Cloudera Data Platform – “лучшее из двух миров”

Интервью с представителями компании Cloudera: Яном Кунигком – главным техническим директором по эксплуатации в регионе EMEA и Флорианом фон Вальтером – старшим менеджером по инженерным решениям в Германии, Австрии, Восточной Европе и России.

Ян Кунигк *(Jan Kunigk)* – главный технический директор по эксплуатации в регионе EMEA, Cloudera.

Флориан фон Вольтер *(Florian von Walter)* – старший менеджер по инженерным решениям в Германии, Австрии, Восточной Европе и России, Cloudera.

SN. Мы знаем компании Cloudera и Hortonworks как разработчиков hadoop-решений для больших данных с большим числом технологических партнеров, которые использовали их разработки в своих on-premise решениях. После объединения решения Cloudera позиционируются как внешние облачные сервисы. Будут ли доступны новые разработки дистрибутивов?

Cloudera. Да. В марте 2019 года Cloudera анонсировала новую корпоративную стратегию «Cloudera Data Platform» (или, кратко, CDP), которая предоставляет первое «Enterprise Data Cloud», основанное на лучших из обоих миров из предыдущих дистрибутивов Cloudera и Hortonworks. Платформа Cloudera Data Platform доступна как в центре обработки данных, так и в публичном облаке. Cloudera также продолжает поддерживать существующие дистрибутивы (CDH – Cloudera Distribution of Hadoop и HDP – Hortonworks Data Platform).

Приступая к новому этапу развития архитектуры больших данных, мы потратили много времени на понимание текущих требований наших клиентов, которые являются крупнейшими компаниями мира – в Западной Европе, Азии, США, включая несколько крупных компаний в России. Наши клиенты выразили желание, чтобы Cloudera была на 100% открытой и расширяемой, предоставляя многофункциональную платформу, которая защищена и управляется на корпоративном уровне. С другой стороны, они ищут простоту использования, эффективность, гибкость (agility) и сервиспригодность – вещи, которыми Hadoop традиционно не был известен. Мы можем резюмировать, что нашим клиентам хотелось бы иметь опыт работы в облаке, когда облако не обязательно совпадает с одним из внешних поставщиков общедоступного облака. Наша новая платформа Cloudera Data Platform управляет данными в любой среде, включая несколько общедоступных облаков, «голое железо» (bare metal), частное облако, а также гибридное облако, и, соответственно, имеет несколько форм-факторов развертывания.

CDP – это интегрированная платформа данных, которая проста в развертывании, управлении и использовании. Упрощая операции, CDP сокращает время на внедрение новых вариантов использования по всей организации. Она использует машинное обучение для интеллектуального автоматического масштабирования рабочих нагрузок вверх и вниз для более экономичного использования облачной инфраструктуры.

С помощью Cloudera Shared Data Experience (SDX), возможностей безопасности и управления в CDP, ИТ-специалисты могут уверенно предоставлять безопасную аналитику, работающую с данными в любом месте. CDP – это новый подход к корпоративным данным – от Edge до AI (рис. 1).

CDP предоставляет простую в использовании аналитику, поддерживая самые требовательные варианты использования. Среди ее основных преимуществ:

– *законченность*: доступны все функции, необходимые для приема, преобразования, запроса, оптимизации и прогнозирования на основе данных, что устраняет необходимость в точечных продуктах;

– *интегрированность*: унифицированные аналитические функции упрощают создание приложений и конвейеров для больших данных;

– *консистентность*: стандартизированный пользовательский интерфейс для разных функций позволяет быстрее и проще анализировать данные.

CDP Data Center – полностью переосмысленная платформа данных. Она объединяет новейшую аналитику с открытым исходным кодом – Spark, Impala, Hive, HBase, Kafka, Hadoop и другие – в многофункциональную систему аналитики и управления данными, которая включает в себя:

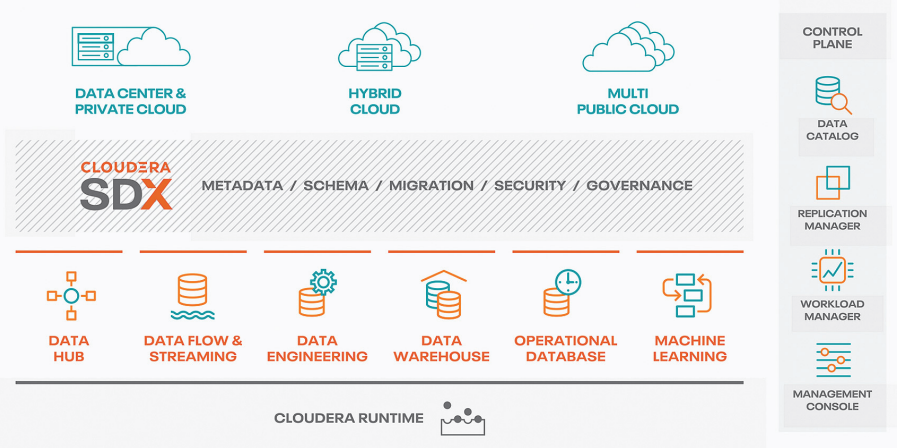


Рис. 1. CDP – это новый подход к корпоративным данным, от Edge до AI.

Табл. 1. Доступные компоненты при переходе с CDH на CDP.

Ranger	• Dynamic row filtering • Dynamic column masking • Attribute-based access control • SparkSQL fine-grained access control
Hive 3	• ACID support - simplify development with transaction guarantees • Comprehensive ANSI SQL 2016 coverage
Hive on Tez	• Better ETL performance

Табл. 2. Доступные компоненты при переходе с HDP на CDP.

Cloudera Manager	• Manage multiple clusters • Automated wire encryption setup • Fine-grained RBAC for administrators • Streamlined maintenance workflows
Impala	• Better fit for Data Mart migration use cases (interactive, BI style queries)
Hue	• Built-in SQL editor
Kudu	• Better performance for fast changing / updateable data

Табл. 3. Доступные компоненты при переходе на CDP, независимо от того, чем ранее вы пользовались.

Virtual Private Clusters	Отделение вычислений от хранилища и поддержка многопользовательского
Atlas 2.0	Расширенное обнаружение данных и поддержка Spark
Ozone Object Storage (Tech Preview)	HDFS Erasure Coding для надежной масштабируемости и интерфейса в стиле S3 для взаимодействия с очень большими файловыми системами и наборами данных
HDFS Erasure Coding	Экономия на хранении

– высокопроизводительную SQL-аналитику;

– потоковую обработку и управление в реальном времени;

– детальный контроль доступа на основе атрибутов;

– динамическую фильтрацию столбцов и маскирование строк;

– 10-кратное увеличение масштабируемости файлов и объектов.

При переходе с CDH на CDP вы получаете доступ к Ranger, Hive 3, Hive on Tez и, соответственно, все их преимущества (табл. 1).

При переходе с HDP на CDP вы получаете доступ к Cloudera manager, Impala, Hue, Kudu и, соответственно, все их преимущества (табл. 2).

При этом независимо от того, чем ранее вы пользовались, при переходе на CDP вам доступны: Virtual Private Clusters, Atlas 2.0, Ozone Object Storage, HDFS Erasure Coding и, соответственно, все их преимущества (табл. 3)

Арун Мурти *(Arun C Murthy)*, наш директор по продажам (*Chief Product Office*), недавно написал отличный пост в блоге о нашем видении Data Cloud: *https://medium.com/swlh/hadoop-evolution-decade2-ca46e5514713*.

SN. Что будет с партнерскими соглашениями после объединения?

Cloudera. Cloudera продолжает работать с партнерской экосистемой и выполняет партнерские соглашения на различных уровнях.

SN. Как будут объединяться разработки Cloudera и Hortonworks? Что будет в результате?

Cloudera. Cloudera анонсировала новую платформу под названием «Cloudera Data Platform» (или, кратко, CDP), которая объединяет лучшее из обоих миров из предыдущих дистрибутивов Cloudera и Hortonworks. CDP уже доступен как локальный дистрибутив для установки в вашем собственном центре обработки данных, а также как облачная версия в общедоступном облаке (в настоящее время доступна в Amazon AWS и Microsoft Azure, а позднее в 2020 году также и в Google Cloud Platform). В течение 2020 года мы также будем внедрять CDP Private Cloud, что привносит преимущества облачных сервисов в центр обработки данных, работающий на локальной контейнерной инфраструктуре.

CDP – это первое корпоративное облако данных, доступное на рынке на сегодняшний день, которое обеспечивает сбор и интеграцию данных, аналитику данных (data warehousing), возможности обработки данных (data engineering) и возможности data science с общим уровнем управления от периферии до искусственного интеллекта гибридным способом, то есть поддержку локальных и мультиоблачных сценариев использования. Для получения дополнительной информации – *https://www.cloudera.com/products/cloudera-data-platform.html* или посмотрите вебинар по требованию (*https://www.cloudera.com/about/events/cloudera-now-cdp.html?keyplay= cdp&utm_campaign=FY21-Q1_VE_AMER_CNow_CDP_OD-HP-_2020-02-05&cid=7012H000001OdDV&internal_link=h01*).

SN. Для повышения эффективности при анализе больших данных (включая технологии AI/ML/DL) используется большое количество специализированных решений, фреймворков, нейроускорителей, FPGA с "заточенным" ПО, энергонезависимая память (типа optane DC) для повышения производительности БД, предобученных моделей и др. Непонятно как такое многообразие планируется интегрировать в общие облачные сервисы?

Cloudera. Специализированное аппаратное обеспечение (например, графические процессоры – GPUs, программируемые матрицы – FPGAs) и программное обеспечение (например, Spark, Python и связанные с ними библиотеки данных – Tensorflow, MXNet, PyTorch, Keras и т.д.) играют важную роль в современной экономике, управляемой данными. Cloudera хорошо осознает важность этих функций и поддерживает специализированное оборудование, такое как графические процессоры или FPGA, либо непосредственно в вашем центре обработки данных, либо с помощью соответствующих стандартных предложений в общедоступном облаке (например, со специализированным виртуальным оборудованием, которое также обеспечивает доступ к графическим процессорам и FPGAs). Что касается специализированных фреймворков, Cloudera также может поддерживать их в зависимости от конкретной технологии. Например, интегрированные среды ML, такие как Tensorflow или MXNet, поддерживаются "из коробки" с нашими предложениями по data science (науке данных) и машинному обучению, такими как "Cloudera Data Science Workbench" (*https://www.cloudera.com/products/data-science-and-engineering/data-science-workbench.html*) или "CDP Machine Learning" (*https://www.cloudera.com/products/machine-learning.html*).

SN. Около пяти лет назад все ключевые разработчики СУБД стали предлагать БД, позволяющие развертывать OLTP- и OLAP-приложения на одной инфраструктуре/платформе – и это большой плюс (с учетом снижения стоимости ОП). В вашей концепции – DATA WAREHOUSE в облаке. Почему?

Cloudera. Возможность использовать одну и ту же технологию базы данных для OLTP и OLAP, безусловно, стала большим достижением в технологии. Однако по-прежнему распространена практика иметь отдельные инфраструктуры для приложений OLTP и OLAP по нескольким причинам, некоторые из которых являются оперативными или организационными. Во многих случаях требования к инфраструктуре для анализа больших объемов данных (например, приложений для анализа больших данных) отличаются от требований к приложениям для анализа OLTP баз данных. Поэтому оптимизация затрат за счет интеграции OLTP- и OLAP-инфраструктур во многих случаях не могла быть реализована. Другие способы снижения затрат и оптимизации, а также ускорения выхода на рынок были представлены с помощью облачной инфраструктуры.

Уже много лет Cloudera предоставляет технические возможности для хранения и анализа огромных объемов данных вплоть до эксабайтного диапазона. Cloudera по-прежнему предоставляет эти возможности на базе традиционных дистрибутивов для больших данных (CDH или HDP), а также на базе новой платформы Cloudera Data Platform. Из-за измененного поведения в том, как инфраструктура больших данных распределяется и используется между пользователями в облаке, Cloudera стремится облегчить миграцию своих клиентов в облако, предоставляя широкие возможности по анализу данных (data warehousing) в облаке, которые оптимизированы как для публичной облачной инфраструктуры, так и для частного облака. Публичные облачные предложения с соответствующими предложениями для частного облака от Cloudera появятся позже в 2020 году.

В целях упрощения корпоративными ИТ-отделами обслуживание потребностей своих бизнес-пользователей, мы представили Cloudera Data Warehouse (CDW) – автоматически масштабируемую, высококонкурентную и экономически эффективную аналитическую услугу, которая принимает (ingests) крупномасштабные данные в любом месте, от структурированных и неструктурированных источников, а также от периферийных/граничных устройств. CDW позволяет бизнес-подразделениям развертывать виртуальные склады по требованию, работающие

в контейнерах, в течение нескольких секунд. Хранилища данных (warehouses) могут увеличиваться и уменьшаться по мере изменения нагрузки (количества параллельных запросов) и автоматически останавливаться, если не используются для ограничения использования ресурсов. CDW основана на технологиях SQL с открытым исходным кодом, длительное время поддерживаемыми и развиваемыми Cloudera и Hortonworks и сегодня доступными в публичном облаке. В ближайшем будущем мы планируем выпустить сервис в рамках CDP Private Cloud.

SN. В настоящее время развивается большое направление edge computing с поддержкой технологий AI/ML/DL на граничных устройствах. У вас для IoT-решений – сбор потоковых данных в облаке. Приведите примеры, когда это целесообразно?

Cloudera. С помощью Cloudera Data Flow (CDF) Cloudera предоставляет набор инструментов для поддержки потоковой передачи данных от края к центру обработки данных или облаку по мере необходимости. Технология пограничного агента (edge agent), такая как Minifi, позволяет собирать данные с периферии для передачи их в центральную облачную инфраструктуру, но также может запускать модели ML на подходящих пограничных устройствах. Примерами простого сбора данных с использованием потоковой передачи могут быть сбор данных журнала из распределенной инфраструктуры центра обработки данных, такой как серверы, маршрутизаторы, коммутаторы, сбор данных о продажах с POS-устройств, сбор данных о транзакциях из банкоматов или устройств кредитных карт, сбор данных датчиков с железнодорожных путей, поездов, грузовых автомобилей, кораблей или других устройств, развернутых в полевых условиях. Одним из последних дополнений, с точки зрения поддерживаемых протоколов, является OPC UA, который позволяет потоковой технологии Cloudera собирать данные датчиков с промышленных предприятий.

SN. Как планируется обеспечивать безопасность данных во внешнем облаке, получаемых по ETL?

Cloudera. Безопасность данных может быть обеспечена на нескольких уровнях. В настоящее время широко поддерживается использование безопасности транспортного уровня посредством шифрования TLS. Стандарты безопасности, такие как Kerberos, для аутентификации и использования VPC (виртуальные частные облака), VPN и другие стандарты безопасности, гарантируют, что облачная ИТ-инфраструктура изолирована от других рабочих нагрузок, работающих в облаке. Данные в состоянии покоя, то есть данные, хранящиеся в собственном облачном хранилище, могут быть зашифрованы аналогично тому, как шифрование данных в состоянии покоя обеспечивается распределенными файловыми системами, такими как HDFS, в локальном центре обработки данных в течение длительного времени. Наконец, если требования безопасности очень высоки, можно использовать шифрование для обеспечения еще большей безопасности. Все эти меры уже доступны сегодня либо через общедоступную облачную инфраструктуру, либо предоставлены Cloudera.

SN. Как осуществляется обучение/дообучение моделей?

Cloudera. Cloudera поддерживает обучение модели ML, предоставляя огромные объемы хранения для данных обучения, которые используются для обучения модели через HDFS или Hadoop Ozone (в будущем), на базе вычислительных ресурсов, таких как инфраструктура распределенных вычислений, которая управляется фреймворками с использованием контейнеров, такими как YARN или Kubernetes. Для ускорения обучения модели ML сегодня используется специализированное оборудование, такое как графические процессоры или FPGA. Для поддержки совместной работы ученых-данных в командах, работающих над одним и тем же проектом, Cloudera предоставляет такие инструменты, как «Cloudera Data Science Workbench» в локальной среде или «CDP Machine Learning» в облаке. Они позволяют ученым данных стандартизировать свой жизненный цикл разработки ML (MLops) через стандартизацию на языках программирования и фреймворков; проводить эксперименты с обеспечением гарантированных и изолированных рабочих сред с использованием контейнерных технологий, таких как контейнеры Docker. Дополнительная информация – официальные документы и вебинары: *https://www.cloudera.com/products/machine-learning.html*.

SN. Есть какая-либо интеграция с существующими разработками (например, с SAS)?

Cloudera: Да, многие технологические партнеры Cloudera обеспечивают интеграцию с экосистемой Cloudera. Список наших Software Technology партнеров по ссылке: *https://www.cloudera.com/partners.html*.

SN. Есть ли примеры реализованных проектов, использующих аналитику в реальном времени в бизнесе?

Cloudera. Сегодня существует множество примеров для аналитики в реальном времени: улучшение качества обслуживания клиентов, мониторинг сетей мобильных телефонов для отслеживания инцидентов в реальном времени, обнаружение и предотвращение мошенничества с кредитными картами, кибербезопасность (например, обнаружение вторжений), онлайн-игры и ставки, подключенные транспортные средства, умные города (предотвращение аварий), управление и оптимизация потоков и многое другое. С последним добавлением Apache Flink в потоковую платформу Cloudera “Cloudera Data Flow” возможности реализации аналитических проектов в реальном времени, основанных на потоковой технологии, еще больше улучшились. Галерея примеров использования и истории успеха клиентов: *https://www.cloudera.com/solutions.html*.

Увеличение производительности Apache Kudu block cache с помощью Intel Optane DCPMM

Kudu – это программное обеспечение с открытым исходным кодом, лицензируемое для Apache 2.0 (*https://kudu.apache.org/overview.html#architecture*). Apache Kudu – новое дополнение к экосистеме Apache Hadoop, которое дополняет уровень хранения Hadoop, обеспечивая быструю аналитику на быстрых данных.

Kudu ориентирован на поддержку OLAP-хранилищ и, как их большинство, Kudu организует свои данные по столбцам, а не по строкам, что позволяет достичь гораздо большей эффективности по сравнению с построчным хранением. Хранение в столбцах также значительно сокращает объем ввода-вывода данных, необходимый для обслуживания аналитических запросов. Используя такие методы, как lazy data materialization и predicate pushdown, Kudu может выполнять запросы по углубленной детализации (drill-down) и “поиск иголки в стоге сена” над миллиардами строк и терабайтами данных в считаные секунды (*https://kudu.apache.org/kudu.pdf*).

В отличие от других хранилищ для анализа больших данных, Kudu это – не просто формат файла, это – оперативная система хранения, которая поддерживает доступ к отдельным строкам с задержкой на уровне миллисекунд. Для доступа в стиле «NoSQL» можно выбирать между API Java, C ++ или Python. И, конечно, эти API произвольного доступа можно использовать в сочетании с пакетным доступом для машинного обучения или аналитики.

API-интерфейсы Kudu разработаны так, чтобы ими было легко пользоваться. Модель данных полностью типизирована, поэтому не нужно беспокоиться о двоичном кодировании или экзотической сериализации: можно просто хранить примитивные типы, например, при использовании JDBC или ODBC.

Kudu не предназначен для OLTP-системы, но если есть некоторое подмножество данных, которое помещается в память, Kudu предлагает конкурентоспособную производительность произвольного доступа. Возможность выполнения сетевых рабочих нагрузок с малой задержкой в том же хранилище, что и на внутреннем анализе данных, может значительно упростить архитектуру приложения.

Kudu специально разработан для случаев использования, которые требуют быстрой аналитики быстрых (быстро меняющихся) данных, и таким образом, чтобы использовать преимущества аппаратного обеспечения и обработки в памяти следующего поколения. Kudu значительно снижает задержку запросов для Apache Impala и Apache Spark.

Кластер Kudu хранит таблицы, которые выглядят так же, как таблицы, к которым привыкли из реляционных (SQL) баз данных. Строки могут быть эффективно прочитаны, обновлены или удалены по их первичному ключу. Простая модель данных Kudu позволяет легко переносить устаревшие приложения или создавать новые: не нужно думать о том, как закодировать данные в двоичные объекты, или разбираться в огромной базе данных полной трудно интерпретируемого формата JSON (JavaScript Object Notation). Таблицы имеют самоописание, поэтому можно использовать стандартные инструменты, такие как движки SQL или Spark, для анализа данных.

Kudu предоставляет комбинацию быстрых вставок/обновлений (inserts/updates) и эффективное поколоночное сканирование для обеспечения нескольких аналитических рабочих нагрузок в реальном времени на одном уровне хранения. Kudu в качестве нового дополнения к HDFS и Apache HBase предоставляет архитекторам гибкость для поддержки широкого спектра применений без экзотических обходных путей.

Интеграция Kudu с другими средами обработки данных достаточна проста: можно передавать данные из источников данных реального времени в режиме реального времени, используя клиент Java, а затем обрабатывать их сразу при получении, используя Spark, Impala или MapReduce. Также можно прозрачно объединять таблицы Kudu с данными, хранящимися в других хранилищах Hadoop, таких как HDFS или HBase.

Kudu реализован на C ++, поэтому он может легко масштабироваться до больших объемов памяти на узел. Поскольку ключевые структуры хранения данных спроектированы так, чтобы быть в высокой степени параллельными, они могут легко масштабироваться до десятков ядер. Благодаря in-memory поколоночному доступу, Kudu достигает хорошего параллелизма на уровне команд, используя операции SIMD из наборов команд SSE4 и AVX на процессорах Intel.

Чтобы масштабировать данные до больших наборов данных и больших кластеров, Kudu разбивает таблицы на более мелкие единицы, называемые таблетками (tablets). Это разделение может быть сконфигурировано для каждой таблицы на основе хеширования, разделения диапазона или их комбинации. Это позволяет легко выбирать между параллелизмом для аналитических рабочих нагрузок и высокой степенью параллелизма для большего количества онлайн-нагрузок.

Чтобы данные всегда были в безопасности и были доступны, Kudu использует согласованный алгоритм Raft для репликации всех операций для данного таблетки. Raft, как и Paxos, гарантирует, что каждая запись сохраняется, по крайней мере, двумя узлами, прежде чем ответить на запрос клиента, гарантируя, что никакие данные никогда не будут потеряны из-за сбоя компьютера. Когда происходит аппаратный сбой, реплики переконфигурируют себя в течение нескольких секунд, чтобы поддерживать чрезвычайно высокую доступность системы.

Использование консенсуса большинства обеспечивает очень низкие задержки хоста, даже когда некоторые узлы могут испытывать нагрузку от одновременных рабочих нагрузок, таких как задания Spark или тяжелые запросы Impala.

Apache Kudu может использовать постоянную память Intel Optane DC (Optane DCPMM) для своего внутреннего блочного кэша. Чтобы сравнить, как Apache Kudu работает с блочным кэшем в DRAM, с тем, как он работает при использовании Optane DCPMM был использован тест YCSB (Yahoo! Cloud Serving Benchmark – тестовая среда с открытым исходным кодом, которая часто используется для сравнения относительной производительности баз данных NoSQL). Чтобы получить максимальную производительность для реализации Kudu block cache, использовался комплект для разработки постоянной памяти (PMDK – Persistent Memory Development Kit).

Таблет-серверы Kudu хранят и доставляют данные клиентам. Каждый таблет-сервер имеет выделенный блочный кэш LRU, который сопоставляет ключи со значениями. Kudu block cache использует внутреннюю синхронизацию и может безопасно использоваться одновременно из нескольких потоков. Если данные не найдены в кэше блоков, они будут считаны с диска и вставлены в кэш блоков. Для кэша постоянной памяти мы выделили пространство для данных из постоянной памяти вместо DRAM.

Некоторые преимущества persistent memory block cache:

– уменьшение объема DRAM, необходимого для Apache Kudu;
– поддержка производительности, максимально близкой к скорости DRAM;
– увеличение общей емкости кэша для кэширования бОльшего количества данных и повышения производительности всей системы.

DCPMM поддерживает 2 режима работы: memory mode и App Direct mode.

Режим App Direct позволяет операционной системе монтировать диск DCPMM в качестве блочного устройства. Именно этот режим был использован для тестирования пропускной способности и задержек блочного кэша Apache Kudu.

Для оценки производительности Apache Kudu выполнялись рабочие нагрузки чтения YCSB на двух компьютерах: один компьютер имел только DRAM, другой – и DRAM и DCPMM (конфигурация – табл. 1, *https://blog.cloudera.com/maximizing-performance-of-apache-kudu-block-cache-with-intel-optane-dcpmm/*).

Были протестировали два набора данных: маленький (100 ГБ) и большой (700 ГБ).

Табл. 1. Конфигурации сравниваемых компьютеров.

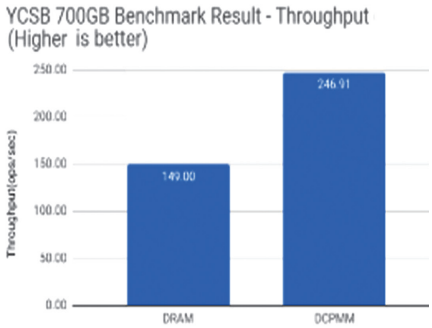
	Config1 – DRAM	Config2 –
OS	Fedora	Fedora
Kernel	4.20.6-200.fc29.x86_64	4.20.6-200.fc29.x86_64
Software versions	Apache Kudu 1.11.0	Apache Kudu 1.11.0
	ycsb 0.16.0	ycsb 0.16.0
Storage – application drivers	8x 1.1TB Intel SSDSC2BB01	8x 1.1TB Intel SSDSC2BB01
Kudu-specific flag values		
	Config1 – DRAM	Config2 –
maintenance_manager_num_threads	4	4
block_cache_type	DRAM	NVM
block_cache_capacity_mb	550000	920000

Первый предназначен для размещения полностью внутри кэша Kudu на обеих машинах. Второй набор данных спроектирован так, чтобы превышать емкость кэша блоков Kudu в DRAM, и в то же время полностью помещаться в кэш блоков в DCPMM.

Для большого (700 ГБ) теста конфигурация на основе DCPMM показала увеличение пропускной способности в 1,66 раза по сравнению с конфигурацией на основе DRAM. При измерении задержки чтения на 95-м процентиле (чтение с наблюдаемой задержкой, превышающей 95% всех остальных задержек), наблюдалось увеличение в **1,9** раза в конфигурации на основе DCPMM по сравнению с конфигурацией на основе DRAM (рис. 1).

Для небольшого (100 ГБ) теста (набор данных меньше емкости DRAM) производительности в конфигурациях на основе DCPMM и DRAM были сопоставимы.

Вывод: добавление модулей DCPMM для блочного кэша Kudu может значительно ускорить запросы, которые многократно запрашивают данные из одного и того же временного окна. Более подробная информация об аналитике временных рядов с Kudu на платформе данных Cloudera: *https://www.cloudera.com/campaign/time-series.html.*



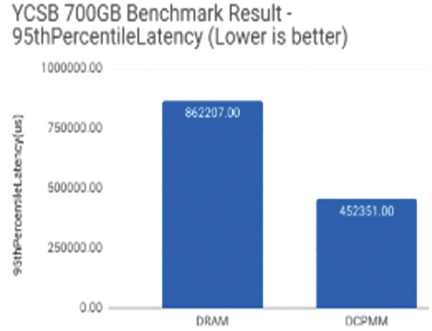


Рис. 1. Результаты тестирования большого набора данных – 700 Гбайт – только с DRAM и с DRAM + DCPMM.