

АНАЛИТИЧЕСКИЙ ОБЗОР МНОГОПОЛЬЗОВАТЕЛЬСКИХ РЕШЕНИЙ ДЛЯ VR РАЗРАБОТКИ

Кейних Д.Д.¹, Видман В. В.²

¹ТПУ ИШИТР ОИТ, студент гр. 8ПМ22, e-mail: ddk10@tpu.ru

²ТПУ ИШИТР ОИТ старший преподаватель, e-mail: vidman@tpu.ru

Аннотация

Данная работа посвящена обзору сетевых решений для многопользовательских тренажеров в виртуальной реальности с целью улучшения процесса обучения, увеличения качества и скорости усвоения материала, а также снижения затрат на обучение.

Ключевые слова: сетевая разработка, Unity, многопользовательские приложения.

Введение

В современном мире виртуальная реальность (VR) становится все более широко используемой в различных областях, включая образование, медицину, военное дело, промышленность и развлечения. Одной из наиболее перспективных областей применения VR является обучение и тренировка, где виртуальная среда может эффективно моделировать различные сценарии и ситуации для подготовки специалистов. Для достижения максимальной эффективности обучения в VR необходимы многопользовательские тренажеры, которые позволяют нескольким участникам совместно участвовать в обучающих сценариях. Для этого требуются специализированные сетевые решения, которые обеспечивают связь между участниками виртуального обучения.

Цель данного исследования заключается в анализе существующих сетевых решений для многопользовательских VR тренажеров с целью выявления их преимуществ и недостатков, а также определения наиболее перспективных направлений развития.

Требования к проекту

На базе Томского политехнического университета ведется разработка VR – тренажеров с возможностью кооперативного режима, к которым выявлены следующие требования:

1. Быстрая интеграция. Интеграционные инструменты и документация должны быть доступны и понятны для разработчиков всех уровней.
2. Количество активных пользователей 2-200. Количество одновременно активных пользователей может варьироваться в зависимости от цели использования:
 - Совместное прохождение VR-тренажера – 2-3 человека;
 - Проведение занятий в виртуальной реальности – 16 человек. Согласно собственным наблюдениям, группа студентов состоит максимум из 30 человек, соответственно, подгруппа состоит из 15 человек и одного преподавателя.
 - Проведение потоковых лекций до 200 человек.
3. Голосовой чат. Система должна включать в себя функциональность голосового чата, позволяющую пользователям общаться в реальном времени во время обучения или тренировки. Голосовой чат должен быть интегрирован непосредственно в приложение и обеспечивать надежную передачу звука без значительной задержки.
4. Возможность создания параллельных серверов. Система должна поддерживать создание и управление несколькими параллельными серверами для обеспечения распределенной обработки нагрузки и повышения отказоустойчивости. Параллельные сервера должны быть легко настраиваемыми и интегрируемыми в общую систему управления и мониторинга.
5. Гибкость и расширяемость. Система должна быть гибкой и расширяемой, позволяя разработчикам легко добавлять новые функции и модули. API и SDK должны быть хорошо документированы и обеспечивать доступ к основным функциональным возможностям системы для создания собственных расширений и интеграций.
6. Поддержка различных платформ. Система должна поддерживать различные платформы и устройства, включая различные VR-гарнитуры, компьютеры и мобильные устройства. Клиентские библиотеки и SDK должны быть доступны для всех основных операционных систем и устройств.

Обзор решений

В настоящее время на рынке представлено много сетевых библиотек для создания многопользовательского режима среди которых есть актуальные и устаревшие, коммерческие и открытые, низкоуровневые и высокоуровневые.

В данной главе рассмотрим самые популярные решения: Photon PUN, Mirror, Darkrift 2, Fishnet и MLAPI. Каждое из этих решений имеет свои особенности, преимущества и недостатки, а также предлагает различные подходы к реализации сетевой части приложений. Достоинства и недостатки по каждому сетевому решению представлено в таблице 1.

Таблица 1

Сравнительная таблицы сетевых библиотек

Название	Достоинства	Недостатки
Photon PUN	<ul style="list-style-type: none">• Быстрая интеграция• Обширная документация• Голосовой чат• Высокоуровневые абстракции• Обратные вызовы	<ul style="list-style-type: none">• Условно бесплатный• 1 вид подключения (Client)• Ограничение на количество активных пользователей• Невозможно запустить несколько параллельных сервисов
Mirror	<ul style="list-style-type: none">• Бесплатный• Быстрая интеграция• Хорошая документация• Интеграция с библиотеками голосового чата• 3 вида подключения (Host, Server, Client)• Отсутствие ограничений для личного и коммерческого использования• Отсутствие ограничений на количество активных пользователей• Большое количество уровней синхронизации• Создание параллельных серверов на виртуальных машинах	<ul style="list-style-type: none">• Сложность интеграции с другими сервисами
Darkrift 2	<ul style="list-style-type: none">• Бесплатный, с возможностью приобрести исходный код• Низкоуровневость• Интеграция с библиотеками голосового чата• Отсутствие ограничений для личного и коммерческого использования• Отсутствие ограничений на количество активных пользователей• Создание параллельных серверов на виртуальных машинах	<ul style="list-style-type: none">• Медленная интеграция• Слабая документация• 1 вид подключения (Client)

Fishnet	<ul style="list-style-type: none"> • Бесплатный, с возможностью перехода на версию Pro • Хорошая документация • Отсутствие ограничений для личного и коммерческого использования • Отсутствие ограничений на количество активных пользователей • 3 вида подключения (Host, Server, Client) • Создание параллельных серверов на виртуальных машинах 	<ul style="list-style-type: none"> • Некоторые необязательные функции доступны в платной версии
MLAPI	<ul style="list-style-type: none"> • Бесплатный • 3 вида подключения (Host, Server, Client) • Хорошая документация • Отсутствие ограничений для личного и коммерческого использования 	<ul style="list-style-type: none"> • Частично устарел

Photon PUN – инструмент с большим количеством сервисов, хорошо работающих друг с другом, а также большой документацией и хорошей поддержкой. Его просто использовать и внедрять, что делает его одним из самых быстрых по интеграции в проект.

Darkrift 2 – мощный инструмент, который позволяет настроить сервер и передачу данных через него как это необходимо проекту, однако его сложность использования и скорость интеграции не позволяют в быстрые сроки внедрить его в проект и использовать его весь функционал. Также отсутствие других видов подключения не позволяют его использовать без автономного сервера.

Fishnet – молодой и развивающийся инструмент для внедрения кооперативного режима в свои проекты, но является полукommerческим проектом и не позволяет на бесплатной версии использовать весь функционал пакета.

MLAPI - решение появившееся из NetCode от Unity. Удобный и практичный инструмент, но частично устарел.

Пакет Mirror быстрый, мощный и настраиваемый, исходя из этого, подходит для использования на большинстве проектов.

По результатам таблицы 1 построена блок-схема, представленная на рисунке 1, с отличительными особенностями для быстрого выбора сетевой библиотеки.

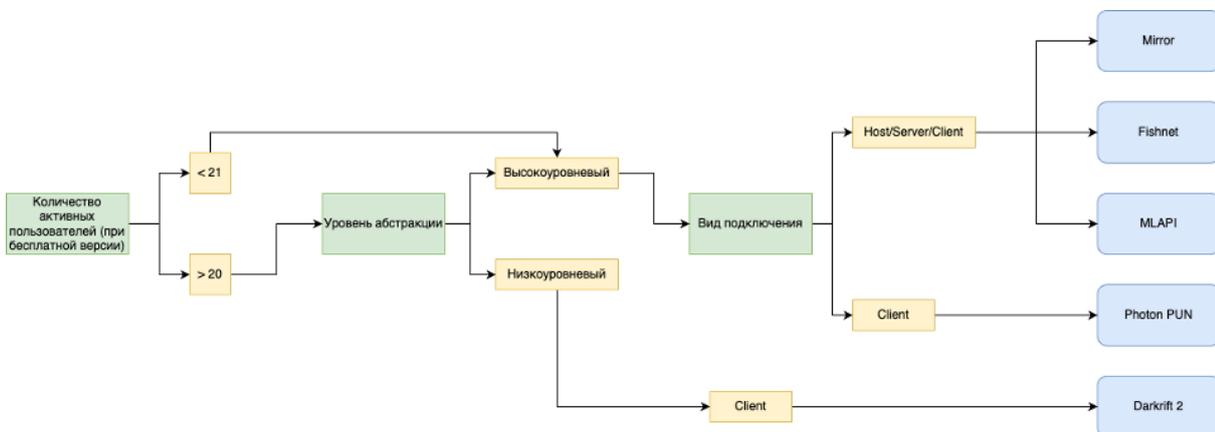


Рис. 1. Блок-схема выбора сетевой библиотеки

Прототипирование

При наложении требований на сетевые библиотеки выбраны 2 решения, которые подходят для быстрого старта: Photon PUN и Mirror. Они имеют схожий функционал, но отличаются внедрением и использованием.

Для проверки решения созданы два прототипа и проверены требования. Результаты проверки сетевых решений представлены в таблице 2.

Таблица 2

Сравнительная таблица Photon PUN и Mirror

Требование	Photon PUN	Mirror
Скорость интеграции	Данный пакет можно поставить простой установкой через PackageManager и простым добавлением компонентов на сцену проекта	Данный пакет можно поставить простой установкой через PackageManager и простым добавлением компонентов на сцену проекта
Количество активных пользователей	Бесплатно до 20 пользователей с использованием Photon Cloud. С увеличением количества пользователей необходимо доплатить за CCU (активных пользователей) и трафик. Также при развертывании собственного сервера необходимо переходить новый тарифный план Photon Server, в котором отсутствует бесплатное использование	Ограничивается только мощностями сервера
Голосовой чат	Имеет собственное решение Photon Voice, но Photon Voice также имеет ограничение на количество пользователей	Не имеет собственного решения, но легко интегрируется со многими сервисами, например, Dissonance или UniVoice
Возможность создания параллельных серверов	Photon Cloud позволяет создать несколько параллельных облаков, но имеет ограничение на общее количество пользователей. Photon Server не имеет ограничение на количество собственных серверов и пользователей	Не имеет ограничение на использование на нескольких параллельных серверах
Открытость исходного кода	Исходный код частично закрыт для просмотра и редактирования	Исходный код полностью открыт

Поддержка различных платформ	Необходимость дополнительной настройки для VR гарнитур при использовании Photon Server	Ограничений не имеет
Гибкость и расширяемость	Легко расширяется, но без возможности внесения изменений в исходный код	Легко расширяется и обновляется

По итогам сравнительной таблицы 2 для проекта, разрабатываемого на базе Томского политехнического университета, где требуется легко масштабируемое количество пользователей с возможностью разделения трафика на нескольких серверах, возможностью модифицирования и быстрого расширения, а также поддержки различных платформ, в большей степени подходит пакет Mirror.

Заключение

В ходе данного обзора были рассмотрены несколько популярных сетевых решений для создания многопользовательских VR тренажеров: Photon PUN, Mirror, Darkrift 2, Fishnet и MLAPI. Каждое из этих решений обладает своими особенностями и преимуществами, что делает их подходящими для различных типов проектов и задач.

Выбранные сетевые решения, Photon PUN и Mirror, представляют собой мощные инструменты для создания многопользовательских приложений в виртуальной реальности. Photon PUN обеспечивает удобное развертывание и масштабирование сетевых приложений, а также предоставляет инструменты для управления соединениями и обменом сообщениями между клиентами. Однако, его ограничения в настройке и контроле могут быть недостаточными для проектов с особыми требованиями. Mirror, в свою очередь, предлагает более гибкий подход к созданию сетевых приложений, позволяя разработчикам полностью контролировать сетевую архитектуру и процесс обмена данными. Это делает Mirror более подходящим для проектов с уникальными требованиями к сетевой части приложения, такими как сложные сценарии виртуального взаимодействия и асинхронная сетевая логика.

По итогу, как Photon PUN, так и Mirror представляют собой значимые сетевые решения для разработки многопользовательских VR тренажеров, и выбор зависит от конкретных потребностей и целей проекта, но исходя из поставленных требований и выявленных ограничений двух библиотек - Mirror является более предпочтительным решением для проекта, разрабатываемого на базе Томского политехнического университета.

Список использованных источников

1. Photon Engine // Photon Engine: сайт. – 2024. – URL: <https://doc.photonengine.com/pun/current/getting-started/pun-intro> (дата обращения 27. 03. 2024).
2. Mirror Networking // Mirror: сайт. – 2024. – URL: <https://mirror-networking.gitbook.io/docs/> (дата обращения 27. 03. 2024).
3. Darkrift 2 // Darkrift 2: github. – 2024. – URL: <https://github.com/DarkRiftNetworking/DarkRift> (дата обращения 27. 03. 2024).
4. Fishnet // Fishnet: сайт. – 2024. – URL: <https://fish-networking.gitbook.io/docs/> (дата обращения 27. 03. 2024).
5. MLAPI // MLAPI: сайт. – 2024. – URL: <https://docs-multiplayer.unity3d.com/netcode/current/about/> (дата обращения 27. 03. 2024).