

# Towards an Ontological Language for Game Analysis

**José P. Zagal, Michael Mateas, Clara Fernández-Vara, Brian Hochhalter,  
Nolan Lichti**

College of Computing and School of Literature, Communication and Culture  
Georgia Institute of Technology  
Atlanta, GA 30332, USA

{jp,michaelm}@cc.gatech.edu, clara.fernandez@lcc.gatech.edu, brianhoch@mac.com, nolanlichti@hotmail.com

## **ABSTRACT**

The Game Ontology Project (GOP) is creating a framework for describing, analyzing and studying games, by defining a hierarchy of concepts abstracted from an analysis of many specific games. GOP borrows concepts and methods from prototype theory as well as grounded theory to achieve a framework that is always growing and changing as new games are analyzed or particular research questions are explored. The top level of the ontology (interface, rules, goals, entities, and entity manipulation) is described as well as a particular ontological entry. Finally, by engaging in three short discussions centered on relevant games studies research questions, the ontology's utility is demonstrated.

## **Keywords**

Game analysis, game design, ontology

## **INTRODUCTION**

This paper introduces the Game Ontology Project (GOP) as a framework for describing, analyzing and studying games. We begin by positioning the GOP in the context of other projects in the field of game studies. Next, we present the theoretical and methodological influences that have shaped both our conceptual understanding of the GOP and its development. We continue with an overview describing its structure and overall hierarchy. Finally, we outline a few ways in which the GOP scaffolds and affords the exploration of interesting research questions and describe some future directions our work will take.

## **RELATED WORK**

Game designers have called for a design language [5, 6, 16, 17], noting that designers currently lack a unified vocabulary for describing existing games and thinking through the design of new ones. Many of the proposed approaches focus on offering aid to the designer, either in the form of design patterns [2, 3, 16], which name and describe design elements, or in the closely-related notion of design rules, which offer advice and guidelines for specific design situations [9, 10]. Other analyses draw methods and terminology from various humanistic disciplines- for example,

Proceedings of DiGRA 2005 Conference: Changing Views – Worlds in Play.

© 2005 Authors & Digital Games Research Association DiGRA. Personal and educational classroom use of this paper is allowed, commercial use requires specific permission from the author.

games have been analyzed in terms of their use of space [14], as semiotic systems [18], as a narrative form [4, 21], in terms of the temporal relationships between actions and events [8], or in terms of sets of features in a taxonomic space, using clusters in this space to identify genres. [1]

The Game Ontology Project's approach is to develop a game ontology that identifies the important structural elements of games and the relationships between them, organizing them hierarchically. Our use of the term *ontology* is borrowed from computer science, and refers to the identification and (oftentimes formal) description of entities within a domain. Often, the elements are derived from common game terminology (e.g. level and boss) and then refined by both by abstracting more general concepts and by identifying more precise or specific concepts. An ontology is different than a game taxonomy in that, rather than organizing games by their characteristics or elements, it is the elements themselves that are organized.

Our work is distinct from design rule and design pattern approaches that offer imperative advice to designers [9, 10]. We do not intend to describe rules for creating good games, but rather to identify the abstract commonalities and differences in design elements across a wide range of concrete examples. The ontological approach is also distinct from genre analyses and related attempts to answer the question "what is a game?" Rather than develop definitions to distinguish between games/non-games or among different types of games, we focus on analyzing of design elements that cut across a wide range of games. Our goal is not to classify games according to their characteristics and/or mechanics [20], but to describe the design space of games.

Our approach is well suited to exploring issues and questions regarding games and gameplay. The GOP provides a framework for exploring, dissecting and understanding the relationships between different game elements. A few examples of research questions we have already begun to explore include: "How can we understand interactivity in games?", "How is gameplay regulated over the progress of a game?", and "What roles does space play within games?"

In summary, we present an ontology in which we identify abstract elements capturing a range of concrete designs. Our ontology allows for generalizations across this range of concrete design choices as embodied in specific games. Its primary function is to serve as a framework for exploring research questions related to games and gameplay; it also contributes to a vocabulary for describing, analyzing and critiquing games.

## **THEORETICAL BACKGROUND AND METHOD**

The purpose of the Game Ontology Project is to categorize things we see in games. More specifically, it defines and classifies the things essential to the "gameness" of games. We have consciously chosen to focus on things that cause, effect and relate to gameplay in order to help characterize and classify the design space of games.

The traditional Aristotelian view of classification was based on the notion of similarity or resemblance. Categories were, according to George Lakoff, "assumed to be abstract containers, with things either inside or outside the category. Things were assumed to be in the same category if and only if they had certain properties in common. And the properties they had in common

were taken as defining the category.”[19] Research in cognitive science has shown that the classical view is insufficient and sometimes incorrect.<sup>1</sup> For many concepts there is no such list of properties that supports a binary category membership function. Membership can be highly dependent on context and culture.

Prototype theory provides an alternative to traditional classification. While a comprehensive review of prototype theory is beyond the scope of this paper, we highlight some key concepts borrowed from prototype theory, explaining how they are useful in the generation of the ontology. Unless otherwise noted, the key concepts are taken from [19].

1. Cognitive economy: categories must be both specific enough to reflect all essential information and general enough not to overwhelm with irrelevancies.
2. Centrality: Some members of a category may be better examples of that category than others.
3. Membership gradience: Some categories may have degrees of membership and no clear boundaries.
4. Centrality gradience: Members (or subcategories) of a category may be more or less central
5. Reference point, or “metonymic”, reasoning: Part of a category (that is, a member or subcategory) can stand for the whole category in certain reasoning processes.
6. Basic Level categorization: Categories are organized hierarchically. Additionally, they are organized so that cognitively basic categories are in the middle of a general-to-specific hierarchy [13].

Rosch et. al [22] claim that the process of categorization is principled and depends on the attributes of what is perceived as well as the characteristics of the perceptual apparatus itself. In other words, we can only categorize on the basis of what we perceive and, all things being equal, that which is more easily perceived will be of greater significance to the categorization process [15]. This is relevant since the majority of the analyses of videogames that inform our ontology are black-box analyses. We do not necessarily presume to have any inside knowledge of the game designer’s ideas or intentions, how the game was implemented or even the internal functioning of a game; our ontology is primarily based on that which we can perceive or experience as players.

According to the concepts of cognitive economy and basic level categorization, the mid-level elements of our ontology are presumably the most easily identifiable since they are observed over a broad range of games. If we observe something interesting in very few games, it is either too specialized to belong, or should be in the lowest (most specific) part of the ontology.

---

<sup>1</sup> For example, robins are more representative of the category “bird” than ostriches. This contradicts the classical view in which each member of a category is just as good an example as any other.

Many parts of the ontology have fuzzy boundaries regarding what games exemplify them (or have aspects that exemplify them). In general, we have tried to describe examples that could be considered central on account of how well known the game is or how closely it matches the definition of the ontological element. We also recognize that there are games whose use as an example deviates from the central or ideal with respect to the ontological definition. These examples are important because they help define the center of the category, and illustrate the nuances and interpretations an ontological definition may have. This is why we have developed the idea of “strong” and “weak” examples to account for the membership and centrality gradience we observe when applying the ontology to specific games.

While prototype theory describes the theoretical foundations that support our ontology, it does not explain how the ontology is generated. Our methodological approach borrows from grounded theory [12]; the elements and structure of our ontology are inductively developed from data gathered in the real world., including field notes of sessions of gameplaying activities, observations of other people playing games and documentation associated with games (manuals, reviews, screenshots, etc.) We make use of theoretical sampling and comparisons, in order to verify whether our ontology remains grounded in real games as opposed to being generated abstractly. These additional observations help identify new abstract categories which can be instances of the ones observed or generate specific categories that had not been salient from previous observations. Thus, our method is iterative, adaptive, and organic. The ontology is constantly adapting to reflect our knowledge and observations.

The ontology is not developed from the top (more abstract) or the bottom (concrete and specific). Rather, as prototype theory suggests, our ontology grows in a middle-out fashion- the obvious (most readily observable) categories tend to exist in the middle of the ontology. As we refine and revisit them, we discover both more abstract and more specific concepts.

## **THE ONTOLOGY**

This section describes some general characteristics and structure of the ontology. It describes the highest level of the ontology in detail as well as some of the elements directly beneath them. It also explains what has been purposefully excluded from the ontology and provides an example of a particular entry in the ontology.

Our ontology abstracts away the representational details of games. Issues of setting (e.g. medieval castle, spaceship), genre (e.g. horror, sci-fi), and the leveraging of representations from other media (e.g. player’s knowledge of the Star Wars universe) are bracketed by our analysis. Because our goal is to characterize the game design space, such bracketing is necessary in order to achieve broad coverage without having to abstractly characterize notions of setting and genre. Thus, we avoid the Sisyphean task of building an abstract model of the whole of human culture. A deep reading of any one particular game would require an analysis of its representational conventions, allusions and connotations. Our ontology helps position the more formal or structural elements of the game within the game design space; other methods and techniques would be required to unpack representational issues.

The top level of the ontology consists of five elements: interface, rules, goals, entities, and entity manipulation. The *interface* is where the player and game meet, the mapping between the

embodied reactions of the player and the manipulation of game entities. It refers to both how the player interacts with the game and how the game communicates to the player. The *rules* of a game define and constrain what can or can't be done in a game; they lay down the framework, or model, within which the game shall take place. Rules regulate the development of the game and determine the basic interactions that can take place within it (see [23] for an overview of other definitions of rules in the context of games). *Goals* are the objectives or conditions that define success in the game. *Entities* are the objects within the game that the player manages, modifies or interacts with at some level. This definition is broader than "game tokens" [6] since it also includes objects that are not controlled by the player. Finally, *entity manipulation* encompasses the alteration of the game made either by the player or by in-game entities. Entity manipulation thus refers to the actions or verbs that can be performed by the player and by in-game entities.

Each ontology entry consists of a title or name, a description of the element, a number of strong and weak examples of games that embody the element, a parent element, potentially one or more child elements, and potentially one or more part elements (elements related by the part-of relation). The examples describe how the element is concretely reified in specific games. As explained previously, we include both strong and weak examples; the weak examples describe border cases of games that partially reify the element. The parent/child relationship captures the notion of subtype (subset); child elements are more specific or specialized concepts than the parent element. Finally, the part-of relation captures the notion of compound elements that are constructed out of other elements (parts). Table 1 shows an example of a mid-level entry from the ontology.

**Table 1: Example Ontology Entry - "To Own"**

<b>Name</b>	To Own
<b>Parent</b>	Entity Manipulation
<b>Children</b>	To Capture, To Possess, To Exchange
<b>Description</b>	<p>Entities can own other game entities. Ownership does not carry any inherent meaning, other than the fact that one entity is tied to another. Changes in ownership can not be initiated by the owned entity. Ownership can change the attributes or abilities of either the owned or owning entity. Ownership can be used to measure performance, either positive or negative. Ownership is never permanent; the possibility of losing ownership separates ownership from an inherent attribute or ability of an entity. Ownership of an entity can change in variety of ways, including voluntary and involuntary changes of ownership.</p> <p>It is important to note the difference between owning an entity, and using an entity. For example, in <i>Super Mario Bros</i>, when Mario collides with a mushroom, the mushroom is immediately used and removed from the game world. Mario never owns the mushroom.</p>
<b>Strong Example</b>	In <i>Super Mario World</i> Mario can collect mushrooms (or fire flowers or feathers) to use later. Mario owns these entities and can make use of them later.
<b>Weak Example</b>	In <i>Ico</i> , the player character must protect a girl called Yorda. While the player only directly controls Ico, his actions are very closely tied to leading, guiding and protecting Yorda. One could argue that Ico, in effect, owns Yorda because of the way they are tied to each other.

## Interface

The interface provides the means by which the player experiences the game and takes action within the game. The presentation provides the sensory experience of the game, input devices provide a mechanism for the player to choose between physically discriminable signals, and the input method maps the signal selected via the input device onto a game action (entity manipulation).

Since our ontology abstracts the representational details of games, the presentation hierarchy focuses on presentation as it directly serves gameplay, rather than on a cultural analysis of issues such as setting, tone, or genre. For example, in analyzing a game such as *Grim Fandango*, rather than exploring the game's excellent use of film noir and art deco visuals or the rich flavor that those visuals and the game's voice acting and music provide, we instead focus on the functional aspects of how presentation influences gameplay. Presentation consists of three parts: the cardinality of the game world, the presentation hardware and the presentation software. The cardinality of the game world can be distinct from the cardinality of the gameplay; see "The Role of Space in Games" in the Discussion section below. The presentation hardware describes the physical details of the visual, audio and haptic displays used in the game, while the presentation software describes how the game state is communicated by means of the hardware. Entries in the presentation software hierarchy include point of view, categorizations of the information displayed in heads-up displays, and categorizations of types of game state feedback.

An input device is a piece of hardware used to gather input from the user. This includes such items as joysticks, joypads, game paddles, fishing controllers, light guns, pushbuttons, pedals, electrosensitive mats, mice, and other devices through which players send messages to the game software.

Input devices differ from input methods in that the devices translate human action (typically motion) into electronic messages (physically discriminable signals) which are then accepted and interpreted by the game software. Together, the input device and input method translate the physical actions of the player into a game action (entity manipulation). See "How Can We Understand Interactivity in Games?" in the Discussion section below for a description of this mapping process. Input devices and methods have changed significantly over the history of electronic games, typically following a trajectory of increasing bandwidth at the device level and more sophisticated handling of that input at the software level. While these advances have often appeared in parallel, we have chosen to keep the hardware and software layers of input handling distinct in the ontology since they can vary independently from game to game. That is, one can vary the hardware device from which a game receives input without altering the game's interpretation of that input.

## Rules

In the context of our ontology, the rules and constraints of a game define what can or cannot be done in a game. They lay down the framework, or model, within which the game takes place. Rules regulate the development of the game and determine the basic interactions that can take place within it. [25] We note that rules should not be considered static or fixed; in some games, the rules can change as you play.

We define two types of rules: gameplay and gameworld rules. Gameworld rules define the virtual world where the game takes place, while gameplay rules impose rules and constraints on top of the gameworld. An example of a gameworld rule is “gravity” (unsupported objects tend to fall); an example of a gameplay rule is that the player has three “lives” (the game ends when the player has been killed three times). We will explore some specific gameplay rules in the discussion section.

The distinction between gameplay and gameworld helps us to understand the difference between “abstract” and “simulator” games. In abstract games, most, if not all the rules are gameplay rules. A simulator game, on the other hand, may rely almost entirely on the gameworld to frame the game, making use of few, if any gameplay rules.

Games which make use of the same physics and graphics engines<sup>2</sup> share many of the same gameworld rules. The differences between the games are due primarily to differences in gameplay rules.

Additionally, we needed to account for emergent situations that were the result of many rules together, without having to resort to a list of the particular rules that caused these emergent situations; we called these, rules synergies. Examples of rules synergies include economies of scale as well as transitive and intransitive relationships. Thus, the rules section of the ontology has three main branches: gameplay rules, gameworld rules and rules synergies.

## Goals

The goals section of the ontology accounts for the in-game objectives or conditions that the player must meet in order to succeed at the game. These are goals defined by the game, though they may or may not be explicitly communicated to the player: in fact, in the eyes of the player, they may not even be defined.

When analyzing a game, we discover goals with different levels of granularity. For all games the highest level goal may be win the game or play as well as possible. However, in order to achieve that goal the player may have to find a key or defeat a monster. Goals must be considered at the level in which they affect the decisions the player is making. Some goals may be short-term (cross the room) while others may be long-term. (solve the mystery).

We consciously decided not to enumerate all the specific goals commonly seen in videogames. In the first place, we could never hope to cover all the specific goals; specially those that are narrative in nature (save the princess, save the universe, etc.). More importantly, there exists a one-to-one correspondence between goals and entity manipulation. This is not surprising, since entity manipulation is the means by which players perform actions in the game, as a means of achieving goals. For example, if the goal is to reach the finish line, then the player will traverse the gameworld (traversal is in the entity manipulation hierarchy). Therefore, goals that are equivalent to actions, usually the lowest level of goals, are not included in this part of the ontology.

---

<sup>2</sup> Such as the Quake, Unreal for graphics and Havok for physics.

Some goals are explicitly player-defined. For example, a person playing *Sim City* may decide to build a representation of the city she lives in. Another player may decide that he wants to play *Quake* without using any weapon other than the shotgun. Goals that players self-impose on their game-playing experience are also not covered in the ontology either.

Finally, though they are not goals per se, we account for how games evaluate and provide player feedback with regard to the degree of goal success. Thus, the ontology also has a branch dealing with goal metrics such as score, success level, etc. The goals section of the ontology has three main branches: agent goals, game goals, and goal metrics.

## Entities

Entities are the objects that make up the reality of the game world (e.g. agents, walls, power ups, etc.). The entity hierarchy is currently the least developed section of our ontology. This may appear strange, since identifying the objects that compose game worlds seems like a logical first step in an ontological analysis. However, in keeping with our focus on gameplay, we initially focused on the actions (entity manipulation) that the player can take in the game world. Entities tend to be defined implicitly by entity manipulations. While we certainly intend to make the entity hierarchy explicit in the future, our current implicit definition of entities via entity manipulation has not been an impediment to our work.

## Entity Manipulation

Game world objects (entities) possess a set of attributes (e.g. velocity, damage, owner, etc.) and a set of abilities (e.g. jump, fly, etc.). Entity manipulation consists of altering the attributes or abilities of game world entities.

Abilities are the “verbs” of entities, that is, the actions that entities are able to perform. Static entities have no abilities. They usually serve as obstacles, platforms (in games with gravity), or items/collectables. Dynamic entities possess abilities. Abilities can be gained permanently (gaining the speed boost in *Super Metroid*) or temporarily (eating a super pellet gives *Pac-man* the ability to eat ghosts).

Attributes are the “adjectives” of entities, and are altered by abilities. For example, the ability to move changes an entity’s location attribute. The ability to vary speed changes the velocity attribute. Attributes can also be altered permanently (changing character statistics in a role playing game) or temporarily (receiving a power up that changes the power of a character’s punch for a short period of time). Abilities can also alter the existence of an entity (can instantiate or destroy an entity).

There are cases where the line between ability and attribute is fuzzy. For example, in *Zelda: Wind Waker*, when Link gains the bow, he now has the ability to attack from a distance. One could argue that one of Link’s attributes has been changed so that using the “attack” verb now does it at a distance. Contrarily, one could argue that Link now has the new ability “To Attack from a Distance”. To distinguish between abilities and attributes in cases like this, we use the following heuristic: if the attribute/ability is utilized through an explicit player choice, then it is an ability; otherwise, if it occurs automatically, without an explicit player decision, it is an attribute. So, in our example, we would say that the bow bestows a new ability to Link, as the



player must explicitly choose to use the bow during an attack in order to be able to attack at a distance.

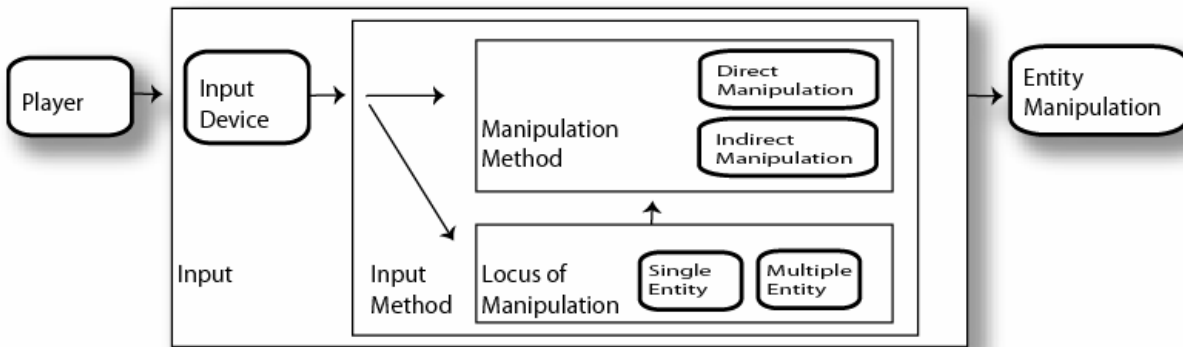
Abstract examples of entity manipulation in our ontology include To Collide, To Create, To Own, and Compound Action (actions composed of several atomic actions).

## DISCUSSION

This section of the paper illustrates how the GOP provides a framework for reasoning, exploring and discussing issues in game studies. We present three short discussions related to research questions we have explored. Space constraints prevent us from fully fleshing out each discussion; their purpose is illustrative of the ontological approach rather than an attempt at providing fully developed answers to these questions.

### How can we understand interactivity in games?

It is commonly noted that interactivity is one of the central characteristics of games [7, 21]. Focusing on the player input side of interactivity, our ontology helps us to understand what it means for a player to perform actions in a game.



**Figure 1:** The Input Path of Interactivity

If a person is playing a game on his computer using a mouse and a keyboard, is this any different from, say, a player using a gamepad? It depends on the game. For some games, the **Input Device**<sup>3</sup> that is used is a greater mediator or factor in interactivity than others. **Input Devices** translate human action (typically motion) into electronic messages which are then accepted and interpreted by the game software (the **Input Method**). So understanding the effect of different input devices requires understanding the relationship between the **Input Device** and the **Input**

<sup>3</sup> For the mini-discussions, words in bold text correspond to actual elements that are part of the Game Ontology. Their definitions can be found online at <http://egl.gatech.edu/gamedesign/>

**Method.** Consider for example a discussion of the reasons why *Halo's*<sup>4</sup> novel **Input Method** allowed it to succeed while being hampered by an **Input Device** (a gamepad) which lacked the affordances that a mouse and keyboard have towards traditional first person shooter games (aiming in FPS games is generally considered much easier to do with a mouse than a gamepad) .

We have unpacked the term “input” past the level of the particular devices and can start to ask questions related to what is actually being manipulated in a game and how. While **Input devices** constitute how user input gets translated into electronic signals, **Input Methods** are the manner in which the game software interprets those electronic messages. Therefore, is a player manipulating several entities in a game or only one (e.g. multiple units in an RTS game vs. an individual character in a platformer)? This is the **Locus of Manipulation**. Additionally, there are games where the manipulation is mapped directly to the input device. For example, if a player controls a spaceship and presses the “left” button on the controller, the space ship moves left. This form of **Direct Manipulation** is very different from the **Indirect Manipulation** present in a game where the player selects the actions he wants his avatar to perform from a menu, such as in the battles of the *Final Fantasy* series. So, the **Manipulation Method** (direct/indirect) as well as the **Locus of Manipulation** play central roles in defining the interactivity of a game. We could also delve deeper and explore the relationships between them in a particular game. Are some entities controlled directly while others are controlled indirectly? What particular forms of **Indirect Manipulation** are there and what are their particular affordances?

This mini-analysis mirrors the overall hierarchy of the **Input** branch of the game ontology (under **Interface**). Asking more questions led us to explore deeper (more specific) parts of the ontology. Each level of “depth” poses its own questions, while the overall structure situates the discussion relative to other design issues.

### **The Role(s) of Space in Games**

Space is a complicated issue to discuss in the context of videogames. Part of the difficulty lies in the fact that there are multiple views of space within a game that, while related, are not necessarily equal. The broadest way of discussing space is at the level of representation. If one considers only what can be seen on the screen, what are the characteristics of that space? Is the representation two-dimensional or three-dimensional? How is that representation achieved? These aspects are part of the **Presentation** and are perceived by the player from a particular **Point of View**.

Most games convey a notion of place to the player, which we call the gameworld. However, we often find games where the **Point of View** describes a space that is different from what the representation suggests. For example, maybe all the characters in the game are rendered in a style that makes them appear as three dimensional objects but they only act in two dimensional ways, as is the case of *Super Smash Bros Melee*. In this case the **Cardinality of the Gameworlds'** space is different from the represented space. Additionally, consider what options of navigation or movement the player has within the gameworld. Is the **Cardinality of**

---

<sup>4</sup> *Halo* is a first person shooter game originally for Microsoft Xbox. *Halo*, and its sequel *Halo 2*, are the most successful games for Xbox in terms of sales and reviews.

**Gameplay**, the space the player can effectively act in, different from that of gameworld and the representation? So there are potentially three different levels at which one could discuss issues of spatiality in games (see Figure 2). They are interrelated and possibly share the same characteristics, but a comparative analysis of various games would have to take them into account.

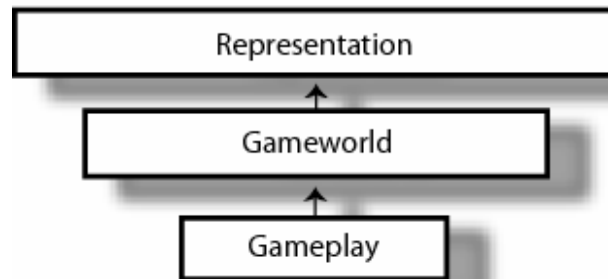


Figure 2: Levels of Spatiality in Games

Consider an example using a familiar game such as *Space Invaders*, but with a twist. Let's imagine that the invaders are not flat two dimensional sprites but rather are beautifully rendered in 3D. At the representational level, we could argue that this version of *Space Invaders* is 3D. On another level, we observe what is happening in the game. The invaders march across the screen from left to right and also, down towards the player. All their actions occur in a two dimensional plane. *Space Invaders* has a **Two Dimensional Gameworld**. The player, however, can only move his spaceship from side to side. The space of movement for the player is only one dimensional. Thus, we say that *Space Invaders* has **One Dimensional Gameplay**.

This mini-discussion shows how different parts of the game ontology relate to each other, in this case, elements under the top level **Rules** and **Presentation** hierarchies. Interested readers are invited to [11] for further discussion of these issues.

### Regulation of Gameplay Over Time

“Level”, “Wave” and “Checkpoint” are all common terms used to describe videogames. What do the terms really mean and what role do they play in a game? Our exploration of such a question led us to discover that all three terms are related to each other in unanticipated ways. Levels, waves and checkpoints are ways of breaking down gameplay into smaller/shorter elements. In other words, they are each a form of **Segmentation of Gameplay**.

**Segmentation of Gameplay** is a more abstract concept than **Level** or **Wave**, thus it lies higher in the ontology. As we explored more games, we not only found new forms of segmenting gameplay, but also realized that they could be organized according to the general way in which they were applied. For example, some forms of segmentation were spatial in nature since they affected the gameworld, while others applied to time. Figure 3 shows how the concept of segmentation has evolved and grown within the ontology. As of this writing, the sub-hierarchy under **Segmentation of Gameplay** has more than twelve elements (not pictured in Figure 3). For

an in-depth discussion of the concept of segmentation of gameplay, readers are invited to refer to [24].



**Figure 3:** Part of the evolution of Segmentation of Gameplay within the ontology

The purpose of this mini-discussion is to illustrate how the process of generating the game ontology is situated in the context of concrete examples. We also emphasize that the ontology is not only growing constantly but also changing and adapting as we dig deeper into specific gameplay-related issues.

## CONCLUSIONS

Our ontology, currently consisting of more than 150 elements, is a tool designed to inform and guide the analysis of games as well as provide a framework for the discussion and exploration of game design space. The future development of the ontology will be structured around specific questions such as those described in the Discussion section above.

## ACKNOWLEDGEMENTS

The authors would like to thank the members of the Experimental Games Lab for their comments and contributions. Also, the following people have contributed to this project over the years as well as its many incarnations. In alphabetical order: Chaim Gingold, Yusun Jung, Heather Logas, Janet Murray, Marleigh Norton and Daniel Rachels.

## REFERENCES

1. Aarseth, E., Smedstad, S. and Sunnanå, L., A multi-dimensional typology of games. in *Level Up: Digital Games Research Conference 2003*, (Utrecht, The Netherlands, 2003).
2. Bjork, S. and Holopainen, J. *Patterns in Game Design*. Charles River Media Inc., Hingham, Massachusetts, 2005.
3. Bjork, S., Lundgren, S. and Holopainen, J., Game Design Patterns. in *Level Up: Digital Games Research Conference 2003*, (Utrecht, The Netherlands, 2003).
4. Carlquist, J. Playing the Story: Computer Games as a Narrative Genre. *Human IT*, 6 (3). 7-53.
5. Church, D. Formal Abstract Design Tools *Game Developer*, 1999.
6. Costikyan, G. I have no words & I must design *Interactive Fantasy*, 1994.
7. Crawford, C. *The Art of Computer Game Design*. Osborne/McGraw-Hill, Berkeley, 1984.
8. Eskelinen, M. Towards Computer Game Studies. *Proceedings of SIGGRAPH 2001, Art Gallery, Art and Culture Papers*. 83-87.
9. Fabricatore, C., Nussbaum, M. and Rosas, R. Playability in Action Videogames: A Qualitative Design Model. *Human Computer Interaction*, 17 (4). 311-368.
10. Falstein, N. The 400 Project, 2004.

11. Fernández-Vara, C., Zagal, J. and Mateas, M., Evolution of Spatial Configurations In Videogames. in *International DiGRA Conference 2005*, (Vancouver, Canada, 2005).
12. Glaser, B. and Strauss, A. *The Discovery of Grounded Theory: Strategies for Qualitative Research*. Aldine, Chicago, 1967.
13. Green, S.P. Prototype Theory and the Classification of Offenses in a Revised Model Penal Code: A General Approach to the Special Part. *Buffalo Criminal Law Review*, 4 (1). 301-339.
14. Jenkins, H. Game Design as Narrative Architecture. in Wardrip-Fruin, N. and Harrigan, P. eds. *First Person: New Media as Story, Performance and Game*, The MIT Press, Cambridge MA, 2003.
15. Johnson, R.K. Prototype Theory, Cognitive Linguistics and Pedagogical Grammar. *Working Papers in Linguistics and Language Training* (8). 12-24.
16. Kreimeier, B. The Case for Game Design Patterns, 2002.
17. Kreimeier, B. Game Design Methods: A 2003 Survey, Gamasutra, 2003.
18. Kücklich, J. Perspectives of Computer Game Philology. *Games Studies: The International Journal for Computer Games Research*, 3 (1).
19. Lakoff, G. *Women, Fire, and Dangerous Things: What Categories Reveal About the Mind*. The University of Chicago Press, Chicago, 1987.
20. Lundgren, S. and Bjork, S., Describing Computer-Augmented Games in Terms of Interaction. in *Technologies for Interactive Digital Storytelling and Entertainment (TIDSE)*, (Darmstadt, Germany, 2003).
21. Murray, J.H. *Hamlet on the Holodeck: The Future of Narrative in Cyberspace*. The Free Press, New York, 1997.
22. Rosch, E., Mervis, B., Gray, W.D., Johnson, D.M. and Bayes-Braem, P. Basic objects in natural categories. *Cognitive Psychology*, 8. 382-439.
23. Salen, K. and Zimmerman, E. *Rules of Play: Game Design Fundamentals*. The MIT Press, Cambridge, Massachusetts, 2004.
24. Zagal, J., Fernández-Vara, C. and Mateas, M. Gameplay Segmentation in Vintage Arcade Games. in Bogost, I. and Bittanti, M. eds. *Ludologica Retro, Volume 1: Vintage Arcade (1971-1984)*, Forthcoming.
25. Zagal, J., Nussbaum, M. and Rosas, R. A Model to Support the Design of Multiplayer Games. *Presence: Teleoperators and Virtual Environments*, 9 (5). 448-462.