



Rethinking Game Architecture with Immutability

JACOB DUFAULT

SPONSER: DR. BERNHARD

Milestone 5 Progress Summary

- ▶ Create poster
- ▶ Implement automatic synchronization error detection
 - ▶ For robustness, the implemented method serializes the game state and then computes a hash based on that state. These hashes can then be sent across the network and compared – if they differ, then the game simulations have diverged.
- ▶ Continue work on Unity integration
 - ▶ Bug fixes and feature polish, primarily within the reflection code for the inspector logic
- ▶ Begin work on XNA/MonoGame integration
 - ▶ Basic rendering is working; bindings will be 2D only

Milestone 5 Progress Summary

- ▶ (from milestone 6) Begin work on sample game
 - ▶ Milestone 5 completed sooner than expected
 - ▶ Basic gameplay implemented
 - ▶ Could be more complicated
 - ▶ Demo running in Unity
 - ▶ <http://i.imgur.com/1G9n9Uq.gif>

Milestone 6

- ▶ Work on XNA/MonoGame bindings
 - ▶ Just a fully working renderer
 - ▶ No editor – that would require a GUI, which is out of scope
- ▶ Game sample running in both Unity and XNA/MonoGame
 - ▶ Currently runs in Unity – convert to 3D
 - ▶ Needs to run in XNA
 - ▶ (stretch) Have it be multiplayer between the two different renderers
- ▶ Write user manual and documentation
 - ▶ Guide going through the sample
 - ▶ API docs generated from extensive comments in the code base

Questions?

Thanks!