СБЕРБАНК
Всегда рядом

БИБЛИОТЕКА СБЕРБАНКА

Джудит Гурвиц
Алан Ньюджент
Ферн Халпер
Марсия Кауфман

Просто о больших данных

Judith Hurwitz, Alan Nugent,
Dr. Fern Halper and Marcia Kaufman

Big Data For Dummies

Джудит Гурвиц, Алан Ньюджент
Ферн Халпер, Марсия Кауфман

Просто о больших данных

Перевод с английского

СБЕРБАНК

Москва • 2015

УДК 004.738.5
ББК 32.973.202
Г95

Гурвиц, Джудит.

Г95 Просто о больших данных / Гурвиц Джудит, Ньюджент Алан, Халпер Ферн, Кауфман Марсия : [перевод с английского]. — Москва : Эксмо, 2015. — 400 с. — (Библиотека Сбербанка. Т. 58).

ISBN 978-5-699-85806-4

Большие данные — один из важнейших технологических трендов, который кардинально меняет возможности использования информации в бизнесе. Многим понятие Big Data пока в новинку, поэтому, прежде чем с ним работать, необходимо уяснить суть как технических, так и бизнес-требований к применению этих технологий.

Вы узнаете:

- как наиболее эффективно работать с данными;
- как преобразовывать большие объемы данных в полезную информацию;
- какие архитектурные решения нужны для работы с большими данными.

В книге освещается широкий круг вопросов, которые помогут сформировать представление о больших данных и определить, для каких целей их можно использовать в конкретной организации.

УДК 004.738.5
ББК 32.973.202

© 2013 by John Wiley & Sons, Inc., Hoboken, New Jersey
This translation published under license with the original publisher John Wiley & Sons, Inc.
© Перевод. АНО ДПО «Корпоративный университет Сбербанка», 2015
© Оформление. ООО «Издательство «Эксмо», 2015
Все права защищены.

ISBN 978-5-699-85806-4 (рус.)
ISBN 978-1-118-50422-2 (англ.)

Оглавление

Предисловие 7

Введение 9

Часть I. Приступая к работе с большими данными 15

Глава 1. Введение в большие данные 17

Глава 2. Типы больших данных 39

Глава 3. Новое использование старых решений: распределенные вычисления 53

Часть II. Технологические основы больших данных 63

Глава 4. Компоненты технологии больших данных 65

Глава 5. Виртуализация и распределенные вычисления 83

Глава 6. Облачные сервисы и большие данные 95

Часть III. Управление большими данными 111

Глава 7. Операционные базы данных 113

Глава 8. Основные принципы MapReduce 133

Глава 9. Мир Hadoop 145

Глава 10. Экосистема Hadoop 157

Глава 11. Программно-аппаратные комплексы и хранилища больших данных 167

Часть IV. Аналитика и большие данные 179

Глава 12. Что такое аналитика больших данных 181

Глава 13. Обзор текстовой аналитики в контексте больших данных ... 195

Глава 14. Специальные подходы к анализу больших данных 213

Часть V. Применение больших данных 229

Глава 15. Интеграция источников данных 231

Глава 16. Работа с потоками данных в режиме реального времени и обработка сложных событий 247

Глава 17. Использование больших данных в операционной деятельности 257

Глава 18. Применение больших данных в вашей организации 269

Глава 19. Безопасность и управление средами больших данных 285

Часть VI. Решения на основе больших данных 297

Глава 20. Важность больших данных для бизнеса 299

Глава 21. Анализ данных в движении: практические примеры 309

Глава 22. Оптимизация бизнес-процессов с помощью анализа больших данных: практические примеры 321

Часть VII. Раздел «десятка лучших» 333

Глава 23. Десять лучших способов работы с большими данными 335

Глава 24. Десять основных источников больших данных 343

Глава 25. Десять рекомендаций: что следует и чего не следует делать в работе с большими данными 349

Глоссарий 354

Указатель 369

Об авторах 390

Благодарности 393

Предисловие

Дорогие коллеги!

С огромным удовольствием представляю вам прекрасную книгу Джудит Гурвиц с соавторами о больших данных, углубленной аналитике и о том, как с их помощью можно трансформировать бизнес. Книга — настоящая энциклопедия современных подходов, технологий, решений и концепций, и каждый — от менеджера до ИТ-архитектора и от аналитика до пользователя — найдет здесь что-то для себя.

Мы издали эту книгу, чтобы компенсировать дефицит качественной литературы на русском языке об использовании, анализе и внедрении больших данных — теме, которая уже не первый год волнует очень и очень многих и которая нашла отражение в стратегических инициативах нашего банка. Та литература, которая имеется сейчас на русском языке по этой тематике, либо не обладает достаточной глубиной изложения, либо рассчитана на людей, имеющих серьезную подготовку в области ИТ-технологий и математики. Требовалась книга, которая станет проводником в мире современных технологий для широкого круга читателей, своего рода поваренной книгой и справочником, охватывающим все аспекты подхода Big Data.

Джудит и ее коллеги проделали огромную работу по сбору и систематизации информации в области, которая меняется с необычайной скоростью. Само понятие больших данных за последние пару лет стало такой же неотъемлемой частью нашей жизни, как электричество или интернет.

Книга, повторюсь, во многом уникальна. Джудит — одна из немногих авторов, которые откровенно заявляют, что в больших данных нет

«серебряной пули» — некоего универсального решения, которое пригодно для любой цели. Задача всегда первична — технологии же идут вслед за ней.

В книге вообще можно найти «освежающую» информацию. Например, о том, что подходы углубленной аналитики, в том числе машинное обучение, с помощью которых и предполагается искать скрытые закономерности, появились не с рождением понятия «большие данные», а много десятилетий назад, но высший менеджмент долгое время не уделял им достаточного внимания.

А также о том, что после завершения анализа больших данных потребуется интегрировать результаты этого анализа в бизнес-процессы компании, если вы хотите увеличить прибыль. Часто это может оказаться намного сложнее, чем собрать и проанализировать большие данные, потому что потребуется объединить усилия множества людей в самых разных подразделениях и, более того, трансформировать сам бизнес и отношение к данным. Это не та задача, которую сможет решить горстка людей в ИТ или самые талантливые аналитики — здесь требуются усилия всех сотрудников компании.

Первый шаг на этом пути — ознакомиться с теми возможностями, которые открывают для нас новые технологии и подходы.

Вадим Валерьевич Кулик,
Заместитель Председателя Правления
ПАО Сбербанк

Введение

Большие данные — один из важнейших технологических трендов, которые кардинально меняют возможности использования информации в бизнесе для повышения удовлетворенности клиентов качеством обслуживания и для преобразования бизнес-моделей. Как наиболее эффективно работать с данными? Как преобразовывать большие объемы данных в полезную информацию? Из этой книги вы узнаете, как технологические изменения в программном обеспечении, оборудовании и моделях взаимодействия позволяют использовать данные по-новому.

Большие данные сами по себе не являются единым рынком. Это скорее сформировавшееся в течение длительного времени сочетание технологий управления данными. Технологии больших данных позволяют организациям хранить огромные объемы данных, управлять ими и обрабатывать их так, чтобы своевременно получать информацию, необходимую для принятия решений. Главное в работе с большими данными — руководствоваться бизнес-требованиями и кругом решаемых задач. Большинство компаний только начинает внедрять большие данные. Многие экспериментируют с методами сбора больших массивов данных и выявления скрытых закономерностей, которые могут быть первым сигналом важных тенденций — например, изменений в потребительских предпочтениях — или появления новых условий, которые необходимо своевременно учесть.

Современные технологии работы с большими данными открывают новые возможности в самых разных отраслях. К примеру, на производстве обработка данных, поступающих с датчиков, позволит определить, какие изменения необходимо внести в производственные

процессы для предотвращения аварий. В розничной торговле анализ данных в реальном времени даст возможность уже в момент совершения сделки убеждать покупателя приобрести сопутствующие товары. Технологии больших данных могут применяться в медицине для диагностики и определения методов лечения.

Тем не менее большие данные — это не изолированное решение. Внедрение систем больших данных требует инфраструктуры, которая обеспечивает возможности для их масштабирования, распределения и управления ими. Соответственно, для реализации всех преимуществ этой новой технологии необходима согласованная бизнес-стратегия и техническая политика компании.

Таким образом, для эффективного применения больших данных важно сначала научиться ориентироваться в этой сфере, понять особенности инновационных программных платформ, таких как Hadoop и MapReduce, и механизмов управления базами данных, исследовать опыт организаций, успешно использующих большие данные в своей деятельности. В этом и призвана помочь настоящая книга, представляющая собой обзор ключевых сведений о данном технологическом тренде и его возможностях.

О чем эта книга

Большие данные многим пока в новинку, поэтому, прежде чем с ними работать, необходимо уяснить суть как технических, так и бизнес-требований к применению этих технологий. Сведения о больших данных требуются специалистам различного профиля. Одних интересуют технические подробности, другие изучают экономические аспекты использования больших данных; руководителям высшего звена важно понять, как большие данные могут влиять на процесс принятия решений. Внедрение среды больших данных требует как архитектурного, так и бизнес-подхода, а также большого объема планирования.

В книге освещается широкий круг вопросов, которые помогут сформировать представление о больших данных и определить, для каких целей их можно использовать в конкретной организации.

- Какие архитектурные решения нужны для работы с большими данными? Как обеспечить бесперебойное управление большими объемами данных?

- Когда нужно интегрировать результаты анализа больших данных в хранилище данных?
- Какие требования к безопасности и управлению предъявляет использование больших данных? Как обеспечить надежную защиту корпоративной информации?
- В чем преимущества различных технологий обработки данных? Когда выбор технологии является неотъемлемой частью стратегии по внедрению больших данных?
- Какие типы источников больших данных могут использоваться в аналитике? Как применять различные виды аналитики для решения бизнес-задач?

Для кого эта книга

Очевидно, что при всей своей многогранности тема больших данных достаточно узкая и требует некоторых специальных знаний. Итак, данная книга адресована:

- тем, кто интересуется информационными технологиями и старается следить за новыми тенденциями, но пока мало осведомлен о больших данных;
- тем, кто занимается бизнесом, напрямую не связанным с информационными технологиями, однако стремится узнать о возможностях применения больших данных;
- тем, кто профессионально занимается информационными технологиями и хочет получить полное представление об особенностях больших данных.

Как устроена эта книга

Книга состоит из семи частей, которые можно читать в любом порядке.

Часть I. Приступая к работе с большими данными

В этой части вводятся фундаментальные понятия, необходимые для формирования общего представления о больших данных как с технической, так и с экономической точек зрения.

Часть II. Технологические основы больших данных

Часть II адресована специалистам в области информационных технологий и руководителям и описывает различные типы компонентов больших данных и концепции, лежащие в основе этой технологии, а также требования к инфраструктуре, которые обеспечивают удобство работы с большими данными.

Часть III. Управление большими данными

Часть III также ориентирована на ИТ-специалистов и менеджеров. В ней более подробно рассказывается о возможностях различных баз данных и инновационных технологиях, таких как MapReduce и Hadoop.

Часть IV. Аналитика и большие данные

Как анализировать огромное количество данных, которые поступают в инфраструктуру больших данных компании? В этой части рассматриваются различные виды аналитики, которые применяются в работе с большими данными.

Часть V. Применение больших данных

В этой части подробно рассказывается об управлении данными, в том числе об особенностях их использования, обеспечения безопасности и конфиденциальности данных.

Часть VI. Решения на основе больших данных: реальные примеры

В этой части рассматриваются особенности работы с большими данными в организациях различного профиля.

Часть VII. Раздел «десятка лучших»

В этой части собраны ключевые рекомендации по работе с большими данными, основанные на опыте компаний, которые уже их внедрили.

Глоссарий

Большинство специальных терминов, встречающихся в книге, объясняются в тексте. В сжатом виде эти определения сведены в глоссарий.

Условные обозначения



Критически важная информация.



Детали, которые не следует упускать из виду.



Практические рекомендации, на которые стоит обратить внимание.



Более подробные сведения, адресованные техническим специалистам.

С чего начать

Первые четыре главы посвящены общему обзору больших данных и содержат основные сведения, необходимые для того, чтобы ориентироваться в этой теме. Главы 5 и 6 знакомят с ключевыми вопросами инфраструктуры, применяемой в работе с большими данными. Следующие восемь глав рассматривают различные типы структур данных, которые являются основополагающими для больших данных.

Главы и части можно читать подряд, а можно пользоваться книгой как справочником. Более подробная информация по ряду вопросов, затронутых в данном издании, содержится в блогах и статьях на вебсайте компании Hurwitz & Associates по адресу: www.hurwitz.com.

Часть I

Приступая к работе с большими данными

В ЭТОЙ ЧАСТИ:

- Эволюция управления данными
- Определение больших данных и их технологических компонентов
- Типы больших данных
- Структурированные и неструктурированные данные
- Данные в реальном времени и неоперативные данные
- Распределенные вычисления при работе с большими данными

Введение в большие данные

В ЭТОЙ ГЛАВЕ:

История управления данными
Почему большие данные важны для бизнеса
Как эффективно применять большие данные в бизнесе
Основополагающие элементы больших данных
Перспективы больших данных

Управление данными и их анализ дают любой организации, независимо от ее размера и области деятельности, значительные преимущества и открывают большие перспективы. Однако с развитием экономики в целом и каждой отдельной компании сбор информации о потребителях, продуктах и услугах становится сложнее. Когда речь идет о небольшом количестве клиентов, которые приобретают один и тот же продукт одним и тем же путем, вести учет таких сведений нетрудно. Но со временем растут и компании, и их рынки сбыта; конкуренция заставляет создавать новые линейки продукции, способы продвижения и продажи товаров также диверсифицируются — и объем информации возрастает многократно. Трудности с использованием данных возникают не только в бизнесе: например, научно-исследовательским организациям не хватает вычислительных мощностей для работы со сложными моделями, обработки изображений и других источников научных данных.

Безусловно, работа с данными сопряжена с многочисленными сложностями. Часть данных структурирована и может храниться в традиционной реляционной базе данных (БД). Однако такие данные, как документы, отзывы клиентов, фотографии и видеозаписи, являются неструктурированными. Также необходимо учитывать воз-

можность использования новых источников данных. Это могут быть данные, автоматически генерируемые различными устройствами, например датчиками. Огромное количество информации генерируется с участием человека, к примеру данные из социальных сетей или сведения о действиях пользователя на веб-сайте, и очевидно, что распространение новых, более мощных мобильных устройств в сочетании с расширением возможностей доступа к глобальным сетям приведут к созданию новых источников данных.

Управление каждым источником данных в отдельности и поиск в нем необходимой информации не представляют большой сложности — затруднения возникают тогда, когда нужно анализировать разноплановые данные из ряда источников в совокупности. Традиционные методы управления данными и работы с ними оказываются непригодны, когда речь идет об огромном количестве информации, представленной в различных формах. Конечно, большие объемы данных существовали и прежде, но к настоящему времени они значительно увеличились, не говоря уже о том, что возросло разнообразие видов информации и приобретает все большее значение степень их актуальности. Кроме того, появляются принципиально новые способы использования информации. Таким образом, сложилась потребность в новом подходе к управлению данными. Это и преимущество, и трудность работы с большими данными. Рассмотрим, чем же было обусловлено появление больших данных как нового явления в информационных технологиях и как они могут применяться в бизнесе.

Эволюция управления данными

Насколько проще было бы осваивать новые тенденции, если бы каждая инновация в управлении данными была независима от прошлых достижений! Однако в основе большинства новых способов управления данными, как вносящих кардинальные изменения, так и внедряемых поэтапно, лежат методы, применявшиеся и ранее. Эволюцию управления данными, как правило, рассматривают с точки зрения программного обеспечения, но на самом деле этот вопрос требует комплексного подхода. Управление данными учитывает технологические достижения в области аппаратных средств и систем хранения данных, сетей и вычислительных моделей, такие как виртуализация

и облачные вычисления. Конвергенция новейших технологий и снижение их стоимости (начиная со стоимости хранения и заканчивая стоимостью вычисления циклов) коренным образом изменили среду обработки данных и сделали доступными ряд возможностей.

Все эти технологические факторы в совокупности изменяют способы использования данных и управления ими, и наиболее актуальная тенденция, возникшая благодаря указанным факторам, — это большие данные. Что же собой представляют большие данные и почему они так важны? Дефиницию термина приведем позже. Для начала определим большие данные как любой источник данных, который характеризуется по меньшей мере тремя общими параметрами:

- чрезвычайно большими объемами данных;
- чрезвычайно высокой скоростью обработки данных;
- чрезвычайным разнообразием данных.

Большие данные ценны тем, что позволяют организациям хранить большие объемы информации, управлять ими и обрабатывать их с нужной скоростью и в требуемое время в соответствии с актуальными бизнес-задачами. Прежде чем перейти к деталям, ненадолго остановимся на истории управления данными и проследим, какие процессы привели к появлению больших данных. Технология больших данных возникла не на пустом месте — это результат научно-технических достижений последних пятидесяти лет.

Сегодня в управлении данными наступил переломный момент. Позади осталась эпоха, когда технологии в бизнесе применялись только для решения конкретных узкоспециальных задач (например, для ведения статистики продаж). Теперь в распоряжении любой организации значительно больше информации из значительно большего числа источников. Данные — это потенциальная «золотая жила», но как и на прииске, чтобы добраться до настоящего золота, нужно обработать внушительное количество материала. В этом и технологическая сложность: как извлечь ценную информацию из таких данных, если в них так трудно выявить тенденции и закономерности, которые наиболее важны для принятия решений? Как наладить в организации эффективную работу с огромным объемом данных? Прежде чем ответить на эти вопросы, посмотрим, как связаны между собой разные этапы развития управления данными.

Этапы развития управления данными

Управление данными всякий раз переходило на новый этап в своем развитии, когда возникала необходимость решать задачи, связанные с определенным типом данных. Здесь действует причинно-следственная связь: как только на рынок выходит технологическая инновация, появляются различные способы с ней работать, в свою очередь стимулирующие создание новых технологических решений. Когда появились реляционные базы данных, потребовался комплекс средств, которые позволяли бы изучать связи между элементами данных. Когда организации начали накапливать неструктурированные данные, для их обработки понадобились новые инструменты анализа на основе естественного языка. Интернет-поисковики обнаружили, что располагают колоссальными объемами информации, из которой можно извлечь выгоду, но для этого нужно было внедрить инновационные подходы и средства.

Технология больших данных — это кульминация полувекового развития процессов управления данными. Чтобы понять, на какой стадии мы сегодня находимся, важно знать, что лежало в основе предыдущих этапов, и не забывать, что при переходе с одного этапа на другой инструменты, технологии и методы, которые применялись раньше, зачастую не выходят из употребления, а напротив, трансформируются для решения новых задач.

ЭТАП 1. Создание управляемых структур данных

Когда компьютерные системы в конце 1960-х годов вышли на коммерческий рынок, данные хранились в так называемых плоских файлах, которые состояли из однородных записей, без внутренней иерархии данных. Получить подробные сведения из таких баз данных можно было только путем перебора. Извлекать информацию из такого файла крайне трудоемко, и вот почему. Представьте, что вы заполняете файл с данными сотрудников и указываете дату рождения в формате ДД.ММ.ГГ и сначала прописываете фамилию, а потом имя, а ваш коллега, обрабатывая вторую половину компании, сохранил дату в формате ГГГГ.ММ.ДД. и указывает фамилию после имени, а не перед, как вы. Программист должен это учитывать в своей работе. Что это неудобно, известно любому человеку, который сводил данные, полученные хотя бы от 2–3 человек для подготовки отчета. В 1970-х все изменилось — была изобретена реляционная модель данных и по-

явились реляционные системы управления базами данных (реляционные СУБД), которые позволили структурировать данные и повысить производительность работы с информацией, а главное, реляционная модель добавила уровень абстракции (язык структурированных запросов (SQL), генераторы отчетов и утилиты управления данными). Благодаря этому программистам стало проще удовлетворять растущие потребности бизнеса в извлечении необходимой информации из имеющихся данных.

Благодаря реляционной модели стало возможным создание экосистемы программного обеспечения, разработанного большим количеством перспективных компаний. Таким образом была удовлетворена растущая потребность бизнеса в улучшении организации своих данных, появилась возможность сравнивать транзакции в разных областях. Кроме того, руководители стали применять такие сведения, как результаты сопоставления данных инвентаризации с данными заказов клиентов, в процессе принятия решений. Но вместе с расширением возможностей новые технологии породили и новые проблемы. Хранение все увеличивающихся объемов информации обходилось дорого, а доступ к таким данным был медленным. Более того, данные часто дублировались, так что фактическую ценность информации для бизнеса было сложно определить.

Возникла острая необходимость в новых технологиях для поддержки реляционной модели. Была создана модель «сущность-связь» (ER), которая добавляла дополнительную абстракцию, повышающую эффективность использования данных. В этой модели каждый объект был определен независимо от его применения, таким образом, разработчики могли выстраивать новые отношения между источниками данных, не прибегая к сложному программированию. Это был огромный шаг вперед, который позволил разработчикам расширить границы технологии и создать более продвинутые модели, обеспечивающие сложные методы объединения объектов. Рынок реляционных баз данных пережил в семидесятые настоящий бум и остается динамичным до сих пор. Такие БД широко применяются для управления транзакциями высоко структурированных данных.

Когда объем данных, которыми требовалось управлять, вырос до состояния неконтролируемого, положение спасло появление хранилищ данных. Они позволили ИТ-подразделениям компаний выбирать ряд параметров сохраняемых данных таким образом, чтобы упростить анализ информации. Благодаря уменьшению объема хранимых данных и их

отбору в соответствии с потребностями конкретной сферы деятельности, хранилища данных помогали компаниям справляться с возрастающим количеством информации, которую требовалось анализировать. Разделение процессов поддержки принятия решений на оперативном и стратегическом уровнях управления компанией положительно сказались на производительности. Кроме того, в хранилищах данных часто содержатся данные за предыдущие годы, необходимые для оценки эффективности работы организации, выявления тенденций и закономерностей, в том числе в моделях поведения потребителя. Поступающая из различных источников и агрегированная в хранилище данных информация также может быть использована для анализа. Коммерческие реализации хранилищ данных появились на рынке в 1990-е годы. На сегодняшний день и в системах управления контентом, и в хранилищах данных реализованы такие преимущества, как масштабируемость оборудования, технологии виртуализации и возможности создания интегрированных программно-аппаратных комплексов.

Иногда такие хранилища данных сами по себе были слишком сложными и большими и не обеспечивали необходимую скорость и оперативность обработки информации. Решением стало дальнейшее сужение объема хранимых данных с помощью витрин данных — узкоспециализированных подсистем хранения. Такие малые хранилища данных, в отличие от более крупных, были ориентированы на конкретные бизнес-задачи, что обеспечивало требуемую быстроту реагирования на запросы. С развитием хранилищ данных в них стали применяться такие инновационные решения, как интегрированные системы и устройства данных.

Хранилища данных и витрины данных решили ряд проблем для компаний, нуждавшихся в системном подходе к управлению массивными транзакционными данными. Однако для работы с огромными объемами неструктурированной или слабоструктурированной информации возможностей хранилищ данных уже было недостаточно. Сама эта технология не могла развиваться в таком направлении, которого требовала меняющаяся обстановка. Положение осложнялось тем, что хранилища данных, как правило, пополняются в пакетном режиме — еженедельно или ежедневно. Этого достаточно для нужд планирования, финансовой отчетности и традиционных маркетинговых кампаний. Но для функционирования все более возрастающего числа бизнес- и потребительских сред реального времени такой темп является крайне медленным.

Как же удалось преобразовать традиционные подходы к управлению данными, чтобы обрабатывать нарастающий объем элементов неструктурированных данных? Решение было найдено не сразу. Поскольку компаниям требовалось хранить неструктурированные данные, поставщики начали использовать в технологических решениях такие новые возможности, как, например, массивы двоичных данных BLOB (большие двоичные объекты). Это позволяло хранить элемент неструктурированных данных в реляционной БД как один непрерывный фрагмент данных. Такой фрагмент может быть найден по запросу, если снабжается меткой набора данных или ссылкой, однако увидеть, что находится внутри такого объекта, невозможно. Очевидно, что это было лишь временное решение, которое не могло удовлетворить постоянно меняющиеся требования потребителей и бизнеса.

Тогда появились объектно-ориентированные системы управления базами данных (ODBMS). В объектно-ориентированной базе данных большие двоичные объекты хранятся как адресуемое множество фрагментов, таким образом, можно увидеть, что находится внутри объекта. В отличие от внедрения больших двоичных объектов, которое по сути добавляло независимый элемент к традиционной реляционной базе данных, появление объектно-ориентированных баз данных обеспечило единый подход к работе с неструктурированными данными. Объектно-ориентированные базы данных включают в себя язык программирования и структуру элементов данных, благодаря чему стало значительно легче манипулировать различными объектами данных без необходимости программирования и сложных соединений с базой. Объектно-ориентированные базы данных стали инновацией, которая стимулировала переход процесса управления данными на следующий этап.

ЭТАП 2. Управление сетью и контентом

Не секрет, что большинство существующих сегодня в мире данных — неструктурированные. Тем не менее компании вкладывали средства в развитие систем, работающих со структурированными данными. Это, как правило, были специализированные транзакционные системы, тесно связанные с получением доходов. В 1980-е годы функционал систем управления информационными ресурсами предприятий расширился, чтобы более эффективно управлять неструктурированными данными — в основном это были текстовые документы. В 1990-е годы, с развитием Интернета, компании уже не могли ограничиваться

работой с текстами — возникла необходимость хранить веб-контент, изображения, аудио- и видеозаписи, а также управлять ими.

Разрозненные решения, с которых началось развитие этого рынка, сменила унифицированная модель — появились платформы, объединившие в себе все необходимые элементы. Такие системы обеспечивали управление бизнес-процессами, контроль за версиями, распознавание информации, управление текстом и организацию совместной работы. В новом поколении таких систем стали использоваться метаданные (как правило, они включали сведения об организации и характеристиках хранимой информации). До сих пор подобные решения широко востребованы организациями, которым необходимы механизмы управления такими типами данных. В то же время требования к управлению данными выходят на качественно новый уровень, что стимулирует новый виток эволюции технологий управления данными. Возникновение таких новых требований обусловлено слиянием таких факторов, как появление интернет-технологий, виртуализации и облачных вычислений. Стало понятно, что организациям предстоит управлять новым поколением источников данных и иметь дело с ранее невиданными объемами и разнообразием данных, которые необходимо обрабатывать с огромной скоростью.

ЭТАП 3. Управление большими данными

Являются ли большие данные принципиально новой технологией или же результатом эволюции в области управления данными? Ответ на оба вопроса утвердительный. Как и достижения прошлых этапов развития управления данными, большие данные основаны на опыте лучших разработок последних пяти десятилетий. Однако появился совершенно новый фактор, который привел к кардинальным изменениям в технологиях управления данными — стоимость вычислительных циклов и хранения данных впервые существенно снизилась. Почему это так важно? Еще несколько лет назад компании были вынуждены ограничиваться лишь срезами или выборками важной информации, поскольку стоимость хранения и обработки данных не позволяла сохранять все доступные данные для последующего анализа.

Во многих ситуациях такой компромисс был оправдан. К примеру, производственная компания контролировала состояние систем, собирая технические данные каждые две минуты. Но это не исключало возможность того, что в срез данных не попадает информация о но-

вом типе дефектов, и таким образом проблема могла оставаться незамеченной в течение нескольких месяцев.

Сегодня технологии больших данных сделали хранение данных эффективным благодаря виртуализации и экономным благодаря облачным хранилищам. Кроме того, повышение скорости и надежности сетей позволило устранить физические ограничения, препятствовавшие управлению огромными объемами данных в приемлемом темпе. Усовершенствование компьютерной памяти и снижение ее стоимости также открыло новые возможности. В результате всех этих технологических изменений компаниям стали доступны такие способы использования данных, которые еще пять лет назад были немыслимы.

Но технологические изменения не происходят сами по себе. Их вызывают насущные потребности бизнеса, которые могут быть удовлетворены лишь за счет наличия и развития соответствующих технологий. Многие технологии, лежащие в основе больших данных, используются уже несколько десятков лет. Это виртуализация, параллельная обработка, распределенные файловые системы и базы данных в оперативной памяти. Углубленная аналитика также появилась несколько десятилетий назад, но не находила широкого применения в бизнесе. С другой стороны, такие технологии, как Hadoop и MapReduce, появились сравнительно недавно. Такое сочетание испытанных временем и новых технологий позволяет сегодня решать ряд бизнес-задач, обеспечивая компаниям возможность получать ценную информацию и практические результаты путем обработки данных различного типа с желаемой скоростью независимо от их объема.

В перспективе станет возможно обрабатывать петабайты данных (что эквивалентно 20 млн. картотечных шкафов с бумажными документами или 13,3 года видеоматериалов в высоком разрешении) и уверенно выявлять в них закономерности и отклонения. Благодаря большим данным использование информации в бизнесе выйдет на качественно новый уровень. Но развитие этих технологий связано не только с их коммерческим применением: наука и деятельность государственных органов также способствуют расширению возможностей больших данных. Расшифровка генома человека и обработка данных астрономических наблюдений, сбор и анализ информации для нужд антитеррористической деятельности — вот яркие примеры того, что использование больших данных не ограничивается только сферой бизнеса.

Подход к обработке данных зависит прежде всего от типа данных. Все данные можно разделить на две большие категории: передаваемые данные (также встречается термин «данные в движении») и хранимые данные («данные в покое»). Передаваемые данные используются, например, для анализа качества продукции непосредственно в ходе производственного процесса, что позволяет устранить проблемы на ранней стадии. А бизнес-аналитик, исследуя модели покупательского поведения, работает с хранимыми данными, которые описывают все аспекты взаимоотношений с клиентами, включая сведения о продажах, данные из социальных сетей и центров обслуживания.

Не стоит забывать, что мы только в начале пути — но именно большие данные в будущем позволят комплексно оценивать деятельность компании и прогнозировать изменения в ожиданиях клиентов. Пока что технологии, необходимые для удовлетворения потребностей бизнеса, изолированы друг от друга, и для достижения желаемого результата необходимо одновременно использовать достижения всех трех этапов развития управления данными. Большие данные не сводятся к какому-либо одному инструменту или технологии — это комбинация лучших наработок прошлого и актуальных тенденций, которая позволяет в нужное время получать нужную информацию из нужного объема данных независимо от того, генерируются такие данные людьми, устройствами или сетью.

Что такое большие данные

Большие данные — это не отдельная технология, а скорее сочетание опробованных и вновь внедряемых технологий, которое позволяет компаниям извлекать из имеющихся данных информацию для использования в их деятельности. Большие данные можно определить как способность управлять большими объемами разнородных данных со скоростью, достаточной для анализа таких данных в реальном времени и своевременного реагирования. Как было отмечено выше, большие данные обладают тремя основными признаками:

- объем: количество данных;
- скорость: быстрота обработки данных;
- вариативность: количество различных типов данных.



Вышеуказанные признаки не всегда выражены в больших данных в равной степени. Например, методы работы с большими данными справедливо применять для управления сравнительно небольшим объемом разнообразных и сложных данных или при обработке очень простых данных, но в огромных количествах. Такие простые данные могут быть как полностью структурированными, так и неструктурированными. Кроме того, существует четвертая определяющая характеристика: достоверность. Насколько точно рассматриваемые данные позволяют прогнозировать бизнес-ситуации? Соотносятся ли результаты анализа больших данных с заключениями на основе других методов?

Очень важно верно оценить свои задачи. Всегда должна быть возможность проверить данные на точность и соответствие контексту. Например, инновационное предприятие будет заинтересовано в том, чтобы анализировать огромное количество данных в режиме реального времени; это позволит моментально оценить значимость клиента и предоставить ему дополнительные предложения. Необходимо определить, данные каких типов и в каком объеме анализировать, чтобы повлиять на эффективность работы предприятия. Большие данные включают в себя любые структурированные и неструктурированные данные, в том числе из электронной почты, социальных сетей, текстовых потоков и многое другое. Таким образом, управление большими данными требует эффективного использования как структурированных, так и неструктурированных данных.

Разработка успешных архитектур управления большими данными

Прошло время, когда можно было внедрить базу данных для нужд отдельного проекта и этим ограничиться. Поскольку данные стали движущей силой роста и инноваций, наличие базовой архитектуры приобретает все большее значение для удовлетворения растущих потребностей бизнеса в эффективной работе с информацией.

Для начала: собрать, систематизировать, обобщить, проанализировать и действовать

Прежде чем детально рассматривать архитектуру, важно понять функциональные требования к работе с большими данными. Из рис. 1.1 видно, что первым этапом цикла является сбор данных, затем следует их систематизация и обобщение. После этого данные могут быть про-



Рис. 1.1. Цикл управления большими данными

анализированы с учетом конкретной задачи. Наконец, на основе результатов анализа предпринимается действие: например, Amazon.com рекомендует клиенту книгу, руководствуясь его прошлыми покупками, или магазин выдает покупателю определенного продукта купон со скидкой на сопутствующие товары.

На первый взгляд все просто, однако некоторые особенности указанных функций могут вызывать затруднения. Особенно важен вопрос проверки данных. При объединении данных из разных источников нужно убедиться, что они друг с другом соотносятся. Кроме того, некоторые источники данных могут содержать конфиденциальную информацию, для которой необходимо обеспечить соответствующий уровень защиты и управления. Более подробно об управлении данными рассказывается в главе 7.



Безусловно, любое использование больших данных должно начинаться с определения решаемой задачи. Это позволит понять, какие именно данные необходимы и какая архитектура подойдет для работы с ними.

Архитектура: закладываем фундамент

Архитектура должна не только удовлетворять функциональным требованиям, но и обеспечивать достаточную производительность. Необходимые ресурсы — вычислительные мощности, скорость — определяются типом проводимого анализа информации. Даже если большинство данных будет анализироваться в режиме реального времени, некоторый объем данных все равно потребуется сохранять.

Архитектура также должна иметь запас избыточной мощности, чтобы избежать непредвиденных задержек и простоев.

Требования к производительности зависят от конкретной организации и ее потребностей. Подумайте над следующими вопросами.

- Каким объемом данных управляет ваша организация в настоящее время и планирует управлять в перспективе?
- Как часто потребуется управлять данными в режиме реального времени или в близком к реальному времени режиме?
- Какова приемлемая степень риска? Действуют ли в отрасли особые требования к безопасности, нормативные и регуляторные ограничения?
- Насколько важна скорость управления данными?
- Какова приемлемая степень точности или достоверности собираемых данных?

Чтобы работать с большими данными, нужно представлять себе компоненты требуемой архитектуры. Архитектура управления большими данными должна включать разнообразный спектр сервисов, позволяющих быстрым и эффективным способом работать со множе-



Рис. 1.2. Архитектура больших данных

ством источников данных. Приводимая схема (рис. 1.2) наглядно отображает компоненты архитектуры больших данных и их взаимосвязи, которые будут рассмотрены ниже.

Интерфейсы и потоки данных

Прежде чем рассматривать самое важное и интересное — технологический стек больших данных, — ненадолго остановимся на интерфейсах и потоках для обмена данными. На рис. 1.2 они показаны по бокам от основной схемы. Чтобы понять, как большие данные функционируют на практике, необходимо уяснить, что обмен данными имеет первостепенное значение. Большие данные потому и становятся большими, что основаны на сборе большого количества данных из множества источников. Таким образом, открытые интерфейсы прикладных программ (API) критически важны для любой архитектуры больших данных. Такие интерфейсы действуют и на каждом уровне стека, и между ними. Большие данные не могут функционировать без сервисов интеграции.

Избыточная физическая инфраструктура

Обеспечение бесперебойной работы физической инфраструктуры имеет фундаментальное значение для реализации и масштабируемости архитектуры больших данных. Фактически без надежной физической инфраструктуры большие данные вряд ли стали бы настолько важной тенденцией. В отличие от решений, ориентированных на традиционные данные, физическая инфраструктура для работы с большими данными должна справляться с непредвиденными и непрогнозируемыми объемами данных. В основе такой физической инфраструктуры лежит модель распределенных вычислений. Это означает, что данные могут физически храниться в разных местах и связываются друг с другом посредством сетей, с помощью распределенной файловой системы и различных инструментов для анализа больших данных и приложений.

Избыточность инфраструктуры важна, поскольку мы имеем дело с огромным количеством данных из разных источников. Избыточность может воплощаться по-разному. Располагая частным облаком, компания может предусмотреть избытки в рамках этой частной облачной среды, чтобы горизонтально масштабировать ее в зависимости от рабочей нагрузки. Если компания не стремится наращивать собственную ИТ-инфраструктуру, то в дополнение к внутренним

ресурсам можно использовать внешние сервисы облачных вычислений. В некоторых случаях избыточность обеспечивается технологией программного обеспечения как услуги (SaaS), когда сложный анализ данных осуществляется как сторонняя услуга. Подход SaaS позволяет снизить затраты, быстро преодолеть начальный этап становления бизнеса и непрерывно развивать базовые технологии.

Инфраструктура безопасности

Чем важнее анализ больших данных для компании, тем важнее защита таких данных. К примеру, медицинская организация может быть заинтересована в применении больших данных для исследования демографических процессов или изменений в потребностях пациентов. Защита такой информации обусловлена законодательно и должна обеспечивать безопасность персональных данных клиентов. Поэтому должен быть четко определен круг лиц, имеющих доступ к этой информации, и условия такого доступа. Защита данных в таком случае должна обеспечивать идентификацию пользователей данных и при этом не допускать их применения для идентификации пациентов. Подобные требования безопасности должны учитываться в структуре больших данных изначально, а не внедряться впоследствии.

Источники операционных данных

Использование больших данных предполагает привлечение всех источников данных, которые дают представление о работе компании с разных сторон, и оценку влияния таких данных на функционирование бизнеса. Традиционно источник операционных данных предоставлял высокоструктурированные данные, относящиеся к конкретному направлению деятельности и управляемые с помощью реляционной базы данных. Однако в настоящее время операционные данные должны охватывать множество источников данных, в том числе разнообразные неструктурированные источники, такие как данные о клиентах и данные из социальных сетей.

Большие данные предполагают новые перспективные подходы к управлению данными, включая текстовые, табличные, столбцовые и геопространственные архитектуры баз данных, которые в совокупности называют NoSQL-базами («не только SQL») — распределенными нереляционными базами данных с открытым исходным кодом. По сути, необходимо сопоставить архитектуры данных с видами транзакций, чтобы обеспечить доступность ненужных данных в нужное

время. Кроме того, необходимы такие архитектуры данных, которые поддерживают сложный неструктурированный контент. Для работы с большими данными применяются как реляционные, так и нереляционные базы данных. Чтобы получить всестороннее представление о компании, потребуется использовать и неструктурированные источники данных, такие как системы управления контентом.

Все источники операционных данных имеют несколько общих характеристик.

- Они представляют собой системы учета, отслеживающие критические данные, необходимые для ежедневного функционирования бизнеса в режиме реального времени.
- Они постоянно обновляются, причем обновления происходят по результатам операций, проводящихся в рамках структурных подразделений компании и через Интернет.
- Чтобы давать точное представление о бизнесе, они должны объединять структурированные и неструктурированные данные.
- Они должны быть масштабируемыми, чтобы осуществлять поддержку тысяч пользователей на постоянной основе. Сюда же могут входить системы обработки совершенных через Интернет операций, системы управления взаимоотношениями с клиентами или приложения для центра обработки вызовов.

Проблемы производительности

Архитектура данных должна взаимодействовать со вспомогательной инфраструктурой организации. К примеру, нужно построить модель и определить, безопасно ли бурить нефтяную скважину в акватории. Для этого необходимы поступающие в реальном времени данные о температуре, минерализации, повторном образовании взвеси отложений. Потребуется множество других данных о биологических, химических и физических свойствах столба воды. На традиционном сервере запуск и выполнение такой модели может занять несколько дней. Однако теперь модель распределенных вычислений позволяет осуществить это за несколько минут.

Производительность также влияет на выбор типа базы данных. Если требуется выявить взаимосвязь между совершенно разными элементами данных (к примеру, обсуждением популярной темы в социальных сетях и ростом продаж), типичный запрос, который можно

применить к структурированным реляционным базам данных, не поможет. В таком случае лучше выбрать графовую базу данных, поскольку в ней «узлы» (то есть сущности) отделены от «свойств» (определяющей сущность информации) и от «границ» (взаимоотношений между узлами и свойствами). Использование правильной базы данных также способствует увеличению производительности. Чаще всего графовые базы данных применяют в научных и технических приложениях.

Еще один важный тип баз данных — это столбцовые базы данных, которые эффективны при представлении хранимой информации в столбцах, а не в строках. Поскольку ввод и вывод информации осуществляется чрезвычайно быстро, это обеспечивает более высокую производительность. Если необходимо использовать геопространственные данные, следует обратить внимание на пространственные базы данных. Они оптимизированы для хранения и осуществления запросов к данным на основе взаимосвязи объектов в пространстве.

Службы и инструменты организации данных

Не все используемые данные относятся к операционным. Все больше данных поступает из разнообразных неупорядоченных или сложных источников, таких как объемные источники публичных и частных данных, данные межмашинного взаимодействия и данные с датчиков. Еще несколько лет назад большинство компаний не имели возможности осуществлять сбор или хранение такого огромного объема данных — это было слишком дорого или обременительно. Даже в том случае, когда удавалось собирать такие данные, компании зачастую не располагали инструментами для последующей работы с ними. На тот момент лишь немногие программные решения были способны обрабатывать настолько большие объемы информации, при этом их использование представляло трудность и требовало значительных временных затрат. В конечном счете компании, которые все же отваживались анализировать подобные данные, были вынуждены работать лишь со срезами данных. Такой подход был чреват тем, что важные события, не попадавшие в срез данных, оставались незамеченными.

MapReduce, Hadoop и Big Table

Благодаря развитию вычислительных технологий стало возможным управление огромными объемами данных. Ранее такие данные могли обрабатываться только с помощью суперкомпьютеров, что требовало

больших денежных затрат. Но постепенно цены на программно-аппаратные системы снизились, и в результате новые методы распределенных вычислений получили массовое распространение. Настоящий прорыв в области больших данных произошел в тот момент, когда такие компании, как Yahoo!, Google и Facebook, стали искать возможности для извлечения выгоды из огромных объемов данных, создающихся при использовании их продуктов.

Им требовались новые технологии, которые обеспечивали бы доступ к большим объемам данных, их хранение и анализ в режиме, приближенном к реальному времени. Таким образом обладание огромным количеством данных, генерируемых при использовании их сетей, могло принести этим компаниям реальный доход. Созданные для этих целей решения коренным образом изменили рынок управления данными. В частности, такие инновации, как MapReduce, Hadoop и Big Table, послужили толчком к появлению нового поколения систем управления данными. Эти технологии позволяют решить одну из основных задач — они обеспечивают экономически эффективную и своевременную обработку огромных массивов данных.

MapReduce

MapReduce — это разработанная компанией Google технология эффективного выполнения наборов функций, обрабатывающих большие объемы данных в пакетном режиме. Компонент «map» (карта) распределяет задачи между большим количеством систем, уравновешивая их нагрузку, а также управляет восстановлением после сбоев. После завершения распределенных вычислений функция «reduce» (редукция) собирает все элементы воедино и формирует результат. Пример использования MapReduce — определение количества страниц на каждом из языков в книге с текстами на 50 языках.

Big Table

Big Table — это система распределенного хранения данных, разработанная компанией Google и предназначенная для управления масштабируемыми структурированными данными. Данные организованы в форме таблиц со строками и столбцами. В отличие от традиционных реляционных баз данных, Big Table представляет собой разреженную распределенную постоянную многомерную сортированную карту, предназначенную для хранения больших объемов данных на стандартных серверах.

Hadoop

Hadoop — это программная платформа Apache на базе MapReduce и Big Table. Hadoop обеспечивает работу приложений на основе MapReduce на больших кластерах со стандартным оборудованием. Платформа лежит в основе вычислительной архитектуры, осуществляющей поддержку коммерческой деятельности компании Yahoo!. Платформа Hadoop организует параллельную обработку данных на множестве вычислительных узлов. Это позволяет ускорить вычисления и сократить задержки. Hadoop включает в себя два основных компонента: высокомасштабируемую распределенную файловую систему, способную хранить петабайты данных, и высокомасштабируемый механизм MapReduce, осуществляющий вычисления в пакетном режиме.

Традиционная и углубленная аналитика

Каким же образом компания может эффективно использовать имеющиеся разнообразные данные? Анализировать их можно по-разному, в зависимости от решаемой задачи. Одни типы анализа используют традиционные хранилища данных, другие требуют применения углубленной прогнозной аналитики. Для успешного планирования бизнеса необходимо использовать различные подходы к управлению большими данными.

Аналитические хранилища данных и витрины данных

После классификации огромного количества имеющихся данных целесообразно выделить те, что позволяют выявлять закономерности, и преобразовать эти данные в доступную для предприятия форму. Обыкновенные хранилища и витрины данных обеспечивают сжатие, многоуровневое разбиение и архитектуру параллельных вычислений.

Аналитика больших данных

Возможность управлять петабайтами данных и анализировать их позволяет в полной мере использовать массивы информации, что может оказать существенное влияние на работу компании. Для этого требуются аналитические инструменты, позволяющие управлять такими распределенными данными и получать результаты, пригодные для решения бизнес-задач. Вместе с тем аналитика больших данных может быть достаточно сложной. К примеру, некоторые компании

используют прогнозные модели, объединяющие структурированные и неструктурированные данные для предотвращения мошенничества. Аналитика больших данных включает в себя анализ данных социальных сетей, текстовую аналитику и другие виды анализа. Более подробно об аналитике больших данных рассказывается в главах 12, 13 и 14.

Отчетность и визуализация

Имеющиеся в организациях данные всегда использовались для составления различных отчетов. Это позволяло получить представление о состоянии дел: от объемов ежемесячных продаж до прогнозов роста. Большие данные изменяют способ использования таких данных. Если компания имеет возможность собирать и анализировать достаточное количество данных и управлять ими, то можно применить инструменты анализа нового поколения, которые представят собранные данные в контексте решаемых бизнес-задач. Составление отчетов и визуализация больших данных позволяют выявить взаимосвязь между данными, а также влияние такой взаимосвязи на будущее.

Приложения, задействующие большие данные

Исторически в бизнес-среде данные применяли для того, чтобы определить, какие действия нужно предпринять и в какое время. Часто данные представляли в виде полей в стандартных бизнес-приложениях. Однако с началом эпохи больших данных ситуация изменилась — теперь появляются специализированные приложения, функционал которых основан на использовании особенностей и преимуществ, характерных именно для больших данных.

Эти решения применяются в таких областях, как здравоохранение, управление производством, управление транспортными потоками и т.д. Что общего у приложений разного назначения, работающих с большими данными? Они меняют конъюнктуру на рынках именно благодаря огромным объемам, высокой скорости и разнообразию обрабатываемых данных. Например, в медицинской организации приложение, использующее большие данные, может применяться для контроля состояния недоношенных детей, чтобы точно и своевременно определить необходимость вмешательства. На производстве такие приложения могут предотвратить отказ оборудования в ходе производственного процесса. В сфере управления транспортными потоками приложение, использующее большие данные, способно

уменьшить количество пробок на городских магистралях, что в свою очередь снизит вероятность ДТП, расход топлива и уровень загрязнения окружающей среды.

На пути к большим данным

Бизнесу всегда приходилось иметь дело со значительным количеством данных в различных формах. Большие данные меняют возможности работы с такой информацией. Наличие соответствующих технологий позволяет использовать большие данные для прогнозирования и решения бизнес-задач, а также быстро реагировать на появление новых возможностей. Анализ больших данных находит применение в любой сфере: от управления городским хозяйством, безопасности на производстве и научных экспериментов до регулирования транспортных потоков, повышения удовлетворенности клиентов и улучшения качества продукции. Новые технологии и инструменты, описанные в данной книге, помогут вам понять и раскрыть огромный потенциал больших данных, который способен изменить привычный мир.

Глава 2

Типы больших данных

В ЭТОЙ ГЛАВЕ:

Понятие структурированных и неструктурированных данных
Требования, предъявляемые к типам данных в режиме реального времени и операционным данным
Интегрирование типов данных в среду больших данных

Как говорится, вся прелесть жизни в разнообразии. Именно разнообразие является одним из ключевых признаков больших данных. В главе 1 уже упоминалась важность управления данными различных типов. А большие данные предполагают самую разнообразную информацию — от сведений о финансовых транзакциях до твитов, фотографий и аудиозаписей. Таким образом, прежде чем воспользоваться преимуществами больших данных, сначала необходимо обобщить всю имеющуюся информацию, чтобы ее можно было комплексно анализировать и управлять ею. Однако реализовать это значительно сложнее, чем кажется. В этой главе рассматриваются две основные группы в составе больших данных: структурированные и неструктурированные данные.

Эра больших данных привнесла в сложившуюся систему управления данными два новых фактора:

- принципиально новые источники данных — смартфоны, планшеты, датчики;
- эффективные и экономически выгодные технологии сбора, хранения и анализа информации, способные работать со сколь угодно большими объемами данных.

Существует множество способов применения больших данных для решения конкретных задач. Например, для мониторинга информации о транспортных потоках необходимо работать с данными, поступающими в реальном времени. Напротив, если требуется собрать и обра-

ботать большое количество данных для выявления неизвестных ранее закономерностей, анализ проводится в пакетном режиме, и управлять данными в реальном времени нет необходимости. Кроме того, иногда программное решение в области больших данных требует интегрировать несколько источников данных (возможные предпосылки такой интеграции рассматриваются ниже). Иными словами, выбор конкретного технологического решения, которое необходимо внедрить, определяется целями использования структурированных и неструктурированных данных.

Что такое структурированные данные

Под структурированными данными понимаются данные, которые имеют определенную длину и формат. Примерами таких данных являются числа, даты, а также комбинации слов и чисел (например, имя, адрес клиента и т.д.). В контексте структурированных данных все они называются «строками». По мнению большинства экспертов, структурированные данные составляют около 20% от общего объема данных. Структурированные данные — это, пожалуй, наиболее привычный формат данных. Обычно они хранятся в базе данных, к которой можно делать запросы с помощью специального языка программирования, к примеру SQL (язык структурированных запросов). Подробнее о языке SQL рассказано ниже, в разделе о неструктурированных данных.

Возможно, ваша компания уже ведет сбор структурированных данных из «традиционных» источников. Это могут быть данные системы управления взаимоотношениями с клиентами (CRM), данные об оперативном планировании ресурсов предприятия (ERP) и финансовые данные. Для удобства анализа элементы таких данных, как правило, объединяют в хранилище данных.

Источники структурированных больших данных

Хотя структурированные данные, казалось бы, всем хорошо знакомы, в мире больших данных они приобретают совершенно новое значение. Эволюция технологий порождает новые источники структурированных данных, которые зачастую поступают в реальном времени и в больших объемах. Источники данных делятся на две категории:

- данные, генерируемые машинами: это данные, создаваемые компьютерами и другими устройствами без вмешательства человека;
- данные, генерируемые человеком: это данные, которые создаются человеком при взаимодействии с компьютером.

Некоторые эксперты выделяют также третью категорию — гибридные данные, однако мы ее рассматривать не будем.

Генерируемые машинами структурированные данные могут включать:

- данные с датчиков: данные, создаваемые метками радиочастотной идентификации (RFID-метками), интеллектуальными счетчиками, медицинскими приборами, системой глобального позиционирования (GPS) и т.п. В частности, быстро завоевывает популярность технология радиочастотной идентификации (RFID), использующая крошечные компьютерные чипы, благодаря которым, к примеру, можно отслеживать перемещение контейнеров с продукцией. Передаваемая метками информация поступает на сервер, где ее затем можно проанализировать. Эта технология важна для управления поставками и учета товаров. Еще один пример такого типа данных — информация с GPS-датчиков в смартфонах, открывающая новые возможности для исследования поведения потребителей;
- данные лог-файлов: все виды данных о функционировании серверов, приложений, сетей и т.д., собираемые в процессе их работы. Объем таких данных может быть огромным. Они находят применение, например, в соглашениях об уровне предоставляемых телекоммуникационных услуг или в оценках информационной безопасности;
- данные, поступающие из точек продаж: все данные, связанные с продуктом, которые генерируются при считывании штрихкода на кассе. Достаточно подумать, какое количество товаров приобретается в мире ежедневно, чтобы представить, насколько велик объем таких данных;
- финансовые данные: данные, поступающие из финансовых систем. Большинство таких систем в настоящее время алгоритмизировано, то есть они функционируют на основе предопределенных правил, автоматизирующих процессы. Примером могут служить данные, относящиеся к торговле на бирже. Обозначение компании и ее стоимость в долларовом выражении, безусловно, относятся к структурированным данным. Нужно отметить, что не все финансовые данные являются машинно-генерируемыми.

Примеры генерируемых человеком структурированных данных включают:

- входные данные: это любые данные, вводимые человеком в компьютер на различных сайтах, в социологических опросах и т.д., к примеру, имя, возраст, уровень дохода. Такие данные могут быть весьма полезными при изучении потребительского поведения;
- данные о посещениях веб-сайтов: данные генерируются всякий раз, когда пользователь проходит по ссылке на какой-либо сайт. Анализ этих данных помогает изучать потребительское поведение и привычки;
- игровые данные: любое действие в игре может фиксироваться. Это позволяет исследовать поведение конечных пользователей в рамках игры.

Это лишь некоторые примеры. Такие данные необязательно сами по себе объемны — но если учесть, что их предоставляет каждый из миллионов пользователей, можно представить, какое громадное количество данных генерируется в результате. Кроме того, значительная часть таких данных содержит компонент реального времени, поэтому они могут быть полезны при определении закономерностей, имеющих влияние на потенциальную прибыль. Иными словами, структурированные данные могут служить очень мощным и многофункциональным инструментом.

Реляционные базы данных и большие данные

Термин «обеспечение сохраняемости данных» означает свойство базы данных сохранять свои прошлые состояния после изменений. Предшественником хранилищ сохраняемости стала система управления реляционными базами данных, или реляционная СУБД. На ранних этапах развития вычислительной техники использовались методы обеспечения сохраняемости данных, которые сейчас представляются примитивными.

Можно вспомнить «плоские файлы» или «сетевые» хранилища данных, которые были распространены до 1980-х годов. Хотя подобные механизмы выполняли требуемую задачу, ориентироваться в них было сложно, так что системным программистам приходилось писать специальные программы для работы с данными.

Реляционная модель была разработана в 1970-х годах доктором Эдгаром Коддом, сотрудником IBM, и применялась в продуктах таких

компаний, как IBM, Oracle, Microsoft и т.д. Она по-прежнему широко распространена и играет немалую роль в эволюции больших данных. Мы подробно останавливаемся на особенностях реляционных баз данных, поскольку для работы с большими данными применяются различные типы БД и необходимо понимать принципиальные различия между ними.

В реляционной модели данные хранятся в таблице. Такая база данных содержит схему, то есть структурное представление того, что именно хранится в базе данных. Например, схема реляционной базы данных определяет таблицы, поля в таблицах и связи между ними. Данные хранятся в столбцах (по одному столбцу для каждого атрибута) и в строках. На рис. 2.1 показана схема простой базы данных, состоящая из двух таблиц. В первой таблице хранится информация о продукте, во второй — демографические данные. Каждая из них содержит различные атрибуты (номер клиента, номер заказа, код продукта и т.д.). В каждую таблицу можно вносить дополнительные данные, читать их, обновлять или удалять. Как правило, в реляционной модели эти функции осуществляются с использованием языка структурированных запросов (SQL).

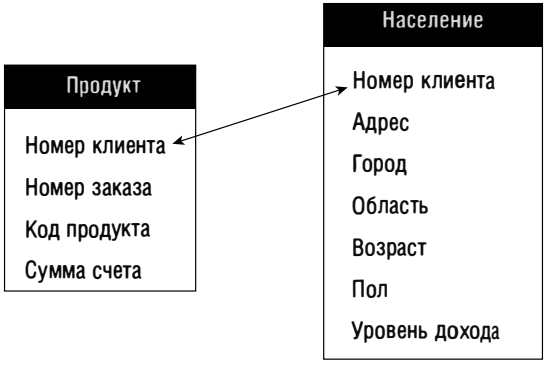


Рис. 2.1. Связь между таблицами

Важный аспект реляционной модели с использованием SQL — возможность формулирования запроса к таблицам с помощью общего ключа, который связывает их друг с другом. На рис. 2.1 общим ключом двух таблиц является атрибут «Номер клиента».

К примеру, с помощью запроса можно определить, потребители какого пола чаще приобретали тот или иной товар. Такой запрос может выглядеть следующим образом:

```
Select CustomerID, State, Gender, Product from
          "demographic table", "product table" where
          Product= XXYY
```

Последние несколько десятилетий реляционные базы данных доминировали на рынке, однако в них трудно работать с огромными потоками различных типов данных. Но поставщики реляционных БД стараются идти в ногу со временем — сегодня появляются реляционные базы данных, ориентированные на работу с большими данными. Кроме того, для управления большими данными были разработаны и новые модели БД (о NoSQL и потоковых базах данных упоминалось в главе 1; более подробно об этих и других технологиях рассказывается в части III).



PostgresSQL (www.postgressql.org), более подробная информация о которой изложена в главе 7, является распространенной свободной объектно-реляционной системой управления базами данных. Благодаря расширяемости и совместимости с различными платформами PostgresSQL стала основой для многих реляционных баз больших данных.

Что такое неструктурированные данные

Неструктурированные данные — это данные, не имеющие определенного формата. Если в среднем 20% данных любого предприятия являются структурированными, то остальные 80% — неструктурированными. Таким образом, большая часть информации, с которой мы сталкиваемся, относится именно к неструктурированным данным. До недавнего времени не существовало технологий, которые справлялись бы с их обработкой, и неструктурированные данные можно было только сохранять и анализировать вручную.

Изучение источников неструктурированных данных

Неструктурированные данные окружают нас повсюду. В сущности, многие виды деятельности строятся на взаимодействии с неструктурированными данными. Как и структурированные данные, они могут генерироваться либо машиной, либо человеком.

Примеры машинно-генерируемых неструктурированных данных:

- спутниковые снимки: метеорологические данные, результаты спутниковой фотосъемки (Google Earth — самый яркий пример);
- научные данные: графики сейсмической активности, результаты исследования атмосферы, данные из области физики элементарных частиц;
- фото- и видеодокументы: записи с камер наблюдения и видеорегистраторов;
- данные радаров и сонаров: данные дорожного, метеорологического, океанографического, сейсмологического наблюдения.

Некоторые примеры неструктурированных данных, генерируемых человеком:

- документация организаций: тексты служебных документов, журналов учета, результатов опросов, электронных писем и т.д. Генерируемая на предприятиях информация составляет немалую часть в глобальном объеме текстовых данных;
- данные социальных сетей: такие данные генерируются в рамках YouTube, Facebook, Twitter, LinkedIn и Flickr и других платформ;
- мобильные данные: к этой категории относятся, к примеру, данные текстовых сообщений и информация о местоположении;
- информационное наполнение веб-сайтов: такие данные поступают с любого сайта, поставляющего неструктурированный контент, например YouTube, Flickr или Instagram.



Некоторые считают термин «неструктурированные данные» неточным, поскольку любой документ имеет свою структуру или форматирование в зависимости от программного обеспечения, в котором он создан. Тем не менее само содержимое документа является неструктурированным. Неструктурированные данные, безусловно, составляют большую часть общего объема информации в мире, и возможности применения неструктурированных данных стремительно расширяются. Например, текстовая аналитика (подробнее о ней рассказано в главе 13) используется для извлечения из неструктурированных текстов релевантной информации и ее преобразования в структурированные данные, которые могут затем обрабатываться различными способами. В социальных сетях с помощью аналитики рассматриваются массовые обсуждения. В сочетании с анализом записей call-центров, электронных писем, комментариев в анкетах и опросах и других документов обработка информации из социальных сетей, поступающей из десятков миллионов источников, позволяет всесторонне оценивать потребительское поведение и степень удовлетворенности клиентов качеством обслуживания.

Слабоструктурированные данные

Слабоструктурированные данные обладают признаками как структурированных, так и неструктурированных данных. Они не соответствуют четкой схеме (то есть структуре), однако могут быть самоописываемыми и иметь простые пары «метка/значение» (к примеру, <семья>=Джонс, <мать>=Джейн и <дочь>=Сара). К слабоструктурированным относятся данные в форматах стандартов EDI, SWIFT и XML. Данные этого типа можно рассматривать как промежуточные средства обработки сложных событий.

Системы управления контентом (CMS) в управлении большими данными

Неструктурированные данные можно хранить в базах данных. Однако чаще для этого используют системы управления контентом (CMS), осуществляющие контроль над полным циклом существования контента, который может представлять собой контент веб-сайтов, содержимое документов и т.д.

Ассоциация по системам управления обработкой информации и изображений (AIIM; www.aiim.org) — это некоммерческая организация, которая предоставляет образовательные услуги, проводит исследования и изучает передовой опыт обработки информации. Ассоциация AIIM так определяет процесс управления информационными ресурсами предприятия (ECM): «стратегии, методы и инструменты, используемые для сбора, хранения, доставки контента и связанных с организационными процессами документов, а также управления ими». Технологии управления информационными ресурсами предприятия включают в себя управление документами, управление отчетностью, визуализацию, управление рабочими процессами, управление веб-контентом и организацию совместной работы.

Процесс управления контентом привел к появлению отдельной индустрии. Многие поставщики систем управления контентом модернизируют свои продукты, предлагая горизонтально масштабируемые решения для обработки больших объемов неструктурированных данных. При этом появляются и новые технологии для анализа неструктурированных данных. Некоторые из них поддерживают обработку как структурированных, так и неструктурированных данных. Доступные на рынке решения, такие как Hadoop, MapReduce и Streaming, позволяют обрабатывать потоки данных реального вре-

мени. Более подробно об этих технологиях рассказано в главах 8, 9 и 10 соответственно.

Системы хранения контента в рамках CMS уже не представляют собой отдельные функциональные решения, а зачастую входят в состав комплексных решений для управления данными. Например, мониторинг ленты Twitter при выполнении некоторого заданного условия может алгоритмически запускать CMS-поиск, и автор соответствующего твита получит ответ с информацией о том, где можно купить продукт, в котором он может быть заинтересован. Особенно эффективно такое взаимодействие, если оно осуществляется в реальном времени. Данный пример демонстрирует ценность одновременного использования неструктурированных, структурированных (профиль автора твита) и слабоструктурированных (контент в CMS) данных в реальном времени.



Чаще всего для решения задач в области больших данных используется гибридный подход. К примеру, не имеет смысла сохранять весь новостной контент организации в корпоративную платформу обработки данных Hadoop только потому, что она ориентирована на работу с неструктурированными данными.

Обзор требований, предъявляемых к обработке данных в реальном времени

Как уже отмечалось выше, во многих случаях использование больших данных открывает возможности, которые ранее были недоступны из-за отсутствия технологических возможностей или высокой стоимости. Одним из самых значительных изменений, связанных с большими данными, стало исчезновение необходимости в сложном программировании для обработки огромных объемов данных. На данный момент многие организации, активно работающие с информацией, испытывают затруднения в управлении большими объемами сложных данных. Внедрение систем больших данных поможет таким компаниям справиться с обработкой данных в условиях дальнейшего увеличения их объема, скорости и разнообразия. Не менее важную проблему всегда представляла необходимость быстрой обработки данных, и многим организациям приходилось ограничиваться анализом срезов данных, а не всего их объема. Однако в такие срезы не всегда по-

падала критически важная информация, которая могла содержаться в полном объеме данных. Развитие и внедрение технологий больших данных даст возможность анализировать данные с меньшими затратами и получать результаты, которые позволят принимать обоснованные решения и своевременно реагировать на изменения.

Использование больших данных в режиме реального времени способно решить ряд масштабных проблем. Как влияет на работу организации возможность управлять потоковыми данными в реальном времени? Прежде всего этот подход актуален, когда требуется оперативно принимать критически важные решения — например, проверить работу медицинского оборудования или прогнозировать риск несанкционированного вторжения в систему. Вот еще несколько примеров использования данных реального времени для быстрого достижения положительного результата:

- контроль нештатных ситуаций при обнаружении попыток мошенничества или несанкционированного сбора информации;
- мониторинг новостных лент и социальных сетей для определения событий, которые могут повлиять на финансовые рынки, как, например, реакция клиентов на анонс нового продукта;
- анализ ленты Twitter во время важного спортивного мероприятия и оперативное изменение приуроченной к нему рекламной кампании;
- предоставление клиенту купона на скидку с учетом купленного им ранее товара в точке продажи.

В ряде случаев потоковые данные поступают с очень высокой скоростью из небольшого количества источников. Возможны также ситуации, когда данные поступают из целого ряда различных источников. Иногда вариативность источников существенно разнится — от небольшой до весьма значительной. Если вы планируете обрабатывать информацию в реальном времени, подумайте над следующими вопросами. Можно ли решить задачу с помощью традиционных систем управления информацией или же для решения необходимы новые возможности? Не создадут ли большие объемы или высокая скорость данных (а нередко и то, и другое) чрезмерную нагрузку на системы?

Итак, вы пришли к выводу, что возможности для обработки данных в реальном времени вам действительно необходимы. Какая для них нужна инфраструктура? Более подробно эта тема освещается в главе 3,

а пока приведем ключевые характеристики системы для загрузки, обработки и анализа данных в режиме реального времени:

- малая задержка. Задержка — это время запаздывания выполнения задач в среде. Чем меньше значение задержки, тем быстрее осуществляется реакция системы. Обработка данных в реальном времени требует минимальной задержки, а это предъявляет серьезные требования как к вычислительным мощностям, так и к инфраструктуре сети;
- масштабируемость — способность поддерживать определенный уровень производительности при увеличении нагрузок;
- универсальность: система должна поддерживать как структурированные, так и неструктурированные потоки данных;
- поддержка форматов исходных данных. Преобразование данных требует временных и финансовых затрат, поэтому в режиме реального времени предпочтительно использовать данные в их исходном формате. Возможность обработки сложных процессов взаимодействия данных, инициирующих события, может иметь решающее значение.



Необходимость обработки постоянно увеличивающихся объемов различных данных является одним из ключевых факторов, стимулирующих внедрение облачных сервисов. Облачная модель является широкомасштабной и распределенной. Более подробно об облаке рассказано в главе 6.

Объединение больших данных

Выбор той или иной технологии зависит от того, с какой целью вы будете использовать структурированные и неструктурированные данные. Цель также определяет необходимость понимания структуры входящих данных для их эффективного хранения.

Управление разными типами данных

На рис. 2.2 представлены типы систем управления данными, используемые для различных категорий больших данных. Их характеристики подробно рассматриваются в следующих главах.

	ПАКЕТНАЯ ОБРАБОТКА	ПОТОКОВЫЕ ДАННЫЕ	СЛОЖНЫЙ ЗАПРОС
Структурированные	Hadoop	Ключ / значение	Реляционная СУБД
Неструктурированные	Документальная	Пространственно-графовая	Столбцовая
Оба типа данных	Гибридная	Гибридная	Гибридная

Таблица 2.1. Типы систем управления данными

Интегрирование типов данных в среду больших данных

Еще один важный аспект больших данных заключается в том, что зачастую нет необходимости владеть данными, чтобы их использовать. Например, можно эффективно анализировать данные социальных сетей, отраслевую статистику из сторонних источников и даже данные со спутников. Достаточно представить себе контент любой социальной сети. Очень часто возникает необходимость в интеграции различных источников. Данные могут поступать из всех внутренних систем, одновременно из внутренних и внешних источников или только из внешних источников, но в любом случае есть вероятность, что эти данные прежде хранились и обрабатывались изолированно друг от друга.

Вовсе не обязательно, чтобы данные поступали в режиме реального времени — большой объем накопленных данных в совершенно разных форматах тоже может управляться средствами больших данных. Разумеется, на практике нередко приходится работать и с данными, которые обладают всеми тремя признаками: поступают в огромном количестве, на высокой скорости и в различных форматах. В любом случае источники таких данных нельзя рассматривать как совокупность изолированных хранилищ информации — чтобы извлечь из них пользу для бизнеса, разные источники необходимо интегрировать.

Для этого предназначены соединители и метаданные.

Соединители

Соединители — это программные средства, которые позволяют получать большие данные из разных источников. Их можно использовать, например, в Twitter или Facebook, а можно с их помощью соединить

внутреннее хранилище организации с внешним источником больших данных, чтобы проанализировать эти данные совместно. Подробнее о соединителях рассказано в главе 15.

Метаданные

Метаданные — один из важнейших компонентов для объединения любых данных. Метаданные представляют собой определения, схемы расположения и другие характеристики, которые указывают путь к данным организации и программным компонентам, описывают условия доступа и использования этих данных. Пример метаданных — данные о номере банковского счета, которые могут содержать собственно номер, описание, указание на тип данных, имя, адрес и телефон владельца и уровень доступа к информации.

Использование метаданных помогает упорядочить хранилища данных и реагировать на появление и изменение источников данных. Принцип метаданных сам по себе не нов, однако в контексте больших данных этот компонент совершенствуется и развивается. Традиционно метаданные подразумевают наличие каталога, который обеспечивает единое представление обо всех источниках данных, но в сфере больших данных, когда для контроля доступны лишь некоторые источники данных, каталог должен быть устроен иначе, а для работы с метаданными может понадобиться дополнительный аналитический инструментарий.

Глава 3

Новое использование старых решений: распределенные вычисления

В ЭТОЙ ГЛАВЕ:

Развитие технологий распределенных вычислений
Составные элементов распределенных вычислений
Преимущества современных технологий программного и аппаратного обеспечения в распределенных вычислениях

Распределенные вычисления — это не новая технологическая концепция, они используются уже около 50 лет. Первоначально эта технология применялась для исследований в области вычислительной техники, чтобы масштабировать вычислительные задания и решать сложные задачи без затрат на большие вычислительные системы. Одним из наиболее успешных начинаний в области распределенных вычислений был проект, финансировавшийся Управлением перспективных исследований и разработок министерства обороны США (DARPA), — его результатом стала разработка первой распределенной вычислительной сети, которая получила название Интернет. Можно сказать, что появление Интернета стало началом революции, которая привела к изменениям во всем: от торговли, здравоохранения и транспорта до коммуникаций между людьми и межмашинного взаимодействия. Рассмотрим, что же собой представляют распределенные вычисления и почему данная технология является основополагающей для больших данных.

Обзор истории распределенных вычислений

Распределенные вычисления лежат в основе всех важнейших тенденций последних десятилетий, включая облачные вычисления, предоставление сервисов как оказание услуг, виртуализацию и большие данные. Другими словами, ни одно из этих достижений не было бы возможным без применения распределенных вычислений. Распределенные вычисления — это технология, позволяющая отдельным компьютерам, географически удаленным друг от друга, объединяться в сеть, как если бы они были единой средой (рис. 3.1). У распределенных вычислений широкая сфера применения: в некоторых топологиях отдельные вычислительные ресурсы просто осуществляют передачу друг другу сообщений, в других ситуациях все элементы распределенной вычислительной среды пользуются общими ресурсами, начиная с памяти и заканчивая сетями и хранилищами. Все модели распределенного вычисления имеют общее свойство: они представляют собой группу объединенных сетью компьютеров, которые работают совместно для выполнения определенного объема работы или процесса.

Благодаря DARPA

Наиболее известная модель распределенных вычислений — Интернет. Она является основой всех современных технологий, начиная от онлайн-коммерции и заканчивая облачными вычислениями, управлением услугами и виртуализацией. Разработка Интернета финансировалась Управлением перспективных исследований и разработок министерства обороны США (DARPA); этот научно-исследовательский проект был призван создать взаимосвязанную сетевую систему для нужд совместных некоммерческих научных исследований. В начале эры Интернета связь осуществлялась посредством телефонных линий (рис. 3.1); те, кто еще застал коммутируемый доступ в Интернет, помнят, насколько медленными и неустойчивыми были такие соединения.

В течение следующего десятилетия технология совершенствовалась; были разработаны общие протоколы, такие как TCP (протокол управления передачей данных), которые способствовали распространению технологии и сети. С появлением интернет-протокола (IP) проект превратился из закрытой сети для группы ученых в потенциально коммерческую платформу для передачи электронной по-

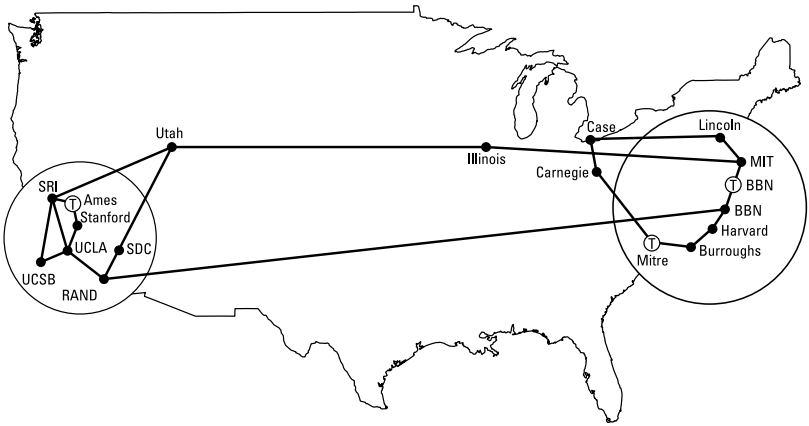


Рис. 3.1. Модель соединений на начальном этапе существования Интернета

чты по всему миру. В 1980-е годы на рынке начали появляться новые интернет-сервисы, которые стали коммерческими альтернативами сети DARPA. В 1992 году конгресс США принял закон о научных и передовых технологиях, который впервые разрешил коммерческое использование этой мощной сетевой технологии. С тех пор Интернет продолжает бурно развиваться, и сегодня это поистине глобальная распределенная сеть и ярчайший пример возможностей распределенных вычислений.

Достоинства согласованной модели

Как же инновация DARPA сказалась на развитии распределенных вычислений? До коммерциализации Интернета сотни компаний и организаций создавали инфраструктуры программного обеспечения, поддерживающие высокораспределенную вычислительную среду, однако при этом каждый поставщик или организация по стандартизации разрабатывали свои собственные вызовы удаленных процедур (RPC), которыми должны были пользоваться все клиенты, разработчики коммерческого программного обеспечения и партнеры данной организации. Вызов удаленной процедуры (RPC) — это примитивный механизм отправки задачи на удаленный компьютер; как правило, он не позволял переключиться к другим задачам, пока удаленная работа не завершена.

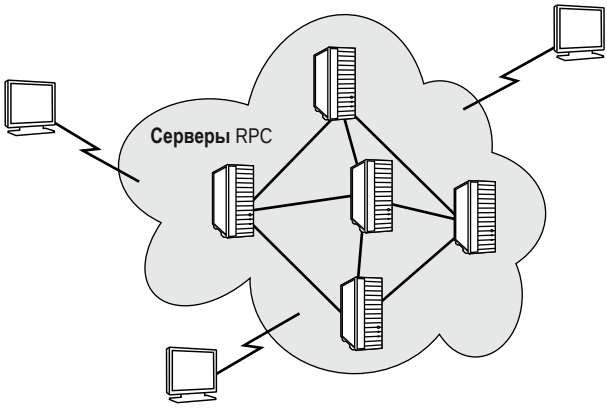


Рис. 3.2. Пример использования вызовов удаленных процедур (RPC) в первых распределенных вычислительных методах

Чем больше поставщиков внедряли свои проприетарные (закрытые) RPC, тем более очевидно становилось, что ни одной компании не удастся самостоятельно создать универсальный стандарт распределенных вычислений, который приняли бы все остальные. К середине 1990-х годов на смену примитивным процедурам RPC пришли интернет-протоколы, которые стали основой для используемых в настоящее время технологий распределенных вычислений. Это открыло путь к распространению сетевых вычислений. Сегодня мы считаем самим собой разумеющимся тот факт, что можно создать сеть слабосвязанных компьютеров, которые могут обмениваться информацией и общаться с нужной скоростью и в нужное время (рис. 3.2).

Основы распределенных вычислений

Единой распределенной вычислительной модели не существует, поскольку вычислительные ресурсы могут быть распределены различными способами. Например, можно физически разместить пакет программ на одном сервере и использовать для передачи между ними информации сервисы обмена сообщениями. А можно назначить общее задание ряду различных систем или серверов, так что каждый из них будет использовать для его выполнения собственную память. И в том, и в другом случае это будут распределенные вычисления.

Зачем нужны распределенные вычисления при обработке больших данных

Не все задачи требуют распределенных вычислений. При отсутствии существенных временных ограничений комплексную обработку можно осуществлять удаленно посредством специализированной службы. Раньше для проведения сложного анализа информации компании передавали данные на внешний объект или сервис, располагавший необходимыми мощностями. Конечно, на получение результата уходило немало времени, но в большинстве случаев было экономически нецелесообразно приобретать вычислительные мощности для самостоятельного проведения таких работ. Зачастую организации даже не стремились собрать весь имеющийся объем информации, поскольку это обходилось очень дорого, и ограничивались срезами данных. Аналитикам оставалось только надеяться, что в такие срезы попадала вся важная информация.

Ключевые достижения в области аппаратного и программного обеспечения коренным образом изменили управление данными. В первую очередь инновации и спрос увеличили мощность и снизили стоимость оборудования. Стало разрабатываться программное обеспечение, которое максимально использовало преимущества нового оборудования, например автоматически оптимизировало и балансировало нагрузку между огромными кластерами узлов. Программное обеспечение содержало встроенные правила, которые учитывали, что разным уровням рабочих нагрузок требуется определенная производительность. Программное обеспечение задействовало все узлы вычислительной сети как единый массив для вычислений и хранения данных. В случае сбоя на одном из узлов выполняемые процессы без прерывания работы перемещались на другой узел. Для этого применялись технологии виртуализации. Более подробно об использовании виртуализации в сфере больших данных рассказано в главе 5.

Изменение стоимости вычислений

Еще несколько лет — и новое оборудование и программное обеспечение возымели экономический эффект. Стоимость приобретения вычислительных ресурсов и ресурсов хранения данных снижалась на глазах. Поддерживаемые технологиями виртуализации типовые серверы, объединенные в кластеры, а также соединяемые в сеть блэйд-

серверы коренным образом изменили экономику вычислений. Наряду с появлением инновационных решений для автоматизации это заметно улучшило управляемость таких систем. Использование методов параллельной обработки и распределенных вычислений существенно изменило общую ситуацию и значительно уменьшило время ожидания. В отдельных случаях, например в высокочастотной торговле на бирже (HFT), малая задержка может быть достигнута лишь путем размещения серверов физически близко друг к другу.

Проблемы задержки

Неизменная проблема управления данными (особенно это касается больших объемов данных) — негативное влияние задержки. Время задержки — это всякого рода запаздывания в системе при выполнении операций. Это общая проблема для всех сфер вычислений, включая коммуникации, управление данными, производительность систем и многое другое. Достаточно вспомнить паузы в телефонном разговоре по междугородной или международной связи, когда передача информации между собеседниками запаздывает. Иногда задержка не оказывает большого влияния на степень удовлетворенности потребителей, например, когда компания анализирует достигнутые результаты, чтобы понять, когда следует планировать выпуск новой версии продукта. Здесь скорость поступления информации и быстрота реагирования на нее не будут критически важны. Однако чем быстрее требуется принять решение, тем большее значение имеет задержка.

Параллельная обработка и распределенные вычисления заметно влияют на время задержки для пользователя информации (заказчика, поставщика или партнера). Во многих приложениях больших данных необходимо низкое время задержки, что объясняется повышенными требованиями к скорости обработки, а также объемами и разнообразием данных. В среде с большими задержками эффективно использовать приложения больших данных не удастся. От задержки зависит и возможность проверки данных в режиме, близком к реальному времени. В главе 16 описаны особенности потоковой передачи данных реального времени и обработки сложных событий, которые критически важны для приложений больших данных. Эффективность работы с данными в реальном времени в значительной мере определяется низким или высоким значением задержки.

Спрос определяет решения

Распространение Интернета в качестве платформы для всех видов деятельности, начиная с торговли и заканчивая медициной, увеличило спрос на новое поколение решений для управления данными. В конце 1990-х годов интернет-компании, такие как Google, Yahoo! и Amazon.com, получили возможность расширить свои бизнес-модели, используя недорогое аппаратное обеспечение для вычислений и хранения данных. В дальнейшем таким компаниям потребовалось новое поколение программных технологий, чтобы монетизировать огромные объемы данных, поступающих от пользователей. Инновационные лидеры информационного рынка не собирались подолгу ждать результатов аналитической обработки — им нужна была возможность обрабатывать и анализировать данные в режиме, близком к реальному времени.

Оптимизация производительности

Для обеспечения необходимого уровня производительности при обработке больших данных недостаточно просто иметь быстрый компьютер. Необходима возможность распределять компоненты сервиса обработки больших данных между несколькими узлами (рис. 3.3). Узел в распределенных вычислениях — это элемент, находящийся внутри кластерной системы или серверной стойки, который обычно включает в себя центральный процессор, память и какой-либо диск. Иногда узел может быть представлен блэйд-сервером — сверхкомпактным компьютером с процессором и памятью, использующим вместо диска общее хранилище данных в серверной стойке.

В среде больших данных такие узлы, как правило, сгруппированы, чтобы обеспечить возможность масштабирования: например, когда возрастает количество анализируемых источников данных, к анализу постепенно подключаются дополнительные узлы в кластере. Однако простого увеличения числа узлов в кластере недостаточно. Гораздо важнее иметь возможность направить часть анализа больших данных в другие физические среды — и здесь выбор среды и подхода к управлению данными имеет критическую важность.

В некоторых сложных ситуациях для обеспечения требуемой скорости анализа возможно параллельное выполнение множества раз-

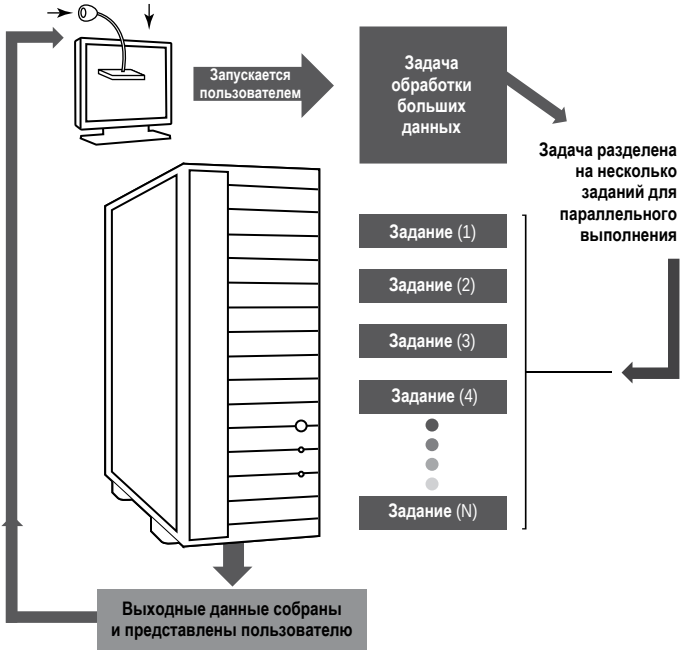


Рис. 3.3. Распараллеливание выполняемых задач

личных алгоритмов даже в рамках одного кластера. Дело в том, что распределение функций между вычислительными мощностями, расположенными физически близко друг к другу, повышает скорость их выполнения. Конечно, анализ больших данных можно распределять и по сети, когда есть такая возможность. Распределение должно быть обусловлено требованиями к производительности. В одних случаях скорость обработки не имеет принципиального значения, в других является приоритетом — и тогда необходимо, чтобы вычислительные мощности были приближены друг к другу. Другими словами, среда больших данных должна быть оптимизирована под тип решаемых с ее помощью аналитических задач.

Поэтому именно масштабируемость является ключом к успешному функционированию больших данных. Управлять средой больших данных в рамках одной крупной среды теоретически возможно, но на практике такой подход неэффективен. Чтобы понять, почему большие данные нуждаются в масштабируемой среде, достаточно посмотреть,

как функционирует облачная среда. Подобно облачным вычислениям, работа с большими данными требует быстрой сети и наличия недорогих кластеров аппаратного обеспечения, которые для увеличения производительности могут быть объединены в стойки. Такие кластеры управляются и контролируются с помощью программных систем, осуществляющих динамическое масштабирование и балансировку нагрузки.

Разработка и реализация модели MapReduce — хороший пример того, как с помощью распределенных вычислений можно сделать работу с большими данными прозрачной и экономически выгодной. Подробнее о MapReduce рассказано в главе 8. В сущности, сейчас можно наблюдать переломный момент в развитии вычислительных технологий: объединение преимуществ существующих технологических концепций и подходов позволяет решать задачи, которые ранее оставались за пределами возможностей. Комбинация распределенных вычислений, современных аппаратных средств и прикладных решений, таких как MapReduce и Hadoop, коренным образом изменяет способ управления данными.

Часть II

Технологические основы больших данных



В ЭТОЙ ЧАСТИ:

- Элементы стека больших данных
- Интеграция аналитика и приложений с большими данными
- Понятие виртуализации
- Влияние виртуализации на большие данные
- Абстракция
- Облачные вычисления и их роль в больших данных

Глава 4

Компоненты технологии больших данных

В ЭТОЙ ГЛАВЕ:

Общая информация о стеке больших данных
Избыточная физическая инфраструктура
Инфраструктура безопасности
Интерфейсы и потоки данных из приложений и в приложения
Операционные базы данных
Службы и инструменты организации данных
Аналитические хранилища данных
Введение в аналитику больших данных
Введение в приложения больших данных

Как уже отмечалось в предыдущих главах, большие данные — это высокоскоростные потоки данных различных типов в больших объемах. Для опытного разработчика программного обеспечения и архитектуры не представляет труда создать решение, которое позволяет работать с данными, обладающими одним или двумя из этих признаков. Например, для того чтобы обеспечить высокую отказоустойчивость при обработке больших объемов транзакционных данных, можно задействовать резервные кластеры реляционной базы данных в центре хранения и обработки данных с очень быстрой сетевой инфраструктурой. Или, к примеру, для интеграции различных типов данных из многих известных и неизвестных источников можно воспользоваться метамоделью, которая будет управлять хранилищем данных с учетом потребностей клиента.

Однако в динамично развивающемся мире больших данных невозможно разрабатывать индивидуальное решение для каждого. Когда лишь часть данных, с которыми работает компания, находится у нее

под контролем, а в остальном приходится полагаться на внешние ресурсы, в подобной гибридной среде требуется особая архитектурная модель, способная адекватно реагировать на динамический характер больших данных и одновременно позволяющая решать конкретные бизнес-задачи. В этой главе рассмотрены критерии, которым должна соответствовать архитектура технологического решения для работы с большими данными, и более подробно описан технологический стек больших данных, о котором упоминалось в главе 1.

Стек больших данных

Как и в случае с любой важной архитектурой данных, для работы с большими данными необходимо создать модель, которая обеспечивает целостное представление о том, каким образом ее элементы должны быть связаны для эффективного функционирования. Построение такой модели требует времени, однако это ускорит дальнейшие этапы доработки архитектуры и позволит создать более отказоустойчивую систему. К большим данным следует подходить не как к одноразовому проекту, а как к стратегии.

Идет ли речь о хранении, анализе больших данных, их применении для составления отчетов или решения прикладных задач, среда для работы с большими данными должна создаваться (и развиваться) в соответствии с четкими принципами проектирования. Необходимо учитывать аспекты, связанные с оборудованием, инфраструктурным, системным и управляющим программным обеспечением; следует проработать вопросы, связанные с правильным выбором интерфейсов прикладного программирования (API) и даже средств разработки программного обеспечения. Архитектура должна обеспечивать выполнение всех основных функций (см. главу 1):

- сбор;
- обобщение;
- систематизацию;
- анализ;
- действие.

На рис. 4.1 представлена многоуровневая ориентированная архитектура, упоминавшаяся в главе 1. Эта схема может быть взята за основу — она соответствует функциональным требованиям, предъявляемым к реализации проектов больших данных.



Рис. 4.1. Технологический стек больших данных

Схема достаточно сложна, и для начала можно сосредоточиться лишь на некоторых ее аспектах, необходимых в том или ином конкретном случае, однако в дальнейшем для решения всего спектра задач потребуется понимание всего стека в целом, чтобы в зависимости от решаемых задач эффективно использовать различные его элементы.

УРОВЕНЬ 0. Избыточная физическая инфраструктура

На нижнем уровне стека находится физическая инфраструктура — аппаратное обеспечение, сеть и т.д. Если в организации уже имеется центр хранения и обработки данных, приобретены необходимые объекты физической инфраструктуры, то имеет смысл найти способ эффективного их использования для работы с большими данными. Внедрение систем больших данных предъявляет специфические требования ко всем элементам базовой архитектуры, и эти требования необходимо последовательно рассмотреть, чтобы убедиться, что выбранная реализация обеспечит масштабируемость и желаемую функциональность. Перед началом внедрения больших данных важно

уяснить общие принципы, на которых должен строиться подход к этому процессу, и определить приоритеты организации.

- Производительность. Насколько быструю реакцию должна обеспечивать система? Производительность тесно связана со временем задержки, которое зачастую оценивается в рамках одной транзакции или одного запроса как интервал между началом запроса и получением ответа. Самые быстрые инфраструктуры (обладающие высокой производительностью и низким временем задержки), как правило, и самые дорогостоящие.
- Доступность. Требуется ли гарантировать работоспособность сервиса на 100%? Каково допустимое время простоя в случае перебоя или отказа в работе сервиса? Высокодоступные инфраструктуры также очень дорого стоят.
- Масштабируемость. Насколько большая инфраструктура необходима вашей организации? Какой объем дисковой памяти требуется на данный момент и на перспективу? Какая нужна вычислительная мощность? Обычно, определив необходимые значения этих характеристик, требования к ним немного повышают, чтобы иметь возможность маневра в случае непредвиденного увеличения нагрузки.
- Гибкость. Насколько быстро можно будет ввести новые ресурсы в уже имеющуюся инфраструктуру? Сколько потребуется времени на восстановление инфраструктуры после сбоя? Наиболее гибкие инфраструктуры могут быть дорогостоящими, однако можно сэкономить средства, обратившись к облачным сервисам, которые взимают плату только за реально используемые мощности и объемы (см. главу 6).
- Стоимость. Какую систему вы можете себе позволить? Поскольку инфраструктура представляет собой набор компонентов, можно сочетать лучшие из доступных на рынке сетевые технологии с более дешевыми хранилищами и наоборот. Необходимо сформулировать требования к каждому из компонентов исходя из общего бюджета и решить, какие из них приоритетны.

В связи с тем что большие данные характеризуются высокой скоростью, большим объемом и разнообразием, физическая инфраструктура является решающим фактором успеха их внедрения. Как правило, приложения больших данных должны быть постоянно готовы к работе, следовательно, сети, серверы и физические хранилища

должны одновременно быть отказоустойчивыми и обладать достаточным резервом. Отказоустойчивость и резервирование тесно связаны: инфраструктура или система является отказоустойчивой к сбоям или изменениям только при наличии достаточного количества готовых к использованию резервных ресурсов. Но в любой, даже в самой совершенной и отказоустойчивой, сети может произойти, например, аппаратный сбой. Наличие резервов гарантирует, что неисправности такого рода не вызовут перебоев в работе всей системы.

Отказоустойчивость позволяет исключить ситуации, когда отказ отдельных компонентов инфраструктуры приведет к отказу всей системы. Например, если компания осуществляет доступ к Интернету лишь через одно сетевое соединение, то есть не располагает резервными интернет-каналами, ее инфраструктура не является отказоустойчивой и уязвима для общих перебоев в работе сети. В больших центрах хранения и обработки данных, которые должны соответствовать требованиям к устойчивости функционирования, необходимый объем резервов присутствует и может быть использован для создания среды больших данных. Ответственность за эффективность создаваемой среды, ее производительность, соответствие потребностям бизнеса и понесенным затратам целиком и полностью лежит на разработчиках.



Все больше поставщиков предлагают платформы, основанные на облачных вычислениях, и соответственно в большинстве случаев именно на поставщиков услуг ложится ответственность за разработку аппаратной инфраструктуры.

Это означает, что компания-клиент не решает самостоятельно многочисленные технические и эксплуатационные вопросы, а получает набор услуг, для каждой из которых оговариваются требуемая производительность, доступность, условия восстановления и т.д. Все эти условия описываются в соглашении об уровне обслуживания (SLA) и, как правило, устанавливаются совместно поставщиком услуг и клиентом, как и штрафы за невыполнение обязательств.

Например, если вы заключаете договор с поставщиком услуг управления данными, теоретически вы освобождаетесь от забот, связанных со спецификой работы физической среды и основных компонентов центра хранения и обработки данных. Сети, серверы, операционные системы, структура виртуализации, необходимые инструменты управ-

ления и повседневная эксплуатация — все это оговорено в соглашении об уровне обслуживания. В сущности, вы пользуетесь виртуальным центром хранения и обработки данных. Но даже в этом случае полезно знать, как развертывать и вводить в эксплуатацию систему больших данных, — это поможет в выборе наиболее подходящих предложений и услуг. Хотя соглашение об уровне обслуживания регламентирует техническую сторону вопроса, в конечном счете производительность услуг зависит от того, как с ними работать.

Резервирование сетевой инфраструктуры

Сети должны обладать достаточным резервом и запасом производительности для обработки определенного объема входящих и исходящих данных с приемлемой скоростью — в дополнение к обработке «обычного» сетевого трафика компании. А поскольку решения больших данных внедряются как неотъемлемая часть стратегии обработки данных, логично предположить, что производительность и скорость потребуется наращивать и в дальнейшем.

Задача разработчика инфраструктуры — учесть объем дополнительных ресурсов, которые могут потребоваться, и сделать физическую инфраструктуру «эластичной». И сетевой трафик, и необходимость в дополнительных ресурсах варьируются во времени. Инфраструктура должна обеспечить возможность постоянного мониторинга, чтобы оперативно привлекать дополнительные ресурсы при изменении рабочих нагрузок.

Управление аппаратным обеспечением: хранилища и серверы

Аппаратное обеспечение (хранилища и серверы) должно поддерживать достаточную скорость и производительность для обработки всех планируемых объемов больших данных. Нет смысла при медленных серверах вкладываться в высокоскоростную сеть — вероятнее всего, серверы будут задерживать работу всей системы. Напротив, очень быстрая связка сервер-хранилище компенсирует переменную производительность сети. Разумеется, при ненадежной работе сети никакое оборудование не обеспечит высокую производительность.

Эксплуатация инфраструктуры

Еще один важный аспект проектирования инфраструктуры — управление ее эксплуатацией. Только хорошо управляемая среда может быть высокопроизводительной и достаточно гибкой. Руководители центров хранения и обработки данных должны уметь предвидеть и предотвращать внезапные отказы. Это позволит обеспечить целостность данных, а соответственно, и непрерывность бизнес-процессов. ИТ-компании часто недооценивают значимость данного аспекта и вкладывают недостаточно средств в развитие инфраструктуры. Более подробно об эксплуатации инфраструктуры больших данных рассказывается в главе 17.

УРОВЕНЬ 1. Инфраструктура безопасности

Требования к безопасности и конфиденциальности больших данных аналогичны требованиям, предъявляемым к обычным средам данных. Требования к безопасности должны четко соответствовать конкретным потребностям бизнеса. Когда большие данные становятся частью стратегии, необходимо учесть следующие специфические параметры.

- Доступ к данным. Технические требования к доступу пользователей к необработанным исходным или вычисляемым данным примерно одинаковы и для больших, и для традиционных данных: просматривать и изменять данные могут только те сотрудники, которым это необходимо для выполнения их обязанностей. Большинство основных платформ хранения данных содержит строгие схемы соблюдения требований безопасности. Зачастую они поддерживают средства разграничения доступа, что позволяет вводить соответствующие степени доступа для разных уровней архитектуры.
- Доступ приложений к данным. Обеспечение доступа приложений к данным относительно просто организуется с технической точки зрения. Большинство интерфейсов прикладного программирования (API) обеспечивает защиту от несанкционированного использования или доступа. Такого уровня защиты достаточно для большинства вариантов реализации больших данных.
- Шифрование данных. Шифрование данных — самый сложный вопрос обеспечения безопасности в среде больших данных. Шифрование и дешифрование данных значительно усиливает нагрузку на

ресурсы систем, а в среде больших данных эта проблема усугубляется большим объемом и разнообразием данных, а также высокой скоростью их обработки. Самый простой подход — шифровать все элементы подряд и обеспечить большую скорость вычислений. Однако это дорогостоящее решение, особенно если необходимо также обеспечить высокую отказоустойчивость. Более умеренный подход заключается в том, чтобы выявить элементы данных, в отношении которых требуется обеспечить необходимый уровень безопасности, и применять шифрование только к таким элементам.

- Обнаружение угроз. Использование мобильных устройств и социальных сетей в качестве источников информации значительно увеличивает как объемы данных, так и возможные угрозы безопасности. Сегодня в защите нуждается не только периметр корпоративной сети — необходим комплексный подход к защите информации.

О безопасности данных и управлении ими подробно рассказано в главе 19, где также рассмотрены возможности применения больших данных для обнаружения угроз и решения других проблем безопасности.

Интерфейсы и передача данных между приложениями и Интернетом

Итак, физическая инфраструктура — это основа функционирования среды больших данных, а инфраструктура безопасности защищает все ее элементы. Следующий уровень стека — интерфейсы, которые обеспечивают двусторонний доступ ко всем компонентам стека: от корпоративных приложений до потоковых данных из Интернета. Важный аспект проектирования таких интерфейсов — создание совместимой структуры коллективного доступа в пределах, а возможно, и за пределами компании. Это необходимо для взаимодействия с технологическими партнерами и партнерами по бизнесу.

Интерфейсы прикладного программирования (API) уже десятки лет применяются для обеспечения доступа к программным продуктам и организации их взаимодействия между собой. Поставщики этих интерфейсов и связанных с ними инструментов стремятся к тому, чтобы создание новых приложений с помощью их продуктов было простым и понятным. Тем не менее ИТ-специалистам иногда требуется разраба-

тывать специализированные корпоративные или закрытые коммерческие API-интерфейсы. Это непростая задача, но такие интерфейсы способны принести конкурентные преимущества, удовлетворить особые потребности компании или помочь в достижении иных целей. Чтобы API-интерфейсы приносили бизнесу пользу, их необходимо снабжать хорошо задокументированными описаниями и обеспечивать их поддержку. Для ускорения этой сложной и важной работы некоторые компании делают выбор в пользу инструментальных средств разработки.

Интерфейсы API, созданные с использованием таких средств, обладают рядом преимуществ перед создаваемыми «с нуля» силами корпоративных программистов. Во-первых, комплекты инструментальных средств разработки интерфейсов прикладного программирования — это продукты, создаваемые, управляемые и поддерживаемые независимой третьей стороной. Во-вторых, они хорошо приспособлены для конкретных технических решений. Например, в настоящее время широко представлены пакеты средств разработки API-интерфейсов для веб-сервисов и мобильных приложений.

Технология передачи репрезентативного состояния (REST)

Обзор интерфейсов прикладного программирования для больших данных не будет исчерпывающим без упоминания технологии REST (Representational State Transfer, передача репрезентативного состояния). REST — самый распространенный механизм подключения одного веб-источника (сервера) к другому веб-источнику (клиенту), разработанный специально для Интернета. Интерфейс RESTful API обеспечивает стандартизированный способ установления временных взаимоотношений (слабого связывания) между веб-ресурсами. Как следует из названия, слабосвязанные ресурсы жестко не связаны между собой. Это обеспечивает отказоустойчивость при изменениях в сетях или в других компонентах инфраструктуры. Например, у вас ночью сломался холодильник, и вы хотите приобрести новый. Но для этого придется подождать сначала открытия магазина, а потом доставки. Примерно так же функционируют веб-сервисы, использующие RESTful API: запрос остается без ответа, ожидая, пока освободится соответствующий сервис. Многие, если не все, технологии больших данных поддерживают REST. Об этом рассказывается в последующих главах.

Использование больших данных требует специфического подхода к разработке или внедрению интерфейсов прикладного программирования. Поскольку значительная доля данных неструктурирована и генерируется за пределами использующей их организации, для взаимодействия между большими данными и прикладными программа-

ми предпочтительно использовать новый метод, известный как NLP (Natural Language Processing, обработка естественного языка). NLP позволяет сформулировать запросы на естественном языке, а не на формальном языке запросов наподобие SQL. Большинству пользователей гораздо проще запросить «Список всех женатых мужчин в возрасте от 30 до 40 лет в таком-то городе, которые смотрят гонки NASCAR», чем составлять SQL-запрос из нескольких десятков строк.



Одним из способов реализации интерфейсов является внедрение «фабрики соединителей», которая добавляет к процессу уровень абстракции и прогнозирования. Этот способ позволяет использовать возможности сервисно-ориентированной архитектуры (SOA).

Большинство процессов сбора и перемещения данных имеют очень схожие характеристики, что позволяет создать набор служб для сбора, очистки, преобразования, нормализации и хранения элементов больших данных практически в любой системе хранения данных. Для обеспечения необходимой гибкости «фабрикой» можно управлять посредством языка описания интерфейсов XML (eXtensible Markup Language, расширенный язык разметки). Использование уровня абстракции позволяет быстро и легко создавать соответствующие интерфейсы, что избавляет от необходимости разрабатывать отдельные службы для каждого источника данных.

На практике можно создавать описания интерфейсов приложений SAP или Oracle с помощью XML или подобных ему инструментов. Каждый интерфейс будет использовать одно и то же базовое программное обеспечение для перемещения данных между средой больших данных и средой приложений, независимо от специфики приложений SAP или Oracle. Аналогичным образом можно собирать в Интернете данные социальных сетей (Facebook, Google+ и т.д.). Достаточно описать интерфейсы сайтов на XML, а затем пользоваться сервисами перемещения данных в необходимом направлении. Как правило, такие интерфейсы задокументированы, что позволяет их применять и техническим специалистам компаний, и сторонним разработчикам.

УРОВЕНЬ 2. Операционные базы данных

Основа любой среды больших данных — ядро системы управления базами данных (СУБД), содержащей совокупность элементов данных, необходимых предприятию. СУБД должны быть быстрыми, масштабируемыми и надежными, однако каждая из них имеет свою специфику. Некоторые среды больших данных более эффективно функционируют

с одним ядром СУБД, нежели с другим, в иных случаях предпочтительно комбинировать несколько СУБД, чтобы подбирать наиболее подходящее решение для каждой сферы применения. Хотя можно для работы со всеми большими данными использовать реляционные системы управления базами данных, на практике это неэффективно ввиду возникающих трудностей с производительностью и масштабированием или по причине их стоимости. Сориентироваться в разнообразии представленных на рынке технологий баз данных непросто. Подробнее о базах данных рассказывается в главе 7.

Единственно правильного подхода к выбору языка базы данных не существует. Несмотря на то что SQL на сегодняшний день является наиболее распространенным языком запросов базы данных, другие языки могут оказаться более эффективными для решения той или иной конкретной задачи. Ядра и языки баз данных — это внушительный набор инструментов на все случаи жизни, и важно научиться выбирать наиболее подходящий.

К примеру, при работе с реляционной моделью запросы обычно осуществляют на SQL, но это не исключает возможности использования и других языков, таких как Python или Java. Очень важно понимать, какие типы данных могут быть обработаны базой данных и поддерживает ли она транзакционное поведение. Такое поведение сводится к соответствию четырем пунктам, которые обозначают аббревиатурой ACID.

- Атомарность (atomicity). Транзакция является атомарной, если она осуществляется по принципу «все или ничего». Если какая-либо часть транзакции не выполняется или происходит сбой в базовой системе, вся транзакция завершается неудачей.
- Согласованность (consistency). В базе данных выполняются только транзакции с подтвержденными данными. Если данные повреждены или не подходят, транзакция не будет завершена и данные не будут записаны в БД.
- Изоляция (isolation). Множественные одновременные транзакции выполняются независимо друг от друга. Все подтвержденные транзакции выполняются в том порядке, в котором они были представлены для обработки, и до тех пор, пока не будут завершены.
- Устойчивость (durability). После того как данные транзакции заносятся в базу данных, они остаются в системе «навсегда».

В таблице 4.1 представлена сравнительная характеристика баз данных на основе SQL и NoSQL.

Ядро	Язык запросов	MapReduce	Типы данных	Транзакции	Примеры
Реляционное	SQL, Python, C	Нет	Типизированные	ACID	PostgreSQL, Oracle, DB/2
Столбцовое	Ruby	Hadoop	Предопределенные и типизированные	Да, если используются	HBase
Графовое	Walking, Search, Cypher	Нет	Нетипизированные	ACID	Neo4J
Документальное	Commands	JavaScript	Типизированные	Нет	MongoDB, CouchDB
Ключ-значение	Lucene, Commands	JavaScript	BLOB, полутипизированные	Нет	Riak, Redis

Таблица 4.1. Основные параметры баз данных SQL и NOSQL

Определившись с техническими требованиями, спектром собираемых данных и областью их применения, необходимо организовать данные так, чтобы их можно было использовать для аналитики, генерации отчетов или решения прикладных задач.

УРОВЕНЬ 3. Службы и инструменты для организации данных

Службы и инструменты для организации данных осуществляют сбор, проверку и интеграцию различных элементов больших данных в контекстно-зависимые совокупности. Современные методы и инструменты учитывают огромные объемы больших данных и позволяют осуществлять их обработку эффективно и непрерывно. Рассматриваемая в главе 8 модель распределенных вычислений MapReduce является одним из наиболее распространенных методов: достаточно отметить, что многие из служб, предназначенных для организации данных, строятся на ядрах MapReduce, оптимизированных для организации потоков больших данных.

Службы организации данных по сути являются экосистемой инструментов и технологий, применяемых для сбора и объединения данных при подготовке их к дальнейшей обработке. Такие инструмен-

ты должны обеспечивать интеграцию, перевод, нормализацию и масштабирование данных. На данном уровне применяются следующие технологии.

- Распределенная файловая система. Необходима для декомпозиции (разбиения) потоков данных и обеспечения масштабирования и необходимого объема хранения.
- Службы сериализации. Отвечают за постоянное хранение данных и осуществление многоязычных удаленных вызовов процедур RPCs (Remote Procedure Calls, удаленные вызовы процедур).
- Службы координации. Необходимы для создания распределенных приложений (блокировки и т.д.).
- Инструменты для извлечения, преобразования и загрузки (ETL). Необходимы для загрузки и преобразования структурированных и неструктурированных данных в систему Hadoop.
- Службы рабочих процессов. Обеспечивают планирование заданий и синхронизацию элементов процессов на разных уровнях.

В главах 9 и 10 Hadoop рассматривается более подробно. Это наиболее широко используемый набор продуктов для организации больших данных, открытое программное обеспечение, поддерживаемое фондом Apache Foundation.

УРОВЕНЬ 4. Аналитическое хранилище данных

Обычные хранилища и витрины данных (по сути являющиеся малыми хранилищами данных) давно применяются для оптимизации данных, используемых в процессе принятия решений. Как правило, обычные и малые хранилища данных содержат нормализованные данные, собранные из различных источников и объединенные для оптимизации процедур бизнес-анализа. Обычные и малые хранилища данных упрощают процедуру создания отчетов и визуализации различных элементов данных. Обычно в их основе реляционные БД, многомерные БД, плоские файлы и объектные БД — в сущности, для них подходят любые архитектуры хранения данных. В традиционной среде производительность может и не быть фактором первостепенного значения, и выбор технологий построения корпоративных систем обработки данных определяется требованиями к анализу, отчетности и визуализации.

Чтобы обеспечить организацию данных и их готовность к анализу, хранилища, как правило, актуализируют данные методами пакетной обработки. Однако для нужд больших данных этого во многих случаях уже недостаточно. Ввиду значительных нагрузок обработка высокоскоростных потоков данных потребует от хранилищ больших данных функционирования в режимах реального времени. Это не означает, что аналитические хранилища и витрины данных, наполняемые посредством пакетной обработки, уйдут в прошлое. Напротив, вполне вероятно, что различные хранилища данных будут применяться параллельно, а их масштаб и производительность будут зависеть от задач, которые решают с их помощью аналитики и руководители.

Многие обычные и малые хранилища содержат данные из различных источников внутри компании, и не стоит забывать, что эти данные также необходимо подвергать очистке и нормализации. В работе с большими данными из внутренних источников появляется своя специфика:

- традиционные потоки данных (поступающие в результате осуществления транзакций, от приложений и так далее) могут создавать большое количество различного рода данных;
- появились десятки новых источников данных, которые должны быть определенным образом обработаны, прежде чем их можно будет эффективно использовать для нужд бизнеса;
- источники контента также должны пройти очистку, причем она может потребовать подходов, отличных от тех, которые применяются в работе со структурированными данными.

Раньше данные из хранилищ организовывались и поступали к руководителям, ответственным за разработку стратегии и планирование. Большие данные вызвали появление целых групп специалистов, которые используют данные для принятия решений. Множество вариантов реализации больших данных обеспечивает возможности работы в реальном времени, что позволяет направлять результаты анализа данных тем сотрудникам, которые могут предпринять с их помощью конкретные меры. Аналитику можно использовать для своевременного (в режиме, близком к реальному времени) решения таких задач, как поддержка клиентов, обеспечение сбыта и предотвращение перебоев в обслуживании. Таким образом, использование больших данных способствует перераспределению бизнес-активности с административного уровня на уровень взаимодействия с клиентами.

В главе 11 описывается применение многих технологических подходов к организации хранилищ больших данных и приводятся рекомендации по эффективному использованию таких хранилищ.

Аналитика больших данных

Для работы с большими данными можно применять уже существующие аналитические инструменты и методы. Однако трудность заключается в том, что алгоритмы в составе таких инструментов не всегда могут справляться с обработкой больших объемов различного рода данных в режиме реального времени. Кроме того, требуется соответствующая инфраструктура, о которой говорилось выше, а на поставщиков аналитических инструментов ложится обязанность обеспечить работу алгоритмов во всех распределенных системах. Логично предположить, что наряду с существующими средствами будут применяться и инструменты нового поколения, приспособленные для обработки больших данных.

Отметим три основных типа инструментов, которые используются на этом уровне архитектуры. Они могут применяться как по отдельности, так и в совокупности.

- Отчеты и панели мониторинга. Эти инструменты обеспечивают представление информации из различных источников в удобной для пользователя форме. Они широко применяются для обработки традиционных данных, но к большим данным пока только приспосабливаются. Некоторые традиционные инструменты развивают новый функционал и приобретают возможность работать с новыми типами баз данных, которые известны под общим названием NoSQL (Not Only SQL, «не только SQL»). О базах данных NoSQL рассказывается в главе 7.
- Визуализация. Инструменты этого типа выводят создание отчетов на новый этап. Выходные данные визуализации, как правило, взаимосвязаны друг с другом и имеют очень динамичный характер. Другим важным отличием визуализированного представления результатов от традиционных отчетов является применение анимации. Такие технологии визуализации, как динамически меняющиеся диаграммы, карты, схемы связей, инфографика, позволяют наглядно представить процессы и результаты в движении. Составление отчетов и визуализация осуществляются, как правило, по за-

вершении проекта. Хотя данные впоследствии могут быть переданы другому инструменту для дальнейших расчетов или изучения, визуализация является заключительным этапом решения задачи.

- Аналитика и углубленная аналитика. Инструменты этого типа обрабатывают данные в хранилище, представляя результаты в виде, доступном для восприятия человеком. Углубленная аналитика позволяет выявлять тенденции или события, которые могут оказать существенное и даже революционное воздействие на существующие бизнес-процессы. Примерами таких инструментов являются прогнозная аналитика и аналитика мнений. Подробнее об аналитике рассказывается в части IV (главы 12, 13, 14).

Приложения, задействующие большие данные

Пользовательские приложения и приложения сторонних производителей предлагают альтернативные способы изучения и совместного использования больших данных. Каждый уровень архитектуры важен, но именно в разработке приложений реализуется творческий и инновационный потенциал компании.

Приложения могут быть как горизонтальными (для решения задач, общих для нескольких сфер деятельности), так и вертикальными (для решения задач, характерных для одной отрасли). Ассортимент приложений чрезвычайно широк и продолжает расти, и с распространением технологий больших данных следует ожидать и быстрого увеличения количества приложений, предназначенных для работы с ними. Стоит отметить приложения анализа данных лог-журналов (Splunk, Loggly), рекламные, мультимедийные приложения (Bluefin, DataXu) и маркетинговые приложения (Bloomreach, Myrrix). Решения, задействующие большие данные, разрабатываются также для нужд медицины, промышленного производства, логистики и многих других сфер деятельности.

Как и любая другая инициатива по разработке пользовательских приложений, создание приложений больших данных требует определенной структуры, соблюдения стандартов, технических требований и реализации удобных API-интерфейсов. Большинство бизнес-приложений, использующих большие данные, должны быть доступны через API-интерфейсы в масштабах всего стека. Возможно, появится необходимость обрабатывать исходные данные из низкоуровневых массивов

и объединять их с синтезированными данными из хранилищ. Естественно, на первое место выходит адаптация приложений под индивидуальные нужды пользователя — а это еще один фактор, который нужно иметь в виду.

Большие данные поступают очень быстро и изменяются моментально, так что разработчикам программного обеспечения потребуется оперативно создавать приложения, способные решать актуальные бизнес-задачи. Возможно, в некоторых компаниях понадобится организовать специальные группы экспертов-разработчиков, которые смогут быстро реагировать на изменения в бизнес-среде и создавать и развертывать приложения по требованию. В сущности, такие приложения можно назвать «полупользовательскими», поскольку их разработка предполагает не столько написание низкоуровневого кода, сколько компоновку элементов.



Можно ожидать, что со временем определенные типы приложений будут создаваться конечными пользователями — они смогут сами собирать нужные решения из множества доступных компонентов. Разумеется, структура и стандартизация в таком случае будут иметь первоочередное значение, поэтому для быстрого развертывания приложений больших данных важно создать совместимые стандартизированные среды и новые методы разработки.

Глава 5

Виртуализация и распределенные вычисления

В ЭТОЙ ГЛАВЕ:

Что такое виртуализация
Что такое гипервизор
Основы абстракции и виртуализации
Применение виртуализации для работы с большими данными

Виртуализация — это базовая технология, применяемая для внедрения как облачных вычислений, так и больших данных. Она служит основой для многих атрибутов платформы, необходимых для доступа к компонентам распределенных вычислений, их хранения, анализа и управления ими в средах больших данных. Виртуализация — это процесс использования одних ресурсов компьютера для имитации других ресурсов, что позволяет повысить эффективность имеющихся ИТ-ресурсов, привлечь дополнительные ресурсы, а также обеспечить масштабируемость. Один из основных вариантов применения виртуализации — объединение серверов. Это помогает увеличить коэффициент использования физических серверов и потенциально сэкономить затраты на инфраструктуру. Вместе с тем у виртуализации есть и другие применения. Компании, ранее применявшие исключительно виртуализацию серверов, приходят к пониманию того, что виртуализацию можно использовать в масштабах всей ИТ-инфраструктуры, включая программное обеспечение, системы хранения и сети.

В этой главе мы дадим определение виртуализации, а также расскажем о преимуществах и проблемах виртуализированных сред. Основное внимание уделено роли виртуализации в работе с большими данными.

Суть виртуализации

Виртуализация отделяет ресурсы и сервисы от лежащего ниже физического уровня, т.е. от физической среды доставки. Это позволяет создавать множество виртуальных систем в рамках одной физической системы. На рис. 5.1 показана типичная среда виртуализации.

Одна из главных целей внедрения виртуализации — повышение производительности и эффективности обработки различных рабочих нагрузок. Вместо того чтобы присваивать каждому набору задач совокупность физических ресурсов, можно зарезервировать пул виртуальных ресурсов в том объеме, который необходим для всех рабочих нагрузок. Использование пула виртуальных ресурсов позволяет уменьшить время задержки. Возможность увеличения скорости и эффективности сервисов напрямую связана с распределенной природой виртуализированных сред и помогает оптимизировать затрачиваемое время и снизить расходы.



Рис. 5.1. Использование программного обеспечения виртуализации для создания нескольких виртуальных систем в рамках одной физической среды

Использование распределенной совокупности физических ресурсов (таких как серверы) более гибким и эффективным способом позволяет значительно снизить затраты и повысить производительность. Виртуализация имеет ряд преимуществ:

- виртуализация физических ресурсов (таких как серверы, хранилища данных и сети) позволяет существенно повысить эффективность их использования;
- виртуализация позволяет повысить производительность ИТ-ресурсов и усилить контроль за их использованием;
- при оптимизации вычислительной среды виртуализация может обеспечить необходимый уровень автоматизации и стандартизации;
- виртуализация обеспечивает основу для применения облачных вычислений.

Виртуализация ресурсов — чрезвычайно эффективное мероприятие, но оно требует определенных затрат в связи с тем, что управление виртуальными ресурсами должно происходить с учетом обеспечения их защиты. С помощью виртуального образа злоумышленник может получить прямой доступ к важным системам. Кроме того, если не удалять неиспользуемые образы, со временем это приведет к потере эффективности функционирования системы.

Важность виртуализации для больших данных

Использование больших данных тесно связано с управлением большим количеством высокораспределенных хранилищ данных, имеющих значительный объем, и вычислительных приложений. Следовательно, для обработки больших данных нужна высокоэффективная ИТ-среда. Виртуализация — способ повышения эффективности среды, благодаря которому использование платформ больших данных становится повседневной реальностью. Отметим, что виртуализация не является обязательным требованием для осуществления анализа больших данных. Однако используемые в средах больших данных программные платформы, такие как MapReduce (см. главу 8), наиболее эффективны именно в виртуализированной среде. Чтобы масштабировать среду больших данных, ее элементы необходимо виртуализировать — это сделает доступным практически безграничный объем масштабирования.

Виртуализация имеет три характеристики, обеспечивающие необходимую для сред больших данных масштабируемость и эксплуатационную эффективность:

- секционирование. В виртуализации множество приложений и операционных систем поддерживаются в рамках единой физической

системы путем секционирования (разделения) доступных ресурсов;

- изоляция. Каждая виртуальная вычислительная машина изолирована от своей основной физической системы и других виртуализированных машин. Благодаря этому, если один виртуальный экземпляр выходит из строя, сбой не влияет на другие виртуальные машины и основную систему. Кроме того, данные не используются совместно в разных виртуальных экземплярах (не являются общими);
- инкапсуляция. Виртуальная машина может быть представлена (и даже храниться) в виде одного файла так, чтобы ее можно было легко идентифицировать, основываясь на типе предоставляемых сервисов. К примеру, файл, содержащий инкапсулированный процесс, может быть бизнес-сервисом полного цикла. Такую инкапсулированную виртуальную машину можно загрузить в приложение в качестве полноценной структурной единицы. Таким образом, инкапсуляция обеспечивает защиту каждого приложения, чтобы оно не препятствовало работе других приложений.

Одним из ключевых параметров в работе с большими данными является обеспечение уровня производительности, достаточного для анализа больших объемов и большого разнообразия данных. При работе в таких средах, как Hadoop и MapReduce, важно располагать вспомогательной инфраструктурой, обладающей запасом масштабирования. Виртуализация повышает эффективность всех уровней ИТ-инфраструктуры и помогает обеспечить необходимый уровень масштабируемости.

Внедрение виртуализации в качестве комплексного подхода дает ряд преимуществ в работе с большими данными и другими типами рабочих нагрузок. Комплексный подход означает, что ошибки будут исправляться намного быстрее, как того требует среда больших данных. При работе с большими данными инфраструктура должна быть подготовлена к управлению данными, потенциально имеющими огромный объем, высокую скорость и большое разнообразие форматов.

Поэтому потребуется оптимизировать все уровни ИТ-среды, начиная с сети и заканчивая базами данных, хранилищами и серверами. Если виртуализировать только серверы, то другие элементы инфраструктуры, такие как система хранения и сети, могут превратиться в узкие места и ограничить возможности всей среды. Виртуализация

лишь одного из элементов инфраструктуры не дает возможности существенно снизить время задержки и достичь необходимой эффективности и даже может повлечь дополнительные затраты и повышенные риски для безопасности.

На практике большинство организаций не пытается одновременно виртуализировать все элементы собственной инфраструктуры. Многие начинают с виртуализации серверов, обеспечивают определенный уровень улучшения эффективности и на этом останавливаются. Правильнее было бы постепенно виртуализировать и остальные элементы инфраструктуры, чтобы повысить производительность и эффективность всей системы в целом. Далее рассмотрим, как виртуализация каждого элемента ИТ-среды — серверов, хранилищ данных, приложений, данных, сетей, процессоров, памяти и сервисов — помогает в анализе больших данных.

Виртуализация серверов

При виртуализации серверов один физический сервер разделяется на несколько виртуальных. Аппаратное обеспечение и ресурсы системы, в том числе оперативную память (ОЗУ), центральный процессор (ЦП), жесткий диск и сетевой контроллер, можно виртуализировать (логически разделить) на ряд виртуальных машин, на каждой из которых запускается своя операционная система и выполняются собственные приложения. Виртуальная машина (ВМ) — это программное представление физической машины, которое может исполнять те же функции, что и сама физическая машина. На аппаратное обеспечение устанавливается специальное программное обеспечение, называемое тонким клиентом. Такое ПО обеспечивает работу монитора виртуальной машины, или гипервизора. Гипервизор можно рассматривать как технологию, которая управляет потоком данных между виртуальной и физической машинами.

Гипервизор при виртуализации серверов обеспечивает эффективность использования физических ресурсов. Естественно, что установка, конфигурация и административные задачи непосредственно связаны с настройкой виртуальных машин. Такая настройка включает в себя управление лицензиями, управление сетью, управление рабочими нагрузками, а также планирование производительности.

Виртуализация серверов дает возможности масштабирования вашей платформы, необходимые для анализа больших объемов и раз-

личных типов данных. Поскольку не всегда можно определить до начала анализа, какой объем структурированных и неструктурированных данных будет использоваться, потребность в виртуализации серверов возрастает — этот метод позволяет справиться с непредвиденными нагрузками при обработке огромных массивов данных.

Кроме того, виртуализация серверов обеспечивает основу для использования различных облачных сервисов в качестве источников данных при осуществлении анализа больших данных. Виртуализация повышает эффективность облака, что упрощает оптимизацию многих сложных систем и позволяет обеспечить необходимую производительность, а это, в свою очередь, дает возможность работать с ранее недоступными для применения или трудными для сбора данными. Платформы больших данных все чаще используются как источники огромного количества данных о предпочтениях, мнениях и поведении клиентов. Эту информацию можно интегрировать с данными о внутренних продажах и продуктах, чтобы изучить предпочтения клиентов и формировать направленные и индивидуализированные предложения.

Виртуализация приложений

Виртуализация инфраструктуры позволяет эффективно управлять приложениями для решения задач, связанных с покупательским спросом. Приложения инкапсулируются таким образом, что становятся независимыми от основной физической компьютерной системы. Это повышает общую управляемость и переносимость приложений. Кроме того, виртуализация инфраструктуры приложений обычно допускает применение кодифицированной технической и бизнес-политики. Это гарантирует, что каждое из приложений использует виртуальные и физические ресурсы предсказуемым образом. Повышение эффективности достигается благодаря тому, что ИТ-ресурсы легко распределить между приложениями в соответствии с их важностью для решения той или иной задачи; другими словами, наиболее важные приложения при необходимости получат высший приоритет использования доступной вычислительной мощности и объемов хранилища.

Виртуализация инфраструктуры приложений наряду с виртуализацией серверов способствует выполнению требований, описываемых в соглашениях об уровне обслуживания (SLA). Виртуализация серверов позволяет контролировать использование центрального про-

цессора и оперативной памяти, но не распределение ресурсов в соответствии с приоритетами. К примеру, если всем приложениям по умолчанию задан высокий приоритет, внедрение виртуализации инфраструктуры приложений в дополнение к виртуализации серверов позволит обеспечить приложениям, используемым для решения конкретной бизнес-задачи, первоочередной доступ к ресурсам.

Ввиду больших объемов или скорости генерации выполнение приложений больших данных может потребовать значительного количества ИТ-ресурсов. Необходим достаточный уровень предсказуемости и воспроизводимости среды больших данных, чтобы обеспечить приложениям своевременный доступ к необходимым ресурсам. Виртуализация инфраструктуры приложений гарантирует, что каждое развертываемое для анализа больших данных приложение получает доступ к необходимой вычислительной мощности на основе присвоенного ему приоритета. Кроме того, виртуализация инфраструктуры приложений упрощает запуск приложений на разных компьютерах, позволяя одновременно выполнять несовместимые или неподдерживаемые приложения на одной и той же физической машине, таким образом, не придется создавать отдельные версии приложения для разных операционных систем, например для Windows и Linux.

Платформы больших данных, разработанные для поддержки высокораспределенных и информационноемких приложений, будут работать лучше и быстрее именно в виртуальной среде. Это не означает, что необходимо виртуализировать все приложения, связанные с большими данными: к примеру, приложение для текстовой аналитики эффективнее выполняется в независимой среде, и виртуализация никак не повысит его производительность.

Виртуализация сетей

Виртуализация сетей (программно-конфигурируемое сетевое взаимодействие) позволяет эффективно использовать сети в качестве пула ресурсов соединения. Виртуализация сетей осуществляется таким же образом, как и виртуализация других физических ресурсов: для управления потоками данных создается множество виртуальных сетей, которые выполняют функции физической сети. Это полезно в том случае, когда необходимо использовать две сети с разными показателями производительности и мощности, например, если одна оптимизирована для сбора данных, а другая — для запуска приложений. Кроме того,

виртуализация позволяет справиться с узкими местами, возникающими из-за ограничений сетевого уровня, и улучшить управление большим количеством необходимых для анализа распределенных данных.

Виртуализация процессора и памяти

Виртуализация процессора оптимизирует использование процессора и увеличивает его производительность. Виртуализация памяти отделяет память от серверов.

При анализе больших данных для поиска новых закономерностей и тенденций могут потребоваться повторяющиеся запросы больших массивов данных и алгоритмы углубленной аналитики. Такая углубленная аналитика требует большого объема вычислительной мощности процессора (ЦП) и памяти (ОЗУ), в противном случае некоторые расчеты займут очень много времени. Виртуализация процессора и памяти может ускорить процесс обработки данных и способствовать более быстрому получению результатов анализа.

Виртуализация данных и накопителей информации

С помощью виртуализации данных можно создать платформу динамически связанных служб данных, которая облегчает поиск данных и их связывание между собой, поскольку они используются как единый источник. В результате виртуализация данных позволяет с помощью абстрактного сервиса представить данные в совместимых форматах вне зависимости от физической базы данных в их основе. Кроме того, виртуализация данных предоставляет всем приложениям кэшированные данные, а это повышает производительность.

Виртуализация системы хранения данных объединяет физические ресурсы для их более эффективного совместного использования, что снижает стоимость хранения и упрощает управление хранилищами, необходимыми для анализа больших данных.

Виртуализация данных и виртуализация системы хранения уменьшает затраты на хранение, извлечение и анализ больших объемов разных типов высокоскоростных данных и облегчает их анализ. Кроме того, хранение большого количества неструктурированных данных представляет трудность для традиционных методов хранения, а виртуализация системы хранения данных позволяет решить эту проблему. В среде больших данных к ряду хранилищ оперативных данных

целесообразно иметь доступ по требованию. Например, если столбцовая база данных используется лишь изредка, ее можно хранить в виде виртуального образа и запускать только при необходимости, не задействуя ценные ресурсы или свободное дисковое пространство центров обработки и хранения данных.

Виртуализация и проблемы управления и безопасности

Чтобы виртуализация повышала эффективность и экономила средства, необходимо правильно управлять виртуализированными средами. При использовании больших данных для решения аналитических задач необходимо обеспечить виртуальной среде такую же защиту и надежность управления, как и физической среде. При ненадлежащем использовании преимущества виртуализации, такие как удобство конфигурирования, могут обернуться проблемами в сфере управления и безопасности. Так, виртуализация позволяет разработчику легко создать виртуальный образ (копию) ресурса. Оборотная сторона этого преимущества — сложность контроля над растущим количеством виртуальных образов. Вот некоторые возможные проблемы, связанные с виртуализацией:

- создание большого количества виртуальных образов может повлечь резкое снижение производительности сервера и памяти;
- отсутствие контроля за жизненным циклом виртуальных образов создает уязвимости в системе обеспечения безопасности;
- переизбыток виртуальных образов увеличивает затраты на хранение данных, таким образом, виртуализация перестает быть экономически оправданной;
- преднамеренные и непреднамеренные вредоносные действия администраторов при управлении виртуальными образами повышают уровень риска;
- если не осуществлять надлежащим образом мониторинг регистрационных журналов виртуальной инфраструктуры, возникает риск нарушения технических требований.

Гипервизор и управление виртуализацией

Гипервизор — это технология, организующая распределение ресурсов системы и оборудования одним и тем же образом в соответствии с предъявляемыми требованиями. Это своего рода регулировщик движения, который позволяет операционным системам функционировать, не мешая друг другу. Гипервизор создает и запускает виртуальные машины. Он располагается на нижних уровнях аппаратной среды

и динамически распределяет ресурсы, используя тонкий программный клиент. В результате, с точки зрения пользователя, любой из операционных систем доступны все имеющиеся физические ресурсы.

В мире больших данных используются различные операционные среды, и гипервизор — это идеальный механизм для связи между технологическими компонентами стека больших данных. Гипервизор позволяет вам запустить одно и то же приложение на множестве различных систем без необходимости физически копировать это приложение в каждую систему. Дополнительное преимущество архитектуры гипервизора заключается в возможности загружать множество различных операционных систем в качестве обычных приложений. Таким образом, гипервизор — это практичный способ быстро и эффективно осуществить виртуализацию.



Гипервизор устроен скорее как серверная операционная система, нежели как Windows. Каждая запущенная на физической машине виртуальная машина называется гостевой, а операционные системы, запущенные на виртуальных машинах, гостевыми ОС. Гипервизор, таким образом, распределяет доступ гостевых операционных систем ко всем ресурсам, в том числе к процессору, памяти, дисковому пространству и другим механизмам ввода-вывода. Виртуализация позволяет с помощью гипервизора распределить физические ресурсы компьютера. К примеру, между двумя гостевыми ОС ресурсы могут быть распределены в пропорциях 50/50 или 80/20.

Гипервизор берет на себя самую сложную часть работы. Гостевая операционная система «не знает», что она запущена в виртуальном разделе, и функционирует так, как будто в ее распоряжении весь компьютер.

Гипервизоры можно разделить на два основных типа.

- Гипервизоры первого типа запускаются и работают непосредственно на аппаратной платформе и потому намного эффективнее других.
- Гипервизоры второго типа запускаются и работают в основной операционной системе. Они позволяют поддерживать широкий спектр устройств ввода-вывода.

Абстракция и виртуализация

Перед осуществлением виртуализации ИТ-ресурсов и сервисов необходимо отделить их от основной физической среды. Этот процесс называется абстракцией. Абстракция — одна из ключевых концепций работы с большими данными. Распределенные вычислительные среды

MapReduce и Hadoop широко задействуют абстракцию. Объект абстрагируется таким образом, что разработчику или аналитику нет нужды беспокоиться о том, где фактически находятся элементы данных.

Абстракция упрощает представление объекта, исключая нерелевантные подробности. Например, если вы договариваетесь о встрече с человеком, которого раньше не видели, он сообщит вам свой цвет волос, рост, опишет одежду, но не станет рассказывать о том, где родился, сколько зарабатывает и как зовут его детей, поскольку эта информация не имеет значения для конкретной задачи — распознать человека в толпе. Так же работает и абстракция: описание объекта делается на высоком уровне и не касается деталей его функционирования. К примеру, в облачной модели доставки IaaS (Infrastructure as a Service, инфраструктура как услуга) детали физической и виртуальной инфраструктуры абстрагированы от пользователя.

Применение виртуализации для работы с большими данными

Благодаря виртуализации ИТ-среда может осуществлять анализ больших данных намного эффективнее. Оптимизация всех элементов инфраструктуры, включая аппаратные средства, программное обеспечение и систему хранения, обеспечивает требуемый уровень производительности для обработки большого количества структурированных и неструктурированных данных и управления ими. При работе с большими данными доступ к структурированным и неструктурированным данным, управление такими данными и их анализ осуществляются в распределенной среде.

Работа с большими данными предполагает распределение. На практике любой тип MapReduce будет намного эффективнее работать именно в виртуализированной среде, которая позволяет перемещать рабочие нагрузки между системами в зависимости от требований к вычислительной мощности и хранению.

Виртуализация дает возможность быстро решать непредвиденные проблемы — никогда не известно, насколько быстро может потребоваться масштабировать систему.

Виртуализация позволяет поддерживать различные операционные хранилища больших данных. К примеру, графовая база данных может быть представлена в виде образа.

Наиболее очевидное преимущество виртуализации — более эффективное функционирование ядра MapReduce благодаря повышению масштабируемости и производительности. Задачи Map (сопоставление) и Reduce (свертка) должны выполняться независимо друг от друга. Распараллеливание ядра MapReduce и его настройка для работы в виртуальной среде позволяют снизить затраты на управление им, а также варьировать рабочие нагрузки при выполнении отдельных задач. MapReduce по своей сути является параллельной и распределенной платформой. Инкапсуляция ядра MapReduce в виртуальный контейнер дает возможность запускать необходимые процессы в удобное время. Таким образом, виртуализация повышает эффективность имеющегося оборудования и программного обеспечения, превращая их в универсальные пулы ресурсов.

Глава 6

Облачные сервисы и большие данные

В ЭТОЙ ГЛАВЕ:

Что такое облако
Облачные модели развертывания и взаимодействия
Для чего необходимо облако при работе с большими данными
Как применяется облако в системах обработки больших данных

Преимущество облака заключается в том, что пользователи могут получить доступ к необходимым вычислительным ресурсам и ресурсам хранения практически без информационно-технической поддержки или необходимости приобретать дополнительное аппаратное или программное обеспечение. Одной из основных характеристик облака является эластичная масштабируемость: пользователи могут добавлять или наращивать ресурсы в режиме, близком к реальному времени, в зависимости от изменяющихся требований. В мире больших данных облако играет очень важную роль: совместное использование облачных инфраструктурных компонентов с новыми методами управления данными дает кардинально новые возможности. Горизонтально расширяемая и оптимизированная инфраструктура является основой практической реализации больших данных.

В этой главе мы рассмотрим основы облачных технологий и их возможности применительно к работе с большими данными.

Облачные вычисления в контексте больших данных

Облачные вычисления — это способ обеспечения доступа к общему множеству вычислительных ресурсов, которые включают в себя приложения, вычисления, систему хранения, сети, разработки, платфор-

мы развертывания, а также бизнес-процессы. Облачные вычисления объединяют традиционные вычислительные ресурсы в совместно используемые пулы ресурсов на основе Интернета. В рамках облачных вычислений в качестве услуги может быть предоставлено все, от вычислительной мощности до вычислительной инфраструктуры, от приложений и бизнес-процессов до данных и аналитики. Чтобы это работало на практике, облако должно быть реализовано с помощью общих стандартизированных процессов и механизмов функционирования.



Многие компании используют облачные сервисы во всех аспектах деятельности, начиная с резервного копирования и заканчивая возможностями программного обеспечения как услуги (SaaS) (например, системы управления взаимоотношениями с клиентами (CRM). С распространением мобильных устройств все больше потребителей, специалистов и организаций создают и используют данные посредством облачных сервисов. Клиент магазина получает онлайн-купон на скидку, специалист службы контроля качества собирает и анализирует данные со множества различных датчиков на заводе. Все это строится на облачной инфраструктуре сервисов данных.

Яркий пример эффективной работы с большими данными в облаке — опыт Google и Amazon.com. Обеим компаниям необходимо управлять огромными объемами данных, а для этого нужны соответствующие инфраструктуры и технологии поддержки приложений. Достаточно представить себе почтовый сервис Gmail компании Google, в рамках которого ежедневно обрабатываются миллионы сообщений. Google удалось оптимизировать операционную систему Linux и ее программную среду для наиболее эффективной поддержки электронной почты и таким образом создать удобные условия для работы сотен миллионов пользователей. Что еще более важно, Google имеет возможность собирать и использовать огромное количество данных о пользователях своей электронной почты и поисковой системы.

Инфраструктура компании Amazon.com (включающая центры хранения и обработки данных IaaS, инфраструктуры как услуги) оптимизирована в соответствии с огромными рабочими нагрузками, что позволяет компании предлагать новые сервисы и поддерживать растущее количество заказчиков без значительных издержек. Чтобы наращивать объемы розничных продаж, Amazon необходима возможность управлять данными о товарах, покупателях, а также о каналах сбыта партнерских компаний. Успех компании в значительной степени зависит от целевой (таргетированной) рекламы, основанной на моделях покупательского поведения клиентов. В настоящее время Amazon.com и Google предлагают широкий спектр облачных услуг для больших данных. Они будут рассмотрены ниже.

Облачные модели развертывания и взаимодействия

Общедоступные и частные облака — две основные облачные модели, которые следует рассмотреть применительно к большим данным. Большинство организаций, которые внедряют облачные модели развертывания и взаимодействия, комбинируют использование частных вычислительных ресурсов (центров обработки данных, частных облаков) и общедоступных сервисов (управляемых сторонними компаниями, которые за плату предоставляют мощности для общего их использования различными клиентами). Выбор соотношения между общедоступными и частными сервисами зависит от ряда факторов, таких как конфиденциальность, допустимые задержки, цели использования данных. Необходимо иметь общее представление об облачных средах и их возможностях применительно к большим данным. Например, можно вести работу с большими данными в общедоступном облаке IaaS (см. ниже) или продолжать хранить все данные локально, а можно сочетать оба подхода. Для начала рассмотрим разные модели развертывания и взаимодействия, затем подробнее поговорим об их значении для больших данных.

Модели развертывания облачных сервисов

Модели развертывания облачных вычислений делятся на общедоступные и частные. В отличие от моделей взаимодействия (см. ниже), ориентированных на специфические потребности, модели развертывания предназначены для стандартных вычислительных задач. Рассмотрим особенности и возможности применения общедоступных и частных облачных моделей.

Общедоступное облако

Общедоступное облако — это совокупность принадлежащих сторонней организации и эксплуатируемых ею аппаратных средств, сетей, систем хранения, сервисов, приложений и интерфейсов, которые предназначены для использования другими компаниями и частными лицами. Коммерческие поставщики общедоступных облаков создают центры обработки данных с высокой масштабируемостью, где детали лежащей в их основе инфраструктуры скрыты от потребителя. Общедоступные облака конкурентоспособны, поскольку, как правило, управляют повторяющимися или сравнительно простыми рабочими

нагрузками. К примеру, электронная почта является очень простым приложением, следовательно, облачный провайдер может оптимизировать среду для работы с большим числом клиентов даже при условии хранения большого количества сообщений.

Поставщики общедоступных облаков, предлагающие сервисы хранения данных или вычислительные услуги, также оптимизируют свое вычислительное оборудование и программное обеспечение, чтобы поддерживать специфические типы рабочих нагрузок.

Напротив, типичный центр обработки данных обеспечивает работу очень большого количества различных приложений и рабочих нагрузок, поэтому труднее поддается оптимизации. Общедоступное облако целесообразно использовать в случае, если организация выполняет сложный проект анализа данных и нуждается в дополнительной вычислительной мощности, или, к примеру, для хранения больших объемов информации, когда облачное хранение обходится гораздо дешевле, чем приобретение в собственность крупного хранилища данных. Главные проблемы общедоступных облаков применительно к большим данным — требования к безопасности и допустимое время задержки.

Общедоступные облака очень разнообразны. Одни обеспечивают высокий уровень безопасности, управления услугами и масштабируемости.

Другие менее надежны и менее безопасны, но и менее затратны в использовании. В конечном счете выбор определяется характером проектов, задействующих большие данные, и допустимой степенью риска.

Частное облако

Частное облако — это совокупность аппаратных средств, сетей, систем хранения, сервисов, приложений и интерфейсов, предназначенных для использования сотрудниками, партнерами и клиентами организации, которой они принадлежат. Даже если частное облако создано и управляется сторонней организацией, его использование все равно будет ограничено одним предприятием-заказчиком. Частное облако — это тщательно контролируемая среда, закрытая для посторонних пользователей. Соответственно, оно отделяется межсетевым экраном. Частное облако отличается высокой степенью автоматизации, особенно в области управления, безопасности и соответствия техническим требованиям. Автоматизация заменяет большую часть ручных про-

цессов управления ИТ-сервисами в рамках поддержки клиентов, поэтому бизнес-правила и процессы могут быть реализованы внутри программного обеспечения таким образом, чтобы среда была более прогнозируемой и управляемой. Частное облако — отличное решение для реализации проекта больших данных, связанного с обработкой больших объемов информации, поскольку оно обеспечивает оптимальную безопасность и приемлемое время задержки.



Гибридное облако — это частное облако, которое обращается к общедоступным облачным сервисам и имеет с ними одну или несколько точек соприкосновения. В результате создается хорошо управляемая облачная среда, которая позволяет использовать преимущества различных моделей и объединять данные из облаков разного типа.

Модели взаимодействия с облаками

Помимо модели развертывания облачные сервисы различаются также по способам взаимодействия с ними. Рассмотрим четыре наиболее популярные модели взаимодействия.

Инфраструктура как услуга

Инфраструктура как услуга (IaaS) — одна из самых простых облачных вычислительных услуг. IaaS подразумевает предоставление вычислительных услуг (включая аппаратные средства, сети, хранение данных и место в центре хранения и обработки данных) напрокат. Потребитель услуги получает ресурсы и оплачивает их в соответствии с объемом и продолжительностью их использования. Существуют как общедоступные, так и частные модели IaaS. Общедоступные модели обычно оплачиваются банковской картой; как только оплата прекращается, прекращается и предоставление ресурса. Частные модели IaaS, как правило, используют специально созданную ИТ-организацией или интегратором инфраструктуру для предоставления ресурсов по требованию внутренним пользователям, а иногда и внешним партнерам.

Платформа как услуга

Платформа как услуга (PaaS) — это механизм для объединения IaaS с абстрактным набором промежуточных межплатформенных сервисов, программного обеспечения и инструментов, который дает возможность создавать и развертывать приложения как в облаке, так

и локально. PaaS предлагает совместимый набор программного обеспечения или промежуточных межплатформенных сервисов, позволяющих разработчикам создавать приложения в облачной среде. Среда PaaS сводит воедино процессы разработки и развертывания, чтобы более эффективно управлять созданием, развертыванием и масштабированием приложений. Для реализации модели PaaS требуется модель IaaS.

Программное обеспечение как услуга

Программное обеспечение как услуга (SaaS) — это бизнес-приложение, созданное и организованное поставщиком в рамках мультитенантной модели. Мультитенантность означает, что запущенный в облачной среде один экземпляр приложения одновременно обслуживает разные организации (тенанты), но при этом их данные отделены друг от друга. Клиенты платят за услугу помесячно или раз в год в расчете на одного пользователя. SaaS устанавливается поверх PaaS и IaaS.

Данные как услуга

Данные как услуга (DaaS) — модель, тесно связанная с SaaS. Это кроссплатформенный сервис, который позволяет подключаться к облаку для хранения и извлечения данных, а также предоставляет ряд специализированных услуг передачи данных, которые чрезвычайно полезны для работы в среде больших данных. К примеру, Google предлагает сервис, способный обработать по запросу 5 терабайт данных за 15 секунд. Обычный центр обработки данных затратил бы на такую задачу в десять раз больше времени. Такие компании, как IBM, разрабатывают сотни специализированных аналитических сервисов DaaS.

Почему облако необходимо для работы с большими данными

Комбинации моделей развертывания и моделей взаимодействия создают множество различных решений, которые могут применяться для работы с большими данными в облаке. Например, модель IaaS реализуется и в частных, и в общедоступных облаках. Но в чем ценность этих моделей применительно к большим данным и почему именно облачные системы идеально подходят для работы с большими данными? Дело в том, что работа с большими данными предполагает использо-

вание распределенных кластеров вычислительной мощности (см. главу 3) — то есть именно тот подход, который применяется в архитектуре облака.

Важная роль облачных решений в экосистеме больших данных обусловлена рядом факторов.

- Масштабируемость. Масштабируемость оборудования означает способность обеспечивать как высокую, так и низкую вычислительную мощность при неизменной архитектуре. Применительно к программному обеспечению масштабируемость означает стабильность уровня производительности на единицу вычислительной мощности при наращивании количества аппаратных ресурсов. Возможности облака к масштабированию огромны; распределенные вычисления, будучи неотъемлемой частью облачной модели, работают по принципу «разделяй и властвуй». Таким образом, при работе с большими объемами данных их можно разделить между несколькими облачными серверами. Важной характеристикой IaaS является возможность динамического масштабирования: если в процессе работы потребность в ресурсах превысит запланированную, легко получить дополнительные ресурсы. Эта возможность напрямую связана с понятием эластичности.
- Эластичность. Эластичность — это способность вычислительного ресурса увеличиваться или уменьшаться в реальном времени в зависимости от потребностей. Одним из преимуществ облака является возможность для клиента получить услугу именно в том объеме, который требуется, в нужный момент. Это особенно важно в работе с большими данными, когда количество данных и их скорость могут непредсказуемо возрасти и потребовать увеличения объема вычислительных ресурсов. Однако эта особенность облака, выгодная для конечного пользователя, обязывает поставщика услуг обеспечить оптимизацию архитектуры платформы для такого типа сервиса.
- Создание пула ресурсов. Облачные архитектуры позволяют эффективно создавать группы совместно используемых ресурсов, благодаря чему облако является экономически целесообразным решением.
- Самообслуживание. В отличие от традиционного процесса получения ресурсов в центре хранения и обработки данных, когда их необходимо запрашивать у ИТ-подразделения, в облачной системе пользователь через браузер или интерфейс портала самостоятель-

но получает ресурсы, необходимые, например, для запуска огромной прогностической модели.

- Низкие начальные затраты. Использование облачного поставщика позволяет сэкономить на приобретении оборудования или аренде дискового пространства, необходимых для работы с большими данными. Учитывая экономические преимущества масштабируемости, облачная среда может представляться идеальным решением, однако предварительно нужно проанализировать ситуацию, чтобы понять, использовать ли общедоступное, частное или гибридное облако либо сделать выбор в пользу других технологий (см. далее «На что обращать внимание при использовании облачных сервисов»).
- Оплата по факту использования. Как правило, услуги облачного провайдера оплачиваются по факту использования (PAYG), то есть по установленному тарифу исходя из фактически использованного объема ресурсов. Это целесообразно, например, если в начале проекта трудно оценить, какое количество ресурсов потребуется (при условии, что в бюджет проекта заложено достаточно средств на обработку данных).
- Отказоустойчивость. Облачные сервисы обладают отказоустойчивой архитектурой, обеспечивающей бесперебойное предоставление услуг при возможном отказе одного или нескольких элементов системы.



Поставщик услуг не всегда может удовлетворить потребности клиента собственными силами и в некоторых случаях обеспечивает требуемые мощности, привлекая сторонних поставщиков услуг. Как правило, клиент при этом не знает, что имеет дело с облачными услугами от нескольких разных поставщиков.

Как использовать облако для работы с большими данными

Очевидно, что облако по самой своей природе является идеальной вычислительной средой для работы с большими данными. Каким же образом можно использовать большие данные вместе с облаком? Вот несколько примеров.

- IaaS в общедоступном облаке. Чтобы не создавать собственную физическую инфраструктуру, компания развертывает сервисы больших данных в инфраструктуре провайдера общедоступного облака. IaaS позволяет создавать виртуальные машины с почти безграничными возможностями системы хранения и неограниченной вычислительной мощностью, поддерживающие ряд операционных систем и динамическое масштабирование среды. Яркий пример — использование сервиса Amazon EC2 (эластичное вычислительное облако Amazon, см. ниже) для запуска модели предсказания в реальном времени, которая требует массовой параллельной обработки данных. Такая модель может обрабатывать данные розничной торговли в гипермаркетах или анализировать миллионы элементов данных о посещениях веб-сайтов и переходах по ссылкам, чтобы сделать целевую рекламу более эффективной.
- PaaS в частном облаке. PaaS — цельная инфраструктура, функционал которой можно применять для проектирования, внедрения и развертывания приложений и сервисов в общедоступной или частной облачной среде. PaaS позволяет использовать ключевые промежуточные сервисы без необходимости управления отдельными аппаратными и программными элементами. Поставщики PaaS начинают внедрять в свои решения технологии больших данных, такие как Hadoop и MapReduce. К примеру, требуется создать специализированное приложение для анализа большого объема медицинских данных, которое будет работать как с данными, поступающими в режиме реального времени, так и ранее собранными. Для хранения и обработки таких данных необходимо будет использовать Hadoop и MapReduce. PaaS в этом случае дает огромное преимущество: возможность быстро развернуть приложение, не дожидаясь, пока корпоративное ИТ-подразделение полностью освоит новые технологии — достаточно, чтобы оно готово было осуществлять поддержку приложения. Использование PaaS открывает широкие возможности для экспериментов.
- SaaS в гибридном облаке. Это решение может использоваться для анализа «народного мнения» из разных источников. Многие компании справедливо полагают, что высказывания клиентов о продукции и услугах содержат ценнейшую информацию и позволяют сделать важные аналитические выводы касательно потребительского поведения. Клиенты все чаще оставляют отзывы и выражают свое мнение на общедоступных сайтах в Интернете. Эти данные

могут быть чрезвычайно полезны для бизнеса, если их надлежащим образом проанализировать. Поставщик SaaS способен предоставить и платформу для анализа, и данные из социальных сетей, и возможность включить в анализ данные системы управления взаимоотношениями с клиентами из частной облачной среды организации.



Некоторые профильные участники рынка, например Amazon.com и LinkedIn, относят приложения, работающие в облаке с большими данными, к приложениям больших данных. Другие считают, что применять этот термин по отношению к облачным приложениям некорректно, поскольку они являются SaaS-приложениями, ориентированными на конкретные бизнес-задачи. Как и в любой динамично развивающейся отрасли, терминология пока не устоялась.

Поставщики услуг на рынке облаков для больших данных

Участники рынка облачных сервисов чрезвычайно разнообразны и предлагают множество различных продуктов. Среди них и широко известные бренды, и новые компании. В числе облачных провайдеров, предлагающих услуги IaaS, которые возможно использовать для больших данных, стоит отметить Amazon.com, AT&T, GoGrid, Joyent, Rackspace, IBM и Verizon/Terremark.

При этом облачные компании и поставщики облачных услуг предлагают и специфическое программное обеспечение, предназначенное специально для работы с большими данными. Оно описано ниже в данной главе.

Публичное эластичное вычислительное облако Amazon

В настоящее время одним из наиболее известных поставщиков услуг IaaS является инфраструктура платформ облачных веб-сервисов Amazon Web Services с эластичным вычислительным облаком Amazon EC2. Первоначально компания Amazon не стремилась на рынок сервисов больших инфраструктур, а фокусировалась на создании массивной инфраструктуры для поддержки собственного розничного бизнеса. Однако, обнаружив, что ресурсы этой инфраструктуры используют-

ся лишь частично, компания решила задействовать их в иной форме. Сервис Amazon EC2 был запущен в 2006 году и продолжает активно развиваться.

В Amazon EC2 масштабируемость находится под контролем пользователя; для оплаты использования ресурсов применяется почасовая тарификация. Показательно использование в названии сервиса термина «эластичный». Эластичность в этом случае — возможность для пользователей увеличивать или уменьшать объем ресурсов инфраструктуры, выделенный для удовлетворения их потребностей.

Компания Amazon также предлагает набор услуг Amazon Web Services и другие сервисы больших данных:

- Amazon Elastic MapReduce. Сервис предназначен для обработки огромных объемов данных. Веб-сервис Amazon Elastic MapReduce основан на платформе Hadoop (см. главу 9) и использует в работе EC2 и сервис хранения Amazon Simple Storage Service (Amazon S3). Пользователи могут запускать HBase (распределенное столбцово-ориентированное хранилище данных);
- Amazon DynamoDB. Полностью управляемый сервис баз данных NoSQL (см. главу 7). Отказоустойчивый, высоконадежный сервис баз данных с самообслуживанием, не видимой для конечного пользователя масштабируемостью и простым администрированием. Для большей надежности и производительности сервис реализован на SSD-накопителях (твердотельных дисках);
- Amazon Simple Storage Service (S3). Веб-сервис облачного уровня, предназначенный для хранения любого объема данных. Основной упор сделан на производительность и масштабируемость, поэтому сервис не имеет такого дополнительного функционала, как другие хранилища данных. Данные хранятся в так называемых «контейнерах»; пользователь может выбрать, в каких регионах мира физически будут храниться данные, чтобы сократить задержки при их использовании и соблюсти нормативные требования;
- Amazon High Performance Computing. Сервис для решения специализированных задач, предоставляющий доступ к кластерам высокопроизводительных вычислений с низкой задержкой. Такие сервисы преимущественно используются в научной среде, однако в настоящее время распространяются и в коммерческих сферах благодаря появлению предложений от Amazon и других поставщиков высокопроизводительных вычислений. Кластеры высоко-

производительных вычислений Amazon предназначены для специфических рабочих нагрузок и легко поддаются конфигурации в зависимости от выполняемых задач;
- Amazon RedShift. Сервис хранилищ данных на масштабируемой архитектуре MPP, рассчитанный на объемы от петабайта. Этот сервис Amazon является безопасной и надежной альтернативой локальному хранилищу данных и при этом совместим со многими популярными инструментами бизнес-аналитики.

Сервисы больших данных Google

Компания Google также предлагает ряд облачных услуг для работы с большими данными.
- Google Compute Engine. Сервис виртуальных вычислительных машин, основанный на облачных технологиях. Compute Engine представляет собой безопасную и гибкую вычислительную среду, располагающуюся в энергоэффективных центрах обработки данных. Google также предлагает решения по управлению рабочими нагрузками от нескольких технологических партнеров, которые оптимизировали свои продукты для использования с Google Compute Engine.
- Google Big Query. Средство осуществления запросов типа SQL к большим объемам данных (потенциально до миллиардов строк) на высокой скорости. Недостаток Big Query в том, что данные после помещения в запрос не могут быть изменены. Можно рассматривать это решение как разновидность системы оперативной аналитической обработки (OLAP) больших данных. Big Query подходит для автоматизированной системы отчетности или поискового анализа.
- Google Prediction API. Основанный на облачных технологиях самообучаемый инструмент прогнозирования, предназначенный для обработки огромных объемов данных. Prediction API выявляет закономерности в данных, запоминает их и при каждом последующем использовании может обнаружить новую информацию о выявленных ранее закономерностях. Анализ закономерностей позволяет, в частности, выявлять мошеннические действия, исследовать мнения потребителей и процессы оттока клиентов. Подробнее о прогнозировании рассказывается в главе 12.

Microsoft Azure

Компания Microsoft в рамках единого PaaS-решения выпустила на рынок продукт, основанный на абстракции Windows и SQL, который включает в себя набор инструментов для разработчика, поддержку виртуальной машины, сервисы управления и медиасервисы, а также сервисы для мобильных устройств. PaaS-сервис Microsoft на базе Azure становится очевидным выбором для клиентов, имеющих опыт работы с.Net, SQLServer и Windows.

Для удовлетворения возрастающих потребностей интеграции больших данных в решения на базе Windows Azure, Microsoft также разработан сервис Windows Azure HDInsight. Это средство на платформе Hortonworks Data Platform (HDP), которая, согласно заявлению Microsoft, на 100% совместима с Apache Hadoop. HDInsight поддерживает работу Microsoft Excel и других инструментов обработки информации. Помимо Azure, HDInsight может также быть развернут на операционной системе Windows Server.

OpenStack

Созданный по инициативе компаний Rackspace и NASA комплекс проектов свободного программного обеспечения OpenStack (www.openstack.org) реализует открытую облачную платформу, предназначенную как для общедоступных, так и для частных облаков. Компания OpenStack Foundation контролируется Rackspace, но формально от нее отделена. Несмотря на то что OpenStack допускает создание проприетарных реализаций, предполагается, что пользователи этого сервиса придерживаются стандартной его реализации.

Задача OpenStack — предоставить широкомасштабную мультитенантную облачную спецификацию, совместимую с любым оборудованием. OpenStack создает большую экосистему партнеров, заинтересованных в адаптации данной облачной платформы. Это Dell, HP, Intel, Cisco, Red Hat, IBM и еще более 100 компаний, которые используют OpenStack в качестве основы для своих облачных решений. В сущности Openstack является IaaS-инициативой с открытым исходным кодом на базе Ubuntu — операционной системы на основе дистрибутива Debian Linux. OpenStack также может работать и под управлением операционной системы Red Hat Linux.

OpenStack предлагает широкий спектр услуг, среди которых вычисления, хранение объектов, поддержка каталогов и хранилищ (ре-

позитарии), компактное отображение информации, идентификация и сетевое взаимодействие. Применительно же к большим данным Rackspace запускает общедоступный облачный сервис OpenStack на основе Hadoop, сертифицированный и поддерживаемый компанией Hortonworks (поставщиком платформы управления данными с открытым исходным кодом на основе Apache Hadoop), для быстрого создания сред больших данных.

На что обращать внимание при использовании облачных сервисов

Сервисы на базе облачных вычислений предлагают экономически выгодное решение ваших задач, связанных с большими данными. Однако использование облака может повлечь за собой и проблемы, поэтому, прежде чем переносить большие данные в облако, необходимо учесть ряд аспектов.

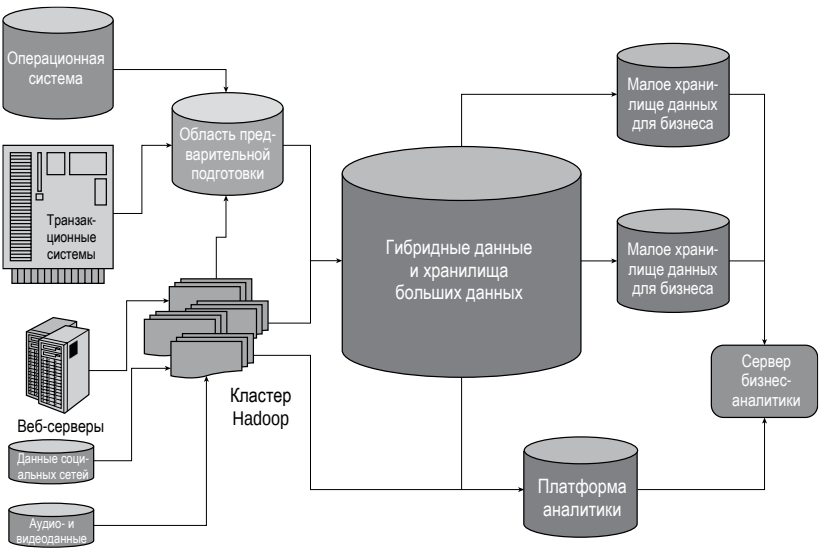
- Целостность данных. Использует ли поставщик облачных услуг все необходимые средства контроля, чтобы гарантировать целостность данных?
- Соответствие. Удовлетворяет ли выбранный поставщик всем нормативным требованиям компании или отрасли?
- Затраты. Небольшие затраты, накапливаясь, могут вылиться в крупную сумму. Внимательно ознакомьтесь со всеми условиями договора. Соответствуют ли они вашим потребностям в облачных услугах?
- Передача данных. Каким способом вы будете загружать данные в облако? Одни поставщики принимают данные на физических носителях, другие требуют загружать их исключительно через сеть, что может обойтись довольно недешево.
- Производительность. Прописаны ли в соглашениях об уровне обслуживания (SLA) все требования к доступности, поддержке, производительности? К примеру, если поставщик услуг утверждает, что данные доступны пользователям 99,999% времени, не поленитесь прочитать документ: возможно, эта величина включает также периоды планового технического обслуживания.
- Доступ к данным. Как реализована защита от несанкционированного доступа к данным? Иными словами, какие средства контроля гарантируют, что посторонние лица не получат доступ к данным?

Защита может включать механизмы идентификации, в таком случае необходимо позаботиться о безопасности личной идентификационной информации, чтобы доступ к вычислительным ресурсам, приложениям, данным и услугам находился под должным контролем.

- Местоположение. Где физически будут храниться данные? Законодательство некоторых стран или корпоративные стандарты могут запрещать хранение и обработку данных на устройствах, расположенных за рубежом.

Часть III

Управление большими данными



Потоки данных в хранилищах и витринах больших данных

В ЭТОЙ ЧАСТИ:

- Различия между базами данных в мире больших данных
- Использование MapReduce для анализа больших данных
- Основы Hadoop
- Повышение эффективности распределенной файловой системы Hadoop с использованием языков программирования и программных инструментов
- Разработка приложений больших данных для аналитики
- Использование программно-аппаратных комплексов при управлении большими данными

Глава 7

Операционные базы данных

В ЭТОЙ ГЛАВЕ:

Обзор реляционных баз данных
Нереляционные базы данных и базы данных пар «ключ-значение»
Документальные и столбцовые базы данных
Графовые и пространственные базы данных
Полиглотная сохраняемость

Большие данные играют все более важную роль в решении задач, связанных с использованием больших объемов информации. Но большие данные не пребывают в изоляции. Для эффективной работы, как правило, требуется объединять результаты анализа больших данных с данными о текущем состоянии бизнеса; другими словами, большие данные невозможно рассматривать отдельно от источников операционных данных. Для работы с операционными данными существует целый ряд служб. В этой главе мы рассмотрим, как такие данные используются совместно с большими данными.

Одна из главных задач операционных БД — обеспечение сохраняемости данных. Сохраняемость гарантирует, что данные, находящиеся в базе, изменяются только с соответствующего разрешения, а их доступность будет обеспечена в течение всего периода, когда в них есть необходимость. Какой смысл работать с базой данных, если она не защищает помещенные в нее данные? Это первое требование, которое нужно учесть при выборе БД. Также необходимо определить, какие виды данных будут сохраняться, как будет осуществляться их обновление и доступ к ним и как эти данные будут использоваться для принятия решений. От выбора ядра базы данных, который делается на этом уровне, в конечном счете зависит эффективность работы с большими данными.

Предшественником средств, обеспечивающих сохраняемость данных, является система управления реляционными базами данных, или реляционная СУБД. На ранних этапах развития вычислительной техники использовались методы обеспечения сохраняемости данных, которые сейчас считаются примитивными. По сути, это были системы ведения записей. В них компании хранили любую информацию: от данных о сделках с клиентами до сведений о функционировании бизнеса. Несмотря на устаревшие технологии, многие подобные системы до сих пор остаются в употреблении, так как компании продолжают активно пользоваться хранящимися в них данными; заменить такую систему на новую не легче, чем поменять двигатель на летящем самолете. Примерно до 1980-х годов были распространены «плоские файлы» и «сетевые» хранилища данных — функциональные, но трудные в управлении средства, применение которых требовало от системных программистов разработки специальных программ для каждой конкретной задачи. Реляционная модель по-прежнему широко используется и играет важную роль в развитии больших данных.

Реляционная база данных построена на одном или нескольких отношениях и представлена в виде таблиц. Столбцы таблиц служат определителями, а в строках хранятся данные. Первичным ключом часто является первый столбец. Непротиворечивость базы данных достигается за счет «нормализации» данных, то есть их преобразования из исходного формата в общий для всей базы. Например, в одной БД телефонные номера записываются в виде XXX–XXX–XXXX, а в другой — XXXXXXXXXX. Чтобы с данными можно было работать, их необходимо привести к единообразию — нормализовать значения полей. Существует пять уровней стандартов нормализации. Какой формат будет считаться «нормой», обычно определяет проектировщик базы данных, чтобы упростить работу конечным пользователям. Коллекция таблиц, ключей, элементов и других составляющих называется схемой базы данных.

На протяжении многих лет вместе с технологией реляционных СУБД развивался и язык структурированных запросов (SQL). На сегодняшний день SQL — наиболее широко используемый механизм создания, поддержки и использования реляционных баз данных. Этот круг задач в англоязычной практике описывают аббревиатурой CRUD (Create, Retrieve, Update, Delete — создание, получение, обновление и удаление). Эти взаимосвязанные операции осуществляются напря-

мую или через программный интерфейс API. Язык SQL первоначально был ориентирован на реляционные СУБД, но нашел широкое применение и в работе с нереляционными базами данных. Подробнее об этом рассказано ниже.

Эволюция реляционных баз данных

Развитие реляционных баз данных постоянно сопровождалось разработкой технологий, предназначенных для устранения недостатков ранних реализаций реляционных СУБД. Так появились объектные базы данных, базы данных контента, хранилища данных, витрины данных и т.д. Чтобы применять эти технологии, каждой компании приходилось создавать собственные решения и интегрировать их с существующими приложениями реляционных СУБД. Это требовало больших затрат времени и средств, а результат не всегда оправдывал ожидания. Со временем эти новые технологии органично встроились в реляционные СУБД и стали неотъемлемой частью функционала доступных на рынке продуктов. Это устранило необходимость в разработке дополнительных, теперь уже избыточных решений. Вероятнее всего, так произойдет и с большими данными. Учитывая фундаментальные различия между большими данными и традиционными решениями для работы с данными, можно предположить, что встраивание технологий больших данных в реляционные СУБД займет несколько лет. А вот большие данные, со своей стороны, уже начали использовать язык SQL и другие традиционные для реляционных СУБД функции. Примером тому является MapReduce.

Значение реляционных СУБД в среде больших данных

В большинстве компаний основной объем важных операционных данных, как правило, хранится в реляционных СУБД. Многие компании используют несколько разных реляционных СУБД для различных направлений деятельности. Например, поставщик хранит данные о сделках в одной базе данных, а данные о клиентах — в другой. Понимать, где и какие данные хранятся, чрезвычайно важно при внедрении больших данных. Маловероятно, что в качестве основы для внедрения больших данных вы станете использовать реляционные СУБД, однако данные, хранящиеся в таких базах, потребуются для эффективного применения больших данных. На рынке представлено множество коммерческих решений, в том числе от таких компаний, как Oracle, IBM и Microsoft, но прежде всего необходимо упомянуть о реляционной базе данных с открытым исходным кодом под названием PostgreSQL.

Реляционная база данных PostgreSQL

PostgreSQL (www.postgresql.org) — это самая распространенная реляционная база данных с открытым исходным кодом. Она была разработана специалистами Университета Калифорнии в Беркли более полутора десятков лет назад и продолжает активно развиваться по сей день. Популярность PostgreSQL обусловлена несколькими факторами. Как реляционная СУБД с поддержкой стандарта SQL, она имеет все стандартные функции для работы с базами данных. Как продукт, известный уже много лет и широко распространенный, PostgreSQL считается надежным и проверенным решением. Наконец, эта СУБД работает практически со всеми операционными системами, от персональных компьютеров и до мейнфреймов.

Но ее преимущества не ограничиваются лишь надежностью и наличием необходимого функционала. PostgreSQL также поддерживает многие функции, доступные только в дорогих коммерческих реляционных СУБД:

- возможность прямой обработки объектов в реляционной схеме;
- внешние ключи (ключи ссылок между таблицами);
- триггеры (хранимые процедуры, которые автоматически запускаются при определенных действиях с базой);
- сложные запросы (подзапросы и вложенные табличные подзапросы);
- целостность транзакций;
- управление конкурентным доступом с помощью многоверсионности.

Самой сильной стороной PostgreSQL является ее расширяемость. Пользователи и программисты баз данных могут добавлять новые возможности, не затрагивая базовые операции и не влияя на надежность базы данных. Расширения применимы в том числе к следующим объектам:

- типы данных;
- операторы;
- функции;
- методы индексации;
- языки процедур.

Широкие возможности адаптации к решаемым задачам позволяют эффективно использовать PostgreSQL там, где жестко определенный функционал коммерческих продуктов зачастую неспособен обеспе-

чить требуемую гибкость. Возможности расширения PostgreSQL практически безграничны.

Наконец, лицензия PostgreSQL дает право на изменение и распространение продукта в любой форме — как в открытом, так и в закрытом исходном коде. Каждый разработчик решает сам, делать ли свои модификации общедоступными.

Хотя реляционные базы данных, и в частности PostgreSQL, играют ключевую роль в «корпоративных» больших данных, существует и ряд альтернативных подходов.

Нереляционные базы данных

Нереляционные базы данных, в отличие от реляционных, не используют модели «таблица-ключ». Каждая из описанных ниже технологий нереляционных баз данных обладает уникальными возможностями для решения определенных задач, которые недоступны традиционным реляционным СУБД. Для решения специализированных задач больших данных также потребуются особые методы их обработки, и здесь могут быть полезны нереляционные базы данных — хотя универсальными средствами они не являются.

Широко распространенный новый класс нереляционных баз данных известен как NoSQL. Это название расшифровывается как «not only SQL» («не только SQL»), хотя может быть прочитано как «не SQL». Это напрямую связано с историей этой технологии. Первоначально разработчики NoSQL предполагали создать СУБД, которая не использовала бы реляционную модель и язык запросов SQL, но к тому времени, как продукт вышел на рынок, стало понятно, что полностью отказаться от SQL не удастся, и название класса стали расшифровывать как «не только SQL», относя к нему базы данных, которые не поддерживают реляционную модель, но все же используют SQL в качестве основного средства работы с данными. Реляционные и нереляционные базы данных имеют схожие фундаментальные основы, однако реализуют их по-разному. Для всех технологий нереляционных баз данных характерны следующие общие черты.

- Масштабируемость. В этом контексте масштабируемость понимается как возможность записывать данные в несколько хранилищ одновременно, невзирая на физические ограничения базовой инфраструктуры. Важным фактором здесь является бесшовность: ба-

зы данных должны масштабироваться в зависимости от объемов потоков данных незаметно для пользователей.

- Данные и модель запросов. Вместо строк, столбцов и ключей нереляционные базы данных используют специальные структуры для хранения данных. Эти структуры имеют обязательный набор специальных API-запросов, которые обеспечивают интеллектуальный доступ к данным.
- Проектирование сохраняемости. Сохраняемость остается критически важным элементом и в нереляционных базах данных. В связи с высокой скоростью передачи, разнообразием и объемом больших данных такие базы данных задействуют механизмы дифференциации для сохранения данных. Наиболее производительным вариантом является хранение «в памяти», когда вся база данных находится в очень быстрой системе памяти серверов.
- Разнообразие интерфейсов. Хотя большинство этих технологий поддерживает интерфейсы RESTful API в качестве интерфейса доступа, они также предоставляют широкий набор механизмов подключения, доступных программистам и администраторам баз данных, в том числе средства анализа, создания отчетов и визуализации.
- Модель «согласованность в конечном счете». В реляционных СУБД согласованность данных определяется параметрами ACID (Atomicity, Consistency, Isolation, Durability — атомарность, согласованность, изолированность, долговечность). Факторы согласованности в нереляционных СУБД описываются моделью BASE (Basically Available, Soft state и Eventual Consistency — базовая доступность, гибкое состояние и согласованность в конечном счете). Из этих факторов наибольшую важность имеет согласованность, поскольку она обеспечивает разрешение конфликтов при перемещении данных между узлами распределенных систем. Состояние данных поддерживается программно, а модель обеспечения доступа использует базовую доступность.

Рассмотрим самые популярные классы и реализации нереляционных баз данных с открытым исходным кодом.

Базы данных пар «ключ-значение»

Самые простые базы данных NoSQL используют модель пар «ключ-значение» (key-value pair, KVP). В отличие от реляционных СУБД, базы данных пар «ключ-значение» не нуждаются в схеме и обеспечивают более высокую масштабируемость и гибкость. Эти базы данных не удовлетворяют требованиям ACID (т.е. атомарности, согласованности, изолированности и долговечности), поэтому при их внедрении необходимо применять дополнительные решения для размещения данных, репликации и обеспечения отказоустойчивости. Базы данных пар «ключ-значение» не являются типизированными, поэтому большинство данных хранится в виде строк. В табл. 7.1 приведены некоторые примеры пар «ключ-значение».

Ключ	Значение
Цвет	Синий
Напиток	Пиво
Герой	Солдат

Таблица 7.1. Примеры пар «ключ-значение»

Это довольно упрощенный набор ключей и значений. При работе с большими данными у разных людей будут самые разные ассоциации с цветами, напитками и героями (табл. 7.2).

Ключ	Значение
ПользовательFacebook1234_Цвет	Красный
ПользовательTwitter67890_Цвет	Серо-буро-малиновый
ПользовательFoursquare45678_Напиток	«Белое вино»
ПользовательGoogle+24356_Напиток	«Водка без пива — деньги на ветер»
ПользовательLinkedIn87654_Герой	«Самый эффективный продавец»

Таблица 7.2. Пары «ключ-значение» в больших данных

С увеличением числа пользователей отслеживать ключи и связанные значения становится труднее. Если речь идет о миллионах пользователей, то количество связанных пар «ключ-значение» возрастает чрезвычайно быстро. Если не ограничивать варианты значений, то в целом представление пары «ключ-значение» строкой обеспечивает гибкость и удобочитаемость.

Чтобы упорядочить данные в базе пар «ключ-значение», можно использовать дополнительные решения. Как правило, они предоставляют возможность агрегации ключей (и связанных значений) в коллекцию. Коллекция может состоять из любого числа пар «ключ-значение» и не требует исключительного контроля за отдельными элементами пар.

База данных пар «ключ-значение» Riak

Одна из широко используемых баз данных пар «ключ-значение» с открытым исходным кодом называется Riak. Она разработана и поддерживается компанией Basho Technologies (www.basho.com), доступ предоставляется в рамках лицензии на программное обеспечение Apache Software License версии 2.0.

Riak — это очень быстрая и масштабируемая реализация базы данных пар «ключ-значение». Она не требует значительных ресурсов и поэтому применима для поддержания сред быстро меняющихся данных большого объема. База данных Riak особенно эффективна при анализе в режиме реального времени торгов на финансовых рынках и биржах. В качестве механизма упорядочивания коллекций ключей и значений используются так называемые контейнеры. Реализациями Riak являются кластеры физических или равноправных виртуальных узлов. Отсутствие главного узла обусловливает отказоустойчивость и хорошую масштабируемость. Все данные и операции распределены внутри кластера. Кластеры Riak отличаются интересным профилем производительности: большие кластеры (с большим количеством узлов) работают лучше и быстрее, чем кластеры с малым числом узлов. Связь в кластере обеспечивается специальным протоколом Gossip, который сохраняет сведения о состоянии кластера и предоставляет общие сведения о контейнерах.

Riak имеет много функциональных возможностей и является частью экосистемы, состоящей из перечисленных ниже компонентов.

- Параллельная обработка. С помощью технологии MapReduce база данных Riak поддерживает возможность разбивки и повторной компоновки запросов в масштабах всего кластера. Это обеспечивает вычисления и анализ в режиме реального времени.
- Ссылки и переход по ссылкам. Использование ссылок в Riak позволяет имитировать графовую базу данных. Ссылку можно рассматривать как связь между парами «ключ-значение». Описание совокупности таких ссылок позволяет построить карту отношений между парами «ключ-значение».
- Поиск. База данных Riak имеет функции распределенного и устойчивого полнотекстового поиска. Контейнеры можно индексировать для быстрого разделения их ключей по значениям.
- Вторичные индексы. Разработчики могут помечать хранимые значения одним или несколькими ключевыми значениями. После этого приложение может сформировать запрос к индексу и получить список совпадающих ключей. Это может оказаться очень полезным при работе с большими данными — такая операция является атомарной и поддерживает поведение реального времени.

Реализации базы данных Riak эффективно применяются в следующих сферах:

- пользовательские данные в социальных сетях, сообществах, играх;
- сбор и хранение мультимедийных данных большого объема;
- слои кэширования для подключения баз данных NoSQL и реляционных СУБД;
- мобильные приложения, требующие гибкости и функциональной надежности.

Документальные базы данных

Существует два вида документальных баз данных. Первый вид часто описывают как репозиторий для контента, полностью состоящего из документов (файлы Word, веб-страницы и т.д.). Второй вид — это база данных, предназначенная для сохранения компонентов документов в целях постоянного хранения в качестве статических записей или для использования таких частей в целях динамического включения в другие документы.

Структура документов и их частей обеспечивается форматом JSON (JavaScript Object Notation) и двоичным форматом Binary JSON (BSON). Документальные базы данных особенно полезны в тех случаях, когда нужно подготовить большое количество отчетов, для генерации которых используются часто меняющиеся элементы. Примером может служить заполнение медицинской документации, когда в стандартную форму требуется внести разноплановые данные о пациенте (пол, возраст, адрес и т.д.), медицинской страховке, льготах и т.п. Для работы с большими данными важны оба вида документальных баз данных.

JSON — это, по сути, формат обмена данными на основе подмножества языка программирования JavaScript. Хотя этот формат частично состоит из языка программирования, по существу он является текстовым, поэтому очень удобен для чтения и ввода данных. Кроме того, его преимуществом является простота обработки на компьютере. JSON характеризуется наличием двух базовых структур, которые поддерживаются если не всеми, то основными современными языками программирования. Первая базовая структура — это коллекции пар «имя-значение». Программно они могут быть представлены в виде объектов, записей, индексированных списков и т.п. Вторая базовая структура — это упорядоченные списки значений, которые могут быть представлены в виде массивов, списков и последовательностей. BSON — это двоичная реализация структур JSON, предназначенная для повышения производительности и масштабируемости.

Документальные базы данных становятся «золотым стандартом» внедрения больших данных. Рассмотрим две самые популярные реализации таких баз данных.

MongoDB

MongoDB (www.mongodb.com) — проект по созданию «гигантской базы данных», получивший свое название от английского слова «hu (mongo) us», что и означает «гигантский». Это проект с открытым исходным кодом, он бесплатно распространяется в рамках лицензии GNU AGPL версии 3.0. Также доступны коммерческие лицензии с полной технической поддержкой.

MongoDB набирает популярность; применительно к большим данным он может использоваться в качестве хранилища данных. MongoDB состоит из баз данных, содержащих «коллекции»; каждая

коллекция состоит из записей («документов»), а каждый документ — из полей. Как и в случае с реляционными базами данных, коллекцию можно индексировать для повышения эффективности поиска данных. Однако, в отличие от других баз данных, запрос к базе MongoDB возвращает так называемый «курсор», который указывает на данные. Это очень полезная возможность, которая позволяет считывать и классифицировать данные без их непосредственного извлечения. MongoDB естественным образом поддерживает BSON, двоичную реализацию документов JSON.

Экосистема MongoDB включает следующие компоненты:

- службы обеспечения доступности и репликации для масштабирования в локальных и глобальных сетях;
- кластерную файловую систему (GridFS), которая позволяет хранить крупные объекты путем их разделения на несколько документов;
- набор программных средств MapReduce, обеспечивающих аналитику и агрегацию различных коллекций и документов;
- службу сегментирования, которая распределяет одну базу данных по кластеру серверов одного или нескольких центров обработки данных. Эта служба работает на основе ключа сегментации. Ключ сегментации используется для интеллектуального распределения документов по нескольким экземплярам;
- службу запросов, которая поддерживает неструктурированные запросы, распределенные запросы и полнотекстовый поиск.

Сферы, где наиболее эффективно применение MongoDB:

- управление большими объемами контента;
- взаимодействие через социальные сети;
- архивация;
- аналитика в режиме реального времени.

База данных CouchDB

Еще одна популярная нереляционная база данных — CouchDB (http://couchdb.apache.org). Как и MongoDB, Couch DB имеет открытый исходный код. База данных поддерживается Фондом Apache Software Foundation (www.apache.org) и доступна в рамках лицензии Apache License v2.0. В отличие от MongoDB, база данных CouchDB об-

ладает свойствами, близкими к веб-серверу или веб-приложению. Например, CouchDB отличается отказоустойчивостью при сбоях в сети и беспрепятственно работает при нестабильном доступе к сети. Она одинаково хорошо функционирует и на смартфонах, и в центрах обработки данных. Однако имитация веб-серверов и приложений имеет и недостатки. CouchDB отличается высокой задержкой, поэтому с ней предпочтительно использовать локальное хранилище данных. Несмотря на возможность нераспределенной работы, CouchDB мало подходит для небольших проектов. Таким образом, надо решать в зависимости от ситуации, перевесят ли преимущества CouchDB ее недостатки при работе с большими данными.

Базы данных CouchDB состоят из документов, которые, в свою очередь, состоят из полей и вложений, а также «описания» документа в виде метаданных, автоматически поддерживаемых системой. Технологии базы данных CouchDB реализуют все механизмы обеспечения согласованности данных ACID. Преимущество CouchDB перед реляционными СУБД заключается в том, что данные уже упакованы и готовы к обработке или хранению, а не распределены по строкам и таблицам.

В состав экосистемы CouchDB входят следующие функциональные возможности.

- Сжатие. Базы данных имеют механизмы для устранения нерационально используемого пространства при достижении определенного уровня пустоты базы. Это повышает производительность и обеспечивает эффективную сохраняемость.
- Модель представления. Это механизм фильтрации, упорядочивания данных и составления отчетов, использующий набор определений, которые хранятся в базе данных в виде документов. В моделях представления баз данных существует связь «один-ко-многим», так что продольные и поперечные срезы данных можно представить в различных видах.
- Службы репликации и распределения. Хранилище документов поддерживает двунаправленную репликацию. Для обеспечения распределения на основе задаваемых критериев или для миграции на устройства с ограниченным подключением можно создавать частичные реплики. Механизмы репликации поддерживают иерархию узлов; обычно репликация осуществляется между одноранговыми узлами, но возможно организовать репликацию между ведущими и ведомыми узлами, только между ведущими узлами и др.

CouchDB эффективно применяется:

- для управления большими объемами контента;
- для масштабирования в широком диапазоне (от смартфонов до центров обработки данных);
- в приложениях с ограниченным или медленным подключением к сети.

Столбцовые базы данных

Реляционная модель предполагает представление данных в виде строк — то есть данные каждой табличной строки в БД хранятся вместе. В столбцовой, или столбцово-ориентированной, базе данных данные хранятся в столбцах. Казалось бы, это несущественное отличие, однако это важнейшая характеристика в основе столбцовых баз данных. В таких СУБД очень просто добавлять столбцы, но данные можно также вносить построчно, что обеспечивает высокую гибкость, производительность и масштабируемость. Столбцовую базу данных удобно использовать для работы с большим объемом разнообразных данных. Она легко адаптируется — достаточно добавлять столбцы.

Столбцовая база данных HBase

Одна из самых популярных столбцовых баз данных — HBase (http://hbase.apache.org). Она также является продуктом фонда Apache Software Foundation, распространяемым в рамках лицензии на программное обеспечение Apache License v2.0. В качестве основы хранилища данных в HBase используется файловая система Hadoop (см. главу 9) и набор программных средств MapReduce (см. главу 8).

В основе HBase лежит высокопроизводительная СУБД Google BigTable, обеспечивающая эффективные формы хранения нереляционных данных. Поэтому реализации HBase представляют собой высокомасштабируемые, разреженные, распределенные, постоянно упорядоченные многомерные карты. Карта индексирована по ключу строки, ключу столбца и отметке времени; каждое значение в карте представляет собой неинтерпретируемый байтовый массив. HBase станет подходящим решением, когда для использования больших данных потребуется прямой произвольный доступ к базе для чтения и записи в режиме реаль-

ного времени. Как правило, эта база данных используется для сохранения результатов с целью последующей аналитической обработки.

Основные характеристики HBase:

- Согласованность. База данных HBase не основывается на модели «согласованности в конечном счете» и не реализует механизм ACID, но при этом обеспечивает строго согласованные чтение и запись данных. Это означает, что ее можно применять при потребности в обработке данных на высоких скоростях, если нет необходимости в дополнительных функциональных возможностях реляционных СУБД, таких как полная поддержка транзакций или типизированные столбцы.
- Сегментация. Так как данные распределяются поддерживающей файловой системой, HBase обеспечивает автоматическое разделение и перегруппировку контента незаметно для пользователя.
- Высокая доступность. Развертывание региональных серверов HBase обеспечивает отказоустойчивость и восстановление на уровне локальных (LAN) и глобальных (WAN) сетей. Основой ядра системы является главный сервер, ответственный за мониторинг региональных серверов и метаданных всех кластеров.
- Клиентский API. HBase предоставляет программный доступ через интерфейс Java API.
- ИТ-поддержка в процессе эксплуатации. Встроенный веб-интерфейс позволяет контролировать производительность и другие показатели.

Реализации базы данных HBase подходят для решения следующих задач:

- сбор и обработка постоянно возрастающих больших объемов данных;
- информационный обмен в режиме реального времени (например, обмен сообщениями);
- обработка часто меняющегося контента.

Графовые базы данных

Фундаментальная структура графовых баз данных называется «узел-связь». Такая структура особенно эффективна для работы с сильно взаимосвязанными данными. Свойства узлов и связей описаны и хра-

нятся в виде пар «ключ-значение». Навигация по графовым БД осуществляется за счет использования описанных отношений. В реляционных СУБД такой тип хранения и навигации не применяется из-за жесткой структуры таблиц и невозможности следовать по подключениям между данными. Графовая база данных может использоваться, например, для управления географическими данными нефтеразведки, а также для моделирования и оптимизации сетей поставщиков телекоммуникационных услуг.

Графовая база данных Neo4J

Одна из наиболее распространенных графовых баз данных — Neo4J (www.neo4j.org). Это проект с открытым исходным кодом, который лицензируется в соответствии с универсальной общедоступной лицензией GNU версии 3.0. Поддерживаемая коммерческая версия предоставляется компанией Neo Technology как по коммерческой лицензии, так и по свободной лицензии GNU AGPL версии 3.0. Neo4J — транзакционная база данных, удовлетворяющая требованиям ACID и обеспечивающая высокую доступность за счет кластеризации. Свойства фундаментальной структуры «узел-связь» и интуитивно понятное описание взаимосвязей, близкое к схеме отношений между людьми в обществе, обеспечивают простоту моделирования баз, надежность и масштабируемость. Neo4J обладает большой гибкостью, поскольку не требует ни типизации данных, ни описания структуры данных (схемы).

Однако следствием такой гибкости является ряд ограничений. Узлы не могут ссылаться непосредственно сами на себя. Например, человек не может являться своим собственным отцом (одновременно «быть» узлом и «состоять в отношениях отцовства» с этим узлом), однако может «состоять в отношениях отцовства» с другими узлами. Однако в реальности встречаются ситуации, когда узел может и должен ссылаться сам на себя, и в этом случае графовая база данных будет неэффективна. Несмотря на исключительно хорошие возможности репликации, Neo4J может реплицировать только графы целиком, ограничивая при этом общий размер графа (приблизительно 34 млрд. узлов и 34 млрд. отношений).

Основные характеристики Neo4J:

- Интеграция с другими базами данных. Neo4J поддерживает управление транзакциями с возможностью отката, что позволяет взаимодействовать с неграфовыми хранилищами данных.

- Службы синхронизации. Neo4J поддерживает поведение на основе событий через шину событий. Также обеспечивается периодическая синхронизация, при которой в качестве главного узла синхронизации используется сам узел или реляционная СУБД, а также традиционная пакетная синхронизация.
- Отказоустойчивость. Neo4J поддерживает «холодное» (т.е. при неработающей базе данных) и «горячее» (при работающей базе данных) резервное копирование, а также режим кластеризации высокой доступности. Стандартные уведомления позволяют осуществлять интеграцию с существующими системами управления и администрирования.
- Язык запросов. Neo4J поддерживает декларативный язык Cypher, разработанный специально для запросов графовых данных и их компонентов. Команды Cypher в общих чертах основаны на синтаксисе SQL и предназначены для нерегламентированных запросов графовых данных.

Применение реализаций Neo4J:

- взаимодействие через социальные сети;
- классификация в сфере биологии или медицины;
- формирование динамичных сообществ по интересам или общим занятиям.

Пространственные базы данных

С пространственными данными мы встречаемся ежедневно, порой не отдавая себе в этом отчета: когда прокладываем маршрут с помощью GPS в автомобильном навигаторе или ищем с помощью поисковой системы итальянский ресторан в определенном районе, мы пользуемся приложениями, которые используют пространственные данные. Стандарты пространственных данных, такие как OpenGIS (географические информационные системы), появились благодаря усилиям Открытого геопространственного консорциума (OGC; www.opengeospatial.org).

Пространственные базы данных являются реализацией стандартов OGC, которые могут отвечать или не отвечать определенным потребностям бизнеса. Пространственная база данных необходима, когда требуется использовать несколько различных измерений данных. Например, для метеорологического исследования нужно сохранить

и проанализировать данные об урагане, такие как температура, скорость ветра и влажность, а затем построить на основе полученных результатов трехмерную модель.

В простейшей форме пространственные базы данных хранят данные о двухмерных, 2,5-мерных и трехмерных объектах. Двухмерные и трехмерные объекты всем хорошо знакомы, мы постоянно с ними взаимодействуем. Двухмерный объект характеризуется длиной и шириной. Трехмерный объект помимо длины и ширины имеет также высоту. Страница книги — двухмерный объект, книга — трехмерный. А что такое 2,5-мерные объекты? Это особый тип пространственных данных — двухмерные объекты, которые находятся на некотором постоянном удалении относительно третьего измерения; величина этого удаления и является таким дополнительным «половинным» измерением. Большинство 2,5-мерных пространственных баз данных содержит картографические сведения и часто называется геоинформационными системами (ГИС).

Атомарными элементами пространственных баз данных являются линии, точки и многоугольники (полигоны). Они могут объединяться любым образом для представления любого объекта, ограниченного 2, 2,5 или 3 измерениями. В связи с особой природой объектов пространственных данных разработчики создали механизмы индексации (пространственные индексы), предназначенные для поддержки нерегламентированных запросов и визуальных представлений содержимого базы данных. Например, пространственный индекс позволит найти ответ на запрос «Каково расстояние между двумя точками?» или «Пересекает ли эта линия определенный набор многоугольников?». Выполнение таких запросов представляется непростым. И действительно, эффективная работа с пространственными большими данными — одна из сложнейших задач в сфере больших данных, которую еще только предстоит решить.

Пакет приложений PostGIS/OpenGEO

PostGIS (www.postgis.org) — это проект с открытым исходным кодом, поддерживаемый организацией Refractions Research (www.refractions.net) и лицензируемый в соответствии с универсальной общедоступной лицензией GNU. PostGIS также поставляется как часть приложения OpenGeo Suite community edition (www.opengeo.org) в рамках корпоративной лицензии.

PostGIS отличается от других баз данных, описываемых в данной главе. Это специализированная многослойная реализация, работающая на реляционной СУБД PostgreSQL, и она удачно сочетает преимущества обеих моделей. Во-первых, PostGIS присущи такие важные характеристики реляционной СУБД SQL, как транзакционная целостность и поддержка требований ACID. Во-вторых, она позволяет выполнять специализированные операции, необходимые для пространственных приложений, такие как повторная проекция, поддержка геодезии, геометрические преобразования и т.п.

Конечно, сама по себе база данных очень важна, но для работы с пространственными приложениями понадобятся и другие решения. К счастью, PostGIS входит в состав экосистемы компонентов, ориентированной именно на такие задачи. Помимо PostGIS пакет приложений OpenGEO содержит следующие компоненты:

- GeoServer (сервер обработки и совместного использования геопространственных данных). GeoServer реализован на Java и предоставляет доступ к пространственной информации из нескольких основных источников пространственных данных в сети. Может интегрироваться с приложением GoogleEarth и обладает веб-интерфейсом для удобства администрирования;
- OpenLayers (библиотека для создания карт). Библиотека, написанная на JavaScript, которая используется для отображения карт и других представлений пространственных данных в веб-браузере. Может управлять изображениями карт из большинства картографических источников в Интернете, включая карты Bing, Google, Yahoo!, OpenStreetMap и т.д.;
- GeoExt (библиотека для создания приложений веб-картографии). Это решение предназначено для разработчиков веб-приложений. Оно обеспечивает доступ к картографическим данным из OpenLayers. Виджеты GeoExt можно использовать для создания, редактирования, просмотра, стилизации геопространственных данных и другого интерактивного взаимодействия через веб-сервисы;
- GeoWebCache (надстройка кэширования данных сервера GeoServer). Средство ускорения доступа к данным, размещенным на сервере и обработанным для отображения в браузере. Формирует кэш из элементов изображений («тайлов») и подготавливает их для быстрой доставки на устройство отображения.

В основном использование пространственных баз данных связано с картами и географическими координатами, но появляются и новые сферы применения пространственных данных:

- трехмерное моделирование человеческого тела, зданий, атмосферы и т.п.;
- сбор и анализ данных из сетей датчиков;
- интеграция с архивными данными для изучения трехмерных пространств и объектов по прошествии определенных промежутков времени.

Полиглотная сохраняемость

Полиглотом называют человека, который говорит или пишет на нескольких языках. В контексте информационных технологий этот термин означает набор приложений, использующих несколько ключевых технологий баз данных. Конечно, такой набор приложений найдет применение в работе с большими данными. Даже если сфера использования больших данных в вашей организации очень ограничена, трудно остановиться на одном подходе к сохраняемости данных. База данных с полиглотной сохраняемостью позволяет работать над сложной задачей, разбивая ее на части и применяя к ним различные модели баз данных. После этого результаты агрегируются в гибридное решение для хранения и анализа данных. В чем преимущества работы с полиглотной сохраняемостью?

- Полиглотная сохраняемость уже применяется во многих рабочих процессах. В любой достаточно крупной организации используются разнообразные реляционные СУБД, хранилища данных, витрины данных, плоские файлы, серверы управления контентом и т.п. Эта гибридная среда является общераспространенной, на основе этой среды будут внедряться и большие данные, так что в ней необходимо хорошо ориентироваться, чтобы принимать обоснованные решения относительно интеграции, анализа, актуальности и видимости данных и т.д.
- Если представить себе идеальную среду, в которой применяется только одна технология сохраняемости, скорее всего, окажется, что она не подходит для решения задач, связанных с большими данными, а значит, для работы с ними потребуется применять базы данных других типов и иные дополнительные технологии.

- В зависимости от скорости сбора больших данных и их разнообразия может возникнуть необходимость использовать различные базы данных в рамках одной внедряемой системы. Следует подумать и о требованиях транзакционной целостности: обеспечивать ли соответствие требованиям ACID или ограничиться BASE.

Рассмотрим пример. Требуется определить всех покупателей некоторого товара длительного пользования, которые совершили покупку за последние 12 месяцев и прокомментировали это в социальных сетях, установить, обращались ли эти клиенты в службу технической поддержки (и если да, то когда, сколько раз и как были решены их проблемы), где был приобретен товар и как доставлен покупателю (и была ли стоимость доставки сопоставима с затратами на энергопотребление), сколько заплатил каждый клиент и каким образом осуществлялся платеж, посещали ли покупатели веб-сайт компании (и если да, то сколько раз и какие действия производили на сайте) и так далее, чтобы в результате этого исследования провести таргетированную рекламную акцию, отправляя на смартфон клиента купон на скидку в тот момент, когда он заходит в магазин компании (или ее партнеров).

Это типичная задача из области больших данных. Чтобы получить всю желаемую информацию, потребуется собрать и проанализировать данные с разнообразной структурой из ряда источников, затем определить, кто из клиентов может участвовать в акции, и в режиме реального времени отправлять им купоны с интересным предложением.

Ни одна технология баз данных сама по себе не позволит эффективно и без значительных затрат решить такую задачу. Конечно, часть требуемой информации относится к транзакционным данным и, скорее всего, хранится в реляционной СУБД, однако другие данные являются нереляционным, так что для работы с ними потребуется как минимум два типа модулей сохраняемости (пространственный и графовый) — а это уже полиглотная сохраняемость.

Глава 8

Основные принципы MapReduce

В ЭТОЙ ГЛАВЕ:

Происхождение MapReduce
Функция Map
Функция Reduce
Объединение функций Map и Reduce
Оптимизация задач MapReduce

Тема больших данных стала широко обсуждаться в последние годы. Однако проблемы масштабных вычислений сопровождают всю историю развития вычислительной техники. С появлением каждой новой, более быстрой и мощной компьютерной системы, расширявшей вычислительные возможности, обнаруживались и новые задачи, которые эта система решить еще не могла. Когда появились локальные вычислительные сети, для решения задач, которые становились все более масштабными, системы вычисления и ресурсы хранения данных начали объединять в сети. В основе решения любой задачи, связанной с большими данными, лежит распределение приложений, требующих объемных вычислений и данных. Для обеспечения надежного и масштабируемого распределения понадобились новые технологии и новые подходы. Одним из таких подходов стала программная платформа MapReduce, которая позволяет разработчикам писать программы, способные параллельно обрабатывать огромные объемы неструктурированных данных, используя для этого группу распределенных процессоров.

История MapReduce

В начале 2000-х годов прогнозы компании Google показали, что применяемые на тот момент решения для таких задач, как сбор данных в Интернете, частота запросов и т.д., в целом соответствовали имеющемуся уровню потребностей, однако не смогут эффективно работать в условиях увеличивающейся сложности, связанной с появлением все нового и нового количества пользователей. Инженеры Google пришли к выводу, что проблему можно решить, распределив работу между недорогими компьютерами, объединенными в сеть и составляющими «кластер».

Но само по себе распределение еще не обеспечивает необходимой эффективности. Требуется осуществлять распределение работы параллельно, и тому есть три причины:

- обработка должна поддерживать автоматическое наращивание и сокращение используемых ресурсов;
- обработка должна выполняться независимо от сбоев сети или отдельных систем;
- поскольку разработчики, применяющие этот подход, должны создавать службы, которые могли бы использовать и другие разработчики, необходимо обеспечить независимость рабочих процессов от того, где находятся данные и выполняются расчеты.

Платформа MapReduce разрабатывалась как общая модель программирования. Уже на раннем этапе ее реализации удалось обеспечить выполнение всех ключевых требований к параллельному выполнению, отказоустойчивости, балансировке нагрузки и обработке данных. Проект получил название MapReduce, так как в нем сочетались две возможности существовавших языков программирования: map (сопоставление) и reduce (редукция).

Компания Google разработала платформу MapReduce для решения конкретного круга задач, поэтому практически одновременно с разработкой модели программирования была выполнена и ее реализация, которая впоследствии стала эталонной для других продуктов, основанных на этой же модели. Эталонная реализация продемонстрировала практические возможности и эффективность этой концепции и помогла ее широкому распространению в компьютерной отрасли. Затем стали появляться и другие реализации модели MapReduce, как с открытым исходным кодом, так и коммерческие.

Функциональная и процедурная модели программирования

Map и reduce — это операции модели функционального программирования. Функциональное программирование представляет собой один из двух способов создания программ для решения бизнес-задач. Другой способ — это процедурное программирование. У обеих моделей программирования есть свои преимущества.

Процедурные программы жестко структурированы и содержат пошаговые инструкции по обработке входных данных. Порядок обработки имеет существенное значение; входные данные изменяются по мере перехода от одного шага программы к другому. Примеры процедурных языков — FORTRAN, COBOL, C и C++. Процедурные программы оптимально применять в тех случаях, когда допустимо преобразовывать значения входных данных или когда необходимо сравнить значения, вычисленные на одном шаге, чтобы определить, следует ли продолжить обработку или выйти из программы, предоставив результат.

Напротив, функциональные программы не изменяют входные данные. Они проводят поиск по всему объему данных, выявляя закономерности, затем применяют правила для идентификации важных элементов и собирают эти элементы в списки. Порядок обработки не имеет значения, так как операции осуществляются независимо. Примеры функциональных языков — LISP, Scheme, Prolog и R. Функциональные программы не меняют входные данные и чаще всего используются, если необходимо неоднократно обрабатывать имеющиеся данные для выявления различных закономерностей. Например, по результатам обработки данных о голосовании на последних выборах в США можно получить список округов, поддержавших республиканскую партию, а затем снова обработать эти данные и получить список округов, голосовавших за демократов, таким образом, на выходе будут два разных списка.

Функция map (функция сопоставления)

Эта функция входит в состав многих функциональных языков программирования. Популярность ей принес язык LISP, предназначенный для моделирования искусственного интеллекта, а затем функция map обрела новую жизнь, став основной технологией обработки списков элементов данных (ключей и значений). Чтобы понять, почему эта функция (вместе с reduce) хорошо подходит для работы с большими данными, нужно иметь представление об основах функционального программирования.

В функциональных языках операторы не меняют структуры данных, а создают на выходе новые. Важно, что исходные данные остаются неизменными — это позволяет многократно использовать функ-

цию сопоставления, не опасаясь за сохранность ценных данных. Еще одно преимущество функционального программирования заключается в том, что нет необходимости непосредственно контролировать перемещение или поток данных, таким образом, программисту не нужно явным образом управлять выводом и размещением данных. В распределенной среде часто возникает проблема размещения данных, но функция сопоставления благополучно ее решает. Наконец, в функциональном программировании последовательность операций с данными не является строго заданной, и это весомое преимущество для вычислительного кластера, где задачи выполняются параллельно.

Итак, как же работает функция сопоставления? Функция применяется к каждому элементу списка (который определяется как пара «ключ-значение») и формирует новый список. Например, так можно создать программу, которая будет подсчитывать количество символов в последовательности. Рассмотрим работу функции map на примере, используя для наглядности некоторые элементы программного кода.

Для начала определим входные данные и создадим список. В данном примере мы будем работать со статистикой голосования по округам на последних выборах в США.

```
mylist = ("all counties in the US that participated in the
          most recent general election")
```

Затем с помощью функции сопоставления создаем функцию для отбора округов с численностью населения более 50 000 человек.

```
map howManyPeople (mylist) = [ howManyPeople "county 1";
          howManyPeople "county 2"; howManyPeople "county
          3"; howManyPeople "county 4"; . . . ]
```

Теперь можно получить новый выходной список, где будут все округа с населением более 50 000 человек.

```
(no, county 1; yes, county 2; no, county 3; yes, county 4;
          ?, county nnn)
```

Выполнение функции не вносит никаких изменений в исходный список. Как видно из кода, в выходном списке каждый элемент сопоставлен с соответствующим элементом входного списка с указанием значения yes или no — «да» или «нет». Если округ удовлетворяет требо-

ванию к численности населения, функция сопоставления помечает его значением «да», в противном случае присваивается значение «нет». Эта важная особенность имеет особое значение для функции редукции.

Функция reduce (функция редукции)

Как и функция сопоставления, функция редукции уже много лет является частью языков функционального программирования. В некоторых языках такая функция называется fold (свертывание). Функция редукции использует выходные данные функции сопоставления и «редуцирует» список в соответствии с заданными условиями. На первом этапе выполнения функции редукции значение помещается в так называемый «аккумулятор», место хранения исходного значения. После сохранения исходного значения в аккумуляторе функция редукции обрабатывает каждый элемент списка и выполняет над ним нужную операцию. Обработав весь список, функция редукции возвращает значение в зависимости от выполненной операции. Вернемся к примеру со статистикой голосования.

Предположим, нам нужно определить округа, в которых большинство проголосовало за демократическую партию. С помощью функции сопоставления мы оценили каждый элемент во входящем списке и получили выходной список округов с населением более 50 000 человек (со значением «да») и менее 50 000 человек (со значением «нет»).

```
(no, county 1; yes, county 2; no, county 3; yes, county 4;
          ?, county nnn)
```

Теперь этот список будет входными данными для нашей функции редукции. Теперь нужно, чтобы в списке остались только округа, поддержавшие на выборах демократов.

```
countylist = (no, county 1; yes, county 2; no, county 3;
          yes, county 4; ?, county nnn)
reduce isDemocrat (countylist)
```

Функция редукции обрабатывает каждый элемент списка и возвращает список всех округов с населением более 50000 человек, где большинство голосовало за демократов.

Теперь представим, что необходимо узнать, в каких округах с населением более 50 000 человек большинство голосов было отдано за республиканцев. Для этого снова задействуем функцию редукции, только поменяем оператор isDemocrat на isRepublican.

```
reduce isRepublican (countylist)
```

Функция возвратит список округов, где большинство голосов было отдано в поддержку республиканской партии. Поскольку элементы списка округов не изменялись, можно продолжить выполнение функции редукции над входными данными, например, посмотреть, как распределялись голоса независимых избирателей, или ограничить результаты лишь частью страны.

Объединение функций сопоставления и редукции

Иногда лишь формирования выходного списка недостаточно. В других случаях требуется только выполнить определенные операции с каждым элементом списка. Но, как правило, необходимо перебрать большое количество входных данных, выбрать определенные элементы данных, а затем обработать их для получения нужной информации. Входные данные могут принадлежать и третьей стороне, так что не нужно вносить изменения во входной список — возможно, что впоследствии его понадобится использовать с новыми данными и новыми задачами.

Приложение функционирует на основе алгоритма. Алгоритм — это последовательность действий, которые необходимы для достижения общей цели. Алгоритм похож на кулинарный рецепт: вы берете отдельные элементы (мука, яйца, сахар...), выполняете пошаговые инструкции (замесить тесто, сформовать изделие, выпечь...) и получаете результат (пирог). Для эффективной совместной работы функций сопоставления и редукции также необходим алгоритм. Он может выглядеть следующим образом.

1. Взять большое количество данных или записей.
2. Выполнить циклический перебор данных.
3. Применить функцию сопоставления для извлечения нужных сведений и создания выходного списка.
4. Упорядочить выходной список для оптимизации дальнейшей обработки.

5. Использовать функцию редукции для вычисления набора результатов.
6. Сформировать окончательные выходные данные.

Таким образом создаются любые виды приложений. Пока мы рассмотрели применение MapReduce на очень простых примерах. А можно ли использовать этот же алгоритм для работы с огромным, порядка терабайта, объемом входных данных? Оказывается, что можно!

На рис. 8.1. представлена схема работы с данными в MapReduce. Создается впечатление, что операции осуществляются независимо друг от друга. Так и есть! Преимущество MapReduce — способность работать по принципу «разделяй и властвуй», то есть разбивать большую задачу на несколько меньших, более управляемых, работать с каждой частью отдельно, а затем объединять результаты. Более того, функция сопоставления является коммутативной — порядок выполнения для этой функции не имеет значения.

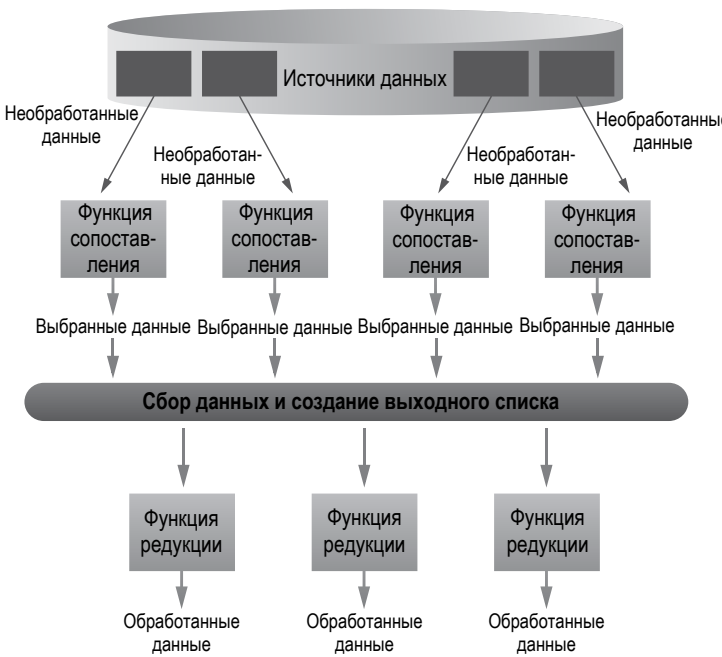


Рис. 8.1. Поток данных в MapReduce

Если вы вообще помните алгебру, вы можете вспомнить, что когда что-то коммутативно, результат тот же, независимо от порядка элементов. Например:

```
5 + 7 = 7 + 5
```

или

```
3 * 4 = 4 * 3
```

Поэтому MapReduce может выполнять свою работу на разных машинах в сети и получать такой же результат, как если бы вся работа выполнялась на одной машине. Он также может использоваться из нескольких источников данных, внутренних или внешних. MapReduce отслеживает свою работу, создавая уникальный ключ, чтобы гарантировать, что вся обработка связана с решением одной и той же проблемы. Этот ключ также используется для вывода всех результатов вместе в конце всех распределенных задач.

Когда функции map и reduce используются таким образом, они работают коллективно, чтобы работать как одно задание внутри кластера. Все деление и захват прозрачно выполняются каркасом выполнения механизма MapReduce, и вся работа распределяется на один или несколько узлов в сети.

Вам нужно понять некоторые характеристики структуры выполнения, чтобы вы могли лучше понять, почему все работает так, как они делают. Это может помочь вам разработать лучшие приложения, а также оптимизировать выполнение для производительности или эффективности. Ниже перечислены основы поведения MapReduce:

- Планирование. Задания MapReduce разбиваются на отдельные задачи для сопоставления и редукции частей приложения. Поскольку сопоставление должно быть завершено до того, как редукция может иметь место, эти задачи приоритетны в зависимости от количества узлов в кластере. Если у вас больше задач, чем узлов, среда выполнения будет управлять задачами сопоставления до тех пор, пока все не будут выполнены. Тогда задачи редукции будут выполняться с одинаковым поведением. Весь процесс завершен только тогда, когда все задачи редукции успешно выполнятся.

- Синхронизация. Когда несколько процессов выполняются одновременно в кластере, вам нужен способ для бесперебойной работы. Механизмы синхронизации делают это автоматически.

Зная, программа выполняет сопоставление и редукцию, платформа отслеживает, какие шаги уже были выполнены и когда. После за вершения всех действий по сопоставлению начинается редукция. Промежуточные данные по мере возникновения копируются по се ти; для этого используется механизм, называемый «перетасовкой и сортировкой». Его задача - сбор и подготовка всех сопоставленных данных к редукции.

- Совместное размещение кода и данных. Наиболее эффективной обработка будет в том случае, если код, реализующий функции сопоставления, размещен на одном компьютере с данными, которые обрабатываются этим кодом. Средство планирования процессов учитывает это обстоятельство и может перед выполнением размещать код и обрабатываемые им данные на одном узле.
- Обработка сбоев и ошибок. Если произойдет сбой, скорее всего, он почти не скажется на результатах. Большинство модулей MapReduce обладает высокой устойчивостью к сбоям и надежной системой обработки ошибок. Учитывая количество узлов в кластере и элементов данных на каждом узле, сбой рано или поздно где-нибудь произойдет. Однако модуль быстро определяет, что появилась проблема, и вносит корректировки в процесс: например, если некоторые задачи сопоставления не выполняются, их перенаправляют на другой узел. Модуль автоматически распознает незавершенную задачу и назначает ее другому узлу, чтобы довести выполнение функции до конца.

Оптимизация задач MapReduce

Для повышения надежности и производительности работ с MapReduce оптимизировать можно не только код приложения, но и другие элементы: топологию оборудования и сети, синхронизацию и файловую систему.

Топология оборудования и сети

Быстрое оборудование и быстрые сети обеспечат быструю работу любого программного обеспечения. Особое преимущество MapReduce — возможность работы на недорогих кластерах, построенных на базе стандартного компьютерного и сетевого оборудования. Но если не уделять внимания физической организации серверов, невозможно

обеспечить оптимальную производительность и высокую надежность, необходимые для работы с большими данными.

Недорогое стандартное оборудование часто размещается в стойках центра обработки данных. Близкое расположение оборудования в пределах одной стойки обеспечивает более высокую производительность, так как не тратятся время и ресурсы на передачу данных и кода от одной стойки к другой. Модуль MapReduce можно настроить таким образом, чтобы он учитывал расположенное близко друг к другу различное оборудование. Близкое размещение данных и кода — один из лучших способов оптимизировать производительность MapReduce. Чем ближе элементы оборудования друг к другу, тем ниже величина задержки.

Синхронизация

Поскольку хранить все результаты сопоставления на одном узле неэффективно, механизмы синхронизации копируют результаты сопоставления на узлы редукции сразу же после завершения сопоставления, что позволяет незамедлительно начать дальнейшую обработку. Все значения, связанные с одним ключом, передаются одному и тому же средству редукции, что также обеспечивает более высокую производительность и эффективность. Выходные данные после редукции записываются непосредственно в файловую систему, поэтому она должна быть сконфигурирована и настроена наилучшим образом.

Файловая система

Успешная реализация MapReduce базируется на использовании распределенной файловой системы. Основное отличие между локальной и распределенной файловыми системами — емкость. Для обработки огромных объемов данных файловые системы должны быть распределены по различным компьютерам или узлам в сети. Реализации MapReduce полагаются на метод распределения с использованием главного и вспомогательного узлов: главные узлы хранят все метаданные, права доступа, сопоставления, места размещения файлов и блоков и т.д., а вспомогательные узлы служат для хранения данных. Все запросы поступают на главный узел, а затем обрабатываются соответствующим вспомогательным узлом. Выбирая файловую систему, необходимую для поддержки реализации MapReduce, нужно учитывать следующие факторы:

- обеспечение «горячего» резервирования. Логично, что нагрузка на главный узел очень высока — вся обработка начинается с него. Если на главном узле произойдет сбой, до его восстановления вся файловая система будет недоступна. Поэтому следует предусмотреть создание «горячего» резервного главного узла, который незамедлительно включится в работу в случае неполадки на главном узле;
- чем больше, тем лучше. Речь идет о размере файлов. Следует избегать чрезмерного накапливания файлов небольшого объема (менее 100 МБ) — распределенные файловые системы, поддерживающие модули MapReduce, лучше всего работают с небольшим количеством больших по объему файлов;
- анализ показателей работы. Поскольку рабочие нагрузки обрабатываются пакетно, высокая производительность и устойчивая работа сети имеют большее значение, чем быстродействие средств сопоставления и редукции. Оптимальный подход — потоковая передача кодом больших объемов данных при чтении из файловой системы и при записи в нее;
- обеспечение безопасности. Безопасность обеспечивать нужно, но главное — не переусердствовать. Многоуровневая система безопасности заметно снижает производительность распределенной файловой системы. Права на доступ к файлам служат для защиты от непреднамеренных действий, а не целенаправленного причинения вреда. Лучше всего организовать защиту так, чтобы доступ к среде центра обработки данных имели только авторизованные пользователи, а распределенная файловая система была защищена от внешних атак.

В следующей главе рассмотрим более подробно самую распространенную реализацию MapReduce и ее экосистему.

Глава 9

Мир Hadoop

В ЭТОЙ ГЛАВЕ:

Что такое Hadoop и в чем его преимущества
Обзор распределенной файловой системы Hadoop
Подробнее о Hadoop MapReduce
Как работать с Hadoop

Для обработки источников с большим объемом данных традиционных подходов недостаточно. Большинство технологий отступали перед объемом, скоростью и разнообразием больших данных, поэтому для решения этих новых задач должны были появиться новые технологии. Одной из них стала MapReduce — но это всего лишь алгоритм, общие принципы работы с данными. Чтобы воспользоваться всеми преимуществами MapReduce, требуется нечто большее, чем просто алгоритм, — необходим набор продуктов и технологий, разработанных для решения задач, связанных с использованием больших данных.

Что такое Hadoop

Компании Yahoo! и Google, первопроходцы в области поисковых систем, были заинтересованы в том, чтобы упорядочить огромные массивы данных, собираемых поисковыми системами: понять, что за информация собирается и как монетизировать эти данные в рамках бизнес-модели. Проект Hadoop получил развитие, поскольку предложил самый прагматичный подход к управлению большими объемами данных. Платформа Hadoop позволила разделять крупные задачи на небольшие элементы, что обеспечивало скорость и экономическую эффективность анализа данных.

Разбивка комплексных задач по работе с большими данными на небольшие элементы, которые можно обрабатывать параллельно, позволяет эффективно обработать информацию и представить объединенный результат такой обработки.

Проект Hadoop (http://hadoop.apache.org) изначально был создан инженером компании Yahoo! Дагом Каттингом. Теперь это проект с открытым исходным кодом, который поддерживается Фондом Apache Software Foundation. Он доступен в рамках лицензии Apache версии 2.0. Наряду с другими проектами, которые мы рассмотрим в главе 10, Hadoop является основополагающим решением для сохранения и обработки больших данных. Платформа Hadoop организует параллельную обработку данных на множестве вычислительных узлов, благодаря чему сокращается длительность вычислений и величина задержек. В основе Hadoop лежат два компонента:

- распределенная файловая система Hadoop. Это надежный, экономный по затратам кластер хранилища данных с высокой пропускной способностью, который упрощает управление связанными файлами на различных компьютерах;
- модуль MapReduce. Это реализация алгоритма MapReduce, предназначенная для высокоэффективной параллельной или распределенной обработки данных.

Hadoop предназначается для обработки больших объемов (от терабайтов до петабайтов) структурированных и неструктурированных данных. Он реализуется на стойках стандартных серверов в виде кластера Hadoop. Серверы можно динамически добавлять в кластер и удалять из кластера, так как Hadoop обеспечивает самовосстановление. Другими словами, Hadoop обнаруживает изменения (в том числе сбои) и корректирует работу в зависимости от этих изменений, обеспечивая бесперебойное функционирование.

Теперь подробнее рассмотрим распределенную файловую систему Hadoop (HDFS) и реализацию технологии MapReduce в Hadoop.

Обзор распределенной файловой системы Hadoop (HDFS)

Распределенная файловая система Hadoop — это гибкая и надежная кластерная технология управления файлами в среде больших данных. HDFS — это не конечный пункт назначения файлов, а скорее служба,

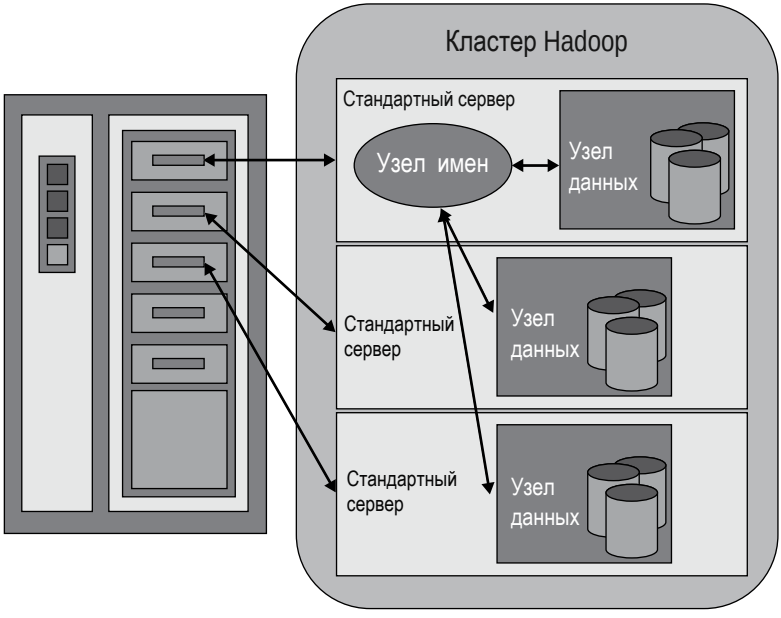


Рис. 9.1. Оборудование, входящее в кластер Hadoop

которая предоставляет уникальный набор возможностей для быстрой обработки больших объемов данных. В HDFS данные записываются один раз, а впоследствии многократно считываются, в других же файловых системах запись и считывание происходят постоянно. Именно поэтому файловая система HDFS идеально подходит для поддержки анализа больших данных. Служба включает узел имен (NameNode) и несколько узлов данных, которые работают на кластере, состоящем из стандартного оборудования. Самый высокий уровень производительности обеспечивается в том случае, если весь кластер находится в одной физической стойке одного центра обработки данных. По существу, узел имен отслеживает физическое размещение данных. На рис. 9.1 представлена базовая архитектура HDFS.

Узлы имен

HDFS разбивает большие файлы на фрагменты, которые называются блоками. Блоки сохраняются на узлах данных, а узел имен отслеживает, какие блоки на каких узлах данных в совокупности составляют

единый файл. Узел имен также регулирует все случаи доступа к файлам, включая операции чтения, записи, создания, удаления и репликации блоков на узлах данных. Полная коллекция всех файлов в кластере иногда называется пространством имен файловой системы. Управление этим пространством осуществляется на узле имен.

Несмотря на наличие жесткой связи между узлом имен и узлами данных, сопряжение между узлами данных является свободным. Это обеспечивает динамическое поведение элементов кластера и позволяет добавлять и убирать серверы по мере увеличения или уменьшения потребностей. В стандартной конфигурации на одном физическом сервере в стойке присутствует один узел имен и, возможно, один узел данных. Другие серверы выполняют роль исключительно узлов данных.

В отличие от узла имен, функций у узлов данных не очень много. Узлы данных постоянно опрашивают узел имен для получения заданий, требующих выполнения. Благодаря этому узел имен располагает сведениями о том, какие узлы данных присутствуют в системе и насколько они загружены. Узлы данных также общаются друг с другом, чтобы совместно работать над выполнением стандартных операций файловой системы, поскольку часто блоки одного файла хранятся на нескольких узлах данных. Так как узел имен критически важен для правильной работы всего кластера, он может и должен реплицироваться.

Узлы данных

Для узлов данных важна не многофункциональность, а отказоустойчивость. В кластере HDFS блоки данных реплицируются по разным узлам данных, а доступом к ним управляет узел имен. Оптимальная эффективность механизма репликации достигается при установке всех узлов кластера в одну стойку. По сути, узел имен использует «идентификатор стойки» для отслеживания узлов данных в кластере; принадлежность к той или иной стойке является важной характеристикой для кластеров HDFS. Узлы данных также посылают регулярные сообщения о своем статусе («heartbeat-сообщения») для обнаружения и поддержки подключений между узлом имен и узлами данных. При отсутствии heartbeat-сообщения узел имен исключает узел данных из кластера и продолжает работу. При возобновлении поступления (или появлении новых) heartbeat-сообщений узел данных добавляется в кластер незаметно для пользователей или приложений.

Как и в любой файловой системе, целостность данных является ключевым элементом. HDFS поддерживает ряд возможностей, предназначенных для обеспечения целостности данных. Естественно, что при разбивке файлов на блоки и распределении по различным серверам кластера любая операция с любым элементом может повлиять на целостность данных. В HDFS для обеспечения целостности в рамках кластера используются журналы транзакций и проверка контрольных сумм.

Журналы транзакций широко применяются в файловых системах и базах данных. В них отслеживается каждая операция, поэтому они используются для ревизии и повторного создания файлов системы в случае возникновения нештатных ситуаций.

Проверка контрольных сумм позволяет подтвердить неизменность содержимого файлов, входящих в HDFS. Когда клиент запрашивает файл, содержимое файла проверяется на соответствие контрольной сумме, и если контрольная сумма совпадает, то операции с файлом будут разрешены. В противном случае регистрируется ошибка. Для предотвращения случайного или намеренного изменения контрольных сумм они хранятся в скрытых файлах.

Сохраняемость на узлах данных обеспечивается использованием локальных дисков сервера. Все блоки данных хранятся локально, преимущественно для повышения производительности. Блоки данных реплицируются по узлам данных, поэтому отказ одного сервера не обязательно влечет повреждение файла. Степень репликации, количество узлов данных и пространство имен HDFS задаются при создании кластера. Так как файловая система HDFS является динамической, все параметры могут быть изменены во время работы кластера.

Как устроена распределенная файловая система HDFS

Как уже отмечалось выше, HDFS справляется с большими трудностями больших данных — объемом, скоростью и разнообразием — путем разбивки файлов на связанную коллекцию блоков меньшего размера. Эти блоки распределяются по узлам данных в кластере HDFS и управляются узлом имен. Размер блоков можно настроить; как правило, он составляет 128 МБ или 256 МБ, то есть файл объемом 1 ГБ по умолчанию распределяется по восьми блокам по 128 МБ. Файловая система HDFS является отказоустойчивой, поскольку эти блоки реплицируются по всему кластеру на случай отказа сервера. Для отслеживания всех этих частей файлов используются метаданные файловой системы.

Метаданные — это «данные о данных». Они уже много лет используются в вычислительных технологиях под разными названиями; в последнее время их также часто называют тегами. Можно сказать, что метаданные HDFS служат шаблоном для подробного описания характеристик файлов и кластеров. Метаданные содержат следующую информацию:

- время создания файла, его открытия, изменения, удаления и т.д.;
- время сохранения блоков файла в кластере;
- права на просмотр или изменение файла;
- количество файлов, сохраненных в кластере;
- количество узлов данных в кластере;
- местоположение журнала транзакций кластера.

Метаданные HDFS хранятся на узле имен. Во время работы кластера все метаданные загружаются в физическую память сервера узла имен. Чем больше кластер, тем, соответственно, больше место, занимаемое метаданными, поэтому для оптимальной производительности сервер узла имен должен иметь большой объем памяти. Идеальный вариант — это большое количество твердотельных накопителей.

Как уже говорилось в начале главы, узлы данных по своей сути очень просты. Это серверы, которые содержат блоки определенного набора файлов. Узлы данных можно рассматривать как «серверы блоков», так как работа с блоками и есть их основная функция. В процессе осуществления этой функции сервер блоков решает следующие задачи:

- сохраняет блоки данных в локальной файловой системе сервера и возвращает их. Файловая система HDFS может работать с различными операционными системами и одинаковым образом функционирует в среде Windows, Mac OS или Linux;
- сохраняет метаданные блока на основе шаблона метаданных в локальной файловой системе узла имен;
- выполняет периодические проверки контрольных сумм файлов;
- отправляет на узел имен регулярные отчеты о доступных для работы блоках файлов;
- предоставляет по запросу клиентам метаданные и данные. HDFS поддерживает прямой доступ к узлам данных из клиентских приложений;
- в конвейерном режиме перенаправляет данные на другие узлы данных.

Размещение блоков на узлах данных имеет критическое значение как для репликации, так и для поддержки конвейерного режима обработки данных. HDFS локально сохраняет по одной реплике каждого блока, затем размещает вторую реплику на сервере какой-либо удаленной стойки, чтобы обеспечить защиту на случай полного отказа одной из стоек, и передает третью реплику на другой сервер в той же удаленной стойке. Наконец, система также может отправить дополнительные реплики в случайным образом выбираемые разделы на локальном сервере или на удаленных кластерах. Механизмы репликации данных и обеспечения отказоустойчивости реализованы в HDFS весьма эффективно. Для клиентских приложений расположение блоков не имеет значения — клиент перенаправляется на ближайшую к нему реплику, что обеспечивает максимальную производительность.

HDFS поддерживает возможность создания конвейеров данных. Конвейер — это соединение между несколькими узлами данных, которое обеспечивает перемещение данных между серверами. Клиентское приложение передает блок на первый узел конвейера данных, узел данных принимает блок и перенаправляет данные на следующий узел конвейера, и так далее, пока все данные и реплики данных не будут записаны на диск. Затем клиентское приложение повторяет процесс, записывая следующий блок в файл. Как будет показано ниже, это очень важная функция для Hadoop MapReduce.

Как же система балансирует огромное количество файлов, блоков и серверов? Если оставить это на волю случая, одни узлы данных будет переполняться, а другие пустовать. Чтобы этого не происходило, в HDFS используется служба балансировки нагрузки, которая выравнивает нагрузку на узлы данных в зависимости от степени заполнения локальных дисков каждого сервера. Служба балансировки нагрузки работает, когда кластер активен; этот процесс можно приостанавливать, чтобы не отвлекать на него сетевой трафик. Основная задача файловой системы HDFS — управление файлами и блоками, а балансировка кластеров вторична.

Служба балансировки нагрузки очень эффективна, но не располагает возможностями для адаптации. К примеру, она не позволяет создать пользовательские шаблоны для доступа и загрузки вместо применяемых по умолчанию установок и не выявляет «проблемных» данных, требующих корректировки системы. Однако можно предполагать, что такие функции появятся в следующих версиях HDFS.

Hadoop MapReduce

Необходимо отличать MapReduce как алгоритм от реализации MapReduce. Hadoop MapReduce — это реализация указанного алгоритма, созданная и поддерживаемая в рамках проекта Apache Hadoop.

Реализацию MapReduce можно сравнить с двигателем в автомобиле: пользователь «заправляет» его входными данными, MapReduce быстро и эффективно преобразует входные данные в выходные данные, и пользователь получает желаемый результат. Чтобы эффективно использовать Hadoop для решения бизнес-задач, необходимо понимать принципы его работы. Итак, рассмотрим реализацию Hadoop MapReduce более подробно.

Рабочий процесс Hadoop MapReduce состоит из нескольких этапов; на каждом из них производятся важные операции, которые в совокупности способствуют получению информации из больших данных. Процесс начинается с пользовательского запроса на запуск програм-

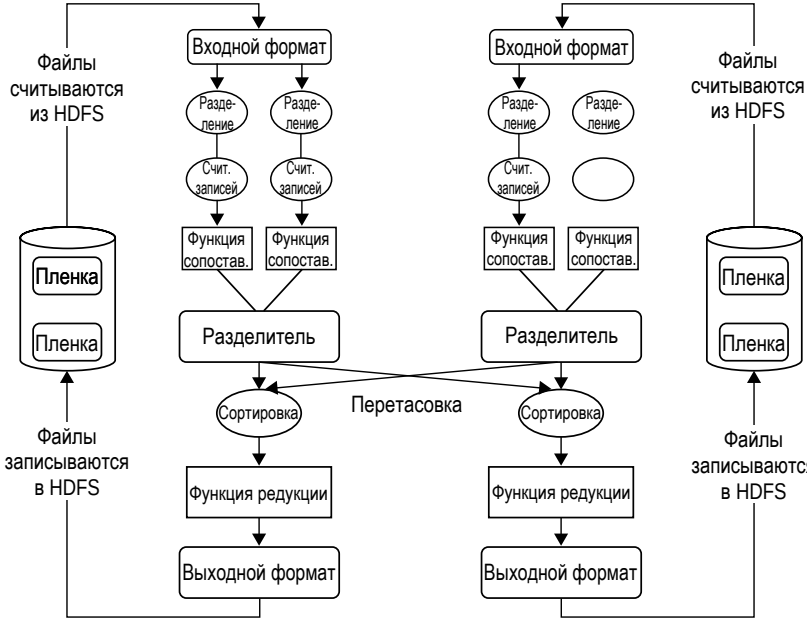


Рис. 9.2. Рабочий процесс и перемещение данных в небольшом кластере Hadoop

мы MapReduce и продолжается вплоть до записи результатов обратно в HDFS. Схема процесса приведена на рис. 9.2.

HDFS и MapReduce работают на узлах кластера, которые размещены на стандартных серверах, установленных в стойку. Для простоты на схеме показаны только два узла.

Подготовка данных

При запросе клиентом на запуск программы MapReduce осуществляется нахождение и чтение входного файла исходных данных. Формат файла произвольный, однако данные должны быть преобразованы в формат, который сможет обработать программа. Это задача устройства входного формата (InputFormat) и считывателя записей (RecordReader).

Устройство входного формата с помощью функции разбивки входных данных InputSplit определяет, как файл будет разбит на меньшие фрагменты. Затем назначается считыватель записей, который обеспечивает преобразование исходных данных для работы функции сопоставления. Как рассказывалось в главе 8, функция сопоставления работает с парами ключей и значений входных данных. Вместе с Hadoop поставляются несколько типов считывателей данных, что позволяет использовать различные варианты преобразования. Преобразование — одна из возможностей, позволяющих Hadoop управлять широким спектром типов больших данных.

Сопоставление

Итак, данные приведены к виду, приемлемому для функции сопоставления. Для каждой входной пары создается отдельный экземпляр сопоставления, который позволяет обработать данные. У функции сопоставления есть две дополнительные возможности, которые обеспечивают обращение с обработанными выходными данными и их отслеживание. Так как функции сопоставления и редукции работают во взаимосвязи, программа должна собрать выходные данные со всех средств сопоставления и передать их средствам редукции. Эта задача выполняется сборщиком выходных данных (OutputCollector), а функция составления отчетов (Reporter) передает сведения, полученные от средств сопоставления, позволяя следить за результативностью выполнения задач сопоставления.

Все это осуществляется одновременно на нескольких узлах кластера Hadoop. В некоторых случаях бывает необходимо до начала редукции свести воедино выходные данные определенных процессов сопоставления. В других случаях перед редукцией требуется обработать промежуточные результаты. Возможно также, что некоторые из этих выходных данных будут размещаться не на том узле, где запущены средства редукции для их обработки. Поэтому промежуточные результаты подвергаются сборке и перетасовке с помощью средств разбиения и сортировки. Результаты выполнения задач сопоставления направляются в особый раздел, после окончания сопоставления эти результаты перетасовываются и распределяются по средствам редукции так, чтобы обеспечить оптимальную обработку.

Редукция и объединение

Каждая выходная пара подвергается редукции. Как и при работе функции сопоставления, выходные данные собираются по мере выполнения всех заданий. Редукция начинается только после того, как выполнены все действия по сопоставлению, и завершается только тогда, когда окончена работа всех экземпляров. Выходными данными редукции также являются ключ и значение. Зачастую это не самый удобный выходной формат для приложения, поэтому в Hadoop применяется функция выходного формата (OutputFormat), которая работает аналогично функции входного формата (InputFormat). Функция выходного формата обрабатывает пары «ключ-значение» и упорядочивает выходные данные для записи в HDFS. Последний этап процесса — фактическая запись данных в HDFS, которая выполняется средством занесения записей (RecordWriter), аналогичным средству считывания записей (RecordReader), но работающему в обратном порядке. Данные функции выходного формата (OutputFormat) записываются в HDFS в той форме, которая соответствует требованиям приложения.

В более ранних версиях Hadoop координацией всех этих действий управлял планировщик заданий. Такой планировщик обеспечивал лишь простейший функционал, и по мере изменения характера заданий и увеличения их количества стало понятно, что требуется другой подход. Основным недостатком прежних реализаций планировщика было отсутствие управления ресурсами. В последней версии Hadoop используются новые возможности, которые мы подробно рассмотрим в главе 10.

Hadoop MapReduce является основой системы Hadoop. Она располагает всеми средствами, необходимыми для разбивки больших данных на меньшие, более управляемые части, обеспечивает параллельную обработку этих данных в распределенном кластере и дает возможность дальнейшей обработки данных или их представления пользователю. При этом система устойчива к сбоям и отказам. И это только начало. Экосистема Hadoop — большой развивающийся набор инструментов и технологий, разработанных специально для решения задач больших данных путем их деления на множество небольших подзадач.

Глава 10

Экосистема Hadoop

В ЭТОЙ ГЛАВЕ:

Почему экосистема Hadoop является основой для больших данных
Управление ресурсами и приложениями с помощью Hadoop YARN
Хранение больших данных в HBase
Интеллектуальный анализ больших данных с помощью Hive
Взаимодействие с экосистемой Hadoop

Итак, Hadoop MapReduce и распределенная файловая система Hadoop (HDFS), о которых рассказывалось в главе 9, являются мощными технологиями, предназначенными для решения задач, связанных с большими данными. Но дело в том, что эффективное использование этих компонентов требует глубоких познаний в информатике и навыков программирования. Поэтому была разработана экосистема Hadoop, которая делает применение Hadoop доступным для менее подготовленного пользователя. На протяжении ряда лет разработчики по всему миру создают и тестируют инструменты (как коммерческие, так и с открытым исходным кодом), повышающие удобство использования Hadoop и расширяющие сферу применения этой технологии. Одно за другим появляются решения, совершенствующие те или иные элементы экосистемы Hadoop, и таким образом весь проект развивается. При этом обеспечивается надлежащая степень безопасности и контроля.

В этой главе мы рассмотрим технологии в составе экосистемы Hadoop.

Формирование основы системы больших данных с помощью экосистемы Hadoop

Приступать к решению задач, связанных с использованием больших данных, не располагая специализированными инструментами, технологиями и службами, — все равно что пытаться вычерпать море ложкой. Ключевые компоненты (Hadoop MapReduce и HDFS) постоянно совершенствуются, но необходимы и другие средства. Экосистема Hadoop предлагает постоянно расширяющийся набор инструментов и технологий, специально предназначенных для упрощения разработки, развертывания и поддержки решений, обеспечивающих работу с большими данными. Но прежде чем описывать важнейшие компоненты экосистемы Hadoop, рассмотрим ее в целом и охарактеризуем роль, которую Hadoop играет в сфере больших данных.

Ни одно здание нельзя построить без фундамента. Однако устойчивость — это не единственный важный критерий. Каждая часть здания должна соответствовать его общему назначению, выполнять собственные функции и в то же время эффективно взаимодействовать с другими. Это справедливо и для экосистемы Hadoop. Фундаментом, который служит опорой всей системы и поддерживает связь между компонентами, являются MapReduce и HDFS. Они обеспечивают базовую структуру и службы интеграции, необходимые для выполнения ключевых требований, предъявляемых к решению задач больших данных. Остальная экосистема — это компоненты, необходимые для создания приложений, решающих прикладные задачи обработки больших данных, а также для управления этими приложениями.

В отсутствие сложившейся экосистемы (как часто бывает с инновациями, которые только начинают внедряться) разработчикам, администраторам базы данных, сетевым и системным администраторам и другим специалистам пришлось бы определять и согласовывать набор технологий для создания и развертывания решений больших данных. Свести в единую систему различные технологии на новом, неосвоенном рынке — чрезвычайно сложная задача. Именно поэтому экосистема Hadoop является фундаментальной основой для успешной реализации обработки больших данных. Это самая полная коллекция инструментов и технологий, доступных на сегодняшний день для решения проблем, связанных с большими данными. Экосистема упрощает создание новых возможностей для больших данных и расширяет их применение различными компаниями и организациями.

Управление ресурсами и приложениями с помощью Hadoop YARN

Планирование и отслеживание выполнения заданий — неотъемлемая часть Hadoop MapReduce. Более ранние версии Hadoop также поддерживали базовую систему отслеживания, однако прежние реализации планировщика не справлялись с постоянно менявшимся составом задач, поддерживаемых Hadoop. В частности, они не позволяли управлять заданиями, не связанными с MapReduce, и не могли оптимально использовать кластеры. Для решения этих проблем и для повышения гибкости, эффективности и производительности была разработана новая технология.

YARN (Yet Another Resource Negotiator, еще один ресурсный посредник) — это ключевая служба Hadoop, которая, в свою очередь, обеспечивает работу двух основных служб:

- глобального управления ресурсами (ResourceManager);
- управления приложениями (ApplicationMaster).

ResourceManager (диспетчер ресурсов) — это основная служба, которая управляет диспетчером узла (NodeManager) кластера Hadoop. В ResourceManager имеется планировщик, главной задачей которого является выделение системных ресурсов под определенные работающие приложения (задачи). Однако сам планировщик не отслеживает состояние этих приложений — вся необходимая системная информация хранится в контейнере ресурсов. Там находятся подробные сведения о ЦП, диске, сети и прочих важных атрибутах ресурсов, которые необходимы для работающих на узле и в кластере приложений.

Каждый узел имеет диспетчер узла NodeManager, который подчиняется глобальному диспетчеру ресурсов кластера ResourceManager. Диспетчер узла отслеживает использование приложениями ЦП, диска, сети и памяти, а также передает все эти сведения диспетчеру ресурсов. Каждое приложение, работающее на узле, контролируется соответствующим мастером приложения ApplicationMaster. Если для работы приложения требуется больше ресурсов, ApplicationMaster уведомляет NodeManager, а NodeManager согласовывает с планировщиком ResourceManager выделение приложению дополнительных мощностей. NodeManager также несет ответственность за отслеживание состояния и хода выполнения заданий на этом узле.

Хранение больших данных в HBase

HBase — это распределенная нереляционная (столбцовая) база данных, которая использует файловую систему HDFS для обеспечения сохраняемости. Она смоделирована на основе Google BigTable и поддерживает хранение очень больших таблиц (миллиарды столбцов и строк), поскольку работает на кластерах Hadoop, использующих стандартное оборудование. HBase обеспечивает неограниченный доступ к большим данным, их чтение и запись в режиме реального времени. HBase отличается гибкостью настройки, что обеспечивает эффективную обработку больших объемов данных. Таким образом, HBase может применяться для решения задач, связанных с большими данными.

HBase — это столбцовая база данных. Все данные хранятся в таблицах, где есть строки и столбцы, аналогично реляционным системам управления базами данных (СУБД). Пересечение строки и столбца называется ячейкой. Одним из основных отличий таблиц HBase от таблиц реляционных СУБД является поддержка управления версиями (версионность): значение каждой ячейки имеет атрибут версии, который является не чем иным, как отметкой времени, которая уникально идентифицирует ячейку. Версионность позволяет отслеживать изменения в ячейке и получать необходимую версию содержимого. HBase хранит данные в ячейках в порядке убывания (согласно отметкам времени), поэтому первыми всегда считываются самые новые значения.

Столбцы в HBase могут группироваться в семейства. Имя семейства столбцов используется в качестве префикса для идентификации членов этого семейства. Например, «фрукты: яблоко» и «фрукты: банан» — это члены семейства столбцов «фрукты». Реализации HBase настраиваются на уровне семейства столбцов, поэтому важно решить, как пользователи будут получать доступ к данным и насколько большими будут столбцы.

Строки в таблицах HBase также имеют связанный с ними ключ. Ключ имеет очень гибкую структуру. Ключом может быть вычисленное значение, строка и даже другая структура данных. Ключ используется для управления доступом к ячейкам в строке, хранящимся в порядке возрастания их значений.

Вместе все эти компоненты составляют схему. Схема должна быть определена и создана до сохранения каких-либо данных, тем не менее после создания и запуска базы данных можно создавать таблицы и до-

бавлять новые семейства столбцов. Такая расширяемость оказывается очень полезной при работе с большими данными, поскольку степень разнообразия потоков данных не всегда известна изначально.

Интеллектуальный анализ больших данных с помощью Hive

Hive (то есть куст) — это пакетно-ориентированный слой хранения данных, основанный на ключевых элементах Hadoop (HDFS и MapReduce). Он предоставляет пользователям, знакомым с SQL, простую реализацию этого языка — HiveQL, обеспечивая полноценный доступ через средства сопоставления и редукции. Hive позволяет воспользоваться преимуществами обеих систем: доступом к структурированным данным, который дает SQL, и комплексным анализом больших данных, обеспечиваемым MapReduce.

В отличие от большинства хранилищ данных, Hive не ориентирован на быстрое реагирование на запросы. Фактически на выполнение запроса, в зависимости от его сложности, может уходить от нескольких минут до нескольких часов. Таким образом, Hive лучше всего использовать для анализа данных и детализованной аналитики, не требующих выведения результатов в режиме реального времени. Поскольку в основе Hive лежат механизмы Hadoop, эта база хорошо расширяется, масштабируется, отличается отказоустойчивостью, что является ее преимуществами по сравнению со среднестатистическим хранилищем базы данных.

Hive использует три механизма упорядочивания данных.

- Таблицы. Таблицы Hive не отличаются от таблиц реляционных СУБД, состоящих из строк и столбцов. Так как Hive основан на файловой системе Hadoop HDFS, таблицы сопоставлены с каталогами в файловой системе. Кроме того, Hive поддерживает таблицы, хранящиеся в других файловых системах.
- Разделы. Таблица Hive может содержать один или несколько разделов. Эти разделы сопоставлены с подкаталогами в базовой файловой системе и отражают распределение данных по всей таблице. Например, если таблица называется «Автомобили», где значение ключа равно 12345, а значение производителя — Ford, путь к разделу будет /hivewh/autos/kv=12345/Ford.

- Контейнеры. В свою очередь, данные могут быть разделены на контейнеры, которые хранятся как файлы в каталоге разделов в используемой файловой системе. Контейнеры основаны на хэше столбца в таблице. В предыдущем примере возможно наличие контейнера с именем Focus, который содержит все атрибуты автомобиля Ford Focus.

Метаданные Hive хранятся вне базы данных, в так называемом «хранилище метаданных». Хранилище метаданных — это реляционная база данных, содержащая подробные описания схемы Hive, включая типы столбцов, их владельцев, значения ключей и данных, статистику таблиц и пр. Хранилище метаданных может синхронизировать данные каталога с другими службами метаданных в экосистеме Hadoop.

Язык HiveQL поддерживает многие операции SQL, такие как select, join, aggregate, union all и т.п. За счет общего использования входных данных в одной инструкции HiveQL этот язык также поддерживает многотабличные запросы и вставки. HiveQL может быть расширен для поддержки задаваемой пользователем агрегации, преобразования столбцов и встроенных сценариев MapReduce.

Взаимодействие с экосистемой Hadoop

Создание программ или использование специальных языков-запросов не является единственным способом взаимодействия с экосистемой Hadoop. ИТ-подразделения, ответственные за управление инфраструктурой, должны управлять Hadoop и созданными для этой платформы приложениями для работы с большими данными. Однако по мере распространения больших данных к ним начинают обращаться специалисты из разных областей, необязательно обладающие соответствующими техническими познаниями. Поэтому в составе экосистемы Hadoop представлен ряд технологий, ориентированных на менее подготовленных пользователей.

Pig и Pig Latin

Экосистема Hadoop в первую очередь создавалась для нужд разработчиков программного обеспечения, и именно в этой сфере ее преимущества проявляются наиболее ярко. Однако сфера применения

Hadoop расширяется. Среда Pig призвана сделать платформу Hadoop более доступной для тех, кто не связан с разработкой программ. Pig — это интерактивная или основанная на сценариях среда выполнения, которая поддерживает Pig Latin — язык, используемый для работы с потоками данных. Язык Pig Latin поддерживает загрузку и обработку входных данных с использованием ряда операторов, преобразующих входные данные и формирующих нужные выходные данные.

Среда выполнения Pig поддерживает два режима.

- Локальный режим. Здесь все сценарии выполняются на одном компьютере. Наличие Hadoop MapReduce и HDFS не является обязательным.
- Режим Hadoop. Он также называется режимом MapReduce; все сценарии работают в определенном кластере Hadoop.

Фактически Pig создает набор заданий сопоставления и редукции. Пользователю не нужно писать код, компилировать его, упаковывать, использовать в среде и получать результаты. Во многих отношениях среда Pig аналогична SQL в мире реляционных СУБД. Язык Pig Latin предоставляет абстрактный способ получения ответов на основе больших данных. Он дает возможность пользователю сконцентрировать свое внимание на работе с данными, а не на структуре программы. Среда Pig значительно упрощает создание прототипов. Например, сценарий Pig можно запустить для небольшой выборки из массива больших данных, чтобы убедиться, что результаты соответствуют требованиям конкретной задачи, прежде чем приступить к обработке всех данных.

Программы Pig могут выполняться тремя различными способами, каждый из которых совместим с локальным режимом и режимом Hadoop.

- Сценарий. Это простой файл, содержащий команды Pig Latin и имеющий расширение PIG. Команды интерпретируются Pig и выполняются последовательно.
- Grunt. Grunt — это интерпретатор команд. Можно вводить команды Pig Latin в командной строке Grunt. Эта возможность оказывается очень полезной при создании прототипных сценариев по принципу «что если».
- Встраивание. Программы Pig могут выполняться как часть Java-программ.

Язык Pig Latin отличается очень богатым синтаксисом. Он поддерживает операторы, предназначенные для следующих действий:

- загрузка и сохранение данных;
- потоковая передача данных;
- фильтрация данных;
- группировка и соединение данных;
- упорядочивание данных;
- объединение и разделение данных.

Pig Latin также поддерживает различные типы, выражения, функции, операторы диагностики, макросы и команды файловой системы.

Подробную информацию и примеры использования Pig можно найти на веб-сайте http://pig.apache.org.

Sqoop

Поскольку многие компании хранят данные в реляционных СУБД и других хранилищах данных, в экосистеме Hadoop предусмотрено средство для перемещения данных между этими хранилищами и Hadoop. Иногда данные необходимо перемещать в режиме реального времени, но чаще всего данные загружаются и выгружаются пакетами. Sqoop (SQL-to-Hadoop) — это инструмент, который дает возможность извлекать данные из хранилищ данных, не основанных на Hadoop, преобразовывать данные в форму, которая может использоваться Hadoop, а затем загружать данные в HDFS. Этот процесс называется ETL (Extract, Transform, Load — извлечение, преобразование, загрузка). Перенос данных в Hadoop критически важен для обработки данных с использованием MapReduce, но не менее важна и возможность извлекать данные из Hadoop и передавать их во внешний источник для использования в других приложениях. Для этого тоже применяется Sqoop.

Как и Pig, Sqoop является интерпретатором командной строки. Пользователь вводит команды Sqoop в интерпретатор, и они последовательно выполняются. Четыре ключевые возможности Sqoop:

- пакетный импорт. Импорт в HDFS отдельных таблиц или баз данных целиком. Данные сохраняются в собственных каталогах и файлах системы HDFS;
- прямой ввод. Импорт и сопоставление реляционных баз данных SQL непосредственно в Hive и HBase;

- взаимодействие с данными. Создание классов Java, что дает возможность программного взаимодействия с данными;
- экспорт данных. Экспорт данных непосредственно из HDFS в реляционную базу данных с использованием определения целевой таблицы на основе характеристик целевой базы данных.

Sqoop обрабатывает импортируемую базу данных и выбирает подходящую для исходных данных функцию импорта. После распознавания входных данных выполняется чтение метаданных таблицы (или базы данных) и создается определение класса импортируемых данных. Можно принудительно задать в Sqoop режим, позволяющий получать только отдельные таблицы вместо загрузки всех входных данных и последующего поиска нужной информации. Это позволит значительно сэкономить время. Импорт из внешней базы данных в HDFS выполняется с помощью задания MapReduce, создаваемого Sqoop в фоновом режиме.

Sqoop — пример эффективного инструмента для пользователей-непрограммистов. Важно отметить, что он тесно связан с базовыми технологиями экосистемы, такими как HDFS и MapReduce. Это характерно для многих элементов экосистемы Hadoop.

Zookeeper

Лучшим методом Hadoop для работы с большими данными является возможность «разделять и властвовать». После разделения задачи задействуются методы распределенной и параллельной обработки по всему кластеру Hadoop. Иногда при работе с большими данными интерактивные инструменты не обеспечивают необходимые для принятия решений качество или скорость предоставления результатов анализа. В таких случаях требуется создавать распределенные приложения для работы с большими данными. Zookeeper — это используемый в Hadoop способ координации всех элементов таких распределенных приложений.

Сервис Zookeeper — несложное по сути технологическое решение, однако его функциональные возможности весьма значительны. Пожалуй, без него не удалось бы создавать отказоустойчивые распределенные приложения Hadoop. Приведем некоторые возможности Zookeeper.

- Синхронизация процессов. Zookeeper координирует запуск и остановку узлов в кластере, что гарантирует выполнение процессов об-

работки в нужном порядке. Последующий процесс начинается исключительно после того, как завершены все предыдущие.

- Управление конфигурацией. Zookeeper можно использовать для отправки атрибутов конфигурации на все узлы кластера или настройки конкретного узла. Если обработка зависит от определенных ресурсов, доступных на всех узлах, Zookeeper обеспечивает согласованность конфигураций.
- Самостоятельное определение ведущего узла. Zookeeper имеет информацию о структуре кластера и может назначить ведущую роль одному из узлов. Этот ведущий узел обрабатывает все запросы клиентов и распределяет их в кластере. В случае сбоя на ведущем узле из числа оставшихся узлов выбирается новый ведущий узел.
- Надежный обмен сообщениями. Несмотря на то что рабочие нагрузки в Zookeeper слабо связаны, для работы распределенных приложений может потребоваться наличие связи между узлами кластера. Zookeeper реализует механизм «публикация-подписка», который позволяет создать очередь. Он гарантирует доставку сообщений даже в случае отказа узла.

Так как Zookeeper управляет группами узлов, обслуживая одно распределенное приложение, то лучше всего его развертывать на нескольких стойках сразу — в отличие от кластеров, к которым предъявляется требование размещения в пределах одной стойки. Объяснение этому очень простое: для работы Zookeeper нужны производительность, отказоустойчивость и сохранение работоспособности, обеспечиваемые на надкластерном уровне. Поскольку кластер Hadoop уже обладает высокой отказоустойчивостью и способен восстанавливаться, Zookeeper требуется дополнительно обеспечить только собственную отказоустойчивость.

Экосистема Hadoop и поддерживаемые коммерческие версии постоянно развиваются: вводятся новые инструменты и технологии, существующие технологии улучшаются, на место устаревшим приходят более совершенные. В этом и заключается одно из самых замечательных преимуществ открытого системного кода. Другим преимуществом является использование технологий открытого исходного кода коммерческими компаниями. Они улучшают такие продукты и делают их более удобными для пользователей, обеспечивая техническую поддержку и обслуживание по приемлемой стоимости. Именно так развивается экосистема Hadoop, поэтому она является хорошим решением для работы с большими данными.

Глава 11

Программно-аппаратные комплексы и хранилища больших данных

В ЭТОЙ ГЛАВЕ:

Особенности хранилища больших данных
Создание новой модели, соответствующей меняющимся требованиям
Гибридная система хранилища больших данных
Развитие моделей хранилищ больших данных

Концепция хранилища данных появилась тридцать лет назад. Хранилища данных позволили решить серьезные проблемы, с которыми сталкивались клиенты, использовавшие несколько различных изолированных операционных систем. Кроме того, со временем требовалось заменять неэффективные системы обеспечения принятия решений более простыми и удобными. Возникла необходимость в единой архитектурной модели, которая значительно упростила бы принятие решений в бизнесе. Поэтому такой подход к хранению, воплощенный в хранилищах данных и более простых витринах данных, стал основным. Однако с развитием больших данных концепция хранилища данных стала меняться в соответствии с новыми задачами. Это не означает, что традиционные хранилища данных уходят в прошлое — они будут по-прежнему широко использоваться для анализа накопленных операционных данных, что необходимо для принятия решений. Однако новые типы хранилищ данных будут специально приспособлены для решения задач, связанных с использованием больших данных. В настоящей главе рассказывается о тенденциях в развитии хранилищ данных, учитывающих специфику больших данных.

Интеграция больших данных и традиционных хранилищ

В отличие от традиционных рабочих систем и приложений баз данных, хранилище данных использовалось сотрудниками компаний и финансовыми аналитиками для принятия стратегических решений. Для этого требовалось собрать данные из различных источников реляционных баз данных, затем обеспечить согласованность метаданных, а также очистку и надлежащую интеграцию самих данных. Билл Инмон, который считается создателем концепции современного хранилища данных, сформулировал ряд принципов построения хранилища данных. Приведем некоторые из них.

- Хранилище данных должно быть субъектно-ориентированным.
- Оно должно быть организовано таким образом, чтобы все соответствующие события были связаны.
- Должна быть гарантирована сохраняемость информации, чтобы ее нельзя было изменить непреднамеренно.
- Хранилище данных должно обеспечивать сохранение информации из всех используемых в работе источников. Информация должна храниться таким образом, чтобы обеспечивалась согласованность и актуальность.

Оптимизация хранилища данных

Хранилища данных традиционно поддерживали структурированные данные и были тесно связаны с операционными и транзакционными системами предприятия. Однако в настоящее время эти отработанные элементы и схемы взаимодействия претерпевают значительные изменения — предприятия стремятся расширить и модифицировать хранилище данных, приспосабливая их к новым требованиям, предъявляем большими данными. Хотя технологии больших данных и хранилищ данных пересекаются, до объединения им еще очень далеко. Традиционное хранилище данных можно воспринимать как систему учета данных для бизнес-аналитики, подобно системам управления взаимоотношениями с клиентами (CRM) или бухгалтерским системам. Это высокоструктурированные системы, оптимизированные для решения специфических задач. Кроме того, эти системы записей являются высоко централизованными. На рис. 11.1 показана типичная схема обработки потоков данных в хранилищах и витринах данных.

Отличие структур хранилищ данных от больших данных

Бесспорно, хранилища данных и далее будут применяться для управления структурированными и операционными данными, выступая в некотором роде в качестве систем учета данных. К ним по-прежнему будут обращаться бизнес-аналитики, которым требуется рассматривать ключевые данные, исторические тенденции и т.д. Однако развитие больших данных не только заставило пересмотреть роль хранилищ данных — потребовалось дополнить традиционный подход новыми методами. В настоящее время отношения между хранилищем данных и большими данными развиваются в направлении их объединения в гибридную структуру. В этой гибридной модели высокоструктурированные оптимизированные операционные данные находятся в очень хорошо контролируемом хранилище данных, в то время как распределенные данные, которые могут меняться в режиме реального времени, управляются инфраструктурой на основе Hadoop (или аналогичной NoSQL).

Операционные и структурированные данные неизбежно будут взаимодействовать в мире больших данных, где источники информации

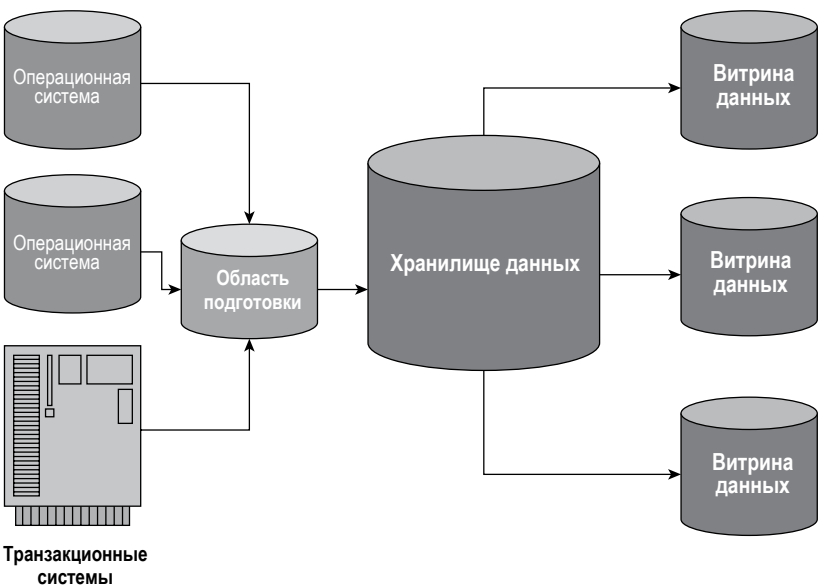


Рис. 11.1. Потоки данных в традиционных хранилищах и витринах данных

не всегда проходят очистку и профилирование. Кроме того, в бизнесе все чаще возникает потребность комбинировать данные из традиционных хранилищ, являющиеся источниками проверенной архивной информации, с большими данными из менее структурированных и менее регулируемых источников. Решать подобные бизнес-задачи позволяет гибридный подход, который обеспечивает работу как с традиционными хранилищами, так и с источниками больших данных.

Пример гибридного процесса

Представим себе управление данными веб-портала туристической компании. Компания предоставляет широкий спектр услуг, включая авиаперелеты, круизы, бронирование номеров в гостиницах и санаториях и многое другое, причем все эти услуги предоставляются различными способами. Например, имеется общедоступный веб-сайт, где размещаются отзывы о поездках, о гостиницах и т.д. Этот веб-сайт связан с сайтами партнеров — страховых компаний, экскурсионных бюро и других организаций, предоставляющих услуги в сфере туризма. Существуют также специализированные сайты, посвященные разным странам. Кроме того, для крупных компаний действуют корпоративные программы. Очевидно, что этой туристической компании необходимо управлять огромным объемом данных, по-разному представляя их в зависимости от того, кто получает к ним доступ. Для отслеживания транзакционных и операционных данных компания использует хранилище данных. Однако хранилище данных не отслеживает веб-трафик, поэтому компания применяет также инструменты веб-аналитики, которые позволяют оценивать взаимодействие с клиентами.

Например, какой элемент сайта выбрал клиент, щелкнув на нем мышкой? Какие предложения предоставлялись различным клиентам и какие предложения были выбраны? Была ли стоимость решающим фактором? Пользовались ли клиенты возможностью самостоятельно формировать туры или предпочитали приобретать готовые программы? Какие направления были наиболее популярны? Какие партнеры приносят наибольший доход?

Такая информация могла оказаться очень полезной для стратегического планирования, однако хранить все эти данные или хотя бы большую их часть в хранилище данных было бы слишком трудоемко и затратно, поэтому почти все такие данные компания удаляла после

ознакомления. Со временем стало ясно, что для более четкого понимания тенденций и нюансов бизнеса эти данные желательно сохранять.

Подразделение компании, отвечающее за управление информацией, решило не создавать специального хранилища для всех таких данных, а использовать распределенные вычисления Hadoop, выполняемые на стандартных серверах. Теперь компания имеет возможность сохранять все данные о взаимодействии клиентов с веб-сайтом. Они хранятся в большом массиве серверов, на которых работают Hadoop и MapReduce. С помощью Flume и Sqoop данные можно перемещать в Hadoop и обратно и передавать в реляционную модель, чтобы формировать к ним запросы с помощью знакомых инструментов SQL.

Теперь компания может быстро менять предложения, если станет понятно, что у определенного круга клиентов появились новые потребности. Кроме того, стало возможным прогнозировать изменения в тарифах на авиаперелеты и учитывать их в ценообразовании пакетов услуг. Часть данных остается в среде платформы Hadoop и обновляется в режиме, близком к реальному времени. Другие элементы данных проходят очистку и перемещаются в хранилище данных, где их можно сравнивать с архивными данными о клиентах и партнерах. Имеюще-

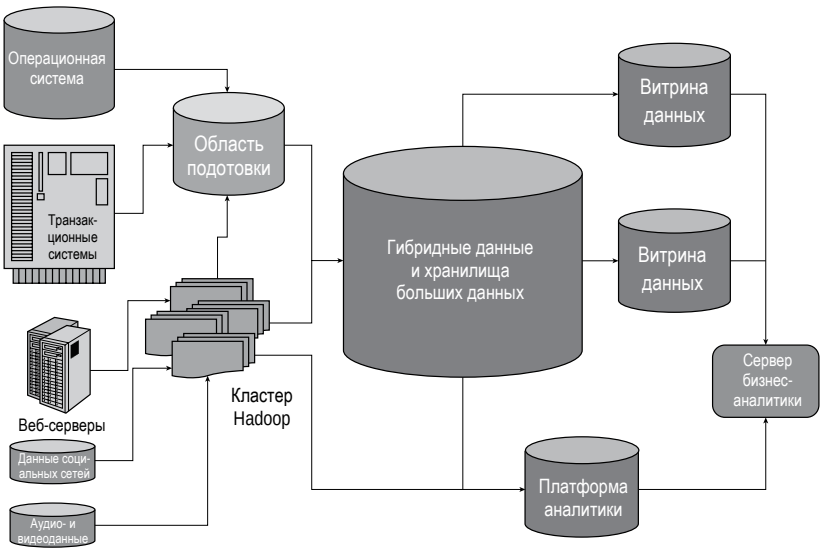


Рис. 11.2. Потоки данных в хранилищах и витринах больших данных

еся хранилище данных позволяет увидеть общую картину, в то время как среда Hadoop отслеживает актуальные процессы. Таким образом, сочетание традиционного подхода с динамической системой больших данных дает компании ценнейшую возможность использовать для дальнейшего развития информацию, полученную из анализа огромных объемов данных, которые генерируются на ее интернет-ресурсах. На рис. 11.2 приведен пример схемы гибридизации традиционного хранилища данных и хранилища больших данных.

Анализ больших данных и хранилище данных

Итак, объединение возможностей хранилища данных и среды больших данных способно дать компании значительные преимущества. Для взаимодействия больших данных с хранилищем данных требуется создать гибридную среду. Прежде всего необходимо учесть, что технология хранилища данных в ее современной форме в ближайшее время не изменится. Поэтому разумно использовать хранилище данных по прямому назначению, то есть помещать в него хорошо проверенные данные, необходимые для анализа. Это могут быть сведения об определенной линейке продукции компании, о клиентах и поставщиках компании, а также сведения о транзакциях за целый год. Информация в хранилище данных или витрине данных должна быть хорошо организована и обладать корректными метаданными. По мере роста объема веб-контента необходимо иметь возможность анализировать этот огромный объем информации и сопоставлять его с архивными данными. Именно для таких задач эффективно использовать гибридную модель.

Некоторые аспекты взаимодействия данных из хранилища с большими данными довольно просты. К примеру, многие источники больших данных содержат собственные исчерпывающие метаданные. Сайты электронной коммерции работают с четко определенными элементами данных (клиент, цена и т.д.). Поэтому анализ данных из хранилища совместно с данными из источника больших данных зачастую предполагает работу с двумя наборами данных, в каждом из которых уже представлена организованная модель метаданных, так что их несложно соотнести друг с другом.

Конечно, в некоторых случаях источник информации содержит недостаточно метаданных, и прежде чем аналитик сможет совместить

архивные транзакционные данные с менее структурированными большими данными, придется провести подготовительную работу. Зачастую предварительный анализ нескольких петабайт данных уже выявляет интересные закономерности, позволяющие спрогнозировать изменения в бизнесе или уточнить диагноз пациента. Предварительный анализ можно провести с помощью таких инструментов, как MapReduce на платформе распределенной файловой системы Hadoop. Результат позволит понять, будет ли полезна выбранная информация для решения рассматриваемой задачи. Для анализа одинаково важно и исключить нерелевантные данные, и выделить те, что непосредственно относятся к решаемой задаче. После завершения этого этапа оставшиеся данные необходимо преобразовать, чтобы определения метаданных были точными. Тогда объединение больших данных с традиционными архивными данными из хранилища данных даст значимые результаты.

Основа интеграции

Чтобы описываемый процесс можно было использовать на практике, необходима хорошо определенная стратегия интеграции данных. Более подробно вопросы интеграции данных рассматриваются в главе 15. Интеграция данных очень важна как при управлении большими данными, так и при гибридном анализе, для которого используется хранилище данных. В целом процесс извлечения и преобразования данных в гибридной среде мало отличается от такового в традиционном хранилище данных. К примеру, когда в хранилище данных передаются данные из системы управления взаимоотношениями с клиентами (CRM) и системы управления ресурсами (ERP), элементы данных из этих разноплановых систем также необходимо правильно согласовать.

Извлечение, преобразование и загрузка данных в новом контексте

В хранилище данных помещают и таблицы реляционных баз данных, и плоские файлы, и нереляционные источники. Хорошо структурированное хранилище данных должно предусматривать преобразование данных в единый формат — это необходимо для точной и согласован-

ной обработки запросов. Сохраняемые файлы должны преобразовываться в соответствии с бизнес-правилами и процессами той предметной области, для анализа которой предназначено хранилище данных. Например, часто в хранилище данных в качестве вычисляемого поля хранится информация о покупной цене, поскольку она нередко фигурирует в запросах руководителей. В хранилище данных могут быть предусмотрены процессы проверки вычисляемых результатов и оценки их соответствия требованиям бизнеса. Эти принципы, которые для организации хранилища данных являются основополагающими, применяются и в концепции сочетания больших данных и хранилища данных. Другими словами, извлечение из разных источников и совместное использование элементов больших данных должно обеспечивать безопасность обработки и получение нужных результатов. Кроме того, источники нужно преобразовывать, чтобы они позволяли анализировать отношения между архивными и динамически изменяющимися в режиме реального времени большими данными.

Загрузка информации в модели больших данных имеет свою специфику. В модели традиционного хранилища загруженные данные после кодификации уже не изменяются, и для анализа предоставляется выборка данных, связанных с конкретной бизнес-задачей (например, информация о запасах товара на складе или запланированных объемах продаж). Поскольку большие данные обладают распределенной структурой, при работе с ними загрузка осуществляется иначе: сначала данные загружают на ряд узлов, а затем выполняют их извлечение и преобразование. В гибридной модели, сочетающей традиционное хранилище данных и большие данные, распределенность среды больших данных способна существенно повлиять на возможность анализа огромных объемов данных.

Изменение роли хранилища данных

У управления данными в традиционном и гибридном хранилище есть и общие черты, и различия.

Сходство двух методов управления данными выражается в следующих требованиях:

- наличие общих определений данных;
- извлечение и преобразование ключевых источников данных;
- соответствие данных бизнес-процессам и правилам.

Различия же таковы:

- в работе гибридной модели ключевую роль играет распределенная модель вычислений, характерная для больших данных;
- в гибридной модели анализ больших данных имеет приоритет, а данные из традиционного хранилища используются в качестве дополнительной информации (историческая перспектива, статистика транзакций).

Изменение моделей внедрения в эпоху больших данных

По мере распространения больших данных меняются модели внедрения новых средств управления данными. Как правило, традиционное хранилище данных развернуто в одной большой системе центра обработки данных. Для снижения затрат, связанных с этой моделью внедрения, хранилища обычно оптимизировали и ограничивали область и объемы помещаемых туда данных. Но для использования огромных объемов информации из источников больших данных ограничения традиционных моделей уже неприемлемы. Поэтому программно-аппаратный комплекс, предназначенный для хранения данных, стал удобным решением для создания оптимизированной среды при переходе к новому способу управления информацией.

Модель программно-аппаратного комплекса

Когда необходимо объединить структуру корпоративного центра обработки данных с большими данными, использование модели программно-аппаратного комплекса позволяет решить проблему масштабирования. Как правило, программно-аппаратный комплекс — это интегрированная система, которая состоит из оборудования (обычно размещенного в стойке), оптимизированного для хранения управления данными. Поскольку такие программно-аппаратные комплексы являются независимыми и самодостаточными, их относительно просто и быстро развертывать, а их эксплуатация и обслуживание обходятся недорого. Такая система уже имеет предустановленные реляционную базу данных, платформы Hadoop, MapReduce, программное обеспечение, включающее средства загрузки и обработки данных,

и аналитические модули и инструменты, которые позволяют упростить анализ данных из различных источников. Интерфейсы программно-аппаратного комплекса, как правило, существенно облегчают подключение к существующему хранилищу данных.

Облачная модель

Облако все чаще применяется для управления большими данными. В сочетании с локальными средами облако может использоваться для создания гибридной среды. Благодаря инновационным подходам к загрузке и передаче данных облачная среда становится гибкой платформой для хранения больших данных. Например, компания Aspera, специализирующаяся на быстром переносе данных между сетями, в рамках партнерского соглашения с Amazon.com предоставляет услуги по управлению облачными данными. Другие поставщики решений, такие как FileCatalyst и Expedition, также сосредотачивают свое внимание на этом рынке. В сущности, эта технология оптимизирует работу сети для уменьшения задержек при перемещении файлов. Можно предположить, что со временем стандартом станет хранение систем больших данных в облаке, взаимодействующем с хранилищами данных, как облачными, так и расположенными в локальных центрах обработки данных.

Будущее хранилищ данных

Появление больших данных стимулировало значительные изменения на рынке хранилищ данных. В прошлом было экономически невыгодно хранить большие объемы архивных данных из разных систем. Отсутствие эффективных с экономической точки зрения архитектур распределенных вычислений означало, что хранилище данных должно было строиться как единая большая система. Поэтому хранилища данных создавались и оптимизировались для решения конкретного круга задач. Кроме того, хранилища данных предусматривали тщательную проверку, чтобы гарантировать четкость определения и управления данными. Благодаря этому хранилища данных стали эффективным источником надежных данных. Однако контроль и требования к надежности ограничили возможности использования хранилища как среды работы с источниками динамически изменяющихся больших данных.

Несомненно, в перспективке хранилища данных приспособятся к работе с большими данными, но происходить это будет медленно.

Хранилища и витрины данных будут и далее оптимизироваться, чтобы лучше соответствовать потребностям бизнес-анализа. Однако новое поколение таких систем объединит в себе преимущества хранилищ высокоструктурированных и архивных данных с различными возможностями хранения больших данных. Во-первых, хранилища больших данных предоставят возможность анализировать гигантские объемы данных в режиме, близком к реальному времени. Во-вторых, хранилища больших данных смогут согласовывать метаданные результатов анализа больших данных с требованиями традиционных хранилищ данных.

Часть IV

Аналитика и большие данные

Тип анализа	Описание
Базовая аналитика для получения извлечения знаний	Продольные и поперечные срезы данных, отчеты, простая визуализация, базовые возможности мониторинга
Углубленная аналитика для подробного анализа	Более сложный анализ: прогнозное моделирование и другие способы выявления закономерностей
Операционализированная аналитика	Аналитика становится частью бизнес-процесса
Монетизированная аналитика	Аналитика используется непосредственно для повышения прибыли

Анализ больших данных

В ЭТОЙ ЧАСТИ:

- Большие данные для составления отчетов, визуализации и прогнозного анализа
- Примеры структурированной аналитики
- Неструктурированные данные
- Текстовая аналитика
- Интеграция неструктурированных и структурированных данных
- Модели использования больших данных

Глава 12

Что такое аналитика больших данных

В ЭТОЙ ГЛАВЕ:

Как использовать большие данные для достижения результатов
Специфика анализа больших данных
Трудности анализа больших данных
Обзор аналитических инструментов для больших данных

В предыдущих главах мы подробно описали инфраструктуру, необходимую для поддержки инициатив, связанных с большими данными. Но большие данные приносят пользу, когда есть возможность их обрабатывать. Итак, каким же образом анализировать большие данные?

Такие компании, как Amazon и Google, имеют большой успешный опыт анализа больших данных и научились извлекать из них конкурентные преимущества. Достаточно посмотреть на механизм рекомендаций Amazon: сведя воедино историю покупок клиента, доступную информацию о нем и его покупательских привычках и обработав ее вместе с информацией о других клиентах с похожими предпочтениями, интернет-магазин способен предложить конкретному клиенту то, что его с наибольшей вероятностью заинтересует. Это маркетинговый механизм, который чрезвычайно эффективно функционирует благодаря результатам анализа больших данных.

Анализ больших данных предоставит уникальные возможности любой организации. Вы сможете расширить спектр доступных вам методов анализа и уже не будете ограничиваться выборками из больших массивов данных — вы сможете анализировать гораздо более полные и более сложные данные. Конечно, анализ больших данных сопряжен и с некоторыми трудностями. Зачастую даже самые простые методы

анализа применительно к большим данным требуют изменения алгоритмов и технологий.

Итак, поговорим об аналитике больших данных. В этой главе описаны разновидности анализа, которые используются с большими данными, и некоторые существенные различия между аналитикой больших данных и традиционной аналитикой. Работе с неструктурированными данными посвящена глава 13, в настоящей же главе сосредоточимся на анализе структурированных данных.

Как использовать большие данные для достижения результатов

Первый вопрос, на который нужно ответить, прежде чем приступить к анализу больших данных, — для чего вам это нужно? Возможно, вы даже не представляете, какую цель преследуете, — просто решили проанализировать имеющиеся данные, понимая, что из них можно получить ценную информацию. Но если вы не знаете, что ищете, вы рискуете оставить без внимания закономерности, выявленные при анализе, потому что не подозреваете об их существовании.

Вы поймете, что какая-то конечная цель обязательно присутствует: спрогнозировать потребительское поведение клиентов, чтобы предотвратить их уход к конкурентам; проанализировать стиль вождения автомобиля клиентами, чтобы точнее рассчитывать страховые выплаты; обработать данные системного журнала, чтобы спрогнозировать возможные технические неполадки. В зависимости от сложности поставленной задачи определяется наиболее подходящий тип аналитики. С другой стороны, при отсутствии конкретной бизнес-задачи можно направить аналитику на проблемные области деятельности компании. Наконец, можно выбрать тот или иной метод анализа (табл. 12.1) и применить его в той сфере, где он будет наиболее эффективен.

Тип анализа	Описание
Базовая аналитика для получения извлечения знаний	Продольные и поперечные срезы данных, отчеты, простая визуализация, базовые возможности мониторинга
Углубленная аналитика для подробного анализа	Более сложный анализ: прогнозное моделирование и другие способы выявления закономерностей

Тип анализа	Описание
Операционализированная аналитика	Аналитика становится частью бизнес-процесса
Монетизированная аналитика	Аналитика используется непосредственно для повышения прибыли

Таблица 12.1. Анализ больших данных



Прежде чем приступить к анализу данных, необходимо провести их предварительную обработку. Эта тема освещается в главе 15.

Базовая аналитика

Базовая аналитика подходит для предварительного изучения данных, когда еще не ясно, как они могут быть применены. Сюда входят простые визуализации и статистическая информация. Базовый анализ часто применяется к большим объемам несопоставимых данных. Вот несколько примеров:

- продольные и поперечные срезы данных. Весь массив данных разбивается на небольшие выборки, которые более доступны для изучения. Например, данные исследования толщи воды в различных точках включают значения разнообразных параметров: температуры, давления, прозрачности, количества растворенного кислорода, уровня pH, минерализации и т.д., — измерявшихся с помощью множества датчиков за некоторый период. Результатом анализа таких данных могут быть простые графики или схемы, которые позволят соотнести разноплановую информацию, например температуру с уровнем pH или прозрачность с минерализацией; или же статистические показатели, например среднее значение или диапазон значений для каждого параметра, каждой точки измерения или каждого периода. По итогам базового анализа можно будет сформулировать более конкретные задачи и запросы. Отличие базового анализа больших данных от традиционной бизнес-аналитики в том, что ввиду огромного объема обрабатываемых данных заранее неизвестно, сколько понадобится ресурсов на выполнение запросов, и при этом обработка, скорее всего, будет производиться в реальном времени;

- базовый мониторинг. Изменения больших объемов данных отслеживаются в режиме реального времени. Так, в примере с характеристиками толщи воды может возникнуть необходимость ежесекундно отслеживать ряд параметров в сотне различных точек и на разной глубине. В результате образуется огромный набор данных. Или, например, нужно отследить рост обсуждения определенного продукта в социальных сетях каждый раз, когда для него запускается рекламная кампания. Причем если в случае с наблюдениями за водой результаты будут относительно структурированными, то мониторинг социальных сетей потребует сбора большого количества разрозненных типов данных из различных источников в Интернете;
- идентификация аномалий. Обнаружение событий, которые не совпадают с ожидаемыми, например необычного поведения оборудования. Аномалии могут свидетельствовать о проблемах в бизнес-процессах, сбоях на производстве и т.п. Например, анализ записей с производственной линии позволит определить, связаны ли возникающие на ней проблемы с конкретным оборудованием или оператором. Здесь могут использоваться простые статистические данные, например вычисление скользящего среднего значения срабатываний аварийной сигнализации оборудования.

Углубленная аналитика

Углубленная аналитика реализует алгоритмы комплексного анализа структурированных и неструктурированных данных. Они включают сложные статистические модели, машинное обучение, нейронные сети, текстовую аналитику (см. главу 13) и прочие методы интеллектуального анализа данных (см. ниже врезку «Что такое интеллектуальный анализ данных?»). Среди областей применения углубленной аналитики поиск тенденций, прогнозирование и обработка сложных событий.

В статистике и математике углубленная аналитика используется уже десятки лет, однако в бизнесе вплоть до недавнего времени не была широко распространена. К слову, 20 лет назад корпоративные специалисты по статистике могли прогнозировать, кто из клиентов откажется от услуг компании, с помощью оценки функции выживания или машинного обучения. Однако на административном уровне значимость такого анализа и его возможности для получения конку-

рентных преимуществ зачастую недооценивали. Кроме того, не существовало доступных вычислительных ресурсов для интерпретации данных, которые со временем менялись.

Сегодня углубленная аналитика применяется все чаще. Рост вычислительных мощностей, улучшенная инфраструктура данных, появление новых алгоритмов создали для этого технические возможности, а с резким увеличением объемов данных их качественный детальный анализ превратился в средство обеспечения значительных конкурентных преимуществ.

Несколько примеров применения углубленной аналитики больших данных:

- прогнозное моделирование. Один из самых распространенных примеров использования углубленной аналитики больших данных. Прогнозная модель — это решение для статистической обработки или интеллектуального анализа данных, которое включает алгоритмы и методы, работающие как со структурированными, так и с неструктурированными данными (совместно или раздельно) и позволяющие решать задачи прогнозирования. Например, телекоммуникационная компания с помощью прогнозной модели определяет, какие клиенты могут отказаться от услуг компании. Большие данные предоставляют множество параметров, которые могут использоваться для прогнозирования, на основе огромного количества наблюдений. В прошлом для выполнения прогнозной модели требовалось несколько часов, если не дней, и это требовало хранения огромного объема данных на компьютере. Теперь при наличии соответствующей инфраструктуры больших данных прогнозную модель за то же время можно циклически выполнить сотни раз;
- текстовая аналитика. Процесс анализа неструктурированного текста, извлечения соответствующей информации и преобразования ее в структурированную информацию, которую можно использовать различными способами. Поскольку неструктурированные данные являются весьма значительной частью больших данных, тестовая аналитика стала очень важным компонентом экосистемы больших данных. Процессы анализа и извлечения данных, которые используются в текстовой аналитике, задействуют методы компьютерной лингвистики, статистики и информатики. Текстовая аналитика применяется во всех видах анализа, начиная от прогнозирования оттока клиентов, анализа социальных сетей и за-

канчивая выявлением мошеннических действий. Текстовая аналитика настолько важна, что мы посвятили ей значительную часть главы 13;

- прочие алгоритмы статистики и интеллектуального анализа данных. Сюда входят расширенное прогнозирование, оптимизация, анализ кластера на сегментацию или даже микросегментацию, а также анализ сходства.

Что такое интеллектуальный анализ данных?

Интеллектуальный анализ данных включает изучение больших объемов данных и поиска в них закономерностей с применением статистических методов, искусственного интеллекта, а также некоторых технологий управления базами данных. В общем, целью интеллектуального анализа данных является либо классификация, либо прогнозирование. При классификации требуется распределить данные по группам. Например, маркетологу нужно определить, какие покупатели отреагировали на рекламную кампанию, а какие остались равнодушны к рекламе. Результатом анализа будет разделение покупателей на два больших класса. При прогнозировании определяется вероятное значение непрерывной (т.е. недискретной) переменной — например, количество тех, кто отреагирует на проводимую рекламную кампанию.

Приведем некоторые типичные алгоритмы, используемые при интеллектуальном анализе данных.

- Деревья классификации. Это популярный способ интеллектуального анализа данных, используемый для классификации категориальной зависимой переменной на основе измерений одной или нескольких переменных прогноза. В результате получается дерево с узлами и ссылками между узлами, которые могут быть использованы для формирования правил «если-то».
- Логистическая регрессия. Это статистический метод, который является вариантом стандартной регрессии, но обеспечивает при этом и классификацию. Формула логистической регрессии позволяет прогнозировать вероятность возникновения состояния в качестве функции независимых переменных.
- Нейронные сети. Это программный алгоритм, который моделирует параллельную архитектуру интеллекта животного. Сеть состоит из входных узлов, скрытых слоев и выходных узлов. Каждой единице назначается вес. Данные предоставляются входному узлу, затем посредством проб и ошибок алгоритм регулирует вес до соответствия определенным предельным критериям. Иногда этот подход называют «черным ящиком», поскольку его внутренние процессы могут не поддаваться наблюдению.

- Методы кластеризации, в частности метод K-ближайших соседей. Это методы, которые определяют группы похожих записей. Метод K-ближайших соседей рассчитывает расстояние между записями и сопоставляет его с историческими (тренировочными) данными, а затем относит классифицируемую запись к классу ближайшего соседа в наборе данных.

Рассмотрим пример дерева классификации. Телефонной компании необходимо определить, кто из клиентов с большой долей вероятности откажется от ее услуг. Информация телефонной компании состоит из следующих атрибутов: как долго абонент пользовался услугой, сколько он потратил на услугу, были ли у него проблемы с услугой, нашел ли он для себя подходящий тарифный план, где он живет, сколько ему лет, пользуется ли он пакетом услуг, которые также включены в его тарифный план, конкурентные сведения о тарифных планах других операторов, продолжает ли он пользоваться услугой или же отключил ее. Конечно, это лишь немногие из возможных атрибутов. Последним атрибутом будет выходная переменная, которую программное обеспечение использует для распределения клиентов по двум группам — условно назовем их «лояльные» и «рискованные».

Затем массив данных разбивается на данные для обучения (тренировочные) и тестовый набор данных. Данные для обучения состоят из наблюдений (атрибутов) и выходной переменной (в случае модели классификации она будет двоичной), в данном примере принимает значения «лояльные» и «рискованные». Алгоритм применяется к данным для обучения, после чего формируется дерево, которое можно воспринимать как последовательность правил. Например, если клиент в течение десяти лет принадлежал к «лояльным» и сейчас старше 55, скорее всего, он и останется «лояльным».

Затем эти правила применяются к тестовому набору данных, чтобы проверить, насколько модель применима к «новым данным». Точность прогноза модели определяется, например, с помощью матрицы неточностей. Эта матрица представляет собой таблицу, в которой содержатся сведения о том, сколько случаев было классифицировано корректно, а сколько нет. Если точность модели приемлема, ее можно применять к другим данным по мере их поступления (то есть прогнозировать новые случаи отказа от услуги). По результатам применения модели компания может, к примеру, сделать специальные предложения клиентам, которые попали в категорию «рискованных».



Углубленная аналитика не обязательно требует больших данных. Однако применение углубленной аналитики к большим данным может принести ценные результаты.

Операционализированная аналитика

Операционализированная аналитика — это аналитика, которая стала составной частью бизнес-процессов. Например, специалисты по статистике построили для страховой компании модель, которая прогнозирует вероятность того, что заявление на получение страховой выплаты подано с целью мошенничества. Такая модель, наряду с некоторыми правилами принятия решений, может быть включена в систему обработки заявлений. Это позволит выделить заявления, которые с большой долей вероятности будут подложными, чтобы направить их в отдел расследования для более детального изучения. В других случаях конечный пользователь не взаимодействует с моделью непосредственно. Например, в центре обработки вызовов применяется модель, которая определяет клиентов, склонных к увеличению объема покупки. Оператор центра обработки вызовов, приняв вызов клиента, получает системное сообщение о дополнительных продуктах, которые можно предложить этому клиенту. Оператор при этом может и не знать о существовании прогнозной модели.

Монетизированная аналитика

Аналитика позволяет оптимизировать бизнес, принимать более обоснованные решения и, как результат, увеличивать итоговую и базовую прибыль. Однако аналитика больших данных может использоваться для увеличения прибыли напрямую. Сформированный в результате анализа уникальный массив данных может заинтересовать другие организации: например, компании-эмитенты кредитных карт и финансовые учреждения продают аналитические продукты, созданные на основе обработанных ими данных. Телекоммуникационные компании предлагают предприятиям розничной торговли информацию о потребительском поведении в различных районах — дело в том, что его можно спрогнозировать, обрабатывая по отдельности или в комплексе данные о платежах, местоположении, текстовых сообщениях, просмотре веб-страниц и т.д. Разумеется, такие исследования должны осуществляться в соответствии с законодательством.

Модификация продуктов бизнес-аналитики для обработки больших данных

Традиционные продукты бизнес-аналитики изначально не предназначены для обработки больших данных. Они ориентированы на работу с хорошо структурированными и понятными данными, которые хранились в репозитории реляционных данных и отображались на экране персонального компьютера. Традиционная бизнес-аналитика, как правило, применяется к отдельным выборкам данных, а не ко всему доступному объему данных. Что же меняется, когда эти методы используются для анализа больших данных?

Данные

Как уже отмечалось в главе 2, большие данные состоят из структурированных, слабоструктурированных и неструктурированных данных. Объем этих данных может быть огромным, а сами данные — достаточно сложными. Вот основные факторы, которые следует иметь в виду:

- данные могут поступить из источников, не заслуживающих доверия. Анализ больших данных часто включает агрегацию данных из различных источников, как внешних, так и внутренних. Проблема обычно заключается в невозможности проверить надежность внешних источников информации. Достоверна ли информация из сообщения в Twitter? Она может быть основана на непроверенных сведениях. Кроме того, необходимо учитывать целостность данных. Подробнее о безопасности данных и управлении ими рассказано в главе 19;
- данные могут быть некорректными. Некорректные данные — это неточные, неполные или ошибочные данные: от текста с орфографическими ошибками до показаний плохо откалиброванного датчика, а также дублированные данные. До сих пор нет единого подхода к тому, как следует «очищать» данные: в пределах какого-либо интервала или все в режиме реального времени. Есть также мнение, что некорректные данные вообще не следует очищать, так как они могут содержать интересные аномалии. Стратегия очистки определяется источником, типом данных и целью анализа. Например, при разработке фильтра против спама важно определять подозрительные элементы данных, поэтому «чистить» их не следует;

- соотношение «сигнал-шум» может быть низким. Иными словами, полезная информация может составлять лишь малую долю имеющихся данных, а все остальное будет «шумом». Преимущество анализа больших данных как раз в том, что он способен извлечь из «шумных» данных мельчайший сигнал — но этот сигнал может оказаться действительно незначительным;
- данные могут поступать в режиме реального времени. Во многих случаях приходится анализировать потоки данных в режиме реального времени. Особенности анализа таких данных описаны в главе 16.



Управление большими данными — важная часть процесса аналитики. Для повышения эффективности бизнес-аналитики потребуются более совершенные решения по управлению данными, которые обеспечат достоверность данных, взятых из новых источников. Это особенно важно, если такие данные объединяются с имеющимися достоверными данными, которые размещены в хранилище. Также понадобится усовершенствовать обеспечение безопасности и конфиденциальности данных. Это позволит лучше управлять большими данными, которые хранятся с использованием новых технологий. Управлению данными и их безопасности посвящена вся глава 19.

Аналитические алгоритмы

Когда аналитика выходит за пределы одного компьютера — как при работе с большими данными, — используемые алгоритмы могут потребовать доработки, однако такие внутренние изменения не должны причинить ущерба внешним функциональным возможностям. Достоинство инфраструктуры больших данных заключается в том, что она позволяет за считаные минуты обработать модель, выполнение которой ранее занимало часы и даже дни. Появляется возможность циклически применять модель сотни и сотни раз. Однако при регрессии миллиардов строк данных в распределенной среде необходимо помнить о требованиях к ресурсам, связанных с объемом данных и их расположением в кластере. Алгоритмы должны учитывать различные особенности обрабатываемых данных.

В настоящее время появляются новые аналитические возможности, которые способны обрабатывать большие данные непосредственно в источниках, не требуя предварительного их сохранения. Такой подход

позволяет сохранять только значимые данные, а также анализировать информацию на более ранних этапах, обнаруживая ключевые события заблаговременно, что очень важно для принятия решений в режиме реального времени. Более подробно эти методы описаны в главе 14.

Конечно, аналитика будет развиваться и в дальнейшем. Например, для отображения постоянно меняющихся данных могут потребоваться возможности визуализации в режиме реального времени. Но как на практике построить график, состоящий из миллиардов точек? Или, к примеру, как без потерь в скорости и качестве анализа применять алгоритмы прогнозирования к постоянно растущему набору сложных данных? Это лишь некоторые перспективные направления разработок в области аналитики.

Поддержка инфраструктуры

Инфраструктура, необходимая для поддержки больших данных, подробно описана в главе 4. Отметим только ключевые требования к платформе для работы с большими данными:

- Интеграция технологий. Инфраструктура должна интегрировать новые технологии больших данных с традиционными технологиями. Это позволит обрабатывать все виды больших данных и делать их пригодными для традиционных методов анализа.
- Хранение больших объемов разрозненных данных. Для этого может применяться система Hadoop корпоративного уровня, способная обрабатывать, хранить и контролировать большие объемы структурированных, слабоструктурированных и неструктурированных данных.
- Обработка перемещаемых данных. Датчики, интеллектуальные устройства, аудио- и видеоустройства, журналы систем постоянно генерируют большие потоки перемещаемых данных, которые могут быть использованы в процессах обеспечения принятия решений в режиме реального времени.
- Данные в хранилище. Решение, оптимизированное для операционных и пиковых рабочих нагрузок, позволит анализировать и сохранять растущие объемы заслуживающих доверие данных, а также управлять этими данными.

И конечно, необходима возможность интеграции уже имеющихся данных с результатами анализа больших данных.

Примеры аналитики больших данных

Аналитика больших данных находит разнообразное применение. Рассмотрим примеры некоторых интернет-компаний и других организаций.

Orbitz

www.orbitz.com — один из наиболее популярных в США веб-сайтов, связанных с туризмом и путешествиями. Компания Orbitz была основана в 1999 году, веб-сайт начал действовать в 2001 году. Ежедневно пользователи Orbitz выполняют более миллиона поисковых запросов, а компания собирает сотни гигабайтов необработанных данных. Специалисты компании решили, что анализ содержимого лог-файлов, собранных с помощью инструментов веб-аналитики, может дать ценную информацию о взаимодействии потребителей с сайтом.

В частности, компания была заинтересована в том, чтобы на основании анализа поведения пользователей определить наилучший способ отображения результатов поиска гостиниц, позволивший бы увеличить частоту бронирований. До начала этого исследования Orbitz не использовала эти собранные данные из-за высоких затрат на их хранение. Специалисты компании внедрили системы Hadoop и Hive, которые были развернуты на стандартном оборудовании. Платформа Hadoop обеспечила распределенную файловую систему, а система хранения Hive позволила использовать интерфейс типа SQL. В несколько этапов данные были перенесены в Hive; затем были применены методы машинного обучения, т.е. подход, ориентированный на данные (и интеллектуальный анализ данных, см. врезку в начале этой главы), чтобы выявить тенденции, скрытые в этих данных, и обеспечить возможность их анализа. Подробнее о Hadoop и Hive рассказано в главах 9 и 10.

Nokia

Nokia производит устройства беспроводной связи и оказывает услуги в этой отрасли. Эта компания считает данные стратегическим активом. Служба анализа больших данных компании использует платформу, которая выполняет десятки тысяч заданий и обрабатывает петабайты данных ежедневно. Сюда входит использование углубленной

аналитики терабайтов потоковых данных, например, для того, чтобы понять, как пользователи работают с приложениями Nokia — какие функции наиболее популярны, каким образом используют те или иные возможности, как пользователи переключаются между различными функциями, не слишком ли сложно устроены приложения. Такой уровень детализации позволяет компании внедрять новые функции в свои приложения и удерживать клиентов.

NASA

NASA использует прогнозные модели для анализа данных о безопасности на воздушных судах, чтобы определить, как на нее может повлиять внедрение той или иной новой технологии. Бесспорно, NASA имеет дело со значительными объемами данных. Ежесекундно каждый самолет записывает тысячи параметров, причем некоторые данные передаются в потоковом режиме. Компания также получает текстовые данные из отчетов, составленных пилотами и членами экипажей. Кроме того, в анализ включаются метеорологические данные (которые меняются в пространстве и во времени). Таким образом точность прогнозирования значительно повышается.

Решения для анализа больших данных

Сегодня на рынке представлен целый ряд решений для работы с большими данными, в том числе от известных производителей и поставщиков.

- IBM (www.ibm.com) использует корпоративный подход к большим данным и межплатформенной интеграции, в том числе встраивание и пакетное предоставление возможностей аналитики. Продукты этой компании включают хранилище данных (InfoSphere), которое имеет собственные встроенные возможности интеллектуального анализа данных и построения OLAP-кубов. Новые системы PureData (набор технологий углубленной аналитики для встраивания в интегрированные платформы) включают различные аналитические методы, входящие в единый пакет. Продукт InfoSphere Streams тесно интегрирован со статистическим программным обеспечением Statistical Package for the Social Sciences (SPSS) (статистический пакет для общественных наук) и поддерживает прогноз-

ный анализ в режиме реального времени, включая возможность динамического обновления моделей на основе данных режима реального времени. В комплекте предоставляется лицензия на Cognos Business Intelligence с соответствующими ключевыми возможностями платформы больших данных (Hadoop корпоративного уровня, потоковые вычисления и решения хранилища данных).

- SAS (www.sas.com) обеспечивает комплексный подход к анализу больших данных благодаря использованию собственной высокопроизводительной инфраструктуры аналитики и статистического программного обеспечения. SAS предоставляет несколько возможностей распределенной обработки, в число которых входит аналитика непосредственно в базе данных, аналитика в памяти и сетевые распределенные вычисления. Системы могут развертываться как на локальных ресурсах, так и в облаке.
- Tableau (www.tableausoftware.com), компания, выпускающая программное обеспечение для бизнес-аналитики и визуализации данных, предоставляет свои возможности визуализации для работы на лучших программно-аппаратных комплексах и других инфраструктурных решениях, предоставляемых целым рядом партнеров, включая Cirro, EMC Greenplum, Karmasphere, Teradata/Aster, HP Vertica, Hortonworks, ParAccel, IBM Netezza и др.
- Oracle (www.oracle.com) предоставляет целый ряд инструментов, дополняющих основную платформу больших данных, которая называется Oracle Exadata. В их число входит углубленная аналитика с использованием языка программирования R, а также возможность размещения базы данных в памяти благодаря решению «компьютер в памяти» Oracle Exalytics и хранилищу данных Oracle. Хранилище данных Exadata интегрировано с аппаратной платформой.
- Pentaho (www.pentaho.com) предоставляет решение для бизнес-аналитики с открытым кодом; имеются общедоступная и корпоративная версии. Pentaho поддерживает ведущие дистрибутивы на основе Hadoop, а также соответствующие возможности, такие как высокопроизводительная файловая система MapR NFS.

Глава 13

Обзор текстовой аналитики в контексте больших данных

В ЭТОЙ ГЛАВЕ:

Типы неструктурированных данных
Что такое текстовая аналитика
Примеры использования неструктурированного анализа
Размещение неструктурированных данных вместе со структурированными
Инструменты текстовой аналитики больших данных

Большинство данных относится к неструктурированным. Неструктурированные данные включают как информацию, которая хранится на внутренних ресурсах компании (например, документы, электронная переписка, корреспонденция клиентов), так и данные из внешних источников, которые могут быть полезны компании (сообщения в Twitter, блоги, видео на YouTube, спутниковые фотографии). Объем и разнообразие таких данных стремительно растут, и в бизнесе все чаще возникает потребность анализировать большие объемы данных, чтобы понять актуальные и перспективные тенденции.

В то время как анализ изображений и видео пока находится на раннем этапе развития и еще не получил широкого распространения, текстовая аналитика уже стала одной из основных технологий обработки больших данных. Рассмотрим пример использования текстовых данных для поддержки процесса принятия бизнес-решений. Крупный автоконцерн задался целью устранить проблемы с качеством автомобилей. Оказалось, что анализ текстовых сообщений партнерских автосервисов позволяет идентифицировать проблемы, связанные с ка-

чеством автомобилей, поступающих на рынок. Такой анализ служит системой раннего предупреждения: чем раньше компания обнаружит проблему, тем больше у нее возможностей внести изменения в производственный процесс и тем самым повысить удовлетворенность клиентов. До того как использовать текстовую аналитику, концерн в течение многих лет обрабатывал информацию из своих бизнес-систем, включая номера деталей и коды дефектов. Этот способ давал хорошие результаты, но только применительно к ситуациям, о которых руководство уже знало. Сложившаяся система не позволяла выявить скрытые проблемы, которые были хорошо известны сотрудникам, непосредственно общавшимся с клиентами.

Огромная разница! Текстовая аналитика применяется в работе с большими данными в самых разных областях: от анализа социальных сетей до анализа гарантийных случаев и выявления мошенничества. Кроме того, компании все чаще начинают анализировать объединенные структурированные и неструктурированные данные, чтобы получить всестороннюю оценку ситуации. В этой главе мы рассмотрим особенности текстовой аналитики и примеры ее использования, в том числе для объединения неструктурированных и структурированных данных, и расскажем о некоторых поставщиках инструментов текстовой аналитики для больших данных.

Изучение неструктурированных данных

Неструктурированные данные отличаются от структурированных тем, что их структура непредсказуема. Как уже упоминалось в главе 2, некоторые считают термин «неструктурированные данные» некорректным, поскольку каждый документ имеет свою специфическую структуру или форматирование, определяемые программным обеспечением, в котором он создан, и фактически неструктурированным является только содержание документа.

Вот типичные примеры текстовых данных.

- Документы:
  2.1. За пользование займом Заемщик уплачивает Заимодавцу 15% (пятнадцать процентов) годовых от фактической суммы срочной задолженнности в долларах США или ставку, установленную в соответствии с п. 2.6.

За просроченную задолженность Заемщик уплачивает 20% (двадцать процентов) годовых в долларах США от фактической суммы неуплаченного в срок долга.
2.2. Проценты по полученному займу уплачиваются Заемщиком ежемесячно…

- Электронные сообщения:
  Сэм, привет! Как там глава для книги? В пятницу крайний срок. Джоан.
- Файлы журналов:
  222.222.222.222- — [08/Oct/2012:11:11:54–0400] "GET / HTTP/1.1" 200 10801 "http://www.google.com/search?q=log+analyzer&ie=…..
- Сообщения в Twitter:
  #Большие_данные — наше все!
- Посты в Facebook:
  ИМХО, кино так себе, но за наводку спс!:)

Очевидно, что некоторые текстовые данные более структурированы, чем другие. Например, договор займа имеет общепринятый шаблон и следует определенному стилю. В электронном письме также прослеживается некоторая структура. Сообщение в Twitter или Facebook может использовать специальные символы и сокращения. Файл журнала также может быть по-своему структурирован.

Как же анализировать эти разноплановые неструктурированные текстовые данные?

Общие сведения о текстовой аналитике

Существуют различные методы анализа неструктурированных данных. Исторически сложилось так, что эти методы пришли из технических областей, таких как обработка естественного языка (NLP), обнаружение знаний, интеллектуальный анализ данных, получение данных и статистика. Текстовая аналитика — это процесс анализа неструктурированного текста, извлечения из него важной информации и ее преобразования в структурированный вид, в котором ее можно обрабатывать различными способами. Процессы анализа и извлечения данных, которые используются в текстовой аналитике, задействуют методы компьютерной лингвистики, статистики и информатики.

Рассмотрим пример. Оператор сотовой связи вывел на рынок два тарифных плана: «А» и «Б». Отдел маркетинга не может понять, почему «А» не пользуется ожидаемым спросом. Прояснить ситуацию может неструктурированный текст из заметок центра обработки вызовов (рис. 13.1).

Клиент XYZ позвонил по поводу «А», расспрашивал про условия тарифов. Клиент считает, что нужен перенос неиспользованных минут на следующий месяц.

Клиент ABC позвонил по поводу «А». Сказал, что невозможность переноса неиспользованных минут на следующий месяц — это идиотизм.

Потенциальный клиент позвонил по поводу «А». Сказал, что план очень дорогой.

Потенциальный клиент позвонил по поводу «А». Сказал, что 4 ГБ данных недостаточно.

Клиент XYT позвонил по поводу «А». Ругался, что в тариф включен слишком маленький объем данных.

Рис. 13.1. Пример записей центра обработки вызовов

Подчеркнутые слова отражают информацию, которая может быть полезна для рассмотрения ситуации. Видно, что тарифный план «А» вызывает много вопросов и у пользователей, и у потенциальных клиентов. Слова «перенос неиспользованных минут», «4 ГБ данных», «объем данных» и «дорогой» указывают на проблемы в тарифном плане. Слова «ругался» и «идиотизм» характеризуют эмоциональный отклик абонентов, в данном случае негативный.

Процесс текстового анализа задействует для обработки неструктурированного текста различные алгоритмы, такие как понимание структуры предложения, которые позволяют извлечь из текста информацию и преобразовать ее в структурированные данные. Структурированные данные, извлеченные из неструктурированного текста, приведены в табл. 13.1.

Идентификатор	Продукт	Проблема	Эмоции
Клиент XYZ	A	Перенос неиспользованных минут	Нейтрально
Клиент ABC	A	Перенос неиспользованных минут	Отрицательно
XXXX	A	Высокая цена	Нейтрально

Идентификатор	Продукт	Проблема	Эмоции
XXXX	A	Объем данных	Нейтрально
Клиент XYT	A	Объем данных	Отрицательно

Таблица 13.1. Формирование структурированных данных на основе неструктурированного текста

Казалось бы, ничего нового — ту же самую информацию мы получили, просто прочитав записи центра обработки вызовов. Однако в данном примере приведен лишь крошечный элемент огромного набора данных, который составляют записи тысяч операторов центра обработки вызовов. Каждый оператор сам по себе не может отследить масштабные тенденции в развитии каждого тарифного плана — он не обязан, да и физически не может делиться этой информацией со всеми остальными операторами. Однако после агрегации такой информации и ее обработки с помощью алгоритмов текстовой аналитики в этих неструктурированных данных четко прослеживается тенденция. Именно в этом преимущество текстовой аналитики.

Отличие текстовой аналитики от поиска

Обратите внимание, что в рассмотренном примере речь шла об извлечении текста, а не о поиске по ключевым словам. Поиск означает получение документа, удовлетворяющего запросу, — в этом случае конечный пользователь уже знает, что именно он пытается найти. Текстовая аналитика означает обнаружение информации. Вместе с тем текстовая аналитика может дополнять методы поиска: например, для улучшения категоризации или классификации документов, а также для получения краткого обзора или сводки по документам.

В табл. 13.2 показаны четыре метода работы с текстовыми данными: запрос, интеллектуальный анализ данных, поиск и текстовая аналитика. В левой колонке — методы, создающие выборку данных. Результатом их применения становится некоторый объем данных, соответствующих заданным критериям. Например, пользователь выполняет запрос к базе данных, чтобы узнать, сколько клиентов прекратило пользоваться услугами компании за прошедший месяц. Запрос возвращает одно число. Чтобы получить достаточно сведений, кото-

рые позволят понять, почему компания теряет клиентов, пользователю потребуется выполнить еще ряд запросов. Или, к примеру, поиск по ключевым словам позволит найти документы, где упоминаются конкуренты. Но для того, чтобы получить ответ на конкретный вопрос, пользователю придется просмотреть все эти документы.

	Выборка	**Анализ**
Структурированные данные	Запрос: возвращает данные	Интеллектуальный анализ данных: аналитика на основе структурированных данных
Неструктурированные данные	Поиск: выдает документы	Текстовая аналитика: анализ на основе текста

Таблица 13.2. Запрос, интеллектуальный анализ данных, поиск и текстовая аналитика

Технологии в левой колонке возвращают элементы информации и требуют участия пользователя для ее синтеза и анализа. Технологии в правой колонке позволяют получить нужный результат гораздо быстрее. Таким образом, ценность текстовой аналитики для бизнеса становится все более очевидной.

Методы анализа и извлечения информации

Перейдем к техническим аспектам. Как правило, решения текстовой аналитики используют сочетание статистических методов и методов обработки естественного языка (NLP) для извлечения информации из неструктурированных данных. Обработка естественного языка — это комплексная дисциплина, которая развивалась на протяжении последних двадцати лет. Ее цель — определение значения текста. Обработка естественного языка опирается на лингвистические концепции, такие как грамматические структуры и части речи. Обычно задача такого анализа данных сводится к тому, чтобы подробно описать ситуацию: кто что сделал, каким образом, когда, где и почему.

Обработка естественного языка позволяет анализировать текст на различных уровнях.

- Лексико-морфологический анализ исследует характеристики отдельных слов на уровне частей слова и на уровне частей речи. Это

позволяет понять значение конкретного слова в рассматриваемом контексте. Для лексического анализа применяются различные языковые словари, а также отраслевые и корпоративные глоссарии. В примере с тарифным планом словарь сообщит, что «дорогой» — это прилагательное в мужском роде, имеющее значения «стоящий больших денег», «добытый ценой больших усилий» или «любимый». Лексико-морфологический анализ покажет, что в данном контексте следует использовать первое значение (а заодно определит, что это слово не является формой существительного «дорога» или наречия «дорого»).

- Синтаксический анализ членит предложения на грамматически допустимые словосочетания, позволяя уточнить контекст отдельных слов. На этом этапе можно выстроить схемы возможных сочетаний слов, выявить часто встречающиеся грамматические структуры, последовательности слов или чисел (например, даты или денежные суммы).
- Семантический анализ определяет возможные значения предложения. Он включает изучение порядка слов и структуры предложения, а также сопоставление синтаксиса словосочетаний, предложений и абзацев для снятия неоднозначности.
- Дискурсивный анализ работает уже не с отдельными предложениями, а на уровне абзацев и всего текста целиком.

На практике извлечение информации из различных источников иногда требует выработки определенных правил. Эти правила могут быть очень простыми.

Имя начинается с заглавной буквы.

На веб-сайте университета перед названием каждого курса приводится трехзначный индекс, а после названия ставится точка с запятой.

На каждой странице в определенном месте должен стоять логотип компании.

Конечно, правила могут быть значительно более сложными. Они составляются вручную, автоматически или с применением обоих подходов.

- При составлении правил вручную последовательность правил извлечения информации может быть задана с использованием ведомственного языка. Ручное составление правил вкупе с обработкой словарей и созданием глоссариев требует больших затрат времени, но обеспечивает очень точные результаты.

- При автоматическом создании правил применяется машинное обучение и другие статистические методы. Программное обеспечение создает правила на основе набора данных для обучения и тестовых данных; сначала система обрабатывает набор похожих документов (например, газетных статей), чтобы выработать (выучить) правила, а затем пользователь обрабатывает тестовые данные, чтобы проверить точность правил.

Извлеченная информация

Описанные выше методы часто комбинируются с другими статистическими или лингвистическими методами для автоматизации сортировки текстовых документов и получения разных видов информации, таких как:

- термины, или ключевые слова;
- объекты (именованные объекты) — примеры прямых или косвенных абстракций. Объектами могут быть имена, названия, контактные данные, даты, время, валюты, должности и т.д. Например, программное обеспечение текстовой аналитики может извлечь объект «Яна Дулина» и идентифицировать его как имя человека, объект «3 марта 2007» — как дату, и т.д. Многие поставщики предоставляют готовые возможности извлечения объектов;
- факты (отношения). Указывают на простые отношения между двумя объектами: «Иван Кузнецов — исполнительный директор», «аспирин снижает температуру»;
- события. Иногда используется как синоним терминов «факт» и «отношение». Разграничение терминов «факт» и «событие» основано на том, что события имеют временной компонент и могут изменять факты: например, событие «Увольнение Ивана Кузнецова» повлияет на факт «Иван Кузнецов — исполнительный директор»;
- концепции. Это наборы слов и фраз, которые отражают определенную идею или тему, важную для пользователя. Категоризация может быть выполнена вручную или с помощью статистических методов на основе правил или гибридных методов. Например, концепция «недовольный клиент» может включать слова «сердится», «разочарован», «запутался», а также фразы «отключенная услуга», «не перезвонили», «потеря денег» и т.д. Это позволяет идентифицировать в тексте «недовольного клиента», даже если слова «недовольный» и «клиент» там не встречаются. Концепции

могут задаваться пользователями в соответствии с решаемыми задачами;

- эмоции. Анализ эмоций выявляет в тексте точки зрения и эмоционально окрашенные высказывания. Некоторые методы достигают этого путем классификации текста как субъективного (мнение) или объективного (факт) с использованием инструментов машинного обучения или обработки естественного языка. Анализ эмоций находит широкое применение в приложениях для изучения мнения клиентов.

Классификации

Классификации часто являются критически важными для текстовой аналитики. Классификация — это метод упорядочивания информации в соответствии с иерархическими отношениями (по категориям). Так как классификация определяет отношения между терминами, используемыми компанией, она упрощает поиск и анализ текста.

Например, поставщик телекоммуникационных услуг предоставляет услуги проводной и беспроводной связи. Услуги беспроводной связи, предоставляемые компанией, могут включать сотовую связь и доступ в Интернет. Услуги сотовой связи могут быть разделены на категории несколькими путями, например, по тарифным планам, по типам телефонов, по срокам действия и т.д. Любую из этих категорий можно дробить и дальше.

В классификациях также могут использоваться синонимы и синонимические выражения — например, чтобы распознавать «сотовый» и «мобильный» как одно понятие. Такие классификации могут быть достаточно сложными и требовать длительного времени на разработку.



Некоторые поставщики решений заявляют, что использование их продуктов не требует предварительного определения классификаторов, — конечные пользователи могут упорядочивать уже извлеченную информацию. На деле необходимость использования классификации зависит от решаемых задач; если тематика источников информации очень сложная или узкоспециализированная, без классификации не обойтись.

Совмещение результатов текстового анализа со структурированными данными

Структурированную информацию, извлеченную из неструктурированных данных, можно объединить с другой структурированной информацией, например, содержащейся в вашем хранилище данных, и затем использовать их совместно для дальнейшей бизнес-аналитики или интеллектуального анализа данных.

В таблице 13.3 приведен пример объединения результатов текстовой аналитики со структурированными данными из биллинговой системы. Отличие от таблицы 13.1 только одно — появился столбец «Сегмент», где условно отражен уровень дохода. Таким образом, сведения, хранящиеся в биллинговой системе, можно сопоставить со сведениями из заметок операторов. Конечно, о потенциальных клиентах, звонивших в центр обработки вызовов, информации в системе нет, поэтому в столбце «Сегмент» отображается значение «XXXX».

Идентификатор	Продукт	Проблема	Эмоции	Сегмент
Клиент XYZ	A	Перенос неиспользованных минут	Нейтрально	Золотой
Клиент ABC	A	Перенос неиспользованных минут	Отрицательно	Серебряный
XXXX	A	Высокая цена	Нейтрально	XXX
XXXX	A	Объем данных	Нейтрально	XXX
Клиент XYT	A	Объем данных	Отрицательно	Бронзовый

Таблица 13.3. Объединение структурированных и неструктурированных данных

Результат объединения данных показал, что как минимум один из клиентов в списке имеет «золотой» статус, так что стоит приложить дополнительные усилия, чтобы сохранить этого абонента. Конечно, на практике объемы обрабатываемых данных будут гораздо больше, чем в данном примере.

Применение больших данных на практике

Пример с сотовым оператором и неудачным тарифным планом показывает, как текстовая аналитика помогает осмыслить имеющиеся данные. А если необходимо работать с большими данными? Тогда анализируемые неструктурированные данные будут характеризоваться очень большим объемом, очень большой скоростью или и тем, и другим. Рассмотрим несколько примеров.

Мнение клиентов

Оптимизация взаимодействия и повышение лояльности клиента являются ключевыми задачами для многих организаций в секторе услуг. Чтобы понять, как обеспечить требуемую эффективность в этой сфере, нужно ответить на ряд вопросов.

- В каких направлениях возникает больше всего жалоб? Остаются ли эти направления проблемными на протяжении длительного времени?
- Насколько клиенты удовлетворены теми или иными услугами?
- Каковы преобладающие причины ухода клиентов к конкурентам?
- Какие ключевые сегменты клиентов потенциально готовы к приобретению сопутствующих товаров?

Сообщения электронной почты, поступившие в компанию, опросы об уровне удовлетворенности клиентов, заметки центра обработки вызовов и другие внутренние документы содержат много сведений о мнениях и эмоциях клиентов. Текстовый анализ позволяет определить причины неудовлетворенности клиентов и своевременно решить проблемы, пока они не распространились.

Является ли это примером больших данных? Вполне, если речь идет о большом объеме информации, например, поступающей в пакетном режиме. Эту информацию можно объединить со структурированными данными, как отмечалось выше.

Аналитика социальных медиа

Еще одна форма изучения потребительского поведения и управления взаимодействием с клиентами — аналитика социальных средств коммуникации. Это сравнительно молодой тренд, но он дал существен-

ный стимул развитию рынка текстовой аналитики. При использовании аналитики социальных медиа осуществляется сбор и совместная обработка разнообразных данных из интернет-источников, таких как блоги, микроблоги, новостные порталы, форумы и др. Затем этот огромный поток данных подвергается анализу, часто с использованием текстовой аналитики, чтобы получить ответы на ряд вопросов.

- Что говорят люди о бренде?
- Что им нравится в бренде?
- Что им не нравится в бренде?
- Каковы позиции бренда по сравнению с конкурентами?
- Насколько лояльны клиенты бренда?

Социальные медиа важны не только для маркетологов. К примеру, с их помощью правоохранительные органы выслеживают террористов, а медицинские организации прогнозируют эпидемии.

Аналитика социальных медиа — пример использования больших данных (особенно если провайдер информационных услуг располагает достаточными мощностями, чтобы обрабатывать потоки сообщений в Twitter наряду с другими данными).

IBM Watson

Возможно, вы слышали о том, как в 2011 году компьютер участвовал в телевикторине Jeopardy! (российский аналог — «Своя игра») и одержал победу. Это был суперкомпьютер IBM Watson. Watson — это набор технологий, которые обрабатывают и анализируют огромные объемы структурированных и неструктурированных данных. Watson может обработать и проанализировать информацию, содержащуюся в 200 миллионах книг, за три секунды. Однако в основе этих мощных вычислительных способностей доступные коммерческие технологии, которые были дополнены «секретным ингредиентом» — специальными разработками IBM Research. Это сочетание технологий программного обеспечения для работы с большими данными и контентом, инструменты прогнозной аналитики и специальное программное обеспечение. IBM совместно работает с медицинскими организациями над применением возможностей Watson для нужд здравоохранения. Этот суперкомпьютер должен положить начало целой линейке мощных программно-аппаратных комплексов.

Итак, что собой представляет «секретный ингредиент»? Watson понимает естественный язык, формирует и оценивает предположения, может адаптироваться и обучаться.

Во-первых, Watson использует обработку естественного языка. С помощью ряда аннотаторов IBM извлекает информацию о симптомах, возрасте пациентов, их местонахождении и т.д. Watson быстро обрабатывает огромные объемы таких неструктурированных данных, используя специально созданную для этого архитектуру.

Во-вторых, Watson формирует предположения, которые являются возможными ответами на вопросы. Компьютер обучается путем обращения к системе данных в виде вопросов и ответов: другими словами, ему показывают репрезентативные вопросы и правильные ответы. Такой прием называется эмпирическим обучением. Он ставит своей целью формирование модели, которая может давать ответы с высокой степенью достоверности (логистическая регрессия с большим набором атрибутов). Watson начинает с использования общей статистической модели, затем оценивает первый вопрос и ответ и с его помощью корректирует коэффициенты. По мере накопления фактов он продолжает корректировать коэффициенты, пока не «скажет», что достоверность велика. Обучение Watson имеет ключевое значение — по сути, его «учителя» строят статистические модели с системой оценки. В результате обучения Watson получает систему, характеризующуюся определенными векторами и моделями, что в конечном счете позволяет использовать вероятностные модели для оценки ответов. В работе Watson нет детерминизма (то есть правила не используются) — она является вероятностной, и именно это делает суперкомпьютер очень динамичным.

Когда Watson формирует предположение, он оценивает его на основе фактов. Его целью является получение правильного ответа на правильных основаниях. (Теоретически, например, если диагноз характеризуется пятью положительными симптомами и четырьмя отрицательными, а Watson располагает сведениями только о четырех из девяти симптомов, он запросит дополнительную информацию.) Выдается предположение с наивысшей оценкой. В конце анализа Watson «уверен», знает ли он ответ или не знает.

Приведем пример. Пациент подозревает у себя проблемы с щитовидной железой и обращается к врачу с жалобами на сильное сердцебиение, усталость, выпадение волос и мышечную слабость. Доступ в систему Watson позволит врачу ускорить постановку диагноза. Watson уже загрузил и обработал всю информацию из книг и журналов относительно заболеваний щитовидной железы, а также располагает хранящимися в электронных медицинских картах данными о диагнозах других пациентов, наблюдающихся в этой же клинике, и о случаях из практики других врачей.

На основе набора симптомов, указанных пациентом, компьютер формирует предположения вместе с соответствующими вероятностями (например, 60% гипертиреоз, 40% нервное перенапряжение и т.д.). Система может запросить дополнительную информацию, например архивные медицинские карты пациента из других клиник; по мере обработки новых данных Watson будет уточнять предположение и оценивать вероятность его точности. Когда вся доступная информация будет обработана, компьютер «выберет» диагноз с самым высоким

уровнем достоверности и представит его врачу. Если же Watson «не знает» ответа, он сообщит о том, что не удалось получить ответ с достаточно высоким уровнем достоверности или что у него недостаточно информации.

Специалисты IBM утверждают, что процесс обучения Watson воплощает те же подходы, что применяются в педагогике, когда ребенка учат учиться. Чтобы получить знания, ребенок может прочитать книгу, а может усвоить тот же материал, слушая учителя, который ставит вопросы и разъясняет ответы.

Инструменты текстовой аналитики больших данных

Расскажем о некоторых игроках этого рынка. Среди них и небольшие компании, и известные крупные корпорации. Не все поставщики позиционируют свои продукты как решения для текстовой аналитики больших данных — часть из них предлагает инструменты для обычной текстовой аналитики, которые, однако, могут применяться для работы с большими данными.

Attensity

Attensity (www.attensity.com) — компания, специализирующаяся исключительно на текстовой аналитике. Она начала разрабатывать и продавать аналитические продукты более десяти лет назад; сейчас у нее более 150 корпоративных клиентов и одна из крупнейших в мире групп разработки решений по обработке естественного языка. Attensity предоставляет несколько модулей текстовой аналитики, в число которых входят Auto-Classification, Entity Extraction и Exhaustive Extraction. Exhaustive Extraction — это флагманская технология Attensity, которая автоматически извлекает факты из синтаксически проанализированного текста и упорядочивает эту информацию.

Компания сосредоточилась на аналитике социальных медиа и многоканальной текстовой аналитике. Возможен анализ текстов как из внутренних, так и из внешних источников с последующим предоставлением результатов анализа бизнес-пользователям. Недавно Attensity приобрела компанию Biz360, которая занимается социальными медиа и агрегирует из них большие потоки данных. Attensity разработана система сетевых распределенных вычислений, которая обеспечивает высокопроизводительную обработку огромных

объемов текста в режиме реального времени. Для хранения данных Attensity использует платформу Hadoop (MapReduce, HDFS и HBase). Кроме того, компания имеет систему опроса данных, которая распознает пиковые нагрузки во входящих данных и распределяет обработку по большему или меньшему количеству серверов, в зависимости от потребностей.

Clarabridge

Еще один специализированный поставщик средств текстовой аналитики — Clarabridge (www.clarabridge. com), бывший филиал компании Claraview, которая оказывала консультационные услуги по вопросам бизнес-аналитики. Clarabridge выделилась в отдельную компанию, специализирующуюся на работе с неструктурированными данными. По заказу ряда организаций Clarabridge проводит комплексное изучение клиентов, определяет ключевые возможности взаимодействия и проблемы и помогает организовать эффективную работу в компании в режиме реального времени. Сюда входит определение эмоциональной окраски и классификация данных клиентских отзывов в режиме реального времени, а также формирование основы для будущей обработки информации в системе Clarabridge.

В настоящее время Clarabridge предлагает ряд интересных специализированных инструментов. Они включают поиск и анализ основных причин изменения объема текстовых потоков, эмоциональной окраски сообщений или степени удовлетворенности клиентов в контексте возникающих проблем. Clarabridge также предлагает свое решение в сфере программного обеспечения как услуги (SaaS).

IBM

Титан программного обеспечения IBM (www.ibm.com) предоставляет несколько решений в сфере текстовой аналитики в рамках своего стратегического предложения Smarter Planet. Помимо Watson и IBM SPSS (о SPSS подробнее рассказано в главе 12), IBM также предлагает разработанную на основе исследований подразделения IBM Research систему IBM Content Analytics с Enterprise Search (ICAES).

IBM Content Analytics используется для преобразования контента в проанализированную информацию, которая доступна для детального анализа, аналогичного инструментам анализа структуриро-

ванных данных в пакетах для бизнес-аналитики. Ранее IBM Content Analytics и Enterprise Search были двумя разными продуктами; конвергентное решение ориентировано на обеспечение расширенного поиска по корпоративным ресурсам, задействующего текстовую аналитику, а также обеспечивающего отдельные потребности в анализе контента. Система ICAES хорошо интегрирована с платформой IBM InfoSphere BigInsights, что позволяет осуществлять крупномасштабные поисковые запросы и поддерживать работу набора средств для анализа контента.

OpenText

OpenText (www.opentext.com) — канадская компания, которая больше всего известна благодаря своей ведущей роли в разработке решений по управлению корпоративной информацией (EIM). Ее бизнес-концепция заключается в управлении, обеспечении безопасности неструктурированных данных корпораций и извлечении из них полезных сведений. Компания предлагает на рынке решение, которое она называет «семантическим промежуточным ПО». В перспективе, по мнению специалистов OpenText, эта семантическая технология «обеспечит аналитику в режиме реального времени с высокой степенью точности при использовании больших наборов данных (то есть контента). Анализ можно проводить для различных языков и в разных форматах представления данных из разных отраслей». Основой семантического промежуточного ПО является семантика, которая может использоваться на различных уровнях вместе с разными технологиями (например, управлением документооборотом, прогнозной аналитикой и т.д.) для решения различных бизнес-задач. Другими словами, текстовая аналитика может быть задействована при необходимости где угодно. OpenText предоставляет это промежуточное ПО как самостоятельный продукт, который может использоваться в различных решениях, а также встраивает его в собственные продукты.



Идея подключаемых семантических модулей имеет неплохие перспективы. Небольшие игроки рынка также обращаются к этим модулям, изучая возможности их использования в работе с большими данными.

SAS

Компания SAS (www.sas.com) уже длительное время занимается комплексными проблемами, связанными с большими данными. Несколько лет назад компания приобрела поставщика инструментов текстовой аналитики, компанию Teragram, что позволило расширить возможности продуктов анализа как структурированных, так и неструктурированных данных и обеспечить интеграцию возможностей работы с такими данными в решения компании для описательного и прогнозного моделирования. Теперь возможности текстовой аналитики компании являются частью общей аналитической платформы, а текстовые данные рассматриваются просто как еще один источник данных.

SAS продолжает инновационные разработки в области высокопроизводительной аналитики, добиваясь сокращения временных затрат на выполнение аналитических операций. Например, SAS High Performance Analytics Server — это решение, размещаемое в памяти, которое позволяет разрабатывать аналитические модели с использованием всего объема данных, не прибегая к созданию выборки агрегированных данных. SAS заявляет, что может обеспечить анализ тысяч переменных и миллионов документов. Решение работает на программно-аппаратных комплексах EMC Greenplum или Teradata, а также на стандартном оборудовании с использованием распределенной файловой системы Hadoop (HDFS).

Глава 14

Специальные подходы к анализу больших данных

В ЭТОЙ ГЛАВЕ:

Новые модели и подходы, способствующие развитию анализа больших данных
Полностью специализированный и частично специализированный анализ
Оптимальная среда для анализа больших данных
Преобразование больших данных в малые

Замечательная особенность больших данных заключается в том, что теоретически весь массив данных, как внутрикорпоративных, так и внешних, может использоваться для выполнения необходимого анализа. В идеале это означает, что любое увеличение объема или разнообразия анализируемых данных позволит получить новые, более содержательные результаты анализа.

Как уже отмечалось в главах 12 и 13, многие инструменты, используемые для анализа больших данных, являются лишь развитием уже доступных на рынке решений для бизнес-аналитики и углубленной аналитики. Сюда входят программное обеспечение для интеллектуального анализа данных, прогнозное моделирование, углубленная статистика и текстовая аналитика. Кроме того, зачастую поставщики приспосабливают уже отработанные алгоритмы выполнения программного обеспечения для новой инфраструктуры больших данных.

Данная глава посвящена верхним уровням инфраструктуры (рис. 14.1).

Как отмечают некоторые эксперты, анализ больших данных предполагает иной тип мышления по сравнению с традиционным анализом — готовность экспериментировать и приспосабливать свои



Рис. 14.1. Слои эталонной архитектуры больших данных, необходимые для клиентского анализа

стратегии к данным, а не наоборот. С этим мнением можно не соглашаться, но очевидно, что экосистема анализа больших данных нуждается в новых технологичных платформах, алгоритмах и навыках осуществления такого вида анализа. Это особенно актуально, когда речь заходит о расширении горизонта возможностей. Поскольку применение больших данных пока что находится на ранней стадии, большая часть аналитических инструментов разрабатывается или настраивается «под заказ». В этой главе рассматриваются некоторые инновации в этой сфере и специфика работы с ними.

Выработка новых моделей и подходов для поддержки больших данных

Анализ больших данных становится очень популярной темой, и это не случайно. Многие компании приветствуют новую возможность использовать и анализировать собираемые данные, которыми прежде не могли эффективно управлять. Теперь ценная информация, которой они обладают, благодаря средствам интеллектуального анализа дан-

ных может применяться для практических решений. Это замечательный аспект в развитии анализа больших данных. В числе новых возможностей — визуализация огромных объемов разобщенных данных, графическое представление потока обработанных после углубленного анализа данных в режиме реального времени и многое другое. Такое развитие, с одной стороны, является эволюционным, а с другой — революционным.

Характеристики анализа больших данных

Итак, что меняется, когда компания решает расширить горизонты своих возможностей с помощью анализа больших данных? В главе 12 уже упоминалось, что для поддержки анализа больших данных требуется другая инфраструктура, а алгоритмы должны быть изменены таким образом, чтобы учитывать используемую инфраструктуру.

Анализ больших данных можно рассматривать с двух ракурсов:

- ориентация на решения;
- ориентация на действия.

Анализ, ориентированный на решения, близок традиционной бизнес-аналитике: рассматриваются репрезентативные выборки данных из крупных источников информации, затем результаты анализа применяются в процессе принятия бизнес-решений. Конечно, эти решения могут повлечь за собой значительные изменения; однако анализ играет в них вспомогательную роль, дополняя процесс принятия решений.

Анализ, ориентированный на действия, обеспечивает быстрое реагирование на обнаружение определенной тенденции или выявление специфических типов данных. Некоторые применения анализа больших данных уже отмечались выше, теперь же рассмотрим их подробнее. Использование преимуществ анализа больших данных предоставляет значительные возможности для всех, кто решил задействовать эти технологии уже на ранних этапах.

Специально созданные приложения для поиска, обработки и анализа больших данных могут ускорить получение от них выгоды. Наиболее эффективно либо создавать подобные специализированные приложения «с нуля», либо адаптировать под новые нужды уже существующие платформы и компоненты. Этот аспект будет рассмотрен ниже.

Сначала отметим некоторые характеристики анализа больших данных, которые отличают его от традиционных видов анализа (помимо ранее упомянутых скорости, объема и разнообразия обрабатываемых данных).

- Анализ может быть программируемым. Одно из важнейших изменений состоит в том, что раньше работа велась с наборами данных, которые можно было вручную загрузить в приложение, визуализировать и изучить. При анализе больших данных ввиду их огромного объема исходные данные предпочтительно обрабатывать программно (используя код) даже для изучения на начальном этапе.
- Анализ может быть ориентированным на данные. Многие специалисты по работе с данными используют подход к анализу данных на основе предположений (выдвижение гипотезы и сбор данных, чтобы проверить ее правильность). Однако данные можно использовать и как основополагающий фактор анализа, особенно если собирается их огромный объем. Например, можно использовать алгоритм машинного обучения (см. главу 12) для выполнения подобного анализа без формирования предположений.
- Анализ может использовать множество атрибутов. Если в прошлом приходилось работать с сотнями атрибутов или характеристик источника данных, то сегодня счет атрибутов пошел на тысячи, а сами данные могут занимать сотни гигабайт и представлять собой результаты миллионов наблюдений.
- Анализ может быть итеративным. Увеличение вычислительных мощностей означает, что модель может циклически выполняться, пока не будет найдено оптимальное решение. Предположим, вы создаете модель, которая ищет прогностические параметры определенного поведения клиентов в отношении определенных продуктов. Можно начать с извлечения подходящего образца данных или подключения к источнику данных. Можно построить модель для проверки предположения.

Для эффективной работы этой модели (перебора всех итераций, необходимых для обучения алгоритма) потребуется значительный объем памяти. Возможно, также понадобится использовать более совершенные методы компьютерных вычислений, такие как обработка естественного языка или нейронные сети, которые автоматически развивают модель на основе обучения по мере добавления данных.

- Анализ может быть быстрым, если прибегнуть к использованию облачной инфраструктуры как услуги. Благодаря платформам инфраструктуры как услуги (IaaS), таким как Amazon Cloud Services (ACS), можно быстро подготовить кластер компьютеров для быстрой обработки и анализа больших наборов данных.

Теперь рассмотрим некоторые средства анализа больших данных.

Подходы к анализу больших данных

Во многих случаях результаты анализа больших данных предоставляются пользователю в виде отчетов и визуализаций. Так как разнообразие необработанных данных может быть колоссальным, необходимо полагаться на инструменты и методы анализа, которые позволяют представить данные в доступной для понимания форме. Традиционные отчеты недостаточно ясно представляют результаты анализа и зачастую не содержат информации, необходимой руководителям, ответственным за принятие решений. Весьма эффективны в таких случаях методы визуализации данных, но и они нуждаются в улучшении — для работы с большими данными требуются сложные инструменты с широким функционалом.

Привычных отчетов и визуализации недостаточно — анализ больших данных требует создания новых подходов и приложений. Можно, конечно, дождаться, пока подходящие решения не выйдут на рынок, но это чревато потерей времени и преимущества, к тому же такие решения могут не обеспечить конкретных потребностей вашей организации. Итак, для использования больших данных на ранних этапах необходимо создавать новые приложения, соответствующие не только требованиям, предъявляемым анализом, но и требованиям ко времени, затрачиваемому на выполнение этих задач.

Эти новые приложения можно разделить на две категории: специализированные (код написан «с нуля») и частично специализированные (созданные на основе платформ или компонентов). Рассмотрим некоторые примеры, чтобы понять, как и зачем эти подходы в перспективе могут применяться для эффективного использования больших данных в повседневной работе.

Специализированные приложения для анализа больших данных

Специализированное приложение создается для определенной цели или ряда связанных задач. Определенные области бизнеса или организации всегда нуждаются в специальном наборе технологий, соответствующих специфике деятельности или обеспечивающих требуемые конкурентные преимущества. Например, если вы оказываете финансовые услуги, то, естественно, заинтересованы в том, чтобы ваши биржевые инструменты работали быстрее и эффективнее, чем приложения ваших конкурентов. А биллинговые приложения при этом для вас не являются приоритетными, поэтому для них можно использовать типовой коммерческий продукт.

В рамках анализа больших данных целью разработки специализированного приложения является ускорение времени принятия решения или времени реагирования. По мере развития больших данных и как научной дисциплины, и как рынка поставщики традиционных программных решений будут все медленнее выводить новые технологии на рынок. Инфраструктура больших данных сама по себе не имеет особой ценности, если из-за недостаточных возможностей анализа, соответствующего требованиям конкретной сферы бизнеса, преимущества этой инфраструктуры не находят применения для принятия решений или быстрого реагирования на ситуацию. Как уже отмечалось в главах 12 и 13, некоторые пакеты поддерживают различные методы анализа больших данных. Поставщики, о которых мы говорили в этих главах, могут использовать собственные технологические компоненты, чтобы создавать решения для своих клиентов. Однако дело в том, что даже самое совершенное приложение не превратится сразу же в комплексное средство обработки больших данных. Рассмотрим некоторые дополнительные возможности из области специализированных (индивидуализированных) приложений для анализа больших данных.

Среда R

Среда R основана на языке статистики и анализа S, который был разработан в 1990-х годах компанией Bell Laboratories. Она поддерживается в рамках проекта GNU и доступна в качестве соответствующей лицензии GNU. За многие годы пользователи S и R внесли большой вклад в базовую систему, расширяя и развивая ее возможности. Несмотря на сложность освоения системы, ее глубина и гибкость обеспе-

чили ее популярность как среди разработчиков приложений аналитики, так и среди опытных пользователей. Кроме того, в рамках проекта CRAN R (полноценная архивная сеть R) поддерживается глобальный набор FTP- и веб-серверов, содержащих самые актуальные версии среды R. Компания Revolution Analytics из Пало-Альто (Калифорния) (www.revolution-computing.com) также предоставляет корпоративную версию R с коммерческой поддержкой.

R является интегрированным пакетом программных инструментов технологии создания специализированных приложений, упрощающих обработку данных, расчеты, анализ и визуальное отображение. Ниже перечислены некоторые возможности пакета:

- эффективные компоненты для обработки данных;
- операторы для выполнения расчетов с использованием массивов и других типов упорядоченных данных;
- инструменты, подходящие для различных видов анализа данных;
- расширенные возможности визуализации;
- язык программирования S, разработанный программистами для программистов, имеет много знакомых конструктивных элементов, включая условия, циклы, пользовательские рекурсивные функции и целый ряд средств ввода и вывода. Большинство функций, предоставляемых в самой системе, написано на языке S.

R — это механизм, предназначенный для разработки новых методов интерактивного анализа больших данных. Он был быстро разработан и расширен благодаря большой коллекции дополнительных пакетов. Он оптимизирован для создания и обеспечения работы приложений, ориентированных на анализ источников больших данных.

Интерфейс Google Prediction API

Интерфейс Google Prediction API — пример развивающегося класса инструментов для анализа больших данных. Он доступен на вебсайте для разработчиков Google и хорошо задокументирован. Google Prediction API обеспечивает несколько механизмов доступа с использованием различных языков программирования. Для ознакомления с возможностями этого интерфейса предлагается возможность бесплатно пользоваться им в течение шести месяцев (с некоторыми функциональными ограничениями), прежде чем приобретать полную лицензию, которая ориентирована на проектную работу и отличается умеренной ценой.

Интерфейс Prediction API довольно прост: он ищет закономерности и сопоставляет их с проскрипционными, прескрипционными и другими существующими закономерностями. При этом интерфейс «обучается» — чем больше сопоставлений он проводит, тем «умнее» становится. Каковы возможности применения интерфейса Prediction API? Например, для изучения потребительского поведения можно исследовать закономерности на основе записей в Facebook, Twitter, Amazon и Foursquare. Производителю потребительских товаров изучение информации в социальных сетях подскажет, когда и как предложить новые или уже имеющиеся продукты. Кинокомпания с помощью этих же средств сможет уведомлять пользователей социальных сетей о выходе нового фильма с их любимым актером. Интерфейс Prediction API дает возможность прогнозировать (или даже формировать) поведение в будущем путем анализа привычек и предыдущих действий.

Интерфейс Prediction является реализацией интерфейса RESTful API с поддержкой языков NET, Java, PHP, JavaScript, Python, Ruby и многих других. Google также предоставляет сценарии для получения доступа к интерфейсу API наряду с клиентской библиотекой R.

Прогнозный анализ — одна из самых мощных потенциальных возможностей больших данных, а интерфейс Google Prediction API становится очень полезным инструментом для создания специализированных приложений.

По мере развития больших данных на рынке будут появляться все новые средства создания специализированных программных приложений, и ориентированные на подготовленного пользователя, как среда R, и представленные в виде интерфейсов API или библиотек, которые программисты смогут использовать для создания новых способов обработки и анализа больших данных. На практике далеко не в каждой компании есть квалифицированные разработчики программного обеспечения, способные создавать специализированные приложения. К счастью, уже имеются и продолжают появляются новые средства, позволяющие удовлетворить специфические потребности пользователей в анализе.

Частично специализированные приложения для анализа больших данных

Зачастую приложения, которые представляются специализированными, на самом деле созданы с использованием готовых пакетов или сторонних компонентов, таких как библиотеки. Далеко не всегда не-

обходимо создавать новое приложение «с нуля» (если только речь не идет о продукте, не имеющем даже приблизительных аналогов). Использование пакетных приложений и компонентов позволяет разработчикам или аналитикам написать код, «сшивающий» эти компоненты в специализированное рабочее приложение. Вот преимущества такого подхода:

- скорость внедрения — так как не нужно создавать каждую часть приложения, время разработки значительно сокращается;
- стабильность — использование хорошо написанных, надежных сторонних компонентов позволяет делать специализированные приложения более устойчивыми к отказам;
- высокое качество — к пакетным компонентам часто применимы более высокие стандарты качества, так как предполагается, что они будут работать в самых различных средах и областях;
- повышенная гибкость — если появляется более совершенный компонент, он может быть задействован в приложении, что продлевает срок эксплуатации, адаптивность и удобство использования специализированного приложения.

Другой тип частично специализированного приложения — это приложение с открытым исходным кодом, который модифицируется под определенные задачи. Это эффективный подход, позволяющий встроить в частично специализированное приложение полезные компоненты, такие как:

- TA-Lib — библиотека технического анализа, широко используемая разработчиками программного обеспечения для создания инструментов технического анализа данных финансовых рынков. Доступна в виде открытого исходного кода в рамках лицензии BSD и может быть встроена в частично специализированные приложения;
- JUNG (платформа Java Universal Network Graph) — библиотека, предоставляющая общую платформу анализа и визуализации данных, которые могут быть представлены в виде графа или сети. Эффективно применяется для анализа социальных сетей, измерений важности (рейтинг страницы, количество посещений) и интеллектуального анализа данных. Доступна в виде открытого исходного кода в рамках лицензии BSD;
- GeoTools — набор геопространственных инструментов с открытым исходным кодом, предназначенный для управления различными формами ГИС-данных. Позволяет анализировать пространствен-

ные и непространственные атрибуты данных, создавать графы и сети данных. Предоставляется в рамках лицензии GPL2, что позволяет интегрировать его в частично специализированные приложения.

Поддержка мобильных устройств

Многие (если не все) мобильные приложения являются специализированными. Некоторые сторонние поставщики пакетов предоставляют возможности мобильного доступа, часто с помощью приложения для мобильных устройств, однако эти приложения мало применимы за пределами сферы интересов поставщика. В результате многие выходящие на рынок разработчики специализированных компонентов создают технологии, позволяющие быстро создать мобильные приложения для работы с большими данными.

Скорость больших данных и их разнообразие форсируют переход к наблюдениям в режиме реального времени. Это, в свою очередь, позволит принимать более обоснованные решения или быстрее реагировать на ситуацию. По мере развития рынка для решения задач, связанных с наблюдением, скорее всего, будут использоваться специализированные приложения. Они расширят возможности реагирования на происходящие перемены. Платформы и компоненты анализа позволяют быстро, легко и эффективно создавать, менять и обслуживать такие приложения.

Характеристики платформы анализа больших данных

Несмотря на разнообразие новых наборов инструментов, позволяющих более эффективно контролировать и анализировать большие данные, не всегда среди них окажется идеально подходящее решение. Кроме того, целый ряд технологий, о которых говорилось выше, поддерживает анализ больших данных и удовлетворяет требованиям к доступности, масштабируемости и производительности. В числе этих технологий программно-аппаратные комплексы больших данных, столбцовые базы данных, базы данных, размещенные в памяти, нереляционные базы данных и модули масштабной параллельной обработки. Более подробно они освещены в главах 1, 4 и 7.

Итак, чего же ждут бизнес-пользователи от анализа больших данных? Ответ на этот вопрос зависит от типа бизнес-задачи, которую необходи-

мо решить. Ранее в этой главе были упомянуты два основных типа решения бизнес-задач — ориентация на решения и ориентация на действия. У этих двух подходов есть ряд общих характеристик, что естественно, так как решения часто ведут к действиям. При выборе платформы анализа больших данных следует учитывать следующие факторы.

- Поддержка многочисленных типов данных. Многие организации осуществляют или планируют в рамках своих решений для обработки больших данных использование всех типов данных: структурированных, слабоструктурированных и неструктурированных.
- Управление пакетной обработкой и потоками данных в режиме реального времени. Ориентация на действия обеспечивается анализом потоков данных в режиме реального времени. Ориентация же на решения адекватно поддерживается пакетной обработкой. Некоторым пользователям необходимо и то, и другое по мере того, как они задействуют новые формы анализа.
- Использование имеющихся ресурсов среды. Чтобы лучше соответствовать специфике решаемых задач, может потребоваться использовать в платформе анализа больших данных уже существующие данные и алгоритмы.
- Поддержка NoSQL и других более современных форм доступа к данным. Запросы SQL продолжают широко применяться, но многие организации также ищут новые способы получения доступа к данным, чтобы сократить время реагирования и время принятия решений.
- Уменьшение времени задержки. При работе с данными, которые характеризуются высокими скоростями, необходима платформа, отвечающая требованиям к скорости и производительности.
- Дешевое хранение данных. Работа с большими данными означает потенциально большие объемы хранения — следовательно, затраты на управление хранилищем и само отказоустойчивое хранилище являются очень важными факторами.
- Интеграция с облачными технологиями. Облако может предоставлять по требованию хранилище и вычислительные мощности. Все больше и больше компаний используют облако как площадку для экспериментов в области больших данных и тестирования своих решений. Еще одно преимущество облака — возможность интеграции существующих систем с облачными сервисами (общедоступными или частными) в единую гибридную модель. Кроме того, появляются облачные сервисы для работы с большими данными (см. главу 11).

Помимо вышеуказанных характеристик необходимо прежде всего учитывать, что наиболее ощутимое преимущество создания приложений на основе платформы — это быстрота их внедрения. Рассмотрим пример перспективной платформы анализа больших данных.

Развертывание платформы, поддерживающей разработку и внедрение приложений для работы с большими данными, может быть выполнено в виде одного экземпляра, частного облака или общедоступного облака. Какой-либо адаптации к среде, где будет использоваться платформа, не требуется. Сама платформа представляет собой набор технологий, разработанный для решения любых задач работы с большими данными на низком уровне. В качестве конструктора приложений используется подключаемый модуль, который позволяет разработчику создавать, тестировать и выполнять отладку в знакомой среде.

На такой платформе могут быть реализованы следующие возможности:

- поддержка потокового анализа данных и реагирования в режиме реального времени;
- унифицированный интерфейс API, который устраняет необходимость создавать инфраструктуру приложений для работы с большими данными;
- простой интерфейс запросов и поддержка подключаемых обработчиков запросов позволяют быстро получать требуемые результаты;
- доступные для запросов через объединенный интерфейс API наборы данных и таблицы;
- считывание и запись данных независимо от входных и выходных форматов или особенностей используемых базовых компонентов (операции с данными на основе Hadoop);
- транзакционная обработка событий;
- многомодальное развертывание на одном узле или в облаке.

Такой подход обеспечит поддержку разработки приложений благодаря обилию инструментов и технологий, необходимых для создания среды работы с большими данными. Когда структура или уровень абстракции обеспечивают управление особенностями базовых компонентов, у разработчиков появляется возможность использовать высокоуровневые интерфейсы API, что позволяет создавать высококачественные, надежные приложения, которые без труда могут быть модифицированы и развернуты.



Не все компании могут позволить себе создавать средства для анализа больших данных «с нуля». Поэтому лучше всего рассматривать развертывание больших данных как экосистему, объединяющую людей, процессы и технологии. Анализ больших данных — это не изолированная система, он связан со многими другими средами и бизнес-процессами всего предприятия. Сейчас, на раннем этапе развития технологий больших данных, многие проекты в этой сфере представляют собой эксперименты, отделенные от общей среды вычислений. Однако обязательно нужно предусматривать возможность интеграции этих средств. Когда анализ больших данных станет широко распространен, он должен оставаться отделенным от остальной среды управления данными.

Разработчики программного обеспечения редко работают в изолированной среде. Специалисты по обработке данных и эксперты в сфере аналитики делятся друг с другом опытом использования имеющихся в их распоряжении ресурсов. Потребность в совместной работе и обмене успешным опытом особенно велика в области развивающихся технологий, и отсутствие возможностей для такой работы может дорого обойтись. Использование инструментов, обеспечивающих совместную работу, даст значительные преимущества крупным организациям. Очень часто сотрудники, выполняющие смежные или похожие задачи, не знают о работе друг друга, что приводит в лучшем случае к дублированию результатов. Это напрасный расход не только денег, но и ресурсов. Быстрый запуск проекта на основе существующих решений может значительно повлиять на качество продукта и время выхода на рынок.

Хорошим примером платформы приложений является OpenChorus (www. openchorus.org). Помимо быстрого развертывания приложений для анализа больших данных она также поддерживает совместную работу и многие другие функции, важные для разработчиков программного обеспечения. Например, она обеспечивает интеграцию инструментов, контроль версий и управление конфигурациями.

OpenChorus — это проект, поддерживаемый EMC Corporation и доступный в рамках лицензии Apache 2.0. EMC также создает и предоставляет коммерческую версию Chorus. И OpenChorus, и Chorus имеют динамичные сети партнеров, а также большое количество индивидуальных и корпоративных соавторов.

OpenChorus — это базовая платформа. Ее основным преимуществом является возможность создания узла для общего доступа к источникам больших данных, соответствующей аналитике, методам

анализа и визуализациям. OpenChorus предоставляет следующие возможности:

- репозиторий инструментов анализа, артефактов и методов с полной поддержкой версий, отслеживанием изменений и возможностями архивации;
- самообеспечивающиеся и простые в управлении рабочие пространства и среды тестовых испытаний;
- визуализацию, включая тепловые карты, временные ряды, гистограммы и т.д.;
- федеративный поиск по всем ресурсам данных, включая Hadoop, метаданные, репозитории SQL и комментарии;
- совместную работу через функции, напоминающие социальные сети, которые позволяют делиться информацией и осуществлять мозговые штурмы;
- возможности расширения для интеграции сторонних компонентов и технологий.

По мере развития больших данных будут внедряться все новые разновидности платформ приложений. Одни будут ориентированы на разработку мобильных приложений, другие — на работу с вертикальными приложениями. В любом случае, это важный инструмент на ранних этапах внедрения больших данных.

От большого к малому: парадокс больших данных

Есть один нюанс, связанный с анализом больших данных. Как ни парадоксально, это малые данные. Малые данные являются продуктом анализа больших данных. Это не новая концепция, малые данные хорошо знакомы любому аналитику данных. Среда работы с данными расширилась в целом, но ответы все равно лежат в сфере «малой» аналитики.

Традиционный анализ данных начался с баз данных, в которых содержались сведения о клиентах, продуктах, транзакциях, данные телеметрии и т.д. Даже тогда объем данных перекрывал возможности эффективного анализа. Системы, сети и программы не обеспечивали достаточной производительности и мощности — они не соответствовали масштабам анализируемых данных. Для решения этой проблемы большие наборы данных разбивали на части.

Однако и эти меньшие наборы данных были довольно обширными; обнаружились и новые проблемы, основной из которых было несоответствие между данными и спецификой решаемой задачи. Сотрудник отдела расчетов с поставщиками в поисках нужной информации прорабатывал сравнительно большое количество данных, не связанных напрямую с его задачами. И снова отрасль реагировала на это путем создания меньших, контекстуально связанных наборов данных: от большого к малому и еще меньшему.

Легко связать этот процесс с переходом от баз данных к хранилищам и витринам данных. Во многих случаях данные для хранилищ и витрин данных выбирались на основе произвольных или экспериментальных параметров, которые, в свою очередь, формировались методом проб и ошибок. Необходимые или просто возможные результаты не достигались, так как снижение производительности не было связано с вычислительными мощностями.

Но вот появились большие данные с огромными скоростями, объемами и разнообразием, и проблема усугубилась. Совершенствование инфраструктуры позволило хранить и обрабатывать большие объемы дополнительных данных — но одновременно потребовалось внедрить новые технологии, чтобы управлять большими данными.

И все же, несмотря на сложности, это замечательная тенденция. Мы не можем себе вообразить объемы данных, с которыми работаем, но обладаем возможностями сохранять и контролировать их. На этом этапе важнее всего способность своевременно анализировать именно те данные, которые нужны для принятия решения. Нам по-прежнему требуется сокращать наборы данных до оптимальных размеров, но теперь это можно делать автоматически. Таким образом, обработка больших данных превращает их в малые данные, которые проще изучать, — но эти малые данные более точны, а поскольку получены из значительно большего объема, то лучше соответствуют характеру решаемых задач.

Часть V

Применение больших данных



Применение больших данных

В ЭТОЙ ЧАСТИ:

- Определение и интеграция нужных данных для реализации больших данных
- Оценка точности результатов
- Потоковые данные
- Обработка сложных событий
- Ввод больших данных в эксплуатацию
- План действий по внедрению больших данных

Глава 15

Интеграция источников данных

В ЭТОЙ ГЛАВЕ:

Определение необходимого круга данных
Общие основы интеграции больших данных
Использование Hadoop в качестве ETL
Обзор передового опыта интеграции данных

Чтобы извлечь максимальную выгоду для бизнеса от использования больших данных, эти данные должны быть интегрированы в бизнес-процессы. Результаты анализа больших данных необходимо интерпретировать в контексте рабочих данных. Приобретение конкурентных преимуществ за счет более качественных бизнес-решений зависит от ряда факторов, и среди них на первый план выходит интеграция внутренних и внешних источников данных. Обычно это традиционные реляционные данные и более новые формы неструктурированных данных. На первый взгляд задача кажется сложной, но здесь поможет предыдущий опыт интеграции данных, как теоретический, так и практический. Задействуя результаты анализа в процессах принятия решений, необходимо в первую очередь обеспечить высокое качество исходных данных. Однако чтобы свести воедино среду больших данных и среду корпоративных данных, понадобится использовать новые методы интеграции, которые поддерживают Hadoop и другие нетрадиционные среды больших данных.

В этой главе освещаются две основные категории процессов интеграции больших данных: интеграция нескольких источников больших данных в средах больших данных и интеграция неструктурированных

источников больших данных вместе со структурированными корпоративными данными. Мы опишем традиционные формы интеграции, такие как ETL (извлечение, преобразование и загрузка), а также расскажем о новых решениях, предназначенных для платформ больших данных.

Определение необходимого круга данных

Прежде чем приступить к планированию интеграции больших данных, необходимо составить перечень всех типов данных, с которыми вам предстоит иметь дело. Зачастую оказывается, что компания раньше не использовала весь потенциал внутренних данных. Новые инструменты анализа позволяют получать новые результаты благодаря работе с внутренними источниками неструктурированных данных, которые прежде не находили применения в этой сфере. В их числе сообщения электронной почты, записи об обслуживании клиентов, показания датчиков и сведения из журналов безопасности. Кроме того, растет потребность в анализе данных из внешних источников, от данных социальных сетей и геолокации с мобильных телефонов до транспортных сводок и прогнозов погоды.

Анализ может потребовать агрегации ряда таких источников данных. Прежде чем приступить к анализу, необходимо собрать большие объемы данных из файлов регистрации, сообщений Twitter, от тегов радиочастотной идентификации, метеорологических показаний и т.д., а затем интегрировать все эти элементы в системы распределенных данных. После завершения анализа большие данные может потребоваться интегрировать с операционными данными. Например, исследователи в области здравоохранения изучают неструктурированную информацию из медицинских карт пациентов в сочетании с традиционными данными, такими как результаты обследований, и применяют результаты анализа для повышения качества медицинского обслуживания. В такой анализ можно включить и источники больших данных, например, информацию с медицинских приборов и данные клинических экспериментов.

Приступая к анализу больших данных, трудно предсказать, какими окажутся его результаты. Вначале объемы обрабатываемой информации могут составлять несколько петабайт, но по мере выявления зако-

номерностей будут сужаться. Процесс анализа включает три основных этапа:

- исследование;
- кодификацию;
- интеграцию и инкорпорирование.

Этап исследования

На начальных этапах анализа необходимо выявить закономерности в данных. Новые и неожиданные отношения и корреляции между элементами могут стать очевидными только после изучения очень больших объемов данных (терабайты и петабайты). Например, такие закономерности могут рассказать о предпочтениях клиентов касательно нового продукта. Чтобы упорядочить большие данные для поиска этих тенденций, понадобится платформа, такая как Hadoop.

Как уже было рассказано в главах 9 и 10, Hadoop широко используется в качестве базового структурного элемента, предназначенного для сохранения и обработки больших данных. Возможности платформы Hadoop позволяют ускорить обработку больших данных и выявить тенденции в огромных массивах данных, затрачивая на это относительно небольшое время. Два основных компонента Hadoop — распределенная файловая система Hadoop (HDFS) и MapReduce — используются для управления и обработки больших данных.

На этапе исследования проблема интеграции с операционными данными еще не стоит. Об этом нужно будет задуматься в дальнейшем.

Использование FlumeNG для интеграции больших данных

Но одна разновидность интеграции на этом этапе все же необходима. Для поиска скрытых закономерностей в массивах больших данных нужно собрать, агрегировать и переместить огромное количество потоковых данных. Традиционные инструменты интеграции, такие как ETL, не обеспечивают достаточной скорости перемещения больших потоков данных, необходимой для выполнения ответственных задач в режиме реального времени. С этим справляется FlumeNG (расширенная версия исходного приложения Flume), загружая данные в режиме реального времени путем потоковой передачи данных в Hadoop.

Как правило, Flume используется для сбора больших объемов данных из журналов регистрации с распределенных серверов. Flume отслеживает все физические и логические узлы, которые указаны при установке. Узлы агентов устанавливаются на серверах и несут ответственность за управление передачей отдельных потоков данных и их обработку. Кроме того, сборщики данных группируют данные в более крупные потоки, которые записываются в файловую систему Hadoop или в другой контейнер хранения больших данных. Flume обеспечивает масштабирование и может постоянно добавлять дополнительные ресурсы в систему для эффективной обработки больших объемов данных. Выходные данные Flume могут быть интегрированы с Hadoop и Hive для анализа. Flume также имеет функции преобразования, которые можно применять к данным, что позволяет превратить инфраструктуру Hadoop в источник потоковых неструктурированных данных.

Поиск закономерностей в больших данных

Можно найти немало примеров тому, как компания начинает активно использовать конкурентные преимущества, получаемые благодаря аналитике больших данных. Для многих организаций потоки данных из социальных медиа становятся неотъемлемым компонентом стратегии цифрового маркетинга. Например, Wal-Mart анализирует пользовательские данные на основе местоположения, сообщения в Twitter и других социальных медиа, чтобы сформировать целевые рекомендации по продуктам, а также обеспечить наличие в продаже товаров, которые пользуются спросом. В 2011 году компания Wal-Mart приобрела Kosmix, компанию, занимающуюся социальными медиа, и получила доступ к ее технологической платформе для поиска и анализа потоков данных в режиме реального времени. На этапе исследования эта технология может использоваться для быстрого поиска по большим объемам потоковых данных и нахождения закономерностей, связанных с определенными продуктами или клиентами. Результаты могут использоваться для оптимизации запасов определенных продуктов на основе мнения покупателей в том или ином районе.

По мере поиска закономерностей в больших данных отфильтровывается неподходящая информация, и огромный объем данных постепенно сокращается.

Этап кодификации

Чтобы перейти от идентификации закономерностей к применению полученных результатов в бизнес-процессах, необходимо следовать определенной процедуре. Например, крупная розничная компания обнаруживает в социальных медиа многочисленные разговоры о предстоящем футбольном матче на стадионе рядом с одним из магазинов. Как воспользоваться этой информацией? При наличии сотен магазинов и многих тысяч клиентов необходимо использовать повторяемый, воспроизводимый процесс, который позволит в дальнейшем быстро переходить после определения закономерности к реализации новых продуктов и целевому маркетингу. Если процесс налажен, розничная компания быстро закупит и поставит в нужную торговую точку товары с символикой команд. Любые интересные результаты анализа больших данных необходимо кодифицировать и сделать их частью общего бизнес-процесса, в частности, важно обеспечить взаимодействие между аналитикой больших данных и системами инвентаризации и поставки продуктов.

Чтобы кодифицировать отношение между результатами анализа больших данных и операционными данными, эти данные нужно интегрировать.

Этап интеграции и инкорпорирования

Большие данные оказывают значительное влияние на многие аспекты управления данными, включая интеграцию данных. Традиционно основное внимание при интеграции данных уделялось задачам перемещения данных посредством промежуточного ПО, включая определение характеристик передаваемых сообщений и требований к интерфейсам программирования приложений (API). Эти методы подходят для статичных данных, но не для постоянно меняющихся. Работа с неструктурированными и потоковыми данными вынуждает сменить традиционную концепцию интеграции данных. Чтобы встроить анализ потоковых данных в бизнес-процесс, необходима расширенная технология, работающая достаточно быстро, чтобы позволять принимать решения в режиме реального времени. Одна из важнейших целей аналитики больших данных — поиск закономерностей, которые применимы к деятельности конкретной компании, и сужение набора данных до решаемых ею задач. Таким образом, анализ боль-

ших данных — это лишь один шаг реализации. После завершения анализа больших данных потребуется найти подход, позволяющий интегрировать результаты этого анализа в бизнес-процессы и действия компании в режиме реального времени.

Многие компании приветствуют возможность извлечь пользу из анализа больших данных, в том числе внутренних больших данных, таких как журналы безопасности, которые раньше не подвергались анализу из-за технологических ограничений. Применение технологий высокоскоростной передачи очень объемных и быстрых данных становится обязательным для интеграции распределенных источников больших данных. То же относится и к интеграции больших и операционных данных.

Часто нужно быстро перемещать данные из неструктурированных источников на значительные географические расстояния для общего доступа и совместной работы. Такая необходимость возникает в самых разных сферах: от крупных научно-исследовательских проектов и до создания и предоставления развлекательного мобильного контента.

Например, ученые, как правило, работают с очень большими наборами данных. Сегодня исследователи легко получают доступ к данным и организуют совместную работу благодаря сочетанию аналитики больших данных и облачных технологий. Один из примеров большого набора данных, доступных ученым со всего мира, — это проект «1000 геномов». Эпидемиологи исследуют его данные, чтобы определить и скомпилировать различные генетические вариации для изучения различных заболеваний и борьбы с ними. Данные этого объемного каталога генетической вариативности человека по всему миру размещены на веб-службах Amazon Web Services (AWS) и доступны международному научному сообществу. Службы AWS поддерживают передачу очень больших файлов на высоких скоростях по Интернету (700 мегабайт в секунду) с помощью решения, предоставляемого компанией Aspera, — технологии fasp, которая обеспечивает высокоскоростную передачу файлов. Это программное обеспечение переносит большие данные на скоростях, которые гораздо выше технологий передачи файлов на основе TCP-протоколов, таких как FTP и HTTP, и позволяет передавать очень большие по размерам файлы на большие расстояния.

Связка традиционных источников и больших данных — многоэтапный процесс, который начинается после оценки данных, передаваемых с помощью потоков больших данных, и определения соответствующих тенденций. Если в начале анализа его задачи могли быть

нечеткими, то теперь полученная ценная информация позволяет перейти от исследования данных к их практическому применению. Для этого нужно позаботиться о метаданных, правилах и структуре данных, а после сужения объемов данных, необходимых для управления и анализа, следует подумать об их интеграции.

Компания, которая использует большие данные для прогнозирования заинтересованности клиентов в новых продуктах, должна установить связь между большими данными и операционными данными по клиентам и продуктам, чтобы вовремя предпринять нужные действия. В случае если эти сведения нужно использовать для приобретения новых продуктов, изменения ценообразования или управления запасами, необходимо интегрировать операционные данные с результатами анализа больших данных. Сфера розничной торговли — это один из рынков, где компании начинают использовать аналитику больших данных для укрепления отношений с клиентами и создания более персонализированных и клиентоориентированных предложений, и ключевым фактором успеха здесь является интеграция больших и операционных данных. Предположим, клиент регистрируется на веб-сайте торгового центра, предоставляя номер своего мобильного телефона и адрес электронной почты. Сегодня клиент получает сообщения электронной почты о продажах и купоны для покупок товаров в магазинах или через Интернет. В будущем такие магазины планируют задействовать сервисы геолокации: если с мобильного телефона клиента поступили данные, показывающие, что он находится в определенном магазине, клиент получит соответствующее предложение. К примеру, зайдя в отдел электроники, он получит сообщение с купоном на скидку при покупке проигрывателя Blu-ray-дисков. Для этого компания должна обеспечить в режиме реального времени интеграцию потоков больших данных (сведения о местоположении) с операционными данными об истории покупок клиентов и запасах в магазинах. Анализ должен осуществляться незамедлительно и одновременно с установлением связи с клиентом. Десятиминутное опоздание — и взаимодействие с клиентом не сработает.

Общие основы интеграции больших данных

Элементы платформы больших данных позволяют по-новому подходить к управлению данными по сравнению с традиционной реляционной базой данных. Это связано с необходимостью масштаби-

рования и высокой производительностью, которые требуются для управления структурированными и неструктурированными данными. Компоненты экосистемы больших данных, начиная с Hadoop и заканчивая NoSQL DB, MongoDB, Cassandra и HBase, обеспечивают собственный подход к извлечению и загрузке данных. В результате персоналу понадобятся новые навыки, необходимые для управления процессом интеграции этих платформ. Однако многие методы управления данными, которые уже используются на сегодняшний день, станут еще более значимыми по мере вашего перехода в мир больших данных.

Большие данные предоставляют новый уровень комплексной интеграции, но и базовые принципы на них распространяются. Достижение бизнес-целей тесно связано с качественным и надежным использованием данных компании в нужное время и в нужном контексте. Для этого необходимо определить общие правила оценки и обеспечения качества данных, акцентируя внимание на точности и целостности данных. Кроме того, понадобится комплексный подход к разработке корпоративных метаданных, отслеживанию происхождения данных и управлению интеграцией ваших данных.

В то же время развиваются и традиционные инструменты для интеграции данных. Это позволяет обрабатывать постоянно растущее разнообразие неструктурированных данных, справляться с увеличивающимся их объемом и скоростью их изменения. Несмотря на то, что традиционные формы интеграции по-новому определяются в мире больших данных, все технологии интеграции нуждаются в общей платформе, которая поддерживает качество и профилирование данных.

Чтобы принимать обоснованные управленческие решения с использованием результатов анализа больших данных, исходная информация должна быть надежной и понятной на всех уровнях организации. На этапе исследования будет неразумно расходовать значительные финансовые и временные ресурсы, чтобы обеспечить высокое качество данных. Однако в дальнейшем качество и надежность должны играть важную роль, если результаты будут использоваться для решения задач, стоящих перед бизнесом. Информация должна предоставляться всем подразделениям компании надежным, управляемым, постоянным и гибким способом, независимо от требований к отдельным системам или приложениям. Для достижения этой цели применяются три базовых принципа.

- Единая система определений данных. На начальных этапах анализа больших данных трудно обеспечить такой же уровень контроля над определениями данных, как в работе с операционными данными. Но когда будут определены закономерности, наиболее важные для конкретной компании или сферы деятельности, понадобится возможность определять соответствие элементов данных общему определению. Затем это общее определение переносится в операционные данные, хранилища данных и бизнес-процессы.
- Набор служб данных, которые позволят определять качество данных, обеспечивать их согласованность, а значит, и надежность. При интеграции источников неструктурированных и больших данных с операционными данными необходимо быть уверенным в правильности получаемых результатов.
- Упрощенный способ интеграции источников больших данных и систем записи. Чтобы принимать обоснованные решения на основе результатов анализа больших данных, необходимо предоставлять информацию в нужное время и в правильном контексте. Процесс интеграции больших данных должен обеспечить согласованность и надежность данных.

Интеграция данных в средах смешанных приложений означает возможность приема данных из одной среды (источник) и передачу их в другую среду данных (цель). В традиционных средах хранилищ данных для этого использовались технологии извлечения, преобразования и загрузки (ETL). Сейчас роль ETL меняется — эта технология используется для управления данными в новых средах, таких как Hadoop. В среде больших данных может понадобиться объединить инструменты, поддерживающие процессы пакетной интеграции (с помощью ETL), со средствами интеграции и федерации нескольких источников в режиме реального времени. Например, фармацевтической компании потребуется объединить данные, хранящиеся в системе управления основными данными (Master Data Management — MDM), с источниками больших данных о результатах приема пациентами тех или иных лекарств. Компании используют управление данными для упрощения сбора, агрегирования, консолидации и предоставления согласованных и надежных данных в рамках своих бизнес-процессов. Поддержка интеграции сред больших данных осуществляется с помощью новых инструментов, таких как Sqoop и Scribe. О технологии ELT подробнее рассказано ниже.

Определение традиционного понятия ETL

Инструменты ETL (extract, transform, load — извлечение, преобразование, загрузка) объединяют важные функции, необходимые для получения данных из одной среды данных и переноса их в другую среду.

Традиционно ETL используются для пакетной обработки в средах хранилищ данных. Хранилища данных обеспечивают бизнес-пользователю способ консолидации информации из разных источников, например, системы планирования ресурсов предприятия (ERP) и управления отношениями с клиентами (CRM). Это позволяет анализировать и использовать для составления отчетов данные, имеющие отношение к определенным бизнес-задачам. Инструменты ETL используются для преобразования данных в формат, требуемый для хранилища данных. Фактически преобразование осуществляется в некотором промежуточном буфере, до загрузки данных в хранилище. Многие поставщики программного обеспечения, в том числе IBM, Informatica, Pervasive, Talend и Pentaho, предоставляют программные инструменты ETL.

Эти инструменты обеспечивают базовую инфраструктуру для интеграции и выполняют три важные функции:

- извлечение — чтение данных из исходной базы данных;
- преобразование — конвертацию формата извлеченных данных для соответствия требованиям целевой базы данных. Преобразование осуществляется с использованием правил или посредством объединения различных данных;
- загрузку — запись данных в целевую базу данных.

Инструменты ETL продолжают совершенствоваться и теперь поддерживают интеграцию не только традиционных хранилищ данных, но и транзакционных систем, операционных хранилищ данных, платформ бизнес-аналитики, узлов MDM, облака и платформ Hadoop. Поставщики программных решений ETL расширяют возможности, предлагая функции извлечения, преобразования и передачи больших данных между Hadoop и традиционными платформами управления данными. Инструменты ETL и программные средства для процессов интеграции данных, таких как очистка, профилирование и аудит данных, позволяют обеспечить надежность информации. Инструменты ETL интегрируются со средствами обеспечения качества данных, многие из которых обеспечивают очистку и сопоставление, а также идентификацию происхождения данных. Инструменты ETL позволяют извлекать только те данные, которые нужны для интеграции.

Инструменты ETL необходимы для загрузки и преобразования структурированных и неструктурированных данных в Hadoop. Более совершенные инструменты ETL могут считывать и записывать параллельно несколько файлов, взаимодействуя с платформой Hadoop и упрощая объединение данных непосредственно в процессе преобразования. Некоторые решения содержат библиотеки предварительно созданных преобразований ETL для транзакционных данных и данных взаимодействия, которые можно применять в Hadoop или в инфраструктуре вычислительной grid-сети.

Преобразование данных

Преобразование данных — это процесс изменения формата данных для их дальнейшего использования в других приложениях. В частности, формат, в котором хранятся данные, преобразуется в формат, необходимый приложению, которое будет эти данные использовать. Этот процесс также включает функции сопоставления, поэтому приложениям даются инструкции по получению данных, которые необходимо обработать.

Преобразование данных усложнилось из-за стремительного роста объема неструктурированных данных. Бизнес-приложение, например система управления взаимоотношениями с клиентами или система управления продажами, как правило, имеет определенные требования к хранению используемых данных. Скорее всего, данные структурированы и упорядочены в столбцах и строках реляционной базы данных; если столь строгие требования к форматам не предъявляются, то данные будут слабоструктурированными или неструктурированными. К примеру, информация, содержащаяся в сообщении электронной почты, считается неструктурированной. Некоторая важная корпоративная информация может находиться в неструктурированной или слабоструктурированной форме, например документы, сообщения электронной почты, сообщения комплексного формата, записи служб поддержки клиентов, транзакции, а также информация, которая поступает из пакетных приложений, таких как ERP и CRM.

Инструменты преобразования данных изначально не ориентированы на работу с неструктурированными данными. Поэтому, чтобы иметь возможность применять неструктурированную информацию в процессах принятия управленческих решений, приходилось вручную писать достаточно объемный код для выполнения процедур ин-

теграции данных. В настоящее время, учитывая растущую значимость неструктурированных данных для процесса принятия решений, инструменты ETL от крупнейших поставщиков все чаще предоставляют стандартизированные средства преобразования неструктурированных данных. Это облегчает интеграцию неструктурированных и операционных структурированных данных.

ELT (извлечение, загрузка и преобразование)

Инструменты ELT (extract, load, transform — извлечение, загрузка и преобразование) фундаментально отличаются от ETL только порядком выполнения функций. В первых базах данных не было технической возможности преобразовывать данные, поэтому инструменты ETL извлекали данные в промежуточный буфер для выполнения преобразования перед загрузкой данных в хранилище. В настоящее время благодаря развитию таких технологий, как системы масштабной параллельной обработки и столбцовые базы данных, это ограничение уже не действует. Инструменты ELT могут преобразовывать данные в исходной или целевой базе данных без использования ETL-сервера. Они полезны для обработки больших данных, поскольку повышают производительность и упрощают масштабируемость. Для преобразования данных инструменты ELT используют язык структурированных запросов (SQL). Многие традиционные инструменты ETL также предоставляют возможности ELT, поэтому в зависимости от конкретной ситуации можно использовать оба набора инструментов.

Приоритеты качества больших данных

В мире больших данных трудно однозначно оценить надлежащее качество данных.

При работе с большинством данных изначально предполагается, что они не были очищены. Однако именно значительное количество данных, кажущихся случайными, несвязанными и избыточными в информационных потоках социальных медиа, является одним из основных факторов, делающих эти данные полезными для бизнеса. Приступая к поиску закономерностей в петабайтах данных, трудно знать заранее, каким будет результат, но сразу можно предположить, что в этом объеме информации будет очень много «шума». Конечно, боль-

шие данные из некоторых источников, такие как данные радиочастотной идентификации или показания датчиков, будут более упорядоченными, нежели данные из социальных сетей. Данные с датчиков, как правило, относительно чистые, однако в них могут встречаться ошибки. В конечном счете ответственность за обеспечение уровня качества данных при анализе огромных объемов данных несет пользователь. В обеспечении качества данных важны две стадии.

Стадия 1. Поиск закономерностей в больших данных без учета качества самих данных.
Стадия 2. После нахождения закономерностей и формирования результатов, важных для бизнеса, к ним применяют те же стандарты качества данных, которые используются для традиционных источников данных. Следует избегать сбора и обработки тех больших данных, которые не являются важными для бизнеса и могут повредить другие элементы данных в Hadoop или других платформах обработки больших данных.

По мере встраивания результатов анализа больших данных в бизнес-процессы станет понятно, что высокое качество данных является важным фактором для принятия компанией обоснованных управленческих решений. Это относится как к большим, так и к традиционным данным. Качество данных связано с такими их характеристиками, как согласованность, точность, надежность, целостность, своевременность, взвешенность и допустимость. Программные инструменты для обеспечения качества данных формируют однозначное представление элементов данных в различных хранилищах и системах данных, что позволяет повысить согласованность данных.

Например, в одном хранилище данных могут использоваться две строки для хранения адреса клиента, а в другом хранилище — только одна строка. Это различие в представлении данных может привести к появлению неточных сведений о клиентах — например, один и тот же человек будет восприниматься как два разных. У каждой организации есть несколько форм наименования (полная и краткие), и в разных документах могут фигурировать разные варианты. С помощью программы для обеспечения качества данных можно определить все варианты наименования компании в различных хранилищах данных и получить все сведения о взаимодействии с ней. Этот процесс называется обеспечением единого представления по клиенту или продук-

ту. Программа для обеспечения качества данных сопоставляет данные в различных системах и очищает или удаляет избыточные данные. Процесс обеспечения качества данных предоставляет компании сведения, которые проще использовать, интерпретировать и понимать.

Инструменты профилирования данных используются в процессе обеспечения качества данных. Они обеспечивают оценку содержимого данных, их структуру и состояние. Можно сгруппировать данные по характеристикам в базе данных или другом хранилище данных, чтобы потом начать преобразование данных в форму, заслуживающую большего доверия.

Эти инструменты анализируют данные для выявления ошибок и несогласованностей. Они могут вносить изменения и исправлять ошибки. Эти инструменты проверяют данные на соответствие шаблону и на нахождение их значений в допустимых диапазонах. Это позволяет выявлять наложение данных. К примеру, процесс профилирования данных проверяет, какие данные должны быть введены: буквенные или цифровые. Эти инструменты также проверяют зависимости и сравнивают данные с элементами данных, уже содержащимися в других базах данных.

Инструменты профилирования больших данных имеют функции, аналогичные профилированию традиционных данных. Инструменты профилирования данных для Hadoop предоставляют важные сведения о данных в кластерах Hadoop, а также могут использоваться для поиска сопоставлений и удаления дубликатов в больших массивах данных, чтобы в результате обеспечить полноту и согласованность больших данных. Такие инструменты Hadoop, как HiveQL и Pig Latin, могут использоваться в процессе преобразования.

Использование Hadoop в качестве ETL

Многие организации, внедрившие платформы больших данных, обеспокоены тем, что инструменты ETL слишком медленны и сложны для работы с огромными объемами данных. Некоторые считают, что для обработки некоторой части процесса преобразования лучше использовать Hadoop, а для общего улучшения процессов и подготовки данных лучше подходит ETL. Процесс интеграции данных можно ускорить, загрузив и неструктурированные данные, и операционные, и транзакционные данные непосредственно в Hadoop независимо от

исходной структуры. После загрузки в Hadoop данные можно интегрировать с использованием традиционных инструментов ETL. При использовании Hadoop в качестве вспомогательного средства для процесса ETL процесс аналитики ускоряется.

Производители программных средств продолжают работать над реализацией полноценного использования Hadoop в качестве инструмента интеграции. Поставщики традиционных решений ETL, такие как IBM, Informatica, Talend, Pentaho и Datameer, встраивают Hadoop в свои интегрированные предложения. Используя возможности Hadoop как системы масштабной параллельной обработки, разработчики могут реализовывать ранее недоступные функции преобразования и обеспечения качества данных. Однако платформа Hadoop не может сама по себе стать заменой инструментам ETL.

Наилучшие способы интеграции в мире больших данных

Большие данные обладают огромным потенциалом как в научной, так и в бизнес-сфере. Они позволяют изучать задачи с новых точек зрения, искать ответы на неразрешенные вопросы, быстро реагировать на изменения ситуации и достигать выдающихся результатов. В главах 21 и 22 приводятся примеры решений, связанных с большими данными, — и это лишь немногие перспективы новой технологии. Но именно сейчас, делая в ней первые шаги, необходимо обратить внимание на базовые принципы.

Фактически интеграция больших данных встраивается в общий процесс интеграции данных в вашей компании. Нельзя просто отбросить опыт и наработки интеграции традиционных источников данных. К управлению большими данными применяются те же правила. Вот ключевые приоритеты работы с данными.

- Обеспечивайте качество данных с учетом перспектив. Степень внимания к качеству данных зависит от этапа анализа больших данных — так, при выполнении предварительного анализа контролировать качество данных не удастся. Однако после того, как первоначальный объем сужается до определенной выборки самых полезных данных, целесообразно уделить качеству особое внимание. Особенно важно качество данных, если нужно сопоставить полученные результаты с имеющимися архивными данными. По-

скольку аналитика играет все более важную роль в планировании, качество данных может гарантировать успех или стать причиной неудачи.

- Учитывайте требования обработки данных в режиме реального времени. На передний план работы с большими данными выходят потоковые данные. Поэтому необходимо четко понять, как интегрировать перемещаемые данные в среду компании для целей прогнозного анализа.
- Не создавайте новые обособленные хранилища информации. Несмотря на пристальное внимание, которое уделяется Hadoop и другим неструктурированным и слабоструктурированным источникам, необходимо управлять этими данными исходя из особенностей и задач конкретной компании, поэтому такие источники данных должны быть интегрированы с ее операционными данными и хранилищем данных.

Работа с потоками данных в режиме реального времени и обработка сложных событий

В ЭТОЙ ГЛАВЕ:

Потоковая передача данных
Специфика обработки сложных событий
Влияние потоковых данных и обработки сложных событий на большие данные
Использование потоковых данных и обработки сложных событий на практике
Потоковые данные в действии

Большие данные создают новые перспективные подходы к предоставлению качественно нового уровня анализа и более высокой степени детализации при управлении данными. Во многих примерах, рассмотренных в предыдущих главах, большие данные анализировались в пакетном режиме. Зачастую главная трудность заключается в том, что объемы данных чрезвычайно велики, но при этом их надо обрабатывать и контролировать с нужной скоростью, чтобы вовремя получать нужные результаты. Однако такая аналитика связана в основном с крупномасштабным анализом и процессом принятия решений. Анализ огромных объемов данных из многочисленных источников может дать организации представление о значении данных, поможет составить планы на будущее, позволит предвидеть изменения на рынке и смену потребительских предпочтений.

Зачастую данные требуют быстрого реагирования — например, идентифицировать проблему с медицинским оборудованием, отправить клиенту купон во время осуществления покупки или скорректировать размещение оборудования из-за меняющихся погодных условий. Эти разноплановые задачи объединяет необходимость анализировать данные в процессе их перемещения. Вот как сформулировал проблему один из крупных руководителей: «Нам необходимо иметь возможность обрабатывать и анализировать потоковые данные с датчиков в режиме реального времени, то есть когда данные все еще находятся в движении».

Эта глава посвящена двум методам управления потоком данных. Технология потоковой передачи тесно связана с объемом данных, тогда как для обработки сложных событий объем имеющихся данных играет второстепенную роль по сравнению с необходимостью сопоставления данных с правилами. Потоковая передача данных и обработка сложных событий приобретают все большее значение во многих сферах деятельности. Эти технологии особенно важны «на передовой», в ситуациях, где успех или поражение зависят от быстроты реагирования.

Введение в потоковые данные и обработку сложных событий

Итак, что такое потоковая передача данных, что такое обработка сложных событий и чем они отличаются? На эти вопросы нелегко дать ответ, так как управление данными является непрерывным процессом. Потоковые вычисления предназначены для обработки постоянного потока больших объемов неструктурированных данных, в то время как обработка сложных событий (Complex Event Processing — CEP), как правило, связана с несколькими переменными, которые должны быть соотнесены с определенным бизнес-процессом. Во многих случаях обработка сложных событий зависит от потоков данных, однако для потоковой передачи данных обработки сложных событий не требуется. Как и потоковая передача данных, обработка сложных событий связана с анализом данных, передаваемых в потоках. Обе технологии применимы только к данным в движении (передаваемым данным), но не к данным в покое (хранимым данным). Рассмотрим потоковую передачу данных и ее применение в различных сферах деятельности.

Применение потоковых данных

До сих пор мы говорили о сборе огромных объемов данных из разных источников и обработке этой информации для получения ценных аналитических сведений. В целом такие получаемые данные считаются данными в покое. А как же выглядят данные не в покое? Они связаны с системами, которые управляют активными транзакциями, поэтому должны характеризоваться сохраняемостью. В этих случаях данные будут сохранены в операционном хранилище данных. Однако в других случаях анализируются данные уже выполненных транзакций, как правило, размещенные в хранилище или витрине данных. Это означает, что информация обрабатывается в пакетах, а не в режиме реального времени. При планировании на перспективу организации важно анализировать все виды данных, начиная с того, о чем говорят клиенты, и заканчивая тем, что эти клиенты хотят купить и почему. Важно понимать типичные индикаторы изменений. Если потребительские предпочтения клиента меняются, как это повлияет на продукты и услуги, которые будет предлагать организация в следующем году или даже через три года? Многие исследовательские организации используют этот тип аналитики больших данных для разработки новых лекарственных средств. Страховые компании соотносят закономерности возникновения дорожно-транспортных происшествий со сводками погоды в широком географическом охвате. В вышеприведенных примерах управление информацией в режиме реального времени не дает никаких преимуществ. Очевидно, что анализ должен выполняться достаточно быстро с практической точки зрения, кроме того, организации нередко анализируют данные несколько раз, чтобы выяснить, не появились ли новые тенденции.

Однако если необходимо быстро обработать значительный объем данных в режиме, близком к реальному времени, чтобы получить ценные аналитические сведения, лучшим решением станет работа с перемещаемыми данными в виде потоковой передачи данных.

Потоковая передача данных

Обработка потоковых данных связана с использованием аналитической вычислительной платформы, предназначенной для повышения скорости обработки данных. Соответствующие приложения создают постоянные потоки зачастую неструктурированных данных, кото-

рые требуют обработки. Поэтому данные непрерывно анализируются, преобразуются в памяти, а затем сохраняются на диске. Обработка потоков данных выполняется путем обработки «временных окон» данных в памяти на кластере серверов, таким же образом как Hadoop управляет данными в покое (см. главу 9). Основным отличием является скорость. В кластере Hadoop данные собираются в пакетном режиме, а затем обрабатываются. Скорость имеет меньшее значение для платформы Hadoop, чем для обработки потоковых данных. Некоторые ключевые принципы определяют, когда лучше всего применить потоки данных:

- определение возможности розничной покупки в момент взаимодействия с клиентом через социальные сети либо личные сообщения (при получении от клиента такого разрешения);
- сбор информации о перемещении по режимному объекту;
- ситуации, требующие быстрого реагирования, например отказ системы или изменение состояния пациента;
- вычисление затрат на основе таких переменных, как использование и доступность ресурсов, в режиме реального времени.

Потоковая передача данных эффективна, когда аналитика должна выполняться в режиме реального времени, пока данные находятся в движении. Ценность анализа, а зачастую и данных, со временем уменьшается. Например, невозможность своевременно провести анализ и принять меры может привести к тому, что сделка сорвется или что угроза останется незамеченной.

Рассмотрим несколько примеров.

На электростанции должна быть обеспечена высокая степень безопасности и исключено проникновение на территорию посторонних. Обычно по периметру объекта размещают датчики движения. Однако тут кроется проблема. Есть большая разница между зайцем, бегущим вдоль ограды, и машиной, которая проезжает близко к станции на высокой скорости. Поэтому огромный объем данных, поступающих с этих датчиков, должен анализироваться в режиме реального времени, чтобы поднимать тревогу только при наличии реальной угрозы.

Телекоммуникационная компания, работающая на рынке с высокой конкуренцией, стремится отслеживать все случаи перебоев в обслуживании, чтобы оперативно решать проблемы. Телекоммуникационные системы генерируют огромные объемы данных, которые должны анализироваться в режиме реального времени, чтобы была

возможность предпринять соответствующие меры. Несвоевременное обнаружение ошибки может серьезно повлиять на удовлетворенность клиентов.

Буровая компания перед развертыванием нефтяной платформы должна точно знать, где находятся залежи нефти и какие внешние факторы могут повлиять на работу, поэтому компания собирает сведения о глубине и температуре воды, движении льда и т.п. Во избежание ошибок необходимо обработать и проанализировать огромные объемы данных.

Группе диагностов необходимо обрабатывать огромные объемы данных томографии мозга и анализировать результаты в режиме реального времени, чтобы установить причину заболевания и быстро помочь пациенту.

Итак, речь идет об огромном объеме данных, который нужно обрабатывать и анализировать в режиме реального времени. Поэтому очень важна физическая среда, обеспечивающая соответствующее реагирование. Среды потоковой передачи данных, как правило, нуждаются в кластеризованном аппаратном решении, а иногда для анализа требуется масштабная параллельная обработка. Значимая особенность анализа потоковых данных в том, что такой анализ является однопроходным, — другими словами, когда передача данных завершилась, повторно проанализировать эти данные уже нельзя. Это критически важно для приложений, отслеживающих наличие и отсутствие данных. К примеру, в телекоммуникационной сети реакция на отсутствие сигнала о нормальном функционировании должна быть своевременной. Если данные все же требуется анализировать многократно, их придется поместить в особое хранилище, где будет выполнен дополнительный анализ. Например, часто это необходимо для сравнения потоковых данных с архивными. Такое сопоставление может многое рассказать о долговременных тенденциях и их важности для компании.

Необходимость наличия метаданных в потоках

Большинство специалистов по управлению данными знакомы с необходимостью управлять метаданными в средах управления структурированными базами данных. Эти источники данных являются строго типизированными (например, первые десять символов являются именем) и предназначены для работы с метаданными.

Можно было бы предположить, что у неструктурированных данных нет метаданных, но это не так. Структура присутствует в любом виде данных. Возьмем, к примеру, видео: можно не знать содержания ролика, но форматы видеоданных обладают определенной структурой. Посмотрев на неструктурированный текст, можно понять, на каком языке он написан, и применить подходящие инструменты и алгоритмы для его интерпретации. Работа с неструктурированными данными описывалась в главе 13.

Так как в неструктурированных данных имеются (хоть и не явные) метаданные, можно синтаксически анализировать эту информацию с помощью языка XML. XML представляет собой метод представления файлов неструктурированного текста с использованием тегов, придающих данным определенное значение. Технология в основе XML была одной из основополагающих для реализации клиентоориентированных сервисов.

Примерами продуктов, предназначенных для работы с потоковыми данными, являются IBM's InfoSphere Streams, Twitter's Storm и Yahoo's S4.

IBM InfoSphere Streams

InfoSphere Streams обеспечивает непрерывный анализ огромных объемов данных.

Это решение предназначено для выполнения комплексной аналитики гетерогенных типов данных, включая текст, изображения, аудио, голос, VoIP, видео, веб-трафик, электронные сообщения, данные GPS, данные финансовых операций, спутниковые данные и показания датчиков. Infosphere Streams может поддерживать все типы данных. Это приложение может выполнять анализ в режиме реального времени, прогнозный анализ регулярно генерируемых данных с использованием цифровой фильтрации, анализ закономерностей и корреляций, а также декомпозицию и геопространственный анализ.

Twitter's Storm

Twitter's Storm — это модуль аналитики режима реального времени с открытым исходным кодом, разработанный компанией BackType, которую Twitter в 2011 году купил в том числе и потому, что Twitter использует Storm в своей системе. Проект по-прежнему доступен в виде открытого исходного кода и пользуется значительной популярностью среди развивающихся компаний. Это решение может быть использо-

вано с любым языком программирования для таких приложений, как аналитика в режиме реального времени, непрерывные вычисления, распределенные удаленные вызовы процедур (RPC) и интеграция. Решение Storm разработано для работы с существующими технологиями запросов и баз данных. Storm используют в своих приложениях для работы с большими данными такие компании, как Groupon, RocketFuel, Navisite и Oolgala.

Apache S4

Четыре S в названии расшифровываются как Simple Scalable Streaming System — простая масштабируемая потоковая система. Apache S4 — разработка компании Yahoo!, распределенная масштабируемая, частично отказоустойчивая подключаемая платформа общего назначения. S4 позволяет программистам разрабатывать приложения для обработки непрерывных потоков данных. Ядро платформы написано на Java и было выпущено компанией Yahoo! в 2010 году. Через год она была предоставлена Apache в рамках лицензии Apache 2.0. Клиентские приложения для отправки и приема событий могут быть написаны на любом языке программирования. S4 разработана как система с высокой степенью распределения. Ее пропускная способность может быть увеличена линейно за счет добавления в кластер новых узлов. Apache S4 больше всего подходит для крупномасштабных приложений интеллектуального анализа данных и машинного обучения в производственной среде.

Использование обработки сложных событий

И обработка потоков, и обработка сложных событий предназначены для управления перемещаемыми данными. Однако использование этих двух технологий сильно различается. Обработка потоков предназначена для анализа больших объемов данных в режиме реального времени. Обработка сложных событий является инструментом отслеживания, анализа и обработки данных по мере возникновения событий. Эта информация подвергается дальнейшей обработке и управляется на основе бизнес-правил и описаний процессов. Основная концепция обработки сложных событий — это возможность установки корреляции между потоками информации и сопоставление выявляемых закономерностей с определенным поведением, таким как устранение угрозы или использование возможности.

Обработка сложных событий — это расширенный метод, основанный на обработке простых событий, который собирает и объединяет данные из различных соответствующих источников, чтобы найти события и тенденции, которые могут привести к определенному действию.

Приведем пример. Сеть розничной торговли создает многоуровневую программу лояльности, чтобы повысить количество повторных продаж. Программа прежде всего ориентирована на клиентов, которые тратят более 1000 долларов США в год.

Для компании важно создать платформу, которая будет побуждать этих критически важных клиентов опять совершать покупки и тратить больше денег. При использовании платформы обработки сложных событий, как только «ценный» клиент воспользуется программой лояльности, система запустит соответствующий процесс, и клиенту будет предложена дополнительная скидка на сопутствующий продукт. Другое правило обработки событий может предоставить клиенту неожиданный сюрприз — дополнительную скидку или образец нового продукта. Компания также добавляет новую функцию, которая связывает программу лояльности с мобильным приложением. Когда лояльный клиент проходит мимо магазина, ему приходит текстовое сообщение с предложением скидки. В то же время, если лояльный клиент оставит негативный отзыв в социальных медиа, отдел по работе с клиентами получит уведомление, и клиенту будут принесены извинения. Итак, компания будет иметь дело с огромным объемом данных о взаимодействиях с большим числом клиентов. Однако простого механизма анализа потоковых данных в этом случае будет явно недостаточно. Чтобы осуществить поставленные задачи, потребуется запустить процесс, который будет адекватно реагировать на результаты, полученные в рамках анализа.

Многие отрасли используют преимущества обработки сложных событий. Компании — эмитенты кредитных карт применяют обработку сложных событий, чтобы бороться с мошенничеством. При выявлении признаков мошеннических действий компания может заблокировать кредитную карту до того, как понесет значительные убытки. Базовая система сопоставляет входящие транзакции, отслеживает поток данных событий и при необходимости запускает процесс. Обработка сложных событий реализуется в том числе в финансовых торговых приложениях, приложениях прогноза погоды и приложениях для управления продажами. Общее для всех этих приложений — наличие заранее установленной нормы (температуры, давления, размера тран-

закции, объема продаж и т.п.), и как только значение параметра выходит за пределы этой нормы, инициируется действие. Так, в современных автомобилях предусмотрены индикаторы давления в шинах.

Решения для обработки сложных событий предлагает ряд поставщиков. Многие представленные на рынке инструменты по обработке сложных событий позволяют создавать приложения, работающие в режиме реального времени, и реагировать на возникающие события. Эти приложения могут загружать данные из потоков, но также поддерживают работу с традиционными базами данных. Большинство предложений имеет такие возможности, как графическая среда разработки, основанная на преимуществах Eclipse, подключение к потокам данных в режиме реального времени, а также интерфейсы API для доступа к источникам архивных данных. Большинство этих продуктов включает язык графического отображения потока событий и поддерживает SQL. Ключевые поставщики в этой сфере — Esper (поставщик решений с открытым исходным кодом), IBM с продуктом IBM Operational Decision Manager, Informatica с RulePoint, Oracle с Complex Event Processing Solution, Microsoft StreamInsights и SAS DataFlux Event Stream Processing Engine, а также Streambase CEP. На этом рынке появилось также множество продуктов, предлагаемых компаниями-стартапами.

Различия между обработкой сложных событий и потоковыми данными

Итак, в чем разница между решениями для обработки сложных событий и потоковых данных? Потоковые вычисления, как правило, применяются для анализа больших объемов данных в режиме реального времени. Обработка сложных событий ориентирована на решение сцецифических задач, связанных с возникновением событий и действий. В то же время анализ потоковых данных часто используется в приложениях для обработки сложных событий. Как отмечалось выше, приложения для потоковой передачи данных часто управляют большим количеством данных и обрабатывают их с высокой скоростью. В связи с большим объемом данных они, как правило, управляются в хорошо распределенной, кластеризованной среде.

С другой стороны, обработка сложных событий обычно не требует управления таким объемом данных, поэтому может осуществляться на менее сложном оборудовании. Кроме того, отличаться будет и тип

анализа. Очень важно, чтобы приложения по обработке сложных событий могли подключаться к ключевым системам учета данных, т.е. к системам управления взаимоотношениями с клиентами (CRM) или к средам управления транзакциями. Среды обработки сложных событий часто имеют дело лишь с небольшим количеством переменных, которые используются в очень сложных моделях и процессах. Системы обработки сложных событий применяют сложный интеллектуальный анализ или статистические модели, но в основе их функционирования лежит механизм выполнения правил, который запускает обусловленное действие при возникновении события.

Значение потоковых данных и обработки сложных событий для бизнеса

И обработка потоковых данных, и обработка сложных событий заметно влияют на то, как компании используют большие данные для достижения стратегических целей. При обработке потоковых данных компании могут анализировать эти данные в режиме реального времени, чтобы незамедлительно получить результаты анализа. Зачастую требуется организовать двухэтапный процесс, в продолжение которого будут проанализированы полученные результаты, которые ранее могли остаться незамеченными. Благодаря обработке сложных событий компании могут организовывать потоки данных, а затем анализировать их с точки зрения бизнес-правил и бизнес-процессов. Возможности получить в результате анализа новые знания, которые приведут к появлению инноваций или новым направлениям деятельности, — вот принципиальная ценность использования анализа потоковых данных.

Глава 17

Использование больших данных в операционной деятельности

В ЭТОЙ ГЛАВЕ:

Большие данные как часть вашего бизнеса и операционной деятельности
Понимание операций с потоками больших данных
Гарантия валидности, достоверности и волатильности больших данных

Большие данные в бизнесе способны создать значительные преимущества. Но зачастую их применение вызывает ряд вопросов. Как сделать большие данные частью бизнес-процессов, чтобы их можно было эффективно использовать? Что, если объединить традиционный процесс принятия решений с анализом больших данных? Как сделать большие данные доступными для специалистов, ответственных за принятие решений, чтобы они могли воспользоваться преимуществами бесчисленных источников данных для улучшения бизнес-процессов? Чтобы сделать большие данные частью общего процесса управления данными, прежде всего необходимо составить план. В этой главе мы поговорим о том, что нужно для объединения результатов анализа больших данных с существующими операционными данными. Это сочетание может стать мощной основой для вывода вашего бизнеса на принципиально иной уровень.

Большие данные как часть операционной деятельности

Лучший способ сделать большие данные частью бизнес-процессов — это начать с планирования стратегии интеграции. Данные, будь то традиционный источник данных или большие данные, должны быть интегрированы таким образом, чтобы стать естественной частью внутренних рабочих процессов вашей организации.

Могут ли большие данные быть вспомогательными инструментами в бизнес-процессах? Да, могут, но только если практически или полностью отсутствует зависимость между транзакционными и большими данными. Конечно, работа с большими данными может быть и параллельной деятельностью. Однако для максимально эффективного использования больших данных их следует интегрировать в существующие операционные бизнес-процессы.

Интеграция больших данных

Доступ к источникам больших данных еще не решает задачу их использования. Как сориентироваться в петабайтах данных и сотнях механизмов доступа к ним? Какие потоки данных и какие типы данных выбрать? «Нужные» источники данных определяются простым и испытанным алгоритмом:

- понять проблемы, которые надо решить;
- определить задействованные процессы;
- определить информацию, необходимую для решения проблемы;
- собрать данные, обработать их и проанализировать результаты.

Этот алгоритм в разных вариациях используется в бизнесе десятки лет. Так есть ли принципиальная разница в его применении к большим данным? Да, есть. Хотя большие объемы данных нам уже не в новинку, большие данные внесли в нашу профессиональную и личную жизнь новые типы данных. Ленты Twitter и Facebook, данные с датчиков, журналы безопасности, видеоданные и многие другие источники информации возникают и обновляются практически ежедневно. С развитием таких источников больших данных активизируется и поиск возможностей их применения для улучшения взаимодействия с клиентами, партнерами и поставщиками, а также для прогнозирования. Рассмотрим пример, демонстрирующий важность интеграции больших данных с операционными процессами.

В сфере здравоохранения постоянно генерируются данные в различных формах и в гораздо больших объемах, чем в других областях деятельности. Неудивительно, что именно в этой сфере использование новых форм больших данных в перспективе будет наиболее эффективно. В настоящее время медицинские специалисты, страховые и исследовательские организации часто делают выбор в пользу того или иного варианта лечения пациента на основе неполных или нерелевантных данных. Отчасти причина кроется в том, что очень трудно эффективно собирать и обрабатывать данные по каждому пациенту. Нередко элементы данных сохраняются и управляются различными организациями в самых разных местах. Кроме того, по всему миру ведутся клинические исследования, которые накапливают эмпирические данные о различных заболеваниях и подходах к их лечению. Большие данные могут в этом помочь.

Итак, применим наш алгоритм к стандартному сценарию использования данных в сфере здравоохранения (рис. 17.1).

1. Проблема:
    а) Лечение пациента с определенным типом рака
2. Задействованные процессы:
    а) Диагноз и анализы

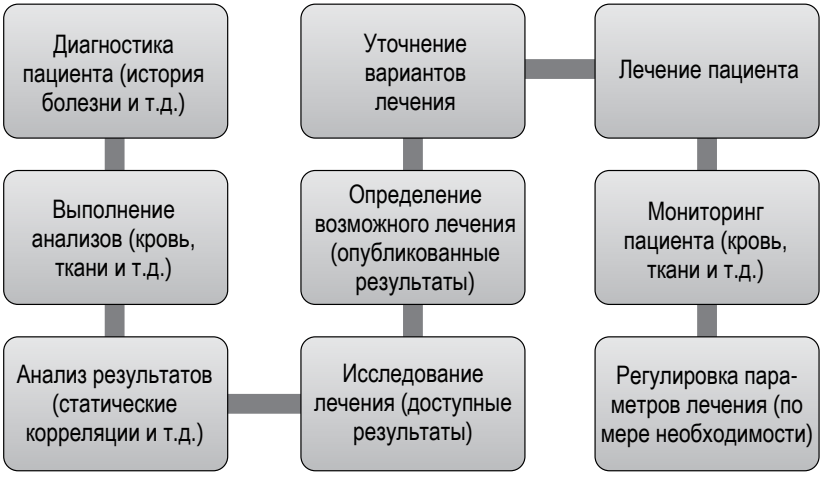


Рис. 17.1. Последовательность традиционного процесса диагностики пациентов

б) Анализ результатов, включая исследование вариантов лечения
в) Определение лечебного протокола
г) Мониторинг состояния пациента и корректировка лечения при необходимости

3. Сведения, необходимые для решения проблемы:
   а) История болезни
   б) Кровь, ткани, результаты анализов и т.д.
   в) Статистические результаты вариантов лечения
4. Сбор данных, их обработка и анализ результатов:
   а) Начало лечения
   б) Мониторинг пациента и корректировка лечения при необходимости.

Так ведется работа с пациентами сегодня. Большинство данных хранится локально в сети медицинского учреждения, а докторам не хватает времени, чтобы выходить за пределы доступных данных и искать новейшую информацию, включая практические рекомендации.

Внедрение систем больших данных в практику медицинской диагностики

Источники больших данных для здравоохранения создаются по всему миру и становятся доступными для интеграции в существующие процессы. В повседневных процессах медицинского учреждения могут быть использованы данные клинических испытаний, генетических исследований, белковой терапии и др. Социальные медиа могут применяться для дополнения существующих данных и процессов и обеспечения более персонализированного подхода к лечению и терапии. Новые медицинские приборы будут контролировать лечение и передавать данные телеметрии для их изучения в режиме реального времени и других типов аналитики. Задача состоит в том, чтобы освоить эти новые источники данных и дополнить существующие данные и процессы новыми типами больших данных.

Итак, как будет выглядеть рассматриваемый процесс в сфере здравоохранения после начала использования больших данных для диагностики и управления лечением пациентов (рис. 17.2)?

1. Проблема:
   а) Лечение пациента с определенным типом рака

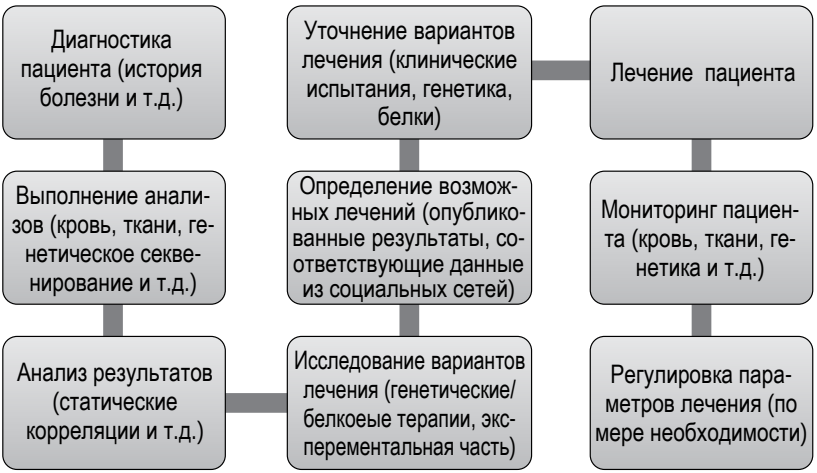


Рис. 17.2. Процесс диагностики в сфере здравоохранения с использованием больших данных

2. Задействованные процессы:
   а) Диагностика и анализы (определение генетических мутаций)
   б) Анализ результатов, включая рассмотрение возможностей лечения, анализ клинических испытаний, генетический анализ и белковый анализ
   в) Определение лечебного протокола, возможно с использованием генетической или белковой терапии
   г) Мониторинг пациента и корректировка лечения при необходимости с использованием нового беспроводного устройства для персонализированного лечения и мониторинга. Пациент использует социальные сети для описания общего впечатления от лечения.
3. Сведения, необходимые для решения проблемы:
   а) История болезни
   б) Кровь, ткани, результаты анализов и т.д.
   в) Статистические результаты вариантов лечения
   г) Данные клинических испытаний
   д) Данные генетических исследований

е) Данные исследования белков
ё) Данные социальных медиа
4. Сбор данных, их обработка и анализ результатов:
а) Начало лечения
б) Мониторинг состояния пациента и корректировка лечения при необходимости

Это оптимальный пример, в котором нет необходимости создавать новые процессы для поддержки интеграции больших данных. Пока процессы остаются относительно неизменными, базовые технологии включают приложения, которые потребуется изменить для учета характеристик больших данных, в том числе объема данных, разнообразия источников данных и скорости, необходимой для обработки данных.

Введение в процесс управления здравоохранением больших данных способно сыграть огромную роль в развитии более эффективных средств диагностики и управления в сфере здравоохранения. Тот же операционный процесс может быть применен в различных отраслях, начиная с добычи нефти и газа и заканчивая финансовыми рынками и розничной торговлей. Каковы ключевые аспекты успешного применения больших данных в операционных процессах? Вот основные шаги на этом пути:

- изучить текущий процесс;
- выявить зоны нехватки информации;
- определить релевантные источники больших данных;
- разработать процессы для естественной интеграции имеющихся данных и данных, которые могут появиться в будущем;
- отрегулировать процесс анализа и процессы принятия решений с учетом применения больших данных.

Потоки операций с большими данными

Рабочий процесс состоит из одного или нескольких рабочих потоков, имеющих отношение к общей цели процесса. Процессы разрабатываются как высокоуровневые сквозные процедуры, которые определяют порядок принятия и исполнения решений внутри компании или организации. Напротив, рабочие потоки ориентированы на задачи и часто требуют более конкретных данных, чем процессы.

Во многом рабочие потоки больших данных аналогичны рабочим потокам стандартных данных. Фактически в любом рабочем потоке на различных этапах возникает потребность в новых данных для выполнения всех необходимых задач. Рассмотрим рабочий поток из вышеприведенного примера — взятие крови на анализ. Этот простейший рабочий поток решает задачу, необходимую для выполнения общего процесса диагностики. Если данные об анализе крови не будут созданы (или будут утеряны), это окажет прямое воздействие на точность или достоверность всего процесса.

А если представить рабочий поток, зависящий от источника больших данных? Хотя для больших данных можно использовать и существующие рабочие потоки, невозможно гарантировать, что процесс или рабочий поток будет функционировать надлежащим образом при замене источника больших данных на стандартный источник, поскольку стандартные методы обработки данных не обеспечат необходимой функциональности или производительности для обработки разнообразия и сложности больших данных.

Рабочая нагрузка в контексте проблем бизнеса

Продолжим рассматривать пример с анализом крови. В стандартном рабочем потоке данных кровь типизируется, затем выполняются определенные химические анализы на основе требований, формулируемых медицинским работником. Маловероятно, что этот рабочий поток определяет, какие анализы необходимы для определения тех или иных биомаркеров или генетических мутаций. Если к этому потоку подключить источники больших данных о биомаркерах и мутациях, рабочий поток пройдет с ошибкой, так как он изначально не рассчитан на работу с большими данными и должен быть модифицирован, чтобы обеспечить их поддержку.

Чтобы эффективно использовать большие данные в рабочем потоке, рекомендуется выполнить следующие действия:

- определить источники больших данных, которые следует использовать;
- сопоставить типы больших данных с типами данных рабочих потоков;
- убедиться, что обеспечивается достаточная скорость обработки и доступа к хранилищу для поддержки рабочего потока;

- выбрать хранилище данных, которое лучше всего подходит для используемых типов данных;
- изменить существующий рабочий поток для использования больших данных или создать новый рабочий поток данных.

Если уже имеются рабочие потоки больших данных, необходимо настроить эти рабочие потоки, чтобы они не испортили анализ избыточными или лишними данными. Например, многие источники больших данных не имеют хорошо описанных определений данных и метаданных, касающихся элементов данных этих источников. В некоторых случаях эти источники данных не очищены. И конечно, прежде чем использовать источник больших данных, необходим достаточный уровень знаний о нем.

Гарантия валидности, достоверности и волатильности больших данных

Большие объемы, большое разнообразие и высокая скорость — вот основные характеристики больших данных. Эти характеристики подробно рассмотрены в главе 1. Однако другие характеристики больших данных также важны, особенно для применения больших данных к операционным процессам. Вот вторая триада характеристик, которые являются ключевыми для ввода в операционную деятельность больших данных.

- Валидность. Отличаются ли данные правильностью и точностью для использования их в конкретных целях?
- Достоверность. Значимы ли результаты для области данной проблемы?
- Волатильность. Как долго необходимо хранить эти данные?

Валидность данных

Очевидно, что от анализа больших данных ожидаются точные результаты. На начальных этапах анализа, когда объемы измеряются петабайтами, валидность, или допустимость каждого элемента данных не имеет первостепенного значения — исходный поток больших данных может быть достаточно «грязным», поскольку важнее оценить, существуют ли какие-либо отношения между элементами в составе этого источника данных, а не гарантировать допустимость всех элементов.

Однако после того, как удалось определить, что некоторые элементы исходного анализа данных важны для решения задачи, этот набор больших данных необходимо проверить, так как теперь он будет применен в рабочих условиях. Когда данные перемещаются от этапа исследования к этапу начала действий, их следует проверить. Допустимость источников больших данных и последующий анализ должны быть точными, если результаты будут использоваться для принятия решений или для других важных целей. Допустимые входные данные и последующая их правильная обработка должны привести к формированию точных результатов. В работе с большими данными важно уделять особое внимание допустимости данных. Например, в сфере здравоохранения можно использовать данные клинических испытаний, которые могут иметь отношение к симптомам болезни пациента, однако лечащий врач без предварительной проверки (валидации) таких данных не может их использовать, как если бы они были связаны непосредственно с состоянием пациента.

Между потоком данных Twitter и данными телеметрии, которые поступают с метеорологического спутника, существенная разница. Казалось бы, зачем интегрировать два таких источника данных, на первый взгляд, ничем не связанных? Представьте, что метеорологический спутник показывает, что в определенном регионе формируется ураган. Как этот ураган повлияет на людей, которые живут на пути его прохождения? В Twitter более полумиллиарда пользователей, поэтому с помощью анализа потоков Twitter можно определить, как ураган воздействует на население районов, где он прошел. Таким образом, использование Twitter в сочетании с данными с метеорологического спутника поможет исследователям понять степень точности прогнозов погоды.

Наличие только данных со спутника еще не означает, что эти данные являются достоверным представлением о погоде в определенном географическом регионе. Чтобы получить достоверное представление о погоде, следует сопоставить поток данных из социальных сетей (например, Twitter) с данными, полученными со спутника для определенной области. Если люди, живущие в этой области, публикуют свои наблюдения о погоде и эти наблюдения соответствуют данным со спутника, достоверность текущего состояния погоды установлена. Хотя допустимость и достоверность связаны, они являются независимыми индикаторами эффективности данных и всего процесса.

Волатильность данных

Как долго сохраняется потребность в хранении данных, которые отвечают требованиям валидности и достоверности? В стандартных условиях данные можно хранить десятилетиями: со временем складывается понимание того, какие данные имеют значение для той или иной деятельности, вырабатываются правила, регулирующие актуальность и доступность данных, которые соответствуют рабочим процессам. Например, компания хранит в своей бизнес-системе данные по клиентам и транзакциям только за последний год — это обеспечивает быстрое получение нужной информации по мере возникновения необходимости. Если нужны данные за более давний период, ИТ-отдел по запросу восстанавливает их с автономного носителя. С большими данными ситуация сложнее.

Если объем хранилища ограничен, необходимо оценить источники больших данных, чтобы определить, что следует собирать и как долго это следует хранить. При работе с некоторыми источниками данных бывает необходимо просто собрать данные для быстрого анализа. Например, для анализа опыта владельцев гибридных автомобилей достаточно подключиться к каналам Facebook и Twitter для сбора всех записей о гибридных автомобилях и затем сохранить эту информацию локально для дальнейшей обработки. Если места в хранилище недостаточно для размещения всех этих данных, можно обрабатывать их «на лету» (по мере их сбора) и сохранять локально только релевантные элементы. Продолжительность хранения больших данных зависит от нескольких факторов.

- Как много данных имеется в источнике?
- Необходимо ли повторно обрабатывать данные?
- Необходимо ли обрабатывать данные, собирать дополнительные данные и выполнять дополнительную обработку?
- Установлены ли правила или нормативные требования по хранению таких данных?
- Зависит ли от этих данных работа клиентов?
- Эти данные все еще имеют значение или уже не являются релевантными?

Вследствие таких факторов, как объем, разнообразие и скорость больших данных, необходимо понимать степень их волатильности. Некоторые источники всегда будут содержать данные, к другим источ-

никам это будет относиться не всегда. Чтобы определить требования и политику хранения больших данных, нужно понимать суть каждого используемого источника данных и знать, как долго они будут оставаться доступными.

Большие данные и аналитика открывают путь ко всему разнообразию новых видов информации обо всем, с чем мы сталкиваемся в повседневной жизни. Потребителю большие данные помогают решить, когда и где приобретать товары и услуги, пациенту клиники благодаря большим данным поставят более точный диагноз и подберут подходящее лечение, ну а бизнес-специалист с их помощью определит оптимальные способы создания и распространения продуктов и услуг. Но для этого необходимо, чтобы большие данные были интегрированы в операционные процессы компаний и организаций.

Глава 18

Применение больших данных в в вашей организации

В ЭТОЙ ГЛАВЕ:

Экономика больших данных
Интеграция управления данными предприятия и управления большими данными
Составление плана действий по внедрению
Начало работы

Итак, собранные и проанализированные большие данные предоставляют ценные сведения о существующих проблемах, а также открывают возможности для решения вновь возникающих проблем. Использование больших данных потенциально способно справиться с задачами, которые ранее казались невыполнимыми, однако этому сопутствуют некоторые затраты. Для применения больших данных требуется внести корректировки в операционные процессы (см. главу 17), добавить в рабочую среду новые типы данных, кроме того, понадобятся новые типы анализа, которые позволят понять значимость больших данных и их связь (или ее отсутствие) с существующими данными. Наконец, должны быть внедрены новые технологии, удовлетворяющие требованиям, связанным с большими данными.

Понимание экономики больших данных, особенно способов реализации и интеграции больших данных в вашей среде, позволит ответить на следующие вопросы:

- Как лучше всего использовать большие данные в вашей организации?

- Как выполнить гибкое, экономически эффективное внедрение данных?
- Как работать с большими данными?
- Как уменьшить отрицательные результаты деструктивного влияния технологий?

Экономика больших данных

Лучший способ понять экономику больших данных — рассмотреть различные методы использования больших данных в вашей организации. Хотя конкретные затраты могут различаться в зависимости от размера организации, ее покупательной способности, отношений с поставщиками и т.д., сами классы расходов являются сравнительно постоянными. Экономику больших данных анализируют по следующим направлениям:

- идентификация типов и источников данных;
- изменение бизнес-процессов или создание новых процессов;
- изменение технологий или создание новых технологий для больших данных;
- привлечение нового квалифицированного персонала и обучение имеющихся сотрудников;
- возможности повышения окупаемости инвестиций в большие данные.

С учетом растущей популярности больших данных соответствующую экономику лучше всего рассматривать с двух ракурсов: начало работы и поддержка стабильного функционирования. Мы охарактеризуем эти направления и оценим экономический эффект и преимущества больших данных.

Идентификация типов и источников данных

По мере развития технологий, связанных с большими данными, понадобится учитывать новые и вновь появляющиеся типы и источники данных. Одни вы сможете контролировать, другие будут определять некоторые ваши действия. Вот ряд вопросов, которые необходимо учесть:

- Какие данные потребуются для решения рассматриваемой бизнес-задачи?
- Из каких источников можно получить эти данные?
- Что можно сделать с этими данными?
- Как часто необходимо взаимодействовать с данными?
- Кому будут принадлежать эти данные и рабочие продукты?
- Как долго необходимо хранить данные?
- Можно ли доверять данным и их источнику?

Рассмотрим на примере практические аспекты экономики больших данных. Бренд-менеджер в компании, продающей потребительские товары, решает использовать большие данные, чтобы лучше оценить потребности, привычки, тенденции и лояльность клиентов. Соответственно, необходимо найти данные об эмоциональном отклике клиентов, опыте использования продукта, его удобстве, конкурентоспособных альтернативах и др. Некоторые такие данные будут доступны в традиционных формах, таких как записи системы управления взаимоотношениями с клиентами (CRM), и в существующих хранилищах данных. Однако традиционными данными наш бренд-менеджер ограничиться не может и будет стремиться рассмотреть проблему с разных ракурсов.

Анализ больших данных для возможности прогнозирования

Наш бренд-менеджер хочет не только задавать вопросы, но и иметь возможность прогнозировать изменения требований или привычек клиентов. Как правило, некоторые закономерности можно проследить в уже существующих данных, однако если этих данных для анализа недостаточно, то закономерности будут выглядеть как аномальные значения или даже как ошибка. Однако возможность прогнозирования даже самых незначительных изменений позволит бренд-менеджеру вовремя на них отреагировать, например, зная о тенденции в потребительских запросах, заранее протестировать новые услуги или новую упаковку, пока конкуренты еще не представляют, что ситуация меняется.

Поиск подходящих источников данных

Следующим шагом является определение источников данных. Необходимо установить, где получить желаемые данные, какими должны быть форма или тип данных и какой степенью качества и надежно-

сти они должны обладать. Источниками данных об эмоциональных откликах являются социальные сети, такие как Facebook, Foursquare, Yelp, Pinterest, Twitter. Выбор источников может быть определен привычками клиентов: например, ваш идеальный клиент очень активен в социальных сетях. Однако ваша компания может осуществлять продажи только для юридических лиц. Помогут ли социальные сети улучшить обслуживание ваших клиентов? Предпочтительно включить в анализ данные с сайтов для взаимодействия юридических лиц (B2B). Это огромный объем данных, и порой приходится искать ту самую иголку в стоге сена: несколько бит данных с эмоциональным откликом клиента о продукте или бренде, скрытые в необъятной шири социальных сетей. Кроме того, структура и типы этих данных различаются от сайта к сайту, что повышает сложность и может увеличить затраты. Нашему бренд-менеджеру нужно будет определить ценность источников данных и результатов их анализа для получения значимых сведений. Некоторые из этих источников могут быть изучены без особого труда и затрат, тогда как другие потребуют подробного анализа окупаемости инвестиций для определения потенциальной ценности информации, содержащейся на том или ином сайте.

Что вы можете делать с данными?

После определения источников и типов данных необходимо понять, что можно делать с данными. Можно ли их изменить? Можно ли их сохранить локально для дальнейшего использования? Есть ли ограничение на сбор данных за определенный период времени? Как часто можно брать данные из источников? Ограничивается ли скорость «скачивания» данных? Как часто меняются данные в источнике? Как долго хранятся данные в источнике? Ответы на эти вопросы позволят бренд-менеджеру понять экономический эффект от использования больших данных. Например, для анализа может потребоваться неограниченный доступ к определенному источнику, однако такой уровень доступа может оказаться достаточно дорогим. Насколько важно наличие постоянного доступа? Если этот источник использовать каждую неделю, повлияет ли это на прибыль или дальнейшую разработку продуктов? Если ответ положительный, то доступ к такому источнику стоит приобрести.

Чтобы контролировать затраты, важно понимать, как часто данные используются внутренними системами. Если необходимо анали-

зировать эмоции клиентов в разных социальных сетях в режиме реального времени, то затраты будут очень высоки. Если анализ можно выполнять менее часто или использовать меньшее число источников данных, затраты будут более управляемыми и не столь значительными (в случае если платить надо только при использовании данных). Использование данных и степень их актуальности — это ключевые факторы экономики больших данных.

Некоторые поставщики источников больших данных пожелают остаться владельцами своих данных, лицензируя их только для определенного использования без нарушения их целостности. Другие смогут предоставить данные по небольшой стоимости или бесплатно без особых требований к их использованию. Бренд-менеджер в нашем примере должен будет оценить каждый источник и убедиться в обеспечении надлежащего подхода к правам на владение данными и рабочими продуктами, полученными с использованием этих данных. Некоторые виды лицензирования данных ограничат использование только вычислениями и последующим уничтожением данных. Данные можно будет использовать в рамках аналитического процесса, однако после завершения всех вычислений их надо будет уничтожить. Другие поставщики могут разрешить использование данных, однако потребуют их «возврата» после завершения вашего анализа или вычислений, чтобы источник данных был доступен для других последующих пользователей. Следует с особой тщательностью подходить к защите информации и рабочих продуктов компании по мере интеграции больших данных в рабочую среду.

В рассматриваемом примере бренд-менеджеру, вероятнее всего, потребуется отслеживать эмоции клиентов в течение определенного периода. Изменилась ли общая картина эмоций клиентов за последний месяц? За шесть месяцев? За год? А эмоциональный отклик на предложения конкурентов за те же периоды? Если понять, как часто потребуется получать доступ к данным, можно будет спрогнозировать затраты, связанные с объемом, доступностью и актуальностью данных.

Экономику больших данных следует рассматривать с двух позиций: начало работы и поддержка стабильности. Начальные затраты можно снизить, найдя открытые данные или свободно доступные источники. Если необходимо больше ресурсов центра обработки данных, можно рассмотреть возможность использования облачных услуг, где оплачиваются только использованные ресурсы. Эта стратегия открытых

источников и облачных услуг открывает большие возможности для экспериментов. Итак, для начала следует выбрать подход, который позволит проще всего достичь поставленных бизнес-целей, а затем можно приступать к его реализации. Возможность реализации выбранного подхода к использованию больших данных позволит рассчитывать на его стабильное функционирование. Конечно, при планировании необходимо учитывать возможность роста затрат и возникновения новых проблем, однако надежный фундамент для расчетов уже будет заложен.

Изменение бизнес-процессов или создание новых процессов

После того как бренд-менеджер из нашего примера подготовит источники данных, ему нужно определить, какие процессы будут затронуты. Например, определение новых клиентов на основе источников больших данных и добавление этих клиентов (в качестве потенциальных) в существующую базу данных клиентов будет иметь незначительное воздействие. В других случаях может потребоваться создание новых процессов, чтобы понять, как источники больших данных могут использоваться для продвижения бренда или получения сведений о лояльности клиентов и возможности удержания этих клиентов. В любом случае, важно моделировать затраты, требуемые для изменения существующих или создания новых рабочих потоков. Правильное экономическое воздействие больших данных должно обеспечивать баланс затрат, связанных с этими изменениями, и потенциальной выгоды.

Воздействие потоков больших данных на технологии

Итак, бренд-менеджер определил типы и источники больших данных, а также оценил необходимые изменения бизнес-процессов. Теперь ему необходимо понять, как это повлияет на технологии. В идеальном мире для обработки рабочих потоков больших данных можно использовать много существующих технологий и приложений. Однако, скорее всего, для получения максимальной экономической ценности от инвестиций в большие данные потребуется внедрение новых технологий.

Современные разнообразные инструменты для работы с большими данными уже описывались в главах 8, 9 и 10. Если бренд-менеджеру

необходимо собрать данные с нескольких разных сайтов социальных сетей, каждый из которых характеризуется особым типом данных, ему необходимо совместно с ИТ-подразделением выбрать технологию, которая лучше всего будет соответствовать бизнес-требованиям и требованиям к затратам. Например, если необходим модуль MapReduce, можно ли использовать Hadoop, или же коммерческая реализация лучше будет соответствовать решаемым задачам? Можно ли использовать существующие хранилища данных или необходимо внедрить Apache Hive? Может ли реляционная СУБД хранить большие данные, или же требуется использовать другое хранилище данных? Оптимальным вариантом для освоения популярных тенденций будет реализация продуктов, сочетающих в себе элементы Hadoop и Hive; однако такой подход не позволит решить все разнообразие задач, которые могут возникать в различных ситуациях использования больших данных.

При внедрении больших данных в рабочую среду новые технологии обязательно понадобятся. Дело в том, что существующие технологии либо неустойчивы, либо узкоспециализированы и потому не справятся с нагрузками, связанными с приложениями для больших данных. Это означает, что для перехода на следующий уровень использования больших данных придется понести соответствующие расходы. Именно поэтому так важен экономический анализ.

Поиск квалифицированного персонала для поддержки проектов больших данных

В нашем примере бренд-менеджеру понадобится создать новые процессы и технологии. В свою очередь, каждое из этих требований приведет к необходимости развить у персонала новые навыки или актуализировать имеющиеся в первую очередь, в отделах ИТ и бизнес-аналитики.

Бизнес-аналитики должны стать настоящими специалистами по работе с данными. Это можно осуществить на начальных стадиях посредством привлечения консультантов, однако по мере освоения темы следует озаботиться и штатным персоналом. Один специалист по работе с данными вряд ли сможет сделать всю работу, если речь не идет о предприятии малого и среднего бизнеса. Самое эффективное решение — это сформировать группу специалистов, которые будут опреде-

лять источники больших данных, осуществлять анализ и интеграцию с бизнес-процессами.

Сотрудники ИТ-подразделений должны в полной мере овладеть знаниями о технологиях больших данных. Также возможно дополнительное обучение и привлечение сторонних консультантов. Со временем необходимо будет принимать на работу новых квалифицированных специалистов. Можно и нужно прибегать к помощи консультантов, чтобы организация уверенно начала внедрять свои инициативы, связанные с большими данными. Кроме того, следует учитывать, что в ряде учебных заведений есть курсы соответствующей тематики, которые позволят быстро восполнить пробелы в знаниях. В долгосрочной перспективе поставщики решений должны создать более удобные в использовании инструменты для работы с большими данными, которые позволят работать с задачами любой сложности.

Расчет окупаемости инвестиций в большие данные

В нашем примере бренд-менеджеру необходимо проанализировать рентабельность инвестиций в большие данные, чтобы лучше понять и спрогнозировать новые способы расширения клиентской базы. Все обсуждаемые затраты должны быть сопоставлены с возможными объемами инвестиций. Через какое время окупятся инвестиции в проект, связанный с большими данными? Как и во многих других случаях, «все зависит от обстоятельств». Если бренд-менеджер создает решение, ориентированное исключительно на его сферу деятельности, окупаемость инвестиций, скорее всего, окажется меньше, чем при использовании более общего подхода, применяемого в различных областях бизнеса для работы с большими данными. Если же новые возможности будут доступны другим бренд-менеджерам, службе поддержки клиентов или специалистам по продажам, то окупаемость инвестиций может стать весьма значимым фактором. Важнее всего при создании модели окупаемости инвестиций тщательно изучить все экономические аспекты, чтобы иметь полное представление о проблемах и возможность лучше прогнозировать результаты.

Итак, разобравшись, что именно предстоит сделать, чтобы начать использовать большие данные в вашей организации, вы сможете лучше понять, как этого достичь.

Управление данными предприятия и большие данные

Управление данными предприятия (EDM) — это важный процесс для понимания и контроля экономики данных в компании или организации. Управление данными предприятия не является обязательным требованием применительно к большим данным, однако правильное использование этой технологии позволит добиться лучшей интеграции, контроля и более эффективного использования больших данных.

Что такое управление данными предприятия

Управление данными предприятия — это комплексный подход к определению, управлению, обеспечению безопасности и поддержке качества всех данных, используемых в бизнес-процессах организации. Управление данными предприятия включает формирование политики использования ключевых типов и источников данных, определяет их владельцев, а также позволяет создать стратегию и технологические основы управления жизненным циклом данных. Основной целью процессов управления данными предприятия является поддержание «достоверности» информации. На рис. 18.1 показаны ключевые компоненты управления данными предприятия.

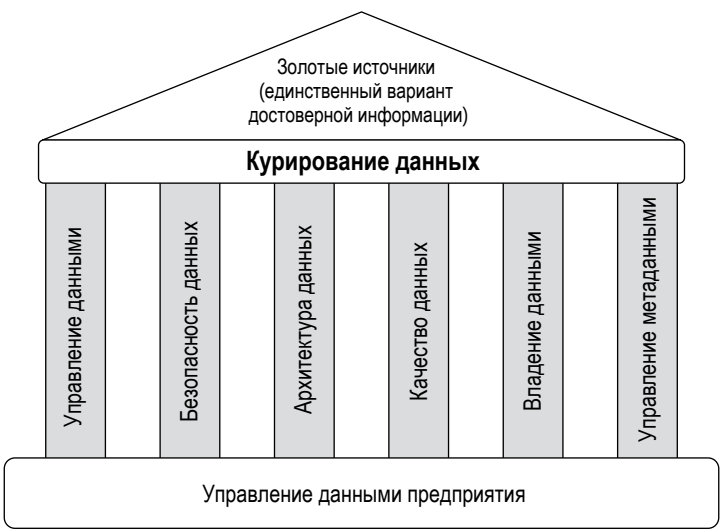


Рис. 18.1. Управление данными предприятия

Версия управления данными предприятия в вашей организации может несколько отличаться, но в ее основе будут те же самые принципы. Данные должны быть смоделированы, управляемы и защищены, чтобы вы могли доверять результатам обработки в рамках процесса принятия решений. Следует также отметить, что отсутствие управления данными предприятия на уровне организации не является преградой для внедрения больших данных — оно может использоваться на уровне отдела или даже на уровне проекта, хотя их эффективность в этом случае будет ниже. Учитывая это, необходимо определить, повлияет ли внедрение систем больших данных на принципы управления данными предприятия. Для того чтобы продумать использование больших данных в сочетании с существующей системой учета данных, необходимо хорошо понимать принципы управления и курирования всей средой управления данными, ведь это повлияет на стратегические решения. Поэтому крайне важно, чтобы элементы данных, которые подвергаются сравнению или анализу, были точными, значимыми и безопасными. Управление и соответствие нормативным требованиям подробно освещаются в главе 19.

Составление плана мероприятий по внедрению больших данных

Планы по внедрению больших данных будут различаться в зависимости от ваших бизнес-целей, зрелости среды управления данными и степени риска, которая приемлема для организации. Итак, для начала при составлении плана действий требуется учесть ключевые факторы, в числе которых:

- роль фактора времени в вашем бизнесе;
- прогнозируемые мощности;
- предпочтительные методы разработки программного обеспечения;
- доступный бюджет и наличие необходимых навыков;
- склонность к риску.

Теперь рассмотрим эти факторы применительно к некоторым образцам планов действий, которыми можно руководствоваться при разработке собственного решения.

Понимание роли фактора времени в вашем бизнесе

Создается впечатление, что компании, стремящиеся к лидерству на рынке, начинают внедрять самые новые и передовые технологии сразу же, как только они появятся. Иногда организация может продемонстрировать, что доступ к важным источникам больших данных способен привести к формированию новых стратегий; в этих случаях разумно сразу сформировать стратегию и составить план. Ошибочно предполагать, что внедрение и использование больших данных является готовым «пакетным» решением — использование больших данных влияет на различные аспекты стратегии управления данными в компании, поэтому в любом плане действий должно быть четко определено время, необходимое для разработки решений для работы с большими данными. Кроме того, задачи проектирования никогда не следует рассматривать как второстепенные или вовсе исключать из плана. Это значительно снизит ценность любой инициативы по использованию больших данных. Например, может понадобиться задействовать новые источники данных, приобрести новое оборудование и программное обеспечение или оформить подписку на облачные сервисы, которые предоставляют уровень обслуживания и гарантии, соответствующие требованиям вашей компании.

Планирование необходимой производительности

Определив, что введение больших данных в вашу среду является необходимостью, нужно ответить на два вопроса: «Сколько данных понадобится?» и «Как быстро необходимо их анализировать?». Это даст возможность определить длительность этапов планирования, реализации и тестирования в вашем плане действий. Например, если необходимо собрать 200 Гб данных и сохранить их до завершения анализа, внедрение этого решения больших данных может занять 2–4 недели. С другой стороны, если вы рассчитываете получить 2 петабайта исходных данных, далее получать еще по 0,5 петабайта в месяц и хранить все данные в течение многих лет, на внедрение в плане следует отвести несколько месяцев. И в любом случае, если предполагается собирать и анализировать данные в режиме, близком к реальному времени, продолжительность внедрения в плане необходимо увеличить на 50%.

Выбор эффективной методологии разработки программных средств

В большинстве компаний и организаций есть ИТ-подразделения, которые имеют опыт работы в соответствии с рекомендованными процедурами внедрения и методами разработки. Некоторые из этих методов разработки хорошо подходят для внедрения больших данных, другие, к сожалению, неприменимы.

Проекты больших данных лучше всего подходят для гибкого и итерационного процесса разработки. Итерационные методы задействуют короткие временные циклы и характеризуются быстрым получением результатов, постоянным участием пользователей в реализации бизнес-решения, поэтому неудивительно, что именно итерационный процесс является самым эффективным методом разработки для внедрения больших данных.

Доступный бюджет и наличие необходимых навыков

Всегда трудно спрогнозировать бюджетные требования для проекта нового типа, в особенности для такого, как внедрение систем больших данных. Нужно четко понять ожидаемые затраты и последующие преимущества от внедрения больших данных и лишь после этого формировать соответствующий бюджет для реализации этой инициативы. Итерационный подход лучше всего позволяет определить рациональный путь формирования бюджета проекта. Поэтому размер бюджета может быть определен сразу же, а финансирование может осуществляться при достижении результатов того или иного этапа проекта.

Еще одна задача — подбор персонала с подходящими для проекта навыками. Как правило, самые ценные специалисты работают сразу над несколькими инициативами. Увеличение численности сотрудников часто позволяет решить проблемы, связанные с ресурсами, однако в развивающейся области деятельности, характеризующейся быстрым ростом, это сделать сложнее. Иногда навыки, необходимые для работы с большими данными, возникают по мере развития рынка. Со временем появится больше возможностей для обучения и больше квалифицированных специалистов. Рекомендуем ознако-

миться с передовым опытом и приобрести базовые навыки в сфере работы с данными, связанные с проектированием и планированием, внедрением Hadoop и NoSQL и использованием параллельных и кластерных вычислений. Также полезно будет протестировать продукты и получить навыки, выполнив пилотное или экспериментальное внедрение в облачной среде. Обучение на базе собственной компании позволит значительно быстрее добиться успеха в применении больших данных.

Определение допустимой степени риска

Каждая организация характеризуется собственной корпоративной культурой, которая определяет, какая степень риска допустима для этой организации. Если вы работаете на весьма конкурентном рынке, то вынуждены рисковать, внедряя инновационные для рынка решения, в отличие от компании, чьи продукты востребованы клиентами при малом количестве конкурентов. Но даже организации, имеющие большой опыт работы в конкурентной среде, должны осторожно и взвешенно оценивать потенциальные риски. Итак, прежде чем приступать к реализации проекта больших данных, вам необходимо понять динамику своей организации. Например, возможно, потребуется сначала реализовать ограниченный пилотный проект, который продемонстрирует руководству открывающиеся возможности, или представить прогнозный план, основанный на прогнозной аналитике больших данных, который позволит вывести взаимодействие с клиентами на качественно иной уровень.

Если ваша организация не склонна к риску, лучше наблюдать за эволюцией больших данных и предлагать руководству проекты по мере появления на рынке готовых решений. Изучение успешного опыта других организаций позволит внедрить новые технологии без серьезных потрясений для организации.

Даже организации, характеризующиеся большой склонностью к риску, должны со всей осторожностью подходить к применению больших данных. Очень легко красочно преподать преимущества или переоценить ожидания, связанные с большими данными. Необходимо помнить о том, что развитие и внедрение новой технологии или решения может сопровождаться неудачами. Используйте гибкие технологии, которые позволят найти причины быстрого успеха или же неудачи, — это позволит правильно оценить ожидания.

Разработка плана мероприятий по работе с большими данными

Ниже представлены две схемы, которые позволят вам оценить возможности вашей организации и сформировать соответствующий им план мероприятий по внедрению больших данных.

Это примеры успешного внедрения новых технологий в инструментарий компании на ранних этапах их существования. Эти схемы могут быть взяты за основу.

Пример плана действий на рис. 18.2 можно легко адаптировать для вашей первой инициативы, связанной с большими данными. Здесь предполагается практическое или полное отсутствие процессов или технологий, связанных с высокими требованиям к объему, разнообразию и скорости больших данных. Очевидно желание продвигать проект вперед уверенно, но без лишней спешки. Спешка в реализации проекта — верный способ поставить под сомнение возможности больших данных в качестве основы для последующего принятия решений в вашей организации.

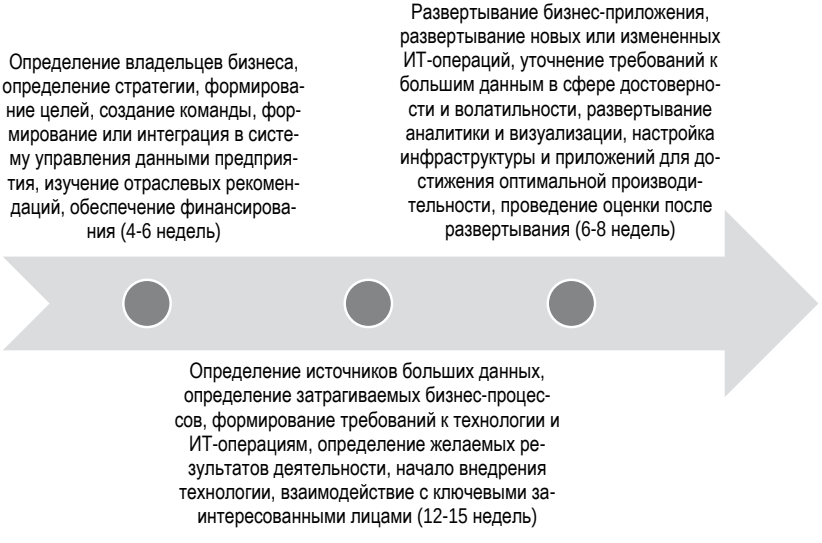


Рис. 18.2. Основные этапы первого внедрения больших данных

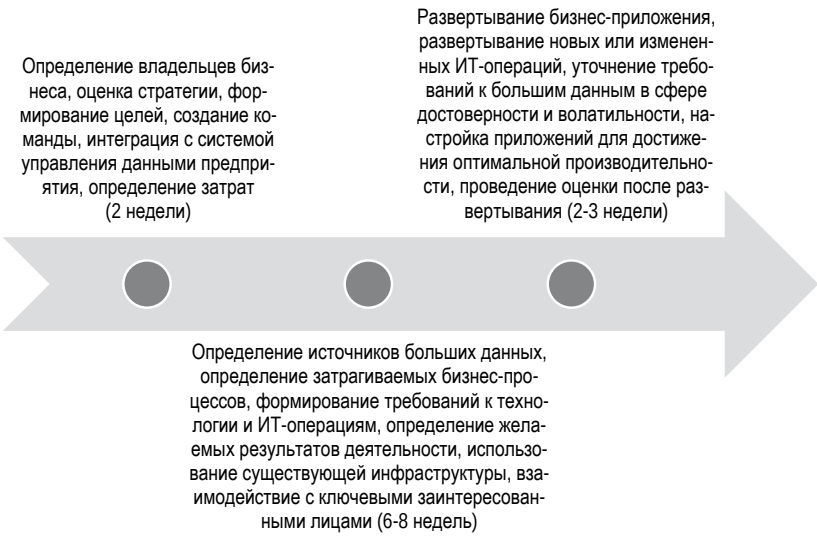


Рис 18.3. Основные этапы внедрения больших данных опытными специалистами

Если в вашей организации есть опыт работы с приложениями для бизнес-аналитики, достаточно развита практика управления данными, сформирована высокопроизводительная инфраструктура, обеспечивающая надлежащие операции, то тогда задача внедрения больших данных значительно упрощается. Это не гарантирует успех или снижение риска. Наличие этих возможностей сигнализирует о зрелости вашей компании и ее стремлении задействовать новую информацию, технологии и навыки. Примерный план действий по внедрению для более опытных специалистов показан на рис. 18.3.

Большинство действий схожи между собой и могут выполняться параллельно — если опыт организации это позволяет. Результата может добиться каждый. Если вы тщательно провели послепроектную оценку и внесли корректировки в план на основе опыта первого или второго проекта, в дальнейшем это позволит уменьшить время, необходимое на проектирование, развитие, проверку, реализацию и развертывание инициатив, связанных с большими данными.

Начинать всегда проще, если некоторые из участвующих в работе специалистов уже выполняли подобные проекты. Однако в таком новом и революционном деле, как большие данные, вы вряд ли найдете много людей, которые могут похвастаться предыдущим опытом. Вот несколько советов.

- Не бойтесь обращаться за помощью. Пригласите одного-двух консультантов, готовых к тесному сотрудничеству. Убедитесь, что они действительно хорошо знакомы с тематикой и способны обучить ваших сотрудников.
- Обучайтесь. Курсы, книги и материалы в Интернете, специализированные конференции — прекрасная возможность освоиться в новой среде. Это поможет в принятии всех последующих решений.
- Экспериментируйте. Учитывайте в планах возможность неудач и сбоев. Практически любая современная компания, основой развития которой являются технологии, терпела неудачу в начале своего пути. Известно, что на своих ошибках учатся — но не стоит пренебрегать и чужим опытом.
- Сформируйте адекватные ожидания. Как говорится, залог счастья — не ждать от жизни слишком многого. В бизнесе правильно сформулированные ожидания помогут отличить успех от неудачи. Успешный проект можно рассматривать как неудачу, если ожидаемые преимущества были преувеличены или внедрение проекта заняло на 50% больше времени. Большие данные несут огромный потенциал для вашего бизнеса, однако вы сами должны аккуратно определить их ценность, затраты и время реализации.
- Мыслите глобально. Попытайтесь оценить любую инициативу, касающуюся больших данных, со всех сторон. Если проект развивается по графику и в соответствии с бюджетом, однако пользователи не обучены работе с ним, проект может завершиться провалом. Опытные руководители проектов прекрасно понимают, что успех на любом уровне связан с людьми, процессом и технологиями.

Безопасность и управление средами больших данных

В ЭТОЙ ГЛАВЕ:

Требования к безопасности больших данных
Управление большими данными
Подходы к созданию экосистемы больших данных
Управление анализом больших данных
Разработка защищенной среды больших данных
Обеспечение безопасности среды больших данных

Обычно исходят из того, что данные, используемые при анализе и планировании, имеют высокое качество и надежно защищены. Как правило, эти данные уже прошли обработку с помощью процессов очистки и профилирования и, следовательно, могут считаться надежными. Использование больших данных связано с целым рядом новых проблем и препятствий, усложняющих управление данными и обеспечение их безопасности. Многие специалисты и компании, работающие с большими данными, полагают, что безопасность и управление данными не требуют пристального внимания. Следовательно, нет необходимости обдумывать и планировать связанные с этим мероприятия. Тем не менее, когда источники больших данных становятся активными, ситуация кардинально меняется. В этой главе рассматриваются аспекты, которые необходимо принимать во внимание при использовании источников больших данных для анализа и планирования.

Безопасность в контексте больших данных

Несмотря на то что компании очень обеспокоены общими вопросами обеспечения безопасности и управления своими данными, они не готовы к сложностям, связанным с управлением большими данными.



Управление информацией — это способность создать информационный ресурс, которому могли бы доверять сотрудники, партнеры и клиенты компании, а также государственные учреждения.

Зачастую анализ больших данных связан с использованием значительного числа источников, данные из которых не могут считаться проверенными. Кроме того, ваша организация должна быть осведомлена о правилах защиты информации и управления данными, применяемых к различным источникам больших данных. Может возникнуть необходимость оценивать важность новых данных, которые ваша организация получает в больших объемах из многочисленных разнородных источников, неструктурированных и слабоструктурированных. Что, если к вам поступят данные о состоянии здоровья, защищаемые законом о преемственности и подотчетности медицинского страхования (Health Insurance Accountability and Portability Act, HIPAA), или персональные данные, такие как имена и адреса? Если система управления данными не обеспечивает их надлежащую защиту, то ваша компания нарушит закон, работая с такими данными. Некоторая часть этих данных не потребуется, и компания должна позаботиться о надлежащем их удалении. Остальным данным необходимо обеспечить защиту и управление. Таким образом, какую бы информацию вы ни использовали, необходима четко определенная стратегия безопасности.

Безопасность должна всегда находиться в фокусе вашего внимания, поскольку обстановка постоянно меняется. Помимо стратегии безопасности необходима и согласованная с ней стратегия управления данными. Безопасность и управление данными в совокупности должны гарантировать ответственность всех сторон, участвующих во внедрении системы управления информацией. Ответственность за информационную безопасность так или иначе несет вся организация в целом. Даже внедрение всех новейших средств безопасности не устранит имеющиеся риски, если ваши конечные пользователи не будут понимать своей роли в защите данных, с которыми работают.

Оценка рисков для бизнеса

Большие данные становятся критически важными для руководителей бизнеса, стремящихся выработать направления развития новых продуктов компании, понять требования клиентов и оценить состояние

окружающей деловой среды. Данные из множества непроверенных источников, используемые компанией, становятся источниками рисков и непредсказуемых последствий. Таким образом, необходимо одновременно следить за целым рядом факторов. Когда дело касается управления большими данными, обеспечение безопасности — это реагирование на постоянные изменения. Безусловно, обучение играет ключевую роль в том, чтобы каждый сотрудник организации понимал свои функции в обеспечении безопасности и свою ответственность.

Риски, заключенные в использовании больших данных

Несмотря на то что безопасность и управление данными являются приоритетными вопросами корпоративного масштаба, большие данные обладают определенной спецификой, которую вам необходимо иметь в виду.

Например, если вы собираете неструктурированные данные из таких источников, как социальные сети, необходимо проверять контент с этих сайтов на наличие вирусов и фальшивых ссылок. Если вы используете такие данные в своей аналитической системе, это может подвергнуть рискам вашу компанию. Кроме того, следует учитывать природу первоначального источника данных. Источник неструктурированных данных об определенном типе клиентов может предоставлять полезную информацию, но при этом содержать «посторонний шум». Поэтому вы должны принимать во внимание характер источника, из которого берете данные. Проверены ли данные? Надежен ли источник? Хорошо ли он защищен от постороннего вмешательства? Например, сайты популярных социальных сетей внимательно отслеживают признаки вредоносного поведения и удаляют страницы злоумышленников до того, как они успевают нанести вред. Для этого требуется серьезный анализ больших данных, который способен провести не каждый сайт. Возможна ситуация, когда ваша организация обнаруживает полезный сайт и использует его контент для корпоративной платформы анализа больших данных, а затем вдруг выясняется, что этот сайт был ранее взломан хакерами. Последствия могут быть самыми серьезными. Не все угрозы безопасности являются преднамеренными. Безусловно, вы не хотите пользоваться источником больших данных, который содержит в себе конфиденциальную информацию, позволяющую идентифицировать ваших клиентов. Это может нанести ущерб клиентам и репутации вашей компании.

Способы защиты данных

Некоторые эксперты полагают, что способ защиты данных зависит от их типа и поэтому шифрование данных в облачной среде иногда является излишним. Вы можете шифровать любые данные когда вам угодно. Например, при записи на ваш жесткий диск, при передаче поставщику облачных услуг и при сохранении в его базе данных. Данные можно шифровать на любом уровне.

Шифрование на всех уровнях снижает уязвимость перед угрозами, но уменьшает производительность работы. Например, многие эксперты рекомендуют пользователям самостоятельно управлять ключами, а не делегировать эту задачу поставщику облачных услуг. Однако управление большим количеством ключей может оказаться очень обременительным. Кроме этого, такой подход к использованию шифрования порождает и другие проблемы. Например, если вы шифруете содержимое базы данных, то вам придется контролировать данные при их передаче (шифрование на канальном уровне), а также при сохранении их в базе данных. Эта процедура может оказаться затратной и сложной. И даже если вы полагаете, что зашифровали все возможные данные и тем самым обеспечили их полную защиту, это не всегда соответствует действительности.



Один из самых серьезных недостатков стратегий использования криптозащиты в том, что данные уязвимы как до шифрования, так и после него. Например, в 2008 году произошла крупная утечка данных из супермаркетов Hannaford. Хакеры в течение нескольких месяцев похищали данные о платежах банковскими картами, осуществлявшихся в точках продаж. Утечка данных происходила до их шифрования.

Использовать большое количество ключей непрактично, а управлять их хранением, архивированием и доступом к ним сложно. Чтобы решить эту проблему, следует генерировать и использовать ключи шифрования по необходимости. Это упрощает работу с ключами и повышает защищенность данных.

Ниже перечислено еще несколько методов защиты данных.

- Обезличивание данных: удаление всех данных, которые позволяют идентифицировать человека (имя, номер социального страхования или кредитной карты). Этот метод способен обеспечить защиту от неправомерной идентификации личности (а следовательно,

конфиденциальность). Но несмотря на это, необходимо очень осторожно относиться к персональной информации, которую вы используете. Если не вся персональная информация будет удалена, то хакеры смогут определить, кому принадлежат данные.

- Использование токенизации: защита конфиденциальных данных путем замены их случайными значениями аналогичного размера и типа (токенами) или ссылками, которые бесполезны для злоумышленника, получившего несанкционированный доступ к этим данным. Это снижает вероятность использования данных в случае их кражи. Токенизация может использоваться для защиты информации о кредитных картах, паролей, персональных данных и т.д. Некоторые эксперты считают, что токенизация надежнее шифрования.
- Облачное управление базами данных: метод, при котором средства управления доступом встроены в базу данных и целиком защищают ее содержимое — это устраняет необходимость шифрования отдельных элементов данных.

Проблема управления данными

Управление данными является важной задачей компании независимо от того, какие источники данных используются и как управляются. Обычно компании, которые используют традиционные хранилища данных и системы управления реляционными БД, создают четкие и понятные правила защиты данных. Например, в медицине критически важной задачей является обеспечение конфиденциальности данных о пациентах. Данные о пациентах можно хранить и анализировать при условии, что их имена, номера социального страхования и другая персональная информация скрыты. Вы должны позаботиться о том, чтобы исключить возможность несанкционированного доступа к частной или закрытой информации. Что происходит, когда в вашу среду загружаются большие данные из различных источников? Некоторые из этих источников принадлежат сторонним коммерческим поставщикам, которые тщательно фильтруют данные и удаляют из них конфиденциальную информацию.

Тем не менее, весьма вероятно, что ряд источников больших данных окажутся небезопасными, незащищенными и будут содержать значительный объем персональных данных. Начальная обработка,

скорее всего, выявит большое количество данных, не представляющих интереса для вашей организации. Соответственно, вы вряд ли станете затрачивать ресурсы на защиту таких данных и управление ими, поскольку не планируете хранить их. Тем не менее, если конфиденциальные персональные данные передаются по корпоративной сети, то ваша компания рискует непреднамеренно нарушить закон. Начальный анализ данных с неизвестным содержанием безопаснее всего проводить во внутренней закрытой сегментированной среде или в облаке.

Если вы решите, что часть этих данных следует проанализировать более глубоко и включить результаты анализа в корпоративный бизнес-процесс, то важно обеспечить скрупулезное выполнение требований, предъявляемых к управлению данными.

При интеграции непроверенных источников данных в среду организации следует принимать во внимание следующие аспекты.

- Заранее определите пользователей, имеющих право доступа к новым источникам данных сразу после их включения в информационную среду, а также к результатам изучения и анализа этих данных.
- Выясните, каким образом эти данные будут отделены от данных других компаний.
- Выясните, какую ответственность накладывает на вас использование этих данных. Если у данных есть владелец, то убедитесь в том, что вы соблюдаете обязательства, налагаемые договорами и условиями использования этих данных. Условия использования некоторых данных определяются договором с их поставщиком.
- Выясните физическое расположение ваших данных. Возможно, вы используете данные о существующих и потенциальных корпоративных клиентах, которые находятся в странах со строгими правилами защиты конфиденциальности. Для того чтобы избежать нарушения закона, необходимо иметь соответствующие сведения об этих источниках.
- Выясните, что нужно сделать с данными при изменении их физического расположения. Будете ли вы хранить какую-либо часть этих данных у поставщика облачных услуг? Какие обязательства о месте хранения данных и степени их защиты возьмет на себя поставщик облачных услуг?

Даже если вы создали процесс защиты и управления традиционными источниками данных, вы не можете рассчитывать на то, что сотрудники и партнеры автоматически распространят эти правила на

новые источники данных. Вам необходимо учесть два ключевых параметра: видимость данных и надежность работающих с ними пользователей.

- Видимость данных: может оказаться, что ваши бизнес-аналитики и партнеры с удовольствием работают с новыми источниками данных, однако вы не осведомлены о том, как эти данные используются и контролируются. Другими словами, вы не «видите», что происходит внутри ресурсов, которые работают вне вашего контроля. Эта проблема стоит особенно остро в случае, если необходимо, чтобы ваш поставщик следовал правилам регулирования или действующей нормативно-правовой базе, а также если вы поручили управление данными поставщику облачных услуг, поскольку управление облачным хранилищем обходится очень недорого.
- Ненадежные сотрудники: даже если ваша компания тщательно проверяет репутацию всех своих работников, вы в любом случае вынуждены полагаться на полное отсутствие злоумышленников среди сотрудников, работающих вне ИТ-отдела, а также на то, что поставщик облачных услуг столь же скрупулезно проверяет своих сотрудников. Это очень серьезная проблема, поскольку около половины нарушений безопасности происходит по инициативе сотрудников организации либо при их соучастии. Если ваша компания планирует широко использовать новые источники данных, то вам необходимо позаботиться о противодействии как внешним, так и внутренним угрозам.



Ваша задача — убедиться, что новые источники больших данных не создают уязвимостей из-за непредвиденных угроз и управленческих рисков. Также в круг ваших обязанностей входит обеспечение эффективности процессов защиты, управления и обучения в масштабе всей корпоративной информационной среды.

Как и в любом процессе разработки и внедрения новой технологии, для этого необходимо сформировать процесс оценки способности всех подразделений вашей организации соблюдать требования безопасности и регуляторные требования. Если в вашей организации уже существуют процессы защиты, обеспечения конфиденциальности и управления данными в имеющихся базах и хранилищах данных, то для внедрения больших данных эти процессы необходимо соответствующим образом расширить.

Например, осведомлен ли директор по безопасности о новых источниках данных, используемых в различных подразделениях компании? Су-

ществуют ли четкие правила использования сторонних источников данных? Если ваша компания интегрирует в свою информационную среду большие данные, которые принадлежат другим организациям, то она рискует нарушить соответствующие авторские права. Если вы создаете среду с большим количеством новых источников больших данных, то позаботились ли вы о сокрытии всей необходимой частной информации? Соблюдаете ли вы при этом правила защиты конфиденциальности данных, принятые в других странах, где работает ваша компания?

Аудит процесса работы с большими данными

Ваша конечная цель — продемонстрировать внутренним и внешним аудиторам, что ваша организация соблюдает правила ведения бизнеса. Для этого вам будет необходимо предъявить файлы регистрации, журналы учета или другие документы, подтверждающие защищенность и достоверность используемых вами данных, а также объяснить происхождение этих данных. Сможете ли вы минимизировать риски вашей компании за счет проверки результатов? Возможно, вам нужно будет подтвердить, что вы заархивировали данные, которые используются вашей компанией при принятии и исполнении решений. При использовании традиционных баз и хранилищ данных это, скорее всего, не составит труда, однако обычные процедуры не охватывают источники неструктурированных больших данных.

Несмотря на то что внешние аудиторы могут не сверять содержимое хранилищ данных с информацией из внешних источников больших данных, ваша внутрикорпоративная процедура должна требовать тщательной синхронизации этих источников друг с другом. Например, при наличии набора достоверных основных данных в хранилище данных метаданные в источниках больших данных могут быть недокументированы. Соответственно, управление внешними источниками данных должно обеспечивать единую систематизацию определений метаданных для всех источников. Успех бизнеса в значительной степени зависит от того, насколько хорошо организован этот процесс.

Выявление ключевых пользователей больших данных

Одна из особенностей больших данных заключается в том, что они, как правило, используются определенными бизнес-подразделениями компании. Например, отдел маркетинга хочет использовать большие объемы данных, генерируемых сайтами социальных сетей, таких как

Facebook, Twitter и др., а отдел логистики — управлять цепочкой поставки с помощью данных RFID-идентификации. Отдел кадров заинтересован в том, чтобы отслеживать информацию, публикуемую сотрудниками на сайтах социальных сетей, для профилактики нарушений внутренних и внешних правил. Отделу страховых возмещений необходимо отслеживать изменения правил регистрации обращений пациентов в карточках медицинского страхования, чтобы обеспечивать надлежащую конфиденциальность персональных данных. Поскольку все перечисленные отделы могут работать в рамках одной компании, крайне важно, чтобы все сотрудники одинаково понимали правила, которые им необходимо соблюдать. При этом инфраструктура организации должна быть надежна, что также ведет к снижению рисков.

Формирование соответствующей организационной структуры

Как правило, компании начинают изучать возможность использования больших данных с выяснения степени их влияния на бизнес-стратегию. Когда становится понятно, что большие данные будут играть стратегическую роль в корпоративной информационной среде, возникает необходимость сформировать такую оргструктуру, которая обеспечит надлежащую поддержку и защиту организации.

Прежде чем определять правила работы с информацией, вам нужно понять, с какими данными вы будете иметь дело. Например, предполагаете ли вы работать с транзакционными системами, данными социальных сетей или автоматически генерируемыми данными? Будете ли вы проводить комбинированный анализ информации, получаемой из различных источников? Если вы не планируете ограничить знакомство с большими данными одиночным экспериментом, то следует скорректировать стратегию управления таким образом, чтобы обеспечить безопасность работы с новыми данными.

Подготовка к контролю и управлению рисками

Какую бы стратегию управления информацией вы ни избрали, необходимо обеспечить достаточный уровень контроля информационной среды. Это типовая практика, которая применима и в случае больших данных. Тем не менее, после добавления источников больших

данных в информационную среду может возникнуть необходимость изменения процедуры контроля данных. Например, вам может потребоваться отдельный специалист для мониторинга социальных сетей, поскольку получаемая из них информация отличается по происхождению и структуре от традиционных реляционных данных. Чтобы специалист по контролю определенного типа данных мог работать со всеми отделами компании, для которых важен анализ этих данных, необходимо четко определить роль такого специалиста. Например, он должен понимать правила хранения данных компании либо взаимодействовать с сотрудниками соответствующей компетенции, а также знать требования к скрытию персональных данных независимо от их источника.

Определение правил управления и требований к качеству данных

Работа организации с большими данными представляет собой не однократное мероприятие, а долговременный цикл. Непоследовательность в применении правил и процессов создает серьезные риски для бизнеса, даже если сами правила и процессы хорошо согласованы между собой. Кроме того, необходимо обеспечивать достаточный уровень качества данных с точки зрения управления. В политику защиты организации необходимо включить следующие основные элементы.

- Определите наилучшие методы работы, внедренные вашими коллегами. Свод процедур и правил обеспечения безопасности следует сформулировать так, чтобы у всех сотрудников организации было одинаковое понимание того, что от них требуется.
- Сравните ваши правила с нормативно-правовой базой и требованиями регуляторов в сфере деятельности вашей организации. При необходимости скорректируйте политику соответствующим образом.
- Действуют ли в вашей организации правила, определяющие длительность хранения информации? Применяются ли они к данным, получаемым из внешних источников (например, из дискуссионных групп для клиентов и с сайтов социальных сетей)?
- Какова важность источников данных, интегрируемых в компанию для ее бизнеса? Есть ли у вас стандарты качества, требующие, чтобы для принятия решений использовались только прошедшие

проверку достоверности и надлежащим образом задокументированные данные? Компания может легко поддаться соблазну воспользоваться большими данными для того, чтобы получить недостижимые ранее аналитические результаты. Тем не менее, если на основе такого анализа будут сделаны неверные заключения, это создаст угрозу для вашего бизнеса. Даже информация, собираемая с датчиков, может подвергаться сторонним воздействиям, что способно повлечь неправильные выводы.

Разработка управляемой и защищенной среды больших данных

Продуманный подход к вопросам безопасности способен сократить многие риски ее нарушения. Ваша задача — разработать защищенную среду больших данных. Для начала оцените текущее положение дел. Это поможет вам сформировать собственную стратегию обеспечения безопасности данных. Обратите внимание на следующие вопросы:

- Как вы оцениваете стратегию защиты своих обычных данных?
- Как контролируются права доступа к данным в приложениях, базах и хранилищах данных, расположенных внутри вашей компании и вне ее? Кто обладает правом доступа к имеющимся информационным ресурсам и новым источникам больших данных? Каким образом осуществляется контроль за тем, чтобы доступ к приложениям и информации имели только надлежащие пользователи?
- Обеспечена ли возможность обнаружения и устранения уязвимостей данных и рисков?
- Существует ли возможность отслеживать риски безопасности во времени и беспрепятственно делиться поступающей информацией с заинтересованными сторонами?
- Обеспечена ли непрерывная защита всей корпоративной инфраструктуры от внешних угроз? Если это не так, то для безопасности ваших данных существует серьезный риск.
- Используя шифрование, управляете ли вы ключами самостоятельно или получаете их у надежного, доверенного поставщика? Пользуетесь ли вы стандартными алгоритмами? Применили ли вы уже этот стандарт к новым источникам данных, критически важным для вашего бизнеса?

- Есть ли у вас возможность следить за количественными и качественными показателями рисков безопасности в реальном времени?
- Способны ли вы последовательно применять правила безопасности и управления ко всем источникам данных, в том числе к расположенным в облачной среде?
- Обеспечена ли защита всех данных независимо от места их хранения?
- Удовлетворяются ли требования к аудиту и отчетности об использовании данных независимо от места их хранения?
- Соблюдаются ли требования отраслевых стандартов?
- Каково содержание планов послеаварийного восстановления? Как обеспечивается непрерывность обработки критически важных источников данных?

Часть VI

Решения на основе больших данных

В ЭТОЙ ЧАСТИ:

- Обзор моделей использования больших данных
- Применение больших данных для обнаружения мошенничества
- Использование отзывов клиентов в социальных сетях при принятии корпоративных решений
- Совершенствование медицинской диагностики с помощью больших данных



Используйте большие данные в качестве инструмента бизнес-планирования

Этап 1: осуществите планирование на основе данных

Этап 2: выполните анализ

Этап 3: проверьте результаты

Этап 4: реализуйте план

Этап 5: наблюдайте за выполнением плана в реальном времени

Этап 6: корректируйте результаты

Этап 7: экспериментируйте

Глава 20

Важность больших данных для бизнеса

В ЭТОЙ ГЛАВЕ:

Бизнес-планирование с использованием больших данных
Внедрение систем больших данных в процесс корпоративного планирования
Принятие бизнес-решений с использованием больших данных
Знакомство с большими данными

Идея развития бизнеса с помощью анализа данных отнюдь не нова. Проблема больших данных существует с тех пор, как организации собирают информацию о своих бизнес-процессах, продуктах, существующих и потенциальных клиентах. Раньше компаниям было непросто и экономически невыгодно обеспечивать эффективное управление всеми корпоративными данными. По этой причине в течение предыдущих 30 лет использовались компромиссные решения. Специалисты по управлению данными стояли перед выбором: сохранять только данные, необходимые в текущий момент, или же создавать отдельные базы для хранения сегментов данных. Компании пытались применять временные меры для объединения данных, чтобы повысить качество процесса принятия управленческих решений. Для этого часто требовалось разрабатывать сложные программы, позволяющие получать правильную интерпретацию данных с точки зрения бизнеса.

Сложность и многофакторность не позволяли компаниям извлекать максимальную пользу для бизнеса из своих данных — это было слишком затратно. Среди этих факторов были следующие:

- большие затраты на приобретение достаточного количества систем и накопителей для физического размещения данных;
- невозможность управления слишком объемными базами данных, трудность архивации таких баз и обращения к ним;

- несовершенство технологий, необходимых для управления разнородными данными с требуемой скоростью;
- сложность написания и последующей поддержки программного кода для интеграции элементов данных;
- сложность актуализации данных и приведения их в соответствие новым бизнес-требованиям

В этой главе рассматриваются причины перехода компаний к использованию больших данных, а также примеры влияния больших данных на результативность бизнеса.

Большие данные как инструмент бизнес-планирования

Каких результатов рассчитывает достичь компания с помощью больших данных? Ответить на этот вопрос непросто. Подход к управлению корпоративными данными зависит от сферы, в которой работает компания. Тем не менее, в планировании и реализации бизнес-стратегий существует ряд типовых проблем, которые можно решать с помощью больших данных. Несмотря на то что во многих компаниях налажены механизмы управления взаимоотношениями с клиентами, становится гораздо сложнее обнаруживать меняющиеся запросы клиентов, даже выявляя взаимосвязи между данными из многочисленных источников.

Самая сложная задача для бизнеса — прогнозирование изменений и их причин. Компаниям необходимо быстро и эффективно принимать обоснованные решения. Знания нужны им для принятия мер, которые могут обеспечить требуемые результаты. Руководителям компаний необходимо оценивать те нюансы бизнеса, которые влияют на продуктовые линейки и партнерские экосистемы. В наиболее эффективных организациях используется комплексный подход к данным. Организация работы с большими данными включает в себя четыре этапа: планирование, выполнение, контроль и реагирование.

ЭТАП 1: планирование на основе данных

Компаниям доступно огромное количество данных. Однако опасность заключается в прогнозировании на основе лишь какой-то одной выборки из этих данных. Для того чтобы создать у руководителей сба-

лансированное представление обо всех элементах бизнеса, необходимо четко уяснить взаимосвязи между источниками данных. Проблема заключается в том, что компании, как правило, располагают гораздо меньшим объемом данных, чем требуется для принятия обоснованных решений. Необходимо выяснить, какие данные потребуются для выработки новых стратегий и направлений развития бизнеса.

Например, если ваша компания намерена расширить ассортимент своих услуг, то вам необходимо проанализировать как можно больше данных о текущих запросах покупателей и характере их изменений. Что нравится и что не нравится покупателям в ваших продуктах? Какие продукты предлагают конкуренты? Какие глобальные тенденции возникают, и как они влияют на изменение запросов покупателей? Как покупатели относятся к вашим продуктам по сравнению с продуктами конкурирующих фирм? С помощью эффективных методов обработки данных вы сможете создать мощный инструмент планирования. Данные могут подтвердить правильность выбранной вами стратегии. Но данные также могут свидетельствовать о необходимости ее изменения, зачастую весьма неожиданным образом. Использование широкого круга данных при планировании дает возможность проверять допущения и смотреть на бизнес с разных точек зрения.

ЭТАП 2: выполнение анализа

После того как бизнес-цели вашей организации определены, можно приступать к анализу данных. Анализ данных не является обособленным процессом, он требует изучения целого ряда новых инструментов и освоения новых навыков. Многие организации нанимают специалистов по обработке больших данных для того, чтобы они разобрались в огромном количестве разнородных данных и связали их с конкретными проблемами или перспективами бизнеса.



Поскольку рынок средств анализа больших данных развит очень слабо, количество универсальных и простых в использовании аналитических инструментов весьма невелико. Поэтому вам, возможно, придется прибегнуть к помощи опытных профессионалов из консалтинговых компаний. Это поможет достичь успеха. Анализ больших данных — очень динамичная, быстро меняющаяся сфера. Отсутствие развитых аналитических инструментов в сочетании с необходимостью анализа все большего количества источников данных вынуждает бизнес расширять

границы своих возможностей. Компании, которые научились планировать свой бизнес с помощью больших данных, понимают нюансы и изменения, влияющие на его эффективность. Например, если компания занимается электронной коммерцией, то необходимо анализировать результаты деятельности новых партнеров и определить, стимулируют ли они интерес покупателей и новые продажи. Возможно, вас заинтересуют отзывы в социальных сетях о новых услугах. В то же время вам потребуется сформировать четкое представление о предложениях основных конкурентов, способных снизить вашу прибыль.

ЭТАП 3: проверка результатов

Процесс анализа данных может оказаться настолько увлекательным, что можно забыть соотнести его результаты с действительностью. Отражает ли результат анализа реалии бизнеса? Достаточно ли точны те данные, которые вы используете? Действительно ли полезны для планирования данные, получаемые из тех или иных источников? Настало время убедиться в том, что источники данных ведут вас в правильном направлении. Многие компании пользуются сторонними источниками и не всегда оценивают качество их данных. Если вы строите планы и принимаете бизнес-решения на основе анализа, то позаботьтесь о максимальной достоверности исходных данных.

ЭТАП 4: выполнение плана

После завершения этой стадии анализа начинается реализация плана. Выполняемые действия должны укладываться в общий корпоративный цикл планирования, особенно в условиях растущей динамики рынков. Каждый раз, когда бизнес приступает к реализации новой стратегии, важно периодически оценивать его состояние с помощью больших данных. Принцип «действовать на основе анализа больших данных и проверять успешность выполнения бизнес-стратегии» является ключевым для успеха организации. Большие данные играют важнейшую роль в проверке правильности реализации стратегии компании по достигнутым результатам. Если фактические результаты не соответствуют ожиданиям компании, то в одних случаях она изменяет свою стратегию, а в других начинает развиваться в ином, более перспективном направлении.

Расширение цикла планирования

Появление больших данных меняет подходы к бизнес-планированию. Чем больше компаний использует облачные технологии, чтобы предложить новые услуги клиентам, тем важнее становится роль анализа данных. По этой причине целесообразно обратить внимание на следующий этап процесса планирования. После формирования начального плана и стратегии цикл работы с данными можно дополнить тремя этапами: наблюдением, корректировкой и экспериментальным моделированием.

ЭТАП 5: наблюдение в реальном времени

Средства анализа больших данных позволяют наблюдать за данными практически в реальном времени и предвосхищать изменения. Такая возможность может принести существенную пользу вашему бизнесу. Если ваша компания проводит клинические испытания фармацевтических препаратов, то в случае опасности судебного преследования вы можете заблаговременно изменить план испытания или вовсе отказаться от него. Производственная компания может наблюдать за показаниями датчиков и своевременно устранять ошибки в производственных процессах во избежание крупного ущерба.

ЭТАП 6: внесение изменений в процесс

Компания, имеющая в своем арсенале инструменты для постоянного наблюдения за данными, способна корректировать свои процессы и стратегию на основе анализа этих данных. Благодаря своевременному получению информации организация может оперативно изменять процессы и улучшать свои результаты. Для большинства компаний такой подход является инновационным: раньше аналитики, как правило, получали возможность обрабатывать результаты наблюдений лишь тогда, когда становилась очевидной та или иная проблема. Анализ позволял определить, почему возникла проблема, продукт потерпел неудачу на рынке, а клиенты остались недовольны оказанными услугами. Понять причину неудачи важно, однако еще важнее иметь возможность ее избежать.

ЭТАП 7: экспериментирование

В динамично меняющемся мире важно иметь возможность опробовать новые продукты и предложения. Но это неизбежно связано с риском. Экспериментирование, которое не позволяет быстро оценить результат, лишь запутывает покупателей и партнеров. Тем не менее, вы можете усовершенствовать свою бизнес-стратегию, сочетая эксперименты с наблюдением данных в реальном времени и быстрой корректировкой процессов. Наличие нужных данных позволит легко менять направления и результаты, а следовательно, меньше рисковать при экспериментировании.

Новые аспекты аналитики данных

Большие данные начинают серьезно влиять на стратегию бизнеса. Если в компании существует стратегия работы с большими данными, то руководство может пользоваться ими на протяжении всего цикла планирования, а не только в его конце. С развитием рынка больших данных компании смогут вести бизнес, основываясь на реальных данных об окружающей деловой среде. Так, с помощью прогностического анализа организация способна своевременно обнаруживать небольшие изменения потребительских предпочтений и оперативно корректировать свою стратегию. Компания Walmart заранее узнает с помощью социальных сетей, какие новые продукты начинают пользоваться спросом у покупателей. Розничные торговые организации мало что могут изменить в продуктах, когда они уже находятся на прилавках. Однако, предсказывая изменения покупательских предпочтений на полгода вперед, компания способна кардинально улучшить свои бизнес-результаты.

Может сложиться впечатление, что достаточно создать платформу для работы с большими данными, и корпоративная стратегия сформируется сама собой. Разумеется, в действительности все гораздо сложнее. Несмотря на то что большие данные становятся важным бизнес-инструментом, рассчитывать лишь на одни данные слишком рискованно. Руководителю необходимо рассматривать результаты анализа больших данных во взаимодействии с другими факторами, которые сложно свести к конкретному алгоритму. Не все стратегии можно беспрепятственно реализовывать в меняющихся условиях.

Анализ не всегда помогает выявить возникающие тенденции и изменения в конкурентной среде. Представители высшего руководства опираются не только на анализ данных, но и на собственные знания и интуицию. Не рассматривайте большие данные как панацею, способную решить любой стратегический вопрос, придерживайтесь взвешенного подхода.

Выбор правильной основы

Итак, с чего начать создание среды, в которой вы сможете экспериментировать с большими данными и использовать приобретенный опыт работы с ними? Нужно ли вкладывать средства в оснащение центра обработки данных новыми технологиями? Выгодно ли пользоваться облачными службами? Ответ на оба эти вопроса положительный. Вам придется пойти на изменения, чтобы обеспечить поддержку больших данных. Для начала определите, какие типы данных важны для вашей организации и какие из доступных сред позволяют управлять ими. Каждое из этих нововведений приспособлено для решения тех или иных задач.

Например, если ваш приоритет — быстрая обработка данных, то изучите базы данных, работающие в оперативной памяти. Если требуется обрабатывать большие объемы данных в реальном времени, стоит обратить внимание на технологии, работающие с потоковыми данными. Существует большое количество продуктов, обрабатывающих пространственные данные. В главах 1 и 4 описаны продукты и архитектуры, поддерживающие разнообразные структуры и процедуры анализа данных. Кроме того, полезно изучить облачные технологии, которые позволяют с низкими затратами хранить большие объемы информации. Благодаря облачным службам анализа данных организации теперь могут использовать сложные инструменты, которые раньше стоили слишком дорого.

Выбор стратегии работы с большими данными

Стратегия работы с большими данными — это, конечно, новые технологии. Но в первую очередь нужно сформировать команду специалистов, компетентных и в технике, и в бизнесе. Необходимо, чтобы руководство компании участвовало в планировании стратегии ис-

пользования нового поколения продуктов и услуг. Следует понимать, каких вопросов можно ожидать и как отвечать на эти вопросы.

Поэтому лучше всего начать с формирования команды. Ищите опытных консультантов, которые помогут вам освоить эффективные методы работы с большими данными. Помните, что на текущем этапе развития сферы больших данных вам придется пользоваться низкоуровневыми программными средствами. Работа с ними требует серьезных знаний в области программирования. По мере появления новых инструментов экспериментируйте с ними. Берите на вооружение все преимущества инноваций. Некоторые поставщики и консультанты создают на основе своего опыта комплексные информационные продукты, которые можно адаптировать к рынкам и бизнес-модели вашей компании.

Тем не менее, чтобы воспользоваться преимуществами новых технологий, вы должны сосредоточиться на главном. Выбирая элементы данных для анализа, вы должны быть уверены, что получите необходимый результат. Чтобы минимизировать риски, необходимо иметь возможность работать только с теми данными, которым вы доверяете. Каждый источник данных имеет свою особую структуру. Не всегда эти источники могут быть подвергнуты тщательной проверке. Перед тем как такие источники будут использованы для аналитики, удостоверьтесь в корректности их метаданных.

Начинайте работать с большими данными не спеша. Большинство компаний приходит к полнофункциональному общекорпоративному анализу больших данных не одномоментно. Переход к использованию больших данных является постепенным расширением традиционных методов анализа, которые уже хорошо изучены. И тем не менее, компании планируют и осуществляют экспериментальные проекты в сферах бизнеса, требующих быстрой обработки новых типов данных.

Планирование использования больших данных

С помощью больших данных можно решать почти неограниченный круг бизнес-задач. Возможность повышать эффективность бизнеса за счет сбора и анализа данных есть практически в любой отрасли. В одних сферах применение больших данных более очевидно, в других — менее. Вы можете сами привести сотни примеров того, как компании могут улучшить планирование и бизнес-процессы, используя данные

из социальных сетей. Тем не менее, диапазон применений больших данных гораздо шире — от наблюдения за производственными процессами до диагностики заболеваний. Руководители страховых компаний с помощью больших данных определяют, какие продукты оптимальны для конкретного клиента и при этом создают минимальный риск для страховщика.

Практически в любой отрасли руководители хотят выявлять тенденции и делать прогнозы с помощью анализа всех типов структурированных и неструктурированных данных. Компании изучают состояние дел в своем бизнесе, запрашивая данные у отделов обслуживания клиентов, собирая информацию с датчиков и знакомясь с записями системных журналов. Большие данные позволяют компаниям максимально эффективно сокращать риски и использовать имеющиеся возможности. В главах 21 и 22 пойдет речь о применении больших данных в различных отраслях.

Преобразование бизнес-процессов посредством больших данных

Все больше организаций осознает возможности, открывающиеся благодаря новым способам использования разнообразной информации. Развитие технологий позволит руководителям компаний расширить горизонты своих бизнес-стратегий. Здесь мы лишь кратко рассмотрели потенциальные возможности больших данных. С их помощью компании могут предотвращать мошенничество и избегать потери денег. Они позволяют осуществлять планирование в реальном времени, исходя из знаний о том, что клиенты покупают и что хотят купить. Врачи, имеющие доступ к большим объемам методической информации, способны лучше, быстрее, дешевле и результативнее лечить своих пациентов. Конечно, все это — лишь краткая оценка перспектив, которые открываются благодаря появлению больших данных. Для того чтобы подготовиться к необходимым изменениям бизнес-процессов, вашей организации нужно исследовать потенциал новых технологий.

Глава 21

Анализ данных в движении: практические примеры

В ЭТОЙ ГЛАВЕ:

Зачем компаниям нужны данные в движении
Потоковые данные для защиты окружающей среды
Потоковые данные в общественной жизни
Потоковые данные в здравоохранении
Потоковые данные в энергетике

Для успешного использования больших данных вам нужно научиться по-новому подходить к решению задач. В прошлом решение задач осложняли в первую очередь такие факторы, как отсутствие данных, ограниченная емкость хранилищ и вычислительная мощность, а также высокие затраты. Технологии больших данных устраняют прежние ограничения и открывают новые возможности для инноваций в различных сферах деятельности. Сегодня ежедневно генерируются огромные объемы данных, в которых вы, владея соответствующими технологиями, можете обнаруживать полезную информацию для бизнеса.

Скорее всего, вы уже вложили немало ресурсов в управление и анализ структурированных данных для получения информации о клиентах, организации работы компании и выполнения финансовых обязательств. Тем не менее, сегодня все наиболее интенсивно развивается другой тип данных. К нему относятся информация из социальных сетей, новостные ленты, биржевые котировки, лог-файлы, пространственные данные с датчиков, информация с медицинских устройств

и GPS-навигаторов. Все это данные, появляющиеся в реальном времени. Эти новые источники данных позволяют решать весьма сложные вопросы, поскольку обеспечивают мгновенный доступ к необходимым знаниям. Потоковые данные, или данные в движении, позволяют получать информацию о событии в тот момент, когда оно происходит.

В этой главе вы узнаете об организациях и целых отраслях, в которых внедряются инновационные методы работы с данными, непрерывно передаваемыми по коммуникационным каналам. Мы покажем, как можно решать бизнес-задачи с помощью технологий, описанных в предыдущих главах книги. Обработка и анализ непрерывных потоков данных с помощью потоковых вычислений подробно рассмотрены в главе 14, а подходы компаний к интеграции источников потоковых данных с другими источниками структурированных и неструктурированных данных — в главе 15.

Зачем нужны данные в движении

Оплата кредитной картой, операция на фондовой бирже, отправка электронного письма — все эти процессы предполагают перемещение данных из одного места в другое. Хранимыми данными, или данными в покое обычно называют данные, которые находятся в центре обработки данных или в облаке. Напротив, передаваемые данные, или данные в движении — это данные, которые перемещаются из одного хранилища в другое. Компании, которым требуется обрабатывать большие объемы данных в почти реальном времени, как правило, имеют дело с данными в движении. Данные в движении необходимы тогда, когда требуется быстро реагировать на их текущее состояние.

Как правило, данные в движении имеют большой объем. На практике многие источники, генерирующие большое количество данных, являются потоковыми.

- От работы медицинского оборудования зачастую зависят жизнь и здоровье. Такие устройства снабжены датчиками, которые оповещают персонал о любых отклонениях от нормальной работы. Регистрируемые данные постоянно находятся в движении. Они предоставляют техническим специалистам информацию о потенциальных сбоях. Это позволяет своевременно корректировать работу оборудования и обеспечивать безопасность пациентов.

- Телекоммуникационное оборудование, используемое для мониторинга больших объемов данных, позволяет обеспечить должный уровень обслуживания клиентов.
- Анализ данных, генерируемых и собираемых в точках продаж в реальном времени, позволяет понять потребительские предпочтения. Данные обрабатываются и анализируются в момент контакта с клиентом, возможно, с использованием информации о его местоположении и сведений, получаемых из социальных сетей.
- Финансовые организации непрерывно обмениваются сообщениями с данными о платежах и торговле акциями. Для обеспечения безопасности этих сообщений часто используются такие протоколы, как Advanced Message Queuing Protocol (AMQP, усовершенствованный протокол очередей сообщений) и MQSeries, разработанный компанией IBM. Оба протокола включают службы защиты информации.
- Сбор информации с датчиков отслеживания движения в охраняемой зоне позволяет отличить безопасные движения (например, бегущего кролика) от подозрительных (машина, приближающаяся к объекту на большой скорости).
- Медицинские устройства способны генерировать большие объемы данных о состоянии пациента, проверять наличие угрозы его жизни или здоровью, а также выявлять различные отклонения.

Ценность потоковых данных

Перемещаемые данные (часто в потоковой форме) приобретают все большее значение для компаний, которым критически важно быстро принимать решения. Анализ данных в реальном времени позволяет компании оперативно реагировать на ситуации, чтобы корректировать свои действия и избегать нежелательных их последствий. Сложность работы с потоковыми данными заключается в извлечении из них полезной информации в процессе создания и передачи данных. Такая обработка осуществляется до того, как данные оказываются в месте назначения. Потоковые данные могут оказаться очень полезными для вашей компании, если вы сумеете воспользоваться ими немедленно после создания или получения.

Необходимо обрабатывать и анализировать их в реальном времени, пока они находятся в движении и не помещены в хранилище. Это позволит оперативно реагировать на изменения. Вам необхо-

димо учитывать контекст этих данных и их отношение к собранной ранее информации, а также иметь возможность интегрировать их с традиционными данными. В первую очередь необходимо понимать особенности обрабатываемых потоковых данных. Нужно также понимать, какие результаты вы хотите получить. Например, вы работаете в производственной компании и следите за чистотой веществ, участвующих в технологическом процессе. Вас должны интересовать показания соответствующих датчиков, и это является веской причиной для использования потоковых данных. Однако иногда у вас есть возможность собирать большие объемы потоковых данных, которые не обязательно представляют интерес для бизнеса. Иными словами, наличие такой возможности не означает, что вы обязаны непременно воспользоваться ею.

Как же можно изменить бизнес с помощью потоковых данных? В следующих разделах вы познакомитесь с тем, как организации в различных отраслях извлекают пользу из данных в движении. Одни компании повышают эффективность работы с уже имеющимися у них данными. Другие начинают собирать данные, которые были недоступны им ранее. Некоторые организации получают возможность собирать гораздо больше данных, которые раньше могли накапливать лишь выборочно. Потоковые данные позволяют компаниям повышать качество обслуживания клиентов, пациентов, жителей целых городов, а возможно, и приносить пользу всему человечеству. Коммерческие организации используют потоковые данные, чтобы повлиять на решения, принимаемые клиентами в точках продаж.

Потоковые данные для защиты окружающей среды

В рамках исследований окружающей среды ученые измеряют и отслеживают различные характеристики озер, рек, океанов, морей, родников и других водных объектов. Важным этапом исследований в сфере сохранения и рационального использования водных ресурсов является наблюдение за подводными экосистемами и происходящими в них изменениями. Для чего это делается? Изменения окружающей среды способны оказывать огромное влияние на экономическое, физическое и культурное благополучие отдельных людей и целых групп населения во всем мире. Чтобы эффективнее прогнозировать воздействия на окружающую среду, ученые из университетов и экологических орга-

низаций различных стран мира постепенно включают анализ перемещаемых данных в свои исследования.

Процесс научного исследования предполагает непрерывный сбор больших объемов данных о состоянии водных ресурсов и погодных условиях. Это позволяет снижать риски для людей и реагировать на катастрофы, угрожающие природным ресурсам. Прогнозирование таких характеристик, как уровень подтопления в определенной местности или величина ущерба, наносимого нефтяным пятном морской экосистеме, осуществляется с помощью математических моделей. Для этого используются разные типы данных — от сведений о температуре воды и концентрации содержащихся в ней веществ до скорости потока. Кроме того, для одних и тех же водных объектов полезно сравнивать новые данные наблюдений с предыдущими.

Существует много сложных исследовательских программ, нацеленных на поиск методов защиты естественных водных ресурсов. Например, реки и прилегающие к ним болотистые и затопляемые территории нуждаются в защите, поскольку являются средой обитания диких животных и растений. Многие населенные пункты используют реки в качестве транспортных артерий, туристических объектов, источников питьевой воды и еды, для генерации электроэнергии. Кроме того, наблюдение за состоянием рек позволяет прогнозировать наводнения и заблаговременно предупреждать о них жителей прибрежных поселков. Ученые надеются, что проведение таких исследований в реальном времени поможет существенно улучшить жизнь людей.

Получение информации о реках и океанах с помощью датчиков

В одном научном центре в США с помощью датчиков ведется сбор данных о физических, химических и биологических характеристиках рек. Датчики отслеживают пространственные изменения температуры, давления, минерализации, мутности и химического состава воды. В перспективе планируется создание единой сети для наблюдения за состоянием рек и их эстуариев в реальном времени. Исследователи предполагают, что в будущем они смогут прогнозировать изменения в характеристиках рек так же, как метеорологи сегодня прогнозируют погоду. Другой научный центр, расположенный в Европе, собирает с помощью буйков, оснащенных датчиками и средствами радиосвязи,

данные об океане (в том числе о высоте волн и волновой активности). Эти потоковые данные, объединенные с другими данными о погоде и окружающей среде, обеспечивают рыбаков и ученых актуальными сведениями о состоянии океана.

В обоих описанных выше примерах датчики генерируют большие объемы данных в реальном времени. Несмотря на разнообразие инфраструктурных платформ, данные, собранные с датчиков, как правило, интегрируются с помощью связующего ПО с уже находящимися в хранилищах данными. Научно-исследовательские организации пользуются также внешними источниками данных, такими как картографические базы, геоинформационные системы, данные с принадлежащих третьим сторонам датчиков и т.д. По мере поступления из перечисленных источников данные анализируются и обрабатываются. Пусть организация создает интегрированную сеть датчиков, роботов и мобильных систем мониторинга. Она использует получаемую информацию для построения сложных моделей, в том числе многопараметрических систем моделирования в реальном времени. Эти модели позволяют изучать динамику взаимодействия элементов внутри локальных экосистем рек и эстуариев.

Преимущества данных, поступающих в реальном времени

Анализ данных об окружающей среде в реальном времени позволяет ученым глубже понимать ключевые экологические проблемы. Технологии обработки потоковых данных открывают новые области для исследований и задают направления развития методов сбора и анализа научных данных. Ученые по-новому рассматривают уже собранную информацию и получают возможность использовать данные из новых источников. Большой объем информации дает уже периодическое наблюдение за изменениями значений температуры или химического состава воды. Однако при наблюдении лишь в определенные интервалы времени есть риск не заметить важные изменения или закономерности.

Анализ потоковых данных по мере их поступления способен восполнить этот недостаток информации. Сбор данных о состоянии рек и погоды в реальном времени позволяет прогнозировать изменения и реагировать на них. Ученые надеются, что прогнозировать изменения окружающей среды можно будет так же, как и погоду. Они продолжают изучать влияние глобального потепления и делать выводы

из наблюдений за миграцией рыб. Как наблюдения за транспортировкой ядовитых веществ помогут избежать загрязнения окружающей среды?

Имея доступ к уже собранным данным, аналитики могут гораздо эффективнее объединять их с данными, получаемыми в реальном времени. Это позволяет глубже анализировать информацию и точнее предсказывать будущие события. Результаты такого комплексного анализа могут использовать другие специалисты, которым также необходима перечисленная информация. Эти данные можно сохранять в облачной среде, доступной исследователям со всего мира, которые смогут добавлять к ним новые данные и использовать их в решении смежных экологических проблем.

Потоковые данные в общественной жизни

Сбор данных присутствует практически во всех сферах городской жизни: сведения о разного рода платежах, контроль за состоянием зданий и мостов, наблюдение за транспортными потоками, определение географического положения пользователей, данные о криминогенной обстановке. Для обеспечения безопасной и удобной жизни в городах создаются программы их развития. Это требует анализа большого количества данных из различных источников. Сбор значительной части данных для исследований, связанных с развитием городов, выполняется различными муниципальными организациями. Раньше для анализа этих данных требовались месяцы, а иногда и годы (примеры — данные ежегодных переписей населения, полицейские и налоговые досье). Даже в пределах одной муниципальной организации (например, в полиции) данные могут накапливаться раздельно по каждому району города. Собранные таким образом данные весьма непросто будет использовать для решения проблем, затрагивающих и город в целом, и его окрестности.

Таким образом, руководство города располагает огромным объемом информации о влиянии проводимой политики городского развития на качество жизни горожан за предыдущие годы. Однако использовать такие данные для принятия решений, полезных для города, в реальном времени — крайне непростая задача. Использование изолированных друг от друга баз данных и хранилищ существенно усложняет работу с информацией. Это вызывает проблемы, поскольк-

ку различные процессы управления городом напрямую связаны друг с другом. Городские чиновники постепенно осознают, что для усовершенствования системы муниципального управления необходимо использовать как оперативную информацию, так и данные, связанные с ранее принятыми управленческими решениями. Чем сложнее городское хозяйство, тем более важную роль играют данные в решении проблем города и его жителей.

Ситуация меняется по мере того, как чиновники, ученые и новаторы объединяют усилия для выработки программ управления городом с использованием данных в движении. Например, чтобы создать и внедрить программу по борьбе с заторами на дорогах, необходимы данные о численности и занятости городского населения, состоянии дорог и погоде. Многие из этих данных уже собраны, однако размещены в разных хранилищах и носят ретроспективный характер. Формирование предложений и рекомендаций на основе актуальной информации требует нового подхода. Исследователи одного из европейских технических университетов собирают информацию о дорожном движении. В реальном времени они получают данные из различных источников — от транспортных средств с системой GPS (Global Positioning System, глобальной системой позиционирования), радарных датчиков и метеостанций. С помощью интеграции и анализа потоковых данных удалось сократить заторы и повысить эффективность дорожного движения. Анализ структурированных и неструктурированных данных в реальном времени позволяет системам оценивать текущие дорожные условия и строить оптимальные маршруты. Глобальная цель исследования — существенно улучшить движение транспорта. Для того чтобы рекомендации системы соответствовали текущей ситуации, поступающие актуальные данные сопоставляются с накопленными ранее.

Потоковые данные существенно влияют на снижение уровня преступности. Например, один из отделов полиции с помощью прогностического анализа выявляет закономерности в месте и времени совершения определенных преступлений. Если где-то вдруг отмечаются сходные условия, то отдел своевременно направляет туда полицейских. Впоследствии собранные данные можно использовать для анализа, чтобы выявить новые признаки, характерные для преступной деятельности.

Потоковые данные в сфере здравоохранения

Большие данные играют очень важную роль в сфере здравоохранения. Они применяются в самых разных ее областях — от генетических исследований и сложной рентгенографии до анализа качества медицинского обслуживания. Анализ больших данных в этих областях важен для проведения углубленных исследований, однако его ключевое достоинство заключается в том, что его результаты можно использовать в клинической медицине. Сбор достаточного количества данных, их быстрое, правильное и своевременное применение помогают спасать людям жизнь. Благодаря потоковым данным практикующие врачи и исследователи могут быстрее принимать решения и улучшать процесс лечения пациентов в стационарах.

В процессе лечения больных врачи используют большие объемы данных, актуальных в течение определенного времени. Это, например, результаты анализов и гистологических исследований, томограммы и рентгеновские снимки. Врачи также измеряют показатели жизнедеятельности пациентов, такие как давление, пульс и температура, с помощью медицинских приборов. Несмотря на то что приборы оповещают врачей об отклонениях показаний от нормы, они не дают заблаговременных предупреждений и не позволяют профилактировать такие отклонения. Как правило, небольшие изменения состояния здоровья пациента бывает сложно обнаружить путем осмотра, однако приборы могут зафиксировать эти изменения при наличии мгновенного доступа к данным.

Устройства, используемые при интенсивной терапии, генерируют тысячи показаний в секунду. Раньше, в силу ограничений, фиксировалось некое среднее значение показателя за период от 30 до 60 минут. Медицинские устройства позволяли получать огромные объемы данных, однако из-за технологических ограничений анализировать эти данные было невозможно.

Сохранение потока данных

С помощью технологии потоковых данных группа исследователей университетской клиники получает информацию от прикроватных мониторов и обрабатывает ее с применением алгоритмов раннего обнаружения признаков серьезных инфекционных заболеваний. Полученные данные анализируются в реальном времени. Это позволяет

заблаговременно предупреждать врачей об изменениях в состоянии пациента. Иногда врачам удается принимать превентивные меры на 24–36 часов раньше, чем при отсутствии технологии обработки потоковых данных. Потоковые данные также позволяют врачам сравнивать результаты анализов пациента с хранящимися в базе данных результатами его предшествующих обследований. Такое сравнение может дать врачу дополнительную полезную информацию.

Потоковые данные в энергетике

Снижение энергопотребления, повышение его эффективности и поиск новых возобновляемых источников энергии — приоритеты современной энергетики, обеспечивающие защиту окружающей среды и экономический рост. Для достижения этих целей все больше данных собирается и анализируется в реальном времени.

Многие крупные организации принимают комплексные меры, чтобы обеспечивать себя энергетическими ресурсами как сейчас, так и в будущем. Из-за дефицита углеводородного топлива и его удорожания все больший интерес вызывают нетрадиционные источники энергии — ветряные турбины, солнечные электростанции и волновые энергетические установки. Такие организации самостоятельно генерируют и сохраняют энергию. Для поддержания баланса между спросом и предложением им необходимо получать достоверную информацию в реальном времени. С помощью потоковых данных эти организации измеряют и контролируют объемы спроса и предложения электроэнергии для того, чтобы понимать свои потребности и принимать решения об объемах потребления электроэнергии в реальном времени.

Повышение энергоэффективности с помощью потоковых данных

Организации повышают свою энергоэффективность с помощью больших данных, что наглядно иллюстрируется следующими двумя примерами.

Крупный университет изучает потоковые данные о внутреннем энергопотреблении и интегрирует их с метеоданными. Это позволяет

в реальном времени корректировать объемы производимой и потребляемой энергии.

Участники бизнес-сообщества совместно используют и анализируют потоковые данные, что позволяет эффективнее расходовать энергию и сокращать затраты на нее. Потоковые данные позволяют компаниям отслеживать спрос и предложение, учитывать объемы генерируемой и потребляемой электроэнергии. Это позволяет изменять генерируемую мощность в соответствии с текущими потребностями.

Использование потоковых данных для развития альтернативных источников энергии

Компании постепенно начинают использовать потоковые данные для эффективного исследования и разработки альтернативных источников энергии, как показывают следующие два примера.

Научно-исследовательский институт с помощью потоковых данных изучает возможность использования морских волн в качестве возобновляемого источника энергии. Для этого необходимо собирать информацию о большом количестве различных параметров, в том числе о температуре, о фазах луны, о приливах и отливах, геопространственные данные. Для наблюдения и анализа шума, создаваемого волновой электростанцией, используются контрольно-измерительные устройства, коммуникационные технологии, облачные вычисления и инструменты анализа потоковых данных. Исследователи изучают влияние шума на рыб и других морских обитателей.

Компания по эксплуатации ветровых электростанций использует потоковые данные для часовых и суточных прогнозов объемов генерируемой электроэнергии. Проводится сбор данных о состоянии турбин, температуре, атмосферном давлении, влажности, осадках, направлении и скорости ветра на различных высотах (от 0 до 90 м над уровнем земли). Источниками этих данных являются тысячи метеорологических станций, расположенных по всему миру, а также принадлежащие компании ветроэлектрические установки. Как компания может использовать полученные данные? Можно смоделировать воздушные потоки, определить розу ветров и характер вихреобразования возле существующих установок. Результаты анализа используются для оптимизации размещения ветроустановок и сокращения затрат на единицу генерируемой электроэнергии.

Объединение потоковых данных с ретроспективными и другими источниками данных в реальном времени

Приведенные в этой главе примеры демонстрируют возможности применения потоковых данных. Наиболее важно для эффективного использования потоковых данных соотносить их с ситуацией. Например, врач видит, что результаты анализов пациента указывают на определенное заболевание. Однако исследование данных других пациентов может показать, что при схожих симптомах и результатах анализов существуют альтернативные варианты диагноза. В сложном современном мире данные позволяют правильно действовать лишь при условии, что они рассматриваются применительно к решаемой задаче.

Оптимизация бизнес-процессов с помощью анализа больших данных: практические примеры

В ЭТОЙ ГЛАВЕ:

Для чего компаниям необходимо анализировать большие данные
Как повысить качество обслуживания клиентов с помощью текстовой аналитики
Обеспечение индивидуального подхода к клиентам с помощью больших данных
Применение больших данных для борьбы с мошенничеством
Конкурентные преимущества для бизнеса от интеграции новых источников данных

Сегодня становится все очевиднее, что люди, принимающие решения в бизнесе, должны научиться по-новому интерпретировать данные, чтобы их компании могли справиться с резкими изменениями на рынке. Топ-менеджеры понимают, что фундаментом будущего успеха организации являются инновации, внимательный, упреждающий и индивидуальный подход к клиентам. Результаты бизнеса тесно связаны с минимизацией рисков, решением вопросов корпоративного управления и безопасности. Чтобы соответствовать меняющимся условиям бизнес-среды, компаниям необходимо своевременно получать достоверную информацию. Во многих компаниях анализ больших данных занимает центральное место в бизнес-стратегии, что позволяет укреплять отношения с клиентами и партнерами, а также ускоряет принятие управленческих решений.

В этой главе вы познакомитесь с отдельными организациями и целыми отраслями, внедряющими новые методы управления и анализа структурированных и неструктурированных данных. Это позволяет справиться с обработкой растущих объемов разнообразной, динамично поступающей информации и получать достоверные результаты. Вы узнаете о том, как технологии, описанные в предыдущих главах этой книги, используются для решения реальных бизнес-задач.

Осознание потребности компаний в анализе больших данных

Информация, с помощью которой компании уникальным образом удовлетворяют нужды своих клиентов и партнеров, больше не хранится исключительно в традиционных базах данных. Очевидной становится ценность неструктурированных данных, поступающих из новых источников. Руководители осознали, что быстрый анализ неструктурированных данных (как текстовых сообщений в системах поддержки клиентов, так и публикаций на сайтах социальных сетей) позволяет получать важную информацию. Анализируя большие объемы данных и в реальном времени сопоставляя полученные результаты с решениями клиентов, компания может значительно увеличить свою прибыль. Совместное использование структурированных и неструктурированных данных в бизнес-процессах позволяет компании быстро реагировать на изменения, быть более гибкой, а главное, увеличить прибыль.

Повышение степени удовлетворенности клиента качеством обслуживания с помощью текстовой аналитики

Многие компании накапливают гигантские объемы неструктурированных данных, содержащих отзывы своих клиентов, но не в полной мере используют их возможности. Неструктурированные данные содержатся в электронных письмах и текстовых сообщениях, заметках информационно-справочных служб, комментариях в опросных листах, твитах и блогах. На долю этой категории приходится около 80%

данных, поступающих в компании, и этот показатель продолжает расти. Как правило, неструктурированные данные требуют многочасовой обработки вручную, и многие компании не уделяют их анализу достаточного внимания. Однако становится очевидно, что своевременный анализ таких данных позволяет узнавать о неудовлетворенности клиентов или возможных недостатках продуктов и вовремя реагировать на них. Компании рассматривают развитие текстовой аналитики как важную возможность, позволяющую глубоко анализировать большие объемы неструктурированных данных в реальном или почти реальном времени, а полученные результаты использовать в процессе принятия решений. Текстовая аналитика предполагает анализ неструктурированного текста, извлечение полезной информации и ее структуризацию. Полученные данные можно использовать различными способами (более подробно текстовая аналитика рассматривается в главе 13).

Рассмотрим, как это происходит на практике. Компания по прокату автомобилей испытывала серьезное давление со стороны молодых фирм, которые вели бизнес с меньшими издержками, и была вынуждена вести с ними конкурентную борьбу. Компания решила, что ключом к победе в этой борьбе является готовность реагировать на пожелания клиентов, и стала кардинально улучшать качество обслуживания, используя текстовую аналитику. Компания запрашивала у клиентов отзывы о своих услугах через интернет-опросы, электронную почту и текстовые сообщения. Клиенты информировали компанию о недостатках в обслуживании — чрезмерно долгом ожидании, неквалифицированной работе агентов, случаях предоставления автомобиля, отличного от заказанного. Тем не менее, компания проявляла непоследовательность при интерпретации клиентских отзывов и реагировании на них. Подход был выбран верный, но компания отвечала слишком медленно и анализировала данные бессистемно. Менеджеры читали электронные письма и комментарии, полученные в ходе опросов, и распределяли их по категориям для дальнейшего изучения. К сожалению, это занимало много времени; кроме того, менеджеры не сформировали единых принципов распределения комментариев по категории. Они часто не замечали системных признаков неудовлетворенности клиентов и проблем, которые были бы очевидны при одновременном рассмотрении большого количества комментариев.

Была поставлена задача: ускорить обработку клиентских отзывов, чтобы обнаруживать потенциальные проблемы в реальном времени и разрешать их на ранних стадиях. Менеджеры внедрили решение для текстовой аналитики, которое позволяло быстро находить интересующую информацию сразу во всех типах источников структурированных и неструктурированных данных. Они также использовали решение для анализа эмоциональной окраски высказываний, которое автоматически обнаруживало отзывы, требующие немедленного вмешательства. Таким образом, менеджеры компании получили возможность собирать и анализировать в реальном времени большие объемы данных об удовлетворенности клиентов и оперативно на них реагировать.

Ценность использования аналитики больших данных для бизнеса

Этой компании удалось существенно повысить уровень удовлетворенности своих клиентов. Теперь она эффективно отслеживает уровень качества услуг проката автомобилей и оборудования, обнаруживает возникающие проблемы и быстро устраняет их. Компания хорошо осведомлена о проблемных областях своего бизнеса и выявляет трудности гораздо быстрее, чем раньше. Благодаря новому аналитическому инструменту менеджеры заблаговременно получили информацию о проблемах в одном из регионов, где работала компания. Менеджеры исправили ситуацию и повысили удовлетворенность местных клиентов услугами компании.

Использование аналитики больших данных для повышения эффективности бизнеса

Сегодня клиент сам выбирает, как ему взаимодействовать с продавцом или поставщиком услуг. Такое положение дел характерно для телекоммуникационной, страховой, банковской сферы, розничной торговли и других отраслей. Покупатель имеет доступ к многочисленным каналам продаж, хорошо информирован и все чаще принимает решения о покупках с мобильного устройства. Чтобы сохранить конкурентоспособность на динамичном и хорошо информированном

рынке, вам необходимо выстраивать взаимоотношения с каждым конкретным покупателем на основе подробных знаний о нем. Каким образом компания может сделать покупателю подходящее предложение в тот момент, когда он принимает решение о покупке? Как научить специалистов по работе с клиентами определять запросы конкретного клиента и его ценность для компании? Как интегрировать и анализировать многочисленные источники структурированной и неструктурированной информации, чтобы делать самые заманчивые предложения клиенту уже в момент его привлечения? Как быстро определить ценность клиента, понять его потребности и осуществить продажу так, чтобы покупатель остался доволен?

Руководство компаний все чаще расценивает анализ больших данных как секретное оружие, позволяющее в условиях жесткой конкуренции формировать индивидуальный подход к клиентам. Изучение личных запросов конкретного клиента с помощью аналитики играет важную роль в работе с агрессивными и хорошо информированными потребителями. Компании все активнее используют социальные сети и мобильные сервисы, чтобы вовремя получать доступ к своим клиентам через желаемые средства коммуникации. Для успешной работы с клиентами в современном мобильном мире компании должны предоставлять им максимально персонализированные предложения. Чтобы получить конкурентные преимущества, компании интегрируют технологии быстрого анализа больших данных в реальном времени в свои аналитические платформы. Они преследуют несколько целей:

- лучше понимать уникальные потребности каждого клиента. Своевременно пользоваться получаемыми знаниями;
- демонстрировать желание удовлетворять запросы клиента при общении с ним;
- интегрировать данные, поступающие в реальном времени из разных источников, с аналогичными архивными данными об уже сделанных покупках, чтобы предлагать клиентам персональные рекомендации в момент продажи;
- предоставлять специалистам по работе с клиентами знания, позволяющие им выстраивать оптимальную схему взаимодействия с конкретным клиентом;
- повышать удовлетворенность клиентов и эффективнее поддерживать их лояльность;

- делать клиентам предложения, которые с максимальной вероятностью будут приняты.

Какие же технические решения обеспечивают индивидуальный подход к клиенту? Компании интегрируют и анализируют большие объемы неструктурированных и потоковых данных. Их источниками являются поступающие электронные письма, текстовые сообщения, заметки информационно-справочных служб, формы интернет-опросов, аудиозаписи, данные с GPS-устройств и из социальных сетей. Иногда компании находят новые варианты использования данных, которые в силу слишком большого объема, интенсивности поступления или неподходящей структуры раньше не поддавались обработке при помощи аналитических и прогностических моделей. Теперь компании могут строить более сложные модели и с их помощью обрабатывать данные, поступающие из разнообразных источников в реальном времени. Корпоративные аналитики ищут в данных закономерности, позволяющие делать выводы о предпочтениях и поведении клиентов. Важнейшим требованием к обработке данных является ее скорость. Если вы хотите быть успешным в динамичном, мобильном мире, то вам нужна модель, которая очень быстро прогнозирует оптимальное следующее действие.

Современные технологии позволяют компаниям прорабатывать и принимать решения в течение нескольких минут, а не часов или недель, как это было ранее. Для прогнозирования оптимального следующего действия часто требуются сложные алгоритмы машинного обучения для когнитивных вычислительных сред, подобных системе Watson компании IBM. Среда Watson обрабатывает большие объемы данных, анализирует данные в движении, позволяет компаниям изучать своих клиентов и моментально генерировать персональные предложения в ответ на их запросы. Это делается с помощью когнитивных вычислений (более подробная информация о системе Watson приведена в главе 13).

Рассмотрим несколько реальных примеров компаний, которые работают в сфере финансовых услуг и вкладывают значительные средства в новые технологии изучения клиентов и реагирования на их запросы.

Страховая компания хочет повысить результативность и эффективность работы сотрудников своей информационно-справочной

службы. Агенты компании не смогли быстро определить весь набор бизнес-направлений, необходимых потребителям, поэтому оказалось трудно выделить важных клиентов, требовавших к себе особого внимания. Затрачивалось очень много времени на поиск записей, создававшихся в процессе предшествующего общения с тем или иным клиентом, и необходимая информация, позволяющая решить проблему, обнаруживалась слишком поздно. Интенсивное взаимодействие компании с клиентами стало все чаще вызывать недовольство. В сложившейся ситуации компания внедряет решение, которое преобразует аудиозаписи телефонных разговоров в тексты, а также обнаруживает и анализирует ключевые слова. Полученная информация объединяется с архивными данными о клиенте. Это позволяет выявлять приоритетных клиентов, требующих немедленного внимания к себе, а также своевременно и надлежащим образом реагировать на остальные обращения.

Международный банк обеспокоен тем, что обработка информации о клиентах занимает слишком много времени. Банк хочет предоставлять сотрудникам своей информационно-справочной службы более подробные данные о клиентах, а также лучше понимать круг общения клиентов (в семье, на работе и в социальных сетях). Руководство располагает большими объемами как структурированной, так и неструктурированной информации о клиентах. В том числе это электронные письма, отзывы, переписка в чатах и аудиозаписи разговоров с информационно-справочной службой. Банк внедрил решение для анализа больших данных, которое улучшает качество поддержки клиентов. Представители банка получили оперативный доступ к информации о потребностях каждого клиента. Платформа, на которой построено это решение, определяет круг общения клиента, собирая информацию из социальных сетей, и объединяет данные из множества внутренних и внешних источников. С помощью этих данных банк может узнавать о важных событиях, происходящих в жизни конкретного клиента, а агенты — выбирать оптимальные дальнейшие действия. Например, если ребенок клиента оканчивает школу, будет уместно предложить клиенту оформить кредит на оплату высшего образования.

Организация, эмитирующая кредитные карты, собирается повысить эффективность мониторинга удовлетворенности и реагировать на индивидуальные запросы клиентов. Компания заинтересована в том, чтобы добиться действительно индивидуального подхода к клиентам,

а не просто нацеленного на различные социальные категории. Для этого была разработана система анализа больших данных, которая интегрирует информацию из традиционных структурированных источников (например, данные о транзакциях клиентов) с неструктурированными и потоковыми данными (сведениями о посещении сайтов, лентами твиттера и другими данными социальных сетей). Чтобы получить возможность формировать целевые предложения, компании требуется детально сегментировать своих клиентов. Благодаря быстрому анализу больших объемов информации разработанное решение позволяет компании эффективно выделять клиентов, мотивированных на совершение покупок, и формировать для них оптимальные предложения.

Применение аналитики больших данных для предотвращения мошенничества

Согласно многочисленным оценкам, не менее 10% выплат страховые компании совершают по сфальсифицированным претензиям. Сумма таких выплат в мировом масштабе составляет миллиарды, а возможно, и триллионы долларов. Проблема мошенничества в страховой сфере существует давно и со временем только усугубляется, а злоумышленники пользуются все новыми и новыми методами. Мошенничество происходит во всех видах страхования, в том числе в страховании автомобилей, здоровья, бизнеса и сотрудников компаний. Мошенником может оказаться как клиент, который инсценирует падение в торговом центре и затем предъявляет поддельную претензию в связи с переломом руки, так и сотрудники страховщика, вовлеченные в процесс возмещения ущерба от несчастных случаев, врачебных ошибок или другие процедуры работы с претензиями. Страховое мошенничество широко распространено, порой создаются целые преступные группы, связанные с деятельностью в области ремонта автомобилей и жилья, медицинской помощи, юридических услуг и других сферах, где часто задействованы страховые претензии.

Как страховые компании могут с помощью анализа больших данных распознавать мошенничество? Страховым компаниям необходимо предотвращать мошенничество заранее — до начала процедуры обработки претензии. Разрабатывая прогностические модели

на основе архивных и текущих данных о заработных платах, выплатах по медицинским страховкам, затратах на услуги адвокатов, демографических показателях, состоянии погоды, заметках информационно-справочной службы и аудиозаписях переговоров, компании могут эффективно обнаруживать подозрительные обращения на ранних стадиях общения с клиентами. Например, при рассмотрении заявления о возмещении вреда необходимо выяснить, подлинное ли медицинское заключение предоставлено и не был ли несчастный случай инсценирован. Организации отмечают в сфере медицинского и автомобильного страхования рост числа случаев мошенничества, совершаемого организованными преступными группами. Злоумышленники могут применять одни и те же схемы мошенничества в различных регионах и подавать претензии от заявителей с вымышленными именами. Анализ больших данных позволяет быстро обнаружить закономерности, сходство с уже подававшимися претензиями, своевременно остановить процедуру возмещения по сомнительному заявлению.

Эксперты по управлению рисками и борьбе с мошенничеством, специалисты по страхованию и руководители страховых компаний отмечают, что анализ больших данных приносит значительную пользу их бизнесу, поскольку позволяет прогнозировать попытки мошенничества и предотвращать их. Цель анализа — обнаружение криминала при первом уведомлении об ущербе, т.е. в момент, когда возникает необходимость участия специалиста по страховым расчетам или страхового агента.

Рассмотрим следующий практический пример. Страховая компания стремится эффективнее принимать решения в реальном времени при обработке поступающих обращений. Ее расходы, в том числе судебные издержки, связанные с мошенничеством от неправомерных выплат, неуклонно растут. В компании действуют подробные регламенты, помогающие агентам оценивать законность обращений. Но агенты были лишены своевременного доступа к информации, необходимой для принятия обоснованных решений. Компания реализовала платформу для анализа больших данных, которая обеспечила интеграцию и анализ данных из различных источников. Она широко использует данные из социальных сетей, потоковые данные, а специалисты могут получить доступ к информации в реальном времени. Сотрудники информационно-справочной службы уже при первом обращении

заявителя в страховую компанию могут быть осведомлены о возможных закономерностях поведения заявителя и его взаимоотношениях с поставщиками услуг.

Например, в момент подачи нового обращения специалист получает оповещение о том, что заявитель фигурирует в качестве свидетеля в аналогичном обращении, поданном полгода назад. Если будет обнаружено еще что-то необычное или подозрительное, заявителя об этом уведомят, и процесс обработки сомнительного обращения удастся прервать в самом начале. Иногда в социальных сетях обнаруживаются данные, которые свидетельствуют о том, что события, изложенные в претензии, не происходили в указанный заявителем день. Например, заявитель утверждает, что его автомобиль утрачен по причине наводнения, однако информация в социальных сетях показывает, что во время наводнения застрахованный автомобиль находился в другом городе.

Страховые компании несут из-за мошенничества столь большие убытки, что это стимулирует руководителей оперативно внедрять системы анализа больших данных и другие передовые технологии для их снижения. Убытки негативно влияют не только на страховщиков, но и на их клиентов, которые вынуждены приобретать страховку по более высоким ценам, компенсирующим эти убытки для компании. Анализ большого объема структурированных и неструктурированных данных, содержащихся в обращениях, позволяет компаниям выявлять закономерности поведения злоумышленников и в реальном времени обнаруживать попытки мошенничества. Отдача от внедрения технологий анализа данных может оказаться существенно больше вложений в них. Компании получают возможность за несколько минут проанализировать сложную информацию и сценарии аварий. В отсутствие платформы обработки больших данных это заняло бы от нескольких дней до нескольких месяцев.

Конкурентные преимущества для бизнеса от интеграции новых источников данных

Анализ больших данных позволяет компаниям находить новые решения давно существующих проблем. Коммерческие фирмы традиционно концентрируют свои усилия на повышении качества обслуживания клиентов, своевременном предложении им подходящих товаров

и услуг, сокращении рисков и борьбе с мошенничеством. Сегодня ситуация трансформируется. Компании меняют взгляд на собственный бизнес, интегрируя новые источники неструктурированных данных (блоги, заметки информационно-справочных служб, электронные письма, записи журналов регистрации и геопространственные данные) с традиционными источниками данных о транзакциях, клиентах и бизнес-процессах. Появилась возможность собирать недоступные ранее данные, выявлять с их помощью поведенческие закономерности и получать ценную для бизнеса информацию. Интеграция разнообразных источников данных позволяет компаниям глубже понимать своих клиентов, продукты и риски.

Часть VII

Раздел «десятка лучших»

В ЭТОЙ ЧАСТИ:

- Наилучшие методы работы с большими данными
- Источники больших данных
- Отраслевые стандарты, касающиеся больших данных
- Конференции, посвященные большим данным
- Прогнозирование тенденций в сфере больших данных

Глава 23

Десять лучших способов работы с большими данными

В ЭТОЙ ГЛАВЕ:

Осознание ваших целей
Определение плана действий
Оценка имеющихся данных и потребности в данных
Обеспечение безопасности управления и надежности данных
Проверка полезности данных

Несмотря на то что мы находимся в начале пути освоения больших данных, вы можете начать изучение с эффективных методов их применения. Затем можно подумать, как использовать полученные теоретические знания и практический опыт. Когда вы имеете дело с новой и перспективной технологией, важно понимать, как ее использовать, и подготовить четкий план действий. В этой главе описаны десять ключевых методов, которые необходимо освоить, прежде чем вы начнете работать с большими данными.

1. Определите ваши цели

Многие организации начинают экспериментировать с большими данными в рамках проекта, который способен принести конкретные результаты. Выбрав такой проект, вы получаете возможность свободно тестировать технологии, при этом не рискуя понести капитальные затраты. Тем не менее, если вы ограничитесь лишь серией однократных экспериментов, то, скорее всего, к тому времени, как вы убеди-

тесь в целесообразности использования больших данных в вашей компании, у вас еще не сложится четкий план работы с ними. Таким образом, когда вы проведете эксперименты и четко уясните потенциальные возможности больших данных, следует определить краткосрочные и долгосрочные цели. Какие результаты вы рассчитываете получить, используя большие данные? Повысится ли доходность каких-либо направлений вашего бизнеса благодаря прогнозированию поведения клиентов и выявлению их потребительских предпочтений с помощью большего количества данных? Для того чтобы хорошо сформулировать цели, важно организовать сотрудничество между техническими и бизнес-подразделениями компании.

Определение целей использования больших данных — лишь первый шаг в вашей работе. Теперь предстоит разобраться с потенциальными проблемами. В этот процесс должны быть вовлечены все участники бизнеса. Большие данные затрагивают все аспекты работы компании. Это, в частности, и хранение архивных данных, и управление источниками информации, осуществляемое различными бизнес-подразделениями. В некоторых подразделениях может возникнуть необходимость изучения тех источников данных, с которыми ранее работали лишь немногие менеджеры. Создание рабочей группы по использованию больших данных — хороший способ объединить представителей различных подразделений компании. Это позволит выявить возможные проблемы обработки данных. С течением времени эта группа может начать содействовать внедрению эффективных методов работы с большими данными в различных подразделениях компании. В состав рабочей группы должны входить топ-менеджеры, определяющие стратегию и направление развития компании.

2. Определите план действий

На этом этапе вы уже провели эксперименты с большими данными и определили цели компании. Вы четко понимаете, какие результаты должны быть достигнуты руководством компании и ее бизнес-подразделениями. Настало время выработать план действий. План действий описывает то, что подлежит выполнению на практике. Безусловно, вы не можете одновременно работать над всеми проектами и удов-

летворять все запросы внутри компании. Составление плана следует начинать с определения перечня базовых служб вашей компании, которые будут работать с большими данными. План действий должен предусматривать использование существующих служб обработки данных, а также описывать разумные и достижимые промежуточные результаты. Если вы будете слишком амбициозны, то не сумеете продемонстрировать руководству конкретные результаты выполнения плана. Не нужно планировать на десять лет вперед. Начните с плана, рассчитанного на один-два года, с четко определенными целями и результатами. План действий должен включать в себя как технические, так и бизнес-цели.

3. Определите ваши данные

Не существует компаний, которые жалуются на недостаток данных, — напротив, они буквально утопают в информации. Проблема заключается в том, что компании часто не знают, как извлекать из этих данных пользу, — прогнозировать будущее, выполнять важные бизнес-процессы или просто корректировать свои представления о текущей ситуации. Стратегия и план использования больших данных должны быть направлены на создание новых способов их обработки, повышающих точность прогнозирования бизнес-результатов. И тем не менее, не стоит торопить события. Для начала следует определить, какие данные вам необходимы. Вам нужно знать, какими данными вы уже располагаете, где они находятся, кто владеет и управляет ими, а также как они используются на текущий момент. На какие сторонние источники данных опирается ваша компания? В рамках этого исследования вы получите много полезной информации. Например, это позволит узнать, сколько источников данных вы используете и в какой степени они дублируют друг друга. Определение необходимых данных выявит пробелы в ваших знаниях об этих источниках. Возможно, вы располагаете большим количеством повторяющихся данных об одной сфере бизнеса, а о другой почти не имеете данных. Возможно, точность сторонних данных, которые вы используете, неудовлетворительна. Уделите процессу определения необходимых данных достаточно времени. Это закладывает основу для планирования стратегии работы с большими данными и ее последующего выполнения.

4. Определите, какими данными вы не располагаете

Теперь, когда вы знаете, какие данные доступны, следует определить, каких данных не хватает. Воспользуйтесь возможностями созданной рабочей группы: руководители бизнеса — лучший источник информации. Они как никто другой знают, что мешает им принимать более эффективные решения. Приступая к выявлению недостающей информации, старайтесь стимулировать людей к нестандартному мышлению. Например, спросите их, какие данные для поддержки бизнеса они хотели бы получить, если бы имели быстрый и бесплатный доступ к любой интересующей их информации? Разумеется, доступ к информации не бесплатен, но вы задаете этот вопрос, чтобы руководители компании задумались над тем, что им необходимо для развития бизнеса. Инновационные технологии работы с данными иногда позволяют воплощать в жизнь даже весьма экстравагантные идеи и замыслы.

5. Продумайте возможные варианты применения технологий

Теперь вы понимаете цели компании, знаете, какие данные у вас есть и каких данных не хватает. Как реализовать избранную стратегию? Вы должны знать о том, какие технологии существуют и как компания может с их помощью улучшить результаты своей деятельности. Проведите самостоятельную работу: изучите возможности инструментов работы с потоковыми данными и сложной обработки событий, таких как Hadoop, а также различные типы баз данных — пространственные, работающие в оперативной памяти и др. Вы должны ознакомиться с новыми инструментами и методами экосистемы больших данных. Важно, чтобы ваши коллеги обладали достаточно широким техническим кругозором для принятия взвешенных решений.

6. Планируйте мероприятия по безопасности использования больших данных

Несмотря на то что компании всегда называют обеспечение безопасности данных одной из приоритетных задач, зачастую они не готовы столкнуться с проблемами управления распределенными и сложными

данными. Аналитики не защищают большие данные на начальных этапах анализа, поскольку лишь небольшая часть этих данных будет использована для дальнейшего изучения. Тем не менее, когда аналитик выбирает данные, которые интегрируются в информационную среду компании, возникает необходимость их защиты от внутренних и внешних рисков. Часть этих данных станет той конфиденциальной информацией, которая должна быть скрыта от всех неавторизованных пользователей. Для того чтобы обеспечить эффективную защиту больших данных, нужно хорошо ее спланировать.

7. Планируйте внедрение стратегии управления данными

Управление информацией — это способность создать информационный ресурс, которому доверяют сотрудники, партнеры и клиенты компании. Ответственность за стратегию управления информацией лежит не только на IT-подразделениях, но и на бизнес-структурах компании. Эта стратегия должна включать в себя жесткие правила управления большими данными, в том числе их защиту в различных ситуациях в соответствии с требованиями законодательства и правилами регулирования. Сведения о состоянии здоровья людей должны храниться с соблюдением требований, предъявляемых к обработке персональных данных. На финансовых рынках действуют особые законодательные и регуляторные требования к управлению данными, которые компаниям необходимо неукоснительно выполнять. Сбор и анализ больших объемов информации в отсутствие необходимых мер для ее защиты может стать причиной серьезных проблем. Кроме того, сами источники данных могут быть частной собственностью. Это может накладывать ограничения на объемы и цели использования таких данных внутри организаций. Ответственность за корректное использование информации лежит в основе эффективной стратегии управления данными.

8. Планируйте оперативное управление данными

Полагаясь на то, что результаты анализа данных всегда правильны, можно легко попасть в ловушку. Руководители любят работать с цифрами и использовать их при принятии решений, однако некорректное управление данными может привести к опасным последствиям. Допустим, вы используете данные из пяти или шести различных источников. Вы хотите определить, каким клиентам компании будет наиболее интересен новый продукт, проанализировав 10–15 источников данных. Имеется ли у вас единый набор метаданных для всех этих источников? Если нет, то существует ли процедура проверки достоверности источника и его пригодности для использования? Компания, которая пользуется источниками данных, основанных на разных метаданных и допущениях, рискует пойти неверным путем. Будьте осмотрительны и позаботьтесь о том, чтобы собираемые данные давали компании возможность принимать максимально взвешенные и точные решения. Для этого вам необходимо понимать, как интегрировать новые источники данных с хранилищами архивных данных.

9. Осуществляйте постоянную проверку допущений

Вы постепенно убедитесь в том, что ваша компания может гораздо точнее прогнозировать будущее. Это становится возможным благодаря новым источникам данных и обработке большого объема информации. С помощью данных вы сможете в почти реальном времени принимать решения и определять наилучшие дальнейшие действия в отношении своих клиентов. Даже если вы выполняете все процессы для надлежащего управления данными и корректного определения метаданных, постоянно проверяйте результаты своих действий. Какие результаты вы получаете с помощью анализа? Кажутся ли они вам верными? Если вы получаете результаты, в правильности которых сомневаетесь, проверяйте их корректность. Чем точнее данные, которые вы анализируете, тем точнее и результаты. Тем не менее, иногда вам придется сталкиваться с неочевидными проблемами. Не полагайтесь на то, что ваши данные всегда верны! Проверяйте свои допущения и представления о бизнесе.

10. Изучите наилучшие практики и используйте проверенные методы работы

По мере развития рынка больших данных компании будут расширять свой опыт, осваивать эффективные методы работы с ними и достигать требуемых результатов. Осваивать передовые методы обработки данных можно несколькими разными способами. Для повышения результативности бизнеса обменивайтесь опытом с коллегами, изучающими варианты использования больших данных. Обратите внимание на поставщиков и системных интеграторов, которые разрабатывают и предлагают клиентам решения на основе опробованных методов работы. Всегда лучше учиться на чужих ошибках, чем повторять их. С развитием рынка больших данных вы получите доступ к многочисленным эффективным методам работы с ними. Это значительно повысит эффективность разработки и реализации вашей корпоративной стратегии.

Глава 24

Десять основных источников больших данных

В ЭТОЙ ГЛАВЕ:

Источники больших данных
Стандарты больших данных
Бесплатные ресурсы

Существует множество ресурсов, которые помогут вам освоиться в мире больших данных. Организации, разрабатывающие стандарты, устраняют проблемы эффективного совместного использования источников данных. Решения с открытыми исходными кодами позволяют легко экспериментировать с большими данными и изучать их возможности. Большим данным посвящено большое количество конференций и форумов. Разумеется, все поставщики решений для больших данных проводят исследования, выпускают обзоры, посвященные эффективным методам работы. Все эти материалы, как правило, доступны для ознакомления. В этой главе мы предложим вашему вниманию ряд полезных информационных ресурсов.

HurWitz & Associates

www.hurwitz.com

Авторы этой книги — партнеры и совладельцы компании Hurwitz & Associates. Мы будем рады помочь вам в решении вопросов, касающихся больших данных. Мы оказываем услуги по обучению и страте-

гическому планированию, ведем блоги и исследовательскую работу. Если вы являетесь поставщиком больших данных, то мы поможем вам лучше узнать потребности ваших клиентов и соответствующим образом позиционировать свою компанию и продукцию. Подписывайтесь на наши блоги и посещайте наш веб-сайт!

Организации по стандартизации

Для развития больших данных необходимы стандарты. Ряд организаций ведет активную работу по объединению поставщиков больших данных для совместной разработки этих стандартов. Вот некоторые из таких организаций.

Фонд открытых данных (The Open Data Foundation)

www.opendatafoundation.org

Фонд открытых данных (The Open Data Foundation, ODaF) — некоммерческая организация, содействующая внедрению международных стандартов метаданных и разработке инструментов с открытыми исходными кодами для обработки статистических данных. Организация занимается повышением качества метаданных в сфере экономики, финансов, здравоохранения, образования, труда, обществоведения, технологий, сельского хозяйства, строительства, охраны окружающей среды.

Альянс информационной безопасности облачных данных (Cloud Security Alliance)

https://cloudsecurityalliance.org/research/big-data

Альянс информационной безопасности облачных данных (Cloud Security Alliance) занимается внедрением эффективных методов обеспечения безопасности облачных вычислений и обучением в области использования облачных технологий для защиты в IT-сфере. Союз сформировал рабочую группу по большим данным (Big Data Working

Group, BDWG), целью которой является поиск масштабируемых методов защиты информации и решение проблем обеспечения конфиденциальности больших данных.

Национальный институт стандартов и технологии (National Institute of Standards and Technology, NIST)

www.nist.gov/itl/ssd/is/big-data.cfm

Национальный институт стандартов и технологии (National Institute of Standards and Technology) — американская государственная организация, разрабатывающая новые стандарты. Ею проделана большая работа по определению и описанию различных концепций — от облачных вычислений до больших данных. В марте 2012 года институт NIST запустил проект по большим данным, цель которого — обучение организаций использованию больших данных для научных исследований, охраны окружающей среды, развития биомедицины, образования и обеспечения национальной безопасности. NIST планирует создать секцию по большим данным совместно с Центром исследований производительности гибридных многоядерных вычислений (Center for Hybrid Multicore Productivity Research, CHMPR) Национального научного фонда (National Science Foundation, NSF). В ведении секции будут находиться следующие вопросы:

- стандартизация современных фундаментальных технологий сбора, хранения, управления, анализа и совместного использования больших данных;
- измерение точности и надежности методов, использующих эти технологии.

Фонд программного обеспечения Apache (Apache Software Foundation)

http://hadoop.apache.org

Фонд программного обеспечения Apache обеспечивает организационную, юридическую и финансовую поддержку широкого круга проектов по разработке ПО с открытыми исходными кодами. Он основан в 1999 году как некоммерческая организация, обеспечивающая

дальнейшее развитие проектов по лицензии Apache, создаваемых разработчиками-энтузиастами. Одним из ключевых проектов фонда является управление платформой Hadoop — библиотекой с открытым исходным кодом, которая обеспечивает стандартные методы кластерной обработки наборов больших данных.

OASIS

www.oasis-open.org

OASIS (Organization for the Advancement of Structured Information Standards, организация по развитию стандартов структурированной информации) является одной из ключевых организаций в области стандартизации. OASIS ведет некоммерческую деятельность по разработке стандартов больших данных. Этот процесс находится в начальной стадии, однако мы предполагаем, что OASIS продолжит работу в сфере стандартизации больших данных.

Сайты поставщиков

Все ключевые поставщики средств управления данными предлагают очень полезные интернет-ресурсы. Рекомендуем обратить внимание на следующих поставщиков:

Google: http://research.google.com/

Amazon: http://aws.amazon.com/big-data/

IBM: http://bigdatauniversity.com/and http://www-01.ibm.com/software/data/bigdata/

Oracle: http://www.oracle.com/us/technologies/big-data/index.html

Microsoft: http://research.microsoft.com/en-us/projects/bigdata/

Cloudera: http://www.cloudera.com/content/cloudera/en/why-cloudera/hadoop-and-big-data.html

SAS Institute (Институт SAS): http://www.sas.com/

Teradata: http://www.teradata.com/business-needs/Big-Data-Analytics/

Это далеко не полный список. Существуют сотни компаний, предоставляющих полезную информацию по большим данным. Рекомендуем ознакомиться с блогами ведущих сотрудников этих компаний.

Сайты для совместной работы в реальном времени

Многие вновь возникающие группы поддерживают совместную работу специалистов над большими данными. Ряд таких групп, заслуживающих внимания, имеется в социальной сети LinkedIn: http://www.linkedin.com/groups/Big-Data-Analytics-Strategy-FP-1814785?home=&gid=1814785&trk=anet_ug_hm. Рекомендуем также ознакомиться с форумами, спонсируемыми программой Apache Hadoop Initiative, они доступны по ссылке http://hadoop. apache.org/.

Конференции по вопросам использования больших данных

Существует большое количество конференций, целиком посвященных использованию больших данных или включающих секции на эту тему. Список конференций в сфере больших данных постоянно пополняется, и ниже перечислены лишь некоторые из них:

The Data Warehousing Institute (TDWI, Институт изучения вопросов хранения данных): спонсирует многие конференции, семинары и образовательные форумы, посвященные хранению и работе с большими данными.

Big Data Conference (Конференция по большим данным): обеспечивает доступ к аналитике и приложениям для работы с большими данными.

Форум по работе с большими данными в сфере розничной торговли (Big Data Retail Forum): конференция по вопросам анализа информации в реальном времени для компаний, работающих в сфере розничной торговли или производства потребительских товаров.

Hadoop World: спонсируется O'Reilly

O'Reilly Strata и StrataRx: http://strataconf.com.

StructureData: http://event.gigaom.com/structuredata.

Десять рекомендаций: что следует и чего не следует делать в работе с большими данными

В ЭТОЙ ГЛАВЕ:

Четкий начальный план
Взаимодействие с бизнесом
Интеграция больших данных в процессы создания и выполнения планов

Многие компании приступают к изучению больших данных, находясь на ранних стадиях развития и не обладая развитыми навыками работы с такими данными. Большинство компаний проводит экспериментальные проекты, чтобы определить, можно ли изменить процесс принятия решений, включив в него источники больших данных. При использовании больших данных легко совершить ошибки, наносящие вред корпоративной бизнес-стратегии. В этой главе приведены рекомендации о том, что следует и чего не следует делать в процессе освоения больших данных.

Привлекайте все структурные подразделения организации к работе над стратегией использования больших данных

Работа с большими данными — это не обособленный процесс. Задействовав огромные объемы данных, можно получить более глубокие знания о клиентах, процессах и событиях, чем при работе только с выборками данных. Правильное использование больших данных позволяет существенно повысить эффективность бизнес-стратегии. Некоторые компании все же анализируют данные, выходящие за пределы норм, а не отказываются от них как от недостоверных. В результате это дает возможность обнаруживать новые тенденции изменения запросов покупателей. Подразделения компании, которые с самого начала участвуют в разработке стратегии использования больших данных, способны извлечь из нее существенную выгоду.

Развивайте все модели предоставления больших данных

Если вы работаете с петабайтами данных, то естественно предположить, что единственным способом хранения этих объемов информации и управления ими является центр обработки данных. Тем не менее, технологии меняются: теперь для управления данными можно и нужно использовать облачные хранилища и облачные вычисления. Оцените потенциал использования облачных служб и определите, какие из них обеспечат вам необходимую эффективность.

Подумайте о включении ваших традиционных источников данных в общую стратегию больших данных

Многие компании, которые научились извлекать пользу из анализа больших данных, полагают возможным отказаться от традиционных хранилищ данных. Однако это не так. Крайне важно планировать совместное использование результатов анализа больших данных и возможностей вашего хранилища данных. Поскольку в хранилище дан-

ных находится информация о функционировании вашей компании, при принятии решений крайне важно сравнивать полученные с помощью больших данных результаты с целевыми показателями.

Планируйте использование корректных метаданных

Вполне вероятно, что результаты анализа большого набора данных будут целиком соответствовать некоторому шаблону. С помощью этих данных ваша организация может начать углубленное исследование какого-либо нового аспекта изучаемой проблемы. Имейте в виду, что эти данные могут поступать из непроверенных источников, например с сайта службы поддержки клиентов или из социальных сетей. Прежде чем использовать такие данные, необходимо сначала удостовериться в том, что вы работаете с согласованным набором метаданных, — и лишь затем вводить данные в корпоративную среду и анализировать их.

Используйте распределенное хранение и обработку данных

Не рассчитывайте на то, что при использовании больших данных вам будет достаточно единственного сервера для управления всей имеющейся информацией. Изучите методы распределенных вычислений (такие как Hadoop), позволяющих управлять данными, которые имеют различный объем, степень разнообразия и требуют различной скорости обработки.

Не полагайтесь на единственный подход к аналитике больших данных

Такие продукты, как Hadoop и MapReduce, весьма агрессивно продвигаются на рынке. Попытайтесь понять, позволяет ли функционал этих программ решать стоящие перед вами задачи. Существует множество технологий, которые могут оказаться важными в вашей работе, на-

пример текстовый и прогностический анализ среды потоковых данных и анализ геопространственной информации. Уделите время изучению разнообразных технологий и найдите то, что подходит именно вам. Экспериментируйте и исследуйте решения, которые ведут вас к успеху.

Не внедряйте большие данные, пока не будете полностью готовы к этому

Вы оценили весь спектр возможностей, которые открываются перед вашей компанией благодаря использованию больших данных. Большие данные позволят вам опередить конкурентов на новых перспективных рынках, однако не предвосхищайте события. Начните с пробных проектов и наберитесь опыта. Работайте с экспертами — это убережет вас от серьезных ошибок.

Не упускайте из виду вопросы интеграции данных

Невозможно эффективно использовать источники больших данных, которые изолированы друг от друга. Передовые технологии упрощают интеграцию результатов анализа больших данных с другими источниками данных. Позаботьтесь не только об анализе, но и об интеграции.

Не забывайте о необходимости безопасного управления данными

При внедрении анализа больших данных компании часто забывают обеспечивать тот же уровень безопасности и управляемости данных, что и в традиционных информационных средах. Когда объем анализируемой информации начинает измеряться петабайтами, вы обычно не сразу направляете усилия на то, чтобы замаскировать данные, составляющие тайну частной жизни. В вашем распоряжении оказывается подмножество данных, критически важных для выбора оптимального пути продвижения вашей компании на новый рынок. Вам необходимо сначала защитить эти данные, чтобы не поставить бизнес под угрозу.

Некоторые из этих данных становятся интеллектуальной собственностью вашей компании и должны быть защищены соответствующим образом. Возможно, вам также придется позаботиться о соблюдении требований, предъявляемых к обработке конфиденциальных данных. Все перечисленные аспекты обеспечения безопасности должны быть учтены в цикле работы с большими данными. Кроме того, некоторые данные могут поступать из сторонних источников, использование которых требует лицензирования. Удостоверьтесь в том, что вы имеете право пользоваться этими данными и не нарушаете требования законодательства.

Не пренебрегайте управлением производительностью

Современные технологии позволяют обрабатывать беспрецедентно большие объемы данных с недостижимой ранее скоростью. Получаемая в результате информация приносит компаниям огромную пользу, однако неэффективное управление большими данными может вызывать серьезные проблемы. По этой причине необходимо уделять должное внимание вопросам управления при планировании работы с большими данными.

Глоссарий

Абстракция — концепция упрощения предмета путем исключения подробностей и представления только важной информации. Абстракция представляет собой высокоуровневое описание предмета, в отличие от подробной характеристики его отдельных особенностей. Например, в облачной модели IaaS инфраструктура абстрагирована от пользователя.

Архитектура — в обработке информации — подход к проектированию, используемый при разработке программы или системы.

Архивирование — процесс копирования редко используемых или устаревших баз данных или файлов на более экономичный накопитель для хранения или последующего аудита. В качестве архивного накопителя может использоваться сетевой диск, оптический диск или ленточный накопитель. Компании используют для архивирования данных облачные среды.

Аудит — проверка эффективности выполнения задачи или группы задач, а также процесса управления ими и их документирования. С помощью аудита организации также удостоверяются в том, что их данные надежно защищены и соответствуют требованиям регулирующих органов.

Аутентификация — процесс проверки подлинности личности или компьютерного процесса.

База данных — компьютерная система для организованного надежного хранения больших объемов информации. Большинство баз данных обеспечивает пользователям удобный доступ к данным и ценные поисковые возможности.

База данных в памяти — база данных, в которой управление и обработка информации происходят в оперативной памяти, а не на диске.

База пространственных данных — база, оптимизированная для работы с данными о пространственном положении объектов.

Бизнес-процесс — совокупность взаимосвязанных или взаимодействующих видов деятельности для достижения целей организации.

Бизнес-правила — реальные коммерческие ограничения или операции, которые могут встраиваться в средства управления сервисами или бизнес-приложения.

Биометрия — использование компьютером уникальных физических особенностей человека (например, отпечатка пальца или особенностей голоса) для его идентификации.

Большие данные — способность управлять большими объемами разнородных данных со скоростью, достаточной для их анализа и реагирования в реальном времени. Как правило, большие данные классифицируются по трем характеристикам: объему (количеству данных), скорости (быстроте обработки данных) и вариативности (количеству различных типов данных).

Брокер — в информатике — программа, которая получает запросы от одного программного компонента или уровня и преобразует их в форму, понятную другому компоненту или уровню.

Веб-хостинг Linux — подавляющее большинство веб-сайтов работает на операционной системе Linux под управлением службы веб-хостинга с использованием программного стека LAMP (Linux, Apache, MySQL, PHP).

Веб-сервис — программный компонент, включающий в себя интерфейс, который состоит из определения WSDL, схемы XML и WS-Policy. В совокупности компоненты можно назвать сервисным контрактом или API. См. также: API, WSDL (Web Standard Definition Language), WS (Web Standard) и XML (eXtensible Markup Language).

Виртуализация, виртуальная память — хранение активных областей памяти на диске с целью увеличить объем памяти, которая воспринимается пользователем как доступная. В виртуальной среде на одном компьютере выполняется ПО, эмулирующее другой компьютер. Такую эмуляцию часто называют виртуализацией. См. также: Эмуляция.

Витрины данных — часть хранилища данных, предназначенная для определенного набора бизнес-информации.

Время отклика — продолжительность периода с момента отправки транзакции пользователем или приложением до момента получения ими результатов обработки транзакции.

Выделение ресурса — предоставление пользователям и программам доступа к ресурсам. Система выделения ресурсов обеспечивает доступ пользователей к приложениям и доступ приложений к ресурсам серверов.

Вычисления как коммунальная услуга — измеряемый сервис, который предоставляется по модели коммунальных услуг: пользователь платит

за использование определенного отмеренного объема компонента или актива.

Гранулярность — важная концепция разработки ПО (в особенности компонентов), которая характеризует объем функциональных возможностей, предоставляемых компонентом сервиса. Гранулярность варьируется от мелкой до крупной: один компонент может выполнять единственное простое действие, например вычислять квадратный корень, а другой — детально реализовывать сложный бизнес-процесс или правило. Считается, что первый компонент имеет мелкую гранулярность, а второй — крупную. Часто разработчики создают бизнес-сервисы, объединяя сервисы с мелкой гранулярностью в сервисы с более крупной гранулярностью.

Гибридное облако — вычислительная среда, которая обеспечивает согласованное использование общедоступных и частных облаков, а также ресурсов центров обработки данных.

Гипервизор — оборудование, обеспечивающее одновременную работу нескольких операционных систем на одном компьютере. Гипервизор располагается на нижних уровнях аппаратной среды и динамически распределяет ресурсы, используя тонкий слой программного кода. Благодаря гипервизору у пользователя создается впечатление, что операционной системе доступны все имеющиеся ресурсы.

Доступ к данным — См.: Управление доступом.

Жизненный цикл приложения — процесс поддержания согласованности и предсказуемости кода в процессе его изменения согласно бизнес-требованиям.

Журнал аудита — последовательность событий, зарегистрированная в компьютере или системе документооборота. Как правило, аудит позволяет выявить действия по созданию и изменению любого элемента системы, исполнителей этих действий, а также, по возможности, их причины.

Задержка — длительность интервала перед выполнением сервиса в среде. Некоторые приложения требуют малой задержки и работают в почти реальном времени, в то время как для других приложений величина задержки менее важна.

Интеллектуальный анализ данных — процесс исследования и анализа больших объемов данных с целью выявления закономерностей.

Интеграция информации — процесс связывания источников данных в различных подразделениях или региональных филиалах организации для более надежного, согласованного и качественного накопления информации при помощи программных средств.

Инфраструктура — базовые системы, обеспечивающие нормальную работу любой структуры — от государства до ИТ-отдела. Инфраструктура системы жизнеобеспечения людей включает в себя транспортную сеть,

сети электро- и водоснабжения. В сфере информационных технологий под инфраструктурой понимается базовое компьютерное оборудование, сети, операционные системы и ПО, обеспечивающее работу приложений.

Изолированное приложение — приложение с единственной узконаправленной ключевой функцией (например, управление человеческими или складскими ресурсами), не предназначенное для использования другими приложениями.

Инфраструктурные сервисы — сервисы, предоставляемые инфраструктурой. В сфере информационных технологий к таким сервисам относят все ПО, обеспечивающее коммуникацию между устройствами.

Каталог — этот термин используется как в компьютерных технологиях, так и в телефонии и обозначает структурированную схему устройств, файлов или людей.

Каталог сервисов — перечень ИТ-сервисов, предоставляемых на предприятии, который содержит описания сервисов, их владельцев и права доступа.

Качество данных — включает в себя такие характеристики данных, как согласованность, точность, надежность, полнота, своевременность, обоснованность и валидность. ПО, обеспечивающее качество данных, гарантирует единообразное представление данных в различных хранилищах и системах, а также повышает надежность данных в рамках предприятия.

Компонент — элемент ПО, который может использоваться в составе более крупных систем. Компоненты могут являться частью бизнес-приложений, доступ к которым осуществляется с использованием стандартов и технологий, обеспечивающих работу веб-сервисов: например, WSDL, SOAP и XML. См. также: Веб-сервис.

Коммерческая услуга — отдельная функция или операция, которая непосредственно используется для получения прибыли в бизнесе.

Контейнер — в информатике — структура данных или объект, используемый для организованного управления коллекциями других объектов.

Конфигурация — полное описание физического и функционального взаимодействия составных элементов программного продукта.

Критически важное приложение — приложение, без которого функционирование бизнеса невозможно в любой момент времени.

Кэширование — эффективный метод хранения данных в памяти, ускоряющий доступ к ним при повторных запросах.

Масштабируемость — в контексте оборудования — способность обеспечивать как высокую, так и низкую вычислительную мощность при неизменной архитектуре. Этот термин также применяется к программным продуктам (например, к базам данных) и означает сохранение величи-

ны производительности на единицу мощности при наращивании аппаратных ресурсов.

Метаданные — определения, соответствия и другие характеристики, описывающие поиск, доступ и использование данных и ПО компании.

Модель «объект-отношение» (ER-модель) — способ управления данными, при котором отношения между ними отображаются графически. Он позволяет разработчикам создавать новые отношения между источниками данных без сложного программирования.

Моделирование бизнес-процесса — метод представления бизнес-операций в виде кодифицированных описаний, на основе которых создаются программные продукты.

Мультитенантность — этот термин описывает ситуацию, в которой на серверах поставщика SaaS работает один экземпляр приложения, обслуживающий разные организации (тенанты), при этом их данные отделены друг от друга. В мультитенантной архитектуре данные и настройки приложения секционируются, что позволяет каждому клиенту работать с индивидуальным виртуальным экземпляром приложения.

Мэшап — программа (часто установленная на веб-странице), которая собирает контент из нескольких источников, например из Google Maps и риелторской базы данных.

Неструктурированные данные — данные, не имеющие определенного формата, например текст, видео, изображения и др. См. также: Структурированные данные.

Облачные вычисления — вычислительная модель, предоставляющая организациям такие ИТ-ресурсы, как серверы, промежуточное ПО и приложения в виде сервисов, доступных в режиме самообслуживания.

Обработка в реальном времени — вид обработки данных, при котором компьютерная система одновременно принимает, обновляет данные и незамедлительно возвращает результаты, влияющие на источник данных.

Обработчик событий в реальном времени — класс приложений, которые требуют своевременной реакции на события, происходящие в реальном мире. Типичными примерами таких приложений являются системы биржевой торговли и радиочастотной идентификации. См. также: RFID.

Общедоступное облако — ресурс, который доступен любому пользователю бесплатно или с оплатой за транзакцию. Общедоступное облако не обеспечивает высокого уровня безопасности и соглашений об уровне обслуживания.

Очистка данных — программное обеспечение, обнаруживающее потенциальные проблемы, связанные с качеством данных. Если один и тот же человек многократно упоминается в базе данных клиентов из-за того,

что его имя написано по-разному, то ПО для очистки данных вносит коррективы, помогающие стандартизировать данные.

Операционная совместимость — способность продукта взаимодействовать с большим количеством других продуктов. Как правило, этот термин применяется по отношению к ПО.

Операционная аналитика — интеграция аналитики в бизнес-процесс.

Отказоустойчивость — способность системы продолжать обслуживание при отказе одного или нескольких компонентов.

Открытый исходный код — направление разработки ПО, при котором пользователям предоставляется доступ к исходным кодам программ, а также возможность изучать и дорабатывать эти программы. Все изменения, вносимые в исходный код, становятся доступными всему сообществу.

Пакетная обработка — неинтерактивный процесс, который выполняется в очереди, как правило, в периоды минимальной загрузки системы, и обеспечивает эффективную последовательную обработку информации. Первые компьютеры обрабатывали данные только в пакетном режиме.

Передовая практика — эффективный способ выполнения какого-либо действия, от написания программного кода до управления IT-инфраструктурой.

Перемещаемые данные — данные, передаваемые по сети или перемещаемые в память для обработки в реальном времени.

Позднее связывание — создание связей между приложениями в моменты, когда впервые появляется необходимость в этих связях. Позднее связывание обеспечивает гибкость, облегчающую внесение изменений в ПО, однако увеличивает время обработки.

ПО как услуга (Software as a Service, SaaS) — предоставление компьютерных приложений через Интернет.

Потоковые данные — аналитическая вычислительная платформа, нацеленная на повышение скорости обработки данных. Данные непрерывно анализируются, преобразуются в памяти, а затем сохраняются на диске. Такая платформа позволяет анализировать большие объемы данных в реальном времени.

Прогностическая аналитика — решение для статистической обработки или интеллектуального анализа данных, которое включает в себя алгоритмы и методы, работающие как со структурированными, так и с неструктурированными данными (совместно или раздельно) для решения задач прогнозирования. С помощью такого решения можно выполнять прогнозирование, оптимизацию, предсказание, моделирование и многие другие операции.

Протокол — набор правил, с помощью которых компьютеры устанавливают связь и взаимодействуют друг с другом.

Процесс — высокоуровневая глобальная структура, которая определяет порядок принятия и исполнения решений внутри компании или организации. См. также: Рабочий процесс.

Платформа — инфраструктура, поддерживающая разработку программных продуктов и управление ими.

Преобразование данных — процесс изменения формата данных для использования их в других приложениях.

Промежуточное ПО — многоцелевое ПО, которое в распределенной вычислительной среде располагается в слое между операционной системой и приложением.

Пропускная способность — в технике — диапазон частот, в котором устройство может передавать или принимать сигналы. Этот термин также обозначает максимальную скорость передачи данных по каналу, измеряемую в битах в секунду.

Пропускная способность — скорость выполнения транзакций в системе.

Профилирование данных — метод выявления содержания и структуры данных, а также взаимоотношений между ее элементами. Этот процесс также позволяет проверять соответствие данных техническим требованиям и бизнес-правилам.

Пул ресурсов — набор вычислительных служб, служб хранения и обработки данных, совместно работающих в гибридных средах.

Рабочий процесс — последовательность задач, которые необходимо выполнять для осуществления бизнес-процесса. См. также: Процесс.

Раннее связывание — задание связей между приложениями, выполняемое перед обработкой данных с целью ее ускорения. Раннее связывание также ограничивает гибкость. См. также: Позднее связывание.

Распределенные вычисления — выполнение и управление выполнением алгоритмов на множестве узлов вычислительной среды.

Реестр — единый источник всех метаданных, необходимых для получения доступа к веб-сервису или программному компоненту.

Резервное копирование — утилита, копирующая базы данных, файлы или подмножества баз данных и файлов на какой-либо носитель. Копия может использоваться для восстановления данных при отказе системы.

Репозиторий — база данных для ПО и компонентов с особыми возможностями контроля изменений и управления конфигурациями (другими словами, место, в котором хранятся «правильные» данные).

Репозиторий метаданных — контейнер согласованных определений бизнес-данных и правил, которые связывают данные с их реальным физическим расположением в системе.

Руководство — способность обеспечивать соблюдение правил и предписаний, установленных корпоративными стандартами и регулирующими органами. Руководство связано с соблюдением требований и обеспечением безопасности в вычислительных средах.

Связывание — установление связей, обеспечивающих взаимодействие между программными компонентами.

Секционирование оборудования — разделение элементов физического сервера на изолированные части, каждая из которых может обеспечивать работу операционной системы или приложения.

Семантика — в информатике — значение данных (в отличие от правил их форматирования — синтаксиса).

Сервис — целенаправленное действие, которое выполняется с определенной целью. Как правило, сервис состоит из группы сервисов-компонентов, некоторые из которых, в свою очередь, тоже состоят из компонентов. Сервисы всегда что-либо преобразуют, а их работа завершается определенным результатом.

Сетевые распределенные вычисления — развитие технологии распределенных вычислений, при котором большое количество подключенных к сети компьютеров (часто географически распределенных и, как правило, обладающих разнородными архитектурами и возможностями) объединяются для решения общей задачи. Сетевые распределенные вычисления могут использоваться вместо виртуализации в условиях, требующих обработки в реальном времени, когда задержки недопустимы.

Сеть — соединение компьютерных систем (узлов) между собой с помощью каналов связи и соответствующего ПО.

Система управления реляционными базами данных, СУРБД (relational database management system, RDBMS) — СУБД, организующая данные в таблицах с заданной структурой.

Система управления базами данных, СУБД — ПО, управляющее хранением, доступом, удалением, защитой и обеспечением целостности первично структурированных данных, хранимых в базе.

Система управления контентом — система, обеспечивающая методы и инструменты сбора, управления, хранения и доставки контента и документов, относящихся к организационным процессам. Эта система включает в себя средства управления документооборотом, записями, изображениями, рабочими процессами, веб-контентом и совместной работой.

Скриптовый язык — интерпретируемый язык программирования, имеющий доступ ко всем ресурсам операционной системы или большей их части. Самые известные скриптовые языки — Perl, Python, Ruby и JavaScript. Программировать на таком языке относительно легко, но создаваемые на нем программы, как правило, работают медленнее, чем программы, написанные на компилируемых языках, таких как C и C++.

Слабое связывание — подход к созданию распределенных приложений, в котором компоненты взаимодействуют друг с другом, передавая данные и запросы другим компонентам стандартизованными методами, что минимизирует зависимость между компонентами. Главные особен-

ности этого подхода — простота и автономность. Каждый компонент предоставляет другим компонентам небольшой набор простых служб.

Служба технической поддержки — единый контактный центр, в который корпоративные пользователи и клиенты могут обращаться по любым вопросам, связанным с использованием ИТ-систем (или, в некоторых случаях, с ИТ-службой).

Сохранность — гарантия того, что данные, хранящиеся в базе, не будут изменяться без соответствующего разрешения и будут доступны в течение всего периода их необходимости для бизнеса.

Стандарты — базовый набор воспроизводимых передовых практик и протоколов, общепринятых в бизнес-сообществе или индустрии. Как правило, поставщики, группы пользователей в различных сферах деятельности и конечные пользователи совместно разрабатывают стандарты, основанные на широких знаниях большого числа экспертов. Организации могут использовать эти стандарты как единый фундамент для собственных инноваций.

Столбцовая база данных — база данных, в которой данные хранятся в столбцах — в отличие от реляционных баз данных, где данные хранятся в строках.

Структурированные данные — данные, имеющие определенную длину и формат. К структурированным данным относятся числа, даты, группы слов и чисел, называемые строками (например, имя, адрес и др.).

Текстовая аналитика — процесс анализа неструктурированного текста, извлечения из него важной информации и ее преобразования в структурированный вид, в котором ее можно использовать различными способами.

Транзакция — действие компьютера, соответствующее бизнес-событию, например списанию средств со счета. Всякая начатая транзакция должна быть завершена или полностью отменена.

Углубленная аналитика — алгоритмы комплексного анализа структурированных и неструктурированных данных. Они включают в себя сложные статистические модели, машинное обучение, нейронные сети, текстовую аналитику и другие продвинутые методы интеллектуального анализа данных. К углубленной аналитике не относятся запросы к базам данных, формирование отчетов и построение кубов OLAP.

Управление удостоверениями — отслеживание удостоверения пользователя или актива при использовании системы или множества систем.

Управление сервисами — мониторинг и оптимизация сервиса для достижения результатов, ценных для клиента и заинтересованных сторон.

Управление доступом — процесс, который определяет, что, когда и как могут делать пользователи.

Управление изменениями — процесс управления корректировкой текущей деятельности организации и бизнес-процессов. Управление из-

менениями играет важнейшую роль в управлении информационными системами и внедрении новых процессов разработки. Для того чтобы все компоненты ПО работали корректно, их необходимо синхронизировать.

Управление конфигурациями — как правило, конфигурационная информация хранится в базе данных, позволяющей изменять эту информацию и управлять ею при необходимости.

Управление активами — программное обеспечение, позволяющее организациям регистрировать всю информацию об их оборудовании и ПО. Большинство таких приложений осуществляет сбор информации о затратах, лицензиях и т.д. Эта информация хранится в базе данных, предназначенной для управления конфигурациями. См. также: CMDB.

Устаревшее приложение — любое приложение, созданное как минимум несколько лет назад. К устаревшим относятся приложения, которые невозможно легко удалить или заменить. Если отдельные фрагменты кода преобразовать в бизнес-сервисы с новыми стандартизованными интерфейсами, то устаревшие приложения смогут по-прежнему приносить пользу.

Федерация — сочетание разнородных объектов, обеспечивающее их согласованную работу (федерация государств, данных или учетных записей), а также применение всех необходимых правил.

Федерация данных — доступ к данным из разных хранилищ с использованием единых правил и определений, позволяющих работать со всеми хранилищами как с единым ресурсом.

Хранимые данные — данные, помещенные в хранилище и не используемые в реальном времени.

Хранилище данных — большой массив архивных данных организации, используемый главным образом для их интеллектуального анализа. Хранилище также обеспечивает учет данных.

Центр компетенции — группа ключевых сотрудников из всех областей и структур бизнеса, сконцентрированных на выработке передовых практик. Центр компетенции позволяет рабочим и проектным группам вести совместную деятельность внутри компании. Этот центр также является движущей силой изменений, поскольку передает непрерывно накапливающиеся знания бизнес-подразделениям компании.

Частное облако — в отличие от общедоступного облака частное облако представляет собой множество внутрикорпоративных вычислительных ресурсов, которые используются только для нужд компании, но управляются аналогично облаку.

Черный ящик — компонент или устройство со входом и выходом, внутренние механизмы работы которого не должны быть известны или доступны пользователю.

Шина — технология соединения нескольких компонентов, которая позволяет им взаимодействовать друг с другом. По сути, шина представляет собой способ связи. Шина может быть программной (например, сервисная шина предприятия) или аппаратной (шина памяти).

Эластичность — способность вычислительного ресурса увеличиваться или уменьшаться в реальном времени в зависимости от потребностей.

Эмуляция — воспроизведение функциональных возможностей одной компьютерной системы аппаратным, программным или аппаратно-программным способом в другой системе. Поведение второй системы с высокой точностью копирует работу первой системы. См. также: Виртуализация.

Язык разметки — способ кодирования информации, в котором в обычный текст включаются специальные теги, часто помещаемые в угловые скобки (<и>). Для стандартизации обмена информацией между различными компьютерными системами и службами используются специальные языки разметки на базе XML. См. также: XML.

ACID — сокращение от: atomicity (атомарность), consistency (согласованность), isolation (изоляция) и durability (устойчивость) — основных требований для гарантированной обработки транзакции.

API (application programming interface, интерфейс программирования приложений) — протокол, с помощью которого компьютерные программы могут пользоваться функциями и данными других программ.

Bigtable — система распределенного хранения данных, разработанная компанией Google и предназначенная для управления масштабируемыми структурированными данными. Данные организованы в таблицах в виде строк и столбцов. В отличие от традиционных реляционных баз данных Bigtable представляет собой разреженную распределенную постоянную многомерную сортированную карту, предназначенную для хранения больших объемов данных на стандартных серверах.

BPaaS — См.: Бизнес-процесс как услуга (Business Process as a Service).

BPEL (Business Process Execution Language, язык выполнения бизнес-процессов) — компьютерный язык на основе WSDL (Web Services Description Language — язык описания веб-сервисов в формате XML), предназначенный для создания коммерческих программных продуктов. См. также: XML.

BPM (business process management, управление бизнес-процессами) — технология и методология ручного и автоматического управления различными видами деятельности для формирования бизнес-функций.

Business Process as a Service (BPaaS, бизнес-процесс как услуга) — бизнес-процесс целиком предоставляется в виде услуги, то есть фактически как программный интерфейс (например, служба доставки посылок).

CMDB (configuration management database, база данных управления конфигурациями) — репозиторий данных для управления сервисами.

COBIT (Control Objectives for Information and Related Technology, задачи управления для информационных и смежных технологий) — В ИТ-сфере — набор стандартов для управления техническими и бизнес-рисками.

Complex Event Processing (CEP, обработка сложных событий) — метод отслеживания, анализа и обработки данных при возникновении определенных событий. Эта информация подвергается дальнейшей обработке и управляется с использованием бизнес-правил и процессов.

CRM (customer relationship management, управление взаимоотношениями с клиентами) — программное обеспечение для управления работой отделов продаж и поддержки клиентов.

ELT (extract, load, transform — извлечение, загрузка, преобразование) — инструменты поиска и загрузки данных в бизнес-приложение для дальнейшего преобразования. Эти инструменты схожи с ETL (см. соответствующую статью), однако относятся к процессам интеграции данных.

ERP (enterprise resource planning, планирование ресурсов предприятия) — пакет бизнес-приложений, в котором бизнес-правила, процессы и средства управления данными объединены в интегрированную среду, поддерживающую функционирование бизнеса.

Enterprise Service Bus (ESB, сервисная шина предприятия) — пакет промежуточных служб, обеспечивающих надежную и предсказуемую коммуникацию между бизнес-сервисами.

ETL (extract, transform, and load, извлечение, преобразование и загрузка) — инструменты, предназначенные для поиска данных в хранилище и доступа к ним (извлечение), изменения их структуры или формата для применения в бизнес-приложении (преобразование) и передачи данных в бизнес-приложение (загрузка).

Hadoop — программная платформа Apache на базе MapReduce и Bigtable. Hadoop обеспечивает работу приложений на базе MapReduce в больших кластерах со стандартным оборудованием. Платформа Hadoop организует параллельную обработку данных на множестве вычислительных узлов, благодаря чему сокращается длительность вычислений и задержки. Hadoop включает в себя два ключевых компонента: масштабируемую распределенную файловую систему, в которой могут храниться петабайты данных, и масштабируемый механизм MapReduce, осуществляющий вычисления в пакетном режиме.

Hadoop Distributed File System (HDFS, распределенная файловая система Hadoop) — гибкая и надежная кластерная технология управления файлами в среде больших данных. HDFS является для файлов не «пунктом назначения», а скорее «сервисом», обеспечивающим уникальный набор возможностей для быстрой обработки больших объемов данных.

IaaS — См.: Инфраструктура как услуга.

Infrastructure as a Service (IaaS, инфраструктура как услуга) — инфраструктура, включающая в себя интерфейс управления и связанное с ним ПО, которая предоставляется компании как облачный сервис.

ISO (International Organization for Standardization, Международная организация по стандартизации) — организация, разработавшая более 17 000 международных стандартов, в том числе для управления ИТ-сервисами и корпоративного управления информационными технологиями.

ITIL (Information Technology Infrastructure, библиотека инфраструктуры информационных технологий) — платформа и набор стандартов ИТ-руководства, основанных на передовых практиках.

LAMP (Linux, Apache, MySQL, PHP, Perl или Python) — набирающие популярность средства создания веб-приложений с открытым исходным кодом. LAMP представляет собой пакет ПО, в состав которого входят операционная система Linux, веб-сервер Apache, база данных MySQL и скриптовый язык PHP, Perl или Python.

Linux — операционная система с открытым исходным кодом, основанная на ОС UNIX и схожая с ней. В облачных вычислениях ОС Linux занимает лидирующие позиции, поскольку поддерживается многими поставщиками.

MapReduce — разработанная компанией Google технология эффективного выполнения наборов функций, обрабатывающих большие объемы данных в пакетном режиме. Компонент «map» (карта) распределяет задачи между большим количеством систем, балансируя их нагрузки, а также управляет восстановлением после сбоев. После завершения распределенных вычислений функция «reduce» (сокращение) собирает все элементы воедино и формирует результат.

MOM (Message Oriented Middleware, промежуточное ПО для передачи сообщений) — технология, предшествующая сервисной шине предприятия. См. также: Enterprise Service Bus (ESB, сервисная шина предприятия), набор пакетов промежуточных сервисов.

MySQL — вариант SQL с открытым исходным кодом.

NoSQL (not only SQL, не только SQL) — набор технологий, формирующий широкий круг нереляционных СУБД. Особенность этих СУБД заключается в том, что они не используют SQL в качестве основного языка запросов. Кроме того, эти СУБД работают с распределенными хранилищами данных.

Object-oriented database management system (OODBMS, объектно-ориентированная система управления базами данных, ООСУБД) — СУБД, в которой данные хранятся в виде объектов, тесно связанных с приложением.

P2P (peer-to-peer, одноранговая сеть) — сеть, в которой узлы непосредственно обмениваются данными друг с другом без участия центрального сервера.

PaaS — См.: Платформа как услуга.

Platform as a Service (PaaS, платформа как услуга) — облачный сервис, при котором абстрагируются вычислительные службы (включая операционное ПО, а также жизненный цикл разработки, внедрения и управления). Такой сервис работает поверх IaaS.

PostgreSQL — самая распространенная реляционная СУБД с открытым исходным кодом.

REST (Representational State Transfer, передача репрезентативного состояния) — самый распространенный механизм подключения одного веб-источника (сервера) к другому веб-источнику (клиенту), разработанный специально для Интернета. RESTful API обеспечивает стандартизированный способ установления временных отношений между веб-ресурсами, также называемый слабым связыванием.

RFID (radio frequency identification, радиочастотная идентификация) — технология, которая использует небольшие и недорогие микросхемы, размещаемые на упаковках продукции (и даже на животных) и передающие на небольшое расстояние уникальный идентификационный номер, который принимается специальным приемопередатчиком радиосигналов.

RPC (remote procedure call, удаленный вызов процедур) — способ, с помощью которого компьютерная программа может запустить подпрограмму на другом компьютере.

SaaS — См.: ПО как услуга.

SAML (Security Assertion Markup Language, язык разметки с защитой данных) — стандартная платформа обмена аутентификационной и авторизационной информацией (т.е. учетными данными) в формате XML — т.н. подтверждениями.

SAN (storage-area network, сеть хранения данных) — высокоскоростная сеть взаимосвязанных устройств хранения данных: серверов, оптических дисков или других накопителей. Различие между SAN и NAS (Network Attached Storage, сетевым хранилищем) заключается в том, что SAN работает быстрее, чем NAS, а NAS легче устанавливается и обеспечивает файловую систему.

SLA (service-level agreement, соглашение об уровне обслуживания) — документ, описывающий соглашения о качестве и своевременности предоставляемых услуг и ресурсов между пользователем и поставщиком.

SOA (service-oriented architecture, сервисно-ориентированная архитектура) — подход к созданию приложений, реализующий бизнес-процессы и сервисы в виде набора слабосвязанных закрытых компонентов, кото-

рые обеспечивают четко определенный уровень обслуживания путем скоординированной совместной работы.

SOAP (Simple Object Access Protocol, простой протокол доступа к объектам) — спецификация протокола обмена данными. Этот протокол наряду с технологией REST используется в облачном хранилище Amazon для хранения и извлечения данных. См. также: REST.

SQL (structured query language, язык структурированных запросов) — самый распространенный компьютерный язык для доступа к базам данных и работы с ними.

SSL (Secure Sockets Layer, уровень защищенных сокетов) — распространенный метод создания защищенных интернет-соединений, впервые предложенный компанией Netscape.

TLS (Transport Layer Security, безопасность транспортного уровня) — новое название технологии SSL. См. также: SSL.

TQM (Total Quality Management, глобальное управление качеством) — распространенная программа повышения качества.

WS (Web Standard, веб-стандарт) — свод правил, позволяющих описывать возможности, требования и характеристики программных компонентов системы веб-сервисов.

WSDL (Web Service Definition Language, язык описания веб-стандартов) — формат описания веб-сервисов на базе XML (eXtensible Markup Language, расширяемый язык разметки).

XML (eXtensible Markup Language, расширяемый язык разметки) — представление данных в виде текстовых файлов, ставшее общепринятым языком сервисной архитектуры (SOA). В XML, как и в HTML, данные разделяются тегами, заключенными в угловые скобки (<и>), однако в XML теги имеют гораздо больше значений. См. также: SOA.

XML Schema — язык, позволяющий определять и описывать структуру документов XML.

XSD (XML Schema Definition, определение XML Schema) — описание допустимого содержимого документа XML.

XSLT (eXtensible Stylesheet Language Transformation, расширяемый язык стилей для преобразований) — компьютерный язык на основе XML, который определяет, каким образом преобразовать один документ XML в другой. См. также: XML.

Указатель

A

Абстракция
  и виртуализация *93—94*
  определение *354*
Альянс информационной безопасности облачных данных (Cloud Security Alliance) *344—345*
Анализ больших данных, подходы *217*
  интерактивный *219*
  персонализированные приложения *218—220*
  полуперсонализированные приложения *220—222*
  Google Prediction API *219—220*
  R-среда *218—219*
Анализ и извлечение больших данных *202*
  виды классификации *203*
  извлеченная информация *202—203*
  ключевые слова *202*
  концепции *202*
  отношения *202*
  события *202*
  термины *202*
  факты *202*
  эмоциональная окраска *203*
  NLP (Natural Language Processing, обработка естественного языка) *200—202*
Анализ, выполнение *301—302*
Аналитика
  аналитика социальных медиа *205—206*
  базовая *183—184*
  больших данных *35—36*
  визуализация *36*, *79—80*
  выбор оптимального следующего действия *324—328*
  и углубленная аналитика *80*
  монетизированная *183*, *188*
  новые аспекты аналитики данных *304—305*
  операционная *183*, *188*, *359*
  отчетность *36*
  отчеты и панели мониторинга *79*
  потребность *322*
  предотвращение мошенничества с помощью *328—330*
  прогностическая *359*
  текстов *197—199*, *322—324*
  типы *182—183*
  углубленная аналитика *362*
  углубленная *184—187*
  хранилища данных *35*

Аналитика в организации эмитирующей кредитные карты *330*
Аналитика международного банка *327*
Аналитика страховой компании *329—330*
Аналитическая платформа
интеграция с развернутыми облаками *223*
использование существующих алгоритмов *223*
использование существующих данных *223*
обеспечение дешевого хранилища *223*
ориентация на действие *215*
поддержка NoSQL *223*
поддержка типов данных *223*
преодоление низкой задержки *223*
Open Chorus *225—226*
Аналитические решения
IBM *193—194*
Oracle *194*
Pentaho *194*
SAS *194*
Tableau *194*
Аналитические решения SAS *194*, *211*
Аналитические решения Tableau *194*
Архивирование, определение *354*
Архитектура больших данных
избыточная физическая инфраструктура *30—31*
интерфейсы *30*
инфраструктура безопасности *31*
источники операционных данных *31—32*
потоки *30*
производительность *35*
NoSQL (не только SQL) *31—32*
Архитектура больших данных
диаграмма *67*
избыточная физическая инфраструктура *67—71*
инфраструктура безопасности *71—72*
обзор *66*
операционные базы данных *74—76*
распределенная файловая система *77*
службы и инструменты организации данных *76—77*
службы координации *77*
службы рабочих процессов *77*
службы сериализации *77*
Уровень 0 *67—71*
Уровень 1 *71—74*
Уровень 2 *74—76*
Уровень 3 *76—77*
Уровень 4 *77—79*
хранилища аналитических данных *77—79*
ETL (extract/transform/load, извлечение/преобразование/загрузка) *77*
Архитектура управления *29*
Архитектура, определение *354*
Архитектурная основа
избыточная физическая инфраструктура *30—31*
интерфейсы *30*
инфраструктура безопасности *31*
источники операционных данных *31—32*
потоки *30*
производительность *35*
NoSQL (не только SQL) *31—32*
Аудит процесса обработки больших данных *292*
Аудит, определение *354*
Аутентификация, определение *354*

Б

База данных в памяти, определение *354*
База данных пар «ключ-значение»
  Riak
  вторичные индексы *121*
  описание *120—121*
  параллельная обработка *121*
  переход по ссылкам *121*
  поиск *121*
  реализации *121*
  ссылки *121*
Базовая аналитика
  базовый мониторинг *184*
  идентификация аномалий *184*
  продольные и поперечные срезы *183*
Базы данных
  см. также нереляционные базы данных; RDBMS (relational database management systems, системы управления реляционными базами данных)
  столбцовые *125—126*, *362*
  база данных в памяти, определение *354*
  графовые базы данных *126—128*
  документальные базы данных *121—125*
  определение *354*
  пространственные *128—131*, *354*
  KVP (key—value pair, пара ключ—значение) *119—121*
Базы данных пар «ключ-значение»
  примеры *119*
  Riak *120—121*
Базы пространственных данных
  определение *354*
  описание *128—129*
  пакет приложений PostGIS/OpenGEO *129—131*
Безопасность
  в контексте *286*
  наилучшие методы *295—296*
  обезличивание данных *288—289*
  оценка риска *286—287*
  планирование *338—339*
  средства управления облачными базами данных *289*
  токенизация *289*
  управление *295—296*
  управления данными *352—353*
  HIPAA (Health Insurance Accountability and Portability Act, закон о преемственности и подотчетности медицинского страхования) *286*
  PHI (personal health information, персональная информация о состоянии здоровья) *286*
  PII (personal identifiable information, персональная информация, позволяющая идентифицировать личность) *286*
Безопасность инфраструктуры
  доступ к данным *71*
  доступ к приложениям *71*
  обнаружение угроз *72*
  шифрование данных *71—72*
Бизнес-подразделения, вовлечение в формирование стратегии *350—351*
Бизнес-правила, определение *355*
Бизнес-процессы
  моделирование *358*
  модификация *274*
  определение *355*
  преобразование *307*
Биометрия, определение *355*
Большие данные *226—227*
  см. также структурированные данные
  анализ *301—302*
  архитектурная основа *28—30*
  в качестве инструмента бизнес-планирования *300*

важность *19*
вариативность *26*
внесение изменений в процесс *303*
выбор основы для *305*
выбор стратегии *305—306*
выполнение плана *302*
источники *39—40*
наблюдение в реальном времени *303*
объем *26*
определение *19*, *355*
парадокс *226—227*
планирование с помощью *300—301*
поиск закономерностей в *234*
приложения *36—37*
проверка результатов анализа *302*
скорость *26*
сравнение с маленькими данными *226—227*
технологии *24—26*
управление *190*
управление *24—26*
функциональные требования *27—28*
характеристики *68—69*
цикл управления *28*
экономика *270—276*
экспериментирование *304*
Брокер, определение *355*

В

Валидность данных *264—265*
Варианты защиты данных
обезличивание данных *288—289*
средства управления облачными базами данных *289*
токенизация *289*
шифрование *288*
Веб-сайт Amazon *346*
Веб-сайт Refractions Research *129*
Веб-сайт SAS Institute *346*
Веб-сайт Teradata *346*
Веб-сайты
см. также интернет-сайты
Веб-сервис, определение *355*
Веб-управление *23—24*
Виртуализация
важность *85—87*
выгода от *93—94*
гипервизор *87*
данные *90—91*
данные и накопители информации *90—91*
диаграмма *84*
и абстракция *92—93*
изоляция *86*
инкапсуляция *86*
накопители информации *90—91*
определение *83*, *355*
приложения *88—89*
применение *93—94*
проблемы безопасности *91*
проблемы в управлении *91*
распределенные ресурсы *84*
секционирование *85—86*
серверы *87—88*
сети *89—90*
суть *84*
управление с помощью гипервизора *91—92*
характеристики *85—86*
Виртуализация накопителей информации *90—91*
Виртуализация памяти *90*
Виртуализация процессора *90*
Виртуализация сервера *87—88*
Виртуализация сети *89—90*
Витрины данных *22*, *35*, *355*
Влияние на общественную жизнь с помощью потоковых данных *315—316*
Время отклика, определение *355*

Входные данные *42*
Выделение ресурса, определение *355*
Вычисления как коммунальная услуга, определение *355*

Г

Гибридное облако, определение *356*
Гипервизор, определение *356*
  использование совместно с виртуальной системой *87*
  программный клиент *91—92*
  типы *92*
  управление виртуализацией с помощью *91—92*
  устройство *92*
Гранулярность, определение *356*
Графовые базы данных
  описание *126—127*
  Neo4J *127—128*

Д

Данные
  в движении *26*
  и виртуализация накопителей информации *90—91*
  интеграция *352*
  неструктурированные *196—197*, *358*
  определение *337*
  распределение *351*
  сравнение структурированных и неструктурированных *200*
  хранимые *26*, *363*
Данные лог-файлов *41*
Данные о посещении сайтов *42*
Данные реального времени, преимущества *314—315*
Данные с датчиков *41*
Данные точек продаж *41*
Данные, относящиеся к компьютерным играм *42*
Датчики, предоставление информации в реальном времени *313—314*
Диагностика заболеваний *260—262*
Документальные базы данных
  описание *121—122*
  CouchDB *123—125*
  JSON (JavaScript Object Notation, нотация объектов JavaScript) *122*
  MongoDB *122—123*

Ж

Жизненный цикл приложения, определение *356*
Журнал аудита, определение *356*

З

Заболевания, диагностика *260—262*
Задачи MapReduce
  аппаратная топология *141—142*
  сетевая топология *141—142*
  синхронизация *142*
  файловая система *142—143*
Задержка
  определение *356*
  проблема с *58*
Здравоохранение
  потоковые данные в *317*
  сохранение потока данных *317—318*

И

Избыточность физической инфраструктуры
  гибкость *68*
  доступность *68*
  масштабируемость *68*
  оборудование *70*
  общие сведения *67*
  отказоустойчивость *69*

производительность *68*
серверы *70*
сети *70*
сложность *69—71*
стоимость *68*
хранилище *70*
эксплуатация *71*
SLAs (service-level agreements, соглашения об уровнях обслуживания) *69—70*
Избыточность, важность *30—31*
Изменчивость данных *264—267*
Изолированное приложение, определение *357*
Изоляция *86*
Инкапсуляция *86*
Инструмент Sqoop (SQL-to-HadoopSQL в Hadoop)
взаимодействие с данными *165*
описание *164—165*
пакетный импорт *164*
прямой ввод *164*
экспорт данных *165*
Инструменты текстового анализа
аналитические решения SAS *211*
Attensity *208—209*
Clarabridge *209*
IBM Content Analytics *209—210*
OpenText *210*
Интеграция больших данных
бизнес-цель *238*
качество данных *238—239*
определения данных *239*
службы данных *239*
совершенствование *239*
ELT (extract/load/transform, извлечение/загрузка/преобразование), определение *239*
ETL (extract/transform/load, извлечение/преобразование/загрузка) *239*
MDM (Master Data Management, управление главными данными) *239*
Интеграция данных, наилучшие методы *245—246*
Интеграция информации, определение *356*
Интеграция источника данных и инкорпорирование *235—237*
исследовательский этап *233*
определение необходимых данных *232—233*
поиск закономерностей *234*
этап кодификации *235*
AWS (Amazon Web Services, веб-службы Amazon) *236*
FlumeNG *233—234*
Интеллектуальный анализ данных
алгоритмы *186—187*
классификационные деревья *186*
логистическая регрессия *186*
метод K-ближайших соседей *187*
методы кластеризации *187*
нейронные сети *186*
общие сведения *186—187*
определение *356*
Интернет-сайты
Альянс информационной безопасности облачных данных (Cloud Security Alliance) *344—345*
аналитические решения SAS *211*
веб-сайты для совместной работы *347*
Фонд открытых данных (Open Data Foundation) *344*
AIIM (Association for Informationand Image Management, Ассоциация по системам управления обработкой информации и изображений) *46*

Apache Software Foundation *345—346*
Attensity *208—209*
Clarabridge, текстовая аналитика *209*
CouchDB, база данных *123—125*
Hadoop, программная платформа *145*
HBase, столбцовая база данных *125—126*
Hurwitz&Associates *343—344*
IBM Content Analytics *209—210*
IBM *193—194*
MongoDB, база данных *122—123*
Neo4J, графовая база данных *127—128*
NIST (National Institute of Standards and Technology, Национальный институт стандартов и технологий) *345*
O'Reilly Strata and StrataRx конференция *347*
OASIS *346*
OGC (Open Geospatial Consortium, открытый геопространственный консорциум) *128—129*
OpenChorus *225—226*
OpenGeo Suite *129*
OpenStack, поставщик облачных услуг *107—108*
OpenText, текстовая аналитика *210*
Oracle *194*
Pentaho *194*
PostgreSQL *44*, *116—117*
Refractions Research *129*
Revolution Analytics *219*
Riak, база данных пар «ключ-значение» *120*
SAS *194*
Tableau *194*
Интернет-сайты для совместной работы *347*
Интерфейсы и потоки *72—74*
Инфраструктура
  определение *356—357*
  сервисы *357*
Источники данных
  интеграция с большими данными *330—331*
  объемы данных *19*
  разнообразие данных *19*
  скорость обработки данных *19*
Источники данных реального времени, использование потоковых данных с *320*

К

Каталог сервисов, определение *357*
Каталог, определение *357*
Качество больших данных
  двухстадийный подход *243*
  инструменты профилирования данных *244*
  приоритизация *242—244*
Качество данных, определение *357*
Кодд, Эдгар *42—44*
Коммерческая услуга, определение *357*
Компонент, определение *357*
Контейнер, определение *357*
Контроль данных, планирование *293—294*, *340*
Контроль, планирование *293—294*, *340*
Конференции
  форум по работе с большими данными в сфере розничной торговли (Big Data Retail Forum) *347*
  Big Data Conference (конференция по большим данным) *340*
  Hadoop World *347*
  O'Reilly Strata и StrataRx *347*

Structure Data *347*
TDWI (The Data Warehousing Institute, Институт изучения вопросов хранения данных) *347*
Конференции о данных *347*
Конференция Structure Data *347*
Конференция TDWI (The Data Warehousing Institute, Институт изучения вопросов хранения данных) *347*
Конфигурация
определение *357*
управление *363*
Критически важное приложение, определение *357*
Кэширование, определение *357*

M

Масштабируемость
важность *59—61*
как свойство облака *101*
определение *357—358*
поддержка *85*
Медицинская диагностика *259*
Метаданные
общие сведения *51*, *358*
репозиторий *360*
согласованность *351*
Модели предоставления, развитие *350*
Модели программирования
процедурная *135*
функциональная *135*
Модель ER (Entity-Relationship, объект-отношение) *21*, *358*
Модель соединений Интернета *55*
Монетизированная аналитика *188*
Мошенничество, предотвращение с помощью аналитики *328—330*
Мультитенантность, определение *358*
Мэшап, определение *358*

H

Научные исследования, использование потоковых данных в *313*
Нереляционные базы данных
см. также базы данных
возможности *117—118*
данные и модель запросов *118*
масштабируемость *117—118*
проектирование сохраняемости *118*
разнообразие интерфейсов *117—118*
согласованность в конечном счете *118*
NoSQL (нетолькоSQL) *117*
Неструктурированные данные *196—197*
видео *45*
источники *44—47*
контент веб-сайтов *45*
мобильные *45*
научные данные *45*
определение *44*, *358*
радаров *45*
сгенерированные людьми *45*
сгенерированные машинами *45*
социальные сети *45*
спутниковые изображения *45*
сравнение со структурированными данными *200*, *204*
структурирование *198—199*
текстовые *45*
фотографии *45*
характеристики *50*
эхолокаторов *45*

O

Облака
гибридные *99*, *103—104*
сравнение частного и общедоступного *97—99*, *75*
Облако
общедоступное *358*

частное *363*
Облачная доставка модели
Daas (Data as a Service, данные как услуга) *100*
IaaS (Infrastructure as a Service, инфраструктура как услуга) *99*, *103*
PaaS (Platform as a Service, платформа как услуга) *99—100*, *103*
SaaS (Software as a Service, ПО как услуга) *100*, *103—104*
Облачные вычисления
использование *96*
определение *95—96*, *358*
преимущества *96*
Облачные сервисы
доступ к данным *108—109*
затраты *108*
передача данных *108*
предупреждения *108*
производительность *108*
расположение данных *109*
соблюдение требований *108*
целостность данных *108*
Обработка в реальном времени, определение *358*
Обслуживание клиентов, оптимизация *205*
Обслуживание клиентов, повышение качества с помощью текстовой аналитики *322—324*
Общедоступное облако, определение *358*
Океаны, предоставление информации в реальном времени об *313—314*
Операционализация больших данных
диагностика заболеваний *259—262*
здравоохранение *259—260*
интеграция *258—260*
медицинская диагностика *260*
Операционная аналитика *188*, *359*
Операционная совместимость, определение *359*
Операционные базы данных *74—76*, *113—115*
Оптимальное следующее действие, выбор *324—328*
Организационная структура
внедрение *293*
контроль *293—294*
определение качества данных *294—295*
правила управления *294—295*
управление рисками *293—294*
Отказоустойчивость, определение *359*
Открытый исходный код, определение *359*
Отчетность и визуализация *36*
Очистка данных, определение *358*

П

Пакет приложений PostGIS/OpenGEO *129—131*
описание *129—130*
GeoExt *130*
GeoServer *130*
GeoWebCache *130*
OpenLayers *130*
Пакетная обработка, определение *359*
Передача данных и интерфейсы *72—74*
Передовая практика
безопасность *295—296*
для рабочих процессов *263—264*
изучение *341*
интеграция данных *245—246*
использование шаблонов *341*

контроль данных *340*
обнаружение данных *337*
обнаружение недостающих данных *338*
определение *359*
определение целей *335—336*
планирование безопасности *338—339*
проверка допущений *340*
стратегия управления данными *339*
технологические возможности *338*
Перемещаемые данные
данные точек продаж *311*
датчики *310*
медицинские устройства *311*
определение *359*
потоковые данные *311—312*
сбор информации *311*
сообщения *311*
телекоммуникации *311*
Петабайты данных, анализ *25*
План обеспечения целостности
бюджеты *280—281*
мыслите глобально *284*
навыки *280—281*
обучайтесь *284*
основные этапы *282—283*
планирование производительности *279*
помощь *284*
разработка ПО *280*
разработка *282—284*
риск *281*
определение действий *336—337*
фактор времени *279*
формирование ожиданий *284*
экспериментирование *284*
этапы *282—284*
Планирование с помощью данных *300—301*
Платформа анализа больших данных
интеграция с развернутыми облаками *223*
использование существующих алгоритмов *223*
использование существующих данных *223*
обеспечение дешевого хранилища *223*
пакетная обработка *223*
поддержка NoSQL *223*
поддержка типов данных *223*
преодоление низкой задержки *223*
OpenChorus *225—226*
Платформа анализа больших данных
возможности *224*
Платформа, определение *360*
Плоские файлы *42*
Позднее связывание, определение *359*
Поиск, сравнение с текстовым анализом *199—200*
Полиглотная сохраняемость *131—132*
Политика защиты организации *294—295*
Поставщики облачных услуг
Amazon EC2 *104—106*
Google *106*
MicrosoftAzure *107*
OpenStack *107—108*
Потоковая передача данных
в энергетике *318*
влияние на бизнес *256*
влияние на общественную жизнь *315—316*
датчики *313—314*
и сложная обработка событий (Complex Event Processing, CEP) *248*, *255—256*

использование в научно-исследовательском институте *319*
использование на ветровой электростанции *319*
использование *249—253*
источники архивных данных *320*
источники данных реального времени *320*
медицинская отрасль *317*
метаданные *251—252*
научное исследование *313*
общие сведения *248*, *359*
полезность *250*
пример с группой врачей-диагностов *251*
пример с нефтеразведкой *251*
пример с телекоммуникациями *250—251*
пример с электростанцией *250*
примеры *250—251*
принципы *250*
с воздействием на окружающую среду *312—315*
ценность *311—312*
Apache S4 *253*
IBM InfoSphere Streams *252*
Twitter's Storm *252—253*
Потоковые данные
воздействие на технологии *274—275*
наилучший метод работы *263—264*
общие сведения *262—263*, *359*
рабочая нагрузка *263—264*
Преобразование данных, определение *360*
Преобразование и анализ концепции *202—203*
извлеченная информация *202—203*
классификации *203*
ключевые слова *202*
отношения *202*
события *202*
термины *202*
факты *202*
эмоциональная окраска *203*
NLP (Natural Language Processing, обработка естественного языка) *200—202*
Приложения
большие данные *104*
вертикальные *80*
виртуализация *89—90*
гибкость *221*
горизонтальные *80*
данные лог-журналов *80*
качество *221*
персонализация *80—81*
специализированные *218—220*
частично специализированные *220—222*
скорость внедрения *221*
стабильность *221*
GeoTools *221—222*
JUNG (Java Universal Network Graph, универсальный сетевой Java-граф) *221*
TA—Lib (Technical Analysis library, библиотека технического анализа) *221*
Приложения для обработки данных лог-журналов *80*
Приложения, работающие с большими данными *104*
Применение аналитики
бизнес-ценность *324*
международный банк *327*
организация, эмитирующая кредитные карты *327—328*
страховая компания *326—327*
Проверка данных, важность *28*
Прогностическая аналитика, определение *359*

Программное обеспечение
большие данные *104*
вертикальные *80*
виртуализация *89—90*
гибкость *221*
горизонтальные *80*
данные лог-журналов *80*
качество *221*
персонализация *80—81*
специализированные *218—220*
частично специализированные *220—222*
скорость внедрения *221*
стабильность *221*
GeoTools *221—222*
JUNG (Java Universal Network Graph, универсальный сетевой Java-граф) *221*
TA—Lib (Technical Analysis library, библиотека технического анализа) *221*
Программы
большие данные *104*
вертикальные *80*
виртуализация *89—90*
гибкость *221*
горизонтальные *80*
данные лог-журналов *80*
качество *221*
персонализация *80—81*
специализированные *218—220*
частично специализированные *220—222*
скорость внедрения *221*
стабильность *221*
GeoTools *221—222*
JUNG (Java Universal Network Graph, универсальный сетевой Java-граф) *221*
TA—Lib (Technical Analysis library, библиотека технического анализа) *221*
Продукты для бизнес-аналитики
аналитические алгоритмы *190—191*
данные *189—190*
модификация *189*
поддержка инфраструктуры *191*
NASA *193*
Nokia *192—193*
Производительность
важность *85*
выполнение алгоритмов *59—60*
инструменты *33*
масштабируемость *59—60*
распределенные вычисления *59—60*
службы данных *33*
факторы *33—35*
BigTable, система хранения данных *34*
Hadoop, программная платформа *34*
MapReduce *35*, *61*
Производительность обработки данных, управление *353*
Промежуточное ПО, определение *360*
Пропускная способность, определение *360*
Протокол, определение *359*
Профилирование данных, определение *360*
Процедурное программирование *135*
Процесс обработки больших данных, аудит *292*
Процесс, определение *360*
Пул ресурсов, определение *360*

Р

Раннее связывание, определение *360*
Распределенные вычисления
задержка *58*

изменение стоимости *57—58*
необходимость *57*
общие сведения *53*
определение *360*
протоколы *54—55*
развитие *59—60*
согласованная модель *55*
спрос и решения *59*
узлы *59*
DARPA *54—55*
RPCs (remote procedures calls, удаленные вызовы процедур) *55—56*
Распределенные ресурсы, использование *84*
Расчет окупаемости инвестиций в большие данные *276*
Реализация аналитики больших данных
международный банк *327*
организация эмитирующая кредитные карты *327—328*
страховая компания *326—327*
ценность для бизнеса *324*
Реестр, определение *360*
Резервное копирование, определение *360*
Реки, предоставление информации в реальном времени *313—314*
Рекомендации и предостережения
безопасное управление данными *352—353*
интеграция данных *352*
источники данных и стратегии работы с ними *350—351*
корректные метаданные *351*
оценка моделей доставки *350*
постепенный рост *352*
привлечение структурных подразделений *350*
производительность данных *353*
разнообразие подходов *351—352*
распределение данных *351*
Репозиторий, определение *360*
Ресурсы
Альянс информационной безопасности облачных данных (Cloud Security Alliance) *344—345*
организации по стандартизации *344*
Фонд открытых данных (Open Data Foundation) *344*
Apache Software Foundation *345—346*
Hurwitz & Associates *343—344*
NIST (National Institute of Standards and Technology, Национальный институт стандартов и технологий) *345*
OASIS *346*
Решения для аналитики больших данных
IBM *193—194*
Oracle *194*
Pentaho *194*
SAS *194*
Tableau *194*
Риск, оценка для бизнеса *286—287*
Руководство
важность *289*
видимость данных *291*
ключевые пользователи *292—293*
наилучшие методы *295—296*
ненадежные сотрудники *291*
определение *360*

C

Сайты
Альянс информационной безопасности облачных данных (Cloud Security Alliance) *344—345*

аналитические решения SAS *211*
веб-сайты для совместной работы *347*
Фонд открытых данных (Open Data Foundation) *344*
AIIM (Association for Informationand Image Management, Ассоциация по системам управления обработкой информации и изображений) *46*
Apache Software Foundation *345—346*
Attensity *208—209*
Clarabridge, текстовая аналитика *209*
CouchDB, база данных *123—125*
Hadoop, программная платформа *145*
HBase, столбцовая база данных *125—126*
Hurwitz&Associates *343—344*
IBM Content Analytics *209—210*
IBM *193—194*
MongoDB, база данных *122—123*
Neo4J, графовая база данных *127—128*
NIST (National Institute of Standards and Technology, Национальный институт стандартов и технологий) *345*
O'Reilly Strata and StrataRx, конференция *347*
OASIS *346*
OGC (Open Geospatial Consortium,открытый геопространственный консорциум) *128—129*
OpenChorus *225—226*
OpenGeo Suite *129*
OpenStack, поставщик облачных услуг *107—108*
OpenText текстовая аналитика *210*
Oracle *194*
Pentaho *194*
PostgreSQL *44*, *116—117*
Refractions Research *129*
Revolution Analytics *219*
Riak, база данных пар «ключ-значение» *120*
SAS *194*
Tableau *194*
Свойства облака
масштабируемость *101*
начальные затраты *102*
отказоустойчивость *102*
самообслуживание *101*
слабосвязанные ресурсы *75*
эластичность *101*
PAYG (Pay as You Go, оплата по факту использования) *103*
Связывание, определение *361*
Секционирование оборудования, определение *361*
Секционирование *85—86*
Семантика, определение *361*
Сервис, определение *361*
Сервисы больших данных Google
Big Query *106*
Compute Engine *106*
Prediction API *106*
Сетевые распределенные вычисления, определение *361*
Сетевые хранилища данных *42*
Сеть, определение *361*
Скриптовый язык, определение *361*
Слабое связывание, определение *361—362*
Слабоструктурированные данные *46*
Служба технической поддержки, определение *362*

Службы и инструменты организации данных *76—77*
Советы
безопасное управление данными *352—353*
интеграция данных *352*
источники данных и стратегии работы с ними *350—351*
оценка моделей доставки *350*
постепенный рост *352*
привлечение структурных подразделений *350*
производительность данных *353*
разнообразие подходов *351—352*
распределение данных *351*
корректные метаданные *351*
Соединитель, общие сведения *50—51*
Сохранность, определение *362*
Социальные медиа *205—206*
Среда выполнения Pig
задания map (карта) и reduce (сокращение) *163*
локальный режим *163*
Hadoop *163*
Стандарты, определение *362*
Столбцовая база данных *362*
описание *125*
HBase *125—126*, *160—161*
Стратегия управления, планирование *339*
Структурированные данные
см. также большие данные; неструктурированные данные
ввод *42*
данные лог-файлов *41*
данные о посещении сайтов *42*
датчиков *41*
источники *40—42*
определение *40*, *362*
относящиеся к компьютерным играм *42*
сгенерированные компьютерами *41*
сгенерированные людьми *42*
сгенерированные машинами *41*
сравнение с неструктурированными данными *200*, *204*
строки *40*
точек продаж *41*
финансовые *41*
характеристики *50*
Схема (БД) *43*
Сценарий в здравоохранении *259—262*

Т

Таблицы
запрос в СУРБД *43—44*
связь в СУРБД *43*
Текстовая аналитика
записи информационно-справочной службы *197—198*
обработка *197—198*
определение *362*
сравнение с поиском *199—200*
улучшение качества обслуживания клиентов *322—324*
Технологические возможности, осведомленность о *338*
Типы данных
интеграция *50—51*
метаданные *51*
соединители *50—51*
характеристики *49—50*
Транзакционное поведение, поддержка *75*
Транзакция, определение *362*
Требования
не реального времени *47—49*
реального времени *47—49*
Требования к качеству данных *294—295*
Требования реального времени
адаптивность *49*

малая задержка *49*
масштабируемость *49*
универсальность *49*

У

Углубленная аналитика *184—185*, *362*
алгоритмы интеллектуального сбора данных *185*
прогнозное моделирование *185*
статистические алгоритмы *186*
текстовая аналитика *185—186*
Узлы в распределенных вычислениях *59*
Управление удостоверениями, определение *362*
Управление данными
веб-управление *23—25*
структуры данных *20—22*
управление контентом *23—25*
этапы *20—26*
Управление доступом, определение *362*
Управление изменениями, определение *362—363*
Управление контентом *23—24*, *286*
Управление рисками, подготовка к *293—294*
Управление сервисами, определение *362*
Управление удостоверениями, определение *362*
Устаревшее приложение, определение *363*

Ф

Фабрика соединителей, реализация *74*
Федерация, определение *363*
Федерация данных, определение *363*
Финансовые данные *41*
Фонд открытых данных (Open Data Foundation) *344*
Форум по работе с большими данными в сфере розничной торговли (Big Data Retail Forum) *347*
Функциональное программирование
общие сведения *135*
операторы *136*

Х

Характеристики анализа
интерактивный *216*
использование атрибутов *216*
на основе данных *215*
ориентированный на действие *215*
ориентированный на решение *215*
программный *216*
IaaS (Infrastructure as a Service, инфраструктура как услуга) *217*
Хранилище данных *26—27*, *39*
см. также операционные базы данных
анализ больших данных *172—174*
загрузка *173—174*
извлечение *173—174*
изменение роли *174—175*
интеграция больших данных и *168*
история *167*
методы управления *174—175*
модели внедрения *175—176*
модель программно-аппаратного комплекса *175—176*
облачная модель *176*
определение *363*
оптимизация *168*
основа для интеграции *173*
перспективы *176—177*

потоки данных *169*, *171*
преобразование *172—173*
пример гибридного процесса *170—172*
сравнение со структурами больших данных *169—170*

Ц

Цели, определение *335—336*
Центр компетенции, определение *363*

Ч

Частное облако, определение *363*
Черный ящик, определение *363*

Ш

Шина, определение *364*
Шифрование, уязвимости *288*

Э

Экономика больших данных
изменения бизнес-процессов *274*
источники данных *270—274*
начало использования *272—273*
поддержка стабильности *273*
поиск персонала *275—276*
принадлежность данных *273*
расчеты *259—270*
типы данных *270—274*
частота использования данных *272—274*
ROI (return on investment, доход на инвестиции) *276*
Эластичность, определение *364*
Эмуляция, определение *364*
Энергетика, потоковые данные в *318—320*
Эталонная архитектура, слои *214*

Я

Язык разметки, определение *364*
Языки баз данных *75*

A

ACID
атомарность *75*
изоляция *75*
определение *364*
согласованность *75*
устойчивость *75*
AIIM (Association for Informationand Image Management, Ассоциация по вопросам управления обработкой информации и изображений) *46*
Amazon EC2, поставщик облачных услуг *103*
DynamoDB *105*
HPC (High Performance Computing, высокопроизводительные вычисления) *105—106*
MapReduce *105*
Redshift *106*
S3 (Simple Storage Service) *105*
Apache Hadoop, программная платформа
веб-сайт *146*
определение *35*, *145—146*, *365*
разработка *146*
MapReduce, технология *146*
YARN (Yet Another Resource Negotiator, еще один ресурсный посредник) *159*
Apache S4 *253*
Apache Software Foundation *345—346*
API (application program in interfaces, интерфейсы программирования приложений) *72—74*, *364*
Attensity, текстовая аналитика *208—209*
AWS (Amazon Web Services, веб-службы Amazon) *236*

B

Big Data Conference (конференция по большим данным) *347*
Bigtable, определение *364*
Big Table, система хранения данных *34*
BLOBs (binary large objects, большие двоичные объекты) *23*
BPaaS (Business Process as a Service, бизнес-процесс как услуга) *364*
BPEL (Business Process Execution Language, языки исполнения бизнес-процессов) *364*
BPM (business process management, управление бизнес-процессами) *364*

C

CEP (Complex Event Processing, сложная обработка событий) *248*, *365*
  влияние на бизнес *256*
  использование *253—255*
  сравнение с потоками *255—256*
Clarabridge, текстовая аналитика *209*
Cloudera, веб-сайт *346*
CMDB (configuration management data base, база данных управления конфигурациями) *363*
CMSs (content management systems, системы управления контентом) *46—47*
COBIT (Control Objectives for Information and Related Technology, задачи управления для информационных и смежных технологий) *365*
CouchDB, база данных
  модель представления *123*-124
  описание *123—124*
  распределенные службы *124*
  реализации *124*
  репликация *124*
  сжатие *124*
CRM (customer relationship management, управление взаимоотношениями с клиентами) *365*
CRUD (create/retrieve/update/delete, создание/получение/обновление/удаление) *114—115*

D

DaaS (Data as a Service, данные как услуга) *100*
DARPA *54—55*
DBMS (data base management system, система управления базами данных) *361*

E

EC2, поставщик облачных услуг
  DynamoDB *105*
  HPC (High Performance Computing, высокопроизводительные вычисления) *105—106*
  MapReduce *105*
  RedShift *106*
  S3 (Simple Storage Service, простая служба хранилища) *105*
EDM (enterprise data management, управление данными предприятия)
  архитектура *277*
  безопасность *277*
  качество *277*
  метаданные *277*
  определение *277—278*
  принадлежность *277*
  принципы *278*
  управление *277*
  фундамент *277*

ELT (extract/load/transform, извлечение/загрузка/преобразование)
использование Hadoop в качестве *244—245*
определение *239*, *242*, *365*
ERP (enterprise resource planning, планирование ресурсов предприятия) *365*
ESB (Enterprise Service Bus, сервисная шина предприятия) *365*
ETL (extract/transform/load, извлечение/преобразование/загрузка) *239*, *365*
в пакетной обработке *240*
инфраструктура для интеграции *240*
преобразование данных *241—242*

F

FlumeNG, использование для больших данных
интеграция *233—234*

G

GeoTools *221—222*
Google Prediction API *219—220*
Google, веб-сайт *346*

H

Hadoop MapReduce
возможности *151—153*
перемещение данных *152*
подготовка данных *153*
привязка данных *153—154*
рабочий процесс *152*
редукция и объединение *154—155*
HadoopWorld, конференция *347*
Hadoop, экосистема
основа для применения больших данных *158*
Pig, среда выполнения *162—164*
PigLatin, язык *162—164*
Sqoop *164—165*
Zookeeper *165—166*
Hadoop, программная платформа
веб-сайт *146*
общие сведения *145—146*
определение *35*, *145—146*, *365*
разработка *146*
MapReducetехнология *146*
YARN (Yet Another Resource Negotiator, еще один ресурсный посредник) *159*
HBase, столбцовая база данных *125—126*
высокая доступность *126*
клиентский API *126*
поддержка работы ИТ *126*
сегментация *126*
согласованность *126*
хранение больших данных с помощью *160—161*
HDFS (Hadoop Distributed File System, распределенная файловая система Hadoop)
блоки *146—148*, *151*
кластер *148*
конвейеры данных *150—151*
метаданные *150*
общие сведения *145—146*
определение *365*
пространство имен *148*
средства проверки контрольных сумм *149*
узлы данных *148—149*
узлы имен *147—148*
целостность данных *149*
HIPAA (Health Insurance Accountability and Portability Act, закон о преемственности и подотчетности медицинского страхования) *286*

Hive
интеллектуальный анализ данных с помощью *161—162*
контейнеры *162*
метаданные *162*
разделы *161*
таблицы *161*
Hurwitz&Associates *343—344*

I

IaaS (Infrastructure as a Service, инфраструктура как услуга) *99*, *103*, *366*
IBM
аналитические решения *193—194*
веб-сайт *346*
InfoSphere Streams *252*
IBM Watson
гипотезы *206—208*
обучение на основе фактов *206—207*
NPL (Natural Language Processing, обработка естественных языков) *207*
ISO (International Organization for Standardization, международная организация по стандартизации) *366*
ITIL (Information Technology Infrastructure Library, библиотека инфраструктуры информационных технологий) *366*

J

JSON (JavaScript Object Notation, нотация объектов JavaScript) *122*
JUNG (Java Universal Network Graph, универсальный сетевой Java-граф) *221*

L

LAMP (Linux, Apache, MySQL, PHP, Perl или Python) *366*
Linux
определение *366*
веб-хостинг *355*

M

MapReduce *33—34*, *105*
алгоритмы *137—138*
и виртуализация *93—94*
инфраструктура выполнения *140*
источники *133—134*
обработка сбоев/ошибок *141*
определение *366*
планирование *140*
поведение *140—141*
поток данных *139*
разработка *61*
реализации *61*
синхронизация *140—141*
совместное расположение кода и данных *141*
функции map (карта) и reduce (сокращение) *135—138*
функция map (карта) *135—137*
функция reduce (сокращение) *137—138*
MDM (Master Data Management, управление главными данными) *239*
Microsoft Azure, поставщик облачных услуг *107*
Microsoft, веб-сайт *346*
MOM (Message Oriented Middleware, промежуточное ПО для передачи сообщений), определение *366*
MongoDB, база данных *122—123*
MySQL, определение *366*

N

NASA, использование прогностических моделей *193*
Neo4J, графовая база данных *127—128*
  интеграция с базами данных *127*
  отказоустойчивость *128*
  реализации *128*
  службы синхронизации *128*
  язык запросов *128*
NIST (National Institute of Standards and Technology, Национальный институт стандартов и технологий) *345*
NLP (Natural Language Processing, обработка естественного языка) *73—74*
  анализ дискурсного уровня *201*
  лексический анализ *200—201*
  морфологический анализ *200—201*
  общие сведения *200*
  семантический анализ *201*
  синтаксический анализ *201*
Nokia *192—193*
NoSQL (не только SQL) *31—32*, *76*, *117*, *366*
NPL (Natural Language Processing, обработка естественных языков) *186—187*

O

O'Reilly Strata and StrataRx, конференция *347*
OASIS (Organization for the Advancement of Structured Information Standards, организация по развитию стандартов структурированной информации) *346*
ODBMS (object data base management systems, системы управления объектными базами данных) *23*
OGC (Open Geospatial Consortium, открытый геопространственный консорциум) *128*
OODBMS (object-oriented data base management system, система управления объектно-ориентированными базами данных) *366*
OpenChorus, прикладная платформа *225—226*
OpenGeo Suite, веб-сайт *129*
OpenStack, поставщик облачных услуг *107—108*
OpenText, текстовая аналитика *210—211*
Oracle, аналитические решения *194*
Oracle, веб-сайт *346*

P

P2P (одноранговая сеть), определение *367*
PaaS (Platform as a Service, платформа как услуга) *99—100*, *103*, *367*
Pentaho, аналитические решения *194*
PHI (personal health information, персональная информация о состоянии здоровья) *286*
PigLatin, язык *164*
Pig, программы
  встроенные *163*
  запуск *164*
  операторы *164*
  сценарный файл *163*
  Grunt, командный интерпретатор *163*
PII (personal identifiable information, персональная информация, позволяющая идентифицировать личность) *286*

PostgreSQL
веб-сайт *44*
определение *367*
реляционная база данных *116—117*

R

RDBMSs (relational data base management systems, системы управления реляционными базами данных)
см. также столбцы
PostgreSQL *44*, *116—117*
базы данных *114—115*
важность *21*
запросы к таблицам *43—44*
изобретение *42—43*
использование SQL в *75*
недостатки *44*
определение *361*
основание *115*
первичный ключ *114*
развитие *115*
связь между таблицами *43*
схема *43*
таблицы *114*
хранилище данных *43*
REST (Representational State Transfer, передача репрезентативного состояния) *73*, *367*
Revolution Analytics *219*
RFID (radio frequency identification, радиочастотная идентификация) *367*
RPCs (remote procedures calls, вызовы удаленных процедур) *55—56*, *367*
R-среда *218—219*

S

SaaS (Software as a Service, ПО как услуга) *100*, *103—104*
SAN (storage-area-network, сеть хранения данных), определение *367*
SLAs (service-level agreements, соглашения об уровнях обслуживания) *69*, *367*
SOA (service-oriented architecture, сервисно-ориентированная архитектура), общие сведения *367—368*
SOAP (Simple Object Access Protocol, простой протокол доступа к объектам) *368*
SQL (structured query language, язык структурированных запросов) *368*
использование в реляционной модели *75*
сравнение с NoSQL *76*
SSL (Secure Sockets Layer, уровень безопасных сокетов), общие сведения *368*

T

TA—Lib (Technical Analysis library, библиотека технического анализа) *221*
TLS (Transport Layer Security, безопасность транспортного уровня) *368*
TQM (Total Quality Management, глобальное управление качеством) *368*
Twitter's Storm *252—253*

U

U.S. DARPA *54—55*

V

VM (virtual machine, виртуальная машина) *87*

W

Watson *см.* IBM Watson
WS (Web Standard, веб-стандарт), общие сведения *368*
WSDL (Web Service Definition Language, язык описания веб-стандартов) *368*

X

XML (eXtensible Markup Language, расширяемый язык разметки) *74*, *368*
XML Schema, определение *368*
XSD (XML Schema Definition, определение XML Schema) *368*
XSLT (eXtensible Stylesheet Language Transformation, расширяемый язык стилей для преобразований) *368*

Y

YARN (Yet Another Resource Negotiator, еще один ресурсный посредник) *159*

Z

Zookeeper
  надежный обмен сообщениями *166*
  описание *165*
  самостоятельное определение ведущего узла *166*
  синхронизация процессов *165—166*
  управление конфигурацией *166*

Об авторах

Джудит С. Гурвиц — президент и глава совета директоров исследовательской и консалтинговой компании Hurwitz & Associates. Сфера деятельности компании — инновационные технологии в бизнесе, в частности облачные вычисления, большие данные, аналитика, разработка программного обеспечения, управление сервисами, а также обеспечение безопасности и стратегическое управление. Джудит Гурвиц — идеолог компании, специалист в области информационных технологий и бизнес-стратегий, автор и соавтор ряда книг о системах распределенных вычислений (*Smart or Lucky? How Technology Leaders Turn Chance into Success,* 2011; *Hybrid Cloud For Dummies,* 2012; *Cloud Computing For Dummies,* 2010; *Service Management For Dummies,* 2009; *Service Oriented Architecture For Dummies,* 2009; эксклюзивные корпоративные издания для CloudBees (*Platform as a Service For Dummies,* 2012) и IBM (*Cloud For Dummies,* 2011; *Private Cloud For Dummies,* 2011; *Information on Demand For Dummies,* 2008). В качестве консультанта сотрудничала со многими ведущими организациями в различных отраслях, способствуя внедрению инновационных информационных платформ. Основатель группы компаний Hurwitz Group, член консультативных комитетов ряда организаций. Окончила бакалавриат и магистратуру в Бостонском университете, удостоена звания «Выдающийся выпускник колледжа гуманитарных и естественных наук Бостонского университета» (2005), член Совета выпускников Бостонского университета, лауреат премии Массачусетского совета по технологическому лидерству (MASS TLC) (2005).

Алан Ф. Ньюджент — главный консультант Hurwitz & Associates, работает в области инновационных информационных технологий бо-

лее тридцати лет. В прошлом исполнительный и технический директор компании Mzinga, Inc. — ведущей организации в области разработки и обслуживания облачных программных решений для работы с большими данными, аналитики в режиме реального времени, исследования социальных медиа и управления сообществами; в качестве исполнительного вице-президента и технического директора компании CA, Inc. занимался разработкой стратегических направлений развития технологий компании, до этого в должности первого вице-президента и руководителя подразделения систем управления сетями масштаба предприятия (ESM) осуществлял контроль над портфелем продукции для инфраструктуры и управления данными. До работы в CA, Inc. Алан Ньюджент был первым вице-президентом и техническим директором фирмы Novell и способствовал переходу компании к использованию открытого кода и решений в сфере идентификационной информации. Сотрудничая с компанией BellSouth в качестве внештатного технического директора, возглавил программу по объединению и преобразованию оперативных данных компании. Независимый член совета директоров компании Adaptive Computing, председатель консультативного совета компании SpaceCurve, член консультативного совета компании N-of-one. Автор многочисленных работ в сфере бизнеса и технологий, участник отраслевых конференций.

Алан Ньюджент имеет лицензию пилота-любителя, более 30 лет профессионально играет в покер, в свободное время занимается реконструкцией старинных американских автомобилей и мотоциклов, увлекается высокой кухней, коллекционирует антикварные книги, американские и итальянские вина.

Ферн Халпер — доктор наук, научный сотрудник Hurwitz & Associates, директор по исследованиям в области углубленной аналитики Института изучения вопросов хранения данных (TDWI). Более 20 лет работает в области анализа данных, бизнес-анализа и разработки стратегий, автор ряда статей и научных исследований, активный участник отраслевых конференций. Ведет блог, посвященный развитию технологий. В прошлом работала на руководящих должностях в компании AT&T Bell Laboratories and Lucent Technologies, где отвечала за развитие инновационных систем анализа данных, а также за разработку стратегии и линеек продуктов в сфере интернет-коммуникаций. Вела курс по информационным технологиям в ряде университетов. Окончила бакалавриат Колгейтского университета, получила степень доктора наук в Техасском университете A&M. Соавтор ряда

книг, включая эксклюзивные корпоративные издания (*Hybrid Cloud For Dummies,* 2012; *Cloud Computing For Dummies,* 2010; *Service Oriented Architecture For Dummies,* 2009; *Service Management For Dummies,* 2009; *Cloud For Dummies,* 2011 (для IBM); *Platform as a Service For Dummies,* 2012 (для CloudBees); *Information on Demand For Dummies,* 2008 (для IBM)).

Марсия А. Кауфман — партнер-основатель и главный операционный директор Hurwitz & Associates, автор многочисленных трудов, посвященных вопросам коммерческого применения виртуальных технологий и облачных вычислений, в частности развитию облачной инфраструктуры и инновационным бизнес-моделям, шифрованию данных и безопасности, а также обработке транзакций в реальном времени в облачных средах. Более 20 лет занимается стратегическим развитием коммерческой деятельности, отраслевыми исследованиями, программным обеспечением для распределенных вычислений, обеспечением качества ПО, управлением информацией и аналитикой. Марсия Кауфман работала в финансовых организациях, в сфере производства и в сфере услуг. Разрабатывала сложные отраслевые модели и прогнозы в компании Data Resources, Inc. (DRI). Бакалавр Коннектикутского колледжа (математика и экономика), получила степень магистра делового администрирования в Бостонском университете. Соавтор книг *Hybrid Cloud For Dummies,* 2012; *Cloud Computing For Dummies,* 2010; *Service Management For Dummies,* 2009; *Service Oriented Architecture For Dummies,* 2009; эксклюзивных корпоративных изданий для CloudBees (*Platform as a Service For Dummies,* 2012) и IBM (*Cloud For Dummies,* 2011; *Private Cloud For Dummies,* 2011; *Information on Demand For Dummies,* 2008).

Благодарности

Сердечно благодарим наших друзей из издательства Wiley, в особенности нашего редактора Николь Шолли. Кроме того, мы хотели бы поблагодарить нашего технического редактора Бренду Майкельсон за ее вклад в создание этой книги.

Особая благодарность ведущим экспертам, которые любезно согласились поделиться своими знаниями по широкому кругу вопросов, связанных с гибридным облаком, за брифинги, предоставленные материалы исследований и эмпирические данные, а также за критику. Нам помогали:

Форрест Карман из Context Relevant;

Мэтт Уокен из Dell;

Боб Зурек из Epsilon;

Рик Клементс, Дэвид Карриган, Фил Франциско, Стивен Голд, Глен Хинце, Джефф Джонс, Нэнси Коп, Дэйв Линдквист, Анхель Луис Диаз, Билл Мэтьюз, Ким Майнор, Трейси Мустачио, Боб Палмер, Крейг Райнхарт, Ян Шауэр, Брайан Вайл, Глен Циммерман из IBM;

Майкл Хайски, Стив Миллард из Kognitio;

Джейкоб Споэльстра из Opera Solutions;

Рамон Чен, Дейдре Махон из RainStor;

Малком Александер, Майкл Эймс из SAS Institute;

Крис Кин из VMware;

Майкл Лэмбл из Xtremedata.

Авторы

Джудит Гурвиц посвящает эту книгу
своему мужу Уоррену,
детям Саре и Дэвиду, матери Элейн,
а также памяти своего отца Дэвида.

Алан Ньюджент посвящает эту книгу
своей жене Джейн
в благодарность за ее любовь и поддержку,
детям Крису, Джеффи и Грегу,
а также памяти своих родителей.

Ферн Халпер посвящает эту книгу
своему мужу Клэю,
дочерям Кэти и Линдси,
а также своей сестре Адриенне.

Марсия Кауфман посвящает эту книгу
своему мужу Мэттью,
детям Саре и Эмили,
а также своим родителям Глории и Ларри.

СБЕРБАНК