Disruptive Game Design:
A Commercial Design and Development Methodology for Supporting Player Cognitive Engagement in Digital Games

By Peter Howell

December 2015

This thesis is submitted in partial fulfilment of the requirements of the award of the degree of Doctor of Philosophy of the University of Portsmouth
Abstract

First-person games often support the player’s gradual accretion of knowledge of the game’s rules during gameplay. They thus focus on challenging and developing performative skills, which in turn supports the player in attaining feelings of achievement and skills mastery. However, an alternative disruptive game design approach is proposed as an approach that encourages players to engage in higher-order thinking, in addition to performative challenges. This requires players to cognitively engage with the game at a deeper level. This stems from the player’s expectations of game rules and behaviours being disrupted, rather than supported, requiring players to learn and re-learn the game rules as they play. This disruptive approach to design aims to support players in satiating their needs for not only achievement and mastery at a performative level but also, their needs for problem-solving and creativity.

Utilising a Research through Design methodology, a model of game space proposes different stages of a game’s creation, from conceptualisation through to the final player experience. The Ludic Action Model (LAM), developed from existing game studies and cognitive psychological theory, affords an understanding of how the player forms expectations in the game as played. A conceptual framework of game components is then constructed and mapped to the Ludic Action Model, providing a basis for understanding how different components of a game interact with and influence the player’s cognitive and motor processes. The Ludic Action Model and the conceptual framework of game components are used to construct the Disruptive Game Feature Design and Development (DisDev) model, created as a design tool for ‘disruptive’ games. The disruptive game design approach is then applied to the design, development, and publication of a commercial game, Amnesia: A Machine for Pigs (The Chinese Room, 2013). This application demonstrated the suitability of the design approach, and the proposed models, for establishing disruptive game features in the game as designed, developing those features in the game as created, to the final resolution in the game as published, which the player will then experience in the game as played.

A phenomenological template analysis of online player discussions of the game shows that players tend to evaluate their personal game as played (i.e. their personal play experience) in relation to their a priori game as expected (i.e. the experience that they expected the game to provide). Players reported their play experiences in ways that suggested they had experienced cognitive engagement and higher-order thinking. However, player attitudes towards this type of play experience were highly polarised and seemingly dependent on the correspondence between actual and expected play experiences. The discussion also showed that different methods of disruption have a variable effect on the player experience depending on the primacy of the game feature being disrupted. Primary features are more effectively disrupted when the game’s responses to established player actions are subsequently altered. Secondary game features, only present in some sections, are most effectively disrupted when their initially contextualised behaviour is subsequently altered, or recontextualised. In addition, story-based feature disruption is most effected when the initial encoding stage is ambiguous, thus disrupting players’ attempts to form an initial understanding of them. However, these different methods of disruption may be most effective when used in conjunction with each other.
Risk-aversion and 'conservatism' in contemporary game design, adapted from Lepetit and Bridgman (2014).
## Contents

**CHAPTER 1: APPROACHES TO GAME DESIGN**

1.1: Contemporary Game Design Discourse 1

1.2: Design Intent and the Structure of Game Space 3
   1.2.1: Game Design Space and Game Development Space 4
   1.2.2: Game Publication Space and Influences on the Player from Factors in World Space 8
   1.2.3: Game-Player Space and the Player 10

1.3: Disruptive Game Design 14
   1.3.1: Defining Cognitive Engagement 17
   1.3.2: Properties of Disruptive Game Design 19

1.4: Research Question 20

1.5: Contributions to Knowledge 22

1.6: Structure of Thesis 25

**CHAPTER 2: A METHODOLOGY FOR IMPLEMENTING AND EVALUATING DISRUPTIVE GAME DESIGN IN A COMMERCIAL GAME PROJECT** 27

2.1: Intentionality and Ecological Validity in Methods of Games Research 27

2.2: Defining Action Research and Design Research 31
   2.2.1: Canonical Action Research (CAR) 32
   2.2.2: Design Research (DR) 33

2.3: Modes of Design Research 35
   2.3.1: Research through Design’s (RtD) Output in the Context of Game Design and Development 36
   2.3.2: Research through Design (RtD) with Commercial Constraints 37

2.4: An Iterative, Stepwise Method for a Research through Design (RtD) Project 38
   2.4.1: ‘Awareness of Problem’ and ‘Suggestion’ 42
   2.4.2: ‘Design’ 42
   2.4.3: ‘Development’ 43
   2.4.4: ‘Publication’ 43
   2.4.5: ‘Evaluation’ 43
   2.4.6: ‘Conclusion’ 44

2.5: Chapter Summary 44

**CHAPTER 3: THE LUDIC COGNITION MODEL OF THE GAME AS PLAYED** 46

3.1: A ‘Systems Model’ of Cognition 48

3.2: Developing the Ludic Cognition Model (LCM) 52
   3.2.1: Working Memory and its Interaction with the Body and World 52
   3.2.2: Encoding Stimulus Information for Long-Term Memory Storage 55
3.2.3: Refining the Definition of the ‘Dynamic Schema’ 58

3.3: Ludic Knowledge Types 61
  3.3.1: Encoding Specificity 61
  3.3.2: Intraludic Knowledge 62
  3.3.3: Transludic Knowledge 62
  3.2.4: Extraludic Knowledge 63
  3.2.5: The Game as Expected and Ludic Knowledge Types 63

3.4: Query and Recall of Knowledge Types in a Gameplay Session 64

3.5: Ludic Knowledge Types within the Ludic Cognition Model 67

3.6: Implications for the Disruptive Game Design Philosophy of the Construction and Recall of Ludic Knowledge Types 70

3.7: Chapter Summary 71

CHAPTER 4: THE LUDIC ACTION MODEL AND THE PLAYER’S INTERACTION WITH THE GAME AS PUBLISHED 73

4.1: The Gameplay Cycle 74
  4.1.1: The Circular Model of Gameplay 75
  4.1.2: The Heuristic Circle of Gameplay 78

4.2: Developing the Ludic Action Model (LAM) 82
  4.2.1: The Cognitive Processing Cycle 83
  4.2.2: Performative Processes and Performed (Re)actions 84
  4.2.3: Stimuli from World Space and the Player’s Effect on World Space 86
  4.2.4: Modification of Game Object Properties 87
  4.2.5: The Ludic Action Model and the Primary Cycle of Current Gameplay 88

4.3: Modes of Disruption, Cognitive Process Targets, and Cognitive Engagement 89
  4.3.1: Encoding Disruption 91
  4.3.2: Recall Disruption 92
  4.3.3: Action Plan Disruption 93

4.4: Modes of Disruption and their Knowledge Type Targets 96

4.5: The Ludic Action Model in Game Space 97

4.6: Chapter Summary 98

CHAPTER 5: A CONCEPTUAL FRAMEWORK FOR THE GAME AS CREATED 99

5.1: The Ludodiegesis 100

5.2: The ‘Elemental Tetrad’ Structure of Game Components 101
  5.2.1: Technology 101
  5.2.2: Aesthetics 103
  5.2.3: Story 104
  5.2.4: Mechanics 106
5.3: Mapping the Conceptual Framework of the Game as Created to the Ludic
Action Model

5.3.1: Mapping ‘Technology’ to the Ludic Action Model
5.3.2: Mapping ‘Aesthetics’ to the Ludic Action Model
5.3.3: Mapping ‘Told Story’ and ‘Constructed Narrative’ to the Ludic Action Model
5.3.4: Mapping ‘Mechanics’ to the Ludic Action Model
5.3.5: Implications of Mapping the Conceptual Framework to the Ludic Action Model for Operationalising the Disruptive Game Design Philosophy

5.4: Disruptive Game Feature Properties

5.5: The Conceptual Framework in Game Space

5.6: Chapter Summary

CHAPTER 6: A DESIGN AND DEVELOPMENT MODEL FOR THE GAME AS DESIGNED AND THE GAME AS CREATED

6.1: The Disruptive Game Feature Design and Development (DisDev) Model

6.2: Examples of Disruptive Game Features Utilising the Disruptive Game Feature Design and Development Model

6.2.1: Encoding Disruption of Audibly Perceived Story Information
6.2.2: Recall Disruption of Haptically Perceived Mechanic Information
6.2.3: Action Plan Disruption of Visually Perceived Mechanic Information

6.3: Constraints on Disruptive Game Design Utilising the Disruptive Game Feature Design and Development Model

6.3.1: Disrupting the ‘Technology’ Property Class of the ‘Game Component’ Property Category
6.3.2: Disruption in the Context of the Individual Game

6.4: Understanding a Project’s Contextualising Factors that may Influence Design Decisions

6.4.1: Context of the Commercial Development Process
6.4.2: Context of the Medium of Games
6.4.3: Context of the Game’s ‘Genre’, ‘Mode’, and ‘Milieu’
6.4.4: Context of Series, Developer, or Publisher History
6.4.5: Updating the Disruptive Game Feature Design and Development Model with Contextualising Factors

6.5: Prototyping as a Bridge between Game Feature Design and Game Feature Development

6.5.1: Prototyping in the Disruptive Game Feature Design and Development (DisDev) Model
6.5.2: Prototyping in the Game Space Model

6.6: Constructing an Integrated Model
CHAPTER 7: DESIGNING, PROTOTYPING, AND DEVELOPING DISRUPTIVE GAME FEATURES IN AMNESIA: A MACHINE FOR PIGS

7.1: Commercial Context of the Design and Development of the Game Artefact

7.1.1: Commercial Situation for the Project’s Initiation
7.1.2: Balance of Creative Control between Development Studio and Publishing Studio
7.1.3: Design Ethos of Development and Publishing Studios
7.1.4: Initial Pitch, Project Funding, and Proposed Development Timescale

7.2: Application of DisDev Model Stage 1: Assessment of Project Context

7.2.1: Implications of the Commercial Development Process
7.2.2: Implications of the Series, Developer, and Publisher History
7.2.3: Implications of the Medium of Games
7.2.4: Implications of the Game’s ‘Genre’, ‘Mode’, and ‘Milieu’
7.2.5: Summary of DisDev Stage 1: Assessment of Project Context for Amnesia: A Machine for Pigs

7.3: Application of DisDev Model Stage 2: Initial Non-Disruptive ‘Framework’ for the Game as Designed

7.3.1: Setting
7.3.2: Characters
7.3.3: Plot

7.4: Application of DisDev Model Stage 2: Disruptive Game Features in the Game as Designed

7.4.1: The ‘Infection’ System
7.4.2: Emulating Peripheral Vision
7.4.3: Procedurally Generated Tesla Field ‘Maze’
7.4.4: Personality-Driven Enemy Artificial Intelligence
7.4.5: ‘Hardcore’ Difficulty Mode
7.4.6: Providing the Player with Hints
7.4.7: Handling the Death of the Player-Character
7.4.8: Hiding Places
7.4.9: Item Inventory Removal
7.4.10: The Player-Character’s Lantern
7.4.11: Manipulation of ‘Set Dressing’ Entities and Props
7.4.12: Ambiguous Stimulus Information and the ‘Pig Mask’ Motif
7.4.13: Enemy Audio Cues
7.4.14: Summary of the Game as Designed for Amnesia: A Machine for Pigs

7.5: Application of DisDev Model Stage 3 and Stage 4: Prototyping and Developing Disruptive Game Features in the Game as Created

7.5.1: Removal of the ‘Infection’ System
7.5.2: Removal of System to Emulate Peripheral Vision
7.5.3: Removal of Procedurally Generated Tesla Field ‘Maze’
7.5.4: Addition of 'Non-Euclidean' Space 200
7.5.5: Removal of Personality-Driven Enemy Artificial Intelligence 208
7.5.6: Addition of Multiple Enemy Types 209
7.5.7: Removal of 'Hardcore' Difficulty Mode 219
7.5.8: Providing the Player with Hints 220
7.5.9: Handling the Death of the Player-Character 225
7.5.10: Hiding Places 227
7.5.11: Item Inventory Removal 229
7.5.12: Addition of 'Flashing Lantern Warning System' to the Player-Character’s Lantern 236
7.5.13: Enemy Audio Cues 242
7.5.14: Manipulation of ‘Set Dressing’ Entities and Props and the Pig Mask Motif 245
7.5.15: Summary of the Prototyping Process and the Game as Created for Amnesia: A Machine for Pigs 248

7.6: Significant Differences between the Game as Created and the Game as Published 251

7.7: Observations from the Application of Design Theory to Design and Development Practice 254
7.7.1: The DisDev Model’s Applicability to Commercial Game Projects 254
7.7.2: Designing for Different Modes of Disruption and Disruption of Different Knowledge Types 255
7.7.3: ‘Interludic’ Knowledge as an Additional Ludic Knowledge Type 256

7.8: Summary of Disruptive Game Features in Amnesia: A Machine for Pigs 257
7.8.1: Summary of Disruptive Game Features in the Game as Designed 257
7.8.2: Summary of Disruptive Game Features in the Game as Created 259

7.9: Chapter Summary 260

CHAPTER 8: PHENOMENOLOGICAL STUDY OF ONLINE PLAYER DISCUSSION OF AMNESIA: A MACHINE FOR PIGS 262
8.1: The Phenomenological Psychological Perspective and the Game as Reported 262
8.2: The Contextual Constructivist Position in Phenomenology 264
8.3: Sourcing Data from Players 265
8.4: Selection of Appropriate Phenomenological Methodology 267
8.4.1: Descriptive Phenomenology 267
8.4.2: Interpretive Phenomenological Analysis, Hermeneutic Phenomenology and Template Analysis 268
8.5: Ethical Considerations 270
8.6: Participant Demographic Data and Sample Size 270
8.7: Data Collection, Organisation, and Analysis Apparatus 271
8.8: Coding the Player Data 272
8.9: Coding Results: Identification of ‘FIT’ Descriptions 275

8.10: Analysis and Discussion of Study Findings 279
8.10.1: Disruptive Game Features with a Correspondence between Intended and Reported Disruptive Impacts 280
8.10.2: Disruptive Game Features without a Correspondence between Intended and Reported Disruptive Impacts 291
8.10.3: Potential Association between Modes of Disruption and Disruptive Game Feature ‘Primacy’ 298
8.10.4: Concept Mapping and Further Themes Identified in the Player Discussion Data 300

8.11: Implications of Findings from the Game as Reported for Disruptive Game Design Theory 308
8.11.1: Implications for the Game Space Model 308
8.11.2: Implications for the Ludic Cognition Model (LCM) 310
8.11.3: Implications for the Ludic Action Model (LAM) 312
8.11.4: Implications for the Disruptive Game Feature Design and Development (DisDev) Model 312
8.11.5: Implications for the Integrated Model 314

8.12: Chapter Summary 317

CHAPTER 9: CONCLUSIONS AND RECOMMENDATIONS FOR FURTHER WORK 318

9.1: Contributions to Knowledge 319
9.1.1: Contributions to the Field of Game Studies and the New Model of Game Space 319
9.1.2: Contributions to Game Design Theory and the New Definition of Disruptive Game Design 320
9.1.3: Contributions to Practice-Based Game Studies and a New Research through Design Methodology for the Field 321
9.1.4: Contributions to the Understanding of Player Cognition during Gameplay and a New Ludic Cognition Model 322
9.1.5: Contributions to the Understanding of Player Interaction with a Game and a New Ludic Action Model 324
9.1.6: Contributions to the Theory of Conceptual Game Structure and a New Conceptual Framework for Games 326
9.1.7: Contributions to the Theory and Practice of Game Design and Development, the New Disruptive Game Feature Design and Development Model 328
9.1.8: Contributions to the Theory and Practice of Game Design and Development and the Integrated Model 330
9.1.9: Contributions to the Theory and Practice of Designing Games with the Disruptive Game Design Philosophy 333
9.1.10: Contributions to the Understanding of how Players Experience and Report on ‘Disruptive’ Games 334
9.1.11: Contributions to the Understanding of how a Game’s Marketing Material Influences the Game as Expected 337
9.1.12: Contributions to the Understanding of Methodological Approaches to Qualitative Player Studies 337

9.2: Limitations of the Research 337

9.3: Suggestions for Further Work 339

9.4: Closing Remarks 345

REFERENCES 347

APPENDIX A: DEVELOPMENT OF THE PSYCHOLOGICAL FOUNDATION FOR THE SYSTEMS MODEL OF COGNITION 371

A.1: Behavioural Psychology and Games 371

A.2: The Cognitive Approach to Memory and Knowledge 372

A.2.1: Cognitive Models of Memory and Recall 374

A.2.2: Structure of Knowledge Storage in Long-Term Memory 382

A.2.3: Cognitive Theories of Knowledge Organisation, Recall, and the Construction of Meaning 388

A.2.4: Knowledge Organisation in Procedural, Semantic and Episodic Memory 401

A.2.5: The Problems of Initial Schema Acquisition and Schema Flexibility 408

APPENDIX B: TRANSFER OF KNOWLEDGE FROM GAME(S) SITUATIONS TO REAL WORLD SITUATIONS 411

APPENDIX C: OVERVIEW OF HPL2 ENGINE FEATURES WITH POTENTIAL APPLICATIONS OR LIMITATIONS FOR DISRUPTIVE GAME FEATURES 412

C.1: Coherent Hierarchical Culling System 412

C.2: Shadow Maps 412

C.3: AngelScript-based Application Programming Interface (API) 412

C.4: Newton Game Dynamics Physics Engine 412

C.5: HPL2 Developer Tools 412

APPENDIX D: WE ARE THE PIG CONCEPT PITCH 414

APPENDIX E: AMNESIA: A MACHINE FOR PIGS DESIGN DOCUMENT 415

APPENDIX F: AMNESIA: A MACHINE FOR PIGS POST-MORTEM REPORT 440

F.1: What Went Right 441

F.1.1: Creative Freedom 441

F.1.2: Development Tools 444

F.1.3: Development Team and Communication 445

F.1.4: Environmental Storytelling 446

F.1.5: Streamlining Gameplay Experience 448

F.2: What Went Wrong 449

F.2.1: Rapid Prototyping and Project Scheduling 449
Figures

Figure 1.1: The interaction of *game design space*, game objects in *game publication space*, and *game-player space* ................................................................. 3

Figure 1.2: The four layers of *game space*, detailing *game design space* and *game development space* .................. 5

Figure 1.3: The four layers of *game space*, detailing *game publication space*, *game-player space* and factors from *world space* that may influence the *game as played* ................................................ 8

Figure 1.4: Maslow’s Hierarchy of Needs, following Schell (2014, p.127) and Maslow (1943) ........................................... 11

Figure 1.5: Player-supportive and counter-supportive design mapped against Maslow’s hierarchy .................. 14

Figure 1.6: Retrieving a star by creating a ladder object in Scribblenauts (Coldewey, 2009) ................................. 15

Figure 1.7: ‘Survival instinct’ in Tomb Raider highlights all puzzle-related entities ....................................... 16

Figure 1.8: Bloom’s Revised Taxonomy (Krathwohl, 2002) ............................................................... 18

Figure 2.1: Structure of Pinchbeck’s (2010) development-led research methodology with additional iteration loop between projects .................................................................................................. 39

Figure 2.2: Design Research Process Model, following Vaishnavi and Kuechler (2004, p.7) ................................. 40

Figure 2.3: Research through Design (RtD) process model for a game design and development project, adapted from Vaishnavi and Kuechler’s (2004, p.7) Design Research process model ....................................... 41

Figure 3.1: Chapter 3 focuses on the player in game-player space ................................................................. 47

Figure 3.2: A ‘Systems Model’ of Cognition .................................................................................................. 49

Figure 3.3: Developing the Ludic Cognition Model from the Systems Model, Stage 1: Working memory and its interaction with the body and world ............................................................. 53

Figure 3.4: Developing the Ludic Cognition Model from the Systems Model, Stage 2: Long-term memory storage and encoding of stimulus information ........................................................................... 56

Figure 3.5: Developing the Ludic Cognition Model from the Systems Model, Stage 3: Refining the definition of the ‘dynamic schema’ .................................................................................................. 59

Figure 3.6: Section of the *game space* model with ludic knowledge types added to the factors that influence the *game as expected* .................................................................................. 63

Figure 3.7: LCM modified to include the ludic knowledge types within the long-term memory stores .......... 67

Figure 3.8: Transfer of knowledge from real world analogous situations, and experiences in other game instances, to a new game instance .................................................................................. 69

Figure 4.1: Chapter 4 focuses on the interaction between the *game as published* and the *game as played*. ........................................................................................................................................... 73

Figure 4.2: Heaton’s (2006) Circular Model of Gameplay (left) and Perron’s (2006) Heuristic Circle of Gameplay (right). ........................................................................................................... 75

Figure 4.3: Heaton’s Circular Model of Gameplay embedded within *game space*; diagram style adapted from Heaton (2006, p.2) ........................................................................................................... 76

Figure 4.4: The perceptual cycle (Neisser, 1976, p.21) .................................................................................. 78

Figure 4.5: The components of Perron’s Heuristic Circle of Gameplay embedded within *game space*; diagram style adapted from Perron (2006, p.66) ........................................................................... 79
Figure 4.6: *FTL: Faster Than Light* (Subset Games, 2012) allows players to analyse, evaluate, and give input while the game state is paused. ................................................................. 81

Figure 4.7: Developing the Ludic Action Model from the Heuristic Circle of Gameplay, Stage 1: The Cognitive Processing Cycle. .................................................................................. 83

Figure 4.8: Developing the Ludic Action Model from the Heuristic Circle of Gameplay, Stage 2: Performative Processes and Performed (Re)actions. ................................................................. 84

Figure 4.9: *Super Hang-On* (Sega AM2, 1987) arcade machine flyer (The Arcade Flyer Archive, 2009). .... 85

Figure 4.10: Developing the Ludic Action Model from the Heuristic Circle of Gameplay, Stage 3: Stimuli from *world space* and the player’s effect on *world space*. ........................................ 86

Figure 4.11: Developing the Ludic Action Model from the Heuristic Circle of Gameplay, Stage 4: Modification of *game object* properties ................................................................. 87

Figure 4.12: The complete Ludic Action Model................................................................. 88

Figure 4.13: Specific cognitive processes within the player that may be able to be disrupted. .......... 90

Figure 4.14: Impact of Encoding Disruption on cognitive engagement. ................................. 92

Figure 4.15: Impact of Recall Disruption on cognitive engagement. ........................................ 93

Figure 4.16: Impact of Action Plan Disruption on cognitive engagement............................... 95

Figure 4.17: The Ludic Action Model incorporates a number of different levels and components of *game space*. ........................................................................................................... 97

Figure 5.1: Chapter 5 focuses on the conceptual framework of the *game as created* in *game development space*. ........................................................................................................... 99

Figure 5.2: The ludodiegesis, containing the homodiegesis and heterodiegesis. ......................... 100

Figure 5.3: The ‘elemental tetrad’ structure of a game (Schell, 2014, p.51). ................................. 101

Figure 5.4: Lunar: Silver Star Story Complete (Game Arts & Japan Art Media, 1999) collector’s edition, with cloth map and hardback manual. ................................................................. 102

Figure 5.5: The game’s 'Technology' in relation to the ludodiegesis, homodiegesis and heterodiegesis. ................................................................. 102

Figure 5.6: ‘Scratch’n’Sniff’ cards that shipped with copies of *EarthBound* (Ape & HAL Laboratory, 1995). Image sourced from Starmen.net (n.d). ................................................................. 103

Figure 5.7: Adding the game’s 'Aesthetics' in relation to the ludodiegesis, homodiegesis and heterodiegesis. .......................................................................................................................... 104

Figure 5.8: Adding the game’s 'Told Story' and 'Constructed Narrative' in relation to the ludodiegesis, homodiegesis and heterodiegesis................................................................. 106

Figure 5.9: Removing enemy agents from the game world and simplifying the game state in *Bioshock Infinite* (Irrational Games, 2013). ................................................................. 107

Figure 5.10: Choosing between two brooch designs for Elizabeth to wear in *Bioshock Infinite*. ........ 108

Figure 5.11: Adding the game’s ‘Mechanics’ in relation to the ludodiegesis, homodiegesis and heterodiegesis, forming a refined conceptual framework of the *game as designed*. ................................................................. 109

Figure 5.12: Psycho Mantis using game save data from the memory card in *Metal Gear Solid* (Konami Computer Entertainment Japan, 1998) to identify a player’s other gaming preferences. .......... 110

Figure 5.13: Mapping the conceptual framework for the *game as created* onto the Ludic Action Model. .......................................................................................................................... 111
Figure 5.14: Steel Battalion’s (Capcom Production Studio 4 and Nude Maker, 2002) bespoke input device that shipped with the game. 

Figure 5.15: Lost Odyssey’s (Mistwalker and feelplus, 2008) menu screen contains small text that is difficult to read on standard-definition displays.

Figure 5.16: Visual representation of disruptive game feature property categories and classes.

Figure 5.17: The conceptual framework provides a means of understanding the structure of the game as created in game space.

Figure 6.1: Chapter 6 focuses on the game as designed within game design space and the development process that leads to the game as created in game development space.

Figure 6.2: The Disruptive Game Feature Design and Development (DisDev) model, demonstrating the design process for disruptive and non-disruptive game features within the game as designed.

Figure 6.3: DisDev model for Encoding Disruption of audibly perceived Story information.

Figure 6.4: DisDev model for Recall Disruption of haptically perceived Mechanic information.

Figure 6.5: Wall-mounted flame trap in The Elder Scrolls V: Skyrim (Bethesda Game Studios, 2011).

Figure 6.6: DisDev Model for Action Plan Disruption of visually perceived Mechanic information.

Figure 6.7: Exploding barrels in DOOM (Id Software, 1993) (left) and Half-Life 2 (Valve Corporation, 2004) (right).

Figure 6.8: The game space model, with contextualising factors and their sources summarised alongside.

Figure 6.9: Mapping 'mode' to the structure of ludodiegesis.

Figure 6.10: Mapping 'milieu' to the structure of ludodiegesis.

Figure 6.11: Mapping 'genre' to the structure of ludodiegesis.

Figure 6.12: The Disruptive Game Feature Design and Development (DisDev) model, with Stage 1 (Assessment of Project Context) and Stage 2 (Game Feature Design) of the process specified.

Figure 6.13: Fullerton’s (2014, p.414) stages of development (left) and Adams and Rollings’ (2007, p.53) stages of the design process (right).

Figure 6.14: The Disruptive Game Feature Design and Development (DisDev) model, with Stage 3 (Game Feature Prototyping) and Stage 4 (Full Development) of the process specified.

Figure 6.15: The model of game space modified to include the prototyping stage of the design and development process.

Figure 6.16: The Integrated Model (Part1): design philosophy through to release candidate of game object.

Figure 6.17: The Integrated Model (Part 2): ludodiegetic structure of the game as created, which becomes the game as published via the publication process and is then experienced by the player as the game as played.

Figure 7.1: Research through Design (RtD) process model for a game design and development project, adapted from Vaishnavi and Kuechler’s (2004, p.7) Design Research process model.

Figure 7.2: Adaptation of the RtD process model, adding an awareness of commercial context requirement and defining the focus of each stage of the process.

Figure 7.3: Amnesia: The Dark Descent (Frictional Games, 2010).

Figure 7.4: DisDev model Stage 1: Assessment of Project Context.
Figure 7.5: Stage 1 of the Disruptive Game Feature Design and Development (DisDev) model applied to *Amnesia: A Machine for Pigs* ................................................................. 161

Figure 7.6: DisDev model Stage 2: Game Feature Design ................................................................. 162

Figure 7.7: Victorian London skyline concept artwork. © Ben Andrews, used with permission ........ 163

Figure 7.8: Reference used in development as the basis for the character of Oswald Mandus. Image originally obtained from Moda Historica (http://modahistorica.blogspot.co.uk/) .................................................. 163

Figure 7.9: Representation of Huitzilopochtli, the Aztec Sun God. Image Licensed under Public Domain via Wikimedia Commons ..................................................................................................................... 164

Figure 7.10: In-game screenshot of Edwin and Enoch, following the revelation of Mandus’ actions in *Mexico* .................................................................................................................. 165

Figure 7.11: Early concept artwork of the Manpig .............................................................................. 166

Figure 7.12: St. Dunstan’s Church Altar concept artwork. © Ben Andrews, used with permission ...... 167

Figure 7.13: Tesla Field concept artwork. © Ben Andrews, used with permission .............................. 168

Figure 7.14: Aztec Temple concept artwork. © Ben Andrews, used with permission ........................ 169

Figure 7.15: The *game space* model ............................................................................................... 170

Figure 7.16: Insanity system effects in *A:TDD* when looking at an enemy agent. ......................... 171

Figure 7.17: Receiving damage in *Battlefield: Bad Company 2* ....................................................... 171

Figure 7.18: DisDev Stage 2 elements forming the design for the ‘Infection’ System ...................... 172

Figure 7.19: DisDev Stage 2 elements forming the design for the emulation of peripheral vision ..... 173

Figure 7.20: The ‘Lost Woods’ consists of a room-based, door selection puzzle with a single correct solution ......................................................................................................................... 174

Figure 7.21: DisDev Stage 2 elements forming the design for the procedurally generated Tesla Field ‘maze’ ...................................................................................................................... 175

Figure 7.22: DisDev Stage 2 elements forming the design for the personality-driven enemy artificial intelligence ..................................................................................................................... 177

Figure 7.23: DisDev Stage 2 elements forming the design for the ‘hardcore’ difficulty mode ............... 178

Figure 7.24: An early memento automatically provided to the player in *A:TDD* .............................. 179

Figure 7.25: Interacting with the organic tissue provides players with a further piece of explicit information ............................................................................................................................................. 179

Figure 7.26: DisDev Stage 2 elements forming the design for the method for providing the player with hints ............................................................................................................................................. 181

Figure 7.27: Example level layout demonstrating a Pig Nest with two return routes to the main game level with one-way access into both ............................................................................................................. 183

Figure 7.28: DisDev Stage 2 elements forming the design for the method for handling the death of the player-character ............................................................................................................. 183

Figure 7.29: Players in *A:TDD* can use cupboards such as this to hide from pursuing enemies ........ 184

Figure 7.30: DisDev Stage 2 elements forming the design for hiding places ..................................... 185

Figure 7.31: *A:TDD*’s main inventory screen, showing resources such as health, sanity, lantern oil level, and collected items ................................................................................................................ 186

Figure 7.32: DisDev Stage 2 elements forming the design for the removal of the item inventory ........ 187

Figure 7.33: Typical environment in *A:TDD* illuminated with the omnidirectional lantern light ........ 188
Figure 7.34: DisDev Stage 2 elements forming the design for the player-character’s lantern. 189
Figure 7.35: DisDev Stage 2 elements forming the design for the manipulation of ‘set dressing’ entities and props. 191
Figure 7.36: Concept artwork for the porcelain pig mask. 191
Figure 7.37: DisDev Stage 2 elements forming the design for the ‘pig mask’ motif’s behaviour. 192
Figure 7.38: DisDev Stage 2 elements forming the design for the enemy audio cues. 194
Figure 7.39: Stage 2 of the Disruptive Game Feature Design and Development (DisDev) model applied to Amnesia: A Machine for Pigs. 195
Figure 7.40: The game space model. 196
Figure 7.41: DisDev model Stage 3: Game Feature Prototyping and Stage 4: Full Development. 197
Figure 7.42: Prototype of Infection System demonstrating the four stages of infection and their impact on the player’s vision (A:AMFP March 2012 Build). 198
Figure 7.43: Example of Antichamber’s (Bruce, 2013) non-Euclidean environments. 201
Figure 7.44: DisDev Stage 2 elements forming the design for the use of non-Euclidean space. 202
Figure 7.45: Front view of ‘portal’ object showing different environment on the other side. 203
Figure 7.46: Rear view of ‘portal’ object demonstrating non-Euclidean nature of environment as the player is able to walk around the back of the ‘portal’. 203
Figure 7.47: Prototype version of A:AMFP’s seventh level, Tunnels (A:AMFP March 2012 Build). 204
Figure 7.48: Final version of the seventh level, Tunnels (A:AMFP September 2013 Release Build). 206
Figure 7.49: Section of Tunnels level demonstrating the use of ‘impossible architecture’ and its setup in the HPL2 level editor (A:AMFP May 2013 Build). 206
Figure 7.50: DisDev Stage 2 elements forming the design for the use of multiple enemy types. 211
Figure 7.51: A Wretch approaches, but does not immediately attack the player-character (A:AMFP September 2013 Release Build). 211
Figure 7.52: The game’s fifth level, Church, is designed to force enemy-to-player interaction to occur in confined spaces (A:AMFP September 2013 Release Build). 213
Figure 7.53: An Engineer charges at the player-character (A:AMFP September 2013 Release Build). 214
Figure 7.54: The Tesla Pig attacks the player-character in Tesla02 (A:AMFP September 2013 Release Build). 217
Figure 7.55: The Tesla Pig encounter area in Tesla01 (A:AMFP September 2013 Release Build). 219
Figure 7.56: A broken fuse emitting smoke and sparks in the Cellar level (A:AMFP June 2012 Build). 220
Figure 7.57: The direction of the pipeline between the machine and the off-screen steam-activated door (red arrows), with steam jets (red circles) (A:AMFP July 2012 Build). 221
Figure 7.58: A:AMFP’s game options menu, with ‘Show hints’ defaulted to inactive (AAMFP September 2012 Build). 223
Figure 7.59: A usability hint in A:AMFP explaining how to access the hints in the journal (September 2013 Release Build). 224
Figure 7.60: Two example ‘hints’ written in A:AMFP’s ‘diary entry’ style (A:AMFP September 2013 Release Build). 225
Figure 7.61: Hiding Place System, Prototype 1; the two cages shown as white wireframes each have different ‘hiding properties’ (A:AMFP July 2012 Build). 227
Figure 7.62: The Quick Bar in Neverwinter Nights (BioWare, 2002), allows players to use abilities and items with a single button press (in this case, the twelve function keys F1 to F12). ........................................232
Figure 7.63: A concept mock-up of the ‘quick bar’-style inventory system (A:AMFP April 2012 Build). ...232
Figure 7.64: Journal screen for players to read written pages located during gameplay (A:AMFP March 2013 Build) ........................................................................................................................................234
Figure 7.65: Miasmata’s (IonFX, 2012) diary system keeps the player in active gameplay. ..................235
Figure 7.66: System Shock 2’s inventory overlays the game screen and does not pause game execution while it is open........................................................................................................................................235
Figure 7.67: DisDev Stage 2 elements forming the design for the ‘Flashlight Lantern Warning System’. ...237
Figure 7.68: Semi-randomised ‘set dressing’ manipulation scenario in A:AMFP’s Mansion01 level (A:AMFP September 2013 Release Build, screenshots taken in development tool). .............................................................246
Figure 7.69: The pig mask motif in two locations in A:AMFP’s first level, Mansion01; in a storage area (top) and attached to the face of a bust (bottom) (A:AMFP September 2013 Release Build);........247
Figure 7.70: All four stages of the Disruptive Game Feature Design and Development (DisDev) model applied to Amnesia: A Machine for Pigs (Part 1). .................................................................248
Figure 7.71: All four stages of the Disruptive Game Feature Design and Development (DisDev) model applied to Amnesia: A Machine for Pigs (Part 2). ........................................................................249
Figure 7.72: The game space model (identical to Figure 6.15). ................................................................251
Figure 7.73: The colour grading in the final build of the game as created (top) and the modified colour grading with ‘blue fog’ in the game as published (bottom). ......................................................253
Figure 8.1: The relationship between noema and noesis, adapted from Ihde (1986, p.44) and Langdridge (2007, p.15) ........................................................................................................................................263
Figure 8.2: The difference between ‘lived experience’ and reflection on prior ‘lived experience’..........264
Figure 8.3: Portrait ‘hallucination’/insanity effect in A:TDD .................................................................281
Figure 8.4: Section of one theme map focusing on the positive (green) discussion topics and negative (red) discussion topics grouped under the broad theme, Working to Find Meaning(s) in the Game........300
Figure 8.5: Interludic Knowledge as a subset of Transludic Knowledge. ..............................................305
Figure 8.6: The game space model, with the game as reported included.............................................309
Figure 8.7: The Ludic Cognition Model, with interludic knowledge included as a subset of transludic knowledge ........................................................................................................................................311
Figure 8.8: The Ludic Action Model (LAM), with interludic knowledge included within long-term memory. ........................................................................................................................................312
Figure 8.9: Disruptive Game Feature Design and Development (DisDev) model, accounting for interludic knowledge and game feature/mechanic primacy. .........................................................313
Figure 8.10: Comparison of changes made to property division ordering in Stage 2 of the DisDev model. ..................................................................................................................................................314
Figure 8.11: Integrated Model, Part 1..................................................................................................315
Figure 8.12: Integrated Model, Part 2..................................................................................................316
Figure 9.1: The game space model. ..................................................................................................319
Figure 9.2: Research through Design (RtD) process model for game studies. .................................321
Figure 9.3: The Ludic Cognition Model (LCM). .................................................................................323
Figure 9.4: The Ludic Action Model (LAM). ........................................................................................................ 325
Figure 9.5: New conceptual framework of game components within the ludodiegetic structure .......... 327
Figure 9.6: Conceptual framework of game components with additional mapping of 'genre', 'mode', and 'milieu'. ........................................................................................................................................... 327
Figure 9.7: Disruptive Game Feature Design and Development (DisDev) model........................................ 329
Figure 9.8: The Integrated Model, Part 1 ........................................................................................................ 331
Figure 9.9: The Integrated Model, Part 2 ........................................................................................................ 332
Figure A.1: Basic model of memory, following Waugh and Norman (1965, p.93). ........................................ 375
Figure A.2: Atkinson and Shiffrin's (1968) Multi-Store Model of Memory .................................................. 376
Figure A.3: Model of working memory, following Baddeley (2002, p.86). ....................................................... 378
Figure A.4: Working Memory model showing episodic buffer and links to long-term memory stores, following Baddeley (2000, p.5) ........................................................................................................ 379
Figure A.5: 'Systems' model of memory and recall for the disruptive game design philosophy ............... 382
Figure A.6: Schematic diagram of memory type and consciousness relationships, following Tulving (1985b, p.3). ............................................................................................................................................... 384
Figure A.7: Adapted Systems Model of memory and recall incorporating separate long-term memory stores .......................................................... 386
Figure A.8: Further adaptation of Systems Model of memory and recall, refining working memory into parallel declarative working memory and procedural working memory stores.............................................. 387
Figure A.9: An example of a hierarchical network model of knowledge organisation according to Collins and Quillian's (1969) model, following Malim and Birch (1998, p.298). ...................................................... 389
Figure A.10: Hypothetical concept relatedness links in a spreading activation model, adapted from Collins and Loftus (1975, p.412) ................................................................. 391
Figure A.11: The 'Give' schema, following Rumelhart and Ortony (1977, p.102) ............................................. 397
Figure A.12: A simple procedural schema for making Mario enter a pipe and the resulting in-game events ........................................................................................................................................... 398
Figure A.13: Adapted procedural schema for making Mario enter a pipe whilst avoiding an enemy .......... 402
Figure A.14: Hierarchical, embedded schema structure in semantic memory .................................................. 403
Figure A.15: Example semantic schema structure for the 'Firearm' concept ................................................... 404
Figure A.16: Further adapted Systems Model of memory and recall adding representation of organisational structure to memory types .................................................................................... 407
Figure A.17: Further adapted model of memory and recall adding dynamic schema instance generation via induction as a process linked to the Central Executive of working memory .................................................. 408
Figure H.18.1: Tracking text coding in NVivo via the coding stripes tool ..................................................... 464
Tables

Table 2.1: Criteria for describing Canonical Action Research (CAR), following Cole (2005, p.328) .......... 32
Table 2.2: Criteria for describing Design Research (DR), following Cole (2005, p.327) ............... 34
Table 2.3: Summary of Frayling’s (1993, p.5) three modes of design research.......................... 35
Table 3.1: Summary of ludic knowledge types with shorthand identifiers.................................. 65
Table 3.2: Ludic knowledge types in relation to operating a hand gun in a first-person-shooter game. .... 66
Table 4.1: Summary of possible knowledge type targets of each mode of disruption.................. 96
Table 5.1: Disruptive game feature properties................................................................................. 116
Table 7.1: The Chinese Room’s development team structure for A:AMFP and associated high-level skills audit. ................................................................. 155
Table 7.2: Combining genres with themes to create definable sub-genres...................................... 158
Table 7.3: Overview of proposed enemy agent personality types and key behavioural traits (Pre-Development, December 2011). ............................................................ 176
Table 7.4: Summary of all disruptive game features present in A:AMFP’s game as designed .......... 258
Table 7.5: Summary of all disruptive game features present in A:AMFP’s game as created .......... 260
Table 8.1: Coding template (Version 3). ...................................................................................... 274
Table 8.2: Colour coding scheme for list of disruptive game features as reported on by players........ 275
Table 8.3: Summary of FIT descriptions for each disruptive game feature in A:AMFP, their intended disruptive impacts, and their player-reported disruptive impacts.............................. 278
Table 8.4: Approximation of associations between modes of disruption and disruptive game mechanic/feature primacy, with example ‘successful’ disruptive game features from A:AMFP.... 299
Table 9.1: Suggestions for further research................................................................................... 345
Table A.1: Characteristics of long-term memory types, following Glassman and Hadad (2004, p.161). .... 386
Table H.18.1: F-Codes in the initial coding template (Version 1). .................................................... 459
Table H.18.2: I-Codes in the initial coding template (Version 1). .................................................... 460
Table H.18.3: T-Codes in the initial coding template (Version 1). .................................................... 461
Table H.18.4: Initial coding template (Version 1). ......................................................................... 463
Table H.18.5: I-Codes in the refined coding template (Version 2). New codes in this version are highlighted in bold and underlined.............................................................. 465
Table H.18.6: N-Codes in the refined coding template (Version 2). ................................................. 466
Table H.18.7: C-Codes in the refined coding template (Version 2). ................................................. 467
Table H.18.8: Refined coding template (Version 2). ..................................................................... 468
Table H.18.9: T-Codes in Version 3 of the coding template............................................................ 469
Table H.18.10: N-Codes in Version 3 of the coding template......................................................... 470
Table H.18.11: I-Codes in Version 3 of the coding template......................................................... 470
Table H.18.12: O-Codes in Version 3 of the coding template........................................................ 471
Table H.18.13: Refined coding template (Version 3). .................................................................... 473
Acknowledgements

Thanks firstly to Dr. Dan Pinchbeck and Dr. Steve Hand for having the confidence in me to support the initial research proposal and for making this research possible. Additionally, thanks must go to Dan for sharing his wealth of games research and design knowledge along with his sheer enthusiasm for the field, which is unrelentingly infectious.

Thanks to Dr. Brett Stevens for always being on hand with a critical eye and a pack of red pens to provide a steady stream of questions, comments, feedback, and all-important sanity checks.

Thanks to Dr. Mark Eyles for providing a patient sounding board, particularly for my early morning, pre-coffee thoughts and ramblings, and for providing a multitude of in-depth discussions that helped develop those ramblings into something more concrete.

To the rest of the Creative Technologies staff at the University of Portsmouth that are simply too numerous to thank individually, thank you for creating such a supportive and inspiring place to work.

To the players and reviewers of Amnesia: A Machine for Pigs and specifically, to the users of the official discussion boards, thank you for playing, for having opinions, and for not hesitating to voice those opinions. This research would have been impossible without you.

Thanks go to my family and friends who have all been unwavering in their support, despite being on the receiving end of many rants. Your support has kept me sane. Although, I do look forward to family gatherings no longer being punctuated with “haven’t you finished that PhD yet?”

To my wonderful wife Katie, you have my eternal gratitude and admiration for putting up with me during this entire process and for providing constant encouragement, belief, and cups of tea. I promise you can have your husband back now.

Lastly, thank you to my little boy, Oscar. While I’ve been teaching you all about life, you’ve been teaching me what life is all about.
Declaration

Whilst registered as a candidate for the above degree, I have not been registered for any other research award. The results and conclusions embodied in this thesis are the work of the named candidate and have not been submitted for any other academic award.

Word count (excluding appendices and references): 80,000
In chronological order, the following publications, presentations and products have stemmed from the research and development work contained within this thesis:

2011:


2013:


2014:


Howell, P. (2014). Presentation – *A Model of ‘Game Space’ and the Development of Disruptive Game Design.* Presented at the University of Portsmouth Faculty of Creative and Cultural Industries Post-Graduate Research Conference, Portsmouth, UK.
The best games are the ones that sandwich a kernel of something familiar yet ever-so-slightly off between all the layers of howling insanity. Like a painting that won’t hang properly no matter how many times you tilt it. (Grayson, 2013)

Amnesia: A Machine for Pigs is a game of rare patience and intelligence. It’s a horror game that is designed to tweak your understanding of how video games are “supposed to work”. Amnesia: A Machine for Pigs knows what you know about video games and uses that knowledge against you whenever possible. (Groen, 2013)
Chapter 1:
Approaches to Game Design

1.1: Contemporary Game Design Discourse

The dominant discourse within contemporary computer game design, according to Wilson and Sicart (2010, p.41), is player-centric, monologic design aimed at ensuring the needs and desires of players are met and the game's accessibility to players is prioritised.

Monologic play can be thought of as a conversation in which only the player speaks, while the designer merely nods along – hardly a conversation at all. In the monologue of player narcissism, the player (the customer) is always right. Design becomes a rote catering to a user, devoid of any possibility of nurturing an open dialogue between creator and user. Players become mere customers, and designers become mere providers. (Wilson and Sicart, 2010, p.41)

Wilson and Sicart define monologic design as representing an inherent conservatism within game design. Indeed, this ‘conservative view’ of design in computer games has been identified by both academics and games industry professionals for a number of years (Herz, 1998, 2013).

In 1998, Costikyan (1998) argued that this conservatism stemmed from risk-averse publishers more readily providing financial backing to ‘safer’, ‘proven’ game concepts than to more unique concepts carrying greater financial risk. Costikyan also noted however that there was possibly an inherent lack of highly creative concepts being pitched to publishers, suggesting that this conservatism may exist in terms of the “constrained imaginations” (1998, para.8) of designers. Such imaginative constraint may itself have stemmed from, and been perpetuated by, the knowledge that publishers were more likely to invest in certain types of game than others. Herz (1998) similarly argued that at that particular point in the games industry’s growth, the conservatism “rut” stemmed from the commercial, financial drivers of publishers as well as from technological advances. She argued that too much focus was placed on technological innovation while innovation in the field of design suffered. Herz similarly qualified this statement as Costikyan did, stating that the “industry’s long-term survival” required not only publishers but designers to look to other sources of inspiration, move away from the body of existing ideas and be more imaginative in the ways that they conceived of design concepts.

These issues may have been raised in 1998, however a republication of Herz's article in 2013 (Herz, 2013) identifies the continuing relevance of them and again, emphasises a perceived division between innovation in technology and innovation in design creativity. It should be noted however that these discussions of ‘conservatism’ are in terms of ‘the games industry’ as a whole. In the context of the current games industry structure, such a broadly applied discussion is less appropriate, as there will be differences in ‘innovation’ and ‘creativity’ between small developers and large developers, or between projects with different development budgets, for example.
Wilson and Sicart (2010) present an alternative design approach in *abusive design*, which they frame as a direct reaction to the perceived ‘conservative’ approach. This eschews what they suggest is the conservative, ‘acceptable’, idea of challenging the player only “within the limits of what an implied player model suggests, always maintaining a desired and “positive” experience of the game” (Wilson and Sicart, 2010, p.41). Their approach aims to set up a ‘dialogic’ scenario, as opposed to monologic, in which the designer is no longer in the background creating fixed systems for the express purpose of satisfying the player. Instead, they are directly antagonising and provoking the player through the mediator of the game. Thus, the player achieves a sense of playing ‘with’ or ‘against’ the designer rather than only playing the game.

Abusive design however may be seen as an extreme reaction to the perceived ‘conservative’ approach. Varying degrees and types of abuse are described, such as physical, social and aesthetic abuse (e.g. games designed to cause physical or emotional pain or discomfort), through to unfair game design (e.g. games that are designed to cheat, or favour computer-controlled players/avatars) and lying to the player (e.g. games that purposely provide the player with incorrect information about game functionality). Of the examples provided by Wilson and Sicart (2010) however, only a small selection may be deployable in, or are drawn from, a large-scale commercial game project. The majority are drawn from niche titles such as Desert Bus from the unreleased *Penn and Teller’s Smoke and Mirrors* (Imagineering, Unreleased), smaller independently developed titles such as *Flywrench* (Essen, 2007) and *I Wanna Be The Guy* (O’Reilly, 2007), ‘installation’ or ‘exhibition’ games such as *PainStation* (Morawe and Reiff, 2001), through to a non-digital live-action role-playing (LARP) example in *Fat Man Down* (Østergaard, 2009).

The “modalities of abuse” (Wilson and Sicart, 2010, p.3) that are present in these games vary. Physical abuse via electric shocks is present in *PainStation*, with ‘punishment’ being administered to players each time they lose a point during an otherwise ‘conservative’ game that replicates *Pong* (Atari, 1972). Physical abuse is present in *Desert Bus* in a less overt manner as the game consists of driving a bus from Tucson to Las Vegas without leaving the road and in accurate real-time, making the journey a full eight hours in length. This is coupled with damaged steering, meaning that the bus constantly veers towards the right, making any attempt to cheat by keeping the accelerator held down, and walking away to do something else, impossible. In *Flywrench*, aesthetic abuse is present, with a soundtrack that is discordant and grating. As Wilson and Sicart (Wilson and Sicart, 2010, p.4) state, *Flywrench* is “already a very difficult game on its own, [yet further] dares the player to maintain focus amid a sonic maelstrom”. *I Wanna Be The Guy* represents the modality of “unfair design” (Wilson and Sicart, 2010, p.3) in which a game is designed specifically to be as user-unfriendly as possible and to be unfair in how it implements game mechanics. For example, one sequence in the game presents players with a Windows XP (Microsoft, 2001) error dialog box while seemingly crashing the game, before then turning the dialog box into a lethal in-game object that falls and crushes the player character.
As a reaction to perceived conservatism in the games industry, abusive design offers a unique perspective and a clear departure from the ‘conservative’ idea of supporting the expected positive experience of players that are only challenged within established acceptable limits. It does not however propose suggestions for how such an approach may proactively mitigate conservatism in the industry via application to more ‘mainstream’ games.

The work of Wilson and Sicart (2010) in conjunction with the identification of a ‘conservative’ approach to game design presents two distinct design approaches; conservative design and abusive design. Moreover these two approaches could be conceived as two points on a ‘spectrum’ of design approaches, with conservative design seeking to challenge the player “within acceptable limits” and abusive design seeking to challenge the player outside of those suggested limits. This raises the question of whether there is another approach to design that may exist between conservative and abusive design; one that seeks to challenge the player by pushing at the suggested acceptable limits, without becoming wholly ‘abusive’.

To explore this question, the broader structure of game space must be defined. That is, how a game is situated within ‘the world’ and how it interacts with designers and players that come into contact with it during the design, development and playing processes. In the context of the current research, ‘the world’ is defined as a container for phenomena and objects; the ‘perceivable reality’. This world space is suggested to contain a large number of other ‘spaces’ that provide further contextual meaning to the phenomena and objects within them. Game space is suggested to be one such space, in which games as objects and game-related phenomena exist.

1.2: Design Intent and the Structure of Game Space

In the current research, game space is suggested to consist of at least four components in relation to any complete, published, commercial game. A player must have a published (either by a publisher, or self-published by a developer) game object (i.e. a playable game artefact) to engage with and a game object cannot exist without both designer and developer input.

Figure 1.1: The interaction of game design space, game objects in game publication space, and game-player space.

Thus, it is suggested that game space in this context consists specifically of game design space, game development space, game publication space, and game-player space (Figure 1.1). It should be noted that this game space exists within world space and that, in theory, other ‘game spaces’ may exist.
for other methods of designing, developing and playing a game. For example, a game that is
designed and developed by an individual as a method of learning a skill such as programming but
which is subsequently never in a form that can be played by other players, would not conform to
the structure in Figure 1.1. However, it is this particular structure that will form the basis for the
current research. The interactions between these suggested spaces may be explored by
considering the logical separations between the spaces, the intentions of those operating in game
design space, and the flow of information from designer to player.

Design intention is specifically raised by Wilson and Sicart (2010, p.3) in relation to abusive design.
They suggest that the intentions of a game designer, or team, are vital if one is discussing
approaches to design. If those intentions are not known, one may only speculate about the
intended design of existing games (i.e. game objects in game publication space). For example, a
game may appear to demonstrate evidence of having been designed from an ‘abusive’ perspective,
but one cannot reliably state that this was the intention. It may simply be that it appears abusive
through a combination of ‘failings’. This can be perceived in a similar way to unintentional
emergent gameplay, such as ‘rocket-jumping’ in early first-person-shooters. This technique sees
players using the additional force generated by explosive weapons to jump higher than they
otherwise would be able to. Intentionality applies as an important consideration in any analysis or
application of a particular design approach. The interaction between game design space, game
development space, and game publication space can thus be suggested to consist of an intentional
design and development process, which results in the release of a game object.

The interaction between game design space and game-player space moves through the mediator of
the game object. A player therefore does not respond to design approaches, they respond to
representations of design approaches as embodied in developed game objects. This is not to suggest
that, outside of this established game space, players are not able to also interact with design
approaches, perhaps discussing them in world space. However, within game space, the transaction
between designer and player must move through, and be mediated by, a game object. Thus, if one
is to ask questions about design approaches in relation to the player experience, it is necessary to
form a representation of that approach as a playable game object.

1.2.1: Game Design Space and Game Development Space

Within this definition of game design space, the design approach could also be described as the
design philosophy being applied to a project. It should be noted that it is not the aim of this thesis
to enter into discussion of the wider field of Philosophy of Design, which seeks to answer
fundamental questions about the nature of design itself (Love, 2000). Instead in this context, a
design philosophy is defined simply as the underlying principles that drive the design of a game in a
particular direction so as to fulfil the aim of the designer, or team, for the intended impact of the
game on the player. Indeed, Harteveld et al. (2009) specifically define a ‘design philosophy’ in the
context of game design as the starting point for developing more robust ‘design theories’ and it is
in this same vein that the term should be read in the current research.
Design philosophies may work towards fulfilling different aims; for example, a particular game may be designed with the aim of providing the player with a specific type of experience (e.g. to horrify, to disgust, or perhaps to relax the player). Other designers may, for example, utilise an approach that specifically aims to demonstrate the capabilities of new technology, such as in the design of technical demonstration games (e.g. *Serena* (Senscape, 2014)) or software that is a ‘flagship’ title for a console (e.g. *Killzone 2* (Guerrilla Games, 2009)). The qualities of the player experience are secondary (although they are still a necessary consideration, influencing them is not the primary aim) in this case to the more important technical qualities of the game.

Within the context of conservative and abusive design as described in Section 1.1, the high level concepts of ‘conservatism’ and ‘abusiveness’ may be considered as philosophical approaches. However, these approaches must be operationalised in some way if they are to produce games. The philosophical principles (e.g. “challenging the player within an implied player model” (Wilson and Sicart, 2010, p.41)) must be translated into components of a game product (e.g. game mechanics that can be demonstrated to have been designed specifically to provide such challenge).

It is thus necessary to apply some form of design framework in addition to a philosophical approach. This framework may then be used to design game features intended to operationalise the principles of the design philosophy. This may be the application of a ‘formal’ design framework, for example, utilising the MDA framework (Hunicke, LeBlanc and Zubek, 2004), the DPE (Design, Play, Experience) framework (Winn, 2009) (an extension of the MDA framework for use in the design of educational games), or other ‘formal’ design framework such as the ‘6-11 framework’ (Dillon, 2010, 2011). It is also possible that the ‘framework’ applied is a lack of framework, or at least a lack of ‘formal’ framework, in favour of an ad hoc approach.

![Figure 1.2: The four layers of game space, detailing game design space and game development space.](image-url)
The combined design philosophy and accompanying framework, within game design space, thus informs the game as designed. This can be conceived as the ‘idealised’ design for a game, as viewed through the lens of the design philosophy. This version of the game includes game components and aesthetic choices (i.e. choices regarding the visual and auditory properties of the game) that appear to support the design philosophy to the fullest extent. However, this ‘idealised’ game has not yet been subjected to the reality of implementation; it is at this stage likely to be primarily conceptual, possibly supported by some physical design documentation. Figure 1.2 differentiates between conceptual and physical stages in each section of game space, with white stages (e.g. the game as designed) being either wholly or primarily conceptual, with grey stages (e.g. the game as created) being physical. The conceptual game as designed is thus not necessarily possible within commercial, industrial or technological constraints.

The game as designed can be used as a basis for developing features within a game during the development process within game development space. However, within game development space, a number of additional factors will place constraints on the creation of the features in the game object, such as hardware and software constraints (e.g. console limitations, game engine capabilities) and development team skills, resources and time. The result of the development process, and thus the output of game development space, will be the release candidate of the game object, or the game as created (Figure 1.2).

The game as created is released into game publication space (Section 1.2.2), where it may pass through any necessary publication processes such as localisation and translations for different international markets, or compliance testing necessary in order to release the game onto a particular platform or service. However, the game as created (i.e. the game that was passed from the design and development team) should not be fundamentally altered within game publication space. The publication process may introduce additional components to the game as created (e.g. different language options) but may not modify the game as created in a way that distorts the design and development team’s design aim. There may be exceptions to this, for example, if a game fails to comply with censorship laws (e.g. Carmageddon (Stainless Games, 1997)), then changes to the game as created may be necessary in order to have the game approved for release. However, such changes should be considered an exception rather than the norm.

It is the game object within the container of the game as published that will be played and experienced by players (i.e. the game as played). It is important to note that, even if the publication process does not make changes to the game as created, the game as created will nevertheless likely differ, possibly significantly, from the game as designed (i.e. the ‘idealised’ game). Thus, designers looking to implement a particular design philosophy must apply that philosophy throughout both design and development. That philosophy must interact with the reality of game development as well as informing the game’s design.
In order to understand how a player may experience a game, it is necessary to not only consider the game design space from the designer’s perspective, but to also consider it from the player’s perspective. While the player does not interact with the game design space directly, the output from this space (i.e. the game object) will be interacted with. This game object can be conceived as being ‘unpacked’ (i.e. experienced) by the player in the opposite direction to that in which it was designed. This concept is posited already in the MDA framework (Hunicke, LeBlanc and Zubek, 2004, p.2), which suggests that the designer interacts first with the game mechanics, then with its dynamics, or systems, and lastly with its aesthetics. The player conversely interacts first with the game’s aesthetics, then with the game’s dynamics, and finally with individual game mechanics. However, it is necessary to consider this same differentiation between the perspectives of designer and player at a higher conceptual level than simply the components of the game object itself.

While the designer can directly interact with the design philosophy, the game as designed, and the game as created, the player only has access (within the confines of game space) to the game object within the game as published. The player only interacts with the game as published, not the game as designed (i.e. the ‘idealised’ game). The player also does not have access to the design philosophy that informed the aims of the design. Within game space, the player will only ever interact with a representation of the design philosophy embodied within the game object.

This limited player interaction places limits on the possible questions that may be asked about the nature of design philosophies as experienced by the player. Galle (2007, p.3) raises this issue more broadly as one that is of particular interest in the field of philosophy of design.

What ontological and epistemological assumptions should be made to explain the apparent fact that designers can know or predict the properties of an artefact, which is not there to have properties? (Galle, 2007, p.3)

One cannot reliably identify the impact of a particular design philosophy alone on a player because it is always interpreted through a ‘lens’ which, in the case of game design, is the game object combined with the individual characteristics of a particular player. As Popper (1974, p.33) states,

we all have our design philosophies, whether or not we are aware of this fact, and our philosophies are not worth very much. But the impact of our philosophies upon our actions and our lives is often devastating. This makes it necessary to improve our philosophies by criticism. (Popper, 1974, p.33)

A philosophy potentially unknown and unseen by the player is of little consequence unless that philosophy can be demonstrated to have been intentionally operationalised and embodied within a game object that is then experienced by the player (i.e. the game as played). Only then can criticism of the game as played in turn provide criticism of the design philosophy that created it. Thus, in seeking to provide criticism of a design philosophy, it is only possible to identify and ask questions about the effect on a player’s game as played of a particular game object that has been intentionally designed and developed to provide a representation of a design philosophy. However, even then the
player experience of this representation of a design philosophy within a designed game object is influenced through its situating within game publication space and through a player’s knowledge of factors within world space.

1.2.2: Game Publication Space and Influences on the Player from Factors in World Space

The game space model must be expanded to elaborate game publication space and the factors within world space that a player may have knowledge of that could influence their experience of the game as played.

Figure 1.3: The four layers of game space, detailing game publication space, game-player space and factors from world space that may influence the game as played.
The game object is released into game publication space through a publishing process, either via a publisher or through self-publishing, resulting in the game as published that the player will interact with. However, alongside game publication space in world space, there are a number of influencing factors that may serve to associate additional information with the game object itself before it is played by the player (Figure 1.3). This additional information, when perceived by a player prior to engaging with the game, will influence a player’s game as expected; that is, an anticipated version of the game. This game as expected exists within game-player space (Section 1.2.3), which can be conceived as being within the ‘mind’ of the individual player.

In game publication space, the game as created is turned from a collection of game assets (e.g. code, graphics, and audio) into a saleable product, which is then either packaged and distributed as a physical product, or distributed via one of the many digital distribution channels available. Turning the game into a saleable product will add extra material that may influence the player’s game as expected. This material may be elements of the game product, such as (in the case of a physical product) the casing, cover artwork and instruction manual, or (in the case of a digital product) the digital storefront page for the game or additional digital material such as a downloadable soundtrack. The game product is situated in world space and provides input into game publication space, along with the game as created, to form the game as published.

However, in world space it is not only the materials of the game product that may influence the game as expected. Cues that may cause a player to expect that a game is going to provide a particular gameplay experience may come from sources of knowledge that are, to various degrees, outside of the designer’s and publisher’s control. Lindley and Sennersten (2008) note this in relation to one such source, game genre archetypes, stating that

[. . .] computer game genres, such as role-playing games (RPGs) and first-person shooters (FPSs), imply particular sets of design features supporting expectations that prospective players have about the nature of the player experience that games support, based upon past experiences with other games in the same genres. (Lindley and Sennersten, 2008, p.1)

Genre archetypes may be suggested to fall under the more general categorisation of games culture as a source of information that may influence the player’s game as expected, along with other ‘cultural’ factors. For example, how a game series or franchise is situated in ‘popular’ games culture, information that is intentionally acquired (e.g. ‘user’ or ‘player’ reviews on online shopping sites), or information that is unintentionally acquired (e.g. ‘spoilers’) by engaging with games culture. Other significant sources may include the game’s marketing materials such as trailers, magazine previews, reviews or features, and pre-release game demos.

The game as expected may also be influenced through the individual player’s own prior knowledge and experience (e.g. they played and enjoyed previous games in the series or from the same developer and thus expect to enjoy another game in that context). These prior experiences,
contained within game-player space, are combined with the influence of marketing and games culture to create an individual player’s game as expected.

It is beyond the scope of this thesis to consider all possible sources of information that may influence the game as expected prior to players engaging with the game itself, or indeed not engaging with the game at all because of them. It is suggested though that, at least, the four main categories described (Figure 1.3) can be considered in terms of how much control the design and development team may have over them. Firstly, the materials that form the game product which, in collaboration with a publisher, the design and development team have control over; secondly, marketing, which is likely to also be influenced by the design and development team, and potentially a publisher as well; thirdly, games culture which may be indirectly influenced (e.g. through a developer’s prior release history); lastly, prior experiences, which cannot be influenced directly by the designer and cannot be known by the designer.

This is not to suggest that the game as expected is, in itself, a negative influence on a player’s experience of the game as played. Indeed, it would be almost impossible in a real world scenario for a player to not have some form of expectation about a game before playing it. It is whether these established expectations are subsequently met, or not met, when the player actually interacts with the game as published that is a more important consideration (i.e. what is the game as played?).

1.2.3: Game-Player Space and the Player

The game as played is compared to the player’s individual game as expected both during play and after play. The attitude taken towards the expectations of the player before and during gameplay is what differentiates the approaches of conservative and abusive design. These attitudes can be considered within the context of a framework for understanding the player themselves and in turn, the aims of these two design philosophies can be interpreted.

1.2.3.1: A Framework for Understanding the Player

One of the important roles of the game designer according to Fullerton (2008, p.35) is to be an “advocate for the player”, identifying the desires of players and then designing systems that fulfil those desires. Such an approach places the player at the centre of the design process and frames them as an entity requiring satiation. Adams (2010, p.30) defines this player-centric approach as placing two obligations on the designer. Firstly, that the designer has a “duty to entertain” and that such entertainment is the primary function of a game. Secondly, that the designer has a “duty to empathise” and to “build the game to meet the player’s desires and preferences for entertainment”. It is this approach that Wilson and Sicart’s abusive design is a reaction to. To discuss this player-centricity further, it is necessary to identify what ‘needs’ a game may be able to ‘satisfy’.
Schell (2014) provides such an identification in his work discussing a range of conceptual ‘lenses’ that can be used to view the design of games in different ways. “The Lens of Needs” (2014, p.127) focuses explicitly on the fulfilment of player needs through the playing of a game. Schell uses the Hierarchy of Needs (Figure 1.4) as proposed by Maslow (1943), as the basis for this lens and notes that while the hierarchy may have some exceptions, it can be applied in understanding the majority of player motivations in games.

In his original work Maslow (1943) states that the lower level needs of the hierarchy must be satiated before an individual can focus on satiating those needs that are higher up. In a ‘conservative’ game, the game systems are designed in a manner that aims to support players in satiation of the low level needs. A player-centric game for example does not aim to physiologically damage the player or place their ‘physical’ safety in question, as an abusive game may aim to (e.g. PainStation). Indeed, common gameplay features such as a ‘pause’ function, ‘save game’ functions, or convenient separations between game levels actively assist players in the satiation of low level needs (e.g. being able to pause a game allows the player to attend to their physiological needs, such as eating, whilst being able save game data supports sleeping). Moreover, games that incorporate social aspects of play such as local or online multiplayer game modes may also allow the satiation of aspects of a player’s social, or ‘belonging’, needs.

Thus, the majority of player-centric games, as Schell suggests, “are about achievement and mastery, which places them at level four, self-esteem” (2014, p.127). This is reflected in other research into psychological need satiation through gameplay that suggests that mastery and a sense of competence during play are highly important factors in player enjoyment (Peng et al.,

Figure 1.4: Maslow’s Hierarchy of Needs, following Schell (2014, p.127) and Maslow (1943).
2012, Przybylski, Ryan and Rigby, 2010, Ryan, Rigby and Przybylski, 2006). Even within this context however, the approach of a player-centric design philosophy is to actively support the player’s needs for achievement and skill mastery through the game’s design. The player is thus only challenged, as previously stated, “within the limits of what an implied player model suggests, always maintaining a desired and “positive” experience of the game” (Wilson and Sicart, 2010, p.2).

An example of this type of player support in achievement and skill mastery can be identified in *Prince of Persia* (Ubisoft Montreal, 2008). As VanOrd discusses in his critical review of the game, there is a checkpoint at almost every platform, so aside from possibly having to repeat a few seconds of gameplay, there is absolutely no penalty for plummeting to your doom. You will never see the words “game over”, and you won’t need to save and reload before difficult sequences. Nor will you need to ever puzzle over how to make it from point A to point B: Elika [a key non-player character that assists the player] can fire off a magical homing orb that will show you the precise way of getting to your destination [. . .] these facets make *Prince of Persia* one of the easiest games you’ll play all year. (VanOrd, 2008, para.4)

While VanOrd describes the game simply as “easy”, the source of this ease can be seen in this particular example to stem from the support systems that have been designed into the game. Frequent checkpoints support consistent player achievement, combined with the lack of requirement to reload between ‘deaths’, which minimises ‘downtime’ between those moments of achievement. Cognitive challenge in terms of route planning through an environment is minimised, as the non-player character is able to highlight the correct route for the player to take. This reduces the core of the game’s challenge to repeated completion of performative tasks, such as running, jumping and fighting enemies, and even this challenge is presented with minimised punishment for failure. A sense of achievement and mastery on behalf of the player may be supported by the game’s design but the types of challenge that must be overcome and the nature of the game’s risk-reward balance may prevent these feelings being experienced by players. Moreover, the minimal cognitive challenge presented does not support the satiation of needs at the top level of Maslow’s hierarchy, self-actualisation (Figure 1.4). Hussain (2011, para.5) specifically identifies the lack of “challenge”, “gratification” and “fulfilment” felt after completing many contemporary games, which can be seen to align with the needs noted in the hierarchy of ‘problem solving’ and indeed, ‘fulfilment’ specifically.

As a reaction to a perceived ‘conservative’, player-centric design approach, it might be expected that the ‘abusive’ design approach aims to affect this support structure and introduce greater cognitive demand and higher level need satiation. However, in an ‘abusive’ game, it can be suggested that the modalities of abuse in fact primarily focus on the prevention of need satiation at the lower levels of the hierarchy. For example, physical abuse clearly targets the lowest levels of physiological wellbeing and of physical safety. Aesthetic abuse, depending on how it is presented, may also be physiological in nature (i.e. audio-visual material presented in particular ways may be physically uncomfortable). The modality of lying to the player or of unfair design can be viewed as
primarily targeting the ‘security’ and ‘stability’ components of the hierarchy’s second level in that they break the fundamental ‘trust’ between player and game system to play ‘by the rules’. However, abusive design further targets concepts of security and stability in its proposal for a move away from monologic design (the suggested ‘norm’) towards dialogic design.

Thus, abusive design aims to remove the game-system-based player support and to create game objects “that appeal [to] the player to face and understand the designer” and “to go beyond the object designed” (Wilson and Sicart, 2010, p.6). In doing this not only does abusive design aim to produce games that do not support the needs of players, at the lower levels of Maslow’s hierarchy, but it also potentially causes a significant difference between the game as expected and the game as played.

Depending on the reasons that an individual player has for engaging with a game, this misalignment between the expectation and the reality of play may have a negative impact on the likelihood of players continuing to play the game, or as Schell (2014, p.127) suggests,

> if a player imagines [or expects] that playing your game is going to make them feel better about themselves, or get to know their friends better, and your game doesn’t deliver on these needs, your player will move on to a game that does. (Schell, 2014, p.127)

This statement may be an over-generalisation, however while some players may find pleasure in a game that does not meet their expectations or their needs, perhaps by interpreting it as a pleasant ‘surprise’, other players may be likely to simply stop playing in favour of a game that does meet their expectations and needs. A player’s attitude and their reasons for playing a game, while they cannot be definitively known during a game’s design and development, will have a significant impact on the eventual player response to the game.

Therefore in the context of the needs of the player and the game system as a designed ‘needs-satiating machine’, the perceived ‘conservative’ design approach, or player-centric approach, may be considered as ‘player-supportive’ design. Indeed, Peng et al.’s (2012) research into the requirement for games to satiate needs for competence, autonomy, and relatedness specifically refers to need-satisfaction-supportive game features. Conversely then, ‘abusive’ design may be referred to in this context as ‘counter-supportive’ design in that it actively aims to restrict or prevent the satiation of player needs.

Thus, it is clear why ‘abusive’ design is not generally described in relation to its applicability to what may be considered ‘mainstream’ design and development applications. Modalities of abuse such as physical abuse may require additional hardware or specialised equipment for example. But also notions such as unfair design or lying to players may be difficult to pitch to investors, publishers or marketers.

Thus, with reference to the different layers of game design space and game development space (Figure 1.3), the ‘abusive’ design approach provides a design philosophy and allows the
conceptualisation of the game as designed. However, it becomes harder to consistently apply it within the constraints of realistic commercial game development in order to move from the game as designed to the game as created.

This poses the question, of whether an alternative design philosophy exists that is able to consider the issues that have been identified regarding ‘player-supportive’ design, such as lack of challenge and especially cognitive challenge, but respond to them in a manner that may be more realistically deployable throughout all levels of the game design space and game development space than ‘abusive’, or ‘counter-supportive’ design appears to be?

1.3: Disruptive Game Design

Figure 1.5: Player-supportive and counter-supportive design mapped against Maslow’s hierarchy.

In order to investigate a potential new design philosophy within game design space, it is necessary to refer back to Maslow’s hierarchy and to now consider it in terms of the ‘space’ left un-catered for by both player-supportive and counter-supportive design. To do this, these two design philosophies are mapped against the structure of Maslow’s hierarchy (Figure 1.5).

There are some important clarifications to make in relation to this mapping. Firstly, this applies to design philosophies and does not apply to individual game objects. As Wilson and Sicart (2010, p.3) state in the case of ‘abusive’ design, it is only the design process that can be labelled as ‘abusive’ and not individual games themselves. This separation between the labelling of the design process and the game applies also when discussing player-supportive and counter-supportive design and links to the importance of understanding designer intention (Wilson and Sicart, 2010, p.3). While it is possible to argue that a game object itself has player-supportive or counter-supportive properties through the way it appears to have been designed, this is not the same as a design philosophy having these same properties (i.e. the intended design).
Secondly, the mapping of player-supportive and counter-supportive design does not intend to suggest that all existing games may be conceived as having been designed utilising one philosophy or the other. A design that does not appear player-supportive is not therefore counter-supportive by default. The design of Scribblenauts (5th Cell, 2009) (Figure 1.6) provides an example to illustrate this point. In Scribblenauts players can create any one of more than 22,000 possible objects based on objects drawn from various encyclopaedias and dictionaries (Slaczka in Ohannessian, 2009, para.3). Players are tasked with solving a series of problems by combining these different objects in whatever way they think will be successful.

If this game is considered from the perspectives of player-supportive and counter-supportive design, it is evident that it does not appear to fit either philosophy. There are still game components that can be considered player-supportive, such as the ability to quickly restart a level after making a mistake, being able to pause gameplay, and save game data. However, the game design cannot be considered player-supportive due to the lack of guidance players are provided with in terms of how to solve the different problems the game presents them with. Players must discover through trial and error what objects the game is able to create and then how they may be creatively combined to achieve different outcomes.

Thus, the game does not actively support players in solving the presented problems. However, the game cannot be described as counter-supportive either, as it also does not actively seek to prevent or restrict the satiation of player needs. The design of Scribblenauts can thus be described as belonging to a third, different design space, that of player-unsupportive design, characterised by a game’s withdrawal of certain player-supportive components with a simultaneous lack of actively counter-supportive components.
Thus considering the mapping (Figure 1.5), it may be suggested that player-supportive and counter-supportive game design philosophies undermine the satiation of player needs at the highest level (Level 5) of Maslow’s hierarchy. Player-supportive design does this by providing support for the meeting of lower level needs and focusing primarily on achievement and skill mastery, meaning players are not required to engage at a cognitive level to a significant extent. Counter-supportive design does this by preventing the satiation of lower-level needs, which in turn prevents the satiation of higher level needs. However, in the case of *Scribblenauts*, the game can be seen to provide potential satiation of higher level needs such as creativity and problem-solving.

![Image](image.png)

*Figure 1.7: 'Survival instinct’ in Tomb Raider highlights all puzzle-related entities.*

Even games that may appear to provide opportunities for greater cognitive engagement and satiation of Level 5 needs may incorporate player-supportive game components that mitigate this. For example, *Tomb Raider* (Crystal Dynamics, 2013) appears to present players with cognitively challenging problem-solving scenarios but simultaneously provides a player-supportive game component that immediately minimises the cognitive effort required in the form of ‘survival instinct’ (Figure 1.7). Using this mode, players can quickly identify all puzzle-related entities in an environment, minimising the cognitive engagement and problem-solving required.

It would be highly unlikely for a game to be designed to be in every possible way player-supportive, or counter-supportive. For example, *PainStation* is physically abusive through its interface device’s ability to electrocute players, but the version of *Pong* that is played via the device is not abusive in and of itself. Thus, any game may contain components that in and of themselves are player-supportive or counter-supportive and the combination of these various components determine to what extent overall the player may feel that their needs have been satiated at different levels.
Player-supportive, counter-supportive, and player-unsupportive design, are thus suggested to exist as three related but separate design philosophies, within game design space. Both player-supportive and counter-supportive design philosophies undermine the satiation of high-level player needs. Player-unsupportive design (e.g. as exemplified in *Scribblenauts*) appears to move closer to allowing the satiation of such needs by withdrawing some player-supportive design components. However, such a design philosophy cannot be definitively associated with this game due to a lack of information regarding design intention (Wilson and Sicart, 2010, p.3). It is evident however that there is scope within game design space for such philosophies to exist and to thus be intentionally applied to games in the future.

However, while player-unsupportive design provides initial opportunities during the game’s early stages for players to engage with it at a cognitive level and thus, to potentially satiate Level 5 needs, these opportunities are not consistently available to players throughout a player-unsupportive game. *Scribblenauts*, for example, suggests a vast range of possibilities for solving problems. However, once players identify particularly useful items, those same items can be used from then on during gameplay to solve similar problems (Dutton, 2009). Thus, the level of cognitive challenge is decreased and performance of learned skills becomes the primary focus once more.

Thus, this raises the opportunity to consider how the initial level of cognitive engagement and potential for the satiation of Level 5 needs may be maintained more consistently throughout a player’s experience of a complete game. This would require a new design philosophy within game design space that may ‘disrupt’ the common pattern of linear development of ‘player skills’ whilst supporting high levels of cognitive engagement and in turn, satiation of Level 5 needs. This proposed new design philosophy is hence termed disruptive game design.

1.3.1: Defining Cognitive Engagement

The term ‘cognitive engagement’ requires more precise definition. Within games research literature, ‘engagement’ as a stand-alone term possesses a degree of definitional malleability. It is beyond the scope of the current research to enter into a detailed analysis of the term. However, some literature tends towards a definition that describes ‘engagement’ as low-level interaction with a game; for example, a player may be engaged without being ‘immersed’ (a similarly malleable term) (Brown and Cairns, 2004). However, this thesis takes a perspective similar to that presented by Douglas and Hargadon (2000, 2001), that ‘engagement’ with a game is a more demanding and self-aware form of interaction than, for example, ‘immersion’. ‘Engagement’ with a game from this perspective is necessarily ‘cognitive’ in nature. At a broad level, this perspective suggests that an ‘immersed’ player becomes less aware of the mediated nature of the play experience, moving from action to action with minimal thought and effort. Meanwhile an ‘engaged’ player may be more aware of the fact that they are ‘playing a game’ and be thinking more consciously about their actions.
‘Cognitive engagement’ is a term that is in frequent use across pedagogical literature and it is from this domain that a more specific definition to be used in the current research can be developed. The term is often defined in relation to the ‘quality’ of a student’s learning or the ‘depth’ of engagement with stimulus material. Davis, Summers and Miller (2012, p.22) for example define cognitive engagement as “the quality of students’ psychological engagement in academic tasks, including their interest, ownership, and strategies for learning”. Stoney and Oliver (1999) refer instead to cognitive engagement being linked to students “giving sustained, engaged attention to a task requiring mental effort and [. . .] extended engagement in optimally complex cognitive activities”.

![Figure 1.8: Bloom’s Revised Taxonomy (Krathwohl, 2002).](image)

‘Quality’ of psychological engagement and a student’s ‘mental effort’ can be considered against the structure of Bloom’s Revised Taxonomy (Anderson and Krathwohl, 2001, Krathwohl, 2002) that maps the skills within the cognitive process dimension of learning (Figure 1.8). Each of the layers of the taxonomy refers to different cognitive processes that manipulate ‘knowledge’ in different ways. The processes can be split into ‘lower-order’ thinking skills (i.e. remember, understand, and apply) and ‘higher-order’ thinking skills (i.e. analyse, evaluate, and create).

Newman (1990) defines lower-order thinking as involving routine recall and application of existing knowledge to situations or tasks that readily suit that knowledge (e.g. restating learned factual knowledge, or inserting new values into a learned mathematical formula). Applied to gameplay, this type of thinking can be mapped to the previously described performance of skills to overcome performative challenges. Newman’s definition of higher-order thinking meanwhile suggests that the individual should be challenged to “interpret, analyse, or manipulate information” (Newman, 1990, p.44) in new ways. For example, applying information selectively to abstract problems (analogous problem solving), or creating new strategies for approaching different tasks.
In games, this may include the creation of new combat strategies by combining the use of different game mechanics in creative ways, or having to dynamically respond to unexpected changes to game scenarios or game rules during gameplay.

In aiming to design games that encourage cognitive engagement, Bloom’s Revised Taxonomy can thus be used as a qualitative framework. Stoney and Oliver (1999) refer to higher forms of cognitive engagement being linked to the levels of higher-order thinking. Thus, heightened cognitive engagement in the current research can be evaluated by the type of higher-order thinking being engaged with by players. Each level of the taxonomy has a selection of action verbs associated with it, such as the list described in Wong and Wong (2009). This links, for example, the ‘evaluate’ level of the taxonomy to a selection of associated verbs including appraise, judge, compare, assess, contrast, and measure. Whereas in teaching, such verbs may be used to write appropriate learning outcomes for a lesson, in games, the same verbs may be used in the design of different challenges for the player with the aim of supporting higher degrees of cognitive engagement.

1.3.2: Properties of Disruptive Game Design

Drawing on one of the core principles of player-supportive design, disruptive design retains the aim of seeking to satiate, or allow the player the opportunity to satiate, the lower level needs of ‘physiology’ (Level 1), ‘safety’ (Level 2) and ‘belonging’ (Level 3); for example, by presenting players with expected game functionality such as being able to pause and being able to save and load games between play sessions.

Drawing then on the design philosophy of player-unsupportive design, disruptive design aims to present players with an initially unsupportive game framework that requires cognitive engagement (i.e. within the levels of higher-order thinking), problem solving and creative thought to understand and then interact with. This may be achieved for example by removing common player-supportive game components such as ‘tutorial’ sequences that teach players how to use game mechanics or interact with in-game objects ‘correctly’, in a safe environment. Instead, players may be provided with basic player-supportive game components, such as a ‘standard’ input device and ‘standard’ key binding configuration but be presented with in-game challenges or problems for which no support is provided to assist in finding solutions.

This initial player-unsupportive setup can then become disruptive if players are allowed to construct some understanding and thus, expectations, of game components while being afforded some time during gameplay to feel that they are learning and developing their knowledge of the game’s rules. That is, players are provided with short sections of gameplay in which they may feel some sense of increasing competence but not ‘mastery’. This established knowledge can then be disrupted by presenting players with game scenarios in which that learned knowledge and associated expectations become ineffective and inaccurate. Where a player-supportive game may present players with a linear “difficulty curve” (Bostan and Ogut, 2009) that requires players to
perform established actions or apply established knowledge with greater and greater refinement, a disruptive game requires players to constantly identify and learn new knowledge, or analyse and re-evaluate ‘established’ knowledge.

Furthermore, if a particular game system has certain properties that remain ambiguous to players, then designers have the flexibility to change those ambiguous properties, without warning the player, at any point during gameplay. This means that a player cannot assume, or have an expectation (Section 1.2.2) that, a game will behave consistently throughout its play time. Thus, changes to game systems that are not clearly communicated to players (as would be necessary if following a player-supportive design philosophy) have the potential to generate opportunities for a game to more effectively satiate needs at Level 5 of the hierarchy if the changes made remain possible for players to think through and work out.

The removal of clear and obvious player support and the consistent disruption of established in-game knowledge and expectations may encourage greater cognitive engagement (i.e. higher-order thinking) and problem solving, as well as provide more consistent opportunities for a player to be creative and spontaneous in their decision making and their actions. This in turn may provide a more fulfilling player experience upon game completion.

Conversely however, it has been suggested that players that are unable to establish a feeling of ‘competence’ during gameplay may demonstrate frustration and anger towards the game (Przybylski et al., 2014). Przybylski et al.’s work identified that games that purposely undermined feelings of player competence broadly tended to increase levels of aggression in those players (Przybylski et al., 2014, p.452-453).

However, while disruptive design may indeed disrupt what Przybylski et al. refer to as competence at a number of different points during gameplay, it does not seek to specifically prevent players from experiencing some periods of perceived competence, as abusive design would. It is thus necessary to consider not only ‘disruption’ as a design philosophy, but also in the game as created, the possible outcome of disrupting ‘too much’ or disrupting ‘too frequently’ during gameplay.

1.4: Research Question

This new disruptive game design philosophy requires exploration in terms of the opportunities it presents (or indeed, if it presents any at all) for creating enjoyable gameplay. This exploration may thus specifically focus on how the design philosophy supports the satiation of low-level player needs (Levels 1-3), while disrupting the satiation of mid-level needs (i.e. Level 4, achievement and performative skill mastery) and player expectations at different intervals during gameplay, to provide greater support for the satiation of high-level (Level 5) needs (Figure 1.4).

The previously defined structure of game space (Figure 1.3) can be used to guide this exploration. Within game design space and game development space, an initial question may consider whether or
not it is possible to implement such a design philosophy within a game. However, this only considers the highest level issues of theoretical and practical viability, which can be answered relatively easily. Theoretically it is possible to create game designs that disrupt player expectations and mid-level needs constantly throughout their play time. Practically, these designs could then be translated into playable game objects; there is no immediate barrier to this production process. However, this question does not consider whether the game as played following this process will be enjoyable for players. It also does not consider the impact on player enjoyment that a mismatch between the game as expected and the game as played may have.

Thus, to consider the player experience, it is important to understand the individual differences between players. It may indeed be possible to disrupt player expectations in a number of different ways, but it is unlikely that each of these will be equally effective for creating an enjoyable experience for all, or even the majority, of players. Each player’s individual prior game experiences and prior life experiences will influence how they respond to games designed using the disruptive game design philosophy. Thus, a more relevant research question may be proposed that asks how the disruptive game design philosophy may be operationalised into game features that provide opportunities for different individual players to have cognitively engaging gameplay experiences.

This question may be answered in an academic research context through the testing of different disruptive game components in purpose-developed games for research. However, if disruptive game design is to be evaluated in terms of its viability for use in the wider games industry in commercial games, data gathered from research-focused games is going to be of limited applicability. A non-commercial game will be responded to differently by players, because their game as expected will be influenced by different factors. A research-focused game for example may be more likely to be represented differently in its marketing (if it is ‘marketed’ at all) and may thus be situated differently in games culture to a commercial product. The question of operationalising the disruptive game design philosophy can thus be refined to specify that operationalisation occurring specifically within a commercial game development project that is designed, developed, and published, using industry-standard processes and played by a player-base that respond to it as a commercial product. This thus enhances the ecological validity of the player responses to the game as played.

As stated in Section 1.2.1, it is necessary to improve one’s philosophies via criticism (Popper, 1974). Releasing a commercial game that implements the disruptive game design philosophy that can then be played, analysed and evaluated by players provides an opportunity to receive such criticism. Analysis of this player feedback can then provide data regarding the impact of the disruptive game design philosophy on player reception of the game as played, in terms of their expectations (i.e. their individual game as expected), and the satiation of their various needs during gameplay.
The thesis therefore will aim to explore two interlinked questions.

1. **Within the constraints and limitations of game development space**, is it possible to operationalise the disruptive game design philosophy within a commercial game as created so that it supports cognitive engagement during the game as played?

2. Do players report their experiences of a ‘disruptive’ game as played in a way that suggests cognitive engagement with the game and satiation of ‘high-level’ needs, and if so what are their attitudes towards this type of game experience?

The first question aims to explore the application of a design philosophy from game design space (i.e. a ‘conceptual’ game), through game development space (i.e. a ‘physical’ game) and game publication space (i.e. a ‘physical’ game) resulting in a game as played (i.e. a ‘conceptual’ game). Through this process, the practical viability of the disruptive game design philosophy within a commercial game design and development project can be evaluated.

The second question then aims to assess, by way of player responses, the effect of the game developed using the disruptive game design philosophy on the game as played. It aims to assess whether players appear to have experienced cognitive engagement and high-level need satiation during play. In answering this second question it will be possible to draw conclusions about the broader commercial, cultural, and player-based viability of the disruptive game design philosophy. That is, whether the design philosophy has the potential to be used as a basis to make games that are responded to positively by players.

The analysis of the player responses to the game may also provide insight into the effect, if there is any, of differences and similarities between an individual player’s game as expected and game as played. Both of these questions, when combined, aim to provide data that can be used to evaluate the theoretical basis of the disruptive game design philosophy itself. This in turn may allow further refinement of the design philosophy’s principles and may lead to the future development of more effective, successful disruptive game designs.

### 1.5: Contributions to Knowledge

This thesis develops and presents a new model of game space (Section 1.2) that provides a new perspective on the different stages through which a game progresses; from conceptualisation through to being played and reported on by players (Figure 1.3). This contributes to knowledge by providing a structure through which to examine each of the specific stages of this process. The individual stages themselves (i.e. the game as designed, the game as created, the game as published, and the game as played) also each provide further contributions to knowledge as they are expanded (Chapter 3 to Chapter 8). This model also holds potential to be applied to other academic research as a framing device for the analysis of other games. The game space model is also used as a supporting framework to demonstrate how external factors outside of the
designer's control can influence a player's experience of a game by setting up initial expectations before the player even engages with the game itself (i.e. *the game as expected*) (Section 1.2.2).

This thesis presents a definition of the disruptive game design philosophy (Section 1.3) and proposes this philosophy as a new approach to designing games that support a player's cognitive engagement and the satiation of their high-level needs. The design philosophy is supported by the construction of the Ludic Cognition Model (LCM) (Section 3.2) which affords a new perspective on how a player's cognitive processes of information encoding, recall, and application, operate during gameplay (i.e. the last stage of the game space model; the *game as played*). The Ludic Cognition Model is used to propose new definitions for different types of 'ludic knowledge' (Section 3.3), along with a suggestion of how those knowledge types (i.e. intraludic, transludic, interludic, and extraludic knowledge) are constructed and used during gameplay (Section 3.4 and Section 8.10.4.2).

With the Ludic Cognition Model as a basis for understanding player cognition, the Ludic Action Model (LAM) is then constructed (Section 4.2) that embeds these cognitive processes within a new model of interaction between the player and the game. The Ludic Action Model provides a connection between the *game as played* and the *game as published* within the game space model. This Ludic Action Model combines existing psychology and game studies literature with new concepts to describe the interaction between the player and the game in a way that presents opportunities for disruption to occur (Section 4.3 and Section 4.4).

Following this, a conceptual framework of game components is proposed (Chapter 5), combining existing, separate works on the 'ludodiegesis' (Section 5.1), game components (Section 5.2), and later, the concepts of 'genre', 'mode', and 'milieu' (Section 6.4.3) into a new structure for considering the design and analysis of games. This conceptual framework provides the structure of the *game as created* within the game space model. However, as a contribution to theory in the field, it can stand alone as a potential basis for further research, or a potential lens through which to analyse other games.

The conceptual framework of game components is mapped to the Ludic Action Model (Section 5.3), enabling the proposal of a list of specific properties that disruptive game features may possess (Section 5.4). This list of properties, in conjunction with existing literature focusing on the game design and development process, is used to construct the new Disruptive Game Feature Design and Development (DisDev) model (Section 6.1 to Section 6.5). This model provides a tool for implementing games utilising the disruptive game design philosophy, whilst also providing a structure that is flexible enough to potentially be adapted to suit other design philosophies or approaches as well. The DisDev model expands the *game as designed* within the game space model.

With the game space model, Ludic Cognition Model (LCM), Ludic Action Model (LAM), conceptual framework of game components, and the Disruptive Game Feature Design and
Development (DisDev) model proposed, they are lastly combined into a single integrated model of game space (Section 6.6). This integrated model provides a single, comprehensive overview of player interaction with a game through the lens of the disruptive game design philosophy, embedded within the high-level structure of game space.

In order to assess the viability of the disruptive game design philosophy within an industrial game design and development context, it was necessary to apply the design philosophy to the creation of a commercial game artefact. This was achieved through the application of a bespoke Research through Design (RtD) process model (Section 2.4), constructed following exploration of existing methodological approaches within Design Research more broadly (Section 2.2 and Section 2.3). This new RtD process model is tailored to game studies and is linked to the stages of the game space model (Section 1.2), providing a potential methodological model for application to future research and development projects.

The game that was created alongside this research was the first-person horror-adventure game, Amnesia: A Machine for Pigs. The contextualisation (Section 7.1 and Section 7.2), design (Section 7.3 and Section 7.4), development (Section 7.5), and publication (Section 7.6) of the game are fully documented, with each stage providing insights into the practicality of applying the disruptive game design philosophy at each stage of a commercial game project. In addition to this, the game artefact itself represents a contribution to games culture in the form of an object that can be played, explored, and studied in its own right, potentially providing a basis for future academic research or a point of reference for future design and development projects.

A phenomenological template analysis of online player discussion data about their experiences whilst playing the game is then conducted, following the game’s public release. This analysis provides a number of specific findings that contribute to knowledge (detailed in Section 8.10) in multiple areas. Firstly, how players experience and report on ‘disruptive’ games and how their expectations prior to playing a game can have a significant impact on their attitude towards the type of experience the game actually offers (Section 8.10.1 and Section 8.10.2). Secondly, different types of disruption of different types of ludic knowledge may be more or less effective depending on the primacy of the game mechanics or features that are being disrupted (Section 8.10.3). Thirdly, the significant impact that marketing material can have on player expectations, which can then negatively affect the eventual player experience of the game (Section 8.10.4.1). Lastly, the data also provides interesting insights into the influence of a game or game franchise’s branding on player expectation and experience (Section 8.10.4.2), along with insights into player attitudes towards change and experimentation within the field of game design more broadly (Section 8.10.4.3).

Additionally, through the phenomenological template analysis, a further contribution to knowledge is made through the application of the template analysis method to rich, qualitative data obtained from a ‘naturally occurring’ (i.e. no experimental intervention) online player
discussion forum. This is a notably underutilised, yet beneficial, qualitative research method within the field of game studies, applied to a similarly underutilised data source. This thesis thus provides a detailed case study of the methodology’s application that others may utilise as a basis for designing a diverse range of future research projects. The phenomenological study also highlighted the need to obtain data about the game as played via the previously unconsidered game as reported (Section 8.1). This demonstrates that not only did this methodological approach contribute to understanding of design practice and player experience, it also contributed to the underpinning theory (i.e. the game space model) as well.

1.6: Structure of Thesis

The structure of this thesis follows the different stages of game space (Figure 1.3), with the disruptive game design philosophy (Section 1.3) being discussed in relation to each stage in turn. The thesis can be divided into three main sections: (1) the disruptive game design philosophy and theory, (2) the application of the theory to the design, development, and publication of a game, and (3) the analysis of player experiences of that game.

The formation of the underpinning theory for the disruptive game design philosophy forms the first section of the thesis and can be summarised in four parts.

Firstly, a model of Ludic Cognition is constructed (Section 3.2) incorporating Working Memory theory (Section 3.2.1, Section 3.2.2), Schema theory (Section 3.2.3), and Encoding Specificity theory (Section 3.3.1). This model is used as a basis to propose nine different knowledge types that a player may use during gameplay in order to understand and respond to the stimuli presented to them by the game (Section 3.3 to Section 3.6).

Secondly, the model of Ludic Cognition is placed in the context of a cyclical gameplay model. Perron’s Heuristic Circle of Gameplay (2006) is co-opted as a basis (Section 4.1), with additional components being proposed to provide a more comprehensive model capable of supporting the conceptual elements of the disruptive game design philosophy (Section 4.2). This model is termed the Ludic Action Model and is used to propose three specific types of disruption that a player may experience during play, based on the different stages of memory encoding, recall of knowledge, and performance of an action (Section 4.3, Section 4.4).

Thirdly, with the Ludic Action Model providing a basis for understanding how the player interacts with the game, a conceptual framework for the component structure of the game itself is proposed. Via Pinchbeck’s (2007, 2009b) ludodiegesis and Schell’s (2014) Elemental Tetrad of game components, this framework is split into the homodiegesis and heterodiegesis, within which the four elements of Technology, Aesthetics, Story, and Mechanics are placed (Section 5.1, Section 5.2). This conceptual framework is mapped to the Ludic Action Model (Section 5.3) providing a link between the components of the game itself and the cognitive and motor
processes involved in players’ interaction with it. This thus forms a framework that was applied to the design and development of a game that utilises the disruptive game design philosophy.

Lastly, Section 6.1 proposes the Disruptive Game Feature Design and Development Model as a practical tool for designers to utilise as a means of incorporating disruptive game features into their games. Alongside this model, the importance of considering an individual project’s contextualising factors (e.g. budget, development time, target audience, genre) is discussed (Section 6.4). To support this discussion, specific definitions of ‘genre’, ‘mode’, and ‘milieu’ are provided and mapped to the previously constructed conceptual framework of game components (Section 6.4.3). Section 6.6 links together all of the previously proposed theory into an Integrated Model for designing and developing disruptive games, forming the basis for the second section of the research.

The second section of the thesis documents the design (Section 7.1 to Section 7.4), development (Section 7.5), and publication (Section 7.6) of a commercial game title (Amnesia: A Machine for Pigs). This documentation provides a detailed case study on the use of the Disruptive Game Feature Design and Development Model in a live commercial context, allowing an opportunity for analysis and critique of the model’s benefits and limitations (Section 7.7).

The third section of the thesis presents a phenomenological study of player discussion data following their play of the game (Chapter 8). Through a Template Analysis, this study explores which disruptive game features were/were not experienced by players as they were intended to be experienced by design. The findings of this study are also used to make adjustments to the previously proposed theory underpinning the disruptive game design philosophy (Section 8.11) as well as being used as a basis for suggesting potential future avenues for ongoing research (Section 9.3).
Chapter 2:
A Methodology for Implementing and Evaluating Disruptive Game Design in a Commercial Game Project

Disruptive game design was defined (Section 1.3) as a design philosophy to provide greater support for satiation of player needs at the highest level (Level 5) of Maslow’s hierarchy (i.e. that of problem solving, creativity, spontaneity, and fulfilment). This fills a design space not catered for by traditional player-supportive design, player-unsupportive design, or counter-supportive design (Section 1.2.3.1). The necessity to design, develop and release a commercial game through which to operationalise the disruptive game design philosophy and obtain player response data following their play of that game was also identified. This necessity is due to the need to ensure that ecological validity of player responses to a ‘disruptive’ game as played is maximised and in turn, ensuring that applicability of findings to further commercial game design and development is also maximised.

However, commercial game design, as a methodological approach to carrying out industry relevant and ecologically valid research and development, requires exploration and evaluation (Section 2.1). Such an exploration firstly requires understanding of available methodological approaches to practice-based research within the field of ‘design’ (Section 2.2). Secondly, it requires understanding of how such approaches may be adapted to function specifically within a commercially constrained game design and development project (Section 2.3). The selected overarching research and development methodology (Section 2.4) is utilised to translate the disruptive game design philosophy, via the different stages of the game space model (i.e. game design space, game development space, game publication space and game-player space), into a playable game (Chapter 6).

2.1: Intentionality and Ecological Validity in Methods of Games Research

In evaluating the benefits of practical game design and development as a research approach, an initial question may ask why it is not possible to analyse existing games and evaluate their designs in the context of the disruptive game design philosophy. The textual analytical approach to the study of games represents one such method and has a number of notable examples, including Consalvo’s (2003) detailed analysis of sexuality and sexual representation in Final Fantasy IX (Square, 2001), Ekman and Lankoski’s (2009) analyses of sound and emotion in Silent Hill 2 (Konami Computer Entertainment Tokyo, 2001) and Fatal Frame (Tecmo, 2002), as well as Soderman’s (2010) analysis of political and economic representation in Every Day the Same Dream (Molleindustria, 2009).
Frayling (1993, p.5) describes such deep critical readings of texts, within the broader context of ‘art and design’, as being “Research into Design”. This approach focuses on properties and qualities of existing works, or on analysis of works from a particular perspective (e.g. social, political, or cultural). Such research provides new perspectives from which to consider games, based on the individual analyses of different researchers. However, intentionality is a vital consideration if one is attempting to evaluate a design approach or philosophy (Section 1.2) and there are fundamental differences between this type of “third-person study of the past” (i.e. games that have been developed and released) and the “first person” participatory study of the present (i.e. games in the process of being developed) (Chandler and Torbert, 2003, p.133). Note that here, intentionality is used to refer simply to the intended outcome of design decisions made during the game creation process and it is not the aim of the current research to discuss the philosophical interpretation of the term.

Thus, without directly engaging with the designers and developers of a game, it is not possible to reliably assign motive or intentionality to the design of any component of a game under analysis. A researcher cannot know whether a component was designed to behave that way, or whether it is simply a product of emergent behaviour as game systems interact with each other dynamically. Similarly, a researcher cannot know whether a perceived ‘bug’ in a game is indeed a bug, or whether it is a designed ‘feature’.

Equally, the interpretive nature of textual analysis (McKee, 2003, p.1) is fundamentally subjective. A researcher’s interpretation of a game may not align with the designer’s original intentions. Thus, it cannot be known whether such a reading is in fact the precise interpretation that the designer intended players to have. Therefore, textual analysis of games is able to analyse the game as played (Figure 1.3) by the researcher but not the game as designed (Figure 1.3).

The purpose and aim of a research project is important to consider in determining how it is to be carried out and also, how it is going to be disseminated to appropriate audiences. Wilson (in Goodwin, 2012a) elaborates on this, stating that

[. . .] if you’re asking concrete questions about game design, then sure, it might make a lot of sense to build things. But it might not. For example, consider ethnographers who study various aspects of game culture. Many of those researchers aren’t trying to "test" existing theories. They might instead be telling a particular story about a particular community, or simply raising new questions [. . .] We should remember that not all games research is focused on design [. . .] Research can have practical applications, but we shouldn’t demand that it does. (Wilson in Goodwin, 2012a, para.7-8)

Individual subjectivity is one of the strengths of textual analysis, as it allows the development of new perspectives on games, along with new research questions that may otherwise not have been identified. However, these must be considered in relation to the interpretive process through which they were generated. As McKee (2003, p.2, citing Grisprud, 1995) describes, the impact of
subjective interpretation can be mitigated by cross referencing that interpretation of a text with the interpretations of others to identify consistencies and differences.

However, even in cases where developers are willing to speak directly to researchers, providing a potentially useful source of data for defining intentionality of design or meaning in a game, they themselves are likely to frame their responses to ensure their company or development team appear in a positive light. If a feature was not specifically designed into a game, but subsequently emerged and was favourably responded to by players, the developer may assign a post-hoc explanation of it. Thus, textual analysis of existing games even in cases where researchers have access to the development team is still an analysis of data that has been filtered. It is filtered by the development team, potentially filtered by a publisher or marketing department and then filtered by the researcher’s own perspectives and aims for the particular analysis being written.

Thus, to be able to state definitive ‘intentionality’, research artefacts must be created for the express purpose of implementing and analysing a particular design approach or game component. If an artefact is designed and developed for the research, then the analysis is of the actual game-based data against the intended game-based data. It becomes possible to ask the question of whether for example a design philosophy, or game component, is received (i.e. in the game as played) as intended (i.e. in the game as designed/game as created). To carry out a robust analysis and evaluation of the effect of a disruptive game design philosophy it is therefore necessary to be able to clearly state intention. Thus, a methodology that affords the production of a game, or at least some component part thereof, is necessary.

El-Nasr et al. (2009) demonstrate this experimental approach in their design, development and testing of a system for dynamic lighting in game environments that assists in directing a player’s visual attention. The system is demonstrated to operate effectively in experimental testing conditions, using game environments specifically designed to test it. However, in this instance if such a system were to be introduced into a commercial game, it is not suggested how practical it would be to implement or indeed, how well it may operate alongside other game systems. Various other factors, not only a player’s visual attention, may be influenced by lighting (e.g. mood and emotional response to the game) and the proposed dynamic lighting system experiment does not consider the effect on these.

Interestingly, previous work (El-Nasr et al., 2007) describes a separate system for utilising dynamic lighting for enhancing tension in games, and affecting the player’s emotional experience. However, the two systems are not considered as a whole, nor are either incorporated into a commercial game. While both are demonstrated to operate well in experimental (reductionist) environments and independently of one another, there is no evidence to suggest that either system has real-world application in the forms presented. However, it is this latter context of real-world application that needs considering if the disruptive game design philosophy is to be considered as a possible, practical, design approach for commercial game design to utilise.
Thus, whilst the issue of identifying intentionality as an issue for post-hoc analysis may be resolved by designing research artefacts, the issues of ecological validity and in turn, real-world application, must be considered.

The current research requires an approach which affords an evaluation of how players relate their game as played to their game as expected. It is thus necessary to consider the potential differences between a player’s game as expected for a non-commercial (i.e. freely available) game and a commercial (i.e. paid-for) game. There is less ‘risk’ involved for the consumer when a product is free. If a consumer downloads a free game that they do not enjoy, they have only sacrificed their time. A paying customer however, loses both time and money if they do not enjoy the game. Thus, a player’s game as expected, in relation to a commercial game, will be informed by different elements of game culture and marketing (Figure 1.3) (e.g. awareness of ‘market value’ of games and cultural perception of ‘value for money’) than it would do in relation to a non-commercial game. Hence, a customer’s game as played will potentially be evaluated differently to a consumer’s game as played in relation to these different expectations. For example, a non-commercial game that the player does not particularly enjoy may still not be responded to as negatively as a commercial game that the player also does not enjoy. Therefore, to achieve a high degree of validity in the current research with regard the similarities of the outcomes to what may be expected in other commercial game scenarios, exposing a game artefact to customer feedback is vital.

In addition to research validity, developmental, practical, research is important in bridging the divide between theoretical work and commercial development work. While Pinchbeck (in Goodwin, 2012b, para.7) notes that this divide is not necessarily a large one, it nevertheless exists. If research is being carried out under similar restrictions to those that would be present in commercial game development, this serves to potentially make the research output more immediately applicable to further real-world commercial game development. Thus, the research is more ecologically valid in terms of the player comparison of game as played to game as expected, as well as in terms of the design and development processes applied to the game as designed and the game as created.

Moreover, although the reliability of the data gathered from player feedback on a commercial game released to the public may be decreased in comparison to, for example, a laboratory experiment, a justifiable and robust theoretical foundation (i.e. the design philosophy and its defined principles) for the development process provides a means for maintaining a degree of reliability. Even within this approach though, a rigorous methodological structure is necessary for carrying out developmental research. To this end, the closely related methodologies of Action Research and Design Research are of potential use. However, they must first be evaluated to determine the most appropriate methodology for the current research, or to identify components of them that may be restructured into a bespoke methodology.
2.2: Defining Action Research and Design Research

There is ongoing debate surrounding the methodologies of Action Research (AR) and Design Research (DR); the latter is sometimes also referred to as Design Science (Baskerville, 2008, Hevner et al., 2004, Papas, O'Keefe and Seltsikas, 2012) and Design Science Research (Iivari and Venable, 2009, Peffers et al., 2007, Sonnenberg and vom Brocke, 2012). Thus, it is necessary to identify precisely what these terms define.

The labelling of these methodologies implicitly suggests similarities or dissimilarities between them. Goldkuhl (2013) notes that if any comparison is to be made between AR and DR, it is first necessary to ensure that the labels are themselves equivalent. If one is aiming to compare “a design oriented research approach with action research, then the label design research should be used” (2013, p.2). This is a similar equivalency in labelling as applied by Cole et al. (2005), however it is noted in this case that assuming a labelling system such as this implies further similarities between the methodologies in terms of their ontological, epistemological and axiological properties and that these implied similarities may not necessarily be correct.

There is a division in existing literature in this field. There is work that suggests that indeed, AR and DR share a number of similarities and may be considered as concordant methodologies (Cole et al., 2005, Järvinen, 2007), while others suggest that the two are clearly separated by fundamentally different aims and principles (Baskerville, 2008, Iivari and Venable, 2009). Further to this, there are a number of examples of work that present efforts to combine elements of both methodologies to generate hybrid methodologies (Sein et al., 2011, Wieringa and Morali, 2012). The issue of defining AR and DR is further complicated by the existence of multiple different types of AR (Baskerville and Wood-Harper, 1998, Chandler and Torbert, 2003, Iivari and Venable, 2009), listed in Davison, Martinsons and Kock (2004, p.67) to include: Canonical Action Research (CAR), Information Systems Prototyping, Soft Systems, Action Science, Participant Observation, Action Learning, Multiview, ETHICS, Clinical Field Work, and Process Consultation.

In the context of the current research though, Canonical Action Research (CAR) (Cole et al., 2005, Iivari and Venable, 2009) possesses a number of properties that make it particularly suitable. Indeed, as Davison et al. (2004, p.68) state, “CAR is unique among all the forms of AR in that it is iterative, rigorous and collaborative, involving a focus on both organisational development and the generation of knowledge”. Moreover, CAR and DR can both be defined using criteria that projects utilising them must follow. These can be evaluated to assess their suitability for the current research.
2.2.1: Canonical Action Research (CAR)

<table>
<thead>
<tr>
<th>Criterion</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Principle of Researcher-Client Agreement</td>
<td>The Researcher-Client Agreement provides the basis for mutual commitment and role expectation.</td>
</tr>
<tr>
<td>Principal of Cyclical Process Model</td>
<td>The Cyclical Process Model’s stages are diagnosing, action planning, action taking, evaluating and specifying learning.</td>
</tr>
<tr>
<td>The Principle of Theory</td>
<td>Theory must play a central role in Action Research.</td>
</tr>
<tr>
<td>The Principle of Change through Action</td>
<td>Action and change are indivisible research elements, related via intervention focused on producing change.</td>
</tr>
<tr>
<td>The Principle of Learning through Reflection</td>
<td>Considered reflection and learning allow a researcher to make both a practical and theoretical contribution.</td>
</tr>
</tbody>
</table>

Table 2.1: Criteria for describing Canonical Action Research (CAR), following Cole (2005, p.328)

Canonical Action Research (CAR) focuses on five principles (Table 2.1). Iivari and Venable (2009) cite Rapoport (1970) as providing a widely accepted definition of the overarching aim of AR in general, as aiming “to contribute both to the practical concerns of people in an immediate problematic situation and to the goals of social science by joint collaboration within a mutually acceptable ethical framework” (Iivari and Venable, 2009, p.3). CAR is a client focused approach, aiming to solve a particular problem by providing a solution that is appropriate to that particular problem’s context, or environment, as well as the client involved. The client in this definition may be an individual or an organisation. In the field of Information Systems design in which much AR/CAR literature is based, the client is commonly a particular organisation.

In CAR, the researcher must themselves be a part of the research environment to enable them to interpret their observations in an appropriate context. Researchers must ensure that their interpretations are considerate of the individual nature of each different research environment, such as the influence of culture and the social values of the individuals within it. Cole (2005, p.327) thus suggests that CAR requires an idiographic method of enquiry in which individuals from the research environment or research community are incorporated into the research as collaborators so as to enable a greater degree of contextualisation of data.

Within the context of games research and game development, this notion of ‘collaboration’ taking place between the ‘researcher’ and the ‘client’ (the Researcher-Client Agreement in Table 2.1) may be viewed as taking place between the ‘researcher/developer’ and the ‘player(s)’ (the publisher may also be a part of the Researcher-Client Agreement if they are a separate party involved in the project). However, in the context of testing and analysing the disruptive game
design philosophy (Section 1.3), the notion of a Researcher-Client Agreement is problematic. It may indeed be possible to include players within the development and testing process. However to do so in a way that allows players access to the underpinning theory of disruptive game design would be to defeat the purpose of such design. Once players are aware that a game has been designed with disruption in mind, their expectations will be influenced and thus the disruption itself may potentially be less effective. Design Research (DR) provides a research approach that does not require this collaboration with the ‘client’ group and thus may be more appropriate for this particular research.

### 2.2.2: Design Research (DR)

In livari and Venable's (2009, p.3) comparison of AR and DR (which they refer to as DSR, or Design Science Research), they state that “when compared with AR, an essential difference is that DSR assumes neither any specific client nor joint collaboration between researchers and the client”. DR then becomes problem focused rather than client focused. The aim of DR is still to provide a practical solution to a problem, but that problem is not tied to one particular client or situation. It is a broader ‘problem space’ that potentially affects a number of different ‘clients’, or situations. In approaching the issues themselves within the problem space, one may also be able to offer approaches to specific problems for specific ‘clients’ associated with that space. However, this is not a necessary component of DR; the higher-level problem space is of primary interest.

| Design Research |
|-----------------|--------------------------------------------------|
| **Criterion**   | **Description**                                  |
| Design as an Artefact | Design research must produce a viable artefact in the form of a construct, a model, a method or an instantiation. |
| Problem Relevance | The object of design research is to develop technology-based solutions to important and relevant business problems. |
| Design Evaluation | The utility, quality, and efficacy of a design artefact must be rigorously demonstrated via well-executed evaluation plans. |
| Research Contributions | Effective design research must provide clear and verifiable contributions in the areas of the design artefact, design foundations, and/or design methodologies. |
| Research Rigor | Design research relies upon the application of rigorous methods in both the construction and the evaluation of the design artefact. |

*Continued on next page*
**Table 2.2: Criteria for describing Design Research (DR), following Cole (2005, p.327)**

<table>
<thead>
<tr>
<th>Criterion</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design as a Search Process</td>
<td>The search for an effective artefact requires using available means to reach desired ends while satisfying laws in the problem space.</td>
</tr>
<tr>
<td>Communication of Research</td>
<td>Design research must be presented effectively both to technology-oriented as well as to management-oriented audiences.</td>
</tr>
</tbody>
</table>

DR is based on a set of seven criteria (Table 2.2). The analysis of the disruptive game design philosophy via DR could thus be described as an exploration of a ‘problem space’, within game design space (Section 1.2.1), following the requirements of each of these criteria. There is no one particular client, rather the potential practical and theoretical contributions of the current research may be applicable to the wider game development industry and also the field of academic games research.

Further analysis of the criteria for DR set against the requirements of the research questions (Section 1.4) demonstrates the appropriateness of the methodology in this instance. To appropriately test and analyse the implementation of the disruptive game design philosophy, that testing and analysis must be carried out in conjunction with the production of a game artefact (Section 2.2). This is an ‘instantiation’ (Table 2.2). The problem space (Section 1.1) is relevant to the business of developing games in its search for a new design approach that may be developed into a financially beneficial approach for commercial game design. Furthermore, designing and developing a game artefact within the identified and defined commercial constraints (i.e. the ‘laws’ of the problem space) further fulfils the requirement of the ‘design as a search process’ criterion (Table 2.2).

Other criteria to address are the design evaluation and a method for conducting that evaluation, as well as the identification and appropriate communication of research findings. To consider how these criteria may be met, different versions of DR may be considered so as to move from a generalised understanding of the approach towards a specific DR methodology applicable to the current research.
2.3: Modes of Design Research

<table>
<thead>
<tr>
<th>Research into Design</th>
<th>Research through Design</th>
<th>Research for Design</th>
</tr>
</thead>
<tbody>
<tr>
<td>▪ Historical Research of a field.</td>
<td>▪ Materials Research.</td>
<td>▪ The end product is the artefact itself.</td>
</tr>
<tr>
<td>▪ Aesthetic or Perceptual Research.</td>
<td>▪ Developmental, project-based work that produces an artefact.</td>
<td>▪ The research aims and results are embodied in the artefact.</td>
</tr>
<tr>
<td>▪ Research into a range of theoretical perspectives, such as social, political or cultural research of a field.</td>
<td>▪ Action research, including a development report and other supporting documents that contextualise the produced artefact in relation to the research.</td>
<td>▪ The artefact both is, and presents to the viewer the results of, the research.</td>
</tr>
</tbody>
</table>

Table 2.3: Summary of Frayling’s (1993, p.5) three modes of design research.

Frayling (1993), noted previously in Section 2.2.1 for describing “Research into Design” (RiD), further adds two other types of research, namely “Research through Design” (RtD) and “Research for Design” (RfD). DR as a single research approach can be split then into these three types, or modes of DR (Table 2.3).

These different modes of DR each focus on particular aspects of the design process and those involved in it. The artefact also possesses different properties and purposes depending on which mode of DR is being employed. RiD is predominantly historical research into the field of ‘design’ itself and the societal, political and cultural influences upon it. RtD is focused on the materials and tools used in the creation of an artefact, including the theoretical underpinning of the design process, supported by detailed development documentation. RfD focuses not only on the development of an artefact but in presenting the results of the research within and via the artefact; research embodied within the developmental output itself.

Frayling’s definitions are targeted at the field of Art and Design, however they are also identified in Forlizzi, Zimmerman and Stolterman (2009) as being the three main “foci” for DR across a range of disciplines. Furthermore, they are utilised as a basis for describing different perspectives on DR by Lunenfield (2003, p.11), prefacing a collection of DR case studies in which DR within the field of game design in particular is represented (Laurel, 2003). Applying the definitions of these modes of DR to the field of games specifically therefore requires only minimal extension of existing ideas and an awareness of appropriate methodological process.

The current research aims to produce an artefact as part of the developmental work and thus, RtD is appropriate as it aims to focus on ‘materials research’ (i.e. methods and tools used to
implement the game, such as the design philosophy itself and the game technology used to
develop the game). In interviews conducted with a number of DR experts, Zimmerman et al.
(2010) identify a frequently cited issue of RtD projects not intending to provide theoretical
knowledge output (i.e. contributions to the research field) from their initial design. The artefact
itself is prioritised, with research contributions only being identified during reflection on the
development process itself, if they are explicitly identified at all.

[. . .] knowledge production, especially in the form of theory, never seems to be
an intended outcome at the start of a RtD project. Instead, it was either implicit and
remained implicit after a project concluded, or it only emerged from reflection after the fact. Some participants argued that “good” RtD usually does
lead to theory development even though it might not have been the original
intention of the research group. (Zimmerman, Stolterman and Forlizzi, 2010,
p.316)

The interviewees in this study called for greater attention to be given to the theoretical
knowledge contributions of RtD projects from the beginning of such projects. In the context of
the current research, the theoretical contributions are considered from the outset, as defined in
Chapter 1. This theoretical framing will underpin the development of the game artefact and be
used as a basis for analysing player responses to the game as played. Thus, the ‘theoretical
knowledge’ thread will be a core component of the research throughout.

2.3.1: Research through Design’s (RtD) Output in the Context of Game Design and
Development
A cause for the lack of attention being given to theoretical knowledge output (Zimmerman,
Stolterman and Forlizzi, 2010, p.316) is suggested as stemming from a lack of the detailed
contextualising documentation that is a key component of RtD. In the current research however,
the requirement for supporting documentation as a key component of an RtD project (Table 2.3)
can be fulfilled via the game design document (Appendix E), the design and development process
report (Chapter 7), and the development post-mortem report (Appendix F). These documents
have both commercial use (e.g. the design document and the post-mortem report) and research-
oriented use (e.g. the development process report). Additionally, the academic research can be

While the supporting documentation for an RtD project is important, it is necessary to
understand what the content of that supporting documentation is able to provide (i.e. the
‘theoretical knowledge’ contribution). Forlizzi et al. (2009) identify “conceptual frameworks”,
guiding philosophies” and “design methods” as being principal outputs of RtD documentation.

The work can result in the conceptual frameworks for design and evidence of the
value of guiding philosophies for design. In addition, it can result in methods in
support of conceptual frameworks and guiding philosophies. This research can
also produce new problem framings that suggest preferred states for the world.
Finally, this approach results in new product forms that broaden the space of
design. (Forlizzi, Zimmerman and Stolterman, 2009, p.293)
Conceptual frameworks in this understanding are employed as a means of informing the design process and the design decisions made during it. They provide a particular approach to the design process. Guiding philosophies are similar to these conceptual frameworks but are more abstracted and viewed as more of “an attitude than a specified approach to design” (Forlizzi, Zimmerman and Stolterman, 2009, p.292). Design methods are employed alongside both conceptual frameworks and guiding philosophies, and are suggested as being necessary to present alongside them to present a more coherent and complete account of an RtD project. The design method demonstrates the production of an instantiation of a conceptual framework and guiding philosophy, i.e. the game as created (Section 1.2.1), which is an instantiation of the design philosophy (Section 1.3).

2.3.2: Research through Design (RtD) with Commercial Constraints

Another suggested criticism of RtD, is that it allows researchers to operate without concerning themselves with commercial constraints (Gaver and Martin, 2000), such as production or publication practicalities, or external pressures to meet fixed deadlines. Gaver and Martin provide examples of such an unrestricted RtD approach using a collection of “conceptual design proposals” (2000, p.209). From this collection, devices such as the “Dawn Chorus” (2000, p.210) (an intelligent bird-feeder that uses behavioural psychological principles to train birds to sing the owner’s favourite songs) are unlikely to be commercially successful, but the thinking behind their conceptualisation potentially opens new design spaces.

On the one hand, [the concepts] serve as suggestions that digital devices might embody values apart from those traditionally associated with functionality and usefulness. On the other, they are examples of research through design, balancing concreteness with openness to spur the imagination, and using multiplicity to allow the emergence of a new design space. (Gaver and Martin, 2000, p.209)

While the removal of commercial concerns is beneficial for ‘speculative design’, RtD is still possible in a context where commercial constraints remain present. However, the degree of financial risk is increased if research is to be carried out via a commercial product. The design and development methodology for a commercially situated RtD project must thus be clearly structured to enable the assessment of commercial risks (e.g. what external, commercial pressures may affect the project) and implementation risks (e.g. what factors may affect the project’s ability to be completed, or to make a return on investment) effectively.

However, Pinchbeck (in Goodwin, 2012b, para.7) does note that a commercially situated research project that takes risks and fails commercially (e.g. does not at least break even, or is poorly critically received) can still be considered a success in terms of research, presuming that the failure occurs in a way that furthers the discourse and knowledge base around the area of interest. The challenge in a commercially situated RtD project thus becomes assessing and minimising the risk of, and likely impacts of, potential commercial failure, then balancing these risks against the necessary risks that must be taken to fulfil the project’s research aims and objectives (e.g. utilising an untested design approach to be able to analyse and evaluate the effect
of it on the *game as played*). This thus affords a commercially situated RtD project to be successful in at least one capacity; even if the designed and developed artefact fails commercially, it will still provide valuable research output in the form of contributions to knowledge in the area being explored.

In terms of identifying new design spaces, it is important to clarify that RtD does not aim to produce the *best* artefact to describe a design space. Rather, it aims to produce *an* artefact, with clear theoretical justification behind its design and development, that is able to exist within a new design space, to provide what Gaver and Martin (2000, p.216) describe as a ‘placeholder’.

> [The proposed conceptual designs’] overriding function was to serve as landmarks opening a space of design possibilities for future information appliances. As such, the concepts are placeholders, occupying points in the design space without necessarily being the best devices to populate it. (Gaver and Martin, 2000, p.216)

These placeholders act as markers, identifying potential design spaces for future investigation and establishing a research and development dialogue around the issues that the design space potentially includes.

**2.4: An Iterative, Stepwise Method for a Research through Design (RtD) Project**

In aiming to test the practicalities of designing and developing a game utilising the disruptive game design philosophy in a commercial project, the design and development process itself is a vital part of analysing and evaluating those practicalities. It is not only the outcome of the design and development (i.e. how players respond to the *game as played*) that provides important data, but also how that implementation was realised from design (i.e. the *game as designed*), through development (i.e. the *game as created*), and finally through publication and release (i.e. the *game as published*) (Figure 1.3).

It is important to thus identify the specific steps that make up an RtD project and what the outputs of each stage are, as well as how those outputs may then contribute to future understanding within the field of study and future commercial projects.
Within the field of games research, Pinchbeck (2010) provides a simple, six-stage method for the research approach that he terms development-led research (Figure 2.1). Pinchbeck writes that in development-led research,

the work is driven by exploring design vacuums and the serious question of whether there are new game forms, or new understandings of existing game forms located within them. It is based upon a robust, representative and objectively available set of data [...] created by extensive cataloguing and analysis: but always in that order. Catalogue, analyse, identify, design, release, analyse. Break the order and it ceases to be research; fail to catalogue and we are left only with our own 'expertise' that, whilst critical, is never enough. (Pinchbeck, 2010, p.6)

Through this approach, it is firstly necessary to objectively catalogue the focus of the research. This objective data can be drawn from existing games and existing research. From this catalogued data it is then possible to analyse and identify potential avenues for new design philosophies, or for the exploration of new game components. By objectively cataloguing existing design philosophies, or design spaces, it is possible to then identify where gaps between those spaces – what Pinchbeck (2010, p.6) refers to as “design vacuums” – may potentially be situated. This in turn can focus research attention onto what may be available as a target for further investigation. The investigation of such vacuums can be carried out through a process of design that aims to create games to be released into them. Finally, the feedback from players of those games can be analysed to refine the design theory and design/development process that drives future ventures into those design spaces.
The theory 'iteration' therefore occurs between released games, with the second stage of analysis (the analysis of player feedback to a released game) informing the next cycle of identification, design and release. It may also be necessary, depending on the findings of the player feedback analysis, to return to the earlier stages of cataloguing and initial analysis to develop a more robust foundation on which to create further designs.

Figure 2.2: Design Research Process Model, following Vaishnavi and Kuechler (2004, p.7)

The method for development-led research identifies clearly separate stages but does not define what the specific outputs of each of those stages are. However, Pinchbeck’s stages hold a number of similarities to a similar stepwise process for Design Research (DR) (Figure 2.2) proposed by Vaishnavi and Kuechler (2004), building on the work of Takeda (1990). This process provides greater detail in its descriptions of the different stages as well as suggesting what the ‘visible’ output from each stage will be. Importantly, this process also indicates the point at which one or more contributions to research knowledge may be generated. It is this focus on knowledge contribution that defines this process as a design research process, rather than being a design process alone (Vaishnavi and Kuechler, 2004, p.7). Furthermore, by specifying sources of research knowledge output in the model, the previously noted (Section 2.3) issue of RtD projects lacking focus on research (i.e. theoretical) knowledge output, is mitigated.

As demonstrated in the process model (Figure 2.2), the contribution to knowledge is informed by data and documentation from multiple stages in the stepwise process, coming potentially from the Development, Evaluation and Conclusion stages. The body of documentation that describes the development and evaluation steps provides knowledge contributions in the form of process information. This documentation may provide information about how different components of a game (e.g. different game mechanics) were iterated and improved over the course of the project. The conclusion of the project provides knowledge contributions in the form of operational principles and design theories.

It is important to note that, as per Takeda’s (1990) original work, Vaishnavi and Kuechler’s (2004) process model includes circumscription as a means of framing the knowledge contributions from
the development and evaluation stages. This is based on McCarthy’s (1980) definition of circumscription as a form of ‘informal logical reasoning’: “a rule of conjecture that can be used by a person or program for ‘jumping to certain conclusions’” (McCarthy, 1980, p.2) when it is impossible, or impractical, to have access to all information about a problem. Following this definition, Takeda (1990, p.44) states that within the process model, it is assumed that every piece of knowledge is valid only when it is used in certain situations. However, we can only identify the applicability of knowledge when detecting a contradiction. (1990, p.44)

Thus, the knowledge contributions that stem from the development and evaluation stages of the process model are valid within the context of the particular project that generates them. They are then assumed to be correct for other contexts (i.e. other game design projects, in this case) until a context is found in which they become incorrect (i.e. a contradiction is detected). The causes for the incorrectness can then be evaluated and the original knowledge modified in response. This modified knowledge is then subject to a further process of circumscription until other contradictions are detected in future research. Thus, knowledge is continually refined through an ongoing process of research.

The implication of this circumspective reasoning for the RtD process is that iteration of a theory across multiple design projects is required to construct a more complete understanding of the context(s) within which the theory holds true. However, an initial artefact is required to be developed to provide the “placeholder” (Gaver and Martin, 2000, p.216) for the design space in the first instance. Designing and developing this initial artefact to represent the disruptive game design philosophy within game design space is thus the aim of the current research.

![Figure 2.3: Research through Design (RtD) process model for a game design and development project, adapted from Vaishnavi and Kuechler’s (2004, p.7) Design Research process model.](image-url)
Vaishnavi and Kuechler’s (2004, p.7) process model, in conjunction with terminology proposed by Pinchbeck (2010), can be used as a basis for a process model to be applied to a Research through Design (RtD) game design and development project (Figure 2.3).

This model will form the basis for the current research and as such, includes additional stages that contribute to research knowledge, as well as produce additional outputs. These additional stages are still subject to circumscriptive logic (i.e. the knowledge gained from them is assumed to hold true for other contexts until a context is identified in which the knowledge does not apply) and it is still the final conclusion stage that provides knowledge in the form of operational principles and design theories. These stages can now be described within the specific context of the current research into the effect of the disruptive game design philosophy on the game as played and how players compare their play experience to their game as expected.

2.4.1: ‘Awareness of Problem’ and ‘Suggestion’

The current research has identified a problem space in which existing approaches to game design do not appear to support satiation of high-level player needs (Section 1.2.3). This problem space is contextualised within the wider game space (Section 1.2). The problem space definition is informed through ‘cataloguing and analysis’ (Pinchbeck, 2010) (Figure 2.1) of existing academic literature, industry practice and industry texts.

From this, a design philosophy for disruptive game design has been suggested (Section 1.3). This suggestion draws on existing literature in supporting fields (Maslow’s (1943) hierarchy, Bloom’s (Anderson and Krathwohl, 2001, Krathwohl, 2002) revised taxonomy) and is situated within a model of game space (Figure 1.3) that separates game design space, game development space, game publication space, and game-player space as necessarily separate conceptual spaces, each containing a different ‘version’ of a game. The suggestion and the associated design philosophy are abductively drawn from existing knowledge, as per Peirce’s (1931, 1932, 1933, 1934, 1935) theory of abductive logic. That is, based on what knowledge is currently available, the suggestion is the most apparently appropriate method of investigating a particular research question or problem space.

2.4.2: ‘Design’

The design philosophy is to be instantiated into a commercial game artefact (Chapter 7), via a design and development framework (developed across Chapter 3, Chapter 4, Chapter 5, and Chapter 6).

The process of design itself contributes to research knowledge in two ways. Firstly, the output of design documentation provides a record of the design process, problems encountered and solutions determined. This design documentation may include ‘traditional’ games industry documents (e.g. a ‘high concept’, a pitch document, a design document, and a story/narrative document) but should also, if the aim is to understand the rationale behind the design process, include a record of design decisions. This may take the form of, for example, a design diary or commentary that can later be used as reference for the researcher in understanding why decisions
were made at the time that they were made (i.e. this provides a record of the context of the design process). This stage of the process model culminates in the game as designed that will be developed into the game as created.

2.4.3: ‘Development’
As with the design process, the development process (i.e. the creation of game assets) must also be documented to understand the influences of the development process on the original design intention(s). This ensures that the researcher is able to analyse the practicalities of implementing the design philosophy within the constraints of the real-world development process and thus, is able to evaluate the differences between the game as designed and the game as created. The decisions made during development, especially any compromises (or additions) to the original game as designed, must thus be recorded along with the reasoning for them. The output of this stage is the game artefact (i.e. the game as created).

2.4.4: ‘Publication’
The publication stage may be less applicable to some game-based RtD projects (e.g. the game artefact is not being widely released to the general public, instead being used purely within a laboratory context for a specific research project). However, in a commercially situated project such as the current research, the publication stage is a definable, separate stage of the RtD process.

If the publication process involves a project receiving ‘sign-off’ or approval from a company separate to the design and development team (or from senior management or an in-house publishing division in the case of larger development studios), this introduces an additional consideration for the practicalities of the design philosophy. As with design and development, contextualising documentation should be maintained throughout the publication stage also. A project that is reliant on publisher approval to be released may have to be modified to receive that approval. This would thus lead to a situation in which the game as designed, having been potentially modified during the development of the game as created, is modified further still before being released as the game as published. In an RtD project, such modification may be detrimental to the research aims of the project if the game as published differs significantly from the original game as designed or indeed, the game as created. Thus, modifications made due to publisher requirements require contextualising documentation so that those changes can be analysed by the researcher. The output of this stage, following any required modifications (if any), is the game product (i.e. the game as published).

2.4.5: ‘Evaluation’
Once the game artefact is created and published, it must be evaluated based on the expected behaviour of the artefact so as to be able to assess the effect of the intended design on the player experience (i.e. the game as played). In the structure of game space, this evaluation will be an evaluation of the player responses to the game as played. It may also include evaluation of player
discussions regarding the differences and similarities between the *game as expected* and the *game as played*. This feedback and discussion from players provides a means of assessing the performance of the game artefact and, in turn, the design philosophy that underpins it (i.e. the disruptive game design philosophy).

This stage in the process is emphasised by Vaishnavi and Kuechler (2004, p.8) as exposing an “epistemic fluidity” with regard to hypotheses about expected behaviour of the artefact. Whereas a positivist approach may either support or contradict hypotheses and draw conclusions, a design research project considers the artefact behaviour against the hypotheses and must then enter a second phase of problem awareness and suggestion. This then “suggests a new design, frequently preceded by new library research in directions suggested by deviations from theoretical performance” (Vaishnavi and Kuechler, 2004, p.8); that is, hypotheses that were not confirmed regarding the artefact behaviour may provide insight into the direction of future research and the design of future artefacts. A consideration of such possible future directions should be presented as a component of the evaluation stage.

**2.4.6: ‘Conclusion’**

The conclusion must state the knowledge contributions of the design research project (Figure 2.3). In the context of the current research, the dissemination of results may be considered for both academic audiences (e.g. research publications) and industry audiences (e.g. development post-mortem report). Communication of findings must be appropriately focused for each of these different audiences. It is suggested that at this stage, knowledge gained from the research may be categorised in terms of its explicability.

Knowledge gained in the effort is frequently categorised as either “firm” – facts that have been learned and can be repeatably applied or behaviour that can be repeatably invoked – or as “loose ends” – anomalous behaviour that defies explanation and may well serve as the subject of further research. (Vaishnavi and Kuechler, 2004, p.9)

An academic audience may have a greater interest in both what has been suggested to be ‘firm’, as well as any identified ‘loose ends’ which may form the basis of further research. Industry game designers and developers may instead gain more immediate benefit from a consolidated account of ‘firm’ findings (e.g. findings relating to design or development practices) that may be directly deployable in further commercial projects.

**2.5: Chapter Summary**

This chapter presents an argument for the use of a practice-based approach to games research. This argument is rooted in the requirements for understanding of design intent (i.e. *intentionality*) and for ecological validity in relation to the *game as designed*, the *game as created*, the *game as published*, and the *game as played* (Section 2.1). These two factors (i.e. intentionality and ecological validity) influence the applicability of game design research output to industrial game design and development.
Design Research (DR) has been identified as an appropriate methodological basis for the current research as it focuses on problems rather than clients (Section 2.2). Research through Design (RtD) has then been specifically identified as the most appropriate mode of DR to apply due to its focus on both artefact production as well as the development of design documentation and design theory (Section 2.3).

RtD is discussed in the context of a commercially situated game design and development project, with an analysis of stepwise methodologies provided by Pinchbeck (2010) and Vaishnavi and Kuechler (2004) (Section 2.4). While Pinchbeck provides a list of stages for a ‘development-led research’ project, Vaishnavi and Kuechler provide a substantially more robust and defined methodology. Vaishnavi and Kuechler’s process model also explicitly identifies the point in the process where contributions to research knowledge are provided.

Lastly, additional stages have been added to Vaishnavi and Kuechler’s (2004) process model to tailor it towards the defined stages of the game design, development, publication, and play process defined in the model of game space (Section 1.2). The named outputs of each stage have also been specified to make them directly applicable to the current research.

The next stage in the research process is to develop the design and development framework that will be used to guide the application of the disruptive game design philosophy to the game as designed and the game as created.
Chapter 3:
The Ludic Cognition Model of the Game as Played

While the definition of the disruptive game design philosophy (Section 1.3) does not aim to design games for the player (i.e. games that are highly player supportive), it is necessary to nevertheless design to the player (i.e. games that are still able to be interacted with and played by players). As suggested by a number of scholars, games may exist as ‘game objects’ or ‘texts’ in the world, but only exist as games when they are played by players (i.e. the game as played). Calleja (2011, p.8) suggests that until it is played, a game “is only a set of rules and game props awaiting human engagement”, while Ermi and Mäyrä (2005, p.1) state simply that “there is no game without a player”. While there may be exceptions to this perspective (e.g. Björk & Juul’s (2012) definition of ‘zero-player games’), the majority of contemporary commercial games require engagement from a player to move beyond being a set of rules and props forming a static ‘game object’, into a ‘played game’.

Thus, regardless of the type of experience the disruptive game design philosophy (or any game design philosophy) aims to give the player, ultimately the aim is to give the player an experience. To make informed design choices when designing and developing this player experience, it is therefore necessary before the design process begins to form an understanding of how the player ‘experiences’. An understanding of how a player ‘experiences’ a game will in turn allow design decisions to be made based on how they may effect that experience. This understanding of how a player ‘experiences’ must be further supported by an understanding of how the moment-by-moment, individual player’s experience of the game as played is then constructed. The game as played is conceptual, existing ‘within the player’ as an internalisation of that which is presented to them via the stimulus of the game and which may be influenced by other internalised knowledge (i.e. the game as expected and the player’s prior experiences). Thus, to effectively design to the player, a combined understanding of the ‘experiencing’ of a game and the internalised construction of that experience as the game as played is required.

Having an understanding of this conceptual game as played allows the design and development processes to be undertaken with more specific focus on the factors involved in its construction. From the perspective of the disruptive game design philosophy, this may enable the design process to generate disruptive game features that support the primary aim of the philosophy, maintain cognitive engagement during gameplay and hence, fulfil high-level player needs (Section 1.3.1).
Figure 3.1: Chapter 3 focuses on the player in game-player space.

Thus, a model of the game as played, within game-player space (Figure 3.1), is required which takes into account both the game as expected and the prior experiences of the player. In terms of the RtD process model (Section 2.4), this definition forms part of the supporting theory for the design philosophy (i.e. the output of the suggestion stage).

The model, constructed through this chapter, contains three key areas of cognition drawing on existing psychological literature. Firstly, the psychological processes through which players respond to stimuli presented to them during gameplay, both from the game itself and from other co-occurring stimuli from the ‘world’. Secondly, the processes through which those players construct knowledge based on stimulus information. Thirdly, how players construct meaning from both prior and current experiences and thus, decide on appropriate actions to take in response to stimuli. Player expectations (i.e. the game as expected) are formed through these meaning-
construction and knowledge-construction processes. A firm understanding of the game as played is required to then develop a framework for the game as designed that is able to purposely seek to affect the game as played in different ways (i.e. to disrupt it via the principles of the disruptive game design philosophy).

3.1: A ‘Systems Model’ of Cognition

Using behavioural and cognitive psychological theory as a basis, a ‘Systems Model’ of cognition can be constructed (Figure 3.2) that enables an initial understanding of how a player may experience the game as played.

Specifically, this Systems Model is constructed from the structure of Atkinson and Shiffrin’s (1968) multi-store model of memory, in conjunction with the additional structural components and memory processes proposed in Baddeley’s (Baddeley, 1986, 1992, 2000, 2002, 2007, Baddeley and Hitch, 1974, Baddeley and Warrington, 1970) theory of working memory. Within the model, an individual’s memories are separated into three different types of memory, Procedural, Semantic, and Episodic, based on the work of Tulving (1985a, 1985b). These different knowledge types are stored in different memory stores and organised in cognitive schema frameworks (Anderson and Pearson, 1984, Bartlett, 1932, Martin, 1994, Piaget, 1970, Rumelhart and Ortony, 1977, Turner and Turner, 1991, Turner, 1994). Information that is recalled from memory is retrieved from these schema frameworks in a manner comparable to Collins and Loftus’ (1975) spreading activation theory of recall, whereby ‘nodes’ in a memory network (in this case, a network containing schemas) are activated depending on their ‘distance’ from the originally activated node. Nodes that are closely related (e.g. have similar meaning for an individual) are likely to be activated together.

The Systems Model is explained further in this section, while Appendix A contains a full review of the psychological literature used in the formation of the model.
Figure 3.2: A 'Systems Model' of Cognition.
In the Systems Model (Figure 3.2), the ‘stimulus’ and ‘response’ components are drawn from behavioural psychology literature (Pavlov, 1927, Thorndike, 1913, Watson, 1930, Watson and Rayner, 1920) (Appendix A.1), along with the body’s ‘sensory perception’. Stimulus information must be perceived (i.e. sensory perception) and attended to by the body in order to be an instigator of further processing.

The cognitive processes involved in this ‘attending to’ and that then lead to a response require support from the field of cognitive psychology (Appendix A.2). ‘Attended to’ information must be processed in order for knowledge and memories to be constructed and/or recalled and in order for decisions to be made about what response to take in reaction to a stimulus. Early models of memory (Atkinson and Shiffrin, 1968, Waugh and Norman, 1965) provide an initial basis for understanding this information processing (Appendix A.2.1). However, these have been superseded and built upon by a widely accepted ‘working memory’ theory (Baddeley, 1986, 1992, 2000, 2002, 2007, Baddeley and Hitch, 1974, Baddeley and Warrington, 1970) (Appendix A.2.1.3) of how such cognitive processing may occur.

Working memory is suggested to be made up of multiple memory stores that actively manipulate information rather than passively store information. The central executive (Figure 3.2) is the component that controls an individual’s attention, allowing focus to be given to different sensory stimuli. This central executive is of a limited capacity, meaning that only a finite number of stimuli can be given attention at any one time. The central executive is not in itself a storage area, instead acting as a coordinator for attention. This component is then assisted by the two additional systems: the phonological loop which is concerned with short-term memory and rehearsal of audio and verbal information (via what Baddeley (1986) refers to as “subvocal rehearsal”) and the visuospatial sketchpad that provides similar functionality for visual and spatial information (Baddeley, 2002, p.86). The episodic buffer (Baddeley, 2000) lastly, is described as providing a temporary interface between the slave systems (the phonological loop and the visuospatial sketchpad) and LTM [long-term memory]. It is assumed to be controlled by the central executive, which is responsible for binding information from a number of sources into coherent episodes. Such episodes are assumed to be retrievable consciously. The buffer serves as a modelling space that is separate from LTM, but which forms an important stage in long-term episodic learning. (Baddeley, 2000, p.5)

This working memory theory has been developed further with the separation of working memory into declarative and procedural working memory (Oberauer, 2009, 2010) (Appendix A.2.2), where declarative memory contains the ‘contents of memory’ (i.e. the information to be processed) and procedural memory controls the actions and processes that will be applied to the ‘contents’ (i.e. the ‘method’ of processing the information). In the Systems Model (Figure 3.2), working memory is positioned as per Baddeley’s work as the receiver of incoming stimulus information and the instigator of knowledge retrieval from long-term memory, coordinated by the
Central Executive. It is further divided into declarative and procedural working memory reflecting the work of Oberauer.

The knowledge stored within long-term memory is suggested in the Systems Model to be stored in three distinct memory types (i.e. procedural, semantic, and episodic memory) as per Tulving’s (1985a, 1985b) work (Appendix A.2.2). These different types of knowledge are stored in schema frameworks, where schemas are broadly defined according to Bartlett’s (1932) definition:

‘Schema’ refers to an active organisation of past reactions, or of past experiences, which must always be supposed to be operating in any well-adapted organic response. That is, whenever there is any order or regularity of behaviour, a particular response is possible only because it is related to other similar responses which have been serially organised, yet which operate, not simply as individual members coming one after another, but as a unitary mass. (Bartlett, 1932, p.201)

Martin (1994) provides a more contemporary definition of schema that draws on Bartlett’s but further emphasises the primary role of schemas as enabling prediction and planning based on prior experiences.

By recording experience, schemata allow history to operate on future behaviour. This is a critical insight because it is the role of schemata in the prediction and the planning and execution of behaviour that are their primary reason for continued existence. In short, schemata allow memory to affect the behaviour of a system by assimilating information for future use. (Martin, 1994, p.270)

Stored knowledge is accessible through schema-based recall and ‘spreading activation’ of nodes in the schema network (Collins and Loftus, 1975) (Appendix A.2.3). Encoding of information into long-term memory is suggested in the Systems Model to occur in one of three ways as per the work of Rumelhart and Norman (1976); by accretion (i.e. the assimilation of information into an existing schema structure), by tuning (i.e. the modification of a schema’s structure to accommodate new information), or by structuring (i.e. the creation of an entirely new schema to store new information that cannot be contextually related to existing knowledge in other schemas). A full discussion of these three modes of encoding is provided in Appendix A.2.3.3).

Retrieved knowledge is passed, via induction (Gick and Holyoak, 1983), into a ‘dynamic schema’ (Rumelhart et al., 1999) (Appendix A.2.5). Induction is a process that brings together a collection of seemingly relatable knowledge from memory that may be useful in understanding and responding to the current situation. The dynamic schema is thus created as it is required (rather than being recalled fully formed from memory) and in relation to the current context, to interpret the current stimulus in relation to existing stored knowledge. The ‘dynamic schema’ is thus the basis for the body performing a response to that stimulus.

The combination of existing psychological research to form the Systems Model (Figure 3.2) leaves some ambiguity. Specifically, the dynamic creation of schema instances and how this process is coordinated by the Central Executive and working memory, is not clearly defined. The disruptive
game design philosophy focuses on the disruption of player expectations, which may originate from and be influenced by prior experiences and stored knowledge in long-term memory. Thus, it is the very area in which some ambiguity remains (the coordination and creation of the dynamic schema instance using knowledge recalled from long-term memory) that the disruptive game design philosophy still requires greater understanding of.

Furthermore, a model that is to be used as a basis for the game design process requires explicit focus on the key actions and processes engaged with during gameplay (i.e. the game as played). These actions and processes may then be more specifically relevant to game design (i.e. the game as designed) than more generally applicable actions and processes of ‘everyday’ cognition. The Systems Model draws on psychological theory intended as a means of explaining cognition more generally and is therefore not best suited to these requirements. However, the Systems Model nevertheless provides a robust foundation from which to further construct a refined and more specialised gameplay-specific model of cognition.

Thus, a more pragmatic, ‘Ludic Cognition Model’ (LCM) of knowledge construction and recall is required to expand working memory and the dynamic schema into more gameplay-centric components and processes. Note that, in this particular context, the term ‘ludic’ is being used to differentiate between cognition during gameplay (i.e. ludic) and cognition during other activities or everyday life (i.e. non-ludic).

3.2: Developing the Ludic Cognition Model (LCM)

The Ludic Cognition Model (LCM) requires each of the main components of the Systems Model (Section 3.1) to be analysed. Firstly, the ‘body’ and ‘world’ components (i.e. the stimulus, response, and effect, along with sensory perception), followed by the cognitive components within the body (i.e. working memory, the dynamic schema, and long-term memory). Each will then have modifications made to support the construction of a pragmatic model for application in a ludic (i.e. gameplay) context.

3.2.1: Working Memory and its Interaction with the Body and World

In the Systems Model, working memory is described primarily in terms of storage areas (i.e. the visuospatial sketchpad, the phonological loop, and the episodic buffer) (Baddeley, 1986, 1992, Baddeley and Hitch, 1974), while the processing of information is represented by the Central Executive alone. In a ludic context and specifically, in the context of the current research, it is this processing of information that is the focus, rather than the structure of working memory’s storage areas. The first stage of developing the Ludic Cognition Model from the basis of the Systems Model is thus to modify the ‘focus’ of the working memory components. The storage areas are still suggested to exist, however the Central Executive can itself be suggested to contain defined subcomponents (Figure 3.3).
Figure 3.3: Developing the Ludic Cognition Model from the Systems Model, Stage 1: Working memory and its interaction with the body and world.
Drawing on later work by Baddeley (2007), the Central Executive is suggested in this thesis to consist of two specific internal components; the Situation Evaluator and the Action Instruction. Baddeley describes early iterations of the Central Executive (Baddeley, 1986, Baddeley and Hitch, 1974) concept as resembling “a homunculus, a little man in the head who takes all the important but difficult decisions” (Baddeley, 2007, p.118), noting that while there is not yet enough evidence to disregard this view entirely, it is now possible to provide tentative suggestions of specific subcomponents of the Central Executive. These can be summarised, following Baddeley (2007, p.138), as:

1. An attention-focusing component able to direct attention to different tasks within the limited-capacity working memory system. This is a function supported by existing theories of attention (Pashler, 1998, cited in Baddeley, 2007).

2. An attention-switching component that coordinates switching back and forth between two different tasks. Baddeley notes however, that this component is likely to itself consist of multiple executive processes but that these are not yet identifiable.

3. An attention-dividing component able to coordinate attention resources between simultaneously performed tasks or simultaneously ‘attended-to’ stimuli.

4. A long-term memory interaction component that provides a link between long-term memory and working memory.

In the LCM (Figure 3.3), the first three proposed subcomponents (i.e. attention-focusing, attention-switching, and attention-dividing) are contained within the Attention Coordination section of the Situation Evaluator. Attention Coordination operates via the Attentional Loop to assess incoming stimulus information and allocate the body’s attentional resources as necessary.

The rehearsal of stimulus information within working memory (i.e. the process that prevents decay and ‘forgetting’ of short-term memories) is suggested to flow from Attention Coordination, via working memory (i.e. via the previously defined phonological loop and the visuospatial sketchpad), back to Attention Coordination. This is an intentionally simplified conceptualisation of rehearsal to enable the LCM to remain focused on ludic cognition rather than memory systems in general.

The fourth proposed subcomponent (i.e. the long-term memory interaction component) is contained within the Memory Queries & Information Encoding section of the Situation Evaluator. This is a conceptualisation of the episodic buffer in Baddeley’s model of working memory tailored towards ludic cognition specifically. It provides a mechanism for querying long-term memory for information relevant to the current stimulus information and situation, as well as a mechanism for encoding multimodal (i.e. procedural, semantic and episodic) information into long-term memory stores.

The Situation Evaluator and its subcomponents in the LCM are all situated across both procedural and declarative working memory (simplified into a single container in Figure 3.3). This is intended
to retain the interoperation of the two types of working memory as previously described in relation to the Systems Model (Section 3.1). Procedural working memory processes the contents of declarative working memory. Thus, the process of evaluating the current situation (i.e. the Situation Evaluator) necessarily spans both parts of working memory. Similarly, rehearsal requires the processing of declarative content and thus also necessarily flows through both areas of working memory.

The Action Instruction is suggested to be the trigger sent to ‘the body’ to perform the decided upon bodily action. In physiological terms, this is the sending of nerve impulses to trigger muscle movement. The LCM can be conceived as operating in cycles, with each cycle being initiated by a stimulus from the game and ending with an Action Instruction triggering a performed action. It is also possible within the LCM for an Action Instruction to lead to the performance of no action (e.g. if a player is waiting for an opponent to take action) or to the continued performance of the same action (e.g. if a player is currently making their avatar run forwards and intends to continue that action). The Action Instruction thus provides a bridge between cognitive engagement with a stimulus and performative response to a stimulus.

3.2.2: Encoding Stimulus Information for Long-Term Memory Storage

Information may be encoded, via working memory and the Situation Evaluator, to be stored in one of the long-term memory stores. Encoding involves “processing stimulus information for retention in memory” (Glassman and Hadad, 2004, p.159), while storage refers to the actual retention of information in memory. In relation to each long-term memory store, ‘encoding’ can be defined more precisely in relation to the LCM (Figure 3.4).
Figure 3.4: Developing the Ludic Cognition Model from the Systems Model, Stage 2: Long-term memory storage and encoding of stimulus information.
It is important to note the differences between how encoding may occur, as well as the differences between the initial learning of new knowledge and the later refinement of knowledge. With reference to Rumelhart and Norman’s (1976) three proposed modes of learning (accretion, tuning and restructuring) (Appendix A.2.3.3 and Appendix A.2.5), incremental refinement of existing knowledge within a schema may be conceived as the least demanding process of accretion. Greater degrees of refinement, such as modifying a schema structure to better accommodate new information, may be conceived as the next most demanding mode of learning, tuning. Substantial re-evaluation of existing knowledge or the replacement of learned knowledge with significantly different knowledge may be conceived as the most cognitively demanding mode of learning, restructuring, as this is more likely to require the construction of entirely new schemas rather than the modification of existing schemas. The structure of the LCM supports encoding of information for storage into any of the long-term memory stores using each of Rumelhart and Norman’s proposed modes of learning.

Encoding of information for storage in procedural long-term memory is described in the LCM as ‘task/process learning and refinement’; this applies both to the learning of new tasks/processes (which may be likely to require tuning or structuring) as well as the continued refinement and modification of previously learned ones (which may instead require accretion and tuning).

For semantic long-term memory, the encoding of information for storage is described in the LCM as ‘fact learning and schema/network modification’; this applies to the learning of new semantic facts and any associated modification to individual schemas, or schema network associative links required to incorporate the new factual information. Fact learning may require any of accretion, tuning, or structuring, depending on the individual player’s currently existing knowledge schemas and how relatable to that existing knowledge the newly learned information is.

Lastly, in episodic memory the encoding of information for storage is described in the LCM as the encoding of ‘ludic lived experiences’; that is, game-based lived experiences, organised according to spatiotemporal and contextual information ready to be stored as recallable ‘episodes’. For example, a player being able to recall a specific instance of fighting a particular boss in a game, as opposed to simply being able to recall semantic information about ‘the boss’ and ‘the game’.

Encoding of episodic memory may be influenced by existing schema-based knowledge. When an episodic memory is encoded, it may incorporate information that is not relevant to the currently lived experience but that may be relevant to previous similar experiences. For example, an experience of being in a specific office may be formed into an episode in episodic memory. The encoding of this episode is informed by existing knowledge of what is commonly found in a general office environment (the individual’s ‘office’ schema). Later recall of this episode recalls that ‘baked-in’, schema-based information as well, leading to possible errors such as recalling that there was a computer present in the specific office in question (because computers are commonly associated with office environments) when in fact there was not. Errors in memory such as this
have been a subject of research in psychology due to the implications for eyewitness testimony and legal proceedings (Howe, 2013, Loftus, 1981).

3.2.3: Refining the Definition of the ‘Dynamic Schema’

The concept of the ‘dynamic schema’ in the Systems Model (Figure 3.2) provided a means for describing how knowledge was recalled from memory to ‘make sense’ of a current stimulus. However, there remained some ambiguity about how knowledge was combined within that dynamic schema and how the Central Executive may process it. The final stage of developing the LCM thus focuses on a suggested refinement of the dynamic schema concept that follows from the previous modifications made to the working memory and long-term memory components.

This refinement (Figure 3.5) reflects the schema’s multimodal knowledge content (i.e. procedural, semantic, and episodic), along with its situation-dependence, or context-dependence (i.e. its content will be altered dependent on the current stimulus and the context it is encountered in). It also identifies the temporary nature of the schema (i.e. being a dynamically generated schema instance) and hence, it is termed the Multimodal Situational Schema Instantiation (MSSI).
Figure 3.5: Developing the Ludic Cognition Model from the Systems Model, Stage 3: Refining the definition of the dynamic schema.
The MSSI is created ‘as required’, in a comparable manner to that of the ‘dynamic schema’ in the Systems Model of memory and recall (Section 3.1). The MSSI’s content thus has the potential to be different in each cycle of the LCM. The MSSI can be conceived as a player’s internalised representation of a situation and thus, their individual understanding of that situation and appropriate ways to respond to it. Using Neisser’s (1976) definition of ‘schema’, the MSSI can be described as “not only the plan, but also the executor of the plan. It is a pattern of action, as well as a pattern for action”. It is the contents of the MSSI that is passed to the Situation Evaluator following a query of the long-term memory stores and therefore, expectations that an individual player may form through their existing knowledge will be passed to the Situation Evaluator also.

The MSSI therefore is not just an internalised representation of the perceived situation but rather, an internalised representation of the perceived situation along with incorporated expectations, based on how similar situations have been responded to in the past and how appropriate or successful those previous responses were. The content of the MSSI, or the player’s plan of action, in relation to the stimulus is therefore a critical consideration when considering the implementation of the disruptive game design philosophy for the disruption of player expectations.

In response to a query of long-term memory from the Situation Evaluator, the MSSI collates and provides all relevant and available multimodal information (i.e. procedural, semantic, and episodic) in an effort to make sense of, and to respond to, the current stimulus or situation. The MSSI itself is informed by the multimodal contents of all three long-term memory stores. The multimodality comes from the variable types of knowledge that may be recalled. That is, knowledge from procedural, semantic, and episodic long-term memory. The MSSI is able to combine this recalled knowledge into an internalised representation of the perceived situation, including recalled knowledge of previous comparable situations, actions previously performed in those situations, and any recalled consequences of those past actions. The Situation Evaluator can then evaluate the contents of the MSSI (e.g. past successful and unsuccessful performed actions, a comparison of semantic knowledge in the MSSI against perceived semantic information from the stimulus) and make a decision as to the specific, suitable, Action Instruction to send to the body.

It is important to note that the content of the MSSI is only as accurate as the knowledge from long-term memory that it is comprised of. In particular, as described in Section 3.2.2 (and further in Appendix A.2.4.3), information ‘baked-in’ to episodic memory at the time of an episode’s creation may be inaccurate, drawing from existing schemas (e.g. the ‘normal’ content of a typical office environment) but not being directly relevant to the episode in question. Inaccurate knowledge recalled into the MSSI will thus lead to incorrect expectations and inaccurate, or less appropriate, plans of action with which to form an Action Instruction. Thus, the disruptive game design philosophy may aim to consider not only the content of the MSSI as recalled from long-term memory, but the encoding of information into long-term memory during gameplay as well.
3.3: Ludic Knowledge Types

The LCM has been proposed following a previously described Systems Model of memory and recall and has been suggested to operate via a cyclical process of stimulus perception (i.e. from a game), evaluation, and action (i.e. performing an action that provides input back to the game). The different components of the LCM (Section 3.2) provide functionality that affords an explanation of the knowledge construction and recall process, as the flow of information and the activation of components through a single cycle, or series of cycles. The model thus provides a basis for understanding the player within *game-player space*. Furthermore, as a basis for operationalising the disruptive game design philosophy by using it as a basis for designing a game, the LCM can be used to identify potential knowledge and knowledge-based processes that may be targets for being ‘disrupted’. However, in focusing the LCM on ludic cognition rather than general human cognition, it is necessary to consider what types of game-related knowledge a player may be reasonably suggested to construct or to recall from long-term memory. While procedural, semantic, and episodic knowledge provide an initial categorisation, the question remains of how a player may differentiate, for example, knowledge of one particular game from knowledge of another particular game and how that different knowledge may be recalled and used during gameplay.

3.3.1: Encoding Specificity

The encoding specificity principle (Tulving and Thomson, 1973) provides a means for conceptualising different ludic knowledge types. It suggests that “specific encoding operations performed on what is perceived determine what is stored, and what is stored determines what retrieval cues are effective in providing access to what is stored” (Tulving and Thomson, 1973, p.369). That is, the conditions under which information is initially perceived and stored is an important factor in the ability to recall that information later on. This effectively creates stored information that has a level of context dependency. This context-dependency is in line with the organisation of episodic memory (Section 3.2.4.3) which is based on the spatiotemporal context of stored knowledge. As Smith (1994) further explains,

> Context-dependent memory implies that when events are represented in memory, contextual information is stored along with memory targets; the context can therefore cue memories containing that contextual information. There are many different operational definitions of context. ‘Context’ refers to that which surrounds a target, whether the surrounding is spatial, temporal or meaningful in nature. (Smith, 1994, p.168)

Recalling information in a similar environment or situation to when a memory was encoded, results in improved recall. Psychological studies support this for meaningful context (Godden and Baddeley, 1975), spatial context (Smith, 1986), and sensory context (Smith, Standing and Man, 1992); olfactory in that particular instance. It should be noted that the effect of encoding specificity is not strictly causal (i.e. the effect is not guaranteed to occur in all instances of encoding and retrieval), but can instead be described as correlational (Nairne, 2002). The
correlational relationship though is still a generally positive one (Nairne, 2002, p.389) with recall improving when in a contextually similar environment or situation to when encoding occurred.

Based on the evidence supporting context dependency in information recall, the suggestion can be made that when interpreting a new stimulus, the most apparently contextually similar schemas will be activated and compared in an attempt to identify matching schematic knowledge. This suggestion is also supported by spreading activation theory (Collins and Loftus, 1975) (Appendix A.2.3.2), as contextually similar schemas will likely be closer in the memory network and thus, be activated more rapidly than those that are less contextually similar.

While Tulving and Thomson (1973) originally suggested that encoding specificity only applied to episodic memory, later work extended it to semantic memory (Eysenck, 1990, Snodgrass, 2014, originally published 1989). Moreover, procedural memory is inherently context-dependent, as processes and actions are always performed in relation to something else (Appendix A.2.4.1). Therefore, an element of context-dependency and thus, the influence of encoding specificity, can be suggested to be present in relation to all three identified memory types.

Encoding specificity provides a basis to suggest that knowledge of ludic experiences can be subdivided, based on their context, into different knowledge types. The most useful way to conceive these knowledge types within the context of this thesis, is as them being dependent on how closely contextually linked to a particular ludic experience they are. For example, some knowledge may be contextualised as being relevant to a particular, specific, ludic experience (e.g. a particular game). Some knowledge may be contextually relevant to multiple ludic experiences (e.g. to multiple games). Other knowledge may not be contextually relevant to a game-mediated experience at all. These knowledge types may therefore be termed *intraludic* knowledge, *transludic* knowledge and *extraludic* knowledge and collectively referred to as *ludic knowledge types*. Note that other ways of contextualising ludic knowledge may also exist. For example, a player may respond to a visual stimulus by recalling the most contextually relevant visual-based knowledge from memory. In another case, a player may respond to a stimulus that is coloured blue by recalling the most contextually relevant knowledge about other blue stimuli. These other types of context-dependency are beyond the scope of this thesis however.

### 3.3.2: Intraludic Knowledge

Intraludic knowledge (*within-game*) consists of knowledge that is directly relevant to, and has been constructed within, the context of the *play of the current game* (e.g. the meanings and properties of particular character statistics, such as attack power or movement speed, in a particular role-playing game). This knowledge can only be constructed through a player’s interaction with a game.

### 3.3.3: Transludic Knowledge

Transludic knowledge (*across-games*) consists of knowledge related to, and constructed within, the *play of other games*. This could be knowledge from *multiple* other games (e.g. the meanings and properties of particular character statistics in other role-playing games) or it could be more
abstracted knowledge that is still inherently ‘ludic’ in nature (e.g. knowledge that role-playing
game characters generally have their abilities and skills defined through a series of statistics). This
knowledge can only be constructed through a player’s interaction with these games.

Transludic knowledge also includes knowledge from specific games (e.g. the meanings and
properties of particular character statistics in a specific other role-playing game). It is noted that
this specific knowledge may differ to the more general knowledge examples above in how it is
used by players. Evidence would be required to substantiate an explicit definitional split between
the two knowledge types however (Section 7.7.3 and Section 8.10.4.2).

3.2.4: Extraludic Knowledge
Extraludic knowledge (outside-games) consists of the remainder of an individual’s knowledge,constructed outside of a play context. This knowledge may still be recalled during gameplay but was
not itself encoded during play. For example, general knowledge regarding the semantic meaning of
the word ‘statistic’, out of the context of any one particular object (game, or otherwise), is
extraludic knowledge but must be recalled to understand ‘statistics’ in a role-playing game.

3.2.5: The Game as Expected and Ludic Knowledge Types
The game as expected is formed through a number of different sources of information both from
within the player themselves and from sources in ‘the world’.

Figure 3.6: Section of the game space model with ludic knowledge types added to the factors that
influence the game as expected.

These sources are formed of knowledge that can be categorised as belonging to different ludic
knowledge types (Figure 3.6) depending on the context in which the information is likely to be
initially encoded. Knowledge of the game product, marketing materials, and elements of broader
games ‘culture’, is all suggested to be extraludic in nature as it is likely to be experienced outside of
active gameplay. There are likely to be some cases in which some features of the game product may be occasionally interacted with during gameplay (e.g. the instruction manual), although this is unlikely to be frequently or consistently occurring player behaviour. Thus, pragmatically, these factors from world space that influence the game as expected can be suggested to be extraludic factors.

A player’s prior experiences however may consist of a varied range of intraludic, transludic and extraludic knowledge. Players who have played other games will have a collection of transludic knowledge that may, when combined with other information about a game from the world space factors, have an effect on the game as expected. Any other life experiences outside of playing games will form a collection of extraludic knowledge that players may also draw upon and that may influence the game as expected. While a player’s prior experiences may also consist of intraludic knowledge (e.g. if a player is playing a game again after having played it previously), this will not have an effect on the player’s initial construction of the game as expected.

3.4: Query and Recall of Knowledge Types in a Gameplay Session

During play, the perception of stimuli output by the game causes the Situation Evaluator to query long-term memory for relevant information to be used in understanding and responding to the stimuli. ‘Relevance’ can be determined through the context-dependency caused by encoding specificity. The most contextually relevant knowledge is the most likely to be recalled and used as a basis for comparison to the incoming stimulus information (i.e. via the influence of encoding specificity (Section 3.3.1) and memory network proximity).

Based on ‘relevance’ of information as determined through context-dependency and encoding specificity, it can be suggested that a query of long-term memory first attempts to identify intraludic knowledge directly related to the game from which the stimulus is originating. If no suitable schemas are identified, the query expands out to transludic schemas abstracted from previous ludic experiences with other games. If this also fails to identify relevant existing knowledge, then the most seemingly relevant extraludic knowledge is recalled in an effort to understand the incoming stimulus at some level. Thus, even if an individual were to encounter a stimulus that was entirely new and for which no existing intraludic or transludic schemas were appropriate, extraludic knowledge would eventually be identified and used to provide at least a basic understanding of the stimulus from which further knowledge construction could occur. For example, at a very basic level, a stimulus could be interpreted as being an entity or a phenomenon of some sort (e.g. interpreting the stimulus’ basic sensory properties). It is therefore highly unlikely that an individual will encounter a stimulus that they are completely unable to interpret.

The order of recall from each of intraludic, transludic and extraludic would be difficult to meaningfully measure. Pragmatically, encoding specificity and memory network proximity provide a basis to suggest that the player first attempts to identify existing intraludic knowledge, followed by transludic knowledge and then extraludic knowledge. Even if no existing knowledge is available...
in any one of these knowledge types, a check must still be made to identify that fact. The absence of intraludic knowledge appropriate for a query for example would return a 'null' response, before the query continued to transludic and extraludic knowledge.

<table>
<thead>
<tr>
<th>Knowledge Type</th>
<th>Intraludic (in game being played)</th>
<th>Transludic (in other games)</th>
<th>Extraludic (outside of games)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Procedural</td>
<td>INT-P</td>
<td>TRANS-P</td>
<td>Ex-P</td>
</tr>
<tr>
<td>Semantic</td>
<td>INT-S</td>
<td>TRANS-S</td>
<td>Ex-S</td>
</tr>
<tr>
<td>Episodic</td>
<td>INT-E</td>
<td>TRANS-E</td>
<td>Ex-E</td>
</tr>
</tbody>
</table>

Table 3.1: Summary of ludic knowledge types with shorthand identifiers.

The three knowledge types (intraludic, transludic and extraludic) exist within each of the three long-term memory stores (procedural, semantic and episodic) as previously defined in Section 3.2.2. Thus, during gameplay players will always utilise a selection of nine different possible types of knowledge stored in schemas (Table 3.1). As a player progresses through a game, they construct more detailed and more accurate intraludic knowledge related to the game being played. This intraludic knowledge is then recalled first at subsequent stages of the game.

Thus, disruption may be viewed as a mechanism for limiting the applicability of this recalled intraludic knowledge, by changing how stimuli are perceived by the player or how components in a game function. This disruption thus requires players to refer once more to either transludic or extraludic knowledge as a means of interpreting game stimuli and to go through the process of reconstructing new intraludic knowledge. Disruption of transludic and extraludic knowledge may also be possible by specifically designing games that operate in a manner that does not match the player’s ludic prior experiences (i.e. disruption of transludic knowledge) or in a manner that does not match the player’s real-life prior experiences (i.e. disruption of extraludic knowledge).
<table>
<thead>
<tr>
<th>Knowledge Type</th>
<th>Intraludic (in game being played)</th>
<th>Transludic (in other games)</th>
<th>Extraludic (outside of games)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Procedural</strong> (Knowledge of how to…)</td>
<td><strong>INT-P</strong> …equip/shoot/reload the handgun and the input required to do so in this specific game.</td>
<td><strong>TRANS-P</strong> …equip/shoot/reload a handgun, or similar relatable firearm, and the input required to do so in other games.</td>
<td><strong>EX-P</strong> …equip/shoot/reload a handgun, or similar relatable firearm, and the actions required to do so in real-life.</td>
</tr>
<tr>
<td><strong>Semantic</strong> (Knowledge of Concepts and Concept Properties…)</td>
<td><strong>INT-S</strong> …(e.g. within the player’s ‘handgun’ schema, a concept instance for this specific game’s handgun with associated properties).</td>
<td><strong>TRANS-S</strong> …acquired in other games (e.g. handguns in other games may have multiple firing modes such as single shot and semi-automatic, while this specific game’s handgun does not).</td>
<td><strong>EX-S</strong> …acquired outside of games (e.g. a specific concept instance of a handgun that the player has knowledge of in real life).</td>
</tr>
<tr>
<td><strong>Episodic</strong> (Collections of knowledge…)</td>
<td><strong>INT-E</strong> …relating to specific instances of using the handgun in this specific game.</td>
<td><strong>TRANS-E</strong> …relating to specific instances of using a handgun or similar relatable firearm in other games.</td>
<td><strong>EX-E</strong> …relating to specific instances of interacting with/operating a handgun or similar relatable firearm in real-life.</td>
</tr>
</tbody>
</table>

Table 3.2: Ludic knowledge types in relation to operating a hand gun in a first-person-shooter game.

Table 3.2 provides an example of how these nine different knowledge types may apply in the specific game-based context of operating a hand gun in a first-person-shooter.
3.5: Ludic Knowledge Types within the Ludic Cognition Model

Figure 3.7: LCM modified to include the ludic knowledge types within the long-term memory stores.
Ludic knowledge types and the impact of context-dependence can be added to the LCM (Figure 3.7) without the need to modify any other components of the model. The processes of knowledge construction and knowledge recall pass through each long-term memory store from the most contextually relevant ludic knowledge type to the least contextually relevant ludic knowledge type. Long-term memory within the LCM is now described specifically as Ludic Long-term Memory (Figure 3.7) to emphasise that the knowledge being recalled is contextualised in relation to the game being played.

When knowledge is constructed during gameplay, it is constructed through the process of play and contextualised in relation to the game itself as per encoding specificity (Section 3.3.1). Thus, all knowledge constructed during gameplay can be suggested to be constructed as intraludic knowledge. Indeed, whenever knowledge is initially constructed in memory it is stored in relation to the immediate context (Section 3.3.1), although the detail and completeness of that contextualising information may differ under different circumstances (e.g. if the individual is distracted). In terms of game-based knowledge, this is intraludic. In terms of knowledge constructed during an experience in the real world, this may be referred to as intraworld. In terms of knowledge related to a particular activity, such as driving a car, this may be referred to as intradriving knowledge and so forth. Constructed knowledge will be initially stored in appropriately categorised INTRA-type schemas.

Knowledge only therefore becomes TRANS- or EX-type knowledge through the recall process. Consider for example a stimulus that is encountered during gameplay that a player recognises from another prior experience of playing a different game (e.g. a red barrel). Recognising that stimulus and thus recalling appropriate knowledge from another game (e.g. properties and uses of ‘red barrels’ in other games) converts knowledge that was intraludic (when initially constructed during play of the other game) into transludic knowledge. The recalled knowledge is now relevant to two (or indeed more than two) different games. In a further example, a stimulus encountered during gameplay is recognised by the player as similar to an experience they previously had in the real world (e.g. a virtual version of a car in a racing game that the player has real world experience of driving). The original construction of knowledge about the real world experience would have been encoded as intraworld (i.e. within the real world) knowledge. Through recall during gameplay, this knowledge is converted into extraludic (i.e. outside of games) knowledge (Figure 3.8).
Figure 3.8: Transfer of knowledge from real world analogous situations, and experiences in other game instances, to a new game instance.
It should be noted that the conversion of knowledge from one type to another does not alter the knowledge itself, rather, the context in which the knowledge is applied. It should also be noted that this same process operates the other way around. That is, knowledge constructed during gameplay may be recalled in response to a real world situation. In this scenario, intraludic knowledge (i.e. within a game) will be converted to extraworld knowledge (i.e. outside of the real world) during the recall process, because the context in which the knowledge is being recalled has changed.

Recall of knowledge in this way, from a game scenario to a real world scenario, would be frequently required in situations such as game-based training and simulations. The player constructs knowledge during gameplay that is then recalled in the real world situation that the game was simulating. The recall of knowledge in this manner is presented in diagrammatical form in Appendix B. However, this form of recall is not the focus of this thesis and thus is not discussed further here.

3.6: Implications for the Disruptive Game Design Philosophy of the Construction and Recall of Ludic Knowledge Types

With regard to possible methods of implementing disruption within a game, the consideration of the game as expected and prior experiences (as identified initially in Section 1.2.2) informed through a player’s recalled intraludic, transludic, and extraludic knowledge holds potential. Expectations stemming from intraludic knowledge are the most readily identifiable for designers. Design decisions made in regard to the game’s early stages can be tailored towards encouraging the construction of particular intraludic knowledge. Subsequent stages of the game can then be designed in ways that attempt to disrupt that likely established knowledge. For example, an enemy introduced at the beginning of a game can be presented to the player as being weak to a particular type of attack. Scenarios in which this enemy appears early in the game can be designed to emphasise this weakness (e.g. by leaving the player with no choice but to attack the enemy with this particular type of attack). Through repeated exposure to this enemy under these designed circumstances, players will likely develop intraludic knowledge regarding that enemy’s weakness. At a subsequent stage of the game, that enemy’s weakness can then be changed, or modified to only be applicable under specific circumstances, thus requiring players to construct new intraludic knowledge and disregard their existing knowledge, based on their prior in-game experiences. Intraludic-knowledge-based disruption is thus highly controllable by the designer as it relies only on a player’s experience within the specific game being played.

Expectations stemming from existing transludic knowledge may be considered in the design of any game but may be particularly relevant in the context of game series, or game sequels, as it is likely that if players have played previous games in the series, they will have established expectations based on that previous play. A designer looking to operationalise the disruptive game design
philosophy may also be able to directly disrupt such expectations in subsequent games in the series.

Disruption of player expectations stemming from transludic knowledge does not only apply in the case of game series. It is also possible for designers to consider other transludic-knowledge-based expectations a player may have. Considerations such as traditions or tropes within particular genres that players may expect, or specific recently released games that players may have engaged with, could thus make up part of a designer’s research when looking to operationalise the disruptive game design philosophy. Transludic-knowledge-based disruption is less controllable as designers are less likely to have any control over the initial construction of the transludic knowledge (i.e. when it is constructed during the play of other games). However, in the case of a game series, it may be the case that the same designer or members of the development team work on multiple sequential games. In such a scenario, it may be possible for designers to have a similar level of control over the construction (in an initial game) and later disruption (in a subsequent game) of transludic knowledge. This will not be the case in all instances, depending on how a studio organises its design and development teams, but may be possible and useful in some instances. This is important to identify as it may suggest that the disruptive game design philosophy is more flexible and thus potentially more useful when applied to a game that is a later part of a series or franchise, although this would require future research to support or refute.

Expectations stemming from extraludic knowledge are likely to be significantly more varied between players as they will have been constructed through a variety of life experiences that are inherently individual for each player. While it would be unrealistic for specific individual extraludic-knowledge-based expectations to be identified by a designer, consideration can be given to broader potential expectations that may be held by a particular type of player (see Section 6.4.2, Section 6.4.3, and Section 6.4.4 for an analysis of such broader expectations). Thus, a further part of a designer’s research when looking to operationalise the disruptive game design philosophy should be an analysis of the game’s intended target audience (an analysis that would be standard practice in most game design and development projects) and what common extraludic experiences they may share knowledge of. While it is impossible for a designer to know this information for certain, it is nevertheless possible to consider likely shared knowledge and experiences of a game’s intended audience that could be targets for disruption.

3.7: Chapter Summary

The Systems Model (Section 3.1) of cognition draws on a broad base of psychological literature to present a ‘general model’ of cognition. However, this is not tailored specifically for application in a ludic (i.e. gameplay-based) context. The Ludic Cognition Model (LCM) (Section 3.2) has been developed, using the Systems Model as a foundation, to provide this tailored model. The identification of different ludic knowledge types (Section 3.3) provides a pragmatic conceptualisation of how players may encode and recall information during gameplay and in what order players may be likely to recall knowledge of different types. The LCM thus provides a basis
for understanding the player within game-player space. As a basis for operationalising the
disruptive game design philosophy, the LCM can be used to identify potential knowledge (i.e.
intraludic, transludic, and extraludic) and knowledge-based processes (i.e. encoding and recall)
that may be targets for disruption.

The LCM provides an initial design framework that will support the operationalisation of the design
philosophy. However the LCM currently only considers the player in game-player space with only
minimal detail considered regarding the stimulus itself; the game object, or more specifically, the
game object as contained within the game as published. It is now necessary to consider how the
cognitive processes represented in the LCM interact with the physical processes required to
interact with a game object, as well as how the different components of a game object itself (e.g.
audio-visual output and game mechanics) may influence both cognitive and physical player
processes. The LCM thus requires embedding within a model of the gameplay cycle.
Chapter 4:
The Ludic Action Model and the Player’s Interaction with the Game as Published

The Ludic Cognition Model (Section 3.2) provides a model for understanding player cognition within game-player space. It enables an initial understanding of the process by which a player’s prior experiences and the game as expected are formed (i.e. via encoding of multimodal knowledge into long-term memory before and during gameplay), as well as how they are recalled in response to a stimulus (i.e. via induction into the Multimodal Situational Schema Instantiation (MSSI)).

Figure 4.1: Chapter 4 focuses on the interaction between the game as published and the game as played.
In game-player space however, the game as played requires input from the game as published (Figure 4.1). Specifically, the game object itself generates stimuli for the player to perceive. While the LCM does include a ‘stimulus from game’ and an ‘effect on game’, in the model’s current form these could simply be swapped respectively for a stimulus from anywhere else, and an effect on anything else. The LCM is a cognition-centric model that can be applied to the process of gameplay but could be equally applicable to other tasks requiring cognitive processing.

To tailor the LCM for focused application to games, it is necessary to embed the model within a defined ‘gameplay cycle’ (i.e. the process through which a game updates its current state and how these updated states are interacted with and modified by players). By embedding the LCM within a gameplay cycle, the interactivity between the game object (within the game as published) and the game as played can be described in detail and thus, a game-centric model (rather than a cognition-centric model) can be constructed. This in turn can be used as part of a design framework for operationalising the disruptive game design philosophy into instantiated disruptive game features.

4.1: The Gameplay Cycle

The idea of a ‘cyclical’ approach to gameplay is supported across a range of games research. LeBlanc (1999) for example introduced the concept of feedback loops in games as a means of designing game systems to operate in different ways, either cyclically pushing a system away from a target value (e.g. a broken thermostat that reacts to being too hot by increasing the temperature further) or operating to keep a system as close to a target value as possible (e.g. a correctly functioning thermostat that reacts to being too hot by decreasing the temperature). Many racing games, for example Mario Kart: Double Dash!! (Nintendo EAD, 2003), provide a clear ludic analogy for this through their ‘rubber-banding’ systems. To ensure races are as entertaining as possible, a number of game variables are constantly adjusted to prevent leaders getting too far ahead of the pack, whilst simultaneously preventing those in the lower positions getting too far behind the pack (Mark, 2007, 2010). This keeps all of the racers within a specified range, although the full programmatic method of implementing this includes many variables and conditions (e.g. as described in US Patent No. US7278913B2 (2003) in relation to Mario Kart: Double Dash!!).

Salen and Zimmerman (2004) utilise LeBlanc’s definition of feedback loops in their work and also use a thermostat as a means of describing positive and negative feedback loops. Feedback loops are further cited in Adams and Rollings (2007), Fullerton (2008), and Dormans (2009). Each of these suggests, through the use of feedback loops, that the cyclical nature of games exists fundamentally at the low levels of a game’s systems and mechanics.

Crawford (2003, p.262) places the cyclical nature of gameplay at a higher level, co-opting the term in his definition of interactivity in general as being “a cyclical process in which two actors alternately listen, think, and speak to each other”. However, Cook (2007) links the concept of a gameplay cycle more specifically to the construction of skill-based understanding and eventual skill mastery (Section 1.2.3.1) during play in the form of “skill atoms.”
Although the cyclical concept is used frequently, in terms of overall gameplay models, two key examples exist that can be evaluated as a suitable basis for providing this broader context for the interaction between game as played and the game object (Figure 4.2). Heaton’s (2006) Circular Model of Gameplay provides a high-level conceptual view of the interaction process, whilst Perron’s (2006) Heuristic Circle of Gameplay provides a deeper perspective on the player’s cognitive and performative processes. Each of these models contains terms and structures that require closer inspection.

4.1.1: The Circular Model of Gameplay

Heaton’s Circular Model of Gameplay is not intended to describe any one particular game format, aiming instead to be “universal [and] applicable to any game type”. This therefore necessarily simplifies the model’s structure and the depth of detail in its stages.
Figure 4.3: Heaton’s Circular Model of Gameplay embedded within game space; diagram style adapted from Heaton (2006, p.2).

However, this model nevertheless provides a clear and readily understandable representation of the general cycle of gameplay in either a digital or non-digital format and can be adapted (Figure 4.3) to fit within the defined model of game space (Section 1.2).

The cycle flows between the game (i.e. the game object or ‘stimulus’) and the player (i.e. the game as played) via inputs and outputs which, while not explicitly identified by Heaton as such, can be inferred to exist within world space (Section 1.2). Heaton’s model can then also be mapped to the structure of game space, demonstrating the cyclical interaction between game publication space and game-player space.

Heaton suggests that where other approaches to viewing the gameplay cycle as a process of making many decisions over time (such as discussed in Sylvester’s (2005) ‘decision-based’ gameplay design) are common, they are not able to provide a full explanation of gameplay. Heaton suggests that the skills applied by the player before and after a decision are also important to consider, rather than only focusing on the moments at which decisions are actually made.

A decision is in a sense nothing – something which takes up an infinitesimal amount of time. It is a change in the state of the overall intent of the player. The change is discrete and instantaneous. So it is productive to ask what happens between the observation and the decision and between the decision and the action. The answer is that the player uses skills to support this decision [and the following action]. (Heaton, 2006, p.2)
As demonstrated in the model (Figure 4.3), these ‘skills’ are evaluative (i.e. cognition-based) up to the point of the decision (i.e. the perception, analysis and evaluation of the stimulus) and then practical (i.e. motor-skills-based) following the decision in order to implement it (i.e. the ‘action performed’). Heaton suggests that generally, different games require different levels and types of skill within each of these two main categories. Fast-paced action or racing games require greater practical skill, while slower-paced strategy games or turn-based games require greater evaluative, or cognitive, skill.

Heaton provides an example analysis using this model in conjunction with a particular scenario during a race in *Burnout 3: Takedown* (Criterion Games, 2004). This application of the model highlights its main weakness. Heaton suggests that in the analysis phase, the player will assess all of the available game state information to inform them of the likely success of a particular decision. However the explanation does not consider what information players may use to assist in this analysis process. For example, Heaton suggests that one such analysis may be “whether there is a better stretch of track for action just around the corner” (Heaton, 2006, p.2). This is information that only becomes available via intraludic knowledge construction (or via extraludic knowledge recall if the track is a virtual recreation of a real-world track that the player has knowledge of) as players play the game and construct knowledge about its properties. In this case, that knowledge consists of properties of the game’s tracks and what they may be best suited for in terms of approaches to gameplay (e.g. some sections of a track will be more suited to overtaking than others, dependent on factors such as ‘track width’).

Furthermore, this model does not consider in this example the player’s analysis of their own self-assessed practical skills as a factor in the pre-decision analysis process. It is not only the ‘output’ of the game that must be analysed. A decision can only be effectively made if a player also analyses their own skill and their ability to practically implement a particular decision that they make. This may be implied through Heaton’s description of a player “assessing their likely success” (Heaton, 2006, p.2) but is not clearly identified as being related to the player’s own skills. This could equally be perceived as a player considering the likelihood of a game ‘breaking’ if they attempt to cheat, for example by cutting a corner during a race.

Once again, this self-assessment will rely on a range of existing intraludic knowledge, and potentially transludic and extraludic knowledge. A player’s assessment of their skill with a particular game (e.g. a racing game such as *Burnout 3: Takedown*) may inform them of their likely ability to succeed at an intraludic level. Their prior experience of playing other racing games may provide transludic knowledge about their ‘virtual driving’ skill at a more general level. Their extraludic knowledge of ‘driving in real life’ may then also further inform the self-assessment process. Recall during gameplay may include knowledge from all three ludic knowledge types (Section 3.3). This more detailed analysis process is not considered by Heaton. However, this may be due to the previously noted purposeful abstraction of the model so as to be applicable to multiple games and multiple game formats.
Thus, while the fundamental components of Heaton’s model appear appropriate enough to describe the process of recall during gameplay at an abstract level (i.e. not tailored towards describing a particular game, or subset of games), more specificity is required to identify where the processes of ludic knowledge construction and recall may be situated more precisely with regard to the gameplay cycle of digital games.

4.1.2: The Heuristic Circle of Gameplay

Perron (2006) proposes the significantly more detailed model of the Heuristic Circle of Gameplay. Perron’s basis for this model initially stems from cognitive psychological theory, in Neisser’s (1976, p.21) perceptual cycle (Figure 4.4).

![Figure 4.4: The perceptual cycle (Neisser, 1976, p.21).](image)

Neisser’s perceptual cycle identifies the perceived object (e.g. the game) as modifying an individual’s schemas with new perceived information (e.g. the output from the game). These schemas direct the further perceptual exploration of the available object information, via sampling of that information (e.g. by visually exploring the virtual game world displayed on the output device). Perron’s model expands each of Neisser’s perceptual cycle components and in doing so, also addresses the lack of specific detail in Heaton’s (2006) model regarding how particular components in the cycle are suggested to operate.
Furthermore, as previously demonstrated with Heaton’s model, Perron’s model can also be embedded within the defined model of game space (Figure 4.5). The individual components of Perron’s model however require further clarification.

In the Heuristic Circle of Gameplay, Perron combines the cognitive top-down processes with the behavioural bottom-up processes to form a single ‘analysis’ phase (i.e. the dark blue box in Figure 4.5). “This highlights the fundamental perceptual-cognitive activity of gameplay” (Perron, 2006, p.66). Implementation (i.e. practical) skills in the form of sensorimotor capabilities are presented in the ‘implementation’ phase (i.e. the yellow box in Figure 4.5), which is directed by the output of the ‘analysis’ phase. The implementation affects the game state, which then modifies the next ‘analysis’ phase via the game’s output.

However, rather than being a single cycle, Perron’s model in fact contains two cycles that operate simultaneously. The first of these is a cycle of potential gameplay (the cycle of black arrows in Figure 4.5), where potential game states are affected by the player’s potential for implementation,
based on their implementation skills. A game state that requires implementation skills that the
player does not possess cannot be experienced (unless the player acquires the necessary skills).
Similarly, implementation is directed by analysis and thus, the scope of potential implementable
actions the player may perform is limited by the analytical skills they possess. Lastly, a player’s
expectations of potential new game states may modify the analysis process (e.g. a player may make
decisions based on an expectation that they will lead to a desirable potential game state).

The second cycle is the cycle of current gameplay (the cycle of red arrows in Figure 4.5). The
current game state modifies what Perron calls the “schema of the state”, which can be conceived
as the previously proposed Multimodal Situational Schema Instantiation (MSSI) (Section 3.2.3).
This current game state also modifies the player’s current “perceptual exploration” which can be
conceived as the Situation Evaluator within a player’s working memory (Section 3.2.1). The MSSI
and Situation Evaluator direct the player’s response; what Perron refers to as “sensorimotor
(re)actions”. The response provides input to the game, which then generates a new current state.

Perron states that this model aims to “enter into the player’s head” (2006, p.66), or to present an
understanding of the player’s internal cognitive processes, to a greater extent than Heaton’s
model does. Indeed, in the heuristic circle the position and function of schemas within the analysis
phase are vital in enabling a player to both understand a situation in a game, as well as to then plan
an appropriate action to implement. However, while this model provides greater clarity than
Heaton’s in terms of process details, it still does not fully explain the important process of
knowledge construction and recall during gameplay. While the model expands ‘analysis’ to
demonstrate the impact of game data (bottom-up) and schema-based knowledge (top-down) on
directing implementation, it does not explain how the ‘modifies’ process between the game and
the ‘analysis’ stage actually functions. Furthermore, the more granular processes operating at a
knowledge recall level during the analysis process are not considered. Lastly, as with Heaton’s
model, the player’s self-assessment of their implementation skills does not occur in Perron’s
model either. This self-assessment may be implied once more to be contained within a generic
schema, but this is not explicitly identified within the model.

These issues can be partially addressed by modifying the cycle. The analysis process does not have
to be immediately preceded by an updated game state, nor does it need to be immediately
followed by an implementation. Moreover, during a gameplay session, modification of schemas is
possible without that modification being triggered by an immediately preceding output from the
game. A player can still make assessments and analyse and make decisions while the game is
paused, for example.
Indeed, some real-time strategy games encourage just such an approach; for example, FTL: Faster Than Light (Subset Games, 2012), where pausing the game (Figure 4.6) to evaluate a strategy for tackling a situation is often necessary to survive. The ‘cycle’ is maintained in this scenario (i.e. a game output was the initial instigator for the analysis process) but the length of time taken to complete the cycle may be extended for however long the game is paused. Eventually however, the cycle will continue with the performance of an action by the player once the game is resumed.

Analysis carried out and decisions made in this paused state may be based on recalled intraludic knowledge of the game’s previous output, but they may also be based on other recalled knowledge that is recalled as part of the analysis process (i.e. transludic or extraludic knowledge). This may be particularly true in scenarios where the analysis process is extended over a long period and the player has time to consider a wider range of possible actions. Additionally, the implementation of a process need not affect the game immediately. For example, a player that is learning a game that has particularly complex input sequences (e.g. a one-on-one fighting game, such as Tekken (Namco, 1995) or Dead or Alive (Team Ninja, 1998)), or that has input sequences that require notably different physical actions to standard mouse, keyboard, button or joystick input (e.g. arcade machines that utilise bespoke input devices, such as Star Wars: Racer Arcade (Sega AMS, 2000), motion input devices such as the WiiMote or PlayStation Move controllers, or devices such as the Leap Motion (Leap Motion Inc., 2015), the Myo armband (Thalmic Labs Inc., 2015), or the Nod ring (Nod Labs Inc., 2015)), may pause the game to practice ‘implementing’ an input sequence before attempting to implement it during actual gameplay. This could be seen as identical to a ‘practice swing’ carried out by a golf player before taking their actual shot. The implementation practice process does not affect the game state, but it may provide feedback to the player that assists in refining both their analytical and implementation skills.
Thus, a rigid cyclical model is not an appropriate fit in cases such as those described above (e.g. pausing of gameplay and the use of an implementation practice process). A further refined model is required that is able to cater for different potential game types and structures, such as the examples described.

4.2: Developing the Ludic Action Model (LAM)

Using Perron’s (2006) Heuristic Circle of Gameplay as a basis, in conjunction with the Ludic Cognition Model (LCM) (Section 3.2), a refined gameplay cycle model can be constructed. By embedding the components of the cognition-centric LCM within the structure of the game-centric Heuristic Circle model, it is possible to focus on not only the cognitive processes of the player (as the LCM, as a ‘standalone’ model, previously did) but instead, to place equal focus on the actions performed by the player and the responses of the game object to those actions. Thus, the Ludic Cognition Model (LCM) can be expanded into the Ludic Action Model (LAM), providing a gameplay-cognition-centric model.

The LAM can be conceived as consisting of a primary cycle (i.e. the game object creates output that triggers player analysis, the analysis then leads to an action being performed which in turn, generates input to update the state of the game object), with multiple secondary cycles that provide greater detail about each of the ‘stages’ of the primary cycle. The following sections describe these cycles in detail, before presenting the complete LAM.
4.2.1: The Cognitive Processing Cycle

By adapting the ‘analysis’ process of Perron’s Heuristic Circle to include the LCM’s functionality (Chapter 3), the process becomes a self-contained secondary cycle, labelled as the cognitive processing cycle (Figure 4.7). This adaptation still allows for functionality as described by Perron, but further allows the model to describe more complex gameplay scenarios beyond simple input/output loops.

The stimulus from the game is perceived and passed into the Central Executive’s attention coordination system. The Central Executive, via the Situation Evaluator, may then query the contents of long-term memory and the recalled knowledge that is identified as ‘contextually relevant’ to the stimulus is passed into the Multimodal Situational Schema Instantiation (MSSI). This recalled knowledge is likely to be influenced by the player’s game as expected and their prior experiences.

The MSSI can then be evaluated by the Situation Evaluator and used to instigate an Action Instruction. This cycle combines both the querying of, and recall from, long-term memory, along
with the process of encoding new knowledge into long-term memory. It also emphasises the
different types of schema-based content available from long-term memory (procedural, semantic
and episodic in any of intraludic, transludic and extraludic types). However, the original
components of Perron’s Heuristic Circle must be further adapted to take into account the new
context of game space and the broader world space (Section 1.2).

4.2.2: Performative Processes and Performed (Re)actions

![Diagram](image)

**Figure 4.8: Developing the Ludic Action Model from the Heuristic Circle of Gameplay, Stage 2:
Performative Processes and Performed (Re)actions.**

The implementation phase of Perron’s model is expanded to consider the impact of a player’s
performance of actions on both their future implementation ability and on their future analysis of
incoming stimuli (Figure 4.8). This process is thus redefined as *performative processes* to more
clearly describe the identified separation between cognitive and performative challenges (Section
1.2.3.1) and thus, cognitive and performative processes. Performing an action during gameplay
provides immediate feedback, in the form of a stimulus that is perceived via world space, which can
be stored in long-term intraludic procedural (INT-P) (Section 3.2.5) memory and then recalled and
used in future decision making. *An action need not ‘go via the game object’ to do this.*
Figure 4.9: Super Hang-On (Sega AM2, 1987) arcade machine flyer (The Arcade Flyer Archive, 2009).

For example, the physical process of leaning to steer a motorcycle arcade machine, such as that used with Super Hang-On (Sega AM2, 1987) (Figure 4.9), can be carried out and learned from, via feedback, without the game software itself providing outputs in response to player actions. The player can, for example, choose to sit on the bike controller and lean left and right to gain an understanding of the force needed to move the controller itself. This can be done without the game object being interacted with at all. The feedback is perceived via transmission through world space (in this case, the feedback is perceived through the effects of gravity on the body and the bike controller). This process provides extraludic procedural (EX-P) knowledge that can be recalled when the player is using the input device to directly interact with the game object. This feedback process is termed perception of action ‘in the world’ and represents another secondary cycle within the model.

The practice cycle that loops out of the performed re(action) and back into performative processes, via world space, highlights the improvement of performative skills that may stem from motor repetition of an action or sequence of actions. As noted in Lee, Swanson and Hall (1991), whilst repetition of actions is not the only mechanism by which performance of those actions may be improved, it is nevertheless an important factor. The perception of action ‘in the world’, combined with the practice cycle, describe the two key mechanisms involved in the refinement and improvement of performative processes.
4.2.3: Stimuli from World Space and the Player’s Effect on World Space

The world space in which the model is embedded should not be viewed as a passive container in which the game as played occurs. A game does not occur entirely within game space with no impact on the world around it. A player exists within the world whilst they play and thus, both incoming stimulus information from this world space (i.e. extraludic stimuli) along with the effect of the player’s performed actions on the world space must be considered as part of an understanding of the game as played (Figure 4.10).

This consideration is important in the discussion of player knowledge construction and recall as extraludic (e.g. from world space) stimuli may influence the cognitive processing of game-based (intraludic) stimuli. For example, distractions within the real world environment may impact a player’s ability either to perceive game-based stimuli accurately or to provide sufficient attention to them to evaluate them fully. Similarly, a player’s performed actions have observable impacts in world space (e.g. a physical input device is manipulated by the player), as well as providing input back into the game object.

Figure 4.10: Developing the Ludic Action Model from the Heuristic Circle of Gameplay, Stage 3: Stimuli from world space and the player’s effect on world space.
As described in the model of *game space* and specifically in relation to the influences on the player from *world space* (Section 1.2.2), factors such as a game’s product materials (e.g. the manual and the cover artwork), the marketing materials (e.g. trailers, posters, or demos), and the game ‘culture’ that may exist around a title (e.g. online discussion amongst fans, or critical reviews in publications) all influence the game as expected. While the Ludic Action Model is describing the processes and actions of a player *during play*, it would be an oversight to not include the potential influences on the player from these *world space* factors in the model also.

### 4.2.4: Modification of Game Object Properties

![Diagram](image)

*Figure 4.11: Developing the Ludic Action Model from the Heuristic Circle of Gameplay, Stage 4: Modification of game object properties.*

The properties of the *game object* from Perron’s Heuristic Circle can also be examined. "Image and Sound", as the only properties of the game in Perron’s model (Figure 4.5), is restrictive and does not take into consideration other perceivable game outputs, such as haptic feedback (e.g. rumble/vibration features on controllers or force feedback on steering wheel or joystick devices). Thus, this is expanded in the LAM to include all aspects of the *perceivable game* (Figure 4.11), which can also encompass any future technological developments that enable other sensory output types beyond auditory, visual, and haptic output.
Following this, the ‘modifies’ process between the game and the cognitive processing cycle is replaced with the ‘provision of current and potential intraludic stimuli’. This more accurately describes this process as the game object does not directly modify the player’s cognitive processing. The game no more directly modifies an individual’s cognitive processes than the real world does. A stimulus (from either the game or the real world) merely provides an instigator to cognitive processing by the player, which may then result in the player modifying their own knowledge (i.e. via accretion, tuning, or structuring of knowledge within schemas (Section 3.1 and Appendix A.2.3.3)).

The game object also requires an additional programmatic update cycle (Figure 4.11). This is defined as the ongoing real-time updating of the game object (i.e. the continuous execution of game code), which allows the game to generate new states without the need for a player to provide input. Programmatic game state updates may include complex processing of artificial intelligence systems that control the movement and behaviours of non-player characters in the game, through to simple automated updates such as counting down an in-game timer.

4.2.5: The Ludic Action Model and the Primary Cycle of Current Gameplay

Figure 4.12: The complete Ludic Action Model.
The complete LAM (Figure 4.12) retains the previously described concepts of the cycle of current gameplay and the cycle of potential gameplay (Section 4.1.2), represented by the red arrows and black arrows respectively. However, the cycle of potential gameplay is retained for completeness only and does not form part of the game as played in the moment by the player. Thus, the primary cycle within the LAM consists of the cycle of current gameplay (i.e. the red arrows only) as a means of understanding the player experience of the game as played.

In most cases, this current gameplay cycle does indeed operate in a manner similar to that proposed in Perron’s original work, moving consistently from game object, to cognitive processing cycle, to performative actions, and lastly back to the game object. However, for a range of different possible gameplay scenarios, all of which may contribute to the player’s current experience of the game as played, the LAM is able to provide specific detail with regard to the lower level processes and components involved, with particular attention given to the subcomponents of the cognitive processing cycle. These subcomponents are of particular importance for the further exploration of the disruptive game design philosophy and the application of the philosophy to game design practice.

4.3: Modes of Disruption, Cognitive Process Targets, and Cognitive Engagement

The LAM provides a pragmatic framework for operationalising the disruptive game design philosophy into disruptive game features through its identification of specific processes, cycles and interactions during gameplay. Thus, consideration can now be given to the different potential methods of disrupting knowledge and processes within the LAM that may be available to game designers (during the creation of the game as designed and the game as created) and how these different methods may be able to support a player’s cognitive engagement (Section 1.3.1) during gameplay.
Figure 4.13: Specific cognitive processes within the player that may be able to be disrupted.

The disruption of a player’s existing knowledge and any expectations that are based on that knowledge requires the disruption of processes within the player, within the cognitive processing cycle of the LAM. While the game designer will need to manipulate a component of the game object to implement disruption of any kind, it is the player’s cognitive processes that are being disrupted and thus, any definitions of different methods of disruption should be based in these processes.

From this perspective and using the LAM as a basis, it is possible to initially suggest and define three different modes of disruption that may occur during gameplay, based on the individual cognitive processes that combine to form the cognitive processing cycle (Figure 4.13). These processes are encoding of information when it is first perceived, recall of knowledge from long-term memory, and the plan of action that is contained in the MSSI and that will lead to a performed action. Thus, the suggested modes of disruption are termed Encoding Disruption, Recall Disruption, and Action Plan Disruption.
4.3.1: Encoding Disruption

Encoding Disruption is suggested to operate on the initial encoding of information during gameplay and thus, only operate in relation to the player’s construction of intraludic knowledge (in any of procedural, semantic or episodic forms). Encoding Disruption may be possible to operationalise via two mechanisms; firstly, by providing players with incomplete or ambiguous stimulus information (making ‘comprehensive’ encoding of stimulus information impossible), or secondly, purposely presenting players with distracting additional stimuli (increasing the probability of inaccurate or incomplete encoding of information).

An ambiguous or ‘incomplete’ stimulus may, for example, be a previously unheard and unrecognisable sound without an obvious source. Such a stimulus may be encoded initially using existing knowledge as a guide, as well as any contextual information available at the time of encoding (e.g. a new sound reminds the player of a sound that they do know the source of and thus, they associate this unknown sound with potentially coming from that type of source). This may result in inaccurate ‘baked-in’ information being stored that is based-on previous experience of a similar stimulus but that is not directly relevant to the particular stimulus being perceived (Section 3.2.2 and Appendix A.2.4.3). Thus, purposeful presentation of ambiguous stimuli may increase the likelihood of Encoding Disruption.

Encoding Disruption may also occur in situations where a player is not able to attend to a stimulus due to other competing stimuli (i.e. distractions). In the LAM, the Central Executive can be seen to be receiving both intraludic and extraludic stimuli that it must attend to via its Attention Coordination system. The extraludic stimuli may cause distractions that disrupt the accurate encoding of the intraludic stimuli. More importantly from the perspective of a game designer, multiple intraludic stimuli being perceived simultaneously (e.g. multiple sources of auditory and visual stimuli, such as in a battle sequence in a game) must be prioritised by the Central Executive. This may result in some stimuli being partially encoded or indeed ‘missed’ entirely.

Distracting stimuli may also be presented a short time (i.e. one second, as demonstrated in McNab and Dolan (2014)) after the stimuli to be encoded and still have an impact on the encoding process. Thus, game designers may be able to similarly utilise these two types of distraction as a means of implementing Encoding Disruption.

4.3.1.1: Encoding Disruption’s Impact on Cognitive Engagement

Encoding Disruption can be mapped to Bloom’s Revised Taxonomy (Anderson and Krathwohl, 2001, Krathwohl, 2002) (as described as the basis for defining cognitive engagement in Section 1.3.1) to demonstrate its impact on a player’s cognitive processes and thus, degree of cognitive engagement (Figure 4.14).
Encoding Disruption via both the mechanisms described (Section 4.3.1) is suggested to disrupt player cognition at the lowest level, of remembering. The use of ambiguous stimuli in a game increases the likelihood of incorrect information being encoded as knowledge. Later recall of that incorrect information (i.e. remembering) is thus disruptive, as the information does not serve the intended purpose that it was recalled for. What may appear to be a recall error (i.e. incorrect recall of correct knowledge) may instead be the result of an encoding error at the time the knowledge was acquired (i.e. correct recall of incorrect knowledge). Purposeful distractions similarly make accurate encoding more challenging and lead to the same potential disruptive impact on remembering. The impact of Encoding Disruption on remembering requires the player to perform higher-order analysis to identify the inaccuracies in their existing knowledge. Thus the impact of Encoding Disruption on cognitive engagement is to encourage players to move from the lowest level of lower-order thinking, to the lowest level of higher-order thinking.

4.3.2: Recall Disruption
Recall Disruption is suggested to operate on information recalled from long-term memory regarding the perception of a game stimulus’ properties and presentational context. This recalled information may be intraludic, transludic or extraludic in nature and may be in any of procedural, semantic or episodic forms.

If a stimulus is presented to a player that appears identical to a previously encountered stimulus (e.g. the player encounters an enemy early in the game and then encounters an enemy that appears identical later in the game), seemingly understood knowledge stored in long-term memory and related to those previous encounters, will be recalled. If the stimulus then does not behave in a way that this previous experience suggests (e.g. the enemy uses different tactics or different attacks), this will require the player to re-evaluate their understanding of their existing knowledge (i.e. the current understanding of that enemy’s tactical and offensive capabilities). Players will need to construct new knowledge that explains the new stimulus properties (i.e. that
enemy can use these different tactics and attacks as well as the ones previously encountered, or perhaps now utilises these tactics and attacks instead of the ones previously encountered).

Thus, the purposeful presentation of previously encountered stimuli that behave differently or that have otherwise different properties, increases the likelihood of recalling information that is expected to be accurate but in fact is not, resulting in this Recall Disruption. This type of disruption could furthermore be repeated at multiple points throughout a game’s play time, focusing on different game stimuli in different ways.

**4.3.2.1: Recall Disruption’s Impact on Cognitive Engagement**

Recall Disruption can also be mapped to Bloom’s Revised Taxonomy (Figure 4.15).

![Figure 4.15: Impact of Recall Disruption on cognitive engagement.](image)

Recall Disruption disrupts a player’s understanding of information. The information that is recalled is seemingly already understood by the player, based on the properties of the stimulus that triggered the recall process. However, making changes to the behaviour or properties of a familiar stimulus makes the recalled information less accurate and thus, less useful as a means of understanding the stimulus. In response, the player may need to engage in both higher-order analysis as well as evaluation of the recalled information in an effort to identify the incorrect understanding of it and how that may have been caused. Furthermore, the player may need to engage in analysis and evaluation of the stimulus itself once more in an effort to construct a more accurate understanding of it that can then be encoded as new knowledge.

Recall Disruption’s impact on cognitive engagement is thus to encourage players to move from the mid-level of lower-order thinking to the lower and mid-levels of higher-order thinking.

**4.3.3: Action Plan Disruption**

Action Plan Disruption operates on the MSSI within the cognitive processing cycle of the LAM. Specifically, it operates on the in-game consequences of MSSI-driven player inputs. As described in Section 3.2.3, the MSSI within the LAM is both a pattern of, and a pattern for, action (Neisser,
The knowledge that is recalled from the three long-term memory stores (i.e. procedural, semantic and episodic in any of intraludic, transludic or extraludic forms) is passed into the MSSI, which is the action plan for deciding upon and performing a response (a performed (re)action) to a perceived stimulus. If encoding has not been disrupted (i.e. the original experience of the stimulus was not ambiguous or situated alongside distractions) and recall has not been disrupted (i.e. the following experience of the stimulus does not suggest it has changed its behaviour or properties), then the MSSI should in theory contain an appropriate action plan to direct the action(s) that a player will perform in response to the game stimulus.

If the in-game consequence of an MSSI-based Action Decision then does not correspond to previously experienced in-game consequences of that same action in the same context, this requires players to re-evaluate their understanding of the stimulus. It may also require them to re-evaluate their understanding of the action decided upon, which poses multiple questions for the player. These may include whether they (1) misinterpreted the stimulus itself, whether they (2) correctly interpreted the stimulus but then decided upon an incorrect action, or whether they (3) correctly interpreted the stimulus and decided upon and performed the correct action but, their performance itself was poorly implemented. For example, the player encounters an enemy early in the game and attacks and defeats that enemy successfully. The player later encounters an enemy that is identical and thus, the player utilises the tactics previously employed to defeat it. This, time, the player’s attacks do significantly less damage to the enemy. The player must then work out why this may be the case before adjusting their tactics accordingly.

Purposeful implementation of stimuli during gameplay that cause this Action Plan Disruption is a potentially effective method of encouraging higher levels of cognitive engagement over time, as it poses a number of questions that players may consider relating to their interpretation of a stimulus, their understanding of the stimulus and their knowledge related to it, and their ability to perform appropriate actions in response to it.

However, in terms of the knowledge that is actually disrupted, this form of disruption targets only semantic knowledge, in any of intraludic, transludic or extraludic forms. This is because to modify the consequences of an action, a property (or multiple properties) of a game entity must be changed (e.g. the enemy’s weapon resistance in the above example). Properties of objects exist in a player’s semantic knowledge only (i.e. they can be conceived as ‘factual information’ about the game entity). While this type of disruption may be a trigger for the player to consider a number of different questions such as those described, it may be implemented through a comparatively simple modification of a game entity.
4.3.3.1: Action Plan Disruption’s Impact on Cognitive Engagement

Action Plan Disruption can lastly also be mapped to Bloom’s Revised Taxonomy (Figure 4.16).

Figure 4.16: Impact of Action Plan Disruption on cognitive engagement.

Action Plan Disruption disrupts the *application* of seemingly understood information. The consequences of an applied action do not match the expected consequences of that action and thus, players will need to engage in *analysis* and *evaluation* of their understanding of the situation before *creating* a new strategy or approach with which to tackle it. Action Plan Disruption thus encourages players to move from the top level of lower-order thinking through each of the levels of higher-order thinking.
4.4: Modes of Disruption and their Knowledge Type Targets

Not all modes of disruption (i.e. Encoding Disruption, Recall Disruption, and Action Plan Disruption) are able to target all types of knowledge (i.e. INT-P, INT-S, INT-E, TRANS-P, TRANS-S, TRANS-E, EX-P, EX-S, EX-E) (Section 3.3).

<table>
<thead>
<tr>
<th></th>
<th>Encoding Disruption</th>
<th>Recall Disruption</th>
<th>Action Plan Disruption</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Intraludic</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Procedural</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Semantic</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Episodic</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td><strong>Transludic</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Procedural</td>
<td>X</td>
<td>✓</td>
<td>X</td>
</tr>
<tr>
<td>Semantic</td>
<td>X</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Episodic</td>
<td>X</td>
<td>✓</td>
<td>X</td>
</tr>
<tr>
<td><strong>Extraludic</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Procedural</td>
<td>X</td>
<td>✓</td>
<td>X</td>
</tr>
<tr>
<td>Semantic</td>
<td>X</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Episodic</td>
<td>X</td>
<td>✓</td>
<td>X</td>
</tr>
</tbody>
</table>

Table 4.1: Summary of possible knowledge type targets of each mode of disruption.

Encoding Disruption can only apply to the construction of intraludic knowledge (Table 4.1), as a designer cannot control or disrupt a player’s encoding of knowledge outside of the game being currently designed. Similarly, as previously stated, Action Plan Disruption can target only semantic knowledge in each of the three ludic knowledge types. Lastly, a player may recall knowledge of a stimulus in any form during the play of a game and thus, the disruption of this recall process via Recall Disruption is able to potentially target any of the nine different knowledge types (Table 4.1).

While this overview may suggest that Recall Disruption provides the most potential benefit for a designer through its applicability to all nine knowledge types, the potential significant impact on cognitive engagement that Action Plan Disruption may have (i.e. encouraging players to engage with the highest levels of higher-order thinking) may result in that disruption mode having a greater disruptive impact, despite it only targeting semantic knowledge. The uses and applicability of each mode of disruption however, can be further explored through practical application in a designed and developed game artefact (Chapter 7).
4.5: The Ludic Action Model in Game Space

Applied to the game space model (Section 1.2) the LAM can be seen to incorporate a number of different levels and components of game space (Figure 4.17).

The LAM describes the functionality and interactions occurring within the game as played, along with the impact of a player’s prior experiences and their game as expected (influenced by factors from world space such as the game product, marketing and games ‘culture’). The LAM also, through its definition of the gameplay cycle, incorporates the player’s interaction with the game object, within the game as published.
4.6: Chapter Summary

Heaton’s (2006) and Perron’s (2006) cyclical models of gameplay have been analysed (Section 4.1) with a view to understanding the interaction between the game as played and the game object. Using these analysed models as a basis, in conjunction with the Ludic Cognition Model (LCM), a more comprehensive model of the game object and the game as played in the form of the Ludic Action Model (LAM) (Section 4.2) has been constructed. The LAM demonstrates how the process of knowledge construction and recall may be reasonably suggested to practically operate in the context of a cyclical gameplay process.

Moreover, within the cognitive processing cycle of the LAM, the separate processes of encoding, recall, and action planning have been identified as being able to be disrupted (i.e. via Encoding Disruption, Recall Disruption, and Action Plan Disruption) (Section 4.3). These modes of disruption are each able to support the player’s heightened degree of cognitive engagement by encouraging them to utilise higher-order thinking in situations where lower-order thinking may often suffice.

The actual uses and benefits of the different disruption modes will only become apparent however, once they have been implemented into a game (Chapter 7) and players provide feedback on their gameplay experiences (Chapter 8). As discussed in relation to the different modes of disruption and their knowledge type targets (Section 4.4), Recall Disruption may appear the most useful mode of disruption due to its wide applicability to all nine knowledge types. This mode of disruption may not be the most effective in terms of its impact on the player experience however, nor may it be the easiest for designers to design for. Likewise, feedback from players is crucial in acquiring an ecologically valid (Section 2.1) understanding of the impact of disruption.

However, to do this, a component framework for the game as created is required so that the conceptual modes of disruption can be operationalised into disruptive game features that specifically manipulate those components.
Chapter 5:
A Conceptual Framework for the Game as Created

The disruptive game design philosophy (Section 1.3.1) provides the basis for a commercial game design and development project, using the LAM (Section 4.2.5) to understand the game object as experienced in the game as played. However, the design philosophy and the LAM must be linked together, through the game as designed and the game as created, into actual player-facing disruptive game features in that game object. To design a game, it is first necessary to define the conceptual framework that structures the game as created and thus, the game object (Figure 5.1).

Figure 5.1: Chapter 5 focuses on the conceptual framework of the game as created in game development space.
5.1: The Ludodiegesis

A conceptual framework for the game as created needs to first define the different ‘layers’ of the game, from the contents of the virtual environment itself, to the supporting game systems (e.g. menus and loading screens) and potentially, any supporting non-game systems (e.g. console hardware). Differentiating these layers is important for identifying what components of a game may be able to be disrupted through the game’s design and which components may not. These different layers can be defined with reference to the concept of diegesis, or ‘fictive reality’. This is a term co-opted from narratology and specifically, applied by Pinchbeck (2009b) to refer to the ‘fictive reality’ of a game, or the ludodiegesis.

Figure 5.2: The ludodiegesis, containing the homodiegesis and heterodiegesis.

The ludodiegesis can be elaborated further, drawing on the work of Genette (1980), as containing homodiegetic components (i.e. within the fictive reality of the game) and heterodiegetic components (i.e. supporting game systems that are part of the game, but not a part of the fictive reality of the game) (Figure 5.2). Within this definition, homodiegetic components may include, for example, game characters or elements of the game environment such as towns and cities, while heterodiegetic components may include the menu systems and loading screens.

Thus, all components of a game and its supporting systems can be described as being part of the ludodiegesis. However, they may either exist as homodiegetic components or heterodiegetic components within that broader container. With this overarching structure in place, the framework for the game as created can be developed further to identify different game components that exist within the ludodiegetic, homodiegetic, and heterodiegetic structure.
5.2: The ‘Elemental Tetrad’ Structure of Game Components

A conceptual framework for the core components of games is proposed by Schell (2014, p.51), in the form of the “elemental tetrad”.

![Diagram of the Elemental Tetrad Structure of Game Components](image)

This framework consists of four components: **Technology**, **Mechanics**, **Story**, and **Aesthetics** (Figure 5.3). These components are arranged in a manner that presents each as being of equal importance (2014, p.52) and also indicates the relative ‘visibility’ of each component to the player, with underlying technology being least visible to the player and the aesthetics of the game being the most clearly visible to the player.

5.2.1: Technology

Technology is defined by Schell (2014, p.52) as any “materials and interactions that make [the] game possible such as paper and pencil, plastic chits, or high-powered lasers”. Schell’s work is not intended to specifically focus on digital games however and this is emphasised by his statement that ‘technology’ in this context need not refer exclusively to “high technology” (Schell, 2014, p.52) (i.e. electronic and computational technology). In the case of digital games however, ‘high technology’ represents the majority of the materials and interactions that enable the game.

Thus, the array of hardware that may be required to interact with a game object (e.g. central processing units (CPUs), memory (RAM), graphics processing units (GPUs), input, output, and data storage devices) must be considered to be important elements within the ‘technology’ component. The technology of the game provides the medium in which the story is told, the mechanics operate and the aesthetics are presented (Schell, 2014).
It could be suggested that supporting physical materials that are provided with a digital game are also part of the game’s technology (in keeping with Schell’s assertion that technology need not only refer to ‘high technology’). For example, materials such as physical, printed maps may be included in special editions of some games (e.g. Lunar: Silver Star Story Complete (Game Arts & Japan Art Media, 1999), which included a cloth map and a hardback instruction manual (Figure 5.4) in its limited collector’s edition). These items, where present, may be included as part of the gameplay experience and thus form part of the game’s technology.

Figure 5.5: The game’s ‘Technology’ in relation to the ludodiegesis, homodiegesis and heterodiegesis.
In the context of the ludodiegetic structure of a game, the ‘technology’ component of the conceptual framework can be conceived as existing partially within the heterodiegesis (within the ludodiegesis) and partially outside of the ludodiegesis entirely (Figure 5.5). For example, a physical, printed game map is part of the supporting system of the game, but does not exist within the virtual game environment and is thus, part of the heterodiegesis. The console hardware that runs a console game is enabling technology but exists outside of the virtual game environment and outside of the context of the individual game. Thus, the console hardware is situated entirely outside of any single game’s ludodiegesis. This also applies to most arrangements of input and output technology a player is using to interact with a game (e.g. joysticks, gamepads, a monitor, a television, speakers, headphones or haptic feedback devices).

5.2.2: Aesthetics

The aesthetics of a game are defined by Schell (2014, p.52) as “how [the] game looks, sounds, smells, tastes, and feels”. Aesthetics support the game’s story and may be limited in some ways by the game’s underlying technology (e.g. limitations on environmental complexity based on available memory and processing power). This definition requires some refinement to apply more specifically to the current research and to digital games in particular.

The properties of ‘smell’ and ‘taste’ can be removed as these are not readily utilised in commercial digital games (although there are some notable exceptions, such as the ‘scratch’n’sniff’ cards (Figure 5.6) that shipped with copies of EarthBound (Ape & HAL Laboratory, 1995)). The audio-visual properties of the game (i.e. the ‘looks’ and ‘sounds’) are the most evidently relevant to the majority of commercial digital games and thus must remain as elements of the ‘aesthetics’ definition. The majority of commercial game titles can be suggested to include a relative balance of
visual and auditory aesthetics, although once again there are some exceptions, such as the audio game *Papa Sangre* (Somethin' Else, 2013) in which the ‘aesthetic’ properties of the game are predominantly auditory, with only a very minimal graphical interface available to players.

Following the LAM (Section 4.2.5), haptic feedback from a game should also be included in a definition of a game object’s output. Haptic feedback (most commonly instantiated as ‘rumble’ functionality or ‘force feedback’) has been a standard feature in the majority of console gamepads and some PC-based gamepads for a number of years (Buchanan, 2008). Thus, a definition of a game’s aesthetics should also include these haptic properties (i.e. how the game ‘feels’), in addition to the more obvious audio-visual properties.

![Figure 5.7: Adding the game’s ‘Aesthetics’ in relation to the ludodiegesis, homodiegesis and heterodiegesis.](image)

The aesthetics of a game can be conceived as existing partially within the homodiegesis (e.g. in the case of audio-visual and haptic properties of entities within the game environment) and partially within the heterodiegesis (e.g. in the case of audio-visual and haptic properties of game systems, such as front-end menus and configuration screens) (Figure 5.7).

**5.2.3: Story**

The story is the linear sequence of events that occur throughout the run time of the game. It is necessary to clarify that ‘story’ is not being used synonymously in this thesis with ‘narrative’. Schell (2014, p.52) suggests that a game’s story “may be linear and pre-scripted, or it may be
branching and emergent”. A story, in Schell’s definition, is intentionally designed and written, then
told to the player through the linear experience of the game. Even if that linear experience differs
from another player’s linear experience (e.g. in the case of two different story branches), the
story branches themselves are experienced in a linear fashion by each individual player. While
there are many possible structures for a story in a game (see for example Miller’s (2014)
definitions of a range of linear and branching story structures), it is beyond the scope of this thesis
to provide an in depth analysis of each.

However, it is difficult to suggest that a story is explicitly told to the player in all games. For
example, in Tetris (Pajitnov, 1984) there is no explicitly told story that provides reasoning or
meaning for the blocks falling down the screen. However, this is not to say that players are unable
to construct their own interpretations of possible stories. Murray (1997, p.144) for example,
interprets the play of Tetris as representing

[. . .] a perfect enactment of the overtasked lives of Americans in the 1990s – of
the constant bombardment of tasks that demand our attention and that we must
somehow fit into our overcrowded schedules and clear off our desks in order to
make room for the next onslaught. (Murray, 1997, p.144)

These player-constructed stories could be better described alongside Schell’s definition of the
‘told story’ as player-constructed narratives. This is an important distinction to make. While not all
games may have an explicit story to tell, it is still possible for a player to construct their own
narrative while playing, even in the case of games that lack traditional ‘characters’, such as Tetris.
However, these narratives may not bare any resemblance to what the game designer(s) intended
players to experience.

It is equally possible for a player to construct their own narrative even in a game with an explicitly
told story. For example, Final Fantasy VIII (Square, 1999) is a story-driven role-playing game with
an intricate but explicitly told story. However, different player interpretations (i.e. constructed
narratives) can be found in online discussion forums. In particular, the ‘Squall’s Dead’
interpretation provides a full website (Choudhury and Rater, 2011) with an analysis of various
symbols and themes in the game that suggest that the main character (i.e. Squall Leonheart) dies
at the end of the game’s first section (i.e. Disk 1 of 4) and that everything in the following 3
sections is a dream experienced as he dies.

This ability to construct narratives from simple cues is similarly demonstrated in psychological
research, such as the work of Heider and Simmel (1944) in which participants were found to
assign behavioural and personality traits to abstract, animated geometrical shapes and to further
interpret a narrative from the animation’s sequence of events.
Figure 5.8: Adding the game’s ‘Told Story’ and ‘Constructed Narrative’ in relation to the ludodiegesis, homodiegesis and heterodiegesis.

The ‘story’ component of the tetrad is thus suggested to include both told stories and constructed narratives. In terms of the ludodiegesis, the positioning of told stories and constructed narratives differs (Figure 5.8). A told story, delivered as an intentionally designed component of the game, exists wholly within the homodiegesis (note that the homodiegetic told story may still be delivered by a heterodiegetic narrator). A constructed narrative that is created by the player, remains within the ludodiegesis of the game but can instead be conceived as existing within the heterodiegesis. A constructed narrative is part of the player’s understanding of the game and is thus a supporting system of the game. A constructed narrative is constructed using information from homodiegetic cues (e.g. audiovisual information from the game, or text-based story delivery from the game). However, the constructed narrative itself is not homodiegetic (as it may be wholly or partially different to the designed story).

5.2.4: Mechanics

In the context of the elemental tetrad of game components, a definition of ‘mechanics’ is provided by Schell (2014, p.41), who states that they are “the procedures and rules of [the] game. Mechanics describe the goal of [the] game, how players can and cannot try to achieve it, and what happens when they try”. This definition is problematic as it implies that anything within the game that is not linked to the goal of the game is not a mechanic. It also suggests that mechanics are
specifically player focused; what the player can and cannot do. Lastly, it also suggests that mechanics and rules are synonymous; that mechanics are rules.

Sicart (2008, para.6) provides an alternative definition of mechanics, suggesting that they represent “methods invoked by agents, designed for interacting with the game state”. This definition draws on terminology and concepts from the object oriented programming paradigm (i.e. methods and agents). Methods are further defined as the actions available to a game entity for interacting with other entities and the game world, as defined within the game’s rules. Agents are further defined as being either human players, game-controlled ‘players’, or game systems. These game systems are also defined by the game’s rules.

Figure 5.9: Removing enemy agents from the game world and simplifying the game state in Bioshock Infinite (Irrational Games, 2013).

Thus, the definition of mechanics may be reworded as “actions invoked by players or rule-based systems, designed for interacting with the game state”. This interaction with the game state may be clearly apparent. For example, the ‘shooting’ mechanic in Bioshock Infinite (Irrational Games, 2013) causes visual and auditory feedback along with removing enemy agents from the game world, simplifying the game state as the game progresses (Pinchbeck, 2007, p.10) by eliminating dangers (Figure 5.9).
Figure 5.10: Choosing between two brooch designs for Elizabeth to wear in Bioshock Infinite.

However, interaction with the game state may also be much less clearly apparent, or may be much less significant. For example, Bioshock Infinite includes a number of points throughout the game in which players are required to make a decision (Figure 5.10). These choices only interact with the game state in minor ways; in the example above, the player’s decision simply changes the mesh that is used to represent Elizabeth’s brooch throughout the remainder of the game. However, the choice requires an action to be invoked by the player that is designed to interact with the game state and thus can be considered a mechanic.

Sicart’s definition avoids the issues that Schell’s definition presents. Firstly, mechanics are not synonymous with rules as is implied by Schell. The mechanics are invoked by, and constrained by, the rules. However, the rules themselves are fixed and thus not influenced in turn by the mechanics. Rules are fixed while mechanics may be dynamic interactions between rules, the output of which may differ in different contexts.

Secondly, Sicart’s definition does not require a mechanic to be ‘goal-oriented’. The game may indeed have designed ‘goals’ for players to strive towards. However, players may equally create their own personal goals, the meeting of which the game has no method of measuring. If something in the game assists in the meeting of one of these player-constructed goals, it is not clear via Schell’s definition as to whether it is a mechanic or not.
Lastly, the emphasis on actions invoked by any agent, human or otherwise, removes the link that Schell suggests as being necessary between the player and a mechanic, which may be necessary in non-digital games but less critical in a digital game containing agents operated entirely via artificial intelligence processing. This is evident in the LAM as the *programmatic update loop*. A digital game mechanic may be invoked by and used exclusively by a game-controlled ‘player’ (e.g. an artificial intelligence system itself, or a specific attack that is only available to an enemy character) but may still be defined as a mechanic, because it interacts with the game state and is invoked by an agent. Sicart’s definition of mechanics provides a definition that is theoretically narrower in application than Schell’s (applying specifically to digital games rather than all games) but that is deeper in terms of validity within the specific context of digital games. This is therefore the definition that will be used to describe ‘mechanics’ within the conceptual framework of the *game as created*.

![Figure 5.11: Adding the game’s ‘Mechanics’ in relation to the ludodiegesis, homodiegesis and heterodiegesis, forming a refined conceptual framework of the game as designed.](image)

In the same way that a game’s aesthetics exist in both the homodiegesis and the heterodiegesis, a game’s mechanics exist in both as well (Figure 5.11). Homodiegetic mechanics are invoked by an agent and affect a homodiegetic system or entity (e.g. shooting a gun to defeat an enemy in the game, or making a choice in *BioShock Infinite* that has a minor impact on an in-game entity).
Heterodiegetic mechanics are invoked by an agent but affect a heterodiegetic system, such as Psycho Mantis’ ability to read the contents of the player’s real-world memory card (via the game’s save/load system; a heterodiegetic system that supports the game) in *Metal Gear Solid* (Konami Computer Entertainment Japan, 1998) (Figure 5.12). This thus ‘breaks the fourth wall’ by speaking to the player themselves rather than the player’s in-game avatar.

The structure of the ludodiegesis, homodiegesis and heterodiegesis (Figure 5.11) now forms a conceptual framework of the *game as created*. This framework is based on Schell’s ‘elemental tetrad’ of game design (2014, p.42) but utilises refined definitions of key terms, such as mechanics, to more clearly bound what can and cannot be considered to be situated within different components of the framework.

### 5.3: Mapping the Conceptual Framework of the Game as Created to the Ludic Action Model

The conceptual framework (Figure 5.11) can now be mapped to the Ludic Action Model (LAM) (Section 4.2.5).
Figure 5.13: Mapping the conceptual framework for the game as created onto the Ludic Action Model.

This mapping (Figure 5.13) assumes that the game as created will provide the game object that the player interacts with. Thus, the game object is formed of the same conceptual framework (i.e. it contains the same core components) as the game as created.
The coloured arrows in Figure 5.13 flow from the *game as created* to the components of the *game as published*, and the player within *game-player space*, that they are ultimately contained or constructed within. It should be noted that in the case of the game’s Technology components, they can be conceived as being received by the player, via the world, as part of the game’s output.

**5.3.1: Mapping ‘Technology’ to the Ludic Action Model**

Ludodiegetic technology, that includes objects such as printed maps that ship with the game product, can be conceived as existing as part of the *game as published*, but existing outside of the *game object* itself. Non-ludodiegetic technology, that includes objects such as console or computer components that enable gameplay, can be conceived as existing within *game publication space* (i.e. the space in which the *game object* is interacted with) as it is part of the game-player interaction system. However it is outside of the *game as published* as such technology is not (usually) shipped as a core part of the game product.

![Steel Battalion's bespoke input device](image)

Figure 5.14: Steel Battalion’s (Capcom Production Studio 4 and Nude Maker, 2002) bespoke input device that shipped with the game.

It is noted that there may be exceptions to this distinction, such as Steel Battalion (Capcom Production Studio 4 and Nude Maker, 2002), that shipped with a bespoke multifunction input device that was required to play the game (Figure 5.14). In this case, this enabling technology could be considered a component of the *game as published*. However, this type of required technology shipping with every copy of the game product is not a common occurrence across mainstream games and thus, the situating of non-ludodiegetic technology within *game publication space* will be used in this thesis.

**5.3.2: Mapping ‘Aesthetics’ to the Ludic Action Model**

The game’s aesthetics, both homodiegetic and heterodiegetic, can be conceived as existing within the output from the *game object*. This output will be formed of a combination of visual, auditory and haptic outputs.

**5.3.3: Mapping ‘Told Story’ and ‘Constructed Narrative’ to the Ludic Action Model**

The ‘told story’ of the game, being a designed component of the *game object*, can be conceived as being contained within the *game object*. It requires the player to experience it via the game’s
output, however as a conceptual framework component, it is contained within the game itself. Conversely, the ‘constructed narrative’ is constructed within the player; specifically, it is constructed within the player’s long-term memory (via processing through working memory), being updated continually while the player is playing the game and perceiving new stimuli via the game’s output.

5.3.4: Mapping ‘Mechanics’ to the Ludic Action Model
The game’s mechanics, both homodiegetic and heterodiegetic, as with the ‘told story’, can be conceived as existing within the game object.

5.3.5: Implications of Mapping the Conceptual Framework to the Ludic Action Model for Operationalising the Disruptive Game Design Philosophy
A single ‘cycle’ of the LAM during gameplay can be conceived, at a highly simplified level, as consisting of four stages:

1. Input being received by the game technology and the game object within game publication space.
2. This input in turn modifies game mechanics and/or furthers the game’s ‘told story’, creating a new game state.
3. The new game state is represented by the aesthetic outputs of the game.
4. The player in game-player space can then respond to the aesthetic output, via cognitive processing, with further performative input. They can also update their ‘constructed narrative’ in long-term memory, via processing in working memory, with any newly perceived stimulus information from the game object.

During the design and development process, it is possible to create manipulations of ludodiegetic technology, mechanics, and told story. It is also possible to control the aesthetic output of the game object that provides the perceivable evidence of the manipulation of different game components.

However, the player only ever directly perceives the output technology’s representation of the designed aesthetic output. That is, the output technology filters the designed output from the game object. The output of the majority of commercial games is limited by the current, generally available, technology to the modalities of audio, visual and haptic output. Thus, any attempt to operationalise the disruptive game design philosophy into, for example, player-facing disruptive game mechanics must also ensure that the mechanic has a means of being perceived by the player through one or more of these output modes. This means that any disruption of a game component within this conceptual framework must ultimately be aesthetically presentable either visually, audibly or haptically, or a combination thereof.

The potential difference between the designed aesthetic output of the game object (i.e. the aesthetic output sent to the output device as pixel colour information, digital sound information...
or haptic feedback instruction information) and the technologically-represented aesthetics (i.e. what the output device actually provides the player) is important to note, as this may cause a player’s response to the game as played to be a response to a notably different experience to what was intended in the game as designed and the game as created.

Differences may occur, depending on how individual devices are calibrated (e.g. resolution, brightness and contrast settings on a monitor or television, or the physical arrangement of speakers in a room) and these differences may produce slight, or significant, differences between individual players’ experiences of the game. A player’s individual technology arrangement may provide an accurate representation of the game object, or game as created. However, it may also in some cases produce a different game as played experience to what the design and development team intended by filtering the designed aesthetic output in different ways.

Lost Odyssey (Mistwalker and feelplus, 2008) provides an example of different technology filtering the game’s output in a way that is detrimental to the player’s experience of the game as played. The game is a role-playing game that utilises a number of text-based menu screens, for item management, character management, combat and dialogue (Figure 5.15). On standard-definition display devices and particularly on smaller screen sizes, some of the game’s text is difficult if not impossible to read clearly; a problem noted by a number of players. The game as played for these players is thus significantly affected, as any actions during gameplay requiring the use of the menu screens becomes much more challenging and potentially frustrating.

Conversely, different technology arrangements may provide players with an enhanced game as played when compared to other players. For example, playing a competitive multiplayer game such as Call of Duty: Black Ops (Treyarch, 2010) with headphones as opposed to speakers may offer players a tactical advantage, as they may be more likely to hear subtle sound cues such as enemy footsteps, or heavy breathing. This allows players greater spatial awareness around their avatar.
The design and development team have limited control over an individual player’s technology itself, beyond what output is sent to it from the game. For example, the types of display devices that a player may use to play a console-based or PC-based game may vary drastically in display quality, resolution, screen size and brightness. Some players may play with headphones, some with stereo speakers, and some with surround sound systems, of which there are further variants offering differing degrees of fidelity. However, there is no way during development to know the exact hardware interface setup that a player will use to play a game.

This is a consideration in all game design and development projects, not only those utilising a disruptive game design philosophy. Many games include a method either within their graphical options menu or during the process of starting a new game to assist players in correctly setting variables such as screen resolution and brightness and many games will provide appropriate options for, at least, mono and stereo speaker arrangements, if not also for headphones and different surround sound systems. Many issues can also be avoided for PC-based games through the statement of minimum and recommended hardware and software requirements.

In terms of designing and developing disruptive game features, consideration must be given to how those features will interact with likely potential hardware interface setups. For example, it would be poor use of development time, resources, and budget implementing disruptive game features that may not be experienced by players without certain hardware interface arrangements. These considerations become less critical in contexts where the target platform is more specifically constrained (e.g. the majority of console hardware), although similar issues may need to be considered for any games with intended cross-platform releases.

However, opportunities for experimentation and innovation through designing game features that take advantage of emerging technology may be missed if all design decisions are based on ensuring the largest number of players is able to experience the feature. Thus, game features that rely on particular technology being used to play the game in order to be experienced should be evaluated appropriately. For example, a game feature that can only be experienced by a player playing the game via an Oculus Rift device may be worth investing development time and resources into in the context of particular projects (e.g. first-person horror games), while in others such investment may be less justifiable.

### 5.4: Disruptive Game Feature Properties

Using the conceptual framework of the *game as created* (Section 5.1 and Section 5.2), the fundamental properties of a *game feature* can be specified. A game feature is broadly defined in this thesis as any designed element of a game. Thus, a game feature must, as a minimum requirement, be constructed from at least one type of game component (i.e. Technology, Aesthetics, Story, Mechanics) (Section 5.2) and be represented via the game’s available output modes (i.e. auditory, visual, haptic) (Section 5.3).
A disruptive game feature however, requires further structural elements to be considered, which can be defined utilising the LCM (Section 3.2) and LAM (Section 4.2.5). A disruptive game feature, in addition to the basic requirement of a game component and a mode of output, require an intended mode of disruption (i.e. Encoding Disruption, Recall Disruption, Action Plan Disruption) (Section 4.3) and a target knowledge type (i.e. intraludic, transludic, extraludic; procedural, semantic, episodic) (Section 3.3) that will be disrupted in some way. The structural requirements, or properties, of a disruptive game feature are summarised in Table 5.1.

<table>
<thead>
<tr>
<th>Property Category</th>
<th>Source</th>
<th>Category Description</th>
<th>Property Classes</th>
</tr>
</thead>
</table>
| Game Component    | Conceptual framework of the game as created (Chapter 5) | Game component(s) to be manipulated to cause a disruptive effect on the player. | • Aesthetics  
• Mechanics  
• Story/Narrative  
• Technology |
| Mode of Output    | Output constraints identified in the conceptual framework implications (Section 5.3.5) | Mode(s) of output to allow an aesthetic representation of the disrupted game component(s) to be experienced by the player. | • Visual  
• Auditory  
• Haptic |
| Mode of Disruption| The Ludic Action Model (Section 4.2.5 and Section 4.3) | Designer-intended mode(s) of disruption to be experienced by the player. | • Encoding Disruption  
• Recall Disruption  
• Action Plan Disruption |
| Knowledge Type    | Ludic Cognition Model (Section 3.2), Ludic Knowledge Types (Section 3.3) and the Ludic Action Model (Section 4.2.5). | Target knowledge type(s) that contain the specific item(s) of player knowledge to be intentionally disrupted by the designer. | • Intraludic-Procedural  
• Intraludic-Semantic  
• Intraludic-Episodic  
• Transludic-Procedural  
• Transludic-Semantic  
• Transludic-Episodic  
• Extraludic-Procedural  
• Extraludic-Semantic  
• Extraludic-Episodic |

Table 5.1: Disruptive game feature properties.

A disruptive game feature is thus suggested to be structured of four property categories, with each category containing a number of property classes. Different combinations of classes from each category of this structure may be possible and, in the context of a complex commercial title, it is
possible that there will be a degree of overlap. For example, a disruptive game feature may be perceivable only visually, only audibly or only haptically, although it is possible for it to be perceived via a combination of output modes (e.g. an explosion may be seen (visual output), heard (auditory output) and felt (haptic output) simultaneously). A disruptive game feature may similarly target only intraludic-procedural (INT-P) knowledge. However, it may also target intraludic- and transludic-procedural (INT-P and TRANS-P respectively) knowledge or indeed, any other combination of knowledge types.

It is not suggested to be necessary to only utilise a single property class from each category, although it may be that a disruptive game feature will identifiably utilise a primary property class from each category (e.g. being perceived primarily through visual output, with a less significant associated haptic output, or primarily targeting intraludic knowledge with a possible impact also from transludic knowledge).

Figure 5.16: Visual representation of disruptive game feature property categories and classes.

Icons are licensed under a Creative Commons Attribution 3.0 Unported License and have been modified from their original formats in accordance with this license. The original works are attributed to Designmodo, Timothy Miller and Yankoa and were retrieved from the IconFinder database (IconFinder, 2014).
The list of property categories and contained classes presented in Table 5.1 can also be presented visually (Figure 5.16). This provides a simplified overview of how different property categories and their contained property classes may be combined to form different disruptive game features.

5.5: The Conceptual Framework in Game Space

The conceptual framework for the game as created can be added to the game space model (Section 1.2) (Figure 5.17). The framework provides a means of understanding the game as created, as the LAM provided a means for understanding the lower levels of the game space model.

Figure 5.17: The conceptual framework provides a means of understanding the structure of the game as created in game space.
5.6: Chapter Summary

A conceptual framework for the *game as created* has been presented as a means of enabling the operationalisation of the disruptive game design philosophy into player-facing disruptive game features (Section 5.1 and Section 5.2). The conceptual framework structured as a *ludodiegesis* (i.e. a game-based fictive reality), split into the *homodiegesis* (i.e. game components within the virtual environment, such as the player’s avatar) and the *heterodiegesis* (i.e. game components that support the game but exist outside of the virtual environment, such as menu systems and loading screens). Within this structure, four types of game component are defined (i.e. Technology, Aesthetics, Story, and Mechanics) using Schell’s (2014) ‘elemental tetrad’ of game components as a basis. Schell’s component definitions are further refined, with examples of games that do not easily fit within Schell’s structure. Mechanics, specifically, are redefined drawing on the work of Sicart (2008). The refinement of Schell’s definitions within the broad elemental tetrad structure is tailored towards the specific context of digital games.

This proposed conceptual framework has been mapped to the LAM (Section 5.4) to demonstrate the links between the *game as created*, the *game object* within the *game as published*, and the player within *game-player space*. Mapping the conceptual framework to the LAM, in conjunction with the previously defined ludic knowledge types (Section 3.3) and modes of disruption (Section 4.3), enables the definition of the properties of a disruptive game feature (Section 5.4), split into four different property categories, each containing a selection of property classes. A disruptive game feature requires at least one property class from each category to be definable as an intentionally designed, player-perceivable, disruptive game feature.

The final element of *game space* that is next considered is the *game as designed* and how disruptive game features may be designed and then implemented into a game.
Chapter 6: A Design and Development Model for the Game as Designed and the Game as Created

A model is required for the design process leading to the game as designed that enables both the design and then development of disruptive game features that will be contained within the game as created (Figure 6.1). The conceptual framework for the game as created (Chapter 5) can be used as a basis for such a model.

Figure 6.1: Chapter 6 focuses on the game as designed within game design space and the development process that leads to the game as created in game development space.
6.1: The Disruptive Game Feature Design and Development (DisDev) Model

Figure 6.2: The Disruptive Game Feature Design and Development (DisDev) model, demonstrating the design process for disruptive and non-disruptive game features within the *game as designed*. 
Building on the defined disruptive game feature property categories and classes (Section 5.4) and the visual representation of how they combine to form a disruptive game feature (Figure 5.16), a design model can be proposed. This Disruptive Game Feature Design and Development (DisDev) model (Figure 6.2) is intended to provide a practical design and development tool that can be used to guide the design and development of a game based on the disruptive game design philosophy.

The DisDev model, situated as a process within the game as designed, provides a high-level view of the design process for different features in a game designed in line with the principles of the disruptive game design philosophy. Similar models could theoretically be created for games utilising different design philosophies as well.

The design process in this case is simplified into two channels; disruptive game features and non-disruptive game features. It would be possible to create a more detailed model with potentially many more channels, depending on the number of different design philosophies influencing a project. In the current research, the only philosophy being intentionally applied is disruptive game design, thus, the two-channel model is suitable. The model is purposely presented as four stages to encourage the designer to consider how different property classes from each property division are being utilised in their games.

### 6.2: Examples of Disruptive Game Features Utilising the Disruptive Game Feature Design and Development Model

Through taking example scenarios from existing commercial games and proposing suggested alterations to them within the context of the DisDev model, it is possible to demonstrate how disruptive game features could be designed and implemented in otherwise non-disruptively-designed games.

#### 6.2.1: Encoding Disruption of Audibly Perceived Story Information

![Figure 6.3: DisDev model for Encoding Disruption of audibly perceived Story information.](image)

Various games utilise a narrator as a means of delivering the game’s told story, such as Bastion (Supergiant Games, 2011), an action role-playing game with story delivered in short spoken segments. This story information is being perceived and encoded by the player into long-term intraludic semantic (INT-S) memory (i.e. the ‘factual’ story information) and intraludic episodic (INT-E) memory (i.e. the ‘spatiotemporal context’ of the storytelling). This encoding may be disrupted (Figure 6.3) either through purposely providing the player with ambiguous stimuli (i.e. ambiguous story information) or through purposely making the auditory delivery of the information difficult to focus attention on. Thus in this scenario, Encoding Disruption could be
achieved by delivering story information audibly whilst also presenting the player with multiple enemies that must be fought. This requires the player to split their attention between two primary stimuli (i.e. the story information and the visual enemy threat) and thus requires the player to analyse the different stimuli in order to attend to them effectively. The combat itself may then generate further auditory stimuli that may also disrupt the encoding of story information.

### 6.2.2: Recall Disruption of Haptically Perceived Mechanic Information

![DisDev model for Recall Disruption of haptically perceived Mechanic information.](image)

The ‘rumble’ feature of a gamepad may be used to provide haptic feedback on different in-game situations and events, although is typically used for visceral events such as shooting weapons or objects exploding. However, it may also be used in other ways, such as to simulate the player-character’s heart rate as a means of indicating their physical state, or as a means to allow a player to identify potential hazards and traps in the game environment.

![Wall-mounted flame trap in The Elder Scrolls V: Skyrim (Bethesda Game Studios, 2011).](image)

For example, *The Elder Scrolls V: Skyrim* (Bethesda Game Studios, 2011) contains flame traps that emit flames at regular intervals that damage the player (Figure 6.5). A haptic warning may be attached to these traps, with a low-level ‘rumble’ being triggered just before the trap activates, allowing players time to recognise the danger and avoid it. This recognition is based in intraludic-procedural (INT-P) knowledge (performing a series of actions in response to a stimulus), as well as intraludic-semantic (INT-S) knowledge of the properties of the ‘flame trap’ concept instance, and intraludic-episodic (INT-E) knowledge of specific flame traps in the game.
Recall Disruption occurs when a familiar stimulus (i.e. the ‘rumble’-based haptic feedback) is experienced in an unfamiliar context (i.e. not nearby to a flame trap). Recall Disruption of this intraludic-procedural knowledge (Figure 6.4) may thus be achieved by triggering a warning that is then not followed by trap activation. The result of such disruption may be that players approach possible trap locations with more care (i.e. player engagement in active analysis of the situation) as they are unable to reliably predict when a trap may activate. Alternatively, players may decide that because there is no identifiable pattern to the trap activation, it is instead a worthwhile risk to simply move through them as quickly as possible. Both of these decisions however still require the player to actively analyse and evaluate the situation.

If the haptic feedback could be reliably linked to trap activation and a rule or pattern established by the player (e.g. flame traps always activate 1.5 seconds after the ‘rumble’ warning is triggered), then any scenario involving flame traps could be overcome by simply applying that understood information. This would enable greater use of lower-order thinking with minimal need to engage in higher-order thinking.

### 6.2.3: Action Plan Disruption of Visually Perceived Mechanic Information

**Figure 6.6: DisDev Model for Action Plan Disruption of visually perceived Mechanic information.**

Action Plan Disruption occurs when a game component does not respond to player interaction in the manner expected by the player, based on their previously constructed knowledge (Figure 6.6).

**Figure 6.7: Exploding barrels in DOOM (Id Software, 1993) (left) and Half-Life 2 (Valve Corporation, 2004) (right).**

This can be demonstrated using a commonly deployed trope, ‘red barrels’ that explode when shot or damaged. As Pinchbeck (2007, 2013) describes, the common use of the ‘exploding barrels’ concept has extended from its usage in DOOM (Id Software, 1993) initially (where such barrels were not in fact coloured red), to become an established component of the first-person-shooter genre. In many cases, such exploding barrels are now commonly also coloured red (TV Tropes...
Foundation LLC, 2014c) to make them distinguishable from non-exploding ‘prop’ barrels used for environment decoration.

Berg (in Funk, 2011) discusses the design process for the explosive barrels in *Bulletstorm* (People Can Fly and Epic Games, 2011), stating that initially the intention was to make explosive barrels green.

[We tried doing] something different, instead of going with the cliché. In the beginning we had green barrels, but people didn't get it right away [. . .] They got completely ignored by the players and no one guessed or assumed that they were explosive. Why not? Because they weren't red. Everyone knows that only the red barrels are explosive. (Berg in Funk, 2011, para.3)

Other than trying to do something different to the “cliché”, Berg does not provide any further motivation for initially making explosive barrels green, thus making any ‘disruptive’ design intent impossible to confirm. The design may have been a purely artistic decision, or it may have been motivated by wanting to provide a greater degree of cognitive challenge to players. Berg does however place the responses to the green barrels in the context of the fast-paced gameplay of *Bulletstorm* stating that

There's no time to analyze [sic] objects on a detail level, so the shape and color [sic] have to be enough. It became apparent for us that the most efficient way to communicate [the barrel's] purpose was to make it red. (Berg in Funk, 2011, para.4)

However, the use of red exploding barrels nevertheless provides a potential target for implementing Action Plan Disruption, based on the potential contents of a player’s transludic-semantic (TRANS-S) knowledge (i.e. knowledge related to the properties of red barrels in other games), in games where there is time to analyse objects more closely.

Whereas previous examples (Section 6.2.1 and Section 6.2.2) required the game to initially provide an opportunity for the player to encode intraludic knowledge of game components, this example relies on transludic knowledge as the concept itself is an identifiable game trope (particularly prevalent in the broad ‘shooter’ genre). Therefore, immediately providing players with situations in which red barrels do not explode causes Action Plan Disruption; the response of a game component that has semantically defined properties (i.e. a red barrel) to an established player action (i.e. shooting it with a weapon) does not result in the expected outcome (i.e. the barrel exploding).

This initial disruption may then be built upon in a number of possible ways to promote a higher degree of cognitive engagement with the game, by manipulating the semantic properties of the red barrel. Different internal game rules may be established that dictate what types of barrel may explode or under what circumstances. Different coloured barrels may explode, for example, or barrels that have particular symbols on them, regardless of their colour. Barrels may only explode if they are already visibly damaged and hence the contents compromised. Exploding barrels may
have additional detrimental effects, such as releasing poisonous gases that damage the player’s avatar.

Each of these scenarios would require a greater degree of cognitive engagement from the player, requiring them to use higher level analysis, evaluation and problem solving skills rather than being able to rely solely on previously learned knowledge and previously applied responses to the particular stimulus. However, because Action Plan Disruption occurs following the player’s performance of an action (i.e. their application of learned knowledge), it will likely require players to create new plans of action in order to respond more appropriately to the stimulus in future. This thus encourages players to utilise all three levels of higher-order thinking.

6.3: Constraints on Disruptive Game Design Utilising the Disruptive Game Feature Design and Development Model

While disruption is theoretically possible in a number of ways via the DisDev model, there are some potential constraints on what may be able to be disrupted in a manner that will remain enjoyable for players. As stated in Section 1.3, disrupting ‘too much’ or ‘too frequently’ may result in detrimental effects on the player experience. Poorly implemented disruption may not be experienced by players as disruption of their understanding of particular types of knowledge but instead, be experienced as an unintentional error or ‘bug’ in the game. This may confuse, frustrate, or anger players, causing a negative impact on the player’s enjoyment of the game and thus, their overall perception of the game’s quality. A commercial game project must be able to provide an enjoyable player experience to be successful, thus the disruptive game design philosophy and the DisDev model must consider potential threats to that.

6.3.1: Disrupting the ‘Technology’ Property Class of the ‘Game Component’ Property Category

‘Technology’ within the conceptual framework (Section 5.2.1), which is identified as a property class of the ‘game component’ property category (Section 5.4), is problematic as a target for disruption. Separating Technology from the other classes of game component is difficult due to every other game component class being reliant upon the technology in order to function and be perceivable by players. Thus, any disruption of Technology will likely disrupt other game components as well. It is thus questionable whether disruption of Technology alone is possible, or whether the Technology component simply provides a platform for the disruption of other game component classes (i.e. Aesthetics, Story or Mechanics).

Disrupting the non-ludodiegetic game hardware components (e.g. input and output devices) themselves is possible, although it is unlikely to be practical or beneficial in providing an enjoyable experience for the player. For example, altering the game’s input configuration during gameplay, changing the volume level of the game’s audio output, or manipulating the brightness and contrast of the game’s visual output may all disrupt a player’s established knowledge relating to the behaviour and properties of the game hardware. Such examples disrupt the Technology
component without necessarily disrupting other game components. However such disruption may easily result in player frustration rather than cognitive engagement, especially if it is experienced by players as an unintentional game ‘bug’.

However, cases in which Technology may be solely disrupted are likely to be uncommon and as previously stated are unlikely to be beneficial in producing an enjoyable game experience. Even notable examples of Technology-centric disruption, such as the insanity effects in *Eternal Darkness: Sanity’s Requiem*, combine other game components. The game will manipulate the volume level, but will accompany this with a visual volume bar, thus simultaneously disrupting aesthetic game elements as well.

Technology remains as a property class within the game component property category of the DisDev model as it may provide a potential candidate for implementing a disruptive game feature; however it is more likely that the Technology will serve simply as a basis for the disruption of other game component classes or will be disrupted in conjunction with other game components classes.

### 6.3.2: Disruption in the Context of the Individual Game

A game feature that may be experienced as being ‘disruptive’ in one game may appear to players as being a game ‘bug’ in another. As with other design decisions, disruptive game design must be implemented in a manner that suits the particular game. Disruptive game features must make at least some sense within the broader context of the game’s homodiegesis. This can be considered from the perspective of the game’s ‘rules’; disruption is not analogous to simply ‘rule breaking’, rather, it is ‘contextualised rule bending’ that aids in creating an enjoyable and cohesive game experience that also supports cognitive engagement through requiring players to utilise higher-order thinking.

This contextualisation can be elaborated using an example of the common practice of saving a player’s progress during gameplay. Allowing the player to freely save their progress at any point during gameplay, an unlimited number of times, allows a high degree of accessibility and usability as players are not required to commit to playing until the next ‘save point’. Such a method of saving is used in *Half-Life* (Valve Software, 1998) and a number of other PC-based first-person-shooters. However, such a system may be detrimental to both the game’s challenge and its atmosphere, especially in scenarios in which the game is trying to instil tension, fear or anxiety in the player. The player is aware that any mistakes can be easily undone by reloading a previous save state; a practice colloquially referred to as ‘save scumming’ (Schwarz, 2012).

Limiting the number of saves a player is allowed to use during a single play-through of a game (as is the case in *Resident Evil* (Capcom, 1996) and a number of later games in the series) requires players to think strategically about the best places in the game to make use of a save game. In *Resident Evil*, this decision is contextualised within the homodiegesis by requiring players to save (i.e. a non-ludodiegetic technology) using a typewriter that requires individual ink ribbons (i.e. part
of the homodiegetic aesthetics and told story). If a player runs out of ink ribbons, they are unable to save their progress until they find more. This system also contributes to the game's underlying 'survival' theme by requiring close management of a limited number of valuable resources.

If these two different save systems were to simply be swapped between, for example, *Half-Life* and *Resident Evil*, it is unlikely that either would aid in producing a more enjoyable player experience, as the system structures are not rooted within the games' homodiegeses nor do they support the core principles of the gameplay. However, disruption of the save system in either game may be possible by changing the system's rules at certain points in the game.

In the *Resident Evil* format for example, the system could be disrupted by presenting players with a scenario in which a particular typewriter becomes jammed after being used (i.e. Action Plan Disruption of visually perceived Technology/Mechanic information). Thus, regardless of the number of available ink ribbons the player will be forced to move forward to locate another place to save their game rather than relying on the knowledge that they can always return to a previously used typewriter if necessary. This disrupts the established rules of the save system but does so in a manner that is homodiegetically appropriate and in-keeping with the game's 'survival' theme. This disruption thus requires players to adapt their gameplay approach (i.e. to create a new approach to gameplay) and to tune their existing schematised knowledge relating to the game's saving system (i.e. adding information to the 'typewriter' schema that identifies that some typewriters may jam after being used).

What is appropriate, within a particular game's homodiegesis, is a decision that must be made by the designer. It is a necessary consideration in the process of designing disruptive game features, so as to minimise the risk of such features being negatively received by players. Cognitive engagement, through encouraging higher-order thinking, must be the driving principle of any game utilising the disruptive game design philosophy. Thus, any disruptive game features must be assessed in terms of their ability to encourage higher-order thinking, while minimising as far as possible the risk of them being mistaken as unintentional game 'bugs'. This minimising of risk should be considered during the design process but will require analysis and evaluation of feedback from gameplay testing sessions to fully understand.

**6.4: Understanding a Project's Contextualising Factors that may Influence Design Decisions**

Understanding the context of any design project is critical to make appropriate design decisions. Disruptive game features must be developed in a manner that is suitable for the current project's context (Section 6.3.2). Thus it is necessary to further refine the DisDev model (Section 6.1) to include contextualising factors that may influence design decisions.
The structure of *game space* (Section 1.2) suggests potential sources of such contextualising factors that may influence design decisions (Figure 6.8). These apply not only to the disruptive game design approach but to game design more broadly.

Within *game design space* and leading to the release of a *game object* into *game publication space*, a *development process* is required. The constraints placed on this process and the methods by which it is undertaken must contextualise design decisions (i.e. the context of the commercial development process) and this needs to be considered in more detail (Section 6.4.1).

Once a *game object* is released into *game publication space* it is then subject to culturally driven connotations and assumptions being made about it. Two principle influences on these assumptions are of particular interest from a design perspective. Firstly, the players’ perspectives of the *medium* of games and the associated medium-based tropes (Section 6.4.2) (i.e. the context of the medium of games), which are frequently the basis for features and articles in the gaming media; see, for example, Houghton’s (2013) overview of what he deems overused game tropes, Brooks’ (2014) assessment of common game tropes that do not apply to *Dark Souls 2* (From Software, [Image])

---

Figure 6.8: The *game space* model, with contextualising factors and their sources summarised alongside.

The structure of *game space* (Section 1.2) suggests potential sources of such contextualising factors that may influence design decisions (Figure 6.8). These apply not only to the disruptive game design approach but to game design more broadly.

Within *game design space* and leading to the release of a *game object* into *game publication space*, a *development process* is required. The constraints placed on this process and the methods by which it is undertaken must contextualise design decisions (i.e. the context of the commercial development process) and this needs to be considered in more detail (Section 6.4.1).

Once a *game object* is released into *game publication space* it is then subject to culturally driven connotations and assumptions being made about it. Two principle influences on these assumptions are of particular interest from a design perspective. Firstly, the players’ perspectives of the *medium* of games and the associated medium-based tropes (Section 6.4.2) (i.e. the context of the medium of games), which are frequently the basis for features and articles in the gaming media; see, for example, Houghton’s (2013) overview of what he deems overused game tropes, Brooks’ (2014) assessment of common game tropes that do not apply to *Dark Souls 2* (From Software, [Image])

---

Figure 6.8: The *game space* model, with contextualising factors and their sources summarised alongside.
2014), or Smith (in Thier, 2012) discussing the efforts made to avoid the use of “dogmatic” game tropes in the design of *Dishonored* (Arkane Studios, 2012). The second influence is the players' perspectives of the more specific connotations that may be associated with particular game 'genres' (Arsenault, 2009) including a particular game’s ‘mode’ and ‘milieu’ (King and Krzywinska, 2002) (Section 6.4.3) (i.e. the context of 'genre', 'mode', and 'milieu').

*Game publication space* not only exposes the *game object* to cultural influences but also to existing commercial influences. For example, if a game is released that is a sequel, or serialised entry in an established franchise, this is likely to generate different expectations in potential players than a game that is based on a new intellectual property. This may also apply to particular development studios, publishers, or even individual developers (that may be referred to as game ‘auteurs’ (Pfeifer, 2005)) that have an existing catalogue of games of a particular style or type, such as Goichi Suda (‘Suda 51’) or Shigeru Miyamoto. Thus, a further contextualising factor for design decisions is the impact of the studio, developer, or franchise’s established player-facing ‘image’ and the likely expectations of the established player base (Section 6.4.4) (i.e. the context of series, developer, and/or publisher history).

Each of these contextualising factors needs to be further explored before then being integrated into the DisDev model (Section 6.5).

**6.4.1: Context of the Commercial Development Process**

The commercial situation in which a game is developed has an impact on design decisions. The development budget, the size and skills of the development team, the length of time available to complete the development, the hardware and software that is being used to develop the game, and any constraints or requirements put in place by an external publisher or distributor (if one is attached to the project) all influence design decisions. For example, an initial *game as designed* may include twenty separate levels. However, if a publisher requests the game be completed within a shorter period of time, the *game as designed* may be changed so fewer levels are in the *game as created*. Such a change would have an impact on the eventual player experience of the *game as played*. An alternative solution to this situation may be to hire more staff to produce the intended number of levels. This may then have follow-on effects however, for example less budget being available for other aspects of the development process, which may still have an impact on the *game as played*.

While the initial development context (e.g. team size and skills, available budget, hardware and software availability, etc.) should be known at the design stage and can thus be taken into consideration, later changes (e.g. publisher requests) may only become evident during the development process. These will then have to be responded to.

**6.4.2: Context of the Medium of Games**

Players may bring, to a game, expectations formed through their prior experience with games as a medium. That is, a player’s expectations not of the game’s specific properties, informed by its
content or themes, but of the game’s inherent properties acquired through it being an artefact within the medium of games.

It is possible to readily identify within gaming culture a number of established tropes, patterns and behaviours that are present across a range of games in the medium. The previously cited sources taken from the gaming media (Section 6.4) demonstrate a range of examples of such identifiable tropes. Community managed database, TVTropes.org (TV Tropes Foundation LLC, 2014a), provides a range of specific examples to further illustrate these culturally identified concepts within the game medium.

These tropes often involve players of games accepting ‘game logic’ that would otherwise make little sense outside of the medium of games. For example, the use of spider webs as trampolines to reach higher areas (TV Tropes Foundation LLC, 2014b) (in games such as Jak and Daxter: The Precursor Legacy (Naughty Dog, 2001), Sly 2: Band of Thieves (Sucker Punch Productions, 2001), and Pokémon X and Y (Game Freak, 2013)); using the force of explosive weapons as a propellant to jump higher than usual, commonly known as ‘rocket jumping’ (TV Tropes Foundation LLC, 2014e) (in games such as Daikatana (Ion Storm, 2000), Team Fortress 2 (Valve Corporation, 2007) and Borderlands (Gearbox Software, 2009)); or, obtaining items from defeated enemies that are not logical for that enemy to have been carrying, such as obtaining gold coins from defeated animals (TV Tropes Foundation LLC, 2014d) (in games such as Breath of Fire II (Capcom, 1994), Eternal Sonata (tri-Crescendo, 2007), and Kingdom Hearts (Square, 2002)).

The use of such game medium tropes is discussed by Ruch (2010), who suggests that they represent necessary abstractions from reality. Ruch suggests that some tropes are present for pragmatic reasons, such as minimising player frustration or supporting critical game systems. Obtaining gold coins from defeated animals demonstrates this; it would be unlikely to be satisfying or enjoyable for a player to be left with only a dead animal carcass following a combat scenario when material rewards within the game can only be purchased with gold coins. While the animal carcass may be a much more 'ecologically valid' outcome, it is of no benefit to the player. This could be changed if the game was to implement a ‘trade’ or ‘bartering’ ecosystem, although this would then remove the trope from the game.

Thus, in the context of disruptive game design it is possible to suggest that some tropes may have more potential for being a target for ‘disruption’ than others, as disruption of different expectations (in the form of transludic schematised knowledge) may influence a player’s enjoyment of a game to differing degrees, either positively or negatively. Moreover, the context provided by the medium of games (i.e. where abstractions from reality may be necessary to enhance player experience) and the associated medium-based tropes (i.e. manifestations of these abstractions) may provide possible areas in which to consider development of disruptive game features. They also provide a point of reference when evaluating a disruptive game feature against its likelihood to provide an enjoyable gameplay experience for players. One may view the
consideration of the context of the game medium as both preceding and informing the design of disruptive game features initially, as well as then following and evaluating the development of those features. However, these tropes are contextualised themselves in higher order conceptual categories. Some tropes may indeed be seen to apply across ‘all games’, however some will be more contained, applying instead either exclusively or primarily to a particular type, or style of game. For example, the previously described ‘exploding barrels’ trope (TV Tropes Foundation LLC, 2014c) can be found primarily in games within the broad ‘shooter’ category, although examples also exist in other types of game.

6.4.3: Context of the Game’s ‘Genre’, ‘Mode’, and ‘Milieu’

A game’s ‘genre’ is able to provide the established features (i.e. genre-based tropes) that may form the basis for designing disruptive game features. It is thus important to understand the genre in which a game is situated before undertaking design decisions. The term ‘genre’ is potentially problematic (Arsenault, 2009) in its specific meaning due to a degree of “definitional slippage” (Whalen, 2004, p.290) that has occurred in its transition into the medium of games from other media formats. Thus, a definition is required to discuss the specific context of a game’s ‘genre’. While it is beyond the scope of this thesis to enter into the ongoing debate around genre studies in games, existing literature suggests a functional, pragmatic, definition that can be utilised in the current research.

King and Krzywinska (2002, p.26) suggest that ‘genre’ may be defined as the selection of terminology used by the gaming community to “distinguish between broad categories such as action-adventure, driving, or strategy” (emphasis added). Whalen (2004, p.290) expands on this suggesting that ‘genre’ can be understood in terms of categorisation of games based on “the way the game is played or what one must do in order to accomplish the goals of the game”. While these definitions offer a broad suggestion of categorisation based on how a game is played, they can be further refined.

Thus, it is also beneficial to describe ‘genre’ in terms of what it is not. King and Krzywinska (2002, p.26-27) suggest two further defining components of a game that exist alongside its ‘genre’, in the form of the game’s ‘mode’ and ‘milieu’. The mode of a game is the way in which the game is experienced by the player, including for example the game camera (e.g. first-person, third-person), game scenario (e.g. single player, multiplayer) and the temporal flow of the game (e.g. real-time, turn-based). The milieu (King and Krzywinska, 2002, p.27) can be used in a similar way to how the term ‘genre’ is “usually employed in film, to describe the types of worlds reproduced within games in terms such as location and atmospheric and stylistic conventions”. The previously defined structure of ludodiegesis (Figure 5.11) can thus be mapped to each of ‘genre’, ‘mode’, and ‘milieu’ to indicate which game component classes (i.e. Aesthetics, Mechanics, Story, Technology) may be affected by these contexts.
‘Mode’ is suggested to include the configuration of the game’s ludodiegetic and non-ludodiegetic technology, as well as the game’s heterodiegetic aesthetics (Figure 6.9). The ‘mode’ is the way that the game is experienced by the player and thus, necessarily includes the supporting components that enable that experience. For example, different ‘modes’ may include playing games on a PC monitor, a television screen, or a handheld device (e.g. different types of non-ludodiegetic technology). A special edition of a game that is sold with additional heterodiegetic technology (e.g. a printed map of the game world) may be played in a different ‘mode’ to the standard edition that does not include the printed map. Technology in all its forms supports the player experience, from the hardware device used to play the game (non-ludodiegetic Technology), to the software code (heterodiegetic Technology), through to any other ‘technology’ that may ship with the game.

The game’s heterodiegetic aesthetics are the audio-visual and haptic properties of the supporting game systems. This may include the game’s menus or other game systems that support the player experience but are themselves outside of the homodiegetic reality of the virtual environment. For example, the game’s camera (e.g. first-person camera or third-person camera), which in most cases is not a homodiegetic part of the game environment, although some exceptions exist, such as Super Mario 64 (Nintendo EAD, 1996) in which the game camera is a literal in-game camera controlled by a separate in-game character.
'Milieu' is suggested to include the configuration of the game's told story as well as the game's homodiegetic aesthetics (Figure 6.10). Milieu can thus be said to include all audio-visual and haptic properties of the 'fictive reality' of the virtual environment (e.g. game world, game characters, atmosphere, and stylistic conventions) as well as the story that is directly told to the player through that virtual environment and its contents.

It should be noted that the player's constructed narrative will be informed by cues within the game's milieu. For example, the player may construct a personal narrative through interaction with various elements of the game world. However, the constructed narrative itself is not contained within the game’s milieu, only informed by it. Thus, in this mapping, constructed narrative remains separate.
With 'mode' and 'milieu' defined and mapped to the structure of ludodiegesis, it is now possible to lastly suggest how 'genre' can be mapped to the remaining ludodiegetic components (Figure 6.11). With constructed narrative already discussed, the remaining components are the homodiegetic and heterodiegetic mechanics of the game. That is, methods invoked by agents in the game that affect a system or entity either within the game environment (i.e. homodiegetic) or outside of the game environment (i.e. heterodiegetic). This mapping can be seen to be in keeping with Whalen’s (2004) suggestion that genre can be defined through categorisation of what the player must do to accomplish the goals of the game; the player must 'invoke methods' to accomplish the goals of the game.

Historically therefore, one may describe a game as belonging to a particular genre based on a variety of properties. These properties may relate to how the player interacts with the game, such as the ‘first-person-shooter’ or the ‘point-and-click adventure’, the type of activities the player may engage with, such as the ‘puzzle’ game or the ‘platforming’ game, or the aesthetic, thematic or stylistic properties, such as ‘horror’ or ‘metroidvania’ (a commonly used portmanteau referring to games in the style of Metroid (Nintendo R&D1 and Intelligent Systems, 1986) and Castlevania (Konami, 1986)).

However, in considering design decisions, and particularly in the case of designing specific disruptive game features, it is necessary to utilise more specific definitions to consider independently each of genre (i.e. the gameplay-enabling mechanics), mode (i.e. how the game systems are supported, perceived and interacted with) and milieu (i.e. how the game world, content and story are perceived). Each offers potential for disruptive game features and potential
for positive or negative impact on the player’s enjoyment of the game. However, rationales for making disruptive changes to each of these may differ across projects in terms of suitability and impact on the player.

6.4.4: Context of Series, Developer, or Publisher History

As previously suggested (Section 6.4), player expectations may be different when engaging with a game that is part of a series or franchise (being based on a combination of transludic and extraludic knowledge), or that is from an established development studio or publishing house (extraludic knowledge), if the player has an awareness of the series/studio already. This is as opposed to when the player is engaging with a game based on a new intellectual property or that is from a new or less established studio that they do not have existing awareness of. Design decisions must consider these different initial player perspectives.

Faliskz and Wolpaw (2012), discussing the design considerations required during the process of developing Portal 2 (Valve Corporation, 2011), suggest that while sequels in other media are often poorly received “and almost never better than the original”, in the games industry, sequels conversely “have a long tradition of actually improving on the original [. . .] because games, in general, get better with iteration”. This may be through improvements in the underlying technology, or improvements, refinements and additions to the game’s mechanics, story or aesthetic qualities, for example.

However, when developing such games, designers must be aware that as soon as a game has another game against which it will be directly compared, there is potential for players that enjoyed the original to dislike significant changes that may be made to its established ‘properties’ (i.e. its genre, mode or milieu-based properties), even if those changes are intended to improve the player experience. This can be seen to an extent in the case of some Dragon Age: Origins (BioWare, 2009) player and critic receptions of Dragon Age II (BioWare, 2011), which Juba (2011, para.7) describes as catering to an audience that did not “connect” with the first game, whilst simultaneously alienating those players that did. The changes to Dragon Age II included a simplification of the first game’s combat system, character creation system, and equipment and inventory management systems which have been criticised for offering no notable benefit over the more complex systems available in Dragon Age: Origins (VanOrd, 2011).

This risk of players responding negatively to significant changes has been previously identified as one of the major drivers leading to the development of ‘conservative’, ‘iterative’ games (Costikyan, 1998, Dymek, 2010, 2012) (Section 1.1). Thus, making changes to established game properties is a potential influence in the design and development of disruptive game features for game series in particular.
6.4.5: Updating the Disruptive Game Feature Design and Development Model with Contextualising Factors

The DisDev model (Section 6.1) can now be updated to include the requirement for assessing the various contextualising factors for the particular project.

Figure 6.12: The Disruptive Game Feature Design and Development (DisDev) model, with Stage 1 (Assessment of Project Context) and Stage 2 (Game Feature Design) of the process specified.
With this addition to the model, it is suggested that the design process can be split into two distinct stages (Figure 6.12). Stage 1, taking place during the early stages of a project (i.e. what may be referred to as the concept stage (Adams and Rollings, 2007, Fullerton, 2014), is where the design and development team and/or the production staff should carry out the initial assessment of the project context (Section 6.4.1 to Section 6.4.4), along with defining what Adams and Rollings (2007, p.56) refer to as “concept elements”. These include properties of the game such as its theme, focus, genre and target audience (and in turn, the user requirements of users within the target audience), which are fundamental properties that will influence all future design choices. Other requirements of other involved parties will need to be considered in this initial assessment, such as any requirements specified by a publisher, investor, or intellectual property owner (e.g. if a game is using a licensed intellectual property).

The principles of the design philosophy being utilised (i.e. the disruptive game design philosophy in the current research) focus this assessment onto important contextualising factors. For example, the disruptive game design philosophy emphasises the importance of a player’s game as expected and their existing knowledge, thus an important contextualising factor is the possible existing knowledge players may have of previous games played (i.e. the context of the medium of games). The outcome of the Assessment of Project Context in turn places limitations on what can be designed in the current project utilising the design philosophy.

Stage 2 then comprises the creation of game feature designs in the form of design documentation and/or design specifications that will be developed by the development team. These features will be designed with awareness of the project’s context and any limitations or constraints (or possibly freedoms) placed upon it. The output of Stage 2 of the DisDev model is the design documentation that describes the game as designed. This documentation may vary in both depth and complexity between different projects, depending once again on the project’s context. Some studios may create only minimal documentation for their projects, while others may create large ‘design bibles’ detailing each specific component of the game. Even if the game as designed is not formalised through a documenting process, it may still exist, at minimum, within the mind of the designer.

6.5: Prototyping as a Bridge between Game Feature Design and Game Feature Development

Suggesting that a definite, clear division exists between the design and development stages of a game project, in all contexts, is impractical. To do so would suggest that once full development of a game has begun (and thus, the transition has been made from the separate design stage), then the original design specifications (i.e. the game as designed) cannot be modified or iterated based on new understanding acquired through or during the development process (i.e. the development of the game as created). Indeed, some project contexts may not afford such changes, such as if the game’s design specification has been explicitly defined and then contracted out to a development
studio. However, there may be contexts in which returning to the original design and making modifications may be possible and even beneficial to the project.

The currently proposed model of *game space* (Figure 6.1) and the DisDev model however, both suggest a linear progression from the *game as designed* to the *game as created* and both also suggest a clear division between them. In terms of typical project lifecycle models, this is similar to what may be referred to as a ‘pure waterfall’ structure (McConnell, 1996, p.136). Neither the *game space* model nor the DisDev model accounts for the process of prototyping game features that may be undertaken in some projects. Specifically projects making use of iterative lifecycle structures such as the spiral model, or evolutionary prototyping (McConnell, 1996), will have at least one prototyping stage, if not more.

The DisDev model is however intended to describe the design and development processes required for the inclusion of disruptive game features within a game. It is not intended to be constrained to games being developed within specific project lifecycle structures. Thus, it is necessary to afford within the DisDev model (and therefore, also within the *game space* model) the opportunity for prototyping to occur, whilst also affording the option for a project to move directly from design to development without going through prototyping.

### 6.5.1: Prototyping in the Disruptive Game Feature Design and Development (DisDev) Model

Existing game design and development process models (Figure 6.13) provide a basis for adding a prototyping stage into the DisDev model.

![Figure 6.13: Fullerton’s (2014, p.414) stages of development (left) and Adams and Rollings’ (2007, p.53) stages of the design process (right).](image-url)
Prototype development is described by Fullerton (2014, p.197) as being “at the heart of good game design and [. . .] [allowing focus to be placed] on a small set of the game’s mechanics or features to see how they function”. Fullerton places the prototyping process in a separate stage (i.e. the pre-production stage), following on from the concept stage. Similarly, Adams and Rollings (2007, p.53) suggest that prototyping occurs as a separate stage to the concept stage (i.e. what they refer to as the elaboration stage), although their process model specifically emphasises the iterative nature of this stage.

Both of these existing models (Figure 6.13) are able to describe key stages of the design and development process. However, in Fullerton’s (2014, p.414) model if a development problem is encountered during the production stage, there is no mechanism by which new design work and/or prototype development can be carried out to overcome it. The linear flow of stages would suggest such action to not be possible. Adams and Rollings’ (2007, p.53) model avoids this by explicitly demonstrating the iterative nature of the elaboration stage, however this model is also less detailed than Fullerton’s with fewer defined stages.

Being required to enter into a process of re-design and re-prototyping of a particular game feature during the production stage (i.e. once a prototype design has already been tested and approved) may cause issues with regard to budget and milestone deliveries if further prototype development has not been planned for. However, in some scenarios it may be necessary.

For example, it may not become apparent until a number of game mechanics are operating together that particular emergent behaviour caused by their interaction is detrimental to the player experience. This may not have been apparent when initial prototypes were developed and tested either individually, or in incomplete game scenarios. To remedy this problem, a process of feature re-design would be necessary (to re-balance the affected mechanics to prevent the emergent behaviour) and one or more new prototypes may need to be playtested to assess the affect of the redesign on the player experience. In Fullerton’s (2014, p.414) model, this would require the project to return to the pre-production stage. In Adams and Rollings’ (2007, p.53) model, this may be able to be rectified within the elaboration stage, although depending on the game mechanics or systems effected, the decisions made in the concept stage may also require reappraisal.

Prototyping within the DisDev model is thus suggested to provide a ‘bridge’ between the concept and design stages (i.e. the Assessment of Project Context and the game as designed) and the development stage. Furthermore, it is suggested in the DisDev model that issues encountered during the full development stage may require re-design and re-prototyping work to be carried out, thus removing the rigid divisions between the stages of the design and development process.
Figure 6.14: The Disruptive Game Feature Design and Development (DisDev) model, with Stage 3 (Game Feature Prototyping) and Stage 4 (Full Development) of the process specified.
**Game Feature Prototyping** thus forms Stage 3 of the DisDev model, while *Full Development* forms Stage 4 (Figure 6.14). Note that Stage 3 is situated across the *game design space/game development space* transition, emphasising its role as a conduit for transforming features from the *game as designed* into features in the *game as created*.

In Stage 3, the features of the *game as designed* are prototyped, playtested, and evaluated. The evaluation of feature prototypes will be based on playtesting feedback as well as the design philosophy principles underpinning the design process. The evaluation of a game feature may result in one of three outcomes. Firstly, the prototype may be approved for full development (e.g. playtesting feedback has been positive and the technical/practical requirements for fully developing the prototype are viable). Secondly, the prototype may be removed from the game completely (e.g. playtesting feedback was negative and significant design changes would be required to rectify the situation, or the prototyping process identified significant technical or practical issues that would prevent efficient full development of the feature). Thirdly, for many initial prototypes, the prototype may be evidently viable for full development but require some manageable design modifications first (e.g. only minor technical or practical challenges to overcome). This third outcome then leads to a further process of design, followed by a modified prototype being developed, playtested, and evaluated once more.

In Stage 4, the features that are approved for full development are assigned to individuals or teams within the development staff. The features are then developed via the appropriate development pipelines (e.g. modelling, sculpting, UVW unwrapping, texture painting, and exporting to the game engine for art-based features, or writing and debugging game code or game level scripts). During Stage 4, new problems may be encountered (e.g. unforeseen technical challenges that were not apparent in the prototype) and it may be necessary to iterate through the re-design and re-prototyping stages once more for specific game features.

Designers following the stages of the DisDev model and ensuring appropriate consideration is given to the project’s contextualising factors during Stage 1 should minimise the chance of game features having to be removed later at Stage 3. However, because game development is prone to modification and evolution at any stage of the creation process, the structure of the DisDev model emphasises the possible interactions between the *game as created*, the *game as designed* and the *prototypes*. For example, a project may change publisher during development leading to a change in the commercial context of the project (Section 6.4.1). This in turn may mean that previously appropriate features in the *game as designed* are no longer appropriate. These features may have already been prototyped and approved for full development, thus making changes to them or removing them from the game completely requires a return to Stage 1 of the DisDev model in relation to the features in question.

How the DisDev model is used is dependent on the individual project. Regarding the aims that drive philosophical approaches to game design (Section 1.2.1), some projects may have a primary...
aim of presenting players with a high frequency of game-mechanic-based disruptive game features while aiming to present alongside them a non-disruptive story that perhaps utilises common plot structures and devices. Other projects may have the primary aim of presenting a high frequency of disruptive game features related specifically to their story structure or delivery whilst maintaining more commonly recognisable, non-disruptive game mechanics. Other projects may focus instead on aesthetics-based disruptive game features alongside non-disruptive story and mechanics.

The range and combination of disruptive and non-disruptive game features is highly flexible. Thus, it may be necessary to initially consider the non-disruptive game features that will form the framework around which the disruptive game features will be implemented. It may, for example, be necessary to have a full draft of the game’s story (assuming that the aim of the project is not to implement disruptive story-based features) completed before considering the use of the DisDev model to implement suitable disruptive game features. Stage 1 of the DisDev model (i.e. the Assessment of Project Context) aims to allow designers to consider the disruptive game design philosophy in the context of their specific project.

The DisDev model presents a structure that is intended to be industry-facing and usable within multiple game design and development projects. It is also intended to be usable as a basis for further academic research into, not only the disruptive game design philosophy, but other design philosophies as well.
6.5.2: Prototyping in the Game Space Model

The addition of a prototyping stage to the DisDev model must also be reflected in the model of game space (Figure 6.1). The previously proposed model is subject to the same critique applied to the DisDev model in that it presents the design and development process as a clearly linear sequence of stages.

Figure 6.15: The model of game space modified to include the prototyping stage of the design and development process.

This refined model of game space (Figure 6.15) thus includes the prototyping stage and its potential interactions with the game as designed and the development process leading to the game as created. This refined model also acknowledges (via the dotted arrow from the game as created) the possible need to return to the Game Feature Design stage (i.e. the game as designed) from the
**Full Development** stage (i.e. the *game as created*) to respond to issues encountered during development.

This model of *game space* is now supported by the models proposed thus far; the Ludic Action Model (Chapter 4), the Conceptual Framework for the *Game as Created* (Chapter 5), and the Disruptive Game Feature Design and Development (DisDev) model (Chapter 6).

### 6.6: Constructing an Integrated Model

The theory and models presented previously can now be combined to present an Integrated Model of the disruptive game design philosophy and the accompanying design and development framework.

The Integrated Model combines the Ludic Action Model (LAM) (Section 4.2.5) (which itself is expanded from the Ludic Cognition Model (LCM) (Section 3.2)) of the *game as played*, the model of ludodiegesis (Chapter 5) within the *game as created*, and the DisDev model (Section 6.5.1) for creating game features within the *game as designed*. The Integrated Model is situated within the container of the *game space* model (Section 1.2) and tailored in the current research to apply to the disruptive game design philosophy (Section 1.3) specifically.
Figure 6.16: The Integrated Model (Part 1): design philosophy through to release candidate of game object.
The Integrated Model is split into two parts. Part 1 represents the design and development process, from design philosophy through to the creation of a release candidate of the game as created (game object). Part 2 represents the ludodiegetic structure that is contained within the game as created, which becomes the game as published via the publication process. The game as published is then interacted with by the player in game-player space, which allows the player to...
experience the game as played. This game as played may be influenced by their game as expected and prior experiences, drawing on information from sources such as the game product materials, the game’s marketing and games ‘culture’ more broadly.

6.7: Chapter Summary

A structure has been proposed, in the four-stage Disruptive Game Feature Design and Development (DisDev) model, through which the disruptive game design philosophy may be operationalised into player-facing disruptive game features, alongside other, supporting, non-disruptive game features.

The DisDev model is initially based on the conceptual structure of the game as created (Section 6.1). It is then expanded to take into consideration the contextualising factors that may influence a project and that thus should be considered by the design and development team at the initial conceptualisation stage of the project (Section 6.4). Prototyping is then identified as a separate stage that provides a bridge between the game as designed and the game as created (Section 6.5), providing a further expansion of the DisDev model (Section 6.5.1) and also a modification to the model of game space (Section 6.5.2).

An Integrated Model of the disruptive game design philosophy and the accompanying design and development framework is lastly proposed (Section 6.6). The Integrated Model combines the theory and multiple models that have been presented previously into a complete model, situated within the container of the game space model. The Integrated Model is presented in a format intended to be understood and utilised within an industrial game design and development project context, while being underpinned by a robust theoretical foundation rooted in existing research. However, as a ‘theoretical’ model it is necessary to test it via application in a real commercial game design and development project that will allow analysis of the model’s usability and applicability at each stage of design, prototyping, development, publication, and play.
Chapter 7:  
Designing, Prototyping, and Developing Disruptive Game Features in *Amnesia: A Machine for Pigs*

The disruptive game design philosophy (Chapter 1) requires an understanding of how digital games are structured, how players interact with that structure, and how a designer may approach the creation of a game object within that structural framework. Thus, a design framework for the disruptive game design philosophy has been proposed (i.e. the DisDev model) and is situated within an Integrated Model of *game space* providing a design tool usable in both industrial design contexts and academic research contexts (Chapter 6).

This fulfils the Research through Design (RtD) methodology’s (Section 2.4) requirement for awareness of problem (i.e. the need to explore the design potential of the disruptive game design philosophy within *game space*). A suggestion for how to explore the philosophy has then been made, supported by the Ludic Action Model (LAM), the DisDev model, and the Integrated Model.

Now the focus must move on to applying the design philosophy and design framework within the context of a live commercial game development project. This enables the completion of the remaining stages of the RtD methodology culminating in contributions to knowledge (Figure 7.1).

Stage 1 of the DisDev model (i.e. the assessment of project context) however indicates an omission in the RtD model that must first be addressed. In the RtD model (Figure 7.1), the initial output of the *problem contextualisation* only refers to the *research context*; how the problem is situated within the wider research landscape. There is no stage in this current RtD structure to account...
for the commercial context in which the game artefact will be designed, developed, and published. That is, the contextualising factors of the commercial development process (Section 6.4.1), the medium of games (Section 6.4.2), the game’s genre, mode, and milieu (Section 6.4.3), and the series, developer, or publisher history (Section 6.4.4).

This second context is critical to consider alongside the research context, as it will impact both the potential commercial success of the game artefact, as well as potentially influencing the ability of the game artefact to fulfil the research requirements of the project. Thus, any design decisions or issues encountered during the design and development process that stem from this commercial context must be documented and analysed, as they may provide important data for assessing the usability of the design philosophy and framework within a commercial context.

The RtD process model can be adapted (Figure 7.2) to take into account the need to understand the commercial context as well as the research context for a project. This additional stage in the model outputs the documented assessment of project context which provides additional data that will support the eventual knowledge contribution. The adapted RtD process model also defines the specific focus of each stage, moving from Research Context initially, to Commercial Context, then to the Commercial Design, Development, and Production processes, culminating in the Research Completion.

An academic account of a commercial design and development process serves as a case study for the application of the disruptive game design philosophy and design framework to a commercial game project. However, an ‘objective account’ of the creative process can only describe the design decisions, and the discussions that led to them, to the same depth as they were discussed during design and development. Such discussions were necessarily optimised so as to be useful...
within the constraints of commercial game development (i.e. they covered only what was necessary to solve the ‘commercial design and development problem’ in question) and thus, the objective account presented does not aim to present comprehensive academic reviews of the individual topics discussed.

7.1: Commercial Context of the Design and Development of the Game Artefact

The initial commercial situation that provided the starting point for design and development of the game artefact can now be defined and followed by an analysis of the previously described contextualising factors (Section 6.4) as they apply to this particular project.

7.1.1: Commercial Situation for the Project’s Initiation

In 2011, Swedish development studio Frictional Games (FG) (based in Helsingborg) entered into discussion with Brighton-based development studio The Chinese Room (TCR) to discuss the possibility of TCR taking on development of a second game in FG’s Amnesia franchise, to be an indirect sequel to Amnesia: The Dark Descent (Frictional Games, 2010) (Figure 7.3), hereafter referred to as A:TDD. A:TDD is a first-person horror-themed adventure game (although it is referred to as all of survival-horror, horror-adventure, and action-adventure by different distribution channels), developed for PC, MacOS, and Linux.

FG wanted to dedicate their small development team’s time to creating a new version of the studio’s bespoke HPL game engine (named after H.P. Lovecraft for its intended use in creating horror games) and thus, while they had funding to support development of a new game, they did not have the time or personnel available to develop one themselves. However, FG were receiving significant interest from their existing player base regarding a new game in the Amnesia franchise and thus, outsourcing development to an appropriate studio whilst FG maintained production and publication control was considered an appropriate solution.
FG felt that TCR held a complementary ethos toward game design that would operate well within the *Amnesia* mythos, with a focus specifically on allowing players to construct narrative along with environmental storytelling techniques. However, FG emphasised that they wanted TCR to bring their own brand of design, storytelling, writing, and gameplay to the new game and to create an *Amnesia* experience for players that was not just more of the same experience that players would have had from the first game in the series. This initial framing of the project’s commercial situation and intent was also discussed in an early public-facing interview with both TCR and FG, published on *Gamasutra* (Curtis, 2012).

**7.1.2: Balance of Creative Control between Development Studio and Publishing Studio**

FG were both the owners of the intellectual property (i.e. the *Amnesia* franchise and branding), as well as the providers of the funding for the project. The collaboration agreement between the two studios stated that FG would have final sign-off on project milestones and game features and would also be responsible for the final quality assurance process once the release candidate of the game was completed. Whilst FG were responsible for production and publication, the agreement also incorporated significant creative freedom for TCR with regard the design, prototyping, and development of game features. FG did not provide any specific guidance for what should, or should not, be present in the game in terms of game mechanics or story. They emphasised that the game did not even have to explicitly link to the story of *A: TDD*, thus being a sequel in name only. This creative freedom was intended to allow TCR to create a different type of player experience to that which players may have had in *A: TDD*. It also provided substantial scope for prototyping, testing, and developing a range of disruptive game features that would enable the current research to be carried out.

**7.1.3: Design Ethos of Development and Publishing Studios**

It is important to note that one of the key features of both FG’s and TCR’s design ethos is that the game mechanics should never be detrimental to the ‘flow’ of the game’s story and the player experience (Grip, 2011, Pinchbeck, 2011). Specifically, both companies approach the issue of the player-character dying during gameplay with caution, as forcing players to replay sections of the game if they fail directly impedes the player’s experiencing of the game’s story. Thus, design decisions made during design and development of this new game would similarly aim to prioritise a seamless gameplay experience for the player. This is emphasised in TCR’s initial pitch (Section 7.1.4) as well as in the design decisions made in relation to disruptive game features (Section 7.4).

**7.1.4: Initial Pitch, Project Funding, and Proposed Development Timescale**

Before FG released funding for the project, TCR provided a pitch that summarised the story and type of player experience that was being considered for the game (Appendix D). The working title for this pitched game was *We Are The Pig*. In the context of the current research, it is important to note that this initial pitching process was carried out by TCR’s creative director prior to the project being associated with the research into the disruptive game design philosophy (this design
work carried out prior to association with the current research is detailed in Section 7.3). However, the initial pitch was focused on the game’s story primarily, leaving scope for other components of the game, such as the mechanics, to be designed from the outset with the disruptive game design philosophy as a basis.

On the basis of the initial pitch, FG provided funding for a twelve-month development cycle, to cover the costs of TCR’s small development team of seven staff working full-time over that period. The aim of the project at the project initiation stage was to create a short two-to-three hour experience that would be released as a standalone expansion to A:TDD. The working title of We Are The Pig was changed at this stage to the game’s eventual release title of Amnesia: A Machine for Pigs (hereafter referred to as A:AMFP). FG’s funding and proposed development timescale did not cover the costs of creating large numbers of new art, sound, and visual effects assets and TCR were instructed to repurpose existing assets from A:TDD where possible to allow development time to be spent on creating game levels and game mechanics. To this end, TCR were also instructed to make use of the version of the HPL engine that was used to develop A:TDD (the HPL2 engine), although full source code access was provided by FG meaning that engine-level alterations and additions could be made if necessary. It is important to note however that while We Are The Pig was pitched as an expansion to A:TDD, FG were not looking for simply more gameplay in the style of A:TDD; the new game had to be notably differentiated from it.

As the project progressed, this initial scope for the size and length of the game and thus, the funding and development time required, increased. This is discussed further in Section 7.5.5. However, the initial design stage (i.e. Stage 2 of the DisDev model, or the creation of the game as designed) was carried out working within this initial, smaller-scale context.

### 7.2: Application of DisDev Model Stage 1: Assessment of Project Context

![Figure 7.4: DisDev model Stage 1: Assessment of Project Context.](image)

With the commercial context defined, it is now possible to discuss the implications of the different contextualising factors (Section 6.4) for the implementation of the disruptive game design philosophy. This is the assessment of project context as defined as Stage 1 of the DisDev model (Figure 7.4).
7.2.1: Implications of the Commercial Development Process

The collaborative arrangement between TCR and FG had implications for the implementation of the disruptive game design philosophy in the game. Furthermore, the small development team size placed limitations on what would be viable to achieve in the development time available.

7.2.1.1: Constraints of the HPL2 Game Engine Technology

TCR were provided with full source code access to the HPL2 engine, however the documentation provided by FG was minimal. Thus, it was apparent that much of the source code would require significant time investment by TCR’s programmer to interpret the code’s structure and functionality and then make any required additions or changes. The documentation that was provided did include some specifications for the engine, its library dependencies and its key functionality, although these were not comprehensive. This was due to FG developing the engine in-house with no previous intention of providing the source code to other studios, thus negating the requirement for documentation beyond that which was required internally for FG. The engine specifications that were assessed to be of particular potential usage as bases for disruptive game features, or as possible limiters of disruptive game features, are detailed in Appendix C.

This assessment of the game engine technology identified a constraint on the implementation of the disruptive game design philosophy. While all game components (i.e. Technology, Aesthetics, Story, and Mechanics) were readily editable through the development tools and the open-source Application Programming Interfaces (APIs), the HPL2 engine has no support for generating haptic feedback. The engine was created for use exclusively in PC-based titles where the primary form of input is expected to be mouse and keyboard, rather than ‘rumble-enabled’ controllers. This would thus prevent the design of any disruptive game features that make use of the haptic mode of output (Section 5.4), meaning that all disruptive game features would have to provide either a visual or auditory output (or a combination of both).

While it may have been possible for TCR to implement engine-level support for haptic feedback, this would have been potentially of only limited benefit to players, considering the likelihood of players not using compatible input devices. Moreover, the composition of TCR’s development team was not suited to making large, complex changes to the HPL2 engine’s source code.

7.2.1.2: Constraints of Development Team and Available Skill Sets

Ensuring that any disruptive game features (or indeed any other non-disruptive game features) could be viably created utilising the available development team’s skills and the available technology was a necessary consideration within the assessment of the commercial development process context.
<table>
<thead>
<tr>
<th>Core TCR Development Team Member</th>
<th>Development Role</th>
<th>High-level Skills Overview</th>
</tr>
</thead>
<tbody>
<tr>
<td>DP</td>
<td>Creative Director/Writer</td>
<td>Design/Story writing/Gameplay Scripting</td>
</tr>
<tr>
<td>Peter Howell</td>
<td>Game Designer/Gameplay Events Scripter (AngelScript)</td>
<td>Design/Level Design/Gameplay Scripting/VFX</td>
</tr>
<tr>
<td>MD</td>
<td>Engine Programmer (C++)</td>
<td>Engine Programming/Gameplay Scripting</td>
</tr>
<tr>
<td>WT</td>
<td>Lead Artist</td>
<td>Level Design/Art</td>
</tr>
<tr>
<td>BJ</td>
<td>Artist</td>
<td>Level Design/Art</td>
</tr>
<tr>
<td>SJ</td>
<td>Audio Design</td>
<td>Sound Design/Foley</td>
</tr>
<tr>
<td>JC</td>
<td>Director/Composer</td>
<td>Soundtrack Composition</td>
</tr>
</tbody>
</table>

Table 7.1: The Chinese Room’s development team structure for A:AMFP and associated high-level skills audit.

A team overview and high-level skills audit was conducted (Table 7.1 – other core team members are identified by initials only; audit does not include subcontracted staff, or staff that joined the team later in the development cycle). The team structure provided a relatively balanced skill set. The main weakness was the single engine programmer, which would restrict the extent to which the game’s core systems, such as physics and enemy agent artificial intelligence, could be potentially focused on as areas for implementing disruptive game features. This also placed constraints on how much reliance any disruptive (or non-disruptive) game features could have on bespoke code being added to the existing engine source code.

With multiple development team members having both design and scripting ability, the potential for implementing disruptive game features within the game’s event scripting was significant. This potential was further aided by having multiple level designers and artists that could provide rapid development of game environments and appropriate environmental assets to support such features.

7.2.2: Implications of the Series, Developer, and Publisher History
A:AMFP is an indirect sequel to A:TDD, meaning that players with existing knowledge of A:TDD will have initial expectations of A:AMFP based on their previous gameplay experience. Additionally, both the developer (TCR) and the publisher (FG) have a back catalogue of previous releases that may also form a basis for a player’s expectations of A:AMFP. In the case of FG, prior to A:TDD they released the Penumbra (Frictional Games, 2007-2008) series of horror-adventure games. TCR had previously released story-driven first-person adventure game, Dear Esther (thechineseroom,
as well as smaller mods for *Half-Life 2* (Valve Corporation, 2004) and *Doom 3* (id Software, 2004), in *Korsakovia* (thechineseroom, 2009) and *Conscientious Objector* (thechineseroom, 2008) respectively. Depending on an individual player’s knowledge, or lack thereof, of the two companies’ previous releases, their expectations may vary significantly which will in turn have an effect on how they respond to the experience of *A:AMFP*’s game as played.

Thus, *A:AMFP* was to be released to a potentially complex existing player base as well as to new players that may come to the game with little or no existing knowledge of the companies or previous titles. However, the commercial intention for the project was for *A:AMFP* to appeal primarily to players with prior knowledge of both companies’ previous games, while also being enjoyable for players coming to the game without such knowledge.

Disruptive game features thus needed to be considered during design in the context of the two broad player types that may potentially play the game with different expectations (i.e. the game as expected). For example, disruptive game features that targeted transludic knowledge that players with prior knowledge of previous games already have, would not be disruptive to players without that existing knowledge to be disrupted. Players without knowledge of prior games may still be affected by disruptive game features that target intraludic knowledge however, that would be constructed during the play of *A:AMFP*.

It is necessary to also consider the requirements of the research alongside the commercial intent of the project. The intention of the research is to design and implement disruptive game features in *A:AMFP* that provide potential for disruption, and hence cognitive engagement via higher-order thinking, to be experienced by as many players as possible. This thus requires the design and implementation of a range of disruptive game features that target a range of knowledge types. This is not to suggest that a game may not focus on implementing disruptive game features that target only transludic knowledge, for example (e.g. if the intention was to only have players with prior experience of a game series, franchise, genre, etc. experience disruption). However the current research requires a range of features to be implemented so as to increase the potential for usable data to be obtained from a range of players.

The consideration of disruptive game design in the context of the established *Amnesia* brand and the existing FG player base was identified as being likely to cause conflicting interests between the design aims of both companies, as well as the academic aims of the current research. While FG was willing to provide TCR with significant creative freedom, it was anticipated that disruptive game features may have greater difficulty in being approved by FG for inclusion in the final game. However, this potential challenge has already been identified as necessary to address in assessing whether the disruptive game design philosophy may have wider commercial potential or not.

### 7.2.3: Implications of the Medium of Games

The medium of games has its own established standards, or tropes (Section 6.4.2), that players expect to exist and function in particular ways across the majority of titles. In the particular case
of *A:AMFP*, the game’s heterodiegetic game mechanics (Section 5.2.4) were considered to have particularly significant potential for being disrupted. This included systems such as the game’s front-end menu screens, the game’s difficulty settings, the play modes available to the player when starting a game, and the ability for players to use common game functions such as pausing the game, saving games, and loading games.

The game’s front-end menu screens and systems that players had to navigate through to start a new game initially were evaluated as being likely to need to remain non-disruptive, as there would be no benefits to player enjoyment by purposely making this initial process require higher-order thinking. However, systems enabling functionality such as pausing, saving, or loading games appeared to have potential for forming the foundations for disruptive game features, potentially linked to different game modes or difficulty levels (e.g. disabling the player’s ability to save, or placing usage rules on the save system on the game’s higher difficulty settings).

7.2.4: Implications of the Game's 'Genre', 'Mode', and 'Milieu'

*A:AMFP* is a game artefact within the broader medium of ‘games’. However, it is the specific properties of ‘genre’, ‘mode’, and ‘milieu’ (Section 6.4.3), that provide a greater range of established tropes that may form the bases of disruptive game features.

7.2.4.1: Implications of the Game’s ‘Genre’

*A:AMFP* is a horror-themed adventure title. While ‘adventure’ is a widely recognised ‘genre’ label (Whalen, 2004) within games, ‘horror’ contextualised within games, is not. In games it is more common to refer to ‘horror’ games as ‘survival-horror’. However, it remains difficult to suggest what may define ‘survival-horror’ as a ‘genre’ within the previous definition of ‘genre’ (Section 6.4.3) as ‘the configuration of heterodiegetic and homodiegetic game mechanics’. Such games are not defined only by their mechanical components, but require a significant combination of specific milieu-based and mode-based components as well. It is necessary for designing disruptive game features that target different types of knowledge (and specifically, transludic and extraludic knowledge) to define the genre in which the game is situated, as this will have an effect on the initial expectations a player may form about the game.

It is more pragmatic to refer to survival-horror as a sub-genre, as per Whalen’s (2004) definition of the term. Referring to King and Krzywinska’s (2002) terms, genre, mode, and milieu (Section 6.4.3), Whalen states that a sub-genre definition “describe[s] different game qualities in more or less equal proportion and [these] three properties [genre, mode and milieu] combine to form a practical sub-genre” (2004, p.292). Thus, a ‘genre’ can be defined as a particular configuration of heterodiegetic and homodiegetic game mechanics and a ‘sub-genre’ as a particular configuration of a ‘genre’, along with a particular configuration of mode and milieu (which can be referred to collectively as the ‘theme’).
<table>
<thead>
<tr>
<th>Genre</th>
<th>Mode &amp; Milieu (Theme)</th>
<th>Sub-Genre</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action-Adventure</td>
<td>+ Horror</td>
<td>= Survival-Horror</td>
</tr>
<tr>
<td>Adventure</td>
<td>+ Horror</td>
<td>= Horror-Adventure</td>
</tr>
</tbody>
</table>

Table 7.2: Combining genres with themes to create definable sub-genres.

Survival-horror games in this definition are therefore a sub-genre of action-adventure, which itself is a modified form of the adventure genre. A:AMFP, following this same approach, can be defined as a horror-adventure game, as it combines horror-themed mode and milieu components with heterodiegetic and homodiegetic mechanic components from the adventure genre (Table 7.2). It should be noted that A:AMFP has subsequently been independently categorised, post-release, as a horror-adventure title (GamersGate AB, 2014).

The adventure genre, using Wolf’s (2012, p.12) definition as a basis, consists of games ‘set in a game world of interconnected locations framed by a story, requiring players to navigate, explore and locate tools as the principle drivers of gameplay, so that those tools may be used individually or combined in unique ways to achieve a series of objectives’. These features each have the potential to form multiple bases for disruption, in terms of the types of objectives players must achieve, the types of tools they must locate and how those tools must be utilised during gameplay. Similarly, the activities of navigation and exploration may be disrupted by manipulating the structure and properties of the game environments, or the movement properties of the player-character.

It is important to note however, that the commercial context of A:AMFP as a sequel to A:TDD may affect how players form ‘genre’-based expectations of the game. A:TDD is, for example, predominantly defined through retail channels and in critical reviews as a ‘survival-horror’. While A:AMFP is intended to minimise ‘survival’-based game features in favour of ‘adventure’-based game features, it is possible that players with knowledge of A:TDD will base their initial expectations of A:AMFP on the genre and genre-based properties of A:TDD due to them belonging to the same game series. This in turn may lead to a significant mismatch between the game as expected and the game as played which may not always be beneficial to the player’s enjoyment of their game experience.

7.2.4.2: Implications of the Game’s ‘Mode’
Frequently cited mode-based components in horror-themed games include heterodiegetic aesthetic features such as the perspective the game is played from (i.e. the game’s ‘camera’) and heterodiegetic technology, such as the game’s input configuration (i.e. how the player controls the avatar in the game). However, A:AMFP’s mode-based properties are constrained by the commercial context. The HPL2 engine is a first-person game engine, thus preventing any other perspectives being considered for use. While the game is being developed for PC-based play, it is necessary to also provide support for control pad input as well. Thus, the modes of input must be
limited to the type and number that are available on a ‘standard’ control pad (a wired Xbox 360 control pad was used as the ‘standard’ layout in this particular case, on the request of FG). From a disruptive game design perspective however, there is potential to disrupt the first-person perspective and the constrained input modes while operating within the restrictions of the HPL2 engine. A simple example would be to change the control mappings for different in-game actions under different circumstances during play (although such disruption would be likely to be more frustrating than enjoyable for players).

In addition to the camera perspective and controls, audio is also cited as being of particular importance in horror-themed games (see for example, Roux-Girard (2010) and Usher (2013)) due to its ability to create emotion without explicitly showing the player anything ‘horrific’. While audio, in a homodiegetic Aesthetic form, may also be part of the game’s milieu, there is potential for disruptive game features to target heterodiegetic Technology-based audio properties too. For example, manipulating the speakers that audio is played through (e.g. swapping the left and right outputs in a stereo speaker arrangement) can be used to disrupt a player’s knowledge of the virtual space and the direction of potential threats.

Thus, while the commercial context of A:AMFP places significant limits on the game’s ‘mode’, it does not eliminate the potential for disruptive game features to be implemented that operate on mode-based game properties.

7.2.4.3: Implications of the Game’s ‘Milieu’

The common features of horror milieu in games draw on common features of horror milieu in cinema and literature. As Krzywinska (2002, p.208-209) writes, many existing horror games draw on familiar filmic horror themes and settings, such as the ‘zombie horror’ (e.g. the Resident Evil series, The Walking Dead (Telltale Games, 2012), State of Decay (Undead Labs, 2013), and ZombiU (Ubisoft Montpellier, 2012)), which locates itself in familiar apocalyptic settings such as destroyed science laboratories, ruined city streets and buildings, and abandoned, infested rural areas. These games exist alongside titles that draw upon “more traditional gothic and Lovecraftian themes” (Krzywinska, 2002, p.209), such as Call of Cthulhu: Dark Corners of the Earth (Headfirst Productions, 2005), Eternal Darkness: Sanity’s Requiem (Silicon Knights, 2002), and the Alone in the Dark series. Alternatively, there is the ‘aliens-in-space’ theme, which includes titles such as System Shock 2 (Irrational Games and Looking Glass Studios, 1999), Alien: Isolation (The Creative Assembly, 2014), and Enemy Zero (WARP, 1997), and graphic, gore-based horror, including Saw (Zombie Studios, 2009) and The Evil Within (Tango Gameworks, 2014). With such an established range of milieu-based components to draw upon from existing games, there are a range of possibilities for implementing disruptive game features that target a player’s transludic knowledge.

From the initial pitch of We Are The Pig, A:AMFP’s story and thus, its milieu and its internal story logic, aimed to draw significantly on the Lovecraftian mythos and, more broadly, the literary philosophy of cosmicism that presents the concept of an indifferent, unknowable universe in
which humanity pales into insignificance (Mariconda, 1991). This focus is also present in the preceding *Amnesia* game, *A:TMDD*. A key component of this mythos and broader philosophy is the theme of sanity and especially, the sanity of a story’s main protagonist. The initial pitch for *We Are The Pig* included references to the mental state of the game’s protagonist, thus there was scope for developing disruptive game features that co-opted this theme and utilised it in different ways.

For example, it was noted that while the ‘theme’ of sanity and altering one’s perception of reality is commonplace within a range of horror-themed games, the *ludic consequences* of a character ‘losing sanity’ have the potential to be disruptive in nature. Manipulating the aesthetic output of the game may have a disruptive effect on the player via Encoding Disruption (e.g. if the manipulation makes interpreting in-game information difficult). Altering the results of actions that the player takes, dependent on the player-character’s sanity, may result in Action Plan Disruption. Implementing semi-randomised modifications to established in-game entities, again dependent on the player-character’s sanity, may provide a source of Recall Disruption.

However, it was also identified that it would be important for such disruptive game features to be significantly randomised (either in terms of their occurrence chance during gameplay, or in terms of their effect on the player experience) so as to mitigate the likelihood of players being able to construct and refer to accurate knowledge about them. Being able to do so would result in players being able to rely on lower-order thinking and thus, prevent the game features being disruptive.

### 7.2.5: Summary of DisDev Stage 1: Assessment of Project Context for *Amnesia: A Machine for Pigs*

The assessment of project context carried out for *A:AMFP* can be summarised within a customised DisDev model for the project (Figure 7.5). This customised model is presented in full, following the discussion of individual stages, in Section 7.5.15 (Figure 7.70 and Figure 7.71).
Figure 7.5: Stage 1 of the Disruptive Game Feature Design and Development (DisDev) model applied to Amnesia: A Machine for Pigs.

This model summarises the key outcomes of the process in terms of potential targets for disruptive game features, as well as any significant limitations on how the disruptive game design philosophy may be applied to the game. These outcomes will serve as ‘guiding information’ that will in turn, place limitations on how the disruptive game design philosophy may be instantiated within the game as designed and the game as created.

7.3: Application of DisDev Model Stage 2: Initial Non-Disruptive ‘Framework’ for the Game as Designed

Individual game projects will have different design aims. Even if two games are both utilising the disruptive game design philosophy, they may individually be focusing on disruption of different game components (i.e. Technology, Story, Mechanics, or Aesthetics). Thus, it may in some cases be necessary to firstly identify and define the non-disruptive ‘framework’ of the game around which the disruptive game features then operate.
In some projects, the commercial context for the project may dictate certain requirements for what features of the game may or may not be targeted for disruption. In the case of AAMFP there were few limitations imposed by FG as the publishers. However, TCR’s creative director had already constructed a number of details about the game’s setting, plot, and characters, prior to the commercial game project being linked to the current research project. Thus, while these details were still able to be discussed and modified throughout development, the fundamental game properties could not be completely altered for the sake of the research project. Hence, the ‘non-disruptive framework’ for AAMFP (i.e. the left channel of ‘non-disruptive game features’ in Figure 7.6) consists of the game’s story content, its general setting (i.e. where and when the game takes place), and the general properties of the key characters (both the player character and important non-player characters).

As outlined in the Research through Design (RtD) process model, the output of the design stage is design documentation. The full design document for AAMFP is presented in Appendix E, however to provide appropriate contextual information for the following discussion of disruptive game features (Section 7.4), an overview of the non-disruptive ‘framework’ for the game is outlined in the
following sections (Section 7.3.1, Section 7.3.2, and Section 7.3.3). These sections are drawn from the original game synopsis documents developed following FG’s acceptance of the We Are The Pig pitch and are presented here in their original wording (section numbers adapted for consistency).

**7.3.1: Setting**

Figure 7.7: Victorian London skyline concept artwork. © Ben Andrews, used with permission.

The game is set on New Year’s Eve, 1899, in Victorian London (Figure 7.7). The game moves through a number of environments, from a plush townhouse, through the streets and alleys of London, into a cavernous ‘steam punk’ factory complex and beyond into the maze of sewers beneath. Finally, the game concludes in a subterranean Aztec temple beneath the city.

**7.3.2: Characters**

**7.3.2.1: Oswald Mandus**

Figure 7.8: Reference used in development as the basis for the character of Oswald Mandus. Image originally obtained from Moda Historica (http://modahistorica.blogspot.co.uk/)

The player takes on the role of Oswald Mandus (Figure 7.8) (referred to simply as ‘Mandus’ during much of the game), who is a wealthy and forward-thinking industrialist. Having lost his wife during childbirth he has struggled to raise his twin children alone. His business has not gone well. His investment in the latest steam-driven abattoir machinery has left him in poverty.
and facing financial ruin. In desperation, he follows a cryptic passage found in one of his great uncle’s diaries regarding a source of great riches in the newly discovered temples of Mexico.

Following a trip to Mexico with his children, he returns home and falls into a fever brought on by a disease contracted during the trip. The game begins with him regaining consciousness, with no memory of recent events and only a sense that his children are in danger. Finding Mandus’ children is the primary motivation early in the game.

As the game progresses, the player discovers evidence of the man that Mandus once was: greedy, arrogant, and utterly ruthless in his pursuit of power and money, obsessed with progress. Underneath this however, it becomes evident that this dark character was driven by the desperate love of his children and his dedication to creating a better future for them. The game’s plot centres on Mandus coming to terms with the evil he has committed in an insane attempt to secure this better future, and his eventual redemption as he strives to put right the terrible forces he has unleashed.

7.3.2.2: The Machine

Connected to the mysterious force that Mandus discovered in Mexico, the Machine is a vast, sprawling complex, growing ever larger beneath the streets of London. While Mandus was in his fever state, it has been spreading like a tumour through sewers and tunnels, further expanding its vast, steam-driven engines of slaughter. At its heart rests an unworldly force, a consciousness of barely suppressed and malignant power.

The Machine is the dark heart of industrialisation and Empire given life; a creature of steam, cogwheels and metal, but with a disturbingly human character. It is a Victorian version of the great evil computers of science fiction, such as HAL9000. It has a severe insecurity complex, delusions of godhood, as well as an innate understanding of meat processing.

Figure 7.9: Representation of Huitzilopochtli, the Aztec Sun God.

Image Licensed under Public Domain via Wikimedia Commons
The Machine believes it is the incarnation of Huitzilopochtli, the Aztec sun god (Figure 7.9), and it thinks it has a mission to save the world. Mandus gave the Machine life by sacrificing his children on the steps of a lost temple in Mexico; an act committed with the intent of saving a world he saw falling into corruption and ruin. Mandus does not remember this upon recovering consciousness however, thinking instead that his children are in danger and that they can be rescued – a result of the fever-induced titular amnesia.

The Machine has taken this view of a world falling into ruin very literally, and fused it with the method of saving the world that the Aztecs developed: human sacrifice. Sacrifice in large numbers. Sacrifice in huge numbers. It believes that by sacrificing the entire population of London it can avert the apocalypse that it thinks is imminent in the coming 20th century.

The Machine is, on one level, a classic moustache-twirling Victorian villain. It gloats, it crows, it sees itself as a greater being with the right to kill and maim indiscriminately. However, the transition the game aims to create is a realisation by the player as the Machine becomes vulnerable in the final levels, that it is only doing what its “Daddy”, Mandus, wants, and that it genuinely thinks it is saving the world. The game aims to leave the player with an uneasy sympathy for this clockwork, atomic monster.

7.3.2.3: The Professor
A slightly buffoonish character sent by the Ministry to investigate the strange, secretive and sudden rise of Mandus Industries from a ruined parochial butchery company to the dominant meat processors of London. The Professor is shot through with the conceits of the age: an unwavering belief in progress, a disregard for anyone not white, male and upper-class, and an almost sordid love of science and technology.

7.3.2.4: Edwin and Enoch

Figure 7.10: In-game screenshot of Edwin and Enoch, following the revelation of Mandus’ actions in Mexico.
Mandus’ twin boys (Figure 7.10), aged about 9 years old and trapped, so Mandus believes, somewhere in the depths of the Machine. Edwin is the more forceful and mischievous of the two, Enoch is crippled and shy. Mandus sees and hears glimpses of his boys from the opening of the game, pursuing them ever deeper into the heart of the factory.

7.3.2.5: The Manpigs

![Early concept artwork of the Manpig.](image)

The Machine’s gruesome work must be maintained by slaves and what better way to ensure a constant supply than by reanimating the very output of the Machine itself. The Manpigs (Figure 7.11), part broken and twisted pig carcass and part recycled human components can be found throughout the Machine and supporting infrastructure. However, while they may appear aggressive, single-minded slaves working towards the continued sacrifice of further victims, they may not all share the same tendencies. Even following the reanimation process, some dregs of humanity and splintered memories of who they once were may remain.

7.3.3: Plot

Mandus awakes from his fever to hear the sounds of his children and with a sense that something awful has happened. It is dark, he remembers nothing of where or indeed when it is, or why his bed appears to be caged. Drawn by the sound of the boys, he travels through the house, and is startled by a strange earthquake and the sounds of machinery. He discovers a secret passage in the house where someone seems to have been spying in a sinister way upon the occupants. Descending to the ground floor, he finds further disturbing secret rooms that open a path to a bizarre space full of conveyor belts and hydraulics. He finds a telephony device ringing as if waiting for him, where a voice (The Machine, but the player does not know this yet), tells him that his children are trapped in the Machine, which is flooding, and that he must journey deeper into it to save them.
Mandus follows the trail through a cellar and factory complex, outside into a London back alley and finally into a church. The church has been blasphemously re-purposed for the worship of a bastardised, Mesoamerican cult of pig slaughter (Figure 7.12). Mandus finds a secret passage and discovers a horrible kind of prison where people are being stored for ‘processing’. He also stumbles upon and is nearly captured by a vicious and terrifying creature, seemingly stitched together from both human and animal parts. Escaping, he enters the main factory complex, where he finds evidence that the Machine has been deliberately sabotaged. His ‘ally’ telephones again and instructs him to pass through the laboratories to the main bilge pumps to drain the flood water and restart the machine’s central boiler to free his children.

Mandus heads deeper, harried by the creatures and gradually fixing the machine components he finds to gain deeper access. After overcoming many challenges and avoiding the patrolling creatures, he finds the main pump and drains the last of the floodwater. This allows him to enter the reactor chamber, where the player discovers the Machine appears to be a form of strange and unholy Victorian nuclear device. Mandus restarts the reactor, unleashing the Machine. At this point, Mandus sees his children and realises they are already dead. He also realises the Machine is responsible, and that he has been duped into restarting it. The Machine tells Mandus that he is the one who has caused everything, but Mandus refuses it, and says he will destroy what he has created. In return, the Machine attempts to kill Mandus, but he escapes into the factory engine rooms.
Around this time, two things are becoming apparent. The first, that the Machine is not being strictly literal when it speaks of 'pigs' and the second, that it has a plan; a very grand plan that involves many, many pigs being processed in order to save the world.

Figure 7.13: Tesla Field concept artwork. © Ben Andrews, used with permission.

As Mandus escapes from the engine rooms, he emerges into a London awash in flames, screams and panic. The Machine has unleashed its creatures onto the streets to herd the population into its grasp. Mandus, vowing to stop it, is pursued across London back to the Factory, where he gains access to the pig-line (the Machine's primary processing area) in order to strike at the Machine's very core. He journeys through the full horror of the abattoir he has unwittingly constructed below the streets of London but manages to fight through to the Machine's vast steam-powered heart, the Tesla field (Figure 7.13). This is a vast network of towers, filled with strange and advanced machinery that is generating an enormous amount of power for the Machine.
As he crosses the field, the Machine drops its bombshell – that it was resurrected by Mandus sacrificing his own children. Shocked, Mandus resolves to destroy them both. He descends to the centre of the Machine, at this point sensing he is many miles below the city streets, although his precise location at this point is impossible to know for sure. He emerges into a vast subterranean chamber that is housing an Aztec temple (Figure 7.14), unnervingly similar to the one Mandus found in Mexico. Ascending the stairs to the altar at the temple’s summit, Mandus finds the bodies of his children, their hearts providing the Machine’s life force.

As the Machine pleads for its life, desperately trying to argue that the slaughter it has unleashed is nothing compared to the horrors of the coming century, Mandus climbs the altar into an empty throne at its centre. Realising now that the Machine is intrinsically linked to Mandus, he activates the machinery surrounding the throne, impaling and killing himself along with the monster he created.

The lights slowly go out throughout the Machine and the complex. Echoing through the empty corridors, the sound of the Manpigs singing to one another can be heard, as the dust settles on Mandus’ open eyes.

Miles above, the city turns over in its sleep, and the 20th century begins…
7.4: Application of DisDev Model Stage 2: Disruptive Game Features in the Game as Designed

Disruptive game features are created during the game as designed (Figure 7.15). Not all of these features will be retained in the game as created (Figure 7.15) following the prototype development process (Figure 7.15). However, these features demonstrate the differences between the game as designed and the game as created and are the means of providing comprehensive design process documentation as per the output requirements of the RtD process. These disruptive game features are designed via the right-hand channel in Stage 2 of the DisDev model (Figure 7.6).
7.4.1: The ‘Infection’ System

The ‘infection’ system was a core component of the A:AMFP design document (Appendix E) created following FG’s acceptance of the We Are The Pig pitch (Appendix D). The system itself was designed to provide similar effects on gameplay as A:TDD’s insanity system (detailed below), whilst also providing a link between the gameplay and the story’s themes of dirt, sanitation and disease.

![Image of infection system effects in A:TDD](image)

Figure 7.16: Insanity system effects in A:TDD when looking at an enemy agent.

A:TDD’s insanity system applied a range of screen filters and screen effects that would impair the player’s vision whenever the player-character witnessed an ‘unsettling’ or ‘disturbing’ event, or whenever the player-character looked at an enemy agent (Figure 7.16). The degree of vision impairment in A:TDD was not significant enough to be particularly debilitating to the player, consisting primarily of modifications to the player’s field of view (FOV), the use of an ‘image trail’ effect (keeping previously drawn frames on screen for a few additional frames to create a blurring effect), and a radial blurring effect around the edges of the screen.

![Image of damage in Battlefield: Bad Company 2](image)

Figure 7.17: Receiving damage in Battlefield: Bad Company 2.

A primary point of reference during the design of the infection system was the use of damage indicators and on-screen blood spatter that can be seen in a range of first-person and third-person titles. Battlefield: Bad Company 2 (EA Digital Illusions CE, 2010) for example fades in a red, bloody screen effect as the player takes damage from enemies (Figure 7.17).
However, rather than using the screen effects primarily for aesthetic purposes, or as a simplistic warning that the player-character is being injured, the intention in A:AMFP was to make the screen effects significantly more debilitating for the player. The effects would be linked to the player-character becoming infected, either through contact with unsanitary objects in the game world or through being attacked by enemy agents, which would attack the player-character with bodily fluids or by vomiting on him. In line with A:AMFP’s focus on ‘adventure’ gameplay rather than ‘survival’ gameplay (Section 7.2.4), occurrences of the player-character becoming infected would be infrequent and almost always avoidable if players were vigilant of the game environment and careful in their actions. However, becoming infected would have a significant impact on a player’s ability to progress and require immediate treatment to avoid the player-character eventually succumbing to the infection and dying. Treatment items (i.e. syringes of medicine) could be located with relative ease by players that took time to explore the game environments, rewarding players for engaging with the game’s ‘adventure’ focus.

The principle disruptive design aim with this system was to present a highly oppressive, aggressive game feature to players that could have an immediate significant impact on their gameplay. The disruptive nature of the system was designed to produce Encoding Disruption as it impairs the player’s ability to accurately interpret and encode information about other game components (i.e. the screen effects are a simultaneously distracting stimulus). However, the screen effects only disrupt a player’s ability to encode visual information. There is potential therefore via this disruptive game feature to encourage players to engage more deeply with analysis (i.e. the lowest level of higher-order thinking) of auditory stimuli as a means of compensating for their impaired vision.

Figure 7.18: DisDev Stage 2 elements forming the design for the 'Infection' System.

The design for the ‘infection’ system in A:AMFP can be formalised within Stage 2 of the DisDev (Figure 7.18) model as a disruptive game feature that targets Aesthetics, with a Visual mode of output, aiming to create Encoding Disruption of Intraludic Knowledge (INT-P, INT-S, and INT-E).

7.4.2: Emulating Peripheral Vision

The possibility of implementing a game system that could emulate, or use the properties of, the human eye’s peripheral vision was suggested during the design process. More specifically, it was
proposed that the properties of being able to perceive light more clearly through peripheral vision than ‘normal’ (central) vision and of being able to detect movement more easily in peripheral vision, could be interesting concepts to implement in a game. Such a system could also provide significant Recall Disruption of transludic procedural, semantic and episodic knowledge, as it is not something that has been used previously in mainstream games.

At an implementation level, this would mean applying screen effects to simulate the different viewing properties of ‘normal’ and peripheral vision which, as Hecht (2002) summarises, would include the receptive capabilities of ‘rods’ and ‘cones’ in the eye structure such as colour, light levels, motion and definition.

This system could enable potential new types of gameplay to be considered. The concept that could apply particularly well to the context of A:AMFP was the implementation of enemies that could only be clearly seen by placing them in the player-character’s peripheral vision, near the edges of the screen. Thus, to effectively track such enemies and avoid them, the player would have to approach the game very differently to the majority of first-person games, in which the centre of the screen is usually the focal point. Gameplay in a first-person-shooter involves the aligning of enemy agents with the crosshair at the centre of the screen before performing the appropriate action (i.e. usually, pressing the appropriate input to pull the weapon’s trigger) to remove those agents from the game world (Pinchbeck, 2007).

The proposed peripheral vision system disrupts that transludic schema via Recall Disruption of procedural knowledge (e.g. methods of interacting with a typical first-person game), as well as via Action Plan Disruption by changing the semantic properties of enemy agents and the player’s field of view (FOV), thus making the edges of the screen much more important to gameplay than players may expect. This in turn will require players to undertake significant construction of new intraludic knowledge (i.e. creation; the highest level of higher-order thinking) to understand and respond to how enemies appear on screen (i.e. new semantic knowledge construction) and what actions need to be performed to successfully defeat or avoid them (i.e. new procedural knowledge construction).

![Figure 7.19: DisDev Stage 2 elements forming the design for the emulation of peripheral vision.](Image)
The design for the emulation of peripheral vision can be formalised within Stage 2 of the DisDev (Figure 7.19) model as a disruptive game feature that targets Aesthetics, with a Visual mode of output, aiming to create Recall Disruption and Action Plan Disruption of Transludic Knowledge (TRANS-P and TRANS-S).

7.4.3: Procedurally Generated Tesla Field ‘Maze’

The procedurally generated ‘maze’ was designed to be a key feature of the Tesla Field that the player must traverse to reach the Aztec temple.

As a logical system, the design of the ‘maze’ was simple. The player would be tasked with navigating through a series of fifteen square rooms, with doors on each wall. The rooms would be procedurally generated so that, each time the ‘maze’ was played through, they would be connected in different formations and the contents of the rooms would be altered. At set points during the sequence of fifteen rooms, players may encounter a scripted room where a particular action needed to be performed, such as solving a puzzle or, if the rooms were larger, avoiding an enemy agent. Regardless of the route players took through the ‘maze’ (i.e. regardless of which of the four doors players used to exit each consecutive room), after going through the sixteenth door, they would always successfully emerge at the exit.

![Figure 7.20: The 'Lost Woods' consists of a room-based, door selection puzzle with a single correct solution.](image)

Thus, the structure of this environment was in fact a linear labyrinth, designed to be presented as and responded to by players as, a ‘solvable’ maze. This design was intended to provide disruption of players’ transludic semantic and episodic knowledge related to similar environment structures in other games, whilst simultaneously supporting the design ethos of both FG and TCR in ensuring that gameplay never detracts or gets in the way of the player’s experience of the game’s story and world (Section 7.1.3). Mazes similar to this scenario are not uncommon in existing games; for example, The Legend of Zelda: The Ocarina of Time’s (Nintendo EAD Group No. 3, 1998) ‘Lost Woods’ area (Figure 7.20), or Paper Mario’s (Intelligent Systems, 2001) ‘Forever Forest’ area.
These existing examples similarly present sequential ‘rooms’ with four possible exits but both have single correct solutions; taking an incorrect exit returns players to the entrance. Players must identify a particular piece of information within the game that can be used to allow them to successfully navigate the maze. The solution to the ‘Lost Woods’ is found by listening to the volume of the background music as the player-character approaches each exit from a room; the exit that is accompanied by the loudest music is the correct exit. The ‘Forever Forest’ provides clues to the player through a variety of environmental clues that the player must interpret.

The procedural nature of the ‘maze’ proposed for A:AMFP combined with its inevitable ‘solution’ after passing through fifteen rooms was designed to provide a scenario that could disrupt a player’s existing transludic semantic and episodic knowledge of similarly structured mazes in games. However, by presenting the ‘maze’ in a way that clearly co-opted transludic knowledge of similar scenarios in other games, it was designed to encourage players to engage in higher-order thinking via analysis and evaluation of the environment and their decisions, even though such analysis and evaluation is not actually necessary to make progress. Players referring to their established transludic knowledge may expect to find a meaningful clue in the game environment as to whether the doors they are selecting are correct (i.e. they may expect entities in the environment to provide semantic information). With such information unavailable, the design aimed to encourage players to analyse the environments in more detail than they may otherwise.

The design for the procedurally generated Tesla Field ‘maze’ can be formalised within Stage 2 of the DisDev (Figure 7.21) model as a disruptive game feature that targets Mechanics, with a Visual mode of output, aiming to create Recall Disruption and Action Plan Disruption of Transludic Knowledge (TRANS-P, TRANS-S and TRANS-E).

This particular disruptive game feature was identified to have some potential risks with regard to the player’s overall enjoyment of the game experience. Firstly, players may become frustrated if they are unable to identify any ‘clues’ or other information in the environment to provide feedback on their decisions. This however could be counteracted by including a number of false pieces of information (e.g. some doors have symbols on them, or some have different coloured...
lights on them). This may mitigate a player’s frustration at having no feedback while encouraging further analysis and evaluation in an attempt to identify meaning in the false information.

Secondly, players without relevant transludic knowledge, or players that fail to identify the similarities between the ‘maze’ and similar scenarios in other games, will not experience any disruptive effect and may simply find the scenario confusing. However, because the player will always ‘solve’ the maze after going through the sixteenth door, as long as the player is encouraged to continue exploring, this confusion should not last for a significant length of time. Playtesting could further refine this disruptive game feature, identifying how players respond to it (e.g. whether they perceive it as a ‘solvable maze’ or not), as well as the ‘length’ (i.e. number of rooms) that is acceptable for the majority of players without resulting in frustration, boredom, or other negative responses. If necessary, fewer rooms could then be used allowing players to successfully ‘solve’ the maze more quickly.

7.4.4: Personality-Driven Enemy Artificial Intelligence

The initial design of the game’s enemy artificial intelligence system contained three unique sets of behavioural controls. There was only one visual enemy style, however every enemy agent in the game would be assigned one of three possible ‘personalities’, referred to in the game’s code as the ‘Rod’, ‘Jane’ and ‘Freddy’ personality types. These personalities each had a different set of behavioural rules, thus allowing enemy agents that may otherwise appear identical to behave very differently to one another.

<table>
<thead>
<tr>
<th>Enemy Agent Personality Type</th>
<th>Primary Behavioural Traits</th>
</tr>
</thead>
</table>
| ‘Rod’                        | • Will maintain a ‘safe’ distance from the player-character.  
                                • If unable to do so, will approach player character, investigate them (by getting close and smelling them), before continuing its patrol. |
| ‘Jane’                       | • Will maintain a ‘safe’ distance from player-character, whilst observing the player-character’s movements.  
                                • If unable to maintain ‘safe’ distance, will panic and flee.  
                                • If cornered and unable to flee, will attack and knock player-character to floor, then flee.  
                                • Will only attack and kill player-character as a last resort. |
| ‘Freddy’                     | • Will actively hunt the player-character.  
                                • Will attack and kill them if given the opportunity. |

Table 7.3: Overview of proposed enemy agent personality types and key behavioural traits (Pre-Development, December 2011).
The different rules produce three main types of behaviour (Table 7.3). The design of this disruptive game feature aimed to create, primarily, Recall Disruption of transludic and intraludic procedural, semantic, and episodic knowledge. The feature also had the potential to create Action Plan Disruption of intraludic semantic knowledge. Utilising only a single visual style for all enemies in the game prevents players relying on visual recognition to decide upon how to approach an enemy. Thus, players that associate sequences of actions with the semantic properties of the enemy agents may find that the semantic knowledge that was previously correct becomes incorrect when recalled in response to an enemy with a different personality type. If players carry out actions based on this incorrect knowledge, they will experience Action Plan Disruption.

Where many games may use visual cues to aid players in identifying enemy types or properties, this system would allow A:AMFP to differentiate enemies via a less overt method. This could disrupt transludic-knowledge-based expectations of how an apparently single enemy ‘type’ may behave in a predictable manner in every encounter. It could also disrupt a player’s constructed intraludic knowledge of enemy behaviour in A:AMFP, as one encounter with an enemy may result in behaviour that is not exhibited in an otherwise comparable encounter (in terms of the overt audio-visual properties of the encounter) later on, making any recalled knowledge inaccurate and any action decisions made based upon it potentially inappropriate.

This disruptive game feature was designed to encourage players to engage in more consistent analysis and evaluation of the game environment as they play. By making enemies unpredictable, the design aimed to require players to identify the greater importance of knowing key environmental features, such as hiding places, as well as for players to frequently experience Recall Disruption relating to procedural knowledge for handling enemy agents in the game world.

![Figure 7.22: DisDev Stage 2 elements forming the design for the personality-driven enemy artificial intelligence.](image)

The design for the personality-driven enemy artificial intelligence can be formalised within Stage 2 of the DisDev (Figure 7.22) model as a disruptive game feature that targets Mechanics, with a combined Visual and Auditory mode of output (i.e. the enemy agents provide both modes of stimuli), aiming to create Recall Disruption and Action Plan Disruption of Transludic Knowledge (TRANS-P, TRANS-S, and TRANS-E) and Intraludic Knowledge (INT-P, INT-S, and INT-E).
7.4.5: ‘Hardcore’ Difficulty Mode

The concept of a ‘Hardcore’ difficulty mode is not an uncommon feature in games. A particularly notable example can be found in Dead Space 2 (Visceral Games, 2011). In this game, Hardcore mode limits the player to using only three saves throughout the entire game whilst also increasing enemy difficulty and decreasing the regularity of ammunition, health, and credits dropped by enemies or found in the environment.

During initial design discussions, it was suggested that A:AMFP contain its own Hardcore mode but that, given the short length of the game, a game mode such as this could potentially be more punishing. The aim of such a game mode would be to disrupt the player’s potential transludic episodic and semantic knowledge of ‘Hardcore’ modes in other games. To do this, A:AMFP’s mode could increase the damage that enemies were capable of inflicting, making most enemies able to kill the player character in two hits. Additionally, all saving could be disabled, including automatic checkpoints, meaning that death of the player-character would require a complete replay of the game from the beginning (i.e. ‘permadeath’). Lastly, this mode could also disable all pause functionality meaning that players could be vulnerable to attack when leaving their computer. However, to avoid the game becoming ‘abusive’ (Wilson and Sicart, 2010) by preventing a player’s ability to satiate their low level physiological needs (e.g. eating and sleeping), there would be opportunities during play to find ‘safe areas’. However it would be up to the player to analyse the game environment and their intraludic knowledge to identify these safe areas.

To further increase the disruptive potential of this mode, some mode-specific events and enemies could be added or changed. This could limit the potential for players to refer to intraludic semantic and episodic knowledge about, for example, enemy locations, constructed during play of the game in non-hardcore mode.

Figure 7.23: DisDev Stage 2 elements forming the design for the ‘hardcore’ difficulty mode.

The design for the ‘hardcore’ difficulty mode can be formalised within Stage 2 of the DisDev (Figure 7.23) model as a disruptive game feature that targets Mechanics, with a combined Visual and Auditory mode of output (i.e. multiple outputs will change in ‘hardcore’ mode), aiming to create Recall Disruption and Action Plan Disruption of Transludic Knowledge (TRANS-S and TRANS-E) and Intraludic Knowledge (INT-S and INT-E).
7.4.6: Providing the Player with Hints

In A:TDD, hints become available automatically via the player-character’s journal (referred to in the game as *mementos*) that the player can then refer to if they become stuck during the game. While these hints are not a means of providing explicit guidance to players, they often narrow the possible solutions to a situation down significantly.

![Figure 7.24: An early memento automatically provided to the player in A:TDD.](image)

For example (Figure 7.24), suggesting that organic tissue in a barrier can be dissolved removes the possibility of other potential solutions, such as burning or cutting which could be equally viable. Thus, this in turn removes the need for players to engage in analysis of the situation and evaluation of possible solutions, before deciding on one to pursue. The solution (i.e. to dissolve the tissue) is provided, leaving players to simply search the game environment for something to facilitate the dissolving process. The degree of cognitive engagement required is thus reduced. Players are able to rely on understanding of information (i.e. a lower-order thinking skill) provided to them.

![Figure 7.25: Interacting with the organic tissue provides players with a further piece of explicit information.](image)

The hint in the memento is also compounded by a message given to players if they attempt to interact with the organic tissue (Figure 7.25). This additional hint combined with the suggestion that dissolving the organic tissue is the correct solution further narrows the different solutions a player may consider and thus, further reduces the necessary analysis and evaluation (i.e. higher-order thinking) required.
The initial design of A:AMFP made significant alterations to A:TDD’s hint system. A scenario such as the one referred to in Figure 7.24 and Figure 7.25 would require active player experimentation, with environmental feedback, rather than explicit information, informing the player’s understanding of the problem and further decision making. For example, the surrounding game environment may contain tools to facilitate all of cutting, burning and dissolving (e.g. a saw, some matches, and a beaker of acid). The acid may be the only solution that is successful but the player is able to consider the other alternatives, try them for themselves and find out through experimentation that they do not work. The game can then provide contextual feedback to the player following failed attempts that allows the player to construct their own understanding of the problem and why certain actions are unsuccessful (e.g. the saw may be blunt and the matches may be wet).

If the correct solution is hinted at before players have the time to consider all possible solutions for themselves (i.e. as happens in A:TDD), an entire step in the problem-solving process is missed; a step which allows players to heighten their cognitive engagement with the game through application of higher-order thinking. By making significant alterations to the previous A:TDD hint system, this disruptive game feature was designed to implement Recall Disruption and Action Plan Disruption of transludic procedural, semantic, and episodic knowledge relating to the process of solving puzzles in the game and the meaning of puzzle-related game entities. The Recall Disruption was designed to require players to engage in analysis and evaluation of the information provided to them via the game environment and the processes through which they acquired it. The Action Plan Disruption was designed to require players to engage in creation of new problem solving strategies (i.e. interacting with game entities to acquire contextual information, rather than relying on text-based hints).

This approach to providing hints and feedback to the player requires consideration of how much development time is required to ensure contextual feedback is available for different player interactions. It would not be possible to code contextual feedback for all possible interactions. However, limiting the number of ‘useable’ items in an environment and also limiting the areas in which those items may be logically ‘used’ should mitigate this issue and ensure a manageable development workload.
The design for the method for providing the player with hints can be formalised within Stage 2 of the DisDev (Figure 7.26) model as a disruptive game feature that targets Mechanics, with a Visual mode of output, aiming to create Recall Disruption and Action Plan Disruption of Transludic Knowledge (TRANS-P, TRANS-S, and TRANS-E).

7.4.7: Handling the Death of the Player-Character

‘Killing’ the player-character was something that both FG and TCR wanted to avoid in *A:AMFP*, in line with the game's focus on ‘adventure’ qualities rather than ‘survival’ qualities (Section 7.2.4) and also in line with both TCR’s and FG’s design ethos (Section 7.1.3). FG had already employed this same philosophy in *A:TDD* primarily with the aim of avoiding repetition of gameplay sections and thus endeavouring to mitigate the risk of players becoming less fearful of the game through increased familiarity. Grip (2011) of FG, speaking at Game Developers Conference, discusses the company’s experience of handling the death of the player-character in their earlier titles *Penumbra: Overture* (Frictional Games, 2007) and *Penumbra: Black Plague* (Frictional Games, 2008).

Discussing a particular chase sequence in *Penumbra: Overture* in relation to player death, Grip states that:

> Fifty percent of the people that played it, they liked it. Fifty percent, the other half that played it, they just hated it. We asked ourselves, what is the difference between these two groups? It turned out that the people that liked it, they managed to complete these chase sequences in one or two attempts. The other people that took more tries, they didn’t like it. The solution to this is really easy – you just don’t make the player repeat it more than two times. (Grip, 2011)

Avoiding repetition was an important aim for TCR as it aided in working towards the creation of story-based gameplay. Repeating sections of gameplay and thus, repeating sections of the game’s story, pulls the player out of the experience and is detrimental to the player’s sense of continuity and pacing. However, it is important that ‘death’ holds some consequences for the player so that it is still perceived as a threat. If there is no perceived threat from the enemy agents then the player experience is likely to be significantly negatively affected. Thus, a game system that balances the maintaining of perceivable threat with minimal requirement for players to repeat sections of gameplay was necessary. However, working with the disruptive game design philosophy, it was...
also necessary to consider how such a fundamental component of the game could be disrupted in a manner that could provide increased cognitive engagement but would not be detrimental to the positive experience of players.

Thus, the design of A:AMFP’s death-handling system used A:TDD’s system as a basis but made some significant alterations so as to provide Recall Disruption of transludic episodic knowledge. In A:TDD players were respawned in a different location (in an area of the map they had previously visited) upon being ‘killed’ and some events or enemy agents within the level may have been modified. For example, new scripted events may occur if a player had been ‘killed’ once in an area, or enemies may move locations or disappear completely. A:AMFP’s initial design was intended to operate in a similar manner, but was intended to respawn players in areas of the game environments separate from the main map that were not accessible in any other way.

During the creation of the game as designed, these separate areas were referred to as ‘Pig Nests’. The design intent of this disruptive game feature was to produce, in levels containing at least one enemy encounter during which the player could be ‘killed’, a ‘nest area’ placed around the map’s perimeter. These nests were intended to be inaccessible during normal gameplay, being separated from the main game map by mechanisms such as one-way doors, or ledges too high to jump up that could thus also only be traversed in one direction (i.e. by dropping down from them). When players were ‘killed’ by enemies, they could then reawaken in the level’s nest area, having been dragged there by the enemy agent in a short cutscene sequence.

To escape these nest areas an additional puzzle, or series of actions, could be required to be performed, such as throwing debris to knock a ladder down from a wall to climb out of a pit. Players could then be able to return to the main game level and reattempt the section in which they were previously unsuccessful. Enemy locations and behaviour may remain constant in some cases, or some changes may have occurred on a player’s second attempt (e.g. the enemy agent’s patrol path may have altered), thus providing Recall Disruption of transludic, A:TDD-based knowledge of ‘dying’ during gameplay, along with potential Action Plan Disruption of the player’s intraludic semantic and episodic knowledge relating to enemy locations, patrols, and behaviours.

In larger levels, or levels containing multiple enemy encounters, the nests could be designed to be connected to the main level via multiple routes. Depending on the area in which the player is ‘killed’, the appropriate return route could be made accessible, once again through the use of one-way access mechanisms such as doors and ledges.
For example (Figure 7.27), players being ‘killed’ in Enemy Scenario 1 will be moved to the Pig Nest and, after solving a puzzle to escape, will have access to Return Route 1. Likewise, players being ‘killed’ in Enemy Scenario 2 will have access to Return Route 2. Players will only have to complete the escape puzzle once however, so multiple failures will not result in repetition of the same process. Multiple deaths and potential repetition is counteracted by the behaviours or positions of enemies sometimes being modified after the player is ‘killed’. This system thus is able to provide Recall Disruption of A:TDD-based transludic knowledge, as the system functions in a notably different way and requires players to engage in additional cognitive engagement as a means of returning to the main game area. Moreover, placing players in an otherwise inaccessible area of the game level and then requiring players to complete challenges to return to the main game has the potential to provide disruption of more general game-medium-based transludic knowledge about how ‘death’ and ‘respawning’ tend to function. While these concepts are not wholly original (for example, Prey (Human Head Studios and 3D Realms, 2006) requires players to kill a number of enemies in the ‘spirit realm’ to be returned to the main game), they are not commonly utilised and thus, have disruptive potential for a number of players.

<table>
<thead>
<tr>
<th>GAME COMPONENT(S)</th>
<th>MODE(S) OF OUTPUT</th>
<th>MODE(S) OF DISRUPTION</th>
<th>TARGET KNOWLEDGE TYPE(S) TO DISRUPT</th>
</tr>
</thead>
</table>

Figure 7.28: DisDev Stage 2 elements forming the design for the method for handling the death of the player-character.
The design for the method for handling the death of the player-character can be formalised within Stage 2 of the DisDev (Figure 7.28) model as a disruptive game feature that targets Mechanics, with a combined Visual and Auditory mode of output, aiming to create Recall Disruption and Action Plan Disruption of Transludic Knowledge (TRANS-S and TRANS-E) and Intraludic Knowledge (INT-S and INT-E).

### 7.4.8: Hiding Places

**Figure 7.29:** Players in *A:TDD* can use cupboards such as this to hide from pursuing enemies.

One of the most common methods of hiding from enemy agents in *A:TDD* was to hide inside one of the many cupboards placed within the game environments (Figure 7.29). The act of hiding was one that was viewed as necessary to maintain in *A:AMFP* as it is a powerful way of generating anxiety and tension during gameplay with very few resources required. Much of the enjoyment of the process of hiding from enemies is generated within the player's own imagination. The development team need to only implement enough to provide the suggestion that the player is being hunted by enemies, thus potentially significantly reducing the amount of art, sound, and artificial intelligence development work required. However, the hiding places themselves needed to be differentiated from *A:TDD*'s to ensure that *A:AMFP* did not appear to be simply remaking *A:TDD* and trying to replicate identical systems (Section 7.1.4). This differentiation also provided a basis for all of Encoding Disruption, Recall Disruption and Action Plan Disruption.

Thus, rather than relying solely on the player hiding within objects that physically blocked the enemy agents’ lines of sight, a design was instead proposed that called for the creation of script-controlled hiding places. Rather than the artificial intelligence system checking whether enemies could ‘see’ the player-character or not (through a raycast-based system) and basing their behaviour on this (a system that had been demonstrated by *A:TDD* players to contain exploits, such as carrying an object in front of the player-character to prevent enemies from seeing them), the player would be able to use a more varied selection of hiding places (3D volumes within the
game environments) which provided different ‘hiding properties’. These hiding properties would remain ambiguous, thus providing a basis for Encoding Disruption (via ambiguous stimulus information) of intraludic semantic knowledge and requiring players to engage in analysis of the game environment to ascertain the relative safety of different hiding places. Recall Disruption of transludic semantic and episodic knowledge could be achieved through specific use of similar scenarios to *A:TDD*, such as placing cupboards and other furniture throughout the game but making them much less reliable as hiding places. Lastly, by making these changes to how the player is able to hide from enemies, Action Plan Disruption is achieved requiring players to create (i.e. higher-order thinking) new strategies for avoiding enemies and moving through dangerous areas of the game.

![Figure 7.30: DisDev Stage 2 elements forming the design for hiding places.](image)

The design for hiding places can be formalised within Stage 2 of the DisDev (Figure 7.30) model as a disruptive game feature that targets Mechanics, with a Visual mode of output, aiming to create Encoding Disruption, Recall Disruption, and Action Plan Disruption of Transludic Knowledge (TRANS-S and TRANS-E) and Intraludic Knowledge (INT-S and INT-E).

### 7.4.9: Item Inventory Removal

The use of an inventory system that enables the player to store and combine the various items that they collect during gameplay is a common feature across a range of different game types. It is also a fundamental component of games within the ‘adventure’ genre as it affords players the ability to examine the items they find and to identify possible solutions to puzzles that utilise them.
Figure 7.31: A:TDD’s main inventory screen, showing resources such as health, sanity, lantern oil level, and collected items.

A:TDD (Figure 7.31) and FG’s earlier Penumbra series all used an inventory screen that allowed players to view, manipulate, and combine items that they were carrying, most frequently door keys or tools for completing particular puzzles in the game. It also served as a place to store and read journal pages and notes (the main story delivery methods in each of these games) as well as hints that could provide guidance for players if they became stuck or unable to work out a solution to a particular problem in the games.

However, inventory systems are not only a common feature of ‘adventure’ games. They also provide one of the fundamental heterodiegetic mechanics that exists across a wide range of horror-themed games in the form of inventory management.

Therrien (2009) writes on this subject:

[A] common figure in horror video games [is] management. In a context where ammunition is hard to find, players are more likely to use these resources very carefully [. . .] The scarcity of resources thus favours a potentially more methodical attitude with regards to fighting or shooting mechanics. However, "management" takes on a very literal meaning in games where the avatar can gather resources in an "inventory" whose space is restricted [. . .] Even The Lurking Horror, a text-adventure game, made inventory management an integral part of the experience. (Therrien, 2009, p.37-38)

In AAMFP however, the game as designed removed the inventory system completely, as one of the game’s primary disruptive features. Specifically, this could create Recall Disruption of transludic semantic and episodic knowledge of game ‘genre’ as well as potentially at the more specific transludic episodic level of the Amnesia and Penumbra series in particular. The disruptive impacts on the gameplay experience were thus designed to be twofold.
Firstly, by eliminating the need to enter a separate screen that takes players out of active gameplay, players would be able to remain more closely engaged (both at a cognitive and at a motor level) with the real-time game experience. The ability to open an inventory screen (or indeed any separate screen) allows players an ‘easy escape’ from a tense or dangerous situation. For example, if a player is being pursued by an enemy agent, being able to open a separate screen that also pauses the game and temporarily alleviates the tension is potentially detrimental to factors such as immersion, challenge and enjoyment. However, the commonality of this ability across games makes it likely to be an expected feature in AAMFP for many players. Thus, eliminating one of the key methods of invoking that ‘easy escape’ to an inventory screen has the potential to be disruptive at the level of Action Plan Disruption.

Secondly, the removal of the inventory would fundamentally change the way that players would need to approach gameplay involving puzzles and puzzle-related items. Rather than being able to store every item they locate during exploration in an inventory, players would need to manually carry items around the environment from wherever they locate them to wherever they are required to solve a puzzle or complete a particular challenge. This in turn may require a significant degree of additional cognitive engagement as players must plan a safe route, avoiding enemy agents, whilst carrying an item that may potentially make them more vulnerable to attack and less able to make a quick escape from any threats.

This disruptive game feature had the potential to also provide significant Action Plan Disruption of transludic semantic knowledge for players. They could encounter puzzle scenarios and puzzle-related items that could usually be solvable by utilising the inventory system but instead, require a notably different process of planning and action to be taken requiring cognitive engagement through each of analysis, evaluation and creation of new gameplay approaches.

**Figure 7.32: DisDev Stage 2 elements forming the design for the removal of the item inventory.**

The design for the removal of the item inventory can be formalised within Stage 2 of the DisDev (Figure 7.32) model as a disruptive game feature that targets Mechanics, with a Visual mode of output, aiming to create Recall Disruption and Action Plan Disruption of Transludic Knowledge (TRANS-S and TRANS-E).
7.4.10: The Player-Character’s Lantern

As Rouse (2009) states, citing Lovecraft, Poe, Hitchcock, and Kubrick as examples,

The plot of these works is quite simple, the explanation minimal, and what does happen is barely explained, if explained at all. In horror, the way the audience fills in the blanks will be far more disturbing than anything a writer could possibly come up with. (Rouse, 2009, p.17)

While Rouse refers to ‘blanks’ within the plot of a story, aesthetically, one may interpret these ‘blanks’ as being literal areas of darkness. Thus, darkness provides potential space in which the unknown can exist, providing a means of heightening player anxiety and tension. Darkness, being such a well-established horror theme, provided potential as a basis for disruptive game feature design. In A:TDD, players are able to explicitly control the intensity of darkness around them by switching a hand-held lantern on and off. A game feature that could disrupt this notion of control over the darkness for both players with and without A:TDD experience was thus devised for the game as designed.

The lantern in A:TDD projected an omnidirectional light that had a relatively large range in comparison to the size of the game environments. This allowed the player to illuminate a significant amount of the on-screen environment, leaving little immediate space for darkness or the ‘unknown’. While in larger environments the light’s decay rate as it moved away from the player allowed distant areas to be in darkness, the illuminated area around the player was large enough to act as an effective buffer between the player-character and potential threats. Enemies emerging from the darkness ahead of the player-character could be easily moved away from before they reached the player-character.

Figure 7.33: Typical environment in A:TDD illuminated with the omnidirectional lantern light.

A typical area in A:TDD (Figure 7.33) could therefore be easily viewed by the player (larger and smaller areas are also used, but much of the game occurs in relatively consistent size environments). Even in an average size room such as this, the player-character’s lantern illuminates the room clearly with very little effort required by the player, immediately removing darkness and the potential for unseen threats to remain hidden. Thus, the process, or series of
actions the player must take in A:TDD to evaluate a new environment’s threat level consists of activating the lantern and in cases such as the above environment, making small movements with their mouse or gamepad joysticks to ‘sweep’ the light across the room and look for any potential threats.

In A:AMFP, the lantern was designed to produce a very narrow beam of light, rather than A:TDD’s omnidirectional light. Thus, this would require more player effort to evaluate an environment. Significantly more player input (i.e. mouse/gamepad joystick movement) would be required to ‘sweep’ the light across an environment. Additionally, there could be areas of darkness on the player’s screen at nearly all times during gameplay, meaning that to maintain a sense of safety players would need to perform this sweeping motion and be analysing and evaluating the environment more consistently. This additional task would therefore require players to have a greater level of cognitive engagement with the game due to the reduced ability for players to have access to a ‘full set’ of visual information about their current environment at any one time (i.e. some areas will always be in darkness). This places additional demand on the player to ensure they are regularly sweeping the light across the environment in addition to whatever other tasks they are performing (e.g. attempting to solve a puzzle), thus requiring greater cognitive engagement to effectively switch between these activities.

This disruptive game feature could therefore provide Recall Disruption of transludic procedural, semantic, and episodic knowledge (primarily knowledge acquired in A:TDD, although other games may use similar lantern designs), as well as Action Plan Disruption based on the player’s transludic semantic knowledge (again, primarily from knowledge acquired in A:TDD, although other games may require similar actions to be performed by players, such as Doom 3 which uses a narrow-beamed flashlight). Cognitive engagement was designed to be achieved through requiring players to analyse and evaluate the game environment consistently throughout gameplay, utilising a strategy created specifically for use in A:AMFP.

Figure 7.34: DisDev Stage 2 elements forming the design for the player-character’s lantern.

The design for the player-character’s lantern can be formalised within Stage 2 of the DisDev (Figure 7.34) model as a disruptive game feature that targets Aesthetics, with a Visual mode of
output, aiming to create Recall Disruption and Action Plan Disruption of Transludic Knowledge (TRANS-P, TRANS-S, and TRANS-E).

7.4.11: Manipulation of ‘Set Dressing’ Entities and Props

Entity consistency with regard to a game’s environmental assets is a general transludic-knowledge-based expectation that players may be likely to have, although it is not something that they may consciously consider. This expectation will also be supported by extraludic knowledge. For example, the location of entities such as doors, light fixtures, tables, ornaments or other furniture in an environment are unlikely to be expected to move, change or otherwise be manipulated during gameplay in some way that is not clearly a part of the game’s story or that was not clearly caused by the actions of the player or a non-player-character (i.e. based on transludic knowledge). Likewise, entities in the real world are unlikely to be expected to move without rational explanation (i.e. based on extraludic knowledge).

Such manipulation was identified during the design process as a potentially effective method of disrupting that fundamental transludic semantic and episodic knowledge via Recall Disruption. For example, placing a locked door along the wall of a corridor that is replaced by a solid wall when players look away from it, or moving smaller objects such as books, bottles or chairs around a room when players are facing away from them. While methods such as these may be utilised in other games to indicate the ‘mental state’ of the player character (e.g. a reflection of the character’s level of ‘sanity’ (Section 7.2.4.3)), the design intent in this instance was to implement entity manipulation throughout the game in a manner that was not directly linked to the ludodiegetic state of the character (e.g. occurring across a range of different scenarios in multiple different environments and not always in ways that may be clearly signposted to the player). Thus, the manipulation was less likely to be understandable by players as simply the game providing a reflection of the player character’s sanity.

Further to this transludic- and extraludic-based disruption (i.e. entities and props are not likely to be expected to move in such ways in other games or in the real world) such a disruptive game feature also provided a means for Recall Disruption of intraludic semantic and episodic knowledge. Players would perceive and remember a particular environment (i.e. store their experience of that environment as an episode in episodic memory) which upon being perceived again would have been subtly altered through the manipulation of some of the environment’s smaller entities. Thus, the player’s previous episodic memory becomes inaccurate. Additionally, players may also have to reconstruct their semantic understanding regarding the properties of the entities that have been modified (i.e. assign those entities the new property of ‘may change without explanation’, or similar).
The design for the manipulation of ‘set dressing’ entities and props can be formalised within Stage 2 of the DisDev (Figure 7.35) model as a disruptive game feature that targets Aesthetics, with a Visual mode of output, aiming to create Recall Disruption Transludic Knowledge (TRANS-S and TRANS-E), Extraludic Knowledge (EX-S and EX-E), and Intraludic Knowledge (INT-S and INT-E).

To prevent these environmental changes becoming potentially predictable, the scenarios could also be semi-randomised. Thus, particular events experienced by some players may not be experienced by other players. Likewise, what was experienced during one play-through of the game may not occur in the next and an entity that a player may assign the property of ‘may change without explanation’ may not necessarily ever be manipulated again during the same play-through of the game. The differences could thus be designed to be subtle and to have no direct impact on the ‘main thread’ of gameplay. However, their presence and unpredictable nature would prevent players being able to construct a wholly reliable intraludic understanding of the game environment and to thus encourage consistent analysis of the game environment during gameplay.

7.4.12: Ambiguous Stimulus Information and the ‘Pig Mask’ Motif

Encoding Disruption may be achievable by providing the player with incomplete or ambiguous stimulus information, or by purposely distracting the player by presenting them with multiple simultaneous stimuli (Section 4.3.1). In AAMP a disruptive game feature was designed to provide a source of Encoding Disruption via the first method; ambiguous stimulus information.
This disruptive game feature would present players with a consistent motif throughout the game in the form of a porcelain pig mask (Figure 7.36). This motif would appear in seemingly random locations (at an implementation level, this functionality could be based on a random selection from a predefined list of possible locations in each game level) and would also be prone to appearing and disappearing itself (in the same manner as ‘set dressing’ entities (Section 7.4.11)), again seemingly at random.

While this disruptive game feature may appear similar in functionality to the manipulation of other ‘set dressing’ entities and props, the unique intraludic nature of the mask itself (i.e. it is an entity designed for this particular game, unlike other transludic entities such as doors or generic furniture) allow it to be used in different ways and for different disruptive purposes.

For example, the properties of the object itself, as a mask, allow it to be potentially attached to other game entities, hung from hooks on walls or ceilings, or to be placed among other small game entities on desks and shelves or among debris on the floor. Thus it has more positional flexibility within the game environments. This flexibility was coupled with the mask not being designed to have any explicit in-game explanation for its presence or purpose. This thus could provide an opportunity for players to experience Encoding Disruption of intraludic semantic knowledge as they may infer a range of different meanings for the mask. Depending on the particular inferred meaning, players may then experience Recall Disruption of that intraludic semantic knowledge at later stages of the game when the mask motif is encountered again, depending on if these later encounters are supported by their original interpretation of the mask’s meaning or not. In terms of cognitive engagement, the consistent presence of a stimulus with ambiguous meaning was designed to encourage players to engage in ongoing analysis of the possible meaning(s) of the pig mask’s presence, along with ongoing evaluation of any previously assigned understandings the player may have created during gameplay.

From the perspective of the current research, the use of the pig mask motif along with the manipulation of other ‘set dressing’ entities and props (Section 7.4.11) also provides potential for a comparative analysis of the two disruptive game features, as they are functionally similar but operate on different knowledge types in different ways.

Figure 7.37: DisDev Stage 2 elements forming the design for the ‘pig mask’ motif’s behaviour.
The design for the ‘pig mask’ motif’s behaviour can be formalised within Stage 2 of the DisDev (Figure 7.37) model as a disruptive game feature that targets a Cue for Constructed Narrative, with a Visual mode of output, aiming to create Encoding Disruption and Recall Disruption of Intraludic Knowledge (INT-S).

7.4.13: Enemy Audio Cues

During the design process, research was carried out by the development team on the A:TDD official online discussion forum with the aim of identifying commonly cited features of the game that players responded to particularly positively or negatively. While little was identified that players specifically identified as being detrimental to their game experience, the use of audio cues to communicate enemy threat levels to the player was discussed across multiple forum threads, all framing this particular feature as detrimental to the game’s degree of challenge. Consideration of this particular feature in A:TDD provided a basis for a potential disruptive game feature in A:AMFP.

The use of audio cues to signify enemy presence or enemy behaviours is common in a number of games. Minecraft (Mojang, 2011) for example utilises iconic sound effects for different enemy types, notably Creepers and Ghasts, which players can utilise as a means of identifying the type of enemy that is approaching and then reacting appropriately. System Shock 2 attaches different lines of dialogue along with different voices to different enemy types making it possible, through close attention to the game’s audio, to adapt tactics and weaponry prior to confronting enemies. Such audio cues were also heavily utilised in A:TDD, which triggered a specific audio track whenever a player was spotted by an enemy agent. This audio track faded back out again once the enemy had lost sight of and stopped actively hunting for the player.

Such audio cues may be used to build tension and anxiety, as per Perron’s (2004) suggestion. Citing Lazarus’ (1964) concept of ‘anticipatory fear’ and Cantor, Ziemke and Sparks’ (1984) work on building tension and anxiety in film viewers, Perron goes on to suggest that

\[
\text{[. . .] intuitively, prior knowledge about an upcoming frightening event would seem to reduce its emotional impact by decreasing uncertainty about what will happen, [however, this] is not what actually happens. In fact, on the contrary, the notion “forewarned is forearmed” does not lead as much to “emotional defences” or effective coping strategy as to a build up of lasting arousal prior the event. (Perron, 2004, p.4)}
\]

The radio in Silent Hill (Konami Computer Entertainment Tokyo, 1999) is an example of such forewarning, producing static as enemies draw near to the player-character. However, once the player learns the rules of the warning system or device, there is potential for that ‘anticipatory fear’ to diminish each time the warning is encountered.

Moreover, players will form expectations of what actions they should take upon hearing certain warning sounds and will react to them accordingly. This can be likened to classical conditioning (Pavlov, 1927), with players becoming conditioned to respond to the warning sounds regardless of
immediacy of an actual threat (Appendix A.1). This thus potentially detracts from the level of cognitive engagement required to analyse and respond to the warning sound.

This is cited in the specific case of A:TDD by multiple players. One player, offering a high-level generalised description of enemy encounters in the game states that

> From what I've experience [sic], all enemy encounters, scripted or non-scripted, work out the same way. You'll hear it, then the intense music will cue, and then you go hide in a room. After about a minute or so, the music will stop, and at this point you are free to continue on your path. (Forum Post 01, 2011)

A second player similarly notes that

> There are a lot of scary sounds in the game, but playing through I could quickly tell that most of them were just atmospheric and could be ignored. When a real monster actually shows up, you hear a distinctive moan and music. I then hid somewhere until the music and groaning stopped. It was thus easy for me to tell when it was safe and when to be careful. (Forum Post 02, 2010)

Such apparent expectation based upon the use of audio cues attached to enemy agents provided an opportunity for designing a disruptive game feature. The presence of audio cues in A:TDD, as well as in other games (e.g. Silent Hill), provided both franchise-based and game-medium-based potential for the disruption of transludic semantic and episodic knowledge via Recall Disruption and Action Plan Disruption. This was designed to be achieved by making the use of enemy audio cues throughout AAMFP much less identifiably consistent, both in terms of whether the audio cues were always used when enemies were present, as well as the types of audio cues used and what those cues may mean (i.e. what type of enemy may be present and whether they are actively hunting for the player or not).

Furthermore, with sound being a comparatively ‘cheap’ game element to experiment with in terms of time and resources required (to create placeholder sounds for testing), there was potential for the disruption of intraludic semantic and episodic knowledge over the course of the game (by, for example, manipulating the types of sounds used by enemies at different points over the course of the entire game).

Figure 7.38: DisDev Stage 2 elements forming the design for the enemy audio cues.
The design for the enemy audio cues can be formalised within Stage 2 of the DisDev (Figure 7.38) model as a disruptive game feature that targets Aesthetics, with an Auditory mode of output, aiming to create Recall Disruption and Action Plan Disruption of Transludic Knowledge (TRANS-S and TRANS-E) and Intraludic Knowledge (INT-S and INT-E).

7.4.14: Summary of the Game as Designed for Amnesia: A Machine for Pigs

The game as designed for AAMFP consists of the defined non-disruptive framework of setting (Section 7.3.1), characters (Section 7.3.2), and plot (Section 7.3.3), along with the disruptive game features (Section 7.4.1 to Section 7.4.13) that have been intentionally designed following the principles of the disruptive game design philosophy (Section 1.3).

![Figure 7.39: Stage 2 of the DisDev model applied to AAMFP](https://example.com/image.png)

Stage 2 of the DisDev model in the context of AAMFP (Figure 7.39) summarises the specific disruptive game features that will have prototypes developed (the categories that combine to form disruptive game features have been truncated in this example). The non-disruptive game features are summarised in part, with specific focus placed on the previously defined non-disruptive framework. There are many other non-disruptive game features (by definition, any feature in the game that is not specifically defined as disruptive), thus it would be impractical to
196
Moreover, these features are not the focus of the current research and thus are not considered in detail.

7.5: Application of DisDev Model Stage 3 and Stage 4: Prototyping and Developing Disruptive Game Features in the Game as Created

Disruptive game features proposed in the game as designed (Section 7.4) must undergo prototyping (Figure 7.40) to assess both their viability for full development and inclusion in the game as created (Figure 7.40) as well as to assess how playtesters respond to them, either positively or negatively (or indeed, whether certain features go unnoticed during gameplay).
Figure 7.41: DisDev model Stage 3: Game Feature Prototyping and Stage 4: Full Development.

This section describes each of the disruptive game features as they were prototyped during Stage 3 of the DisDev model, Game Feature Prototyping, and Stage 4 of the DisDev model, Full Development (Figure 7.41). The prototyping process is first described for each feature, with key changes made between iterations defined and the rationale behind those changes explained. The outcome of the prototyping process for each feature is identified, with each being either fully developed into a part of the game as created (in most cases after going through multiple prototype iterations requiring additional design work to be carried out), or being removed from the game. Where a disruptive game feature was approved for full development, the implementation of that feature within the game as created is further discussed.

As is demonstrated in this section, not all of the features initially designed were feasible once the development process was underway and practicalities of implementing the features were encountered first hand. However, while some features were not fully implemented or were otherwise significantly simplified, some features did remain in place from the ‘game as designed’ requiring little or no modification from their original designs.

Additionally, a number of features were added during these stages (i.e. Stage 3 and Stage 4) that had not been originally considered. These features either emerged as a natural part of the discussions during the development process, or were responses to changing requirements of the project, such as other features not working as intended and alternatives being needed. These additional features are discussed in the following sections first in terms of their design within the DisDev Stage 2 framework, before then being discussed in terms of the development of their prototype(s) and lastly their full development within the game as created.

The emergence of these additional features following the creation of the game as designed demonstrates the importance of maintaining the disruptive game design philosophy throughout the design, prototyping and development processes (i.e. not only applying it during the early design
process but keeping its principles in mind throughout the development cycle), as some disruptive
game features may only emerge as part of the natural progression and evolution of the project.

7.5.1: Removal of the ‘Infection’ System
The infection system was developed to a first prototype stage over the course of the first two
months of development, with game code and placeholder art and sound assets in place.

![Prototype of Infection System demonstrating the four stages of infection and their impact on the player's vision (A:AMFP March 2012 Build).](image)

This prototype enabled the player to experience four distinct levels of infection (Figure 7.42),
from mild infection that caused a subtle overlay to appear around the edges of the screen,
through to heavy infection that caused highly intrusive screen overlays to appear across the
screen, along with significant warping and distorting of the player’s view.

However, as the development process continued it became apparent that the system was not
providing an enjoyable and cohesive play experience, based on feedback from both testing by the
development team and a small number of external play testers assigned to the project by TCR
and FG. Thus, approximately six months into development, the decision was taken to remove the
system from the game. This decision was made for two reasons.

Firstly, the focus of the game’s story shifted towards having a greater emphasis on political and
social aspects (i.e. commentary on capitalism, automation and social class structure), rather than
on the disease and infection aspects. The infection system was thus somewhat detached as a game
mechanic with minimal relevance to the told story and the game world (i.e. the milieu). From the
perspective of disruptive game design, such detachment is not in itself problematic; indeed, that
sense of detachment may itself heighten the disruptive impact of the disruptive game feature by
placing it more notably outside a player’s intraludic reference schemas. However, the infection
system was to be a central game mechanic in *A:AMFP* and it was the opinion of the team as a whole that the game would be unnecessarily confusing for players if the system remained in place.

Secondly, the infection system was difficult to embed alongside other game components, within gameplay, in a manner that made it seem like a natural part of the game. Whereas the first issue described above related to the situating of the system within the game’s told story and game world, it was evident that the system was not coexisting well alongside the other ‘mechanic’ elements of the game either. Playtesters during early testing sessions stated that it felt like a ‘mechanic for the sake of being a mechanic’. That is, it was being perceived as a system that was present simply to add something else for the player to deal with and was not being perceived as a naturally occurring part of the game.

The original aim of this disruptive game feature was to create Encoding Disruption of intraludic knowledge while the player was under the effects of the infection system. Thus, the perception of the system during playtesting as ‘something else for the player to deal with’ could be viewed as the intended outcome. However, the impact on cognitive engagement was not reported as positive. Rather than the system creating an enjoyable challenge for players, it instead created frustration or confusion.

The removal of this system was a significant design decision, given that the system was the primary mechanic originally pitched in the early game design document, replacing *A:TDD*’s ‘sanity’ mechanic (Appendix E). From the perspective of disruption research however, its removal was not significantly problematic due to the inclusion of a range of other disruptive game features. Indeed, the reasons for the removal of the infection system demonstrate the use of the disruptive game design philosophy and framework in a manner that is able to function within a live development project with changing requirements. The removal of the infection system also demonstrates that even following the consideration of the project’s contextualising factors, there is still the possibility of disruptive game features being removed from the game as the project progresses due to changes in the game design’s direction, or circumstances that could not have been predicted during the project’s early stages.

The removal of the infection system is discussed further in the post-mortem report (Appendix F).

### 7.5.2: Removal of System to Emulate Peripheral Vision

Due to technical limitations of the HPL2 engine that were not apparent when the peripheral vision concept was initially proposed, this system did not move beyond the early design stage. The significant additions to the engine’s rendering system and also the additional artificial intelligence requirements needed to make enemies behave in a way that would make the system purposeful would have required too much additional investment of time and resources, primarily from the development team’s engine programmer. These required additions were however identified early in the prototyping stage and thus, this feature was removed from the game before any significant
time had been spent on developing a fully functional prototype. Hence, its impact on the prototype development of other game features was negligible.

However, the feature itself nevertheless has the potential to form the basis of future research work given its relative absence from existing available games and its potential for affording different types of gameplay. Indeed, FG was positive regarding the design of the system itself but was also clear in stating that implementing it would require significant additional work on the HPL2 source code. It was only once the prototype development stage was reached however that the full requirements of such additions to the HPL2 source code became apparent. This thus demonstrates the potential differences between the game as designed and the game as created once the practicalities of the game development process have been accounted for.

7.5.3: Removal of Procedurally Generated Tesla Field ‘Maze’

While concepts were created for the contents of the maze's various rooms, attempts at creating a functional prototype revealed that the amount of processing that would be required during runtime would not be possible within the constraints of the HPL2 engine. The prototype aimed to create a system for the procedural generation of just two interconnected rooms. However, this significantly simplified prototype raised a number of issues, such as how to store different room configurations in memory, how to ensure players could not become stuck within geometry as they passed between rooms, and how to ensure the connecting doors between rooms did not themselves become stuck when a new room was generated behind them.

Issues such as these (especially issues involving the physics engine, such as the doors between rooms becoming stuck) could not have been predicted during the assessment of project context (DisDev Stage 1), or during the creation of the game as designed. The prototype development process was necessary to identify such issues, through practice, highlighting the benefit of utilising the RtD methodology. The eventual removal of this feature demonstrates again that even with careful initial planning and evaluation of the game as designed against the unique contextualising factors of a project, transferring features of the game as designed into the game as created is a process that can present significant, unforeseen issues.

7.5.4: Addition of ‘Non-Euclidean’ Space

While the procedurally generated maze (Section 7.4.3 and Section 7.5.3) was not able to be implemented due to its technical complexity, TCR still wanted to make use of game features utilising the same underlying concept of ‘unpredictable environments’. Moreover, this was also of interest from the perspective of the current research as a means of implementing disruption.

Thus, time was allocated to return to the design stage (i.e. DisDev model, Stage 2) and consider whether more technologically accomplishable disruptive game features could be conceived. The concept of creating localised areas within specific game levels in which predefined sections of the environment would be modified as the player moved through them, was identified. Specifically, this would draw upon the concept of ‘non-Euclidean’ geometry.
The concept of non-Euclidean space (i.e. space that appears to ‘fold into itself’ or overlap) is not a new concept within games. Recent examples of games utilising the idea include Antichamber (Bruce, 2013) (Figure 7.43) and The Stanley Parable: HD Remake (Galactic Cafe, 2013). However, in these examples the use of non-Euclidean space is clearly demonstrated to the player. In Antichamber it is one of the game’s primary ‘aesthetic’ and ‘mechanic’ features, while in The Stanley Parable: HD Remake the scenarios in which non-Euclidean space is used are frequently devoid of any significant environmental ‘distractions’; players are expected to notice that the environments are non-Euclidean and thus, ‘impossible’ spaces.

The use of such ‘impossible’ spaces occurs in film also, associated with cognitive or memory disorders in characters, hallucinations, or other forms of distorted reality. Eternal Sunshine of the Spotless Mind (Gondry, 2004) for example, utilises the effect to signify the warping of time and space in the main character’s mind as his memories are erased. In the context of ‘horror’, The Shining (Kubrick, 1980), utilises impossible architecture and suggested non-Euclidean space throughout the film, both inside the Overlook Hotel and in its surrounding grounds (Ascher, 2013). This serves to disorient and confuse the viewer but also serves as an ongoing motif throughout the film that reminds the viewer of the supernatural forces at work throughout, as well as the main character’s unstable mental state. The Shining’s usage of such impossible architecture is less overt than some other examples in films and in games, although the intended effect – to confuse and disorient the viewer – is emphasised by supporting methods such as long, uncut camera shots that follow characters through winding corridors that change direction rapidly. Such experiences of impossible architecture, in contexts such as film, may provide players with extraludic semantic knowledge that they can utilise while attempting to understand such architecture in games.

Thus, there was scope within A:AMFP to implement a disruptive game feature that utilised this potential extraludic (and transludic, depending on the player’s knowledge of other games)
expectation that environmental or architectural ‘impossibility’ would often be overt, often be clearly signposted for players and be linked to the mental state of the main character. This disruptive game feature would aim to implement non-Euclidean architecture on a small, less overt and potentially therefore less immediately noticeable scale. Through also not explicitly linking the occurrences of such architecture to events in the game’s told story, or linking them to the main character’s mental state, transludic and extraludic knowledge may be less readily able to be used to understand the meaning behind the impossible architecture (i.e. Recall Disruption of transludic and extraludic semantic and episodic knowledge). By not linking the occurrences of such architecture to any explicit in-game explanation, the potential for players to experience Encoding Disruption of intraludic semantic and episodic knowledge via ambiguous stimulus information can also be created.

Figure 7.44: DisDev Stage 2 elements forming the design for the use of non-Euclidean space.

This additional design can be formalised within Stage 2 of the DisDev (Figure 7.44) model as a disruptive game feature that targets Mechanics, with a Visual mode of output, aiming to create Encoding Disruption and Recall Disruption of Intraludic Knowledge (INT-S and INT-E), Transludic Knowledge (TRANS-S and TRANS-E), and Extraludic Knowledge (EX-S and EX-E).

7.5.4.1: Non-Euclidean Space, Prototype 1

The implementation of non-Euclidean space requires the use of a variety of techniques within the game engine technology. The method utilised in games such as The Stanley Parable: HD Remake connects two separate parts of a game environment together through a ‘portal’. The surface of the ‘portal’ displays the ‘view’ of a virtual camera (referred to as camera projection) placed in the game environment. This allows players to see what is on the other side of the ‘portal’ surface.
The portal surface itself has collision deactivated allowing players to pass through. Doing so triggers a script that instantly teleports the player to the location of the virtual camera. Woodman (2013) demonstrates this method using a simple environment (Figure 7.45 and Figure 7.46).
Initial prototype development of this new disruptive game feature was carried out using an early version of one of the game’s levels (Figure 7.47) that featured a complex network of corridors and rooms. The level was designed to utilise a system similar to the ‘portal’ example previously described to connect these corridors and rooms in various ways.

However, experimentation with this method indicated that the HPL2 engine did not have the existing capability to implement the virtual camera projection method. TCR were aware that such functionality had not been previously required in other products developed using the HPL2 engine and that thus, fully implemented support for it would not be present. However, it was not anticipated that the underlying technology to support the ‘portal’ method (i.e. the capability to project a ‘virtual camera view’ onto a geometric surface) would itself not be available. Thus, without the required support in the HPL2 engine, implementing ‘true’ non-Euclidean architecture that connected physically separated rooms in this manner was not possible without considerable additional resources being used to add that capability to the engine. This presented the same problem that had been previously encountered with the procedurally generated ‘maze’ concept.

With this initial prototype failing to meet the requirements of the context of the development process (i.e. being able to be implemented using the available technology and the available team skills and resources) a further, less technologically ambitious, version of this disruptive game feature was required.

7.5.4.2: Non-Euclidean Space, Prototype 2
The second prototype design aimed to create the illusion of non-Euclidean space without having to overcome the significant technical challenges related to the ‘portal’ method (Section 7.5.4.1). Thus, this prototype would not be implementing non-Euclidean space as per its mathematically
correct definition (i.e. the bending, folding or warping of space). Instead game environments would be built in a manner that followed the laws of Euclidean space with rooms and corridors physically connected to each other as they would be in reality. With this as a basis, wall sections, doorways and other architectural components would disappear and reappear as players moved through the environment, utilising the same engine-level functionality as the manipulation of ‘set dressing’ entities and props (Section 7.4.11).

This fulfilled the aim of creating the illusion of non-Euclidean, or ‘impossible’, level geometry without placing significant strain on the game engine technology (i.e. the only functions required are the activation and deactivation of environmental objects), or the level designer (i.e. the game levels can be designed wholly within the level editor without requiring code-based generation of any parts of the levels). The illusion of non-Euclidean space still provided potential for players to experience disruption of transludic and extraludic knowledge, as the disruptive effect is reliant on the player’s perception of the game environment rather than the specifics of the implementation.

Due to the ease with which this prototype could be implemented and its ability to still potentially provide a disruptive effect on players, it was approved for full development and implementation into the main game. The prototype level (Figure 7.47) was itself moved into full development with the aim of making the illusion of non-Euclidean space the level’s primary ‘theme’.

7.5.4.3: Non-Euclidean Space, Full Feature Development

As full development of this feature and the Tunnels level (Figure 7.47) progressed, additional issues were encountered with the level’s complexity and size. While the prototype level structure was functional in its ‘blocked-out’, low detail version, its loading time and in-game framerate started to increase and decrease respectively as more environmental details and props were added. This level was the largest in the game (in terms of ‘player-accessible’ space) and indeed, significantly larger than any other levels in FG’s previous games that made use of the HPL2 engine. Thus, the level required optimising to retain an acceptable loading time and framerate.
Figure 7.48: Final version of the seventh level, *Tunnels* (A:AMFP September 2013 Release Build).

The level shown in Figure 7.47 was made notably smaller and less complex as a result. Figure 7.48 shows the version of the *Tunnels* level in the final game (i.e. the game as created), with the two areas highlighted in red showing corridor sections that ‘appear’ following gameplay triggers and ‘disappear’ once the player has moved through them.

Figure 7.49: Section of *Tunnels* level demonstrating the use of ‘impossible architecture’ and its setup in the HPL2 level editor (A:AMFP May 2013 Build).
Figure 7.49 demonstrates a specific example from this level (the right-hand corridor section shown in Figure 7.48) in which the architecture is altered based on player actions. Players enter this space from the left (marked ‘A’), proceeding in the direction indicated by the white line. Players then return to this space from the opposite direction, but will find the entrance marked ‘A’ now covered by walls (highlighted as white wireframes in the image). The wall marked ‘B’ on the right (also highlighted as white wireframes in the image) will now have changed from a solid wall into a doorway with functioning wooden door.

It is important to note that the ‘footprint’ of the game environment was created in Euclidean space; the geometry itself does not warp, distort, or ‘fold over itself’. The corridors themselves are always present within the environment, but shifting geometry at their entrance and exit points suggests non-Euclidean, ‘impossible’ architecture.

This disruptive game feature was only utilised in a small number of locations during the game and only significantly affects the game environment in the example provided from the Tunnels level, which occurs approximately half way through the game. This placement allows players time during the first levels of the game to construct what appears to be accurate intraludic semantic knowledge regarding the properties of the physical environments that make up the game world (i.e. that they appear to follow the same physical and spatial laws as the real world and thus, they can be understood in the context of relevant extraludic knowledge as well). Therefore, later levels that disrupt this established understanding do so via Recall Disruption of intraludic semantic and episodic knowledge. While the ‘rules’ of the virtual environment may not be consciously ‘recalled’ by players, there is potential for players to establish an understanding of such ‘rules’ during the early stages of the game.

Furthermore, the game does not provide any definitive ‘logic’ or reasoning behind the shifting level architecture. Therefore, there is the potential for players to also experience Encoding Disruption via ambiguous stimulus information at this stage. How players interpret this stimulus’ meaning may then further have an impact on how they interpret the game’s ‘told story’ or indeed, the content of any ‘constructed narrative’ they may form whilst playing.

This particular disruptive game feature is not utilised again in this manner (i.e. presenting an illusion of shifting whole corridors and rooms). However, the newly constructed intraludic knowledge may be utilised by players from this point onwards to inform their future understanding of later levels and also, to aid in selecting appropriate plans of action. The knowledge that the physical environment in the game may not remain consistent may thus influence how much attention players give to the level architecture in later levels. This in turn provides a basis for greater cognitive engagement (at the levels of analysis and evaluation, primarily) than players may otherwise be required to exhibit if the properties of the game architecture remained consistent and readily ‘known’ or ‘understood’ (i.e. the lowest levels of lower-order thinking) throughout gameplay.
The issues encountered with the runtime optimisation of the \textit{Tunnels} level (i.e. the loading times and framerate) demonstrate an oversight during Stage 1 of the DisDev model, the assessment of project context. While an analysis of the HPL2 engine and its key features was carried out as part of the assessment process, a robust ‘stress test’ of the engine was not carried out. Undertaking this process may have identified limitations of level size and complexity which in turn could have saved some of the development time that was spent having to redesign the \textit{Tunnels} level.

\textbf{7.5.5: Removal of Personality-Driven Enemy Artificial Intelligence}

The code for each of the three different personality types (i.e. ‘Rod’, ‘Jane’, and ‘Freddy’) was prototyped to a near-completed status, playtested and evaluated to be functioning correctly when enemies were placed in a simple environment (e.g. a network of empty rooms and corridors). As the game environments became more complex, more ‘cluttered’ with small game objects (e.g. crates, barrels, tables, chairs), and began to incorporate more ‘challenging’ (from an artificial intelligence and pathfinding perspective) environment features (e.g. doorways, hinged doors and sliding doors, and non-right-angle corners), the artificial intelligence and path-finding code started to fail, or become unreliable and inconsistent in a number of different scenarios. Enemy agents would become stuck in level geometry, or fail to plot a movement path between navigation nodes and remain static, or select incorrect animation sequences (e.g. selecting the 90° turn animation when navigating a 45° angle). Thus, following the initial feature prototype creation, the game’s designers, programmer, and scripter attempted to find a balance between interesting game environments (i.e. environments the contained a selection of objects and environmental features) and functional enemy agents. This process enabled the problem to be more concisely identified.

The problem was found to stem from the HPL2 engine’s locomotion-driven animation system, coupled with a lack of direct control over how enemy agents ‘perceived’ and utilised the navigation mesh (i.e. the node network used for pathfinding). To select an appropriate animation sequence for an enemy agent, HPL2 queried variables such as velocity and rotation. If an enemy agent was moving above a set velocity, then the engine would assign the relevant ‘run’ animation sequence. However, the enemy agents in \textit{A:AMFP’s} game as designed had a selection of walking, running, charging, and attacking animations that were intended to be selected under more directly controlled circumstances. This proved impossible to implement reliably during prototyping due to HPL2 automatically selecting the animation sequences.

This issue was compounded through the discovery during the prototyping process that, while the HPL2 documentation suggests a number of different controls available to the designer for controlling enemy agent behaviour, in reality many of these control methods do not work as described, or have caveats. During the assessment of project context (DisDev model, Stage 1), the analysis of the HPL2 engine capabilities relied on the available documentation. Thus, with this documentation being discovered to be inaccurate or incomplete much later in the development process, development time was used up ensuring that basic enemy agent functionality could be stabilised and made consistent, rather than further implementing the different enemy personalities.
in completed scenarios in the *game as created*. The navigation difficulties also meant that environment design had to be simplified to a state where the different enemy behaviours (i.e. the ‘Rod’, ‘Jane’, and ‘Freddy’ personalities) were less apparent. The enemy agents could only reliably navigate in large, open environments, or in sequences of wide, relatively uncluttered corridors with 90° corners.

As the prototyping of other game features and further game environments continued alongside the enemy behaviour prototyping, the number of different enemy encounters that players would experience throughout the game also decreased significantly. This was partially due to the failure of the personality-driven artificial intelligence system and partially as a result of the removal of the infection system (which was intended to be linked to the manner in which the enemy agents attacked the player, via infectious attacks rather than dealing physical damage).

This reduced number of enemy encounters similarly reduced the potential for players to firstly identify and then, to understand and respond to, different enemy personality types. This, in conjunction with the simplified environments making initial identification of different personality types much more difficult than originally intended resulted in the personality system becoming a substantial amount of work for minimal player experience benefit. Thus, during prototyping it was decided to remove these different personality types from the *game as created* (although the controlling code for the aggressive ‘Freddy’ personality was retained as a basis for all of the game’s eventual enemy agents) and instead create a series of bespoke script-driven enemy encounters.

Some of these encounters would demonstrate behavioural differences between otherwise identical enemies but to a much lesser extent than originally intended in the *game as designed*. However, the majority of these bespoke encounters would instead utilise three visually different enemy types, rather than the originally designed single visual enemy type.

The development of these different enemies required new art and animation assets which was made possible by additional funding provided by FG approximately ten months into the development process. This funding was provided by FG for expanding the game’s complexity and play time from the originally pitched two-to-three hour experience into a more substantive, standalone, six-to-eight hour experience.

### 7.5.6: Addition of Multiple Enemy Types

As was the case with the ‘infection’ system, enemy agent encounters were a core part of the game experience that had been proposed in the *game as designed*. Thus, the proposed change from a single enemy type to multiple types required a return to the design stage (i.e. DisDev model, Stage 2) to ensure that the changes would still afford potential for players to experience disruption. Moreover, from a non-disruptive design perspective, it was necessary to consider what impact(s) such a significant change to the game’s structure (i.e. frequency and location of enemy encounters) may have on other aspects of the game, such as pacing and told story delivery.
Reverting from the original disruptive design intent (i.e. utilising only a single visual style for enemy agents with different behavioural traits) may appear to prevent the originally intended disruption of transludic semantic knowledge, relating to visual recognition of enemy types. However, with fewer total enemy encounters throughout the game, it became more achievable to implement bespoke scripting and enemy agent controls in individual encounters. Thus, even though players may be able to use visual recognition to identify enemies quickly, bespoke control over individual encounters allows identical enemy types to still appear to behave differently to one another in the game as created. Rather than having enemy agents behave ‘procedurally’ through a set of behavioural rules (i.e. the ‘Rod’, ‘Jane’, and ‘Freddy’ personalities), enemy agent ‘behaviour’ could be hard-coded into the script controlling each individual encounter.

Thus, Recall Disruption and Action Plan Disruption of intraludic knowledge (semantic knowledge of the properties of different enemy visual styles and episodic knowledge of previous enemy encounters in A:AMFP) may still be experienced by players. Indeed, disruption of transludic knowledge may also in fact still occur via Recall Disruption and Action Plan Disruption, based on the transludic understanding that enemies sharing visual properties tend to also share behavioural properties.

The use of enemy agents in A:AMFP presented a range of non-disruptive design challenges as well, not only in terms of the enemy agents’ behaviour but also in terms of how they supported the wider player experience. Specifically, the type of ‘horror’ experience in the game as designed was not intended to be visceral, fast-paced, ‘gory’ horror, as might be associated with games such as *Dead Space* (EA Redwood Shores, 2008) or *Left 4 Dead* (Turtle Rock Studios and Valve Corporation, 2008). As emphasised even in the original pitch for *We Are The Pig* (Section 7.1.4), the design intent instead was to create an experience that provided a more cognitive, psychological and lingering type of horror; a game that players may continue to find unsettling even after they stop playing, in-keeping with the Lovecraftian themes in the game’s milieu. The enemy agents used in the game thus needed to support this type of horror, whilst also presenting a significant, immediate threat to players during specific gameplay sequences.

The portrayal of ‘enemy-to-player’ and ‘enemy-to-enemy’ relationships is able to communicate a range of gameplay and story-based information to the player. Moreover, information communicated through non-verbal enemy-to-enemy interaction retains a degree of ambiguity, requiring players to interpret the possible meaning(s) of what they see and hear. Thus, this provides a means for implementing Encoding Disruption of intraludic semantic and episodic knowledge via ambiguous stimulus information. It also provides a catalyst for further supporting the player’s cognitive engagement with the game world, and the game’s told story.
Additional Design for the use of Multiple Enemy Types

This additional design can be formalised within Stage 2 of the DisDev (Figure 7.50) model as a disruptive game feature that targets Aesthetics and Mechanics (i.e. both the appearance and behaviour of enemy agents), with an Auditory and Visual mode of output, aiming to create Encoding Disruption, Recall Disruption and Action Plan Disruption of Intraludic Knowledge (INT-S and INT-E) and Transludic Knowledge (TRANS-S and TRANS-E).

7.5.6.1: Wretches and Early Game Enemy-to-Player Interaction, Prototype 1

The enemies encountered in the first four levels of the game, known as Wretches (Figure 7.51), were designed to be actively attempting to escape from the player. This is similar to the previously defined ‘Rod’ enemy personality type, meaning that much of the behavioural ‘rules’ for this enemy type had already been prototyped to a functional level.

However, these sequences would now be individually scripted using bespoke control scripts for each encounter. This required adapting the behavioural logic previously created at the level of the HPL2 engine code into the higher level AngelScript gameplay scripting language. With significantly reduced functionality compared to the engine source code, this adaptation process required a
new prototyping process to be undertaken with the engine programmer and scripter working collaboratively to achieve the desired enemy behaviour.

This prototyping process was carried out using the game’s first two levels (Mansion01 and Mansion02) as testing areas, as these offered less complex environments for enemy pathfinding and navigation. These levels were also intended to be the main levels in which the Wretch’s ‘running away’ behaviour would be encountered by players.

The prototyping process identified some minor issues with the reduced degree of flexibility in the AngelScript code compared to the HPL2 engine source code. This was remedied through the engine programmer adding extra bespoke script command support into the engine source code, thus enabling greater control for the scripter via the AngelScript code. With these additional commands in place, the Wretch behaviour was evaluated as being ready to enter full development. The ‘enemy-to-player’ interaction relied on the embedding of enemy encounters into more fully developed game environments and scenarios so that a closer approximation of the ‘end-user experience’ could be identified. Thus, it was expected that the individual bespoke enemy encounters would each require a number of iterations and playtests before being finalised. However, these iterations were expected to be minor, requiring small adjustments to variables such as when and where an enemy spawns, how quickly it moves through the environment, and how much sound it makes as it does so. These iterations are therefore not considered individual ‘prototypes’ but instead a natural process of refinement during the full development stage.

7.5.6.2: Wretches and Early Game Enemy-to-Player Interaction, Full Feature Development

The early sequences in which Wretches interact with the player are carefully scripted to ensure that it is not clear whether they are running away from the player or running towards something else (e.g. a sound that they think may have been caused by the player-character). This is further supported with additional game systems, including an ambient sound generation system that triggers a variety of sound effects (e.g. muffled footsteps, creaking doors and windows, and machinery sounds) from sound sources around the player’s current location. Thus, players are placed in a situation designed to make them question whether the enemies are a threat that they are successfully avoiding, or are in fact not a threat at all. Encoding Disruption of intraludic semantic knowledge is thus provided through the lack of explicit information. Furthermore, Recall Disruption is provided for those players with existing transludic semantic and episodic knowledge regarding enemy agents in other horror games; such transludic knowledge may suggest that enemy agents in such games are usually overtly aggressive towards the player-character.

This potential mismatch between existing transludic knowledge and the perceived enemy behaviour, combined with the lack of explicit information to explain that behaviour, requires players to continuously analyse and evaluate each new enemy encounter and to then, create new intraludic knowledge regarding the enemy agents following each encounter. Because these encounters are occurring at a very early stage in the game, this also provides a number of
opportunities later in the game to further make use of Recall Disruption and Action Plan Disruption of intraludic semantic and episodic knowledge by modifying enemy agent behaviour in different ways and within different contexts.

Between the fourth and eighth levels of the game, Wretches become aggressive towards the player (utilising the previously prototyped and functional code for the 'Freddy' personality type). However, once again these scenarios are carefully orchestrated to avoid suggesting to the player that they are simply aggressive enemies to avoid.

![Figure 7.52: The game's fifth level, Church, is designed to force enemy-to-player interaction to occur in confined spaces (A:AMFP September 2013 Release Build).](image)

Some scenarios, such as in the game's Church level (the fifth level in the game), place players in a situation that requires them to move past a Wretch in a small, confined space (Figure 7.52). The intention of this particular scenario design was to suggest that the Wretch was only becoming aggressive towards the player-character because it felt threatened. Thus, it was acting in self-defence rather than openly attacking the player. This interpretation is supported through the design of the encounter itself, which is foreshadowed by the Wretch appearing to run away from the player-character. However, the lack of any specific, explicit information explaining the motivation of the Wretches again leaves their behaviour open to individual interpretation by players and thus, possible Encoding Disruption via ambiguous stimulus information.
7.5.6.3: Engineers and Enemy-to-Enemy Interaction, Prototype 1

As the behaviour of the Wretches becomes more aggressive and to an extent therefore, more predictable, players will potentially be able to construct more accurate intraludic semantic and episodic knowledge about them and be able to plan actions to avoid them more appropriately. Thus at this stage of the game, the Wretches’ behaviour is modified once again alongside the introduction of a second enemy type, the Engineer (Figure 7.53). This requires players to simultaneously begin constructing new knowledge about this new enemy type whilst having to re-evaluate and reconstruct existing knowledge (i.e. Recall Disruption of intraludic semantic and episodic knowledge) about the established Wretch enemy.

![Figure 7.53: An Engineer charges at the player-character (A:AMFP September 2013 Release Build).](image)

The behaviour of the Engineers was intended to be more clearly recognisable within the context of horror-themed games, with all Engineers being openly and immediately aggressive towards the player-character when encountered (behaviour that would be based on the code for the previously prototyped ‘Freddy’ personality type). The behaviour of the Engineers was not itself intended to be a disruptive game feature. The Engineers were instead intended to provide a disruptive game feature, targeting Recall Disruption of transludic semantic and episodic knowledge along with Action Plan Disruption of intraludic semantic knowledge, by knocking the player-character off his feet when they attacked. Recovering from being knocked over would require player input in the form of mouse movement (moving a cursor to a sequence of ‘hot spots’ on the screen) with quicker, more accurate input resulting in faster recovery and thus quicker escape from the enemy.
This mechanic, attached specifically to the Engineers, was designed to provide short-term Action Plan Disruption through changing the association between enemy agent properties (i.e. intraludic semantic knowledge) and the established player actions required to escape from an enemy after being attacked (i.e. players would not be able to simply turn and run, instead having to perform an additional recovery action first). This disruption would be short-term only (i.e. players would likely recognise the enemy type and method of attack and rapidly establish new intraludic procedural knowledge to handle it) but was intended to have a significant disruptive effect on the player during that short period of gameplay. However, as this new mechanic was not considered during the creation of the game as designed (as the Engineer enemy type was not a part of the game as designed either), a new prototyping process was required. Specifically, it was necessary for the engine programmer to assess whether the combination of the Newton Game Dynamics Physics Engine that HPL2 uses and the existing AngelScript commands that enabled the scripter to directly control the movement of the player-character, were able to produce the effect of being knocked down by an enemy in a convincing manner.

Unfortunately, it became evident that technological restrictions of the game engine prevented this ‘knock down’ mechanic being implemented, primarily due to the highly limited range of script commands available for manually moving the player-character and the attached camera. In addition, the required communication to the player of the size, power and weight of the Engineer enemies was not fully realised. This was fed back by multiple play testers, both within the development team and within the FG testing team. This was a failure in part due to the technical restrictions of the game engine and in part due to the design decisions made. The HPL2 engine’s character animation system prevented control over the specific selection and blending of individual animations, due to it being ‘locomotion driven’ (i.e. animations selected and blended in relation to the velocity of the character’s movements). While TCR’s engine programmer spent a significant amount of time attempting to enable more flexible animation selection, it was eventually found to not be possible within a practical time period for the project. Therefore, the Engineer’s movements and animations failed to fully deliver the intended message of the enemy’s power, size and weight.

While it may have been possible for new bespoke script commands to have been added by TCR’s engine programmer to work around this limitation, the development time required to do so at this later stage of the project was too great. Had the use of the Engineer enemy and the ‘knock-down’ mechanic been part of the initial game as designed it may have been possible to allocate development time earlier in the project to work around the technological restrictions. However, this nevertheless serves to exemplify the evolutionary nature of the design and development process and how any design philosophy, not only the disruptive game design philosophy, must be able to respond to changes throughout a project.

Thus, this first prototype of the Engineer enemy’s attack method was removed from the game. However, the Engineer enemy itself had already been created and FG had also already funded the
inclusion of multiple enemy types in the game. Therefore, the Engineer enemy had to enter full development and use in the game as created so as to not waste the development budget and time that had been spent on developing the assets for it.

7.5.6.4: Engineers and Enemy-to-Enemy Interaction, Full Feature Development

With the Engineer’s ‘knock-down’ mechanic unable to enter full development, the Engineers in the eventual game as created provide a functional but rather ‘simplistic’ enemy type that behaves in a manner that players are likely to expect and respond to easily; removing the Engineers’ ability to knock the player-character over means that the potential for disruption is also eliminated in this case. However, it may nevertheless be useful to analyse how players respond to this enemy type in the context of the previously described Wretch enemies (which provide a source of disruption) and the Tesla enemies (Section 7.5.6.5) (which also provide a source of disruption).

However, while the Engineers are no longer ‘disruptive’ as standalone enemy agents in the game as created, they are still utilised as a means of portraying ‘enemy-to-enemy’ interactions for the player to observe. The Engineer is introduced to players in the game as created in a sequence which depicts an Engineer and a Wretch passing one another in a corridor. The Engineer roars at the Wretch before using its much greater weight and power to swipe the Wretch out of the way, knocking it to the floor. The Wretch then cowers in fear as the Engineer passes by, before running away.

This sequence was designed and created with the aim of having two important impacts on the player. Firstly, it aims to emphasise the size and power difference between the established enemy type (i.e. the Wretch) and the new enemy type (i.e. the Engineer), demonstrating to the player that the new enemy type poses a much greater threat than the Wretches. Secondly, it is designed to encourage players to further question their understanding of the previously experienced Wretch behaviour (i.e. Recall Disruption of intraludic semantic knowledge), given that the Wretches are at this point portrayed as existing at the bottom of a power hierarchy amongst the enemy agents.

7.5.6.5: Tesla Pigs and the ‘Boss Enemy’ Game Trope, Prototype 1

Following the disruptive game feature based on the changing behaviours of the Wretches during the earlier stages of the game, the introduction in the penultimate two levels (Tesla01 and Tesla02) of the third enemy type, the Tesla Pig (Figure 7.54), is intended to provide an intense but short-term disruptive impact on players.
An observable pattern can be identified across a range of games and game types regarding the use of ‘boss enemies’, either at the end of game sections or at the conclusion of the game itself. Adams and Rollings (2007, p.447-448) describe these enemies in terms of the challenge they present to players, either forming a test of the player’s recently acquired skills (i.e. in the levels preceding the ‘boss’) or requiring the incremental development of those skills in a new way. The incremental development of previously acquired skills and intraludic knowledge is summarised well in the following statement, made in reference to the first sequence of levels in *Rainbow Islands: The Story of Bubble Bobble 2* (Taito, 1987).

The boss character is simply a much larger version of a spider the player already defeated. This enhances gameplay by allowing the player to predict some of the boss’s behaviour and gives him a small advantage in knowing what to expect. (Adams and Rollings, 2007)

Other games may use bosses in a similar ‘testing’ capacity but will require the player to utilise all of their existing skills (i.e. rather than just their most recently acquired skills). Some particularly notable examples that demonstrate this structure can be seen in *Okami* (Clover Studio, 2007) (which requires players to use all thirteen of the different acquirable ‘brush techniques’, or skills, in the game), *Metroid Prime* (Retro Studios and Nintendo, 2003) (which requires players to use all of the acquired weaponry against the boss enemy’s first form and all of the acquired scanning-visor types against the second form), and *Chrono Trigger* (Square, 1995) (in which the game’s final boss enemy switches between copies of the game’s previously defeated bosses, providing a test of the player’s ability to remember how to combat each of them as well as a test of the player’s party’s set up, equipment and abilities).

In *A:AMFP*, the Tesla Pig was designed specifically to disrupt this game-medium-based trope and expectations players may have based upon it. As players reach the final stages of the game, they may have developed a ‘preferred’ method of tackling dangerous situations involving enemy agents,
based on the player’s understanding of the semantic properties of those enemy agents. The majority of enemy encounters during the game to this point are designed to be most successfully approached using caution, stealth and slow movement to avoid attracting enemy attention. The two Tesla Pig encounters were both designed to provide Action Plan Disruption of these different pieces of constructed intraludic semantic knowledge by being best approached using different tactics. Specifically, avoiding the use of stealth and caution in favour of sprinting through the dangerous areas containing these enemies as quickly as possible, as Tesla Pigs were designed to always hunt the player and be aware of their location, making stealth an ineffective strategy.

Tesla Pigs initially were not expected to require significant additional prototyping to take place. Their basic behaviour was intended to be drawn from the code for the aggressive ‘Freddy’ personality type, with the addition of deactivating the queries to the ‘line of sight’ between the enemy and the player (so that the Tesla Pigs would never lose ‘sight’ of the player-character) and the ‘audible radius’ of the player-character (again, so that Tesla Pigs would always be able to ‘hear’ the player-character and hunt them effectively).

Indeed, this approach was applied to Tesla Pigs in a simple, corridor-based maze-like area and it produced predominantly positive feedback from playtesters. The encounter with the Tesla Pig enemy was cited as being intense and a significant, enjoyable departure from the enemy encounters in the earlier stages of the game. However, FG stated that the encounters could still be improved, suggesting that enabling the Tesla Pigs to ‘teleport’ around the environment at will, rather than having to physically move around, would provide a greater challenge to players whilst also further differentiating the enemy type from the Wretch and Engineer enemy types.

7.5.6.6: Tesla Pigs and the ‘Boss Enemy’ Game Trope, Prototype 2
A second prototype was thus developed with design and development collaboration from FG’s game engine programmer that implemented the ability for the Tesla Pig to teleport. The code-level implementation of this feature was primarily carried out by FG and thus, the details of the implementation are not available for inclusion in this discussion. However, following approximately two weeks of development time, a functional second prototype was ready that utilised the same corridor-based, maze-like environment but enabled the Tesla Pig to teleport at will. This was responded to positively by playtesters that cited the significantly increased challenge in tracking the enemy, as well as the pleasurable shocks associated with having the enemy teleport in close to the player. From the perspective of disruptive game design, the additional ‘teleporting’ capabilities do not detract from the Tesla Pig’s intended disruptive impact. Indeed, the capability may further enhance the disruptive effect as it further differentiates the Tesla Pig enemy type from those previously encountered by players. This prototype was thus approved for full development and inclusion in the full versions of the Tesla01 and Tesla02 levels.

The disruption attached to the Tesla Pig enemies is also supported by the use of the flashing-lantern ‘warning system’ that is described in Section 7.5.12.
7.5.6.7: Tesla Pigs and the ‘Boss Enemy’ Game Trope, Full Feature Development

Full development of the Tesla Pig enemy type required minimal additional work following the completion of the second prototype. The functionality of the enemy type had been playtested extensively at this stage, thus all that was required was to develop the final scenarios and environments that the enemy type would be encountered in. The corridor-based, maze-like environment used for prototyping was used as a basis for one of these environments, with large glass panels in many of the walls to allow players to see glimpses of the Tesla Pig as it teleported around the environment.

![Figure 7.55: The Tesla Pig encounter area in Tesla01 (A:AMFP September 2013 Release Build).](image)

A second Tesla Pig encounter was also developed which placed the player in a large chamber with many pillar-like structures placed around it. This design once again was intended to allow the player brief glimpses of the enemy whilst preventing the player being able to easily track the enemy continuously.

7.5.7: Removal of ‘Hardcore’ Difficulty Mode

As the game evolved and specifically, after the additional funding was received to extend the game length to a significantly longer (than the initial game as designed) six-to-eight hour experience, the viability of players being able to complete the game in a single session decreased. Players dedicating up to eight hours in a single period of play was considered unlikely. Moreover, with its longer play time and greater number of potentially lethal enemy agent encounters (rather than ‘fake’ encounters where the player is never in real danger), the Hardcore mode was considered to exhibit a greater alignment with the abusive game design philosophy rather than the disruptive game design philosophy. Thus, while the possibility of including this mode in a different format was discussed at different points throughout development, it never entered into the prototyping process formally and was not included in the game as created.
7.5.8: Providing the Player with Hints

The method of the game providing the player with hints during the process of solving puzzles proposed in the game as designed required careful planning in terms of how it would be utilised in the game as created. If minimal explicit hints were to be provided to players, it was critical to minimise the risk of players thinking of a potential solution only to find that the game designers had not also considered that potential solution. This may lead to players performing actions that the game has not been programmed to provide appropriate feedback for. The aim of minimising explicit in-game hints for players was intended to provide both Recall Disruption and Action Plan Disruption of transludic semantic and episodic knowledge related to the process of solving puzzles in similar games to AAMFP. However, if players find themselves in situations where they feel they have identified a logical solution to a puzzle only to find that the developers have not catered for such a solution, this is likely to be frustrating.

While it would not be realistic to aim to pre-empt every possible player-conceived solution to different game scenarios, designers can control what tools and methods a player has available to them and thus consider possible player actions involving them. TCR had accounted for these design considerations when planning this approach to the game and prototyped a test scenario in an early iteration of the game’s Cellar level (the third level of the game) as a means of evaluating the playtesters’ responses to such puzzles.

7.5.8.1: Providing the Player with Hints, Prototype 1

The first iteration of this scenario required players to manually remove a broken fuse from a machine and then locate a new fuse, carry it back to the machine and insert it into the fuse slot.

![Figure 7.56: A broken fuse emitting smoke and sparks in the Cellar level (A:AMFP June 2012 Build).](image)

Environmental cues, in the form of smoke and sparks emitting from it were utilised to indicate that the fuse was broken (Figure 7.56). Once repaired, the machines could be activated to open steam-operated door mechanisms around the level.
Figure 7.57: The direction of the pipeline between the machine and the off-screen steam-activated door (red arrows), with steam jets (red circles) (A:AMFP July 2012 Build).

Once activated, further environmental cues are used to guide the player from the machine to the door that it has opened. The two locations are physically connected with a large pipe which can be used by players to find their way to the opened door, although steam jets are also added along with associated sound effects to further emphasise the pipe to players as both a visual and auditory clue (Figure 7.57).

This use of the environment provides hints to players in a more diegetic fashion than through the use of the memento system, an internal monologue of the player-character, or similar system. From a disruptive game design perspective, this approach to providing information to players fulfilled the requirements for Encoding Disruption via ambiguous stimulus information, as it required players to constantly analyse (i.e. the lowest level of higher-order thinking) possible meanings of the non-explicit environmental cues. Some of these cues may be more readily understandable than others (e.g. smoke and sparks being emitted by machinery is more likely to be easily interpretable as meaning the machine is broken or damaged in some way), however the onus was placed on the player to cognitively engage with the game and question what they may ‘know’ (i.e. the lowest level of lower-order thinking) about the game and the cues it is providing, rather than being explicitly told the ‘correct’ interpretation of in-game stimuli.

7.5.8.2: Providing the Player with Hints, Prototype 2

However, FG did not approve this designed approach to providing the player with no explicit information. Feedback on this build from FG requested much more overt signposting for players, especially in levels that required players to perform a number of different interactions that had effects on game entities that were out of immediate sight of the player when the interaction was performed. The suggested alteration was to use the game’s voiceover system and add short voiceover recordings of the player-character thinking about the different scenarios (i.e. an internal
monologue hint system). For example, when first approaching a machine with a broken fuse, a trigger would start a voiceover stating:

Broken machinery? Looks like I’ll need to replace this fuse to make it operational again… [Oswald Mandus character voice clip, A:AMFP July 2012 Build]

TCR were not supportive of this alteration stating during meetings with FG that it detracted significantly from the cognitive challenge of the game, as well as potentially patronising players by not allowing them to work out the solution to problems themselves. Similarly for the current research, this suggested alteration would effectively eliminate the Encoding Disruption (and associated cognitive engagement) experienced by players, as they would have little opportunity to perceive and analyse the less explicit, environment-based information and construct their own understanding and plans of action based on it. These voiceover recordings would trigger shortly after players encountered a new problem or unusual object in the game. This would cause a similar problem as described in relation to A:TDD and the organic matter (Figure 7.24) in that they immediately place constraints on the player’s thinking by suggesting a ‘best’ course of action.

Gameplay in this case becomes a matter of following instructions (i.e. the instructions are ‘known’ by the player and can be readily ‘understood’ and then ‘applied’, fulfilling only the three stages of lower-order thinking), rather than a process of analysis and evaluation of scenarios followed by forming and then performing one’s own plan of action. As described previously as player-supportive design (Section 1.3), players would in this case be likely to have needs fulfilled such as achievement and mastery through the completion of these prescribed gameplay tasks. However, there is minimal scope for players to significantly engage in processes that may assist them in fulfilling needs such as creativity, spontaneity and problem-solving.

After the voiceover method was implemented in a selection of prototype scenarios and presented to FG however, they agreed that the approach was not working and indeed significantly limited the analytical skills required by players during gameplay. FG did however still request that a different solution (to the first prototype) was implemented (i.e. a request for the game as published to differ to the original game as designed and the intended game as created) to make the puzzle scenarios less ambiguous and provide players with some more explicit guidance.

7.5.8.3: Providing the Player with Hints, Prototype 3

Thus, to approach a solution that would be acceptable for both TCR and FG, whilst also supporting the aims of the current research, consideration was given to how the previous A:TDD hint system could be used as a basis. From this, consideration was then given to designs that could build upon the A:TDD system that would require additional effort by players if they wanted to access hints during the game. This was intended to allow the player to choose how they wanted to play; either, without explicit hints, or with hints, but only as and when they chose to use them. At a game system level, this also offered potential for allowing a simple method of implementing differentiated difficulty modes based on the amount of assistance available to players. With
‘Hardcore’ mode removed (Section 7.5.7), such simple differentiation of difficulty modes would enable a degree of flexibility to cater to different player tastes.

The decision was made to reinstate ‘hints’ within the game’s journal system (described in Section 7.4.9). Thus, players would be able to find journal pages that expanded the game’s story and lore by exploring the game environments. These found pages would be stored in the journal’s Found Documents section. Hints would be added to the My Journal section of the journal as the player made progress; indeed, the adding of a hint to the journal may occur on many of the same triggers originally set up for triggering the internal monologue voiceover clips in the previous prototype (Section 7.5.8.2). At this stage, this setup was only minimally different to that used in A:TDD.

Figure 7.58: A:AMFP’s game options menu, with ‘Show hints’ defaulted to inactive (AAMFP September 2012 Build).

Following FG’s previous feedback, it was considered a reasonable compromise to suggest the inclusion of the journal-based hints but to then, set the default game mode to one that did not automatically provide these hints to players (Figure 7.58). If players accessed the My Journal section of the journal, a message would be available encouraging players to rely on what they can find in the environment to aid them in solving the game’s challenges but also, providing an instruction on how to turn the hint system on via the game’s menu if they wanted to. This provided players with a choice as to how they would play the game. It also meant that the problem of potentially patronising players by automatically providing hints was avoided.

In relation to the current research, this solution demonstrated a system-level balancing of the needs of multiple potential players and player types. By hiding the hints initially, construction of new intraludic knowledge through recalling previous transludic and extraludic experiences was encouraged and thus, cognitive engagement with analysing, evaluating, and creating new
understanding of, the game’s various stimuli was also more significantly encouraged. However, having a system in place but not activated by default allows for alternative play styles. For example, some players may be playing primarily to enjoy the game’s story and thus may have no interest in spending time analysing game scenarios and solving puzzles. In this case, the hint system can be activated and used to aid progression. This thus demonstrates one potential method of utilising disruptive game design within a commercial product in a manner that is less likely to be poorly received by some demographics within the player base. While some of the previously described disruptive game features may not be able to be balanced in this way due to them being intrinsic components of gameplay (e.g. the enemy agent behaviours), more abstract game features that operate separately to the main game or that are ‘heterodiegetic’ support systems, such as the hints, can be more easily differentiated for different player types.

FG however again did not approve of this system structure, stating that the hints being added to the game’s journal should not be an option and should occur regardless of the game options selected. TCR suggested having the option to turn off the journal hints (i.e. having them default to ‘on’ and then allowing players to deactivate them if they wanted to) however this too was not agreed on by FG. It is unclear why FG were not receptive to the suggestion of allowing the hint system to be optional for players as this would appear to be a solution that is of maximum benefit to the largest number of players. The disagreements between the two companies on this particular design decision are further discussed in the post-mortem report in Appendix F.

7.5.8.4: Providing the Player with Hints, Full Feature Development

The version of the game’s hint system that was eventually included in the game as created is significantly different to the original design intention (i.e. in the game as designed). Hints are added to the player’s journal automatically by default and this function cannot be deactivated by players, as was requested by FG following the proposal of the preceding three prototypes.

Figure 7.59: A usability hint in A:AMFP explaining how to access the hints in the journal (September 2013 Release Build).

The option to ‘Show hints’ now only affects the ‘usability’ hints that the game provides (for example, Figure 7.59) which are likely to be the hints that most players will want to see, as they ensure players are able to use the game’s basic controls and systems correctly.
While FG did not provide TCR with flexibility regarding the use of hints, the textual content and the style in which the hints are presented were both able to be designed more freely. Thus, rather than the short, relatively explicit sentences provided in A:TDD, A:AMFP's are written in a 'diary entry' style usually consisting of a paragraph of text representing the player-character's current thoughts (see Figure 7.60 for two examples).

This approach attempts to keep the hints situated within the homodiegesis of the game’s told story while also attempting to provide players with a slightly more cognitively engaging piece of text. However, this cognitive engagement can no longer be described as stemming from disruptive game design. The text itself used in the hints does not disrupt existing player knowledge.

From this journal text, players can extract key pieces of information (i.e. the hint) from the surrounding 'flavour text' (i.e. the text that provides story and/or contextualising information). While this is a significant departure from the original intention in the game as designed of entirely environmentally situated sources of information, this version of the hint system provides players with at least a slightly increased required degree of cognitive engagement in the problem solving process, although that engagement is not based in a disruptive game feature.

The disagreement between the development and publishing teams on this particular element of the game serves to indicate a potential challenge that may face developers in future looking to implement game features based on the principles of disruptive game design. Making changes to what may be perceived by other parties involved (e.g. publishers) as fundamental elements of a game may cause significant disagreement. An analysis of how A:AMFP's difficulty and in particular, the hint system, and how it is received by players is thus a key part of the post-release analysis of the game (Section 8.10).

7.5.9: Handling the Death of the Player-Character

The player-character death sequences had to be modified from the original game as designed due to availability of development time and resources. While these factors had been assessed in the
project planning stage, they imposed greater restrictions on the development of the game as created than had been expected.

### 7.5.9.1: Handling the Death of the Player-Character, Prototype 1

The first modification prototype removed the visuals from the interim sequences between ‘death’ and reawakening in the nest area, so as to eliminate the need for development time and resources to be spent on creating the assets and scripting for them. Instead, the screen would fade to black and a short, audio-only sequence would be triggered that would suggest the player was being dragged somewhere in the level by an enemy agent. This prototype was quick to develop and made use of placeholder sounds.

This prototype was not approved by FG during a milestone review process, with testing staff within FG citing that it was now not clear enough how players were being moved to the nest area and thus, it was potentially too disorientating for players. TCR defended this prototype, stating that it was appropriate for players to feel disoriented following the player-character being knocked out by an enemy agent. However, FG requested that the system be reconsidered. Specifically, FG stated they would prefer the removal of the ‘nest area’ system and a return to a system similar to A:TDD.

This was one of the few occurrences during the development process where the two companies strongly disagreed on the implementation of a specific disruptive game feature. However, having to respond to such situations in an appropriate manner is one of the benefits of a commercially-based Research through Design (RtD) process and thus, provides a potentially useful point for later discussion following data analysis (Section 8.10). This may be especially true if this particular feature is responded to in a strongly positive or negative manner by players.

### 7.5.9.2: Handling the Death of the Player-Character, Full Feature Development

The version of this system that is included in the game as created provides less potential for disruption of transludic knowledge than the game as designed. However, there are still some small differences between the system used in A:TDD and the A:AMFP system that may afford minimal Recall Disruption and Action Plan Disruption based on transludic-knowledge from A:TDD.

The ‘nest areas’ are not functionally implemented in the game as created (although some of the ‘nest areas’ within the game environments have been left in place simply to provide more space for players to explore), with players instead being reawakened at a checkpoint location in the main game level. Changes to the level still occur following a ‘death’ event, most of which consist of alterations to the enemy agent states, or the replacement of real enemy agents with fake enemy agents, whose presence are suggested through the use of script-triggered sound effects. These post-death alterations are not always the same and may also differ between one playthrough and another. Thus, there is potential for Recall Disruption and Action Plan disruption of intraludic (primarily) and transludic (secondarily) semantic and episodic knowledge. Specifically, this would target knowledge based on a player’s understanding of how the ‘death’ system worked.
in A:TDD (and other games), as well as how it works in A:AMFP and what actions may be successful or unsuccessful following a ‘death’ event. It is more likely however in this version of the system that intraludic disruption will be experienced by players, as opposed to transludic disruption. However this can be explored further following analysis of player feedback on the game as played (Section 8.10).

The system for handling the ‘death’ of the player-character is one of the game systems that appeared well suited as a basis for implementing disruptive game features due to its ubiquity across the game medium. It is thus unfortunate that it was one of the systems that caused significant disagreement between the development and publishing teams, and as a result, less was achieved than potentially could have been. It is possible however, that it was this ubiquity that made FG less willing to experiment further with the system, deciding instead to avoid the potential risks of making significant changes to such a fundamental component of the player experience (i.e. the game as played). This system is discussed further in the post-mortem analysis in Appendix F.

7.5.10: Hiding Places

The design and implementation of the hiding place system went through two prototypes. The first prototype was developed during the early stages of the development process, before the decision was made to change the enemy behaviour system and the number of enemy encounters in the game. The second prototype was developed as a response to the changes made in those aspects of the game.

7.5.10.1: Hiding Places, Prototype 1

The first prototype was fully tested in the game’s first enemy encounter in which the player is under threat (i.e. as opposed to the ‘fake’ encounters in early levels intended to build tension), with smaller tests carried out in other game levels.

Figure 7.61: Hiding Place System, Prototype 1; the two cages shown as white wireframes each have different ‘hiding properties’ (A:AMFP July 2012 Build).
In this example (Figure 7.61) from the game’s first enemy encounter, the two cages shown in white wireframe each have differing ‘hiding properties’. When the player-character is inside the cages, the enemies have a reduced chance of being able to see them. This is based on a separate script system rather than being directly linked to the main artificial intelligence system, which uses ray casts to test line of sight from enemy agent locations to the player-character’s location.

This provides much more flexibility and also means that the event scripter can have more direct control over how particular enemy encounters proceed during gameplay. In the example scenario, the cage on the floor is much quicker for players to access, but enemies are likely to see them and attack them if they get close enough. The elevated cage takes more time to get into (using a ladder on the opposite side of the cage) during which time the player is vulnerable, but once inside the cage the enemy agents will be unable to see or attack the player.

Variations of these hiding places were utilised throughout other game levels for further prototype testing. The player-character’s crouching height property was also adjusted to allow players to hide underneath furniture such as tables, offering more possibilities when players are being hunted by enemies. In making these changes in relation to A:TDD, players coming to A:AMFP with existing experience and expectations based on A:TDD would need to construct new knowledge and understanding about how they can use the environment to hide from enemies. This feature was designed to provide Recall Disruption and Action Plan Disruption of A:TDD-based (i.e. transludic) semantic and episodic knowledge by specifically including recognisable environmental assets, such as the large cupboards found throughout A:TDD, but having them locked or otherwise unusable as hiding places. Similarly, recognisable environmental assets that were of no hiding benefit to players in A:TDD (e.g. tables) become usable in A:AMFP, making the semantic properties of those assets significantly more important for players to construct new knowledge of.

The hiding place system design also meant that it would not be possible for players to be able to definitively conclude what the system’s ‘rules’ (i.e. the semantic properties of individual hiding places) were. The differences in ‘hiding properties’ between different locations remained ambiguous. While players may be able to infer that a cage located high up would potentially be safer, this is not made explicitly clear during gameplay. Thus, this system was further intended to create Encoding Disruption of intraludic semantic knowledge as players endeavour to form an accurate understanding of how ‘hiding’ functions and also, of how the enemies in the game function in relation to hiding places when hunting the player.

This prototype was approved for full development based on the functionality of the variety of test scenarios it had been applied to.

7.5.10.2: Hiding Places, Full Feature Development

The hiding place system was retained within the game as created based on the approved prototype. However, the system’s influence on gameplay and thus, the game as played, was significantly reduced when compared to the game as designed (and indeed, the prototype) through
the changes to the game’s enemy encounters (resulting from the significant changes made to the enemy agents described in Section 7.5.5 and Section 7.5.6). Thus, the opportunities for players to construct new knowledge about ‘hiding’ in A:AMFP are significantly reduced and hence, the opportunities for experiencing Encoding Disruption via ambiguous stimulus information are also reduced. This however is not a product of a failed disruptive game feature but rather a product of a number of design decisions made during the development process of the game as created.

7.5.11: Item Inventory Removal
Due to the significance of this disruptive game feature and its potential impact on a number of other game features and systems, it went through a variety of different prototypes. This disruptive game feature was also one of the most problematic when presented to FG for approval.

7.5.11.1: Item Inventory Removal, Prototype 1
The first prototype of this disruptive game feature was very quick and simple to implement, requiring only the deactivation of the relevant systems in the game’s code, which was in the early stages a direct duplicate of the A:TDD version of the HPL2 engine. Thus, the complete inventory system was removed along with the ability to store and read acquired story-related ‘notes’ or journal pages.

With this version of the game, the development team tested a variety of scenarios that were representative of the type of gameplay that the removal of the inventory would require and with these scenarios also, the implications for the delivery of the game’s story. While the primary disruptive aim of removing the inventory was gameplay focused (i.e. requiring players to approach item-based puzzles in a different way and requiring objects to be manually carried around the game environments by the player), it was important to also evaluate early on the potential impact on the ability of the game to deliver the game’s ‘told story’ in an effective manner.

The first of these test scenarios was carried out in an early version of the game’s third level, Cellar, in which players were tasked with repairing broken machinery by locating new fuses around the environment and carrying them back to the machines (see the full description of this puzzle in Section 7.5.8.1). Thus, the gameplay in this area revolved around the core player activities of exploration, carefully analysing the environment searching for what may be a fuse (as the game did not provide an initial example of what a fuse may look like) and then strategically planning a route back to the broken machinery that would avoid any interaction with enemy agents or any unnecessary interactions with other game entities, such as doors or levers (as this would be more difficult when carrying an item).

The avoidance of interaction with other game entities was of particular interest from a design perspective, as with the player-character using their hands to carry objects around the environment, any additional interactions would require first setting down the carried item, then performing the interaction before picking the item back up again. This had two potential beneficial outcomes. Firstly, it may provide scenarios that simulate to an extent a feeling of panic induced
through having to perform additional, accurate interactions whilst potentially being pursued by
enemy agents. Secondly, it may provide a greater level of cognitive engagement when compared
to the gameplay that may be provided by an identical scenario but utilising an inventory screen.

Such a scenario utilising an inventory screen would still require exploration and environmental
analysis but once the player locates the fuse, it is placed in the inventory and is removed from the
player’s attention. Returning to the broken machine then becomes a task of retracing one’s steps,
rather than having to plan a potentially different route back that may be safer. Manually carrying
items requires players to be more consistently engaged with thinking about how their seemingly
‘basic’ environmental interactions may have consequences, even when players are not currently
carrying items around. Leaving a door open, for example, makes it easier for quickly running back
through later whilst carrying items but also makes it easier and quicker for enemy agents to reach
the player. This example also includes potential for transludic-knowledge-based disruption as in
A:TDD it was frequently necessary to ensure doors were closed behind the player as they moved
through the game environments to keep enemy agents from locating and attacking them. While
the construction of new intraludic knowledge about the benefits and risks of leaving doors open
in A:AMFP would only need to occur once, the ongoing effect of that knowledge could have a
lasting effect on a player’s degree of cognitive engagement throughout their entire gameplay
experience. Leaving doors open or closed would potentially become a more considered choice,
based on closer analysis and evaluation of the game context at any given moment during gameplay,
as opposed to leaving doors closed always being the safest option.

The second level in which the inventory removal was prototyped was an early version of the
game’s first level, Mansion01. This scenario was used specifically to assess the impact of removing
the inventory system on the ability of the game to deliver the game’s story in an interesting,
engaging, and enjoyable format for players.

It was decided that the story would be delivered through a combination of environmental
storytelling (i.e. story interpreted by the player through what they see, hear and experience
through their interaction with, and exploration of, the game world), voice-acted internal
monologue of the player-character, and voice-acted but disembodied communication from the
primary antagonist, the Machine. Thus, when combined with the potential for creating more
cognitively engaging gameplay, removing the inventory was a coherent design choice as the other
main function of it (i.e. reading written story segments in notes and journal pages) would not be
utilised in A:AMFP.

The Mansion01 level testing provided positive feedback, allowing the game’s told story to be
delivered through the environment and via voice-acted segments. However, the only potential
problem with this approach was that the environments themselves would need to be extended in
some areas to allow time for voice-acted segments to be delivered. For example, the player-
character may move between voice triggers quicker than the voice-acted segments played,
meaning that they would overlap. It was expected that this issue could be overcome with a combination of expanded environment areas and accurate planning of voice trigger locations.

This inventory removal prototype was tested in two different scenarios and found to be successful (based on initial playtesting feedback) with regard to the different type of gameplay it created for players, although some practical, implementation issues were identified. Firstly, a potential design issue was noted in that creating a series of differentiated and varied puzzles and activities may prove more challenging without having an inventory screen. For example, item combination puzzles (e.g., combining chemicals with a conical flask to create a new chemical) are simple to implement using an inventory screen (by dragging and dropping items around the screen); however, without such a screen, diegetic means of performing such combinations would be necessary (e.g., art, sound and script assets for a functioning mixing device). This would require more development resources and thus, planning of such puzzle scenarios would need to consider the various assets required.

Secondly, it was noted that substantive gameplay testing would be necessary to ensure that any scenarios intended to induce panic in players as previously mentioned actually did have that impact and were not instead perceived as being confusing, frustrating or a product of ‘poor design’. This balance between ‘panic’ and ‘frustration’ (or similar negative reactions) was anticipated to be a challenge to achieve but a challenge that would have significant positive benefit for the play experience if achieved.

Lastly, if all puzzle-related items were to be physically placed in the game environments and able to be picked up and carried, it would be necessary to implement a ‘failsafe’ system that would prevent players becoming stuck if they misplaced items or ‘lost’ them (e.g., by purposely dropping them into inaccessible areas of the environment, or accidentally through game ‘bugs’). This would require a combination of the programmer’s and event scripter’s development time but was considered to be a task worthy of that additional time and resource investment given the potential benefits of this approach to puzzle scenarios.

The noted issue with fitting the delivery of voice-acted story segments into the game environments was expected to be solvable through simply expanding the sizes of the environments (e.g., by lengthening corridors or by otherwise extending the movement time between voice triggers). Thus, this prototype was considered successful in relation to its impact on the delivery of the told story of the game, with the proviso that such modifications to the environment would be made.

However, FG remained uncertain over the complete removal of the inventory system when the feature was proposed to them during the game’s first major publisher review. To attempt to address potential problems and to address FG’s concerns, a second prototype was developed.
7.5.11.2: Item Inventory Removal, Prototype 2

The second prototype aimed to design and implement a varied range of interesting gameplay challenges and puzzles for players to complete without the use of the inventory screen. With only minimal interactions available for the player, it was a challenging design problem to conceive of scenarios that involved more than simply locating an item and carrying it back to a specific location. It was also apparent that by removing the inventory, some scenarios in which a very simple task would have been otherwise sufficient (e.g. a locked door that requires a key) instead required more complex and often convoluted sequences to be used, as moving small individual keys around manually felt unnatural during play.

Figure 7.62: The Quick Bar in Neverwinter Nights (BioWare, 2002), allows players to use abilities and items with a single button press (in this case, the twelve function keys F1 to F12).

Thus, a minimal version of an inventory was suggested, that would only appear on screen when it was required (via a configurable key press), and did not require entering a separate screen to use. This design was comparable to the 'Quick Bar' used in Neverwinter Nights (BioWare, 2002) (Figure 7.62) for quickly using abilities and items during gameplay.

Figure 7.63: A concept mock-up of the 'quick bar'-style inventory system (A:AMFP April 2012 Build).

Figure 7.63 demonstrates a concept mock-up of this prototype being used in the game's Mansion01 level with the player having two items in their possession; a key and an empty test tube. Both of these items could then be selected and used quickly to complete simplistic tasks such as unlocking doors or activating machines.
However, this prototype presented a different design problem. With the small inventory available to players, differentiating between items that could be picked up and placed in the inventory and items that had to be carried around the environments manually became significantly more difficult. There was potential for players to become confused by items behaving differently to what they may expect based on existing intraludic, transludic, or extraludic knowledge in a manner that was frustrating, rather than beneficially disruptive.

FG were also critical of this prototype of this feature, stating they would prefer the game to involve more complex scenarios that did not require an inventory at all, even if some of those scenarios were functionally similar to one another (i.e. carrying an item from one location to another). This would require reverting to the first inventory removal prototype.

Reverting to the first prototype would be beneficial from the perspective of the research, as the complete removal of the inventory system was more overtly disruptive. Some concerns remained about the potential for keeping the gameplay tasks interesting and varied throughout the game. However, it was concluded that some functional repetition (i.e. carrying items from place to place) would be accepted by players provided that the context and setting (i.e. the aesthetic qualities) of the functional gameplay were varied enough between scenarios.

However, the implications for story delivery that were previously discussed (Section 7.5.11.1) needed to be addressed as well. While it was anticipated that modifications to the game’s environments would enable voiceover triggers to be spaced out more appropriately, this anticipated solution needed to be tested. Therefore, simply reverting to the first prototype was not possible and a third inventory system prototype was developed that now focused on resolving the issues with the game’s delivery of the told story.

7.5.11.3: Item Inventory Removal, Prototype 3

As development of the game’s levels progressed, it became evident that the amount of spoken dialogue that made up the majority of the game’s told story delivery would not fit within the physical size of the game’s environments. That is, the player-character’s movement speed combined with the size of levels meant that spoken dialogue sections were overlapping, or were continuing for long periods of time. This was detrimental to the game’s ability to build feelings of suspense, anxiety or tension through ambient sounds and music and thus, detrimental to the gameplay experience as a whole.

The expectation that expanding the game environments would be able to solve this issue was found to be unsuccessful. Expanding the environments by including longer corridors made many areas feel unnecessarily elongated and slowing the player’s movement speed also resulted in an unnatural feeling of the game ‘forcing’ the player to stay in an area long enough to listen to dialogue. The game story’s script was edited to lessen the frequency of voice-acted segments during gameplay, however even this process did not solve the problem completely.
Thus, the decision was made to reinstate the ‘journal’ component of A:TDD’s original inventory system and to convert some of the game’s spoken script into written journal pages that could then be discovered through exploration of the game environment (Figure 7.64).

This alteration had the additional effect of slowing the speed at which many players progressed through the game (reflected during playtesting) because there was now a clear motivation to explore the environments in detail to find more story-related information and thus, enrich the play experience.

Conversely, reinstating the journal component of the inventory system meant that the original intention of always keeping players within active gameplay was no longer achievable. However, with the item-based inventory system removed, players would remain able to stay within active gameplay during puzzle scenarios. The journal would at no point during gameplay be required to be opened; players would be able to choose when they read the journal pages that they discovered. However, this system also meant that players would now be able to easily ‘escape’ (both mentally and temporally) from threatening situations temporarily, by opening the journal screen. Thus this would mean the removal of a potential source of Action Plan Disruption (i.e. ‘pausing’ the game to think or to plan actions would now be a viable tactic, as it is in A:TDD and many other games).
The possibility of retaining this Action Plan Disruption by keeping the process of reading the journal within active gameplay was discussed at a later stage in development. Such a system is employed in a number of games, although *Miasmata*’s (IonFX, 2012) diary system (Figure 7.65) and *System Shock 2*’s overlay inventory system (Figure 7.66) were key references for this discussion.

However, implementing such a system would once more require significant investment of engine programmer time and budget, which were under-resourced within the TCR development team structure. Making the journal system keep the player in active gameplay, whilst being read, would have likely been beneficial to the player experience. However, the work required to implement such a system could not be justified within the constraints of the development process.
7.5.11.4: Item Inventory Removal, Full Feature Development

The version of the inventory system removal that was thus approved for full development and inclusion in the game as created, removed the item inventory completely while retaining the ability for players to view a separate journal screen that would allow delivery of story and hints.

This version of this disruptive game feature retains the original capability for creating Action Plan Disruption by requiring players to consider item-based puzzles in a different way. The further disruption that could have been provided by preventing players from accessing any separate screens at all during gameplay has been lost through the inclusion of the journal system. However, this demonstrates the necessity to ensure that disruptive game design is considered alongside other game design decisions. While the disruptive game design philosophy may be the principle driver behind such decisions, the overall impact on player enjoyment as well as the implementation of other, non-disruptive game features (such as the delivery of the game's told story) must be a factor against which all decisions regarding disruptive game features are evaluated.

7.5.12: Addition of ‘Flashing Lantern Warning System’ to the Player-Character’s Lantern

As game features moved out of the prototype stage and into full development, playtesting of the iterative builds of the game continued both internally within TCR, as well as externally utilising FG’s own testing staff. The playtesting process itself was ad-hoc, with iterative builds of specific game systems being tested as they became available.

During these sessions, FG’s staff stated that they felt that the flashlight that the player character carries with him throughout the majority of the game lacked purpose during gameplay. While the changes made to the lantern in A:AMFP compared to A:TDD were well received (i.e. changing to a directed beam of light rather than an omnidirectional light), the perceived lack of gameplay purpose was problematic. In A:TDD, the lantern was oil-fuelled, rather than the electrically powered one in A:AMFP and thus, players could not keep it on indefinitely. Players were tasked with locating oil to keep the lantern fuelled. This meant that there was an ongoing balance for players to be aware of between keeping the lantern on for safety and visibility but, to do so without using up all of their available fuel. A:AMFP’s electric lantern would never run out of power during the game (based on the power provided by Victorian era batteries) and this was problematic for testers, who stated that it was both damaging to their belief and immersion in the virtual world (despite the historical accuracy of the power provided by the lantern’s power source) as well as damaging to their gameplay experience. The game lacked threat and pacing when players had access to a permanent, perfectly functioning light source.

Following discussions within the development team, it was decided that adding ‘fuel’ in the form of additional batteries would not be supportive of the type of experience TCR were aiming to create for players. Specifically, the Creative Director did not want the game’s storytelling qualities to be
overshadowed by players becoming distracted by searching environments for as many batteries as possible before moving on. From a storytelling perspective, it was thought that this would in fact be more detrimental to the game’s pacing and serve to clearly expose the game’s underlying mechanics to the player.

Instead, the suggestion was made within the TCR development team to implement a degree of imperfection and unpredictability to the lantern, by making it respond to the presence of enemies in the environment by flashing, at varying intensities and rates. This would also serve to differentiate this key feature of A:AMFP from the lantern in A:TDD, which worked without fault while it was turned on.

With this as a basis, there was scope to implement a further disruptive game feature into the game as created that had not been considered in the game as designed.

The design of this new disruptive game feature was based on making the rules of the lantern flashing system able to subtly change at different points throughout the game. The design intent of this was to prevent players being able to reliably depend on the system as a ‘warning system’ to identify potential threats (although they may be able to in some cases). For example, early in the game, the more intense and fast the rate of flashing was, the closer the player may be to a nearby enemy. However, this ‘proximity’ trigger could then be modified throughout the game. For example, the radius from the player within which an enemy is required to be to trigger the flashing may be changed, in turn changing the time available for players to react to the warning.

These modifications to the system’s rules were designed to provide a source of Recall Disruption and Action Plan Disruption. Disruption of the established intraludic semantic knowledge (i.e. the perceived ‘rule’ of the proximity trigger) and intraludic episodic knowledge (i.e. previous experiences during the earlier stages of A:AMFP) would require players to engage in ongoing analysis and evaluation of the game environment and the behaviour of the flashing lantern system to establish (or to re-establish) new intraludic knowledge about it (i.e. creating new knowledge, the highest level of higher-order thinking).

![Figure 7.67: DisDev Stage 2 elements forming the design for the ‘Flashin...](image-url)
This additional design can be formalised within Stage 2 of the DisDev (Figure 7.68) model as a disruptive game feature that targets Mechanics with a Visual mode of output, aiming to create Recall Disruption and Action Plan Disruption of Intraludic Knowledge (INT-S and INT-E).

7.5.12.1: ‘Flashing Lantern Warning System’, Prototype 1

The requirements for creating a functional prototype of the flashing lantern warning system did not place any significant additional pressure on the development team. With no additional art assets required and only a small selection of electrical ‘flickering’ sound effects required, the majority of the prototyping process only required input from the engine programmer and the event scripter. Of these two development team members, the event scripter would be required to implement the system into specific enemy encounters via the game’s AngelScript Application Programming Interface (API). The engine programmer was only required to implement engine-level support for a selection of new script-level functions for the event scripter to use.

This first prototype included script functionality enabling the event scripter to control how any individual enemy agent would influence the flashing lantern system. Engine-level variables, such as the radius around the player within which an enemy agent would trigger the flashing system, were exposed within the script-accessible properties for each enemy agent type. This enabled the event scripter to make minor, or major, changes to the radius size over the course of the game, thus enabling the implementation of Recall Disruption.

With this proximity-based functionality available, the event scripter in conjunction with the Creative Director were able to begin creating opportunities within the early stages of the game for players to construct initial intraludic knowledge that would be disrupted later in the game. The early stages of the game (the first four game levels) provided time for players to identify the connection between enemy proximity and the flashing lantern warning system. These early stages presented the player with multiple enemy encounters designed and developed to emphasise the connection. This was to ensure that when the flashing lantern warning system is disrupted later in the game, the majority of players were likely to have constructed relatively uniform intraludic knowledge related to it and thus, would be likely to experience the effect of that knowledge being disrupted.

In designing the flashing lantern warning system specifically as a basis for providing Recall Disruption and Action Plan Disruption of intraludic knowledge, it was necessary to make the initial encoding of the intraludic knowledge as explicit as possible (in direct contrast to the requirements of Encoding Disruption via ambiguous stimulus information) so that players would be likely to share similar base intraludic knowledge that could be disrupted. This explicit encoding also needed to be implemented without detracting from the player experience as may be the case if, for example, the intraludic information was presented to the player via heterodiegetic (Section 5.1) means (e.g. an on-screen, text-based hint), thus potentially damaging the player’s sense of ‘immersion’.
Playtesting of the first four levels of the game confirmed that the majority of players were indeed identifying a connection between the presence of an enemy agent in the vicinity of the player and the lantern starting to flash. While not all players were specifically identifying the connection between the *proximity* of an enemy agent to the specific increase or decrease in the speed and intensity of the flashing, this was not significant enough to be considered problematic. It would be unlikely for all players to interpret the meaning of a system such as this in an identical way. With the majority of players identifying the link between enemy *presence* and the flashing lantern, there was potential for those players to experience disruption later in the game when the system’s functionality was modified.

7.5.12.2: ‘Flashing Lantern Warning System’, Prototype 2

Once this initial proximity-based system functionality had been established and the player had been presented with multiple enemy-based scenarios in which the system appeared to operate in this understandable way (up to the middle of the game’s sixth level), the rules of the system were modified. This modification was not explicitly communicated to the player. Instead, it was left up to the player to identify and attempt to construct an understanding of the new rules. The disruptive intent of this modification was to provide a basis for Recall Disruption of the player’s previously constructed intraludic knowledge by presenting a familiar stimulus (i.e. the flashing lantern) in a context where its behavioural properties have changed.

It was intended to modify the radius size within which the presence of an enemy agent would trigger the lantern flashing system. However, playtesting revealed that players did not notice this change at all. That is, not only did the change of radius size not appear to have a disruptive effect on the players, it did not appear to have any notable effect, with playtesters reporting no apparent change to the behaviour of the lantern.

A second prototype was thus developed that made the flashing lantern respond not to enemy proximity within a radius but instead, to the presence of an enemy within a forward-facing ‘cone of vision’ from the player-character’s location. This required additional development time to be spent by the engine programmer. However, the additional time investment (approximately two days) was not considered significantly problematic in the context of the overall development process.

With this second prototype of the flashing lantern warning system in place, if the player’s field of vision (regardless of proximity or direct line-of-sight) contains an enemy agent, then the lantern will begin to flash. If the player’s field of vision moves away from the enemy agent, the lantern will stop flashing and return to a solid beam of light.

This functionality provided a means for players to experience Recall Disruption (i.e. a stimulus that appears identical to a previously experienced stimulus but behaves in a different manner). In terms of gameplay this rule change also allowed different potential player experiences. With the proximity trigger removed from the game, enemies were now more able to approach players
from behind with less warning. However, players that identified the rule change and were then also able to construct accurate new understanding of the new semantic properties of the system could gain a useful navigational tool. In the game’s larger, darker areas that occur during the middle and later stages of the game (especially in those that contain multiple enemy agents), being able to identify the direction of an enemy, rather than just an approximation of its proximity, allowed players to move more stealthily and purposefully through the game environments.

These modified rules remained constant again for a period of gameplay (from the end of the game’s sixth level, through to the game’s thirteenth level), allowing the player time to potentially reconstruct new intraludic knowledge and understanding of the warning system.

However, the rules of the system were again modified during the final stages of the game where the player encounters the Tesla Pig enemies. This final modification to the flashing lantern system coincided with the addition of the multiple enemy types to the game (Section 7.5.6). This thus demonstrates again the need to apply the principles of the disruptive game design philosophy throughout the design and development processes so that opportunities for including additional disruptive game features can be acted upon as they emerge as part of the evolution of a project.

7.5.12.3: ‘Lantern-Based Warning System’, Prototype 3
The behaviour of the Tesla Pig was already designed to provide a basis for Action Plan Disruption (Section 7.5.6.5, Section 7.5.6.6, and Section 7.5.6.7). As the Tesla Pig prototypes were developed, the suggestion was made to further enhance the disruptive potential of the Tesla Pig encounters by also disrupting the player’s knowledge of the flashing lantern warning system again at this late stage in the game (i.e. the final two levels).

Specifically, the suggestion was made that the presence of the Tesla Pig enemy could cause the lantern to continuously flash, with randomised intensities and at random rates, for the entire time it is present in the environment. Thus, players would be presented with a scenario in which previously established intraludic knowledge (i.e. that the flashing is based on proximity, or that the flashing is based on the direction the player is looking) becomes inaccurate and one of the key methods of avoiding the enemy threats throughout the game thus far (i.e. the lantern-flashing system) would become unreliable.

This disruption of intraludic knowledge would then be compounded by the new enemy type being significantly faster than previous types and able to teleport around the environment (Section 7.5.6.5, Section 7.5.6.6, and Section 7.5.6.7). Players would therefore need to revert back to existing transludic and, potentially, extraludic knowledge, to decide upon an appropriate course of action. For example, players may try to predict the enemy’s movements and use the layout of the environment to avoid it, or try to outpace the enemy by sprinting through the section and trying to find the exit as quickly as possible.
Thus, this final rule modification would provide not only Recall Disruption (i.e. presenting a familiar stimulus to players that behaves in an unfamiliar way) but also Action Plan Disruption of the intraludic semantic knowledge attached to the lantern-flashing system. To this point in the game, players will have been able to respond to the flashing lantern successfully by either using the flashing as a warning system or, when the flashing starts, putting the lantern away so as to avoid attracting the attention of the nearby enemy agent(s). In this final section of the game however, neither of these established actions will be particularly successful for players. Putting the lantern away will be of no benefit, as the Tesla Pig enemy will not be attracted to it (i.e. it will attack the player regardless of the lantern being active or inactive). Similarly, attempting to use the flashing lantern as a warning system, as has been established throughout the game, will be of limited use, as the flashing system will no longer be linked to any useful trigger. The consequences of performing either of these actions will thus be disrupted (i.e. performing them no longer provides an increased likelihood of safely avoiding the enemy agents), providing Action Plan Disruption.

Implementing this design suggestion required a third functionality prototype to be created although again, the development time required for this process was minimal. The engine programmer added further script support to enable the engine-level lantern flashing code to be overridden, effectively enabling the event scripter to ‘start’ and ‘stop’ the lantern flashing directly through script commands attached to triggers in the game environments.

With this functionality in place, the flashing lantern warning system had enough flexibility and functionality to operate based on enemy proximity (developed in the first prototype, Section 7.5.12.1), to operate based on the player-character’s field of vision (developed in the second prototype, Section 7.5.12.2), and to operate based on explicit script commands (developed in this third prototype). Thus, this disruptive game feature had the potential to provide both Recall Disruption and Action Plan Disruption as per the original design intent (Section 7.5.12), albeit through different eventual methods than intended (i.e. simply making changes to the size of the activation radius around the player-character).

7.5.12.4: ‘Flashing Lantern Warning System’, Full Feature Development

With the functionality of the system in place as a cumulative result of the three different prototypes, full feature development required the full development, playtesting and iteration (as required) of the specific enemy encounter scenarios in which the player would experience the flashing lantern warning system. This process was ongoing throughout the remainder of the game’s development cycle up to the point the game was handed over to FG for publication approval. While no significant changes were made to the flashing lantern warning system itself, enemy encounters were modified in some of the game’s levels (e.g. having their locations moved, or having their trigger position altered). The eventual game as created however still maintains the three key ‘events’ described for the flashing lantern warning system; the opportunity for initial intraludic knowledge encoding to occur, followed by the change in the system’s functionality in
the game’s sixth level, followed by the final change in its functionality in the game’s penultimate two levels.

7.5.13: Enemy Audio Cues
The use of enemy audio cues went through a number of different prototypes and was also subject to some disagreement between TCR and FG.

7.5.13.1: Enemy Audio Cues, Prototype 1
Initially, the development team used a number of test scenarios to assess the impact of completely removing all enemy-based audio cues, other than a small set of basic grunts and snorts that were associated with their normal movement animations. In removing the non-diegetic sound (i.e. the audio track that was present in A:TDD) it was intended that players would be required to learn to detect enemies by listening more closely to the diegetic sounds being emitted by them, rather than relying on their transludic knowledge of audio cues (i.e. if the audio track is playing, hide; once it stops, it is safe to continue through the environment).

Such a radical alteration to an established mechanic of the Amnesia franchise was not approved by FG however, after a version was provided for them to playtest. The primary issue they cited was that enemy encounters now lacked a sense of building tension, climax and denouement, functioning instead as ‘jump’ or ‘shock’ scares when enemies were encountered without prior warning. This feedback is in line with the suggestion of Perron (2004) that forewarning does not lead necessarily to preparedness, rather, it heightens arousal and tension prior to an event. This initial prototype was thus evaluated as in need of further refinement, due to it failing to meet the requirements of the context of the development process (i.e. failing to meet publisher approval).

7.5.13.2: Enemy Audio Cues, Prototype 2
A second iteration of this disruptive game feature was developed, utilising a more expansive set of subtly different musical sound cues that seamlessly blended into and out of one another during gameplay. Each type of enemy agent in the game had three cues associated with it, which linked to the three possible behaviour states that an enemy could be in: patrolling, hunting, or attacking. This was a significant expansion on the system utilised in A:TDD and indeed, the system used in a number of other games whereby the same audio track (or small selection of tracks) are used regardless of the enemy in question.

This prototype could not have been initially considered during the creation of the game as designed, because the different enemy types were not added to the game until later in the development process (Section 7.5.6). Thus, with their addition, the audio cues could also be further refined. Once more, this demonstrates the significant differences between the game as designed and the game as created, along with the importance of maintaining the use of the disruptive game design philosophy throughout the design and development processes.
The reasoning behind this second prototype iteration was that the presence of sound cues fulfilled FG’s requirements as well as providing a means for increasing dramatic tension and player anxiety. However, this iteration maintained potential for player disruption through the inclusion of a sound cue associated with the enemy’s ‘patrolling’ behaviour state. With this audio cue included, enemies would maintain an active audible presence in the environment even when it was ‘safe’ for players to move around (i.e. the enemy was not actively hunting for them or attacking them).

This ‘patrol’ audio cue shared certain musical similarities with the ‘hunting’ and ‘attacking’ cues, such as rhythm, tempo, and particular musical phrases within the audio track itself. While the audio cues in this prototype were still available to players, they would be required to engage with the content of the audio cue to determine the enemy state and thus their relative safety within the environment, rather than being able to make the simple binary assessment of whether there was an audio track playing or not. To do this, construction of new intraludic knowledge would be required to form an understanding of the link between different audio cues and enemy behaviour states. The formation of this knowledge would be based on ambiguous stimulus information (i.e. there is no explicit explanation in the game of the meaning of different audio tracks), providing a source of Encoding Disruption. Further to this Encoding Disruption, the use of different sets of audio for different enemy types introduced over the course of the game would require ongoing construction of new intraludic knowledge throughout the game. With three enemy types, each with three different audio cues, this created nine different audio cues for players to identify and understand over the course of the six hours of play time. Players would not be able to rely solely on their understanding of the musical content of the first set of audio cues they encounter in the game to understand later encounters with different enemies.

However, further testing of this prototype in scenarios involving more complex enemy behaviour and in particular, in scenarios involving multiple enemy agents, identified a number of limitations of the HPL2 engine’s sound handling capabilities. Multiple enemies in particular were problematic as they may be in two different states at the same time (e.g. one may be patrolling, while the other is actively hunting the player). The HPL2 engine did not have a method built in for handling two different enemy audio cues simultaneously. Even if this capability was available, having two enemies in different states simultaneously presented the design problem of how to communicate that clearly to the player.

It was thus determined that to get this version of the audio cue system working to a releasable quality would require significant additional programming time to rewrite the relevant sections of the game engine’s audio code, in addition to having to solve a further design problem regarding the communication of information to the player in an effective manner. Given the time, budget and single engine programmer on the development team this was evaluated to not be viable within the context of the development process.
7.5.13.3: Enemy Audio Cues, Prototype 3

With the difficulties encountered making changes to the functionality of the enemy audio cue system itself in the first and second prototypes, consideration was instead given to how the existing audio cue system could be repurposed without requiring significant code-level changes. The third prototype iteration thus reverted back to a single audio track that played during an encounter with any enemy agent, as per the system that exists in A:TDD and many other games. However, the manner in which this audio cue system was deployed throughout the game was modified to maintain potential for player disruption, albeit through different means than previously intended in the game as designed. For example, the audio cue triggers may not be attached to some specific enemy agents, allowing for ambushes or surprise attacks to be orchestrated by the game designers. Conversely, the audio cues could be manually triggered through the game’s AngelScript, allowing for the presence of enemy agents to be faked through sound alone.

This modification to the system’s deployment in the game required, primarily, development time to be dedicated by the event scripter and designers, rather than the engine programmer. This use of the existing audio cue system also received approval from FG when it was demonstrated in test scenarios. Thus, this prototype was approved for full development.

7.5.13.4: Enemy Audio Cues, Full Feature Development

With the third prototype approved for full development, the design task was then to make use of the system in interesting ways at different points throughout the game. With disruptive game design in mind, the audio cue system presented an opportunity to make use of a similar ‘model’ of disruption that was applied to the flashing lantern warning system (Section 7.5.12). That is, to allow players time to construct seemingly accurate intraludic knowledge during the game’s early stages, which can then be disrupted (via Recall Disruption and Action Plan Disruption) by making changes to the system’s behaviour at subsequent stages in the game.

Thus, early enemy sightings in which players are not actually under any immediate threat (although players do not know this) do not trigger the audio cues associated with ‘real’ enemy encounters. This allows for the creation of ‘false’ intraludic knowledge, which suggests that there is no audio cue attached to enemies. Later in the game, encounters that do make use of the audio cues provide a short-term disruption of the initially constructed intraludic knowledge, although this initial disruption can likely be readily interpreted by players utilising transludic knowledge that suggests that the use of audio cues in conjunction with combat is a commonly occurring feature in a range of games.

However, over the course of the game the audio cue system is further deployed in different ways. For example, Recall Disruption is implemented by triggering the audio cue associated with enemies in an otherwise safe environment, thus causing an established stimulus (i.e. the audio cue) to exhibit behaviour different to the previously established behaviour (i.e. signalling the presence of an attacking enemy). The audio cues are also removed from some instances of ‘real’ enemy encounters, further disrupting the players’ intraludic knowledge.
encounters, or their trigger is delayed so that players may unexpectedly find themselves facing an enemy they were not expecting. This provides short-term Action Plan Disruption as the stimuli (i.e. lack of audio cue) is responded to seemingly appropriately (i.e. by not hiding from an enemy) but this response is then found to not be appropriate in this particular instance.

Manipulation of the audio cues in *A:AMFP* provides a disruptive game element that has a wide range of disruptive potential. Those with prior experience with games, those with prior experience of specifically *A:TDD*, and those with minimal prior game experience, will each potentially experience varying degrees of disruptive effect based on their existing transludic knowledge as well as the intraludic knowledge constructed during gameplay of *A:AMFP*. Thus, this disruptive game feature has the potential to afford a particularly wide range of player experiences.

**7.5.14: Manipulation of ‘Set Dressing’ Entities and Props and the Pig Mask Motif**

The originally proposed designs (i.e. in the *game as designed*) for each of the ‘set dressing’ manipulation and ‘pig mask’ motif game features (Section 7.4.11 and Section 7.4.12) remained effectively unchanged in the *game as created*. The prototyping process was only required to confirm the implementation method for both features.

**7.5.14.1: Manipulation of ‘Set Dressing’ Entities and Props and the Pig Mask Motif, Prototype 1**

At a functional level, both features were simple to implement via the game script. Thus, the only prototyping required for both features was to ensure that the ‘visible’ flags for the in-game entities could be activated and deactivated as required. The only additional process found to be required for this to be achieved was to ensure that any objects being manipulated through script were saved as ‘dynamic’ entities, rather than the ‘static’ entities that would usually be used for entities such as wall sections.

**7.5.14.2: Manipulation of ‘Set Dressing’ Entities and Props, Full Feature Development**

With the implementation requirements confirmed through prototyping, both features could be implemented throughout the game’s levels. Functionality of each instance of the features was identical, thus a full account of each in this report is unnecessary. Representative examples are instead provided.
One example of semi-randomised ‘set dressing’ manipulation occurs if the player enters the side room off of a corridor in the game’s first level, Mansion01. There is a 75% chance that a door (Figure 7.68, top) will be replaced by a solid wall when they come back out of that room (Figure 7.68, bottom).

With consideration given to the availability of development team resources, this disruptive game feature was a particularly useful one due to its ease of implementation and ability to be implemented solely by the level event scripter with only a few lines of script. The conversion of required entities from ‘static’ to ‘dynamic’ was also a simple process that could be completed by the event scripter.

There were potential risks associated with the manipulation of ‘set dressing’ entities and props however. While the intention was to provide a subtle environmental manipulation that may or may not be noticed by players, it was also possible that seemingly ‘random’ environmental alterations, without any explicit reason, may be viewed as being a game ‘bug’ rather than an intended design feature. However, the ‘horror’ setting provided more potential for players to interpret the environmental alterations as logical within the game setting, perhaps interpreting...
them as being signifiers of an unstable mental state in the player-character, or of paranormal activity by unseen enemy agents.

These same considerations applied to the use of the ‘pig mask’ motif (which was also able to appear and disappear during gameplay in different areas of the game levels).

7.5.14.3: Pig Mask Motif, Full Feature Development
As described in the game as designed (Section 7.4.12), the use of the Pig Mask Motif was intended to provide a means for players to experience Encoding Disruption, via ambiguous stimulus information. The design intent was for the Pig Mask object to appear at different locations throughout the game, some of which would be semi-randomised and thus differ between different play sessions and differ between individual players’ experiences.

With no explicit explanation for the meaning of the Pig Masks, players would be left to interpret their meaning themselves. Furthermore, in many instances, a Mask object may disappear and reappear within a single environment multiple times while the player explores.

Figure 7.69: The pig mask motif in two locations in A:AMFP’s first level, Mansion01; in a storage area (top) and attached to the face of a bust (bottom) (A:AMFP September 2013 Release Build).
The Mask object appears frequently in the game’s first three levels and in particular in the game’s first level, Mansion01 (Figure 7.69). This is to increase the likelihood of players noticing the object in the first place and thus, the likelihood that players will then continue to notice the object throughout the rest of the game when the frequency of its appearances decreases.

It is important to note that, while TCR had expectations of likely possible interpretations of the meaning of the Pig Mask Motif, there is no ‘intended reading’ of the Masks’ meaning. For example, it was expected that one common interpretation from players may be that the Masks are being left by Mandus’ children as a means of guiding him to them. However, the positioning of the Masks throughout the game’s levels was not designed to specifically support any one particular interpretation but instead, to allow multiple interpretations to remain potentially ‘correct’ for players. This demonstrates the benefit of being aware of design intent (Section 2.1), which can only be reliably achieved through a research methodology (such as RtD) that allows active researcher involvement in the design and development processes.

7.5.15: Summary of the Prototyping Process and the Game as Created for Amnesia: A Machine for Pigs

The game as created for AAMFP consists of the disruptive game features that were approved for full development following the prototyping process. The game as created also includes non-disruptive game features that have been similarly prototyped and approved (e.g. basic game functionality, such as player-character movement) however, these are not the focus of the current research.

Figure 7.70: All four stages of the Disruptive Game Feature Design and Development (DisDev) model applied to Amnesia: A Machine for Pigs (Part 1).
Figure 7.70 and Figure 7.71 present all four stages of the DisDev model applied to Amnesia: A Machine for Pigs (Part 2). Following the assessment of the project context in Stage 1 and the design of game features in Stage 2, Stage 3 focuses on prototyping those features. There are three possible outcomes of Stage 3 for each feature prototype. Firstly, the prototype may be iterated before being playtested and evaluated again (i.e. the feature is evaluated as being viable for full development after design...
changes are made). Secondly, the prototype may be approved for full development, or thirdly, may be removed from the game entirely. Additionally, the development of a feature prototype or feedback on a particular feature during playtesting may present an opportunity to design and prototype new game features that were not previously considered during the creation of the game as designed (i.e. features added as a result of prototype feedback).

As demonstrated in Figure 7.70, the disruptive game features in A:AMFP provided multiple examples of each of these different outcomes from Stage 3 of the DisDev model. The number of features removed entirely from the game is comparatively low in relation to the features outlined in the game as designed. This is reduced further when considering that some of these features were replaced by other features as the development of the game continued (e.g. the procedurally generated maze was replaced with the ‘non-Euclidean’ space, while the personality-driven enemy A.I was replaced with the addition of multiple different enemy types with bespoke behaviour scripting). This thus demonstrates that the initial assessment of project context that underpinned the proposed disruptive game features in the game as designed was appropriate.

Stage 4 of the DisDev model contains the game features approved for full development in Stage 3. These were allocated to the appropriate development team members before going through the development pipeline as required for each feature. Even following the assessment of project context and the development and approval of feature prototypes, some features nevertheless encountered unforeseen issues once they had entered into full development. These issues were primarily minor, such as changes being required to individual enemy encounter scenarios. However, it is important to identify these changes as they serve to demonstrate that any model of game design and development, such as the DisDev model, must be flexible enough to respond to such unforeseen issues.

The output of Stage 4 of the DisDev model is the ‘release candidate’ of the game as created; the build of the game that TCR felt was ready to be sold to players, pending any required publisher-based tasks, such as language translations, localisation and platform testing for Windows, MacOS, and Linux.
7.6: Significant Differences between the Game as Created and the Game as Published

AMFP was released on September 10th 2013. However, the final build of the game as created (the ‘release candidate’) was passed from the TCR development team to the FG team in April 2013.

As described in the game space model (Figure 7.72), in many project contexts the game as published will have additional features when compared to the game as created, such as additional language translations, or specific localisation features for different release territories. However, more significant alterations to the structure of the game as created would not usually be made by the publisher. If any significant changes were still required at this stage of the process, the
publisher would likely refer them back to the development team, with any incurred costs being paid for by the developer or publisher as appropriate (Chandler and Chandler, 2011, p.26-27).

However, the dynamic that existed between TCR and FG was complex. FG as the publisher did not own TCR as the developer, as may be the case in many project contexts (e.g. large companies such as Nintendo that own a number of other development studios, such as Retro Studios and Intelligent Systems). This thus made TCR an independent developer working on a single-project-contract for FG. However, because FG were themselves also an active developer and owned the HPL2 game engine technology that TCR were using to develop A:AMFP, their degree of engagement in the development process was greater than may be expected in the normal role of a publisher. While this degree of engagement was beneficial during the majority of the project’s life cycle for development support and technical issues with the engine itself, it caused problems when the game as created was handed over to FG.

While FG did not make any changes to the disruptive game features that had been implemented during the creation of the game as created, they were responsible for making some other alterations to the game. These principally took the form of changes to the game’s aesthetic properties and, in particular, the game’s colour balancing that TCR had invested significant time into. Specifically, TCR had invested one month of combined development time from the engine programmer and the lead artist developing a bespoke ‘colour grading’ system. This system allowed different game environments and importantly, sections of the same environment, to have different global lighting colour properties. The system was created to cater for A:AMFP’s multiple game levels that transition from exterior to interior environments multiple times requiring, for example, transitions from yellow-tinted interior lighting to blue-tinted, moonlit exterior scenes.
FG made changes to the colour grading system during the time that they were working with the game and these changes were not initially communicated back to TCR. The changes had the effect of creating a ‘blue tint’, or ‘fog’ across the game screen throughout much of the game (Figure 7.73). While this did not make the game unplayable, it did notably detract from the graphical quality of the released product.

It is important to note that changes such as this are unlikely to be made by the publisher in the majority of game development projects. As shown in the game space model (Figure 7.72), the publisher would be expected to make additions to the game but not to make changes to the structure, content, or implementation of the game as created. In this particular instance, TCR submitted fixed game code back to FG when it became apparent close to the game’s release date that the ‘blue tint’ had been created. However, this fixed code was still not incorporated into the
release build by FG. The reasoning for this is not clear, although it may be due to a miscommunication between the two companies or simply an error in incorporating the wrong version of the game code into the release build.

Within the context of this research, the difference between the \textit{game as created} and the \textit{game as published}, in terms of disruptive game features, is negligible. However, the presence of such an overt graphical issue may have a detrimental effect on the content of any player discussion and debate, drawing player attention away from disruptive game features (or indeed, other intentionally designed and developed game features) towards an unrelated aesthetic game ‘bug’. Thus, in the context of the player-data gathering process for the current research (Chapter 8), errors made in the creation of the \textit{game as published} may be problematic.

7.7: Observations from the Application of Design Theory to Design and Development Practice

From the design and development process, some observations can be made about the ‘reality’ of applying the disruptive game design \textit{theory} to disruptive game design and development \textit{practice}.

7.7.1: The DisDev Model’s Applicability to Commercial Game Projects

The disruptive game design philosophy has been demonstrated to be applied, through the application of the DisDev model, to a commercial game design and development project (\textit{A:AMFP}). Through application of the model in the assessment of the project’s context (Section 7.2), the creation of disruptive game features in the \textit{game as designed} (Section 7.3 and Section 7.4) and the prototyping and development of those features in the \textit{game as created} (Section 7.5), the viability of utilising the model and the underpinning philosophy in a commercial context is supported. However, it is important to note that further work, through application of the philosophy and model to different commercial projects with different contexts, aims, and developer/publisher relationships, would be necessary to assess their viability for use more generally. As demonstrated in the differences between the \textit{game as created} and the \textit{game as published} (Section 7.6), along with the disagreements between TCR and FG regarding some disruptive game features, projects involving separate development and publishing studios may encounter additional challenges in the application of the philosophy and the model. This may be especially apparent if disruptive game features seek to make changes to what a publisher may consider a ‘core’ or ‘established’ feature of games as a medium, particular game genres, or a particular game series. These challenges may be less problematic for an independent development studio that is self-publishing, as creative decisions may not need to pass through an external company’s approval process (unless the game is to be released through a mediated distribution channel, such as Steam).

The application of the disruptive game design philosophy and the DisDev model to \textit{A:AMFP} thus provides an initial, in-depth case study that may be used as a basis for further practice-based research in a variety of other contexts and projects.
7.7.2: Designing for Different Modes of Disruption and Disruption of Different Knowledge Types

The three types of disruption (i.e. Encoding Disruption, Recall Disruption and Action Plan Disruption) previously identified in Section 4.3 have been demonstrated to apply to game mechanics (e.g. the Flashing Lantern Warning System, Section 7.5.12), audiovisual elements of game aesthetics (e.g. Enemy Audio Cues, Section 7.4.13 and Section 7.5.13), as well as in-game cues for constructed narrative and delivery of the game’s told story (e.g. the Pig Mask Motif, Section 7.4.12 and Section 7.5.14). The modes of disruption are applied either individually or more often in combination, as the basis for the design of each disruptive game feature (Section 7.4).

The distribution of disruptive game features in A:AMFP is not balanced evenly between those utilising Encoding, Recall, and Action Plan Disruption, nor between those utilising disruption of procedural, semantic, and episodic knowledge. This reflects the importance of individual project context on the applicability of different parts of the disruptive game design philosophy. For example, in A:AMFP there were few opportunities within the constraints of the project context to disrupt a player’s procedural knowledge (i.e. knowledge of actions and processes). Indeed, two of the disruptive game features in the game as designed that sought to do so were eventually removed from the game during prototyping (i.e. the ‘infection’ system and the procedurally generated ‘maze’). As described in relation to these two features (Section 7.5.1 and Section 7.5.3), their removal from the game was not due to them failing as disruptive game features but instead, failing to meet contextual requirements of the project itself. Thus, while A:AMFP may not have a high number of examples of disruption of procedural knowledge, this is not to suggest that such disruption would not be appropriate in other project contexts.

Furthermore, there is only one occurrence in A:AMFP (in the manipulation of ‘set dressing’ entities and props, Section 7.4.11 and Section 7.5.14) in which Recall Disruption is utilised as the only designed mode of disruption for a disruptive game feature. In all other instances, Recall Disruption is used in conjunction with, primarily, Action Plan Disruption and occasionally, Encoding Disruption. This pairing of Recall Disruption and Action Plan Disruption to form disruptive game features is something that could have only been discovered through practice.

In the original definitions of the modes of disruption (Section 4.3), Recall Disruption is ‘observation’-based (i.e. it is based on a familiar stimulus displaying unfamiliar properties or behaviours to the player). Conversely, Action Plan Disruption is ‘interaction’-based (i.e. it is based on a familiar stimulus responding in an unexpected way to a previously established player interaction with it). Each mode of disruption appears to serve very different purposes. Indeed, in A:AMFP the uses of Recall Disruption without Action Plan Disruption are limited to the manipulation of ‘set dressing’ entities and props and also, the manipulation of the Pig Mask Motif throughout the game (which itself utilises the same functionality as more general ‘set dressing’). Both of these disruptive game features are designed to be disruptive without requiring player
interaction; the appearance and disappearance of game entities during gameplay, if observed by the player, is designed to trigger Recall Disruption.

However, when Recall Disruption is designed to be triggered by disruptive game features that involve player interaction, in many cases the Recall Disruption naturally leads to an opportunity for Action Plan Disruption. For example, in the case of the Flashing Lantern Warning System, changing the trigger ‘rules’ for the flashing mechanism at different points during the game provides Recall Disruption (i.e. a familiar stimulus behaving differently). The rule change also means that player actions that were previously successful may no longer be successful (i.e. a familiar stimulus responds in an unexpected way to player interaction). Because a property or ‘rule’ of a game entity or game system must change to cause the different behaviour required for Recall Disruption, it becomes more likely that player interactions based on the previous properties or rules will become incorrect or unsuccessful. This in turn leads to Action Plan Disruption in addition.

7.7.3: ‘Interludic’ Knowledge as an Additional Ludic Knowledge Type

As demonstrated in relation to a number of different disruptive game features in A:AMFP, many instances of transludic knowledge disruption were in fact more specifically targeting a player’s knowledge from one particular game; the previous game in the Amnesia series, A:TDD.

This raises the question of whether the concept of transludic knowledge (i.e. ‘across-games’ knowledge) is too broad to incorporate the specifics of different forms of ‘across-games’ knowledge. This was raised as a potential issue in the definition of transludic knowledge (Section 3.3.3). However, there was no evidence available at that stage of the current research to substantiate an argument for different types of transludic knowledge.

With the practice-based evidence gained from engaging in the design and development process for A:AMFP, the suggestion can be made that in the context of established game series or franchises in particular, ‘interludic’ knowledge (i.e. ‘between-games’ knowledge) may be a more appropriate description for ‘game-specific transludic knowledge’. Thus, disruption such as designed into A:AMFP that targets a player’s knowledge of one particular game is not defined as belonging to the same broad category as disruption designed to target a player’s general knowledge of ‘games’ as a medium.

To further support a definition of ‘interludic’ knowledge as notably different from ‘transludic’ knowledge however, it is necessary to also consider how players respond to different types of transludic knowledge disruption within A:AMFP. Thus, the concept of interludic knowledge is returned to following analysis of player feedback on the game (Section 8.10.4.2).
7.8: Summary of Disruptive Game Features in *Amnesia: A Machine for Pigs*

This section summarises each of the disruptive game features that are present in *AAMFP’s game as designed* and *game as created*. Each feature is listed along with the type(s) of disruption the feature has the potential for producing and the target knowledge type(s).

### 7.8.1: Summary of Disruptive Game Features in the Game as Designed

This section summarises the disruptive game features in the *game as designed*. Features that are highlighted in orange were removed from the game during the later prototyping process, while the remaining features were retained, in some form, in the release version of the game.

<table>
<thead>
<tr>
<th>Disruptive Game Feature</th>
<th>Potential Mode of Disruption</th>
<th>Potential for Disruption of Knowledge Type...</th>
</tr>
</thead>
<tbody>
<tr>
<td>The ‘Infection System’</td>
<td>Encoding Disruption</td>
<td>INT-P, INT-S, INT-E</td>
</tr>
<tr>
<td>(Section 7.4.1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Emulating Peripheral Vision</td>
<td>Recall Disruption</td>
<td>TRANS-P, TRANS-S</td>
</tr>
<tr>
<td>(Section 7.4.2)</td>
<td>Action Plan Disruption</td>
<td>TRANS-S</td>
</tr>
<tr>
<td>Procedurally Generated Tesla Field ‘Maze’</td>
<td>Recall Disruption</td>
<td>TRANS-P, TRANS-S, TRANS-E</td>
</tr>
<tr>
<td>(Section 7.4.3)</td>
<td>Action Plan Disruption</td>
<td>TRANS-S</td>
</tr>
<tr>
<td>Personality-Driven Enemy Artificial Intelligence</td>
<td>Recall Disruption</td>
<td>INT-P, INT-S, INT-E, TRANS-P, TRANS-S, TRANS-E</td>
</tr>
<tr>
<td>(Section 7.4.4)</td>
<td>Action Plan Disruption</td>
<td>INT-S, TRANS-S</td>
</tr>
<tr>
<td>‘Hardcore’ Difficulty Mode</td>
<td>Recall Disruption</td>
<td>INT-S, INT-E, TRANS-S, TRANS-E</td>
</tr>
<tr>
<td>(Section 7.4.5)</td>
<td>Action Plan Disruption</td>
<td>INT-S, TRANS-S</td>
</tr>
<tr>
<td>Providing the Player with Hints</td>
<td>Recall Disruption</td>
<td>TRANS-P, TRANS-S, TRANS-E</td>
</tr>
<tr>
<td>(Section 7.4.6)</td>
<td>Action Plan Disruption</td>
<td>TRANS-S</td>
</tr>
</tbody>
</table>

*Continued on next page*
<table>
<thead>
<tr>
<th>Disruptive Game Feature</th>
<th>Potential Mode of Disruption...</th>
<th>Potential for Disruption of Knowledge Type...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Handling the ‘Death’ of the Player Character (Section 7.4.7)</td>
<td>Recall Disruption</td>
<td>INT-S, INT-E, TRANS-S, TRANS-E</td>
</tr>
<tr>
<td></td>
<td>Action Plan Disruption</td>
<td>INT-S, TRANS-S</td>
</tr>
<tr>
<td>Hiding Places (Section 7.4.8)</td>
<td>Encoding Disruption</td>
<td>INT-S, INT-E</td>
</tr>
<tr>
<td></td>
<td>Recall Disruption</td>
<td>INT-S, INT-E, TRANS-S, TRANS-E</td>
</tr>
<tr>
<td></td>
<td>Action Plan Disruption</td>
<td>INT-S, TRANS-S</td>
</tr>
<tr>
<td>Item Inventory Removal (Section 7.4.9)</td>
<td>Recall Disruption</td>
<td>TRANS-S, TRANS-E</td>
</tr>
<tr>
<td></td>
<td>Action Plan Disruption</td>
<td>TRANS-S</td>
</tr>
<tr>
<td>The Player-Character’s Lantern (Section 7.4.10)</td>
<td>Recall Disruption</td>
<td>TRANS-P, TRANS-S, TRANS-E</td>
</tr>
<tr>
<td></td>
<td>Action Plan Disruption</td>
<td>TRANS-S</td>
</tr>
<tr>
<td>Manipulation of ‘Set Dressing’ Entities and Props (Section 7.4.11)</td>
<td>Recall Disruption</td>
<td>INT-S, INT-E, TRANS-S, TRANS-E, EX-S, EX-E</td>
</tr>
<tr>
<td>The Pig Mask Motif (Section 7.4.12)</td>
<td>Encoding Disruption</td>
<td>INT-S</td>
</tr>
<tr>
<td></td>
<td>Recall Disruption</td>
<td>INT-S</td>
</tr>
<tr>
<td>Enemy Audio Cues (Section 7.4.13)</td>
<td>Recall Disruption</td>
<td>INT-S, INT-E, TRANS-S, TRANS-E</td>
</tr>
<tr>
<td></td>
<td>Action Plan Disruption</td>
<td>INT-S, TRANS-S</td>
</tr>
</tbody>
</table>

Table 7.4: Summary of all disruptive game features present in A:AMFP’s game as designed.
### 7.8.2: Summary of Disruptive Game Features in the Game as Created

This section summarises the disruptive game features in the game as created. Features that are highlighted in green were added to the game during prototyping (i.e. once the initial ‘design’ stage had been completed) and were thus not present in the game as designed. Some features that were originally in the game as designed changed their disruptive potential and/or their target knowledge types during prototyping. These changes are reflected in this summary.

<table>
<thead>
<tr>
<th>Disruptive Game Feature</th>
<th>Potential for Disruption of Mode...</th>
<th>Potential for Disruption of Knowledge Type...</th>
</tr>
</thead>
<tbody>
<tr>
<td>‘Non-Euclidean’ Space</td>
<td>Encoding Disruption</td>
<td>INT-S, INT-E</td>
</tr>
<tr>
<td>(Section 7.5.4)</td>
<td>Recall Disruption</td>
<td>INT-S, INT-E, TRANS-S, TRANS-E, EX-S, EX-E</td>
</tr>
<tr>
<td>Multiple Enemy Types</td>
<td>Recall Disruption</td>
<td>INT-S, INT-E, TRANS-S, TRANS-E</td>
</tr>
<tr>
<td>(Section 7.5.6)</td>
<td>Action Plan Disruption</td>
<td>INT-S, TRANS-S</td>
</tr>
<tr>
<td>Providing the Player with Hints</td>
<td>Potential for Cognitive Engagement but not stemming from Disruption, following design disagreement between TCR and FG</td>
<td>N/A</td>
</tr>
<tr>
<td>(Section 7.5.8)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Handling the ‘Death’ of the Player Character</td>
<td>Recall Disruption</td>
<td>TRANS-S, TRANS-E</td>
</tr>
<tr>
<td>(Section 7.5.9)</td>
<td>Action Plan Disruption</td>
<td>TRANS-S</td>
</tr>
<tr>
<td>Hiding Places</td>
<td>Encoding Disruption</td>
<td>INT-S, INT-E</td>
</tr>
<tr>
<td>(Section 7.5.10)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Item Inventory Removal</td>
<td>Recall Disruption</td>
<td>TRANS-S, TRANS-E</td>
</tr>
<tr>
<td>(Section 7.5.11)</td>
<td>Action Plan Disruption</td>
<td>TRANS-S</td>
</tr>
</tbody>
</table>

*Continued on next page*
Table 7.5: Summary of all disruptive game features present in A:AMFP’s game as created.

7.9: Chapter Summary

It has proven challenging to balance the aims of the research with the constraints placed upon the project by the technology, the aims of the development team, and the requirements of the publisher (i.e. the project’s contextualising factors). However, in doing so, the disruptive game design philosophy, design framework and the DisDev model have been tested within a live commercial project and demonstrated to be usable in such an environment. The commercial design and development project is concluded with the publication of a ‘post-mortem’ report (i.e. a retrospective analysis of the successes and failures during the game’s development) (Howell, 2014) (Appendix F) which provides industry-facing discussion of the project.

From the process of planning, designing and developing A:AMFP utilising the principles of the disruptive game design philosophy, it can be suggested that in the context of this project, the philosophy and framework, coupled with the DisDev model for the design, prototyping, development, and evaluation of individual disruptive game features has been successful in advancing from a design philosophy, through to having a completed commercial game object (i.e. the game as published).
This does not mean that the player responses to the disruptive game features, or to the game as a whole, will necessarily be positive however. It is important to separate the success of the implementation from the success of the game as received by players (i.e. the game as played). At this stage, it is possible to state that the disruptive game design philosophy, design framework and the DisDev model can be applied to the development of a publisher-constrained commercial game. It is also possible to state that the DisDev model, for providing guidance over the consideration of contextualising factors for a project and over the selection of game elements with which to construct disruptive game features, has been successful in this particular project context.

The next chapter will present the gathering and analysis of player responses to individual gameplay experiences of A:AMFP and evaluate the impact of the disruptive game design philosophy on the player experience of the game as played. This will thus enable the completion of the final stages of the RtD process model, the evaluation of documented player feedback, followed by conclusions being drawn.
Chapter 8:
Phenomenological Study of Online Player Discussion of
Amnesia: A Machine for Pigs

The disruptive game design philosophy has been proposed (Section 1.3) along with a design framework (Chapter 6) for instantiating it. This philosophy and framework was applied to the commercial game title A:AMFP during each of the stages of the assessment of project context (Section 7.2), the game as designed (Section 7.3 and Section 7.4), the game as created (Section 7.5), and the game as published (Section 7.6). However, to evaluate the effect of the disruptive game design philosophy on player experience, it is necessary to understand the game as expected and the game as played for the players of A:AMFP.

Within game space, a player does not respond directly to a game design philosophy. They respond to a representation of that game design philosophy, embodied within a particular game object (i.e. a game as published). However, even that representation is filtered through the player’s perception and interpretation processes. Thus, a player’s response to a game is a response to their individual interpretation of it. The player’s individual experience and how they report it can thus provide insight into their individual interpretation of the game as played and to an extent, their individual game as expected.

Due to the inherent individuality of player expectation and interpretation of experience, any analysis of such should aim to collate data from a large sample of players, to enable an understanding of the common themes reported by players. This will address the second research question (Section 1.4), which asks how players report their experiences of playing a disruptive game, in terms of it matching or not matching their game as expected, along with their experiences of higher-order thinking (i.e. analysis, evaluation, and creation of new knowledge) and satiation of high-level needs (Section 1.3.1).

Lastly, the analysis will also inform design theory and/or operational principles (i.e. design practice(s)), as outlined in the Research through Design (RtD) methodology (Section 2.4). Conclusions drawn can in turn inform future game design and development projects utilising the disruptive game design philosophy, as well as potentially suggesting avenues for further research (Section 9.3).

8.1: The Phenomenological Psychological Perspective and the
Game as Reported

When a player plays a game, their individual ‘lived’ experience of that game (i.e. the game as played) will be influenced by their individual context. For example, their prior experiences and knowledge (i.e. their transludic and extraludic knowledge), the environment in which they are playing, their current state of mind, and their individual ability to play the game or to understand
what the game is presenting (i.e. their intraludic knowledge). Players may also compare their lived
play experience to their expected play experience (i.e. the game as expected), either consciously
or subconsciously.

Collecting and analysing player data in a manner that allows focus to be placed on this lived
experience and the idiosyncratic processes that enable it requires a research approach with a
suitable ontological and epistemological perspective. Phenomenology is a broad label that
encompasses a number of different approaches to the study of ‘lived experience’. Phenomenological psychology, more specifically, provides a suitable perspective:

[. . .] to study experience and how the world appears to people. To this end,
phenomenological psychology employs a set of methods to enable researchers to
elicit rich descriptions of concrete experiences and/or narratives of experiences.
These methods are designed to illuminate the lived world of the participant and
also, possibly, the researcher, along with others who [have experienced, or may
in the future experience], something similar. (Langdridge, 2007, p.5)

To evaluate the disruptive game design philosophy, it is necessary to obtain these rich
descriptions of the individual lived experiences of players that have engaged with a representation
of that philosophy, embodied within a game object (e.g. A:AMFP).

The phenomenological perspective suggests that human awareness, or experience, is always
experience or awareness of something. To support this view of experience, Husserl (2002,
originally published 1931) utilises the terms noema (or the noematic correlate), meaning ‘what is
experienced’, and noesis (or the noetic correlate), meaning ‘how it is experienced’. Ihde (1986, p.44)
suggests that the relationship between noema and noesis can be visualised as a one way
perception process (Figure 8.1).

Thus, it is not possible to obtain a completely unfiltered description of the game as played ‘in-the-
moment’ by the player, as it is experienced within the mind of the player themselves. A pragmatic
solution is thus to gather data from multiple players’ reported game as played through methods
such as, for example, interviews or surveys. Ihde’s (1986) clearly defined one way process appears
suitable for describing an individual’s lived, ‘in-the-moment’, experiences in many instances.
However, this relationship is less appropriate if it is considered in relation to an individual
reflecting upon prior experience, or recalling experiential information from memory (i.e. a
player’s recalled memories of the game as played).
The difference between ‘in-the-moment’ experience (Figure 8.1) and reflection upon past experience (Figure 8.2) is that, following their gameplay session, the individual has to reflect upon that experience to report on it in some way (e.g. discussing their experience with other players online, or writing an article or review about the game). In doing this, the individual experiences their memory of their previous experience of the game object which will be recalled from episodic memory (not the experience itself). This recalled memory of the game as played can be defined as the game as reported (Figure 8.2).

8.2: The Contextual Constructivist Position in Phenomenology

Phenomenological approaches generally hold a contextual constructivist position (King, 2004, p.256). This position is defined by Madill et al. (2000, p.9) as one in which there is not assumed to be one reality that can be revealed or discovered through the use of an appropriate methodology (as is the position of a positivist or realist stance).

Contextualism is the position that all knowledge is local, provisional, and situation dependent. Hence, this perspective contends that results will vary according to the context in which the data was collected and analysed. (Madill, Jordan and Shirley, 2000, p.9)

The data gathered from multiple players’ games as reported may have multiple possible interpretations. However, in the context of the current research it is important that any presented interpretation of player data demonstrates flexibility, reflexivity, or what Stiles (1993, p.602) refers to as permeability. That is, that the interpretation can be demonstrated to be a response to the data available (i.e. the interpretation can be verified and the observation of the data demonstrated to have permeated the researcher’s understanding of the phenomena being studied). Furthermore, if other data were to be interrogated using the interpretation as a basis, that interpretation should be flexible enough so as to be modified to incorporate new findings or alternatively, to compliment a notably different interpretation. The aim of the contextual constructivist is to thus move towards a completeness of data (i.e. a collection of interpretations
that build towards a comprehensive understanding), rather than a singular convergence of data to a one, ‘true’, reality (Madill, Jordan and Shirley, 2000, p.10). In this study, multiple players will be providing their different reports on the game as played, moving towards a completeness of data rather than a singular convergence of data, relating to the effect of the disruptive game design philosophy on the players’ experiences. This thus makes the contextual constructivist position an appropriate position from which to analyse and discuss the gathered results.

8.3: Sourcing Data from Players

A number of potential data collection sources and methods were available for gathering data from players, such as focus groups, interviews with players, observed play sessions, or analysis of critical reviews and reportage of A:AMFP. However, in the context of the current research, such approaches were not the most appropriate. The aim of the research is to critique the application of the disruptive game design philosophy in a commercial game title, in a manner that can provide insight into the potential future commercial application of the design philosophy. High ecological validity (Section 2.1) in the data is therefore important. If data was to be gathered through experimental methods, or in contexts that remove players from a typical commercial game playing environment, the way that those players may report on their game experience (i.e. the game as reported) may be influenced.

Hence, even if a player had otherwise experienced an ‘ecologically valid’ game as expected, gathering data from that player through a non-ecologically-valid method would change the game as reported. Similarly, gathering data from critical reviews or media reportage of the game does not provide data that equates to the experience of the ‘generic’ player. It was therefore necessary to identify a data source that was naturally occurring and that players may choose to engage with as part of their natural game playing process. This data source would also need to afford data gathering in a practical and ethical manner.

With these requirements identified, the data source that was identified as being most appropriate was online discussion forums. Discussion forums can be engaged with by players out of choice and to any degree (i.e. they can post a single message or comment, or they can become involved in detailed conversations with different users across multiple forum ‘threads’). How players choose to write about their game experiences on these forums is similarly down to the individual player’s preferences. Thus, the individual games as reported that may be discussed will have a high degree of ecological validity as they will reflect how players discuss games and what elements of games those players focus on, in a natural context. This does not mean that data will be completely objective, as individual forum users may well have personal preferences, biases or agendas that will influence their comments. However, any such influences will also be ‘naturally occurring’ influences that would exist regardless of whether data was being gathered for research or not.

Online discussion forums are also practical data sources, as data gathering can be performed easily utilising either manual capture tools for saving webpages into PDF format (e.g. Mozilla
Firefox’s (Mozilla Foundation, 2015) ‘webpage to Adobe PDF’ browser plugin, or NVivo 10’s (QSR International, 2012) ‘NCapture’ browser plugin), or by utilising automated website ‘scraping’ software. Similarly, online discussion forums do not raise ethical issues (Section 8.5) regarding the collection and use of data, assuming that the data is anonymised and is taken from a forum that is publically open access.

There are a wide range of online discussion forums that are focused on games, either facilitating general discussion and/or discussion of specific games or specific game-related topics. However, the most appropriate forum to gather data from for this particular study was the Frictional Games (FG) official forum. The official forum for a game may be a first ‘point of contact’ for players wishing to discuss the game, either with each other or directly with the developer. This applies to a wide range of players, from those that have prior experience of previous games in the series, to those that may play A:AMFP without any such prior experience. While The Chinese Room (TCR) are the developers of A:AMFP and FG are the publishers, TCR do not have their own online forum meaning that players only have one official forum to use. Had TCR also had their own forum, it may have been necessary to gather data from both forums for completeness.

Other properties of the forum were also beneficial to the study. The forum structure and the default metadata attached to forum posts enables simple archiving and organisation of the gathered data as well as providing easy to query information when searching for specific data during later analysis. Specifically, each forum post is linked to a username, the date and time it was posted, the thread name it was posted in, and the specific post number within that thread. The forum is also organised into separate, game-focused sections, with each of FG’s previous games having its own forum section. This provides a practical benefit to this study, meaning that the researcher will not have to filter through forum posts unrelated to A:AMFP, as all A:AMFP posts will be stored in a single forum section.

With the official forum selected as the data source, it was also necessary to identify an appropriate time period sample which would form the dataset for analysis. With a confirmed release date for A:AMFP on September 10th 2013, the most appropriate time to take a sample was decided to be mid-January 2014. This allowed time for a number of different player types to potentially be represented in the data set, including early adopters (e.g. those that had pre-ordered the game or bought it upon release), players waiting for a sale before buying the game (e.g. the Halloween and Christmas sales), as well as players in demographics that may not have time to play games outside of holiday periods.

With the data source and sample period identified (Section 8.3), the phenomenological psychological perspective defined (Section 8.1), and the contextual constructivist position justified as a lens for analysing the data (Section 8.2), it is now necessary to determine the specific methodological approach to apply in the analysis of the data.
8.4: Selection of Appropriate Phenomenological Methodology

There are a number of different methodological approaches to phenomenological analysis, including Descriptive Phenomenology, Interpretive Phenomenological Analysis (IPA), Hermeneutic Phenomenology and Template Analysis (TA) (Langdridge, 2007, p.55-56). An analysis and evaluation of these different approaches is necessary to identify the most appropriate for the current research. This analysis and evaluation process will consider the epistemological position of the research (i.e. what it may be possible to ‘know’ about the game as played via the game as reported). It will also consider practical concerns, such as required sample sizes and type(s) of data to be gathered (e.g. text, audio, or video; short-form or long-form participant responses) in the context of the previously described data source, as well as available resources, equipment, and time with which to carry out the data gathering, analysis, and presentation processes.

8.4.1: Descriptive Phenomenology

Descriptive Phenomenology as generated from the work of Husserl (1970) requires the researcher to accept an epistemological position that holds that there are common, discoverable features of an experience, referred to as “universal essences”. These essences are considered to represent the ‘true’ nature of the phenomenon being studied (Lopez and Willis, 2004, p.728). However, this objectivist position conflicts with the previously defined position of the current research within the contextual constructivism paradigm (Section 8.2), which rejects the view that such “universal essences” describing a singular ‘true’ reality, exist.

Furthermore, Descriptive Phenomenology requires the researcher to ‘bracket’ their own expert knowledge, their prior experiences and their personal opinions to be able to identify the lived experiences of others in as objective a manner as possible. The researcher is required to place themselves in a ‘transcendental’ analytical position, outside of the phenomenon being investigated. This bracketing process, also referred to as epoché (Langdridge, 2007, p.21), is not unique to Descriptive Phenomenology and is a component of many qualitative research methodologies (Tufford, 2012, p.80). However, in Husserlian Descriptive Phenomenology the importance of bracketing is more heavily emphasised so as to aid in the process of ‘stepping out of’ (i.e. transcending) the experience to be able to identify the essences of it.

In the current research, the researcher has been directly involved in the design and development of the disruptive game design philosophy (Section 1.3) and framework (Chapter 6), as well as in the design and development of A:AMFP (Chapter 7) which is the object of study. Bracketing such a deep level of involvement, to the extent required for Descriptive Phenomenology, may thus not be possible.

Practical issues also reduce the appropriateness of Descriptive Phenomenology in the context of the current research. Principally, Descriptive Phenomenology requires maximum variation sampling (Langdridge, 2007) to be able to make claims about a single reality that is commonly experienced. Such a sample is not practical in the current research. This would require access to
participant demographic data (to enable purposeful sampling of different players) that is not readily available from online discussion forums, which effectively anonymise users. This demographic data may be obtainable using other data collection methods (e.g. interviews, focus groups or gameplay diaries), however such methods are not appropriate for the current research.

### 8.4.2: Interpretive Phenomenological Analysis, Hermeneutic Phenomenology and Template Analysis

Interpretive Phenomenological Analysis (IPA), Hermeneutic Phenomenology and Template Analysis can all be applied to studies utilising a contextual constructivist epistemology. All three methodologies favour interpretation of meaning from data (and accept the likely existence of multiple valid interpretations), rather than objective description of a ‘true’ reality. Similarly, all three methodologies accept that bracketing can only be achieved to an extent and thus, a true transcendental perspective is unlikely to be obtainable.

Bracketing in the context of interpretive methodologies is a means of demonstrating validity in the data collection and analysis process (Ahern, 1999, p.407), as far as is reasonably possible in an interpretive process. Indeed, while bracketing is a means of validating data analysis, it is also noted that subjective awareness and preconceptions can often aid researchers. They enable researchers to identify issues or themes emerging from the data by allowing the researcher to identify how that which is emerging fits within the broader picture (Ahern, 1999, p.408).

The three methodologies share a number of similarities in terms of analytical process but differ in key areas that impact the specific contexts in which they are most appropriate to apply. IPA studies for example are “inductive, grounded in the data rather than pre-existing theory, and invariably idiographic, focusing, initially at least, on a single case before moving on to other cases and more general-knowledge claims” (Langdridge, 2007, p.108).

The focus on ‘cases’ in IPA is problematic for the current research. In gathering individual games as reported it may not be possible or practical to gather substantive amounts of data from each individual. Specifically, in the context of online discussion forums, some players may only provide small amounts of data in individual forum entries. The same individual may comment multiple times, however data in this format (i.e. multiple individual ‘comments’) is not comparable to the self-contained, detailed, interview-style ‘cases’ that are implied as the focus of IPA. Referring to short comments on a forum as individual ‘cases’ may thus not be appropriate, as they may not contain substantive data and may not be readily comparable to other comments. Data could alternatively be gathered from more substantive individual sources, such as critical reviews and articles, which could be more appropriately referred to as ‘cases’, although they are again not ecologically valid (Section 8.3).

Hermeneutic Phenomenology raises different incompatibilities with the context of the current research. Specifically, in the methods it favours and the focus it places on creative presentation of analysis findings. Hermeneutic Phenomenology places an emphasis on the researcher as a ‘co-
constructor of meaning’ and thus, tends towards data collection methods that allow flexible communication between researcher and participant, such as unstructured and semi-structured interviews (Langdridge, 2007, p.123). Furthermore, Hermeneutic Phenomenology places importance on the style of the written presentation of research findings and the use of language, creative writing and anecdote in vividly conveying the richness of the experience. While this may have benefits in research that is focused on a more holistic experience of a phenomenon, the current research’s specific focus on the player experience (i.e. the game as played as described in the game as reported) of specific disruptive game features does not require such creative presentation.

Template Analysis is comparable to IPA (King, 2004, p.257) but does not require the analysis to be purely grounded in the data (i.e. it is not a wholly inductive methodology). A set of a priori analysis codes can be deductively generated (e.g. based on the research question or the areas of interest relevant to the specific study) and these codes used to examine the data from a particular perspective. It is expected that the a priori codes are used as a guide, but are “not fixed and [are] very likely to change as the data [is] examined” (Langdridge, 2007, p.125).

Template Analysis is able to be applied from a range of epistemological positions (King, 2004, p.257), from realist “mainstream psychology” through to interpretive positions such as contextual constructivism (Brooks and King, 2012, p.2). Template Analysis is a highly flexible methodology that allows it to be tailored to individual research projects (Brooks and King, 2012, p.3). This is primarily because Template Analysis is a collection of methods with recommendations for use, rather than a rule-defined, distinct methodology in the strictest sense (King, 2004, p.257). The absence of a prescribed set of rules for carrying out Template Analysis potentially reduces the time required for carrying out analysis (e.g. by removing the necessity to carry out steps not appropriate within a particular research project context).

In the current research, the aim is to examine player responses to specific disruptive game features (Chapter 7) in terms of specific psychological impacts (i.e. cognitive engagement through higher-order thinking processes and the reported satiation of high-level psychological needs). In any data collection method that aims to elicit this information from players without alerting those players to the focus of the study (thus potentially influencing the data provided), this information is likely to be embedded within discussion of other, unrelated information about different aspects of the game experience. Template Analysis allows more precise focus through its deductive creation of an initial set of codes, while remaining flexible enough to incorporate codes that emerge from the data (i.e. in the same way that IPA does).

Lastly, Template Analysis is able to be applied to larger numbers of individual sources of data where each source may provide a smaller amount of information. Where purely inductive methodologies require deeper insight into individual ‘cases’ to allow codes to emerge from the data, Template Analysis is able to use the a priori code template to deductively identify specific
meaning in the data, alongside allowing for inductive creation of new codes as required. In practice, this means that less focus is placed on individual ‘case’ analysis and greater focus placed on analysis of trends between individual data sources and across the data set as a whole. “The net effect of these differences is that Template Analysis is generally somewhat less time-consuming than IPA, and can handle rather larger data sets more comfortably” (King, 2004, p.257). Template Analysis is thus an appropriate methodological approach to apply in the current research.

8.5: Ethical Considerations

This study has been approved by the University of Portsmouth Faculty of Creative and Cultural Industries Ethics Committee. This approval was obtained prior to the collection of any data to be used in the study. Documentation pertaining to the ethical approval process can be found in Appendix G.

The data gathered for this study was retrieved from a publically viewable online discussion forum. Users may only post to this forum if they complete the sign up process, however non-members are able to view posts and any attached information freely. Users posting to the forum are made aware of these access settings during the sign up process, and thus are aware of the publically viewable nature of their postings. Users have complete freedom to remove, edit or lock any of their own posts at any time should they wish something removed or made private. Therefore users have been informed of the possible accessibility of their posts and also have the right to withdraw their posts at any time from this public forum.

All data gathered has been stored as PDF (Portable Document Format) files which are direct duplicates of individual web pages. As such, the only identifying information in these raw documents is individual users’ forum aliases as appear next to their forum posts. Any references to specific posts within the research results and discussion however, will also have this information removed to ensure anonymity of individual forum users in the final thesis.

The FG forum does not contain any explicit copyright statement regarding ownership of content posted to its discussion forum. However, individual forum comments can be categorised as user generated content, following the definition provided by the Organisation for Economic Cooperation and Development (Moens, Li and Chua, 2014, p.8). That is, they are published online, required a degree of creative effort to generate, and were not produced as part of a professional process or professional practice. Thus, while the copyright status of such content remains problematic in law (Erickson, 2014), without any specific clause stating otherwise present on the forum, the use of forum comments for the purposes of academic enquiry falls under ‘fair use’.

8.6: Participant Demographic Data and Sample Size

The level of participant anonymity is beneficial from an ethical perspective; however it does limit the available data regarding study participants. Some forum users list basic demographic
information on their forum profiles, such as gender and geographic location, however there is no way to verify the accuracy of this information nor is its presence consistent amongst all users as publication of it on user profiles is optional.

Furthermore, the retrieval of such information is only possible to registered forum users, suggesting a higher level of privacy in relation to the openly accessible forum posts. Retrieval and use of such information for the purposes of research without user consent could therefore be viewed as unethical practice. However, it is unnecessary to retrieve such information as the focus of this research is on player experience rather than statistical comparisons of player data.

A participant factor that is of interest for this work however is players’ level of interest in and experience with playing games, in particular horror and/or adventure games both within FG’s past titles as well as in titles from other developers. This again is not explicitly available, however is more readily able to be inferred from available information. For example, the users of the forum have all demonstrated interest, either positive or negative, beyond that which could be expected of many players, through the process of registering for and voicing their opinions on the forum itself. It is possible therefore, that the player experience data being sought may be representative only of the experience within this self-selected group. This is unavoidable in the context of the methodological approach being used and is a discussed limitation of the research in Section 9.2.

The forum statistics available on the Member’s List database indicated a total member population of approximately 13,460 users at the time of data collection. Of these, approximately 2,240 were active during the period from which data for this study was gathered (indicated by their ‘Most Recent Post’ database information). It is also notable that the forum’s highest simultaneous user count since its creation was at 21:18 on September 10th 2014 (i.e. the release date for A:AMFP), where 490 active users were on the forum.

One final consideration must be noted regarding the participant sample. While the forum is an English-speaking forum, there are no rules that specify posts must be written in English. It is unlikely that non-English posts will be a significant issue however, as forum users may be less able to reply to them thus making posting them initially less worthwhile.

8.7: Data Collection, Organisation, and Analysis Apparatus

The following apparatus was used for collecting, organising and analysing the data:

- A desktop computer with internet access.
- NVivo 10’s NCapture add-on for Internet Explorer 8 (Microsoft, 2009), used for the conversion of web pages into PDF documents ready for coding and analysis.
- NVivo 10, used for collating and coding data.
- Coding memo notebook for keeping written notes during the coding process.
- Microsoft Excel (Microsoft, 2012), used for organising code lists for use during the coding process (i.e. hard copies of working coding template versions).
• CMap Toolset (Florida Institute for Human and Machine Cognition, 2014) used for generating theme maps that provide the basis for discussion topics.

8.8: Coding the Player Data

The process of Template Analysis can be split into distinct stages. These stages are summarised here, with detailed, code-by-code descriptions of each individual stage provided in Appendix H.

First, an initial coding template must be created. King (in Clarke and Gibbs, 2012) recommends that this initial template then be applied to a subset of the data. This initial application assesses the accuracy and relevance of the a priori codes and also allows new codes to emerge from the data subset that can then be incorporated back into the template. The refined template can then be used to examine the remainder of the data set, again being modified to incorporate new codes or to reorganise existing codes into a coding hierarchy. The number of levels within the hierarchy is flexible and can contain as many levels as required to accurately represent the meaning emerging from the data. Additionally, parallel coding (i.e. coding the same data with multiple codes) is acceptable within Template Analysis (King, 2004, p.258), supporting its epistemological position of multiple potential interpretations of a player’s reported experience.

The development of the template took place over a two month period, with three versions of the template being created during this time. Version 1 (Appendix H) of the template was the pre-analysis template (e.g. the a priori code template) based on the key features of the disruptive game design philosophy and its implementation developed in Chapters 1 to 6. Version 2 (Appendix H) consisted of the adapted and expanded template resulting from the subset analysis. Version 3 (Table 8.1) was the resultant template following completed analysis of the full dataset and a second coding pass of the full dataset to ensure consistency.

Three types of codes formed the first version of the coding template. These were ‘F-Codes’ (i.e. Feature Codes), ‘I-Codes’ (i.e. Influence Codes), and ‘T-Codes’ (i.e. Theory Codes). This enabled individual game features to be assigned a ‘FIT’ description. That is, a reference to a particular disruptive game feature, along with its reported influence(s) on the player experience and how the player data for the feature relates to the theory underpinning the disruptive game design philosophy and framework.

Following the exploration of the data subset using the first version of the coding template, as anticipated, a number of new themes emerged from the data. Similarly, some of the disruptive game features were not reported on by players or only reported on in a small number of cases. The coding template was refined into Version 2, to accommodate these new themes. The F-Codes (i.e. disruptive game features) remained unchanged as they referred directly to fixed features in the released game. Likewise, the T-Codes remained unchanged as they also referred to defined components of the previously established theory underpinning the disruptive game design philosophy and framework. Additionally, Version 2 of the template incorporated two new code
types, N-Codes (i.e. ‘Not-Codes’) and C-Codes (‘Concept-Codes’). N-Codes referred to data in which players were discussing game features that had not been designed with a disruptive intent. C-Codes were used to code data in which broader ‘themes’ were being discussed, such as the game’s perceived ‘value’ for players. These were themes that were not directly linked to specific game features but were nevertheless discussed frequently in the data.

With Version 2 of the coding template, a full analysis of the dataset was carried out, with modifications to the template being made as necessary. This resulted in Version 3 of the template once analysis of the full dataset had been completed once. With Version 3 of the template constructed, a second full coding pass of the dataset was carried out. This second pass aimed to ensure that the full dataset had been coded with consideration given to all of the codes within the template. Thus, data that had been coded earlier (during development of the first and second template versions) was reconsidered against codes that had been added to the template later in the process.

A small number of coding changes were made during this second pass (i.e. data being recoded to different codes). Additionally, one structural change was made to the template, with C-Codes being renamed to O-Codes (‘Other-Topic-Codes’), as it became apparent that the conceptual themes being codes previously were too far detached from the main focus of the research to be focused on in depth. Thus, Version 3 of the template (Table 8.1), with O-Codes replacing C-Codes, was used as the basis for the presentation and analysis of the data. A full list of definitions for the codes in Version 3 of the template is provided in Appendix J, however the following discussion of the coded data will provide contextual descriptions of the codes as they are analysed.
## Disruptive Game Features (F-Codes)

- **F_ENVIRONMENT**
  - F_SetDressing
  - F_PigMask
  - F_Non-Euclidean
- **F_ENEMYBEHAVIOUR**
  - F_Wretch
  - F_Engineer
  - F_TeslaPig
- **F_DIFFICULTY**
  - F_CognitiveDifficulty
  - F_PerformativeDifficulty
  - F_Hints
- **F_PLAYERDEATH**
- **F_HIDINGPLACES**
- **F_ITEMINVENTORY**
- **F_LANTERNLIGHT**
- **F_LANTERNF LASHING**
- **F_ENEMYAUDIO**

## Disruptive Game Design Theory Components (T-Codes)

- **T_DISRUPTIVEGAMEDesign**
  - T_Expectation
  - T_IntraludicKnowledge
  - T_TransludicKnowledge
  - T_ExtraLudicKnowledge
  - T_EncodingDisruption
  - T_RecallDisruption
  - T_ActionPlanDisruption
  - T_LowerOrderThinking
  - T_HigherOrderThinking
  - T_SupportHighLevelNeeds
- **T_PLAYERSUPPORTIVEDesign**
  - T_SupportLowLevelNeeds
  - T_IncrementalAccretion
- **T_COUNTERSUPPORTIVEDesign**
  - T_RestrictLowLevelNeeds
  - T_RestrictMidLevelNeeds

## Non-Disruptive Features (N-Codes)

- **N_LACKINGMechanics**
  - N_Sanity
  - N_Puzzles
  - N_EnvironmentInteractivity
- **N_STORY**
  - N_StoryDelivery
  - N_GameplayStoryMismatch
- **N_ATMOSPHERE**

## Influence on Player Experience (I-Codes)

- **I_CHALLENGE**
- **I_COGNITIVEEngagement**
- **I_ENJOYMENT**
- **I_SURPRISE**
- **I_IMMERSION**
  - I_NotRelatable
- **I_HORRIFYING/Psychological**
- **I_TERRIFYING/Visceral**
- **I_TREAT**
- **I_LASTINGImpact**
  - I_EmotionalImpact
  - I_Post-PlayDebate
- **I_CONFUSION**
- **I_FRUSTRATION**
  - I_Patronising
- **I_Boredom**
- **I_DecreasedChallenge**
- **I_DecreasedCognitiveEngagement**
- **I_NotScary**
- **I_FeelingBetrayed**
- **I_Predictable**
- **I_Disappointed**
- **I_NoExploration**
- **I_WhatIsImportantToPlayers**
- **I_UndecidedOpinion**
- **I_HowYouPlayTheGame**

## Other Topics (O-Codes)

- **O_Diff.InterpretationsOfFeatures**
- **O_FranchiseComparisons**
- **O_GenreComparisons**
- **O_MechanicsThatWereRemoved**
- **O_IntendedAudience**
- **O_BaitAndSwitch**
- **O_ConSideringTheDeveloper**
- **O_EvolvingGames**
- **O_Fans**
- **O_RoseTintedMemoryOfATDD**
- **O_UnsatisfiedMoreLikelyToPost**
- **O_Value**
  - O_GamePrice
  - O_GameLength

---

Table 8.1: Coding template (Version 3).
8.9: Coding Results: Identification of ‘FIT’ Descriptions

Carrying out matrix coding queries on the code template in NVivo, it was possible to identify cases of code co-occurrence (i.e. parallel coding). Identifying clusters of co-occurring ‘F-Codes’, ‘I-Codes’, and ‘T-Codes’ enabled ‘FIT’ descriptions to be created, linking disruptive game features (‘F-Codes’), experiential influences on the player (‘I-Codes’), and elements of the disruptive game design theory (‘T-Codes’).

Summarised ‘FIT’ descriptions (Table 8.3) were created for each of the disruptive game features that were present in the game as created (Section 7.8.2). Alongside each F-Code, the I-Codes are listed in descending order of how frequently they were co-occurring alongside that F-Code. The T-Codes are combined into a modality of disruption (i.e. Encoding Disruption, Recall Disruption, and Action Plan Disruption) and disrupted knowledge type(s) (i.e. intraludic, transludic, extraludic; procedural, semantic, episodic). In some instances, multiple modalities and knowledge types have co-occurred alongside player discussion of different F-Codes, in which case they are also organised in descending order of co-occurrence frequency.

Alongside the FIT descriptions, two further columns present the intended disruptive impact(s) of the disruptive game features in the game as created alongside the eventual reported impacts of those features in the game as reported. Each feature in the table is colour coded (Table 8.2) to represent the extent to which the intended disruptive impact(s) appear to have been experienced and reported on by players.

<table>
<thead>
<tr>
<th>Colour</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>All of the reported disruptive impact(s) of the game feature correspond closely to intended disruptive impact(s) of the game feature.</td>
</tr>
<tr>
<td>Blue</td>
<td>At least one of the reported disruptive impact(s) of the game feature corresponds closely to the intended disruptive impact(s) of the game feature.</td>
</tr>
<tr>
<td>Orange</td>
<td>None of the reported disruptive impact(s) of the game feature correspond closely to the intended disruptive impact(s) of the game feature.</td>
</tr>
</tbody>
</table>

Table 8.2: Colour coding scheme for list of disruptive game features as reported on by players.
<table>
<thead>
<tr>
<th>Disruptive Game Features (F-Codes)</th>
<th>Co-Occurring Influences on the Player Experience (I-Codes)</th>
<th>Co-Occurring Links to Disruptive Game Design Theory (T-Codes)</th>
<th>Intended Disruptive Impacts of Game Feature in the Game as Created</th>
<th>Reported Disruptive Impacts of Game Feature in the Game as Played</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-Euclidean Space (Section 7.5.4)</td>
<td>Horrifying/Psychological Cognitive Engagement Post-Play Debate Confusion</td>
<td>Encoding Disruption of Intraludic Semantic Knowledge</td>
<td>Encoding Disruption (INT-S)</td>
<td>Encoding Disruption (INT-S)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>NOT REPORTED</td>
</tr>
<tr>
<td>Wretches and Early Game Enemy-to-Player Interaction (Section 7.5.6.2)</td>
<td>Emotional Impact Horrifying/Psychological Not Scary</td>
<td></td>
<td>Encoding Disruption (INT-S)</td>
<td>Action Plan Disruption (INT-S)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Recall Disruption (TRANS-S, TRANS-E)</td>
<td>Action Plan Disruption (INT-S)</td>
</tr>
<tr>
<td>Engineers and Enemy-to-Enemy Interaction (Section 7.5.6.4)</td>
<td>Emotional Impact Terrifying/Visceral Cognitive Engagement</td>
<td>Action Plan Disruption of Intraludic Semantic Knowledge</td>
<td>Recall Disruption (INT-S)</td>
<td>Action Plan Disruption (INT-S)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tesla Pigs and the ‘Boss Enemy’ Game Trope (Section 7.5.6.7)</td>
<td>Terrifying/Visceral Not Scary</td>
<td>Action Plan Disruption of Intraludic Semantic Knowledge</td>
<td>Action Plan Disruption (INT-S, TRANS-S)</td>
<td>Action Plan Disruption (INT-S)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Providing the Player with Hints (Section 7.5.8)</td>
<td>Decreased Immersion Decreased Cognitive Engagement Frustration</td>
<td>NO CO-OCCURRING T-CODES</td>
<td>Encoding Disruption (INT-P, INT-S)</td>
<td>NOT REPORTED</td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Disruptive Game Features (F-Codes)</th>
<th>Co-Occurring Influences on the Player Experience (I-Codes)</th>
<th>Co-Occurring Links to Disruptive Game Design Theory (T-Codes)</th>
<th>Intended Disruptive Impacts of Game Feature in the Game as Created</th>
<th>Reported Disruptive Impacts of Game Feature in the Game as Played</th>
</tr>
</thead>
<tbody>
<tr>
<td>Handling the Death of the Player Character (Section 7.5.9)</td>
<td>Decreased Challenge Not Scary</td>
<td>NO CO-OCCURRING T-CODES</td>
<td>Recall Disruption (INT-S, INT-E, TRANS-S, TRANS-E)</td>
<td>NOT REPORTED</td>
</tr>
<tr>
<td>Hiding Places (Section 7.5.10)</td>
<td>How Different Players Play the Game Decreased Cognitive Engagement</td>
<td>Action Plan Disruption of Transludic Semantic Knowledge</td>
<td>Recall Disruption (TRANS-E)</td>
<td>Action Plan Disruption (TRANS-S)</td>
</tr>
<tr>
<td>Removal of Item Inventory (Section 7.5.11)</td>
<td>Enjoyment Increased Immersion Decreased Cognitive Engagement</td>
<td>Recall Disruption of Transludic Procedural and Episodic Knowledge</td>
<td>Action Plan Disruption (TRANS-S)</td>
<td>Recall Disruption (TRANS-P, TRANS-E)</td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Disruptive Game Features (F-Codes)</th>
<th>Co-Occurring Influences on the Player Experience (I-Codes)</th>
<th>Co-Occurring Links to Disruptive Game Design Theory (T-Codes)</th>
<th>Intended Disruptive Impacts of Game Feature in the Game as Created</th>
<th>Reported Disruptive Impacts of Game Feature in the Game as Played</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enemy Audio Cues (Section 7.5.13)</td>
<td>Not Scary Horrifying/Psychological Enjoyment Confusion Predictable Cognitive Engagement</td>
<td>Recall Disruption of Transludic Semantic Knowledge Action Plan Disruption of Transludic Semantic Knowledge</td>
<td>Recall Disruption (INT-S, INT-E)</td>
<td>Recall Disruption (TRANS-S)</td>
</tr>
<tr>
<td>Manipulation of ‘Set Dressing’ Entities and Props (Section 7.5.14.2)</td>
<td>Horrifying/Psychological</td>
<td>Recall Disruption of Intraludic Semantic Knowledge</td>
<td>Recall Disruption (INT-S, INT-E, TRANS-S, TRANS-E)</td>
<td>Action Plan Disruption (TRANS-S)</td>
</tr>
<tr>
<td>Pig Mask Motif (Section 7.5.14.3)</td>
<td>Horrifying/Psychological Post-Play Debate</td>
<td>Encoding Disruption of Intraludic Semantic Knowledge</td>
<td>Encoding Disruption (INT-S)</td>
<td>Encoding Disruption (INT-S)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Recall Disruption (INT-S)</td>
<td>Recall Disruption (INT-S)</td>
<td>NOT REPORTED</td>
</tr>
</tbody>
</table>

Table 8.3: Summary of FIT descriptions for each disruptive game feature in A:AMFP, their intended disruptive impacts, and their player-reported disruptive impacts.
With regards to the coding of the player data in relation to the disruptive game design theory (i.e. T-Codes), it is important to emphasise that the identification of different modes of disruption required a degree of researcher interpretation in the application of the coding template. This is because players were unaware of the disruptive game design philosophy underpinning the game’s design and development and thus, did not discuss disruption explicitly. Thus, coding of the data with appropriate T-Codes was based on researcher identification of ‘disruption-like’ experiences that presented clear parallels with the definitions for Encoding Disruption, Recall Disruption and Action Plan Disruption (Section 4.3).

Where these disruption-like experiences also referred to knowledge that was in line with the previously defined types of knowledge (i.e. INT-P, INT-S, INT-E, TRANS-P, TRANS-S, TRANS-E, EX-P, EX-S, and EX-E) (Section 3.3), the knowledge types were added to the mode of disruption to create fuller descriptions of the disruption-like experience (e.g. Action Plan Disruption of INT-S knowledge). T-Code co-occurrence was consistent enough in some instances to allow a single disruption mode and knowledge type to form the ‘FIT’ description. For example, The Pig Mask Motif (F_PigMask) co-occurs exclusively with T_EncodingDisruption and T_IntraludicKnowledge. In other instances, multiple players reported evidence of different modes of disruption and also effects on different knowledge types. The Flashing Lantern Warning System (F_LanternFlashing), for example, co-occurs in different player comments with T_EncodingDisruption, T_RecallDisruption, T_ActionPlanDisruption, T_IntraludicKnowledge, and T_TransludicKnowledge.

In terms of context, 3 disruptive game features (i.e. those coloured green) have demonstrated a significant correspondence between all intended disruptive effects and all reported disruptive effects (Table 8.3), while another 3 (i.e. those coloured blue) have demonstrated at least a partial correspondence. Of the remaining 6 disruptive game features (i.e. those coloured orange), only 2 were not associated with any reported disruptive impacts at all, while 4 were associated with disruptive impacts that were different to those intended in the game as created.

8.10: Analysis and Discussion of Study Findings

The first stage of the analysis process (Section 8.10.1 and Section 8.10.2) is to explore the individual disruptive game features in depth, drawing on individual player comments to investigate possible reasons underpinning how players reported their game experiences in relation to those game features. This analysis also enables identification of indicators in the data of cognitive engagement by players and also, of high-level need satiation being achieved during play.

Half of the reported disruptive impacts of the different disruptive game features are either wholly, or partially, in line with the intended disruptive impacts (Table 8.3). This may be suggestive of, at least, successful design and development work leading to an intended player experience outcome. Conversely however, a number of disruptive game features have resulted in players reporting experiences that are representative of different modes of disruption to those that were intended, or disruption of different knowledge types to those that were intended. This presents an issue in
defining what is and is not, disruptive game design. In the context of the previous definition of disruptive game design as needing to be intentional (Section 1.2), disruptive game features that have a different disruptive effect, than designed, cannot be considered ‘successful’. However, they can also not be considered ‘failed’ disruptive game features if they had an identifiable disruptive effect on players. Thus, in such situations it is necessary to consider whether the originally designed disruptive effect was the most appropriate and whether the design was correctly implemented (i.e. in the game as created). That is, does the cause for players not experiencing the intended disruptive impact lie in an incorrect theoretical understanding of the disruption modality itself, or simply an incorrect implementation at the practical, developmental level.

8.10.1: Disruptive Game Features with a Correspondence between Intended and Reported Disruptive Impacts
Six disruptive game features achieved a correspondence between intended and reported disruptive impacts. These features can be grouped into two sets for the purpose of discussion, based on the contexts in which they were discussed by players. Firstly, discussion of ‘Environmental’ disruptive game features (i.e. Non-Euclidean Space, the Pig Mask Motif, and the Manipulation of ‘Set Dressing’ Entities and Props) (Section 8.10.1.1) and secondly, discussion of ‘Enemy Agent Interaction’ disruptive game features (i.e. The Flashing Lantern Warning System, the Enemy Audio Cues, and the Tesla Pig enemy) (Section 8.10.1.2).

8.10.1.1: ‘Environmental’ Disruptive Game Features
The use of non-Euclidean space (i.e. \( F_{\text{Non-Euclidean}} \)), the use of the pig mask motif in the game (i.e. \( F_{\text{PigMask}} \)), and the manipulation of ‘set dressing’ entities and props (i.e. \( F_{\text{SetDressing}} \)) each achieved at least partial correspondence between their intended and reported disruptive impacts (Table 8.3). These three disruptive game features are also frequently discussed together in the dataset, likely due to their functional and aesthetic similarities (i.e. they all involve objects in the environment, or parts of the environment itself, moving, appearing, or disappearing by themselves). These disruptive game features specifically demonstrate a correspondence between intended Encoding Disruption and reported Encoding Disruption (in relation to Non-Euclidean space and the pig mask motif), as well as intended Recall Disruption and reported Recall Disruption (in relation to the manipulation of other ‘set dressing’ entities and props). Moreover, they also demonstrate examples of players cognitively engaging with the game beyond its ‘surface level’, supporting the satiation of high-level needs for some players (Section 1.2.3.1).

In relation to each of these disruptive game features, Encoding Disruption was achieved as intended by providing players with ambiguous stimulus information (i.e. pig masks and changes to level architecture/set dressing’ that had no explicit in-game explanations). Players reported a range of effects that these features had on their play experiences. Interestingly, different players reported very similar experiences but framed those experiences in dissimilar ways; as being either positive, enjoyable experiences, or as negative, detrimental, experiences. For example, the
The following forum comment extract is from a player’s description of their “biggest complaints” about the game:

The scene cuts were confusing [. . . ] I turned around and the door disappeared. I actually was shocked by that and wrote it off as a freaky hallucination thing. But then it kept happening. I mean, near the end I was even questioning whether the Machine was even real at all, and Mandus was just an insane person who killed his kids. (Forum Post 03, 2013)

![Portrait 'hallucination'/insanity effect in A:TDD.](image)

Such one-off hallucinations were used in A:TDD if the player-character’s ‘sanity’ level dropped below a certain threshold. For example, the changing portrait of A:TDD’s antagonist, Alexander, from a normal portrait to a warped and twisted version of it (Figure 8.3). It is interesting that while this player does not refer to A:TDD directly, the phrasing used is reminiscent of a key feature of A:TDD. The result of this experience for this player was that they ended up “questioning” what was happening in the game and specifically, what was happening in terms of the game’s ‘told story’. The player has however kept their evaluation within the diegesis of the game series, attempting to use it as a cue for constructing a diegetically situated narrative, rather than interpreting it as an unintentional game ‘bug’. This player associates a sense of “confusion” with a negative game experience (as they have placed this comment under the heading of “biggest complaints” in the full forum comment). This association is not however found in many other players’ comments.

However, even if some responses to the Non-Euclidean space disruptive game feature were negative, the ongoing use of this feature along with the ‘set dressing’ manipulation and pig mask motif features has encouraged players to cognitively engage with the game beyond its ‘surface level’. That is, the game content is not presented in a way that clearly supports a single, explicitly provided explanation but instead in a way that supports the emergence of different, individual, player-constructed narratives. These constructed narratives are any narratives constructed using
in-game cues but that were not explicitly intended by the design and development team, as per Dansey and Stevens’ (2008, p.2) definition of emergence as being any outcome of play not explicitly intended by the designer. This potential for emergence in turn supports players’ cognitive engagement with the game by encouraging active construction of knowledge (i.e. the high-level need for creativity) and using that knowledge to interpret the game content (i.e. the high-level need for problem solving).

The particular comment in Forum Post 03 was framed as being a negative experience for this particular player. However, similar descriptions are provided by many other players that frame this same opportunity for cognitive engagement with the game beyond its ‘surface level’ properties as being a positive part of their play experience.

In reference to the non-Euclidean architecture feature, two players specifically cite confusion as a positive experience:

There were a few cool moments when objects or even doors and passages would appear/disappear leaving you a little confused or freaked out for a moment, but in a good way! (Forum Post 04, 2013)

I really like the reshaping of the levels and how it confused me (especially in [the level] where you have to look for the ingredients). (Forum Post 05, 2013)

However, the positive player responses to this type of engagement, stemming primarily from Encoding Disruption via ambiguous stimulus information, can be more readily demonstrated by drawing on player comment examples from the F_PigMask code. This particular disruptive game feature attracted a number of more substantive comments from players than non-Euclidean space did. Different players appear to have assigned different meanings to the pig masks, considering them from different perspectives.

Each time you notice a pig mask, it just appeared, and it means probably something changed in the level, or a door just closed. I was sure of it at the first place you mention, you take a corridor that loops back to the place you came from (there's steam so you can't [cross] the catwalk), when you come back out of it, both doors disappear (and the steam is gone). It's made obvious [near] the end (you clearly get teleported, then a bunch of pig mask drop [from above]). Not sure how it is to be taken. (Forum Post 06, 2013)

This player describes an interpretation of the pig masks as being linked to the ‘mechanical’ functionality of the game. This interpretation suggests that the appearance of a pig mask signifies an ‘update’ to the layout of the game level (such as doors closing). However, this player recognises at the end of this comment that they are not certain how the masks are intended to be interpreted, leading other players to consider the possible story-based meaning(s) of the pig masks.

The Pig Masks! Who was putting them everywhere? What was their purpose? I know they are like Aztec masks like people were talking about before the game came out (The Jaguar masks for sacrifice), but I don’t know anything else about
them! I especially want to know why they kept appearing everywhere. (Forum Post 07, 2013)

It [the pig mask] was like the trail. Hansel and Greta [sic], breadcrumbs, etc. (Forum Post 08, 2013)

They [the pig masks] appeared because we are insane I assume. That's all there is to it probably. (Forum Post 09, 2013)

The interpretations presented in Forum Posts 07, 08 and 09 discuss the pig masks’ meaning in terms of the game’s ‘told story’. Forum Post 07 assumes that a homodiegetic character or entity is placing the masks around the game levels for some reason. Forum Post 09 assumes that they are indicators of the player character’s insanity. ‘Sanity’ is not used as a game mechanic in A:AMFP as it was in A:TDD and thus it is interesting to see that players once again are describing features of A:AMFP using terms and themes from A:TDD. However, there is no direct evidence to suggest that this is an example of transludic knowledge influencing player interpretation of the game. Forum Post 08’s interpretation that makes a link between what this player views as a ‘trail’ of pig masks throughout the game does however provide a strong example of a player utilising extraludic knowledge (knowledge from, for example, the Hansel and Gretel fairy tale) to construct understanding of a disruptive game feature.

The aim of disruptive game features that utilise Encoding Disruption is to make initial construction of knowledge more demanding and thus, more cognitively engaging for players. In practice, as can be seen in relation to the pig mask motif in particular, this may cause post-play discussion and debate between players in their games as reported, about specific in-game stimuli that were ambiguously presented. In encouraging this discussion around the game in online forums, an active community can be constructed which provides a form of ‘added value’ for players after they have finished playing the game. This ‘added value’ is referred to by Ang, Zaphiris and Wilson (2010) as “extrinsic play”, or play that occurs around the game without direct engagement with the game itself. Their work suggests two components of extrinsic play, in reflective play and expansive play.

Players are motivated to play by reflective play in which they want to talk about the game with others and be part of the player community; and players are also motivated to explore what they can do with the game to test the game boundary and to expand the game through expansive play. (Ang, Zaphiris and Wilson, 2010, p.372)

The analysis of players’ engagement in reflective play in particular (i.e. entering into discussion with other players via, for example, online discussion forums) demonstrated that reflective play “is not just an action that resolves contradictions [between player experiences] but also an activity that contributes directly to the fun and enjoyment of game play” (Ang, Zaphiris and Wilson, 2010, p.372). Thus, Encoding Disruption via ambiguous stimulus information that provides a catalyst for this reflective play has the potential to provide added player enjoyment of the game beyond the experience of the game as played. Indeed, this evidence (Ang, Zaphiris and Wilson, 2010) suggests
that the player’s creation of the game as reported may itself be part of the player’s overall enjoyment of the game.

The frequency of discussion of these particular disruptive game features is relatively low in comparison to other game features, both disruptive and non-disruptive (as indicated by code occurrence frequency). The specific low frequency of discussion of the pig mask motif and other ‘set dressing’ manipulation is explicitly noted by one forum user as well, who identifies that, even in the mainstream gaming media’s coverage of the game, these features are rarely mentioned.

One great thing A:AMFP did, that as far as I’ve read is not mentioned at all in "big" magazines/sites review, is the fascinating subtlety of the changes in the game world since the very beginning (maybe because they are very easy to overlook). Pig mask appearing/changing, one mask that is visible when looking behind a painting but not when you are physically in the room, the owl statue that goes to the other side of the “hunting” room and more. This was really a high point of the game for me. (Forum Post 10, 2013)

Indeed, the manipulation of ‘set dressing’ entities and props, as well as the manipulation of the pig mask motif throughout the game are, by design, subtle changes within the game environments. Thus, this may explain the low frequency of discussion between players, although this cannot be confirmed. The subtlety of the changes made to the environment and game entities, combined with those changes having little direct impact on the player’s primary objectives of exploration and progression, could mean that even if players noticed the changes they may have deemed them not important enough to form a part of their game as reported. Hence, the low frequency of discussion of these features in the dataset may not necessarily be an accurate indicator of their impact on the player experience in the game as played, rather, an indicator of their perceived importance to report in the game as reported.

Even within the small number of player comments regarding these disruptive game features (as well as non-Euclidean architecture), there is evidence to suggest that Encoding Disruption of intraludic semantic knowledge has occurred as intended in the original design and has provided an opportunity for players to reach a range of different understandings of the meaning of these different game features (i.e. emergent constructed narratives). To place this data in a broader context, further supporting evidence of post-play discussion and different individual interpretations of the pig mask in the game can be identified on the game’s player-run wiki, which has a page dedicated to the pig mask and speculation about its meaning (Wikia Inc., 2014).

8.10.1.2: ‘Enemy Agent Interaction’ Disruptive Game Features

There are further disruptive game features that also achieved at least a partial correspondence between their intended and reported disruptive impacts. These are the flashing lantern system (i.e. F_LanternFlashing), the audio cues attached to the enemy agents in the game (i.e. F_EnemyAudio), and the player encounters with the Tesla Pig enemy (i.e. F_TeslaPig). These are often discussed in the context of, or in direct relation to, each other. Thus, they can be similarly discussed together here. Each of these three disruptive game features has been reported on by
players with reference to in-game experiences that are ‘disruption-like’ in nature and that provide
evidence for Action Plan Disruption. Both Encoding Disruption and Recall Disruption are also
evident in player discussion of these features but to a lesser extent.

Player discussion of these disruptive game features makes frequent references and direct
comparisons to similar features in A:TDD. However, there is an interesting split between players
that make such comparisons as a means of critiquing A:AMFP and those that make such
comparisons as a means of praising the game and how different it is in its mechanical design from
A:TDD. While the comparisons being made by players are often identical, the ways in which those
comparisons are framed, varies. It is also notable that a number of players discuss the features of
A:AMFP without any reference to A:TDD (or indeed, other games) and that many of these players
present positive opinions on the game. Further discussion of how players frame their comments is
provided in Section 8.10.2.1.

The flashing lantern system was designed and created (Section 7.5.12) with the intent of providing
different modes of disruption at different points over the course of the entire game's play time.
This was the only disruptive game feature that was designed to notably ‘evolve’ over the course of
the game, with other features designed instead with the intent of providing one, or at most two
modes of disruption, usually at specific points or within specific sections of the game.

Some players noted that the flashing lantern system was functionally similar to the use of enemy
audio cues in A:TDD but that the A:AMFP system was able to instil a greater sense of uncertainty
and anxiety.

Do you not remember the monster spawning noises that gave you a warning [in
A:TDD]? Personally, I find this flickering [in A:AMFP] a LOT scarier because you
have no idea where they are except close. (Forum Post 11, 2013)

This player is reporting experiencing Action Plan Disruption of intraludic semantic knowledge. If
the player performs an identical action in response to the flashing (flickering) lantern in different
scenarios, there may be different consequences each time. This is caused by the system only
providing players with partial information about the location of enemies, making learning a
‘correct’ action impossible. Players must respond in a contextually appropriate way for each new
enemy encounter, thus requiring consistent cognitive engagement throughout the game to
effectively approach each new enemy encounter. This example also demonstrates Recall
Disruption of transludic semantic knowledge, with the player recognising the concept of a warning
system for enemy locations (from A:TDD) but being unable to apply that existing knowledge to the
context of A:AMFP.

The monster and chase music [in A:TDD] are a pretty obvious
"ooooooooooooooohhhh there's a monster" clue so by omitting them they keep
you on your toes. Also like that they have "random" music playing from out of
left field to toy with our preconceptions. The lantern and lights flickering also
function as a 'tell' but, again, they don't guarantee an encounter or anything.
That's just there to fuck with the player. (Forum Post 12, 2013)
When they [the enemy agents] did show, I would know in advance because the lights would flicker. But many times it didn’t mean that the monsters would come to attack, so I don’t think it was a bad thing (like removing the surprise factor). (Forum Post 13, 2013)

My favourite feature about them is the fact that instead of spawning by sounding off loudly like the Grunts and Brutes [the enemies in A:TDD], the lights and lantern will instead flicker if they are nearby. (Forum Post 14, 2013)

Forum Post 12 and Forum Post 14 demonstrate Recall Disruption of transludic semantic knowledge. Forum Post 12 states that A:AMFP “toys with preconceptions” by playing “random” music. This statement follows on from a description of the enemy audio cues in A:TDD earlier in the post and thus, it is likely that the player is referring to preconceptions formed via reliance on this transludic knowledge of A:TDD’s enemy audio cues. Likewise, Forum Post 14 refers to knowledge of enemy behaviour (i.e. Grunts and Brutes) in A:TDD. Forum Post 13 meanwhile demonstrates Action Plan Disruption of intraludic semantic knowledge, with the player identifying that the lantern can be used to predict enemy encounters, but cannot be used consistently for this purpose (i.e. the flickering does not guarantee an encounter).

While there were a number of further positive responses to the lantern flashing system, the majority of these did not specify a particular reason behind the positive response (e.g. players stating simply that they “liked the way the lantern functioned” or that they “enjoyed the flashing lantern mechanic”). It is useful to know that these players responded positively to the feature however. Disruptive game features should exist as a coherent part of the game and, as previously discussed, whether a player experiences a sense of disruption in response to any particular disruptive game feature is dependent on their individual existing schemas. Thus, a disruptive game feature that is responded to by players positively even if they do not place any particular emphasis on it suggests that it is successfully existing as a coherent part of the game; a disruptive game feature for some players, while simply an enjoyable part of the game for others.

The flashing lantern system was generally positively received; however a number of negative responses were also identified in the data. Where positive responses were based on the lantern flashing system’s ability to create uncertainty during gameplay, negative responses appear to be based on players feeling that the system was consistent, predictable and thus, made the game too easy and less scary.

The way your latern [sic] works when you point it at the monster is a very questionable design decision. I hope this isn’t considered spoiler btw [by the way]? Anyways [sic], when you look in the general monster direction with your latern [sic] out, it will flicker. Therefore, in all scenarios except tight corridors, having your latern [sic] out is THE best way to be safe against the monster. Since it will react to the monster even if the monster is behind an obstacle, like a crate, you will know that the monster is there about 2 or 3 seconds before it shows up in your view (and gets a chance to spot you for lighting him up). So you can always put the latern [sic] away and hide and wait for him to pass by. (Forum Post 15, 2013)
While this player notes that there is a difference in the game’s tight corridors, it is felt that having the lantern equipped constantly is the best way to play the game and detect enemies. Indeed, the design of the lantern flashing system is intended to serve as warning system for players, which balances the threat presented by the fast enemies with an opportunity to be forewarned of their presence. However, it is evident that this player has not experienced the intended Action Plan disruption as they have not noted the rule changes over the course of the game. Similarly, some other players did not identify the designed inconsistency in the lantern flashing system, referring to being able to easily detect enemies.

Make the light flickering random. So you can’t detect the monsters every time. (Forum Post 16, 2013)

The flickering light every time an enemy is near? Not only does it feel ripped straight from FEAR [assumed to be F.E.A.R. 2: Project Origin (Monolith Productions, 2009) which uses a flickering flashlight to indicate supernatural activity], it was the one bad thing about FEAR. Why would you want to know when shits [sic] about to go down? Ruins any suspense. (Forum Post 17, 2013)

While these negative responses were in the minority, they nevertheless raise some issues regarding disruptive game design more broadly. The first of these issues is that subtle changes to a game feature over the course of a game may be overlooked by players. Forum Post 16 for example, suggests making the flashing random. However, there are already multiple ‘fake’ enemy encounters in the game that trigger a ‘false positive’ lantern flash, as well as some encounters where the lantern flashing does not begin until the player is much closer to the enemy agent than may be expected. This semi-randomisation was evidently not experienced in this player’s case however. This demonstrates that the game as played is an inherently individual experience and thus may not be the same as the game as designed, the game as created, or the game as published.

In some instances, failure to notice subtle changes such as this may not be problematic. If the disruptive game feature is experienced simply as an enjoyable (e.g. Forum Post 14’s response to the flashing lantern system), non-disruptive feature, then the overall player experience may still be positive. However, the opposite effect of not noticing these subtle changes is that players perceive a ‘static’ game feature that never changes over the course of the game.

Given that disruptive game design is intended as a means of increasing cognitive engagement with a game, this outcome can be considered as a ‘failure’. A ‘static’, unchanging game feature is even less cognitively engaging than a ‘conservative’, incrementally accretive game feature, as even incremental changes require some minimal cognitive effort from players to incorporate them into their existing schemas. In A:AMFP, only a small number of players appear, from the game as reported, to have perceived the flashing lantern system as consistent and unchanging, even though the total number of players that have commented on the flashing lantern in some way is comparatively high. However, it may equally be possible that players that perceived the flashing lantern as ‘static’ throughout the game did not feel it necessary to comment on. This is a
limitation of attempting to understand the game as played via the game as reported, as data is filtered through individual players’ ‘importance’ filter before being committed to text.

Consideration should thus be given in future applications of disruptive game design, to the potential benefits and risks of each disruptive game feature if experienced by players as disruptive or non-disruptive. Specifically, designers should consider the question of what may happen if players do not experience disruption (e.g. if they do not possess the necessary schematised knowledge to be disrupted). In this case, it is necessary to consider what the possible alternative interpretations or understandings of the disruptive game feature may be.

The second issue raised specifically by Forum Post 17 is the effect of transludic knowledge on player responses to a disruptive game feature. Disruption of transludic knowledge is intended to be a means of encouraging players to learn new intraludic knowledge and limit their ability to rely on prior learned knowledge from other game experiences. However, the effect of a player’s previous responses (i.e. positive or negative responses) to features also in other games has not been considered thus far in this research. While Forum Post 17 does state a specific reason for disliking the flashing lantern system (i.e. it “ruins any suspense” in the game), it raises the question of whether a player is more likely to respond negatively to a game feature that they have previously responded negatively to in other games. Forum Post 17 states that the flashing light was “the one bad thing about FEAR”. Thus, despite the fact that A:AMFP’s system functions differently, the initial perception of the familiar aesthetic qualities of the flashing lantern feature may have predisposed the player to responding negatively to it based on their existing transludic knowledge, although there is not enough other evidence to investigate this possible effect further. However, this could be addressed in future studies (Section 9.3) and may also provide further insight into how players form initial impressions of a game whilst playing it.

The enemy audio cues in A:AMFP have been largely successful in terms of their disruptive properties, with evidence of Recall Disruption and Action Plan Disruption in the game as reported, although with some different reported disrupted knowledge types to those intended. As identified in Forum Post 12, not using overt, identifiable, audio cues associated with enemy agents makes the game more demanding for players. Other forum comments also support this.

I also liked that they removed any music when you encounter the monsters (except when they chase you), because I didn’t know what to expect. It wasn’t the same old "I hear music so there’s a monster" routine anymore, which felt refreshing. (Forum Post 19, 2013)

The monster no longer have [sic] music attached to them. Finally, there is no way to know whether there is a monster out there hunting for you, or there isn’t. You don’t get hinted "okay, Mr. Face [a slang term amongst players for the main enemy in A:TDD] spawned, time to get in cover" - and thus you naturally can get surprised by the monster or literally run into it (which was the most frightening experience for me in the game, but I won’t tell you where and under what conditions does [sic] it happen, so you can experience it fresh yourself). (Forum Post 20, 2013)
While these comments are positive, it is interesting to note that both of these players suggest that all enemies in the game are devoid of audio cues (except when they directly chase the player character) when in fact, this is only the case in some encounters. This disruptive game feature was intended to provide Action Plan Disruption of transludic semantic knowledge in the scenarios where audio cues were absent completely, which appears to have been successful. However, Recall Disruption of intraludic semantic knowledge was also intended to occur in scenarios that made use of enemy audio cues in different ways (e.g. attaching audio cues to some enemies, or triggering audio cues without an associated enemy threat). This disruptive mechanism throughout the game does not appear to have been identified by players, in the game as reported. The majority of players appear to apply their initial perception (i.e. that there are no enemy audio sounds used in A:AMFP) to their experience of the game as a whole. A few players conversely state that the enemies do have audio cues attached to them throughout the game but that they are much quieter than the cues in A:TDD. In reality, the game makes use of a combination of different encounters, using enemies with and without audio cues.

While the previously described subtle changes to the flashing lantern system may be understandably missed by some players, it is surprising that a more overt change (i.e. the absence of music in some scenarios and the clear presence of music in other scenarios) appears to also have been largely unnoticed by players. However, the similarity between both the player responses to the flashing lantern system and the player responses to the enemy audio cues is that in both cases, a high number of players appear to apply their initial perception of a disruptive game feature to their experience of the game as a whole. This may suggest that players expect a game to be stable and to signpost any changes that occur during gameplay (e.g. changes to a game’s properties that make the player’s existing knowledge inaccurate). Thus, they become less likely to notice changes as the game progresses because they do not expect changes to occur without them being made explicit. This is only one possible interpretation of the data but this nevertheless is an interesting avenue for future research (Section 9.3), which may lead to insight into how players more broadly apply their attention to different game features at different stages of a game.

Both the flashing lantern system and the enemy audio cues were frequently discussed with respect to the Tesla Pig enemy type and how it appeared to represent the established ‘boss enemy’ game trope. While the Tesla Pig encounters were intended to provide a source of Action Plan Disruption of both transludic (i.e. the cross-game ‘boss enemy’ trope) and intraludic (i.e. the Tesla Pigs’ behaviour compared to other enemies in A:AMFP) semantic knowledge, the data only revealed evidence to support disruption of intraludic knowledge. Players that discussed the Tesla Pig in relation to its transludic ‘boss enemy’ properties did so from the perspective of it fitting that trope, rather than disrupting it in any notable way.

The Action Plan Disruption of intraludic knowledge was intended to stem from the inability of players to respond to the Tesla Pig using strategies developed up to that point in the game. Hiding, or moving slowly while using stealth to avoid enemies, is a viable strategy in response to
the Wretch and Engineer enemy types encountered throughout the game, but is significantly less effective against the faster, more powerful, Tesla Pig. A number of player comments support the interpretation that this disruption has been experienced in multiple cases.

I enjoyed the 'Tesla' encounters. I liked how the game changed the rules, which is a good method for getting players out of their comfort zones. My whole strategy leading up to that point was to creep around in the dark, listening and [turning] the lantern on only when I needed to light up a section or find a pathway. It was enjoyable to find that method didn’t really work against 'Tesla' (I died once, 2nd time I led him away from the control panels and then ducked around the vat things to clear the room). I died in the small maze section [the second Tesla Pig encounter] once as well - from going back to my tried and true method of hiding in a dark corner. Was a pretty good adrenaline rush in comparison to most of the game being a large build up of fear. (Forum Post 21, 2013)

This concisely reports the effect of Action Plan Disruption of intraludic semantic knowledge, specifically referring to ‘rule changes’ and ‘strategies’ for handling enemy encounters in the game. This is further supported by the player’s admission that they returned to their previous method (i.e. attempting to use stealth to avoid the enemy agent) and subsequently ‘died’. The impact of this disruption as reported by this player was the requirement to move out of their ‘comfort zone’. In doing so, the suggestion can be made that moving out of one’s ‘comfort zone’, with respect to gameplay strategies, requires cognitive engagement on the part of the player to evaluate, create and then apply new strategies.

Further evidence of players experiencing Action Plan Disruption in relation to how the game requires them to ‘strategize’ against the Tesla Pig can be identified in other player comments.

But when entering the Tesla area I was suddenly scared beyond everything good and evil. In fact when the Tesla Pig turned up I sat there in a corner and got slaughtered. I could not move my fingers. And just realizing: "Dude, I cannot just HIDE from that thing!" Priceless. (Forum Post 22, 2013)

[The first Tesla Pig encounter] was really panicking [sic] because you are used to pigmen that don’t notice you and are slow. Here suddenly there is this big electrical guy and his [sic] fast, I had to run and look back constantly to see where he was. After both fights I was bloody red (meaning one hit and I die) so it was intense and felt really close. (Forum Post 23, 2013)

While these further comment examples are not as concise, they are nevertheless identifiably referring to the consequences of actions performed by each player. In Forum Post 22, the player refers specifically to being unable to hide from the Tesla Pig (i.e. an inability to apply lower-order thinking, to rely on existing knowledge and the previously, successfully applied strategy for handling enemy encounters). In Forum Post 23, the player refers to the behavioural differences between the Tesla Pig and previously encountered enemies and how this in turn led to having to respond differently to the Tesla Pig (i.e. having to create new knowledge and apply new strategies, requiring higher-order thinking).
The Tesla Pig may *initially* appear to have not provided significant *cognitive* engagement with the game, instead enhancing the game’s *performative* challenge and the player’s sense of fear. However, this may be due to how players choose to report their play experience. In each of the previous example comments, there is evidence of additional cognitive processes being engaged in response to the Tesla Pig enemy. Forum Post 21 reports having to develop a new strategy (i.e. rather than incrementally building upon an existing strategy) for handling the Tesla Pig enemy; Forum Post 22 reports identifying that hiding was not a viable strategy, which may have further developed into having to form a new strategy; Forum Post 23 once more reports having to identify and develop a new strategy. Thus, whilst the player comments primarily focus on this disruptive game feature creating increased fear-like effects on them, an underlying potential cause of this is the cognitive engagement associated with having to develop new strategies for the Tesla Pig enemy type, especially whilst under direct threat from the enemy. As with the enemy audio cues, further research using a different methodology would allow additional evidence to be gathered that may support this interpretation (Section 9.3).

The second intended disruptive impact of the Tesla Pig enemy was Action Plan Disruption of transludic semantic knowledge. This was intended to be caused by presenting players with the new type of late-game enemy in scenarios that suggested that enemy type to be the equivalent of a ‘boss enemy’, as per the established game trope. In identifying a ‘boss enemy’ it was expected that players would be likely to respond to it using strategies previously learned throughout the preceding game levels (i.e. as they may typically respond to ‘boss’ enemies in other games). Indeed, as demonstrated in the previous forum comment examples, players reported this type of experience; however these comments were only framed in terms of the player’s *intraludic* knowledge, not their transludic knowledge. Thus, it is not possible to suggest that this disruptive game feature successfully implemented transludic knowledge disruption.

### 8.10.2: Disruptive Game Features without a Correspondence between Intended and Reported Disruptive Impacts

The remaining (six) implemented disruptive game features did not achieve a correspondence between the intended disruptive impacts and the reported disruptive impacts. Four of these were reported on by players as having disruptive impacts that differed from those intended, while the other two were not reported on by players in a way that suggests any disruptive impact at all. As stated in the initial definition of disruptive game design (Section 1.3), disruption must be intentional in the game’s design to be considered disruptive, rather than an unintended outcome of other design decisions. Thus, the four disruptive game features that did not achieve the intended disruptive impact cannot be considered successful. However, they still present an opportunity to explore disruptive impacts on players and to explore possible causes for the lack of correspondence between design intent and reported player experience. The two disruptive game features not reported on at all, in terms of their disruptive impacts, cannot be considered
successes or failures given the lack of data. However, the other discussion around these features may reveal other useful insights into the player experience more broadly.

Additionally, players made suggestions about how they feel these game features could have been improved. Many of these suggestions are similar, if not identical, to a version of the game feature that was prototyped during the game’s development (i.e. during the development of the game as created) but eventually changed before the game was released (i.e. the game as published).

8.10.2.1: Wretches, Engineers, and Enemy-to-Enemy Interactions

While the Tesla Pig partially achieved its intended disruptive impacts (Section 8.10.1.1), the behaviour of the other enemy types (i.e. Wretches and Engineers), including the developed interactions between the two enemy types themselves (Section 7.5.6.3 and Section 7.5.6.4), did not produce evidence to suggest a notable disruptive effect on players. The majority of player comments framed the enemy agents as, primarily, a tool for creating fear in the game. However, there was a noticeable divide between how players reported on identical sections of the game.

The encounters with the pigmen. Oh gosh, what can I say. The pigmen weren't scary at all in my opinion. The AI [Artificial Intelligence] was bad and there were too few encounters and even those were very mild. Not horrifying. (Forum Post 24, 2013)

The pigs... Amnesia TDD [A:TDD] is known for being one of the scariest games ever made, people were expecting AMFP to be scary too. The pigs were not scary whatsoever. (Forum Post 25, 2013)

I wasn't very fond of the monsters they’ve created in this Amnesia. Walking pigs, electric pigs and electric lurker, what more is there to say? I even felt sorry for them when I was walking through the cells, seeing how they were playing or were chained in their tiny rooms. That is certainly not the impression you want to make from evil monsters that are hunting you through the night. (Forum Post 26, 2013)

Forum Post 26 specifically refers to “feeling sorry” for the enemies during the level in which they can be seen in various holding cells. This player further states that such a reaction “is certainly not the impression you want” to give of an enemy agent. However, this empathetic response from players was the intention behind the design of this particular section of the game. The aim was to strongly emphasise the game’s Lovecraftian horrific qualities rather than the ability of the game to instil more immediate feelings of fear or terror. It was intended for players to develop an understanding of the larger scale horror occurring around them throughout the game and how they, as the player-character, fitted into that horror, rather than for the players to be merely ‘fearful’ of the enemy agents.

Other players however commented on the same section of the game (i.e. the holding cells) as being particularly effective.

Probably my favourite part of A:AMFP was the character of the monsters. They weren’t mindless drones that just kill kill kill. They had personalities and some even were a bit funny (kind of out of place for an Amnesia game though). I’m
thinking of the part of the game when you are going through what looks to be their living quarters. It added a little more life and realism to these 'monsters' and gave a little more depth. Sure, some of the interactions could've been done better but I think the game greatly benefited from its inclusion overall. (Forum Post 27, 2013)

The monsters grew on me, at first I thought they were kind of goofy, but once the story was unveiled it's [sic] clear that they didn't need to be these terrifying creatures, there were wretches to be pitied more than feared, playing with their toys like the children they once were. (Forum Post 28, 2013)

Specifically, Forum Post 28 demonstrates the impact of the expectations and knowledge of the individual player on how they respond to a game. In relation to the enemy behaviour in A:AMFP, it is evident that there are two main player ‘types’ represented in the data; those that play to experience a more immediate, visceral, shock or ‘fear’-based form of horror (i.e. those that view the enemies as being ‘not scary enough’), and those that play to experience a slower, more cognitively engaging and less immediate type of horror (i.e. those that respond positively to feeling empathy with the enemies). This split can also be described as players that get a greater sense of engagement out of horror that happens to the player and players that get a greater sense of engagement out of horror that happens to others.

However, while the player responses to the enemy behaviour in terms of ‘horror’ have included both positive and negative comments, there is no evidence in the data that suggests ‘disruption-like’ experiences. The behaviour of the Wretches in the early stages of the game was not discussed by players and it was in these early stages that Encoding Disruption was intended to occur, with players being uncertain of the motivations and threat levels of the enemies. Recall Disruption of transludic semantic and episodic knowledge was also intended to be experienced in these early stages, based on the ambiguous enemy motivations driving their behaviour (i.e. often running away from the player rather than being aggressive). However, there were no instances in the data of players reporting experiences that appeared to demonstrate this. Forum Post 28 does state that the enemies “weren’t mindless drones that just kill kill kill”, suggesting that such behaviour may have been expected based on this player’s previous gaming experiences. However, this is not strongly supported in this case and is also not identified in any other player comments. It is possible that players may have interpreted the ambiguous motivation behind the enemy behaviour, especially in the early stages of the game, as being the result of poor game design or poorly written game code. Forum Post 24 states that the “AI was bad”, although this is not contextualised further with specific examples. The AI functioned as it was designed, however if a player was expecting the enemy agents to be openly and consistently aggressive, the designed behaviour may easily be interpreted as being poor design or implementation. This again reinforces the care needed during design of ‘subtle’ disruptive game features or ambiguous game stimuli, as was seen previously in relation to players not noticing any of the rule changes applied to the flashing lantern warning system (Section 8.10.1.2).
8.10.2.2: Handling the ‘Death’ of the Player-Character and Hiding Places

As described in Section 7.5.9 and Section 7.5.10, both of these disruptive game features were subject to significant modification during development. The implemented features were therefore not expected to result in ‘disruption-like’ experiences. This was the case, with players providing near consistent negative comments on both features that not only suggested a lack of disruptive impact but also, a fundamental lack of enjoyment of the features.

In relation to how A:AMFP handles the ‘death’ of the player-character, many players report that the game is made too easy in the scenarios in which ‘death’ results in the enemy agent being removed from the game environment upon player respawn. This reduced difficulty in turn had a negative impact on many players’ sense of immersion and agency within the game environment, with some players purposely allowing themselves to be ‘killed’ specifically so that the enemy agent would be removed.

Monsters still despawn on death. Yeah I know they're against trial and error and think that after you’ve experienced a scare once, it won't scare but annoy you, but at the same time it makes bum-rushing monsters and just dying THE tactic to beat the game, because this will despawn them and make it safe for you. [The] worst kind of this despawning happens when you fall into the water with Kaernks [an invisible monster that lives in water in the game] and if they kill you not only [do] they despawn but you are also magically teleported to the top of the ladder you had to grasp from the water [eliminating the need for the player to search for and locate the ladder themselves]. (Forum Post 29, 2013)

Different players will likely approach games with different attitudes and place varying degrees of importance on certain aspects of their play experience. Research into player types and player personalities (Bartle, 1996, Bateman and Boon, 2006, Bryant and Oliver, 2009, Hilgard, Engelhardt and Bartholow, 2013, Peever, Johnson and Gardner, 2012, Quick, Atkinson and Lin, 2012) provides significant evidence in support of this. Not all players will gain equal enjoyment or feel equally engaged by the same game, or features of a game, and it would be naive to attempt to create a game that appealed to all players, regardless of individual factors or variables. In the case of A:AMFP, the game is designed with a certain target audience in mind; namely, it is targeted primarily at players that play games to experience a world and a story within that world, with the ‘gameplay challenges’ taking a secondary role. Thus, it may be expected that players that play games primarily to be challenged through gameplay may not respond favourably to game features designed to keep the story moving forwards, rather than forcing a player to play sections multiple times.

Meades (2012) suggests that games may “demand certain behaviours and ideological positions from their players in order to enjoy the game”. This statement is in relation to multiplayer games requiring player-to-player communication, however it can be suggested to potentially apply to other game types as well. Some players may primarily enjoy combat-heavy, fast-paced games while other players may primarily enjoy slower-paced, story-focused games. The reasons that players of each of these game types choose to play them for will likely be significantly different and may be
rooted in their underlying attitudes towards ‘play’ and what purpose(s) their particular play activities seek to fulfil. However, in A:AMFP, even for players with an attitude that prioritises the ‘experience’ of the game over the ‘challenge’ of the game, the approach to handling the ‘death’ of the player-character implemented in the game as created is flawed.

For example, if a player is ‘killed’, they may not mind the lack of challenge presented by an enemy that is removed from the environment after the player respawns. However, they may instead feel significantly removed from the experience of playing the game, through diminished feelings of agency and immersion. The experience described in Forum Post 29 describes both a lack of challenge as well as being disengaged from the experience by being “magically teleported” to safety. It is evident in the data that not only did the eventual implementation choice for handling the ‘death’ of the player-character fail to provide a disruptive experience for players, but the indecision during development over how to implement it resulted in a game feature that was detrimental to the game experience for a majority of players.

This feature was one that a number of players made suggestions for improving. In the original game as designed (Section 7.4.7), upon ‘death’ players should have been dragged by the enemy agent to a ‘nest area’ and then would have had to solve an additional puzzle to return to the ‘main’ game environment. This was not implemented in the game as created. However, suggestions were made by players that this initial design may have satisfied:

That’s where I think TCR could have done something better. Make a real penalty for getting caught by an enemy. (Forum Post 30, 2013)

Some sort of death sequence would’ve been quite interesting, instead of the same-old scratch marks on the screen like what we got in the original. (Forum Post 31, 2013)

Being dragged to a ‘nest area’ by the enemy would have provided an interesting ‘death sequence’ as suggested in Forum Post 31 and having to complete additional challenges to escape would have added a more tangible penalty for getting caught as suggested in Forum Post 30. This could have then been further enhanced by ensuring that the majority of enemies did not disappear once the player respawned. The failure of this disruptive game feature can be attributed jointly to a failure in communication between TCR and FG and design failures within both the development and publishing teams resulting in a flawed feature being included in the final release of the game. The player suggestions do provide a basis however for potentially implementing a similar system as originally prototyped in a future project.

Hiding places can also be discussed briefly alongside the handling of the ‘death’ of the player character. The removal of the majority of hiding places from A:AMFP was once more a significant change from the designed hiding place system. Instead of including a number of different hiding places with varying levels of ‘safety’ from enemies, A:AMFP does not include anything that players may recognise as being a hiding place. While the functionality remains in the game code, from the
player perspective, hiding places are not present in the game. This absence has been criticised by many players stating that not having to hide from enemies reduces both challenge and fear during gameplay. As with the handling of player character ‘death’, the failure of this disruptive game feature can be attributed to failures in the development process related to communication between TCR and FG (Section 7.5.10).

8.10.2.3: Providing the Player with Hints

The implemented method of providing the player with game hints in the game as created went through a number of prototypes and was the basis of significant disagreement between TCR and FG (Section 7.5.8). This has resulted in an evident trend in the game as reported that suggests that many players found the game too easy.

The game was... easy. No pursuits, no puzzles, no need to think. The hardest part was figuring out the point-the-light-to-lock thingy [a puzzle in A:AMFP’s ‘Tunnels’ level]. The notes kinda gave away every solution to every problem. (Forum Post 32, 2013)

[. . .] I understand his point about "games should not be too frustrating", but this time it went too far. "Puzzles" and accompanying instructions + no physical threats reduce this game to the linear visual novella [. . .] (Forum Post 33, 2013)

In A:TDD you learned about the character and his actions through the memory flashbacks with voice acting. These were well done but in A:AMFP it’s all done with verbose diary entries which I couldn’t care about at all. After the first 5 or so you realise that he’s basically just writing down what you have to do to complete the next puzzle. (Forum Post 34, 2013)

While the aim of the hint system from its initial design (Section 7.4.6) was to provide Encoding Disruption based on ambiguous stimulus information (i.e. minimal hints specifically not provided through explicit channels such as written text), the numerous changes made to the hints during prototyping and development (Section 7.5.8) has resulted in no disruption-like experiences being reported by players in relation to them. In conjunction with the previously discussed failings regarding the game’s puzzle designs, the explicit, written hints provided to the player throughout the game (with no option to deactivate them) has resulted in players responding consistently negatively in the game as reported. In conjunction with the issues encountered with the handling of the ‘death’ of the player-character, this demonstrates the potentially significant problems caused through disagreements between the developer and the publisher. In both cases, compromises have been implemented into the game as created that do not meet the full aims of either TCR or FG (Appendix F) and these features have been received poorly by players.

8.10.2.4: Removal of the Item Inventory

The removal of the item inventory provided insight into the perceived differences created between the different modes of disruption. More specifically, the player comments on this disruptive game feature demonstrate that in some instances, experiencing Recall Disruption alone may cause player frustration.
The intention of the item inventory removal was to allow players to experience Action Plan Disruption of transludic semantic knowledge (Section 7.4.9). Without an item inventory to store important items, players would have to consider navigation and obstructions (such as open or closed doors) more carefully, as they would move slower and thus be more vulnerable whilst moving items around the game environments manually. The success of this disruptive game feature was linked to the design and development team’s ability to create varied, engaging puzzles that required the player to move objects in different, interesting ways. Furthermore, for players to engage with the additional cognitive challenges of navigation and planning their way past obstructions and enemy agents, the puzzles had to take place in areas where the player could be attacked, or at least felt that there was a potential for being attacked. However, failure to fulfil these requirements created a situation in which the item inventory removal was experienced by players solely as Recall Disruption of transludic procedural and transludic episodic knowledge. While some players provided positive comments on this experience, the common themes that were evident were feelings of confusion and of being 'underwhelmed'.

For example, a number of players commented on the positive impact this feature had on their sense of immersion in the game world.

Most of the things changed to Amnesia like the removal of an inventory is a decision I can understand. No inventory improves the immersion and it really was unnecessary most of the time [in A:TDD] anyway. (Forum Post 35, 2013)

Speaking of "immersion", having no inventory is more immersive than having one. I don’t see how a person can carry 4 cogwheels, [a] hammer, acid, and huge machine batteries in his pocket [a reference to the use of an item inventory in A:TDD]. (Forum Post 36, 2013)

Inventory, sanity, tinderboxes and oil removed [in comparison to A:TDD]. I'm not sure how I feel about those. It made the game more immersive when you didn't need to check your inventory every now and then to mess around with the items or check how much oil you have in your lantern. At the same time, they didn't seem to fully take advantage of having no inventory, making the puzzles feel lacklustre. (Forum Post 37, 2013)

While this positive impact on immersion may be perceived as a partial success, it is offset by the issue raised specifically in Forum Post 37. Through failures in the design and development process, the disruptive game feature of removing the item inventory was mostly unsupported by the other game components required to make it successful (i.e. interesting and varied environmental puzzles). This has resulted in a number of players focusing on the absence of the inventory in comparison to A:TDD (i.e. Recall Disruption) rather than the intended changes in player behaviour that should have stemmed from its absence (i.e. Action Plan Disruption).

This is not to suggest that if players experience Recall Disruption they will always focus specifically on the change itself rather than on the consequences of the change. For example, 'set dressing' manipulation and the Pig Mask motif (Section 8.10.1.1.1) showed Recall Disruption was reported along with predominantly positive player comments. However, in the instances of disruptive game
features that successfully provided Recall Disruption, the features themselves were more ‘elements of the game story’ rather than ‘elements of the game mechanics’. Players may be more able to engage with these story elements and the associated disruption whilst continuing to play the game and make progress, because the story elements were not a critical part of that progress.

However, if the Recall Disruption is elicited by a game feature that has a more direct impact on the process of play itself and the ability of the player to make progress (i.e. the item inventory), the player experience appears to be different. The item inventory and the puzzles associated with it are being directly engaged with by the player and will prevent progress if not understood, along with feelings of confusion and frustration. In A:AMFP, much of the data from the game as reported is supportive of the idea of removing the inventory but consistently critical of the game’s simplistic puzzles. This is therefore indicative of a failing in the design and development stages of the game. It was understood during the game’s design (Section 7.5.11) that creating interesting and varied puzzles without an inventory would be a difficult task. As is apparent from the game as reported it is a design challenge that TCR did not overcome. However, the positive comments made by players about the concept of the removal of the item inventory itself are encouraging, as it demonstrates that players are positively receptive to the disruption of an established game feature such as the item inventory. A similar approach could thus be applied in future projects but with a greater awareness and focus on the need for more engaging puzzle design.

At a disruptive game design theory level however, it may be possible that Recall Disruption is more suited to disruptive game features that are not primary game mechanics that the player is constantly interacting with.

8.10.3: Potential Association between Modes of Disruption and Disruptive Game Feature ‘Primacy’

The issues associated with the item inventory removal suggest a broader question relating to the specific implementation possibilities for each of Encoding Disruption, Recall Disruption and Action Plan Disruption. That is, whether or not there an association between the modes of disruption and the ‘primacy’ of disruptive game features that can be implemented using them. In this context, the term ‘primacy’ is co-opted from existing games literature which frequently utilises terminology such as ‘core mechanics’, ‘primary mechanics’, and ‘secondary mechanics’ (Järvinen, 2008, Salen and Zimmerman, 2004, Sicart, 2008). In relation to disruptive game design in particular, three terms will be used: primary game mechanics, secondary game mechanics, and story-based game features. Primary mechanics are defined here as mechanics necessary for players to interact with throughout the game in order to make progress (e.g. the lantern in A:AMFP). Secondary mechanics are necessary for players to interact with at specific points in the game in order to progress past that point, but are not consistently interacted with throughout the game (e.g. the Tesla Pig in A:AMFP). Story-based game features may be any element of the game’s world, environment, entities, or story that the player encounters. These features may not be directly interacted with by players, thus they are features and not mechanics.
<table>
<thead>
<tr>
<th>Encoding Disruption</th>
<th>Secondary Game Mechanics</th>
<th>Story-based Game Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weak Association</td>
<td>Moderate Association</td>
<td>Strong Association</td>
</tr>
<tr>
<td>(e.g. Flashing Lantern Warning System)</td>
<td>(no successful example in A:AMFP)</td>
<td>(e.g. Pig Mask motif)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Recall Disruption</th>
<th>Secondary Game Mechanics</th>
<th>Story-based Game Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moderate Association</td>
<td>Strong Association</td>
<td>Moderate Association</td>
</tr>
<tr>
<td>(e.g. Enemy Audio Cues)</td>
<td>(no successful example in A:AMFP)</td>
<td>(e.g. manipulation of ‘set dressing’)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Action Plan Disruption</th>
<th>Secondary Game Mechanics</th>
<th>Story-based Game Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strong Association</td>
<td>Moderate Association</td>
<td>Weak Association</td>
</tr>
<tr>
<td>(e.g. Enemy Audio Cues)</td>
<td>(e.g. Tesla Pig)</td>
<td>(no successful example in A:AMFP)</td>
</tr>
</tbody>
</table>

Table 8.4: Approximation of associations between modes of disruption and disruptive game mechanic/feature primacy, with example ‘successful’ disruptive game features from A:AMFP.

With the degrees of ‘primacy’ identified, it is possible to cross-reference the primacy of A:AMFP’s disruptive game features to the types of disruption they have successfully elicited, according to the game as reported. This cross-referencing can then suggest an association matrix (Table 8.4).

The story-based disruptive game features of the non-Euclidean space, ‘set dressing’ manipulation and Pig Mask motif each elicited Encoding Disruption, Recall Disruption, or both. The Tesla Pig enemy (i.e. a secondary game mechanic) elicited Action Plan Disruption. The enemy audio cues (i.e. a primary game mechanic) elicited both Recall Disruption and Action Plan Disruption. Each of these features demonstrated correspondence between intended and reported disruptive impacts as well as being responded to with typically positive comments from players. This supports a possible association between Encoding Disruption and story-based features, and Action Plan Disruption and primary game mechanics, with Recall Disruption being primarily associated with secondary game mechanics.

The association matrix (Table 8.4) can thus be viewed in terms of strong, moderate, and weak association based on the available data from this study. However, due to the limited evidence available in the dataset and the limited number of successful disruptive game features, it is not possible to support these associations further at this stage. Moreover, the notion of primary and secondary game mechanics has not been considered previously in the current research and thus, would require more rigorous definition to form the basis for a more thorough investigation. Nevertheless, the available evidence provides a basis for further exploratory research (Section 9.3) that may more closely consider the effect(s) of implementing different modes of disruption via disruptive game mechanics/features of varying primacy. This would be a logical next step in further refining the disruptive game design philosophy and framework as it may allow designers to streamline their design and development process by providing more specific design guidance.
8.10.4: Concept Mapping and Further Themes Identified in the Player Discussion Data

Beyond the discussion of findings immediately relevant to specific disruptive game features within AAMFP (Section 8.10.1, Section 8.10.2, and Section 8.10.3), a number of further themes in the player discussion data emerged during the data analysis process that are relevant to different aspects of the current research and the theoretical underpinning of the disruptive game design philosophy. These findings also provide information that helps to contextualise disruptive game design in the broader field of games research and may provide themes for research, identified through a process of concept mapping.

![Figure 8.4: Section of one theme map focusing on the positive (green) discussion topics and negative (red) discussion topics grouped under the broad theme, Working to Find Meaning(s) in the Game.](image-url)

---

**Figure 8.4:** Section of one theme map focusing on the positive (green) discussion topics and negative (red) discussion topics grouped under the broad theme, Working to Find Meaning(s) in the Game.
This concept mapping process, whereby coded data is grouped into higher level discussion themes, was supported by the concept map creation toolset, CMap Tools. A series of ‘theme maps’ were generated, each one focusing on different types of theme, and these maps then provided the basis for discussion topics in this section, for example, Figure 8.4. Appendix K presents each map created.

8.10.4.1: Game Trailers, the Game as Expected, and the Game as Played

In Section 7.2.4 when considering the contextualising factors for A:AMFP prior to design and development commencing, the potential effect of the game’s genre label on the game as expected was noted. Specifically identified was the potential for players to come to A:AMFP expecting a ‘survival-horror’ experience rather than a ‘horror-adventure’ experience. This mismatch between the game as expected and the game as played was noted as potentially problematic. However, what was not considered at this stage was how the game may be marketed to players. This oversight may have eventually led to negative player responses to the game as played. The forum discussion data contains numerous examples of players citing the apparent differences between what the game’s marketing (specifically, its two trailers (Frictional Games, 2012a, 2012b)) led them to expect (i.e. the game as expected) and what they eventually played (i.e. the game as played). These examples from the data range from balanced discussion and debate about the contents of the game’s trailers, through to accusations of purposeful ‘bait and switch’ marketing tactics.

Can anyone tell me where in the game this [A:TDD]-esque moment (00:33) was? [Referring to a timestamp in A:AMFP’s first teaser trailer] Because I didn’t experience it. Not to say that a moment from a year ago should or should not have been cut from the game, but that advertisement made the game appear to be in the same horror vein of avoidance, hiding, and survival as [A:TDD], which was the element I missed the most in [A:AMFP]. (Forum Post 39, 2013)

The moment referred to in Forum Post 38 was indeed created as a bespoke event never intended to be incorporated into the final game in order to avoid spoiling a playable encounter. This player comments that the trailer made the game appear in the “same horror vein of avoidance, hiding, and survival” as A:TDD which they felt was absent from A:AMFP.

The content design of the game’s trailers provides an example of one of the problems associated with marketing a game designed using the disruptive game design philosophy (and indeed, one of the problems associated with marketing a ‘horror’ game more broadly); showing potential players enough content to allow them to understand and become interested in the game, without giving away key events from the actual game as played. Thus, the two sections of ‘gameplay footage’ from the two trailers were created specifically for them. They were intended to be representative of enemy encounters and gameplay in the full game, without spoiling any actual played encounters.

Further player comments do provide a less specifically targeted critique of the trailer content, discussing instead the stylistic and thematic qualities of the trailer and the type of gameplay indicated. Some comments also specifically note the marketing problem of not wanting to give away sections of the final game.
The issue is not that TCR didn't include some prop or event sequence shown in the trailers. The trailers that were shown marketed tropes and motifs of the original Amnesia title, which the final product did not include at all. (Forum Post 39, 2013)

I just thought the cage scene [the gameplay sequence in the second trailer] was pretty cool. I personally was expecting more interactions with the monsters in that fashion [such as] having to hide in spaces to avoid them. There were some areas like this but not very many and none of them involved the bigger monsters. I understand the whole not wanting to give away the surprises but I agree, it seems a little shady on their end to remove scenes that portray the type of vibe that [A:TDD] had. It could be an example of a bait and switch, or it could honestly just be what Frictional did with [A:TDD] and the levels that were never in the game. But with the game having a completely different vibe than what those scenes imply it feels as if they did it on purpose. (Forum Post 40, 2013)

While Forum Post 39 does not expand on specifically what “tropes and motifs of the original Amnesia title” are being referred to, Forum Post 40 refers specifically to the gameplay actions of hiding and avoiding enemies. However, even following this, Forum Post 40 still appears to suggest deliberate ‘false marketing’ on the part of TCR and FG.

In retrospect, it is possible that this problem could have been mitigated or entirely avoided through releasing further trailers or media nearer to the game’s final release to more accurately reflect the final product. Both trailers were released in 2012 (the second trailer being released in October 2012), thus leaving a gap of eleven months between the most recent trailer and the game’s release. However, given that trailer content was created specifically for trailers and not simply pulled from the game itself, the resources required (e.g. personnel, time, budget) to create new trailers was significant. With a small development team, any time spent on marketing material was time that was not being spent on the game itself.

The trailer definitely made the game seem scarier than it was, but I took that with a grain of salt, so I didn’t really feel betrayed. Making a trailer scary is easy. Making a game scary is much harder. When I saw the [A:AMFP] trailer, I didn’t think "That settles it. It's good!" I thought "I hope it's good." It wasn’t overtly [sic] misleading, and it’s on us to manage our expectations when we consume promotional media like that. (Forum Post 41, 2013)

Frictional Games and The Chinese Room are trying to actually sell their game so they can earn money for their work and continue making games, so I don’t think that "a dark kind of horrific tale, maybe not the most terrifying thing ever, but a good one if you’re into that kind of stuff" would have been a good catch-phrase [. . . ] It’s like when they sell a horror movie as “the most terrifying movie ever". We all know it’s not. I’m not saying that you’re wrong for not enjoying the game, that’s totally fine by me, and I actually understand why. I’m just saying that you can’t blame FG and TCR for that. They said it was going to be different. It is. Like it or not. (Forum Post 42, 2013)

Referring specifically to “expectations”, Forum Post 41 notes that the players themselves have some control in how they consume and interpret promotional media. This is supported by Forum Post 42 that argues that marketing material is, of course, trying to sell the product to customers and should be interpreted as such. However, even with this in mind, a design and development
team have a responsibility to produce material that supports potential players in creating an appropriate game as expected. This is not only to mitigate any risk of receiving claims of ‘false marketing’ or ‘bait and switch’ tactics but also, to ensure that when players eventually play the game (i.e. the game as played), it is the qualities of that experience that are engaged with first and foremost. A significant section of the A:AMFP discussion forums was filled with lengthy discussion and argument equating to the game as expected not matching the game as played. This discussion, while useful in the context of this research for demonstrating the importance of considering both the game as expected and the game as played, may have potentially shifted focus away from the primary research focus. That is, the actual disruptive game features and their effect on the player experience.

The marketing choices made for A:AMFP, including trailers and interviews with media outlets, were made with the primary aim of portraying an accurate ‘flavour’ of the game without giving away specific parts of the game itself. However, this has demonstrably failed with respect to, at least, a subset of players on the game’s discussion forum.

8.10.4.2: The ‘Amnesia Experience’ and Interludic Knowledge

Closely related to the discussion of the game’s marketing and the difference between the game as expected and the game as played is the frequently discussed notion of the ‘Amnesia experience’.

Removing much of the game mechanics and possible interaction in A:AMFP killed the "Amnesia experience". (Forum Post 43, 2013)

TCR did a fantastic job [in my opinion], I just want to make it scarier [in reference to creating a mod for A:AMFP] to appease the people who bought and expected an ‘Amnesia experience’. (Forum Post 44, 2013)

I’ve seldomly [sic] been so disappointed of [sic] a game. It’s mainly because the trailers promised a second A:TDD experience. Which made me pre-order this instantly. (Forum Post 45, 2013)

This selection of examples is of particular note as each appears to equate the ambiguous term ‘Amnesia experience’ (or A:TDD experience) with different aspects of play. Forum Post 45 refers directly to the gameplay shown in the trailers, suggesting a link between their interpretation of the A:TDD experience and hiding from enemies, as emphasised in both trailers. Forum Post 43 more specifically refers to the removal of mechanics in A:AMFP compared to A:TDD (e.g. the item inventory, the oil for refilling the lantern) as well as the removal of interaction possibilities (i.e. the interactive small physics objects in the game environments that were removed for optimisation and framerate purposes). Forum Post 44 however, appears to equate the ‘Amnesia experience’ simply to the game’s ‘scariness’, rather than any specific component of the game (e.g. mechanics) or specific gameplay actions a player may perform (e.g. hiding from enemies).

While describing a particular game ‘experience’ in reference to its branding or franchise may be common, it is interesting to see the degree to which players did this in the context of A:AMFP. Amnesia, as a brand, only consists of one other full commercial title (A:TDD). The number of
players equating this one game to a specific ‘Amnesia experience’ is significant in the dataset and
demonstrates the impact that A:TDD had on them. These players’ games as expected may thus have been heavily influenced by their knowledge of A:TDD and this may have further affected their responses to the game as played. It is also possible that this significant level of expectation may have influenced how A:AMFP marketing material, such as the trailers (Section 8.10.4.1), was interpreted. A number of other players made comments on the forum that addressed the notion of the ‘Amnesia experience’.

A lot of us that loved the game have dealt with the same arguments over and over, on this or other forums. It gets old, having to repeat and repeat oneself on the same points. I see a lot of ‘well it had the Amnesia title on it, it must be the same’, as if the Amnesia series is a long established series with a particular idiom. It isn’t. Let them try new ways of approaching horror instead of the same formulaic, warn out clichés. Personally, psychological horror appeals infinitely more to me than horror that is all shock and jump scares (I find it rather cheap). In the end, I’m probably not going to persuade anyone who has clearly made their mind up. I love the game, as well as its predecessor. (Forum Post 46, 2013)

Did it HAVE to be [an A:TDD] experience to be good? For me, the answer is ‘no.’ But of course, you’re entitled to feel otherwise. Amnesia is far too young a game to have an idiom of its own. The biggest factor for me was originality [sic], storytelling, and overall reaction (not necessarily fear, my gut still turns thinking about some of those notes...). The music still brings a tear to my eye. A:TDD found a winning formula, no doubt. But that doesn’t make it the only one that works. (Forum Post 47, 2013)

There is an identifiable split between the players that were disappointed by A:AMFP due to how much it differed from A:TDD and the players that appreciated the changes made and the new design direction A:AMFP attempted to go in, even if some elements of A:AMFP’s design were not implemented as well as they may have been. With the data available, it is not possible to further explore individual player responses. However, the frequency of comments that make explicit reference to comparisons to, and expectations stemming from, A:TDD suggest that the previously defined ludic knowledge types may require refinement. Specifically, transludic knowledge as a knowledge type that covers a player’s knowledge of all other games they have played previously may be too broad.

Players that have focused on comparisons to A:TDD are indeed making comparisons using transludic knowledge, although such comparisons are clearly more specifically focused than the broad transludic knowledge (i.e. across all games) label would suggest. Section 3.3.3 previously noted that the difference between knowledge of ‘other games’ and knowledge of ‘specific other games’ may be necessary to reflect in the ludic knowledge type definitions and Section 7.7.3 provided some supporting evidence for this from the design and development process undertaken for A:AMFP. The discussion data from this study provides further, player-based support for a split between these two types of transludic knowledge.
Thus, a new ludic knowledge type is suggested that contains a player’s knowledge of other specific games in the same series or franchise as the game being played. This new ludic knowledge type is termed interludic knowledge (i.e. between-game knowledge).

Figure 8.5: Interludic Knowledge as a subset of Transludic Knowledge.

For simplicity, this knowledge type can be conceived as a subset of knowledge within the container of transludic knowledge (Figure 8.5). There is not enough evidence from the current research to suggest a possible hierarchical ordering of transludic and interludic knowledge in terms of which knowledge type may have more influence on a player’s game as expected and reception of the game as played. The expectation, based on the previously defined concepts of encoding specificity and context dependent memory (Section 3.3.1), would be that interludic knowledge is referred to before more general transludic knowledge, as it may be more contextually relevant. Further research (Section 9.3) would be required to investigate this however as other factors may also be involved. For example, the serial position effect (Coleman, 2006, Ebbinghaus, 1913), also referred to as the primacy and recency memory biases, may mean that players would refer more immediately to games played recently, regardless of their transludic or interludic nature. Further exploration would be required to ascertain the extent to which such cognitive effects may influence player recall of different knowledge types during gameplay.

8.10.4.3: Player Attitudes towards Change and Experimentation in Game Design

The identified split between players responding positively and negatively to A:AMFP can be suggested to be based on those players’ underlying attitudes towards games as a medium more generally.

On one side, players appear to respond to A:AMFP positively because of how it changes the previous A:TDD design in different ways. For example:

> I honestly believe that TCR did the right choice by going for something different. As I said, there are hundreds of CS [player-created ‘custom stories’, or mods] to continue with [the A:TDD] vibe, some are even very good, so why would they do that, too? Yes, the game [A:AMFP] has its flaws, and some are disappointing, but I really feel that it was a good call. I prefer waiting for 2 years and ending up with
something new, even if it's flawed, than waiting 2 years to get the exact same thing. What's the point? But that's a personal opinion. (Forum Post 48, 2013)

I think some people need to realize is that this is not the same team who did A:TDD. This is a whole different team with a whole different interactive story philosophy and outlook of what is terrifying. So, I do think this is what the Amnesia "namebrand" needed: a new outlook instead of the same with new story or maybe mechanics. (Forum Post 49, 2013)

It's [A:AMFP] really-really-really good. In many ways it continues the evolution of the story-driven horror adventure that Frictional started with Amnesia and does things even smoother, better paced and more intuitive. It meant losing the most gamey elements with it - the inventory, the tinderboxes, the sanity and health meters. But they never felt truly important or necessary in [A:TDD] [. . .] sanity was, pretty much, pointless (as it did nothing if it ran out), and all the other inventory mechanics were tied with sanity anyway, so they felt pointless as well. They are gone and good riddance, I say. (Forum Post 50, 2013)

Each of these comments demonstrates underlying player attitudes towards games and game design. This provides evidence of players not only responding positively to change but also, demonstrating understanding of the development and marketing implications of implementing such change.

Further to these comments, one player specifically draws a comparison between “experimenting” in game design and the annual release cycle model that game series such as Call of Duty tend towards.

I think they [TCR and FG] try and innovate, do something different. I think a lot of the bad things people claim this game have are right, especially advertising badly, it's far from perfect but don't blame them for trying new things and experimenting with mechanics. If they stop doing that we might be getting a new game every year that is basically the same thing with a new story (sorry but COD [Call of Duty] jumps straight to mind) [. . .] don't bash them for trying, we will all benefit if they perfect a new style for horror games. (Forum Post 51, 2013)

There is a market for certain game franchises to be able to release new instalments with minor graphical upgrades or gameplay tweaks (i.e. what may be essentially “the exact same thing again”) and for those games to continue to sell well, such as Call of Duty and Battlefield. Such series serve a particular audience that seek a particular game experience. However, comments such as the examples above demonstrate that, in the case of the Amnesia franchise, a demographic of players exists that are more responsive to change and experimentation. The question is whether this demographic is more significant than the more negatively responsive demographic. Some of these negative comments can provide an insight into the specific player attitudes that give rise to them. For example:

You cannot really remove features from a game when you make a sequel and get away with it [. . .] even if the game would be better without it, you cannot remove features in sequels [. . .] I agree that a horror game without an inventory would be better than the one with [an inventory] [. . .] I really felt vulnerable carrying some valuable quest items [. . .] Also, all such abstractions such as "Endless bags" inventory [. . .] are quite immersion breaking [. . .] and therefore, if
you’d ask me which would be better for a horror game - with inventory or without, I’d say "without". However, when you’re doing a sequel, you just can’t do it! [. . .] since the original [A:TDD] had [an inventory], Machine for Pigs will feel incomplete without it for many customers. (Forum Post 52, 2013)

This is a particularly noteworthy example, as the player praises the removal of the item inventory from A:AMFP compared to A:TDD. However, this is countered by the player’s insistence that “even if a game would be better without” a particular feature, a sequel “cannot remove features”. This is the only instance in the dataset of a player making such explicit comments regarding removing features from sequels, although the majority of negative comments from players focus on features of A:TDD missing from A:AMFP. Thus, this attitude may not be reflected to this extent in other players. However, it does nevertheless suggest a design consideration for future design and development projects in whether it is more beneficial to retain a game feature in a game even if it was poorly received in previous games in the same series.

Player, or customer, ‘buy-in’ is necessary in order for changes within a game series to be well received. The positive comments selected above, along with others in the dataset, demonstrate that such players appreciate developers attempting to evolve a game’s design to provide unique and varied experiences. Likewise, players that enjoyed an initial experience (i.e. A:TDD) may have little or no interest in a series changing and may associate a series brand with a particular experience. Such players may go to other games for different experiences and thus, feel angry or betrayed when a series they feel has an ‘established idiom’ has significant changes made to it.

This poses a problem for designers in that it is important to understand the attitudes of players in a game’s target demographic. In reality however there is limited opportunity to acquire such understanding in a reliable way, especially when considering that what a player thinks they want may be different to what they actually want once they receive a new game. This problem is directly noted by a player on the forum, who states:

Kinda fun to read all those reviews sort of "disappointed", "where are my inventory and jump scaring monsters". From one side all those so called reviewers do want evolution. But at the same time they're disappointed by evolution of the game. It's just nonsense, unsolvable paradox. Give me evolved new game, but leave everything as I loved it. Everything the same, but a lot of new monsters and everything expanded, and polished, especially graphics, models and textures. (Forum Post 53, 2013)

The individual attitudes of players have a significant impact on how a game is responded to before, during, and after play. This is demonstrated in the mixed feedback received from players in response to A:AMFP. There is scope for future research to more thoroughly examine player attitudes towards aspects of the games industry, such as the place of ’experimentation’ in game design, and how those attitudes bias or influence responses to the game as played (in the context of different games).
8.11: Implications of Findings from the Game as Reported for Disruptive Game Design Theory

The analysis of the game as played, via the game as reported by players on the discussion forum, is the final stage of the Research through Design (RtD) methodology and means that every stage of the game space model (Figure 1.3) has been considered in relation to the design, development, publication, and play of a disruptive game. The findings from the phenomenological study can now be incorporated into the game space model, along with the other previously presented models (i.e. the Ludic Cognition Model (LCM) (Chapter 3), the Ludic Action Model (LAM) (Chapter 4), the DisDev model (Chapter 6), and the Integrated Model (Section 6.6)).

8.11.1: Implications for the Game Space Model

The game space model, previously proposed (Figure 1.3), shows the process from the design philosophy, through the design, development, and publication of a game, culminating in the game as played. This was modified in Section 6.5 to also include the prototyping stage of the development process. However, the game as reported is not included in these previous versions of the model. While the game as reported was expected to simply be a pragmatic means of accessing a reflective account of a player’s game as played, there is evidence from this study (Section 8.10.1.1), supported by the work of Ang, Zaphiris, and Wilson (2010), that engaging in reflection upon the play experience may be an intrinsic part of the enjoyment of a game for many players. Thus, including the creation of the game as reported within the game space model (Figure 8.6) is necessary to demonstrate this reflective practice by players and its potential importance in describing the complete progression of a game from a design philosophy to a player experience. It also highlights the difference between the ‘played’ experience and the ‘reflected upon’ experience, which is of importance for other gameplay studies beyond the context of the current research, that make use of individual player reports as a key source of data.
The *game as reported* may be created by players who are motivated enough by their experience of the *game as played* to share that experience with others through channels such as online discussion forums. The *game as reported* may be created through other means too, from casual conversation with other players through to more detailed, formal reports such as blog posts.

In the context of the *game space* model, the *game as reported* only includes reports that are motivated by, and created for, their *intrinsic value* to the player. Thus, reports such as critical reviews in magazines or reviews on gaming websites would not be included, as they are created through different motivations (e.g. as part of the writer’s paid employment).
Lastly, the game as reported within the game space model is situated across game-player space, game space, and world space. This demonstrates the function of the game as reported in translating a player’s personal experience of the game as played into a form that can be discussed with other players in ‘the world’.

8.11.2: Implications for the Ludic Cognition Model (LCM)

The LCM was previously proposed (Chapter 3) as a means of translating the general Systems Model of Cognition (Section 3.1 and Appendix A) into a pragmatic model able to be applied to cognition during gameplay. The ludic knowledge types (i.e. intraludic, transludic, and extraludic knowledge) were originally proposed as part of this model, based on the principles of encoding specificity and context dependent memory (Section 3.3.1). The LCM can now be updated to include the identification of interludic knowledge (Figure 8.7).

In addition, where the queries of long-term memory were previously labelled simply as ‘query progression’ in the order of intraludic, transludic, and extraludic knowledge, they are now explicitly labelled as ‘context-dependent query progression’. This labelling change takes into account the potential limits of the model structure, in that query progression may occur differently if influenced by other factors (e.g. the serial position effect, or primacy and recency memory biases).
Figure 8.7: The Ludic Cognition Model, with interludic knowledge included as a subset of transludic knowledge.
8.11.3: Implications for the Ludic Action Model (LAM)

The Ludic Action Model (LAM) requires only minimal modification to incorporate the new interludic knowledge type (Figure 8.8).

![Figure 8.8: The Ludic Action Model (LAM), with interludic knowledge included within long-term memory.](image)

This minor change does not impact the core functionality of the LAM, as it simply adds an additional long-term memory knowledge type that will be utilised within the cognitive processing cycle as required, alongside the other already existing knowledge types (i.e. intraludic, transludic, and extraludic).

8.11.4: Implications for the Disruptive Game Feature Design and Development (DisDev) Model

The findings of the phenomenological study have implications for the structure of Disruptive Game Feature Design and Development (DisDev) model. As with the previous models, interludic knowledge must be added as an available knowledge type to be disrupted. However, the DisDev model requires more substantive updates to account for other factors that may influence decision making during design and development, such as game feature/mechanic primacy (Section 8.10.13).
Figure 8.9: Disruptive Game Feature Design and Development (DisDev) model, accounting for interludic knowledge and game feature/mechanic primacy.
The key change to this version of the DisDev model (Figure 8.9) is the new ordering of the four property divisions within Stage 2 (i.e. Mode(s) of Disruption, Game Component(s), Target Knowledge Type(s) to Disrupt, and Mode(s) of Output).

![Diagram of property division ordering]

**Figure 8.10: Comparison of changes made to property division ordering in Stage 2 of the DisDev model.**

The previous property division ordering (Chapter 6) placed the player-based properties of a disruptive game feature below the game-based properties (Figure 8.10), implying that the player experience is secondary to the game components themselves. The new ordering enables a more balanced approach to design, with player-based properties informing decisions about game-based properties. This also takes into account the identified potential for certain modes of disruption to be more or less suited to different types of game feature/mechanic (Section 8.10.3), thus meaning that decisions about an appropriate game component to disrupt should be informed by the intended mode of disruption being designed for. Once the mode of disruption and game component to be disrupted have been identified, the knowledge type(s) to be disrupted can be identified, before finally ensuring an appropriate mode of output will be used to allow players to perceive the disruptive game feature during play.

Some minor graphical changes have also been made to this version of the DisDev model, with visual representations of the four different knowledge types added to enhance the model's usability for designers.

**8.11.5: Implications for the Integrated Model**

The Integrated Model can now also be updated to include the modifications to the previous models that form its component parts.
Figure 8.11: Integrated Model, Part 1.
The new Integrated Model (Figure 8.11 and Figure 8.12) incorporates the changes made to the DisDev model (Section 8.11.4), the inclusion of the new interludic knowledge type (Section 8.11.2 and Section 8.11.3), and the *game as reported* as a possible outcome of the performed actions of the player combined with their cognitive processing cycle, motivated by their experience of the *game as played* (Section 8.11.1).
8.12: Chapter Summary

The use of phenomenology, grounded in the contextual constructivist perspective and via a Template Analysis method applied to online player discussion forum data, has allowed analysis of player perspectives on a game artefact designed and developed using the disruptive game design philosophy. It is possible to state for example, that the disruptive game features that were reported on by players as having the designed-for disruptive effect were predominantly positively received, while those that failed to achieve the intended disruptive effect were predominantly negatively received. This suggests that designing with the aim of disrupting player knowledge can result in successful, positively received features in a commercial game title.

The study further suggests a possible pattern that may provide a basis for further research. Specifically, different modes of disruption may be more or less effective depending on the ‘primacy’ (i.e. story-based, primary, or secondary) of the game feature or mechanic providing the basis for the disruption.

Further findings aided in more broadly contextualising the exploration of the disruptive game design and specifically, the importance of the game as expected’s influence on the game as played. Furthermore, the frequent references by players specifically to the previous Amnesia game, A:TDD provided evidence to support the concept of a fourth ludic knowledge type within transludic knowledge, in the form of interludic (between-game) knowledge. However, with four different ludic knowledge types potentially being used by players, a question can be raised in relation to the general applicability of context-dependent recall in describing the order in which those knowledge types are queried. Further research may explore the potential impact of other cognitive factors (e.g. the serial position effect, or primacy and recency memory biases).

The study has lastly provided evidence to suggest that the game as reported may be a necessary component to include within the game space model. The creation of the game as reported by players may, in some cases, be an intrinsic part of the player’s enjoyment of the game itself.
Chapter 9:
Conclusions and Recommendations for Further Work

The current research has developed the disruptive game design philosophy as a new approach to
game design focusing on supporting a player’s cognitive engagement with a game (Section 1.3).
This design philosophy was proposed in order to fill a proposed gap within game space that other
design philosophies (identified as player-supportive design, counter-supportive design, and player-
unsupportive design) do not appear to fulfil. Two interlinked research questions were thus posed:

1. Within the constraints and limitations of game development space, is it possible to
   operationalise the disruptive game design philosophy within a commercial game as created
   so that it supports cognitive engagement during the game as played?
2. Do players report their experiences of a ‘disruptive’ game as played in a way that suggests
   cognitive engagement with the game and satiation of ‘high-level’ needs, and if so what are
   their attitudes towards this type of game experience?

Using a cognitive psychological foundation combined with modifications to existing game
interaction and game structure models, this disruptive approach suggests designing game features
that target different types of ludic knowledge (e.g. a player’s ‘cross-game’ knowledge, or their
knowledge of a particular game) for disruption at different stages of the knowledge encoding,
recall, and application (i.e. action) process. The theory developed (Chapter 1, Chapter 3, Chapter
4, Chapter 5, and Chapter 6) has then been applied, via a Research through Design (RtD)
methodology (Chapter 2), to the design, development, and publication of a commercial title,
Amnesia: A Machine for Pigs (Chapter 7). Online player discussion of the game, and of gameplay
experiences with it, has then formed the basis of a phenomenological investigation, via Template
Analysis, of the effectiveness of the disruptive game design philosophy (Chapter 8).

Through the definition of the disruptive game design philosophy in the contexts of the Ludic
Cognition Model (Chapter 3) and the Ludic Action Model (Chapter 4), followed by the design,
development and release of a commercial game utilising the philosophy (Amnesia: A Machine for
Pigs), the first question of operationalisation was addressed. Within the constraints of a
commercial game project, that included an external publisher with no involvement with the
current research, it was possible to design, develop, and publish a game underpinned by the
disruptive game design philosophy. Although there were instances in which issues stemming from
the commercial context negatively affected the application of the philosophy (e.g. disruptive game
features being heavily changed to the point where they were no longer disruptive), overall, these
instances were in the minority.

Then, following the release of the commercial game, the analysis of player discussion data has
addressed the second question by supporting an argument for the benefits to the player
experience of the disruptive game design philosophy. In resolving these questions, several contributions to knowledge were made.

9.1: Contributions to Knowledge

The contributions to knowledge are divided into thematic sections based on the different aspects of the research. Specifically, contributions are made to game design theory, game design and development practice, and game studies more broadly. Limitations of the research were however noted (Section 9.2) and thus potential avenues for future study are suggested (Section 9.3).

9.1.1: Contributions to the Field of Game Studies and the New Model of Game Space

![Image of the game space model.](Figure 9.1: The game space model.)
A model of game space was developed (Figure 9.1), throughout the research, demonstrating the progression of an individual game from its underpinning design philosophy, through the game as designed, the game as created (via prototyping), the game as published, the game as played, and concluding with a game as reported. The model itself constitutes the first contribution to knowledge as it provides a new way of visualising the interlinking stages of game creation and play. The detailed analysis and discussion of each of the individual stages (Chapter 1, Chapter 3, Chapter 4, Chapter 5, and Chapter 6) provide further separate contributions to knowledge. These provide a basis for guiding future research and development addressing each stage of the model. The specific contributions to knowledge stemming from each stage of the model are defined in detail in the following sections (Section 9.1.2 to 9.1.11).

The concept of the individual player’s game as expected is a key component of the disruptive game design philosophy and discussion of how this may be influenced by a range of external factors outside of the designer’s control (e.g. through a player’s interactions with a game’s production materials, marketing materials, and various elements of games culture more broadly) constitutes a further contribution to knowledge. The definition of these external factors provides a basis which future research may seek to expand (e.g. by identifying further external factors), or may seek to empirically investigate in more depth (e.g. by attempting to isolate and investigate the influence of specific, individual external factors on a player’s game as expected).

9.1.2: Contributions to Game Design Theory and the New Definition of Disruptive Game Design

Three design philosophies were initially defined (Section 1.2.3.1) in relation to the support of different player needs (Maslow, 1943) and support of player cognitive engagement during gameplay. They were made following the definition of a design philosophy for this thesis as being the underlying principles that drive the design of a game in a particular direction so as to fulfil the aim of the designer, or team, for the intended impact of the game on the player (Section 1.2.1). Each of these definitions represents new knowledge within the field of game studies:

- Player-supportive design was defined as a design approach that creates game features to satiate a player’s need for achievement and mastery through performative skill development. This design approach is the most comparable to what has been more broadly defined as ‘conservative’ game design.
- Counter-supportive design was defined from the basis of ‘abusive design’ (Wilson and Sicart, 2009, 2010) as a design approach that creates game features that actively seek to prevent the satiation of a player’s low-level, basic needs (e.g. physical comfort and safety).
- Player-unsupportive design was defined as a design approach that simultaneously removes support for satiating needs for performative achievement and mastery whilst not actively preventing the satiation of low-level needs.
Following the definitions of these design philosophies, disruptive game design was proposed (Section 1.3) as a new design philosophy in order to fill a proposed gap within game space. The disruptive game design philosophy was hence defined as a design approach that creates game features that actively ‘disrupt’ a player’s established knowledge about, and understanding of, a game. This does not prevent satiation of low-level needs whilst simultaneously seeking to increase the degree of satiation of high-level needs for problem-solving, creativity, spontaneity, and cognitive engagement. This may be achieved by, for example, changing how an established game rule or game mechanic functions at different points throughout a game, requiring players to engage in re-learning of game-based knowledge. Performative achievement and mastery based on previous knowledge is thus only experienced briefly by players before they must engage with the game at a cognitive level again in order to construct new knowledge about the game.

The definitions of each of these design philosophies above provide contributions to knowledge via a new perspective on approaches to game design, broadening the knowledge base in this field. They also provide a basis for potential further investigation that focuses on them individually in more detail (Section 9.3).

9.1.3: Contributions to Practice-Based Game Studies and a New Research through Design Methodology for the Field

Practice-based research within games and, more specifically, games research based in the broad ‘design research’ paradigm, were identified as established methodological approaches within game studies (Section 2.2 and Section 2.3). However, the application of these methodologies directly to a game artefact intended as a commercial product from the outset, rather than a pure research artefact, was found to not be represented within the existing literature.

![Figure 9.2: Research through Design (RtD) process model for game studies.](image)
As a means of addressing this absence, a new methodological model for an iterative, stepwise, Research through Design (RtD) process was proposed that was specifically contextualised within commercial game development (Section 2.4) (Figure 9.2). This model draws on concepts from Design Research more broadly (primarily from Vaishnavi and Kuechler’s (2004) design research process model), with new model components proposed to better suit the specific context of games research. These new model components build on those provided by Vaishnavi and Kuechler and include:

- Addition of extra process steps that apply to a (commercial) game project specifically (i.e. awareness of commercial context and publication).
- Game-specific terminology for each stage’s output as well as defined links to the different stages of the game space model (i.e. the design documentation as the game as designed, the game artefact as the game as created, the game product as the game as published, and the documented player feedback as the game as played).
- The addition of a component to the model that identifies the ‘focus’ of each stage. This visualises the transition from the ‘research’ aims of a project into the ‘development’ or ‘practice-based’ elements of the project and then back to the ‘research’-based evaluation and conclusion.

The proposed methodological model for Research through Design within game studies constitutes a contribution to knowledge in terms of empirical game studies theory, providing a new methodological approach tailored for use in the field. It provides an RtD methodology tailored for games that has been demonstrated to function within a commercial context. This supports researchers conducting practice-based research and thus could aid in expanding the knowledge base in this particular area of game studies (Section 9.3). The model can be applied to both commercial game projects that have a research element, as well as pure research-based game projects, although the latter may remove the awareness of commercial context stage.

9.1.4: Contributions to the Understanding of Player Cognition during Gameplay and a New Ludic Cognition Model

A Systems Model of Cognition was proposed (Section 3.1), building on the wide body of existing cognitive psychology literature (Appendix A). With this as a basis, a new model of Ludic Cognition was proposed (Section 3.2 to Section 3.5) that retains the components of memory and cognition established within the field of psychology, whilst creating a pragmatic, game-player-focused model that emphasises the components and processes most relevant to the play of digital games.
Figure 9.3: The Ludic Cognition Model (LCM).
Within this Ludic Cognition Model (Figure 9.3), the theories of working memory (Baddeley, 2007), multiple memory types (Tulving, 1985a), and encoding specificity (Tulving and Thomson, 1973) were integrated to propose game-specific ‘ludic knowledge types’: intraludic (i.e. within-game), transludic (i.e. across-games), interludic (i.e. between-games), and extraludic (i.e. outside-games) knowledge (Section 3.3). The definitions of these ludic knowledge types constitute new contributions to knowledge and provide a structure that may be used to influence a player’s understanding and enjoyment of a game.

The functionality of the encoding and recall process acting on the ludic knowledge types is suggested to operate within the principles of context-dependent memory and recall via encoding specificity. That is, knowledge is encoded along with contextual cues drawn from the environment or situation in which encoding takes place, while queries of long-term memory (i.e. recall) will be made in a context-dependent manner (i.e. from the most contextually relevant knowledge to the least contextually relevant).

The Ludic Cognition Model presents a new way of conceiving of the process of player cognition during gameplay, drawing together a substantive existing body of cognitive psychology literature into a single, pragmatic, game-centric model. The model is suggested to therefore be a contribution to theory, enabling new understanding and a new perspective on an existing subject of research. Additionally, the proposed ludic knowledge types constitute a separate contribution to knowledge, as they can be understood outside the context of the Ludic Cognition Model. Indeed, the ludic knowledge types have already been successfully presented (Howell, Stevens and Eyles, 2014) separately to the Ludic Cognition Model.

9.1.5: Contributions to the Understanding of Player Interaction with a Game and a New Ludic Action Model

The Ludic Cognition Model (Section 9.1.4) provided a basis for understanding player cognition within a ludic context but did not integrate specific game-based factors, such as the detailed process of how a player interacts with a game and its component systems and outputs. To rectify this, the Ludic Action Model (LAM) was created (Section 4.2 to Section 4.5), of which the Ludic Cognition Model constituted one component (i.e. the Cognitive Processing Cycle).
The Ludic Action Model (Figure 9.4) was constructed following an analysis of existing game interaction models (Heaton, 2006, Perron, 2006) and utilizes Perron's model as a foundation. However, a number of additions are made from this initial basis, providing contributions to knowledge in the understanding of player-game interaction.

The Ludic Cognition Model is embedded within the Ludic Action Model as the Cognitive Processing Cycle. This provides a new, detailed, way of conceiving of the cognitive processes involved in a player's decision making during play. The Cognitive Processing Cycle offers a greater degree of granularity in terms of player cognition when compared to the analyses of both Heaton's (2006) and Perron's (2006) models of gameplay (Section 4.1.1 and Section 4.1.2).

The Ludic Action Model is embedded within 'world space', meaning that the interaction between a player and a game is not assumed to only occur within game space but as a process that takes influences from, and has effects on, the real world. Thus, a player's input to a game also has an impact on the real world (e.g. a controller button is pressed, or the player feels the motion of their body when they provide motion-based input). The player's observation of these real-world effects may in turn influence their decision making in future (e.g. a player may think that a
particular motion input was physically uncomfortable and may decide to try a different, less exaggerated version of that input next time). This is an important contribution to knowledge in understanding player-game interaction as it demonstrates that a player’s physical experience of an action may influence their future input decisions without any additional output from the game being required. This has implications for how, in particular, designers may design input methods/devices, such as motion-based inputs.

Additionally, with regard to the player’s input to a game, the concept of a ‘practice cycle’ was added to the Ludic Action Model, in which players may perform ‘practice’ actions with an input device before performing it ‘live’ during gameplay (Section 4.2.2). As with the above addition of ‘real-world effects’ of game input, this practice cycle may be particularly relevant to further study of how players engage with complex controls in games, such as one-on-one fighting games, or how players engage with non-standard controls or input devices (e.g. motion controls, or emerging technologies, such as the Myo or Nod) (Section 9.3).

Thus, the new Ludic Action Model itself provides a contribution to theory, as it is usable both within the specific context of understanding player interaction with a ‘disruptive’ game, as well as more generally in the context of understanding player interaction with other games. In particular, the importance of the ‘practice cycle’ may be a fruitful avenue for further research into how players learn to engage with non-standard or emergent input devices and technologies (Section 9.3). Thus, this specific component of the Ludic Action Model and its associated discussion in the thesis (Section 4.2.2) constitutes a further contribution to knowledge.

9.1.6: Contributions to the Theory of Conceptual Game Structure and a New Conceptual Framework for Games

A conceptual framework for the game as created was proposed (Chapter 5), following an analysis of existing literature that conceptualises the structure of ‘game’ in different ways. In particular, the concept of ludodiegesis (Pinchbeck, 2009b), and within that, homodiegesis and heterodiegesis (Genette, 1980), was explored (Section 5.1) as a means of delineating the boundaries of a game object and its conceptual, structural components.
Figure 9.5: New conceptual framework of game components within the ludodiegetic structure.

Figure 9.6: Conceptual framework of game components with additional mapping of 'genre', 'mode', and 'milieu'.
In creating this conceptual framework, a number of existing works from game studies are combined in a new way (specifically, Pinchbeck’s (2007) ludodiegetic structure, Schell’s (2014) elemental tetrad of game components, Sicart’s (2008) definition of game mechanics, and King and Krzywinska’s (2002) definitions of ‘genre’, ‘mode’, and ‘milieu’) providing a framework that maps these existing works to each other. Thus, this new conceptual framework provides a contribution to knowledge in the form of a new, unified model of a number of existing concepts. Additionally, the timeliness of this particular contribution to knowledge is supported by other recent work. Firstly, the work of Ralph and Monu (2015), who have also proposed a move towards a unified theory of digital games by mapping together the work of Schell (2014) (i.e. the elemental tetrad) and the work of Hunicke, LeBlanc, and Zubek (2004) (i.e. the Mechanics Dynamics Aesthetics (MDA) framework). Secondly, the work of Koenitz et al. (2013) which presents a move towards a unified theory of interactive digital storytelling. As the field of enquiry matures, the many disparate, separate theories that emerge can begin to be combined, working towards a comprehensive understanding. Thus, any contribution to knowledge that presents ‘unified’ theories, or subsets thereof, is an important contribution.

This conceptual framework was used to propose a list of possible properties for disruptive game features (Section 5.4), utilising the disruptive game design philosophy as a foundation. This list was split into four property categories (Game Component, Mode of Output, Mode of Disruption and Knowledge Type to be Disrupted), with each category containing a number of different classes (e.g. Visual, Auditory, and Haptic within the Mode of Output category). This list of game feature properties provided an initial design tool for designing disruptive game features. It also provided a potential template for similar lists of game feature properties to be created to suit different design contexts or design philosophies (e.g. for counter-supportive, player-supportive, or player-unsupportive game features). This list, and the documentation of the process, thus contributes to knowledge. It demonstrates a key step in translating the disruptive game design philosophy into game features, as well as providing an explicit list of components that a designer could utilise in creating ‘disruptive’ games.

9.1.7: Contributions to the Theory and Practice of Game Design and Development, the New Disruptive Game Feature Design and Development Model

The game feature properties stemming from the conceptual framework (Section 9.1.6) formed the basis for the new Disruptive Game Feature Design and Development (DisDev) model.
Figure 9.7: Disruptive Game Feature Design and Development (DisDev) model.
The model (Figure 9.7) provided a practical design and development tool usable as a means of guiding the design and development of a game based on the disruptive game design philosophy. This tool was intended as a designer-facing tool that can be understood without significant background reading of its underpinning theory. Thus, this provides a contribution to both game design theory and also to practice. The model was demonstrated in principle through applying it to hypothetical design situations based on scenarios in existing games (Section 6.2), thus also demonstrating how the disruptive game design philosophy could be applied alongside other design philosophies if desired. This contributes to knowledge by demonstrating that, while this research has focused on applying the disruptive game design philosophy as the principle driver of design and development decisions, it is equally possible for ‘disruptive’ game features to be inserted into otherwise non-disruptive games to achieve different, localised, effects (e.g. changing how the rules of a single, specific, game feature functions in a game, such as the flame traps in *The Elder Scrolls V: Skyrim* (Section 6.2.2)).

While the DisDev model is intended for use in designing and creating ‘disruptive’ games, a similar model structure could be applied to games utilising different design philosophies. Of the four stages of the DisDev model (i.e. Assessment of Contextualising Factors, Game Feature Design, Game Feature Prototyping, and Full Development) only Stage 2 (Game Feature Design) would require modification to suit different design foci. The prototyping stage (Stage 3) reflects the process undertaken in this specific project and thus, may not apply to other projects that have different approaches to prototype creation. However, this stage could easily be modified to suit different development teams’ prototyping processes as necessary.

**9.1.8: Contributions to the Theory and Practice of Game Design and Development and the Integrated Model**

Following the proposal of the DisDev model along with the previously defined models (i.e. the *game space* model, the Ludic Cognition Model (LCM), the Ludic Action Model (LAM), and the conceptual framework of game components), an Integrated Model was proposed (Section 6.6) that shows the connectivity between them, within the container of the *game space* model.
Figure 9.8: The Integrated Model, Part 1.
The Integrated Model (Figure 9.8 and Figure 9.9) presents the lower-level processes that support the higher-level, conceptual game space model. This Integrated Model is tailored towards the specific requirements of the disruptive game design philosophy. However, the elements contained within it have been designed to be flexible and applicable to other game design contexts. For example, the Cognitive Processing Cycle within the LAM component of the Integrated Model...
could be used as a basis for research and/or design focusing on different elements of player cognition. Thus, the Integrated Model contributes to knowledge by demonstrating how separate components function together and how each can have an influence on the final game as played and game as reported.

9.1.9: Contributions to the Theory and Practice of Designing Games with the Disruptive Game Design Philosophy

Chapter 7 documented the contextualisation (Section 7.1 and Section 7.2), design (Section 7.3 and Section 7.4), development (Section 7.5), and publication (Section 7.6) of the first game to be created utilising the disruptive game design philosophy. Each stage of the DisDev model (Figure 9.7) was utilised and applied to a live commercial game project. A number of challenges were encountered during the design, development, and publication processes which had varying degrees of impact on the ability of the design and development team to fully implement all of the intended disruptive game features. These challenges were overcome whilst still maintaining many of the originally intended disruptive game features. Each decision to modify or remove a disruptive feature was documented and demonstrates the ability to practically apply the disruptive game design philosophy, within the constraints of commercial game development. The application of the DisDev model led to a number of conclusions (Section 7.7), each representing new knowledge relating to the use of the disruptive game design philosophy in a commercial context:

- In the context of a horror-adventure game designed, developed, and published in a collaborative arrangement between two companies, the disruptive game design philosophy and the design and development framework proposed in this thesis, has been successfully applied (Section 7.7.1). This demonstrates that the disruptive game design philosophy is practically usable in at least this instance. While it is not possible to generalise from this single case-study example, this case-study nevertheless provides initial evidence for the potential benefits of a ‘disruptive’ approach to game design and development.

- Designing specific modes of disruption (i.e. Encoding Disruption, Recall Disruption, and Action Plan Disruption) into specific game features frequently requires multiple modalities to be utilised in conjunction (Section 7.7.2). It can be suggested that it is challenging to attempt to only apply one specific type of disruption. For example, Recall Disruption is observation-based (i.e. based on a familiar game stimulus displaying unfamiliar properties or behaviours), while Action Plan Disruption is interaction-based (i.e. based on a familiar game stimulus responding in an unexpected way to a previously established player interaction with it). However, when Recall Disruption is designed to be triggered by game features requiring interaction, in many cases, an opportunity for Action Plan Disruption naturally follows (Section 7.7.2). Thus, it appears that these two modes of disruption may be closely linked and perhaps should be considered jointly by designers, rather than as completely separate modalities.
• The distribution of the different modes of disruption in the game is not evenly balanced (Section 7.7.2). This suggests that in this case, and potentially in other cases, different modes of disruption may be more or less suitable for different genres, modes, or milieus in games. This information provides a contribution to knowledge by demonstrating that the disruptive game design philosophy (and potentially, other design philosophies also) may require adaptation to apply to different game types. This could in turn further focus design and research efforts in future work.

The design and development documentation, produced alongside the game, represents a contribution to both theory and practice, providing a case-study that can be used as a point of reference for both academic and industry work in future. This is supported by additional material from the game creation process, including the post-mortem report (Howell, 2014) that has been independently published (Appendix F) as a separate contribution to practice as an industry-facing article. Lastly, the game artefact itself, Amnesia: A Machine for Pigs, represents a contribution to games culture in the form of an object that can be played, explored, and studied in its own right, potentially providing a basis for future academic research or a point of reference for future design and development projects.

9.1.10: Contributions to the Understanding of how Players Experience and Report on ‘Disruptive’ Games

The phenomenological study of online player discussion data (Chapter 8) provided a range of findings regarding how players experience and then report on the play of ‘disruptive’ games. Specifically, interpretation of the data (Section 8.10) has suggested the following insights, each constituting new contributions to knowledge:

• The data showed responses to the game tended to be split into two groups, with strongly positive or strongly negative responses. In many cases, these two groups of players cited identical game features as being causes for their positive or negative responses to the game. This provided evidence of the importance of a player’s individual expectations in influencing how they experience and report on the same game. This is something that should thus be a consideration in all other studies of gameplay and game experience, as well as being something that designers should be aware of in future projects.

• The data further showed that players that accept the themes and gameplay style of the game (i.e. what players in the data referred to as ‘buy-in’) may (although not always) provide more positive reports on their gameplay experiences. In this particular context, these players often also tended to note the importance of considering A:AMFP as a standalone game, rather than only as a sequel to A:TDD. The negative reports meanwhile tended to (although again, not always) come from players that compared the game directly to A:TDD or that did not appear to ‘buy-in’ to A:AMFP’s gameplay style. While this finding may appear obvious, it is specifically the concept of ‘buy-in’ that is of importance. If a game can achieve a certain level of thematic or contextual buy-in from players, they are
more likely to overlook, or to place less importance on, any weaker aspects of the game. Thus, this provides a potential basis for further research (Section 9.3) into factors that may influence this ‘buy-in’ from players. These factors are associated not only with the game itself, but also the developer, the marketing, and other publications around the game, and each contributes to a player’s game as expected.

- The significant number of players that compared their experience of A:AMFP exclusively to their prior experience of A:TDD provided evidence to suggest that a new ludic knowledge type – interludic knowledge – could be defined as a subtype of transludic knowledge. This new ludic knowledge type was added to the previous definitions of intraludic, transludic, and extraludic knowledge and has been reflected in the final versions of all of the models previously presented in this chapter.

- Similarly, players tend to describe experiences in the game as reported in terms of mechanics or features they are familiar with from other games, especially games in the same series or franchise. This can lead to misidentification or misunderstanding of mechanics/features in the current game. This provides a contribution to knowledge in that, especially in the case of game series, designers should consider the influence of features in previous games on how a player may experience and report on features in a new game. This is linked to the concept of interludic knowledge but in this case, it demonstrates that interludic knowledge has an impact on players not only during play but also, during the construction of a game as reported.

- It is possible that, alongside a player’s ability to recall knowledge and understanding from other games during gameplay (i.e. transludic knowledge), players may also be influenced by their previous feelings or responses towards similar game features in other games. Thus, a player may be predisposed to respond positively or negatively to a particular game feature based on how they have responded to similar features in other games previously. There is some limited evidence to support this from the current research, however this may provide a basis for future research (Section 9.3) to explore the impact of a player’s previous responses to game features in greater depth.

- There is evidence to suggest an ‘association matrix’ in which Encoding Disruption, Recall Disruption, and Action Plan Disruption each has a weak, moderate, or strong association with primary game mechanics, secondary game mechanics, or story-based game features. Of particular interest for understanding the potential effectiveness of the disruptive game design philosophy in different situations are the suggestions that:
  - Encoding Disruption has a strong association with story-based game features.
  - Recall Disruption has a strong association with secondary game mechanics.
  - Action Plan Disruption has a strong association with primary game mechanics.

Thus, the study of online player discussion data offers a contribution to knowledge in the form of an ‘association matrix’ that can be used to provide further structure and robustness to the disruptive game design philosophy. The study has also provided a
contribution to knowledge in the form of four distinct ludic knowledge types (i.e. Intraludic, Transludic, Interludic, and Extraludic knowledge).

- Player data also suggested that Encoding Disruption via ambiguous stimulus information may provide a catalyst for the enjoyable practice of 'reflective play'. When applied to story-based game features, ambiguous information may encourage players to discuss their play experiences with others, thus encouraging play to continue beyond the experience of the game as played.

- There is some evidence to suggest that players may become less likely to notice changes within a game (disruptive changes, or otherwise) the further they progress through the game, as they may expect a game to be stable and consistent and to signpost any such changes to them clearly. Once again, this provides a potential basis for future research to explore in greater depth (Section 9.3). This contributes to knowledge of how players engage with a game holistically, rather than at a purely 'mechanical' level.

- When designing disruptive game features, it may be necessary in future projects to consider the possible alternative interpretations or understandings that players may form of those features if they do not experience them as disruptive.

- Disruptive game features that are significantly compromised during development are likely to be best removed from the game, rather than being included in a 'cut-down' form. As demonstrated in the case of A:AMFP, the disruptive game features that were included in such a 'cut-down' format were poorly received by players across the forum study. This finding provides a contribution to design practice, demonstrating that in at least this particular case-study example, attempting to find compromises that suited two different companies with two different design aims failed to produce game features that were well received. This may have implications for how future collaborative design and development projects are managed.

This study also provides case study evidence of successful implementation (in terms of how players have responded to them) of disruptive game features targeting each of Encoding Disruption, Recall Disruption, and Action Plan Disruption. Hence, the study's primary contribution to knowledge is that the disruptive game design philosophy, defined within this thesis and applied to a commercial game design and development project, has provided players with identifiable gameplay experiences that have supported cognitive engagement and high-level need satiation. As may be expected with exploratory research, not all of the disruptive game features have been successfully implemented. However, as previously described in relation to the Research through Design (RtD) methodology (Section 2.3.2), the aim of such research is not to produce the best artefact to describe a design space; it is to produce an artefact to demonstrate the existence of the design space. The findings from the player discussion data are the final component, within the current research, which shows that this is the first game artefact to demonstrate this disruptive game design space exists.
9.1.11: Contributions to the Understanding of how a Game’s Marketing Material Influences the Game as Expected

As an additional finding from the online player discussion data, the study provides an example case-study of how mistakes in a game’s marketing approach and publicity media can lead to a distorted game as expected in the minds of players, in turn leading to negative comments and responses to the game itself. While this research has not focused on marketing processes in depth, there is evidence in the data that demonstrates the impact that A:AMFP’s marketing had on some players. This data and its analysis (Section 8.10.4.1) constitute a contribution to knowledge that may be beneficial to both theory and practice. Further research into the specific effects of marketing on the game as expected is possible, while the case study of the effects of A:AMFP’s marketing on eventual player responses may be of interest to industry practitioners.

9.1.12: Contributions to the Understanding of Methodological Approaches to Qualitative Player Studies

The phenomenological study, as the last stage of the Research through Design (RtD) process (Section 9.1.3), also provided a contribution to knowledge over and above the high-level methodology itself. Firstly, this study provides a significant, detailed example of the application of the Template Analysis method within game studies, providing new data and new analysis (Section 9.1.10). Secondly, the application of the method is itself a significant contribution to games research practice as it is an infrequently applied method within that field and this research has demonstrated its benefits in dealing with large datasets in particular. This may thus encourage future research to consider this method, especially in work that utilises large datasets, such as online discussion forums.

9.2: Limitations of the Research

This research has presented a new game design philosophy, supported with pragmatic models to guide the creation of games utilising its principles. This new design philosophy has been situated within a much larger game space that likely contains many other, as yet undiscovered, philosophies and approaches to designing games. Via a Research through Design (RtD) methodology, a playable, commercial game artefact has been created which places a ‘flag in the ground’ of ‘disruptive game design space’. This artefact is not expected to be the ‘best’ representation of a game designed utilising the disruptive game design philosophy, rather, it is intended to provide an initial basis from which further games may be developed, in turn continuing to refine the understanding of the design space itself. Indeed, the primary limitation of this research is its lack of generalizability beyond the contextual bounds of the design, development, and publication of A:AMFP. However, given the new area of enquiry that this research has focused on, it can be suggested that a methodology that focuses on a single, detailed case-study application of theory was the most appropriate method to employ. Indeed, in the exploration of human experience (e.g. reported experience during gameplay), truly generalizable theory has been suggested to be likely to be impossible to construct:
Predictive theories and universals cannot be found in the study of human affairs. Concrete, context-dependent knowledge is therefore more valuable than the vain search for predictive theories and universals. (Flyvbjerg, 2006, p.7)

Thus, to understand any aspect of player experience, case-study research is a vital tool for generating ‘concrete, context-dependent knowledge’. By continuing to explore the disruptive game design philosophy as it applies to other, different, case-study examples (i.e. different games), a greater body of knowledge can be developed consisting of other context-dependent exemplars of the philosophy’s application. As further case-study examples are created, the different structural components that have formed the theory and models presented in this thesis may then be able to be isolated, tested, and developed further.

For example, within the game space model, it is likely that the identified influences on the game as expected, from world space, are not comprehensive. In particular, the broad notion of games culture can likely be further unpacked beyond the individual elements already identified in this thesis (Section 1.2.2 and Section 6.4), such as genre labels/archetypes, series/franchise characteristics, user reviews, and spoilers. Indeed, over the five year course of this research, the visibility of YouTube (Google, 2006) personalities in the field of games media and criticism has increased substantially. This is supported by a sharp increase (Google, 2015) since 2009 in searches via Google for ‘Let’s Play’ videos, in which individuals record their gameplay and commentate it. Some ‘Let’s Players’ receive pre-release review copies of games from developers and publishers as a means of generating online discussion and interest in upcoming titles. This is one element of games culture that is not considered in the current game space model and thus, the model has limited scope in its current format.

Likewise, other shifts in games culture during the period of this research may provide previously unconsidered influences on the game as expected as well as potentially, direct influences on the game as played (which would require some restructuring of the game space model). For example, MacCallum-Stewart (2013) discusses the concept of the ‘fan-producers’, or “gamers who make videos, machinima [the use of real-time game engines to create cinematic works] and webcasts”. These types of gamers are involved in games culture at a different participatory level than, for example, the gamer that utilises games culture simply as a source of reviews and critical opinion on games. Other players may not only play games but also, engage in what Comunello and Mulargia (2015) refer to as ‘user-generated video gaming’; the use of a game’s editing tools to create and share new game content (e.g. as notably seen in the LittleBigPlanet series (Media Molecule, 2008-2014) and more recently in dedicated games such as Super Mario Maker (Nintendo EAD Group No. 4, 2015)). These ‘player-creators’ can be said to be involved at yet another participatory level. These greater degrees of participation in games culture may lead to different types of game as expected. They may also lead to different experiences of the game as played, depending on what those individual players particularly ‘value’. The fan-producer may focus on identifying elements of the game that would make good content for webcasts and videos (perhaps then experiencing the game in a different way to the ‘designer-intended’ experience). Likewise,
player-creators may play the game specifically to learn how the game itself was created in order to develop new ideas for their own, user-generated game content. There are therefore inevitable limitations to the game space model presented and substantial scope for further research that may expand and further define the individual stages of the model.

Games are an inherently complex medium and thus, any research approach exploring their design or indeed, their analysis, must be bounded in some way. In the case of exploring the impact of a design approach on the player experience, the current research (and any other similar research exploring the relationship between design and experience) is unavoidably bounded by the inability to obtain an unfiltered, ‘in-the-moment’ understanding of any individual player’s experience. Player experience must always be filtered in some way (e.g. the player’s own self-reporting, or an observer’s interpretations). This is an inherent limitation of player studies and indeed, any research into user experience more generally. However, it is nevertheless important to reiterate this limitation and its impact on what can be reliably ‘known’.

The definition of the disruptive game design philosophy itself has also been bounded through its use of specific underpinning theory. The philosophy has been proposed with respect to cognitive engagement and high-level need satiation and has been argued through the critical, but necessarily selective, use of cognitive psychological theory. It is possible that different psychological theory may offer different ways of conceiving of and defining ‘disruption’ and that these different ways may be more, or less, successful. This is an avenue for potential further work (Section 9.3). The disruptive game design philosophy has also been embedded within a cyclical Ludic Action Model that has been created with the aim of being applicable broadly to all digital games. However, the diversity and range of games currently available across all digital platforms means that there will inevitably be exceptions; there will be games that cannot be wholly described via this model.

Further refinements and improvements to the underpinning theory of the disruptive game design philosophy and the design and development models proposed alongside it, can inevitably be made. While the underpinning theory for the disruptive game design philosophy has been demonstrated to function as a whole in this research, this cannot be taken as evidence that every individual element of that theory is itself the most appropriate for its role. For example, altering the structure of the conceptual framework of games (Chapter 5) may in turn lead to a different set of disruptive game feature properties that allow more, or less, effective disruptive game features to be implemented. Therefore, the next logical step in the progression of this line of research is to explore the individual elements of the theory and models presented in this thesis in more depth.

9.3: Suggestions for Further Work

Ongoing work that investigates the individual theoretical elements in greater depth may be better carried out in a more focused, controlled, experimental context that allows the specific variables associated with each of the different elements to be more easily investigated. In the current research, ecological validity was considered of greater importance than reliability (repeatability),
as the aim was to explore the potential for real-world, commercial application. With that commercial application potential identified, to an extent, further work should seek to focus on enhancing the reliability of the theory, thus strengthening the support for the disruptive game design philosophy.

In addition to this primary suggestion for further work, a number of other potential avenues for further work exist, stemming from different components of this research (Table 9.1). Some of these build on what has been presented in the current research, while others branch out into different areas.

<table>
<thead>
<tr>
<th>Basis in this Research</th>
<th>Suggestion(s) for Further Research</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Model of Game Space</td>
<td>The game space model has scope to be expanded. For example, it is likely that the influences from world space on the game as expected presented in this thesis are not comprehensive. Further research should thus focus on developing deeper understanding of these world space influences and their role in creating the game as expected. Each of the embedded spaces within game space (i.e. game design space, game development space, game publication space, and game-player space) is related to different topic areas in game studies (i.e. design/philosophy, development/practice, publication/business, and gameplay/player studies) and thus, further research in these fields could continue to develop the game space model.</td>
</tr>
<tr>
<td>Theory and Application of the Disruptive Game Design Philosophy and the DisDev Model</td>
<td>The application of the disruptive game design philosophy to A:AMFP in this research, via the DisDev model, has demonstrated the philosophy’s applicability in an example case study. However, the underpinning theory has itself been modified following analysis of player discussion data (e.g. the introduction of interludic knowledge). Thus, the next step is to apply this newly refined philosophy to a new game design and development project, again utilising the DisDev model to guide the design and development process. To further the understanding of ‘disruption’ more, this new project should also differ from A:AMFP in terms of the game’s genre, mode, milieu, design/development process, and/or publication arrangements. This would thus provide evidence for whether the disruptive game design philosophy applies outside of the constraints of the case study presented in this thesis.</td>
</tr>
</tbody>
</table>

Continued on next page
## Basis in this Research

### Context-Dependency in the Recall of Ludic Knowledge

The Ludic Cognition Model has argued that the order in which a player queries different ludic knowledge types while constructing understanding of in-game stimuli is based on context-dependency and encoding specificity. Thus, players will refer first to knowledge constructed during play of the current game and then refer to less and less contextually relevant knowledge. However, other psychological factors cannot be ruled out of this process, such as the serial position effect, or primacy and recency memory biases (Coleman, 2006, Ebbinghaus, 1913). While these were outside of the scope of this research, a useful next step in further work would be to explore the possible influence of these other psychological factors further. This in turn may provide new data that can enhance understanding of the role of the game as expected and how it may be constructed by different players.

### The Practice Cycle within the Ludic Action Model

While the Ludic Action Model primarily describes how a player interacts with a *game object* and how that *game object* then responds to player input, the model also includes the *practice cycle*, identifying actions that a player may take before directly interacting with a game (e.g. practicing complex input sequences, or non-standard input actions, such as motion controls). These practice actions can be considered similar to a ‘practice swing’ in golf.

There is scope for further research to explore how players make use of ‘practice actions’ in relation to, in particular, non-standard input devices or non-standard actions. Such research may be of particular relevance as new hardware becomes increasingly available to consumers (e.g. the Oculus Rift (Oculus VR LLC, 2015), the Razer Hydra (Razer Inc., 2015), the Leap Motion, the Myo armband, and the Nod ring).

### The DisDev Model

The DisDev model was created in this thesis with the primary purpose of structuring the design, prototyping, and development of disruptive game features. However, the model structure is flexible and has potential for adaptation to aid in the design and development of other (‘non-disruptive’) games. Doing so may identify design and development considerations not originally identified in this thesis. Such findings may in turn facilitate the future refinement and understanding of the design and development of ‘disruptive’ games.

*Continued on next page*
<table>
<thead>
<tr>
<th>Basis in this Research</th>
<th>Suggestion(s) for Further Research</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analysis of Player Discussion Data</td>
<td>A key finding from the phenomenological study was the association matrix demonstrating associations of varying strengths between the modes of disruption and the ‘primacy’ of game mechanics/features. Future research should focus on the application of the different modes of disruption to primary and secondary game mechanics and story-based game features to test the accuracy of this proposed association matrix.</td>
</tr>
<tr>
<td>Initial Game Feature Perception</td>
<td>The analysis of player discussion data in relation to the ‘flashing lantern warning system’ and ‘enemy audio cues’ in <em>A:AMFP</em> found that some players appear to apply their initial perception of a disruptive game feature to their experience of the game as a whole. This may suggest that players expect a game to be stable and to signpost any changes that occur during gameplay (e.g. changes to a game’s properties that make the player’s existing knowledge inaccurate). Thus, they become less likely to notice changes as the game progresses because they do not expect changes to occur without them being made explicit. It is also possible that players may simply be less likely to expect the fundamental structure or functionality of primary game mechanics to change without clear signposting and thus, do not identify changes made without such signposting. Conversely, players may be more likely to expect story-based game features to change as a game progresses and thus, may be more likely to notice even minor changes to such features (e.g. the Pig Mask Motif). This would be an important avenue for further research as it may have a significant impact on how and when disruptive game features are most appropriately used in games (e.g. earlier or later in a game, or applied to larger or smaller game features).</td>
</tr>
<tr>
<td>Basis in this Research</td>
<td>Suggestion(s) for Further Research</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-----------------------------------</td>
</tr>
<tr>
<td>Cognitive and Emotional Linking</td>
<td>The interpretation of player descriptions of their encounters with the Tesla Pig enemies in <em>A:AMFP</em> suggested that the increased sense of ‘fear’ players cited may have been caused by the need to identify and use a new strategy to handle this new enemy type, whilst simultaneously having to track a fast-moving entity in the game world. That is, the increased ‘fear’ may have been directly linked to the required cognitive engagement of the player. Further research may explore this possible relationship further, potentially from the perspective of cognitive load theory (Sweller, 1988). This may demonstrate benefits of placing players under increased cognitive load in horror games as a means of inducing short-term, heightened feelings of ‘fear’, ‘anxiety’, or ‘panic’.</td>
</tr>
<tr>
<td>Potential Sadistic Vs. Masochistic Traits in Videogame Players</td>
<td>Player responses to the Wretches and Engineers in <em>A:AMFP</em> demonstrated a divide between players that get greater engagement from horror that happens to <em>the player</em>, and players that get greater engagement from horror that happens to <em>others</em>. Through a psychological lens, this divide may be indicative of underlying sadistic (i.e. ‘horror’ to <em>others</em>) or masochistic (i.e. ‘horror’ to <em>the self</em>) tendencies. Pinchbeck (2009a) and Wilson (2010) both discuss the notion of players ‘submitting’ to a manipulating, controlling game system. Indeed, many games, especially those with extreme levels of difficulty, could be considered as ‘masochistic’ entertainment as players gain pleasure from their successes despite having to firstly endure high degrees of repetition and failure. Further research could explore potential links between player preferences for highly challenging/less challenging games and preferences for types of horror (i.e. horror to <em>others</em> or to <em>the self</em>), as a means of mapping sadistic or masochistic tendencies in players. It is an interesting consideration that player personality types with respect to sadistic or masochistic traits, may have an association with the types of games players choose to engage with. If such an association were to be identified, it may provide useful data that can contribute to the ongoing debates around media violence and horror more broadly.</td>
</tr>
</tbody>
</table>

*Continued on next page*
<table>
<thead>
<tr>
<th>Basis in this Research</th>
<th>Suggestion(s) for Further Research</th>
</tr>
</thead>
</table>
| Player 'Buy-In' Influences the Game as Played and the Game as Reported | Whether or not a player 'buys-in' to a game's themes, design philosophy, or gameplay style has been demonstrated to have an impact on how those players then report on their game experiences. However, this research has demonstrated some instances of players referring to non-game factors that have influenced their level of 'buy-in'; primarily in this case, their knowledge of the developer and publisher and how the two companies presented themselves publicly during the game's development.

Further research may explore the range of factors (both game-based and non-game-based) that may influence the degree of player 'buy-in' as this has been demonstrated to be an important component in forming a player's positive or negative play experience. |

| Past Responses to a Game Feature Influences Response to Similar Features in Other Games | The Ludic Cognition Model and Ludic Action Model were constructed on the basis that different types of knowledge held by a player may influence how they interpret a game's features and how they may then make decisions during gameplay. However, what was not considered was the possible influence of a player’s previous responses (i.e. positive or negative) to features in other games on their responses to a game feature in a new game.

For example, a player that did not enjoy a ‘flashing lantern’ mechanic in a previous game may be predisposed to also not enjoying a similar feature in a different game, even if the two features are functionally very different.

Further research may seek to explore in more depth (perhaps by way of a longitudinal study of multiple players' gaming habits and opinions on different features in the games that they play) the degree to which prior responses influence future responses. |

*Continued on next page*
### Table 9.1: Suggestions for further research.

<table>
<thead>
<tr>
<th>Basis in this Research</th>
<th>Suggestion(s) for Further Research</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>The Research through Design (RtD) Methodology for Game Studies</strong></td>
<td>Practice-based research in games is a developing field and this thesis has provided further data to demonstrate the benefits of such an approach. The presented Research through Design Methodology for Game Studies provides a robust, specialised methodology that can be further refined through ongoing application to other projects.</td>
</tr>
<tr>
<td><strong>Other Identified Design Philosophies</strong></td>
<td>In describing the framework for understanding the player (Section 1.2.3.1), the philosophies of player-supportive design, player-unsupportive design, and counter-supportive design were defined in relation to how they support players in attaining feelings of achievement and performative mastery. Further research may apply these definitions to the analysis of other existing games (as was demonstrated briefly in this thesis), or may develop these definitions further through application of their principles to the design of new game artefacts. Further work of this nature would benefit the field of game studies by expanding understanding of different approaches within design space.</td>
</tr>
</tbody>
</table>

To summarise, this research has provided a range of potential avenues for ongoing research. While ongoing exploration and refinement of the disruptive game design philosophy, its underpinning theory and supporting design and development models is the obvious route for future research, other interesting areas have also been touched upon. These include potential research into cognitive load as it links to player experiences of fear during gameplay, as well as research into sadistic and masochistic player personality traits and associations with preferences for game difficulty. More generally, this research provides an example of the application of practice-based, research through design within the field of game studies, which can be used as a foundation for further research following a similar methodological approach.

### 9.4: Closing Remarks

Burgun (2013, p.xviii) suggests that, in order to continue to advance the field of game design, “we need game design movements driven by a design philosophy”. This approach to game design then suggests new concepts of what games, and thus the player experience, should be. Burgun compares such approaches to the precedent set within art history, in which movements such as expressionism and cubism were driven by artists with a defined view of what art should be.

Throughout this research, and embedded within the design and structure of *Amnesia: A Machine for Pigs*, the primary driving aim has been to explore the potential of a new philosophical approach to game design that provides players with a different experience than existing design approaches.
In defining and implementing the disruptive game design philosophy and demonstrating the benefits of it to the player experience, this research has also provided evidence that there is scope within game space to continue to propose new approaches to design. However, rather than viewing these approaches as each vying for precedence within the field of game design, as is implied by Burgun’s (2013, p.xviii) comparison to movements in art history, this research has shown it is more beneficial to instead consider different design approaches as complimentary to each other. Thus, within game space, it is possible that many other design approaches also exist, whose discovery can continue to expand and diversify the range of player experiences that can be created.
References

Scribbblenauts. [Video Game]. Burbank, California: WB Games.


EarthBound. [Video Game]. Kyoto, Japan: Nintendo.


Bruce, A. (2013). Antichamber. [Video Game]. n.p: Bruce, A.


From Software. (2014). Dark Souls 2. [Video Game]. Tokyo, Japan: Namco Bandai Games.


---

367


Appendix A: Development of the Psychological Foundation for the Systems Model of Cognition

The contents of this Appendix form the initial ‘Literature Review’ that was carried out prior to the construction of the Systems Model of Cognition presented in Section 3.1. The following sections document the gradual development of the model through combining a number of different existing theories and models within cognitive psychology.

A.1: Behavioural Psychology and Games

Stemming from empirical philosophers such as Locke (1995, originally published 1690) and Mill (1829) and later works such as Thorndike (1913), Pavlov (1927) and Watson (1930), behavioural psychology is one of the most influential approaches to psychological research. Though not as dominant as it once was, it is still in use by some contemporary researchers under different terminology. As Malim and Birch (1998, p.21) state, “Behaviourism had a profound influence on the course of psychology during the first half of the twentieth century. Its offshoot, stimulus-response psychology, is still influential today”, while Egenfeldt-Nielsen (2006, p.191) states that behaviourism “continues to be influential in research on educational media, including video games concentrating on the overt and observable behaviours essential for facilitating learning”. Indeed, the processes of learning and knowledge construction often coincide with behaviourist theory across education and training fields.

Behaviourism, at a broad level, focuses on how humans respond to environmental stimuli. This includes how the application of rewards (or reinforcers) and punishments influence those responses, with focus on the types of behavioural modification evident when manipulating the patterns of rewards and punishments (Malim and Birch, 1998, p.21). These reinforcers and punishments can be positive (additive) or negative (subtractive) in their nature. In the context of games, either something is added, such as an extra life (positive), or something is taken away, such as a special ability of the player (negative). Moreover, subsequent behavioural modification can be caused through individuals developing associations between a stimulus and a response, without the need for reinforcement or punishment to be applied. In games, if players are consistently provided with a reward or punishment for a particular behaviour, the strength of association between the stimulus and response increases, which encourages increased or decreased behaviour of that type (dependent on whether the behaviour was rewarded or punished).

However, while a behavioural approach to the study of games is able to propose answers for questions involving more basic levels of player interaction and motivation, such as meeting the desire to be rewarded, this approach alone is not able to describe all behaviour adequately. Suggested behavioural modification processes, such as classical conditioning and operant conditioning assert a need for a trial and error process towards learning and knowledge construction through
repeating behaviour that is a closer and closer approximation of the ‘correct’, or desirable behaviour for a situation. Thus, ‘performative’ skill mastery during gameplay (Section 1.2.3.1) may be reinforced through such conditioning mechanisms.

However, such an approach is unable to explain behaviour exhibited without trial and error, what Watkins (2000, p.95) refers to as behaviour learned through “conscious thought or insight” and what Malim and Birch (1998, p.24) refer to as “spontaneous, novel or creative behaviour”. Ertmer and Newby (1993) summarise various higher level skills that behavioural psychological theory is not able to adequately explain, citing specifically “those that require a greater depth of processing (e.g. language development, problem solving, inference generating, [and] critical thinking)”.

Furthermore, behaviourist psychology’s reductionist approach to human behaviour, along with its highly quantitative, scientific focus could be argued to overly simplify complex cognitive behaviours. Behaviourism ignores cognitive processes, treating the individual as a black box, with the assumption made that in order to assess learning, one must observe only external, viewable reactions to various stimuli (Siang and Rao, 2003, p.1). Processes that occur internally and that cannot be quantitatively measured are thus not a component of this approach.

Behaviourist psychology may have a place within game design when linked to reward structures and motivations for players to continue playing. However, its narrow focus on external stimuli and observable responses, along with its inability to adequately explain creative, novel, behaviour limit how useful it can be as a standalone approach both for disruption of expectation and for game design at a broader level. Thus, although possible to analyse games through the lens of behaviourist psychology, to do so singly would be to devalue the many other elements that comprise games themselves, as well as the many other elements of the gameplay experience that players engage with. In a gameplay context it is clear that any given game situation is more complex than any individual case of behavioural modification, as players are often dealing with multiple different stimuli at once, rather than single, readily controllable stimuli with clearly identifiable responses. In terms of disruptive game design specifically, the ‘black box’ approach is problematic as player expectations and the satiation of high-level player needs will require an understanding of the processes within the ‘black box’. Thus, an approach that exposes such internalised processes is required.

A.2: The Cognitive Approach to Memory and Knowledge

The 1950s and 1960s saw cognitivism gradually replace behaviourism (Robins, Gosling and Craik, 1999). Glassman and Hadad (2004, p.150) illustrate the differences between these two approaches with the question: “is knowing something the same as doing something?” (emphasis original). Using an example of a young child undergoing a problem solving task, they highlight how a purely behaviourist approach to attempting to understand learning and knowledge is flawed in its ability to explain certain types of behaviour.
A young child is given a problem in which pointing to a star leads to getting candy as a reinforcer, but pointing to a circle does not. Gradually the child becomes consistent in choosing the star. However, once they have learned this discrimination, they are given a new problem, involving a square and a triangle. On this problem, the child is consistently correct by the second trial [...]. Behaviourists would describe the task as learning to respond to a discriminative stimulus (the shape), a process which requires trial and error learning. Yet in this situation the child seems to develop a rule for making choices, rather than simply being reinforced for a particular response. (Glassman and Hadad, 2004, p.150)

The formation of these behavioural rules is argued to be evidence of internal cognitive processes influencing the construction of knowledge, and influencing responses to different stimuli. These processes are referred to as mediators (Malim and Birch, 1998, p.24), as they operate on incoming information (stimuli) to mediate, or affect, the subsequent behavioural response. The cognitive approach still recognises the importance of the stimulus-response relationship that behaviourism describes but does not treat the individual as a black box information processor. It is the internal processes, and their impact, that cognitivism seeks to understand.

A review of a range of empirical studies, carried out by Brewer (1974), suggests that there is in fact no definitive evidence that supports a purely ‘behavioural’ (what Brewer refers to as ‘mechanistic’) interpretation of classical or operant conditioning. Brewer suggests a cognitive interpretation in which higher-level cognitive functions mediate what may appear to be ‘conditioned’ responses. Later work reviewing data produced following Brewer’s review also conclude that the mechanistic view of conditioning is poorly supported empirically (Lovibond and Schanks, 2002, Schanks and M. F. St. John, 1994).

Further evidence supporting the argument that the learning process is mediated can be seen in a range of experiments by Köhler (1925). These experiments were intended as intelligence tests for chimpanzees, in which they were provided with a range of problems to solve with different tools. One such experiment required the chimpanzee to reach a reward (a banana) outside of their cage. Having solved a number of similar problems previously, the chimpanzee was provided with a number of hollow sticks, all too short to reach the banana on their own, but of different sizes so they could be slotted together to form a longer stick that could reach the banana. After unsuccessfully trying to reach the banana with the sticks individually, the chimpanzee stopped attempting to reach it and moved to a different area of the cage. As Köhler (1925, p.134) then states, despite the chimpanzee not having attempted to join the sticks together previously, “the solution follows quite suddenly”, with the chimpanzee joining first two sticks together, then adding the third to form a stick long enough to retrieve the banana.

The argument posed by Köhler based on this and other similar experiments was that learning and understanding such as this was not based on trial and error. In the above example Köhler notes that there was no gradual sequence of progressively closer approximations of the solution to the problem. Instead “there was a sudden change in the way Sultan [the chimpanzee in this particular study] organised the elements – insight” (Glassman and Hadad, 2004, p.151-152). The term insight
is used by Köhler to describe the change to how one organises a problem and its elements. This change is usually accompanied by a switch from random responding, or random behaviour, to rule-based and more purposeful behaviour. This 'insightful' behaviour is as previously noted, something that a purely behaviourist approach is unable to adequately explain.

While the concept of insight is somewhat ill-defined in terms of being an observable and measurable phenomenon, the manner in which Köhler describes the formation of mental sets (i.e. cognitive frameworks that group knowledge together) was the basis from which the more commonly used term schema has arisen. These terms each broadly refer to the concept of a person possessing an internalised ‘knowledge framework’ that assists in the organisation and recall of information in response to a situation.

While the terms schema and mental set are theoretically similar, differences between them need to be identified, as they tend to have different interpretations across different texts. For example, Glassman and Hadad (2004) use the terms synonymously, simply stating that they are different terms for the same concept. However, understanding of the term mental set by scholars such as Luchins (1942) differs; he states that a mental set (a way of viewing and understanding a given situation) increases the chance of a particular procedure, or behaviour, being selected because it has been successful in the immediate past; a mental set biases behavioural decision making towards behaviours that have a been recently successful. Whereas, an individual’s prior knowledge (gathered from previous experiences not in the immediate past) is concerned with the initial chance of a behaviour being selected (Öllinger, Jones and Knoblich, 2008, p.270). That is, if no knowledge of recent behaviour is available to form the basis of a mental set, the individual’s collected prior knowledge will provide an instigator to an initial behavioural decision.

This differs from the concept of schema in which all prior gained knowledge, regardless of how recently it was acquired, equally influences understanding and decision making in a given situation. The fields of memory (Section A.2.1 and Section A.2.2), information processing and knowledge construction (Section A.2.3, Section A.2.4 and Section A.2.5) are thus of particular significance to the investigation of how player expectations may be disrupted. If indeed cognitive processes act as mediators of stimulus-response behaviours and these mediators are underpinned by cognitive structures such as mental sets, then this raises the question of whether these processes and underpinning structures can themselves be disrupted.

**A.2.1: Cognitive Models of Memory and Recall**

There are a number of different proposed approaches to the construction, storage and recall of knowledge within cognitive psychology. These can be grouped into two broad categories, defined under slightly differing terms in different texts. These are the information processing approach to memory and the ecological approach to memory, in line with the terminology used by Malim and Birch (1998). However, the term ecological in this particular text simply refers to memory within
non-laboratory experimental contexts and, as such, also includes terms such as everyday memory (Eysenck and Keane, 1995, 2010) and ordinary memory (Neisser, 1988).

**A2.1.1: A Basic Model of Memory**

The information processing model views the brain as being analogous to a computer system, in that it receives input from external stimuli, processes that information, and then stores it in an organised fashion so that it is available for future retrieval. This model of memory has been developed over a number of years and has had a number of different researchers contribute to it.

The underlying principle of this model of memory is that it is divided up into separate storage areas, each with its own distinct properties, and that information from stimuli is processed, in a linear fashion, through these different storage areas. ‘Decay’, or forgetting, occurs at different rates depending on the position in the process that the stimulus information is currently at.

![Figure A.1: Basic model of memory, following Waugh and Norman (1965, p.93).](image)

An early interpretation of such a model (Waugh and Norman, 1965), utilises the terms primary and secondary memory (Figure A.1) as defined by James (1890). In this model, events that have never left conscious thought are held in primary memory and past events that have previously left conscious thought are held in secondary memory. While this model demonstrates a simple view of the function of memory, it lacks key information. It does not suggest how long information can be stored in primary memory without a rehearsal loop for example. It suggests that primary memory is of limited capacity but does not suggest how many pieces of information can be stored in primary memory at once before that capacity is exhausted. The process by which information moves from primary to secondary memory (other than that it involves rehearsal of the information) is similarly vague in its definition.

**A1.2.1.2: Atkinson and Shiffrin’s Multi-Store Model of Memory**

Atkinson and Shiffrin (1968) built upon this initial dualistic model, suggesting possible properties that control processes involved in the different storage areas of memory. They also expand the model to include three, rather than two storage areas in the memory system.
Figure A.2: Atkinson and Shiffrin’s (1968) Multi-Store Model of Memory.

This multi-store model of memory includes the sensory register, short-term memory and long-term memory (Figure A.2). The sensory register receives immediate multi-sensory information from external stimuli. Atkinson and Shiffrin state that in the form of visual information, this sensory process is well understood. They cite the work of Sperling (1963) and the properties of visual registration, such as a "several hundred millisecond decay of a visual image" (Atkinson and Shiffrin, 1968, p.92) in making it possible to identify this register as an explicit component of the memory system. However, they acknowledge the lack of evidence to explain the process by which non-visual information is registered. Further work by Crowder and Morton (1969) suggested though that audio stimuli at least have similar properties in sensory memory to visual stimuli.

The short-term memory store, which Atkinson and Shiffrin briefly refer to also as working memory (a term further used in later research, and discussed in Section 3.2.1.3), receives information from the sensory register and is able to hold this information for much longer before ‘decay’ (i.e. forgetting) occurs. While they tentatively assert a period of 15-30 seconds before information decay occurs in short-term memory, they note that measuring decay rates in both short-term and long-term memory is complicated by differing rehearsal mechanisms and variables dependent on the individual participant under study.

The long-term memory store is capable of storing information that is "relatively permanent (although it may be modified or rendered temporarily irretrievable as the result of other incoming information)" (Atkinson and Shiffrin, 1968, p.93). Information held in the long-term store can also flow back to the short-term store to provide context for information being processed through the sensory register, which Atkinson and Shiffrin suggest would be an important process in "problem solving, hypothesis testing and "thinking" in general" (1968, p.94) and that such transfer is controlled by the individual. They state that the limited capacity of the memory stores can be used for either information storage, or for conscious processing of information, such as rehearsal within the short-term store. The capacity is fixed but how it is used involves a balance between storage and processing functions and that balance is flexible.

This model, while it is useful in organising the theoretical structural components of the memory system and has been used by later theorists as a foundation for refined models of memory (Eysenck and Keane, 2010, p.209), nevertheless has weaknesses. It focuses on the structure and organisation of the memory system, but affords little attention to the issues of the processes
involved in memory (viewing memory as passive rather than active) and importantly, what memory is used for in day to day life.

The lack of ecological validity is a point of particular criticism, with much research being carried out using techniques that do not equate to activities an individual would undertake in a normal daily routine. For example, many experiments make use of word lists as a means of measuring memory storage and recall capabilities, although such an activity is unlikely to be representative of common day to day memory-based tasks.

Furthermore, this model does not consider the types, or modes, of information that are being stored, which poses some questions. For example, is there a difference between how audio and visual information is stored? Or, what is the effect on information storage if information has particular meaning to an individual, or is otherwise distinctive or unusual? In the context of gameplay in which multimodal information, especially combined visual and auditory information, is being perceived by a player, possible differences in how these stimuli are processed and stored and later retrieved from memory may be of particular importance. Expectations linked to different modalities of stimuli and stored knowledge may potentially be influenced by different types of ‘disruption’.

It is also not universally accepted that there is a defined separation between long and short-term memory stores, as argued by the levels of processing theory (discussed in Section A.2.1.4). However, significant evidence in support of a separation between the two has come from a range of clinical studies, primarily of patients suffering from Korsakoff’s syndrome, in which long-term memory is significantly impaired whilst short-term recall functions at a similar level to healthy individuals. Evidence from other patients, suffering brain damage in forms other than Korsakoff’s syndrome, also provide similar data as seen in research such as Zangwill (1946), Milner (1966), Baddeley and Warrington (1970) and Brooks (1975). However, it raises the question of the function of the short-term memory store. This is debated in later works (Baddeley, 2000, 2002, Baddeley and Hitch, 1974, Cattell, 1963, Logie, 1996) and in recognition of the interaction that this storage area has both with the sensory register and with the long-term memory store, it was termed ‘working memory’.

**A.2.1.3: Working Memory**

Working memory is responsible for processing incoming information from the sensory register, it is also able to interact with and retrieve information held in long-term memory. This information can then be processed in relation to the current stimulus information, to assist in the formation of understanding by contextualising incoming information. Baddeley’s work in particular led to the formation of a ‘working memory’ theory as an expansion of the existing idea of ‘short-term memory’.
The multiple functions that the working memory carries out led Baddeley and Hitch (1974) to argue that the working memory store should be conceived as itself containing multiple components (Figure A.3) forming the overall working memory system (as opposed to the alternative view of a single unitary short-term memory store). Additionally, as the terminology implies, this approach to memory views the system as one that consists of active processes that manipulate, rather than passively process that information.

The central executive (Figure A.3) is the component that controls an individual’s attention, allowing focus to be given to different sensory stimuli. This central executive is of a limited capacity, meaning that only a finite number of stimuli can be given attention at any one time. The central executive is not in itself a storage area, instead acting as a coordinator for attention. This component is then assisted by the two additional systems, the phonological loop which is concerned with short-term memory and rehearsal of audio and verbal information (via what Baddeley (1986) refers to as "subvocal rehearsal", and what Malim and Birch (1998, p.299) refer to as "the inner voice") and the visuospatial sketchpad that provides similar functionality for visual and spatial information (Baddeley, 2002, p.86).

An evaluation of this working memory model suggests that, in terms of its ability to explain how memory functions in day to day life (i.e. the model’s ecological validity) it is more readily applicable than the multi-store model (Section A.2.1.2) from which it was developed. The central executive, in acting as a coordination system that selects and prioritises incoming stimulus information, makes it possible to more clearly understand how memory and knowledge organisation systems are able to deal with different types of stimulus (e.g. visual, spatial, auditory, and verbal) and contextualise and apply them to a range of diverse tasks (Malim and Birch, 1998, p.300). However, the model of working memory (Figure A.3) fails to elucidate on the precise function of the central executive, in terms of how it manages attention given both to sensory information and to information being simultaneously retrieved from long-term memory. How these different types of information are combined to create meaning and understanding of a situation is equally poorly defined.
Baddeley (2000) identifies these issues, also noting that the phonological loop process is not only important for retaining information on a short-term basis but also plays a significant role in long-term phonological knowledge construction (2000, p.2). To address these criticisms, Baddeley (2000) proposes a fourth component of the working memory model, in the form of the episodic buffer (Figure A.4). This episodic buffer is, according to Baddeley,

assumed to be capable of storing information in a multi-dimensional code. It thus provides a temporary interface between the slave systems (the phonological loop and the visuospatial sketchpad) and LTM [long-term memory]. It is assumed to be controlled by the central executive, which is responsible for binding information from a number of sources into coherent episodes. Such episodes are assumed to be retrievable consciously. The buffer serves as a modelling space that is separate from LTM, but which forms an important stage in long-term episodic learning. (Baddeley, 2000, p.5)

The episodic buffer is stated as critical in the process of passing information to, and retrieving information from, long-term memory systems. The components of the original model (the central executive, phonological loop and visuospatial sketchpad) are dynamic systems, used for short-term storage and for information processing in a manner that can be conceived as similar to the function of Random Access Memory (RAM) in a computer system. While these dynamic systems are able to interact with their respective long-term memory stores (which could be conceived as the Hard Disk Drive (HDD) in the computer system analogy), the dynamic systems themselves are not able to store information over long periods.

The dynamic systems (i.e. working memory) interact with their respective storage areas within long-term memory; language information from the phonological loop, for example, and visual semantic information from the visuospatial sketchpad. However, these two long-term stores are not the only ones that the fluid systems interact with. The episodic long-term memory store both receives and provides information from and to the episodic buffer, which binds the multimodal information together to formulate understanding in a situation. That understanding is then able to be passed back into episodic long-term memory for future retrieval. An important feature of this episodic memory is that it is able to also store temporal information and organise information

Figure A.4: Working Memory model showing episodic buffer and links to long-term memory stores, following Baddeley (2000, p.5).
chronologically, meaning that it is important in the storage and recall of time dependent information, such as stories or sequential tasks or processes. In the context of games and gameplay, this temporal information as it relates to sequential tasks (e.g. carrying out a specific sequence of button or key inputs to perform a sequence of in-game actions) is of particular relevance.

However, while the working memory model has been influential in shaping much research in cognitive psychology, it is not the only current theory available. Its strengths lie in its ability to indicate the possible processes involved between perceiving sensory information, processing that information in combination with the binding of stored information from long-term memory and then the creation of new understanding that in turn can be stored again for later recall. This is something that the multi-store model (Section A.2.1.2) was unable to do. However, there is another alternative available in the levels of processing theory (Craik and Lockhart, 1972), which firmly rejects the concept of separate memory storage areas, such as short and long-term stores, completely.

**A.2.1.4: Levels of Processing**

In evaluating the evidence against the concept of separate short and long-term storage areas, Craik and Lockhart (1972) highlight a lack of clarity in the definitions of ‘limited capacity’ asserted by multi-store theorists (Section A.2.1.2), conflicting evidence of how information is coded within short and long-term memory (for example, remembering information based on verbal or semantic properties) along with conflicting evidence for mechanisms of ‘forgetting’ across different studies (Craik and Lockhart, 1972, p.673-675).

Rather than a processing of information through a linear system, Craik and Lockhart (1972) propose that memory is dependent on the depth of processing that is applied to information. More deeply processed information will be easier to recall than information that has been processed at a more shallow level. The definitions of shallow and deep processing are defined in terms of structure and semantics (Craik and Lockhart, 1972). Information that has been shallowly processed has been processed in relation to its structural form and physical characteristics only, such as identifying a person as male or female, or identifying whether or not a word is in upper or lower case (i.e. description). Deep processing occurs when information is considered in terms of its semantics (i.e. meaning), for example, asking an individual whether two words are synonyms. Such word-based research was carried out by Craik and Tulving (1975), which identified differing response latencies between the different proposed levels of processing, from structural through to semantic processing.

The results have thus shown that different encoding questions led to different response latencies; questions about the surface form of the word were answered comparatively rapidly, while more abstract questions about the word’s meaning took longer to answer. If processing time is an index of depth, then words presented after a semantic question were indeed processed more deeply [...]

380
Semantic questions were followed by higher recognition of the word. (Craik and Tulving, 1975, p.273)

Further evidence based on structural versus semantic processing of words can be seen in studies by Elias and Perfetti (1973), and Fiske and Schneider (1984). Craik and Tulving note that the data may suggest that it is a matter of processing time more than depth that leads to greater recall accuracy. However, they highlight that in situations where the qualitative nature of the processing was held constant, whilst the processing time was varied, the recall accuracy did not correlate directly to processing time, which they argue as evidence against such an interpretation.

Much of the evidence to support the levels of processing theory comes from experiments that are not representative of daily life (Malim and Birch, 1998, p.302), lessening the theory’s ecological validity for the same reason as the multi-store model (Section A.2.1.2). Additionally, a common argument against the theory is that it lacks a quantitative index of processing ‘depth’ against which to take measurements. This lead to concerns over circular arguments to explain data, for example identifying information as having been deeply processed because it was recalled more accurately, and then using this greater accuracy of recall as evidence of deeper processing (Eysenck, 1978, p.159).

However, while the theory has its weaknesses, it has been influential within the field of memory research and has also undergone a number of modifications and refinements since its introduction. In a retrospective paper (Lockhart and Craik, 1990), it is asserted that the concept of memory performance being strongly linked to the nature of information processing is widely accepted (p.109). It is also noted that the other existing models of memory (Section A.2.1.1, Section A.2.1.2 and Section A.2.1.3) have imperfections and gaps in understanding as well.

A.2.1.5: A Systems Model of Memory for the Disruptive Game Design Philosophy

While the multi-store model of memory (Section A.2.1.2), working memory theory (Section A.2.1.3), and the levels of processing theory (Section A.2.1.4) approach the structure and processes of memory differently, what can be suggested based on the reviewed evidence is that both structure and process are of importance in understanding the workings of memory. The multi-store model, while potentially unrealistic in its fractioning of memory into explicit storage areas, provides a strong framework upon which to base further work. Similarly, the evidence supporting working memory, and the effect of processing levels on accuracy and speed of recall, identifies possible areas for consideration within the current research when identifying means of disrupting the player’s expectation and existing knowledge (Section 1.3).
A combined, ‘systems’ model of memory and recall that brings together the principles of the memory models examined (Sections A.2.1.2, A.2.1.3 and A.2.1.4) can now be proposed (Figure A.5). The instigating stimulus (now situated in ‘the world’), and the sensory perception of that stimulus (now situated in ‘the body’) as suggested by Atkinson and Shiffrin’s multi-store model (1968), are combined with the active construction of knowledge within working memory (Baddeley, 2007) through the merging of incoming information from the stimulus with recalled knowledge from long-term memory, to produce a response from the body that has an effect on the world.

The impact of processing levels (Craik and Lockhart, 1972) is suggested in this Systems Model to occur during the processing of information within working memory, depending on the information being processed, how it is rehearsed in working memory and how it is encoded into the different long-term memory stores. However, this model needs further refinement if it is to be used as a basis for the disruptive game design philosophy. For example, how is knowledge stored within long-term memory specifically? The manner of knowledge storage may influence what methods of disruption may be more, or less, effective. Furthermore, while the levels of processing theory suggests associations between information processing and recall, this aspect of the model is not yet refined enough to be used as basis from which to understand the game as played and thus, to design disruptive games. A greater understanding of knowledge encoding and long-term memory organisation, along with knowledge recall, are thus required.

A.2.2: Structure of Knowledge Storage in Long-Term Memory
Structure is a key word in relation to long-term storage of information. The complexity and range of information that will be stored in long-term memory seems to require there to be a more granular structure than simply a single ‘long-term memory store’ for it to be possible to access and recall information in a timely manner (Malim and Birch, 1998, p.297). While the Systems Model proposed in Section A.2.1.5 places three sub-stores within long-term memory (i.e. the visual semantics, language, and episodic long-term memory stores), these do not appear adequate to include all types of long-term memory.
Tulving (1985a) asserts that there are three clearly definable forms of long-term memory dependent on the type of knowledge being remembered; procedural memory, episodic memory and semantic memory. Episodic in this instance holds some similarities in meaning to Baddeley’s (2000) use of the term in the ‘episodic buffer’ and ‘episodic long-term memory’ components of working memory (Section A.2.1.3) and indeed Tulving’s idea of episodic memory was a contributing factor in the development of the working memory theory. In posing this theory of multiple memory types, Tulving argues that doing so eliminates many of the issues that go alongside theories of memory as being a ‘whole’. As no generalisations can be made that apply to all of memory, attempting to argue a theory that takes such a stance “gives rise to needless and futile arguments [that] would become noncontroversial if their domain was restricted to parts of memory” (1985a, p.385).

Procedural memory enables the constructing of relationships between stimuli and responses, as in behavioural conditioning (Section A.1). However, beyond simplistic relationships, procedural memory also allows the linking of such relationships into complex patterns, or “response chains” (Tulving, 1985a, p.387) and so provides a means of constructing knowledge of processes, such as how to play a piano, or when applied to gameplay, how to use a joystick to move a sprite around a screen.

Semantic memory provides the ability to internalise the representation of concepts that are not perceptually present. For example, it affords the ability to make the statement that “the sky is blue”. The concept of ‘blue’ is not perceptually present – it is a word that provides meaning to a perceived wavelength of light. However, semantic memory allows the linking of the meaning of the verbalisation "blue" to the perception of the matching wavelength of light. General knowledge of the world, such as knowing the capital of a country, is similarly semantic in nature, as it relates to the meaning of information, rather than being related to physical stimulus-response relationships. Tulving (1985a, p.387) also relates semantic memory to the construction of ‘mental models’ as defined by Craik (1943). This concept of mental models holds a number of similarities to the previously noted concepts of mental sets and schemas (Section A.2) and is discussed in greater depth in Section A.2.4.2.

Lastly, episodic memory holds information related to personal life experiences and affords the ability to organise these memories according to their spatiotemporal context, thus allowing an individual to remember events in their lives in a chronological sequence, and in relation to the place in which they occurred.
The three types of memory operate in a monohierarchal fashion, with each subsequent memory type requiring the preceding type to be in place, but with each memory type possessing unique characteristics and not inheriting those of lower levels in the hierarchy. So, for example, it is impossible for an organism to possess episodic memory without the corresponding semantic memory, and impossible for it to possess semantic memory without the corresponding procedural memory, although semantic memory systems can exist independently of episodic systems, and procedural systems independently of semantic systems. (Tulving, 1985b, p.3)

<table>
<thead>
<tr>
<th>Memory System</th>
<th>Type of Consciousness</th>
</tr>
</thead>
<tbody>
<tr>
<td>Episodic Memory</td>
<td>Autonoetic Consciousness (Self-Knowing)</td>
</tr>
<tr>
<td>Semantic Memory</td>
<td>Noetic Consciousness (Knowing)</td>
</tr>
<tr>
<td>Procedural Memory</td>
<td>Anoetic Consciousness (Non-Knowing)</td>
</tr>
</tbody>
</table>

Figure A.6: Schematic diagram of memory type and consciousness relationships, following Tulving (1985b, p.3).

Additionally, Tulving (1985b) links the three memory types to three different levels of consciousness, in the forms of anoetic (non-knowing), noetic (knowing), and autonoetic (self-knowing) consciousness. This combination of memory types and levels of consciousness can be represented schematically (Figure A.6), where arrow direction can be read as “implies the presence of”.

The proposal of a monohierarchal format may be questioned with regard to its accuracy in all circumstances however. Not all semantic knowledge may necessarily require corresponding procedural knowledge. It is not necessary for example, when memorising the semantic knowledge that “Paris is the capital of France”, to also memorise a ‘process’ related to that semantic information. It could therefore be suggested more applicable to day to day memory use to consider the three memory types as only partially dependent on the memory types lower in the hierarchy than themselves.

Tulving’s theory of multiple memory types is reflected in similar theories that each categorise memory types in similar ways, with slightly differing terminology. Ruggiero and Flagg (1976) for example refer to stimulus-response (analogous to procedural), organised (analogous to semantic, as they are categorised according to meaning) and representational (analogous to episodic, in that memories are representative of life experiences). Similarly, Oakley (1981) refers to associative
(analogous to procedural – associating stimuli with responses), abstract (analogous to semantic in that meaning is abstracted from perceived reality) and representational, with the same meaning as employed by Ruggiero and Flagg.

The division between procedural memory and ‘other’ memory is widely accepted (Tulving, 1985a, p.389), however the nature of the ‘other’ memory is subject to debate focused on whether this ‘other’ memory consists of a single or dual store structure.

As was the case with evidence in support of the multi-store model (Section A.2.1.2), evidence to support different memory types and their clear separations comes from clinical research of patients with various forms of brain injury. Nielsen’s (1958) work with such patients found that two distinctive types of amnesia result in patients losing either memory of personal experiences (temporal amnesia) or memory of acquired facts (categorical amnesia), and that the loss of one need not impact the other. Further research (Wheeler, Stuss and Tulving, 1997) that draws upon evidence from “brain imaging, neuropsychological experiments, clinical observations and developmental psychology” (p.331) similarly supports a distinction between memory types. Glassman and Hadad (2004, p.161) note two practical examples highlighting the distinction, stating that a patient may be able to learn how to solve a puzzle (procedural), but be unable to recall that they have previously seen it (episodic), or be able to recite a fact (semantic) without knowing where or when they learned it (episodic).

Arguments against theories of different memory types, such as the critical evaluation provided by McKoon, Ratcliff and Dell (1986) and further in Ratcliff and McKoon (1986), focus on the vagueness of the criteria that separate episodic and semantic memory. It is noted for example that “although intuition suggests that overlearned [consistently rehearsed in memory] personal events (e.g. “the first time I...”) should be part of permanent semantic knowledge, they would have to remain, by definition, part of episodic knowledge” (McKoon, Ratcliff and Dell, 1986, p.296) because they are temporally organised. Such critique however is noted by theorists such as Tulving and has driven further research. As Glassman and Hadad (2004) state, “there is growing evidence for different storage systems in LTM [long-term memory]” (p.162), with some evidence indicating that semantic and episodic memory, while distinct in their function, may also be in some ways interdependent (Menon et al., 2002).
<table>
<thead>
<tr>
<th>Type of Memory</th>
<th>Content</th>
<th>Organisation</th>
<th>Retrieval Process</th>
</tr>
</thead>
<tbody>
<tr>
<td>Episodic Memory</td>
<td>Events, Experiences</td>
<td>Time-based</td>
<td>Deliberate (High Effort)</td>
</tr>
<tr>
<td>Semantic Memory</td>
<td>Facts, Concepts</td>
<td>Cognitive Schemata</td>
<td>Deliberate (Low Effort)</td>
</tr>
<tr>
<td>Procedural Memory</td>
<td>Actions, Processes</td>
<td>Activities</td>
<td>Automatic</td>
</tr>
</tbody>
</table>

Table A.1: Characteristics of long-term memory types, following Glassman and Hadad (2004, p.161).

Glassman and Hadad (2004) summarise the properties of the three memory types, and also add to their definition the type of retrieval process that applies to each (Table A.1). The organisation methods and retrieval processes are of particular interest in terms of the disruptive game design philosophy, as they will enable the identification of theoretical structures of knowledge organisation that may be possible to disrupt, along with processes of retrieval that may be open to disruption via interference.

Figure A.7: Adapted Systems Model of memory and recall incorporating separate long-term memory stores.

The separate procedural, semantic and episodic memory stores can be added to the previously proposed (Figure A.5) Systems Model of memory and recall, with recalled knowledge being passed into working memory and incoming information being passed into appropriate memory stores (i.e. the process of learning, or knowledge construction) (Figure A.7). The separate visual semantics and language stores, being semantic (i.e. meaning-based) in nature, can be suggested to be sub-stores within the broader semantic memory store. However, the separate procedural long-term memory store is problematic. There is no mechanism within Baddeley’s (2007) model of working memory via which procedural information can be processed and passed into this procedural long-term memory store.
Figure A.8: Further adaptation of Systems Model of memory and recall, refining working memory into parallel declarative working memory and procedural working memory stores.

An adaptation to working memory proposed by Oberauer (2009, 2010) provides a means of solving this problem however (Figure A.8). Oberauer suggests that working memory is split into two distinct types; declarative working memory and procedural working memory. Declarative working memory is concerned with the "representation and selection of the contents of cognitive activity" (Oberauer, 2009, p.57, emphasis added). That is, declarative working memory is responsible for what is being cognitively processed; this content will be drawn from episodic and semantic long-term memory and combined with input from external stimuli via sensory perception. Procedural working memory is instead concerned with the "representation and selection of the cognitive operations themselves" (Oberauer, 2009, p.57, emphasis added). Procedural working memory operates alongside declarative working memory to select appropriate processes to apply to the cognitive content. This procedural working memory similarly draws on long-term procedural memory and is also able to send information back to that long-term memory store.

In Oberauer’s proposed model of working memory, declarative and procedural working memory can be conceptualised as separate, but working in parallel with each other. Procedural working memory operates on the content of declarative working memory; thus, changes to declarative working memory content will cause changes in the content of procedural working memory. The procedural working memory’s processing of declarative working memory content is referred to by Oberauer as a primary process (2009, p.69). He separates these from executive processes which instead control the conditions of primary processes by changing their parameters (e.g. by changing the ‘aim’ of cognitive processing from ‘make an accurate decision’ to ‘make a fast decision’, or changing the ‘task’ being processed from ‘play a piece of music on the piano’ to ‘make a cup of tea’), or by changing the representations within declarative and/or procedural working memory. The further adaptation of the Systems Model of memory and recall (Figure A.8) represents this by retaining Baddeley’s (1974) concept of a central executive to provide the ‘executive processes’.
While the Systems Model presented in Figure A.8 is necessarily simplified for the purpose of being applied in a gameplay context, it is based on robust psychological research. Similarly, the conceptual splitting of working memory into declarative and procedural working memory has been experimentally investigated (Souza et al., 2012) and evidence found to support the existence of two analogous working memory subsystems (Souza et al., 2012, p.1006). This splitting of working memory thus supports the structure of long-term memory as consisting of three different memory stores (i.e. procedural, semantic and episodic).

With Oberauer’s separation of declarative and procedural working memory, it is lastly necessary to reconsider Tulving’s (1985b) suggestion of long-term memory stores as being monohierarchical. With a model in which working memory was a singular system, it did not appear necessary in all circumstances for procedural knowledge to be in place before semantic knowledge could be acquired (e.g. knowing that “Paris is the capital of France” did not appear to also require a ‘process’ associated with that knowledge). However, in a model in which declarative and procedural working memory operate in parallel with each other, knowing that “Paris is the capital of France” (i.e. declarative, semantic knowledge) requires procedural knowledge to ‘process’ it; that procedural knowledge may in this case be as simple as the ‘process of forming a sentence in English’ to process the declarative semantic knowledge. More complex knowledge in a gameplay context (e.g. declarative semantic knowledge of an enemy’s weak spots) may require more complex procedural knowledge to process it (e.g. the sequence of actions/inputs required to take advantage of those weak spots and defeat the enemy). Thus, while it may be possible to have semantic knowledge without having overtly associated procedural knowledge, procedural knowledge is still required to process the semantic knowledge when required.

A.2.3: Cognitive Theories of Knowledge Organisation, Recall, and the Construction of Meaning

Models of how memory as a system may function (Section A.2.1), along with how long-term memory may be subdivided into a more granular structure of different memory types (Section A.2.2), suggested some cognitive processes and cognitive structures that may be potentially targeted for disruption via the disruptive game design philosophy (Section 1.3). However, to disrupt a player’s game as played, it is necessary to not only understand how knowledge is stored and organised but also, what triggers its recall from memory and what effect on an individual’s expectations of a current situation recalled knowledge has.

The organisational structure of memory is debated, although certain distinctions are traditionally accepted, such as that there is a distinction between objects (bird, cat, knife, fork) and relations (above, next to, punch, kick) in memory (Eysenck and Keane, 1995, p.234). Broadly though, organisational theories fall into two dominant categories – semantic network theory and schema theory (Grow, 1996), which although not consisting comprehensively of all general theories of knowledge organisation, are two of the most influential (Botelho and Coelho, 1995, p.81). These
need to be then considered in the context of the three long-term memory types (Section A.2.2) and the game as played.

**A.2.3.1: Hierarchical Semantic Networks for Knowledge Organisation**

The concept of knowledge networks as they exist across the fields of psychology and artificial intelligence in contemporary research can be attributed to Quillian (1968), who specifically focused on their usage in relation to word concepts. Although as Sharples et al. (1996, para.16) note, Anderson and Bower (1973, p.9) trace the underlying notion of a network of associatively linked information ‘nodes’ to Aristotle and such a way of thinking has been heavily influenced by philosophers such as Locke (1995, originally published 1690).

There are a number of differing approaches that take different perspectives on how the information within the networks is categorised and linked, although each can be described under the encompassing term of *similarity-based* organisation (Eysenck and Keane, 1995, p.234).

![Figure A.9: An example of a hierarchical network model of knowledge organisation according to Collins and Quillian's (1969) model, following Malim and Birch (1998, p.298).](image)

Studies using the free recall methodology, in which participants are asked to recall lists of concepts (usually single words) in whichever order they come to mind have supported this assertion as words were recalled in semantically themed clusters. For example, once the word *dog* is recalled, the likely follow-up would be other animals. This grouping of recalled information can be seen in a range of studies (Bousfield, 1953, Rips, Shoeben and Smith, 1973, Romney, Brewer and Batchelder, 1993). However, beyond clustering of information, Collins and Quillian (1969, 1970) proposed that those clusters are hierarchically organised into linked networks (Figure A.9).

Collins and Quillian (1969) found that answering questions requiring information access from different hierarchical levels took participants differing lengths of time, depending on the number of links needing to be made between information nodes. Thus, answering the question “can a bird fly?” (Figure A.9) will be answered in less time than the question “can a sparrow fly?” because the
sparrow concept is hierarchically lower than the more general concept of a bird, thus it takes longer to retrieve.

While such a model appears to work well when subordinate concepts in the hierarchy are 'typical' members of respective superordinate concepts (as owl and sparrow could be considered in relation to the bird concept), the model, along with the predictions of recall speed it proposes, become less accurate when subordinate concepts are atypical of their superordinate concepts.

The ostrich concept, as it is positioned in the example network (Figure A.9), should be identified as being a bird in the same length of time as an owl or sparrow are identified as being birds, because they are all one hierarchical level away from the bird concept. As would be assumed in a linked hierarchical model, subordinate concepts inherit superordinate concept properties, however as noted by Sharples et al. (1996, para.23) this is not the case if a subordinate concept has a property that explicitly overrides one of the properties of its superordinate concepts, such is the case with the ostrich’s 'cannot fly' property. This overriding of inherited properties thus makes the concept atypical and as noted in both Kellogg (2003, p.228) and Eysenck and Keane (2010, p.265), this atypicality appears to have an impact on information access time (as seen in response time to questions pertaining to atypical concepts and properties). This, along with arguments asserting that hierarchical network structure was inconsistent, led to further extensions of the basic network model by Collins and Loftus (1975). Their extensions moved towards a spreading activation theory of knowledge network organisation and network-based recall.
A.2.3.2: Spreading Activation Theory

Whilst spreading activation theory retains the original notion of information being stored in nodes, those nodes are not rigidly hierarchically linked.

Figure A.10: Hypothetical concept relatedness links in a spreading activation model, adapted from Collins and Loftus (1975, p.412)

Instead, nodes are grouped in clusters (Figure A.10), with possible links between nodes in the same cluster as well as those in nearby clusters. The ‘distance’ between clusters indicates how likely two information nodes are to be semantically similar and thus how likely they are to be ‘activated’. For example, Figure A.10 uses shorter links to indicate greater relatedness between concepts and colours to differentiate between ‘clusters’ of related information nodes.

Activation spreads from the original node, becoming less likely to activate further nodes as it spreads. Activation is also linked to the period of time a concept is processed for (previously referred to as rehearsal), with longer processing equating to longer continued spread to other nodes. The decrease of activation strength is linked to distance spread and also decreased by intervening activity (i.e. being interrupted or distracted, or switching attention to another task before returning to the original task), which stops processing of the concept. These assumptions are among a list of thirteen different assumptions (Collins and Loftus, 1975, p.411) that support this theory, which the researchers note,
do not bend the [original] theory but merely elaborate it in such a way it can be applied to the kinds of experiments on semantic memory that have been performed recently. The elaboration may itself be wrong, so our mistakes should not be held against Quillian’s theory. (Collins and Loftus, 1975, p.411)

Individuals will thus not have the same links and cluster formations. For example, someone who regularly eats a snack on their way home from work may strongly associate the apple concept with the bus concept. However, the theoretical model is flexible enough to be able to account for such variation.

The spreading activation model also has similarities to the physiological structure of the brain. Malim and Birch (1998, p.298-299) note that “activation of a concept can be thought of as activating a neuron (or nerve cell). The activation then spreads to other neurons producing a pattern of excitation.” The structural analogy between this model and the physical structure of the brain would suggest the model is able to be supported by experimental data. Indeed, criticism of the model tends not to be targeted at the concept of a linked node network with spreading activation functionality itself, but at the concepts and properties assigned to the individual nodes.

While physical objects are likely to be perceived and thus semantically understood by individuals in similar ways, abstract concepts are harder to suggest properties for. Wittgenstein (1958) emphasises this point by discussing properties of the concept of a game.

[. . . ] How would we explain to someone what a game is? I imagine that we should describe games to him, and we might add: "This and similar things are called 'games' " [. . . ] One might say the concept 'game' is a concept with blurred edges. - "But is a blurred concept a concept at all?" - Is an indistinct photograph a picture of a person at all? Is it even always an advantage to replace an indistinct picture by a sharp one? Isn't the indistinct one often exactly what we need? (Wittgenstein, 1958, p.33-34, emphasis original)

As Wittgenstein explains, there are some commonalities between things that may be called 'games' - that they have at least a small set of rules or that they involve certain items like cards, dice or playing pieces. However, there are very few properties of the 'game' concept that would be consistent with every instance of that concept. The instances of the game concept share a common family resemblance without sharing every property. The notion of family resemblance as it is used by Wittgenstein, and further explained by Rosch and Mervis (1975), became a widely accepted position from which to consider concept-similarity-based structures of knowledge organisation (Komatsu, 1992).

Different people will have different network formations. Thus, even with simple physical concepts, such as animals, participants categorise objects differently between different experimental sessions (McCloskey and Glucksberg, 1978). However, Barsalou's (1982, 1983, 1985, 1987, 1993, Barsalou and Medin, 1986, Barsalou and Wiemer-Hastings, 2005) extensive research into 'concept stability' suggests evidence against 'concept-based' theories of knowledge storage and organisation, such as the possibility of different individuals’ understandings of a particular concept differing, or an
individual's understanding of a particular concept changing over time or under different circumstances.

Barsalou argues that concept interpretation, and thus the creation of meaning and understanding in different situations, is based not on static concepts in memory but instead, on concepts that have different linked information (from other 'nodes' or 'clusters') dependent on the context that they are perceived or recalled in. This produces a 'graded' structure of concepts in which some concepts are more wholly 'defined' than others. Barsalou (1987) notes that the notion of graded concept structures can be viewed from two opposing perspectives and thus empirical evidence can be interpreted as being both supportive of, and evidence against, them.

Pessimistically, they can be viewed as showing that graded structure is unreliable and meaningless. Optimistically, they can be viewed as showing that graded structure is highly flexible, with this flexibility being a fundamental property of the human cognitive system. (Barsalou, 1987, p.104)

With the aim of disrupting a player's expectations and existing prior knowledge (Section 1.3), concept-based network theories of knowledge organisation offer some possible foundations. For example, the impact of atypicality (Eysenck and Keane, 2010, Kellogg, 2003) (e.g. the ostrich concept) on the information retrieval process and recall time would suggest that disruption may be achievable by frequently presenting players with game components that are atypical of their containing categories and concepts.

However, operationalising such disruption through a game's design in an effective manner may be difficult using concept-based network theories alone. This is due to the restrictive definition of 'concepts' within them, in particular the inability of such definitions to consider abstract concepts (e.g. emotions, such as love, betrayal, or pride). What is required is a means of combining the structural form of network theory with a less restrictive and inflexible definition of 'concept'.

**A3.2.3.3: Cognitive Schema Theory of Knowledge Organisation**

Schema theory offers some solutions to the issues presented by concept-based network theories by providing a means of accounting for flexible mental concepts and context-dependent information, whilst allowing the notion of interlinked memory 'clusters' to still function alongside. As Grow (1996, p.9) states, "schema theory provides a different view of how knowledge is stored [to that proposed by semantic network theory], though network and schema theory are compatible and often used together".

Lindley and Sennersten (2008) elaborate this further, highlighting the key differences between network structures and schema structures.

While the taxonomical structures of semantic or declarative memory [as asserted by semantic network theory] are comprised of object classes together with associated features and arranged in subclass/superclass hierarchies, the elements of schemas are associated by observed contiguity, sequencing, and grouping in space and/or time. (Lindley and Sennersten, 2008, p.2-3)
Schema theory can be traced back to Plato and Aristotle (Marshall, 1995, p.4), with its inception into contemporary psychology in 1781 (Kant, 2003, originally published 1781). Martin (1994, p.265) identifies that Plato’s description of the term ‘schema’ includes one of its most significant features in the form of summarisation, meaning that “a schema can include important information rather than exhaustive information [. . . ] a schema acts as a reduced description of important aspects of an object or event” (p.265). Martin continues, discussing Kant’s assertion of the concept of learning through assimilation, and that “every experience is an assimilation of sensation through the lens of the schemata” (p.267). This suggests that only that which can be represented by a schema is capable of being experienced. While assimilation originated with Kant, its entry as a term into scientific theories of cognition occurred through the work on schemas of Bartlett (1932) and Piaget (1970).

Bartlett identifies the term ‘schema’ as being problematic, as in psychological writing it is used “to refer generally to any rather vaguely outlined theory” (Bartlett, 1932, p.201) and thus has no precise meaning and application. Thus, Bartlett suggests a regularly cited, more specific definition of the term:

‘Schema’ refers to an active organisation of past reactions, or of past experiences, which must always be supposed to be operating in any well-adapted organic response. That is, whenever there is any order or regularity of behaviour, a particular response is possible only because it is related to other similar responses which have been serially organised, yet which operate, not simply as individual members coming one after another, but as a unitary mass. (Bartlett, 1932, p.201)

Martin (1994, p.270) in discussing the importance of Bartlett’s work suggests that schemas possess a diagnostic property (that Martin terms diagnostcity), in that the recording and categorising of past experience “permits effective action” in future situations.

By recording experience, schemata allow history to operate on future behaviour. This is a critical insight because it is the role of schemata in the prediction and the planning and execution of behaviour that are their primary reason for continued existence. In short, schemata allow memory to affect the behaviour of a system by assimilating information for future use. (Martin, 1994, p.270)

In relation to a player’s expectations of a game during their experiencing of the game as played (Section 1.2.3.1), this ‘diagnostcity’ can be clearly likened to a mechanism through which expectations may be generated (i.e. from prior experiences and the game as expected). Past events therefore may permit effective action in future situations, assuming that those past events and the expectations attached to them remain accurate and undisrupted.

However, if the stored information is disrupted in some way, then the recalled information will similarly be a disrupted version of the original information. Bartlett (1932) documented this in the ‘War of the Ghosts’ experiment, in which British participants were requested to read, remember and later recall a short, unfamiliar Native American folk story. The results showed that participants changed information in the story as they attempted to recall it and a significant
number of these changes reflected what could be considered ‘cultural’ or ‘normative’ British schemas. This experiment led to support for the argument that memory is an active, constructive process, and also that existing schematic knowledge (or a person’s worldview) can distort information recall.

This is an interesting consideration in relation to disruptive game design (Section 1.3), as it suggests that disruptive game components may be disruptive in different ways (or perhaps entirely undisruptive) when perceived by players from different social or cultural backgrounds. In the model of game space (Figure 1.3), the impact of games ‘culture’ on the game as expected is noted. However, Bartlett’s experiment suggests that ‘culture’ more broadly may influence both this game as expected and, importantly for the disruptive game design philosophy, the game as played. This broad cultural impact can be conceived in the model of game space to be contained within the prior experiences as ‘previous life experiences’.

Further work has built upon Bartlett’s initial theory, such as Piaget (1964, 1967, 1970) who investigated the means by which schemas are initially constructed, Minsky (1974), who proposed the theory of ‘frames’ as a way of applying schemas to computer-based logic, and Schank & Abelson (1977) who proposed the theory of ‘scripts’ for describing stereotypical sequences of events.

Piaget’s work to identify how schemas are initially constructed through the study of the development of children focused on their developing cognitive capabilities. Piaget suggests, as did Kant, that learning occurs through a process of assimilation but that information that cannot be incorporated into existing schemas must instead be accommodated. This process utilises existing schema to interpret the information whilst simultaneously adapting the structure of the schema to better incorporate the new information. As Martin (1994) discusses, the concept of accommodation appears to operate against Kant’s argument that anything not represented by a schema cannot be experienced. However,

one interpretation of Piaget’s position is that he assumed the existence of the more fundamental categories and schemata that shape experience […] If so, then what he described as mismatch between schemata and the world, Kant would describe as revealed inconsistencies among the schemata. In other words, certain schemata might conflict with others in mapping sensation to experience. In this case an accommodation process as described by Piaget might provoke changes in schemata without violating Kant’s transcendental schematism. (Martin, 1994, p.270-271, emphasis original)

Thus, accommodation as a means of allowing the construction of schemas can operate alongside previously identified schema properties such as summarisation and assimilation (as per Kant’s (2003, originally published 1781) definitions) and diagnosticity (as termed by Martin (1994)).

Minsky (1974) extends the concept of schemas with ‘frames’. These are hierarchical, networked structures of nodes and node relationships, used for representing a stereotypical situation such as the layout and contents of a room in a house. Each frame has a number of ‘slots’, or ‘terminals’,
that can hold information and Minsky assumes that these will be filled with ‘default’ information
should a particular situation not provide specific information to fill them. Minsky further proposes
that once a frame is identified as being an appropriate situational representation, a ‘matching’
process follows in which an attempt is made to position information relating to the current
situation into the appropriate frame terminals. This process is driven in part by the default
information stored in the frame and in part by the current goals of the frame system in terms of
attempting to understand the current situation.

Minsky’s system-based use of the concept of frames allowed extended application of it to
computer-based system logic. Minsky’s work further provided the possibility that schemas could
be embedded within each other (what Martin (1994, p.272) refers to as recursiveness), although some
schemas must therefore be considered to be ‘atomic’ and unable to be reduced into other
schemas.

However, Schank & Abelson’s (1977) script theory differs from the concept of frames in that it
more specifically focuses on knowledge of procedural, time-based events. Crossover can be seen
here with previous definitions of ‘procedural’-type information in memory such as those provided
by Tulving (1973), Ruggiero and Flagg (1976), Oakley (1981), and Oberauer (2009, 2010) (Section
A.2.2).

The ‘restaurant script’ (Schank and Abelson, 1977, p.42) provides a practical example of this
theory. This script holds information about the stereotyped process of entering a restaurant,
being seated by a waiter, ordering food, eating, paying the bill and then leaving. Script theory
suggests that this same script is activated upon entering any establishment that is recognisably a
‘restaurant’. However, the script may not fit in all circumstances, such as in a fast-food restaurant
where it may be necessary to pay for the meal before receiving it. In these circumstances, the
script allows reasonable interpretation of the sequence of events making use of prior learned
experience and is updated to include such a situation for future reference in similar fast-food
restaurants.

Schemas also formed the basis for ‘story grammars’, knowledge frameworks proposed to aid in
the comprehension of stories, as studied in a range of work (Barsalou, 1982, 1983, 1985) and
applied by Piaget (1970) in his work on the development of the cognitive abilities of children as
they age. Despite the varying terminology and the small differences between these terms, each of
these iterations and advancements of the concept maintain the key underlying argument. That is,
that humans categorise their experiences into thematic mental frameworks, using observed
contiguity between concepts along with their spatiotemporal arrangements, which serve to inform
future life experiences and decision making through the application of pre-acquired and stored
knowledge.

However, the properties of a schema are difficult to categorically state, especially when the range
of research mentioned previously each makes small changes or additions to what a schema
conceptually ‘is’. However, by considering schemas at a general level, as per Rumelhart and Ortony’s (1977) proposed theory, a list of schema properties can be constructed that is applicable in a wide range of situations and that incorporates many of the psychological concepts that have been formed since Aristotle and Plato’s original work. Rumelhart and Ortony state that there are at least four essential characteristics of schemata, which combine to make them powerful for representing knowledge in memory. These are: (1) schemata have variables; (2) schemata can embed one within the other; (3) schemata represent generic concepts which, taken all together, vary in their levels of abstraction; (4) schemata represent knowledge, rather than definitions. (Rumelhart and Ortony, 1977, p.101)

The first property to note is that schemas consist of a number of variables or slots for information. These slots are linked together through a network of relations. The information within the variables may change to reflect the current context; however the relations between variables will always remain constant. This concept is closely related to Minsky’s ‘terminals’ (1974).

![Figure A.11: The ‘Give’ schema, following Rumelhart and Ortony (1977, p.102)](image)

An example using the schema for the word ‘Give’ (Figure A.11) shows that the ‘Give’ schema contains three variables, x, y and z. The agent (x) causes the ‘Get’ event in which the gift/object (z) is received by the recipient (y). The variables can contain any number of different values, but the relations remain constant. Variable (x) always causes the ‘Get’ event, (x) always gives the gift/object (z) and this is always received by (y).

Variables have some restrictions however pertaining to the types of information that can be placed in them. The giver (x) must have something to give and must be able to give it. Conversely, the gift/object (z) may not always necessarily be physical, nor may it always be something that could be considered a ‘gift’ (such as in the phrase, “giving someone a hard time”). However the relationships between the three variables remain constant and the schema is still capable of being used in order understand such a statement.
However, the values contained in some variables will automatically place updated restrictions on the possible contents of other variable slots within the schema. Also, if a situation does not provide information that can be placed into each variable, the known variable restrictions, coupled with prior experience of similar situations make it possible to infer appropriate default values in the empty variable slots. For example, in the sentence “John received the parcel”, there is not enough information to assign a value to the giver variable (x). However, with the known restrictions noted above, it is possible to infer that John received the parcel most probably from another person able to give it. One might then consider their separate ‘parcel’ schema, and find that parcels are often delivered by postmen or couriers and so make an informed guess that the giver variable may contain one of these two possible values. These ‘give’ and ‘parcel’ schemas demonstrate an activation process very similar to spreading activation theory (Section A.2.3.2), with different schemas being referred to when they are activated as they become variable contents in other schemas.

This differs slightly in relation to a schema-based construction of knowledge however, in that rather than always spreading ‘outwards’ through a network of linked nodes, the spreading of activation may also occur in an ‘inwards’ fashion through the embedding of schemas within schemas (Rumelhart and Ortony, 1977, p.106-109). For example, the schema for ‘Human Body’ will contain variables for such components as ‘Arm’, ‘Head’, and ‘Foot’, and in turn the ‘Head’ schema will contain variables for its own components, such as ‘Eye’, ‘Nose’ and ‘Mouth’.

The embedding of schema in this way aids in the grouping of information and the understanding of information. Information is grouped based on spatiotemporal factors, with the currently active schemas providing context. This is analogous to two trees planted next to each other, where the roots of each tree would be closer in space to each other than either set of roots would be to their respective tree’s branches. However, one would likely associate the roots of one tree with the branches of the same tree, because they are both variables within the single ‘Tree’ schema. “Objects are grouped together, but only on the basis of the schemata which are being employed in their interpretation” (Rumelhart and Ortony, 1977, p.109, emphasis original)

A schema is capable of storing information about concepts that are further abstracted from physical reality than the concepts discussed previously in relation to networks, which is one of the main differentiating properties of schema theory over network theories. Furthermore, while concepts within semantic networks are associated with their definitional properties (a bird has wings, a bird can fly, for example), schemas instead “represent knowledge associated with concepts” (Rumelhart and Ortony, 1977, p.111). Schemas allow memory and recall to be viewed as knowledge networks rather than definition networks. This further allows the consideration of the process of forming meaning as an active one, requiring logical thought through the linking of appropriate schematic information and the inferring of missing information where required.
Applied to the disruptive game design philosophy (Section 1.3), an associative knowledge network requiring active formation of meaning through thought provides a structure that is more open to ‘disruption’. Where a definitional network may only practically be disrupted by the disruption of concept properties (e.g. by presenting players with ‘atypical’ concepts), an associative network may be disrupted both in terms of the network’s contents as well as the thought processes required to interpret and form meaning from the network. In Oberauer’s (2009, 2010) model of working memory (Figure A.8), this would suggest it is possible to disrupt both declarative and procedural working memory as they will be drawing upon the associative network of long-term knowledge.

Comparing the above features of schemas with Anderson and Pearson’s (1984, p.272) later summary of schema features below, it is possible to see these same properties being noted, along with the addition of properties relating to their self-modification and self-updating:

- They are abstract, containing summarised information rather than specific details about specific instances or cases.
- They are structured, representing relations between component information.
- They are dynamic, changing, adapting and updating as information is received, as well as interacting with each other in order to combine information from multiple schemas in order to understand new situations.
- They provide context, allowing individuals to interpret similar information in different scenarios as having potentially different meaning.
- They organise experiences and modify themselves to accommodate new experiences.

Anderson and Pearson’s properties of schemas retain Rumelhart and Ortony’s abstraction, relational structure and contextualisation, and also note that schemas update themselves every time relevant knowledge is acquired. For example, if a variable has certain restrictions governing what information is valid within it, they will be updated if an event demonstrates that those restrictions are wrong.

While it is possible to describe what a schema is and how it is structured and made use of, the processes through which an individual acquires and updates these schemas is also necessary to consider for the disruptive game design philosophy (Section 1.3). To disrupt the expectations of the player, it is necessary to understand not only what knowledge a player has informing expectations and how they are stored but also how those expectations change over time. The process is thus as important as the structure for disruption.

Existing theories, such as Schank and Abelson’s (1977) work, suggest a simple process whereby the first time an individual encounters a new situation, they attempt to make sense of it using whatever relatable knowledge they have available, as they have no prior directly relevant knowledge from which to form an understanding of the situation. Later experiences of similar experiences will be compared to this first experience (and so forth with subsequent experiences)
and an abstracted schema will be formed containing the summarised form of that type of experience.

Induction of schematic knowledge from multiple experiences in this way and the application of that schematic knowledge through analogous problem solving can be seen in Gick and Holyoak (1983), who demonstrated the construction of an abstracted ‘problem schema’ from either one or two ‘story analogs’ and the further application of that abstracted schema to the solving of analogous but non-identical problems.

Their results indicated that when only a single story analog was used (i.e. the participants were only shown a single application of a problem-solving method), participants were regularly unsuccessful in applying schematised knowledge in finding a solution. However, two story analogs could “be mapped together to derive a more general schema” (p.32), which was more successful. Furthermore, Gick and Holyoak note that “any device that highlights the causally relevant correspondences will facilitate abstraction of a more optimal schema” (p.32), suggesting that when the analogy is made less abstract, the abstracted schema becomes more concise and thus, potentially more optimal for analogous problem solving.

This process of induction by analogy can be viewed as the replacing of certain values within variables, as previously described by Minsky (1974) as information ‘slots’ or ‘terminals’ and in Rumelhart and Ortony’s (1977) definition of schema. The analogy method of schema induction has however only a limited body of empirical research supporting it, such as Gick and Holyoak (1983), Gentner (1983), Keane (1988), and Gentner, Lowenstein and Thompson (2003). While such evidence presents a convincing argument for the accuracy of this method of schema acquisition and adaptation, it may have unknown limitations.

An alternative approach (Rumelhart and Norman, 1976, 1981) however, suggests that there are three ‘modes’ of learning in the forms of accretion, tuning and restructuring. In identifying and placing separations between these modes, their work offers a more quantifiable basis for understanding how individuals may acquire and store knowledge. Rumelhart and Norman’s proposed modes share some similarities with Piaget’s (1964, 1967, 1970) development process. Piaget posits a cycle of moving from a mental state of equilibrium (existing knowledge is able to make sense of the world), to one of disequilibrium (a new situation is encountered that existing knowledge cannot make sense of), on to either accommodation of the new information into a new schema or the assimilation of the information into an existing schema, depending on how closely related the new experience is to existing knowledge. While Piaget’s theory is situated within the field of developmental psychology, Rumelhart and Norman’s modes are applicable on a more general basis.

The first of these modes, accretion is “the normal kind of fact learning [. . .] your knowledge is merely incremented by a new set of facts” (Rumelhart and Norman, 1976, p.3). This information is incorporated into an existing schema (in the same manner as Piaget suggests assimilation
occurs), without changing the structure or organisation of that existing schema. For example, in the ‘Give’ schema (Rumelhart and Ortony, 1977, p.102), accretion may occur when an individual learns new possible values for the existing variable slots. These new values do not affect the individual’s understanding of the schema structure or the meaning of the interactions between variables. They instead merely expand the applicability of the ‘Give’ schema to include a new variable value.

The second of these modes is tuning, which occurs when information cannot be meaningfully stored within existing schema and a structural change must occur. For example, a modification of the ‘Give’ schema in this fashion may occur when a situation is encountered in which a given gift is not received (e.g. if a parcel were to be lost in the post). This would lead to the modification of the schema’s structure to the updated form in which giving does not necessarily cause the act of receiving. It is this structural modification that is the main difference between Rumelhart and Ortony’s (Rumelhart and Norman, 1976, 1981) work and Piaget’s (1964, 1967, 1970) work.

The third mode, restructuring, occurs when the new information or experience is unique, and cannot be seen to be related closely enough to existing schema to allow schema tuning to occur. In this situation, a new schema is formed (in the same manner as Piaget suggests accommodation occurs) to store the new knowledge, and this in turn can then be used as a basis for further accretion or tuning in response to future experiences. For example, in the case of the recipient in the ‘Give’ schema taking the gift/object without it being willingly given may maintain the ‘Give’ schema’s structure but significantly changes the meaning. This may require the creation of a new ‘Theft’ schema.

Schema theory provides a means of understanding how knowledge construction operates. This can be combined with the suggested structure of semantic network theory and the activation process of spreading activation theory to provide a means of understanding knowledge selection and recall. However, if these are applied alongside the previously defined three memory types (Section A.2.2) it is evident that the differences in the types of knowledge being stored (i.e. processes in procedural memory, concepts and facts in semantic memory and personal experiences in episodic memory) require slightly different approaches to organisation. Thus, before disruption (Section 1.3) can be explored fully, it is first necessary to consider these different approaches in more detail.

**A.2.4: Knowledge Organisation in Procedural, Semantic and Episodic Memory**

Schema theory, as an overarching framework for knowledge organisation operates well within all three memory types, however the differences in how they operate in each memory type need to be considered.
A.2.4.1: Schematic Organisation of Procedural Knowledge

Schemas in procedural memory can be conceived as serialised, predominantly linear processes with associated actions. The work of Turner and Turner (1991) and Turner (1994) suggests that procedural schemas, or p-schemas, describe a pattern of actions that the agent [e.g. the player] can perform to achieve a particular goal. A p-schema is a hierarchical structure. Its steps can suggest primitive actions to take, but they can also suggest other p-schemas to use as subschemas, or they can suggest subgoals, which may in turn be achieved using other p-schemas or primitive actions [...]. Though some p-schemas are quite comprehensive in their effects [...], most p-schemas have restricted scope, usually just the achievement of a single, limited goal. (Turner, 1994, p. 27)

Further to this definition, a procedural schema may also be interrupted and resumed, which is an important feature in allowing appropriate responses to be taken to a dynamic and changing situation, such as a game.

![Figure A.12: A simple procedural schema for making Mario enter a pipe and the resulting in-game events.](image)

A simple game-based procedural schema (Figure A.12) can be constructed for achieving the goal of making Mario enter a pipe in Super Mario Bros. (Nintendo R&D4, 1985). The high-level, goal oriented sequence contains steps that describe in-game processes, with attached lower-level actions that can be performed by a player in the real world. This example also demonstrates the potential for rearrangement of procedural schema stages, as the first two stages of moving and jumping could be switched while still achieving the goal of landing on the pipe.

Procedural schemas contain only process and action information, as per the definition of procedural memory contents (Section A.2.2). They require context in order to be utilised in a timely manner. Thus, outside of the context of playing Super Mario Bros. the procedural schema (Figure A.12) would be relatively meaningless. Turner (1994) suggests that procedural schemas require both contextual schemas and strategic schemas to be utilised in conjunction with procedural schemas so as to allow an appropriate response to a situation. Contextual schemas contain knowledge derived from past experiences that assist in informing an appropriate response, while strategic schemas consist of more abstract, high-level strategies for approaching particular types of challenge, such as operating under time pressure or in an unfamiliar context (Turner, 1994, p.23). Within the three memory type structure previously defined (Section A.2.2), such contextual and strategic schemas are suggested to draw on knowledge from long-term semantic...
(Section A.2.4.2) and episodic (Section A.2.4.3) memory. Semantic knowledge will provide conceptual and factual context (e.g. the properties of a pipe in *Super Mario Bros.*), while episodic knowledge will provide strategic information based on prior experience of performing the process in question.

Figure A.13: Adapted procedural schema for making Mario enter a pipe whilst avoiding an enemy.

Some pipes in *Super Mario Bros.* contain enemies (Piranha Plants) that move in and out of the pipes when Mario moves close to them. The semantic knowledge attached to these enemies (e.g. that they can damage or kill Mario) combined with previous experience with them (e.g. being killed by one when jumping onto a previous pipe) may modify the procedural schema, via what Rumelhart and Ortony termed *tuning*. An additional process, or subgoal, may be added to ensure the procedural schema allows for avoidance of the enemy threat (Figure A.13).

Disruptive game design (Section 1.3) may approach the disruption of procedural schemas in different ways, each with potentially differing ‘degrees’ of disruption. For example, a high degree of disruption may potentially be obtained by breaking the association between process and action (e.g. by changing the result of pressing a button on an input device, meaning the expected process is no longer performed by the action). This would require a new procedural schema to be formed (i.e. *restructuring*). A lesser degree of disruption may be potentially obtained by presenting players with situations that consistently vary the different ‘stages’, or ‘subgoals’ of a single schema, (e.g. by adding or removing hazards, or by changing the types of hazards thus requiring the certain stages of the procedural schema to be changed in different situations). This may require *tuning* of existing schema structures to occur rather than new procedural schemas to be formed, as well as also requiring more cognitive engagement (e.g. at the higher-order thinking levels of analysis and evaluation) to ensure the correct collection and sequence of process stages is being applied in the correct situation.

**A.2.4.2: Schematic Organisation of Semantic Knowledge**

Within semantic memory, a hierarchical model of knowledge organisation is proposed that utilises the structural principles of semantic network theory (Section A.2.3.1) and spreading activation theory (Section A.2.3.2) along with schema-based organisation.
A schema in semantic memory (Figure A.14) is simultaneously a store for nested abstract concepts and their properties (i.e. the nested coloured boxes in Figure A.14), as well as specific knowledge regarding concept instances of abstract concepts (Casson, 1983, Cohen and Murphy, 1984, Komatsu, 1992). For example, the concept *bird* may contain a number of concept instances, such as *sparrow*, *owl* or *ostrich*. This suggests a structure that is similar to previous definitions of schema (Anderson and Pearson, 1984, Rumelhart and Ortony, 1977) but also additionally allows non-abstract concept instances (as would be traditionally associated with a semantic network structure) to exist within the same structure as abstract concepts.

The high-level, abstract schema (of which there will be many stored in semantic memory) contains a selection of lower level, less abstract schemas, which may be a number of ‘layers’ deep, becoming less abstract and more defined as depth increases. Each hierarchical layer inherits the properties of the layers above it and then adds additional properties or refines existing properties to become more precise. At the lowest, least abstract hierarchical layer, concept instances represent real-world (or in the case of games, real-virtual) instantiations.

Activation, or recall, of knowledge within this structure can be viewed as a process of spreading activation between schemas. A property of a particular concept instance may itself be stored elsewhere as a more abstract schema. Thus, the recall of information about a concept instance will also cause the recall of properties of that concept instance as standalone high-level schemas. These in turn may cause further activation until the activation strength weakens as the distance from the original activation source increases.
Figure A.15: Example semantic schema structure for the 'Firearm' concept.

An example of this structure and activation process in games could be the concept of a 'Firearm'. A 'Firearm' schema (Figure A.15) may exist as a relatively abstract concept, embedded within the more abstracted 'Weapons' and 'Ranged Weapons' schemas. The 'Firearm' schema may contain abstract properties such as 'has a trigger', 'requires ammunition' and 'requires aiming skill', along with specific concept instances that fit the abstract schematic definition, such as 'Walther P99', '.44 Magnum' or 'Laser Rifle'. These, in turn would have instance-specific properties attached to them, such as the 'Laser Rifle' being 'fictional' (a property which itself may be stored elsewhere in memory as a high-level schema) and also 'requires recharging'. These specific properties may operate in addition to the inherited abstract properties within the schema by providing instance specific detailed information on a property, such as the specific type of ammunition that is required. They may also temporarily overwrite them such as the 'requires recharging' property overwriting the 'requires ammunition' property for the particular concept instance of the 'Laser Rifle'.

This structure is able to handle 'non-physical' concepts as well as 'physical' (or virtually 'physical') object-based concepts such as 'Firearms'. An abstract 'game' schema for example contains factual knowledge that can be applied to all types of games (Section A.2.3.2). This may have embedded
within it less abstracted (but still high-level) schema relating to 'digital games', 'board games', 'live-action role-playing games' and so forth. Then, specific concept instances of these types of game will be contained within these sub-schemas (e.g. Monopoly or Snakes and Ladders may be concept instances within the 'board games' schema).

This suggested structure combines the benefits of both schema theory and semantic network theory and also accounts for a key weakness of semantic networks (Section A.2.3.2) in their inability to handle abstract concepts (Wittgenstein, 1958).

Disruptive game design (Section 1.3) may, as was the case with procedural knowledge, handle disruption of semantic knowledge in different ways. For example, disruption may be achieved by presenting players with concepts for which they have no existing schema-based knowledge to draw upon in order to understand, thus requiring tuning of an existing schema or restructuring to create a new schema. It is unlikely however that a player may be presented with a concept for which they have absolutely no existing knowledge that can allow them to form at least a basic understanding of it. For example, a player may be able to at least identify a concept as being broadly 'physical' or 'non-physical', or belonging to a very high-level abstract category, such as 'organic' or 'non-organic'. That is, it is likely that player's will be able to use of a process of accretion to initially form a fundamental understanding of the concept, although may then need to engage in tuning or restructuring to develop a more comprehensive understanding.

It is more likely that a disruptive game design philosophy may be able to approach design with the aim of presenting consistently atypical or 'non-static' concepts to players. This would require heightened cognitive engagement to recall information about the atypical concepts but may also encourage more frequent attempt at schema tuning. In this context, 'non-static' may refer to concepts that do not retain consistent properties during gameplay and thus, are difficult to accurately organise into a semantic schema network. For example, a firearm that changes its shooting and recoil properties at random when it is used by the player, thus making properties such as 'accuracy' difficult to predict.

A.2.4.3: Schematic Organisation of Episodic Knowledge

Episodic memories are 'snapshots' of lived experiences, spatiotemporally organised and contextualised. Schemas within episodic memories can be conceived as individual episodes, each of which may be recalled as a high-level schema (in the same way as a high-level schema in semantic memory may function). Each episode will contain a combination of multimodal knowledge (e.g. visual, auditory, olfactory, etc.) related to a particular lived experience. The recall of an episode will then activate (via spreading activation) memory of this multimodal knowledge within the other memory stores (i.e. procedural and semantic).

The construction of individual episodes is influenced by existing schema-based knowledge at the time of construction. Existing knowledge contributes to the content of an episode by being 'baked-in' to the episode, as supported by the work of Brewer and Treyens (1981). This work
suggests that not only will existing knowledge be incorporated into an episode, but existing knowledge will also influence what information about the current experience is selected for inclusion in the episode and is thus available for later recall. For example, as Pezdek et al. (1989) identify, there are potential differences in encoding processes for information that is ‘out of place’ or ‘novel’, compared to information that is in-keeping with an individual’s expectations. Existing knowledge influences expectations and thus may influence construction of episodes in memory.

Because episodic memory may be influenced by existing knowledge it is prone to recall errors and inaccuracies. These errors may be introduced through the recall of ‘baked-in’ information that may be schema-relevant but not episode relevant. For example, an experience of being in an office may be formed into an episode in memory. This episode is informed by existing knowledge of what is commonly found in an office environment (the individual’s ‘office’ schema). Later recall of this episode recalls the ‘baked-in’, schema-based information as well, leading to possible errors such as recalling that there was a computer present in the particular office in question (because computers are commonly associated with office environments) when in fact there was not. It is errors such as this that have driven a large amount of research into the accuracy of eyewitness testimony and the reliability of memory in legal proceedings (Howe, 2013, Loftus, 1981).

Episodic memory as separate from semantic memory and procedural memory is in some ways problematic due to the multimodal, combined, nature of episodes. McKoon, Ratcliff and Dell (1986) suggest that the clear interdependence of episodic memory and other memory types means viewing them as entirely separate may not be an accurate interpretation. However, as Menon (2002) argues, while the interdependence is necessary for the purpose of content, the functionality of each type of memory is notably different. This, combined with the differing structures described (Section A.2.4.1, Section A.2.4.2, and Section A.2.4.3) supports the perception of episodic memory as at least in some ways a pragmatically separate memory type.

Figure A.16: Further adapted Systems Model of memory and recall adding representation of organisational structure to memory types.

With schematic organisation of each memory type identified, the Systems Model of memory and recall introduced in Section A.2.1 and further refined in Section A.2.2 can be refined once more to include representations of organisation within each long-term memory store (Figure A.16).
Procedural memory contains procedural schemas of varying lengths, while episodic memory contains bounded memory episodes. Semantic memory is organised in hierarchical sub-schemas which contain concept instances at their least abstract levels. The Visual Semantics and Language stores remain separate from this schematised structure to emphasise their specific role within the declarative working memory interaction process.

A.2.5: The Problems of Initial Schema Acquisition and Schema Flexibility

One of the criticisms of cognitive schema theory is that it does not describe the process of how initial schema acquisition occurs. The three modes of learning, accretion, tuning and restructuring (Section A.2.3.3) appear to function well in situations where initial schema-based knowledge exists to be used as a basis for comparison to incoming stimulus information. However, these schemas must be initially formed through some cognitive process and cognitive psychology as described thus far does not fully address this problem. The work of Gick and Holyoak (1983) (Section A.2.3.3) on schema acquisition via a process of induction provides a potential explanation, however further exploration of related work is needed to support this explanation.

Rumelhart, Smolensky, McClelland and Hinton (1999) state that:

If schemata are to work as a basis for models of cognitive processing, they must be very flexible objects – much more flexible than they really ever have been in any actual implementations [...] On the one hand, schemata are the structure of the mind. On the other hand, schemata must be sufficiently malleable to fit around most everything [...] How can we get a highly structured schema which is sufficiently rich to capture the regularities of a situation and to support the kinds of inferences that schemata are supposed to support and at the same time is sufficiently pliable to adapt to new situations and new configurations of events? (Rumelhart et al., 1999, p.20)

A proposed answer to this problem of flexibility and also to the issue of initial schema acquisition is a dynamic generation of schemas "at the moment they are needed" (Rumelhart et al., 1999, p.20). In Rumelhart et al.'s proposed theory, knowledge is stored in memory in a manner that resembles a concept network construction. When an input, or stimulus, is perceived, a subset of these interlinked concepts is activated. These interlinked concepts are connected through a "constraint satisfaction network" (Rumelhart et al., 1999, p.33) in which the network's link structure must adhere to particular cross-concept constraints. This can be seen to closely resemble Rumelhart and Ortony's (1977) prior work in which variable 'slots' in a schema have constraints that control their possible contents. The content of one variable may then place constraints on the possible contents of other linked variables.

Rumelhart et al.'s theory suggests that once the stimulus has triggered the initial state of the network, the network then moves towards one of the "goodness maxima" (Rumelhart et al., 1999, p.20). This can be conceived as the most accurate 'goodness of fit' between stored knowledge in the concept network and the perceived stimulus that has triggered that stored knowledge. This theory also successfully merges principles of spreading activation theory, suggesting that:
certain groups, or subpatterns of units tend to act in concert. They tend to activate one another and, when activated, tend to inhibit the same units. It is these coalitions of tightly interconnected units that correspond most closely to what have been called schemata. (Rumelhart et al., 1999, p.20)

The key difference between this connectionist approach to schema theory and the previously described cognitivist views of schema theory is in what is suggested to be permanently stored in memory and how this stored information is influenced through learning processes. Whereas cognitive schema theory suggests that clearly structured schemas are permanently stored in memory and recalled and updated in response to stimulus perception, connectionist schema theory suggests that the only permanently stored information is a network of concepts linked together with associative links of varying strength. Recall is controlled by the strength of connections between concepts, and learning has the ability to change the strengths of concept connections, either strengthening or weakening them or placing constraints on them in some way. However, while Rumelhart et al. (1999, p.21) suggest that "in [the] case [of this theory], nothing [that is] stored corresponds very closely to a schema" (emphasis original), they do note that "as the network is reorganised as a function of the structure of its inputs, it may come to respond in a more or less schema-like way" (Rumelhart et al., 1999, p.21)

This "schema-like" behaviour is an important point to consider. It could be suggested following this that the cognitivist suggestion of three different 'modes' of learning (accretion, tuning and restructuring) may function within long-term memory while moment-by-moment responses to stimuli are generated through a dynamic formation of a schema 'instance' as it is required.

If one considers that certain groups of the stored concept network may come to respond in a manner that is more or less "schema-like" in nature, then it should be possible to apply existing cognitivist schema theory alongside this connectionist approach. Accretion, for example, could be viewed as learning through making minimal adaptations to constraint weightings within an already formed group within the concept network. This is in line with Rumelhart and Norman's (1976) assertion that this type of learning is merely incrementing existing knowledge with new facts. Tuning could be conceived within connectionist theory as the linking of new concepts to an existing group, or the removal of concepts (through breaking existing links) from an existing group. Lastly, Restructuring could be conceived as the formation of whole new groups of concepts, which may include the formation of new individual concepts or simply the drawing together of a number of previously unconnected concepts.

Furthermore, if the definition of 'schema' provided by Anderson and Pearson (1984) and discussed in Section A.2.3.3 is considered in relation to 'concept groups', it is still an accurate description of their structure, purpose and behaviour. It could thus be suggested that schemas may exist, both as concept groups in long-term memory and as dynamic schema instances that are formed via induction (Gick and Holyoak, 1983) from these stored concept groups in relation to the current situational context.
Figure A.17: Further adapted model of memory and recall adding dynamic schema instance generation via induction as a process linked to the Central Executive of working memory.

Considering dynamic schema instance generation in relation to the previously described model of memory and recall (Section A.2.1, Section A.2.2, and Section A.2.4.3) it is possible to identify where such a process may occur. The process is active and relies on the combination of stored knowledge from long-term memory with incoming stimulus information. Thus it could be suggested that this process occurs alongside working memory, coordinated by the Central Executive (Figure A.17). Working memory initiates a search of long-term memory for stored knowledge (in each of procedural, semantic and episodic long-term memory) that is relevant to the currently perceived situation, and this knowledge is returned to the dynamic schema instance via induction. The dynamic schema instance can then be processed by the Central Executive as the means of guiding the body’s response to the stimulus.
Appendix B: Transfer of Knowledge from Game(s) Situations to Real World Situations

Figure B.1: Transfer of knowledge from a single game instance (Game 1), to other game instances (Game 2) and to Real World analogous situations.
Appendix C: Overview of HPL2 Engine Features with Potential Applications or Limitations for Disruptive Game Features

C.1: Coherent Hierarchical Culling System
HPL2 utilises a custom occlusion culling system to control what parts of the game environment are rendered. This may influence the size and complexity of game environments that can be created in the engine, compared to other currently available game engine technology.

C.2: Shadow Maps
HPL2 uses shadow maps, processed by the Graphics Processing Unit, or GPU. This potentially frees up CPU cycles that can allow for other systems to become more complex, such as artificial intelligence, or allowing a greater number of game entities (e.g. enemies, or physics-based objects) to be processed simultaneously.

C.3: AngelScript-based Application Programming Interface (API)
The main HPL2 source code is written in C++ (Stroustrup, 1983). HPL2 allows extended functionality through the use of an open-source Angelscript-based (Jönsson, 2003) Application Programming Interface (API). This allows a significant degree of programmer-independent scripting work to be carried out in terms of the implementation of moment-by-moment gameplay events, puzzles and scenarios. For prototyping of disruptive game features, the ability to work independently of the programmer to test game features during gameplay may be particularly beneficial. The API is however limited by what script support has been written into the engine itself by the programmer initially, as each script function must refer back to a method that is contained in the main engine source code. Thus, any completely new script functions required will still need programmer knowledge and time to implement before they can be used independently by the scripter.

C.4: Newton Game Dynamics Physics Engine
The HPL2 engine utilises the open-source Newton Game Dynamics physics API (Jerez and Suero, 2012). This provides a range of collision detection, rigid body physics and scene management functions. As with the AngelScript API, this provides flexibility through its open-source nature, but is limited by what is implemented at an engine level by the programmer.

C.5: HPL2 Developer Tools
While not created to the same standard in terms of usability and flexibility as more widely commercialised game engine technology (due to the independent nature of Frictional Games when the tools were developed and their intended original purpose of only being used in-house),
the HPL2 developer tools nevertheless provided potential for being able to rapidly prototype and implement disruptive game features. With bespoke editors for handling environments, lighting, materials, particle systems and game entities along with a number of options associated with sounds and music, there was scope to develop prototypes using a range of different game components. Moreover, the accessibility of the individual tools in the toolset meant that prototypes for many different game components could potentially be developed by individual team members without the need to hinder the development of the core game.
Appendix D: We Are The Pig Concept Pitch

The people eat the pigs. And when they are done, the pigs eat the people. And I watch it all, inside my wax egg, on the mantelpiece. Everyone comes to suckle. The machine is the pig, obviously. Catch the pig. Doll with trotters. Milk and money. A machine to make factories. A world eaten by the pig. A pig lays an egg, the egg is a future or a machine. I live inside an egg, laid by a pig, in a machine that makes milk and factories. Other things becoming.

We are the Pig is a short, surreal first-person horror game about dehumanization and delusion. Based around one large environment: a vast factory, the arc of the game is based around a central pivot. In the first part of the game, the player works to assemble, free and lubricate the giant engine at the centre of the factory. In the second, the nature of the engine is revealed and they work to destroy it. The game ends with the revelation that destroying the machine is as terrible a prospect as building it in the first place, trapping the player into an endless cycle.

The core mechanisms of the game are exploration and physics manipulation, but it is not a puzzle game at root. Gameplay is reduced to its simplest level, with focus resting primarily on engaging with the story and the world. Navigating, building and destroying the machine take the form of timed runs, object manipulation, item hunts, code breaking and sequenced actions, none of which should be difficult enough to break the flow of the story. We focus instead on the broken, contradictory and increasingly disturbing voice-overs, which are formed of a central monologue from an unnamed narrator (who may or may not be the player) and scattered conversations with several others.

As the player works and toils within the engine, fractured stories weave around the machines: a child born with the hands and feet of a pig; a factory that grew overnight from an egg in the middle of a marsh; the death of workers in an industrial fire; farm animals turning on each other to feed; a slaughterhouse worker’s attempt to free the bodies of the animals he has just killed…

And the machine provides its images: being chased through a structure by a headless pig corpse; pressurizing a giant udder to feed the engine; pistons crushing limbs into pulp that is fed into the furnace; a nest of eggs and money; dolls with trotters; buzzsaws and electric prods and the trepanator; deafening squeals and grunts and the grinding clatter of machinery.

We are the Pig will not be an easy game to play, instead holding the player through our inherent fascination with the darkness at the centre of the soul - but it is ultimately about the capacity to retain humanity even in a slaughterhouse of one’s own making.

This section contains the original game design document for the game. This document presents a very early version of the game and may be used as a comparison to the final version of the game as a means of seeing the evolution of the project from conception to completion. The material contained in this section is an accurate reproduction of the documentation used during the game’s development and thus is owned by The Chinese Room. This section’s content is written by The Chinese Room’s Writer and Creative Director, Dan Pinchbeck.

Frictional Games: Design Document

Author: D. Pinchbeck

Date: 30.6.11

*Amnesia:*

*A Machine for Pigs*
Summary

The year is 1883. Wealthy industrialist Oswald Mandus has returned home from a disastrous expedition to Mexico. Traumatised by the deaths of his wife and children, he has thrown himself into work for the last four months, constructing “the greatest engine of the age”, a vast complex above and below the London streets. The machine’s purpose is unknown. There are rumours of an unholy technology, of deaths and misadventures, of screams from far below the ground. The stench of stale blood is always upon him.

The game starts with Oswald awakening from a fever, and pursuing what he believes to be the ghosts of his children. He follows their voices through his manor, through a strange airlock structure into a chapel that also seems to function as the entrance to a huge abattoir. Mandus realises he has constructed this machine and it is vast; it also seems connected to his expedition to Mexico. The structure is filthy and swamped with lethal infectants, it also seems to be occupied by a degenerate form of man-pig.

As Mandus travels through the machine, he begins to realise that the disease he is so susceptible to is interwoven with the machine’s purpose. A new voice begins to speak to him, claiming to be a god. Although he initially writes this off as his delusion, Mandus comes to realise that there is indeed another presence in the machine, and it is The Machine itself. This machine intelligence needs Mandus’ help, to flush the infectants from its system and restore purity. Mandus realises that this is the fundamental purpose of the entire machine he has built.

Around this point however, his travels through the machine lead him back above ground into a Hotel where it seems people are being fattened and fed into the machine: that it is designed for the mass slaughter of humans, not animals. Mandus realises that he has brought something else back from Mexico with him, something inhuman, with horrendous designs for London’s population. The machine, believing itself to be a trapped God, requires a massive sacrifice to wipe the filth of humanity from the world and restore its purity and power. It also boasts it has manipulated Mandus not only to construct its body, but then also to sacrifice his children to keep it alive, and then to journey back into it, diverting the filth and disease and building its power.

Mandus realises he must destroy the machine, and travels deeper, finding a steam-powered neural network that forms the machine’s brain. He crosses this Tesla field, and leaps into a chasm at the centre, travelling past the sewer systems used to keep the central processor cool, arriving at the machine’s soul, a pyramidal structure deep underground. At the summit of the pyramid, he finds a strange orb that seems to give the machine its sentience. Determined to destroy it, he uses blasting charges to bring down the cavern roof, breaching London’s sewers into the machine and drowning them both in shit and filth.
HUD + INVENTORY

Amnesia: A Machine for Pigs retains Amnesia’s inventory screen, with a slight re-design. Oil is retained, as is the lantern, and so are tinderboxes for candles. The health counter is retained with a reskinned heart graphic. The sanity meter is replaced with the Infection meter, with a graphic of a pair of lungs, altering like health according to the level. The notebook is also retained. Fewer inventory slots will be needed.

INFECTION EVENTS

Rather than sanity, A Machine for Pigs uses Infection events as a mechanic. Monsters are infectious, rather than inflicting damage directly. Parts of the environment are also infectious.

Like Sanity, INFECTION is staged on x levels:

LEVEL 0: Default – no adverse effects


LEVEL 2: Moderate infection. Player takes a health hit every couple of minutes. Every time this happens, they stop and vomit/cough. More coughing and hoarse breathing. Mild visual distortions and overlays.

LEVEL 3: High infection. Heavy distortions, heavy breathing. Movement impaired (wobble walk). Frequent coughing fits and vomiting. Health hits every minute.


Infections are counteracted by Laudanum and Penicillin (not named as this, but it had precursors going back to 1875!). These are spawned if needed on level loads – in other words, they are there if needed to keep the player going through the game, but player cannot stockpile them. Also some Infection events are capped at the level they can take the player to, so even a poorly playing player will not die at environment infection.

Penicillin – reduces infection by two levels

Laudanum – reduces infection by one level

HEALTH

LEVEL 0: Just dandy – default.

LEVEL 1: Battered but alert – no adverse effects

LEVEL 2: A serious injury – slight visual wobbles.

LEVEL 3: Critically injured – red mist…

Health hits are counteracted by bandages.
MONSTERS

The basic monster is the Manpig, a hideous splice of man and beast. They are highly infectious – each hit pushes the player up another infection level.

Design Walkthrough with maps

1. Mansion

Straightforward exploration, no entities. The player begins in an adult’s bedroom, on the floor, by a four-poster bed that has been caged. This leads to a landing, with a staircase going up to children’s rooms. In the nursery, a model of an Aztec temple complex. The bathroom on the master bedroom floor has a bath full of blood. In a second bedroom, a view out over London. A second hall leads to a split-level entrance hall, at the bottom of which we can follow another corridor to a greenhouse. All the plants are dead, diseased. A door at the back leads to an alley.

Nothing occurs in this section, apart from voice-overs. It’s a tension building exercise, there’s no gameplay as such. The mansion is richly furnished but dark, grimy. Colours are deep: reds and greens of furnishings, wood panelling, portraits and furniture. The children’s rooms are super-dark but we can take a lantern from Bedroom 2 and light our way up there. Candles can be lit around the place too. In the nursery, a rocking horse rocks in the dark until we light it: then it stops.

The action is driven by us pursuing the ‘ghosts’ of children (audio triggers) whilst we are orientated to the basic story of Mandus, his expedition, the death of his children, his return and his collapse into fever.
EVENTS:

OnStart – visual distortions, fixed view rise from bed

Leaving bedroom, childrens voices and footsteps up to nursery – door slams shut to it.

Door on right slams and locks

Lights go out in nursery – voices “Sshh. He’s coming, he’s coming.”

Rockinghorse is going in the dark as we approach. Then the candles light and it stops.

Visual distortion on look at map of Aztec. Machine voice low “Mandus....”

In bedroom with cot, behind player child footsteps again running away.

End door slams again, player approaches “You can’t come in”. Crash of furniture smashing around the room into door.

Bathroom – flash/disort. Seconds of bath filled with blood, saw and implements on the bathroom floor. VO: Child laughing. Door slams somewhere below the player.

In cupboard in bathroom is laudanum


Third bedroom: music swell at skybox view out of the window.

All the lights go out with big crashing below us.

On approach to landing door, it swings open for us.

Both doors below in hall slam shut, laughter and footsteps.

Office ground floor: TB and note. Visual shudder on pick-up.

On approach to greenhouse, sound of pig squealing and smashing glass furniture. It’s a mess when we get there. There’s a pig body lying in broken pile of shit. The greenhouse door at the far end shuts as we approach. “Come on papa, this way!”

2. Cellar / Chapel

Brown and yellow tone to this cellar area that contains crates and barrels. It introduces the disease mechanic – because sooner or later we have to become infected by the tunnel towards the coal cellar. First we have to climb onto a split-level walkway and pull a lever to release the block holding the barrels that impede our path. These roll away and we can descend a short ramp into the second cellar section. In here, another split barrel leaks infectants and we cannot pass. However, if we collect a rock from the coal cellar, we can break the support of a walkway,
dropping it down at an angle so we can reach another lever and drop the grille out from under the split barrel. This gives us access to a third cellar, with steps to an office. Inside the office is medicine to beat back the infection, and a strange airlock. Unlike the brown/yellow cellars, this is metal, smooth, hydraulic. There are strange markings, based on Mayan pictograms.

Beyond this airlock is an alley leading to a chapel. Normal in most regards but it is caked in blood and filth. There are disturbing stained glass windows in the style of Mayan/Aztec stela (a cycle of a man, a machine and a pig entwined, and other scenes of slaughter).

The altar area is barred off. To get to it, we need to find a cross, which, if we look closely, has a pig jesus on it, and insert it into the altar. The bars slide up and we have access to the priest hole and bell tower. Pulling a bell opens a slate in the priest hole. We know we want to go down there because when we are close to it we can hear voices.

It’s very dark down in the hole, but we seem to be in a large room full of cages, creating a mazelike path. Things move and make unpleasant noises in the dark, in the cages. There are patches of low-level infection around, and then something slams against the bars near us and our infection levels shoot up (establishing monsters=infected). After we fell, the priest hole door slammed shut behind us so the only way is forwards, and at the far side of the holding pens, we see the start of a conveyor belt. A huge smear of blood along the belt draws us in.

From the chapel on, things have gone very quiet – no ghosts call anymore. Light and colour shift from the normal, mundane brown hue of the cellar to a colder blue of the chapel, with vivid reds on the stained glass. The holding pens are extremely dark – lit only by our lantern (which we collect from the Office).
EVENTS:

Door to cellar is locked. When the player turns their back on it, it unlocks and opens.

Entering cellar – visual overlays, diseases and filth everywhere, distant pigs squealing, machine VO: “Mandus....”

In between blocking barrels, the player can see a manpig move across the cellar (delete after this event)

INFECTION event in second cellar. Player infected to level 2 as soon as they come down the stairs. (patches of water here from broken barrels)

Body of worker. Steal lantern and use disinfectant (he has 2 batches) on self to get rid of infection. Avoiding infective water, to get back out.

Lever at end of walkway releases ground wedge and rolls barrels out of the way down into cellar 2.

INFECTION event: barrel over grill is broken and seeping. If the player is at level 2 of infection or worse, spawn a disinfectant on the overhead walkway.

Lever missing from slot on walkway, it’s in the bottom room.

Candles light in front of the player on the way down the stairs.

Visual shudder on re-emergence.
Lever causes grill to open, drops barrel into gunk, grille closes again, “Mandus….”

Draw in office desk full of glasses – visual distortion, SFX

In alley, each corner children running “this way! This way” We've found something! Come and see!”

Church doors open, wind sweeps in and knocks over candlesticks, sends bible papers flying.

On altar, Lantern oil, medicine and disinfectant.

Player must ring bell (loud, distortions) to slide open secret wall behind altar (priest hole) – drops through floor to cages.

No lights down here. Beasts bang against cages. Factory key is somewhere in the maze, needed to open final door

3. Factory

We are on a dark conveyor belt, sloping downwards. There are occasional walkways and locked doors to either side. Along the ceiling, a wire runs with meat hooks. There is filth and gore everywhere. Mandus’ VO talks of creating this abattoir, taking advantage of the most modern industrial techniques to vastly increase its processing capacity. We hear The Machine for the first time, and the ghosts. It is a fusion of Victorian walls and these odd instances of more modern tech, conveyors and hooks. We pass a set of mechanised knife arms, hanging limply. Behind these is a control room, wheels and dials that resemble the inside of a Jules Verne bathysphere. Plus armchairs and archaic consoles.
Around a corner, the pigline suddenly enters a much bigger room, dropping away three storeys. Ahead of us, the line crosses the room and ends in a grille. Mandus observes that the filth must be clogging the machine, and he should clear it away to get the engine working again, to open the doors. We see a ghost behind the grille. We can drop down onto a caged roof below us and see three huge pistons inactive in caged areas. Rickety wooden maintenance steps lead down. Here, as elsewhere, we have this strange fusion: wood and metal, archaic and modern. There’s a door in the far corner we could reach by dropping into a piston cage and crossing it, but there’s a beast inside and it’s foully infective. Instead, the only way out of the room is through a hole in another piston cage’s floor. As we pass through, The Machine begins to speak to us properly for the first time. It is sick, dying, drowning in shit. It needs to be clean. It begs for our help.

We drop into cramped pipes. Concrete, grey, slimy. Potentially water at points. Very dark – we will need our lantern. The pipes slope down and down. At points off the main path, there are views onto rooms below. In one, we see a four-poster bed in an opulent bedroom with a man fucking a pig on it. This room is identical to the ones we find later in 07Fattening. In another, we see a freezer with human carcasses hanging from meat hooks, also a scene from 07Fattening. In another, a man with a tube connecting to the stump of his neck. Strange lightning flows along the tube. Eventually the pipe tips sharply downwards and we fall into the sewers.
EVENTS:

Office by line at start: more lantern oil, medicine + dis (spawned if needed)

Entering pig line- big flashback distortions, sounds of squealing, buzzsaws, etc.

Passing first blade section, more distortions “Papa, come and see! What is it Papa?”

Factory this time is covered in filth and shit and rust. VO Machine: “Filthy…. so unclean…”

Mandus VO about cleaning it to start the machine again. Children” “Again! Again! Make it go again Papa!”

Manpig in piston cages – angry, trapped, beating on the bars. Big visual hit (no sanity meter).

Drop into office, massive crash and bang from down below. (has the pig escaped fear shock!)

In office, potential spawn point for supplies

Machine VO’s start here.

At foot of stairs, manpig hurls itself against the bars trying to get at us. On proximity, fire INFECTION event.

As player moves away towards well corner, big SFX of it hitting the bars behind them.

Machine lights gaslamps to lead player towards the well.

Particles up from well?
At end sections of pipesystem, more lamp oil spawns. Windows into rooms with (animated)
sequences.

Caged four poster with pig fucking a corpse on it.

As we approach “no, no, no … arg!” then we see a freezer with bodies hanging from hooks, one
twitching and bleeding

Corpses on tables without heads and blue lightning tubes

4. Sewers

If the cellars were yellow, the church blue and the factory brown and grey, the sewers are green: old, fucked stonework, pure Victoriana. A central trough with a walkway on either side. Some areas have water, others are dry. There are three levels to the sewers. A beast roams this place, we will need to avoid it to avoid being infected. Off the sewer somewhere is a room where something has made it’s nest. If we become too infected we pass out and recover here (there is always one room per map designated as the recovery area – always a nest). The sewers are relatively simple and big, giving us time to deliver big plot VOs: Mandus’ discovery of an Orb in Mexico in the temple complexes, and The Machine explaining how Mandus must divert the flow of effluent into the boiler where it can be burnt off and used as fuel, as well as stopping it infecting the engine. Close ties between Mandus and The Machine, both are susceptible to infection.
At the end of the first level of sewers is a large underground reservoir, lethally infectious. Above us we can see a bridge and above that a ledge. Working our way up to the ledge via the bridge and two more levels of sewer tunnels, we find the Flush Controls, and divert the flow of shit in the sewers. With a great rush, we hear water moving. Returning, the water levels have dropped significantly, opening up another bridge and a new tunnel leading off. These climb upwards, and we follow them until we reach a maintenance hatch and a ladder. This joins another series of gore-streaked pipes, which are less sewage than offal disposal, and as we climb up, we hear snatches of conversations between Mandus and another man, discussing the architecture of the plant and it’s purpose. Mandus is cagey: “There are surely not enough pigs in London to feed the hunger of this machine!” “It all depends on what one classifies as a pig, my dear Dr Olafsen”.

We emerge in an exterior yard, London skyline, fog and smog. Ahead is a brick building (the yard is enclosed with fences, we cannot leave any other way). Inside the building, we can see an offal chute leading back down the way we came, and conveyor belts leading into blackened ovens. It is part kitchen, part crematorium. Wagons of unidentifiable waste. Off this, an office. The desk drawers are full of teeth and glasses. Another door leads us to a maintenance corridor, again, turning from wood panelling to stone. We slope downwards again, until we reach an elevator, which is the only option available to us. We go down.
EVENTS:

Variable water level: Starts on map. Where the busts are, are sluice controls.

INFECTION event – all the water is hugely infective. Occasional bodies float along it.

Manpig / maybe 2 on pathways around the level.

Doors added into junctions. At least one door slam/lock on approach – or bang against it on approach.

Sluice out on ledge by big tank. Distortion – effect – shift sewage to blood? – pieces of meat rain down from up above in the darkness?

At base of stairs leading to upper level stairs – bodies float down stream along the sewer – one suddenly moves and screams.

Upper level: player will be expecting another pig, so can play audio hell with them – also maybe a shadow, or see a pig from a distance vanish around a corner.

2nd upper sluice control – water level drops.

Control room – huge rush of water dropping, loose all the water down plug hole in large tank. – seal off huge doors at water edges of the map.

Tank drop and climb- lots of VOs. Hallucination of Aztec temple.

INFECTION event: automatically take the player to level 2 unless they are there already

Ovens – VO on approach, cut as player enters room

Heavy distortions + VOs in the oven room. Spawn point for medicines in the office on one of the drawers. Shudders on seeing the glasses and teeth.

5. Boiler

The boiler is based around one large chamber, in which a huge metal tank, creaking and drooling green pus and bile stands. A staircase runs around the tank, leading to four separate levels. These stairs and walkways are metal and wooden and don’t look that safe. Pieces can fall off them as we move about. At the top, they join a stone bridge that looks like part of the sewer complex, it leads to a grille like the Factory’s, closed off. The chamber the boiler is in is also stone, sewer like, though the boiler is metal, out of place. We enter on the second-highest floor, into a storeroom.
The next floor down is ignition control, a control panel tells us this, but also has a board of 16 lights, two of the red, fourteen of them green, and the ignition key. Near to this is a fuel rod cap (this is a highly infective item and we need to keep tabs on our medical supplies - they are littered around the boiler). We need to collect another of these from the storeroom on the level below, then continue to the base of the boiler, past the closed door to Rod Control, and under the boiler itself, past the huge burners and into a small corridor that leads to the weird fuel rod tank. Here, the tech is suddenly nuclear, totally out of place. Clean blue water, massive infections. There are sixteen rods in the water, two are missing caps. The ‘solution’ is obvious. Once we’ve placed these, we are so infectious we blackout automatically and have a vision: it’s the pyramid at the end of the game, and we get The Machine speaking of how it can save the world, rid it of the filth. Childrens’ voices, “Look Daddy, we’ve found an egg! It’s a giant stone egg!” The Machine: “Restore me Mandus, start the engine and burn away the filth, we will achieve our purity, I will deliver you the children”.
We wake up in Rod Control and throw the switch, Unholy hum, screen starts shaking. Climb the now falling apart staircase as the boiler throbs and heaves, reach Ignition Control and throw the switch. “I live! I live again!” bellows The Machine in best traditions of Victorian evil villains, and if we now explore, we will find the grille has opened and we can exit this space.

**EVENTS:**

VO’s on pipe crawl approach.

In storeroom – door locked. Children laughing on other side. Crash of furniture, then it opens.

Control board in top room – currently red lights on two of the grid (others are green) – map of the rod room below.

If the player goes left, ladder is currently up. Shudders.

Looking down from top run, see a manpig scuttle along the bottom.

Rod 1 in first storeroom we come too.

Visual distortions in fuel control – sounds of fires and screaming.

Fuel cannot be routed – but there’s a second rod in the cupboard

**INFECTION EVENT:** whilst carrying the rods, the player is always at Level One infection. With two, it’s level TWO.
At foot of the stairs, door to storeroom slams shut. As player approaches it, it bangs.

Manpig in the storeroom, player must lure it out

Ignition control airlock – window in wall, manpig chases in, but only one door can be open at once, and the player can shut it in, then use the wheel at the side to flush the airlock with toxins.

Ignition control – cannot be started.

Spawn point – Ignition control, and under the final stairs.

Highly toxic rod control – player must put the two missing rods in the correct places, then they glow. When second is places, screen starts shaking – INFECTION EVENT - level three! Player can stumble back to ignition control, but once there, or if they ignore it, on the stairs AUTOMATIC blackout.

VO in darkness.

Wake up in ignition control – lights are green, screen shake down a bit. Fire boiler and screen shakes are big, noisy – lights change throughout level. Ladder at top drops, all control panels now on in other rooms.

6. Engines

There’s a lot of noise here – grinding and pumping. We follow the access corridor, and find ourselves in a chimney-like brick structure, shifting hue from the green of the sewer back towards the reddish-brown of the Factory. We are in the engine, directly under the piston room. Below us, three huge cogwheels turn, we need to fall in a gap in the cogs, or the centre of the wheel to the next, and so on, until we reach the bottom. Between the middle and top wheel however, there is a broken vent that leads to a storeroom, where we can grab a box (or crowbar probably) - we’ll need this later on. At the far side of this is an access ladder, which is bolted from underneath, so we can only get into the room from the bottom (if we didn’t get the crowbar, we’ll need to do that later on).
At the bottom of the wheels, we can see they are attached to a set of vertical wheels in a long corridor, but the first is set slightly away and they are not turning so we can’t get through to the left. However, we can take a corridor to the right and follow this around to a control room. The architecture here is dark, greys and blacks, oil and blood around. Cramped and dank, very noisy when we are near moving wheels. In the control room (as before, all the control rooms have that Jules Verne mix of plush Victoriana and bathysphere-chic), there is a gearbox missing the gearstick. Another door leads us to a space under the pistons. Strange boxy engines and machines are dotted about here, creating a simple maze-like structure – it’s very dark, but we might want to not put the lantern on, as there is a beast in here with us. There are two exits to this space. One leads to a storeroom where we can find the gearstick, the other is a ladder that takes us back to the (operational) knife-section of the pigline above – currently grilled off (we go here later). Grabbing the gearstick, we can activate the gearbox in the control room and the vertical wheels in the passage will now start turning. There is a potential here for a puzzle where we have a number of gear controls for each wheel and they spin at different speeds, so you have to set the right combination, but that’s optional. Either way, at the end of the cog corridor is a ladder up. As we approach, the grille slides back and we can go up to Piston Control.

This control room looks out onto the Piston Room. We can activate each piston in turn. 2 & 3 do nothing, but 1 can be used to crush the beast in the piston cage. Once we’ve dispatched the beast, we can head back along the cog corridor, only they have sped up to varying degrees and one is moving too fast to get through. We need to use the crowbar from earlier- if we don’t have it, there is a second route, through a vent shaft, back to the chimney store room via the access ladder, so we can fetch it this way. We make out way back via engine control, and the engine room up the ladder to the pigline. During this, The Machine is crowing about being free and
realising its destiny and saying things like “More pig! More pig!” in an upsetting and frankly pretty fucking disturbing way.

The pigline is now operational and the conveyor is moving. This also means the knife arms are slicing away so we can’t go back. We can ride the line out into the piston room again, only the grille at the far end hasn’t opened as we thought, so the only way to go is back down onto the piston room floor. We can’t get past the piston yet, we have to engage the emergency brake in the far corner, before we can exit.

**EVENTS:**

Drop into slowly rotating wheels.

Wheel in corridor stopped, no access.

Flickering strobe lights in main engine room – manpig in there (trap in storeroom).

Use gearstick in gearbox – wheels now spinning in corridor.

Get to control room – overload circuits (tile puzzle?) – shakes on pig corridor, noise and smoke?

Thundering engine now, ground shaking and low level noise and pistons smashing.

On return to engine room lift- as hit elevator, crashing noise and manpig bursts out of storeroom.

In lift, lift stops, creaks, lights off, down and up again (engine is failing)

On pig line, knives and sawblades are now working fine – avoid them obviously.

No INFECTION events apart from the manpig

7. **Fattening**

We climb a ladder and find ourselves at the back of a broom cupboard, exiting this we are once again in a hallway in a house: paintings, carpets, all normality. There are four doors, two are locked. Both open doors lead to studies. In one there are sketches of the machine – Mandus’ office, clearly – and an elevator in the corner that has a hole for a round key-like object. The other study has obsessive collections of items about Mayan and Aztec gods, and drawings of the Orb feature in many of these. There is a library key in the desk drawer. We travel through the library and find ourselves in another hallway, which leads onto the Banqueting Hall. This is a two level room, with the banqueting table below and landings with many doors above, accessed by a staircase near a set of locked double doors at the far end. Although the décor here is lush, everything is caked in gore and grime. The table is a scene of massacre, filth-strewn and with rotting food everywhere, chairs knocked down, blood sprayed across the walls. There is a carcass at the centre of the table, it looks suspiciously human. The Machine says “It is one of those great
ironies that in order to rid oneself of filth, one must occasionally pander to its needs. But I will be rid of them all soon enough”. Mandus realises this is inhuman, and confronts the Machine, who laughs and says it is only doing what Mandus instructed it to do, that the answers lies buried beneath a temple in Mexico. Exploring the rooms above, several are open – these are the Fattening Chambers. We’ve seen one before, caged four-posted beds swimming in putrid flesh and shit, feeding hoses dangling from the ceiling, a headless body spewing lightning back into the walls. Argument between Mandus and The Machine reaches a head in one room, triggered by being on the bed – an alcove behind a portrait is the bait – The Machine sneers that Mandus is a pig like all the others and will be dealt with the same way – and the bed drops away like a trap door, landing Mandus on the operational pigline.

This section contains the big plot hinge, as Mandus realises The Machine is the enemy and must be destroyed. We travel along moving conveyors – you can’t move against the flow. There is potential for some extensions here as the mapping is pretty simple and repeatable, so if we need to buy time for story, it’s a straightforward matter. We are moving through stone walls, metal conveyors, swinging lights now burning, meat hooks moving on their chains. We pass other conveyors, on one we see a beast skewered on a hook squealing off into the darkness. The Machine rants on “Whores, beggars, orphans, filthy degenerates. Pigs all. But we had our plan, Mandus, you and I, we would purify the streets, cleanse this city, set the great industry free. We would have cleaned the world, made it pure” and later “You have already seen the answer. The Maya believed by shedding blood they would avert the apocalypse. Their tragedy was nothing more than a lack of industrialisation. They simply could not shed enough blood. But you and I,
Mandus, we understand and we have built a great machine capable of spilling more blood than the Mayan’s could ever have dreamed of!”

Meanwhile, the gameplay set-piece requires the player to shift from conveyor to conveyor on the pigline, avoiding being sent into spinning blades (if this happens, the machine jams, the player blacks out and is dragged to the central area by A by a beast – we don’t want a ton of reloads going on). They need to set a series of switches, ride the belt out to C to power up B, then ride it to B, set that, ride back towards C and jump to the lower belt on the way. They ride this over a huge room, passing away into the darkness. “I was born here, Mandus, under these streets, a thousand miles from my jungle temples. When I have amassed enough power, when there has been enough sacrifice, I will split the atom, my egg, and ascend with the ashes to become a god” (remember the nuclear fuel rods – uh oh…)

We climb upwards on the belt – riding one, we find a belt-control room, and in it, the Olmec head key to power the elevator in the study. Riding further upwards and suddenly into a dark area, where we are dumped off the belt into a blue-lit, tiled room. It’s a freezer, the belt has dropped us onto a pile carcasses, part pig, part human. More hang on chains around us. There are three interconnected freezer chambers, the corpses on hooks create swinging walls. A cry tells us we are not alone here. It’s really infectious, we need to take care. We can maybe extinguish some of the gaslights on the walls to help us, maybe use some meat or offal from the floor to throw and distract the monster in here with us. If we make it through, we climb up out of a chute into a kitchen, like the banqueting hall it is a scene of carnage. A head in a pan on the stove, arms and
legs hang like sausages, someone has been boiling what we hope is a doll. “This is all you are. I have seen what you boil down to, and it is nothing more than grease to ease the machine” we are told by The Machine. “You are all the same. The man eats the pig and the pig is eaten by the machine and the man is eaten by the machine. This is as it ever will be”. It tells us that Mandus killed his own children on the steps of a Mayan temple, believing he could avert the coming apocalypse if he shed enough blood. “You built me Mandus. And I, in turn, have built you.”

Out of the kitchen, via the locked doors into the banqueting room, through the library to the elevator in the study. Mandus determined to stop the machine. We ride it down and down, into the heart of the monster.

EVENTS:

(see manpig patrol route)

Loud crash and lights flicker overhead on and off (weird electrical lights here)

Scratching noise from inside the cupboard. When it’s opened, headless corpse falls out (INFECTION event)

FX

Locked, requires key from 6.

Locked – only opens from banqueting side.

Only accessible from 3 – both doors are barricaded, but these can be cleared. Contains library key, lots of blood and gore strewn around it. Head in a bucket – player hears whispering on entrance (Dr Olafson’s!). Bucket is rattling from a distance?

Locked from corridor side, can only access via banqueting.

Lights in library are strobing. When player is inside properly, children’s voices “Sssshhh!” and all the lights go off.

Bookshelf topples over behind the player.

Visual distortions, INFECTION area near table.

In one room, painting behind bed askew to show partly opened safe. When player on bed, cage bars descend from ceiling to trap them, then bed opens up / falls away to dump them on conveyor.

Room is open but door to elevator is locked (Olmec head key must be obvious)

Conveyor belts (needs work in 3D) – switches to change their direction, to allow player to progress.
Get Olmec head key – vision of Aztec temple

Freezer – INFECTION event – manpig in the freezer. Use offal to distract it? Build this into the AI.

8. Tesla

We pass through the sewers, it’s important that the player understands we are beneath them, and into small series of corridors- potentially surgical labs and cells, but this is disposable. It might be better to put the player back through some sewers and then underneath them, with a Mandus voice-over saying something about all the shit flowing above the machine, it’s fear of the filth but it’s need for it, to power it’s plans to slaughter the population of London and become a God.

EVENTS

All the lights go off, then on, then splutter, then go on again. But strobe occasionally throughout the level.

Door slams behind the player, Machine VO about stopping Mandus. Lights go off, children whisper “This was papa” and door opens in darkness. Light comes back on again.

Lever by the side of the door opens this. It also opens the door at the far end of 4, but closes the door leading to 8+9

Manpig is here (INFECTION event) – cannot get at the player though.

Lever here opens door 4, releasing the manpig. Lever on the far side shuts this door and door 4 – so player has to bring manpig out, then trap it in.
Lever here opens 6 and 5 – lever on far side closes, draw the manpig further, then trap between 5 and 6.

Lever here finally opens the doorway to get to 8+9

Manpig body hooked up to machine. Begins to thrash as we go into the room.

Lights go off again, sound of huge engine below us somewhere. Things crash, light splutters, then blue cables along the wall begin to glow, leading us up the stairs and towards the maze.

We end up in the CPU maze. The environment has started to become strange, we now have pylons and cables running around, interwoven with the stonework walls and ceilings of the sewers and embedded factory. We look down on a weird cage of electricity – the CPU. Horrible grinding noises, strobing blue lights, dust falls from the ceiling. This is the central route, ideally, it’s all in a giant 3d cage of metal that is really disorientating from the outside. Bold blue static. Lightning caged. Manpigs squealing and burning within.

Trip switch to fuse the machine. Both trips need to be thrown to shortcircuit the CPU and open the gate to the tesla field (c.) – the wall is otherwise ‘live’ – doing big electrical damage to the player.

The second switch. First switch thrown results in massive screen shake, explosions, dust. Second (no matter which order) – machine screams in fury, explosion, all the lights go out (lantern only!) Fuel by the switch just in case.
Exit route to tesla field – crashing behind as whole CPU electrifies – heading for the player so they have to jump – if not, they are thrown backwards.

As player enters the room next to him (towards switch A), the door slams shut and the manpig is electrocuted.

Ignore the door, the manpig is trapped in here. INFECTION if close though.

Huge boom, lights go off. Engine re-fires, lights come on again, lots of dust and shit.

We exit over a huge construction, a grid of pillars with lightning arching around between them. Suspended in a vast space, seems to go on forever above, around us (lightning constricts our actual exploration). We descend several layers. Again, this sequence buys us time to deliver the build to the climax in terms of the narrative, as The Machine give us it’s mad rants about being a God, buried in the jungle until Mandus found it and hatched it out, the truth about him building the complex and machine, the blood on his hands, and through it all, The Machine’s condemnation: “This world is a machine! A machine fit only for pigs! For the slaughtering of pigs!”. At a bottom level, we find an elevator. Time for the final descent.

8.5 Elevator journey
Potential addition – fade in and out of shots of environments (from before) – huge structures – sense of time passing, of depth - only if there’s time to implement it.

9. Temple

We fall for a long time. Mandus’ ‘final’ voice over, his coming to terms with killing his children, building the machine, his madness, his finding of the Orb. The need for the machine to be stopped, that he doesn’t yet understand what is meant by its plan to bathe the world in cleansing light.

He lands in another rocky cavern, but the ceiling is interwoven with the underneath of the sewer tunnels. Cables run. The cavern opens up into a much larger one. At the centre of this is a Mayan pyramid, with cables running up the stairs haphazardly. There are four stone pillars running up to the roof, clearly the same architecture as the sewers. The one nearest the player is corrupted, split, large cracks run from it to the pyramid, up it, there is rubble at its base. Around this, a body of a beast and a shattered fuel rod, like the ones in the boiler. If the player explores, another smaller cavern off the main one contains a pile of fuel rods. There are also many more piled around the base of the pyramid and arranged on the steps. Mandus realises “If one of these cracked rods was enough to shatter stone, if these are the fuel powering this entire machine, then what hell would be unleashed if all of them were to be channelled into a single mighty explosion. It would be like the very sun on earth”. The lightning from the Tesla field runs vertically downwards to the top of the pyramid, at it’s summit is an altar and the orb, suspended in a beam of lightning.

The way to complete the game is simple, the player must smash rods against the remaining three pillars. Each time, cracks appear, the world shakes, rubble falls, the player becomes increasingly infected. “I will bring the sky down, crack the supports and collapse the sewer systems down onto this infernal engine. I will drown it in the very excrement it fears so much.” When the player cracks the last rod against the last pillar, the game ends: the world shakes violently, rocks and rubble falls, stuff strikes Mandus and he falls, the screen going black for the final voice over as we hear the Machine screaming, water rushing in, the chamber collapsing.

There is the potential for another ending, which is really easy to implement and might take advantage of the fact that some player somewhere is going to attempt to take a rod to the lightning. In this case, instead of going to black, we go to white – Mandus has set off the nuclear explosion and we have The Machine delivering the final voice over, not Mandus. It’s very easy to add this in as an alternative as it doesn’t require any animations...

The following section contains the full post-mortem report (Howell, 2014) written and published on Gamasutra following the AAMFP’s commercial release.

Amnesia: A Machine for Pigs was born out of an ethos shared by Frictional Games (FG) and The Chinese Room (TCR) towards game design and what games as a medium are capable of doing. This focuses on the creation of games that prioritise immersion, emotion and the affective experience of the player, combined with a powerful, thought provoking and well told story.

Figure E.1: Concept artwork for the cut Boiler level.

The game was originally pitched as a short two-to-three hour experience entitled We are the Pig, although this initial design went through numerous changes and expansions during its development. This led to the development time extending from the initially intended 12 months up to 24 months. While the game length increased beyond the original design, some sections still unfortunately did not make their way into the final release such as the Boiler (Figure E.1) and Fattening levels.

TCR came to this development with prior experience developing games such as Korsakovia and Dear Esther that provided a number of lessons to inform the development of Pigs. FG of course had experience from their previous successful horror titles. This project was however the first time that either company had collaborated with another company on a project and this would have very noticeable consequences throughout the development process.

Pigs has released to predominantly strong critical praise, but has clearly proven to be a divisive title amongst players. The game has demonstrated success in a number of areas, however the game and the development process itself were not without their problems. In discussing both the game’s successes along with these problems, this post-mortem aims to provide a comprehensive overview of what was an enjoyable if sometimes arduous development.
F.1: What Went Right

F.1.1: Creative Freedom

FG took a very hands-off approach during early development, allowing a high degree of creative freedom amongst the TCR team. FG must be commended for their willingness and openness towards experimentation with an established intellectual property such as Amnesia. Such experimentation with established formulas is something that arguably should be encouraged more across the industry in order to keep pushing at the boundaries of the established design spaces that we currently operate within.

This invitation to experiment meant that in terms of both gameplay and story, time was spent during early development throwing around a lot of different ideas for plot, puzzle scenarios, game mechanisms, enemy designs (Figure E.2) and enemy encounter scenarios. These ideas ranged from small adjustments to the established Amnesia formula through to more radical and complete departures from it. Some of the ideas that came out of this process had potential for very interesting gameplay, such as enemies that would only appear clearly in the peripheral vision of the player, and a procedurally generated three-dimensional electrified maze, similar to Vincenzo Natali's Cube (Natali, 1998).

![Early concept artwork attempting to define the appearance of different pig creatures.](image-url)
While technical limitations of the HPL2 engine prevented the inclusion of the more complex ideas, a number of less technically demanding ones, encouraged by this creative freedom, did make their way into the final game. The intention from the outset was to develop a game that players could not play simply by relying on their experience with *Amnesia: The Dark Descent*. This was critical in giving players a new experience, and the freedom to experiment provided a catalyst for creating such an experience. The last thing TCR wanted to do was give players "just more of the same" gameplay established in FG's previous titles. While certainly some players would have been happy with this, it was not what TCR wanted to make, and furthermore was not what FG wanted TCR to make. The aim was to bring a fresh approach to the established *Amnesia* gameplay.

Of course in following a critical success such as *The Dark Descent*, it would be both naive and a severe disservice to the established fan base to not consider how a sequel may draw on the most successful elements of the original. Ripping out the heart of what makes *Amnesia*, *Amnesia*, was not the aim; it was more about structuring a new, but horrific body around an established skeleton. This also applies to linking the game to the overarching franchise mythos. Again, FG did not require there to be any notable link between the games in terms of narrative, but TCR felt that a complete departure from the mythos would risk alienating many of the established *Amnesia* fans. Indeed, early media releases that prompted a deluge of theories around the game's plot and characters demonstrated the level of investment the fan base had in the established lore, with forum threads spanning over 500 separate pages analysing the minutiae of the released material on the game.

The response of the community to these early media releases was instrumental in fuelling further creativity during the game's development, as a number of forum posts discussed ideas or different plot interpretations that had not been considered. While of course not all of those posts formed the basis of something that made it into the final released game, a small selection were worked into the fabric of the established game and plot where appropriate. Hopefully, some players may recognise small features within the game that they had previously postulated about!

As for the game's mechanical core, the removal of the sanity meter was a primary aim from the outset. TCR recognised the likely controversy of this, but felt that the system was fundamentally flawed as a means of telling the player they should now be scared, and approximately 'how much' they should be scared. The aim was to create a game that was inherently horrifying, and thus did not require a meter or gauge to tell the player to be scared. However, throughout a long period of the game's development (approximately the first 6-7 months), and as per the game's original design document, the sanity system was to be replaced by an 'Infection' system (Figure E.3).
The infection system would serve some of the same purposes of the sanity system, providing visual and auditory distortions and hallucinations as the infection level increased. This would escalate Mandus having to stop, wretch and/or vomit as the player moved through the game environments. Enemies attacked the player through infectious damage rather than the physical damage that is present in the final game, and players were able to heal themselves through the use of the decontamination chambers in the game. However, this system would be more abstracted than the sanity system, removing the requirement for a gauge-based representation to track 'infection level'. Instead the player would be able to approximate the level of infection through the visual and auditory cues provided during gameplay. The linking of this mechanic to the attacks of the enemy agents and to the lack of cleanliness in the Victorian London setting was intended to blend the mechanical workings of the game to the setting and plot in a more integrated way.

As development continued however it was clear that the system simply was not integrating well into the rest of the game, and felt too much like a 'mechanic' for the sake of being a 'mechanic'. The infection-based attacks from enemies for example, felt weak and unthreatening at best and downright confusing at worst. Similarly, environmental infection events, however they were framed, could not shake the feeling of players walking through luminous green toxic waste in any number of classic shooters. After many attempts at integrating the system more convincingly into the game, the decision was taken to remove it. This removal, while certainly not trivial, allowed much more focus to be paid to the core essence of the game - the story, and the environments through which it is told.

The removal of the sanity system was an important aspect of producing the game that TCR wanted to produce, and has been predominantly successful in doing so, as reflected in the writing of a number of critics that have praised its absence. While it may have been possible to continue with the infection system and build it into the game in a manner that felt more integrated, the
decision to remove it allowed more attention to be given to other aspects of the game. The result is a game that is almost certainly more cohesive across its story, world and gameplay and thus is stronger as a whole. Without the level of trust and freedom provided by FG a change as critical as removing one of the core systems would have likely caused more serious concerns and delays in the development, especially as the system was such a key component of the originally pitched game.

F.1.2: Development Tools

The HPL2 engine is certainly not without its quirks, and a long time was spent during early development working out appropriate asset pipelines and the most efficient ways of implementing different features. However, despite this, once the tools were fully understood by the team, it was clear that their combination was capable of creating excellent products.

For a small development team particularly, it was important that everyone should be able to make use of the most critical tools, such as the level editor and be able to understand and make changes where necessary within the level scripts. The level editor provided this functionality and, with some bespoke adjustments made by TCR’s programmer, it was possible to customise the editor to TCR’s preferred methods of working. The accessibility of the source code for the toolset meant such changes could be completed with little or no assistance from FG, which meant minimal delays during development for tool-based problems.

Similarly, the accessibility of the AngelScript-based API used for writing the game’s level scripts meant that basic events could be implemented or adjusted by multiple team members rather than relying solely on the scripter. Early in development, a number of portable sections of script were produced, fully commented, that could be dropped in to any level and then adjusted as necessary by relevant team members. This was particularly useful as a means of enabling the sound team to work independently with portable scripts for systems such as randomised environmental sounds, ambient soundtrack switching and music integration.

Figure E.4: Viewing G-Buffer contents in-game (Diffuse Colour, Z-Buffer and Surface Normal).
HPL2 also already contained a number of useful debugging tools. While not being comparable to more established game engines for obvious reasons, the included features were targeted specifically at the rapid creation and testing of Amnesia style gameplay and worked very well. Once more TCR were able to implement their own additional debugging features, such as in-game dynamic prop placement to speed up the process of placing small objects accurately in the environment by doing so during run-time, as well as the ability to view the separated contents of the G-Buffer used during the process of colour grading the game (Figure E.4). The flexibility of the tools and debugging options provided a solid foundation upon which to develop the game.

F.1.3: Development Team and Communication

The core development team’s ability to work well together with minimal man-management required was one of the most obvious successes of the development, and without doubt influenced the quality and cohesiveness of the final product. Multiple members of the team had previously worked together on past projects such as Nuclear Dawn (Interwave Studios, 2011) and Off Limits (Off Limits, 2012), which meant there was an established rapport between them. It was then very simple for the more junior staff to find their own suitable fit within the team.

The combination of experienced staff and fresh graduate staff provided a catalyst to a number of useful, if sometimes rather heated, debates about the best way of implementing a particular feature, of lighting a particular area or of scripting a particular enemy encounter. Those with more industry experience of course had more existing knowledge that they could draw upon, but the lesser experienced of the team were often able to bring different ideas to the table as well. Ultimately the combination proved highly productive, even if in some circumstances these discussions took a little longer than needed.

These discussions could not have been achieved however without ensuring communication between the team was simple, reliable and fast. Throughout development, the entire team was working remotely with staff working from the UK in Portsmouth, Brighton, London and Winchester, along with others in Belgium and FG based in Sweden. Communication was carried out primarily through Skype (Skype Technologies, 2003), with email used when paper trails were necessary. The majority of communication occurred in this Skype ‘virtual office’ however and it proved successful. Such a setup has evident drawbacks over a normal co-located office setup, such as not being able to simply turn to a colleague’s screen and explain something, instead having to draw elaborate diagrams (often in Paint) to try and explain the functionality of a particular puzzle or the sequence of events in an area. However, given the limitations the team had to work with, the communication methods were fit for purpose and served the development of the game well - as well as generating a number of excellent pieces of programmer-art and stick-man diagrams to boot.

Lastly in this section it is necessary to mention the team’s passion, and the belief in the story TCR were trying to tell. Such a statement may seem somewhat cliché, but that passion is a critical
component in crafting a game that feels ‘complete’. This passion especially applies to the political element of the story, which was a point of particular interest for the team. Having a singular vision for a game is important regardless of what game is being discussed, but for a game that rests on its ability to strike emotional as well as horrific chords with players, it is even more critical. That singular vision needs to include not only what must be happening on screen or being heard by the player at any given point, but also the emotional state the game is trying to instil in the player at any given point as well. Even with multiple staff changes during development on the art team and the sound team, that singular vision remained consistent. This again is testament to the excellent team rapport and communication, as new staff joining mid-way through development were soon briefed and integrated into the team.

F.1.4: Environmental Storytelling

TCR’s previous games make use of voiced narration and internal monologue as means of telling the player a story. Environment design has been important in these past titles to ensure the story is placed in rich, believable and cohesive worlds that provide context and a sense of place for the player. Story telling through the environment itself can be seen in *Dear Esther*, and in *Pigs* the aim was to bring such environmental storytelling to the fore to a greater extent.

In a horror context, telling a story through the game world itself provides the potential for both greater poignancy as well as greater ambiguity. The poignancy is critical in creating an affective, emotional experience for the player. The game environment has far greater potential for this than a spoken or written dialogue describing that environment. For example, a single picture of the shoe room at Auschwitz, or of the lone protester facing up to a tank at Tiananmen, are far more arresting, far more powerful images than anything that could be described through words alone.

![Figure E.5: Sequences such as the Pig Nest in the Sewer level allow storytelling through the environment and characters alone, without revealing lots of explicit detail.](image-url)
The poignancy of such environmental storytelling is important. However ambiguity is also a key part of this type of storytelling. Even the above examples offer levels of ambiguity despite seemingly portraying quite obvious events. Each pair of shoes at Auschwitz has an implied life story attached to them for example. The picture of the Tiananmen protestors poses questions regarding his thoughts and emotions at the time the picture was taken. It is these associated implications that assist in the creation of the emotionally affective aspects of a story. The image, or environment, is simply a cue to thought and to consideration, rather than an explicit, closed story. Makoto Shibata, director of the Project Zero (Fatal Frame) series discussed such an approach in an interview with The Guardian (Stuart, 2006) stating that it is not about simply showing scary things, but providing players with fragmented information through a variety of means that forces them to consider for themselves the horrific events of the game. Ultimately, that which occurs in the player's head will almost always form itself into something more disturbing and more horrific than anything the game could explicitly portray.

The script for Pigs was nevertheless initially very long and included a substantial amount of voiced internal monologue. It quickly became apparent that the amount of voiced storytelling was going to have to be reduced to prevent players listening to voiceovers for long periods. TCR had also initially underestimated the size of environments that would be necessary to allow such an amount of voiced storytelling to comfortably fit while still allowing the player to move around freely. While small sections of the voiced script were cut, much of it ended up being moved across to written notes that are found throughout the game, thus the overall script itself is largely unchanged from the original version in terms of its content.

These written notes however are intended to support the story suggested by the game world through the design and contents of the different environments that the player travels through. For example, the hidden corridors and rooms in the game's early levels are not explicitly explained by the game, although the script may allude to them at points. The player is left to determine their own interpretation of what they were used for. This same approach applies throughout the game, and it is rewarding to see many different interpretations of the game's overarching narrative, as well as specific objects or characters, appearing across discussion boards online. This individual interpretation was also the reasoning behind the removal of voice acting for the written notes. With a voice actor, it is much harder to leave ambiguity intact as emphasis and intonation will always suggest a 'correct' interpretation. Suggesting such definitive interpretations of plot elements would have unravelled the image that the game builds up of Mandus' questionable mental state.

The ambiguity of the storytelling itself has achieved its aim of encouraging thought and extended discussion amongst players. In many cases TCR have achieved some powerful and emotional responses as well. The game's use of music and sound was pivotal in creating these emotional responses and has been cited in a number of critical and player reviews as one of the game's
strongest components. Its quality is testament to the tireless work and attention to detail of the small audio team working on the game.

**F.1.5: Streamlining Gameplay Experience**

As previously mentioned, *Pigs* has clearly proven to be a divisive game amongst players. For many, the streamlining of the gameplay itself may well be more suited to the ‘What Went Wrong’ section. However, for delivering the type of experience that TCR were aiming to deliver, streamlining the gameplay itself was important, and from this perspective the decisions made to achieve such streamlined gameplay were predominantly successful. The removal of the sanity system, and later the infection system, has already been discussed, as has the shift of voiced narrative to text-based narrative. However, further mechanical alterations contributed to the streamlining of *Pigs’* gameplay.

A sprint limiter was implemented for a while during development, intended to emphasise Mandus’ age and level of fitness as well as making enemy encounters more dangerous by reducing the player’s ability to run long distances if spotted. This was removed later on to provide a more consistent experience for players, as the limiter was more of an unnecessary annoyance than it was an interesting tactical addition in the parts of the game that did not contain enemies. This was especially noticeable in the game’s larger environments where artificially limiting the player’s movement speed provided no experiential benefit.

The most obvious streamlining decision, and again a decision that TCR recognised would likely be controversial was the removal of the inventory. A number of possibilities were considered regarding the inventory with the overarching aim being to always keep the player within active gameplay. Breaking out of active gameplay into a separate, static inventory screen not only breaks game flow, it more critically damages the building of tension, suspense and anxiety that is so vital in horror.

Games such as *Dead Space* and *System Shock 2* successfully integrate inventory systems in ways that keep players within active gameplay as much as possible. *Dead Space* makes the inventory screen part of the diegetic game environment while *System Shock 2* allows the player to decide how much of the inventory screen to view, while keeping the game running in the background and leaving the player open to attack. For *Pigs* TCR discussed inventory solutions such as a quick access item bar on the bottom of the screen (similar to *Neverwinter Nights’* Quick Bar) that would appear on a key press, and a minimalist inventory that would only be available to carry and use a small selection of important items, such as keys. However, in line with the aim to keep players engaged in the game world as much as possible during gameplay, there were only a small number of scenarios in which these inventory solutions would really be necessary. Most of the planned scenarios were designed to allow the physical movement of objects through picking up and carrying items around the game environments.
With this in mind, the complete removal of the inventory was decided to be the best solution, as the inclusion of one that would then only be used in a handful of situations in the game seemed unnecessary. This streamlined the gameplay during puzzle scenarios, removing the need to move between screens, and making it possible to implement a range of different, in-game scenarios for players to interact with and solve. The streamlining of gameplay in this way achieved its purpose – however, these in-game puzzle scenarios failed to reach the potential that the game’s industrial, mechanical setting provided.

F.2: What Went Wrong

F.2.1: Rapid Prototyping and Project Scheduling
The game’s puzzle scenario design and indeed all of the main issues that are discussed in this part of the post-mortem stem from poor project scheduling and a mishandled prototyping phase of the project.

While FG provided TCR with a high degree of creative freedom in terms of the game’s content and design direction, the deliverable milestones requested were much less flexible. The first deliverable FG requested was a near-complete version of the game’s Cellar level (the third level in the final game), which would demonstrate all of the major components of the game, such as puzzle design, enemy encounters, art direction, the infection system and narrative delivery.

Meeting this first milestone prevented TCR from grey-boxing the full game at this critical early stage. Moreover, because the entire game had not been grey-boxed and considered as one complete entity, the version of Cellar that was created for this first deliverable was eventually changed into an almost entirely different level, thus rendering much of the time and effort spent on the first version wasted.

Whether the decision to request a completed level rather than a complete grey-boxed version of the game as the first milestone stems from FG’s lack of prior experience acting in a production role is difficult to say. However, it is likely that having a full grey-boxed version of the game early on in development would have made later decisions much simpler and quicker to make, as well as giving TCR a much better idea of the pacing of the game from start to finish. Critically, this complete version of the full game would have allowed much easier mapping of key events, such as enemy encounters, rather than attempting to implement such events while levels were in the process of being fully built, or worse, after they had been built.

The lack of a full grey-boxing process, and thus no rapid prototyping of the game’s core systems meant that even 6 to 7 months into development, the mechanics of the game (such as the infection system) had not been signed off, nor were they in a state to be signed off. While the game’s narrative and environments were predominantly confirmed at this stage, the mechanical core of the game suffered severely from mistakes in the early stages of development. This noticeably followed through to the final released game which is much weaker mechanically than it
is in terms of aesthetics and narrative. Lastly, a full grey-box of the entire game would have highlighted very early on the issues that were encountered later regarding frame rate drops linked to the number of physics objects being used in a level. This would have allowed design work to be carried out with this limitation in mind from the start, avoiding the need to implement the workaround of making a high number of previously interactive objects static - something that been criticised in a high number of critical and player reviews.

While FG’s milestone requests were detrimental, the development also suffered from TCR’s lack of detailed project schedule. While the team were fully aware of the major tasks that needed completing at any given time, and were adept at prioritising appropriately, there was no central documentation that kept track of tasks, internal milestones or game bugs. This led to a lot of additional communication being required in order to discuss tasks that needed completing, as well as some less critical tasks consistently being overlooked. The lack of a centralised bug database meant that the final testing phase before release needed more time as there was no record of outstanding bugs or of those previously identified and fixed.

This combination of no real prototyping phase of development and lack of detailed schedule and documentation can be pointed to as one of the root causes of many of the other smaller problems throughout development.

F.2.2: Collaborative Working and Compromising the Creative Vision

There was a level of naivety on both sides of the collaboration between TCR and FG, as neither company had worked in such a collaborative scenario previously. While the creative freedom provided was welcomed, the hands-off approach taken by FG may also have caused some of the apparent misunderstandings and contradictory feedback that TCR received on deliverables. Many areas of the game were reworked two, three of even more times based on different and often conflicting feedback on deliverables. Some of these reworked areas are much stronger because of this back and forth between TCR and FG, however in some situations it is evident that TCR should have been firmer when defending their initial design decisions.

One of the notable decisions impacted was removing the game’s equivalent of The Dark Descent’s mementos – hints located in the journal that assist players in completing tasks. The initial design of Pigs removed these entirely, requiring players to rely on the game environments and diegetic information within them to solve the game’s various challenges. TCR felt this made the game more cognitively challenging (something that had been raised in multiple threads on The Dark Descent’s discussion forum as a desired consideration in future games), and more rewarding for players. FG were not responsive to this however and thus the final release of Pigs includes frequent additions to the journal of these hints. The differing opinions of TCR and FG on the ability of the player were clearly apparent in this instance, and TCR should in retrospect have been stronger in defending the initial decision to remove the hints.
Figure E.6: The early Mansion levels are most noticeably affected by the altered colour grading system.

The lack of consistent communication between TCR and FG also meant that when TCR handed over the final build to FG for the final phase of testing, optimisation and polish before the game's release, some key features implemented by TCR were altered without the impact of such alterations being realised. For example, the colour grading process carried out by the TCR art team had drastically changed the look and feel of the game's lighting. This was carried out independently of FG and was contained within the final build sent to them. However, because these changes were not communicated between the two companies, changes made by FG during the final stages of development resulted in a release version that was graphically compromised - the blue 'fog' that has been noted by a number of players. This is a result of the colour grading system being altered, and the results compared to an earlier build (March 2013) are readily apparent when placed side by side, most notably in the game's earlier levels (Figure E.6). While this has been fixed in a later patch, the game should not have been released with the colour grading system in this condition.
F.2.3: Game Difficulty

The disagreements between TCR and FG regarding how challenging the game should be, both in terms of enemy encounters and in terms of puzzle-solving resulted in the difficulty of the final game being much too easy. Once again, TCR should have been much firmer in defending the game that they wanted to develop, but they conceded too many alterations that resulted in much of the game’s difficulty being suppressed. This impacted a number of scenarios in the game.

The Tunnels level was initially approximately four times the size of the version in the final game, consisting of more labyrinthine networks of corridors and claustrophobic rooms. Players had to retrieve chemical containers and use the vacuum tubes (present in the final game) to send them around the level and eventually back to the centrifuge. However, unlike in the final game, players were consistently hunted in this area by enemies, combining enemy threats with cognitive puzzle solving. The size and complexity of this area were eventually reduced drastically as the initial version did not meet FG’s approval. While the intention of the original level was to emulate feelings of confusion, disorientation and of being lost without the frustration of actually being lost, FG felt that even these emulated feelings may result in player frustration.

The Bilge level also contained an additional area that combined puzzle solving and enemy threats. The cogs required to repair the bilge pump (easily found in the final release of the game) were located at the end of a large, partially flooded room. Players had to navigate this area, avoiding a number of aggressive failed experiments in the water, retrieve the cog pieces and return (now burdened with the extra weight) to the machine. This sequence again did not meet FG’s approval and was cut from the game.

Figure E.7: This waste disposal pit in the Bilge level is the only remaining respawn area left over from the initial death handling system.
Enemy encounters similarly were intended to be much less forgiving than they are in the final game. TCR implemented a system attached to a reworked version of the enemy AI. This system made use of a death counter already available in HPL2 to carry out different consequence sequences when players were ‘killed’ by enemies. Rather than being killed and respawned at an arbitrary spawn point, players would be captured and respawned in a waste disposal area, or a specimen storage cage or similar area, separate to the main level. Players would then have to escape the area by solving a puzzle in order to return to the main area of the level in question. One such area can still be found in the final version of the game, located near the end of the Bilge level (Figure E.7) in which players must throw debris at a ladder in order to knock it down and climb out of the waste disposal pit. Furthermore, enemies would not disappear from the level. Each time the player was captured, enemies would either become slightly easier to avoid, or (if players were captured a lot by the same enemy) eventually despawn.

This approach to enemy encounters was felt to provide a good balance of challenge and tension whilst not becoming overly frustrating for players unable to get past particular enemies. However, due to the lack of grey-boxing and of scheduling along with difficulties in making changes to the AI system itself, the issues encountered during the implementation of these encounters could not be satisfactorily fixed before the game was handed over to FG. The result was FG reverting enemy encounters back to a state resembling much closer the behaviour of *The Dark Descent* with enemies despawning if they successfully kill a player. This dramatically reduced both the difficulty of these encounters and the anxiety and fear that should have accompanied them. This is reflected in the critique of both critics and players and is an area of the game that is far from the originally intended design.

While disagreements between the two companies caused a number of the game’s difficulty issues, the designs of some of the game’s puzzles themselves were simply not complex enough from the outset. Once again, some of the game’s puzzle scenarios suffered from being simplified in order to meet FG’s requests. However, TCR similarly failed in some instances to make full use of the potential of the game’s setting. Much of the player’s interaction with the machine is reduced to button presses, lever pulls and valve turns. Partially, this was a limitation of the HPL2 engine in not being designed to support large, complex set-pieces; however this could have been worked around more effectively had time been allowed. The Tunnels scenario in which players move chemicals around using vacuum tubes requires some more thought on the part of the player, although even this was simplified from its original design and was not used in any other areas of the game. Ultimately, the tasks that players carry out in the game are not to the level that TCR aimed for, both in terms of challenge and novelty, and this has been one of the main areas of criticism from players and critics alike.

**F.2.4: Separating Gameplay Types**

The issues with the game’s difficulty were compounded by the game’s design, clearly telegraphing transitions between ‘enemy’ areas and ‘non-enemy’ areas. Analysing the game in light of the player
responses and discussion on the game’s forums, this is clearly one aspect of the game in which TCR failed. The encounter in the game’s fourth level, Alley, highlights this division between the two types of gameplay in the game, and sets up an expectation that players then carry forward into the rest of the game. In Alley, players are tasked with filling a fuel can with fuel in order to move a truck that is blocking their path. While filling the can, the player comes close to being attacked by an enemy Wretch, but the Wretch is unable to break through the nearby door to reach the player. This scenario has suggested that players will not be attacked whilst completing ‘puzzle’ objectives, and thus eliminates the tension from future puzzle scenarios.

The above scenario was originally not a problem, as later in the game in Tunnels, players would be constantly hunted by enemies whilst attempting to complete other objectives, as was the case in a number of other scenarios, such as the previously mentioned Bilge sequence. Thus, these early encounters served to set up a false expectation of safety while solving puzzles that would eventually be subverted. However, as these later scenarios were changed, simplified or removed entirely, this early encounter now serves to set up an expectation that is then never challenged or subverted throughout the rest of the game.

The capabilities of the HPL2 engine and the existing AI system also contributed to the telegraphing of different gameplay sections. Having enemies stalking the player through tight, claustrophobic corridors and rooms filled with various debris and other obstacles was impossible within the limitations of the AI system, meaning enemy encounters had to be moved out into larger rooms and wider corridors. This resulted in players being able to identify likely areas where enemies may be lurking simply based on the level architecture, and this again detracts from the game’s ability to instil fear and anxiety in the player.

Once again, early prototyping of all of the game’s systems in a range of grey-boxed areas would have highlighted these limitations, and the game design would have been able to incorporate those limitations from the outset rather than reacting to them later on.

F.2.5: Public Relations and Marketing

While this is in the ‘What Went Wrong’ section, early marketing by FG in the form of an Alternate Reality Game (ARG) (Figure E.8) served to generate a lot of interest both amongst the established Amnesia fan base as well as across a selection of other websites and discussion forums. However, from this early success later media releases were too few and too far apart given the extended development time and delayed release.
Figure E.8: The ARG, which started with the single blurred image above being released on a dedicated website served to generate a lot of interest in the game.

This was very much a problem of balance from TCR’s perspective. On the one hand, the creation of additional trailers or the release of other teaser media would have maintained interest and maintained the good will of patient fans. On the other hand, the creation of such media requires time, and with no dedicated marketing staff, this time would have been drawn from the core development team, thus causing further delays in the development and release of the actual game. Furthermore, releasing more media risked giving away too much of the game and thus spoiling the eventual enjoyment of it for players.

The initial trailer was purposely created for the sole purpose of being a trailer, and did not contain events intended to be present in the final game (although the area shown in the trailer exists, the events in the trailer to do not occur in the game). This was intended to get across the essence of the gameplay without spoiling an encounter in the final game. However this did not have the desired effect, with a number of players voicing their annoyance that this event was not in the final game. The second trailer similarly was intended to demonstrate the type of gameplay present in the game without using a specific scenario in the final game. It was felt that this trailer was representative of the encounters present in the game, however perhaps this was misjudged given the negative responses of a number of players.

The lack of dedicated marketing staff was certainly a problem for TCR, and it is likely that a more consistent drip-feed of media throughout development would have been a better strategy in keeping interest higher. However, the reality of a small development team is that the game itself must come first, and further delays on an already doubled development time were deemed not acceptable in return for outputting more media.
The early release of a review copy to Youtube personality ‘Pewdiepie’ was an unexpected decision by FG made without TCR’s knowledge, and demonstrates the difference in aims for the game between the two companies. Pigs was intended from the outset to provide a different approach to gameplay than that offered by The Dark Descent and this style was far less conducive to the ‘Let’s Play’ audience, requiring direct engagement with the game itself in order to get the most back from the experience.

It was expected that Pigs would likely appeal to a smaller number of players seeking a deeper level of narrative engagement, rather than the likely wider appeal of The Dark Descent’s more action-heavy gameplay. Thus, this marketing decision was a poor match for the game it was intended to market, as was demonstrated by the responses of the core Amnesia fan base, many of which were as confused by the decision as TCR were. Fortunately many players chose to avoid watching the early footage so as to avoid spoiling the game before they had bought it. However this marketing decision remains an odd decision by FG given the highly successful start to the marketing campaign through a more cryptic, cerebral channel in the form of the ARG.

F.3: Post-Mortem Conclusion

Following in the footsteps of a game as widely recognised and respected as The Dark Descent was never going to be an easy task, and whatever game TCR produced it was evident that they could not please everyone.

The game that they created successfully carries with it their passion for storytelling, for environment and for atmosphere and through this the game has proven to be successful and well received by a large number of critics and players. The mechanical core of the game unfortunately failed to resemble a well-oiled machine, in spite of the best efforts of the development team. While Pigs may be a more forgiving game than TCR had intended, it is entirely possible that a more punishing version would have been poorly received by the very players that have enjoyed the version that was eventually released. In streamlining many of the mechanical elements of the game, the narrative was able to shine all the more brightly and instigate lots of heated and thought-provoking discussion amongst the player community.

Would the game have been better off removed from the branding of Amnesia and the inherent gameplay expectations that come with that name? This is a question that has been raised in a number of articles following its release. There is no way of answering that question with any degree of certainty. Indeed, as has been discussed in an interview with TCR’s Dan Pinchbeck and Jessica Curry in The Guardian (Ellison, 2013), many reviews stated that the removal of the word ‘Amnesia’ from the title would have elevated the game’s score. The expectation attached to a word, or to a franchise, is so powerful as to detract from the enjoyment of an experience that does not ‘fit’ that franchise – regardless in some cases of the individual merits of a particular game. This is apparent even in cases, such as Amnesia, where the prior ‘franchise’ only consists of a single title.
This is very much conjecture of course, as there is no way of knowing how the game may have been received in its original format of *We Are The Pig*. The game that was released has proven divisive but predominantly successful, and has clearly been well received by a great many players. Moreover, the game has generated a great amount of lively and passionate discussion which demonstrates how invested players are able to get in the game. What more can a developer really ask for than that?
Appendix G: Ethical Approval Reference

This research has been reviewed by the University of Portsmouth Faculty of Creative and Cultural Industries Ethics Committee.

The research has been approved, with approval reference number: **FO:07/12-0063**
Appendix H: Development of Coding Templates

H.1: Developing the Initial Coding Template

The \textit{a priori} codes in the initial template were based on the key features of the disruptive game design philosophy and implementation developed in Chapters 1 to 6, along with the specific disruptive game features implemented in the final release of \textit{A:AMFP} (Chapter 7). Codes to represent possible reported influences of disruption on the player experience, both positive (e.g. challenge, cognitive engagement, enjoyment) and negative (e.g. confusion, frustration, boredom) were also incorporated. The intended application of the coding template was to utilise parallel coding throughout, with a ‘disruptive game feature’ code accompanied by an ‘influence on player experience code’ and, potentially, a code that denotes which component of the disruptive game design philosophy and/or framework the data may be referring to. This would enable clear organisation of the data in different ways, both ‘by feature’ as well as ‘by reported positive influence’, ‘by reported negative influence’ and ‘by philosophy/framework component’.

Three types of code formed the first version of the coding template. These were ‘F-Codes’ (i.e. Feature Codes), ‘I-Codes’ (i.e. Influence Codes), and ‘T-Codes’ (i.e. Theory Codes). This enabled individual game features to be assigned a ‘FIT’ description. That is, a reference to a particular disruptive game feature, along with its reported influence(s) on the player experience and how the player data for the feature relates to the theory underpinning the disruptive game design philosophy and framework.

\begin{table}[h]
\centering
\begin{tabular}{ll}
\hline
\textbf{VERSION 1} &  \\
\hline
\textbf{Disruptive Game Features} & (F-Codes)  \\
\hline
\textbullet{} F\_Environment & \textbullet{} F\_Difficulty  \\
\hspace{1em} o F\_SetDressing & \hspace{1em} o F\_CognitiveDifficulty  \\
\hspace{1em} o F\_PigMask & \hspace{1em} o F\_PerformativaDifficulty  \\
\hspace{1em} o F\_Non-Euclidean & \hspace{1em} o F\_Hints  \\
\textbullet{} F\_EnemyBehaviour &  \\
\hspace{1em} o F\_Wretch &  \\
\hspace{1em} o F\_Engineer &  \\
\hspace{1em} o F\_TeslaPig &  \\
\textbullet{} F\_EnemyAudio &  \\
\hline
\end{tabular}
\caption{F-Codes in the initial coding template (Version 1).}
\end{table}

The initial template had one F-Code for each disruptive game feature present in the final release of \textit{A:AMFP} (Table H.18.1). Some features, which are similar in their function and intended influence on the player experience (e.g. the different enemy behaviours), were grouped under a higher-level F-Code (e.g. \textit{F\_Wretch}, \textit{F\_Engineer}, and \textit{F\_TeslaPig} were all grouped under the \textit{F\_EnemyBehaviour} code). The high level code would only be used to code data directly if the sub-codes beneath it were not appropriate (e.g. a player discusses the behaviour of ‘enemies’ without referencing a
specific type). Where possible, more specific codes would be used; this was the same for both I-Codes and T-Codes.

<table>
<thead>
<tr>
<th>Version 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Influence on Player Experience</td>
</tr>
<tr>
<td>(I-Codes)</td>
</tr>
</tbody>
</table>

- **I_Positive**
  - I_Challenge
  - I_CognitiveEngagement
  - I_Enjoyment
  - I_Surprise

- **I_Negative**
  - I_Confusion
  - I_Frustration
  - I_Boredom
  - I_DecreasedChallenge
  - I_DecreasedCognitiveEngagement

*Table H.18.2: I-Codes in the initial coding template (Version 1).*

The initial I-Codes were split into *I_Positive* and *I_Negative* (Table H.18.2). The sub-codes beneath these refer to potential influences on players that were thought to be likely for players to report on, as well as influences that were the specific focus of the current research (e.g. *I_CognitiveEngagement* and *I_DecreasedCognitiveEngagement*).

The *I_Challenge* code was based on the aim of disruptive game design to support cognitive engagement during play but with the understanding that players may not refer to cognitive engagement explicitly. The concept of ‘challenge’ in games is a broadly applied term and thus, was used in this initial coding template as a means of coding player comments that appeared to discuss the game’s ‘challenges’ in a generally positive manner. Conversely, *I_DecreasedChallenge* was intended to be applied to comments that made reference to how a disruptive game feature actively decreased the challenge that the player experienced during play. *I_Enjoyment* was a similarly broad code that was intended to be used to code player comments referring to general, positive engagement with the game; this code was intended to be later subdivided into more specific codes. *I_Surprise* was based on the potential for players to respond to disruptive game features with shock, or surprise but in a manner that they report on as being positive, or pleasurable.

The *I_Confusion* and *I_Frustration* codes are based on the potential risks of disrupting ‘too much’, ‘too frequently’, or in ways that have minimal benefit on the player experience (Section 1.3.2). These are also based on the potential risks noted during development, with regard specific disruptive game features ‘e.g. the removal of the item inventory’s (Section 7.5.11)) potential for causing confusion and/or frustration in players rather than the desired cognitive engagement. *I_Boredom* was lastly intended to be applied to comments that were discussing generally negative engagement with the game. This code was thus intended to provide a ‘negative’ equivalent to the similarly general *I_Enjoyment* and was intended to be subdivided into more specific codes as the analysis process continued.
The I-Code template was expected to be significantly modified as the data was analysed and players report their varying responses to the game. It was also possible that sub-codes under I_Positive and I_Negative may be reported on in opposing manners by different players. For example, confusion (i.e. I_Confusion) was expected to be a negative player experience. However, it was possible that some players may find a degree of confusion enjoyable, for example, if the confusion was seen to be aligned with the state of the player-character then it may aid in enhancing a player’s ability to relate to the player-character more closely.

| VERSION 1 |
| Disruptive Game Design Theory Components |
| (T-Codes) |

Table H.18.3: T-Codes in the initial coding template (Version 1).

Lastly, the T-Codes represent the key components of the disruptive game design philosophy and framework (e.g. modes of disruption and knowledge type disruption) (Table H.18.3). The T_DisruptiveGameDesign code was intended as a ‘container’ code and was not intended to be applied to the data directly, although it would be useful for organising and querying the data within NVivo. The more specific sub-codes would instead be applied to the data directly. These codes are based on the three main ludic knowledge types (Section 3.3) (i.e. intraludic, transludic, and extraludic) and the three modes of disruption (Section 4.3) (i.e. Encoding Disruption, Recall Disruption, and Action Plan Disruption). The subdivision of memory types (i.e. procedural, semantic, and episodic) within each of intraludic, transludic and extraludic knowledge was not reflected in the initial template. This was intended to limit the number of codes being considered during the first coding pass over the data, as attempting to use too many codes early on may make the coding process less efficient. It was expected that specific codes for procedural, semantic and episodic memory references in the data would emerge at a later point in the analysis process.

The T_Expectation code was intended to be used in coding sections of data in which players are discussing their expectations of the game either prior to playing it, or as they evolved during the play process. This code was intended to allow data to be identified that could aid in developing an
understanding of different players’ games as expected, which has been identified as an important basis of comparison in how players form their game as played.

T-Codes were also created that could be used if players described game experiences that reflected features of player-supportive, counter-supportive or player-unsupportive game design (Section 1.2.3.1). The codes relating to these other types of game design may not be required. However, by placing them in the initial template, the data could be analysed in that context in order to mitigate confirmation bias occurring through coding data only in relation to disruptive game design. However, these are only four possible design approaches that exist within a design space containing an unknown number of other possible design approaches. Thus, the data analysis was carried out with the knowledge that some player data may be better described in terms of ‘other’ spaces.
H.2 Application of Initial Coding Template to Data Subset

To assess the initial applicability of the coding template (Table H.18.4) to the data set, a subset of the player data will first be examined using it, following the recommendation of King (in Clarke and Gibbs, 2012). A refined template can then be created that can be used as a basis for examining the rest of the data set.

<table>
<thead>
<tr>
<th>Version 1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Disruptive Game Features</strong> (F-Codes)</td>
</tr>
<tr>
<td>• F:_Environment o F:_SetDressing o F:_PigMask o F:_Non-Euclidean</td>
</tr>
<tr>
<td>• F:_EnemyBehaviour o F:_Wretch o F:_Engineer o F:_TeslaPig</td>
</tr>
<tr>
<td>• F:_Difficulty o F:_CognitiveDifficulty o F:_PerformativeDifficulty o F:_Hints</td>
</tr>
<tr>
<td>• F:_PlayerDeath</td>
</tr>
<tr>
<td>• F:_HidingPlaces</td>
</tr>
<tr>
<td>• F:_ItemInventory</td>
</tr>
<tr>
<td>• F:_LanternLight</td>
</tr>
<tr>
<td>• F:_LanternFlashing</td>
</tr>
<tr>
<td>• F:_EnemyAudio</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Disruptive Game Design Theory Components (T-Codes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• T:_DisruptiveGameDesign o T:_Expectation o T:_IntraludicKnowledge o T:_TransludicKnowledge o T:_ExtraludicKnowledge o T:_EncodingDisruption o T:_RecallDisruption o T:_ActionPlanDisruption o T:_LowerOrderThinking o T:_HigherOrderThinking o T:_SupportHighLevelNeeds</td>
</tr>
<tr>
<td>• T:_PlayerSupportiveDesign o T:_SupportLowLevelNeeds o T:_IncrementalAccretion</td>
</tr>
<tr>
<td>• T:_CounterSupportiveDesign o T:_RestrictLowLevelNeeds o T:_RestrictMidLevelNeeds</td>
</tr>
<tr>
<td>• T:_PlayerUnsupportiveDesign o T:_SupportWithdrawal</td>
</tr>
</tbody>
</table>

Table H.18.4: Initial coding template (Version 1).

The subset of data to be used in this stage is the ‘AMFP Member Review Thread’ from the forum (forum users tend to drop the first ‘A’ from the game’s full title, referring simply to AMFP, ‘Machine for Pigs’, or just ‘Pigs’). The thread consists of 44 individual forum pages out of the total 304 forum pages collected. This thread has been selected as it is the most likely to provide a
broad overview of discussion of many different aspects of the game as forum members provide their own comprehensive reviews of it. This thus allows the initial coding template to be assessed against a range of different potential player experiences, more so than may be possible if the data subset selected consisted of more specific topic discussions (e.g. threads specifically discussing sound in the game, or a particular level or puzzle in the game).

**H.3: Coding Procedure**

![Figure H.18.1: Tracking text coding in NVivo via the coding stripes tool.](image)

The coding of the dataset was supported by *NVivo 10*. Web pages captured from the discussion forum in PDF format were imported into the software and coded using the built in text coding tools. Coding location and density within individual web pages was tracked using *NVivo*’s coding stripe tool (Figure H.18.1).

Throughout the coding process, a separate coding memo book was maintained (see Appendix I for full page scans). Keeping research memos is a practice recommended in Grounded Theory research (Corbin and Strauss, 1990, Creswell, 2007) as a key research tool to enable reflexivity in the work (i.e. awareness of, and critical self-reflection on, the ‘abstract’ thoughts of the researcher whilst analysing the data and how these may influence any eventual discussion of the data). However, making memos during any form of qualitative research is good practice as it provides a form of ‘audit trail’ for the researcher’s thoughts and developing interpretations of the data. Thus, in the current research, the memo book was used to identify possible links between different parts of the dataset, potential points for further consideration, or any elements of the data, ideas, or interpretations that may be useful structural points during the writing up of the study results and discussion.
H.4: Refining the Coding Template after Analysis of Data Subset

Following the exploration of the data subset using the initial coding template, as anticipated a number of new themes emerged from the data. Similarly, some of the disruptive game features were not reported on by players or only reported on in a small number of cases. The coding template was refined into Version 2, in order to accommodate these new themes. The F-Codes (i.e. disruptive game features) remained unchanged as they referred directly to fixed features in the released game. Likewise, the T-Codes remained unchanged as they also referred to defined components of the previously established theory underpinning the disruptive game design philosophy and framework. It is however worth noting that the requirement to create new T-Codes relating to specific instances of data related to procedural, semantic, and episodic memory was not encountered. In this data subset at least, players did not provide data that could be interpreted as referring to specific knowledge types at that level.

<table>
<thead>
<tr>
<th>Version 2</th>
<th>Influence on Player Experience (I-Codes)</th>
</tr>
</thead>
</table>
| **I_Positive** | o I_Challenge  
| | o I_CognitiveEngagement  
| | o I_Enjoyment  
| | o I_Surprise  
| | o I_Immersion  
| | o I_Horrifying  
| | o I_LastingImpact  
| | o I_WhatIsImportantToPlayers  
| | o I_UndecidedOpinion  
| **I_Negative** | o I_Confusion  
| | o I_Frustration  
| | o I_Boredom  
| | o I_DecreasedChallenge  
| | o I_DecreasedCognitiveEngagement  
| | o I_NotScary  
| | o I_NotRelatable  
| | o I_FeelingBetrayed  
| | o I_Predictable  
| | o I_Disappointed  
| | o I_NoExploration  |

Table H.18.5: I-Codes in the refined coding template (Version 2). New codes in this version are highlighted in bold and underlined.

The list of I-Codes is expanded in Version 2 of the template (Table H.18.5). Within the I_Positive top-level code, I_Immersion, I_Horrifying, and I_LastingImpact are added. These reflect, respectively, multiple positive comments from different players relating to their sense of ‘immersion’ during play, their enjoyment of being scared during play, and the lasting impact that the game had on them after they stopped playing (e.g. continuing to think about the story, or the themes of the game). These were cited as positive elements of the game as played and thus were important to identify at this stage, so as to explore whether they had been caused by disruptive game features or not.

Within the I_Negative top-level code, new codes were added to reflect multiple player comments relating to negative elements of their play experience. I_NotScary was used to identify features of the game that reduced or eliminated the player’s sense of fear. I_NotRelatable was used to code player comments that suggested players had a problem relating to the virtual ‘reality’ of the game.
(i.e. the player felt unable to relate to the world, the characters, or elements of the game’s story). 

`I_NoExploration` was used to code game features whose presence in the game was felt to negate the need to explore the game environment. This was cited by multiple players and occurred frequently enough to suggest a possible design flaw that required further exploration. `I_Predictable` was used to code data in which players noted that their expectations had been met and that, in doing so, the game was too predictable. This issue was cited by multiple players and suggests that parts of the game may exist in which either, disruptive game features have explicitly failed at disrupting player knowledge and expectation or, additional disruptive game features may have been beneficial in mitigating the game’s predictability. Lastly, `I_FeelingBetrayed` and `I_Disappointed` are codes that emerged based on repeatedly used words and phrases by some players in reference to how the game experience (i.e. the `game as played`) made them feel more generally. These are particularly notable codes as players make reference to what they expected to get from the game (i.e. their individual `games as expected`).

Outside of the `I_Positive` and `I_Negative` codes, two new top-level codes were added into Version 2 of the I-Codes template; `I_WhatIsImportantToPlayers` and `I_UndecidedOpinion`. `I_WhatIsImportantToPlayers` was used to code data in which players described what they individually felt was important for them in their gameplay experience and how `A:AMFP` met, or did not meet, those criteria. `I_UndecidedOpinion` was used as a general code for data in which players did not definitively state a positive or negative response to a particular game feature.

### Version 2

**Non-Disruptive Features**

(N-Codes)

- **N_LackingMechanics**
  - **N_Sanity**
  - **N_Puzzles**
  - **N_NoInteractivity**

- **N_Story**
  - **N_StoryDelivery**
  - **N_GameplayStoryMismatch**

- **N_Atmosphere**

**Table H.18.6: N-Codes in the refined coding template (Version 2).**

Two new code types were added to Version 2 of the coding template, the first of which was Non-Disruptive Features (N-Codes) (Table H.18.6). These were used to code data in which players referred to game features that were part of the game as designed and game as created but which were explicitly not disruptive game features. The `N_LackingMechanics` was a broadly applicable code used to identify data in which players were discussing specific features of `A:TDD` that were not present in `A:AMFP`. Such discussions demonstrated players making comparisons between two games in a franchise and also, between the `game as expected` and the `game as played`. `N_Story` was used to code player discussion of story elements at a broad level, although the sub-codes of `N_StoryDelivery` and `N_GameplayStoryMismatch` were used more frequently to identify more specific information related to how the told story was delivered to the player and how those methods of delivery worked alongside the actions of the player (i.e. the ‘gameplay’). Lastly,
**N_Atomosphere** was used to code data in which players discussed the ‘feel’ of the game world. Such discussion was often placed in the context of *A:TDD* and the player’s expectations of *A:AMFP* thus making it relevant to an exploration of the importance of the game as expected in influencing the game as played.

**Table H.18.7: C-Codes in the refined coding template (Version 2).**

The second new code type in Version 2 of the template was Conceptual Themes (C-Codes) (Table H.18.7). These were used to group together data that was representative of broader ‘themes’ that may be relevant to discuss in the analysis or that may provide different perspectives on the influence of disruptive game design on players. Some of these themes were not directly relevant to the primary focus of the research however they occurred frequently enough to warrant separate codes that may provide contextual information for other findings. For example, the players’ perceived value of the game (i.e. C_Value) is not directly relevant to investigating the impact of disruptive game design on cognitive engagement. However, perceived value may influence how players construct their *game as played* in relation to their *game as expected* which in turn, may influence how they respond to features in the game (both disruptive and non-disruptive).

Some of the C-Codes are named specifically to reflect phrases and terminology used repeatedly by players on the forum. For example, C_BaitAndSwitch refers to the repeated discussion by some players about how they felt the game’s advertising had not been representative of the final released game, in terms of the type of gameplay that players would be engaging with. These players referred to this as ‘bait and switch’ tactics being used by both The Chinese Room and Frictional Games. Likewise, C_RoseTintedMemoryOfATDD refers to a range of different forum comments across multiple threads in which players were referring to features of *A:AMFP* in relation to features that they remember being in *A:TDD*. A number of these comments were responded to by other players that identified that some of these ‘remembered features’ were not present in *A:TDD* or, that they were being remembered by players as more significant than they actually were. A recurring term used in these discussions was players recalling a ‘rose tinted’ or ‘enhanced’ version of *A:TDD*, then comparing *A:AMFP* to this inaccurate memory.

Version 2 of the coding template is presented in full in Table H.18.8, with codes that have been added since Version 1 highlighted in bold and underlined.
<table>
<thead>
<tr>
<th>Disruptive Game Features (F-Codes)</th>
<th>Influence on Player Experience (I-Codes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• F_Environment</td>
<td>• I_Positive</td>
</tr>
<tr>
<td>o F_SetDressing</td>
<td>o I_Challenge</td>
</tr>
<tr>
<td>o F_PigMask</td>
<td>o I_CognitiveEngagement</td>
</tr>
<tr>
<td>o F_Non-Euclidean</td>
<td>o I_Enjoyment</td>
</tr>
<tr>
<td>• F_EnemyBehaviour</td>
<td>o I_Surprise</td>
</tr>
<tr>
<td>o F_Wretch</td>
<td>o I_Immersion</td>
</tr>
<tr>
<td>o F_Engineer</td>
<td>o I_Horrifying</td>
</tr>
<tr>
<td>o F_TeslaPig</td>
<td>o I_LastingImpact</td>
</tr>
<tr>
<td>• F_Difficulty</td>
<td>• I_Negative</td>
</tr>
<tr>
<td>o F_CognitiveDifficulty</td>
<td>o I_Confusion</td>
</tr>
<tr>
<td>o F_PerformativeDifficulty</td>
<td>o I_Frustration</td>
</tr>
<tr>
<td>o F_Hints</td>
<td>o I_Boredom</td>
</tr>
<tr>
<td>• F_PlayerDeath</td>
<td>o I_DecreasedChallenge</td>
</tr>
<tr>
<td>• F_HidingPlaces</td>
<td>o I_DecreasedCognitiveEngagement</td>
</tr>
<tr>
<td>• F_ItemInventory</td>
<td>o I_NotScary</td>
</tr>
<tr>
<td>• F_LanternLight</td>
<td>o I_NotRelatable</td>
</tr>
<tr>
<td>• F_LanternFlashing</td>
<td>o I_FeelingBetrayed</td>
</tr>
<tr>
<td>• F_EnemyAudio</td>
<td>o I_Predictable</td>
</tr>
<tr>
<td>• F_PlayerDeath</td>
<td>o I_Disappointed</td>
</tr>
<tr>
<td>• F_HidingPlaces</td>
<td>o I_NoExploration</td>
</tr>
<tr>
<td>• F_ItemInventory</td>
<td>• I_WhatIsImportantToPlayers</td>
</tr>
<tr>
<td>• F_LanternLight</td>
<td>• I_UndecidedOpinion</td>
</tr>
<tr>
<td>• F_LanternFlashing</td>
<td></td>
</tr>
<tr>
<td>• F_EnemyAudio</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Disruptive Game Design Theory Components (T-Codes)</th>
<th>Non-Disruptive Features (N-Codes)</th>
<th>Conceptual Themes (C-Codes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• T_DisturbingGameDesign</td>
<td>• N_LackingMechanics</td>
<td>• C_BaitAndSwitch</td>
</tr>
<tr>
<td>o T_Expectation</td>
<td>o N_Sanity</td>
<td>• C_Con­sideringTheDeveloper</td>
</tr>
<tr>
<td>o T_IntraludicKnowledge</td>
<td>o N_Puzzles</td>
<td>• C_EvolvingGames</td>
</tr>
<tr>
<td>o T_TransludicKnowledge</td>
<td>o N_NoInteractivity</td>
<td>• C_Fans</td>
</tr>
<tr>
<td>o T_Extra ludicKnowledge</td>
<td></td>
<td>• C_RoseTintedMemoryOfATDD</td>
</tr>
<tr>
<td>o T_EncodingDisruption</td>
<td></td>
<td>• C_UnsatisfiedMoreLikelyToPost</td>
</tr>
<tr>
<td>o T_RecallDisruption</td>
<td></td>
<td>• C_Value</td>
</tr>
<tr>
<td>o T_ActionPlanDisruption</td>
<td></td>
<td>o C_GamePrice</td>
</tr>
<tr>
<td>o T_LowerOrderThinking</td>
<td></td>
<td>o C_GameLength</td>
</tr>
<tr>
<td>o T_HigherOrderThinking</td>
<td></td>
<td></td>
</tr>
<tr>
<td>o T_SupportHighLevelNeeds</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• T_PlayerSupportiveDesign</td>
<td>• T_CounterSupportiveDesign</td>
<td></td>
</tr>
<tr>
<td>o T_SupportLowLevelNeeds</td>
<td>o T.RestrictLowLevelNeeds</td>
<td></td>
</tr>
<tr>
<td>o T_IncrementalAccretion</td>
<td>o T.RestrictMidLevelNeeds</td>
<td></td>
</tr>
<tr>
<td>• T_PlayerUnsupportiveDesign</td>
<td>T_SupportWithdrawal</td>
<td></td>
</tr>
<tr>
<td>o T_SupportWithdrawal</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table H.18.8: Refined coding template (Version 2).
H.5: Modification of Coding Template during First Coding Pass of Dataset

As coding of the dataset proceeded, new codes emerged while others were removed. This ongoing modification of the coding template is in line with King’s (2014a) recommendations for template development, which suggest that there are four methods of developing a template: adding and deleting codes, changing the scope of code if it is evidently too narrow or too broad, or changing the hierarchical structure of top-level codes and sub-codes.

This modification of the template can be an ongoing process while the data is being coded for the first time, with the expectation that the data will be fully coded at least a second time to ensure that the same codes have been considered in the context of the full data set. The first full coding pass on the complete data set resulted in Version 3 of the template.

| Version 3 |
| Disruptive Game Design Theory Components (T-Codes) |

- T_DisruptiveGameDesign
  - T_Expectation
  - T_IntraludicKnowledge
  - T_TransludicKnowledge
  - T_ExtraludicKnowledge
  - T_EncodingDisruption
  - T_RecallDisruption
  - T_ActionPlanDisruption
  - T_LowerOrderThinking
  - T_HigherOrderThinking
  - T_SupportHighLevelNeeds

- T_PlayerSupportiveDesign
  - T_SupportLowLevelNeeds
  - T_IncrementalAccretion

- T_CounterSupportiveDesign
  - T_RestrictLowLevelNeeds
  - T_RestrictMidLevelNeeds

- T_PlayerUnsupportiveDesign
  - T_SupportWithdrawal

Table H.18.9: T-Codes in Version 3 of the coding template.

A minor modification to the T-Codes section of the template was made in Version 3 (Table H.18.9), with the removal of the T_PlayerUnsupportiveDesign code and its associated T_SupportWithdrawal sub-code. After one full coding pass over the dataset there were no occurrences of either of these codes.

However, the presence of a small number of T_CounterSupportiveDesign and T_PlayerSupportiveDesign codes (1 occurrence of the T_RestrictMidLevelNeeds code and 4 occurrences of the T_IncrementalAccretion code, respectively) demonstrate that there was some, limited, benefit for including these other design approaches in the initial template, rather than only focusing on finding evidence of disruptive game design.
**Version 3**

**Non-Disruptive Features (N-Codes)**

- **N_LackingMechanics**
  - **N_Sanity**
  - **N_Puzzles**
  - **N_NoInteractivity**
  - **N_NoEnvironmentInteractivity**

- **N_Story**
  - **N_StoryDelivery**
  - **N_GameplayStoryMismatch**
  - **N_Atmosphere**

Table H.18.10: N-Codes in Version 3 of the coding template.

A minor modification was also made to the N-Codes section of the template in Version 3 (Table H.18.10), with the removal of the *N_NoInteractivity* code. This was replaced with the more descriptive *N_NoEnvironmentInteractivity* code to reflect the fact that player discussion around the interactivity of objects in the game focused almost exclusively on the lack of interactivity of non-critical, decorative environmental objects, such as furniture, books, bottles and other small items.

**Version 3**

**Influence on Player Experience (I-Codes)**

- **I_Challenge**
- **I_CognitiveEngagement**
- **I_Enjoyment**
- **I_Surprise**
- **I_Immersion**
  - **I_NotRelatable**
- **I_Horrifying/Psychological**
- **I_Terrifying/Visceral**
- **I_Threat**
- **I_LastingImpact**
  - **I_EmotionallImpact**
  - **I_Post-PlayDebate**
- **I_Confusion**

- **I_Frustration**
  - **I_Patronising**
  - **I_Boredom**
  - **I_DecreasedChallenge**
  - **I_DecreasedCognitiveEngagement**
  - **I_NotScary**
  - **I_FeelingBetrayed**
  - **I_Predictable**
  - **I_Disappointed**
  - **I_NoExploration**
  - **I_WhatIsImportantToPlayers**
  - **I_UndecidedOpinion**
  - **I_HowYouPlayTheGame**

Table H.18.11: I-Codes in Version 3 of the coding template.

Multiple changes were made to the I-Codes section of the template in Version 3 (Table H.18.11). The organisation of I-Codes into the two broad *I_Positive* and *I_Negative* codes was removed, as there were multiple instances of players reporting positive experiences that were expected to be negative (e.g. *I_Confusion*). Similarly, experiences that initially appeared to be negative for many players (e.g. *I_NoExploration*) were described in positive terms by other players. This demonstrated that the concept of ‘influences(s) on the player’ could not be readily subdivided into such broad categories as simply positive and negative experiences and would require closer analysis. This change was also supported by the addition of the *I_HowYouPlayTheGame* code, which was used to code data that emphasised the different experiences had by players, due to how they approached the game (i.e. their individual play style, and their ‘attitude’ towards the game, the franchise, the genre, and games more generally). The *I_WhatIsImportantToPlayers* code was also
added to further code data that discussed individual players’ preferences for their game experiences.

Further to this, the language used by players when discussing different types of emotional reaction to the game suggested a broad split between ‘horror’ as a ‘psychological’ response and ‘terror’ as a response to the game’s ‘visceral’ qualities (i.e. the I_Horrifying/Psychological code and the I_Terrifying/Visceral code). For example, this may include the difference between players encountering an enemy agent first-hand (a ‘visceral’/‘terrifying’ experience) and players hearing ambiguous but unnerving sounds in the environment (a ‘psychological’/‘horrifying’ experience). The I_Threat code was added to code data in which players discussed game features that influenced their sense of ‘threat’ either from the enemy agents or from the game environment more generally. These influences were both positive and negative but were grouped at this stage under the single code for closer analysis later in the process.

The I_LastingImpact code was then expanded with two sub-codes. I_EmotionallImpact was used to code data in which players discussed instances in which they continued to be effected by their play experience after they stopped playing, while I_PostPlayDebate was used to code data in which multiple players were discussing different interpretations of elements of the game. Lastly, the I_Frustration code had a sub-code added called I_Patronising which reflected the emerging theme of multiple players citing frustration at what they felt were ‘patronising’ hints provided by the game that did not allow them to think through puzzles and problems on their own.

<table>
<thead>
<tr>
<th>VERSION 3</th>
<th>Other Topics (O-Codes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• O_Diff.InterpretationsOfFeatures</td>
<td>• O_EvolvingGames</td>
</tr>
<tr>
<td>• O_FranchiseComparisons</td>
<td>• O_Fans</td>
</tr>
<tr>
<td>• O_GenreComparisons</td>
<td>• O_RoseTintedMemoryOfATDD</td>
</tr>
<tr>
<td>• O_MechanicsThatWereRemoved</td>
<td>• O_UnsatisfiedMoreLikelyToPost</td>
</tr>
<tr>
<td>• O_IntendedAudience</td>
<td>• O_Value</td>
</tr>
<tr>
<td>• O_BaitAndSwitch</td>
<td>o O_GamePrice</td>
</tr>
<tr>
<td>• O_ConsideringTheDeveloper</td>
<td>o O_GameLength</td>
</tr>
</tbody>
</table>

Table H.18.12: O-Codes in Version 3 of the coding template.

Previously in Version 2 of the coding template, C-Codes were identified as broad conceptual themes that were emerging from the data. However, as more of the dataset was coded, it became evident that these themes were often not relevant to the primary focus of the research or, were only discussed in a small subset of the dataset. A ‘theme’ as specifically defined by King (2014b, para.1) in relation to Template Analysis is a feature “of participants’ accounts characterising particular perceptions and/or experiences that the researcher sees as relevant to the research question”. Thus, the suggested conceptual themes in Version 2 of the template were more accurately describable simply as ‘Other Topics’ that were being discussed in the dataset (i.e. O-Codes) (Table H.18.12). Further analysis of the data may identify that some of these topics may
indeed by research-relevant themes, however this was not readily apparent following the first coding pass.

In addition to the previously identified codes in this section, five new O-Codes emerged following the completion of the first coding pass. **O_Diff.InterpretationsofFeatures** was used to match separate comments in the data in which different players discussed notably different interpretations of the same game feature. **O_FranchiseComparisons** and **O_GenreComparisons** identified data in which players referred to either A:TDD or the Penumbra series (i.e. the previous FG franchises) or to other ‘horror’/’adventure’ games more broadly (i.e. the genre). **O_MechanicsThatWereRemoved** was used to code sections of the data in which players made suggestions of game features they would have liked to have seen in the game, where those suggestions were similar or identical to a game feature in the *game as designed* that was not included in the *game as published*. Lastly, **O_IntendedAudience** was used to code data in which players were discussing what they thought the game’s intended target audience was and how this may have influenced the game’s design and eventual reception by different types of player.

Version 3 of the coding template is presented in full in Table H.18.13, with codes that have been added since Version 2 highlighted in bold and underlined and codes removed since Version 2 struck through.
### Disruptive Game Features (F-Codes)
- **FEnvironment**
  - F_SetDressing
  - F_PigMask
  - F_Non-Euclidean
- **FEnemyBehaviour**
  - F_Wretch
  - FENGINEER
  - F_TeslaPig
- **FDifficulty**
  - F_CognitiveDifficulty
  - F_PerformativeDifficulty
  - F_Hints
- **FPlayerDeath**
- **F-HidingPlaces**
- **F-ItemInventory**
- **FLanternLight**
- **FLanternFlashing**
- **F-EnemyAudio**

### Disruptive Game Design Theory Components (T-Codes)
- **TDisruptiveGameDesign**
  - T_Expectation
  - T_IntraludicKnowledge
  - T_TransludicKnowledge
  - T_ExtraludicKnowledge
  - T_EncodingDisruption
  - T_RecallDisruption
  - T_ActionPlanDisruption
  - T_LowerOrderThinking
  - T_HigherOrderThinking
  - T_SupportHighLevelNeeds
- **TPlayerSupportiveDesign**
  - T_SupportLowLevelNeeds
  - T_IncrementalAccretion
- **TCounterSupportiveDesign**
  - T_RestrictLowLevelNeeds
  - T_RestrictMidLevelNeeds
- **TPlayerUnsupportiveDesign**
  - T_SupportWithdrawal

### Non-Disruptive Features (N-Codes)
- **NLackingMechanics**
  - N_Sanity
  - N_Puzzles
  - N_NoInteractivity
  - N_EnvironmentInteractivity
- **NStory**
  - N_StoryDelivery
  - N_GameplayStoryMismatch
- **NAtmosphere**

### Influence on Player Experience (I-Codes)
- **IChallenge**
- **ICognitiveEngagement**
- **IEnjoyment**
- **ISurprise**
- **IImmersion**
  - I_NotRelatable
- **IHorrifying/Psychological**
- **ITerrifying/Visceral**
- **IThreat**
  - I_LastingImpact
    - I_EmotionaImpact
    - I_Post-PlayDebate
  - I_Confusion
  - I_Frustration
    - I_Patronising
  - I_Boredom
  - I_DecreasedChallenge
  - I_DecreasedCognitiveEngagement
  - I_NotScary
  - I_FeelingBetrayed
  - I_Predictable
  - I_Disappointed
  - I_NoExploration
  - I_WhatIsImportantToPlayers
  - I_UndecidedOpinion
  - I_HowYouPlayTheGame

### Other Topics (O-Codes)
- **ODiff.InterpretationsOfFeatures**
- **OFranchiseComparisons**
- **OGenreComparisons**
- **OMechanicsThatWereRemoved**
- **OIntendedAudience**
- **OBaitAndSwitch**
- **O_ConsideringTheDeveloper**
- **OEvolvingGames**
- **OFans**
- **ORoseTintedMemoryOfATDD**
- **OUnsatisfiedMore Likely ToPost**
- **OValue**
  - O_GamePrice
  - O_GameLength

Table H.18.13: Refined coding template (Version 3)
Appendix I: Coding Memo Book

Data Subset Coding Notes

- What do players respond to first?
  - Aesthetic features?
  - Graphics/Sound?
  - Story?
  - Soundtrack?

- Review Thread PL: Unhappy players more vocal?
- Link to Lit?

- How many "negative" reviews reflect the final "score"?

- The core "complaints" are nearly all "in comparison to ATDD."

- Players identifying things such as:
  - Monster spawns differently.
  - "Safe spots" fading. Players want to be under "attack constants."

- Players want more voiced notes.

- Thinking about game even afterwords: Post-Play engagement.

- Custom Stories seem very important for added value.

- Game as a platform for user creation?

- The mindset of fans is very bizarre: "No choice but to buy a game they knew they wouldn't enjoy."

- ATMP actually "wins ATDD for some players.

- Inconsistency confuses players.
London Steam both loved and loathed by different players.

Clear polarisation of opinion - esp. removal of mechanics.

Is disappointment linked to players that want 'gameplay' over story?

Rovia Thread P(30) - Not one mention of story in crispophobia's leisure.

Look at response to Silent Hill: Shattered Memories?

GENRE EXPECTATION and classification on different review sites.

...Many players seem to feel cheated out of mechanics at an individual level.

Review Thread (P. 26)

"Is it a game?" Review Thread (P. 29)

Players and the Lantern - is there an assumption that the first interpretation will not change?

The recontextualisation of behaviour is troubling debate.

Jump Scores - None, or lets."
Review Thread (P12) - Perfect explanation of Trans Atlantic Disruption on Pig Encounter.

- Blue fog explanation - the submitting to the game - the ludic attitude.

- Enemy Types too limiting (crawling enemies).

- Schema balking during recall - different interpretations

- "I haven't beaten it yet" - Review Thread (P12).

- "Scores" as a review category.

- "A Plimsy Machine" Thread - people will always pick logic holes in stories!

- "A question for you all" (P2) "Human form monsters" (P2) - Eldritch enough?

- Bioshock Infinite plot comparison.

- Potted quote on (P2)

- "Tesla Pig Thread: huge success." - Mismatch, out of place with preceding game style."
The issue of audience
you can't appeal
to anyone.

A ludic attitude

The smell? Episodic olfactory

A comp to ATOD poll

A "realism" in the story—how
is ATOD "realistic"?

Perhaps a discussion of
playing for... (rassism, media,
youtube, etc)

A split X Fons
to separate
Personal Entitlement

Defensive?

Riding off of Arnessia's
success—"cash-in"

Defensive? Buyer's remorse?

Also, post-purchase
rationalisation

Could suggest dataset
weakness?

Best suited to mature... (p.4)
"I couldn't allow me
to imagine..."

Should this be in chap 6?

The lack of voiced notes
to mitigate, suggested
interpreted
Cognitive dissonance?
- use content
- story hidden?
- Exaludic knowledge changes
- ability to immerse.
- X. Bolt and Switch
- game as expected.
- Disappointed in wasted
- potential.
- Already disappointed by
- reviews before playing.

Reference
- Sherry in Outlast
- it's like an honor most fit
- the ATDD template.
- Did we simply misjudge our
- audience?
- players as moderators/creators:
- knowing the tricks?
- Make, make genre
- What do players
- critique first?
- Focus on milieu
- Mission predictable/cliche.
- What is "horror" for players?
- Pych vs. Scary.
& Introductory - those with children more affected

& Horror - not 'scary' but horrifying. - Lovecraft
  fear & sickness

& "Nostalgia goggles" - in review.

& Puzzle logic helps immersion in game.
  - mechanics fluid and instinctive.

- Hearing printed (096) - should have
  (099) been minimal
  - font for not reading, marketing!

- 'Levels' of narrative
  interpretation reflect how cognitively 'invested' players were.
  Descriptive versus 'interative'!

- Assumption that things are there to be overtly scary -
  not unsettling, horrifying.

- Justification at lantern fuel
  darkness by narrative/character.

- Some players clearly
  'meek' given cues
  'game world logic'
  given
Consolidation of Observations and Memos

Players tend to describe or critique certain game components first in their comments - specifically, scores and comparisons to ATDD.

A possible methodological weakness - does the dataset bias towards unsatisfied customers? Is there data to support this in literature?

RELATED TO

Cynical player attributes - suggestion of marketing 'bait and switch'.

An element of buyer's remorse?

Or, attempts to rationalise purchase?

Subtle randomisation of game systems has provoked debate amongst players - this is good!

Some players want to be 'constantly under threat' - how can developers hope to meet such expectations? This could destroy the game's pacing...

What players think they want - this can be linked to Gramscatro post-modern comments perhaps?
<table>
<thead>
<tr>
<th>Players feel betrayed, or why players play more</th>
<th>Cheated by removal of mechanics.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reasons beyond simply enjoyment.</td>
<td>Youtube, tourism, support dev.</td>
</tr>
<tr>
<td>A sequel should keep things that were in the original.</td>
<td>Perhaps more critical because the more is better notion.</td>
</tr>
<tr>
<td>Reconstructed memory of ATDD inaccurate.</td>
<td>Players of mod-able games know the tricks of the game engine.</td>
</tr>
<tr>
<td>The &quot;game as expected&quot; may be based on false memories.</td>
<td>Submitting to the logic of contexts within episodic attitude - explaining things in narrative terms, rather</td>
</tr>
<tr>
<td>Do players remember a game, the mechanical terms, or do they remember the experience of a game and try to justify it post-hoc?</td>
<td>Does disruptive design cater more specifically to those looking for this type of experience?</td>
</tr>
</tbody>
</table>
Appendix J: Coding Template Version 3 Code Definition

List

J.1: Disruptive Game Features (F-Codes)

F_ENVIRONMENT:
Text discussing the structure, layout or aesthetic qualities of the game world.

F_SETDRESSING:
Text discussing the appearance/disappearance of elements of the game world such as doors or ornaments.

F_PIGMASK:
Text discussing the disruptive game feature, the Pig Mask Motif.

F_NON-EUCLIDEAN:
Text discussing the use of the non-Euclidean geometry disruptive game feature.

F_ENEMYBEHAVIOUR:
Text discussing enemy behaviour but that does not refer specifically to any particular enemy type or enemy encounter.

F_WRETCH:
Text discussing encounters with, or the behaviour of, the Wretch enemy type.

F_ENGINEER:
Text discussing encounters with, or the behaviour of, the Engineer enemy type.

F_TESLAPIG:
Text discussing encounters with, or the behaviour of, the Tesla Pig enemy type.

F_DIFFICULTY:
This is a container code for a range of difficulty-based codes. No text is coded with this code directly.

F_COGNITIVEDIFFICULTY:
Text discussing difficulty in the game stemming from non-performative challenge (e.g. logic puzzles).
F_PerformativeDifficulty:
Text discussing difficulty in the game stemming from performative challenges (e.g. out-manoeuvring an enemy agent).

F_Hints:
Text discussing the in-game method for providing the player with puzzle hints.

F_PlayerDeath:
Text discussing player experiences of ‘dying’ during gameplay.

F_HidingPlaces:
Text discussing player experiences of hiding from enemies in the game, or their experiences of not having to hide from enemies.

F_ItemInventory:
Text discussing the removal of the item inventory.

F_LanternLight:
Text discussing the aesthetic and functional changes to the lantern light, especially its narrow, focused beam (compared to ATDD’s omnidirectional light source) and the removal of the need to keep the lantern fuelled (as the player had to in ATDD).

F_LanternFlashing:
Text discussing the lantern flashing system specifically (as opposed to other lantern-based features).

F_EnemyAudio:
Text discussing the audio cues attached to the enemy agents.

J.2: Influences on the Player Experience (I-Codes)
I_Challenge:
Text that discusses a game feature that provided a positive, performative challenge for players.

I_CognitiveEngagement:
Text that discusses a game feature that provided a positive, cognitively engaging experience for players.
I_Engoyment:

This code is used to code text that discusses positive responses to the game experience but doesn’t provide detail as to how that enjoyment is specifically experienced (e.g. statements such as “I really liked not having a separate inventory screen”, or “the final level was really good!”).

I_Surprise:

Text that discusses explicitly either a positive or negative unexpected or surprising experience during gameplay.

I_Immersion:

Text that discusses the impact of a game feature on the player’s reported sense of immersion in the game experience. To avoid researcher subjectivity with this particular term, this code is only used where players explicitly refer to the term ‘immersion’, or associated stem words (e.g. immersed, immersive).

I_NotRelatable:

Text that discusses a player’s negative experiences related to being unable to relate to an element of the game, especially the player-character, other non-player-characters, or elements of the game’s story.

I_Horrifying/Psychological:

Text that discusses how a game feature has impacted a player’s reported state of mind through being ‘horrifying’, ‘unsettling’, ‘disturbing’, or other similar psychological means.

I_Terrifying/Visceral:

Text that discusses how a game feature has impacted a player’s reported state of mind or physiological state by being ‘terrifying’, ‘scary’, ‘visceral’, or other similar ‘jump’- or ‘shock’-based means.

I_LastingImpact:

Text that discusses how the game experience has ‘stayed with the player’ after they have finished playing. Also used to code text that refers to the game’s ‘replayability’ or refers to players playing the game more than once.

I_EmotionalImpact:

Text that discusses specific emotions that players have reported being influenced by the game experience (e.g. empathy, pity, sadness, anger).
**I_Post-Play Debate:**

This code is used to block-code large sections of debate, disagreement or argument between players about features in the game (e.g. interpretations of the game story).

**I_Confusion:**

Text discussing a game feature that confused players. This code is used to code text that frames this confusion as a negative experience. 'Positive' confusion is coded using I_CognitiveEngagement.

**I_Frustration:**

Text discussing a game feature that caused frustration or anger in players. If the frustration (or similar) was not caused by an in-game feature, it is coded instead using O_Fans.

**I_Patronising:**

Text discussing a game feature that players felt was patronising or did not allow them to solve a problem themselves.

**I_Boredom:**

Text discussing a game feature that players felt was tedious, slow, pointless, or otherwise was not entertaining.

**I_DecreasedChallenge:**

Text discussing a game feature that provided a negative, decreased sense of performative challenge for the player.

**I_DecreasedCognitiveEngagement:**

Text discussing a game feature that provided a negative, decreased sense of cognitive engagement for the player.

**I_NotScary:**

Text discussing a game feature that negatively impacted the player’s sense of being ‘scared’ or ‘fearful’.

**I_FeelingBetrayed:**

Text that refers to a player’s emotions towards either The Chinese Room and/or Frictional Games. This is used to group the collection of players that feel let down or betrayed by the developer/publisher because the game is so different to ATDD.
**I_Predictable:**

Text discussing a game feature that made the gameplay and/or story predictable. This is used to code text where this predictability is reported as being a negative factor.

**I_Disappointed:**

Text that refers explicitly to the term 'disappointed', or associated stem words (e.g. disappointment).

**I_NoExploration:**

Text discussing a player’s negative reporting of the game’s ‘lack of exploration’ or ‘linearity’.

**I_WhatIsImportantToPlayers:**

A broad code used in parallel with other codes to provide specific context. This code is used when players refer to ‘their particular gaming preferences’ or ‘they play (horror) games’, or similar idiographic perspectives on what they feel is most important to their game experience.

**I_HowYouPlayTheGame:**

A broad code used in parallel with other codes to provide context. This code is used to code text in which players refer to experiencing identical game features or scenarios differently depending on how they played the game (e.g. play style or attitude).

**J.3: Disruptive Game Design Theory Components (T-Codes)**

**T_DisruptiveGameDesign:**

This is a container code for a range of disruptive game design codes. No text is coded with this code directly.

**T_Expectation:**

A broad code used in parallel with other codes to provide context. This code is used to code text where players refer to their expectations and how the game either met, or did not meet them. This code is used for both positive and negative statements, with parallel codes used to separate the positive and negative statements.

**T_IntraludicKnowledge:**

This code is used to refer to any type of disruption (i.e. Encoding Disruption, Recall Disruption or Action Plan Disruption) of any type of intraludic knowledge. Used in conjunction with T_EncodingDisruption, T_RecallDisruption, and T_ActionPlanDisruption to further organise player statements.
T_TransludicKnowledge:

This code is used to refer to any type of disruption (i.e. Encoding Disruption, Recall Disruption or Action Plan Disruption) of any type of transludic knowledge. Used in conjunction with T_EncodingDisruption, T_RecallDisruption, and T_ActionPlanDisruption to further organise player statements.

T_ExtraludicKnowledge:

This code is used to refer to any type of disruption (i.e. Encoding Disruption, Recall Disruption or Action Plan Disruption) of any type of extraludic knowledge. Used in conjunction with T_EncodingDisruption, T_RecallDisruption, and T_ActionPlanDisruption to further organise player statements.

T_EncodingDisruption:

This code is used to refer to any instances in which player statements are interpreted as representing Encoding Disruption (either via ambiguous stimulus information or purposeful distraction). Used in conjunction with T_IntraludicKnowledge, T_TransludicKnowledge, and T_ExtraludicKnowledge to further organise player statements.

T_RecallDisruption:

This code is used to refer to any instances in which player statements are interpreted as representing Recall Disruption. Used in conjunction with T_IntraludicKnowledge, T_TransludicKnowledge, and T_ExtraludicKnowledge to further organise player statements.

T_ActionPlanDisruption:

This code is used to refer to any instances in which player statements are interpreted as representing Action Plan Disruption. Used in conjunction with T_IntraludicKnowledge, T_TransludicKnowledge, and T_ExtraludicKnowledge to further organise player statements.

T_Trans/ExtraLudicKnowledgeUse:

Text discussing how a player attempted to make sense of a game feature in AAMFP by referring to either transludic or extraludic knowledge.

T_LowerOrderThinking:

This code is used to refer to any instances in which player statements are interpreted as being representative of the use of lower-order thinking skills (i.e. Recall/Remembering, Understanding, and Application of remembered and understood knowledge).
**T_HigherOrderThinking:**

This code is used to refer to any instances in which player statements are interpreted as being representative of the use of higher-order thinking skills (i.e. Analysis and/or Evaluation of new stimulus information, and the Creation of new knowledge).

**T_SupportHighLevelNeeds:**

This code is used to refer to any instances in which player statements are interpreted as representing the experiencing of high-level needs (i.e. Level 5 needs in Maslow’s Hierarchy).

**T_PlayerSupportiveDesign:**

This is a container code used for organisational purposes. No text is coded with this code directly.

**T_IncrementalAccretion:**

Text discussing how a player learned or developed their understanding of a game feature via ‘gradual progression’, ‘incremental accretion’ or similar terms that suggest a steady, ongoing process.

**T.CounterSupportiveDesign:**

This is a container code used for organisational purposes. No text is coded with this code directly.

**T_RestrictMidLevelNeeds:**

Text discussing a game feature that a player feels restricted one or more of their mid-level needs (i.e. needs at level 3 or 4 of Maslow’s hierarchy, such as achievement or mastery).

**J.4: Non-Disruptive Features (N-Codes)**

**N_LackingMechanics:**

Text that contains player discussion framed as ‘AAMFP lacking mechanics’, either in comparison to ATDD, or as a standalone product.

**N_Sanity:**

Text discussing the absence of the ATDD sanity system in AAMFP and/or stating why AAMFP is better or worse without it.
N_Puzzles:

Text discussing ‘puzzles’ within the game. This may be broad statements (e.g. “the puzzles in the game were too easy”), or it may be specific statements (e.g. “working out how to fix the machines in the puzzle in the cellar level was really difficult”).

N_EnvironmentInteractivity:

Text discussing the level of interaction the player has with the in-game environment in AAMFP.

N_LanternFuel:

Text discussing the absence of having to fuel the lantern in AAMFP and/or stating why AAMFP is better or worse without it.

N_Story:

Text discussing the game’s story. Used in parallel with other codes to provide context.

N_StoryDelivery:

Text specifically discussing the story delivery methods in the game (e.g. voice acting, journal entries and collected written notes). Used in parallel with other codes to provide context.

N_GameplayStoryMismatch:

Text discussing what players feel is a mismatch between the gameplay and the game story (e.g. statements such as “it doesn’t make sense that Mandus can run without tiring when he is depicted as both old and ill throughout the story”).

N_Atmosphere:

A broad code used to code text that discusses elements of the game’s atmosphere or ambience. Used in parallel with other codes to provide context.

J.5: Other Prevalent Topics Discussed (O-Codes)

O_DifferentInterpretationsOfGameFeatures:

A broad code used to group together text in which players report notably different interpretations of identical game features. This differs from I_HowYouPlayTheGame. Where that code is used to code text in which players report different experiences, this O-Code is used to code text in which players report different interpretations of how a specific game feature works.

O_FranchiseComparisons:

A broad code used to code text in which players draw direct comparisons between AAMFP and ATDD. Used in parallel with other codes to provide context.
**O_GenreComparisons:**

A broad code used to code text in which players draw direct comparisons between AAMFP and other games that the player determines are within the same 'genre'.

**O_MechanicsThatWereRemoved:**

This code is used to code text in which players suggest improvements or changes that could have been made to AAMFP that are identical or very similar to game features that were originally a part of the 'game as designed' but did not get included in the final release of the game.

**O_Value:**

This is a container code used for organisational purposes. No text is coded with this code directly.

**O_GameLength:**

Text discussing the length of the game in relation to the player's perceived monetary value of the game.

**O_GamePrice:**

Text discussing the price of the game in relation to the player's perceived 'artistic' or 'material' value of the game.

**O_BaitAndSwitch:**

A broad code used to group text referring to what players called a 'Bait and Switch' tactic that they felt was used by The Chinese Room and Frictional Games. This text discusses the view that the marketing of the game suggested very similar gameplay to ATDD when (it is argued by players) that the gameplay is very different in the final release of the game.

**O_ConsideringTheDeveloper:**

A broad code used to group text in which players discuss the qualities of the game in relation to their perspective on The Chinese Room and/or Frictional Games.

**O_EvolvingGames:**

A broad code used to group text in which players discuss the wider topic of 'evolving' games, including issues such as 'what a game should be, or do', 'what constitutes a horror game' and 'what the responsibilities of a game developer to their fan base may be'.
**O_Fans:**

A broad code used to group text in which players discuss, or argue about, the validity of their positive or negative opinions of AAMFP due to their status as ‘fans’ of the franchise, studio(s) or otherwise.

**O_RoseTintedMemoryOfATDD:**

Text that refers to ATDD using superlative descriptions (e.g. “ATDD was the perfect horror game”, or “ATDD was flawless”) that can either be objectively falsified by the researcher or are objectively falsified by other forum users.

**O_UnsatisfiedPlayersMoreLikelyToPost:**

Text in which forum users identify and discuss what they feel is a high level of complaints from unsatisfied players that is not balanced by equal number of posts from satisfied players.

**O_IntendedAudience:**

Text that discusses what players feel was the ‘intended audience’ for AAMFP and whether The Chinese Room and/or Frictional Games misjudged what the fans were expecting and/or wanting from the game.
Appendix K: Complete Series of Theme Maps

K.1: Theme Map 1, Positive and Negative Emotional Influence
K.2: Theme Map 2, Working to Find Meaning(s) in the Game

**WORKING TO FIND MEANING(S) IN THE GAME**

- **O-CODES**
  - O_Diff.InterpretationalFeatures
  - O_MechanicsThatWereRemoved
  - O_ConsideringTheDeveloper
  - O_Value
  - O_GamePlay
  - O_GameLength
- **F-CODES**
  - F_Difficulty
  - F_Frustration
  - F_Horror
  - F_Emotion
  - F_Horror
  - F_PerformanceDifficulty
  - F_Engine
  - F_Terra
  - F_Test
  - F_Environment
  - F_HidingPlaces
  - F_ItemInventory
  - F_Lantern
  - F_PlayerDeath
- **I-CODES**
  - I_HowYouPlayTheGame
  - I_LastingImpact
  - I_EmotionalImpact
  - I_HitPlay
  - I_DecreasedChallenge
  - I_DecreasedCognitiveEngagement
  - I_Disappointed
  - I_Frustration
  - I_threat
  - I_NotScary
  - I_Predictable
  - I_Challenge
  - I_Horror

- **T-CODES**
  - T_DisruptiveGameDesign
  - T_Pressure
  - T_Pressure
  - T_RecalDisruption
  - T_EncodingDisruption
  - T_EducationalDisruption
  - T_SupportHighLevelNeeds
  - T_SupportLowLevelNeeds
  - T_IncrementalAccommodation

- **N-CODES**
  - N_LackingMechanics
  - N_Sanity
  - N_Puzzles
  - N_EnvironmentInteractivity
  - N_Story
  - N_StoryDelivery
  - N_GameplayStoryMismatch

- **Questioning the Rules of the Game/Game World**
  - Rules Ambiguous and Changing
  - Cannot Rely on Incremental Accommodation
  - Pressure from Interpreting/Problem Solving
  - Changing Behavioural Rules of Game Entities
  - Unseen Horror not Scary

- **Ambiguous Story Information**
  - Implicit/Ambiguous Story Information
  - Not What the ATDD Audience Wanted from a New Atmosia Game
  - Multi-Playthroughs Enhances Value

- **Replay Value**
  - Sharing of Different Story Interpretations Among Game Experiences

- **I-CODES**
  - I_HowYouPlayTheGame
  - I_LastingImpact
  - I_EmotionalImpact
  - I_HitPlay
  - I_DecreasedChallenge
  - I_DecreasedCognitiveEngagement
  - I_Disappointed
  - I_Frustration
  - I_threat
  - I_NotScary
  - I_Predictable
  - I_Challenge
  - I_Horror

- **T-CODES**
  - T_DisruptiveGameDesign
  - T_Pressure
  - T_Pressure
  - T_RecalDisruption
  - T_EncodingDisruption
  - T_EducationalDisruption
  - T_SupportHighLevelNeeds
  - T_SupportLowLevelNeeds
  - T_IncrementalAccommodation

- **N-CODES**
  - N_LackingMechanics
  - N_Sanity
  - N_Puzzles
  - N_EnvironmentInteractivity
  - N_Story
  - N_StoryDelivery
  - N_GameplayStoryMismatch

- **O-CODES**
  - O_Diff.InterpretationalFeatures
  - O_MechanicsThatWereRemoved
  - O_ConsideringTheDeveloper
  - O_Value
  - O_GamePlay
  - O_GameLength

- **F-CODES**
  - F_Difficulty
  - F_Frustration
  - F_Horror
  - F_Emotion
  - F_Horror
  - F_PerformanceDifficulty
  - F_Engine
  - F_Terra
  - F_Test
  - F_Environment
  - F_HidingPlaces
  - F_ItemInventory
  - F_Lantern
  - F_PlayerDeath
K3: Theme Map 3, Individual Player Differences
K.4: Theme Map 4, Fan Anger, Perceived Brand 'Ownership' and What Games are 'Allowed' to do

The 'Evolution' and Advancement of the Game Medium

Exploration is a Core Part of Gameplay

Experimentation in Games is Necessary and Makes for More Variation in Experiences

Experimenting with an Established Game is a Betrayal of Fan Trust

Games Must Have Gameplay

AARMF Does Not Have 'Gameplay'

WHAT GAMES ARE 'ALLOWED' TO DO

A Sequel Cannot Remove Features

Removing of ATDD Mechanics

Describing ATDD as Flawless/Perfect

Forum Users Identifying that a Different Developer will Create a Different Type of Play Experience

Expecting Presence of Previous Features

Inventory Management is a Core Part of Gameplay

Game Sequels Have Rules that Cannot be Broken

PERCEIVED BRAND 'OWNERSHIP'

Oregon Trail Should Not Be '$5 on Steam'

Creation of a Game in Another's Style when the Developer Does Not Have the 'Credibility'

Simulation in Games is a Core Part of Gameplay