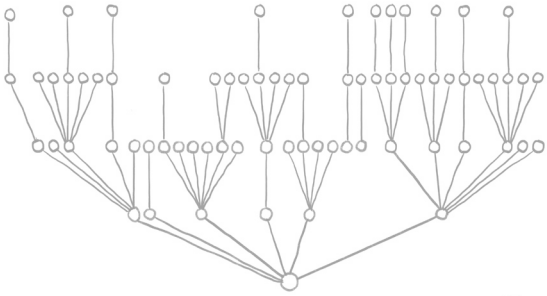


WILL WRIGHT | CHAPTER 03

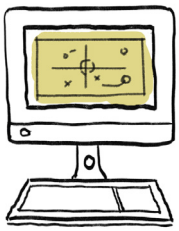
Early Prototyping

TERMS

prototype (n.) The simplest possible execution of a design concept.

paper prototyping (v.) To use simple analog materials to create an interactive experience which answers a design question by testing a game concept or feature.

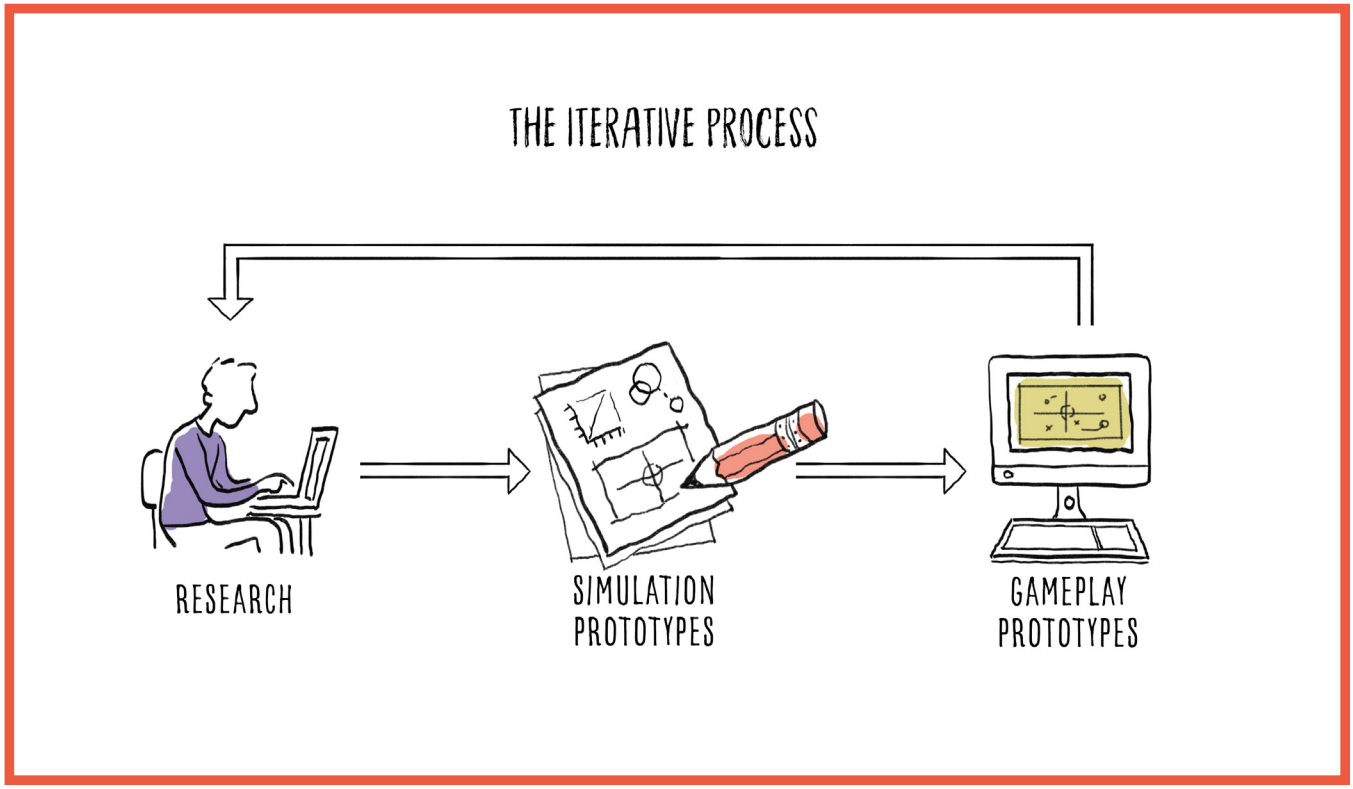
iterative design (n.) A repeating design process in which a prototype is tested, results are analyzed, and the prototype is rebuilt based on the findings.



Rapid prototyping is the central discipline of the game design process. You should build your prototypes as quickly and cheaply as possible, with a specific question or goal in mind. Don't spend any time agonizing over what form the prototype should take. The important thing is to build something interactive as quickly and cheaply as possible, learn a lesson from it, and move on to other branches of your design.

While prototyping, be open to discovering which moments of the play experience are fun, no matter how trivial or incidental they might seem at the time. Then take note of those moments for use in a future prototype and begin reworking the concept from a different angle. Prototypes can take any form, but there are two major categories.

"A prototype is a navigation instrument... it's a compass."



Paper prototyping: These are prototypes that aren't built on a computer. Tangible models help you understand how to give your game nouns authenticity and weight, and what it should feel like to interact with your game in a digital space. (Learn about game nouns in Chapter 7: Develop a Game Language.) Use pencil, paper, scissors, and glue to build an interactive experience that tests a specific concept or system.

Code prototyping: The benefit of prototyping with code is that you can easily tweak variables to explore different behaviors. The downside is that it usually takes more time and effort than paper prototyping. Use the simplest platform at your disposal and only do so if you can produce something quickly.

LEARN MORE

- Read Brian Crecente's *Polygon* article on the development of Ojiro "Moppin" Fumoto's *Downwell*. How did keeping an open mind during the rapid prototyping process help Fumoto find the core mechanic for his game?

- Read the "Eight Tips of Productive Prototyping" from chapter seven of Jesse Schell's book *The Art of Game Design: A Book of Lenses* (CRC Press, 2008). Think about his advice on digital versus paper prototyping. If you primarily prototype digitally, consider trying out paper prototyping, and vice versa.

ASSIGNMENTS

- Quickly build a paper prototype of one of the game concepts you developed in Chapter 2: Generating Game Concepts. Find a clean workspace and gather these materials: blank paper, colored construction paper, pens and pencils in several colors, scissors, glue, as many dice as you can find, a handful of chits or coins to use as resources, and any small objects you'd like to use as player pawns. First, think about your game concept from several different perspectives, and decide which one would be the most fun to inhabit as a player. Then design a set of actions players can take that will allow them to operate from that perspective. Build a responsive environment in which players will execute those actions. Take note of interesting conflicts or dynamics within your new game. Add the prototype to your library, and write a cover sheet for it.

- Find a digital game development environment you're unfamiliar with and watch a tutorial about how it works on YouTube. Write down which scripting language that environment uses, and find a code sample. Read the sample and see how legible it is to you. Try to edit one or two lines of code and see how the script runs afterwards. Here are some free and inexpensive engines for you to explore: Unity, Unreal, Gamemaker Studio, Twine, PuzzleScript, Godot, and HTML5.