

Doom 3 Bfg Concrete Architecture

Basement Gurus

Isaac Chan

Daniel Elmer

Matthew Sherar

Yohanna Gadelrab

Dylan Liu

Kainoa Llyod

Overview

1. Introduction
2. Derivation Process
3. Updated Conceptual Architecture
4. Concrete Architecture
 - ◊ Dependencies
 - ◊ Reflexion Analysis
5. Alternatives
6. Game Logic Subsystem
7. Sequence Diagram
8. Concurrency
9. Limitations
10. Lessons Learned
11. Summary

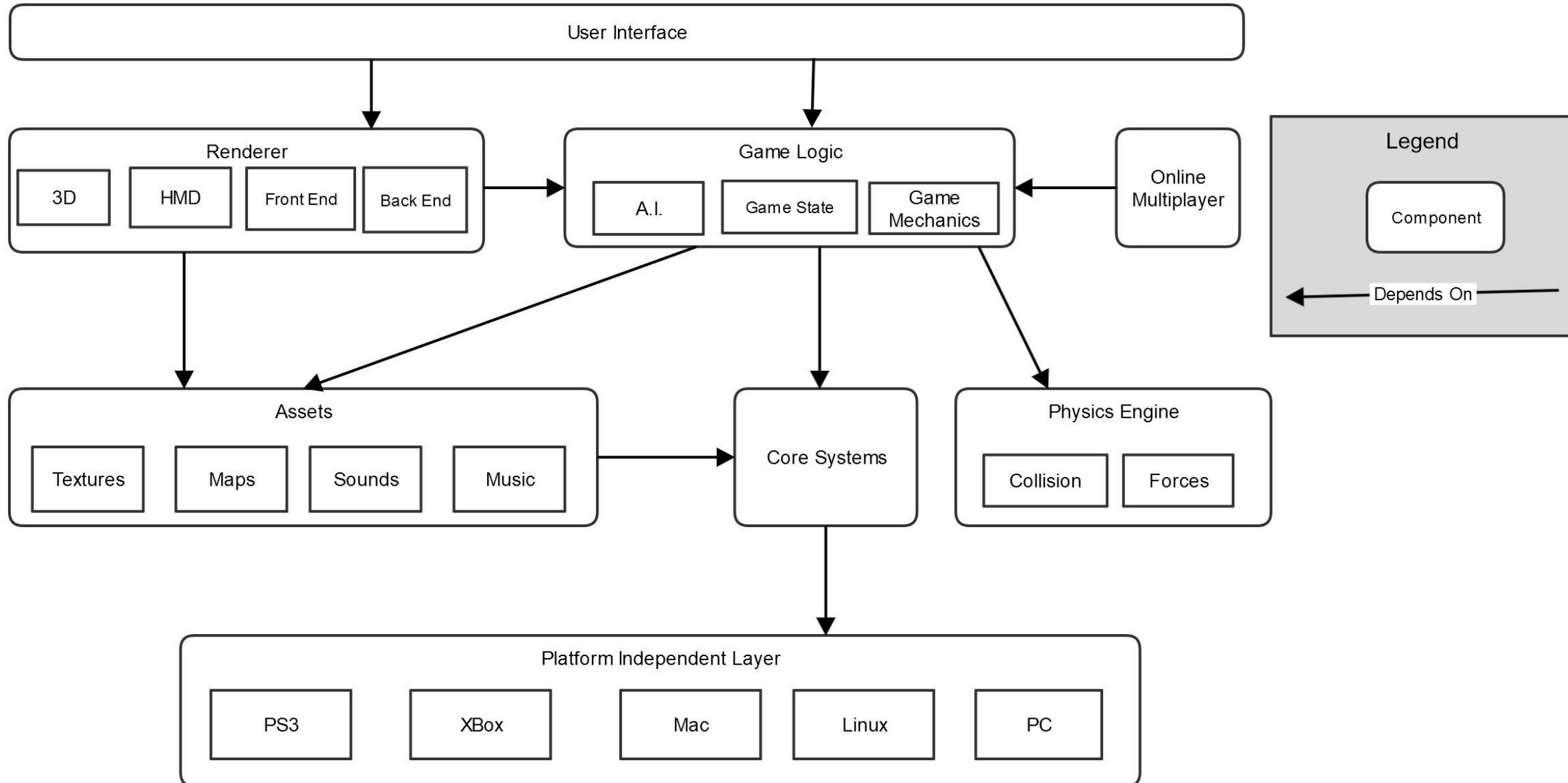
Introduction

- ❖ Concrete Architecture
 - ❖ Object Oriented Layered Architecture
- ❖ Subsystem: Game Logic Subsystem
- ❖ Reflexion Analysis of Concrete vs Conceptual Architecture

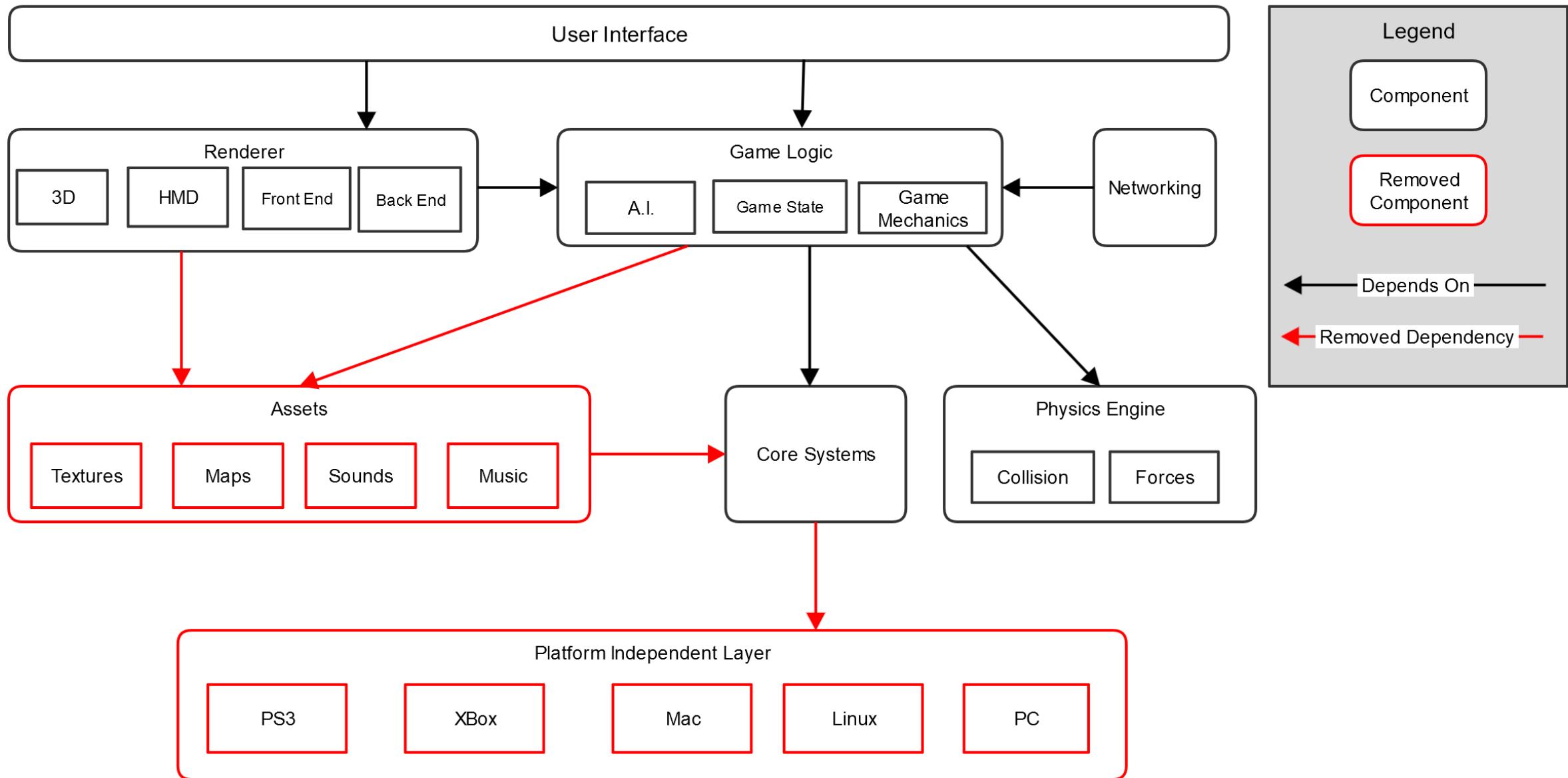
Derivation Process

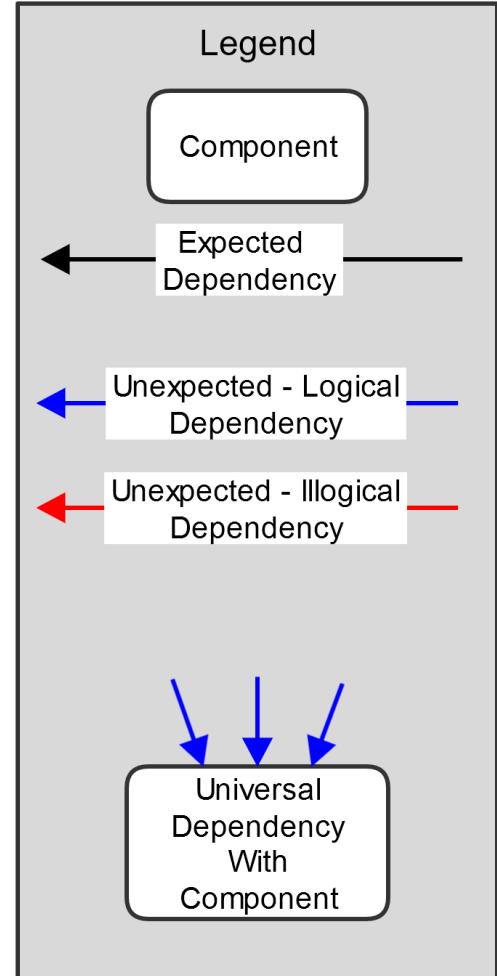
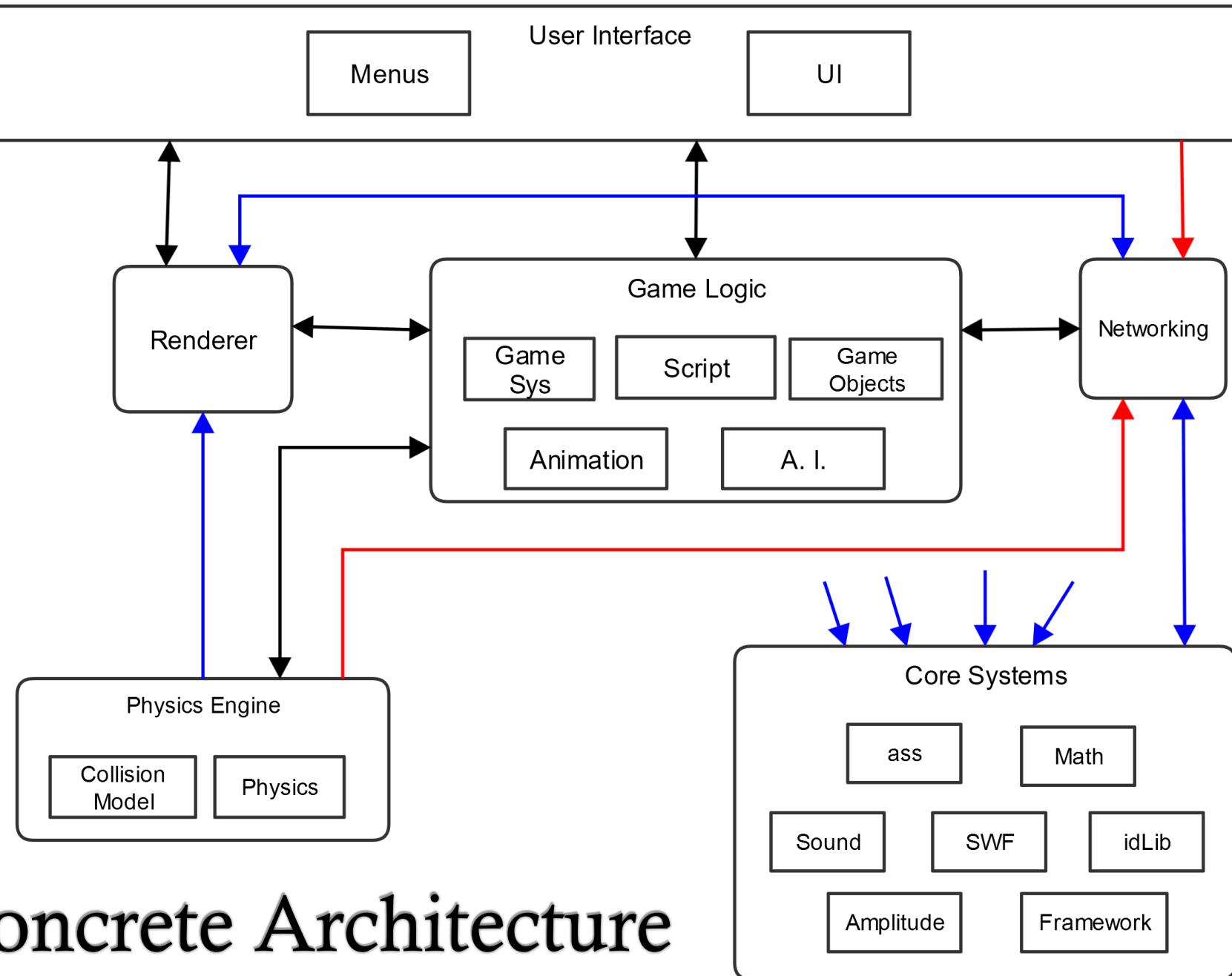
- ❖ Conceptual Architecture as a basis
- ❖ Analysis of source code using Understand
- ❖ Identified subsystems and decided on Game Logic subsystem
- ❖ Analyzed discrepancies

Old Conceptual Architecture



Updated Conceptual Architecture





Concrete Architecture

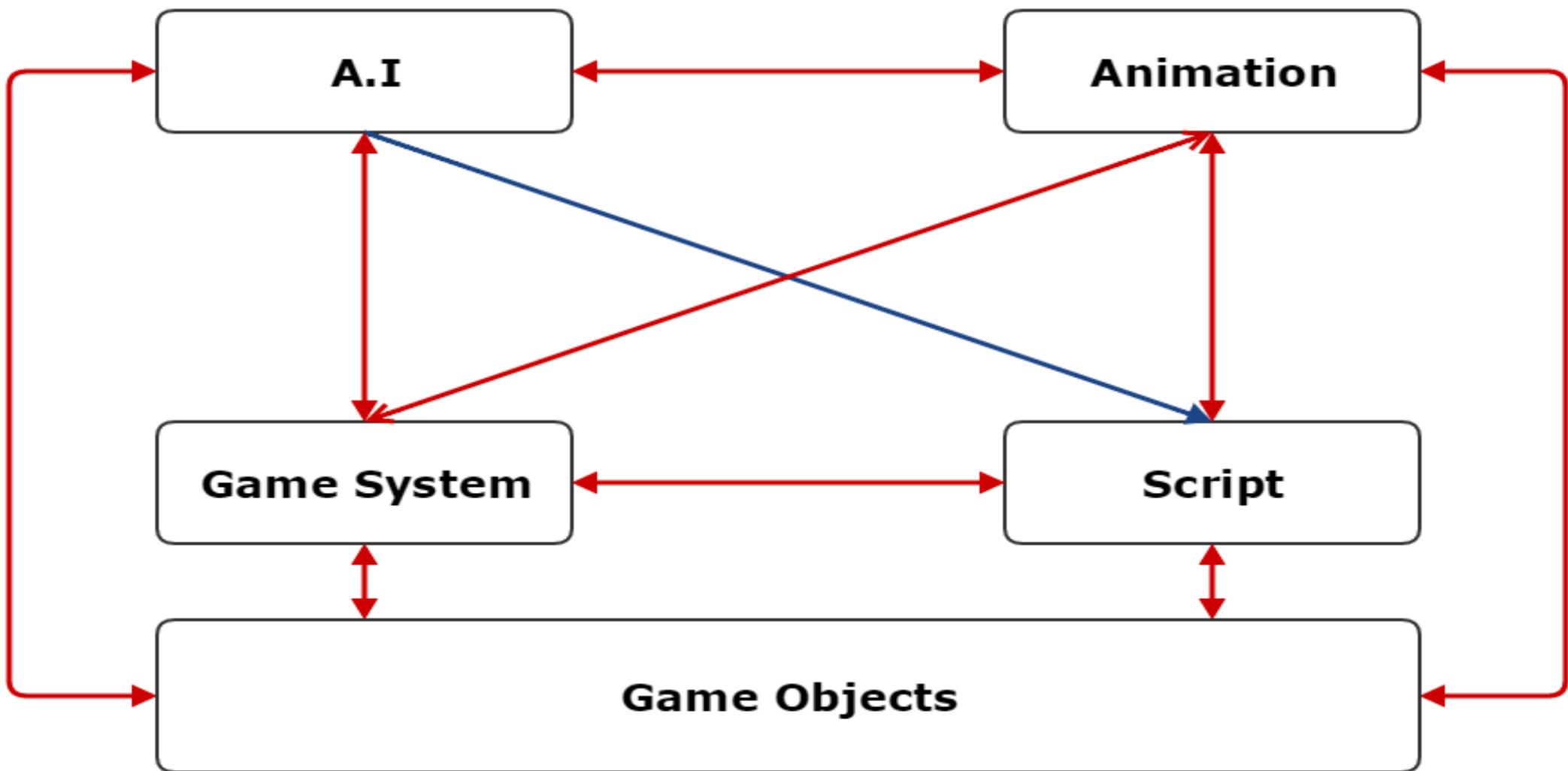
Unexpected Dependencies

- ❖ **Logical:**

- ❖ Universal bidirectional dependency with **Core System**
- ❖ Unidirectional dependency between **Physics** and **Renderer**
- ❖ Bidirectional dependency between **Renderer** and **Networking**

- ❖ **Illogical:**

- ❖ Unidirectional dependency between **UI** and **Networking**
- ❖ Unidirectional dependency between **Physics** and **Networking**



Legend:



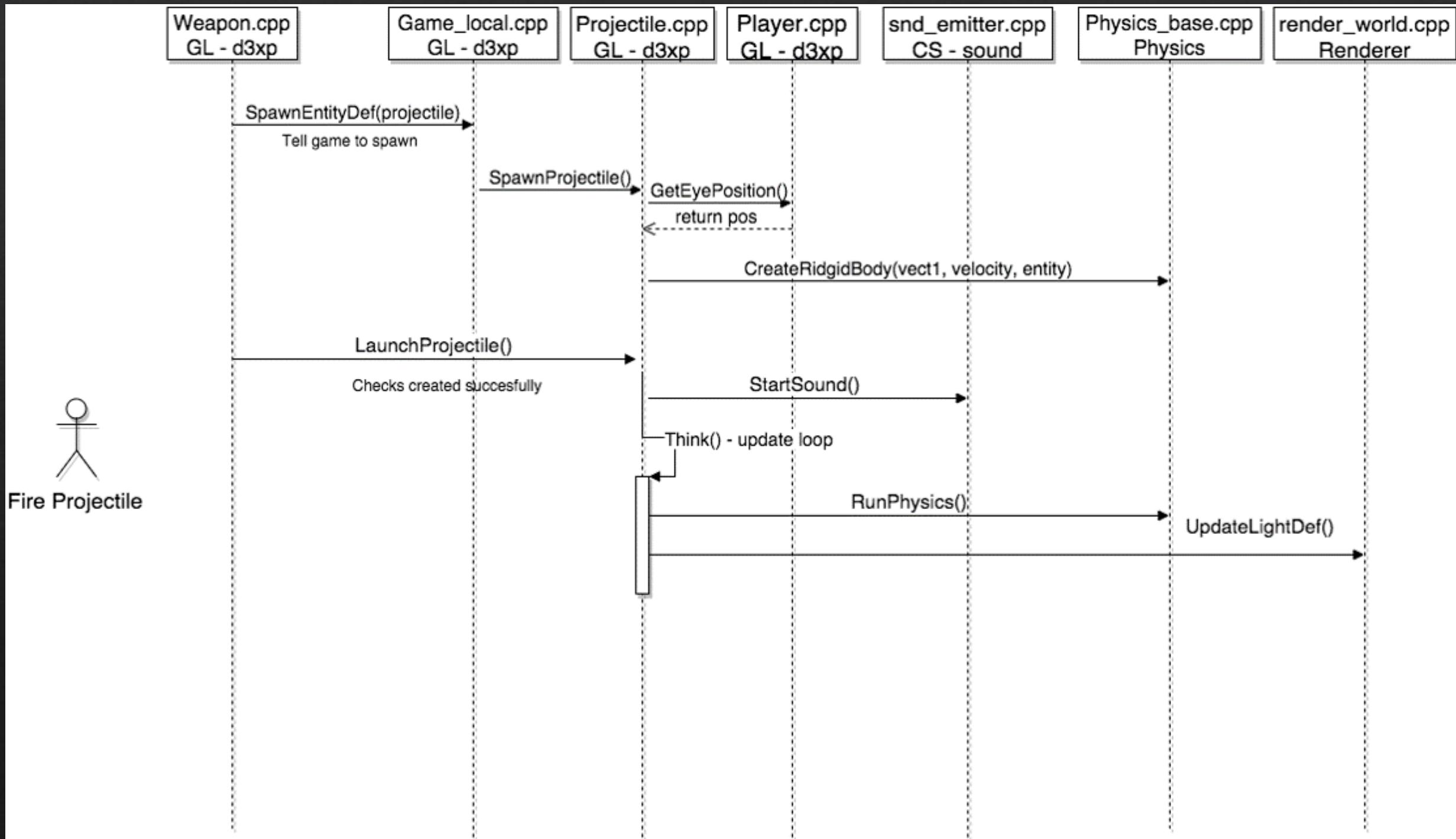
Internal Dependencies

Alternative Architectures

- ❖ Initially had platform independence layer
- ❖ Initially had game assets as it's own component
- ❖ Tried including **idLib** in Game Logic (**precompiled.h** includes many files found in Game Logic Component)
- ❖ Tried **gamesys** in core systems but too tightly coupled with other game logic components

Sequence Diagram

Player Shooting



Concurrency

- ❖ Follows the conceptual triple-threaded multi-threading model with support from Worker threads.
- ❖ Code responsible for maintaining multi-threading exists in the **Core System** under **idLib\sys**.
- ❖ Core System is called by all subsystems – multi-threading is ubiquitous in the program.

Limitations

- ❖ Knowledge of the Understand program
- ❖ Understand Crashing
- ❖ Lack of knowledge of C++

Lessons Learned

- ❖ Use a program to determine dependencies
- ❖ Unexpected dependencies
- ❖ Folder organization doesn't necessarily reflect components and architecture

Summary

- ❖ DOOM 3 BFG uses an Object Oriented Layered Architecture
- ❖ Game logic subsystem uses an Object Oriented Architecture
- ❖ Major changes in our conceptual architecture
- ❖ A few unexpected dependencies
- ❖ Learned how to accurately derive a concrete architecture

