

Game Design Patterns Core Concepts

Chris A Barney
Visiting Assistant Professor
Northeastern University

Art by Jason Weiser

patternlanguageforgamedesign.com

Takeaways

1. **Game Design Patterns** and why would you want them
2. Patterns are connected into a **Pattern Language**
3. You create your own patterns to **solve your design problems**
4. Patterns help you **communicate your design ideas** to others
5. You can **use patterns to design**
6. Patterns help you **coordinate design** in your studio
7. Patterns can **impact the bottom line**

Who am I?

Game Design Student

Created and earned two game design degrees

Industry Veteran

Poptropica.com, FunBrain.com, Indie Board Games and Larp

Game Design Professor

Northeastern University: GAME/GSND

Speaker

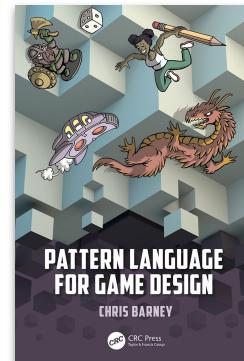
ECGC, GDC, GDC Eu, DevCom, Pax/PaxDev, GaymerX, BFIG

Author

Pattern Language for Game Design, Perspectives in Game Design, Interactive Theater



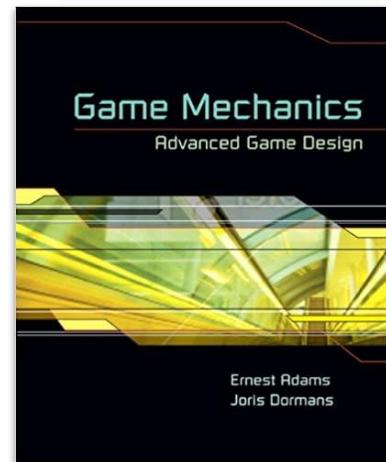
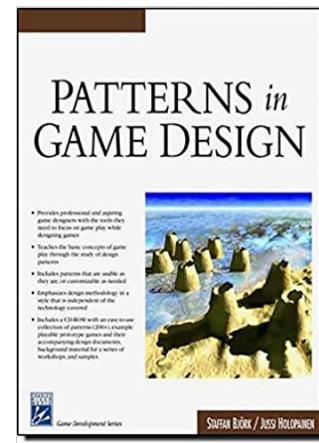
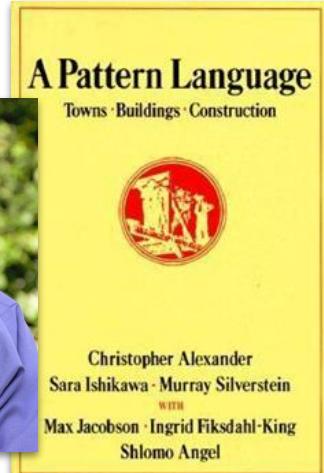
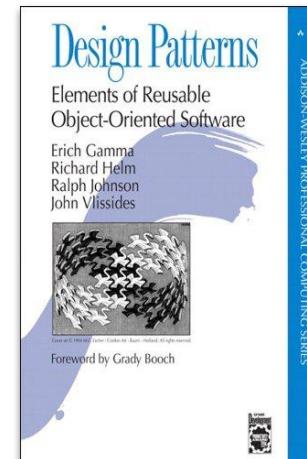
Northeastern University
College of Arts, Media and Design



What is a Pattern

"Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use the solution a million times over, without ever doing it the same way twice."

- Christopher Alexander



What is a Pattern in Game Design? (*According to me!*)

Design Problem

Solution (Pattern Description)

Example Games

Keywords

Related Patterns

Confidence

The Three Pillars of Meaning in Emergent Narrative



Author(s): Chris Barney

Groups(s): Pattern Language for Game Design

Pattern: 2

Confidence: November 17th 2020, 9:03am

Created On: November 19th 2020, 12:48pm

Edited On:

Seed: Exercise 11: Emergent Narrative Patterns — What makes emergent events narratively meaningful?

Keywords:

- Autonomy
- Choice
- Costs
- Difficulty
- Economy
- Emergent Narrative
- Meaning
- Writing/Narrative
- Deep Interlock
- Not Separateness

When developers provide context, motivation, and consequence players can construct meaningful narratives.

Design Problem: Given the reality of limited resources, when creating design elements that encourage emergent narratives, designers need to maximize the narrative potential of every element. How can designers know if a given game element will contribute to meaningful emergent narratives?

Pattern Description: To allow players to construct meaningful emergent narratives, developers should provide players with context, motivation, and consequence for their actions in the game. There are many child patterns that contribute to this pattern; two are listed and 12 other possibilities are provided in the suggested exercise, and there may be more. But each one contributes to either the context of, motivation for, or consequences of an event that could be part of an emergent narrative.

How Do You Make A Pattern

General Patterns Exercise

1. Name a design element.
2. Name (*at least*) 10 games that use that element.
3. Describe how each of those games uses the element you chose.
4. Describe the design problems the games use the element to solve.
5. What are the Patterns in the ways the elements are used that relate to the problems they solve?
6. Pick one of those patterns and describe it. (*on the Pattern Library website*)
7. You may repeat step 6 for each pattern you observed.

Pattern Library Website:

Editing: And Now I Guess We Are Doing This

Pattern Name	And Now I Guess We Are Doing This
Design Problem	Players can get trapped in comfortable gameplay loops, which can both bore them and keep them from taking advantage of all the gameplay possibilities of your game.
Pattern Description	To get players to transition from one gameplay type to another, developers can take advantage of players' tendencies to push their characters' limits and abilities. When you build in lethal consequences to overextending character abilities, you create a situation where the player has to change their way of playing to survive.
Pattern Image	https://patternlanguageforgamedesign.com/PatternLibrary/Images/17_IGuessNowWeAreDoingThis.png
Pattern Image Description	Unexpected events can force players to transition between gameplay styles, in this case between stealth and c This will be used as alt text and caption.
Pattern Confidence	2
Author(s)	Chris Barney X
Group(s)	Pattern Language for Game Design X
Pattern Seed	Select... One of These Days That's Going to Get You Killed
Keywords - Categories - Properties	Abilities X Consequences X Mechanics X Player Motivation X Alternating Repetition X Contrast X Deep Interlock X
Example Games	Sekiro: Shadows Die Twice In order to take advantage of stealth kills the player has to get close to the enemy. If they fail to kill the enemy it is alerted and the gameplay changes from stealth to combat action, or to traversal as the player flees. Anthem The character has very powerful abilities and weapons, but the abilities have cooldowns and the weapons can overheat and can't be fired until they cooldown. Gameplay is frenetic so the player is commonly in situations where they have failed to watch their heat gauge or their abilities are on

Editing: Alan Wake

Alan Wake
Alan Wake is a horror game, which uses dark environments with shadows and a flashlight to create a scary experience for the player. The story follows the protagonist who is a novelist known as Alan Wake, who tries to solve the mysterious disappearance of his wife near the fictional town of Bright Falls.
https://upload.wikimedia.org/wikipedia/en/1/1f/Alan_Wake_Game_Cover.jpg
https://www.youtube.com/watch?v=sSB4QcQMm6E
https://www.youtube.com/watch?v=0RpTykaCVNQ
Steam X
https://store.steampowered.com/app/108710/Alan_Wake/
Notes on the game source.
Add Available Link
Add Info Link
Primary X
Date picker inline 05/14/2010 Calendar icon
Alan Wake
Notes on the game release.
Add Release
Remedy Entertainment X
Add Developer to Game
Enter New Developer

General Patterns Exercise

1. Name a design element

Jumping

Functional design element, for the purpose of this example I am considering jumping in relation to the player controlled character jumping.



General Patterns Exercise

2. Name (at least) 10 games that use that element

- Donkey Kong/Jumpman
- Q-Bert
- Super Mario Bros.
- (Braid as subversion?)
- Mirror's Edge
- Gravity Rush/VVVVVVVV
- Alice / Super Mario World / Crackdown
- Guild Wars 2
- Tomb Raider (Reboot)
- Prince of Persia (2008)
- Poptropica
- Super Meat Boy
- StreetFighter/Soul Caliber/Devil May Cry
- Doom/Quake/Splosion Man
- Tribes
- Assassin's Creed
- Canabalt
- Sonic
- Trials HD



General Patterns Exercise

3. Describe how each of those games uses the element you chose.

- **Donkey Kong / Jumpman, Geometry Dash** - Jumping is used to avoid enemies and traverse the 2d space.
- **Q-Bert** - This game is Pac-Man-like in that it is a reflex-based puzzle game. It uses jumping as its only movement mechanic.
- **Super Mario Bros.** - Jumping is used to avoid enemies, traverse 2d/3d space, and as a way to attack enemies.
- **Mirror's Edge** - First-person jumping as pure traversal
- **Gravity Rush / VVVVVVV** - Jumping with control of physics
- **Alice/Super Mario World /Crackdown** - Jumping with a glide. Also, in-air control?
- **Guild Wars 2** - Jumping for exploration and as a puzzle. Little need in world traversal, none in combat.
- **Tomb Raider (Reboot)** - Jumping as a puzzle mechanic.
- **Prince of Persia (2nd reboot)** - Assisted jumping. The game is single-player so that maybe just a double jump
- **Doom/Quake/Splosion Man/Tribes** - Jumping for world traversal. Jumping assisted by the physics of unrelated systems (Rocket Jumping, Ski Jumping, Bunny Hopping).
- **StreetFighter/Soul Calibur/Devil May Cry** - Jumping for world traversal, jumping as a combat move
- **Poptropica, Super Meat Boy** - Jumping for world traversal with very unrealistic physics.
- **Trials HD** - Jumping in unrealistic environments with very realistic physics.
- **Assassin's Creed** - Jumping 'on rails' for world traversal, jumping to escape enemies

General Patterns Exercise

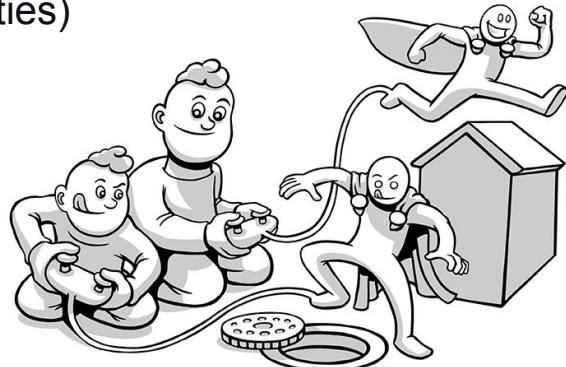
4. What design problems do the games use the element to solve?

- Navigation through the world space (All Games Listed) - All games that I could think of that used jumping used it as part of world traversal. At a base-level jumping gives the player **more movement options**.
- Creating a **sense of autonomy** in the player. (All Games Listed) - I think that this is because jumping increases the players' ability to move in the world. There is some subtlety in how this works, though. Given that in some games jumping serves to make the character able to interact with the world in a way that more closely mirrors the real world, and in others jumping serves to differentiate the character from the player by allowing the character to move through the world in ways that the player can not.
- Creating a **sense of danger** for the player (*Super Mario Bros., Mirror's Edge, Gravity Rush / VVVVVV, Super Meat Boy, Tomb Raider, Prince of Persia, Trials HD*) - Jumping has the real possibility of causing death in all of these games. The ratio of how dangerous jumping is to how much it lets you traverse the world seems to directly relate to the amount of power vs. fear that it creates in the player. In a game like *Super Mario Bros.*, you may die from jumping incorrectly, but mostly it increases your ability to navigate the world. In a game like *Geometry Dash* jumping does allow you to progress through the world, but it mainly the thing that causes you to die when you do it incorrectly.
- **Adding variety** to the ways the player can interact with the world. (All Games Listed except *Q-Bert* and *Geometry Dash* in which jumping is the 'only' way you move through the world. But particularly *Guild Wars 2*, where it is not a primary world traversal tool and mostly used in optional jumping exploration puzzles.)
- Enabling **player mastery** of game systems through creating complicated, intricate systems that require player **skill growth**. (All Games Listed) - The degree to which this is the point seems related to how central a mechanic jumping is and how complicated and subtle the jump mechanics are.
- Enabling **player mastery** of game systems by creating opportunities for the player to **subvert** them: (*Doom/Quake, Tribes*) This is interesting in that, in the case of these games, the mechanics were not intended to allow player subversion. Rocket Jumping and Ski-Jumping were on some level bugs that players found and used to enhance gameplay. The developers recognized the value of the bugs and incorporated them into future games intentionally.
- Character building through **giving the character abilities the player does not have**: (*Mirror's Edge, Gravity Rush, Alice, Tomb Raider, Poptropica, Assassin's Creed*) - Superhuman jumping abilities help make the characters seem superhuman. Improving a character's most basic movement abilities probably more profoundly differentiates them from the player.
- Maintaining immersion in the game world by making player abilities and movement match the player's understanding of **how the real world works**. (Interestingly none of the example games use jumping in this way, but other games do - *Silent Hill 2, Flashback*)
- **Enhancing combat** by enhancing aggressive player actions. (*Street Fighter/Soul Calibur, Super Mario Bros., Devil May Cry*) The jumping itself may not be aggressive, but it serves to amplify the character's aggressive action. A jumping punch to the head is just more impactful than a standing punch to the head.

General Patterns Exercise

5. Are there Patterns in the ways the elements are used that relate to the problems they solve?

- More complex mechanics provide more opportunities for player skill. (Autonomy and Mastery)
- When power has a cost, it's frightening to use (Dangerous Jumping)
- Two great things that go great together (Jumping and Punching)
- She's just like me! vs. I want to be her when I grow up! (Maintaining immersion by creating realistic character abilities vs. Character building through superhuman abilities)



General Patterns Exercise

6. Pick one of those patterns and describe it on the Pattern Library website.



One of these days that's going to get you killed.



Author(s):

Chris Barney

Groups(s):

Pattern Language for Game Design

2

Pattern

June 13th 2020, 11:14am

Confidence:

November 16th 2020, 9:10am

Created On:

Exercise 1: Basic Patterns Exercise — Jumping

Edited On:

Balance

Character Progression

Seed:

Mechanics

Keywords:

Jumping over a dangerous pit and suffering from a weapon overheating are both examples of this pattern in action.

Design Problem: How do you maintain game balance and create tension when giving the player greater power in their interactions with the game world?

Pattern Description: To maintain balance and create tension when designing character abilities, a designer may introduce consequences resulting from using those abilities. The result may be something natural, like falling into a pit of lava you try to jump over, or it may be something mechanical, like weapon heat build-up, or a stamina meter.

Example Games:

Super Mario Bros.

The ability to jump, which increases the character's ability to move through the world and defeat enemies, also puts him in danger. Failing to jump over dangerous obstacles can result in Mario's death. Similarly, failing to jump over an enemy results in the enemy killing Mario.

Sekiro: Shadows Die Twice

Stealth-killing enemies is the easiest way to defeat them, but failing to execute a stealth kill alerts the enemy and nearby enemies and suddenly puts you in a dangerous situation.

Anthem

Firing weapons increases their heat. Failing to manage that resource, to self-limit the damage you are doing, can result in not being able to fire your gun when you most need it.

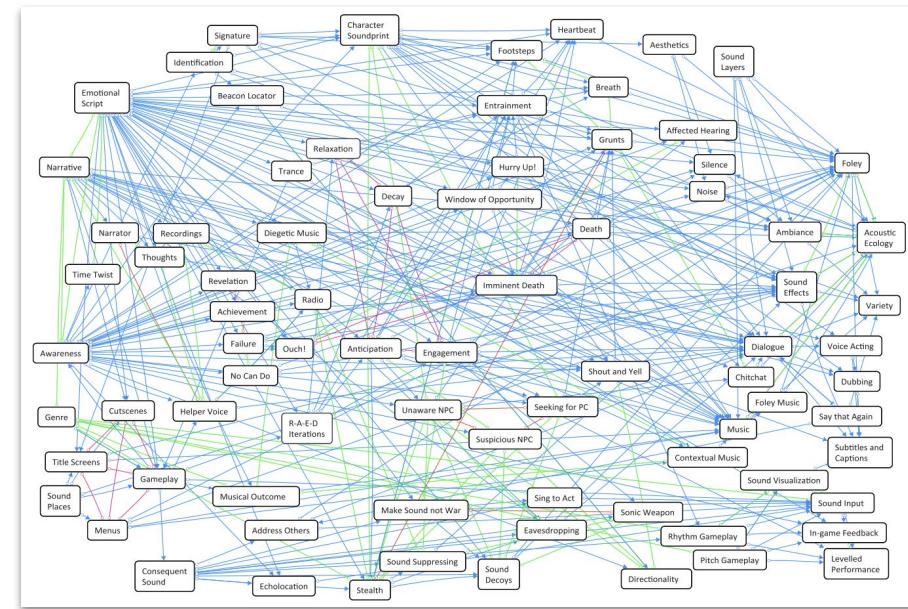
The Legend of Zelda: Breath of the Wild

Link can climb almost anything, but he has a stamina meter, so if he tries to climb something too high he will fall. He can jump off things and glide, but if he runs out of stamina, he falls to his death.

What is a Pattern Language

"In short, no pattern is an isolated entity. Each pattern can exist in the world only to the extent that is supported by other patterns: the larger patterns in which it is embedded, the patterns of the same size that surround it, and the smaller patterns which are embedded in it."

- Christopher Alexander



From Design Patterns in Games: the case for Sound Design
by Valter Alves and Licinio Roque

How Do You Make A Pattern Language

First, Connect the Patterns

- Keywords
- Parents
- Children
- Additive / Subtractive Patterns
- Alternate Patterns

Second, Understand Your Scope

- How broad/deep is your collection?
- How **confident** are you in your patterns?
- How **confident** are you in your links?
- Who is going to use your language?

Third, Make Patterns Findable

- Filter by Keywords
- Link to Related Patterns
- Search Titles
- Search Text
- Share your patterns with others

 Chris Barney

USER PROFILE:
Display Name: Chris Barney
First Name: Christopher
Last Name: Barney
Contact Email: Ryha2000@gmail.com

GROUPS:

Group Name: Pattern Language for Game Design
Group Type: Publication
Group Role: Group Manager — Author
Group Members: ★ Christopher Barney
★ Kamala Colonna
★ Michael Colonna
★ [REDACTED]

Group Name: Northeastern University
Group Type: School
Group Role: Group Manager — Professor
Group Members: [REDACTED]

A Library for Patterns in Game Design

This is one possible implementation of a Pattern Library allowing developers to collect and search patterns.

PATTERN LIBRARY

Active Pattern Search

And Now I Guess We Are Doing This

Players can get trapped in comfortable gameplay loops, which can ...

Bringing About the Apocalypse

Designers often need to apply esthetic themes to the game they're ...

Can I Do This Alone?

The following design problems may result in the inclusion of the lon...

Coercive Ludonarrative Resonance

Sometimes as a designer, you want to give the player a sense of em...

Don't intellectualize my pain!

Designers may need to connect players with the physical and emoti...

Familiarity Breeds Contempt, or at Least High Expectations

Resources are often limited in game development, and there may n...

Fight Like You Live

It can be easy for a developer to copy elements from other "good" ...

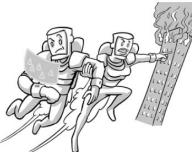
Game, Know Thyself

Several design problems may lead to the attempt to add additional ...

Greater Choice Requires Greater Motivation

How do you keep players motivated to explore and interact with sys...

Greater Choice Requires Greater Motivation



Author(s): Chris Barney

Group(s): Pattern Language for Game Design

Confidence: 2

Created On: March 3rd 2020, 8:39am

Edited On: November 3rd 2020, 12:59pm

Seed: Exercise 3: Lower-order Patterns Exercise – Architectural Whee...

Keywords: Autonomy | Choice | Open World | Player Motivation | Deep Interact | Inner Calm | Levels of Scale

Design Problem: How do you keep players motivated to explore and interact with systems as the scale of the game world and available resources increase?

Pattern Description: The greater the degree of character freedom or choice that is allowed in a game, the stronger the motivation that is required in order to make players feel like the choices they make are enjoyable and rewarding. In a very linear game, every player action visibly leads to progress in the game and the player is unlikely to feel that their actions are not meaningful. When games are nonlinear or allow a great deal of player exploration, or allow the player constant options as to what action to pursue it becomes more likely that players will experience uncertainty as to which actions are optimal or even whether their actions are generating meaningful progress in any aspect of the game, or are just a waste of time.

Active Pattern Search:

Exercise Patterns Only ⓘ

Filter by Keywords

- Abilities x Anticipation x Artificial Intelligence x Player Feedback x Positive Space x

Filter by Authors

- Chris Barney x

Filter by Group

- Pattern Language for Game Design x

Filter by Pattern Name

Enter the pattern name you want to search for.

Please select a Pattern to view.



Why do I need my own?

The problem space of game design is too large for any one group to derive all the patterns that define it.

Your problems are too specific to solve with just patterns created for other problems.

Your own language can become the framework that you hang your understanding of design on.

Add a Pattern
+

Open the New Pattern panel to add a pattern.

A Grinding Rhythm

Game developers want players to get into a rhythm,...

A Nudge Forward

In horror games, players may be too afraid to leave ...

A path for living

Sometimes designers may want players to pass thr...

Ability Signposts

How do you reinforce a game's unique mechanics a...

Action Economy

When a player has a plethora of ways to act within ...

Adapt or Die

How can a game be consistently challenging through...

Aim Of The Game

When designing gameplay systems for your game, i...

Ain't Got No Time For This

Decrease total game time while keeping the game s...

« < 1 2 28 29 30 > »

What would I do with it?

Learning Design

Teaching Design

Understanding your successes and failures

Solving a specific design problem

Planning your next game

Communicating your design ideas clearly

Reducing risk by using proven patterns without creating derivative games



Design Patterns as Pedagogy: *Learning and Teaching Functions as Proposed by Shuell and Moran*

Knowledge Manipulation

- **Encoding:** Developers create their own patterns, deciding how to express the ideas they have observed. They choose the pattern title and image to help solidify the concept the pattern captures.
- **Comparison:** During the first stages of the pattern process, designers compare the use of their seed technique across existing games. Later in the process, they identify the best applications of the pattern that they have articulated as it is applied in existing games and choose the most diverse applications as examples to include in their pattern.
- **Repetition:** Within a specific iteration of the process of pattern discovery, designers analyze ten or more games looking for uses of their seed technique. The larger process of language creation involves repeating the pattern creation process many times and reviewing the created patterns looking for connections.
- **Interpreting:** Designers must examine existing games which contain the techniques they are investigating, understand the use of those techniques, and then articulate the shared aspects of their purpose and implementation in the form of a pattern. Later, they must begin with a pattern and design a game that implements the pattern to achieve the previously stated purpose.
- **Exemplifying:** When completing an exercise, developers must provide examples of the use of the pattern. These examples usually differ from the games that were analyzed as the source for the pattern. Additionally, designers are encouraged to find the most diverse set of examples possible to illustrate the scope of their pattern.

Higher Order Relationships

- **Combination, integration, synthesis:** Individual patterns are created by observing and combining the purpose and implementation of techniques across games. Pattern languages are created by articulating the relationships between patterns in terms of subject, purpose and function.
- **Classifying:** Each pattern must be assigned a set of keywords to place it within the context of existing design theory. Three levels of keywords are provided: keywords which identify the patterns' subject matter, categories which place it in an area of design, and properties which indicate its purpose.
- **Summarizing:** The description of the pattern is a summary of the analysis that the designer has undertaken to derive the pattern.
- **Analyzing:** Patterns are created through the analysis of a set of existing games; these must be decomposed and understood in terms of the seed technique of the pattern exercise.

Learner Regulation

- **Feedback:** As part of the language creation process, patterns are peer reviewed and revised to best form the connections necessary for the language.
- **Evaluation:** On project completion, projects are peer reviewed to analyze the efficacy of their implementation of the patterns.
- **Monitoring:** During the use of patterns in design projects, the implementing designers provide feedback to the designers that developed each pattern.
- **Planning:** The use of patterns in practical design projects is intrinsically a planning process wherein the designers use patterns to structure their design prior to implementation.

Productive Actions

- **Hypothesis generation:** The process of pattern formation consists of analyzing data and forming a hypothesis.
- **Inferring:** Designers take existing design knowledge, examine existing examples of its use, and infer the patterns that it forms.
- **Explaining:** Creating the textual artifact of a pattern using the provided template allows designers to articulate and explain the theory they have constructed. Patterns are then further used to explain the more complex composite concepts that form a complete game design.
- **Applying:** Using patterns as the basis of design in practical game projects allows designers to apply the concepts that they have articulated and validate their efficacy.
- **Producing and constructing:** From simple scene implementations using a signal pattern to complex full game designs, the practical execution of a design into a game provides designers with the opportunity to demonstrate their learning in functional game artifacts.



Can Patterns Scale to Studio-Wide Use?

In the Summer of 2021 40+ students began a year long game design project creating a Pattern Language as the basis for their design.

In the Fall 10 students continued and 40 new students joined the project. Their introduction began with studying the chosen Pattern Language.

Decisions are made and disputes resolved by considering which choices are best supported by the Pattern Language.

The students are currently on track to complete their vertical slice by the end of this semester and begin full production in the Spring with a target of having a shippable game at the end of the academic year.



Takeaways

- A Game Design Pattern can help you understand and solve a design problem.
- A Pattern Language connects patterns so you can use them to solve complex problems or design whole games.
- Patterns you create yourself will be more useful than those you get from others.
- A Pattern Language can be a shared vocabulary to improve communication between developers.
- Patterns can form a useful basis for design and allow developers across a studio to understand and share responsibility for the design at a deep level.
- Validated patterns with high confidence may help reduce risk.

Resources

My Free Pattern Library and Book on Game Design Patterns
patternlanguageforgamedesign.com

My Design Blog with many articles on patterns
perspectivesingamedesign.com

Application of Pattern Language for Game Design in Pedagogy and Design Practice
<https://www.preprints.org/manuscript/202107.0485/v1>

Staffan Björk and Jussi Holopainen's Collection (800+ patterns)
virt10.itu.chalmers.se/index.php/Category:Patterns
 Visualization Tool: <http://gdpv.is/>

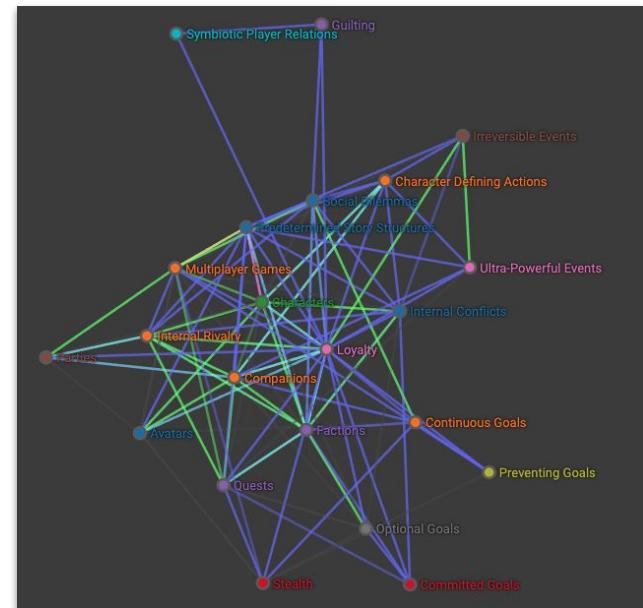
From Design Patterns in Games: the case for Sound Design (Alves and Roque)
www.fdg2013.org/program/workshops/papers/DPG2013/b1-alves.pdf

Patterns in Game Design (Bjork and Holopainen)

Game Mechanics: Advanced Game Design (Adams and Dormans)

Pattern Theory (Helmut Leitner)

A Pattern Language (Christopher Alexander)



Visualization tool for Björk & Holopainen's Collection

Some Advanced Topics

OWL 2 Ontology Language Definition

Using ontology reasoners

Games Reference and Game Demographics

Research and Validation

Generative Design Patterns



Student Pattern Project:

Using Patterns To Understand Techniques: Shooting

Based on a pattern that addressed the potential for stagnate gameplay in games with a shooting core mechanic this group created a game where the character switches dimensions every time they fire their gun creating a kinetic shooting puzzle game with potential on par with Portal.

Shoot to Thrill

Shooting in games is a mechanic that is best utilized in interesting, diverse ways.

Design Problem: Players like shooting games because it allows them to build up precision and skill in an environment where nobody else can compete. Many shooting games often utilize the same general principles, to the point of stagnating the mechanic altogether.

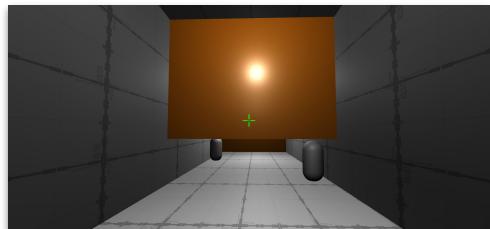
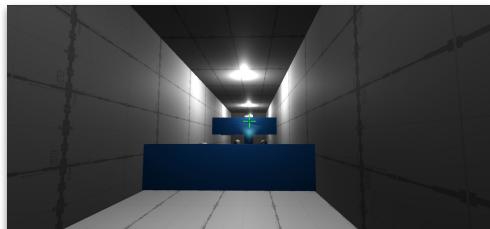
Pattern Description: Designers go about varying up shooting by completely swapping out the types of scenarios it is used for. By changing up how players go about utilizing shooting mechanics, they are able to take one relatively static mechanic and make it dynamic and interesting again.

Example Games:

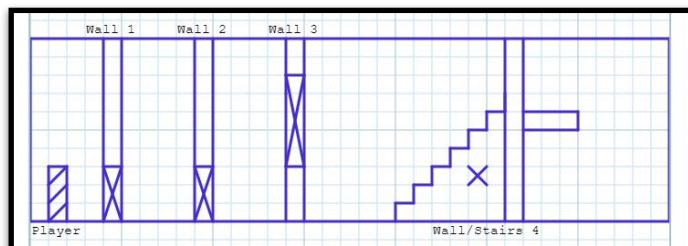
- The Legend of Zelda: Breath of the Wild
- Portal 2
- Fallout 3
- Team Fortress 2
- Splatoon 2
- Rainbow Six Siege

Related Patterns:

- Familiarity Breeds Contempt, or at Least High Expectations



Pattern by: Liam Cristello
 Project by: Christopher Boyd,
 Aharon Leichtman, James
 Mcmanus, Julia Sherbal, Jim Wu



OWL 2 Ontology Language Definition

Classes

Class: example_game	Class: game_release
Class: exercise	Class: game_type
Class: game	Class: group
Class: game_available_link	Class: group_type
Class: game_developer	Class: keyword
Class: game_info_link	Class: pattern
Class: pattern	

Data Properties

DataProperty: game_available_link_notes Domain: game_available_link Range: xsd:string	DataProperty: game_release_date Domain: game Range: xsd:dateTimeStamp	DataProperty: pattern_exercise_name Domain: pattern_exercise Range: xsd:string	DataProperty: hasRelease Domain: game Range: game_release
DataProperty: game_available_link_source Domain: game Range: xsd:string	DataProperty: game_release_name Domain: game_release Range: xsd:string	DataProperty: pattern_exercise_page Domain: pattern_exercise Range: xsd:positiveInteger	DataProperty: hasSuggestedExercise Domain: pattern Range: pattern_suggested_exercise
DataProperty: game_available_link_url Domain: game Range: xsd:string	DataProperty: game_release_notes Domain: game_release Range: xsd:string	DataProperty: pattern_image Domain: pattern Range: xsd:string	DataProperty: hasUser Domain: pattern Range: user
DataProperty: game_description Domain: game Range: xsd:string	DataProperty: game_release_type Domain: game Range: xsd:string	DataProperty: pattern_image_description Domain: pattern Range: xsd:string	DataProperty: owl:topObjectProperty Domain: related_pattern Range: pattern

Object Properties

ObjectProperty: hasAuthor Domain: pattern Range: user	ObjectProperty: hasGamePlatform Domain: game Range: game_platform	ObjectProperty: hasGamePublisher Domain: pattern Range: pattern_states
ObjectProperty: hasExampleGame Domain: pattern Range: related_pattern		ObjectProperty: hasRelatedPattern Domain: pattern Range: related_pattern

ObjectProperty: hasExercise Domain: game Range: game_exercise	ObjectProperty: hasGame Domain: pattern Range: game
ObjectProperty: hasGame Domain: pattern Range: game	ObjectProperty: hasGamePlatform Domain: pattern Range: game_platform

ObjectProperty: hasGamePlatform Domain: pattern Range: game_platform	ObjectProperty: hasGamePublisher Domain: pattern Range: pattern_states
ObjectProperty: hasGamePublisher Domain: pattern Range: pattern_states	ObjectProperty: hasRelatedPattern Domain: pattern Range: related_pattern

ObjectProperty: hasGamePublisher Domain: pattern Range: pattern_states	ObjectProperty: hasRelatedPattern Domain: pattern Range: related_pattern
ObjectProperty: hasRelatedPattern Domain: pattern Range: related_pattern	ObjectProperty: hasSuggestedExercise Domain: pattern Range: pattern_suggested_exercise

ObjectProperty: hasSuggestedExercise Domain: pattern Range: pattern_suggested_exercise	ObjectProperty: hasUser Domain: pattern Range: user
ObjectProperty: hasUser Domain: pattern Range: user	ObjectProperty: relatesTo Domain: related_pattern Range: pattern

OWL 2 Ontology Language Definition

```

Individual: Autonomy
Types: keyword

Individual: Bethesda_Software
Types: game_developer, game_publisher
Facts:
game_developer_name "Bethesda Software",
game_developer_notes "Large US based publisher comprising several AAA studios including Well known for the Elder Scrolls franchise."
Individual: Choice
Types: keyword

Individual: Chris_Baney
Types: user
Facts: hasGroup Northeastern_University

Individual: Exercise_11_Emergent_Narrative_Patterns
Types: exercise
Facts: pattern_exercise_name "Exercise 11: Emergent Narrative Patterns"

Individual: Exercise_24_Theoretical_Patterns
Types: exercise
Facts: pattern_exercise_name "Exercise 24: Theoretical Patterns"

Individual: Greater_Choice_Requires_Greater_Motivation
Types: pattern

Individual: Northeastern_University
Types: group
Facts: hasGroupType School

Individual: Open_World_Action_Game
Facts:
game_type_name "Open World Action Game",
game_type_notes "Typically third person action game that focuses on world exploration and providing a variety of player activities."
Individual: Personal_Computer
Types: game_platform
Facts:
game_platform_name "Personal Computer",
game_platform_notes "Personal Computer running the Windows operating system."

Individual: Published
Types: pattern_states
Facts:
pattern_state_name "Published"

Individual: School
Types: group_type
Facts:
group_type_description "Educational institution, college, or university teaching game design and producing design patterns"
group_type_name "School"

```

```

Individual: Skyrim_Available_Link_1
Types: game_available_link
Facts:
game_available_link_notes "Steam download for the Special Edition release of the game."
game_available_link_source "Steam",
game_available_link_url "https://store.steampowered.com/app/489830/The_Elder_Scrolls_V_Skyrim_Special_Edition/"

Individual: Skyrim_Info_Link_1
Types: game_info_link
Facts:
game_info_link_notes "",
game_info_link_source "Official Website",
game_info_link_url "https://elderscrolls.bethesda.net/en/skyrim"

Individual: Skyrim_Primary_Release
Types: game_release
Facts:
game_release_date "01/01/2011",
game_release_name "Primary",
game_release_notes "Initial release on Windows PC",
game_release_type "Primary"

Individual: The_Elder_Scrolls_V_Skyrim
Types: game
Facts:
hasGameAvailableLink Skyrim_Available_Link_1,
hasGameDeveloper Bethesda_Software,
hasGameInfoLink Skyrim_Info_Link_1,
hasGamePlatform Personal_Computer,
hasGamePublisher Bethesda_Software,
hasGameType Open_World_Action_Game,
hasGameRelease Skyrim_Primary_Release,
game_image "[Skyrim Image URL]",
game_main_storyline "[The Elder Scrolls V: Skyrim]",
video_gameplay "[SRVpNroNoU",
video_trailer "[SRVpNroNoU"

Individual: Three_Pillars_Relationship_1
Types: related_pattern
Facts:
relatedTo Greater_Choice_Requires_Greater_Motivation,
related_pattern_confidence 2,
related_pattern_interpretation "When you have applied the Three Pillars of Meaning to situations where there are emergent narratives and player choices, those choices will be meaningful. The more significant your choices, the more of them your game will be able to support",
related_pattern_type "Child"

Individual: Three_Pillars_Seed
Types: pattern_seed
Facts:
pattern_seed_description "What makes emergent events narratively meaningful?"

```

```

pattern_seed_name "Seed for Three Pillars of Meaning in Emergent Narrative"

Individual: Three_Pillars_Skyrim_Example
Types: example_game
Facts:
hasGame The_Elder_Scrolls_V_Skyrim,
pattern_example_game_description "The degree to which the three pillars are present for emergent narrative elements in this game varies. At worst, the events are isolated and unrelated to the player or the world: a group of bandits in a cave with no associated NPCs or consequences for "riddling the countryside" of them. At best, all are present: killing an NPC in town results in the guards becoming hostile, the character attracting the attention of the assassins' guild, and the inability to wear holy armor due to your evil actions."
Individual: Three_Pillars_Suggested_Exercise_1
Types: pattern_suggested_exercise
Facts:
hasExercise Exercise_24_Theoretical_Patterns,
pattern_suggested_exercise_description "Use Exercise 24: Theoretical Patterns to generate a parent pattern based on the theory that The Three Pillars of Meaning in Emergent Narrative is generalizable to narrative in general."
Individual: Three_Pillars_of_Meaning_in_Emergent_Narrative
Types: pattern
Facts:
hasAuthor Chris_Baney,
hasExampleGame Three_Pillars_Skyrim_Example,
hasExercise Exercise_11_Emergent_Narrative_Patterns,
hasGroup Northeastern_University,
hasKeyword Autonomy,
hasKeyword Choice,
hasPatternSeed Three_Pillars_Seed,
hasPatternState Published,
hasRelatedPattern Three_Pillars_Relationship_1,
hasSuggestedExercise Three_Pillars_Suggested_Exercise_1,
pattern_confidence 2,
pattern_created_date "12/22/2020",
pattern_description "To allow players to construct meaningful emergent narratives, developers should provide players with context, motivation, and consequence for their actions in the game. There are many child patterns that contribute to this pattern; two are listed and 12 other possibilities are provided in the suggested exercise, and there may be more. But each one contributes to either the context, of motivation for, or consequences of an event that could be part of an emergent narrative.",
pattern_design_problem "Given the reality of limited resources, when creating design elements that encourage emergent narratives, designers need to maximize the narrative potential of every element. How can designers know if a given game element will contribute to meaningful emergent narratives?",
pattern_image "[pattern_image.jpg]"
pattern_name "Three Pillars of Meaning in Emergent Narrative"

```