Retro Evolved: Level Design Practice exemplified by the Contemporary Retro Game

Fares Kayali

University of Applied Arts Vienna, TEF - Flow/PPP group
Vordere Zollamtsstrasse 3/ ZG20
1030 Wien
fares@igw.tuwien.ac.at

Josef Schuh

University of Applied Arts Vienna, TEF - Flow/PPP group
Vordere Zollamtsstrasse 3/ ZG20
1030 Wien
josef@igw.tuwien.ac.at

ABSTRACT

The examples of independent games presented in this paper provide fun through strong, innovative and playful game mechanics. The often-resulting combination of retro flair with focused gameplay is what makes the "contemporary retro game". This paper argues that game and especially level design have evolved over time and provides insights on contemporary level design practice. By a qualitative analysis of contemporary game examples and a reflection of the authors' own practice as game and level designers, this paper defines qualities of good level design and presents several distinct level design practices; to expand on strong core mechanics, iterative level design, to design game modes instead of levels, sandboxes, emergent gameplay and object-oriented level design.

Keywords
Game design, level design, retro games, game mechanics, core mechanics

INTRODUCTION

The last years have seen a great rise in popularity for independent games (games created by mostly small publisher-independent teams). This phenomenon can be partially attributed to the emergence of novel distribution platforms and lightweight development tools resulting in a significantly lowered entry barrier. Many examples of independent games presented in this paper feature 2D content and provide fun through strong, innovative and playful game mechanics. The often-resulting combination of retro flair with modern, focused gameplay is what makes the "contemporary retro game". This

paper argues that game and especially level design have evolved over time and provides insights on contemporary level design practice.

The level designer operates at a unique position at the intersection between programming, design and art (Byrne 2004). These aspects are usually combined to evoke a certain mood in the player (Logas & Muller 2005). An important skill for level designers is to distill ideas into solid, workable games (Feil & Scattergood 2005) while leading players through the experience of the game without revealing their designer's presence in a way that distracts the players from their own virtual destiny (Byrne 2004). Level Design can be regarded as the process that implements the game design as well as all kinds of assets in the game. This paper helps clarify the prevalent view that often equates many tasks of game and level design by isolating and situating aspects of level design practice.

The contemporary retro game helps to illustrate a newly found focus on level design and on building level design elements on strong core game mechanics. A methodology will be used that combines a qualitative analysis of several examples of such games with a reflection on the authors' own practice as game and level designers. As results of the analysis this paper will present a series of distinct level design practices that relate to the design of contemporary retro games.

CORE MECHANICS

This paper sees level design elements or patterns (Bjoerk et al 2003) as a result of building on game and core game mechanics. Especially contemporary retro games seem to build on one or more strong core mechanics that are explored and expanded through gameplay. This emergent approach is primarily made possible through level design that playfully experiments with the possibilities that are opened up through a game's set of mechanics. Therefore understanding what game and core mechanics are is the basis for analyzing how these mechanics shape level design practice.

Järvinen (2008) describes game mechanics through verbs, regarding them as the actions that are available to the player. Similarly in their definition of the MDA (mechanics, dynamics, aesthetics) framework Hunicke et al (2004) state that "*Mechanics are the various actions, behaviors and control mechanisms afforded to the player within a game context.*"

Building on these definitions Sicart (2008) describes core mechanics as the game mechanics needed to *"achieve a systemically rewarded end-game state"* while Salen and Zimmerman see them as *"the essential play activity players perform again and again in a game"* (2004).

For the purpose of this examination, core mechanics are regarded as the defining game mechanics of a game. They can be seen as the building blocks around which a level designer can grow a level that plays with these mechanics. For example the platformer VVVVVV's (Cavanagh 2010) core mechanic is that the player can reverse the direction of gravity but is not able to jump. A simple pattern based on this mechanic would be to place obstacles on the floor that the player has to evade by switching gravity to run on the ceiling. Expanding on this mechanic, the player later even navigates between and below moving platforms by switching gravity. The understanding of the importance of core

mechanics in combination with the following account on what good level design practice is builds the foundation for our analysis of level design in contemporary retro games.

QUALITIES OF GOOD LEVEL DESIGN PRACTICE

As a starting point for the research of different approaches to level design the following section explains which qualities of good design practice have to be satisfied to build engaging and well-balanced game levels. The authors' regard level design as a tool to expose the game design vision, centering on the purpose of engaging the player, similarly to Cliff Bleszinski's (2000) "carrot on a stick" metaphor and on handling player expectations (Kayali & Pichlmair 2008). The defining qualities of good level design include:

- To maintain the delicate balance between difficulty and challenge. By avoiding anxiety and boredom players should be kept within an ideal flow channel (Csikszentmihalyi 1990, Chen 2007).

- To achieve a balance between giving players freedom to explore while imposing constraints to ensure continuity and fluent play thereby giving players the illusion of control (Kayali & Purgathofer 2008).

- To provide the possibility to recombine game elements and actions and to achieve elusive mastery of the game. This also results in emergent gameplay and provides an incentive to replay levels. In a way this relates to the quality of malleability that Golan Levin (2000) describes for electronic instruments.

- To balance risk and reward to give players risk incentive (Ryan 1999). A level always should include easier and harder options that are rewarded proportionally.

- To drive the game's narrative. Level design has to ensure individual levels are aligned within the overall progress through the game. To ensure continuity, a level can only introduce new elements as described in the asset revelation schedule.

- To guide the player (Van Best 2010, Schuh 2008). Mirror's Edge (D.I.C.E. 2008) with its distinct use of the color red to mark things the player can cling to is the prime example here (Silli, 2010).

METHODOLOGY

By a qualitative analysis of contemporary retro game examples and the reflection of the authors' own practice this paper defines a series of distinct level design practices. The qualitative analysis uses a "playing research" approach (Aarseth 2003) combining it with a reflection of an explorative design (Ehn & Löwgren 2004) or "design as research" (Burdick 2003, Stapleton 2005) methodology that involves the authors' own games.

To have an approach that combines play with practice the authors asked themselves the following questions. The same questions are used as a basis to look at the qualitative examples.

• How did you approach the level design of your game?

• What lead you to the specific level design practice used in the game?

• What are the benefits of this practice?

• What are the drawbacks?

• How do core gameplay and level design relate to each other?

• At what point in the game production process did most of the level design take place?

• How did constraints (time, budget, assets) shape the level design (practice)?

• How does your level design practice shape player experience?

• How is your game 'retro' / what retro game genre would you align it with?

• How does your game combine retro flair with contemporary gameplay and concepts?

• How did the game controller and interface influence the level design?

Practice: Analysis of the authors' own work

The purpose of this section is to give a deeper understanding of details of the actual design practice that cannot be derived from just playing a game. The authors tried to step back from their own work and act as qualified observers (Lammes 2007) in the following two sections.

Radio Flare REDUX

Radio Flare Redux (Kayali & Pichlmair 2010) is a side-scrolling shoot'em up with multi-touch controls and it is a music-based game where each level is a song. All game elements, from background animations to enemy movement and shooting patterns, are tied to the rhythm of a level's soundtrack. All sound effects triggered by the gameplay blend with the music. Consequently the game's level design process was tightly connected to the music. The first step of designing a level always meant listening to the respective song many times to identify things like sections, recurring elements and different themes. The level files of the game consequently read like a musical score. All events scripted in the level files are already quantized to the beat (Pichlmair & Kayali 2007), meaning that the time index used in level files uses bars and beats as measures. This also means that levels would play slower or faster if used with a song with different BPM (beats per minute). Overall Radio Flare Redux combines the retro style and basic gameplay of a side-scroller with the concept of synesthesia and interactive music. The game's multi-touch controls are tailored to the iPhone and iPad platforms.

In the game you use one thumb to control the ship and the other to target enemies with a swiping motion. When you release your thumb all targeted enemies are destroyed in sync with the music. Radio Flare Redux' core gameplay means alternating between queuing up as many enemies as possible and dodging forward to collect the radio flares released by destroyed enemies. In a sense this core gameplay also happens in rhythm, as does everything else in the game. The rhythmic interaction is designed to connect to the

overall synchronicity between music and visuals to deeply immerse players. The level design follows the music not only in rhythm but also in structure and themes. Ideally different sections of a level read like variations on a theme. The combination of the repetitiveness of enemy waves with timed variations of these patterns is an analogy to a level's electronic music soundtrack that evolves similarly.

What is beneficial about this approach is that the possibility to plan everything exactly tightens the connection between music, visuals and gameplay. It enables the level designer to create a dramaturgy of enemy waves and events that support the flow of a level's music. At the same time this approach results in a very linear and static experience that mostly voids emergent gameplay situations.

Most of the level design for Radio Flare Redux was done close to release. This was a necessity because it was only then that all music licenses were cleared and the soundtrack definitive. The game would have benefited from starting earlier, which might have produced a deeper, and more varied pool of enemy patterns and game elements to draw from. How many elements are needed to build a good, varied level is something that is hard to judge in advance. Also many good ideas arise during the level design process that are mostly lost if the design is done late in the production cycle. The bottom line is that the approach to level design worked very well in connecting the gameplay to the music and thereby providing an immersive experience but had serious drawbacks in terms of gameplay variety and depth.

Hue Shift

Hue Shift (Schuh 2010) is an endless action platformer. The player uses the arrow keys to control an oversized pixel that can shift its color between red, green and blue. The goal is to climb as high as possible while matching the pixel's color to the color of the platforms. Only platforms that have the same color as the player remain solid. If the color does not match, the pixel will fall through the platform and the player has to start again.

Both level design and core gameplay are closely related because the level has to react to the player's actions and change itself accordingly. As a consequence, level and gameplay design happened simultaneously and affected each other during the whole design process. The first approach used in Hue Shift was level based. Challenges were designed manually and the goal for the player was to finish each level as quick as possible. Due to the fact that the player speed is constant, early user testing did show that there is a maximum high score, which does not motivate players. An article by Adam Saltsman (2010) inspired a new approach to the level design, which suggested an endless, self-generated level. With this approach, players were able to play the game until a mistake ended the run - a practice, which led to an active battle for high scores.
The level itself was generated out of 50 predefined blocks, each containing between 2 and 6 platforms. These blocks were split into 3 categories - easy, medium and hard. The first few tiers of the level were designed to be easy, followed by medium and later hard blocks. After a while, blocks were chosen randomly from all difficulties. The biggest challenge was keeping the level solvable. The blocks had to be designed in a specific way so every block could be combined with every other block.

Despite the fact that solving the level design problem took longer than expected, the overall time of the level design process was short and cheap compared to other projects.

Mainly because only a small amount of predefined blocks needed to be designed to ensure a challenging and diversified gameplay experience.

User testing showed that randomized level generation increased player motivation. When a mistake was made, players hoped for a better random level on their next run. Conversely, some players were frustrated after experiencing some tricky random levels. However, they still hoped for a better level in subsequent tries.

Hue Shift is often compared to 'retro' games because it has a very low-end graphic style and limits the player to two actions - jump and change color. Contrary to retro games, one of the major goals of the game was to somehow limit the playtime for each run to approximately 5 to 8 minutes. An approach adopted by many mobile games like Doodle Jump (Lima Sky 2009) or Canabalt (Saltsman 2009).

The level design itself also follows a 'retro' approach. For each run, the level is self-generated by combining predefined blocks that are picked by random. Using only a small number of chunks and combining them as often and varied as possible is a method, which was often used by games that only had very limited disc space at their disposal.

Play: Qualitative Examples

Using literature research and the insights gained from the previous reflection of the authors' own practical work we identified a series of representative game examples that are analyzed in this section. All five selected games are from the contemporary retro game genre. The elegant simplicity of their core game mechanics makes them easy to understand and each of the five games has been selected to illustrate a specific level design practice. The subsequent table aligns the authors' two games and the five qualitative examples with classic game genres. A description of each game's level design practice is given. New school concepts are described to give an understanding why the chosen games, while having a retro game flair, still are modern games that build upon established modern gameplay paradigms and use novel features. The core mechanic is described to understand its importance to the level design.

Game	Retro Genre	Level Design Practice	New School Concept	Core mechanic
VVVVVV	Jump'n'Run	Expand on a strong core mechanic	Accelerated retries and negating the importance of lives & game over	Switch gravity
Galcon	Real-time strategy	Design game modes not levels	Ad-hoc online multiplayer, competitive high scores	Send units to conquer planets
Canabalt	Automatically scrolling platformer	Iterative Level Design	One-Button Game	Control speed and jump across collapsing buildings
Sleep is Death	Text adventure with GUI	Sandboxes and emergent gameplay	Network play and ad-hoc user generated content	Play in a world dynamically created by another player
Bit.Trip Beat	Pong for one player	Object-oriented level design	Play along with the music	Reflect patterns of objects in rhythm
Hue Shift	Automatically scrolling platformer	Iterative level design	In-game score comparison with friends and other players	Match player and platform color
Radio Flare REDUX	Side-scrolling shoot'em up	Object-oriented level design	Interactive music	Swipe to target multiple enemies

Table 1: an overview of the practical work of the authors as well as the qualitative examples and their significance to the presented research.

LEVEL DESIGN PRACTICES

Building on the analysis of qualitative examples the following level design practices used in contemporary retro games could be identified:

Expand on strong core mechanics

"VVVVVV explores one simple game mechanic: you cannot jump - instead, you reverse your own gravity at the press of a button. The game focuses on playing with this mechanic in a variety of interesting ways." (Cavanagh 2010)

The self-description of VVVVVV (Cavanagh 2010) from its product page on the Steam download platform gives a good summary of the game's core gameplay. It is a classic platformer game that builds its potential out of exploring a single and simple core mechanic. It constrains input to left and right movement and to switching gravity between up and down. By reversing gravity players suddenly find themselves running on the ceiling or having to walk across the bottom side of moving platforms. VVVVVV has very difficult yet fair gameplay. Playing is based on retrying small sections very often. To support this the game makes frequent use of well-placed checkpoints so that players never have to replay longer sections to get to the place where they failed. There also are no lives and no game over. Each section presents either a new or slightly refined challenge that builds on using the game's core mechanic the right way and by using exact timing in doing so. The game's reduced graphical style and the use of basic platformer building blocks contribute to VVVVVV being a very focused example of how contemporary retro games can combine a distinct retro style with modern gameplay that is grown out of exploring a strong core mechanic through the game's level design. Similarly physics-based gameplay often relies on one core mechanic; for example rotating the whole game world with the Wii remote in order to find a path through the indie puzzle platformer game *And Yet It Moves* (Broken Rules, 2009).

Iterative level design

Canabalt (Saltsman 2009) is an automatically scrolling platformer in which the player needs to escape an unknown threat. The player is only able to jump across gaps, obstacles or boxes and cannot stop the movement at any time. Only the speed can be slightly modified by running into boxes to slow down a little bit. Otherwise the movement speed is increased over time.
The game contains only one endless level, which changes every time a new session is started and contains only a few distinguished elements.
1. Non-moving platforms (in various forms like rooftops and annunciator panels)
2. Moving platforms (collapsing buildings)
3. Bottlenecks (large buildings where you can only cross through one floor)
4. Random enemy encounters (objects that suddenly fall from the sky)
Those elements are generated procedurally in a similar manner like the levels in Hue Shift, which was described above. In an article, Adam Saltsman (2010) - the creator of Canabalt - defines procedurally generated content as "*anything that exists in a game that wasn't strictly scripted that way by a designer. It might be a dynamically generated dungeon or a randomly rolled adventurer"*. He describes various systems, which can be used for computer-generated levels and uses a tile-based system in Canabalt himself. He states that "*tiles are randomly arranged (though frequently with some constraints) to*

create the terrain or playspace of the game". Although these methods are comfortable to use once they are designed, there are several things to keep in mind when using them. *"All these systems, despite their obvious promise, are fraught with potential disasters, many due to the players' aforementioned need for pseudo-predictable surprises. If your algorithm is too predictable and well ordered, players will become bored or hack the system. If it's too abtuse and unpredictable, players can't explain inconsistencies and will be frustrated or overwhelmed"*, reports Saltsman.

The control of the player in Canabalt happens by pressing one button respectively by touching the screen of a touch-sensitive device. This type of control became popular among small flash and mobile games and even inspired a one-button game programming competition during the Game Developers Conference in March 2010[1]. Simple rules like *"Use a single button as your player input, in a unique or experimental way. Be creative."* inspired developers to games which can be picked up and played immediately. Simple control concepts like only pressing one button can also be found in early arcade and videogames which used a joystick and a maximum of two or three buttons as input method.

Design game modes not levels

Daniel Cook (2011) argues that designing interesting game modes eliminates the need to design levels and makes play varied each session as evidenced in his game Steambirds: Survival (Radial Games & Spry Fox 2011). The authors used pool-based approaches (drawing from a subset of enemy formations at random) in Radio Flare (Pichlmair & Kayali 2008) and Zombies vs. Sheep (Pichlmair & Kayali 2009) to provide variation within a game mode as opposed to the linear wave based approach in Radio Flare REDUX (Kayali & Pichlmair 2010).

The real time strategy game Galcon (Hassey, 2008) uses a very simple game mechanic with a variety of game modes. The game consists of a map with randomly placed planets in different sizes. These are generated procedurally with a Node-based system and use a grid as described in Saltsman (2010). The player starts on a planet and can drag his units (represented by triangles) to other planets. If the amount of sent units is larger than the amount of units on the new planet, the player takes over this planet and produces units on it. The bigger the planet is, the bigger is its production rate. Once all enemy troops are gone, the player wins.

Because of these simple mechanics, various modes of the game can be played.

1. Classic Mode

The player and one AI-controlled player start on the same map, both with one planet under their control.

2. 3-way

The player plays against two AI-controlled players.

3. Vacuum Mode

Only the player starts on the map and has to conquer all planets before a timer runs out.

4. Stealth Mode

Stealth Mode follows the same rules as Classic Mode with the difference that enemy ships cannot be seen.

All these modes can be played immediately once the player understands the core mechanic of the game. After the success of Galcon, another game named Galcon Labs

(Hassey 2009) was released. It introduces even more game modes without changing the core mechanics themselves.

Sandboxes and emergent gameplay

Many examples exist where level design is given to the user as a toolbox. Games like the arcade racing game Trackmania (Nadeo 2003), the 4-player creative jump'n'run Little Big Planet (Media Molecule 2008) or the recent overwhelming success of Minecraft (Mojang 2009) show that passing a good toolbox of level design elements to players can yield endless amounts of interesting levels with emergent gameplay. Examples even include working computers that can be built using the tools provided by both Minecraft as well as Little Big Planet. Trackmania gave birth to the idea of "press forward" sections of a racetrack; a predefined spectacle that relies on the player to accelerate without steering or braking.

The most interesting example in the context of contemporary retro games is Jason Rohrer's (2010) online adventure game *Sleep is Death*. It is a very retro style game, both in appearance and gameplay. It uses pixel graphics and the world is presented from a pseudo top down view. There is no scrolling, the game only switches between individual scenes. Sleep is Death is a text adventure at its core. It has a graphical user interface but player action is determined by typing sentences to trigger actions and dialogue. The novel thing about the game is that two players play it with one of them designing the levels on the fly. The game comes with a large set of scenery, building and character items. Players take turns that are limited to 30 seconds each. One player types in an action and/or something she wants to say while the other (who represents the game) has to react within 30 seconds by moving characters and objects, introducing new things or by responding through dialogue. The game does not impose any limits on what players can try out. By passing on level design to the players, playing Sleep Is Death always results in a novel experience with endless potential for emergent gameplay that is only limited by the players' imagination. The strength of the game design does not lie in the level design itself but in providing a set of level design tools and objects that are easy to use but open up a wide space of possibilities through recombination. The most intriguing potential of Sleep Is Death is it's capability to provide a means for interactive storytelling to players.

Object-oriented level design

Object-oriented design or in short OOLD, is a term coined by Rudolf Kremers (2009). The concept, as the name already implies, is derived from the practice of object-oriented programming. It is an incremental technique, where assets are introduced and then reused with variations. The player first learns basic patterns. These are then expanded and varied during the course of a level (and the game).

Bit.Trip Beat (Gajin Games 2009) is a simple action game where the gameplay is a one-player variant of Pong (Alcorn 1972) meshed with rhythm-based gameplay. The player controls a paddle to reflect incoming patterns of abstract shapes. Different shapes and colors of objects have distinct movement patterns. The reflection of objects is aligned to the rhythm of the music. Sound effects triggered by the paddle blend into it. New patterns are in most cases introduced by a single object first. Then there are waves of the same objects and combinations of these waves. They are either combined with waves of the same kind or with previously introduced patterns. The game uses a simple 8-bit retro

style in both its graphics and music. The repetitive but engaging gameplay is tied to the music beautifully and together an immersive experience is created.

Not incidentally both the qualitative example of Bit.Trip Beat and the authors' own Radio Flare Redux relate object-oriented level design with music. The introduction and subsequent variation of patterns is something naturally found in music and can form a coherent gameplay experience by using an OOLD approach to introduce level design patterns.

Summarized OOLD is a technique that allows for varied gameplay while at the same time saving resources. The drawback of this technique is that it easily leads to levels feeling repetitive. This can be partially overcome by altering not only gameplay but also graphical and acoustic representation of game elements that are subject to an object-oriented design approach. If used well (by employing a balanced ratio of repetition and variation) this practice poses a cognitive challenge through variation while also giving a sense of reward because players know part of the 'solution'. Consequently this balance can be compared to the delicate balance that keeps players in the aforementioned 'flow channel' (Chen 2007), with too much repetition potentially leading to boredom while too many new elements might induce anxiety.

DISCUSSION

This paper presented the level design practices of expanding on strong core mechanics, iterative level design, designing game modes instead of levels, emergent gameplay and sandboxes as well as object-oriented level design. These approaches are not meant to be mutually exclusive. In fact most games might have been shaped by aspects of several of these practices. Especially building on strong core mechanics is also a quality of good game design itself.

Choosing the correct level design practice is an important part of the level design process. As shown in Hue Shift (Schuh 2010), the whole design process can be accelerated after making a good decision on an approach to level design.

For good level design it is essential to start the design process as early as possible in the production timeline. If the level design process starts too late, it cannot be iterated often enough. By observing the level design process of Radio Flare Redux (Kayali & Pichlmair 2010) one can see, that the process started too late and new ideas had to be left out. It is important to detect circumstances, which delay the level design process so they can be bypassed as early as possible.

In addition, early user testing has proven to be an important aspect of level design. It can demonstrate problems, which can be missed by developers like the maximum high score from the level based approach in an early Hue Shift version.

Actually all of the distinct approaches to level design call for level design being considered in early stages of the game design process (Crawford 1997). When building on core mechanics there is a very tight connection between shaping that mechanic and the process of designing individual levels. When using a 'game mode', sandbox or iterative level design strategy the boundary between game and level design becomes blurred. Game design ideally defines a rich toolbox for level designers that supports the discussed

qualities like flow, balance and malleability, resulting in refined more gameplay-oriented experiences for the player.

FUTURE WORK

This paper presented a strictly analytical approach to level design. Although reflective insights on the authors' own practice as designers were presented, a truly explorative approach would benefit the topic. Focused experiments or prototypes could be used to test individual practices and shed more light on specific benefits and drawbacks.

Overall this paper, as can future research into the topic of level design, contributed to form terminologies for game studies to understand the importance of actual design practice to level design.

ENDNOTES

1 http://switchgaming.blogspot.com/2009/12/gamma-4-one-button-game-competition.html [accessed July 19th, 2011].

BIBLIOGRAPHY

Aarseth, E. "Playing Research: Methodological Approaches to Game Analysis." In Game Approaches Conference. Spilforskning, Denmark, 2003.
Alcorn, A. (1972) Pong [Arcade] Atari.
Björk, S., Lundgren, S., and Holopainen, J. "Game Design Patterns." In Level Up- 1st international Digital Games Research Conference, 4-6. Utrecht, 2003.
Bleszinski, C. (2000) The Art and Science of Level Design. Available at http://www.cliffyb.com/art-sci-ld.html (accessed Feb. 21st 2011).
Broken Rules (2009) And yet It Moves [PC game] Broken Rules.
Burdick, A. "Design (as) Research." In Design Research: Methods and Perspectives, edited by B. Laurel: MIT Press, 2003.
Byrne, E. Game Level Design: Charles River Media, 2004.
Cavanagh, T. (2010) Vvvvvv [PC game] Terry Cavanagh.
Chen, J. "Flow in Games (and Everything Else)." Communications of the ACM 50, no. 4 (2007): 31-34.
Cook, D. (2011) Steambirds: Survival: Goodbye Handcrafted Levels. Available at http://www.lostgarden.com/2010/12/steambirds-survival-goodbye-handcrafted.html (accessed Feb. 21st 2011).
Crawford, C. (1997) The Art of Computer Game Design. Sue Peabody, Available at http://library.vancouver.wsu.edu/sites/library.vancouver.wsu.edu/files/ACGD.pdf (accessed Feb. 16th 2011).
Csikszentmihalyi, M. Flow: The Psychology of Optimal Experience. New York: Harper, 1990.
D.I.C.E. (2008) Mirror's Edge [multi platform] Electronic Arts.
Ehn, P., and Löwgren, J., eds. Design [X] Research: Essays on Interaction Design as Knowledge Construction. Vol. 3, Studies in Arts and Communication: Malmö University, 2004.
Feil, J. H., and Scattergood, M. Beginning Game Level Design: Course Technology PTR, 2005.

Gajin Games (2009) Bit.Trip Beat [Wii game] Aksys Games.

Hassey, P. (2008) Galcon [iPhone game] galcon.com.

Hassey, P. (2009) Galcon Labs [iPhone game] galcon.com.

Hunicke, R., LeBlanc, M., and Zubek, R. "Mda: A Formal Approach to Game Design and Game Research." In Challenges in Game AI Workshop. San Jose, CA, 2004.

Järvinen, A. "Games without Frontiers: Theories and Methods for Game Studies and Design." PhD, University of Tampere, 2008.

Kayali, F., and Pichlmair, M. "Intentions, Expectations and the Player." In [player] conference, 2008. Kopenhagen, 2008.

Kayali, F., and Pichlmair, M. (2010) Radio Flare Redux [iPhone game] studio radiolaris / Chillingo.

Kayali, F., and Purgathofer, P. "Two Halves of Play - Simulation Versus Abstraction and Transformation in Sports Videogames Design." Eludamos. Journal for Computer Game Culture 2, no. 1 (2008).

Kremers, R. Level Design: Concept, Theory, and Practice: AK Peters, 2009.

Lammes, S. "Approaching Game-Studies: Towards a Reflexive Methodology of Games as Situated Cultures " In Digital Games Research Association 2007 Conference - Situated play, edited by Akira Baba. Tokyo, Japan: University of Tokyo, 2007.

Levin, G. "Painterly Interfaces for Audiovisual Performance." master, Massachusetts Institute of Technology, 2000.

Lima Sky (2009) Doodle Jump [iPhone game] Lima Sky.

Logas, H., and Muller, D. "Mise-En-ScèNe Applied to Level Design: Adapting a Holistic Approach to Level Design." In DiGRA 2005. Vancouver, CA, 2005.

Media Molecule (2008) Little Big Planet [PS3 game] Sony Computer Entertainment Europe.

Mojang (2009) Minecraft [PC Game] Mojang.

Nadeo (2003) Trackmania [PC game] Valve.

Pichlmair, M., and Kayali, F. "Levels of Sound: On the Principles of Interactivity in Music Video Games." In Digital Games Research Association 2007 Conference - Situated play, edited by Akira Baba. Tokyo, Japan: University of Tokyo, 2007.

Pichlmair, M., and Kayali, F. (2008) Radio Flare [iPhone game] studio radiolaris.

Pichlmair, M., and Kayali, F. (2009) Zombies Vs. Sheep [iPhone game] studio radiolaris / Clickgamer.s

Radial Games, and Spry Fox (2011) Steambirds: Survival [PC game].

Rohrer, J. (2010) Sleep Is Death (Geisterfahrer) [PC game].

Ryan, T. (1999) Beginning Level Design. Gamasutra, Available at http://www.gamasutra.com/view/feature/3329/beginning_level_design_part_1.php (accessed Feb. 20th 2011).

Salen, K., and Zimmerman, E. Rules of Play. Cambridge, Massachusetts: MIT Press, 2004.

Saltsman, A. (2009) Canabalt [PC game].

Saltsman, A. "Constructed Procedural Generation." Game Developer Magazine, no. 2/2010 (2010).

Schuh, J. "Beyond the Map: Navigation Systems for Virtual Worlds." University of Technology, 2008.

Schuh, J. (2010) Hue Shift [PC game].

Sicart, M. "Defining Game Mechanics." Game Studies 8, no. 2 (2008).

Silli, E. (2010) Mirror's Edge - Level Design Challenges & Solutions. Game Developers Conference 2010, Available at http://www.slideshare.net/DICEStudio/e-silli-taome (accessed Feb. 21st 2011).

Stapleton, A. J. "Research as Design-Design as Research." In Digital Games Research Association 2005 Conference - changing views: worlds in play, edited by Akira Baba. Vancouver, 2005.

Van Best, M. (2010) A Trail of Breadcrumbs: Level Design and Player Experience. Nisute, Available at http://www.nisute.com/2010/08/18/needs-images-a-trail-of-breadcrumbs-level-design-and-player-experience/ (accessed Feb. 21st 2011).