A Framework for Analysis of 2D Platformer Levels

Gillian Smith
gsmith@soe.ucsc.edu
UC Santa Cruz

Mee Cha
mc63874@ucsc.edu
UC Santa Cruz

Jim Whitehead
ejw@soe.ucsc.edu
UC Santa Cruz

Abstract

Levels are the space where a player explores the rules and mechanics of a game; as such, good level design is critical to the game design process. While there are many broad design principles, level design is inherently genre-specific due to the wide variety of rules and types of challenge found between genres. Determining genre-specific design principles requires an in-depth analysis of games within the genre. We present such an analysis for the 2D platformer genre, examining level components and structure with a view to better understanding their level design. We then use this analysis to present a model for platformer levels, specifically focusing on areas of challenge. Our framework provides a common vocabulary for these items and provides level designers with a method for thinking about elements of platformers and how to compose them to create interesting and challenging levels.

CR Categories: K.8.0 [Personal Computing]: General—Games; H.1.2 [Models and Principles]: User/Machine Systems—Human information processing

Keywords: level design, game design, game analysis, platform games

1 Introduction

Good level design is vital for creating enjoyable games; as "container[s] for gameplay" [Byrne 2005], levels provide the player with an interactive space and the ability to explore within the context of the gameworld rules. As a result, level design is a complex task, requiring an understanding of all the components of the game and how to fit them together. Furthermore, the skills required differ by genre: a level for a 2D puzzle platformer has very different requirements than a level for a role-playing game. Despite the different skillsets and the complexity of the task, few texts currently address issues in genre-specific level design. Those that do provide only a brief overview for each genre.

We assert that principles underlying level design vary by genre, and a deep understanding of level design requires genre-specific analysis. For example, an FPS level designer must consider the player's need to take cover, regain health, and not get lost navigating an open world. Racing game levels tend not to have any of these requirements, and are instead crafted to provide challenge to the player within the limits of the game's laws of physics. Moving beyond broad design principles requires an analysis of level design within each specific genre.

In this paper we explore level design for 2D platform games by breaking their levels down into their common components and studying how they fit together. We choose this genre because its rules are simple to understand, yet provide for substantial complexity. There are a number of different styles of platformer, which are exemplified by three popular games in the genre: *Super Mario World*, *Sonic the Hedgehog*, and *Donkey Kong Country 2: Diddy's Kong Quest*.

Super Mario World is characterized by a reward structure closely tied to the probability of failure and a single path through the level with relatively few hidden areas accessible from vines or pipes. Players tend to play for the challenge of completing each level through mastery of its dexterity challenges. In contrast, *Sonic the Hedgehog* has rings liberally spread throughout the level. It is easy to gain rewards in this game; a skilled player will collect a large number of rings and extra lives to carry throughout the game. The primary challenge in this game comes from completing levels as quickly as possible. *Sonic* levels frequently have multiple paths to completion; often the player must choose the appropriate path to complete the level. Finally, *Donkey Kong Country 2* is a game of exploration; playing the levels as they are initially presented is not the intended purpose. Instead, the player must find a number of hidden areas to collect rewards and secret coins. Despite the differences between these styles of platformer, there are a number of similarities in their component elements and how they fit together.

The primary contribution of this paper is a common vocabulary for elements of platformer levels and a framework for modeling levels with them. This structural framework pays particular attention to rhythm and pacing, which prove important for providing challenge to the player. Together they provide an improved theoretical understanding of level design for 2D platformers.

2 Related Work

Books on game design typically do not directly address level design. For example, Salen and Zimmerman's *Rules of Play* is primarily about game design, but does discuss important level design concepts such as repetition and interactivity when discussing crafting the play of experience [Salen and Zimmerman 2004]. Books that do address level design talk primarily about its general principles. *Level Design for Games* [Co 2006] discusses the importance of spatial layout and atmosphere. *Fundamentals of Game Design* [Adams and Rollings 2007] and *Beginning Game Level Design* [Feil and Scattergood 2005] also largely focus on general issues, giving a couple of paragraphs on level design for specific genres. Short descriptions of genre-specific level design are not sufficient to provide a detailed understanding of the structure and interrelationships of elements in a level.

While there has not been a great deal of analysis for genre-specific level design, there are a few of notable exceptions. Nelson's article on level elements in *Breakout* is a good example of the kind of detail we need to provide for this level of understanding [Nelson 2007]. His article takes a reductionist view of level design for *Breakout*-style games, first decomposing levels into their genre-specific constituent components, and then giving rules for how to compose level elements into interesting designs. This approach of reducing levels to their base components and giving rules for their recomposition is the essence of the modeling technique we use in this paper. Boutros writes about common design goals in best-selling platform games [Boutros 2006], focusing especially on visuals, controls, and structuring challenges. However, his focus is on helping game developers understand what made these platform games so popular, whereas we wish to provide a common vocabulary and structure for levels to help students and developers discuss and analyze levels.

There are a few other works that analyze games by dividing them into components based on their roll in the game. However, this work tends to be oriented towards game design rather than level design. For example, Bjork and Holopainen discuss an activity-based framework for describing games [Bjork and Holopainen 2004]. Although the framework is intended for entire games rather than levels, their framework is similar to the process we use for determining level components and the ways they combine to form a level. They break down games into a series of patterns in much the same way that we break down levels into a series of patterns. We especially look at structural and temporal aspects of level design.

Also, the Game Ontology Project [Zagal et al. 2005] analyzes a wide variety of games from a number of different genres in an effort to discover similar components, challenges, and goals between games. The project splits games into segments called “Levels”, “Waves”, and “Checkpoints”, labeling levels specifically as a spatial segmentation, whereas “Waves” or “Puzzles” refer to challenge segmentation. In this paper we discuss levels as incorporating challenge as well.

The importance and structure of challenge is central to our model for how level components fit together, and especially in our notion of rhythm groups. In his book describing the role of a level designer, Byrne claims that challenge is the most important aspect of level design [Byrne 2005] because it is the key to the player enjoying the level. Nicollet also discusses the role of challenge in dexterity-based games [Nicollet 2004]. He creates a series of rules for designing challenge, including the importance of rhythm, timing, and handling failure. These rules have helped us understand how to segment levels into rhythm groups.

3 Model Overview

Our model for platformer levels is in two parts: components, and a structural representation for how these components fit together. We focus primarily on the underlying structure of levels, rather than their visual representation. We categorize components by their purpose in the level; our five categories are platforms, obstacles, movement aids, collectible items, and triggers.

Platforms are any object that the player runs or jumps across to traverse the level, such as flat surfaces, loops, or the tops of item boxes.

Obstacles are any object that is capable of imparting damage to the avatar. Gaps between platforms are also considered obstacles, even though they are not explicitly objects in the level.

Movement aids are any object that helps the player traverse the level, such as springs, moveable trampolines, or ropes.

Collectible items are any object that provides a reward to the player; this could be a coin, power-up, or point reward.

Triggers are any object in the level that somehow changes the state of the level. Examples of triggers include switches that turn blocks into coins, buttons that activate platforms for the player to rush across, or objects which change the avatar’s behavior.

We will discuss each of these components in more detail in section 4.

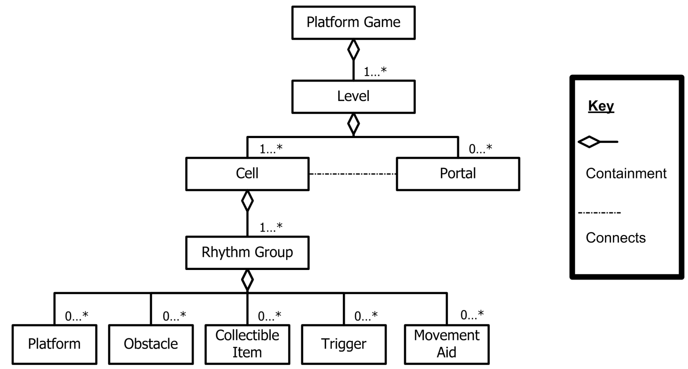

Figure 1: *Conceptual model of our level framework.*

Our structural representation is a hierarchy with the aforementioned components at the bottom and the entire level at the top. In creating this representation we focus on dividing the level into areas of challenge:

- The highest subdivision of a level is called a **cell**, which is a section of linear gameplay. Cells are linked by **portals**.
- In turn, cells are composed of **rhythm groups**, which are non-overlapping sets of the level components. These rhythm groups are often fairly small, encapsulating challenging sections of gameplay. Breaks in rhythm groups are safe places for the player to rest before continuing the level.

Figure 1 shows the hierarchical nature of our model. Cells, portals, and rhythm groups will be explained more fully in section 5.

4 Components of Platformer Levels

We categorize components of platformer levels by their roles in the level. It is common for level components to be members of more than one category; for example, item boxes in *Super Mario World* are considered collectible items, but also as platforms because the avatar can walk along the top of them. Here we describe the different types of components and their typical properties. We also discuss the player avatar as it relates to level design.

4.1 The Avatar

The avatar refers to the character being controlled by the player. Although there are sometimes multiple playable characters, such as Diddy and Dixie Kong in *Donkey Kong Country 2*, the player can usually control only one character at any given time. Avatar choices are sometimes merely cosmetic, but there are often important differences in their behavior. Occasionally the avatar can be swapped at certain times during the level itself, for example at the “stork stations” in *Yoshi’s Island DS*. The ability to change characters mid-level introduces a puzzle aspect to the game, such as in Figure 2, where the player must choose a particular character so Yoshi can float with the wind to the upper ledge.

An avatar’s specific movement ability varies according to the game, but all avatars have control over their horizontal motion and at least limited control over vertical motion, usually in the form of jumping or crouching. More advanced abilities include double jumping and wall jumps. Additionally, action platformers such as *Mega Man* are often provide the avatar with a set of weapons for destroying enemies.

Figure 2: Yoshi's Island DS *has "stork stations" to allow a player to choose a new avatar mid-level.*

4.2 Platforms

A platform is defined as any object that the avatar can walk or run across safely. Often objects that serve some other primary purpose, such as item boxes in *Super Mario World*, double as platforms. Platforms always have a coefficient of friction, size, and slope. Friction and slope affect the avatar's movement across the surface. Platforms can be in constant or occasional motion, and often form paths planned by the designer (Figure 3).

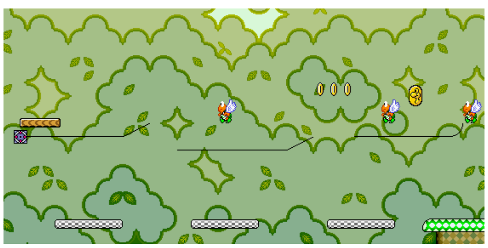

Figure 3: *This platform from* Super Mario World *follows paths marked by the thin black lines. The paths are laid out by the level designer.*

Finally, platforms may be temporary, either on a timer or allowing only a certain number of times the avatar may touch it before disappearing. Temporary platforms may also be deliberately destroyed by the player, such as by using Yoshi's "ground pound" attack. These platforms force the player to make a quick decision and can help create challenging situations. For example, *Sonic the Hedgehog* has platforms are consumed by fire once Sonic has jumped on them. The player must quickly leave the platform before the fire catches up (Figure 4).

There is astonishing variety in types of platforms. Many games have platforms that are flexible, bending or shifting under the "weight" of the avatar (*Yoshi's Island*). Some platforms are invisible; this is especially true of item boxes that double as platforms, where the player must hit the platform to make it appear visible (*Super Mario World*). There are also platforms that are slippery or sticky, forcing the player to react quickly to sudden changes in the avatar's movement.

4.3 Obstacles

Obstacles are a major source of challenge in platform games. Obstacles are any object that is capable of causing damage to the avatar. These can be either scripted moving enemies such as Bullet

Figure 4: *A temporary platform in* Sonic the Hedgehog.

Bills in *Yoshi's Island DS*, or static obstacles such as the spikes in *Sonic the Hedgehog* (Figure 5). Note that platforms that obstruct the player's movement are not obstacles unless they impart damage to the avatar.

Figure 5: *Obstacles can be either moving enemies like Bullet Bills from* Yoshi's Island DS*, or static objects such as the spikes from* Sonic the Hedgehog.

As with platforms, non-static obstacles have an equation modeling their motion. They also have a pattern for spawning; they can either spawn once, a finite number of times, or infinitely. The player must determine whether or not the obstacle can be killed. If it can be destroyed, the obstacle may leave behind a reward, which is sometimes correlated to the difficulty of killing it.

4.4 Movement Aids

Movement aids objects that help the player through the level in a manner other than running or jumping. Examples of movement aids include ladders, ropes, springs, and moveable trampolines, although there can be difficulty in the classification of objects like springs, which might be either platforms or movement aids. We classify springs as movement aids because they are often placed on top of platforms rather than standing alone.

Movement aids can often be manipulated by the player; for example, *Super Mario World* has trampolines that can be moved by Mario. Another example of a movement aid are the ladders that appear in *Mega Man X2*. These two examples are shown in Figure 6.

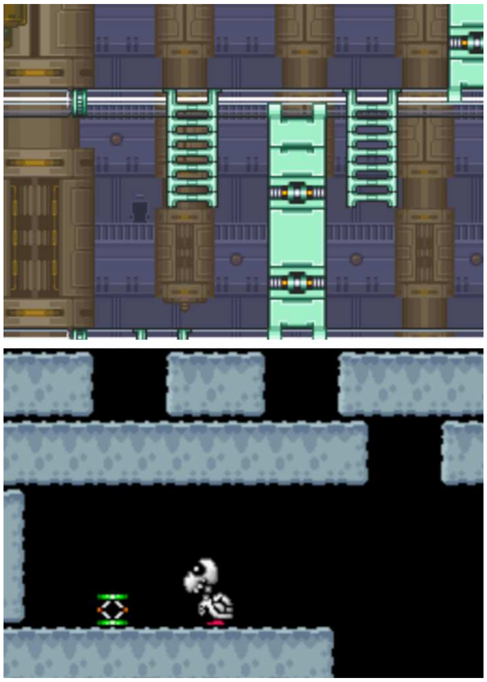

Figure 6: *Examples of movement aids from* Mega Man X2 *and* Super Mario World*.*

Movement aids have an equation modeling their motion, but also allow the possibility for the player modifying that equation. For example, *Yoshi's Island DS* has swings that the player can interact with to be able to swing faster and higher. It is important to note that movement aids do not permanently modify the behavior of the avatar; they only act on the avatar for as long as it is interacting with the object.

4.5 Collectible Items

All platform games have a reward system, and almost always in the form of collectible items. Collectible items are any object in the level that provides a reward, such as coins, rings, power-ups, points, or weapons. They also include non-player characters (NPCs) that the avatar uses to move around the level, such as Yoshi in *Super Mario World* or the rhinoceros in *Donkey Kong Country 2*. These NPCs modify the avatar's behavior in much the same way as a fire flower or feather powerup from *Super Mario World*.

Collectible items often have a reward value associated with them, in the form of points assigned to the player. These points can be redeemed in a variety of ways depending on the rules of the game. Daniel Boutros gives a detailed account of reward systems and design in a variety of platform games [Boutros 2006].

For example, in *Super Mario World*, the main way to collect points comes from collecting coins, which can then be redeemed for extra lives. (Figure 8). Other rewards include power-ups for the avatar or an extra life. Collectible items have a great deal of impact on the player's satisfaction in playing the game; Desurvire et al. use the existence of power-ups and collectible items as a heuristic for evaluating the playability of a game [Desurvire et al. 2004].

Figure 7: *Bananas in the shape of an arrow show the player how to access a hidden area of this* Donkey Kong Country 2 *level.*

Rewards are also important for other reasons, often serving as guides through the level (Figure 7) and compensation for risk [Bleszinski 2000]. For example, Figure 8 shows three kinds of rewards: a large Yoshi coin, four smaller coins, and an unclaimed item box. The Yoshi coin has the largest potential reward and covers an area of risk; the platform below it sinks into the water so Mario cannot stay on it for long. The four coins forming an arc will all be collected if the player performs the optimal jump across to the platform in the lower right. Finally, the item box could contain either a coin or a power-up. Both possibilities serve as a guaranteed reward in case of failure.

Figure 8: *Three types of reward available in* Super Mario World*.*

4.6 Triggers

Triggers are interactive objects that the avatar can use to alter the state of the level, or even game rules such as physics. Figure 9 shows an example of triggers from *Super Mario World*, where Mario can jump on the blue "P" button to turn all the blocks into coins.

Some triggers are also timed; for example, *Yoshi's Island DS* requires Yoshi to jump on a red button that turns red platforms active. Yoshi then has a short amount of time to run across the platforms and continue the level.

Figure 9: *Example of triggers from* Super Mario World.

Triggers can add an interesting puzzle element to a platform game. For example, *N* levels require the player to navigate their way to a series of triggers, with the eventual goal of hitting the trigger that opens the portal from that level to the next one.

To enable triggers, each element of the game knows what it depends on and what depends on it. This introduces the possibility of activating a trigger and setting off a chain reaction.

5 Structural Representation for Levels

Now that we have determined the prominent features of platform games, we can define a structure for each level. This structure can help us analyze existing levels for repeated patterns, challenge, and potential paths through the level. Our framework is inspired by Compton and Mateas's work on procedural level generation for platform games [Compton and Mateas 2006], especially the importance of rhythm and the hierarchical nature of the level representation.

5.1 Rhythm Groups

Rhythm and pacing are key to the player's enjoyment of the game [Bleszinski 2000; Compton and Mateas 2006] and also contribute significantly towards the difficulty of a platformer [Nicollet 2004]. We therefore set up "rhythm groups," composed of the various components discussed earlier. Rhythm groups are short, non-overlapping sets of components that encapsulate an area of challenge. These groups are inspired by a method for representing rhythmic structure in music [Iyer et al. 1997]. Rhythm groups help to identify challenging areas of a level and understand what makes them difficult. They also help us understand patterns and reuse in level design: rhythm groups are quite modular, and it is possible to reuse a rhythm group later in a level, or as a recurring theme in a set of levels, even if their graphical representation differs.

Our rhythm groups are analogous to musical phrases. They have a distinct start, middle, and end, and culminate in a cadence [Rothstein 1989]. This cadence is a place where the rhythm of the player's movement significantly changes. It could also be an area where the player is safe to rest and pause before continuing a new challenging area. Such rest areas are especially important in difficult levels, as long periods of action with little time to pause is more likely to result in player error [Nicollet 2004]. A transition between rhythm groups marks an area of rest, and could also be a place where there is a save point or reward for completing a challenging section of the level.

Rhythm can be found in platformers whenever the player performs actions such as jumping or shooting. These actions map directly to the controller, and the rhythm with which the player must hit the buttons on the controller is the rhythm we are defining. For example, a rhythm could be a series of three short hops, or alternating jumping back and forth up a series of platforms that form a ladder. An example of a rhythm group from *Yoshi's Island* is shown in Figure 10.

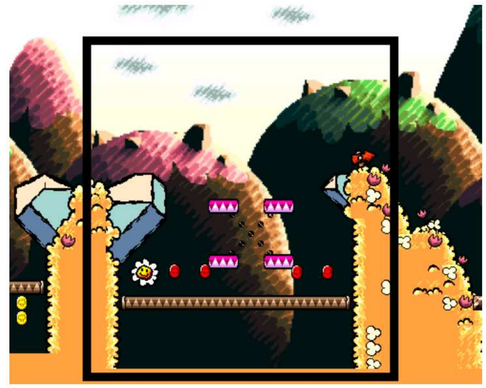

Figure 10: *An example of a rhythm group in* Yoshi's Island. *The components in the rhythm group are surrounded by a black rectangle.*

The player starts at the left of this rhythm group. They must then jump down onto the platform, gather coins and the flower, then perform a series of timed jumps on the rotating platforms back up to the top right. The rhythm group begins at the top left because it is an area where the rhythm has a pause; the player has just completed a similar jump to retrieve the two coins in the bottom left corner. It ends at the top right because the timed jumps around the rotating platforms have ended, and the player will now begin running down the hill towards the next challenge. Both the beginning and end of the rhythm group mark a change in the pace and rhythm of the player's actions. The rhythm group itself marks an area of challenge; once the player has retrieved the red coins he must navigate his way back up to the top right platform without falling down.

Figure 11 shows two rhythm groups from *Super Mario World*. At first glance it may appear as though the two groups shown should actually be a single group, since the platforms form a repeated pattern that the player must jump across. However, coin and enemy placement means that the components make up two separate rhythm groups.

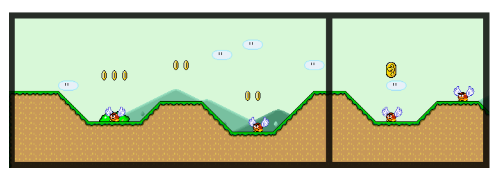

Figure 11: *Two rhythm groups in* Super Mario World. *Each rhythm group is surrounded by a black rectangle.*

The key to this being two rhythm groups rather than a single group is the middle pair of coins in the first group, and their lack of a counterpart at the point where the two groups meet. In the first rhythm group, the player must perform a series of jumps with minimal pauses to reach all the coins and kill the enemies. At the place where the groups meet, the player has an opportunity to pause, allowing him to correctly time his next jump to the Yoshi coin and the enemy. Even though this pause can be quite short, it still forms a break in the rhythm that would not be there if the player had coins to jump for on that second peak.

5.2 Cells and Portals

Cells and portals allow us to represent non-linearity in platformer levels. This non-linearity manifests itself in a variety of ways; for example, in *Sonic the Hedgehog*, there are usually multiple paths through a level. There are moments when the player can switch between these paths, and the player often attempts to find the path that allows him to complete the level in the shortest amount of time. In our structure for levels, each point where the paths meet is a portal.

Cells define regions of non-overlapping, linear gameplay connected by portals. Their boundaries are set by the placement of transitions into and out of the regions, such as a secondary entrance to a level, a transition between paths through a level, or a portal to a secret area in the game. Figure 12 shows two examples of cells and portals: the first is a scene from *Super Mario World* divided into two cells and one portal. The second is a more complex example taken from *Sonic the Hedgehog*; the two platforms marked as portals move up and down and provide a transition to a different path through the game. Note that for the sake of clarity, the figure does not have rhythm groups marked.

In the *Super Mario World* example, the pipe in the top middle of the picture is the portal. The player can go into the pipe and enter a new area. The cells are split at that pipe because it is possible for the player to choose a different path at that point. The two platforms in the *Sonic the Hedgehog* example move up and down, allowing the player to choose between the top and bottom paths through the level. There are therefore four different cells, each representing a linear section of gameplay that ends where the player can choose a new path.

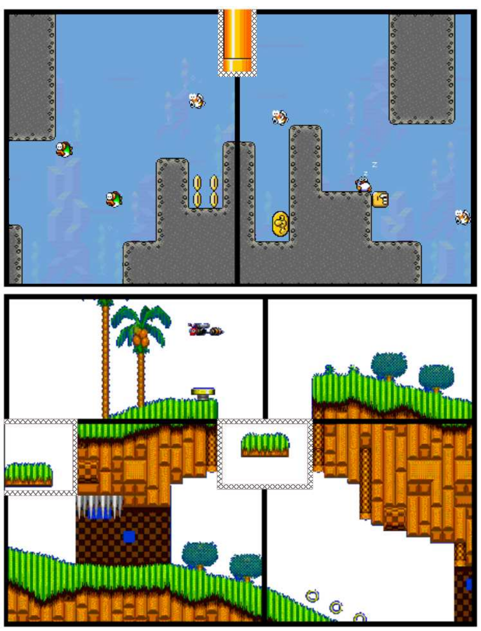

Figure 12: *Scenes from* Super Mario World *and* Sonic the Hedgehog *illustrating cells and portals. Cells are outlined with solid lines; portals with dotted lines.*

Knowing where the cells and portals are in a level helps us analyze their structure and catalog the many paths through a level. These paths may be of different difficulties, depending on the rhythm groups that make up the cells along the path.

6 Analysis of an Existing Level

Having defined our model for platformer levels, we now apply it to a section of a level from an existing game. We have chosen to use an early section of Emerald Hill Zone, Act 1 from *Sonic the Hedgehog 2*. We have split Figure 13 into labeled rhythm groups and portals. Each group is also lightly shaded, indicating that it forms part of a cell. We will discuss what each of these rhythm groups is, and also how they form cells and portals.

1. This rhythm group has the player run uphill and hit a spring to be able to reach the rings and checkpoint in the top left corner. The grassy areas that the avatar runs across are platforms in this section, both in the top left corner and the bottom region. There are no obstacles in this rhythm group. The spring at the end of this rhythm group is a movement aid. The coins are collectible items, and there are no triggers.

2. Another rhythm group in the same cell, this time giving the player several options. The platform on the left moves up and down, allowing the player to jump up to the top of the loop and collect a power-up. The player can then jump back down and complete the loop, rolling down towards the twister. Rhythm groups 1 and 2 form a cell, since the player can move into a different cell if he doesn't have enough speed to complete the twister.

3. This lower rhythm group forms an entire cell on its own. The pace in this group is slow relative to the rest of the level; the player has some speed but must only make one or two jumps to kill the enemies. The rhythm group ends for two reasons: firstly, there is a cell boundary that forces the end of the rhythm group. However, even without that cell boundary it would still end here because the rhythm changes from being one that primarily has the player running to one that has the player jumping more often.

4. This rhythm group begins with a platform we call the "twister". Even though it is possible to fall off the twister, no damage is done to the avatar. If the player has enough speed to complete the twister, he will also have enough to collect the lower set of coins. The group ends with the player collecting a box containing ten rings.

5. This space beneath the twister is a portal, where the player can transition from the upper left cell to either the upper right or lower right cell. This transition may be unintentional, as navigating the twister requires a great deal of speed to be built up already. Note that the portal is *not* a rhythm group, it is merely a region where it is possible to transition between cells.

6. A rhythm group characterized by timed jumps. The player must avoid the fish that jump up towards him from under the bridge; correctly avoiding the fish will reward the player with some rings. Failing to avoid the fish will reduce the player's ring tally to zero, but the coins above the fish serve as a guarantee of retaining at least some rings as a shield for the rest of the level. Groups 6 and 7 form a cell.

7. This last rhythm group consists of spikes that are easy to avoid as long as the player does not accidentally jump into them. It ends at a cell boundary.

Our analysis shows some interesting aspects of level design for *Sonic the Hedgehog*. The designers provide many choices for the player; these choices provide challenge since the main goal of the levels is to complete them quickly. The player must make the optimal choice to complete the level quickly.

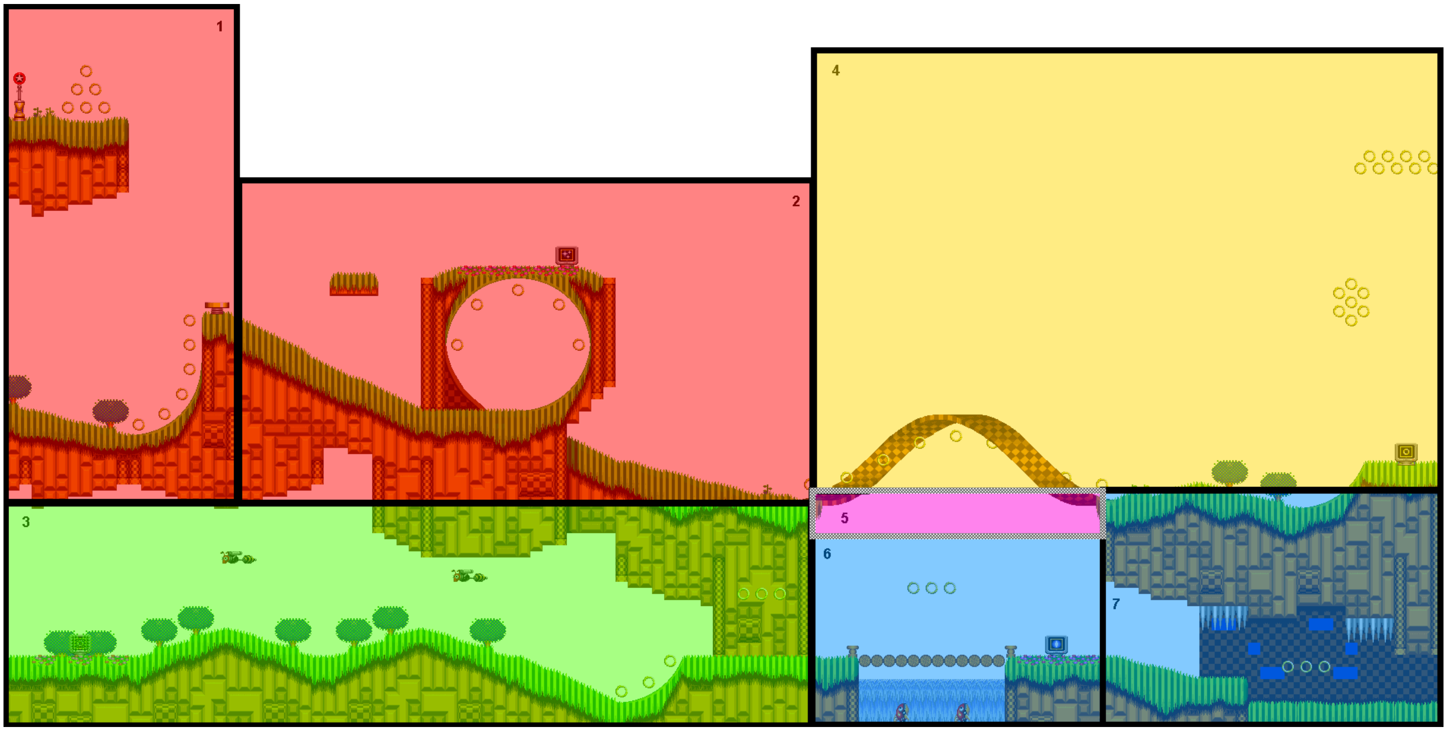

Figure 13: *A section of Emerald Hill Zone, Act 1 from* Sonic the Hedgehog 2*, split into numbered sections. Cells and portals are shaded. Each numbered section is discussed in the paper in terms of rhythm groups, cells, and portals.*

Rhythm group size and allocation reflects the speed with which the player moves. Groups 3 and 4 are quite large, making up their own cells. The size of these groups is deceiving, though; the player is required to jump no more than twice in each group to avoid obstacles, collect items, or destroy enemies.

We also see that the designer provides a second chance to players who fail to pick up enough speed to complete the twister. Region 5 shows the portal that lets the player transition to the lower right cell. Staying on the top path may be preferable, since there are a number of coins available to collect at the end of the top right cell. However, failing to complete the twister still provides compensation in the form of a temporary shield power-up. Indeed, the player may deliberately choose to fall through the twister and collect that power-up if he thinks it will help him later in the level.

7 Future Work and Conclusion

With a formal definition of a platformer level, including a model for rhythm-based challenge, there are a number of different kinds of analysis we can perform. For example, it may be helpful to look at ways of classifying the difficulty of a level based on the pace, length, and beat of a rhythm group. A short, arrhythmic section of the level may be difficult to master and therefore provide more difficulty to the player. Similarly, an extremely long and rhythmic grouping may be easier at first but requires the player to concentrate longer, leading to more points of potential failure [Nicollet 2004].

We would also like to explore automated generation of levels based on our model for their structure. To effectively generate levels, a computer must understand two major ideas: firstly, the components that make up the level, and secondly the way they fit together to create an entire level. It must also be able to reason about constraints on the model so it generates levels that are both playable and well balanced. These constraints are not yet known, and it would be interesting to explore a formal definition for what makes a level enjoyable.

We have presented a framework for analysis of platformer components and level design. This framework can be especially helpful to educators who teach students how to design fun and challenging levels in games. It is our hope that this framework will also provide a common vocabulary for both game analysts and developers and open the field to similar frameworks for other genres.

References

ADAMS, E., AND ROLLINGS, A. 2007. *Fundamentals of Game Design*. Pearson Prentice Hall, Upper Saddle River, New Jersey.

BJORK, S., AND HOLOPAINEN, J. 2004. *Patterns in Game Design*. Charles River Media, ch. 2, 7–32.

BLESZINSKI, C. 2000. The Art and Science of Level Design. `http://www.cliffyb.com/art-sci-ld.html`.

BOUTROS, D. 2006. A Detailed Cross-Examination of Yesterday and Today's Best-Selling Platform Games. *Gamasutra* (August). `http://www.gamasutra.com/features/20060804/boutros_01.shtml`.

BYRNE, E. 2005. *Game Level Design*. Charles River Media.

CO, P. 2006. *Level Design for Games: Creating Compelling Game Experiences*. New Riders.

COMPTON, K., AND MATEAS, M. 2006. Procedural Level Design for Platform Games. In *Proceedings of the 2nd Artificial Intelligence and Interactive Digital Entertainment Conference*.

DESURVIRE, H., CAPLAN, M., AND TOTH, J. 2004. Using Heuristics to Evaluate the Playability of Games. *CHI '04 Extended Abstracts on Human Factors in Computing Systems* (April).

FEIL, J., AND SCATTERGOOD, M. 2005. *Beginning Game Level Design*. Thompson Course Technology PTR.

IYER, V., BILMES, J., WRITE, M., AND WESSEL, D. 1997. A Novel Representation for Rhythmic Structure. In *Proceedings of the 23rd International Computer Music Conference*, 97–100.

NELSON, M. 2007. Breaking Down Breakout: System and Level Design for Breakout-style Games. *Gamasutra* (August). http://www.gamasutra.com/view/feature/1630/breaking_down_breakout_system_and_.php.

NICOLLET, V. 2004. Difficulty in Dexterity-Based Platform Games. *GameDev.net* (March). http://www.gamedev.net/reference/articles/article2055.asp.

ROTHSTEIN, W. 1989. *Phrase Rhythm in Tonal Music*. Schirmer Books, New York, ch. 1, 3–15.

SALEN, K., AND ZIMMERMAN, E. 2004. *Rules of Play: Game Design Fundamentals*. MIT Press, Cambridge, Massachusetts, ch. 23, 313–327.

ZAGAL, J. P., MATEAS, M., FERNANDEZ-VARA, C., HOCHHALTER, B., AND LICHTI, N. 2005. Towards an Ontological Language for Game Analysis. In *Proceedings of the Digital Interactive Games Research Association Conferences (DiGRA 2005)*.

Games Cited

1. *Donkey Kong Country 2: Diddy's Kong Quest*. Nintendo (SNES), 1995.
2. *Mega Man*. Capcom (NES), 1987.
3. *Mega Man X2*. Capcom (SNES), 1995.
4. *N*. Metanet Software (PC), 2005.
5. *New Super Mario Bros.*. Nintendo (DS), 2006.
6. *Sonic the Hedgehog*. SEGA (Genesis), 1991.
7. *Sonic the Hedgehog 2*. SEGA (Genesis), 1992.
8. *Super Mario World*. Nintendo (SNES), 1991.
9. *Super Mario World 2: Yoshi's Island*. Nintendo (SNES), 1995.
10. *Yoshi's Island DS*. Nintendo (DS), 2006.