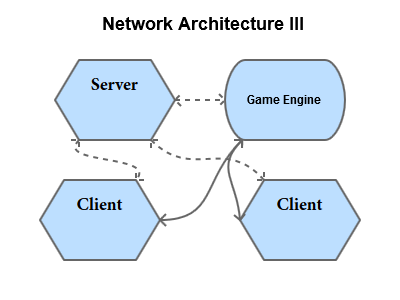
Requirements



# Motivation

Server allows multiple games to interconnect and allow

# Game Engine

* Stores game resources
* Provides game logic
* Coordinates clients.
* Handles chat between players
* Uses HTTP or TCP sockets and JSON for communication

# Server

* Authenticates users
* Identifies Clients
* Provides game engine lookup
* Uses HTTP and JSON for communication
* Python

# Clients

* Input/output for Game Engine
* Displays resources from the server
* Uses HTTP or TCP sockets for JSON communication

# Client Platforms

* Windows Modern
* Android
* Web

# System

* Variety of games
* Multiplayer
* Chat / Social
* Variety of clients
* Variety of platforms

# Games

## Risk

* Up to six players
* Turn-based
* Store Map
* Track units
* Track bonuses

## StarCraft

* Up to eight players
* Store map, exploration, and visibility
* Real-time
* Track units
* Track score

## SpaceTrader

* Multiplayer
* Store map graph
* Turn-based / Real-time
* Player trading
* Track player
* Track upgrades

# Communication Process

1. Game engine registers IP address, port, game name and game parameters with server.
2. Client contacts server with client parameters and game name.
3. Server returns game information to client
4. Client establishes connection to game engine
5. Client and game engine communicate until game completion
6. Connection is terminated and client idles until repeating the process

|  |  |  |
| --- | --- | --- |
| **Client** | **Direction** | **Server** |
| Connect | Connect 🡺 | Accept Connection |
| Request Session with Server | Request Session 🡺 | Authenticate client  Accept/Reject Response |
| If(reject) then end communication  If(accept) then send game request | 🡸 Success OR Fail |
| Game List Request 🡺 | Send all names of games that support the client |
| 🡸 Game list response |
| Game Request 🡺 | If(game = registered) and if(game supports client)  Send game info |
|  |  |
| Store game info  Terminate Session  Disconnect  Disconnect  Play Game | 🡸 Game Info |
| Terminate Session 🡺 | End session  Disconnect  Disconnect  Listen |
| Disconnect 🡺 |
| 🡸 Disconnect |
|  |
| Message | Message 🡺 | If(bad message or server error) return error |
|  | 🡸 Error |

|  |  |  |
| --- | --- | --- |
| **Client** | **Direction** | **Game Engine** |
| Connect | Connect 🡺 | Accept Connection |
| Request Session with Server | Request Game Session 🡺 | Authenticate client  Accept/Reject Response |
| If(reject) then end communication  If(accept) then send begin game | 🡸 Game Session Response OR Fail |
| . . . | Perform command Send game update |
|  | Game Command 🡺 |
| Update game  Issue game command | 🡸 Game update |
|  | . . . |  |
| Terminate Game  Disconnect  Disconnect | Terminate Game 🡺 | End game  Disconnect  Disconnect |
| Disconnect 🡺 |
| 🡸 Disconnect |
|  |  | Listen |
| Message | Message 🡺 | If(bad message or server error) return error |
|  | 🡸 Error |

|  |  |  |
| --- | --- | --- |
| **Game Engine** | **Direction** | **Server** |
| Connect | Connect 🡺 | Accept Connection |
| Request Session with Server | Request Session 🡺 | Authenticate client  Accept/Reject Response |
| If(reject) then end communication  If(accept) then send game info | 🡸 Success OR Fail |
| Game Inform 🡺 | Store Game Info  Send success/fail |
|  | 🡸 Success OR Fail |
| Terminate Session  Disconnect  Disconnect | Terminate Session 🡺 | End session  Disconnect  Disconnect |
| Disconnect 🡺 |
| 🡸 Disconnect |
| Listen |  | Listen |
| Message | Message 🡺 | If(bad message or server error) return error |
|  | 🡸 Error |
| If(bad message or server error) return error | 🡸Message | Message |
| Error 🡺 |  |

# Game Prototype

## Objects

* Player
* Character
* Ship
* Path
* Spaceport
* Commodity
* Credits
* SpaceMap
* Game
* Parameters

## Object Hierarchy

* Game
  + Parameters
    - Clients Supported
    - Number of players supported
  + SpaceMap
    - SpacePorts
      * Paths
        + Distance
        + Target
      * Ships
        + Range
        + Cargo

Capacity

* + - * Commodities
        + Name
        + Price
        + Value
        + Quantity
    - Characters
      * Ships
        + Range
        + Cargo Capacity
      * Credits
      * Commodities
        + Name
        + Price
        + Value
        + Quantity
  + Player
    - History
    - Character
      * Ship
        + Range
        + Cargo

Capacity

Commodities

Name

Price

Value

Quantity

* + - * Credits

## Prototype Details

Support for player trading and chat

Support for NPC encounters (trading)

Minimum of 4 spaceports

Minimum of 9 paths

Support for buying and selling Commodities

Support for upgrading and downgrading ships