# Rethinking Game Architecture with Immutability – Milestone 5 (March 19th)

Student: Jacob Dufault ([jacobdufault@gmail.com](mailto:jacobdufault@gmail.com))

Faculty sponsor: Professor Bernhard

# Progress Matrix

|  |  |
| --- | --- |
| Task | Completion |
| Create Poster | 100% |
| Implement automatic synchronization error detection | 100% |
| Continue work on Unity integration | 100% |
| Begin work on XNA/MonoGame integration | 100% |
| (from milestone 6) Begin work on sample game | 50% |

# Task Summaries

## Create Poster

The poster has been created and looks good to go. Unfortunately, benchmarking the multithreading logic actually showed a decrease in execution time, but this may change as the number of systems rises, as the test logic only has a few (5 or so) systems. To take full advantage of the multithreading logic, there will need to be a large number of systems with some of them executing for a long period of time. This is simply not the case with the current sample game.

## Implement automatic synchronization error detection

In order to ensure the most robust method possible, the game state is serialized and then a hash is generated based on that string. These hashes can then be compared to determine if the games have gone out of synchronization.

## Continue work on Unity integration

While writing the sample game, a number of bugs have been fixed within the Unity integration logic. Additional polish has been added to the integration bindings. An additional dependency component for non-multiplayer games has been added that removes all input delays (which are on by default for multiplayer to remove game update stuttering).

## Begin work on XNA/MonoGame integration

Luckily, the XNA/MonoGame bindings do not need to be as complex as the Unity bindings. Work has begun on these; basic rendering is now working.

## Begin work on sample game

The milestone goals for milestone 5 where achieved sooner than expected. Work has begun on one of the milestone goals for the next milestone, the creation of a sample game. Basic gameplay has been implemented, where a unit walks around a screen with a large number of asteroids and can shoot bullets to destroy these asteroids.

# Next Milestone

|  |  |
| --- | --- |
| Title | Summary |
| Work on XNA/MonoGame bindings | The XNA/MonoGame bindings need to be finished by the end of this milestone, as the game demo depends on them. There will not be an editor component attached to XNA/MonoGame, as that would require an entire GUI framework to be developed, which is simply out of scope for this project. |
| Demo a simple game running in both XNA and Unity | Work has already begun on the game sample, but more gameplay features can be added. Hopefully a networking demo will be doable with the Unity and MonoGame samples connected to each-other. Unity and MonoGame integration need to be completed, as well as potentially introducing more complex mechanics into the game. |
| Writer user manual and documentation | There can always be more documentation. A tutorial going through the game sample will be written, and perhaps there can also be auto generated documentation based on the extensive comments in the codebase. |

# Sponsor Feedback

Signature and Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Feedback:

# Sponsor Evaluation – Rethinking Game Architecture with Immutability – Milestone 5

## Jacob Dufault

Score (0-10):

Signature & Date: