Kaboom! is a Many-Splendored Thing: An interpretation and design methodology for message-driven games using graphical logics

Mike Treanor, Michael Mateas, and Noah Wardrip-Fruin
Expressive Intelligence Studio
University of California, Santa Cruz
Santa Cruz, CA, USA
{mtreanor, michaelm, nwf}@soe.ucsc.edu

ABSTRACT

This paper describes an explicit model for how to interpret and create simple 2D games that reasonably communicate messages through a game's representational layer in a manner that is consistent with its processes.

A few prominent experimental games (e.g. *Kabul Kaboom*, *Passage*) have demonstrated that when the rhetorical implications of a game's processes and its representational layer are in harmony, worthwhile and coherent messages can be communicated. This paper reports the findings of an extensive analysis of Activision's *Kaboom!* (1981) [1] that explores its rhetorical design space in the service of developing a general method for the interpretation of simple message-driven games. The paper then shows how the application of this method to even a simple game like *Kaboom!* reveals an unexpected range of coherent potential messages. The paper concludes with a description of a design process and assistant tool that enables those who are not game designers, or even procedurally literate, to create simple games that present editorial and expressive statements. We see this project as a concrete step forward, both analytically and in enabling production, in the field of procedural rhetoric.

Categories and Subject Descriptors

K.8.0 [**Personal Computing**]: General – *Games.* I.2.4 [**Artificial Intelligence**]: Knowledge Representation Formalism and Methods – *Representations (procedural and rule-based).*

General Terms

Design, Theory

Keywords

Game interpretation, procedural rhetoric, game design

1. INTRODUCTION

As many game designers and game studies scholars have argued, game mechanics and rules—in combination with a game's representational layer—have the ability to communicate messages like other mediums of expression (e.g. films, novels, poems, etc.) [2, 7, 8]. However, these claims have yet to become widely accepted—perhaps because few games have successfully combined the representational and rule layers to achieve the expressive quality of message-driven works in other media.

Figure 1 A screenshot from the Atari VCS version of Activition's *Kaboom!* (1981)

A possible reason for the small number of successful message-driven games is that communication through a game requires a coherent combination of simulation and representation. While the visual arts have many established genres and conventions, there are few agreed upon conventions to use when communicating via simulation. As a concrete example of visual arts conventions, consider the editorial cartoon. Given the existence of newsgames, which are defined to be video game equivalent of editorial cartoons [14, 15], the editorial cartoon is a particularly useful example for comparing the tropes of visual meaning with the nascent tropes of rule-based meaning. In the editorial cartoon, authors express opinions about current events in a single comic frame [13]. The visual conventions of the editorial cartoon help cartoonists to condense their editorial messages into a clear and concise image appropriate for the print newspaper.

Drawing on the widespread familiarity of such artifacts of visual culture, the meaning of most visual tropes of editorial cartoons can extracted by an attentive viewer through examining explicit labels and metaphors as well as by making relational inferences. For example, figure 2 shows an editorial cartoon that makes use of caricature, physical and spatial metaphor and literal signifying images and text to communicate a message about the encroaching

FDG 2010, June 19-21, Monterey, CA, USA

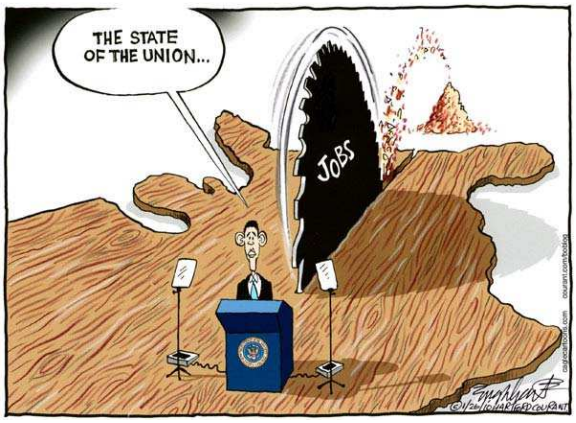

Figure 2 An editorial cartoon that makes use of multiple visual tropes [5].

and dividing problem of unemployment in the United States. The impending disaster is communicated through the use of a destructive entity, the saw, which is merely labeled "jobs." This labeling of objects to make a metaphor explicit is an established convention of the editorial cartoon. Additional relational inferences can be drawn from the fact that the "jobs" saw blade is coming up from behind President Obama, which implies that he is unaware of it. Taking the interpretation even further, the direction of the President's gaze could be interpreted as an assertion as to who the address is being directed towards (the southwestern United States), though this is most likely not an intended message. Interpreting visuals at this level of granularity can feel awkward or unnatural, precisely because we normally perform our learned visual literacy skills *unconsciously*. However, it is exactly this sort of formal common sense reasoning applied to simulation mechanics and their representations that needs to be undertaken by game designers before they can create games that coherently and consistently communicate through gameplay.

1.1 Procedural Rhetoric

While communication through simulation and games is being explored by many in game design and game studies, it is in its infancy compared with studies of visual rhetoric, such as that of editorial cartoons. A key introduction to how simulations express meaning is Frasca's *Simulation 101,* which demonstrates how a simulation can represent thousands of particular assertions, meanings or narratives according to its underlying system's configuration [7]. Simulations allow for the representation of processes.

Bogost describes the term procedural rhetoric as "persuading through processes" [2] or "the way that a videogame embodies ideology in its computational structure" [3]. Creating successful procedural rhetoric requires that designers of games with particular intended meanings should be aware of what exactly it is that their processes are arguing. However, the emergent and unintended implications that arise from interactive systems make this a very difficult goal to reach. With the countless interaction scenarios that arise from even simple games, many games embody unintended procedural rhetoric.

1.2 Operational Logics

In games, a message can be created as a result of the interaction between the player, the processes and the representational layer. Because accounting for *every* gameplay possibility and its meaning in relation to each represented entity is nearly impossible, a more formal and abstract approach can help to make a game's space of potential meanings easier to comprehend. Mateas and Wardrip-Fruin provide such an analytical tool with their concept of *operational logics*. Operational logics are "…an authoring (representational) strategy, supported by abstract processes or lower-level logics, for specifying the behaviors a system must exhibit in order to be understood as representing a specified domain to a specified audience" [11]. Examples of operational logics range from the systems that perform drama management in interactive storytelling to the rules and constraints that govern resource management in real-time strategy games.

This paper is concerned with the relatively small set of logics that apply to movement, collision detection and physics in simple 2D games: the graphical logics. We use these logics as our units of analysis [11]. Graphical logics are well understood by players, as they underlie the mechanics of most early arcade and console games. Because of this, additional meanings have been layered on top of them to form non-spatial meanings, meanings that are not constrained to communicating game state, through the use of coherent *thematic mappings* [12].

1.3 Thematic Mappings

Thematic mappings refer to the specific representations given to game entities (i.e. the *skin* of a game). Thematic mappings can work together with the graphical logics and game rules to create coherent meanings through reasonable relationships between the representations of game entities (e.g. metaphorical relationships). Specific thematic mappings that can be interpreted as rhetorical messages are present in most graphical logic games, but rarely do they amount to meanings beyond combat and antagonistic relationships between game entities. However, in recent years, many people have been attempting to use simple rules and graphical logics with thematic mappings to express editorial opinions and even explore aspects of the human condition [4]. By examining the expressive successes of games of this sort, it is apparent that a *coherence* between graphical logics, rules and thematic mappings is necessary in order to be able to say that a game expresses a reasonably understandable message.

Coherence refers to the degree to which it can be reasonably assumed that a player would interpret a message from the thematic mapping, game rules and operational logics. In the domain of games we are concerned with, the reasonableness of a thematic mapping does not refer to only real world relationships [12], but is extended to rhetorical relationships that form metaphors; as was the case with the previously mentioned editorial cartoon that simply labeled the saw with "jobs" in order to communicate their "dividing force."

As an example of coherence, *Kabul Kaboom* [6] is a slight modification of *Kaboom!* (original described below) where collisions with some of the falling objects award points, and others will result in failure. The player's goal is to collect food falling from the sky while avoiding falling bombs. Through its timely release, title and coherence between game rules and thematic mapping (the player controls a character from Picasso's Guernica), *Kabul Kaboom* is able to express a critique of the United State's foreign policy following the incidents of September 11th. Here, cohesion is achieved because most reasonable interpretations across the thematic mappings, use of graphical

Figure 3 *Kabul Kaboom* communicates a critique of US foreign policy.

logics and game rules yield consistent meanings. In the case of *Kabul Kaboom*, the graphical logics represent physical collisions with the represented entities. The game rules are triggered by these collisions and reward the player for colliding with food, and end the game when the player collides with bombs. The thematic mappings present the player with a character from a famous anti-war painting, as well as a characteristic American meal, the hamburger, to identify the nationality of the people dropping the food and bombs. As a civilian trying to survive a bombing, the rules (being hit by a bomb is failure, finding food is survival) and the operational logic (collision between a bomb and the character means being hit by the bomb, collision between a hamburger and the character means acquiring food) are consistent and harmonious with the thematic mapping. All of these aspects of the game work together to create a consistent editorial message.

However, difficulties can arise while designing message-driven games because the visuals are used to communicate on two interwoven levels: the visual and (potentially) linguistic rhetoric of thematic mappings and the procedural rhetoric of the rules and operational logics. These two levels are not always consistent. For example, as stated before, the bombs in *Kabul Kaboom* are reasonably understood to hurt the visual representation of the player. However, the rules of the game do not account for the explosion that would be expected from the represented ground below the player. Two options that could resolve this slight incoherence are: adding game rules and visuals that account for the explosions or changing the thematic mapping of the bomb to something that a player wouldn't expect to see immediate effects of when it collides with the ground (like biological weapons). This tension between the logic of a game's fiction and the logics of a game's rules is further explored by Juul in his writings about games being *half-real* [9].

When the above two levels of interpretation are in harmony, we say that a game coherently communicates. As demonstrated with *Kabul Kaboom*, by coherently modifying the thematic mappings of existing games, substantive meanings can be expressed. Using Activision's *Kaboom!* as an example, described below is a process for discovering types of meanings that can arise from different thematic mappings on 2D games that use graphical logic and the constraints that make them coherent.

2. ANALYSIS OF *KABOOM!*

Toward the creation of a design methodology that can reliably create games to represent an author's desired meaning, the following analysis of *Kaboom!* illustrates a methodology for bridging the gap between the operational logics and rules, and the implied meanings (visual rhetoric) of the thematic mappings.

2.1 *Kaboom!*

What follows is a very literal interpretation of the original *Kaboom!* (figure 1) followed by a description of how coherent changes to the thematic mappings can completely change its reasonably interpreted meanings.

On the top of *Kaboom!'s* game screen a "Mad Bomber" moves in unpredictable patterns from left to right dropping bombs that fall straight down the screen. At the bottom of the screen is the logo for Activision, the company that released the game. The player controls a set of three "buckets" that can move from left to right. A score counter is incremented for each bomb the player is able to "catch" and is given three chances to miss the bombs before the game ends.

It is clear that the Mad Bomber is intentionally dropping bombs by his animations. Also, while what the buckets represent is left unresolved, it can be inferred by the fact that as the game progresses it becomes increasingly difficult for the player to catch bombs that the Mad Bomber is not intentionally dropping bombs into the buckets. Once three bombs get past the player, the damage done by the Mad Bomber—to a largely-unseen fictional world—is too great for the scenario to continue. Because *Kaboom!* is unwinnable (like many early arcade-style games, the difficulty just keeps progressing until inevitable defeat), this quest to protect the world from damage is ultimately hopeless.

Given the appearance of the Activision logo at the bottom of the screen, one might make the interpretation that *Kaboom!* is about a helpless quest to protect Activision from an insane criminal. This interpretation could be discarded, as it confuses piece of visual representation that is actually a paratext (in this case, a brand identifier), with visual representations that are part of the represented game world. However, for the purposes of our analysis below, the presence of the Activision logo points to the possibility of placing a visual representation at the bottom of the screen that is intended to be *in the game world*; in this way, the unseen fictional world of the original *Kaboom!* can be replaced with an explicitly represented world that the player is trying to protect from the bombs, Considering this additional entity allows for many more potential messages for *Kaboom!'s* mechanics to represent with different thematic mappings (as demonstrated below).

As an example of a protection message, consider modifying the thematic mapping by substituting the sprite of the Mad Bomber with a sprite of a Republican elephant, the buckets with a Democratic donkey and the Activision logo with an image that represent citizens. In concert with *Kaboom!*'s rules and logics, this thematic mapping now makes an editorial cartoon-like statement about how Republicans are attacking the citizens and Democrats are protecting them (figure 4). The investigation that follows seeks to systematically explore the constraints and

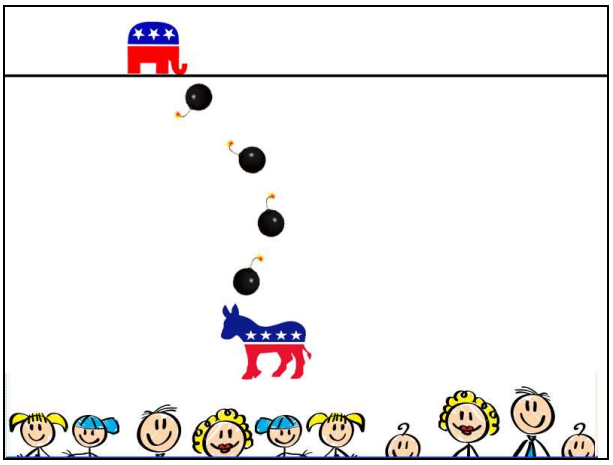

Figure 4 A thematic mapping that expresses a simple political message.

relationships between visual elements of *Kaboom!* to characterize a space of meanings conveyable by the mechanics of *Kaboom!*.

2.2 Method of Interpretation

Games can be interpreted in many ways and from many different perspectives. It is important to stress that the systematic approach that follows is merely one of many possible approaches. For example, without taking into consideration the image at the bottom of the screen, it is still possible to assert that *Kaboom!* is a game about protection, but this message is mostly implied. We know that bombs are destructive and that the player's goal is to intercept them with diffusing buckets, but what the player is specifically trying to stop them from hitting is left to the imagination. The interpretive design choice of including an explicit representation at the bottom of the screen enables designers to create scenarios where the player is protecting a specific entity. Thus, the structure of the model for interpretation shapes the space of potential meanings (figure 5). By choosing and sticking with a systematic approach, many representational structures, and thus expressive possibilities, were discovered that exist coherently with the core rule system of *Kaboom!*. While this method does not describe *all* possible meanings of *Kaboom!*, it does greatly expand the space of its potential meanings. However, it needs to be understood that all of the meanings depend upon the assumptions of the interpretive model. For example, the following analysis could not exactly apply to *Kabul Kaboom* as the additional type of falling object and its associated rules would necessitate a different interpretive model.

Because we are searching for new meanings, it is important to not build into our model assumptions that will limit the results. For that reason, the discussion will refer to the game's entities and rules in as abstract a form as possible. All references to the game's entities will be referred to by arbitrary labels as follows (figure 5): **A** = Mad Bomber, **B** = the buckets, **C** = the bombs and **D** = the area below the buckets (including the Activision logo).

With this, the rules of *Kaboom!* can be described as follows:

- **C** is generated at **A**'s location and falls at a constant rate
- When **C** collides with **D**, the number of remaining attempts to play the game is decremented

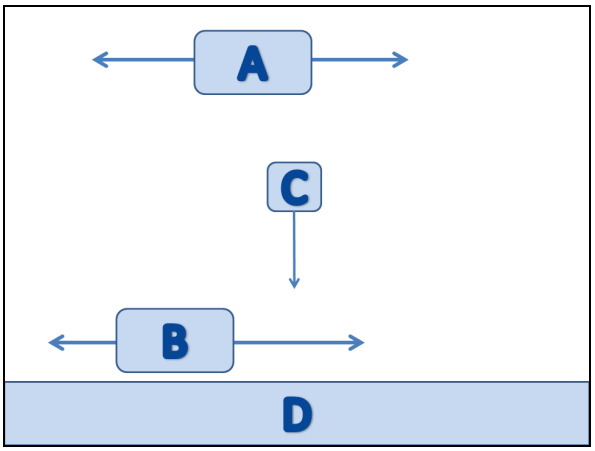

Figure 5 The rules and generalized entities of *Kaboom!*

- When **C** collides with **B**, the score is incremented
- **A** moves left and right
- **B** is controlled by the player
- **B** can move left and right

The interpretations that can arise from this simple set of rules are determined by the relationships between the represented entities (thematic mappings). The interpretive method described below takes the form of asking three types of questions about the represented entities and their relationships: evaluative, volitional and those pertaining to the overall rhetoric. A set of answers to these questions specify a coherent meaning for *Kaboom!*.

2.2.1 Evaluative Statements

Each rhetorically relevant collision in *Kaboom!* can be labeled with an evaluative statement. These evaluative statements describe if a collision is considered "good," "bad" or "not applicable" to an entity. The choice of "not applicable" pertains to representations where imagining a perspective does not make sense to consider (as is the case for inanimate objects such as the bombs in *Kaboom!*). Because the evaluation of a collision between two entities may be different from the perspective of each, a label needs to be considered for each entity involved. With the four entities we are considering in *Kaboom!*, the possible collisions are between **C** and **B**, **C** and **D**, **A** and **C**.

Exactly what the Good, Bad and N/A labels mean is important to clarify when addressing the evaluative statements. A label of Good is to be interpreted as meaning that the event is beneficial to the entity's physical well being, or is in its immediate interests. This implies that the entity has some manner of self awareness, which, as with the bombs, is not always the true. In this case, the option of not applicable (N/A) is appropriate. Of course limiting evaluations to these possibilities is going to limit the meanings that can be created.

2.2.2 Actions of Volition

For every action that an entity takes, we consider the question of whether this action was performed intentionally. Possible answers for these questions are in the form "yes," "no," and "not applicable." "Not applicable" is used to indicate that it does not make sense to consider intention (as would be the case with atmospheric phenomena). Possible actions of volition that can be

considered with the four entities of *Kaboom!* are **A** *moving* left to right, **A** *dropping* **C**, **B** *catching* **C**, and **B** *moving* left and right.

2.2.3 Overall Rhetoric

Differences in overall rhetoric affect interpretation at a higher level than the evaluative and volitional labels. In this context, the rhetoric of a game determines what the game is saying about the situation represented. For example, a high level of difficulty implies that the represented action that the player is pursuing is very difficult to achieve. Most arcade-style games, including *Kaboom!*, are unwinnable. When interpreting a game as representing a meaning, this feature characterizes what has been described as the "rhetoric of failure" [2, 10, 15]. For example, by guaranteeing that the player will lose, the meaning of the *Kabul Kaboom* is understood as the player enacts the rule system. This communicates how a foreign policy that involves dropping food and bombs on the same country will ultimately end in failure and hurt the people that the food is meant for.

To allow for this sort of rhetoric, this method takes a game's difficulty and end conditions into consideration.

2.2.4 Constraining the Space of Interpretations

From the above, some of the evaluative and volitional interpretive considerations can be omitted as they either don't make sense, are not possible to reasonably represent, or are redundant.

From the evaluative statements, the collision between **A** and **C** will not be considered. This is primary because **A** and **C** only collide briefly and the meanings that its consideration enables are mostly captured by the **C**'s evaluative statement for colliding with **D** and whether or not **A** intentionally drops **C**.

We exclude consideration of the volitional actions of **B** moving and trying to catch (or avoid) **C,** as the reality of a player physically playing the game and pursuing its goal implies this. Notice that avoiding **C** is implied by the evaluative judgment of collisions between **B** and **C** by **B**, and thus doesn't require any changes to the set of volitional verbs. Finally, because there is no reasonable way to represent a reason **A** would be moving unintentionally, it is not considered as well.

2.2.5 The Interpretive Considerations

When the interpretive considerations below are specified and labeled, a meaning specification is defined. The variables from which this model of *Kaboom!*'s coherent representational and procedural meaning is formed are:

Evaluative Statements:

- From the perspective of **B**, evaluate a collision with **C**. Responses: **[Good, Bad or N/A]**
- From the perspective of **C**, evaluate a collision with **B**. Responses: **[Good, Bad or N/A]**
- From the perspective of **C**, evaluate a collision with **D**. Responses: **[Good, Bad or N/A]**
- From the perspective of **D**, evaluate a collision with **C**. Responses: **[Good, Bad or N/A]**

Actions of Volition

- Does **A** intentionally drop **C**?
  Responses: **[Yes, No, or N/A]**

Overall Rhetoric

- How difficult is the game?
  Responses: **[Easy, Hard, or Not Possible to Win]**
- Does the game appear to have an ending?
  Responses: **[Yes or No]**

2.3 *Kaboom!'s* Meanings

With the above interpretive method we can describe a space of possible meanings for *Kaboom!*. All combinations of answers to the interpretive considerations represent unique possible meanings. Below we present an exploration of this space that extracts several *classes* of rhetorical meanings that can be created solely through thematic mappings. Classes of meanings are unique descriptions of the primary actions of the player. Each class of meaning can generally describe many specific meanings that all share the class' defining characteristics. For example, a variant of *Kaboom!* where the thematic mapping for **B** was not a Mad Bomber but rather a natural disaster, like a tornado, still belongs to the *protect* class of meanings despite the fact that **A** is no longer intentionally dropping **C**. Classes can also be combined to create specific meanings.

For the sake of brevity, we here introduce a simple notation for referring to the interpretive considerations: *CollisionEval(x, y)*, refers to the evaluation from the perspective of *x* about the collision between *x* and *y* and *Volition(x, action)*, refers to the question of intention for *x* to perform *action*. Because the overall rhetoric makes statements *about* the situation represented, it will not be considered in the formulation of which class of meanings a particular game belongs to. Any combinations of overall rhetoric labels can be applied to a class of meanings to make a statement about the represented goal of the player. Finally, where the label of an interpretive consideration is not absolutely necessary for a game's inclusion in a class of meanings, a wildcard (*) will be used.

Note that the generalized classes presented do not exhaust the space of all possible meanings for *Kaboom!* and its thematic mappings. According to the interpretive considerations chosen, we found that there are specifications that either only make sense in very specific situations, or are hard to even imagine making sense at all. Those included below were chosen because they can be readily applied to the domain of newsgames.

2.3.1 To Protect

As previously stated, the unlikely interpretation of *Kaboom!* as a hopeless quest to protect Activision from an insane criminal, opens up the design space for considering explicitly represented worlds that the player protects from some danger. With the framework described above, we can now make this interpretive claim formal by specifying variable values that leads to this meaning:

*CollisionEval(***Buckets**, **Bombs***)* → N/A
*CollisionEval(***Bombs**, **Buckets***)* → N/A
*CollisionEval(***Bombs**, **Activision***)* → N/A
*CollisionEval(***Activision**, **Bombs***)* → Bad
*Volition(***Mad Bomber**, *Drops(***Bombs***))* → Yes

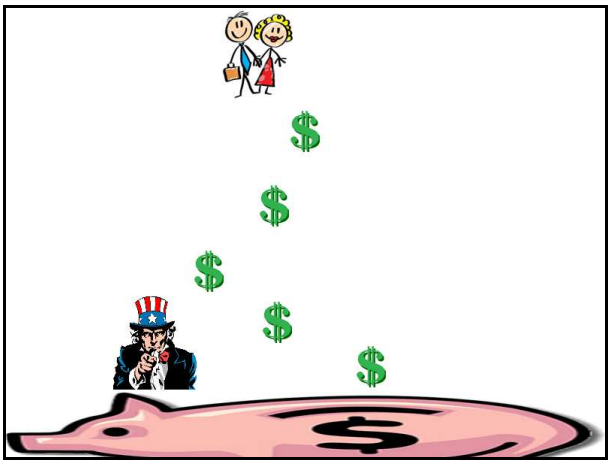

Figure 6 Help Uncle Sam *steal* money from citizens.

Translated, this sequence of specified interpretive considerations describe a situation where the Mad Bomber is intentionally dropping bombs (and implicitly dropping them on the Activision logo by the rules of the game [see section 2.2.4]); to Activision collisions with the bombs are actively bad or undesirable and both the buckets and the bombs are either inanimate or indifferent to the collisions. Despite this last fact, the player controls the buckets and the rules of the game dictate that player try their best to intercept the bombs. This combination of the player actively intercepting something that is *bad* for Activision can be reasonably interpreted as an act of protection.

While the above describes how the meaning of protection is created through the relationships between the representational entities in the original *Kaboom!*, not all responses were important in this formulation. For example, however the bombs evaluate collisions with the buckets or Activision does not affect the game belonging to the protect class of meanings. Additionally, whether the Mad Bomber drops the *bad* things intentionally also does not change the fact that the player, represented as buckets, can be understood to be protecting Activision. While these interpretive labels are not as important as others in the interpretation of this instance of representations belonging to the protect class, they serve as refinements and demonstrate how more nuanced messages are possible under the protect class of meanings. The power of this method lies in providing a formal specification for what relationships must exist between the represented entities (the thematic mapping) in order for it to be reasonably understood as a game about protection.

The general specification for creating a version of *Kaboom!* that can reasonably and coherently communicate protection to a player is:

CollisionEval(**B**, **C**) → *
CollisionEval(**C**, **B**) → *
CollisionEval(**C**, **D**) → *
CollisionEval(**D**, **C**) → Bad
Volition(**A**, *Drops*(**C**)) → *

Below are several other example specifications for thematic mappings that coherently describe other classes of meanings.

2.3.2 To Steal

CollisionEval(**B**, **C**) → *

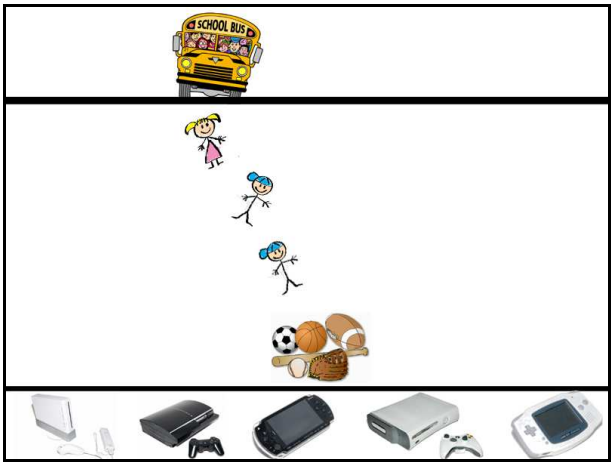

Figure 7 Help after school programs *divert* kids from video games.

CollisionEval(**C**, **B**) → *
CollisionEval(**C**, **D**) → *
CollisionEval(**D**, **C**) → Good
Volition(**A**, *Drops*(**C**)) → Yes

This describes the specification for when **A** intentionally drops something that **D** wants (e.g. common people drop money to a bank) (figure 6). Because the rules of the game specify that it is increasingly difficult to collect the falling objects, it can be inferred that when **A** intentionally drops something, it is intended for **D**. Because the player's goal of the game conflicts with **A**'s intention, this class of game is considered to be about stealing (e.g. Uncle Sam stealing money from the common people) (figure 6).

2.3.3 To Divert

CollisionEval(**B**, **C**) → *
CollisionEval(**C**, **B**) → *
CollisionEval(**C**, **D**) → Good
CollisionEval(**D**, **C**) → Good
Volition(**A**, *Drops*(**C**)) → *

Because **D** is a place that is both good for **C** and happy to receive it, and **B** is preventing this collision, this specification can be described as being about "diverting" **C** from **D**. Figure 7 shows a scenario where kids are falling from a school bus and, by the rules of the game, have a natural inclination to fall toward video games. The player is represented as a bundle of sports equipment, signifying productive activities, and has a goal to intercept, divert the kids from the video games. As an example of how further specifications of classes can result in more nuanced meanings, the additional specification of *CollisionEval*(**C**, **B**) → Bad, allows this game to be understood as a game about **B** *kidnapping* **C**.

2.3.4 To be Self-Destructive

CollisionEval(**B**, **C**) → Bad
CollisionEval(**C**, **B**) → *
CollisionEval(**C**, **D**) → *
CollisionEval(**D**, **C**) → *
Volition(**A**, *Drops*(**C**)) → *

This specification refers to any situation where the goal of *Kaboom!* conflicts with the well being of the representation of the

player (**B**). As a simple example, consider where **B** was a young celebrity, **C** was various forms of drugs and **A** and **D** were not represented (i.e. blank). This would then be a game about a celebrity's insatiable thirst for drugs. The fact that the game rules increase the score with each collision with the drugs could be reasonably assumed to be interpreted as ironic.

2.3.5 *To Rescue*

*CollisionEval(***B**, **C***)* → *
*CollisionEval(***C**, **B***)* → Good
*CollisionEval(***C**, **D***)* → Bad
*CollisionEval(***D**, **C***)* → *
*Volition(***A**, *Drops(***C***))* → *

If the representations are reasonably understood to indicate that **C** would rather collide with **B** than **D**, the game describes a situation that can be understood as **B** rescuing **C** (see section 2.4 for a detailed example).

These are only a few interesting examples from the broadest classes of meanings that *Kaboom!* can coherently represent. These classes of meanings were discovered by enumerating over the possible specifications and interpreting them. Each class contains a multitude of more specific meanings that can be created trough the specification of the wildcards. Additionally, these classes are easily combined and can overlap (as is the case with the steal and divert classes) as long as any of the fully specified considerations do not contradict one another. For example, *self-destructive* and *protect* can be combined to represent the *self-sacrificing hero*. Figure 8 shows a scenario where the player controls Jesus on a crucifix, protecting common people from the attacks of the devil, which are represented by the word *sin*, This crudely represents the story of Jesus' sacrifice for humanity.

2.4 From a Message to *Kaboom!*

With many coherent game meaning classes discovered, it is very simple to create message-driven games by simply imagining thematic mappings that conform to the specifications. Given that a specification is found that communicates the intended message of the designer, the implementation process for creating a game to express that meaning becomes as simple as finding or creating images to insert in an already existing computational version of *Kaboom!*.

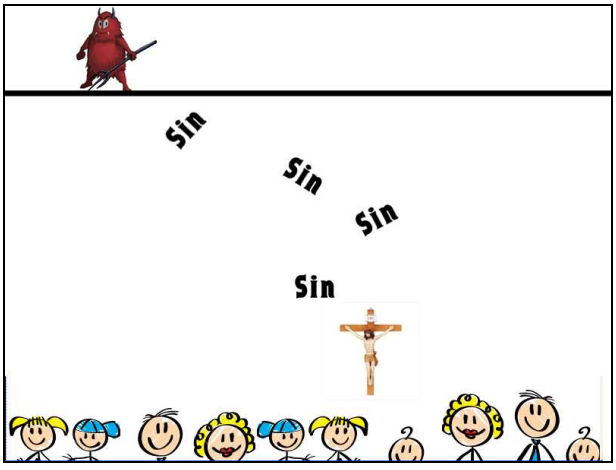

Figure 8 Jesus saves the people from sins with his sacrifice.

2.4.1 *Example*

Newsgames, like *Kabul Kaboom*, are games created in response to current events. These games are almost unavoidably filled with political and editorial opinions [15]. This example will describe an imagined Republican's design process as he creates a game, using the *Kaboom!* meaning classes, to express an editorial opinion about Democratic legislation. The message that he want to express is "President Obama is wasting money" (a simple editorial message, yet representative of the types of messages that are commonly expressed in editorial cartoons).

From a list of potential meaning specifications (see section 2.3), he chooses to use the rescue class. Given that this class describes games where the player's goal is to rescue, he decides to make the primary message about Republicans rescuing wasted money. To represent this, he chooses a Republican elephant to represent **B**, bags of money to represent **C** and President Obama to represent **A**. With **D** still left unspecified, the specification is used to provide assistance for choosing a representation that conforms to the thematic mapping specification. The rescue specification tells us:

*CollisionEval(***C**, **B***)* → Good
*CollisionEval(***C**, **D***)* → Bad

From this, it can be seen that a collision with **D** will need to be a bad thing for **C**. It is then arrived at that **D** can be fire, as it is certainly not a good thing, from the perspective of money, to burn in a fire. Finally, the overall rhetoric needs to be taken into consideration. Because this Republican isn't particularly optimistic, he chooses that the game is unable to be won and there be no appearance of an ending (figure 9).

To check that the thematic mapping communicates the desired message, the designer can review the meaning of the game from the specification and examine it for incoherence:

*CollisionEval(**Republican Elephant**, **Money**)* → Good
*CollisionEval(**Money**, **Republican Elephant**)* → Good
*CollisionEval(**Money**, **Fire**)* → Bad
*CollisionEval(**Fire**, **Money**)* → N/A
*Volition(**President Obama**, Drops(**Money**))* → Yes
Difficulty → Not Possible to Win
Ending → No

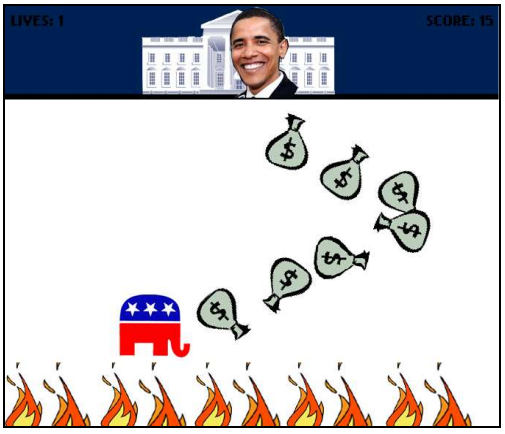

Figure 9 Help the Republicans *rescue* money from firey destruction.

Without the templates that resulted from the interpretive model it is unlikely that someone who is not experienced with game design would imagine that *Kaboom!* could be a game about rescuing and even less likely that the message would be coherent.

3. CONCLUSION

To summarize, this interpretation and design methodology for any video game that uses graphical logic (e.g. *Space Invaders*, *Super Mario Brothers*, etc.) has the following steps:

- **Analysis:** Remove the thematic mapping and all notions of evaluation that are not part of the game rules and identify all of the potentially meaningful events and entities of the graphical logic.
- **Interpretative Model:** Create the list of interpretive considerations (evaluative, volitional and rhetorical).
- **Exploration:** Imagine possible meanings that the game could have with different thematic mappings and specify the constraints that created those meanings in terms of the interpretive considerations. Next, enumerate over the combinations of answers to discover more meaning classes.
- **Design:** From the list, a designer can begin with a message, and describe the message in terms of one or more consistent meaning classes.
- **Implementation:** With a fully specified game, the designer solely needs to provide signifying images to fill the roles specified in the design phase.

With this systematic analytical and interpretive model, it is possible to discover a large and unexplored expressive space of meanings for any given graphical logic game. Because these meanings arise solely from changes in the thematic mappings (skinning) and not the rules and graphical logic, it is relatively simple to create many message-driven games that can be reasonably understood. While the space of meanings discovered is limited by the assumptions that were used to create it, these assumptions are well specified and can be easily modified.

A design assistant tool is in development that will help designers go from a message to a game that represents that meaning. This tool takes the form of a simple *wizard* that presents the user with a series of prompts that systematically describe an intended message in terms of what is possible given the previous responses. Because of the nature of the combinable and somewhat hierarchical meaning specifications, the process for finding a full specification to fit an author's intended message is able to be computationally reasoned over to provide critical feedback about the message that a partially formed specification is creating.

Additionally, because the space of meanings for even a simple game like *Kaboom!* can easily become very large, the process of discovering meaning specifications can be time consuming and requires someone skilled in interpreting procedural rhetoric. Because of this, an area of future work is to design an automated process to assist with the exploration phase by discovering meaning classes by automatically recognizing patterns of game rules and assigning descriptive titles to them.

With this relatively straightforward process, especially if embodied in an assistant tool, individuals of all technical abilities could be enabled to create games that express opinions, express feelings or even simply make jokes to friends. Confusions about the implications of processes and their representations have too long stunted the ability for designers to utilize the medium of video games for message-driven acts of expression. This methodology of analyzing and interpreting graphical logic games is a concrete step forward toward the understanding and creation of games that successfully utilize procedural rhetoric.

4. REFERENCES

[1] Activision. Kaboom! 1981.

[2] Bogost, I. Persuasive Games: The Expressive Power of Videogames. The MIT Press, 2007.

[3] Bogost, I. "Playing Politics: Videogames for Politics, Activism, and Advocacy". In First Monday 11, no. 9., Special Issue #7: Command Lines: The Emergence of Governance in Global Cyberspace

[4] Bogost, I. The Proceduralist Style. Gamasutra. 1/21/2009.

[5] Englehart, B. "The State of the Union..." engletoons@sbcglobal.net <http://cagle.com/news/StateUnion2010/images/englehart.jpg> 1/29/2010

[6] Frasca, G. "Kabul Kaboom" 2001.

[7] Frasca, G. Simulation 101: Simulation versus Representation. 2001. http://www.ludology.org/articles/sim1/simulation101.html.

[8] Humble, R. Game Rules as Art. The Escapist Magazine. 4/18/2006.

[9] Juul, J. Half-Real: Video Games between Real Rules and Fictional Worlds. Cambridge, Massachusetts: MIT Press 2005.

[10] Lee, S. "I Lose, Therefore I Think" in Game Studies, no. 3.

[11] Mateas, M and Wardrip-Fruin, N. Defining Operational Logics. Conference of the Digital Games Research Association - DIGRA. 2009.

[12] Nelson, M.J. and Mateas, M. Towards automated game design. In AI*IA 2007: Artificial Intelligence and Human-Oriented Computing, pp. 626–637. Springer Lecture Notes in Computer Science 4733. 2007.

[13] Press, C. The Editorial Cartoon. Rutherford: Fairleigh Dickinson University Presses. London: Associated University Press. 1981.

[14] Receiver Magazine. "This just in. Playing the news." http://www.vodafone.com/flash/receiver/17/articles/pdf/17_07.pdf

[15] Treanor, M and Mateas, M. Newsgames: Procedural Rhetoric meets Editorial Cartoons. Conference of the Digital Games Research Association - DIGRA. 2009.

[16] Wardrip-Fruin, N. Expressive Processing: Digital fictions, computer games, and software studies. MIT Press, Cambridge MA.,