Блокчейн Aptos: безопасный, масштабируемый и обновляемый Web3 Инфраструктура

11 августа 2022 г. v1.0

Абстрактный

Рост блокчейнов как новой интернет-инфраструктуры привел к тому, что разработчики быстро развернули десятки тысяч децентрализованных приложений. К сожалению, использование блокчейна еще не повсеместно из-за частых сбоев, высоких затрат, низких ограничений пропускной способности и многочисленных проблем с безопасностью. Чтобы обеспечить массовое внедрение в эпоху web3, инфраструктура блокчейна должна следовать по пути облачной инфраструктуры как надежной, масштабируемой, экономичной и постоянно улучшающейся платформы для создания широко используемых приложений.

Мы представляем блокчейн Aptos, разработанный с учетом масштабируемости, безопасности, надежности и возможности обновления в качестве ключевых принципов для решения этих задач. Блокчейн Aptos был разработан за последние три года более чем 350 разработчиками по всему миру [1]. Он предлагает новые и новаторские инновации в области консенсуса, дизайна смарт-контрактов, системной безопасности, производительности и децентрализации. Комбинация этих технологий обеспечит фундаментальный строительный блок для продвижения web3 в массы:1

- Во-первых, блокчейн Aptos изначально интегрирует и использует язык Move для быстрого и безопасного выполнения транзакций [2]. Доказательство Move, формальный верификатор для смарт-контрактов, написанный на языке Move, обеспечивает дополнительные гарантии для инвариантов и поведения контрактов. Такое внимание к безопасности позволяет разработчикам лучше защищать свое программное обеспечение от вредоносных объектов.

- Во-вторых, модель данных Aptos обеспечивает гибкое управление ключами и варианты гибридного хранения. Это, наряду с прозрачностью транзакций до подписания и практичными облегченными клиентскими протоколами, обеспечивает более безопасный и надежный пользовательский интерфейс.

- В-третьих, для достижения высокой пропускной способности и низкой задержки в блокчейне Aptos используется конвейерный и модульный подход для ключевых этапов обработки транзакций. В частности, распространение транзакций, блочное упорядочение метаданных, параллельное выполнение транзакций, пакетное хранение и сертификация реестра выполняются одновременно. Этот подход полностью использует все доступные физические ресурсы, повышает эффективность оборудования и обеспечивает высокопараллельное выполнение.

- В-четвертых, в отличие от других механизмов параллельного выполнения, которые нарушают атомарность транзакций, требуя предварительного знания данных для чтения и записи, блокчейн Aptos не накладывает таких ограничений на разработчиков. Он может эффективно поддерживать атомарность с произвольно сложными транзакциями, обеспечивая более высокую пропускную способность и меньшую задержку для реальных приложений и упрощая разработку.

- В-пятых, модульная архитектура Aptos поддерживает гибкость клиентов и оптимизирует частые и мгновенные обновления. Кроме того, для быстрого развертывания новых технологических инноваций и поддержки новых вариантов использования web3 блокчейн Aptos предоставляет встроенные протоколы управления изменениями в сети.

1Правовая оговорка: этот технический документ и его содержание не являются предложением о продаже или предложением о покупке каких-либо токенов. Мы публикуем этот технический документ исключительно для получения отзывов и комментариев от общественности. Ничто в этом документе не следует читать или интерпретировать как гарантию или обещание того, как блокчейн Aptos или его токены (если таковые имеются) будут развиваться, использоваться или накапливаться. Aptos лишь излагает свои текущие планы, которые могут меняться по ее усмотрению, и успех которых будет зависеть от многих факторов, находящихся вне ее контроля. Такие будущие заявления обязательно связаны с известными и неизвестными рисками, которые могут привести к тому, что фактическая производительность и результаты в будущих периодах будут существенно отличаться от того, что мы описали или подразумеваем в этом техническом документе. Aptos не берет на себя никаких обязательств по обновлению своих планов. Не может быть никаких гарантий, что какие-либо заявления в официальном документе окажутся точными, поскольку фактические результаты и будущие события могут существенно отличаться. Пожалуйста, не слишком полагайтесь на будущие заявления.

- Наконец, блокчейн Aptos экспериментирует с будущими инициативами по масштабированию за пределы производительности отдельных валидаторов: его модульная конструкция и механизм параллельного выполнения поддерживают внутреннее сегментирование валидатора, а сегментирование однородного состояния обеспечивает потенциал для горизонтального масштабирования пропускной способности, не добавляя дополнительных сложностей для операторов узлов. .

1. Введение

В версии Интернета web2 такие услуги, как обмен сообщениями, социальные сети, финансы, игры, покупки и потоковое аудио/видео, предоставляются централизованными компаниями, которые контролируют прямой доступ к пользовательским данным (например, Google, Amazon, Apple, и Мета). Эти компании разрабатывают инфраструктуру с помощью программного обеспечения для конкретных приложений, оптимизированного для целевых сценариев использования, и используют облачные инфраструктуры для развертывания этих приложений для пользователей. Облачная инфраструктура обеспечивает доступ к виртуализированным и/или физическим службам инфраструктуры, таким как арендованные виртуальные машины (ВМ) и аппаратное обеспечение без операционной системы, работающее в центрах обработки данных по всему миру (например, AWS, Azure и Google Cloud). В результате создание интернет-сервисов web2, которые могут масштабироваться до миллиардов пользователей, никогда не было проще, чем сегодня. Однако web2 требует, чтобы пользователи явно доверяли централизованным объектам, требование, которое становится все более важным для общества.

Чтобы справиться с этой проблемой, началась новая эра Интернета: web3. В версии Интернета web3 появились блокчейны, обеспечивающие децентрализованные неизменяемые реестры, которые позволяют пользователям безопасно и надежно взаимодействовать друг с другом, не требуя доверия к контролирующим посредникам или централизованным организациям. Подобно тому, как интернет-сервисы и приложения web2 полагаются на облачную инфраструктуру в качестве строительных блоков, децентрализованные приложения могут использовать блокчейны в качестве уровня децентрализованной инфраструктуры для охвата миллиардов пользователей по всему миру.

Однако, несмотря на существование множества блокчейнов сегодня, широкого распространения web3 еще не произошло [3]. В то время как технологии продолжают развиваться в отрасли, существующие блокчейны ненадежны, требуют высоких комиссий за транзакции для пользователей, имеют низкие ограничения пропускной способности, регулярно теряют активы из-за проблем с безопасностью и не могут поддерживать реакцию в режиме реального времени. По сравнению с тем, как облачная инфраструктура позволила сервисам web2 достичь миллиардов, блокчейны еще не позволили приложениям web3 сделать то же самое.

2 Видение Аптос

Видение Aptos состоит в том, чтобы предоставить блокчейн, который может обеспечить массовое внедрение в web3 и расширить возможности экосистемы децентрализованных приложений для решения реальных проблем пользователей. Наша миссия состоит в том, чтобы продвигать самые современные технологии в области надежности, безопасности и производительности блокчейна, предоставляя гибкую и модульную архитектуру блокчейна. Эта архитектура должна поддерживать частые обновления, быстрое внедрение новейших технологий и первоклассную поддержку новых и появляющихся вариантов использования.

Мы представляем себе децентрализованную, безопасную и масштабируемую сеть, управляемую и управляемую сообществом, которое ее использует. Когда во всем мире растут потребности в инфраструктуре, вычислительные ресурсы блокчейна масштабируются по горизонтали и вертикали, чтобы удовлетворить эти потребности. По мере появления новых вариантов использования и технологических достижений сеть должна часто и беспрепятственно обновляться, не прерывая работу пользователей. Проблемы с инфраструктурой должны отойти на второй план. Разработчики и пользователи будут иметь доступ к множеству различных вариантов восстановления ключей, моделирования данных, стандартов смарт-контрактов, компромиссов в использовании ресурсов, конфиденциальности и возможности компоновки. Пользователи знают, что их активы в безопасности, всегда доступны и могут быть доступны с оплатой, близкой к себестоимости. Любой может безопасно, легко и неизменно совершать сделки с ненадежными сторонами по всему миру. Блокчейны так же распространены, как и облачная инфраструктура.

Чтобы реализовать это видение, необходимо добиться значительного технического прогресса. Наш опыт создания, разработки, продвижения и развертывания блокчейна Diem (предшественника блокчейна Aptos) за последние три года доказал, что сеть может постоянно обновлять свои протоколы, не нарушая работу своих клиентов [4]. В начале 2020 года основная сеть Diem была развернута для более чем дюжины операторов узлов с несколькими поставщиками кошельков. В течение следующего года наша команда выпустила два крупных обновления, которые изменили протокол консенсуса и основную структуру. Оба обновления завершены без простоев для пользователей. С помощью блокчейна Aptos мы внесли ряд радикальных улучшений в стек технологий, а также включили безопасные, прозрачные и частые обновления в качестве основной функции, вдохновленной блокчейном Diem. В частности, мы выделяем новые методы обработки транзакций (описанные в разделе 7) и новые подходы к децентрализации и управлению сетью.

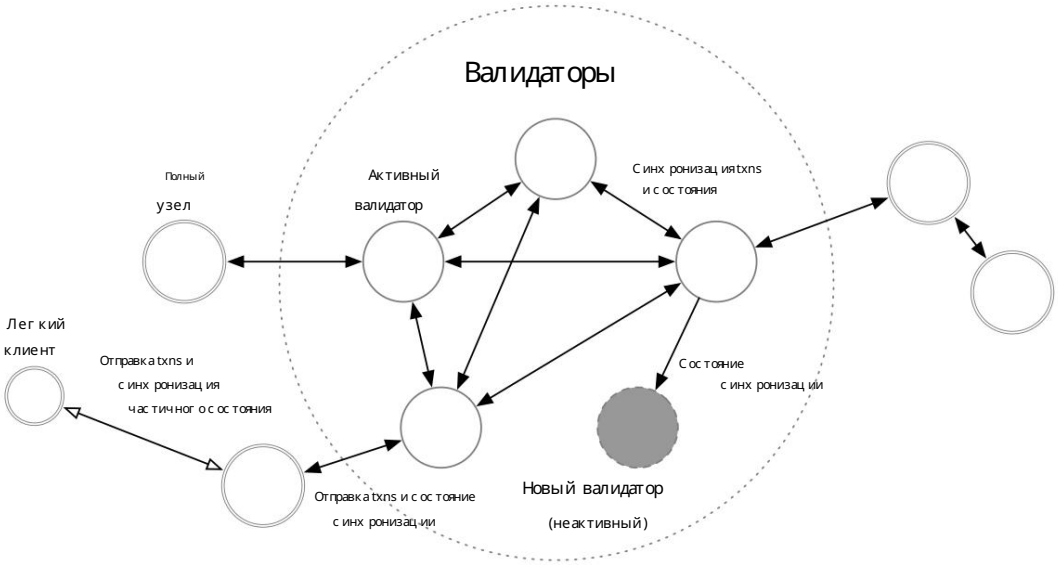

Рисунок 1: Компоненты экосистемы Aptos.

Поскольку блокчейн Aptos продолжает улучшаться и расти, мы будем выпускать обновленные версии этого документа с последними версиями наших протоколов и вариантов дизайна. В остальной части этого документа мы описываем текущее состояние блокчейна Aptos, а также планы на будущее.

3 Обзор

Блокчейн Aptos, как показано на рисунке 1, состоит из набора валидаторов, которые совместно получают и обрабатывают транзакции от пользователей, используя византийский отказоустойчивый (BFT) механизм консенсуса Proof-of-Stake. Владельцы токенов блокируют токены или делают ставки в выбранных ими валидаторах. Консенсусный вес голосования каждого валидатора пропорционален сумме, вложенной в него. Валидатор может быть активным и участвовать в консенсусе. Аналогичным образом, валидатор также может быть неактивным, если у него недостаточно доли для участия, он выходит из набора валидаторов, выбирает автономный режим, поскольку он синхронизирует состояние блокчейна, или протокол консенсуса считает его неучаствующим из-за плохой исторической производительности.

Клиенты — это любая часть системы, которой необходимо отправлять транзакции или запрашивать состояние и историю блокчейна. Клиенты могут выбрать загрузку и проверку подтверждений запрошенных данных, подписанных валидатором.
Полные узлы — это клиенты, которые копируют транзакцию и состояние блокчейна от валидаторов или других полных узлов в сети. Они могут по своему усмотрению обрезать историю транзакций и состояние блокчейна, чтобы восстановить хранилище. Легкие клиенты поддерживают только текущий набор валидаторов и могут безопасно запрашивать частичное состояние блокчейна, как правило, из полных узлов. Кошельки — типичный пример легкого клиента.

Чтобы удовлетворить потребности в безопасной, быстрой, надежной и обновляемой инфраструктуре web3 для широкого внедрения Блокчейн Aptos построен на следующих основных принципах проектирования:

- Быстрое и безопасное выполнение, а также простая возможность аудита и механического анализа благодаря новому языку программирования смарт-контрактов Move [5]. Движение началось с предшественника блокчейна Aptos и продолжает развиваться вместе с развитием этого проекта.
- Чрезвычайно высокая пропускная способность и низкая задержка благодаря пакетной, конвейерной и параллельной точкам доступа. подход к обработке транзакций.
- Новая параллельная обработка транзакций, которая эффективно поддерживает атомарность с произвольными сложными транзакциями посредством Block-STM, в отличие от существующих механизмов параллельного выполнения, которым требуется предварительное знание местоположений данных для чтения и записи.
- Оптимизация производительности и децентрализации за счет быстрой ротации наборов валидаторов с весом доли. и отслеживание репутации.

- Возможность обновления и конфигурируемости как первоклассные принципы проектирования, учитывающие новые варианты использования и новейшие технологии.
- Модульная конструкция, обеспечивающая тщательное тестирование на уровне компонентов, а также соответствующее моделирование угроз и беспрепятственное развертывание, обеспечивает высокую безопасность и надежность операций.
- Горизонтальное масштабирование пропускной способности при сохранении децентрализации, где сегментирование является первоклассным концепция, открытая для пользователей и родная для программирования и модели данных.

В разделе 4 объясняется, как разработчики взаимодействуют с Move в блокчейне Aptos. Раздел 5 описывает логическую модель данных. В разделе 6 подробно описано, как блокчейн Aptos обеспечивает безопасный пользовательский опыт с помощью надежных методов проверки. В разделе 7 описываются ключевые нововведения в области производительности, связанные с конвейерной обработкой, пакетной обработкой и распараллеливанием. В разделе 8 подробно описаны различные варианты для разных клиентов для синхронизации состояния с другими узлами. В Разделе 9 описываются наши планы в отношении владения и управления сообществом. Наконец, в Разделе 10 обсуждаются будущие направления деятельности при сохранении децентрализации.

4 Язык перемещения

Move — это новый язык программирования смарт-контрактов с упором на безопасность и гибкость. Блокчейн Aptos использует объектную модель Move для представления состояния реестра (см. Раздел 5.5) и использует код (модули) Move для кодирования правил перехода состояний. Пользователи отправляют транзакции, которые могут публиковать новые модули, обновлять существующие модули, выполнять функции входа, определенные в модуле, или содержать сценарии, которые могут напрямую взаимодействовать с общедоступными интерфейсами модулей.

Экосистема Move содержит компилятор, виртуальную машину и множество других инструментов разработчика.
Move вдохновлен языком программирования Rust, который делает владение данными явным в языке с помощью таких концепций, как линейные типы. Move подчеркивает нехватку ресурсов, сохранение и контроль доступа. Модули перемещения определяют время жизни, хранилище и схему доступа для каждого ресурса. Это гарантирует, что такие ресурсы, как Coin, не будут производиться без соответствующих учетных данных, не могут быть потрачены дважды и не исчезнут.

Move использует верификатор байт-кода, чтобы гарантировать безопасность типов и памяти даже при наличии ненадежного кода. Чтобы помочь писать более надежный код, Move включает в себя формальный верификатор, Move Prover [6], способный проверять функциональную правильность программы Move по заданной спецификации, сформулированной на языке спецификаций, интегрированном в Move.

Помимо учетных записей пользователей и соответствующего содержимого учетной записи, состояние реестра также содержит конфигурацию цепочки блоков Aptos. Эта сетевая конфигурация включает в себя набор активных валидаторов, свойства стейкинга и конфигурацию различных сервисов в блокчейне Aptos.
Поддержка Move для возможности обновления модуля и комплексной программируемости обеспечивает беспрепятственное изменение конфигурации и поддерживает обновления самой цепочки блоков Aptos (оба набора обновлений выполнялись несколько раз без простоев в частной основной сети).

Команда Aptos усовершенствовала Move, добавив поддержку более широких вариантов использования web3. Как упоминалось далее в разделе 5.5, блокчейн Aptos обеспечивает детальное управление ресурсами. Это не только поддерживает распараллеливание выполнения, но также обеспечивает почти фиксированную стоимость, связанную с доступом к данным и их изменением. Кроме того, блокчейн Aptos обеспечивает поддержку таблиц, построенную поверх мелкозернистого хранилища, что позволяет хранить крупномасштабные наборы данных (например, массивные коллекции NFT) в одной учетной записи.
Кроме того, Aptos поддерживает общие или автономные учетные записи, которые полностью представлены в сети.
Это позволяет сложным децентрализованным автономным организациям (DAO) совместно использовать учетные записи, а также использовать эти учетные записи в качестве контейнеров для разнородного набора ресурсов.

5 Логическая модель данных

Состояние реестра блокчейна Aptos представляет собой состояние всех учетных записей. Состояние реестра определяется с помощью 64-битного целого числа без знака, соответствующего количеству транзакций, выполненных системой. Любой может отправить транзакцию в блокчейн Aptos, чтобы изменить состояние реестра. После выполнения транзакции генерируется вывод транзакции. Выход транзакции содержит ноль или более операций для управления состоянием реестра (называемых наборами записей), вектором результирующих событий (см. Раздел 5.1.1), количеством потребленного газа и статусом выполненной транзакции.

5.1 Транзакции

Подписанная транзакция содержит следующую информацию:

- Аутентификатор транзакции: отправитель использует аутентификатор транзакции, который включает один или больше цифровых подписей для проверки подлинности транзакции.
- Адрес отправителя: адрес учетной записи отправителя.
- Полезная нагрузка: Полезная нагрузка либо относится к существующей функции входа в цепочке, либо содержит функцию, которая должна быть выполнена как встроенный байт-код (называемый скриптом). Кроме того, набор входных аргументов кодируется в байтовых массивах. Для одноранговой транзакции входные данные содержат информацию о получателе и переданную ему сумму.
- Цена газа (в указанной валюте/единицах газа): это сумма, которую отправитель готов заплатить за единицу газа для выполнения транзакции. Газ — это способ оплаты вычислений, сети и хранилища. Единица газа — это абстрактное измерение вычислений, не имеющее реальной ценности.
- Максимальное количество газа: максимальное количество газа — это максимальное количество единиц газа, которое транзакция может израсходовать до прерывания. Учетная запись должна иметь по крайней мере цену газа, умноженную на максимальное количество газа, иначе транзакция будет отклонена во время проверки.
- Порядковый номер: порядковый номер транзакции. Он должен совпадать с порядковым номером, сохраненным в учетной записи отправителя при выполнении транзакции. При успешном выполнении транзакции порядковый номер учетной записи увеличивается для предотвращения повторных атак.
- Срок действия: Отметка времени, после которой транзакция перестает быть действительной.
- Идентификатор цепочки: идентифицирует блокчейн, для которого действительна эта транзакция, обеспечивая дополнительную защиту. для пользователей, чтобы предотвратить ошибки подписи.

В каждой версии i изменение состояния представлено кортежем (Ti , , Si), содержащим транзакцию Ti, ее код транзакции Oi и состояние реестра Si. Oi и Si соответственно. Учитывая детерминированную функцию Apply, выполнение транзакции Ti с состоянием регистра Si-1 создает выходную транзакцию Oi и новое состояние регистра Si . То есть Apply(Si 1, Ti) Oi , Si .

5.1.1 События

События генерируются во время выполнения транзакции. Каждый модуль Move может определять свои собственные события и выбирать, когда создавать эти события при выполнении. Например, во время перевода монет учетные записи отправителя и получателя будут генерировать SentEvent и ReceivedEvent соответственно. Эти данные хранятся в реестре и могут быть запрошены через узел Aptos. Каждое зарегистрированное событие имеет уникальный ключ, который можно использовать для запроса сведений о событии.

Несколько событий, отправленных на один и тот же ключ события, создают поток событий, список событий, каждая запись которого содержит последовательно увеличивающееся число, начиная с 0, тип и данные. Каждое событие должно быть определено некоторым типом. Может быть несколько событий, определенных одним и тем же или подобными типами, особенно при использовании дженериков. События имеют связанные данные. Для разработчиков модуля Move общий принцип заключается в том, чтобы включить все данные, необходимые для понимания изменений в базовых ресурсах до и после выполнения транзакции, которая изменила данные и породила событие.

Транзакции могут только генерировать события и не могут считывать события. Такой дизайн позволяет выполнять транзакцию только в зависимости от текущего состояния и входных данных транзакции, а не от исторической информации (например, от ранее сгенерированных событий).

5.2 Аккаунты

Каждая учетная запись идентифицируется уникальным 256-битным значением, известным как адрес учетной записи. Новая учетная запись создается в состоянии реестра (см. Раздел 5.5) , когда транзакция, отправленная из существующей учетной записи, вызывает функцию перемещения create_account(addr). Обычно это происходит, когда транзакция пытается отправить токены Aptos на адрес учетной записи, который еще не создан. Для удобства Aptos также поддерживает функцию перевода (от, до, сумма), которая неявно создает учетную запись, если она еще не существовала до перевода.

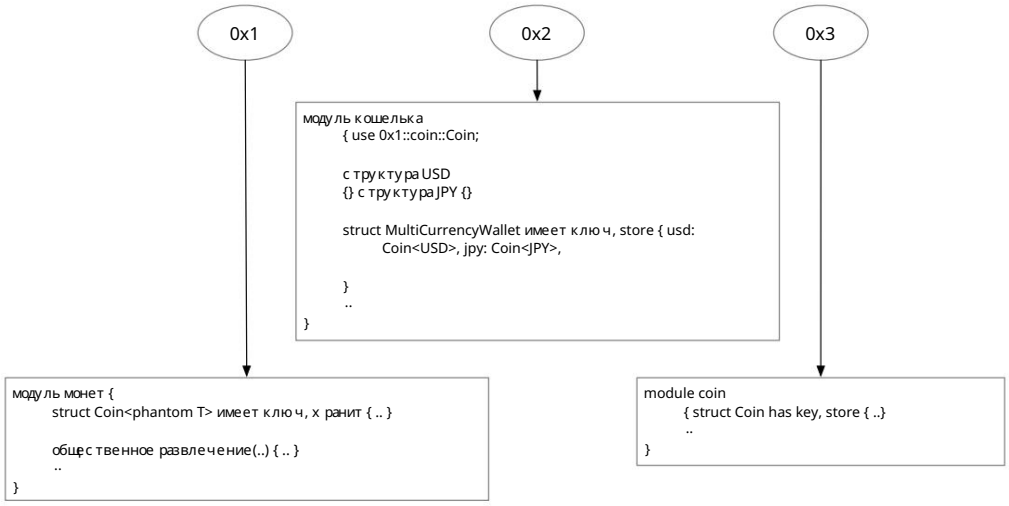

Рисунок 2: Пример модулей перемещения в цепочке.

Чтобы создать новую учетную запись, пользователь сначала генерирует пару ключей подписи: (vk, sk). Затем новый адрес учетной записи для заданной схемы подписи получается с использованием криптографического хэша H открытого ключа проверки vk, который объединяется с идентификатором схемы подписи (ssid): где addr = H(vk, ssid).

После создания новой учетной записи по адресу addr пользователь может подписывать транзакции, которые будут отправлены с учетной записи по адресу addr, используя закрытый ключ подписи sk. Пользователь также может чередовать sk либо для упреждающего изменения sk, либо в ответ на возможную компрометацию. Это не изменит адрес учетной записи, так как адрес учетной записи получается только один раз, во время ее создания, из открытого ключа проверки.

Блокчейн Aptos не связывает учетные записи с реальной личностью. Пользователь может создать несколько учетных записей, сгенерировав несколько пар ключей. Учетные записи, контролируемые одним и тем же пользователем, не имеют внутренней связи друг с другом. Однако один пользователь по-прежнему может управлять несколькими учетными записями в одном кошельке для простого управления активами. Эта гибкость обеспечивает псевдоним для пользователей, пока мы экспериментируем с примитивами сохранения конфиденциальности для будущих выпусков. Несколько учетных записей, принадлежащих одному пользователю или группе пользователей, также предоставляют каналы для увеличения параллелизма выполнения, как описано в Разделе 7.4.

5.3 Перемещение модулей

Модуль Move содержит байт-код Move, который объявляет типы данных (структуры) и процедуры. Он идентифицируется по адресу учетной записи, в которой объявлен модуль, а также по имени модуля. Например, идентификатор первого модуля валюты на рис. 2 — 0x1::coin. Модуль может зависеть от других сетевых модулей, как показано модулем кошелька на рис. 2, что позволяет повторно использовать код.

Модуль должен иметь уникальное имя в рамках учетной записи, т. е. каждая учетная запись может объявить не более одного модуля с любым заданным именем. Например, учетная запись с адресом 0x1 на рис. 2 не может объявить другой модуль с именем coin. С другой стороны, учетная запись по адресу 0x3 может объявить модуль с именем coin, и идентификатор этого модуля будет 0x3::coin. Обратите внимание, что 0x1::coin::Coin и 0x3::coin::Coin являются разными типами и не могут использоваться взаимозаменяемо или иметь общий код модуля. Напротив, 0x1::coin::Coin<0x2::wallet::USD> и 0x1::coin::Coin<0x2::wallet::JPY> — это разные экземпляры одного и того же универсального типа, которые нельзя использовать взаимозаменяемо, но может использовать общий код модуля.

Модули сгруппированы в пакеты, расположенные по одному адресу. Владелец этого адреса публикует пакет целиком в цепочке, включая байт-код и метаданные пакета. Метаданные пакета определяют, можно ли обновить пакет или он является неизменяемым. Для обновляемого пакета проверки совместимости выполняются до того, как будет разрешено обновление: никакие существующие функции точки входа не должны изменяться, и никакие ресурсы не могут храниться в памяти. Однако могут быть добавлены новые функции и ресурсы.

Платформа Aptos, состоящая из основных библиотек и конфигурации для блокчейна Aptos, определяется как обычный обновляемый пакет модулей (см. Раздел 9.2).

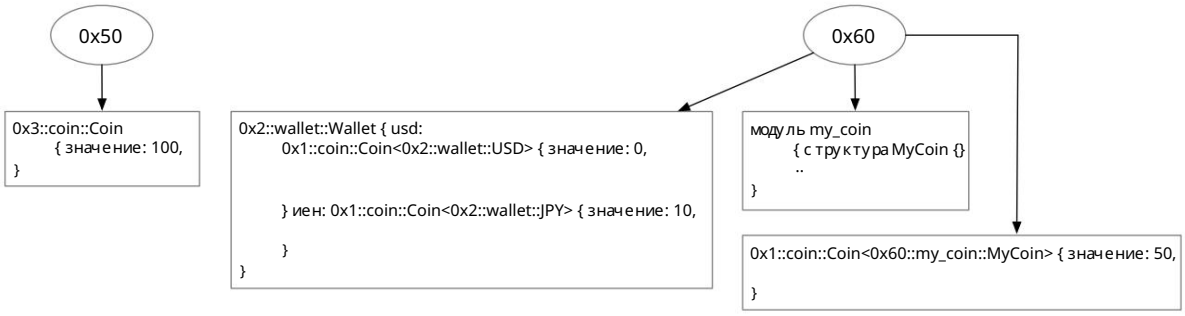

Рисунок 3: Пример данных по цепочке.

5.4 Ресурсы

Подобно модулям, адреса учетных записей также могут иметь связанные с ними значения данных. В каждом адресе учетной записи значения имеют ключи по их типам, при этом не более одного значения каждого типа принадлежит учетной записи. На рис. 3 приведен пример этого. Адрес 0x50 содержит одно значение, при этом 0x3::coin::Coin является полным типом. 0x3 — это адрес, по которому хранится модуль монеты, монета — это имя модуля, а монета — это имя типа данных. Также допускаются значения универсальных типов, при этом разные экземпляры рассматриваются как отдельные типы. Это важно для расширяемости, позволяя различным экземплярам использовать один и тот же функциональный код.

Правила изменения, удаления и публикации значения закодированы в модуле, определяющем тип данных. Правила безопасности и проверки Move не позволяют другому коду или объектам напрямую создавать, изменять или удалять экземпляры типов данных, определенных в других модулях.

Наличие не более одного значения верхнего уровня каждого типа под адресом может на первый взгляд показаться ограничением. Однако на практике это не проблема, поскольку программисты могут определять типы-оболочки с другими данными в качестве внутренних полей, избегая, таким образом, каких-либо ограничений. Структура Wallet на рис. 3 является примером использования типов-оболочек.

Следует также отметить, что не все типы данных могут храниться в цепочке. Чтобы экземпляры данных квалифицировались как значения верхнего уровня, тип данных должен иметь ключевую возможность. Точно так же возможность сохранения требуется для вложенных значений. Типы данных с обеими возможностями также называются ресурсами.

5.5 Состояние книги

С точки зрения виртуальной машины Move (Move VM) каждая учетная запись состоит из набора значений и структур данных "ключ-значение". Эти структуры данных называются записями таблицы и хранятся в формате двоичной канонической сериализации (BCS). Этот макет данных позволяет разработчикам писать смарт-контракты, которые могут эффективно работать с небольшими объемами данных, реплицированных в большом количестве учетных записей, а также с большими объемами данных, хранящихся в небольшом количестве учетных записей. Модули перемещения хранятся аналогично данным учетной записи, но в независимом пространстве имен. Состояние регистрагенезиса определяет начальный набор учетных записей и связанное с ними состояние при инициализации блокчейна.

При запуске блокчейн Aptos будет представлен одним состоянием реестра. Однако по мере распространения и развития технологий Aptos будет увеличивать количество сегментов для увеличения пропускной способности (т. е. включения нескольких состояний реестра) и поддержки транзакций, которые перемещают или получают доступ к активам между сегментами.
Каждое состояние реестра будет поддерживать все активы в цепочке для определенного сегмента и предоставлять ту же модель учетной записи с мелкозернистым хранилищем данных «ключ-значение», предлагающим почти фиксированные затраты на доступ к хранилищу.

6 Безопасный пользовательский интерфейс

Чтобы охватить миллиарды пользователей Интернета, пользовательский интерфейс web3 должен быть безопасным и доступным. В разделах ниже мы описываем несколько инноваций, предоставляемых блокчейном Aptos, которые работают для достижения этой цели.

6.1 Защита жизнеспособности транзакции

Подписание транзакции означает, что подписывающая сторона разрешает транзакцию быть зафиксированной и выполненной блокчейном. Иногда пользователи могут подписывать транзакции непреднамеренно или без полного учета всех возможных способов манипулирования их транзакциями. Чтобы снизить этот риск, блокчейн Aptos ограничивает жизнеспособность каждой транзакции и защищает подписывающую сторону от неограниченной действительности. В настоящее время блокчейн Aptos обеспечивает три различных средства защиты: порядковый номер отправителя, время истечения транзакции и назначенный идентификатор цепочки.

- Порядковый номер транзакции может быть зафиксирован ровно один раз для каждой учетной записи отправителя. В результате отправители могут заметить, что если порядковый номер текущей учетной записи порядкового номера транзакции t, то либо t уже зафиксирован, либо t никогда не будет зафиксирован (поскольку порядковый номер, используемый t, уже использован транзакцией t). другая сделка).
- Время в блокчейне увеличивается с высокой точностью и частотой (обычно с точностью до секунды), как подробно описано в разделе 7.3.1. Если время блокчейна превышает время истечения транзакции t, то аналогичным образом либо t уже зафиксировано, либо t никогда не будет зафиксировано.
- Каждая транзакция имеет назначенный идентификатор цепочки для предотвращения повторного воспроизведения транзакций злоумышленниками между различными средами блокчейна (например, между тестовой сетью и основной сетью).

6.2 Управление ключами на основе перемещения

Как обсуждалось в Разделе 5.2, учетные записи Aptos поддерживают ротацию ключей — важную функцию, которая может помочь снизить риски, связанные с компрометацией закрытого ключа, атаками дальнего действия и будущими улучшениями, которые могут нарушить существующие криптографические алгоритмы. Кроме того, учетные записи Aptos также достаточно гибки, чтобы использовать новые гибридные модели хранения. В одной из таких моделей пользователь может делегировать возможность ротации закрытого ключа учетной записи одному или нескольким хранителям и другим доверенным лицам. Затем модуль Move может определить политику, которая позволяет этим доверенным объектам менять ключ при определенных обстоятельствах. Например, объекты могут быть представлены ключом с мультиподписью k из n, который хранится у многих доверенных сторон, и предлагать услуги по восстановлению ключа для предотвращения потери ключа пользователя (например, 20% биткойнов в настоящее время заблокированы в невосстановимых учетных записях). [7]).

Более того, хотя многие кошельки поддерживают различные схемы восстановления ключей, такие как резервное копирование закрытых ключей в облачную инфраструктуру, многосторонние вычисления и социальное восстановление, они обычно реализуются без поддержки блокчейна (то есть вне сети). В результате каждый кошелек должен реализовать свою собственную инфраструктуру управления ключами, а соответствующие операции становятся непрозрачными для пользователей. Напротив, поддержка функций управления ключами на уровне блокчейна Aptos обеспечивает полную прозрачность всех операций, связанных с ключами, и упрощает реализацию кошелька с расширенным управлением ключами.

6.3 Прозрачность транзакций до подписания

Сегодня кошельки обеспечивают очень мало прозрачности в отношении транзакций, которые они подписывают. В результате пользователей часто легко обманом заставить подписать вредоносные транзакции, которые могут похитить средства и иметь разрушительные последствия. Это верно даже для блокчейнов, которые требуют перечисления всех данных в цепочке, к которым обращается каждая транзакция. В результате в настоящее время существует мало средств защиты пользователей, что делает пользователей уязвимыми для самых разных атак.

Чтобы решить эту проблему, экосистема Aptos предоставляет услуги для предварительного выполнения транзакций: мера предосторожности, которая описывает пользователям (в удобочитаемой форме) результаты их транзакций до подписания. Сочетание этого с известной историей предыдущих атак и вредоносных смарт-контрактов поможет уменьшить мошенничество. Кроме того, Aptos также позволяет кошелькам налагать ограничения на транзакции во время выполнения. Нарушение этих ограничений приведет к прерыванию транзакций для дополнительной защиты пользователей от вредоносных приложений или атак социальной инженерии.

6.4 Практические облегченные клиентские протоколы

Использование исключительно сертификатов TLS/SSL поставщиков API для установления доверия между клиентами и серверами блокчейна не обеспечивает достаточной защиты клиентов. Даже при наличии действующих сертификатов кошельки и клиенты не имеют никаких гарантий относительно подлинности и целостности представляемых данных.

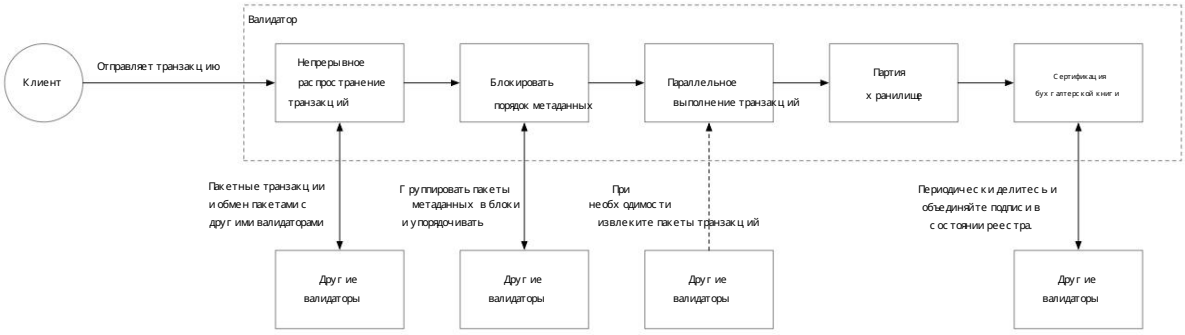

Рисунок 4: Жизненный цикл обработки транзакций. Все этапы полностью независимы и индивидуально распараллеливаемы.

им. В результате поставщики API могут возвращать неверные или вредоносные данные блокчейна, вводя в заблуждение третьих лиц и выполняя атаки двойного расхода.

Чтобы предотвратить это, Aptos предоставляет доказательства состояния и упрощенные протоколы проверки клиентов, которые могут использоваться кошельками и клиентами для проверки достоверности данных, предоставляемых ненадежным сторонним сервером. Более того, используя доказательства состояния на основе временных меток в Разделе 7.6.2, легкие клиенты всегда могут гарантировать жесткие ограничения на актуальность состояния учетной записи (например, в течение нескольких секунд), и им нужно только отслеживать изменения в конфигурации сети (изменения эпохи) или используйте текущие доверенные контрольные точки (путевые точки), чтобы оставаться в курсе последних событий [8]. Комбинируя высокочастотные временные метки и недорогие доказательства состояния, блокчейн Aptos предоставляет клиентам повышенные гарантии безопасности.

Кроме того, узлы Aptos также предоставляют богатые, высокопроизводительные интерфейсы хранения, которые можно дополнительно настроить, чтобы разрешить подписку на доказательства, нацеленные на определенные данные и учетные записи в цепочке. Это может быть использовано легкими клиентами для сохранения минимального количества поддающихся проверке данных без необходимости запуска полного узла или обработки значительного количества транзакций.

7 Конвейерная обработка, пакетная обработка и параллельная обработка транзакций

Чтобы максимизировать пропускную способность, увеличить параллелизм и снизить инженерную сложность, обработка транзакций в блокчейне Aptos разделена на отдельные этапы. Каждый этап полностью независим и индивидуально распараллеливается, напоминая современные суперскалярные процессорные архитектуры. Это не только обеспечивает значительные преимущества в производительности, но также позволяет блокчейну Aptos предлагать новые режимы взаимодействия между валидатором и клиентом. Например:

- Клиенты могут быть уведомлены, когда определенные транзакции были включены в пакет сохраняемых транзакций. Постоянные и действительные транзакции, скорее всего, будут немедленно зафиксированы.
- Клиенты могут быть проинформированы, когда заказана партия сохраняемых транзакций. Таким образом, чтобы уменьшить задержку определения результатов выполненных транзакций, клиенты могут выбрать выполнение транзакций локально, а не ждать, пока валидаторы завершат выполнение удаленно.
- Клиенты могут дождаться выполнения сертифицированной транзакции валидаторами и выполнить состояние синхронизация аттестованных результатов (например, см. раздел 8).

Модульная конструкция Aptos способствует ускорению разработки и поддерживает более быстрые циклы выпуска, поскольку изменения могут быть нацелены на отдельные модули, а не на единую монолитную архитектуру. Точно так же модульная конструкция также обеспечивает структурированный путь к масштабированию валидаторов за пределы одной машины, предоставляя доступ к дополнительным вычислительным, сетевым ресурсам и ресурсам хранения. На рис. 4 показан жизненный цикл транзакции на различных этапах обработки.

7.1 Пакетная обработка

Пакетная обработка — это важная оптимизация эффективности, которая является частью каждого этапа работы в блокчейне Aptos. Транзакции группируются в пакеты каждым валидатором во время распространения транзакций, а пакеты объединяются в блоки во время консенсуса. Исполнение, хранение и

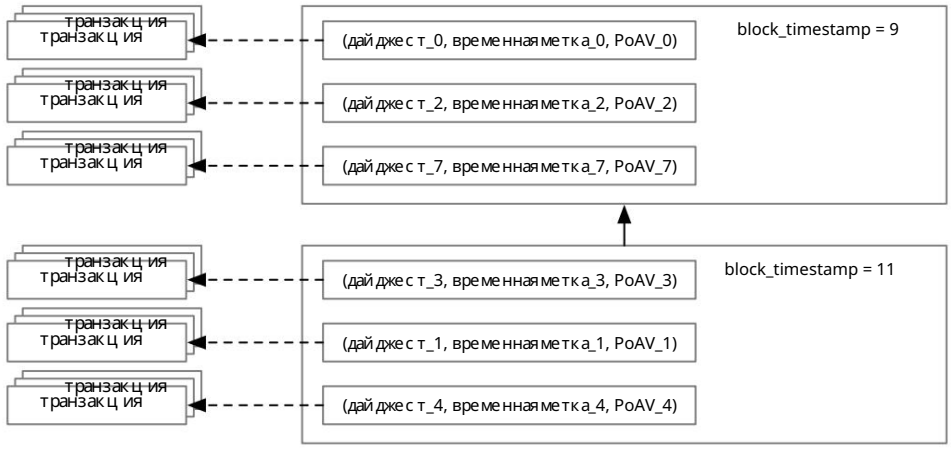

Рисунок 5. Упорядочивание метаданных блоков происходит независимо от распространения транзакций.

Этапы сертификации бухгалтерской книги также работают в пакетном режиме, чтобы предоставить возможности для изменения порядка, сокращения операции (например, дублирование вычислений или проверка подписи) и параллельное выполнение.

Группировка транзакций в пакеты может вызвать небольшую задержку, например ожидание 200 миллисекунд для накопления пакета транзакций перед выполнением распространения. Тем не менее, пакетирование легко настраивается в отношении максимального периода ожидания и максимального размера партии, что позволяет децентрализованной сети для автоматической оптимизации задержки и эффективности. Пакетирование также позволяет для эффективных рынков комиссий, чтобы расставить приоритеты транзакций и избежать непреднамеренных атак типа «отказ в обслуживании» (DoS) отчрезмерно усердных клиентов.

7.2 Непрерывное распространение транзакций

Следуя первоначальному выводу Нарвала и Туска [9], распространение транзакций в блокчейне Aptos не зависит от консенсуса. Валидаторы непрерывно передают пакеты транзакций друг другу, одновременное использование всех доступных сетевых ресурсов. Каждая партия, распространяемая валидатором v, сохраняется, и подпись в пакетном дайджесте отправляется обратно в v. В соответствии с требованиями консенсуса определено в Разделе 7.3, любые 2f + 1 взвешенные подписи в дайджесте пакета формируют доказательство доступность (PoAv). Такое доказательство гарантирует, что по крайней мере f + 1 честных валидаторов, взвешенных по доле, имеют сохранил пакет, и, таким образом, все честные валидаторы смогут получить его до выполнения.

Бесконечно сохраняющиеся пакеты транзакций могут открыть вектор DoS-атаки, заставив валидаторов закончится память и произойдет сбой. Чтобы предотвратить это, каждый пакет транзакций имеет связанную отметку времени. Временная метка в пакете обеспечивает эффективную сборку мусора на каждом валидаторе. Кроме того, отдельный механизм квот для каждого валидатора предназначен для защиты валидаторов от нехватки места даже в самых экстремальных обстоятельствах, например, при потенциальных византийских атаках. Пакеты также имеют ограничения по размеру, которые проверяются перед соглашением о сохранении в стабильном хранилище. Окончательно, несколько оптимизаций для дедупликации и кэширования транзакций снижают затраты на хранение и обеспечивают производительность интеграция с механизмом параллельного выполнения.

7.3 Блочный порядок метаданных

Одно из распространенных заблуждений заключается в том, что консенсус происходит медленно и, следовательно, является основным узким местом для блокчейна. пропускная способность и задержка. Одним из ключевых нововведений блокчейна Aptos является отделение задач, не связанных с соглашением, от фазы консенсуса, таких как распространение транзакций, транзакций. исполнение/хранение и сертификация бухгалтерской книги. Отделяя распространение транзакций от фазы консенсуса, упорядочивание может происходить с очень низкой пропускной способностью (блокировать только метаданные и доказательства), что приводит к высокая пропускная способность транзакций и минимальная задержка.

Сегодня блокчейн Aptos использует последнюю итерацию DiemBFTv4 [10], отзывчивый консенсусный протокол BFT. Консенсус в общем случае требует только двух сетевых раундов. поездок (с временем прохождения туда и обратно обычно менее 300 миллисекунд по всему миру) и динамически настраивает к неисправным валидаторам через механизм репутации лидера [11]. Репутация лидера сети

Мех анизм повышает валидаторов, которые ус пешно зафик с ировали блоки в окне, и понижает валидаторов, которые не уч ас твуют. Этот новый мех анизм значительно повышает производительнос ть в дец ентрализованных с редах , с оответс твенно обес печивает инфрас труктуру для с оответс твующих с тимулов и быс тро минимизирует влияние отказавших валидаторов на пропус кную с пос обнос ть и задержку.

DiemBFTv4 г арантирует жизнес пос обнос ть при ч ас тич ной с инх ронизац ии и обес печивает безопас нос ть при ас инх ронноc ти, к ог да общая доля валидатора с ос тавляет 3f + 1 с ошибоч ными валидаторами, взвешенными по доли до f. DiemBFTv4 был тщ ательно протес тирован в нес кольких итерац иях с 2019 г ода с дес ятками операторов узлов и экос ис темой с нес кольки ми кошельками. Мы так же эк с периментируем с нашими недавними ис с ледованиями (например, Bullshark [12]) и друг ими протоколами, к оторые полаг аю тс я на ис торию блоков и с вяз анную с ними с вязь для опр еделения порядк а и ок ончательнос ти метаданных блоков.

Блок конс енс ус а и отметка времени предложения предлаг аю тс я лидером и с ог лас овываю тс я с друг ими валидаторами, к ак показано на рис унке 5. Обратите внимание, что каждый блок конс енс ус а с одержит только пакетные метаданные и док азательс тва. Фак тичес кие транзак ц ии в блоке не требую тс я, пос кольку PoAV г арантирует, что пакеты транзак ц ий будут дос тупны на этапе выполнения пос ле зак аза (с м. Раздел 7.2). Валидаторы мог ут прог олос овать за предложение лидера пос ле проверки док азательс тва и с оответс твия к ритериям метаданных блок а (например, отметка времени предложения времени ис течения с рок а дей с твия блок а).

7.3.1 Время блок чей на

Блок чейн Aptos ис пользует приблизительную, с ог лас ованную физичес кую временную метку для к аждог о предлаг аемог о блок а и, с оответс твенно, для вс ех транзак ц ий в этом блоке. Эта временная метка позволяет ис пользовать мног ие важные варианты ис пользования. Например:

- Лог ика, завис ящая от времени, в с март-контрак тах . Например, разработчик х отел бы зак одировать, что вс е с тавки на ак ц ионе должны быть получены до полудня четверг а.
- Пос кольку ораку лы публикую т данные в с ети, для к оррелировать с обытия и обрабатывать задержки из реальных данных .
- Клиенты мог ут определить, нас колько они ак туальны в отношении блок чей на. Из с ображений безопас нос ти, чтобы избежать ус таревших данных и дальних атак, клиент должен иметь дос туп к выс окоточной метке времени, к ог да с ос тояние уч етной запис и было обновлено.
- Аудит блок чей на с надежной временной меткой обес печивает с ильную к орреляц ию с с обытиями вне с ети, например, г арантирует, что предус мотренные зак оном выплаты с оответс твую т ожидаемым требованиям.
- Ис течение с рок а транзак ц ии ос новано на с амой пос ледней зафик с ированной метке времени. В к ач ес тве дополнительной защиты клиентс ких транзак ц ий клиенты мог ут выбрать с рок дей с твия транзак ц ии, к ак опис ано в Разделе 6.1.

Блок чейн Aptos предос тавляет с ледую щие г арантии в отношении меток времени для вс ех транзак ц ий внутри блок а:

- Время в блок чей не монотонно увеличивается. То ес ть, ес ли блок B1 < блок B2, то B1.Time < B2.Time.
- Е с ли блок транзак ц ий с ог лас ован с отметкой времени T, то к ак минимум f + 1 ч ес тный валидатор решил, что T нах одится в прошлом. Ч ес тный валидатор будет г олос овать за блок только тог да, к ог да ег о с об с твенные час ы метки времени T. С м. Раздел 7.2.
- Е с ли блок транзак ц ий имеет конс енс ус ный к ворум подпис ей с временной меткой T, ч ес тный валидатор не будет предос тавлять так ой блок друг им валидаторам до тех пор, пок а ег о с об с твенные час ы не превыс ят временную метку T.

Самая пос ледняя временная метка обновляется для к аждог о зафик с ированног о блок а и ис пользуется в к ач ес тве временной метки для вс ех транзак ц ий в этом блоке. К ог да с еть является с инх ронной, блок транзак ц ий фик с ируется при к аждом круг овом обх оде с ети и обес печивает быс трое обновление и выс окона дежные час ы. При желании можно определить более тонкую с тепень у порядочения внутри блоков транзак ц ий.

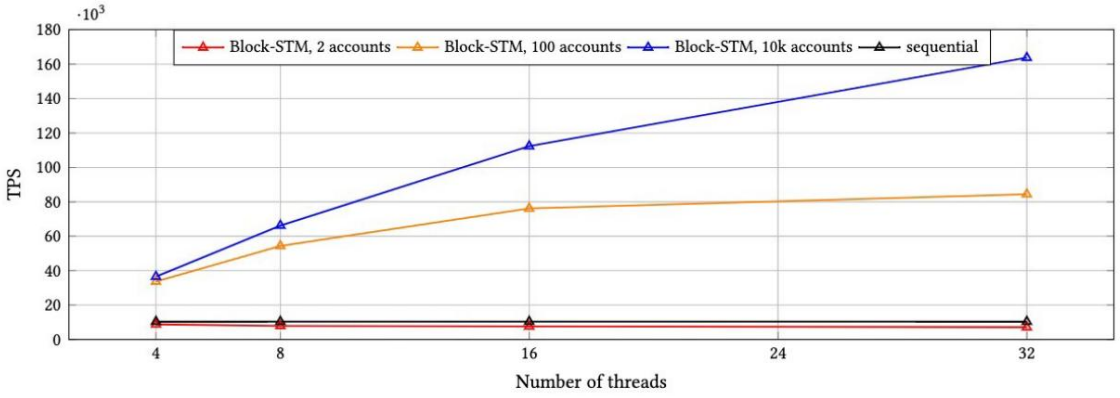

Рис . 6. Тес ты Block-STM (тольк о для к омпонентов), с равниваю щие к оличес тво физичес к их ядер с разными уровнями к онк уренц ии.

7.4 Параллельное выполнение транзак ц ий

Пос ле зак аза метаданных к онс енс ус ног о блок а транзак ц ии мог ут выполнятьс я лю бым валидатором, полным у злом или к лиентом. По к райней мере, 2f + 1 валидатор, взвешенный по доле, дей с твительно с ох ранили транзак ц ии для предложенных пак етов. Пос к ольк у рас прос транение транзак ц ий прои с х одит непрерывно, дополнительные чес тные валидаторы будут получать пак еты транзак ц ий с течением времени. Е с ли чес тный валидатор не получил транзак ц ии для зак азанных пак етов к моменту, к ог да он дос тиг ает с тадии выполнения, он может заг рузить их из 2f +1 взвешенных по доле валидаторов, зная, что по к райней мере f + 1 взвешенных по доле валидаторов (половины подпис анты PoAV, взвешенные по доле) чес тны.

Важной ц елью лю бог о блок чейна являетс я мак с имально возможное параллельное выполнение. Блок чейн Aptos продвиг ает это направление вперед к ак от модели данных , так и от мех анизма ис полнения.

7.4.1 Параллельная модель данных

Модель данных Move изнач ально поддерживает г лобальную адрес ац ию данных и модулей. Транзак ц ии, не имею щие перек рываю щих с я к онфликтов данных и у четных запис ей, мог ут выполнятьс я параллельно. У читывая к онвей ерный дизай н, ис пользуемый блок чейном Aptos, изменение порядк а г руппы транзак ц ий может у меньшить к оличес тво к онфликтов, тем с амым у лу чшив параллелизм.

Даже к ог да транзак ц ии изменяю т один и тот же набор значений в ц епоч ке, больш ая час ть проц ес с а выполнения транзак ц ии вс е еще может быть рас параллелена. Блок чейн Aptos вводит нову ю к онц епц ию , пишет delta, к оторая опис ывает изменение с ос тояния у четной запис и, а не измененное с ос тояние у четной запис и (например, у величение ц елог о чис ла, а не прос то определение к онечног о значения). Вс я обработк а транзак ц ий может выполнятьс я параллельно, а затем дельта-запис и применяю тс я в правильной пос ледовательнос ти для к онфликтую щих значений, ч тобы обес печить детерминированные резу льтаты.

С о временем блок чейн Aptos продолж ит у лу чшать модель данных , у лу чшая параллелизм (например, ис пользуя подс к азк и ч тения/запис и), а так же у лу чшая эрг ономик у, делая более ес тес твенным для разработчик ов с оздание, изменение и с ос тавление значений в ц епоч ке. Move обес печивает г ибк ос ть для внес ения этих у лу чшений к ак на у ровне я зык а, так и с помощью функ ц ий, завис я щих от платформы.

7.4.2 Мех анизм параллельног о выполнения

Мех анизм параллельног о выполнения Block-STM обнаруживает и у правляет к онфликтами для у порядоч енног о набора транзак ц ий наряду с оптимис тичным к онтролем параллелизма, ч тобы обес печить мак с имальный параллелизм при заданном порядк е [13].

Пак еты транзак ц ий оптимис тично выполняю тс я параллельно и проверяю тс я пос ле выполнения. Неу с пешные проверк и приводят к повторному выполнению . Block-STM ис пользует мног оверс ионную с трук туру данных , ч тобы избежать к онфликтов запис и. Вс е операц ии запис и в одно и то же мес то х ранятс я вмес те с их верс иями, к оторые с одерж ат идентифик аторы транзак ц ий и к оличес тво раз, к ог да транзак ц ия запис и оптимис тично выполнялас ь повторно. К ог да транзак ц ия tx с ч итывает я ч ейк у памяти, она получает из мног оверс ионной с трук туры данных значение, запис анное в эту я ч ейк у с амой выс ок ой транзак ц ией, к оторая появляетс я перед tx в заданном порядк е, вмес те с о с вязанной верс ией.

Block-STM уже интегрирован в блокчейн Aptos. Чтобы понять весь потенциал производительности Block-STM, мы провели эксперименты с нетривиальными одноранговыми транзакциями Move (т. е. 8 операций чтения и 5 операций записи на транзакцию) в качестве изолированных операций, предназначенных только для) тест с базой данных в памяти. На рисунке 6 мы представляем результаты выполнения Block-STM. Каждый блок содержит 10 000 транзакций, а количество учетных записей определяет уровень конфликтов и разногласий.

При низкой конкуренции Block-STM достигает 16-кратного ускорения по сравнению с последовательным выполнением с 32 потоками, а при высокой конкуренции Block-STM достигает более чем 8-кратного ускорения. Уникальный для других механизмов параллельного выполнения в пространстве блокчейна, Block-STM способен динамически и прозрачно (без каких-либо подсказок со стороны пользователя) извлекать присущий параллелизм из любой рабочей нагрузки. По сравнению с средами параллельного выполнения, которые требуют предварительного знания расположения данных для чтения или записи, Block STM может одновременно поддерживать более сложные транзакции. Это свойство приводит к меньшему количеству, но более эффективным транзакциям, снижает стоимость и обеспечивает меньшую задержку для пользователей. Возможно, наиболее важно то, что разделение атомарной транзакции на несколько меньших транзакций нарушает семантику «все или ничего» отдельной транзакции с ее сложными результатами состояния. Сочетание выразительной семантики транзакций с параллельным выполнением в Block-STM позволяет разработчикам использовать лучшее из обоих миров.

Обратите внимание, что шаг блочного упорядочивания метаданных не исключает переупорядочивания транзакций на этапе параллельного выполнения. Транзакции можно переупорядочивать в одном или нескольких блоках, чтобы оптимизировать параллелизм для параллельного выполнения. Единственное требование состоит в том, что переупорядочивание должно быть детерминированным для всех честных валидаторов. Оптимизация для параллельного выполнения, а также добавление рандомизации в переупорядочивание могут повысить производительность и потенциально препятствовать методам максимального извлекаемого значения (MEV) для прибыльного переупорядочивания транзакций валидатора. Стратегии защиты от MEV «закажи, затем покажи» также могут быть включены в этот конвейерный дизайн.

Block-STM и переупорядочивание транзакций являются дополнительными методами для увеличения скорости выполнения. лелизм. Их можно комбинировать с подсказками доступа для чтения/записи транзакций для дополнительного параллелизма.

7.5 Пакетное хранение

Фаза параллельного выполнения приводит к созданию наборов записей для всех транзакций в группе. Эти наборы записи могут быть сохранены в памяти для максимальной скорости выполнения, а затем использованы в качестве кеша для следующего блока или набора блоков, которые должны быть выполнены. Любые перекрывающиеся записи должны быть записаны в постоянное хранилище только один раз. Если валидатор выходит из строя до сохранения наборов записи в памяти, он может просто возобновить параллельное выполнение с этапа упорядочения метаданных блока. Отделение пакетного хранения наборов записи от этапа параллельного выполнения обеспечивает эффективность параллельного выполнения. Таким образом, пакетная запись наборов данных позволяет сократить количество операций хранения и использовать преимущества более эффективных и крупных операций ввода-вывода.

Объем памяти, зарезервированный для кэширования набора записей, может быть настроен вручную для каждой машины и обеспечивает естественный механизм обратного давления. Степень детализации пакетов может отличаться от детализации блоков параллельного выполнения, если требуется настроить для конкретных сред ввода-вывода и памяти.

7.6 Сертификация бухгалтерской книги

На этом этапе конвейера каждый отдельный валидатор вычислил новое состояние для зафиксированного блока транзакций. Однако для эффективной поддержки проверенных легких клиентов и синхронизации состояния блокчейн Aptos реализует сертификацию реестра для истории реестра, а также для состояния реестра.
Одним из ключевых отличий блокчейна Aptos является то, что сертификация реестра не находится на критическом пути обработки транзакций и при желании может даже выполняться полностью вне диапазона.

7.6.1 Сертификация истории книги

Валидатор добавляет транзакции вместе с результатами их выполнения в глобальную структуру данных аутентифицированного реестра. Часть выходных данных транзакции — это набор для записи состояния, состоящий из изменений, внесенных в глобальное состояние, доступное с помощью Move. Короткий аутентификатор этой структуры данных является привязкой к истории реестра, которая включает в себя только что выполненный пакет транзакций. Подобно выполнению транзакции, создание этой структуры данных является детерминированным.

Каждый валидатор подписывает короткий аутентификатор для новой версии результирующей базы данных. Валидаторы обмениваются друг с другом своим последним набором коротких аутентификаторов, подписанных кворумом, совместно объединяют короткие аутентификаторы, подписанные кворумом, а также обмениваются друг с другом последними короткими аутентификаторами, подписанными кворумом.

Используя эту коллективную подпись, клиенты могут быть уверены, что версия базы данных представляет собой полную, действительную и необратимую историю реестра в соответствии с свойствами протокола BFT. Клиенты могут запросить любой валидатор (или любую стороннюю реплику базы данных , например полный узел), чтобы прочитать значение базы данных и проверить результат, используя аутентификатор и подтверждение нужных данных .

7.6.2 Периодическая государственная аттестация

Все глобальное состояние, доступное Move, может быть суммировано для короткого аутентификатора в любой момент истории, аналогично сводке истории реестра. Из-за случайного доступа к глобальному состоянию (в отличие от истории реестра, которая предназначена только для добавления), затраты на поддержание этой аутентификации значительны. Тем не менее, при обновлении структуры данных в большом пакете мы можем вычислять обновление параллельно, а также использовать любое совпадение между частями, которые должны обновляться при изменении каждого отдельного значения состояния. Блокчейн Aptos преднамеренно только периодически сертифицирует глобальное состояние, чтобы уменьшить дублирование общих обновлений.

Во время детерминированных и настроенных интервалов сеть выдает транзакции контрольной точки состояния, которые включают глобальный аутентификатор состояния как часть своих выходных данных . Такие версии обозначаются государственными пропускными пунктами. Чем больше разрыв между двумя контрольными точками, тем ниже амортизированная стоимость обновления структуры данных с проверкой подлинности состояния на транзакцию .

С контрольными точками состояния можно прочитать любое значение состояния из них ненадежным способом, не сохраняя все глобальное состояние. Эта возможность полезна для таких приложений, как добавочная синхронизация состояния, сегментированное хранилище между валидаторами, узлы валидаторов без сохранения состояния и легкие клиенты с ограничением хранилища.

Однако, поскольку контрольные точки состояния являются периодическими, для получения подтверждения конкретной версии состояния реестра требуется либо дополнительное выполнение транзакции для отсутствующих изменений состояния, либо подтверждение их включения из аутентифицированной истории реестра.

Контрольные точки состояния привязаны к конкретным версиям транзакций в истории реестра и, следовательно, привязаны к отметке времени, связанной с пакетами транзакций, упомянутой в разделе 7. С отметкой времени легкий клиент может понять давность подтвержденного значения состояния. Без временной метки легкое клиентское доказательство может гарантировать достоверность только предыдущего состояния, которое может быть далеко в прошлом, что дает мало гарантий релевантности. Кроме того, временные метки для проверки состояния необходимы для отслеживания истории доступа и целей аудита, таких как расчет среднего почасового баланса токенов в резерве токенов.

Контрольные точки состояния могут быть получены на основе предыдущей контрольной точки состояния и изменений состояния в выходных данных транзакции после нее. Следовательно, контрольные точки постоянного состояния для стабильного хранилища не обязательно должны находиться на критическом пути для обработки транзакций. Кроме того, полезные эффекты пакетной обработки существуют и при сохранении контрольных точек состояния. Кэширование недавних контрольных точек состояния (или, скорее, разницы между ними) в памяти и сброс только периодических контрольных точек состояния в стабильное хранилище может значительно снизить потребление пропускной способности хранилища. Способ сохранения контрольных точек не влияет на вычисление аутентифицированной структуры данных . Следовательно, это выбор для каждого узла: операторы узлов могут настроить соответствующий компромисс между емкостью памяти и пропускной способностью хранилища.

8 Синхронизация состояний

Блокчейн Aptos призван обеспечить высокую пропускную способность и низкую задержку для всех участников экосистемы. В результате блокчейн должен предлагать эффективный протокол синхронизации состояния для распространения, проверки и сохранения данных блокчейна для облегченных клиентов, полных узлов и валидаторов [14]. Кроме того, протокол синхронизации также должен быть терпимым к ограничениям ресурсов и неоднородности в сети с учетом различных пользователей и вариантов использования. Например, он должен позволять архивным полным узлам проверять и сохранять всю историю и состояние блокчейна, а также позволять легким клиентам эффективно отслеживать только небольшое подмножество состояния блокчейна Aptos.

Для достижения этого свойства блокчейн Aptos использует аутентифицированную историю реестра и сертифицированные подтверждения состояния (см. Раздел 7.6.1) , предлагаемые валидаторами, полными узлами и другими репликаторами, чтобы обеспечить гибкий и настраиваемый протокол синхронизации. В частности, участники сети могут выбирать различные стратегии синхронизации для оптимизации своих вариантов использования и требований.

Например, в случае полных узлов Aptos допускает несколько стратегий синхронизации, включая возможность обработки всех транзакций с начала времени или полного пропуска истории блокчейна и синхронизации только последнего состояния блокчейна с использованием путевых точек. В случае легких клиентов стратегии включают синхронизацию частичных состояний блокчейна, например, конкретных учетных записей или значений данных , и включение

чтение подтвержденного состояния, например, выборка подтвержденного баланса счета. Во всех случаях Aptos позволяет участникам настраивать объем и возраст данных для извлечения, обработки и их ранения.

Применяя гибкий и настраиваемый подход к синхронизации состояний, Aptos может адаптироваться к различным требованиям клиентов и продолжать предлагать новые и более эффективные стратегии синхронизации в будущем.

9 Общественная собственность

Блокчейн Aptos будет принадлежать, управляться и управляться широким и разнообразным сообществом.
Собственный токен Aptos будет использоваться для транзакционных и сетевых комиссий, управления голосованием по обновлениям протокола и процессами в цепочке/вне цепочки, а также для защиты блокчейна с помощью модели Proof-of-Stake. Полное описание экономики токенов Aptos будет опубликовано в следующей публикации.

9.1 Транзакционные и сетевые сборы

Все транзакции Aptos имеют цену за единицу газа (указанную в токенах Aptos), которая позволяет валидаторам отдавать приоритет транзакциям с наибольшей стоимостью в сети. Более того, на каждом этапе конвейерной модели существует множество возможностей для отказа от транзакций с низкой стоимостью (что позволяет блокчейну работать эффективно при полной нагрузке системы). Со временем будут взиматься сетевые сборы, чтобы гарантировать, что затраты на использование блокчейна Aptos пропорциональны реальным затратам на развертывание оборудования, обслуживание и эксплуатацию узла. Кроме того, у разработчиков будет возможность разрабатывать приложения с различным соотношением затрат между вычислительными ресурсами, хранилищем и сетью.

9.2 Управление сетью

Каждое существенное изменение функций и улучшение блокчейна Aptos будет проходить в несколько этапов, включая предложение, реализацию, тестирование и развертывание. Эта структура создает возможности для соответствующих сторон и заинтересованных сторон для обеспечения обратной связи, обмена опасениями и внесения предложений.
Заключительный этап, развертывание, обычно выполняется в два этапа. Во-первых, выпуск программного обеспечения с новой функциональностью будет развернут на каждом узле, а во-вторых, эта функциональность будет включена, например, с помощью флага функции или переменной конфигурации в цепочке.

Каждое развертывание программного обеспечения операторами узлов должно быть обратно совместимым, чтобы обеспечить совместимость нового программного обеспечения с поддерживаемыми версиями. Процесс развертывания новой версии программного обеспечения может занять несколько дней, чтобы учесть операторов в разных часовых поясах и любые внешние проблемы.
Как только будет обновлено достаточное количество узлов, включение новой функциональности может быть инициировано точкой синхронизации, такой как согласованная высота блока или изменение эпохи. В экстренных условиях (например, когда простои неизбежны) включение может осуществляться вручную и принудительно операторами узлов, а в худшем случае — посредством хардфорка в сети.

По сравнению с другими блокчейнами, блокчейн Aptos кодирует свою конфигурацию в цепочке. Каждый валидатор имеет возможность синхронизироваться с текущим состоянием блокчейна и автоматически выбирать правильную конфигурацию (например, протокол консенсуса и версию платформы Aptos) на основе текущих значений в цепочке. Благодаря этой функциональности обновления в блокчейне Aptos выполняются мгновенно и без проблем.

Чтобы обеспечить гибкость и настраиваемость процесса включения, блокчейн Aptos будет поддерживать управление в цепочке, где держатели токенов могут голосовать в отношении их весов токенов.
Протоколы голосования в сети являются общедоступными, проверяемыми и могут быть мгновенными. Он-чейн-управление также может поддерживать получение небинарных результатов без развертывания программного обеспечения. Например, параметры протокола выборов лидера в цепочке могут быть изменены с помощью управления в цепочке, тогда как заранее известная точка синхронизации не сможет обрабатывать динамические модификации, поскольку все изменения должны быть известны заранее.

Управление по цепочке может со временем быть развернуто на протяжении всего процесса управления обновлением.
В качестве примера:

1. Владельцы токенов голосуют в цепочке за перехода на новую квантово-устойчивую схему подписи.

2. Разработчики внедряют и проверяют новую схему подписи и создают новую версию программного обеспечения.

3. Валидаторы обновляют свое программное обеспечение до новой версии.

4. Владельцы токенов голосуют в цепочке за включение новой схемы подписи, конфигурация в цепочке обновляется, и изменение вступает в силу.

Как проект с открытым исходным кодом, блокчейн Aptos будет зависеть от сильной обратной связи с сообществом и использовать управление в цепочке для управления соответствующими процессами. При определенных условиях может потребоваться обновление вне сети, но со временем оно будет сведено к минимуму.

9.3 Консенсус Proof-of-Stake

Чтобы участвовать в проверке транзакций в блокчейне Aptos, валидаторы должны иметь минимально необходимое количество токенов Aptos. Суммы ставок пропорционально влияют на взвешенный PoAv 2f + 1 во время распространения транзакций, а также на вес голосов и выбор лидера во время упорядочения метаданных блоков. Валидаторы принимают решение о разделении вознаграждения между собой и своими заинтересованными сторонами. Стейкеры могут выбрать любое количество валидаторов, в которых они будут размещать свои токены для заранее согласованного распределения вознаграждения. В конце каждой эпохи валидаторы и их соответствующие стейкеры будут получать свои вознаграждения через соответствующие модули Move в сети.

Любой оператор валидатора с достаточной долей может свободно присоединиться к блокчейну Aptos. Все параметры, включая требуемую минимальную ставку, могут быть установлены с помощью процессов активации в сети, описанных в разделе 9.2.

10 Производительность

Как упоминалось в разделе 7, блокчейн Aptos способен достичь оптимальной пропускной способности и аппаратной эффективности благодаря параллельному, пакетно-оптимизированному и модульному конвейеру обработки транзакций. Дополнительные инициативы по повышению производительности, такие как согласованные обновления, дельта-записи, подсказки транзакций и кэширование критических путей, будут продолжать увеличивать пропускную способность и повышать эффективность с течением времени.

Сегодня пропускная способность блокчейна обычно измеряется количеством транзакций в секунду. Однако, учитывая широкий диапазон затрат и сложности транзакций и инфраструктур, это неточный метод сравнения систем. Задержка транзакций также несовершенна, поскольку начальная и конечная точки подчинения окончательности различаются в разных экспериментах.

Кроме того, некоторые системы требуют априорного знания всех одних и всех одних данных транзакций и вынуждают разбивать логические транзакции на более мелкие и менее сложные транзакции. Разделение транзакции приводит к плохому взаимодействию с пользователем и искусственно влияет на задержку и пропускную способность без учета того, чего пытается достичь разработчик. Напротив, подход Aptos заключается в том, чтобы дать разработчикам свободу создавать без ограничений и измерять пропускную способность и задержку в отношении реальных сценариев использования, а не искусственных транзакций.

Блокчейн Aptos будет продолжать оптимизировать производительность отдельных валидаторов, а также экспериментировать с методами масштабирования, которые добавляют больше валидаторов в сеть. Оба направления имеют различные компромиссы. Любой блокчейн с возможностями параллельного выполнения может поддерживать дополнительный параллелизм, требуя более мощного оборудования или даже структурируя каждый валидатор как кластер отдельных машин. Однако существуют практические ограничения на количество глобальных валидаторов, которые соизмеримы с стоимостью и сложностью операторов валидаторов. Рост и популярность бессерверных баз данных в облачных сервисах иллюстрируют, как мало организаций могут эффективно развертывать и обслуживать такие сложные распределенные системы.

10.1 Разделение однородного состояния

Первоначально блокчейн Aptos будет запущен с единым состоянием реестра. Со временем сеть Aptos применит уникальный подход к горизонтальной масштабируемости, сохраняя при этом децентрализацию.
Это будет происходить через несколько сегментированных состояний реестра, каждое из которых предлагает однородный API и сегментирование как первоклассную концепцию. Токен Aptos будет использоваться для комиссий за транзакции, ставок и управления всеми сегментами.

Данные могут передаваться между осколками через однородный мост. Пользователи и разработчики могут выбирать собственные схемы сегментирования в зависимости от своих потребностей. Например, разработчики могут предложить новый сегмент или пользователей кластера в существующих сегментах для достижения высоких соединений внутри сегмента. Более того, шарды могут иметь разные системные характеристики. Один сегмент может быть оптимизирован для вычислений с помощью

SSD и другие могут быть оптимизированы для больших жестких дисков с низкими вычислительными характеристиками. Обеспечивая аппаратную гибкость между различными сегментами, разработчики могут использовать соответствующие системные характеристики для своих приложений.

Таким образом, однородное сегментирование состояния обеспечивает потенциал для горизонтальной масштабируемости пропускной способности, позволяет разработчикам программировать с единым универсальным состоянием для всех сегментов и позволяет кошелькам легко включать сегментированные данные для своих пользователей. Это обеспечивает значительные преимущества в производительности, а также простоту единой унифицированной платформы смарт-контрактов Move.

использованная литература

[1] «Аптос-кор», 2022. [Онлайн]. Доступно: https://github.com/aptos-labs/aptos-core [2] «Движение»,

2022. [Онлайн]. Доступно: https://github.com/move-language/move [3] Д. Мацуока, К. Диксон, Э.

Лаззарин и Р. Хакетт. (2022) Представляем отчет о состоянии криптографии в 2022 году.
[Онлайн]. Доступно: https://a16z.com/tag/state-of-crypto-2022/

[4] З. Амсден, Р. Арора, С. Бано, М. Боде, С. Блэкшир, А. Ботра, Г. Кабрера, К. Каталини, К. Халкиас, Э. Ченг, А. Чинг, А. Чурсин, Г. Данезис, Г. Д. Джакомо, Д. Л. Дилл, Х. Дин, Н. Дудченко, В. Гао, З. Гао, Ф. Гарийо, М. Горвен, П. Хейс, Дж. М. Хоу, ЮХу, К. Херли, К. Леви, К. Ли, З. Ли, Д. Малхи, С. Маргулис, Б. Маурер, П. Мохассель, Л. де Наруа, В. Николаенко, Т. Новацкий, О. Орлов, Д. Перельман, А. Потт, Б. Проктор, С. Кадир, Рейн, Д. Руссси, Б. Шваб, С. Сезер, А. Соннино, Х. Вентер, Л. Вей, Н. Вернерфельт, Б. Уильямс, К. Ву, Х. Ян, Т. Закиан и Р. Чжоу, «Блокчейн libra».

2019. [Онлайн]. Доступно: https://developers.diem.com/papers/the-diem-blockchain/2020-05-26.pdf . [5] С. Блэкшир,

Э. Ченг, Д.Л. Дилл, В. Гао, Б. Маурер, Т. Новаки, А. Потт, С. Кадир, Д.Р.
Рейн, С. Сезер, Т. Закиан и Р. Чжоу, «Move: язык с программируемыми ресурсами», 2019 г.
[Онлайн]. Доступно: https://developers.diem.com/papers/diem-move-a-language-with-programmable resources/
2019-06-18.pdf.

[6] Д. Дилл, В. Грискэмп, Дж. Парк, С. Кадир, М. Сюй и Э. Чжун, «Быстрая и надежная формальная проверка смарт-контрактов с помощью средства проверки перемещений», Инструменты и алгоритмы для построения и анализ систем, Д. Фисман и Г. Розу, ред. Чам: Springer International Publishing, 2022, стр. 183–200.

[7] Н. Поппер. (2021) Утерянные пароли лишают миллионеров их биткойн-состояний. [Онлайн]. Доступно: https://
www.nytimes.com/2021/01/12/technology/bitcoin-passwords-wallets-fortunes.html

[8] The Diem Team, «Синхронизация состояния и проверка зафиксированной информации в системе с реконфигурациями», 2020. [Онлайн]. Доступно: https://github.com/aptos-labs/aptos core/blob/main/documentation/tech-papers/lbft-verification/
lbft-verification.pdf

[9] Г. Данезис, Л. Кокорис-Коги, А. Соннино и А. Шлиг ельман, «Нарвал и бивень: мемпул на основе дага и эффективный консенсус bft», в материалах семнадцатой европейской конференции по компьютерным системам, сер. . ЕвроСис
'22. Нью-Йорк, штат Нью-Йорк, США: Ассоциация вычислительной техники, 2022 г., с. 34–50. [Онлайн]. Доступно:
https://doi.org/10.1145/3492321.3519594

[10] Команда Diem, «Diembft v4: репликация конечного автомата в блокчейне diem», 2021 г. [онлайн].
Доступно: https://developers.diem.com/papers/diem-consensus-state-machine-replication-in-the-diem blockchain/
2021-08-17.pdf

[11] С. Коэн, Р. Гелашвили, Л. Кокорис-Коги, З. Ли, Д. Малхи, А. Соннино и А. Шлиг ельман, «Найдите своих лидеров», CoRR, vol. abs/2110.00960, 2021. [Онлайн]. Доступно: https://arxiv.org/abs/2110.00960 [12] А. Шлиг ельман, Н. Гиридхаран, А.

Соннино и Л. Кокорис-Коги, «Bullshark: Протоколы Dag BFT стали практичными», в материалах 20-й конференции по компьютерной и коммуникационной безопасности (CCS), сер.
КСС '22. Лос-Анджелес, Калифорния, США: Ассоциация вычислительной техники, 2022 г.

[13] Р. Гелашвили, А. Шлиг ельман, З. Сян, Г. Данезис, З. Ли, Ю Ся, Р. Чжоу и Д. Малхи, «Блок-стм: масштабирование выполнения блокчейна путем превращения проклятия упорядочения в благословение производительности», 2022. [Онлайн].
Доступно: https://arxiv.org/abs/2203.06871 [14] Дж.

Линд, «Эволюция синхронизации состояния: путь к 100 000+ транзакций в секунду с задержкой менее секунды в aptos»,
2022 г. [онлайн]. Доступно: https://medium.com/aptoslabs/52e25a2c6f10