TUM Computer Games Laboratory 2015

N. Thuerey, G. Groh, R. Westermann
Organizational

- The GameLab Wiki: [https://gameslab.wiki.tum.de/](https://gameslab.wiki.tum.de/)

- You can request access via: [https://www.wiki.tum.de/join/DQ7RNR7](https://www.wiki.tum.de/join/DQ7RNR7)
TUM Computer Games Laboratory 2015

*with content from*

ETH Game Programming Laboratory

http://chrishecker.com/Advanced_Prototyping

Prototyping

N. Thuerey, G. Groh, R. Westermann
Prototyping

Recommended Reading:

Chapter on Prototyping
Prototyping

- Why a prototype?
- “Creating a game without a prototype is like shooting a movie without a script.”

- A prototype adds more to a game than a script or doc can do:
  - Interactivity
  - Test and exploration
  - Modification
Prototyping

• Prototype
  – Something fast and cheap that allows you to answer a specific question about your game: “Is your game idea fun?”
    • Not something that eventually morphs into a game
    • Not something using the same technology as the production code
    • Not meant to impress others
    • Rather some form of scientific exploration
Prototyping

- **Prototype**
  - Only fundamental mechanics (formal elements)
  - Considers player(s) and the computer
  - Rough approximation of artwork and features
  - Focus on gameplay, abstract from production process
  - Extensible, instrument for radical changes
Prototyping

• The purpose of a prototype
  – Do not create a prototype to show something about the game
    • Rather validate or disprove some concept
  – Do not create a prototype for a game you’ve already committed to
  – Grab the first idea and prototype; With a good prototype it’s easy to see if an idea is worthwhile
  – Prototypes don’t generate ideas from scratch, they validate them
  – Find upside and downside
  – Experiment, persuade and inspire
Prototyping

• The purpose of a prototype
  – Define core gameplay in purest form
  – Learn whether core mechanics hold interest of players
  – Test
    • Game mechanics
    • Balance of rules (too restrictive, too loose, too few, too many)
    • User experience
    • Embedded technologies
  – Discover play patterns and emergent behavior specific to your game
Prototyping

• The question to be answered by the prototype
  – A good question is concise and can be answered in a fairly unambiguous way
  – Ask yourself were you need understanding
  – Can be about game design, but can also be about any other aspect of the game
    • How to make things look heavy by coloring
    • Can I control a pen via my iPhone
    • Does this sound bring me into the mood, etc.
  – Do not try to ask for a good game idea via a prototype
  – You might have to decompose a big problem into smaller tractable ones
Prototyping

• Measure the quality of a prototype
  – Find relevant characteristics; interactivity, robustness, usability, beauty, performance, agility, and many more
  – Evaluate per characteristic quality/cost
Prototyping

• How to generate a prototype
  – Look for the cheapest way to get it
  – Steal it or fake it
  – Use available tools
Prototyping

• How to generate a prototype
  – Let the designer set the problem, let the programmer understand the goal, work together to solve the problem
  – Which prototyping technique? Paper, storyboard, physical, software
  – Code is expensive, use content by default (a physical prototype often works)
  – Only spend code where you need understanding;
  – Don’t restrict your mental capabilities by code
  – Don’t care about robustness, code optimality, software engineering
  – If software, choose between rapid prototype and using a real engine
Prototyping

- Prototyping techniques
  - Paper prototypes
    - Good for testing game mechanics, quick to produce, but cannot convey game experience and action
  - Storyboard and animatics
    - Captures user experience, useful for communicating ideas
  - Software prototypes
  - Physical prototypes
Prototyping

• Physical prototypes - Step 1: Foundation
  – Design the basic game objects and mechanics
    • Cards, paper, pens, toys etc.
  – Test your game idea
  – Restrict yourself to the fundamentals; Don’t try to be too detailed
Prototyping

• Physical prototypes - Step 1: Foundation
  – Example: first-person shooter
  – Core gameplay: simultaneous action
  – Accomplish with action cards
Prototyping

• Physical prototypes - Step 2: Skeleton
  – Prioritize what is most essential and refine
    • Number of spaces player can move
    • Procedures for turning
    • Hit and miss rules for shooting
  – Build upon foundation with structure to support essential parts of game
    • Scoring system
    • Hit points
Prototyping

- Physical prototypes - Step 3: Formal Details
  - Add rules and features for a fully functional fun game
  - Focus on most important formal elements
    - Is objective interesting and achievable
    - Is player interaction ideal
    - Are there missing rules
  - Test each rule individually to determine if it is critical or not
    - Hit percentage, health scoring, …
Prototyping

• Physical prototypes - Step 4: Refinement
  – You have a playable system
  – Play, tweak, play, tweak, play, tweak,…
  – Question smaller and smaller details
  – Especially: Is the game fun?
  – Add new features one at a time
Prototyping

• Demo your prototype
  – Be quiet, watch the users first reaction
  – Do not try to convince them how cool it is; Find out how cool it is
  – Observe, do not suggest
Final game pitch and prototype presentation

• Make your presentation exciting, energetic, visual, professional, and clear
  – Describe the essential design elements and back up your descriptions with sketches, storyboards, and other visuals
  – Motivate the design decisions with respect to the game theme
  – Show your "Big Idea Bullseye"
  – Convince us that your game idea will be a fantastic game!
  – Demonstrate the gameplay using your physical prototype
  – The prototype should support your game pitch by highlighting the core game mechanics and proving that it is a fun, playable game
  – Plan for a 10 minute presentation and practice ahead of time