Сетевая архитектура состоит из одного сервера и нескольких клиентов.

После создания матча запускается сервер и к нему подключаются все клиенты (игроки), в том числе и игрок, который создал сервер.

Общая структура похожа на структуру облачного гейминга.

Сервер выполняет:

1) Чтение и анализ коммант от клиентов (игроков).

1) Вычисление всех свойств всех объектов в каждый момент времени.

2) Подсчёт очков во время раунда.

3) Выбор и генерация следующей карты.

4) Принятие решения о начале/конце раунда/матча.

5) Принатие решения отдельно для каждого игрока о его дальнейшем участии в матче.

Клиент выполняет:

1) Чтение всех свойств всех объектов.

2) Рендеринг всех объектов.

3) Рендеринг меню подсчёта очков.

Для общения со всеми клиентами сервер использует только один поток даллее «Поток работы с клиентами». Сервер общается с клиентом передавая ему различные команды. Все события передаются от сервера к клиенту ввиде команд. Через некоторые промежутки времени (больше времени одного тика), клиент синхронизируется с сервером.

Задача сервера передать такой список команд клиенту, который создаст на его стороне картину аналогичную картине на сервере.

Список команд передаваемый от сервера к клиенту:

1. Spawn <EntityType> <EntityProperties> – Создаёт новую сущность.
2. Delete <EntityType> <EntityId> – Удаляет указанную по id сущность.
3. SetHistoryList <HistoryList> <add | replace | clear> – Устанавливает список событий.
4. EndRound – Завершает текущий раунд.
5. EndMatch <WinnerId> – Завершает текущий матч.
6. StartRound <MapId> – Запускает новый раунд.
7. StartMatch – Запускает новый матч.
8. Update <EntityType> <EntityId> <EntityProperties> – Обновляет укаранную по id сущность.
9. UpdatePlayerFlags <PlayerId> <PlayerFlags> – Обновляет флаги указанного игрока.
10. UpdateGlobalTimer <NewTime> – Устанавливает глобального значение таймера.

Список команд передаваемый от клиента к серверу:

1. NeedUpdate – запрашивает состояние всех сущьностей.
2. UpdatePlayerFlagsOnServer <Flags> – Обновляет флаги на сервере.
3. SetTeam <TeamId> – Устанавливает команду, во время создания лобби.

Список аргументов:

1. <EntityType> – Идентификатор типа сущности.
2. <EntityProperties> – Свойства сущности.
3. <EntityId> – Номер сущности.
4. <WinnerId> – Номер игрока победителя.
5. <MapId> – Номер карты.
6. <PlayerId> – Номер игрока.
7. <PlayerFlags> – Флаги состояния игрока (вращение | ускорение | стрельба)
8. <NewTime> – Счётчик времени.

Все карты, события, текстуры, проч. хранятся на сервере и у клиента.

**Порядок выполнения команд на сервере во время создания лобби.**

Сервер ожидает подключения новых клиентов.

При подключении нового клиента ему даётся номер. 0-й номер даётся хосту.

Сервер ожидает от клиентов команд на изменение их номера команды. А так-же о готовности к началу матча. Если ве клиенты готовы, матч начинается незамедлительно. В противном случае по истечению таймера.

Сервер посылает клиентам информацию об успешном изменении их номера команды.

**Порядок выполнения команд на спервере во время матча.**

Сервер содержит 4 потока, которые работают асинхронно.

1. Поток работы с клиентами.
2. Поток пересчёта физики.
3. Поток отрисовки.
4. Поток вывода информации в консоль.

Каждый поток выполняет определённый список команд.

**Поток работы с клиентами:**

1. Принятие запроса от клиента.
2. Отправка ответа клиенту.

**Поток пересчёта:**

1. Пересчёт всех объектов.

**Поток отрисовки:**

1. Отрисовка всех объектов.

**Поток вывода в консоль.**

1. Вывод текстовой информации в консоль.

**Протокол общения сервера с клиентом.**

Клиент первым посылает серверу запрос. Длинна запроса не ограничена.

В запросе указываются:

1. ID клиента
2. Длинна сообщения
3. Список сообщений

**Список сообщений от клиента.**

1. Запрос обновления определёных сущьностей
2. Запрос обновления всех сущьностей
3. Запрос на перезапись ввода пользователя

Затем сервер посылает клиенту ответ. Ответ начинается со времени его отправки.

**Список сообщений от сервера.**

1. Создание определённой сущности.
2. Удаление определённой сущности.
3. Окончание раунда.

**Порядок выполнения комантд на сервере во время окончания раунда.**

Сервер посылает всем клиентам сообщение об окончание раунда. Затем ожидает от клиентов сообщение о готовности к следующему раунду.

**В сообщение содержится:**

1. Заголовок сообщения.
2. Очки до окончания раунда.
3. Список событий, которые произошли во время раунда.
4. Информацию о продолжение матча.
5. Номер следующей карты. Если матч будет продолжен.
6. Номер победителя, если матч будет заверщён.