

Ganache, мы задаём адрес RPC сервера локального узла в файл конфигурации Truffle – truffle.js (в корневом каталоге). После этого можно переходить к миграции контрактов в блокчейн и использованию их функций из блокчейна.

Командой `truffle console` мы подключаемся к локальному узлу, а с помощью `migrate` – передаём все контракты в блокчейн. При успешном выполнении транзакции мы получаем адрес нашего контракта в блокчейне. При использовании фреймворка Web3.JS он нам нужен как ссылка на созданный контракт, но, т.к. мы используем Truffle, мы можем обратиться к контракту с помощью функций без использования адреса:

```
> VoteSystem.deployed().then(function(instance){ return instance.vote(0); });
```

Таким образом, мы можем обратиться к загруженному контракту VoteSystem и вызвать у него метод `vote` с параметром `0`. Выполнив эту команду, мы голосуем за первый по счёту или нулевой по позиции элемент в голосовании. При повторной попытке проголосовать мы получим ошибку, так как для голосования необходимо, чтобы флаг голосования аккаунта был равен `false`, а после одного удачно отданного голоса он становится `true`. Таким же образом выполняются и другие функции контракта.

Следует отметить, что разработка DAPP не является чем-то сверхъестественным, это очень интересное и многообещающее направление не только в экономической сфере – для безопасного трансфера эфира – но и в социальной – для создания многоуровневых приложений. Если Ethereum продолжит развиваться в таких же темпах и функционал Solidity продолжит расти, то скоро может появиться новая специализация программистов – разработчик децентрализованных приложений.

**Е.М. Хомяков** (ГГУ имени Ф. Скорины, Гомель)

Науч. рук. **В.Н. Леванцов**, ст. преподаватель

## **ОПИСАНИЕ И ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ СЕТЕВОГО ПРОТОКОЛА HTTP/2**

HTTP/2 – вторая версия сетевого протокола прикладного уровня передачи данных, используемая для доступа к Всемирной паутине. Данный протокол основан на SPDY (протокол прикладного уровня для передачи веб-контента). Протокол HTTP/2 был разработан группой Hypertext Transfer Protocol working group компании Internet Engineering Task Force. Протокол был представлен группой на рассмотрение в декабре 2014, и уже 17 февраля 2015 он был утвержден компанией IETF. В мае того же года спецификация данного протокола была опубликована как RFC 7540.

Действительно, протокол HTTP/1.1, используется повсеместно почти 20 лет, и его обновление – это было лишь вопросом времени. В отличие от своего предшественника, HTTP/2 – бинарный, а не текстовый. Поэтому он более эффективен при парсинге, а также более компактен при передаче. Решение о том, чтобы сделать протокол бинарным, было сделано для того, чтобы формирование пакетов проходило легче и проще. В частности, это позволило проще разделять части, которые связаны с протоколом, и те части, которые связаны с пакетом данных.

Основная ставка в данной разработке была на мультиплексировании. Если раньше пакеты множества потоков поставлялись отдельно друг от друга, то в новом протоколе HTTP/2 пакеты в рамках одного соединения смешаны, а их разделение происходит уже на другой стороне. При этом каждый поток имеет свой собственный приоритет для того, чтобы можно было понять, какие потоки считаются более важными, а какие – менее важными. Это актуально в тех ситуациях, когда есть ограничения ресурсов, и сервер вынужден выбирать потоки, которые будут отправлены в первую очередь.

Протокол HTTP построен таким образом, что при отправке запросов также передаются заголовки, которые содержат дополнительную информацию. Сервер, в свою очередь, также прикрепляет заголовки к ответам. Следовательно, все заголовки на веб-страницах могут занимать достаточно большой объем. В связи с этим, в HTTP/2 присутствует сжатие заголовков, которое позволит существенно сократить объем вспомогательной информации, так что браузер сможет отправить все запросы сразу.

Протокол HTTP/2 не требует шифрования канала, однако, может работать как для HTTP URI (то есть без шифрования), так и для HTTPS URI, но только вместе с TLS (протокол защиты веб-страниц, который заменил собой SSL).

Рассмотрим сравнение среднего времени загрузок страницы на примере реального проекта (рисунок 1).

Доступ через протокол версии	Среднее время загрузки страницы
HTTP 1.x без оптимизации	5.44 сек
HTTP 1.x с включенной оптимизацией: *склейка файлов*, *доменное шардирование*	5.07 сек
SPDY/3.1	4.77 сек
HTTP/2	3.91 сек

Рисунок 1 – Сравнение среднего времени загрузок страницы

Протокол HTTP/2 значительно оптимизирован, по сравнению с HTTP/1.1, так что внедрение данной спецификации способно значительно улучшить производительность веб-сервисов.

## Литература

1. AniArt [Электронный ресурс] / Оптимизация в HTTP/2 – Режим доступа: [https://www.aniart.com.ua/blog/detail/ssl\\_2/](https://www.aniart.com.ua/blog/detail/ssl_2/) [Дата обращения: 29 марта 2018].
2. Web-студия «SAMBUSA» [Электронный ресурс] / Ускорение сайтов с HTTP/2 – Режим доступа: <https://sambusa.ru/blog/uskorenie-sajtov-s-http2/> [Дата обращения: 29 марта 2018].
3. Свободная энциклопедия «Википедия» [Электронный ресурс] / HTTP/2 – Режим доступа: <https://ru.wikipedia.org/wiki/HTTP/2> [Дата обращения: 29 марта 2018].

**М.А. Чашев** (ГГУ имени Ф. Скорины, Гомель)  
Науч. рук. **С.Ф. Маслович**, канд. техн. наук, доцент

### РАЗРАБОТКА МОБИЛЬНОГО IOS-ПРИЛОЖЕНИЯ С ИСПОЛЬЗОВАНИЕМ ARKIT FRAMEWORK

В современном мире уже практически невозможно представить свою жизнь без такой мелочи, как мобильный телефон. Это устройство первоначально создавалось только с одной целью – поддерживать связь между людьми, независимо от расстояния между ними. Но прогресс не стоял на месте.

Разработка приложений для мобильных операционных систем является актуальным перспективным делом. Оно позволяет узнать многие особенности создания приложений для портативных устройств, получить опыт при проектировании такого рода приложений.

Цель работы реализация приложения с помощью современных технологий, в котором будет продемонстрирован функционал и возможности такого фреймворка как Arkit и в целом возможностей дополненной реальности на примере создания трёхмерного текста в пространстве.

**А.С. Чеботаревский, Е.А. Левчук** (ГГУ имени Ф. Скорины, Гомель)  
Науч. рук. **Е.А. Левчук**, канд. техн. наук, доцент

### РАЗРАБОТКА БИРЖЕВОЙ ПЛОЩАДКИ ЛОГИСТИЧЕСКИХ УСЛУГ MOVEME.PRO

Постановка задачи включает в себя разработку сервиса с автоматизированными процессами грузоперевозок и эвакуаций автомобилей