

Distributed Application Server for MMO Games

Problem

Online multiplayer games, especially massively multiplayer online (MMO) games such as “World of Warcraft”, face *multiple challenges* that make them different from typical business applications. Online games require:

- *low latency* (opposed to high throughput)
- typically *50%/50% read-write load* (opposed to mostly reads)
- *high scalability* and *high availability*

Many databases, designed for business applications, are too slow for games. As a result all the popular MMO games must implement their own server-side architecture, which is *very hard*. This leads to *huge development costs* and *concurrency bugs*.

And to host thousands of players, MMO games usually must rely on sharding, where *players on one server can not interact with players on another server*, which is a gameplay compromise.

Business Case

A platform that solves the complex details of concurrency, scalability, failover, persistence etc. would *dramatically reduce the cost* of developing online games and virtual worlds. The same platform can also be used for *non-game applications with similar performance requirements*.

Solution

Dimdwarf Application Server (<http://dimdwarf.sourceforge.net>) is an unintrusive, distributed, transactional, scalable, application server and object database for the needs of online games. It allows the application programmer to *write single-threaded code*, which the application server then *executes concurrently in a server cluster*, adding transactionality, persistence, failover and scalability. Only one similar system exists (Project Darkstar).

For students this project will have *multiple interesting challenges*: designing a **distributed fault-tolerant cluster computing** solution, implementing **distributed garbage collection** and other algorithms, **formal verification** of the system’s correctness, **community detection algorithms** for load balancing the data, **on-the-fly data migration** etc. Part of the team may also focus on **implementing online games** using the platform.

The system is developed using TDD and following the SOLID principles. Some technologies used are: Java 6, Scala 2.8, Java bytecode, Guice 2, Apache MINA, CGLIB, ASM, Maven, Git.

“Friends don’t let friends try to make MMORPGs”
Do you know why?

