



Game Design Patterns Consolidation

Amir M. Abu Makhadi

National Research University Higher School of Economics. Moscow, Russia. Email: [abu.amir\[at\]mail.ru](mailto:abu.amir[at]mail.ru)
ORCID <https://orcid.org/0009-0001-9456-1516>

Received: 20 September 2024 | Revised: 2 January 2025 | Accepted: 10 January 2025

Abstract

The field of game design patterns has grown up rapidly because of the necessity of tools supporting game design process on video games, serious games, and education. However the field does not meet criticism from academia and industry such as fragmentation, lack of collaboration, inconsistent template, navigation and patterns contradiction which leads to incompatible pattern collections and as a result to disparate application of game design patterns.

Despite the criticism of fragmentation of game design pattern collections, there is a lack of works aimed at consolidation of game design patterns. The intention of this article is to contribute to closing this gap by examining pattern collections, identifying their weaknesses and revealing practices that would compensate for those weaknesses and become a foundation for patterns consolidation.

Using comparative analysis, the article reviews six works aimed at forming game design pattern collections that behave like a “shared vocabulary”. Each work is compared according to six criteria.

Based on this comparison the article presents three main findings: 1. Despite inconsistent pattern template, existing pattern collections can extend each other without losing collection specific format. 2. Keywords are a reliable way for navigation in pattern collections. 3. “Related patterns” field could be used for describing conflicting and complementing patterns.

According to those findings, a consolidated pattern template was suggested which can be used for consolidation between collections of game design patterns.

Keywords

Game Design; Design Patterns; Pattern Language; Pattern Collection; Pattern Template; Pattern Consolidation; Pattern Navigation; Pattern Contradiction; Game Studies; Ludology



This work is licensed under a [Creative Commons “Attribution” 4.0 International License](https://creativecommons.org/licenses/by/4.0/)



Консолидация паттернов геймдизайна

Абу Махади Амир Мохаммедович

Национальный исследовательский университет «Высшая школа экономики». Москва, Россия.

Email: [abu.amir\[at\]mail.ru](mailto:abu.amir[at]mail.ru) ORCID <https://orcid.org/0009-0001-9456-1516>

Рукопись получена: 20 сентября 2024 | Пересмотрена: 2 января 2025 | Принята: 10 января 2025

Аннотация

Область паттернов геймдизайна быстро развивается из-за необходимости в инструментах, поддерживающих процесс дизайна в видеоиграх, серьезных играх и образовании. Однако коллекции паттернов геймдизайна не выдерживают критики со стороны академического сообщества и индустрии; это, например, относительная фрагментация коллекций, отсутствие консистентного шаблона, сложность навигации и противоречия между паттернами, что приводит к несовместимым коллекциям паттернов и, как следствие, к разрозненному применению паттернов на практике.

Несмотря на критику фрагментации коллекций паттернов, существует недостаток работ, направленных на консолидацию между этими коллекциями. Цель данной статьи — внести вклад в устранение этого пробела, рассмотрев разные коллекции паттернов, раскрывая их слабые и сильные стороны, а также выявив практики, которые могли бы компенсировать недостатки и стать основой для консолидации паттернов.

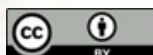
Используя сравнительный анализ, в статье рассматриваются шесть работ, направленных на формирование коллекций паттернов геймдизайна, которые формируют «вокабуляр геймдизайна». Каждая работа анализируется по шести критериям.

На основании сравнительного анализа представлены три основных вывода: 1. Несмотря на несоответствие шаблонов, существующие коллекции паттернов могут расширять друг друга, не теряя при этом специфического для коллекции формата. 2. Ключевые слова являются надежным способом навигации по коллекциям паттернов. 3. Поле «Связанные паттерны» может быть использовано для описания конфликтующих и дополняющих паттернов.

В соответствии с этими выводами статья предлагает шаблон консолидированного паттерна, который можно использовать для объединения коллекций паттернов геймдизайна.

Ключевые слова

геймдизайн; паттерны проектирования; язык паттернов; коллекция паттернов; шаблон паттерна; консолидация паттернов; навигация по паттернам; противоречие паттернов; исследование видеоигр; людология



Это произведение доступно по лицензии [Creative Commons “Attribution” \(«Атрибуция»\) 4.0](https://creativecommons.org/licenses/by/4.0/)



Introduction

This article attempts to address academia and industry criticism addressed to game design pattern collections. Previous research focused on forming pattern collections for game design, but our intention is to find a way for consolidation of pattern collections. The article starts from a brief field review including definition, history of game design patterns and criticism addressed to the field. Then comes extraction of main issues that arise from criticism which are navigation problems, conflicting patterns and an inconsistent pattern template. After that the article identifies the research gap of the field: lack of work focused on game design patterns consolidation. In the materials section it's selected works that are attempting to create a "shared vocabulary" for game design. In the method section the article describes the comparison method used for the research. The results section provides a sum up table with results of the collections comparison. The discussion section returns to issues that come from criticism and using the comparison results attempts to resolve them, including suggestion of a pattern template for consolidated collection and an example of pattern which uses suggested template. It is worth noting that the article does not offer a consolidated collection, but discusses approaches that would allow consolidation.

Game design, as a profession, required a set of theories and practices (often called game design rules or tools) to extract, share and save knowledge between game developers. This hunger of theories has been satisfied by game designers who wrote books and articles in game design cites and journals. Ones of most known of those works nowadays are Christopher Crawford's "The Art of Computer Game Design" (1984), Doug Church's "Formal Abstract Design Tools" (1999), Hal Barwood's "400 Design Rules" (2002), Raph Koster's "Theory of Fun for Game Design" (2004), Jesse Schell's "The Art of Game Design" (2008). This list is not complete, there are many more books, conference papers and articles about game design.

Video game designers and researchers such as Gonsalo Frasca (1999) and Bernd Kreimer (2002) set out to create a language that would allow them to introduce a set of terms and theories that could be used jointly by developers, critics, journalists, and researchers working in various genres. While Frasca's intention is more focused on games academic discipline, called ludology, Kreimer's work is focused on game design theory. His idea was to introduce a consistent and formal documentation of game design theories, insights and ideas (2002, p. 1). Kreimer called this a shared vocabulary of game design: "Game design needs a shared vocabulary to name the objects and structures we are creating and shaping, and a set of rules to express how these building blocks fit together." (2002, p. 1). The foundation for forming such a vocabulary for Kreimer (2002, p. 1) is the Pattern Language of architect and urban planner Christopher Alexander et. al. (1977).



What is a pattern?

Alexander's book "A Pattern Language" describes a language containing 253 patterns which follow a strict order from more general to more specific, from urban design to individual interior design details (Alexander et. al, 1977, p. x-xi). According to Alexander's work a design pattern in the essence is a description of a design problem and then a description of a solution of the problem (1977, p. x). Kreimer via Gamma et. al (2002, p. 2), Björk & Holopainen (2003a, p. 4) quote Alexander's definition of design pattern in their works: "Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of a solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice." (Alexander et al, 1977, p. x). The concept of a design pattern is so simple that it may seem as if there should be something else in it. However, it's nothing else, just a problem and solution. Design pattern structure is following: if problem A exists then solution B can be applied to solve it.

Often the term pattern is rephrased as a tool or a rule. For instance, Cermak-Sassenrath in his work says "Following Alexander, patterns are seen here as rules of thumb." (2012, p. 1). This rephrasing leads to problems with understanding of game design patterns. The reason why tools or rules are overlapping in meaning with patterns is that design patterns were introduced for game developers by Kreimer only in 2002. Before that there have been only game design rules and tools. In his article Kreimer says: "Alexandrian patterns are simple collections of reusable solutions to solve recurring problems. Doug Church's "Formal Abstract Design Tools" or Hal Barwood's "400 Design Rules" have the same objective: to establish a formal means of describing, sharing and expanding knowledge about game design." (Kreimer, 2002, 1). Kreimer's intention was to join several knowledge structures, like rules and tools into a single – pattern language. Kreimer says that patterns are a type of documentation organization in canonical format (2002, p. 11). His work is not about forming new theories, but about converting existing theories at that moment into patterns as it demonstrated on Figure 1. As a consequence, there still exists an understanding of a game design pattern as a rule or a tool which they were before.

Game designers, inspired by the ideas of pattern language, began forming a pattern language for game design. Extensive work was done documenting and formalizing theories and rules in game design using the concept of pattern language including Kreimer (2002), Barwood & Falstein (2002), Björk & Holopainen (2003a), Barney (2020a). There are a lot of works focused on forming patterns for specific game genres such as RPG (Kirk et. al, 2006), oldschool action games (Cermak-Sassenrath, 2012), multiplayer games (Olof, 2002), stealth games (Hu, 2014), cooperative games (Rocha et. al, 2008). Some works are describing patterns for specific fields such as sound design (Alves & Roque, 2010), level design for first person shooters (Hullett & Whitehead, 2010), level design for 2d games (Khalifa et. al, 2019),



game dialogs (Brusk & Björk, 2009), weapons (Giusti et. al, 2012), non-playable characters (Lankoski & Björk, 2007). Other works are focused on patterns in field of serious games and gamification (Ašeriškis & Damaševičius, 2014; Huynh-Kim-Bang et. al, 2010; Reichart & Bruegge, 2015).

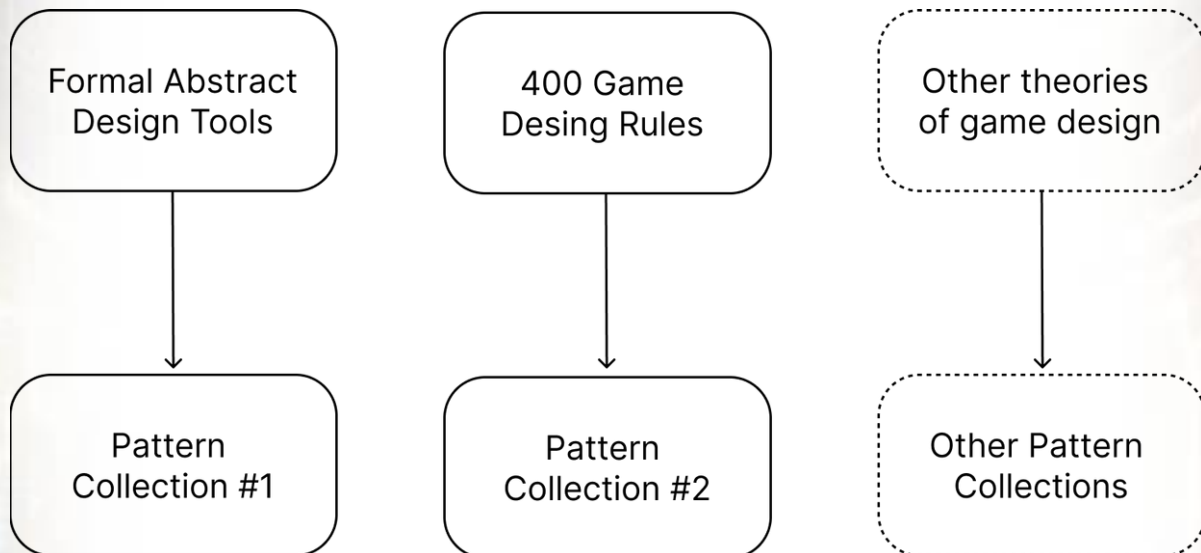


Figure 1. Kreimer's conversion from tools, rules and other game design theories into pattern collections

Criticism of game design patterns

Criticism of game design patterns demonstrates that despite the work done, it can be observed some problems that make game design patterns difficult to apply in practice or education process due to low formality level. Pattern application difficulty is criticised by Dormans (2013) and Barney (2021, p. 3). Moreover, Barney in his work states that novice designers who learn patterns are given too many tools (Barney, 2020a, p. xviii) that it becomes hard to navigate, organize and apply them. Additionally, some authors highlight that patterns belonging to the same collection can contradict or weaken each other, for instance Barney calls them subtractive patterns (2020, p. 7). Barwood says that some of his rules can also contradict each other (2001, p. 92, 105, 112), obviously those rules will continue contradicting in a shape of pattern. Finally, although the authors of collections refer to each other's work, they are developing their own pattern templates and conventions to describe patterns. According to the taxonomy of pattern collections by Liukkonen et. al (2015, p. 4), that leads to overlapping patterns, incompatible pattern templates and fragmentation of the field of game design patterns.

According to Alexander et al., "Each pattern can exist in the world only if it is supported by other patterns – larger ones, [...], as well as smaller patterns embedded in it" (1977, p. xiii). This way patterns form a network of connections



which could be called a pattern language. In Alexander's work each pattern references other patterns that complement it. Alexander insists that this is necessary for the user of the language to perceive the language as a whole. This explicit pattern relationship is omitted by Kreimer when he introduced pattern language to game designers. On pages 2-3 of Kreimer's work (2002) he suggests a pattern template without "related patterns" field while this field exists in each work he reference as an examples on page 3, i.e. Gamma et al. (1994), Meszaros & Doble (1997), Alexander et al (1977). The lack of "related patterns" field leads to game design patterns fragmentation. Different pattern collections don't reference each other and don't avoid overlapping patterns.

As it's outlined above pattern collections which have low continuity between each other. Each of those collections could contain patterns which are useful for some game project and designers would like to apply patterns from different collections to their game, meaning to use them as a single pattern language. Therefore, fragmentation of game design patterns prevents patterns complementing each other, which breaks the whole idea of pattern language.

However, the lack of "related patterns" field is not the only problem leading to fragmentation. Another problem is template inconsistency – every work is using its own conventions to describe patterns. Thus they can't be used as a single pattern language, but only as many small ones which have overlapping and contradicting patterns.

According to the criticism, issues that should be addressed for game design patterns field are as follows:

- What is the approach for navigation in pattern collection?
- What is the approach for contradiction resolution or avoidance between patterns?
- What pattern template should be used?

Despite the above criticism of fragmentation of game design patterns, there is a lack of works aimed at consolidation of game design patterns. The intention of this article is to contribute to closing this gap by examining pattern collections, identifying their weaknesses and revealing practices that would compensate for those weaknesses and become a foundation for patterns consolidation.

What is pattern consolidation?

Game design patterns consolidation is a process where different pattern collections are joined into a single with a consistent template, naming and collection structure, as it demonstrated on Figure 2. However, consolidation is hard due to several reasons: collections use different pattern templates, patterns in several collections could contradict or overlap with each other, united patterns collection



could become too large for navigation as it noted by Liukkonen et. al (2015, p. 4) and Björk & Holopainen (2003a, p. 10).

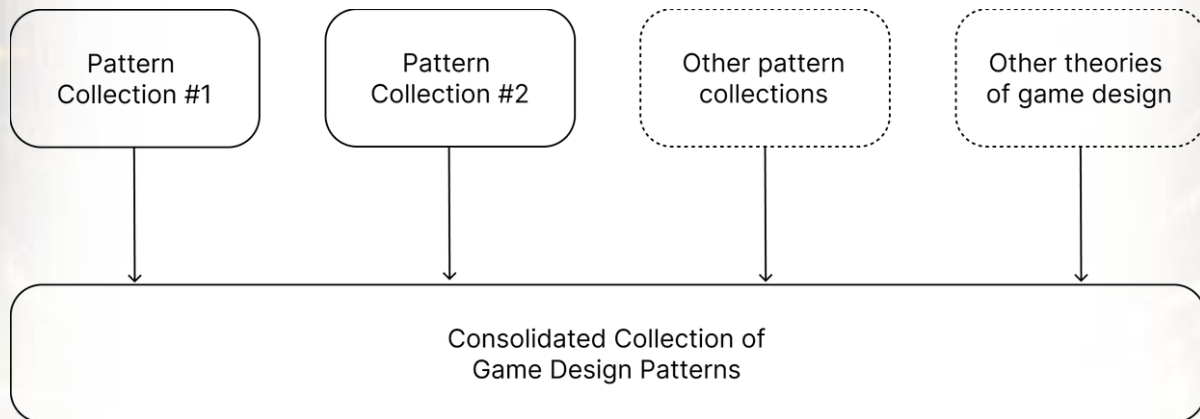


Figure 2. Consolidation of game design patterns

The article's intention is to step towards consolidation of game design patterns. It highly relies on Kreimer's work (2002). Kreimer's idea, to wrap game design theories into a shape of game design patterns, seems valuable since it introduces a single approach to documentation insight and best practices for general game design theory. In contrast to Kreimer's work, this article is aimed at consolidation of different, inconsistent collections of game design patterns into a single collection taking into account issues of navigation, contradiction and pattern template. This step seems important because without such a consolidation, the field of game design pattern will keep fragmented and hard to use. Consolidation should allow collections of game design patterns to behave like a single pattern language as it was supposed by Alexander et al. (1977, p. xiii). By its intention, a consolidated collection of game design patterns is just a shape to represent game design theories which should make it easier for novice designers or game design students to orient within different game design theories and their relations to each other. The goal of consolidated collection – to make it easier to find game design theories or to navigate across them.

Materials and method

Materials under consideration should demonstrate the evolution of game design practices collections which are relative to the field of game patterns. Since the article is addressing issues of consolidation, it is going to examine collections that behave like a shared vocabulary (in definition by Kreimer, 2002, p. 2) for professionals from different fields. Thus, game design patterns collection must include patterns related to game design in general, not only genre or field specific patterns. According to Liukkonen et. al. (2015, p. 3-4) taxonomy of game design pattern collections, "Game(Play) Design Patterns" are going to be studied, which are



The Case for Game Design Patterns (Kreimer, 2002), Game Design Patterns (Björk & Holopainen, 2003), Pattern Language for Game Design (Barney, 2020a). Thus, this research would not include pattern collections which are, in terms of Liukkonen et al., guidelines for intention, guidelines for implementation or game type specific. This article is criticising an absence of “related pattern” field in Kreimer’s template, which calls in doubt Kreimer’s results of converting Formal Abstract Design Tools (Church, 1999) and The 400 Project (Barwood & Falstein, 2002) into patterns. For this reason those works are included into analysis. Moreover, Jesse Schell’s book of lenses (2008) seems to have the same intention to form a “shared vocabulary”, but in the shape of lenses. Schell doesn’t describe why he is not using patterns formalism in his work, because his lenses seem to be easily converted into game design patterns. Schell work is also included into analysis because it seems very important and related to game design pattern’s field. Article’s materials selection seems to be pretty intuitive which is an obvious limitation of this work. However, the advantages of the intuitive selection seems to be higher, because it will demonstrate development of works which are related to game design patterns theory in game design since 1999. While Formal Abstract Design Tools and The 400 Projects inclusion is based on Kreimer’s intuition, Jesse Schell’s Book of Lenses inclusion is based on the authors of this article intuition about importance of Schell’s work for game design theory and a correspondence between Schell’s lens and pattern.

Therefore, comparative analysis would include following works: Formal Abstract Design Tools (Church, 1999), The Case for Game Design Patterns (Kreimer, 2002), The 400 Project (Barwood & Falstein, 2002), Game Design Patterns (Björk & Holopainen, 2003), Book of Lenses (Schell, 2008), Pattern Language for Game Design (Barney, 2020a). All of these works represent the concept of shared vocabulary and problem-solution couples for game designers and could become a basis for consolidated game design pattern language.

The method of the article is to conduct a comparative analysis of several game design pattern collections and highlight how different works could address criticism in order to find out the most suitable approaches addressing the highlighted issues.

The method of the research is comparative analysis which is based on “keyhole” comparison (Walk, 1998). Six sections have been selected to represent different aspects of the works:

- 1) **Template.** This section demonstrates a pattern template with a pattern example from the work, which helps the reader to understand a context of work under discussion. Moreover, the section provides us a way to compare different pattern templates.
- 2) **Game structure.** Game structure presents a set of elements that a game or a developing process can consist of. While designing a pattern, authors can describe which game elements or parts of a development process are



affected by the pattern. Game structure is usually represented as keywords or a fixed structure. A fixed game structure can be described in a work, and each pattern is related to the element from the described game structure. Keywords are usually not fixed, everyone can introduce keywords and each pattern can relate to several keywords if it's required.

- 3) **Navigation.** This section demonstrates which tools the collection provides for users to navigate across patterns.
- 4) **Contradiction.** This section is going to review how different works avoid or handle contradictions between patterns. Usually, when the authors suppose that patterns may contradict each other, they explicitly describe it in a special section.
- 5) **Cooperation.** This section provides an observation on how different works propose to expand the collection of patterns via cooperation with other contributors.
- 6) **Collection structure:** This section describes how the author of the specific work organizes pattern collection. There are two types of collections proper to game design patterns: list and network. List is just a plain set of patterns, while a network assumes relationships between patterns.

The comparison analysis is chosen as methodology because it can demonstrate one aspect of the collection affects others. For instance, how absence of “related pattern” field would affect collection structure, or how approach for navigation effects suggested game structure. To demonstrate the development and inheritance of concepts, works would be examined text-by-text, historically ordered in the results section.

The limitation of the chosen methodology is that it only allows to take a look at best practices within existing pattern collections, while probably it is worth trying to gather best practices and solutions from other fields using patterns concept, such as software engineering. Nevertheless, it seems logical to first analyze collections within game design theory and, if results would be insufficient, to move beyond this field.

Comparative analysis on game design pattern collections has been conducted before by Liukkonen et. al (2015). However Liukkonen et. al's work focuses on the taxonomy of game design patterns.

Results

Formal Abstract Design Tools (1999)

The first move in forming such a language for video games in this area is made by game designer and programmer Doug Church. He argues that game designers need a method to communicate with each other and collect knowledge about their observations, successful and unsuccessful decisions, available to everyone, regard-



less of the game genre they are developing (Church, 1999). Church suggested his own approach to solve these problems called Formal Abstract Design Tools (FADT).

Template: Church's tool consists of a name and a brief description.

“PERCEIVABLE CONSEQUENCE: A clear reaction from the game world to the action of the player.” (Church, 1999)

Game structure: “Abstract tools are not bricks to build a game out of. You don't build a house out of tools; you build it with tools.” (Church, 1999, no page). FADT does not aim to describe what elements a game can consist of; it aims to describe how some elements can interact with each other. Since Church doesn't have an equally formal abstract list of elements, FADT doesn't have a game structure to rely on.

Navigation: Since Doug Church describes only a few tools in his work, he does not address the issue of navigating these tools.

Contradiction: According to Church, some of the abstract tools may contradict each other. Church writes that game design tools should not be applied to every game but should be selected by the game designer depending on the game's vision, meaning the choice of tools depends on which game design elements are key (1999, no page). Thus, FADT doesn't have an embedded tool for pattern contradiction resolution.

Cooperation: As a source for finding tools, Church suggests researching and analyzing video games (1999, no page). However, he does not discuss how different researchers can combine their findings.

Collection structure: Tools are similar to patterns, however they omit a network structure of pattern language. They are using list structure instead.

The 400 Project (2001-2006)

In 2002, at the GDC, Noah Falstein with Hal Barwood announced the intention to compile a list of 400 game design rules. Initially, Noah Falstein stated that he did not know how many game design rules actually exist, and the number 400 was taken arbitrarily (Barwood, 2001, p. 17). Over five years, the authors managed to collect 112 rules (Barwood & Falstein, The 400 Project, n.d.).

Despite Church using the word “tool” and Falstein and Barwood using the word “rule”, these are similar concepts. The description of The 400 Project mentions that rules are tools (2002, p. 32), and the principles of using and properties of rules and tools are identical.

Template: A rule in The 400 Project consists of an identifier, title, brief description, the domain to which the rule applies, and the rule's contributor.

“ID: 3

Imperative Statement: Immersion is easily disturbed -- don't make the player re-calibrate his “suspension of disbelief” and lose touch with your game

Explanation in 250 words or less: Immersion is easily disturbed -- don't make the player re-calibrate his “suspension of disbelief” and lose touch with your game



Domain: Psych

Contributors: Hal Barwood” (Barwood & Falstein, The 400 Project. Rules List, n.d.)

Game structure: The rules are divided into domains: basic, psychology, simplicity, flow, variability, meta, balance, story, localization, and others. It is worth noting that many rules remained without domains, while others belong to several domains. Essentially, a domain is the area to which the rule applies which makes domains similar to keywords.

Navigation: Initially, by adding domains, the authors aimed to facilitate quick navigation through the rules: if it’s needed to find rules related to the game’s world story, users can look among the rules in the Story domain.

Contradictions: In their GDC presentation, they mention that game design ideas are like language rules: they can be broken, contain exceptions, and completely contradict each other (2001, p. 92). However, The 400 project doesn’t have tools to resolve pattern contradictions.

Cooperation: Falstein and Barwood created conditions for anyone to contribute to the project. In 2002, they announced that rules could be sent to them by email. As a result, developers and video game researchers like Jesse Schell, Raph Koster, Warren Spector, Bernd Kreimeier, and others participated in compiling the list of rules (Barwood & Falstein, The 400 Project. Rules List, n.d.). Thus, Falstein and Barwood propose using collaborative knowledge to collect rules.

Collection structure: Unlike Alexander’s patterns, organized as a network, Falstein and Barwood organize their collection as a list.

The Case for Game Design Patterns (2002)

Game programmer and designer Bernd Kreimeier published his text “The case for game design patterns” on gamasutra.com in 2002. Kreimer’s work contains 7 patterns.

Template: The pattern template by Kreimer consists of Name, Problem, Solution, Consequences and sometimes References.

“PAPER-ROCK-SCISSORS

Problem: Avoid a dominant strategy that makes player decisions a trivial choice.

Solution: Introduce nontransitive relationships within a set of alternatives, as in the game of paper-rock- scissors.

Consequence: The player is no longer able to find a single strategy that will be optimal in all situations and under all circumstances. She has to revisit her decisions, and, depending on the constraints imposed by the game, adjust to changing situations, or suffer the consequences of an earlier decision.

Examples: The example given by Andrew Rollings is the set of warrior-barbarian-archer from the Dave and Barry Murray game The Ancient Art of War. He also describes Quake’s weapon/monster relations in similar terms: Nailgun beats shambler, shambler beats rocket launcher, rocket launcher beats zombie, zombie beats nailgun.



References: Chris Crawford provided the first explicit description of the use of nontransitive relationships. Andrew Rollings' discussion of examples uses game theory including detailed payoff, as well as informal fictional designer dialogs." (Kreimer, 2002, p. 6)

Game structure: Kreimer does not discuss game structure.

Navigation: Kreimer says that game design pattern collection should be a kind of documentation which is navigable via special software (2002, p. 11). He compared game design pattern collection to a scriptwriting document and outlined that important thing is a formal standard for describing patterns and software that supports editing and search facilities within game design pattern collection. Therefore, Kreimer believes that the solution for the navigation issue is a software that can perform search over pattern collection.

Contradiction: Kreimer claims that designers could have different opinions of pattern's quality and it's not a problem. As he says: "One person's pattern might well be another's anti-pattern" (2002, p. 10). Then Kreimer says that pattern descriptions should contain only a cause and effect (2002, p. 10). And disagreements on results should be outside of a pattern's scope.

Cooperation: Kreimer states that pattern candidates can be harvested from such sources as lectures, postmortems, discussions (2002, p. 10). The author demonstrates as an example how FADTs can be expressed as patterns (p. 5). Also Kreimer supports collaboration via using an optional section called references to put there information about pattern sources.

Collection structure: Kreimer's collection organized as a list.

Game Design Patterns (2003)

In 2003, Staffan Björk and Jussi Holopainen published two paired articles, one dedicated to game design patterns (2003a), and the other to an Interaction-Centric Structural Framework (2003b). Björk and Holopainen have compiled a collection of 296 game design patterns (Björk & Holopainen, 2024).

Template: The pattern template by Björk and Holopainen consists of the following fields: name, description, consequence, example of application, relationship to other patterns.

"Excluding Goals

Completing an Excluding Goal makes the completion of other goals in the game meaningless or impossible.

Example: Star Wars: Knights of the Old Republic is a single player adventure game, where finishing certain goals exclude other possible goals as the player character moves towards being good or evil.

Instantiates: Conflict, Competition, Incompatible Goals, Closure Points Modulates: Hierarchy of Goals

Instantiated by: Overcome, Race, Preventing Goals, Tiebreakers Modulated by: Selectable Sets of Goals, Dynamic Goal Characteristics Potentially conflicting with: Tied Results, Shared Rewards" (Björk & Holopainen, 2024, p. 23)



Game structure: Björk and Holopainen describe not only game design patterns but also a structural model of video games (2003b). They insist that they can be used separately or jointly. From their perspective, patterns form some connections between elements of the structural framework. Björk and Holopainen work consists of two layers: game design patterns and structural framework.

“The difference between the two is that the design patterns are higher level, hierarchical conceptual models of the possible designs of games which manifest themselves in the structural framework by defining the logical, physical and temporal relations of the elements in the framework.” (Björk & Holopainen, 2003b, p. 2).

Thus, the authors see the possibility of relating patterns and the structural framework while structural framework represents fixed game structure suggested by the authors. Structural framework is a fixed game structure suggested by Björk and Holopainen.

Navigation: According to the authors, the task of names is not to provide an intuitive understanding of what pattern is hidden under it but to provide a mnemonic hint after the pattern has been read (Björk & Holopainen, 2003a, p. 10). This decision is dictated by the problem the authors emphasize: a large number of patterns make it difficult to navigate them, and finding the necessary pattern takes too much time. “With over 200 pattern candidates identified, we have already identified the problem of finding the relevant patterns for any given situation” (Björk & Holopainen, 2003a, p. 10).

Contradiction: In the pattern template it can be seen that the authors are providing a “potentially conflicting” section which describes patterns that could lead to contradiction. The section allows designers to explicitly analyze possible contradictions and avoid them if it’s needed. Thus Björk and Holopainen are suggesting an approach for pattern contradictions resolution.

Cooperation: Björk and Holopainen identified the problem of validating patterns during collection gathering (2003, p. 9). Which patterns are less applicable, which are widely applicable, who should determine the importance of a pattern, and how to avoid subjective decisions. The authors believe that the importance of patterns can only be determined in practice. For this, they publish the patterns online, to put them in use. However, they do not propose an explicit method for assessing the value of a pattern or cooperation between designers.

Collection structure: Instead of organizing patterns as a list, the authors followed Alexander’s approach by organizing them into a network. Each pattern, except for a few base ones, relies on other patterns, complementing and specifying them. “The structure of the pattern collection is not a strict hierarchy but a network with several base nodes.” (Björk & Holopainen, 2003a, p. 9)

An important step in their work was the appearance of a structural framework supporting the patterns and organizing patterns into a network where one pattern can refer to others.



Book of Lenses (2008)

In 2008, game designer Jesse Schell published a book on game design where he describes 113 lenses of game design (Schell, 2008).

Template: Lens template consists of identifier, title, description and set of questions.

“Lens #2: The Lens of Surprise

Surprise is so basic that we can easily forget about it. Use this lens to remind yourself to fill your game with interesting surprises. Ask yourself these questions: What will surprise players when they play my game? Does the story in my game have surprises? Do the game rules? Does the artwork? The technology?” (Schell, 2008, p. 26)

Game structure: As a game structure Schell offers a map (2008, p. 463). “For this book, I have created a map that shows the web of game design relationships” (Schell, 2008, p. xxvii). The map is closely tied to the lenses and shows the connections between them. Each element of the map corresponds to a section of the book and the lenses described in it and demonstrates the game's structure and its elements. The map is a fixed game structure suggested by Schell.

Navigation: Schell suggests that readers purchase a deck of cards with the game design lenses to work with them more easily (p. xxvi). A physical deck of cards with lenses allows the user to lay them out on a surface, sort them into categories, simplifying navigation. Of course this will only work after reading the book, as the reader needs to understand the context, but it is a clear method that can help in working with the lenses.

Contradiction: Schell states that each lens is a perspective on the game (p. xxvi). The lens is a short set of questions the game designer should answer to look at their game from a certain perspective. Schell insists that lenses are not rules or recipes but tools that allow users to check their design. This is what distinguishes this approach; previous authors described a solution to a problem, whereas Schell does not offer a solution but proposes thinking about whether there is a problem at all. It is important to note that by refraining from offering solutions within the lens, Schell solves the problem of contradictory patterns. By refusing affirmative judgments and shifting the dialogue about game design into questions, Schell places the responsibility for the final decision on the game designer. This allows lenses to avoid contradictions between each other, however it partly breaks a concept of pattern that should describe a problem and a possible solution.

Cooperation: Schell does not describe specific methods for expanding the collection of lenses.

Collection structure: Schell's lenses are organized into a list with a rare declaration of connections/relations between lenses if the connection between lenses is crucial, such as in lens #11 (p. 60).

Pattern Language for Game Design (2020)

In 2020, game designer Chris Barney published a book on game design patterns where he described his original approach (Barney, 2020a). Barney



supposed that every designer should create his own pattern language observing games and suggested exercises that help students to extract patterns from their play experience. Additionally, Barney launched a website with a pattern library containing more than 450 patterns (Barney, 2020a).

Template: Barney's pattern template consists of title, image, authors, groups, confidence, creation and editing dates, seed, keywords, design problem, description, example games, related patterns, suggested exercises (Barney, 2021, p. 5-7).

Example pattern "We're Going to a Dark Place Together" (Barney, n.d.) is available online.

Game structure: Barney doesn't directly touch upon game structure in his works. However, according to Barney's template each pattern has a set of keywords. Keywords are forming a game structure for Barney's design patterns as domains do the same in The 400 Project.

Navigation: To resolve the navigation issue Barney has launched a website with pattern search facilities, as it was suggested by Kreimer (2002). On the website patterns can be searched and filtered by authors, pattern name, keywords or group. Site behaves like a hypertext for patterns: users can easily jump to related patterns, patterns with the same keyword, author or group. Additionally, the pattern template contains an image which helps users to memorize the pattern.

Contradiction: There are four pattern relation types described by Barney: parent, child, additive, subtractive (Barney, 2020b). According to Barney, subtractive patterns are those which mitigate another pattern effect. In other words subtractive patterns are patterns which contradict each other. This is similar to what Björk and Holopainen call conflicting patterns.

Cooperation: Barney's website also supports cooperation since every person can participate and write his own patterns. Moreover, every user can compile his own pattern language from existing patterns to use it in his company or personally. Compiled pattern languages can be private or public. A user creating a pattern can omit such fields as example games, related patterns, an image and suggested exercises from the pattern template.

Every user-suggested pattern should pass the review process before becoming visible to other users. That partly addresses the problem of pattern validation touched upon by Björk and Holopainen. Additionally, inheriting Alexander's concept (Alexander et. al, 1977, p. x), every pattern has a confidence field that demonstrates how much a user trusts in the suggested pattern. That also partly addresses the validation issue.

**Table 1. Results of comparison**

	Template	Game Structure	Navigation	Contradiction	Cooperation	Collection Structure
FADT	Title, Description	—	—	—	—	List
The 400 Project	ID, Title, Description, Domain, Contributor	Keywords (Domains)	Domains	—	Via e-mail	List
The Case for game design patterns	Title, Problem, Solution, Consequence, References.	—	Via search facilities	—	Different pattern collections can reference each other	List
Game Design Patterns	Title, Description, Consequence, Example, Children, Parents, Conflict	Fixed (Interaction-Centric Structural framework)	Mnemonic hint	Supported via “conflict” field	—	Network
Book of Lenses	ID, Title, Description, Questions	Fixed (Map)	Deck of cards	Supported via omitting problem solution couple	—	List
Pattern Language for Game Design	Title, Image, Authors, Groups, Confidence, Keywords, Problem, Description, Example, Related	Keywords	Via search facilities: by author, keywords, group or title.	Supported via “subtractive relation” field	Via public website	Network

Discussion

The section is going to analyze and discuss the results given in table 1 to address the issues of consolidation.

What is the approach for navigation in consolidated pattern collection?

According to table 1, some approach for navigation is offered in each collection except FADT (Church, 1999), most of them doesn't seem to be reliable in consolidated collection. The Case For Game Design Patterns (Kreimer, 2002) only offers an idea of software, that should help with navigation, however he is not describing,



what exactly search could be based on, so it's only an idea, without practical proposal. Bjork & Holopainen's (2003a) suggested a mnemonic hint within a name of pattern, however it doesn't seem to be a reliable approach due to number of expected patterns, and furthermore they do not describe how exactly the hint works. Schell's deck of physical cards (2008) could work only with limited collection which is not extend over time, otherwise users would be needed to print or buy additional cards regularly. Schell does not suppose his collection to be extended, since his collection doesn't support cooperation (Table 1), thus his deck of cards approach could work for him, but it does for consolidated patterns collection which is supposed to be extended over time. Keywords game structure are proper to The 400 Project and Pattern Language For Game Design. As was mentioned in the results section, there is no valuable difference between concepts of keywords and domains. Both of them follow an intention to shortly describe content of the pattern and parts of a game the attributed pattern could be applied to. Only these two collections have a wide amount of patterns and support cooperation at the same time. Moreover, those two collections propose the same approach for navigation which is based on keywords (or domains).

According to those observations, the keywords approach is the most suitable to structure the game that helps for navigation.

What is the approach for contradiction resolution or avoidance between patterns?

According to table 1, there were two approaches to deal with pattern contradictions. Book of Lenses (Schell, 2008) approach for contradiction violates the concept of pattern described by Kreimer (2002), because Schell's contradiction avoidance excludes clear problem-solution pairing. Another approach is provided by Game Design Patterns (2003) and Pattern Language For Game Design (2020): the field that explicitly says which patterns can conflict or weaken each other. It is a clear and obvious approach for the final user. The reason for "contradicting patterns" field appearance in Game Design Patterns and Pattern Language for Game Design is that those works use network collection structure, not a list. Network collection structure requires patterns to reference related ones. And contradiction is just a type of relation between patterns. Therefore, "related patterns" field should be used for describing conflicting and complementing patterns.

What pattern template should be used in a consolidated collection?

The first observation about pattern template is that all of the reviewed works contain fields title and description. Kreimer's work doesn't have a field called description, however Kreimer's description field is intentionally splitted into problem and solution fields to express a problem oriented thinking (2002, p. 2-3). It can be noticed that the widest template introduced by Barney keeps an opportunity to omit some fields if they are not necessary. That means that a consolidated collection's pattern template could have optional fields. Optionals fields would help



to join different collections, while the main goal is to define mandatory fields for each pattern.

According to the results summarized in table 1, each pattern collection structured as a network contains fields about pattern relationships, such as “child”, “parent”, “conflicting”, “related”. Following Alexander's concept of pattern language, a network structure should be followed so that patterns support each other.

The 400 Project, The Case For Game Design Patterns and Pattern Language For Game Design are works that support cooperation and their templates have an author field called contributor, references and authors accordingly. Since a consolidated collection concept highly relies on cooperation this field should be mandatory.

This way a mandatory set of fields for pattern template can be proposed.

Name: Shortly describes the content of the pattern.

Keywords: Describes elements of the game related to that pattern. Used for navigation.

Description: Describes a problem, solution and possible consequence in a free form.

Related patterns: Contradicting and complementing patterns.

Reference: This field references the work or the author who suggested the pattern.

This pattern template contains all important fields for consolidated game design pattern collection. Every pattern can have any number of additional fields such as image, confidence, example and so on. Thanks to additional fields every pattern from any collection can be adapted to the provided template without reduction.

To demonstrate the power of the suggested pattern template, the pattern example provided.

Name:

Predictable Consequence

Keywords:

Feedback, Mental-Model, Psych, Meta

Description:

Problem is players can't understand why something is happening in the game world. For instance, why they failed, why they can't climb, why they have been caught and so on. This breaks a player's planning and understanding of the game world.

Solution is making any game world reaction to player action homogeneous and predictable. Homogeneous means that under the same conditions game world reactions to player action should be the same. And every condition that can affect the game world's reaction should be noticeable and clearly perceived by the player.

Related Patterns:



Since *Predictable Consequence* is required for planning, it is complementing to *Intention* pattern. *Predictable Consequence* reinforces *Action* and is supported by *Rules*.

Contradicting pattern to Predictable Consequence is *Surprise* since surprise is something unpredictable. *Indirect-Control* pattern can break *Predictable Consequence*.

References:

PREDICTABLE CONSEQUENCE (Church, 1999), *Predictable consequence* (Kreimer, 2002, p. 4-5), *Lens #24: The Lens of Action* and *Lens #26: The Lens of Rules* (Shell, 2008, p. 144, 150), *Rule 96: Make Consequences of Actions Predictable* and *Rule 79: Preserve Cause and Effect* (Barwood & Falstein, The 400 Project, n.d.), *Predictable Consequence* (Björk & Holopainen, 2024).

Examples:

The actions in Chess and Go have totally Predictable Consequences, and skillful playing of these games consists on being able to predict opponents' actions and planning many actions ahead. (Björk & Holopainen, 2024)

As it can be noticed, the pattern adapted to the suggested template references five of six reviewed pattern collections. Nevertheless, pattern fits into the suggested template and absorbs properties from different pattern collections. Name is from the original Church's work. Keywords from The 400 Project domains. Related patterns section is based on Game Design Patterns. Examples is an optional section which is not required, but allows us to put additional information from Game Design Patterns. References section gives a user of the pattern an opportunity to search other pattern collections for any other required information. This pattern resolves overlapping between collections and can be added to the consolidated collection.

Conclusion

As a still growing industry, game design is looking for the best ways to share knowledge and best practices and patterns is one of those approaches. However, to become a sustainable game design patterns should address criticism. The article compares and analyses the works on game design pattern language and tries to find solutions for the weaknesses in these works.

Comparative analysis was conducted to find the most reliable approaches to meet criticism. The comparison identified that keywords are a good way to solve navigation issues, related patterns tracking is a good way to solve contradiction issues and network structure is most suitable for consolidated pattern collection. Moreover, the article proposes a pattern template for consolidated collection which can express patterns from other pattern collections and suggested that each pattern could have any number of additional fields to be more precise and detailed than the provided template. There're still some not answered questions such as what is the best platform for consolidated collection and how to validate patterns. However those questions are outside of this article's scope and should become a part of future research.



In order to understand whether the result satisfies industry and academia, consolidated pattern collection should be offered to them. The consolidated collection should be more compact than individual collections of game design patterns in sum, so it could be offered to students in game design classes at universities. Whether game design patterns are good for students is not a question. Patterns are just a shape, while their content is actual theory of game design. The real question is whether or not it's easier for students and novice designers to examine a game design theory in the shape of pattern language.

References | Список литературы

- Alexander, C., Ishikawa, S., & Silverstein, M. (1977). *A pattern language: Towns, buildings, construction*. Oxford University Press.
- Alves, V., & Roque, L. (2010). A pattern language for sound design in games. *Proceedings of the 5th Audio Mostly Conference: A Conference on Interaction with Sound*, 1–8. <https://doi.org/10.1145/1859799.1859811>
- Ašeriškis, D., & Damaševičius, R. (2014). Gamification Patterns for Gamification Applications. *Procedia Computer Science*, 39, 83–90. <https://doi.org/10.1016/j.procs.2014.11.013>
- Barney, C. A. (n.d.). Pattern “We’re Going to a Dark Place Together.” Pattern Language for Game. <https://patternlanguageforgamedesign.com/PatternLibraryApp/PatternLibrary/230>
- Barney, C. A. (2020a). *How to create a pattern*. Medium. <https://perspectivesingamedesign.com/how-to-create-a-pattern-43742ad024f9>
- Barney, C. A. (2020b). *Pattern language for game design*. Pattern Language for Game. <https://patternlanguageforgamedesign.com/>
- Barney, C. A. (2021). Application of Pattern Language for Game Design in Pedagogy and Design Practice. *Information*, 12(10), 393. <https://doi.org/10.3390/info12100393>
- Barwood, H. (2001). Four of the four hundred 2001. *GDC Lecture*. GDC lecture. https://www.finitearts.com/Storage/GDC01_4x400_web.zip
- Barwood, H., & Falstein, N. (n.d.-a). *The 400 project*. Finite Arts. <https://www.finitearts.com/Pages/400page.html>
- Barwood, H., & Falstein, N. (n.d.-b). *The 400 project*. Rules list. Finite Arts. https://www.finitearts.com/Storage/400_Project_Master_Rule_List_032306.zip
- Barwood, H., & Falstein, N. (2002). More of the 400: Discovering design rules. *Lecture at Game Developers Conference*. Lecture at Game Developers Conference. https://www.finitearts.com/Storage/GDC02_More400_web.zip
- Björk, S., & Holopainen, J. (2003, January 1). Describing Games: An Interaction-Centric Structural Framework. *Proceedings of DiGRA 2003 Conference: Level Up*. Proceedings of DiGRA 2003 Conference: Level Up. <https://doi.org/10.26503/dl.v2003i1.87>
- Björk, S., & Holopainen, J. (2024). *List of game design patterns*. ResearchGate. https://www.researchgate.net/publication/379684508_gamedesignpatterns.pdf



- Björk, S., Lundgren, S., & Holopainen, J. (2003, January 1). *Game Design Patterns*. *Proceedings of DiGRA 2003 Conference: Level Up*. *Proceedings of DiGRA 2003 Conference: Level Up*. <https://doi.org/10.26503/dl.v2003i1.60>
- Brusk, J., & Björk, S. (2009, January 1). *Gameplay Design Patterns for Game Dialogues*. *Proceedings of DiGRA 2009 Conference: Breaking New Ground: Innovation in Games, Play, Practice and Theory*. *Proceedings of DiGRA 2009 Conference: Breaking New Ground: Innovation in Games, Play, Practice and Theory*. <https://doi.org/10.26503/dl.v2009i1.393>
- Cermak-Sassenrath, D. (2012). *Experiences with design patterns for oldschool action games*. *Proceedings of The 8th Australasian Conference on Interactive Entertainment: Playing the System*, 1–9. <https://doi.org/10.1145/2336727.2336741>
- Church, D. (1999). *Formal Abstract Design Tools*. Game Developer. <https://www.gamedeveloper.com/design/formal-abstract-design-tools>
- Crawford, C. (1984). *The art of computer game design*. Digital Press. https://www.digitpress.com/library/books/book_art_of_computer_game_design.pdf
- Dormans, J. (2013). *Making design patterns work*. *Proceedings of the Second Workshop on Design Patterns in Games, DPG '13, in Association with Foundations of Digital Games, FDG'13*. *Proceedings of the Second Workshop on Design Patterns in Games, DPG '13, in association with Foundations of Digital Games, FDG'13*. <https://automatedgamedesign.wordpress.com/wp-content/uploads/2015/10/dormans-2013-making-design-patterns-work-dormans.pdf>
- Frasca, G. (1999). *Ludology meets narratology: Similitude and differences between (video) games and narrative*. *Ludology*. <https://ludology.typepad.com/weblog/articles/ludology.htm>
- Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). *Design patterns: Elements of reusable object-oriented software*. Addison-Wesley.
- Giusti, R., Hullett, K., & Whitehead, J. (2012). *Weapon design patterns in shooter games*. *Proceedings of the First Workshop on Design Patterns in Games*, 1–7. <https://doi.org/10.1145/2427116.2427119>
- Hu, M. (2014). *Game design patterns for designing stealth computer games* [Master Thesis]. Malmö högskola.
- Hullett, K., & Whitehead, J. (2010). *Design patterns in FPS levels*. *Proceedings of the Fifth International Conference on the Foundations of Digital Games*, 78–85. <https://doi.org/10.1145/1822348.1822359>
- Huynh-Kim-Bang, B., Wisdom, J., & Labat, J. M. (2010). *Design patterns in serious games: A blueprint for combining fun and learning*. Project SE-SG. <http://seriousgames.lip6.fr/DesignPatterns>
- Khalifa, A., De Mesentier Silva, F., & Togelius, J. (2019). *Level Design Patterns in 2D Games*. *2019 IEEE Conference on Games (CoG)*, 1–8. <https://doi.org/10.1109/CIG.2019.8847953>
- Kirk III, W. J., Cantrell, M. R., & Holmes, M. (2006). *Design patterns of successful role-playing games*.
- Koster, R. (2004). *A theory of fun for game design*. Paraglyph Press.
- Kreimer, B. (2002). *The Case For Game Design Patterns*. Game Developer. <https://www.gamedeveloper.com/design/the-case-for-game-design-patterns>
- Lankoski, P., & Björk, S. (2007, January 1). *Gameplay Design Patterns for Believable Non-Player Characters*. *Proceedings of DiGRA 2007 Conference: Situated Play*. *Proceedings of DiGRA 2007 Conference: Situated Play*. <https://doi.org/10.26503/dl.v2007i1.262>



- Liukkonen, T. N., Heimo, O. I., Mäkilä, T., Smed, J., Bakke, S., & Nack, F. (2015). *Comparative analysis on game design pattern collections*. 978-90-77381-91-5, 10–17.
https://tt.utu.fi/ar/wp-content/uploads/2016/09/Comparative-Analysis-on-Game-Design-Pattern-Collections-final_typokorjaus.pdf
- Meszaros, G., & Doble, J. (1997). A pattern language for pattern writing. In R. Martin, D. Riehle, & F. Buschmann (Eds.), *Pattern languages of program design 3* (pp. 529–574). Addison-Wesley.
- Olof, E. (2002). *Multiplayer design patterns*. ABC-Klubben.
https://www.abc.se/~m10383/Haven/General/Multiplayer_Design_Patterns.html
- Reichart, B., & Bruegge, B. (2015). *Serious game patterns for social interactions* (K. Blashki & Y. Xiao, Eds.; pp. 21–24). Las Palmas de Gran Canaria, Spain.
- Rocha, J. B., Mascarenhas, S., & Prada, R. (2008). *Game mechanics for cooperative games*. 72–80.
- Schell, J. (2008). *The art of game design: A book of lenses*. CRC Press.
- Walk, K. (1998). *How to write a comparative analysis*. Writing Center at Harvard University.