Эта книга посвящена сообществу SQL Server, #sqlfamily.
Без этого сообщества удивительный продукт SQL Server
не был бы тем, чем он является сегодня.

Оглавление

Предисловие от издательства 9
Об авторе 10
О техническом рецензенте 11
Предисловие 12
Благодарности 13
Вступление 16

Глава 1. Почему SQL Server 2019? 18
Проект «Сиэтл» 19
Проект Aris 20
«Сиэтл» становится SQL Server 2019 23
Модернизация базы данных с помощью SQL Server 2019 25
Начало работы с SQL Server 2019 32

Глава 2. Интеллектуальная настройка производительности 34
Почему используется термин «интеллектуальная настройка производительности»? 34
Интеллектуальная обработка запросов 35
Упрощенное профилирование запросов 72
База данных в памяти 85
Конфликт вставки на последней странице 94
Резюме 96

Глава 3. Новые возможности безопасности 97
Улучшение достигнутых показателей 97
Стратегия постоянного шифрования с защищенными областями (Always Encrypted with Secure Enclaves) 98
Классификация данных 103
Другие новые функции безопасности 118
Резюме 121

Глава 4. Непрерывная доступность, соответствующая требованиям для систем, критичных к сбоям 123
Поддержка онлайн-индекса 124
Улучшения в группе доступности Always On (Always On Availability Group) 131
Ускоренное восстановление базы данных 132
Использование ускоренного восстановления базы данных 142
Ускорение восстановления базы данных. Основные моменты 146
Резюме 151

Глава 5. Современная платформа разработки 152
Языки, драйверы и платформы 153

Графовая база данных ... 156
Поддержка UTF-8 ... 161
Службы машинного обучения SQL Server ... 163
Расширение языка T-SQL ... 170
Резюме ... 178

Глава 6. SQL Server 2019 для Linux ... 179

История SQL Server для Linux ... 179
Что нового в SQL Server 2019 для Linux ... 181
Улучшения платформы и процедуры развертывания ... 182
Поддержка постоянной памяти ... 186
Репликация SQL Server в Linux ... 187
Сбор данных об изменениях (Change Data Capture, CDC) в Linux ... 187
DTC для Linux ... 188
Active Directory и OpenLDAP ... 190
Службы машинного обучения SQL Server и расширяемость в Linux ... 191
Polybase в Linux ... 196
Резюме ... 197

Глава 7. SQL Server и контейнеры ... 198

Зачем нужны контейнеры в SQL Server? ... 198
Как работают контейнеры с SQL Server ... 202
Что нового для SQL Server 2019 ... 210
Подготовительные шаги для использования примеров, иллюстрирующих использование контейнеров с SQL Server ... 213
Развертывание контейнера SQL Server ... 215
Новый способ обновления SQL Server ... 226
Развертывание контейнера как приложения ... 230
Развертывание контейнеров SQL в промышленной среде ... 236
Контейнеры SQL Server в Windows ... 243
Резюме ... 246

Глава 8. SQL Server и Kubernetes ... 247

Что такое k8s? ... 247
Варианты развертывания k8s ... 250
Подготовительные шаги для использования примеров, иллюстрирующих применение SQL Server и Kubernetes ... 253
Развертывание SQL Server на k8s ... 255
Советы по k8s ... 268
Высокая доступность SQL Server на k8s ... 275
Обновление SQL Server на k8s ... 280
Использование Helm Charts ... 284
Группы доступности SQL Server в k8s ... 285
Резюме ... 287

Глава 9. Виртуализация данных в SQL Server ... 289

Что такое Polybase? ... 289
Как работает Polybase ... 294

Подготовительные шаги для использования примеров, иллюстрирующих применение Polybase и SQL Server 302
Использование внешних таблиц 306
Обсуждение внешних таблиц 316
Резюме 317

Глава 10. Кластеры больших данных в SQL Server 319

Зачем нужны кластеры больших данных, и почему они так называются? 322
Что входит в состав кластеров больших данных? 323
Подготовительные шаги для использования обучающих материалов 326
Развертывание кластеров больших данных 327
Архитектура кластера больших данных 338
Использование кластеров больших данных 349
Развертывание и использование приложений 357
Безопасность 357
Высокая доступность 358
Jupyter Books для кластеров больших данных SQL Server 358
Машинное обучение и кластеры больших данных 359
Резюме 365

Глава 11. Голос клиента и миграция 367

Голос клиента 367
А как насчет бизнес-аналитики? 376
Переход на SQL Server 2019 376
Резюме 393

Предисловие от издательства

Отзывы и пожелания

Мы всегда рады отзывам наших читателей. Расскажите нам, что вы думаете об этой книге – что понравилось или, может быть, не понравилось. Отзывы важны для нас, чтобы выпускать книги, которые будут для вас максимально полезны.

Вы можете написать отзыв на нашем сайте www.dmkpress.com, зайдя на страницу книги и оставив комментарий в разделе «Отзывы и рецензии». Также можно послать письмо главному редактору по адресу dmkpress@gmail.com; при этом укажите название книги в теме письма.

Если вы являетесь экспертом в какой-либо области и заинтересованы в написании новой книги, заполните форму на нашем сайте по адресу http://dmkpress.com/authors/publish_book/ или напишите в издательство по адресу dmkpress@gmail.com.

Скачивание исходного кода примеров

Скачать файлы с дополнительной информацией для книг издательства «ДМК Пресс» можно на сайте www.dmkpress.com на странице с описанием соответствующей книги.

Список опечаток

Хотя мы приняли все возможные меры для того, чтобы обеспечить высокое качество наших текстов, ошибки все равно случаются. Если вы найдете ошибку в одной из наших книг – возможно, ошибку в основном тексте или программном коде, – мы будем очень благодарны, если вы сообщите нам о ней. Сделав это, вы избавите других читателей от недопонимания и поможете нам улучшить последующие издания этой книги.

Если вы найдете какие-либо ошибки в коде, пожалуйста, сообщите о них главному редактору по адресу dmkpress@gmail.com, и мы исправим это в следующих тиражах.

Нарушение авторских прав

Пиратство в интернете по-прежнему остается насущной проблемой. Издательства «ДМК Пресс» и Apress Media очень серьезно относятся к вопросам защиты авторских прав и лицензирования. Если вы столкнетесь в интернете с незаконной публикацией какой-либо из наших книг, пожалуйста, пришлите нам ссылку на интернет-ресурс, чтобы мы могли применить санкции.

Ссылку на подозрительные материалы можно прислать по адресу электронной почты dmkpress@gmail.com.

Мы высоко ценим любую помощь по защите наших авторов, благодаря которой мы можем предоставлять вам качественные материалы.

Об авторе

Боб Уорд (Bob Ward) – главный архитектор технической группы Microsoft Azure Data SQL Server, которая ведет разработку всех версий SQL Server. Боб проработал в Microsoft более 26 лет; он участвовал в выпуске каждой версии SQL Server, начиная с версии 1.1, поставляемой с OS/2, и заканчивая SQL Server 2019, включая Azure. Он популярный докладчик многих конференций; часто выступал с докладами о новых версиях SQL Server, его внутреннем устройстве и производительности на таких мероприятиях, как PASS Summit, SQLBits, SQLIntersection, Red Hat Summit, Microsoft Inspire и Microsoft Ignite. Вы можете подписаться на него в соцсетях, ссылки на его профиль: @bobwardms или www.linkedin.com/in/bobwardms.

Боб является автором книги *Pro SQL Server on Linux* («SQL Server на Linux для профессионалов»), выпущенной Apress Media.

О техническом рецензенте

Аарон Бертран (Aaron Bertrand) – технический специалист, обладающий более чем двадцатилетним опытом работы с SQL Server. Он работал непосредственно с несколькими командами разработчиков продуктов Microsoft и хорошо известен тем, что участвует в развитии технических навыков и передаче опыта для широкого сообщества разработчиков. Он автор многих статей, докладчик конференций и модератор форумов, посвященных технологиям баз данных.

Предисловие

Время, в которое мы живем, – это уникальный переломный момент в истории технологий. Наступает золотой век для данных, аналитики и искусственного интеллекта. Темпы роста данных превосходят все когда-либо достигнутые показатели, и прорыв в области цифровых технологий, случившийся благодаря технологиям искусственного интеллекта и машинного обучения, создал неограниченный потенциал для использования данных с целью обеспечения конкурентного преимущества для бизнеса. С резким ускорением процесса цифровизации основной вопрос, с которым мы сейчас сталкиваемся, заключается в том, как воспользоваться преимуществами этого огромного объема данных, чтобы помочь нашим компаниям и сообществам в их преобразованиях.

Мы видим огромный потенциал в области применения возможностей интеллектуального облака и интернета вещей с элементами искусственного интеллекта. SQL Server не имеет аналогов в отрасли по уровню стабильности и системности, который предоставляется разработчикам, инженерам данных и администраторам в разнообразных вариантах: на периферии, в локальной среде, частном облаке и общедоступном облаке. Наше сообщество SQL Server сыграло очень важную роль в этой эволюции, и я не могу в достаточной мере поблагодарить их за поддержку и отзывы за последние 25 с лишним лет.

Версия SQL Server 2019 – выдающаяся, особенная, и я горжусь тем, что сделала наша команда. SQL Server 2019 основан на инновациях, которые были заложены еще в SQL Server 2016 и SQL Server 2017. Несмотря на то что есть несколько новых возможностей, которые улучшат пользовательский опыт наших клиентов, как и ожидается от каждой новой версии SQL Server, я больше всего рад удивительным инновациям, расширяющим навыки, накопленные нашими клиентами на протяжении нескольких десятилетий, в области управления и извлечения информации из больших данных. Это нововведение сыграет решающую роль в поддержке цифровой трансформации у наших клиентов.

Боб Уорд работает в команде SQL Server с самого начала и оказал заметное влияние на продукт. Очень немногие обладают такими широкими и глубокими знаниями, и это объясняет, как ему удается сделать сложные концепции простыми и понятными. Я надеюсь, вы прочитаете эту книгу с удовольствием.

Рохан Кумар (Rohan Kumar),
корпоративный вице-президент Azure Data в Microsoft

Благодарности

В моей жизни есть множество вещей, за которые я испытываю благодарность, и возможность написать эту книгу – одна из них. Я верю, что все мои дарования приходят от Бога, и именно благодаря Его благодати я познал Иисуса Христа. Прежде всего я должен поблагодарить мою очаровательную и талантливую жену Джинджер. Она – мой партнер, моя родственная душа. Она слышала все мои жалобы, видела, как я засиживаюсь допоздна за работой над книгой, а иногда ей приходилось водить машину, чтобы я мог работать над книгой на пассажирском сиденье. Я не знаю никого, кто имеет более сильную веру, чем моя жена, и мне очень повезло, что рядом со мной находится такой человек. В этом году мы отпраздновали 30-летие нашего брака, и я до сих пор наслаждаюсь каждым моментом, проведенным с ней вместе. Я также хочу поблагодарить моих сыновей Троя и Райана. Сейчас Трой живет в Чарльстоне, штат Южная Каролина, где я написал последнюю часть книги, когда приближался ураган Дориан. Трой – человек, которым я восхищаюсь не только за его характер, но и за его неуклонное стремление сделать наш мир лучше. Райан учится на втором курсе юридического факультета Бейлор (Go Bears!). Он продолжает удивлять меня своим интеллектом, целеустремленностью и способностью сохранять уверенность, но все же находит время, чтобы совершенствоваться в игре в гольф. Я также хочу поблагодарить мою маму Аннетт Гибо (Annette Gibaud), которая была и остается для меня живым примером доброты к окружающим ее людям.

Множество людей помогали мне в работе и внесли свой вклад в эту книгу. Я хочу поблагодарить Apress Media за предоставленную мне возможность написать книгу. Джонатан Дженник (Jonathan Gennick) и Джилл Бальзано (Jill Balzano) поддерживали меня на каждом шагу, помогая мне довести дело до конца. И эта книга не была бы написана вовремя, да и вообще вряд ли появилась бы на свет без технического рецензента Аарона Бертрана (Aaron Bertrand). Когда я еще только обдумывал идею этой книги, Аарон был одним из первых, о ком я подумал как о возможном рецензенте, поскольку он обладает глубочайшими знаниями по SQL Server и заслуженной репутацией эксперта в сообществе. Аарон был сверхчеловеком, столь быстро он давал рецензии для каждой главы книги.

В Microsoft прежде всего спасибо Рохану Кумару (Rohan Kumar), Гейлу Шеппарду (Gayle Sheppard) и Асаду Хану (Asad Khan) за предоставленную мне возможность распространять знания о SQL Server 2019, что помогло мне в детальном изучении продукта для написания этой книги. Я также хочу лично поблагодарить двух моих ближайших коллег, Бака Вуди (Buck Woody) и Анну Хоффман (Томас) (Anna Hoffman (Thomas)). Я посетил много разных стран вместе с Баком и Анной в 2018 и 2019 годах, рассказывая об SQL Server, кластерах больших данных и Azure. Они оба поработали над

улучшением моих навыков выступлений и преподавательского мастерства и составили отличную компанию в дороге. Также я очень благодарен прекрасному коллективу Microsoft SQL Server Engineering. Я в восторге от работы с такими умными и профессиональными людьми, многие из которых помогли мне разобраться с техническими подробностями, которые вы найдете в этой книге. Сначала я хочу поблагодарить Славу Окса (Slava Oks) и Трэвиса Райта (Travis Wright), которые помогли мне рассказать историю о проектах «Сиэтл» и «Арис» и сыграли важную роль в продвижении этой версии SQL Server, в том числе новых возможностей, связанных с кластерами больших данных. Конор Каннингем (Conor Cunningham) не устает удивлять меня своими глубокими знаниями об этом продукте, внося весомый вклад в его улучшение.

Настоящие герои этой книги – члены команды инженеров, создавшие эту версию SQL Server и оказавшие мне помощь в работе над различными главами книги. Спасибо Педро Лопесу (Pedro Lopes), Пэм Лахуд (Pam Lahoud), Амиту Банерджи (Amit Banerjee), Брайану Карригу (Brian Carrig), Техасу Шаху (Tejas Shah), Вин Ю (Vin Yu), Сурабху Агарвалу (Sourabh Agarwal), Михаеле Блендеа (Mihaela Blendea), Нелли Густафссон (Nellie Gustafsson), Абиоле Оке (Abiola Oke), Джеймсу Роуланду-Джонсу (James Rowland-Jones), Скотту Конерсману (Scott Konersmann), Стюарту Пэдли (Stuart Padley), Дэвиду Кризу (David Kryze), Роберту Дорру (Robert Dorr), Митчелл Стернке (Mitchell Sternke), Россу Монстеру (Ross Monster), Мадлен Макдональд (Madeline MacDonald), Дилану Грею (Dylan Gray), Джо Сэку (Joe Sack), Шрее Верма (Shreya Verma), Якубу Шимашеку (Jakub Szymaszek), Йоахиму Хаммеру (Joachim Hammer), Рагхаву Каушику (Raghav Kaushik), Парагу Полу (Parag Paul), Панагиотису Антонопулосу (Panagiotis Antonopoulos), Майклу Нельсону (Michael Nelson), Пранджалу Гупте (Pranjal Gupta), Ярупату Джисароджито (Jarupat Jisarojito), Вэйюн Хуангу (Weiyun Huang), Джорджу Рейнье (George Reynya), Умачандару Джаячандрану (UC) (Umachandar Jayachandran (UC)), Сахаю Сайни (Sahaj Saini), Майку Хаббену (Mike Habben), Вакару Пирзаде (Vaqar Pirzada), Рони Чаттерджи (Rony Chatterjee), Вики Харп (Vicky Harp), Алану Ю (Alan Yu), Джеку Ли (Jack Li), Алексею Эксаревскому (Alexey Eksarevskiy), Джею Чо (Jay Choe), Арджeнису Фернандесу (Argenis Fernandez), Кевину Фарли (Kevin Farlee), Арье Библиовичу (Arieh Bibliowicz), Алексу Умански (Alex Umansky), Маттео Таведжиа (Matteo Taveggia), Капилу Такер (Kapil Thacker), Ли Чжан (Li Zhang) и Дону Цао (Dong Cao).

Я также хочу поблагодарить членов групп Microsoft Marketing и CSS за их помощь: Аншула Рампала (Anshul Rampal), Мэтью Барроуза (Matthew Burrows), Марко Хотти (Marko Hotti), Дебби Лайонс (Debbi Lyons), Суреша Кандота (Suresh Kandoth) и Прадипа М. М. (Pradeep M. M.).

Эта книга и моя работа над ней были бы невозможны без таких партнеров, как HPE, DELL и Red Hat, которые позволили мне рассказать своим клиентам о новой версии SQL Server 2019. Спасибо Венди Хармс (Wendy Harms), Биллу Данмиру (Bill Dunmire), Урсу Ренггли (Urs Renggli), Робер-

ту Сондерсу (Robert Sonders), Луи Имершейну (Louis Imershein) и Стефану Бюро (Stephane Bureau) (тому, кто помогает мне с видео). Особая благодарность также Дэвиду ДеВитту за экскурс в историю Polybase, Брендану Бернсу, совершенствовавшему мои знания в области Kubernetes, и Энтони Ноцентино за глубокое знание Linux и контейнеров.

Наконец, спасибо всемирному сообществу SQL Server. Сейчас мы выпускаем новые версии SQL Server чаще и быстрее, чем когда-либо прежде, но вы все равно проявляете огромный энтузиазм каждый раз, когда я выступаю, рассказывая про SQL Server.

Вступление

Как над моей первой книгой *Pro SQL Server on Linux* («SQL Server на Linux для профессионалов»), так и над этой книгой я работал в разных местах. Я путешествовал в 2019 году больше, чем когда-либо в своей жизни. Это означало, что я должен был быть готов работать над своей книгой где угодно и когда угодно: в самолетах, отелях, поездах, в поездках по городам. Я писал главы этой книги в Сиэтле, Лондоне, Манчестере (Великобритания), Нэшвилле, Лас-Вегасе (я побывал там несколько раз), Сан-Антонио, Остине, Хьюстоне, Орландо, Сент-Люсии (дело было в отпуске), Дженеси (Колорадо), Чарльстоне, Бостоне, Дубае, Йоханнесбурге (Южная Африка), Гринвилле (Южная Каролина) и Индианаполисе. И конечно, я работал над книгой допоздна в своем домашнем офисе, в Норт-Ричленд-Хиллз (штат Техас).

Я подумал, что после завершения своей первой книги я вряд ли возьмусь еще за одну, однако я не мог удержаться от искушения поведать историю SQL Server 2019. Эта книга действительно написана «для души». Я вложил душу в обучение, преподавание, преодоление разнообразных препятствий, документирование, тестирование и работу с SQL Server 2019. Книга объединила все это, а также многое другое.

Я написал эту книгу для разработчиков и профессионалов, работающих с данными, которые обладают фундаментальными знаниями по SQL Server и хотят получить исчерпывающий обзор SQL Server 2019 в одной книге. Эта книга содержит множество примеров, рисунков и ссылок, которые помогут вам разобраться в технических подробностях. Я написал ее, чтобы не только дать своей аудитории полное понимание возможностей SQL Server 2019, но также предоставить возможность использовать ее в качестве справочного материала, к которому можно вернуться в любое время.

Хотя каждую из глав этой книги можно читать отдельно и независимо друг от друга, я настоятельно рекомендую вам начать с главы 1, так она знакомит читателей с историей и предысторией выпуска данной версии. Я также привел в ней обзор всех ключевых возможностей SQL Server 2019 и рассказал, почему я считаю его привлекательным продуктом. Начав с первой главы, вы можете просмотреть все главы по порядку или пропустить некоторые из них. Однако, чтобы извлечь максимальную пользу из главы 10, посвященной кластерам больших данных, сначала необходимо прочитать главы 6, 7, 8 и 9.

Книга разбита на следующие главы:

- глава 1, посвященная истории и обзору версии SQL Server 2019;
- главы 2, 3 и 4 – о производительности, безопасности и доступности. В одних только этих главах есть много интересного об SQL Server 2019;

- глава 5 написана для разработчиков;
- главы 6, 7 и 8 посвящены Linux, контейнерам и Kubernetes;
- глава 9 знакомит вас с виртуализацией данных с использованием Polybase;
- глава 10 – большая глава, посвященная большой теме: кластеры больших данных;
- глава 11 завершает книгу, в ней рассказывается о других новых функциях и миграции.

Мне нравится «учиться на примерах», поэтому в этой книге содержится множество примеров. Вы найдете их почти в каждой главе (и в некоторых случаях я объясняю, как использовать уже пройденный материал). Вы можете найти все примеры, приведенные в этой книге, на GitHub, по ссылке на дополнительные материалы к книге на сайте www.apress.com/9781484254189, или в моем репозитории GitHub по адресу https://aka.ms/bobsqldemos (https://github.com/microsoft/bobsql).

Я также рекомендую вам ознакомиться с бесплатными учебными ресурсами, созданными нашей командой по адресу https://aka.ms/sqlworkshops. Они предлагают бесплатное практическое обучение для SQL Server 2019.

В процессе работы над этой книгой я потратил много времени на размышления «чего хотел бы читатель». Я надеюсь, что вы увидите и почувствуете это, когда будете читать книгу. Если у вас возникнут какие-либо вопросы или замечания в процессе работы с книгой, мне хотелось бы услышать о них. Пожалуйста, напишите мне напрямую на bobward@microsoft.com.

Боб Уорд (Bob Ward)
Норт Ричленд Хиллз, Техас
Сентябрь 2019

Глава 1

Почему SQL Server 2019?

В июле 2017 года я регулярно посещал Редмонд (штат Вашингтон) как член команды разработчиков SQL Server. Я живу в Норт Ричленд Хиллз (в штате Техас), и современные технологии позволяют мне выполнять большую часть своей работы удаленно, вне стен офиса, где собрана основная часть команды SQL Engineering. Но я все еще немного приверженец «старой школы» и считаю, что в определенных случаях ничто не сравнится с личным присутствием, когда люди собираются вместе за работой. К июлю 2017 года я проработал в команде SQL Engineering более года, уделяя основное внимание SQL Server 2016 (пример моей работы над SQL Server 2016 можно найти в интернете по ссылке: https://channel9.msdn.com/Events/Ignite/2016/BRK3043-TS).

До этого времени я был членом знаменитой команды Tiger, но в рамках моего визита в 2017 году меня попросили взять на себя новые задачи, чтобы сосредоточиться на предстоящем выпуске SQL Server 2017. В мои задачи входил SQL Server на Linux, что в конечном итоге привело к написанию моей первой книги, *Pro SQL Server on Linux* («SQL Server на Linux для профессионалов», www.apress.com/us/book/9781484241271). Поэтому во время моего визита я начал встречаться и беседовать с различными членами команды об SQL Server 2017 – о повышении производительности, общем наборе новых функций и технических подробностях работы SQL Server в Linux и о контейнерах. Одним из людей, с которыми я говорил на той неделе, был Слава Окс (Slava Oks). Слава – ведущий менеджер по разработке SQL Server и один из изобретателей SQL Server для Linux. Он написал предисловие для книги *Pro SQL Server on Linux* («SQL Server на Linux для профессионалов», www.apress.com/us/book/9781484241271), и в первой главе этой книги рассказывается об истории его участия в проекте. В то время Слава любил приходить в офис рано; когда я нахожусь в Редмонде, я тоже пытаюсь работать в «техасское время» – а это значит, что я тоже прихожу очень рано.

Поэтому мы часто встречались за кофе тогда, когда большинство других разработчиков еще не добирались до офиса, в здании 16 (хотя сейчас наша команда работает в здании 43). Однажды утром, когда мы со Славой говорили об SQL Server 2017, он спросил меня: «Я рассказывал вам о наших планах относительно следующей версии SQL Server, которая выйдет после SQL Server 2017?» Я, конечно, сделал вид, что помню об этом: «Да, Слава,

я слышал об этом, но не знаю подробностей». Затем он пригласил меня прийти на встречу на следующий день, где он объяснил многим из нашей команды инженеров план проекта. Я провел целый год, сосредоточившись на SQL Server 2016, после чего мне поручили погрузиться в SQL Server 2017 и Linux, а теперь Слава хотел, чтобы я узнал о версии, следующей после той, которая пока еще не была выпущена? Конечно, я не собирался отказывать ему, потому что – ну, это Слава Окс. Это может прозвучать так, будто Слава – какой-то страшный человек, однако он один из самых приятных людей, которых я когда-либо знал в Microsoft.

Поэтому когда я начинал собирать информацию об SQL Server 2017, я пошел по такому пути: я собирался узнать, что мы делаем для будущей версии SQL Server с кодовым названием Проект SQL Server *Сиэтл*.

Проект «Сиэтл»

На встрече со Славой на следующий день я быстро узнал, что за несколько часов мы приступили к одному из самых амбициозных усовершенствований SQL Server, которые я когда-либо наблюдал за всю свою карьеру. Я говорю это, уже зная, что мы выводим на рынок SQL Server под Linux, что ранее считалось невозможным.

Слава и команда проекта выбрали для проекта кодовое название «Сиэтл», потому что в качестве кодового названия SQL Server 2017 использовалось «Хельсинки», и команда и искала новое наименование города для кодового названия проекта. По иронии судьбы, никто в Microsoft раньше не использовал название «Сиэтл», поэтому оно быстро прижилось. Я спросил Славу, когда он впервые начал планировать проект «Сиэтл». Я был поражен, услышав ответ: в январе 2017 года. Тот факт, что такие люди, как Слава, Конор Каннингем (Conor Cunningham) и Трэвис Райт (Travis Wright), планировали проект «Сиэтл», работая над завершением SQL Server 2017 на Linux, стал свидетельством их преданности команде, их стремления удерживать SQL Server на позиции лидера инноваций в отрасли баз данных.

Трудно было поверить, что мы могли так быстро запланировать нечто большее, после того как предоставили так много полезных и инновационных функций в SQL Server 2016 и SQL Server 2017.

В SQL Server 2016 мы добавили новые возможности диагностики производительности с помощью Query Store, а именно новые функции для разработчиков, такие как временные таблицы и интеграция с JSON. Мы повысили безопасность работы благодаря технологии Always Encrypted, динамическому маскированию данных и защите на уровне строк. И мы представили две новые инновационные возможности, выходящие за пределы «обычных» функций для реляционной системы управления базами данных. Одной из них была интеграция языка R для моделей машинного обучения. Второй была интеграция с системами Hadoop при помощи *Polybase* (что в итоге приведет к чему-то большему в 2019 году; однако я

забегаю вперед). Создание возможностей для включения новых сценариев, таких как машинное обучение и большие данные, привело меня и других сотрудников Microsoft к мысли о том, что SQL Server – уже не просто механизм управления реляционными базами данных, а *платформа данных*.

Однако, чтобы создать современную и полнофункциональную платформу данных, нам нужно было расширять возможности приложений в системах, отличных от Windows Server. Это привело к появлению в SQL Server 2017 поддержки Linux и Docker-контейнеров. Запуск на Linux и использование контейнеров стали очень большим шагом вперед для Microsoft, но SQL Server 2017 также включал другие возможности, такие как адаптивная обработка запросов (Adaptive Query Processing), автоматическая настройка, графовая база данных, группы доступности вне кластеров и интеграция с Python, в дополнение к поддержке языка R для служб машинного обучения.

Учитывая все эти новые возможности, как мы можем за короткий период времени спланировать и создать что-то более новое, замечательное и интересное, чем SQL Server 2016 и 2017? Я задал себе этот вопрос, внимательно слушая коллег во время моей первой встречи с командой проекта «Сиэтл». В первые несколько минут меня познакомят с идеей, которая, когда ее позже объявят общественности, будет считаться довольно радикальной. И это новшество было «краеугольным камнем» проекта «Сиэтл», который имеет собственный код проекта: *Aris*.

Проект Aris

В январе 2017 года Рохан Кумар, корпоративный вице-президент Azure Data, поставил перед Славой и руководством команды разработчиков SQL Server задачу разобраться, как интегрировать SQL Server с большими данными. Большие данные – это термин, широко используемый в отрасли и относящийся к системе, которая может обрабатывать большие объемы данных, обычно с использованием распределенной масштабируемой вычислительной платформы. Мне лично нравится определение термина «большие данные», сформулированное моим коллегой Баком Вуди (Buck Woody): «любые данные, которые вы не можете обработать в нужное время с помощью имеющихся у вас технологий». В течение многих лет в качестве системы обработки больших данных наиболее часто выбиралась платформа Hadoop. Итак, в течение нескольких месяцев весной и летом 2017 года команда обращалась к Трэвису Райту (Travis Wright) за консультациями о том, как воплотить в жизнь идею интеграции больших данных. Летом 2017 года у нашей команды Azure Data было несколько проектов с кодовыми названиями «Polaris», «Socrates» (Сократ) и «Plato» (Платон). Я спросил Славу: как ты выбрал кодовое название для проекта – Aris? И получил ответ: Сократ был наставником известного греческого философа Платона, а учеником Платона был Аристотель. Учитывая, что Aris – это часть име-

ни «Аристотель», а также оно входит в состав кодового названия проекта «Polaris», это новое название, Aris, нашло отклик у всех членов команды и у нашего руководства.

Поскольку интеграция с большими данными подразумевает *некоторое* отношение к Hadoop, Трэвис провел несколько встреч с командой, которая представила решение Polybase для SQL Server 2016 и хранилища данных Azure. Идея Polybase заключалась в том, чтобы позволить пользователям SQL Server запрашивать (и принимать) данные из системы Hadoop, используя привычный язык запросов T-SQL, хорошо знакомый нашим пользователям. Кроме того, вместо того чтобы строить простую систему извлечения данных, Polybase могла бы использовать возможности распределенных вычислений, которые существуют в Azure Data Warehouse и Analytics Platform System (ранее известной как Parallel Data Warehouse), для сокращения вычислений и распределенной обработки запросов, чтобы достигнуть увеличения производительности за счет масштабирования при работе с большими наборами данных в целевой системе Hadoop. Я никогда не видел, чтобы Polybase «взлетела» в SQL Server 2016 и 2017, поскольку интеграция систем Big Data Hadoop с реляционными системами, такими как SQL Server, была непростой задачей. Polybase требует значительных усилий по установке и настройке, а ее модель политики безопасности отличается от подходов к защите данных, используемых в системах Hadoop и SQL Server. Кроме того, реализация сокращенных («выталкивающих») вычислений опиралась на концепцию MapReduce, для чего требовалось установить Java на том же компьютере, на котором были размещены службы SQL Server и Polybase. Тем не менее архитектура и концепции интегрированных систем SQL Server и Big Data были пригодны для создания чего-то большего (включая расширение T-SQL под названием EXTERNAL TABLE). Если бы нам удалось упростить историю развертывания и настройки Polybase и добавить больше поддерживаемых источников данных, то такое решение могло бы стать более популярным в отрасли. Кроме того, Трэвис очень быстро понял, что если вы хотите, чтобы вас серьезно воспринимали в мире обработки больших данных, вам нужно рассмотреть другую технологию под названием *Spark*.

Вооружившись этими знаниями, Слава, Трэвис и основная группа членов команды, создавшей SQL Server для Linux, поставили перед собой задачу создать прототип интеграции SQL Server с большими данными, включая Spark. Они устроили многодневную встречу в большом конференц-зале и назвали ее «Хакатон Aris». Членами команды хакатона были Слава Окс (Slava Oks), Трэвис Райт (Travis Wright), Скотт Конерсманн (Scott Konersmann), Стюарт Падли (Stuart Padley), Майкл Нельсон (Michael Nelson), Пранджал Гупта (Pranjal Gupta), Джарупат Джисарохито (Jarupat Jisarojito), Вейюн Хуан (Weiyun Huang), Джордж Рейня (George Reynya), Дэвид Крайз (David Kryze), Умачандар Джаячандран (Umachandar Jayachandran) из Калифорнийского университета и Сахадж Сайни (Sahaj Saini). К моменту завер-

шения хакатона у них был рабочий кластер, который объединил существующую в SQL Server функциональность Polybase с технологией Spark. На рис. 1.1 показана примерная схема кластера, созданного этой командой.

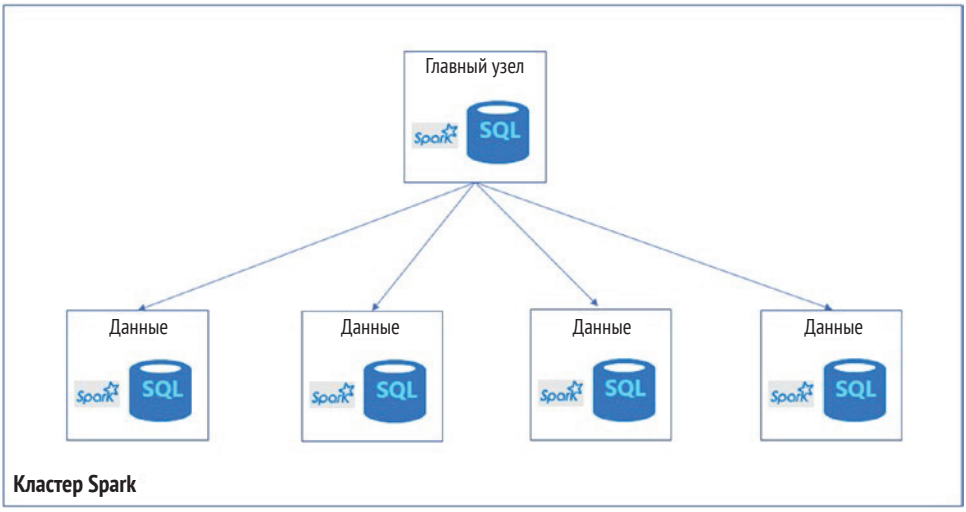

Рис. 1.1. Первый кластер Aris

В своем прототипе они создали кластер Hadoop, включающий компоненты для Apache Spark и HDFS, а также SQL Server Polybase. Они использовали Spark для потоковой передачи данных в узлы данных, а затем применили Polybase для объединения данных на главном узле SQL Server с данными, поступающими со Spark в HDFS. Идея прототипа состояла в том, чтобы доказать, что они могут интегрировать Spark, Hadoop и SQL Server.

Примерно в это же время Трэвис разговаривал с инженерами, которые недавно присоединились к команде. Это были сотрудники из компании Metanautix, которую приобрела Microsoft. Благодаря этому приобретению у нашей команды была технология для подключения к ряду источников данных, в том числе ORACLE, SQL Server, Teradata и MongoDB, через ODBC. Команда подумала, что если мы сможем использовать эту технологию в проекте Aris, то в результате получим довольно интересный пример *виртуализации данных*. SQL Server теперь может быть центром доступа к данным на разных платформах данных и в разных системах без необходимости перемещения данных на SQL Server (с помощью таких методов, как извлечение, фильтрация и загрузка данных (Extract, Transform, and Load, ETL)).

Прежде чем мы смогли предоставить программное обеспечение, которое пользователи могли бы попробовать и использовать, нам нужно было выбрать платформу для запуска всех этих компонентов. Нам была нужна платформа, которая позволила бы легко развернуть все программное обеспечение, включая Polybase, Hadoop и Spark; обеспечить управляемость и безопасность, гибкое масштабирование и высокую доступность. Казалось

логичным использовать контейнеры, учитывая простоту их развертывания, и в SQL Server 2017 мы реализовали поддержку SQL Server в контейнерах. Следующим шагом был выбор Kubernetes в качестве платформы для создания кластера, в котором работают эти контейнеры. Kubernetes быстро приобретал популярность как платформа, поддерживающая распределенные вычисления и масштабируемую производительность. Мы знали, что Linux является предпочтительной ОС для работы с системами Kubernetes и Hadoop, и, поскольку SQL Server уже работает на Linux, он хорошо подходит для создания такой платформы.

Итак, в конце 2017 года наша команда приступила к созданию кластера Aris, который позволил бы реализовать концепцию виртуализации данных и при этом интегрироваться с технологиями больших данных, такими как Spark и HDFS. С самого начала наша команда решила, что все это нужно поставлять как единое решение. Это означает, что если вы приобретаете SQL Server, мы устанавливаем все эти компоненты как часть лицензии (не знаю, будет ли это отдельным видом лицензии, но все это будет включено в SQL Server). Конечный продукт, который вы видите сейчас, –SQL Server 2019 и то, что мы называем *кластерами больших данных* (Big Data Clusters, BDC), – включает в себя гораздо больше, чем ранние прототипы Aris, но идея и концепция, положенные в их основу, одинаковы: предоставить простую в развертывании платформу виртуализации данных, обеспечивающую масштабируемую производительность, безопасность и управляемость.

«Сиэтл» становится SQL Server 2019

В то время как концепция кластеров Aris и больших данных была масштабной, инновационной и, честно говоря, немного пугающей, каждая новая версия SQL Server включала усовершенствования в нескольких областях. Это производительность, безопасность и доступность, три области, которые Конор Каннингем (Conor Cunningham) часто называет «жизненно важными аспектами SQL Server». Наша команда также включила возможность работы с SQL Server на Linux начиная с версии SQL Server 2017. Несмотря на то что первая версия SQL Server для Linux была впечатляющей, оставалось еще несколько возможностей, имеющихся в версии SQL Server для Windows, которые еще не были реализованы для Linux и которые также было необходимо добавить в версию для Linux. Мы также знали, что у контейнеров большое будущее – в том смысле, что они являются перспективным направлением для развертывания и запуска приложений, и SQL Server – не исключение из этого правила. Таким образом, в этом направлении необходимо было проделать определенную работу, включая исследование новых сценариев с кластерами Kubernetes (а не только решение, позволяющее создавать кластер больших данных).

В такой продукт, как SQL Server, вносят свой вклад множество команд разработчиков. Наша команда Enterprise (также известная как Tiger Team)

Конец ознакомительного фрагмента.
Приобрести книгу можно
в интернет-магазине «Электронный универс»
(e-Univers.ru)