

DOI <https://doi.org/10.15407/usim.2019.01.052>
УДК 004.65:004.7:004.75:004.738.5

А.А. УРСАТЬЕВ, канд. техн. наук, старш. науч. сотруд., вед. науч. сотруд.,
Международный научно-учебный центр
информационных технологий и систем НАН и МОН Украины,
просп. Академика Глушкова, 40, Киев 03187, Украина,
aleksei@irtc.org.ua

БОЛЬШИЕ ДАННЫЕ. АНАЛИТИЧЕСКИЕ БАЗЫ ДАННЫХ И ХРАНИЛИЩА: NETEZZA

Статья представляет собой продолжение исследований Больших Данных и инструментария, трансформируемого в новое поколение технологий и архитектур платформ баз данных и хранилищ для интеллектуального вывода. Рассмотрен ряд прогрессивных разработок известных в мире ИТ-компаний, в частности БД Netezza.

Ключевые слова: платформа Data Warehouse Appliance, АМРР — ассиметричная архитектура с массовым параллелизмом, SN (Shared Nothing) среда МРР, SPU (Snippet Processing Unit), FPGA (ПЛИМ) программируемые логические матрицы, SQL-аналитика на Hadoop, аналитика на Apache Spark в платформах Db2 IBM.

Общая характеристика

Данная статья является продолжением цикла статей, опубликованных в № 4, 2017 г. и № 1, № 2, 2018 г.

Netezza (NYSE: NZ) [70, 71] вслед за *Teradata с Database Appliance* — «приборе» или комбинированном продукте, содержащем аппаратное и программное обеспечение, в 2003 г. объявила о своем первом специализированном хранилище данных (*Data Warehouse Appliance*). В этом же году вышли другие стартапы, в том числе *Greenplum. Data Warehouse Appliance* или *Netezza Appliances* — это оптимизированные приложения хранилищ данных, объединяющие хранение, обработку и аналитику в одной системе, предварительно сконфигурированной и подготовленной к быстрому развертыванию для повышения эффективности и производительности.

Информационная поддержка принятия решений в конце 1990-х годов получает все боль-

шее значение не только на стратегическом, но и на операционном уровне. Задачами традиционных баз данных были оперативная обработка транзакций и хранение информации, где необходим быстрый доступ и обновление одиночных записей, но они оказались не очень эффективными при решении аналитических задач на больших наборах данных. Универсальные СУБД не всегда справляются с задачами, присущими технологиям хранилищ данных, — полным просмотром таблицы, объединением таблиц, сортировкой и агрегированием данных. Платой за неадекватность архитектурных решений стали недостаточная производительность, слабая масштабируемость и непомерно высокие затраты на эксплуатацию и администрирование систем [72, 73].

Несмотря на то, что потенциальная ценность аналитических хранилищ данных была четко установлена к началу 2000-х годов [74], все более скептическое отношение к дорогостоящим в проектировании, настройке и

эксплуатации корпоративных хранилищ данных — *EDW (Enterprise Data Warehouse)* с концепцией единой архитектуры для поддержки всех видов анализа, присущих предприятию, существенно затрудняло получение быстрых аналитических выводов.

Поэтому фокус разработчиков БД сместился на приложения и решения для аналитических хранилищ данных, и возможности современных технологий обусловили появление нового класса аппаратно-программных комплексов — специализированных хранилищ данных, способных на быстрое развертывание, простоту и производительность. Проектируемое для определенной цели хранилище данных *Data Warehouse Appliance* должно иметь инструменты, оптимизированные для производительности при аналитических нагрузках (чтение больших блоков, сканирование таблиц, выполнение сложных запросов и др.), а они должны быть масштабируемыми, отказоустойчивыми и простыми в установке, с небольшой настройкой или без нее. Только тогда они смогут приносить прибыль сразу же после доставки на предприятие. В течение 1990-х годов, когда предприятия вкладывали значительные средства в *EDW*, многие архитекторы БД перепрофилировали ПО, предназначенное для использования в *OLTP*-системах, для специализированных хранилищ данных. В большинстве случаев они полагались на моделирование пользовательских данных и сложную настройку для поддержки рабочих нагрузок в *OLAP*-стиле. Исключением была *Teradata*: эта платформа была специально разработана для *OLAP*, но основное программное обеспечение для хранилищ данных занимало самую большую долю рынка *Oracle* и *IBM* [74].

Фактически используемые в *Netezza* технологии не были уникальными: *Teradata* в течение многих лет реализует *MPP*-архитектуру без совместного использования ресурсов, а *Tandem*¹ — первопроходческие отказоустойчивые базы данных [75, 76]. Ключевым новшеством *Netezza* было объединение этих тех-

нологий в единый пакет, конвергентный и предварительно установленный на оборудовании. Ориентация на аналитическую рабочую нагрузку стала ключом к ценности хранилища данных и радикально сократила время готовности к эксплуатации *Netezza Appliances*. Во-первых, за счет устранения необходимости комплексного обеспечения и настройки путем комплектации программного обеспечения и оборудования. Клиенты *Netezza* приобретали оборудование с уже развернутым программным обеспечением. Во-вторых, *Netezza* устранила необходимость предварять число пользователей. В ней заложены технические решения, имеющие возможность поддерживать производительность в условиях увеличивающегося количества пользователей и объема данных. Благодаря высокопроизводительному дизайну *Netezza* легко превосходит хранилища данных с предустановленными размерами и может сделать это достаточно быстро, без анализа сложных требований и моделирования данных. В то же время, традиционные заказные хранилища данных требовали сложного проектирования, конфигурирования и настройки именно потому, что они были адаптированы из БД общего назначения [74].

Netezza, образца 2003 г., представила свой собственный *Performance Server (NPS™)* серии 8000 [77], в котором объединены массовое параллельное интеллектуальное хранилище данных (*Massively Parallel Intelligent Storage*), программное обеспечение СУБД, и вычислительные мощности в компактной системе, оптимизированной для аналитики (рис. 12). Программное обеспечение базируется на открытой СУБД *PostgreSQL*. Ее функции разделены между главным и периферийными узлами: операции высокого уровня реализуются серверным компонентом *DBO_S*², а низкоуровневые встроены на аппаратном уровне в перифе-

2 Модуль *Database Operations* (часть *X2 Framework* для *Node.js*) позволяет приложениям выполнять сложные операции с БД *SQL* без обращения к *SQL*-запросам, анализируя строки набора результатов в иерархические объекты *JSON* и имея дело со спецификой базы данных и драйвера БД. Это позволяет приложениям сосредоточиться на структурах данных высокого уровня и бизнес-логике, <https://github.com/boylesoftware/x2node-dbos/blob/master/README.md>.

¹ *Tandem Computers* — https://en.wikipedia.org/wiki/Tandem_Computers.

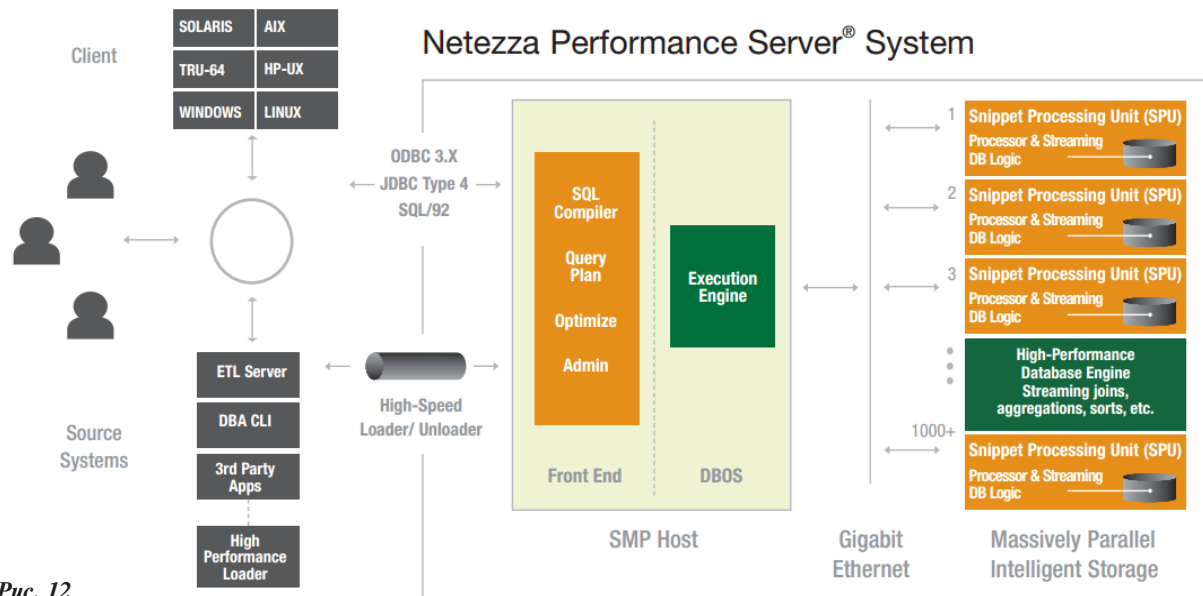


Рис. 12

рийные узлы. Дополнительные программные средства *NPS* включают в себя *Windows*- или *web*-консоль администратора и утилиту массовой загрузки хранилища данных [77, 78].

Ключевая особенность *Netezza NPS Appliances*: тера-масштабируемое высокоскоростное хранилище и архитектура, сочетающая в себе *Symmetric Multiprocessing (SMP³)* и *Massively Parallel Processing (MPP)* для создания

3 В системах обработки данных могут работать два типа бизнес-приложений, одни хорошо масштабируются вверх (например, в рамках одной *SMP*-системы (*Symmetric Multiprocessing* — многопроцессорная обработка)), другие вширь (горизонтально). Масштабирование вверх — вертикальное масштабирование (*scale up*) требует изменения количества процессоров, каналов ввода/вывода, объема памяти, а горизонтальное (*scale out*) масштабирование реализуют подключением дополнительных серверов или созданием кластеров. *SMP*-системы чаще всего применяют для транзакционных БД, небольших хранилищ и витрин данных. Однако такие решения дороги, монолитны, а потому плохо масштабируются. Поэтому крупномасштабные хранилища данных строятся преимущественно на системах с массовым параллелизмом, которые обеспечивают сегментирование данных и распараллеливание обработки запросов. Инфраструктура поддержки выполнения тех или других приложений требует эффективного управления [NGDC: новая реальность ИТ-мира. *Adaptive World* 1/2007, C.2-5]. [ylesoftware/x2node-dbos/blob/master/README.md](https://www.ylesoftware.com/x2node-dbos/blob/master/README.md).

комплекса, нацеленного на быструю обработку петабайт информации. Названа она *асимметричная архитектура с массовым параллелизмом (Asymmetric Massively Parallel Processing, AMPP™)*. Благодаря *AMPP*-архитектуре система сочетает в себе вычислительную мощность *SMP*-сервера с масштабируемостью *MPP* для достижения конкурентоспособной цены/производительности [77].

Двухуровневая архитектура *AMPP Netezza* представляет собой систему, предназначенную для быстрой обработки больших и сложных запросов от ряда пользователей. На первом уровне *host NPS* — высокопроизводительный *SMP*-сервер *Linux* (ОС *Red Hat Linux*) поддерживает каталоги баз данных, оптимизирует для максимальной производительности специальные и сложные запросы, компилируя их в параллельные планы выполнения, и предоставляет необходимое количество вычислительной мощности для сортировки и агрегации больших наборов результатов запроса. Хост делит запрос на последовательность меньших запросов, называемых фрагментами, которые могут выполняться параллельно, и распределяет исполняемый код фрагментов (в программ-

ровании фрагменты исходного кода, пригодные для повторного использования, называют снippetами) во второй уровень для исполнения. Помимо координации компонентов второго уровня, хост также доступен для обработки запросов самостоятельно. Он обычно призван выполнять агрегированные операции, такие как сортировка, объединение и группирование промежуточных результатов. Хост хорошо использует модель общей памяти *SMP* и собственную балансировку нагрузки [77, 78].

Второй уровень архитектуры *AMPP* представляет не делящую ресурсы (*SN*, *Shared Nothing*) среду *MPP*, состоящую из параллельно работающих периферийных узлов или, иначе, модулей обработки фрагментов (*Snippet Processing Unit*, *SPU*). Каждый *SPU* несет ответственность за управление срезом общей базы данных. С этой целью он содержит выделенную память, диск(и) с программируемыми контроллерами и специальный компьютер блейд-сервер *S-Blades* с многоядерными *CPU* на базе процессоров *PowerPC* и жесткой логикой ускорения управления и анализа записей (*Accelerator* для БД *Netezza*) на программируемых логических интегральных схемах (ПЛИС), позволяющих обрабатывать данные ближе к месту их нахождения, избегая ненужной транспортировки.

Каждый периферийный узел выполняет точную аналитическую обработку даже при большой рабочей нагрузке. Он способен обрабатывать множество параллельных фрагментов запроса из нескольких запросов, одновременно принимать потоки данных с диска, проводить вычисления в *CPU* и/или памяти и перемещать результаты через внутреннюю магистральную сеть системы *NPS*. Хотя *SPU* отвечают на запросы от хоста *NPS*, они автономны — выполняют собственное планирование, управление хранением, управление транзакциями, контроль параллелизма и репликацию.

SPUs напрямую не доступны для конечного пользователя или приложения. Последние получают доступ к данным, хранящимся на дисках периферийных узлов, косвенно, обмениваясь данными с *NPS*. Связь между хо-

стом *NPS* и *SPUs* обеспечивается внутренней магистралью сети системы *NPS* с высокой пропускной способностью в стандартном исполнении с коммутаторами *Gigabit Ethernet*.

Высокопроизводительный *SQL Engine* [77] оптимизирует *SQL*-запросы для широкомаштабной параллельной потоковой архитектуры системы *NPS*. Применяя к специальным и сложным запросам сегментацию и распределяя эти фрагменты *SNIP* для выполнения на *MPP*-архитектуре, *Netezza* с архитектурой *AMPP* обрабатывает запрос наиболее оптимизированным способом — *Intelligent Query Streaming* (интеллектуальная потоковая передача запросов). Архитектура *AMPP* позволяет собрать (удалить) все узкие места в поток данных, так что единственным оставшимся ограничением является скорость диска — архитектура «потока данных⁴», где данные перемещаются со скоростью потоковой передачи. Данные, поступающие с диска в поток, должны быть ограничены только необходимыми для анализа данными, а не всей информацией в БД. Это означает, что большая часть исходной работы с БД должна быть сделана в источнике, поэтому разработчики начали с интеллектуальных устройств хранения данных, которые извлекали бы только нужные данные и отправляли их на процессор компьютера в постоянном потоке информации [78].

Благодаря стандартным интерфейсам (*SQL*, *ODBC*, *JDBC*) *Netezza Appliances* полностью совместима с существующими приложениями, инструментами и инфраструктурой *BI*.

В основу интеллектуальной технологии *Intelligent Query Streaming*® *Netezza*® положен подход выполнения интеллектуальной обработки запросов на каждом *SPU* и приведения запроса к данным (это позволило значительно уменьшить трафик данных как между узлами *SPU*, так и хостом *SMP*). Функции запросов

4 Сопереживания об архитектуре потока и его свойствах, изложенные *Foster D. Hinshaw*, соучредителем и техническим директором *Netezza Corp.* (2000–2005гг.), приведены в *Wayne Rash*. 2003 *InfoWorld Innovator: Foster D. Hinshaw*, <https://www.infoworld.com/article/2681904/database/2003-infoworld-innovator--foster-d--hinshaw.html>.

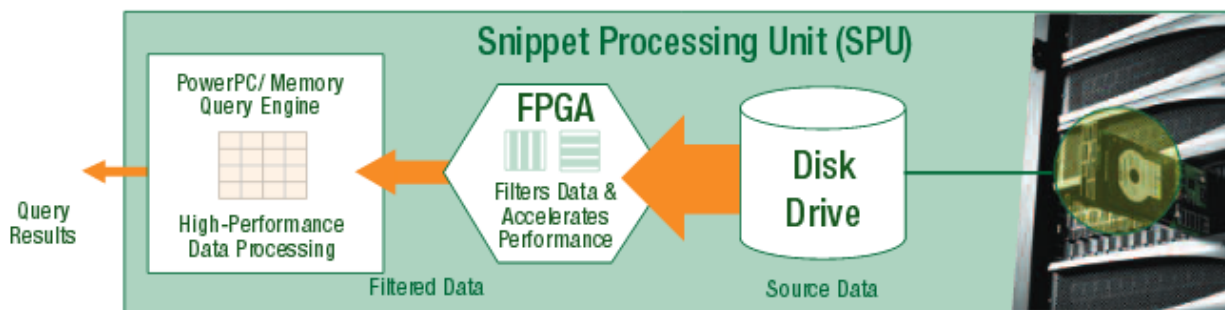


Рис. 13

и обеспечение интеллектуальной потоковой передачи данных на уровне скорости считывания информации с диска реализуется в ПЛИС на основе статических ЗУ — *FPGA (Field Programmable Gate Array)*. *FPGA* — программируемые логические матрицы (ПЛИМ) выступают в роли мультипликатора производительности системы — *Accelerated Streaming Technolog* — и являются ключевым элементом периферийных модулей обработки, обеспечивающих сжатие/декомпрессию⁵ данных, увеличивая тем самым пропускную способность, и выполняющим критически важную функцию сложной фильтрации данных, так что только соответствующие части большого набора данных передаются в высокопроизводительный процессор обработки данных *SQL*-запроса (рис. 13). Так, сортировка, объединение, агрегирование данных выполняются в основном на периферийных узлах (они могут выполняться и на хосте, в зависимости от стоимости обработки этой операции). ПЛИМ выполняют функцию отсева информации с дисков со скоростью ее поступления. Такой процесс вычислений разгружает процессор, память и сеть от обработки чрезмерного объема данных, обеспечивая повышение производительности системы в несколько раз. В результате система выполняет аналитические запросы со скоростью потока данных [78].

Таким образом, *FPGA* — основа запатентованной потоковой архитектуры *Netezza* обеспечивает существенный прирост производитель-

⁵ *Netezza* закладывает основу для постоянных инноваций, новых возможностей и дальнейшего повышения производительности своих потоковых аналитических приборов в устройстве *FAST Engines* [79].

ности системы наряду с основной ее составляющей, опирающейся на *MPP*-архитектуру, которая объединяет значительное число интеллектуальных узлов хранения, чтобы разделить рабочую нагрузку и предоставить ответы на широкий спектр запросов, от простых тактических до выполняемых в почти реальном времени оперативных запросов, вплоть до глубокой аналитики.

Основные преимущества *Netezza*

Итеративная аналитика в режиме реального времени, а также специальные запросы и сложная аналитика с терабайтами данных. Производительность — 10-50X производительности существующих решений для хранилищ данных. Линейная масштабируемость — от 300 Гб до сотен Тб пользовательских данных. Гибкость — простое дополнение модулей обработки для масштабируемости и реконфигурации. Структурная надежность: улучшенные методы зеркалирования обеспечивают надежность с помощью *RAID* и зеркалирования оборудования.

Новые технологии обработки

Поскольку системы *Netezza* все чаще внедряются в средах со смешанными рабочими нагрузками, которые объединяют глубокую специальную аналитику с более короткими, тактическими запросами, то усилия разработчиков направлены на то, чтобы оптимизировать существующие аналитические алгоритмы тактических аналитических запросов. Улучшения производительности обу-

словлены алгоритмической оптимизацией, которая обрабатывает этапы предварительной обработки запросов, такие как оптимизация, компиляция и планирование. Обработка запросов выполняется быстрее, избегая ненужных шагов, агрегируя ответы на уровне связи, уменьшая количество сообщений, обмениваемых в системе, ускоряя связь в системе, увеличивая параллельность операций и фильтрацию данных [80].

Совершенствование технологии изготовления дисков и интерфейсов к ним, поддержание потоковой архитектуры, появление более быстрых и производительных процессоров, а также необходимость снизить цену автономного устройства для аналитики из-за конкурентоспособности на рынке заставило *Netezza* обратиться к стандартному, серийно выпускаемому оборудованию. Так, в 2009 г. компания перешла от процессоров *PowerPC* к *Intel* и товарным блейд-серверам *IBM*. Тогда же *Netezza* представила свое последнее хранилище данных, получившее название *TwinFin* для высокопроизводительного анализа данных до петабайта по более низкой цене ($\$20$ K/terabyte) в сравнении с *NPS* ($\$60$ K/terabyte), чтобы лучше конкурировать с альтернативными решениями от *IBM*, *Oracle* и *Teradata*. Первая, младшая модель (январь 2010 г.) в семействе из четырех устройств *TwinFin* под маркой *Skimmer* ориентирована на небольшие предприятия и целевые отделы в качестве хранилищ и небольших витрин данных. Модели устройств разнятся количеством блейд-серверов (*S-Blades*) и объемами хранимой информации [81, 84].

Netezza запустила устройства *TwinFin*, используя блейд-сервер *IBM BladeCenter* и двухсокетные *S-Blade HS22 Xeon* на базе процессоров X64 и разместив ПЛМ (по одной на ядро *Xeon*) на блейд-сервер. Эта пара известна как *Snippet (SPU)* или *S-Blade*. Устройства *TwinFin* имели от трех до 12 *S-Blades* в корпусе центра обработки данных *BladeCenter-H* и занимали от 8 до 320 Тб емкости хранилища данных. Устройство *Skimmer* помещает *S-Blade* в корпус *BladeCenter-S* в офисном стиле и дает пользователю 10 Тб дискового пространства [84].

Семейство продуктов *TwinFin* основано на одном и том же программном обеспечении — *PostgreSQL*, используемая в *TwinFin*, охарактеризована как сильно модифицированная СУБД общей аппаратной архитектуры, но с новой физической реализацией. Как всегда, архитектура *Netezza* сильно опирается на ПЛМ. Тем не менее, *Netezza* пришла к выводу, что обычные платы процессоров на базе *Intel* лучше согласуются с ПЛМ, чем используемые ранее чипы *PowerPC*. Очевидные последствия перехода *Netezza* на процессоры *Intel*: существенно снижена стоимость одного терабайта пользовательских данных для хранилища; практически любое программное обеспечение, работающее на устройстве хранилища данных, может быть создано, если его еще нет, или перенесено на *Netezza*; аналитическая производительность в некоторых случаях будет значительно повышена (до 100 X, хотя это далеко не всегда).

Дальнейшие модификации продукта будут включать усовершенствования программного обеспечения. Представится возможным создавать приложения на *Java*, *C++*, *Fortran* или *Python*, которые, используя набор *API*-интерфейсов в стеке *Netezza*, могут привлекать ПЛМ для извлечения данных из хранилища и выполнения сложной аналитики, используя инструменты *SAS* или статистический язык программирования с открытым исходным кодом *R*, также позволят подключаться к хранилищу данных через *API*-интерфейсы. При желании использовать *MapReduce* или *Hadoop*, чтобы исследовать большие наборы данных, такие как у *Google*, *Yahoo* и других компаний с *Web 2.0*, есть набор *API*, которые позволяют этим инструментам подключаться к хранилищу данных *PostgreSQL* и многое др. [81, 84].

В связи с выходом на рынок линейки продуктов *TwinFin* компания *Netezza* добавила 28 новых клиентов, приуспевают в цифровых медиа, финансовых услугах, телекоммуникациях и розничной торговле. Она надеется, что радикальное снижение цен и повышение производительности откроют рынок оборудования для

хранения данных для отраслей, которые ранее не рассматривали бы использование такого специализированного и дорогостоящего оборудования как *Netezza* [82, 83].

В 2010 г. *Netezza* объявила, что открыла свои системы для поддержки основных моделей программирования, включая модели *Hadoop*, *MapReduce*, *Java*, *C++* и *Python*.

В этом же году было объявлено о поглощении компании корпорацией *IBM*⁶. Корпорация *IBM* решила, что аналитика и бизнес-аналитика в частности — это рынок, на котором они хотят пребывать, играя отнюдь не последнюю роль. Поняли, что существуют определенные причины, по которым их базы данных *Db2* и *Informix* были менее подходящими для типов проблем, которые решаются автономными устройствами. *Netezza* была, по сути, пионером в создании специализированных хранилищ данных (*Data Warehouse Appliance*). *IBM*, приобретая *Netezza*, получает следующие преимущества: оптимизацию решений по хранению данных и аналитики, достигаемых архитектурой системы, и стандартные интерфейсы сторонних поставщиков.

В эпоху больших данных технологии позволяют потреблять больше разнообразных данных и генерировать новые идеи для конкурентного преимущества. Необходимо иметь возможность быстро реагировать на быстро меняющиеся аналитические требования. В *IBM* признают, что нужны решения для удовлетворения этих потребностей и перевода архитектуры хранилищ данных на следующий уровень.

Решения, ранее присущие устройствам *Netezza*, превратились в *IBM Db2 Warehouse Systems* (прежде *Netezza Appliances*) и входят в состав *IBM PureSystems*, *IBM PureSystems for Analytics* и *IBM Integrated Analytics System*. Эти предложения поддерживают те же ключевые решения, что и *Netezza*, обеспечивая производительность, масштабируемость и встроенную аналитику, при свойственной про-

стоте использования специализированных средств [71].

IBM Netezza — программно-аппаратный аналитический комплекс, объединяющий СУБД, систему хранения данных и вычислительные мощности в компактной системе, предназначенной для быстрого и глубокого анализа больших объемов данных. Выпускается с 2012 г. под маркой *PureData for Analytics* в рамках линейки *PureSystems*. Благодаря технологии *Netezza*, *IBM PureData for Analytics* помогает упростить и оптимизировать производительность служб передачи данных для аналитических приложений и запускает сложные алгоритмы в считанные минуты вместо часов [85, 85].

Программное обеспечение

Продукты линейки *IBM PureData™ System* [86] различаются тем, что каждая модель обеспечивает оптимизацию для конкретной рабочей нагрузки. Соответственно этому продукты *PureData* конфигурируют в целях эффективного решения одной из следующих задач.

1. *Analytics* — аналитическая система оптимизирована для рабочих нагрузок углубленной аналитики, где требуется высокая производительность в сложных исследованиях с привлечением больших массивов данных. Использует встроенные возможности аналитических функций в распределенной вычислительной архитектуре *Shared Nothing*. Аналитические вычисления выполняются непосредственно в БД (*in-database analytics*) с минимальными задержками и максимальным быстродействием без перемещения данных. Таких аналитических функций более 200, включая:

- *Transformations* — профилирование данных, описательные статистики, общая диагностика, статистика, выборка и подготовка данных;
- *Mathematical* — матричная алгебра, гамма- и бета-функции, обширная библиотека статистических и математических функций, поддерживающих широкую направленность аналитических инструментов и языков программирования;
- *Statistics* (статистика) — описательная ста-

⁶ Why did IBM acquire Netezza? — <https://www.quora.com/Why-did-IBM-acquire-Netezza>.

тистика, одномерные и многомерные распределения;

- *Data Mining* (добыча данных) — ассоциативные правила, кластеризация, выделения признаков;

- *Time series* (анализ временных рядов) — авторегрессии, прогнозирование;

- *Predictive* (прогнозирование или предупреждающий анализ) — линейная регрессия, логистическая регрессия, классификация, Байесовский анализ, отбор проб, испытания модели;

- *Geospatial* (геопространственный) — геопространственный тип данных, функции, геометрический анализ, поддержка стандартных промышленных геопространственных приложений *ESRI*.

2. *Operational Analytics* — система оперативной аналитики для принятия решений на основе анализа текущих данных и предоставления информации о работе бизнеса на данных в хранилище. Оптимизирована для аналитики по оперативным данным, для которой характерен высокий процент интерактивных запросов в окне просмотра (*look-ups*), тысячи одновременных оперативных запросов в секунду, смешанные нагрузки интерактивных и аналитических запросов, обработка хранилищ данных в реальном режиме времени. Это не снижает высокой производительности и пропускной способности системы для анализа в БД больших наборов исторических и оперативных данных.

Поступление данных в систему из внешних источников производится в реальном времени без остановки бизнес-анализа и не препятствует принятию решений во время процесса загрузки, что позволяет практически без задержек реагировать на динамическую бизнес-среду.

Использование в системе *PureData* услуги многомерных кубов, обеспечивает быстрый анализ значительных объемов высокочастотных данных. Пользователи могут создавать, редактировать, импортировать, экспортировать и развертывать модели куба над реляционной схемой хранилища данных для анализа ряда бизнес-переменных. Пространственный

анализ ⁷, использующий *ROLAP* на терабайтах данных, может дать исключительную производительность без ограничений физических структур *MOLAP*, известных также как кубы. «*Cubing-услуги*»⁸ существенно повышают производительность *OLAP* запросов, обеспечивая большие возможности пользователям анализировать данные и повысить рентабельность бизнеса. Они позволяют также выполнять интегрированную аналитику структурированных и неструктурированных данных в системе. Поддерживаются стандартные модели интеллектуального анализа данных (в том числе кластеризации, ассоциации, классификации и прогнозирования), выполняемые в производственной среде.

3. *Transactions* — надежная и масштабируемая платформа БД, спроектированная с целью снизить сложность, текущие затраты на управление и ускорить окупаемость; позволяет ИТ-службам оптимизировать задачи транзакционных БД и оперативно обрабатывать транзакции; оптимизирована для транзакционных рабочих нагрузок, которые характеризуются очень низкой латентностью, и ориентированных запросов и обновлений, требующих оптимизацию кэш-памяти.

Система интегрируется с существующими продуктами *IBM*, с них можно автоматически перенаправлять аналитические запросы в *Netezza*, что позволяет строить эффективные гибридные системы, способные как обрабатывать множество транзакций, так и одновременно сканировать большие объемы информации в поисках ответа на аналитические запросы. Для расширения круга задач и организации параллельной обработки данных можно с помощью языков программирования, таких как

⁷ *Five reasons to consider ROLAP with the IBM PureData System for Analytics*. — <http://expertintegratedsystemsblog.com/2014/08/five-reasons-to-consider-rolap-with-the-ibm-puredata-system-for-analytics/>.

⁸ *OLAP-инструменты*, как правило, работают с многомерными БД (МБД). Несмотря на то, что МБД являются наиболее подходящими для оперативной аналитической обработки, эту возможность встраивают в реляционные СУБД или расширяют их *OLAP* для реляционных БД (*ROLAP*). Последние имеют преимущество по масштабируемости и гибкости, но проигрывают по производительности многомерному *OLAP* (*MOLAP*).

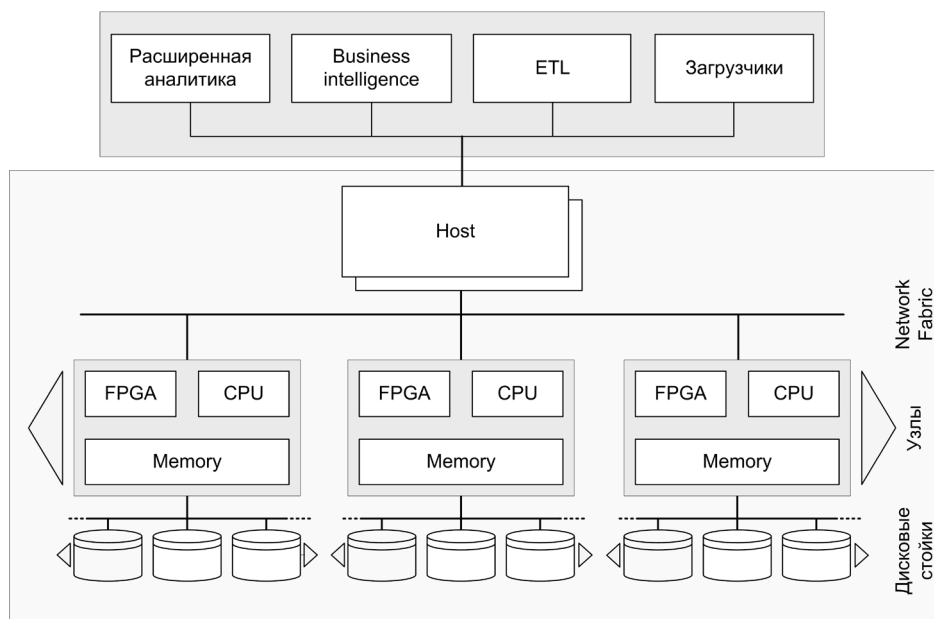


Рис. 14

Lua, Python, Java, Си/C++, Fortran и *R*, писать соответствующие программы.

4. Hadoop — предоставляет услуги передачи данных в программную инфраструктуру с открытым исходным кодом, реализующая одну из основополагающих технологий *BigData*, расширяя имеющееся хранилище для аналитики, и тем самым обеспечивает исследования подключаемых источников данных. Это ускоряет получение ценной информации. Аналитическая платформа предоставляет: высокопроизводительный *SQL*-доступ в *Hadoop*; охват многих источников данных, объединяющих информацию *Hadoop* и *PureData* для *Analytics*; *BigSheets*-инструмент визуализации. Система так же содержит встроенные ускорители аналитики для данных из социальных сетей, анализа данных и текста, а также включает средства архивирования [86, 87].

Аппаратные средства

Несмотря на то, что *Netezza* сменила свою фирменную марку на *PureData for Analytics*, по-прежнему ее эмблема фигурирует при использовании технологии информационных

хранилищ, обсуждении ее аналитических возможностей, архитектуры и других основных особенностей *Netezza* при сохранении ее аппаратной конфигурации (рис. 14). Изменилось только содержание нескольких компонентов: в качестве серверов (*Hosts*) под управлением *Red Hat Linux* использованы *IBM xSeries*, настроенные в конфигурации *активный–пассивный* на повышенную доступность; увеличилось быстродействие внутренней сети⁹ и расширилась функциональность ПЛМ (*FPGA*).

Netezza представила новую ускоренную потоковую технологию *FAST Engine™ Framework*, обеспечивающую увеличение производительности аналитики. Основные функции, которые встроены в ПЛМ, называются «*engines* — движок, механизм, процессор» и составляют структуру *Framework*. Она основана на трех основных концепциях: *FAST Engines* являются основными аналитическими функциями, электронно запрограммированными в ПЛМ для ускорения выполнения

⁹ *Netezza* работает на основе собственного *IP*-протокола, созданного специально для больших объемов данных и интенсивного трафика. Он обеспечивает наилучшее использование ресурсов сети, не создавая ее перегрузки, что позволяет линейно повышать производительность.

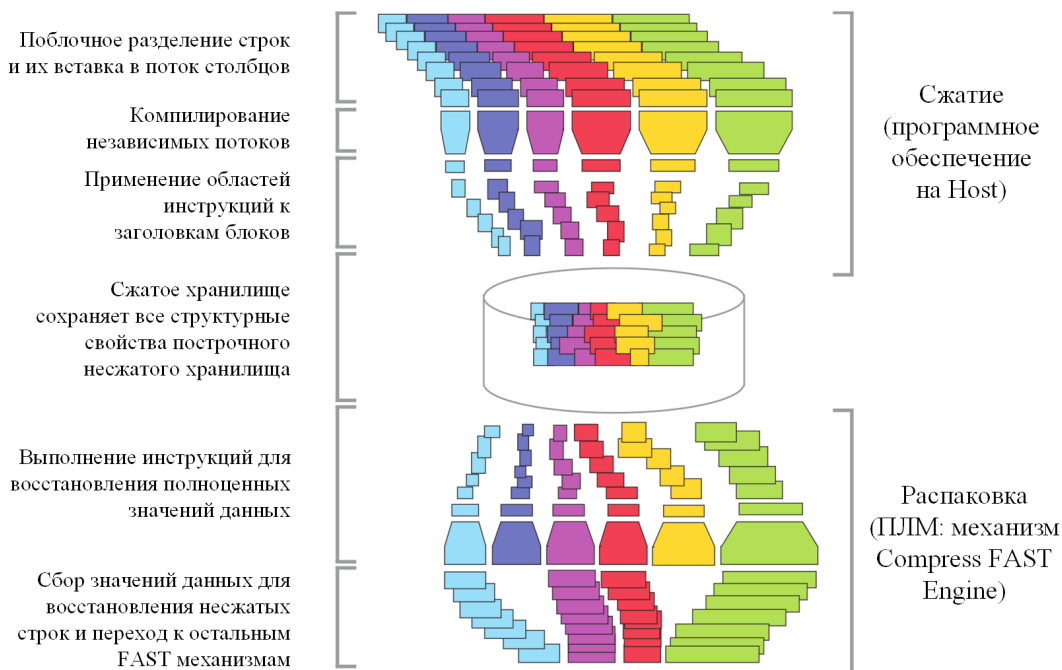


Рис. 15

запросов; они динамически реконфигурируются; настраиваются под каждый фрагмент, выполняемый в сниппете *SPU*, посредством параметров, передаваемых при обработке запроса. Динамическая реконфигурация позволяет каждый из *Engines* модифицировать, отключать или расширять за счет программного обеспечения *Host*. Настройка времени выполнения позволяет ПЛМ включать параметры, передаваемые каждому движку, для оптимизации поведения ПЛМ для конкретного фрагмента запроса.

Действующие механизмы включают *Control*, *Compress*, *Parse*, *Project*, *Restrict* и *Visibility* [79, 81, 85]. Они работают комбинированным последовательным и параллельным образом, и их общий эффект мультипликатора производительности управляет системой [79].

«*Control*» *Engine* (Механизм управления): управляет жестким диском в режимах записи и считывания данных, каналом прямого доступа к памяти (*DMA*) при передаче потоков данных во встроенную память *SPU*. Функции управления повышением производительности хранилища за счет уменьшения количества блоков

данных, считанных с диска, выполняются через *Control Engine* [79].

«*Compress*» *Engine* (Механизм распаковки данных): использует запатентованную технологию поколонной компрессии во всех таблицах БД, что позволяет более эффективно сжимать данные для использования диска и значительно увеличивать скорость потоковой аналитической обработки. *Compress Engine* разработан в первую очередь для повышения производительности потокового запроса в два–три раза (100–200) процентов в дополнение к преимуществам более эффективного использования дисков (рис. 15). Фактически, используемые другими разработчиками интенсивные вычисления, направленные на достижение эффектов сжатия, как правило, приводят к снижению производительности.

В системе производится адаптивная компрессия данных. В зависимости от типов данных используются разные стратегии сжатия (в системе есть 11 алгоритмов сжатия). Поддерживаются все типы данных. Средний эффект сжатия — от 2–2,5X до 4X [81, 87, 88].

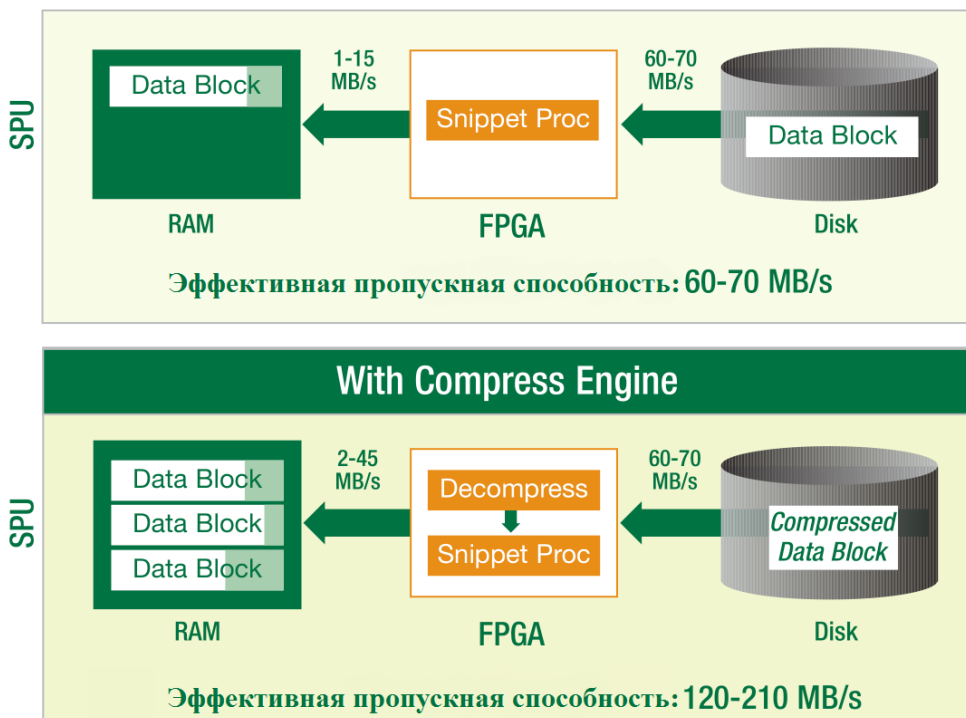


Рис. 16

Compress Engine, как и все другие *FAST Engines*, динамически настраивается и оптимизируется во время выполнения своей задачи.

При записи на диск (например, во время загрузки данных, операций вставки или обновления), исходные данные сжимаются в скомпилированный формат, столбец за столбцом первоначальных данных заменяется набором инструкций *Compress Engine* для декомпиляции.

При чтении данных с диска, *Compress Engine* считывает набор инструкций и, со скоростью потока с диска, распаковывает сжатые и собирает исходные данные, преобразуя каждый блок на диске в 4–8 блоков в памяти (рис. 16). При этом эффективно повышается скорость передачи данных и снижается действующая скорость сканирования на узел *SPU*. Эффект такого решения — повышение постоянных скоростей сканирования данных без привнесения каких-либо изменений в аппаратном обеспечении. Еще одно преимущество *Compress Engine* заключается в том, что он расширяем. Используя тот же базовый

алгоритм компиляции, он может быть использован для обеспечения дополнительных функций, таких как поддержка шифрования данных [79, 85].

«*Parse*» *Engine* (Механизм синтаксиса): играет ключевые роли в структуре *FAST Engines*. Во-первых, он обеспечивает проверку и коррекцию ошибок (*ECC*) потоков распакованных данных с диска, обеспечивая более надежный интерфейс диска и уменьшая необходимость их повторного чтения. Во-вторых, он анализирует этот поток и передает необходимые данные каждому из оставшихся процессоров.

«*Project*» и «*Restrict*» *Engines* (Механизмы фильтрации данных): обеспечивают фильтрацию полей и строк таблицы, опираясь на параметры в условиях *SELECT* и *WHERE* в *SQL*-запросе. Только те столбцы и записи, которые удовлетворяют предложениям, будут разрешены для дальнейшей обработки; в противном случае они будут отфильтрованы и удалены до того, как содержащиеся в них данные будут отправлены на встроенный

процессор и память. Это значительно сокращает последующую обработку фрагментов, возлагаемую для процессора и памяти на каждом *SPU*.

«*Visibility Engine*» (Механизм видимости транзакций): отфильтровывает информацию, которая не должна быть в поле видимости конкретного фрагмента запроса, например, механизм *Visibility* исключает записи (строки) данных, которые не должны быть видимыми для запроса, потому, что они были добавлены в БД после начала текущего запроса. Этот механизм играет важную роль в поддержании *ACID* и настраивается по принципу снимок на скорости передачи данных и без использования каких-либо дополнительных циклов процессора или памяти.

Расширяющийся набор потоковых механизмов *FAST Engines* позволяет создавать *Netezza* новые функции применительно к размерам и типам данных, аналитическим задачам, которые могут быть введены в *FAST Engine™ Framework*.

Другие особенности платформы *Netezza*, направленные на сокращение времени аналитического анализа: зонные (*ZoneMap*)¹⁰ карты [79, 85, 87] и значительное снижение сканирования дисков про помощи технологии зонных карт; многомерный механизм кластеризации с предоставлением данных по четырем измерениям, сокращение сканирования дисков для кластеризованных таблиц и ускоряющие запросы по нескольким измерениям к большим таблицам.

Управление нагрузкой предоставляет дополнительный функционал для регулирования ресурсов и приоритизации выполнения запросов в многопользовательской среде, а также в условиях смешанной нагрузки как то: гарантированное распределение ресурсов, механизм для распределения ресурсов группе пользователей, приоритизированное выполнение запросов и др. [87].

¹⁰ *ZoneMap* работает с естественным порядком записей в хранилищах данных. При использовании данной методики не сканируются записи со значением поля, выходящим за границы запроса.

Применяются диски с самошифрованием для обеспечения дополнительной безопасности и защиты секретных данных в приложениях. Дисковое пространство представлено высокопроизводительными дисками большой емкости, объединенными в *RAID*-массив. Жесткие диски соединены с высокоскоростной сетью *S-Blades*, что позволяет передавать на них данные с максимально возможной скоростью. Пользовательская информация зеркалирована, что обеспечивает ее доступность на 99,99 процентов.

Заключение

Netezza, как и *Teradata*, представляет собой сконфигурированный, подготовленный к быстрому использованию, аппаратно-программный комплекс (*Appliances*), объединяющий хранение и обработку данных в одной системе, изначально спроектированной и оптимизированной под аналитику. Использована не делящая ресурсы среда *MPP* массовой параллельной обработки и головной сервер *SMP* верхнего уровня системы *AMP*-архитектуры.

Помимо координирования работ *SMP*-сервер обеспечивает прирост производительности при увеличении числа клиентских сессий. Архитектурным решением предусмотрено соблюдение принципа непосредственной близости системы обработки и хранения данных, и минимизации перемещения данных при обработке больших объемов информации.

Значительная часть обработки данных выполняется практически на уровне дисковых контроллеров узлов *SPU* — интеллектуальных обработчиков сниппетов. Запросы к системе формулируются на языке *SQL*, а для доступа к системе используются стандартные протоколы *JDBC*, *OLE DB* и *ODBC*. Загрузка данных проводится как штатными утилитами, так и средствами инструментов *ETL*. *Netezza* хочет изменить существующее положение и перейти к реальной параллельной загрузке.

Ключевой особенностью *Netezza* есть мультипликатор производительности аналитического комплекса, дающий существенное аппаратное ускорение выполнению запросов *SQL*. Программируемые логические матрицы ПЛМ на узлах *SPU* выполняют потоковую обработку данных при обращении к дискам. Вследствие этого память и процессоры *SPU* работают с уже отфильтрованными данными, что значительно ускоряет и упрощает дальнейшую обработку. Ускоренная потоковая технология *FAST Engine™ Framework* достигается программированием функций декомпрессии и фильтрации данных, проверки синтаксиса, видимости транзакций и др. Набор потоковых механизмов *FAST Engines* позволяет создавать новые функции применительно к возникающим задачам.

Особое внимание уделено простоте использования и минимальным требованиям в настройках. В *Netezza* практически нечего администрировать. Например, компрессия данных выполняется автоматически и адаптируется к типам данных, не требуя от пользователя указаний на нужные алгоритмы. Нет настройки и проектирования БД, требований к модели данных. Нет индексов и тюнинга, в том числе для выполнения *ad-hoc* запросов — производительность такая, как она есть (из коробки).

Управление нагрузкой предоставляет функционал для регулирования ресурсов и приоритизации выполнения запросов в многопользовательской среде и в условиях смешанной нагрузки. Для расширения круга задач и возможности разработки собственных аналитических процессов есть возможность использовать *C/C++*, *Java*, *Python*, *Fortran*, *R* и поддержку расширяемой, *open-source* интегрированной среды разработки (плагин для *Eclipse*).

Netezza в первую очередь аналитический комплекс, у которого весьма развитые сред-

ства аналитики, такие как *Data Mining*, *OLAP*, *Hadoop* и др., однако он имеет, по данным *Monash Research*, одну из самых низких в индустрии стоимость владения за терабайт пользовательских данных.

Основными конкурентами в этой области *Netezza* являются *Teradata*, *Vertica*, *IBM*, *Greenplum* и др.

Система *Netezza* интегрируется с существующими продуктами *IBM* [24, 67, 71], которые добавляют возможности облачного хранения данных *dbPaaS IBM*, включают в себя использование различных платформ в среде локальных и гибридных облаков, также поддержку аналитики в БД на *Apache Spark* автономных СУБД¹¹ на платформах *Db2* и многое другое. Вместе с тем, складывается впечатление, что *Netezza* остается последовательным приверженцем встраиваемых в *SPU* перепрограммируемых средств (ПЛМ) обработки информации и успешного совершенствования ПО, так как именно только за счет него она получила значительный прирост производительности.

В этой связи напрашивается вопрос, не пойдет ли *Netezza* по пути встраивания в *SPU* производительной, работающей в оперативной памяти, программной конструкции *Apache Spark* со стандартными библиотеками для аналитики больших данных в поддержку уже существующим *framework*, на ПЛМ? По крайней мере, это представляется в рамках *Appliances* более интересным, чем строить эффективные гибридные системы, способные как обрабатывать множество транзакций, так и одновременно сканировать большие объемы информации в поисках ответа на аналитические запросы.

¹¹ *Db2® Warehouse* — это хранилище данных аналитики, которое обеспечивает контроль над данными и приложениями, простое развертывание и управление. Продукт предлагает технологию обработки *BLU* в памяти и аналитику (*in-database analytics*) в БД, масштабируемость и производительность благодаря архитектуре *MPP*. *Db2 Warehouse* обеспечивает совместимость с *Oracle* и *Netezza*[89].

REFERENCES

70. *Netezza*, <https://en.wikipedia.org/wiki/Netezza>.
71. *IBM Analytics*. IBM Data Warehouse Systems (formerly Netezza Appliances), <https://www.ibm.com/analytics/netezza>.
72. Alexandrov, A., 2006. Data Warehouse Machines. Open Systems, 2, <https://www.osp.ru/os/2006/02/1156529>.
73. Aleksandrov, A., 2007. Hardware-software storages, OS, 5, <https://www.osp.ru/os/2007/05/4260303>.
74. *Dinsmore Thomas W.* Disruptive Analytics: Charting Your Strategy for Next-Generation Business Analytics. Apress, 27 aug. 2016, p. 262., <https://www.apress.com/us/book/9781484213124>.
75. *NonStop SQL*. Bauman National Libraru, https://ru.bmstu.wiki/NonStop_SQL.
76. *Fault-tolerant servers of Tandem Computer Inc.*, http://doc.sumy.ua/db/skxbd/glava_17.htm.
77. *Netezza Performance Server (NPS™) 8000*. Wayback Machine, https://web.archive.org/web/20040407102937/http://www.netezza.com:80/products/prod_downloads/Product%20Brochure.pdf.
78. *Foster, D. Hinshaw. A. MPP: combining SMP and MPP to speed database queries*, [https://www.acronymfinder.com/Asymmetric-Massively-Parallel-Processing-\(Netezza-Performance-Server\)-\(AMPP\).html](https://www.acronymfinder.com/Asymmetric-Massively-Parallel-Processing-(Netezza-Performance-Server)-(AMPP).html).
79. *The Netezza FAST Engines™ Framework*. A Powerful Framework for High-Performance Analytics, 2008, <http://www.monash.com/uploads/netezza-fpga.pdf>.
80. *Swoyer, S.* Netezza Says Netezza Performance Server R4 Doubles Query Performance, <https://tdwi.org/articles/2007/09/05/netezza-says-nps-r4-doubles-query-performance.aspx>.
81. *Monash Curt.* Netezza is changing its hardware architecture and slashing prices accordingly. July 30, 2009, <http://www.dbms2.com/2009/07/30/netezza-new-product-family/>.
82. *Netezza launches new data warehouse appliance family*, July 31 2009, <https://www.zdnet.com/article/netezza-launches-new-data-warehouse-appliance-family/>.
83. *Netezza's TwinFin fuels profit surge*, Aug. 27, 2010, <https://www.zdnet.com/article/netezzas-twinfin-fuels-profit-surge/>.
84. *Prickett-Morgan Timothy.* Netezza to bake analytics into appliances, Feb. 24, 2010, https://www.theregister.co.uk/2010/02/24/netezza_data_analytics/.
85. *IBM.* Francisco Phil. Netezza Data Appliance Architecture: High-Performance Data Warehouse and Analytics Platform, <http://www.redbooks.ibm.com/redbooks/pdfs/redp4725-00-ru.pdf>.
86. *IBM PureData System*, <http://www.ndm.net/datawarehouse/IBM/ibm-puredata-system>.
87. *Timchur, A.* Unique IBM Netezza software and hardware for analytical data warehouses. Forum IBM 2012, <https://www.ibm.com/ru/events/presentations/astana2012/at2.pdf>.
88. *Volkov, D.* Netezza Beep Dive. Dsvolk Oracle News: 01.07.11–01.08. 11, <http://dsvolk.blogspot.com/2011/07/>.
89. *IBM Db2 Warehouse overview*. IBM® IBM Knowledge Center, https://www.ibm.com/support/knowledgecenter/en/SS6NHC/com.ibm.swg.im.dashdb.doc/local_overview.html.

Received 14.05.2018

О.А. Урцатьев, канд. техн. наук, старш. наук. співроб., пров. наук. співроб.,
Міжнародний науково-навчальний центр інформаційних
технологій та систем НАН та МОН України, просп. Глушкова, 40, Київ 03187, Україна,
aleksei@irtc.org.ua

ВЕЛИКІ ДАНИ. АНАЛІТИЧНІ БАЗИ ДАНИХ І СХОВИЩА: NETEZZA

Вступ. Стаття є продовженням досліджень Великих Даних і інструментарію, що трансформуються в нове покоління технологій і архітектури платформ БД та сховищ для інтелектуального виводу. У даній частині огляду подано *DB Netezza*. Основну увагу приділено питанням зміни інфраструктури, інструментального середовища і платформи для виявлення необхідної інформації та нових знань з Великих Даних, а початкові відомості про продукт наведено в загальній характеристиці виробу.

Мета. Розглянути та оцінити ефективність застосування інфраструктурних рішень нових розробок в дослідженнях Великих Даних для виявлення нових знань, неявних зв'язків і поглибленого розуміння, проникнення в суть явищ і процесів.

Методи. Інформаційно-аналітичні методи і технології обробки даних, методи оцінки та прогнозування даних, з урахуванням розвитку найважливіших галузей інформатики та інформаційних технологій.

Результати. *Netezza*, як і *Teradata*, являє собою конфігурований, підготовлений до швидкого використання, програмно-апаратний комплекс (*Appliances*), який об'єднує зберігання і обробку даних в одній системі,

спроєктованої і оптимізованої під аналітику. Використана *SN*-середи *MPP*, що не поділяє ресурсів, і головний сервер *SMP* верхнього рівня системи асиметричної *AMPP*-архітектури. Він, крім координування робіт, забезпечує приріст продуктивності при збільшенні числа клієнтських сесій. Значна частина обробки даних виконується практично на рівні дискових контролерів вузлів *SPU* – інтелектуальних оброблювачів шніпетів. Завантаження даних виконується як штатними утилітами, так і засобами інструментів *ETL*. *Netezza* хоче змінити існуючий стан і перейти до реального паралельного завантаження.

Ключова особливість *Netezza* – це мультиплікатор продуктивності аналітичного комплексу, що дає істотне апаратне прискорення виконання запитів *SQL*. Програмовані логічні матриці ПЛМ на вузлах *SPU* виконують потокову обробку даних при зверненні до дисків. Внаслідок цього пам'ять і процесори *SPU* працюють з вже відфільтрованими даними, що значно прискорює подальшу обробку. Ця потокова технологія *FAST Engine™ Framework* досягається програмуванням функцій декомпресії і фільтрації даних, перевірки синтаксису, видимості транзакцій та ін. Набір поточкових механізмів *FAST Engines* дозволяє створювати нові функції стосовно виникаючих завдань.

Особливу увагу приділено простоті використання і мінімальним вимогам в налаштуваннях. У *Netezza* практично нічого адмініструвати. Наприклад, компресія даних виконується автоматично і адаптується до типів даних, не вимагаючи від користувача вказівок на потрібні алгоритми. Немає настройки і проектування БД, вимог до моделі даних. Немає індексів і тюнінгу, в тому числі для виконання *ad-hoc* запитів – продуктивність така, як вона є (з коробки). Управління навантаженням надає функціонал для регулювання ресурсів і пріоритизації виконання запитів в середовищі з багатьма користувачами і в умовах змішаного навантаження. Для збільшення кола завдань і розробки власних аналітичних процесів можливо використовувати *C / C ++, Java, Python, Fortran, R* і підтримку розширяемого *open-source* інтегрованого середовища розробки (плагін для *Eclipse*).

Netezza в першу чергу аналітичний комплекс, у якого досить розвинені засоби аналітики, такі як *Data Mining, OLAP, Hadoop* та ін., проте він має, на думку *Monash Research*, одну з найнижчих в індустрії вартість володіння за терабайт даних користувача.

Основними конкурентами *Netezza* в цій області є *Teradata, Vertica, IBM, Greenplum* та ін.

Висновок. Система *Netezza* інтегрується з існуючими продуктами *IBM*, які додають можливості хмарного зберігання даних *DBaaS IBM* і включають в себе використання різних платформ в середовищі локальних і гібридних хмар, підтримку аналітики в БД на *Apache Spark* автономних СУБД на платформах *Db2* та ще багато іншого. Разом з тим, складається враження, що *Netezza* залишається послідовним прихильником вбудованих в *SPU* перепрограмованих засобів обробки інформації та успішного вдосконалення ПО, так як саме в цьому вона отримала значний приріст продуктивності. У зв'язку з цим напрошується питання, чи не піде *Netezza* по шляху вбудовування в *SPU* продуктивної, працюючої в оперативній пам'яті, програмної конструкції *Apache Spark* зі стандартними бібліотеками для аналітики великих даних на підтримку вже існуючих *frameworks* на ПЛМ? Принаймні, це представляється в рамках *Appliances* більш цікавим, ніж будувати ефективні гібридні системи, здатні як обробляти безліч транзакцій, так і одночасно сканувати великі обсяги інформації в пошуках відповіді на аналітичні запити.

Ключові слова: платформа *Data Warehouse Appliance, AMPP – асиметрична архітектура з масовим паралелізмом, архітектура MPP SN (Shared Nothing), SPU (Snippet Processing Unit) – модулі обробки фрагментів коду, FPGA (ПЛМ) програмовані логічні матриці – компоненти інтелектуальної потокової архітектури передачі запитів (Intelligent Query Streaming® Netezza®), IBM Netezza, SQL-аналітика на Hadoop, підтримка аналітики на Apache Spark у платформах Db2 IBM.*

A.A. Oursatyev, PhD in Techn. Sciences, Leading Research Associate, International Research and Training Centre of Information Technologies and Systems of the NAS and MES of Ukraine, Glushkov ave., 40, Kyiv, 03187, Ukraine, aleksei@irtc.org.ua

BIG DATA. ANALYTICAL DATABASES AND DATA WAREHOUSE: NETEZZA

Introduction. The article is a continuation of the Big Data and tools study, which is transformed into technology of the new generation and architecture of the BD platforms and storage for the intelligent output. In this part the review of DB *Netezza* is presented. The main attention is paid to the issues of changing the infrastructure, the tool environment and the platform for identifying the necessary information and new knowledge from the Big Data, the initial information about the product is given in the product general description.

Purpose. The purpose is to consider and evaluate the application effectiveness of the infrastructure solutions for new developments in the Big Data study, to identify new knowledge, the implicit connections and in-depth understanding, insight into phenomena and processes.

Methods. The informational and analytical methods and technologies for data processing, the methods for data assessment and forecasting, taking into account the development of the most important areas of the informatics and information-technology.

Results. Netezza, like Teradata, is configured and prepared for the quick use. Hardware and software (Appliances), combining data storage and processing in a single system, is originally designed and optimized for analytics. The non-sharing resources MPP environment and the top SMP server of the upper level of the asymmetric AMPP-architecture are used. The SMP-server, in addition to the coordinating work, provides an increase in performance while the number of client sessions growth. A significant part of data processing is performed practically at the level of the SPU nodes disk controllers, the intellectual snippets. Data is loaded using both the regular utilities and ETL tools. Netezza wants to change the existing position and turn to a real parallel download.

A key feature of Netezza is the productivity multiplier of the analytic complex, which provides the significant hardware acceleration for executing SQL queries. Programmable logic matrices FPGA on SPU nodes provide the streaming data processing when accessing disks. As a result, the memory and SPU processors work with already-filtered data, significantly speeding up further processing. FAST Engine™ Framework's usable streaming technology is achieved by programming the decompression and filtering functions, syntax checking, transaction visibility, etc. The set of FAST Engines streaming mechanisms allows to create new functions for emerging problems.

The particular attention is paid to ease the use and minimize the requirements in the settings. Netezza has almost nothing to administer. For example, data compression is performed automatically and adapts to the data types, without requiring the user to specify the necessary algorithms. There are no configuration and designing of the database, no data model requirements. There are no indexes and tuning, including for performing ad-hoc requests - performance is as it is (out of the box). Load management provides functionality for managing resources and prioritizing query execution in a multi-user environment and under mixed load conditions. To expand the range of tasks and develop one's own analytical processes, it is possible to use C / C ++, Java, Python, Fortran, R and support for an expandable, an open-source integrated development environment (plug-in for Eclipse).

Netezza is primarily an analytical complex with highly developed analytics tools, such as Data Mining, OLAP, Hadoop, and others, but, according to Monash Research, it has one of the lowest cost-per-terabyte user data in the industry.

The main competitors of Netezza are Teradata, Vertica, IBM, Greenplum, etc.

Conclusion. Netezza integrates with existing IBM products that add IBM DBPaaS cloud storage capabilities, including the use of various platforms in local and hybrid clouds, as well as support for analytics in the Apache Spark database of standalone DBMS on Db2 platforms and much more. At the same time, it seems that Netezza remains a consistent supporter of the repro-grammable information processing tools and successful software improvements embedded in the SPU, since this is where it received a significant performance boost. In this regard, the question is asked whether Netezza will not go on the way of integrating into the SPU a production Apache Spark software in RAM with standard libraries for analyzing big data in support of the existing frameworks on the FPGA. At the very least, this seems to be more interesting within the framework of Appliances than to build efficient hybrid systems that can both process a lot of transactions and simultaneously scan large amounts of information in search queries for analytical requests.

Keywords: *data warehouse appliance platform, AMPP – asymmetric massively parallel processing, SN (shared nothing) MPP architecture, SPU (snippet processing unit) – modules for processing code fragments, FPGA – programmable logic arrays – Intelligent Query Streaming® Netezza® component, IBM Netezza, SQL analytics on Hadoop, support for analytics on Apache Spark on IBM Db2 platforms*