

1

Что такое сетка данных и зачем она нужна

В ЭТОЙ ГЛАВЕ

- ✓ Даем определение сетки данных
- ✓ Знакомимся с ключевыми понятиями, которые относятся к сетке данных
- ✓ Разбираемся, почему сетка данных — это смена социотехнической парадигмы
- ✓ Перечисляем преимущества сетки данных
- ✓ Выявляем возможные проблемы при внедрении сетки данных

Сетка данных — это парадигма децентрализации. В ней децентрализованы владение данными, их преобразование в информацию и обслуживание. Эти меры направлены на то, чтобы извлекать из данных максимальную ценность, устраняя узкие места этой процедуры.

Концепция сетки данных разрушает пространство данных. Крупные и мелкие компании во всех уголках интернета наперегонки пытаются продемонстрировать свои *наработки, похожие на сетку данных*. Это становится *новой тенденцией*; ее стремится подхватить каждая организация, которая хочет извлекать больше пользы из своих данных. Эта книга описывает парадигму сетки данных как социотехническую архитектуру с упором на *социальную составляющую*. Основное внимание уделяется не технологиям, а людям, процессам и организациям. Сетки данных можно (хотя и не обязательно) реализовывать с помощью тех

же технологий, на которых функционирует большинство современных систем работы с данными.

Но поскольку сетка данных остается предметом постоянных дискуссий, а образцовые методики и стандарты развиваются медленно, мы пришли к выводу, что стоит написать обстоятельную книгу, которая охватывает не только ключевые принципы работы сеток данных, но и примеры с разными вариантами того, как адаптировать их к любой компании. Эта книга предназначена именно для того, чтобы помочь вам приступить к работе со своей собственной сеткой данных.

Книга не претендует на роль всеобъемлющего теоретического исследования, зато предлагает практические рекомендации, которые необходимы, чтобы начать работу.

Сперва мы узнаем, на какие основные идеи опирается понятие сетки данных, а также какие преимущества и проблемы с ней связаны. Мы также дадим свое определение сетки данных, которое сформулировано с точки зрения конечных результатов и практичности.

1.1. ОСНОВЫ СЕТКИ ДАННЫХ

Парадигма сетки данных заключается в первую очередь в том, чтобы децентрализовать ответственность. Например, команда, которая разрабатывает компонент для регистрации клиентов компании, также создает набор данных, с помощью которого можно анализировать зарегистрированных клиентов. При этом команда заботится о том, чтобы этот набор был в удобном формате: она преобразует данные (например, в файл CSV) и предоставляет их так, как нужно потребителям данных (например, через центральную систему обмена файлами).

Но это простое с виду описание связано с далеко идущими последствиями, потому что в большинстве компаний данные рассматриваются как *побочный продукт*. Обычно они приобретают ценность только после того, как в качестве побочного продукта размещаются в хранилище, затем силами централизованной команды по работе с данными попадают в центральный узел системы, а затем их снова получают децентрализованные субъекты. Таким субъектом может быть аналитик в отделе маркетинга, или рекомендательная система, которая используется в маркетинговой кампании, или служба вывода в веб-интерфейс. На рис. 1.1 изображена общая форма архитектуры данных — как с организационной, так и с технической точки зрения. Мы также постарались показать все ее подводные камни.

Здесь просматриваются два уровня централизации:

- Централизованная технология в виде хранилища данных и обычного инструментария в области инженерии и анализа данных.
- Организационная централизация команды по работе с данными.

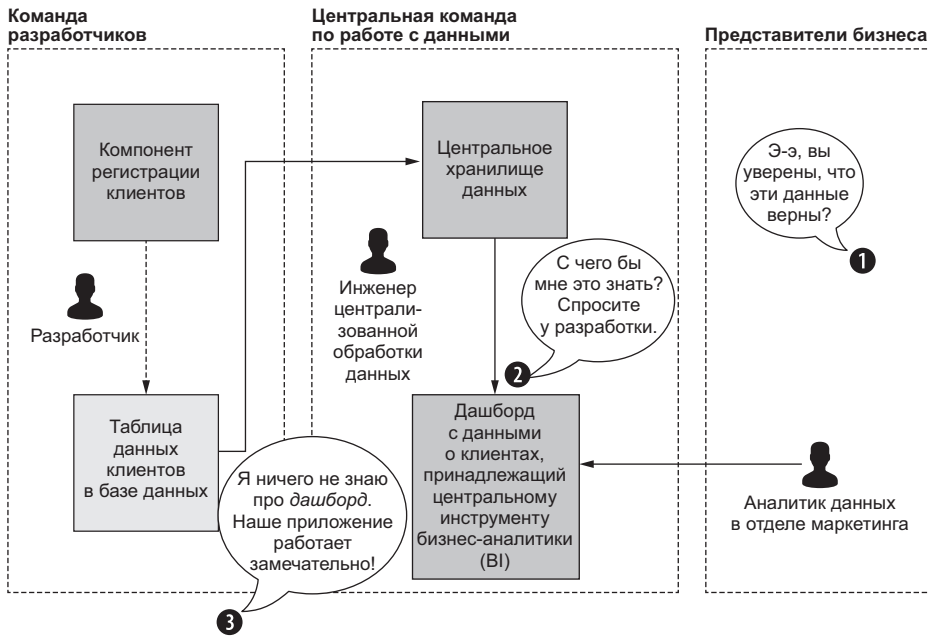


Рис. 1.1. Децентрализованная выработка данных и их централизованное преобразование создают проблемы для пользователей, например, из-за нечеткого понимания, кому принадлежат те или иные данные и кто отвечает за их качество

Поскольку разработчики рассматривают данные как побочный продукт, владение данными неявно приписывается команде по работе с ними. Но такие центральные команды обычно не могут быть в курсе бизнес-знаний, специфичных для разных областей деятельности. Разработчик, который отвечает за регистрацию клиентов, должен знать только язык программирования, на котором написан компонент, а также ориентироваться в обновлениях самого этого компонента и бизнес-процессов, которые с ним связаны. Но центральной команде по работе с данными потребуется также хорошо разбираться в предметной области (домене) для каждого из задействованных доменов.

При такой чрезмерной нагрузке маловероятно, что центральная команда освоит хотя бы один домен в такой же степени, как это делает специализированная группа разработчиков. В результате команда по работе с данными не может сказать, верны ли данные, что они на самом деле означают или в чем смысл тех или иных показателей.

Смена парадигмы на сетку данных требует децентрализовать ответственность за данные, рассматривая их как реальный продукт. Ситуацию, которая представлена на рис. 1.1, можно трансформировать в сетку данных, если, например, команда разработчиков начнет предоставлять продукт данных напрямую аналитикам через стандартизированный порт данных. Этот продукт может быть довольно

простым, например обычным файлом CSV, который размещен в подходящем облачном хранилище так, чтобы у аналитиков был удобный доступ к нему. Такую трансформацию иллюстрирует рис. 1.2.

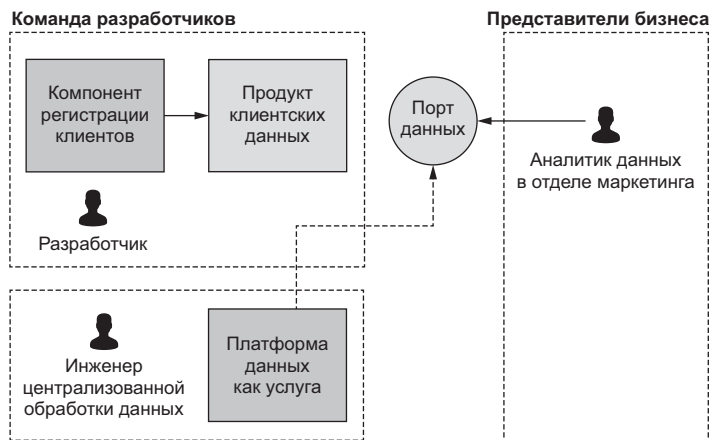


Рис. 1.2. Децентрализованные данные лучше отвечают потребностям потребителей, предлагая простой доступ к четко описанным данным

Тут могла бы помочь *команда платформы*, которая предоставит простую технологию как услугу, чтобы с ее помощью разработчики быстро развертывали подобные продукты данных, включая порты данных. Производители данных сосредотачиваются на том, чтобы создавать *продукты данных*, которые вместе с *потребителями данных* начинают образовывать связи и формируют сеть. Мы называем такую сеть *сеткой*, где отдельные *узлы* — это продукты данных и потребители.

Даже в этом небольшом примере мы наблюдаем значительное операционное изменение в парадигме. К нему относится как перенос ответственности за владение данными (от центральной команды по работе с данными к разработчикам), так и техническая задача — обеспечить работоспособность новой структуры.

Изменения действующей парадигмы вызовут последствия в самых различных областях вашего бизнеса. Чтобы эти последствия не привели к полному хаосу, необходимы некие руководящие принципы. Мы кратко представим их далее в этой главе и подробно рассмотрим в главах 4–7. Но перед этим следует определить наше понятие *сетки данных* и обозначить ее нетехнические аспекты. Жамак Дехгани (Zhamak Dehghani) приложила невероятные усилия, чтобы сформировать идею сетки данных в 2019 году (см. «Как перейти от монолитного озера данных к распределенной сетке данных», <http://mng.bz/nNaK>). Она описала все важнейшие элементы сетки и представила структурированный подход к смене парадигмы, которая обсуждалась ранее.

С тех пор как Дехгани предложила свою концепцию сетки данных, появилось множество примеров, вдохновленных этой концепцией, — как в бизнесе, так

и в теоретических разработках. Не все в этих разработках идеально вписывается в первоначальное понятие сетки данных. Похоже, что большинство компаний даже не могут с уверенностью сказать, что именно соответствует определению сетки данных, а что нет.

В нашей деятельности, как и в этой книге, мы предпочитаем решения, которые в первую очередь применимы на практике (отсюда и название — «Data mesh (сетка данных) в действии»). Поэтому мы постарались сформулировать такое определение сетки данных, которое было бы емким и функциональным и делало бы акцент на мерах по *децентрализации*, которые позволяют максимизировать *ценность*, извлекаемую из данных.

ОПРЕДЕЛЕНИЕ *Сетка данных* — это парадигма децентрализации. В ней децентрализуется владение данными, их преобразование в информацию, а также их обслуживание. Ее цель — повысить извлечение ценности из данных, устранив узкие места в потоке конверсии данных в ценность. Понятие сетки данных опирается на четыре принципа (описанные в разделе 1.4 и в главах с 4-й по 7-ю), которые помогают эффективно масштабировать работу с данными: владение доменом, представление данных как продукта, федеративное вычислительное управление и самообслуживаемая платформа данных. Реализации сетки данных могут различаться по объему и степени использования каждого принципа.

Цель внедрения сетки данных — извлечь наибольшую ценность из активов данных компании. Именно в свете этой цели мы стремились к тому, чтобы определение оставалось простым и позволяло в каждом конкретном случае решать, в какой степени следовать тому или иному принципу. Надеемся, что следующий доступно изложенный пример использования сетки данных объяснит, что мы имеем в виду.

1.2. ЗАЧЕМ НУЖНА СЕТКА ДАННЫХ

Мы видим три основные причины, по которым индустрия данных нуждается в децентрализации в виде сетки данных:

- По мере того как количество источников и потребителей данных увеличивается, центральная команда по их обработке создает организационное узкое место.
- Когда существует много технических систем, которые вырабатывают и потребляют данные, центральное монолитное хранилище этих данных создает техническое узкое место, в результате чего большая часть информации теряется на этом промежуточном этапе.
- Как качество данных, так и владение ими регламентированы неявно, из-за чего возникает путаница и теряется контроль над тем и другим.

За последние 30 лет большинство архитектур данных разрабатывались так, чтобы интегрировать множественные источники данных: центральные команды

по работе с данными объединяли данные, которые поступали из разнообразных источников, и предоставляли «причесанные» наборы данных пользователям, которые, в свою очередь, пытались применять их с пользой для бизнеса. Однако уже более 10 лет проблема «похмельного синдрома» от увлечения большими данными отравляет жизнь компаниям любого масштаба. Эти среды данных страдают от того, что решения плохо масштабируются, информация оказывается неполной, возникают проблемы доступности и т. п. Наверняка и вашей компании нелегко извлекать ценность из данных, верно?

Похоже, что некоторые подходы просто не работают. Десятки отчетов и дашбордов кажутся бесполезными по сравнению с затратами на их создание и поддержку. Многие проекты в области data science застревают на стадии *prototipa*, а действующие приложения, в которых интенсивно используются данные, по-видимому, испытывают с ними массу проблем. По крайней мере, такое впечатление возникает, если учитывать, сколько усилий необходимо приложить, чтобы заставить программный компонент работать.

Одна из причин проблемы с масштабируемостью состоит в том, что количество источников и потребителей данных быстро увеличивается. Очевидное узкое место возникает, когда одна центральная команда по работе с данными владеет и управляет данными на всех этапах: когда данные поступают, преобразуются и приводятся в порядок, а также когда они предоставляются потенциальным пользователям. Если распределить команду по разным участкам конвейера данных, это тоже мало помогает. Когда специалисты, которые отвечают за поступление данных, что-то меняют, они должны сообщить об этом группе, которая занимается их преобразованием, иначе следующие участки конвейера могут дать сбой или, что еще хуже, неправильно обработать данные. Необходимость тесного сотрудничества между специалистами приводит к жесткому сцеплению (*coupling*) всех систем, которые относятся к обработке данных.

Другая проблема возникает из-за монолитной природы платформ данных, таких как хранилища и озера. Им часто не хватает разнообразия, чтобы отразить реальность, представленную в данных, которые поступают из источников и структур, определяемых доменом. Более того, когда структуры данных принудительно приводятся к стандартному формату, становится сложнее извлекать ценную информацию из собранных сведений, потому что на централизованных платформах теряются важные знания, специфичные для домена.

Подобная проблема наблюдалась в одном из проектов, над которым мы работали. Компания по производству автозапчастей покупала сведения о неисправностях различных деталей. Хотя поставщик обладал информацией о происхождении деталей (в какой модели автомобиля они были установлены), у покупателя не было моделей данных, которые позволяли бы хранить эту информацию. В результате компоненты анализировались по отдельности, что мешало специалистам по НИОКР лучше понять общую картину.

Еще два взаимосвязанных фактора усугубляют описанные ранее проблемы. Один из них — нечеткая структура владения данными, а другой — ответственность за качество данных. По мере того как данные проходят через специализированные команды, они утрачивают связь со своим бизнес-назначением. Разработчики централизованных систем обработки данных и приложений не могут и не будут полностью понимать смысл данных, а качество данных невозможно оценивать в отрыве от их смысла.

Похожие проблемы были обнаружены и в других областях разработки ПО, в результате чего появились (и завоевали успех!) предметно-ориентированное проектирование и микросервисы. Когда аналогичное мышление (с акцентом на владение данными и общий инструментарий) стало применяться в области инженерии данных, в итоге удалось сформулировать понятие сетки данных.

1.2.1. Альтернативные модели

Вместо того чтобы децентрализовать ответственность за данные, как предлагает сетка данных, гипотетически можно рассмотреть две альтернативные модели. Мы обсудим их подробнее в главе 6, а здесь дадим краткое описание.

Первый вариант — централизовать как сотрудников, так и технические решения. С такого построения начинается любой стартап, и это вполне приемлемая архитектура по умолчанию — так же, как монолитная архитектура приемлема по умолчанию для программных компонентов. На начальном этапе затраты на децентрализацию перевешивают ее преимущества, а работа значительно упрощается потому, что у нас есть единая команда по работе с данными и единое техническое решение.

ВАЖНО Централизация — это разумный стандартный вариант работы с данными как с организационной, так и с технической стороны. Децентрализация сопряжена с издержками, а централизация может их снизить. Однако этот вариант подразумевает, что из централизованных данных мы получаем примерно такую же ценность, как из децентрализованных.

Вторая альтернатива — разделять работу не по бизнес-доменам, как предлагает сетка данных, а по техническим решениям. Обычно это приводит к тому, что одна главная команда инженеров данных отвечает в основном за то, чтобы принимать данные и предоставлять инфраструктуру для их хранения, а несколько других команд (например, аналитики и специалисты по data science) отбирают необработанные данные, чтобы затем превратить их в нечто значимое. Можно сначала централизовать систему данных, а потом использовать эту схему, чтобы увеличить поток.

В этих двух подходах нет ничего плохого. Они могут быть приемлемыми вариантами по умолчанию, однако ни тот ни другой не помогают создавать ценность, которая тесно связана с бизнес-доменами. Ни один из вариантов не справится с ситуацией, когда в каком-то одном домене что-нибудь внезапно меняется. Как и в случае с микросервисами, которые позволяют быстро извлекать ценность

из одного конкретного сервиса, масштабируя его в отдельности, сетка данных способна масштабировать извлечение ценности даже в одном домене. Альтернативные варианты требуют масштабировать всю систему, чтобы удавалось извлекать больше ценности всего в одном домене.

Так или иначе оба этих варианта в какой-то момент исчерпают свои возможности: окажется, что добавить очередной источник данных или проект data science окажется чрезвычайно сложно и дорого по сравнению с предыдущими добавлениями. Именно в этот момент имеет смысл перейти на сетку данных.

1.2.2. Хранилища и озера данных внутри сетки данных

С сеткой данных связано распространенное заблуждение: иногда ее воспринимают как единственную альтернативу центральному озеру или хранилищу данных. Но это не отражает самой сути сетки данных, в которой сочетаются две составляющие — технология и организация. Сетка данных — это альтернатива ситуации, когда одно централизованное подразделение занимается данными в рамках одного центрального хранилища данных.

Сетка данных позволяет по-прежнему поддерживать центральное хранилище данных, однако обрабатывать данные и владеть ими будут децентрализованные подразделения. Такая реализация распространена в компаниях, которым не нужна максимальная гибкость на стороне производителей данных. Этот подход также часто применяется, чтобы сохранять озера и хранилища данных в рамках групп бизнес-аналитики (BI) или data science. В этом случае озера и хранилища данных становятся узлами в сетке данных (рис. 1.3).

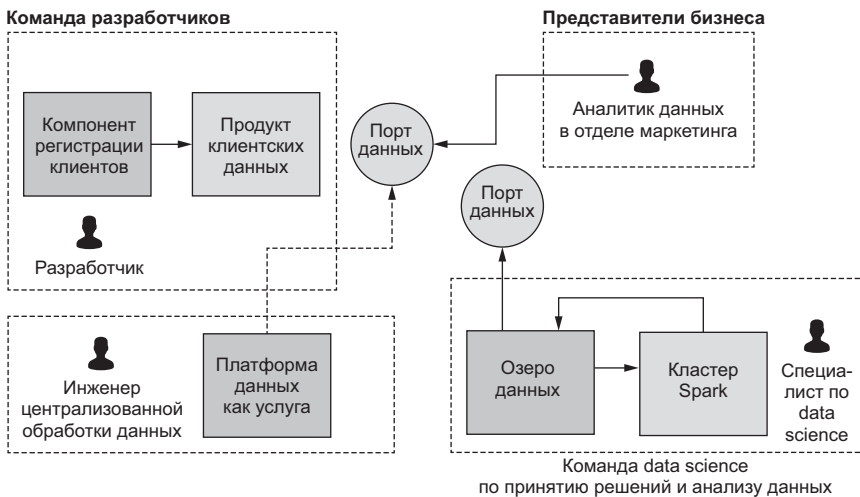


Рис. 1.3. Сетки данных могут по-прежнему использовать озера данных: например, команда data science, которая создает продукты данных, задействует озера данных в качестве узлов внутри сетки

Сетки данных активно используют как озера данных, так и хранилища данных в различных форматах. Как правило, сетки данных не пытаются привязываться к какой-либо конкретной технологии. Мы подробнее обсудим эту дихотомию в разделе 1.6, а пока давайте взбодримся и сосредоточимся на преимуществах сетки данных.

1.2.3. Преимущества сетки данных

Давайте проанализируем потенциал внедрения сетки данных с точки зрения лиц, которые принимают бизнес-решения, а также с точки зрения технолога.

С точки зрения бизнеса

С точки зрения бизнеса данные сами по себе не представляют особой ценности, более того, они означают издержки! Похоже на ересь? Чтобы понять это утверждение и при необходимости донести его до своих бизнес-партнеров, необходимо представлять себе, на каких уровнях люди могут осознавать реальность.

Хорошей иллюстрацией этого феномена служит пирамида DIKW, которая опирается на пьесу Т. С. Элиота «Камень», написанную в 1934 году. Эта модель представляет данные, информацию, знания и мудрость (DIKW, Data — Information — Knowledge — Wisdom) в виде иерархической структуры, где каждый следующий элемент является производным от предыдущего. (См. «Раскрываем тайны данных: модель DIKW» Энтони Фигероа (Anthony Figueroa), <http://mng.bz/v60M>.)

Данные в этом контексте — это просто набор значений (хранение которых стоит денег!). Чтобы извлечь из них пользу, нужно выстроить контекст, который позволит принимать обоснованные решения. Благодаря сетке данных вся пирамида становится надежнее.

Как мы уже упоминали, сырые данные не представляют ценности для тех, кто принимает решения. Можно возразить, что эти люди могут загрузить данные на свои рабочие станции и проанализировать самостоятельно. Это так, однако этот вариант базируется на двух основных допущениях:

- Чтобы скачать данные, они должны быть доступны.
- Чтобы анализ приносил пользу, данные должны быть как можно более полными.

Что касается первого допущения: мы уже говорили и повторим еще раз, что сетка данных в значительной степени ориентирована на то, чтобы сделать данные доступными. (А также чтобы их было легко находить, интегрировать и переиспользовать.) Эта доступность заложена в одном из четырех принципов сетки данных: принцип «*данные как продукт*» посвящен тому, чтобы данные были готовы к использованию.

Полнота данных — это еще одно качество, в котором сетка данных проявляет себя с лучшей стороны. В отличие от большинства архитектур хранилищ или

озер данных, продукты данных и соответствующие модели данных — это не то, что IT-специалисты разрабатывают в отрыве от бизнеса. Разработка становится плодом совместных усилий, в результате чего данные, которые представлены за пределами домена, оказываются достаточно для обоснованных выводов.

Сетка данных также помогает повысить ценность компонентов, расположенных выше в иерархии. Команды, которые преобразуют данные в информацию, знания и мудрость (бизнес предпочитает называть ее *инсайтом*), получают мгновенный доступ к многочисленным источникам данных с высокой совместимостью.

Конечно, для озера данных теоретически тоже можно обеспечить такой немедленный доступ. Однако в реальности, по нашему опыту, это обычно неосуществимо в условиях, когда одна команда управляет техническими аспектами среды, а также правами доступа и передачи данных. А если нужные фрагменты данных хранятся в двух разных озерах (или в четырех, что встречается не так уж редко), заставить их работать вместе практически нереально. Одним словом, если налажен доступ к продуктам данных, которые оптимизированы для чтения, это позволяет быстро прототипировать новые аналитические методы и открывает путь для динамичного развития новых бизнес-возможностей.

С точки зрения технологии

Основное преимущество сетки данных с технической точки зрения заключается в том, что, как мы упоминали ранее, она позволяет сохранять темпы разработки по мере роста организации, а это необходимо, чтобы данные продолжали приносить коммерческую выгоду. Сетка данных устраняет недостатки других архитектур, таких как хранилища или озера данных, благодаря тому, что она децентрализует производство данных и управление ими. Для других архитектур характерно узкое место — центральная команда, которая отвечает за то, чтобы приводить в порядок все данные для целой компании и готовить их к использованию. Таковую единственную команду нельзя масштабировать, чтобы удовлетворить разнообразные потребности растущей организации, связанные с данными. И технические решения, и знания команды быстро превращаются в препятствия для масштабирования. В итоге все больше времени тратится на то, чтобы сопровождать имеющиеся решения, а новые проекты все чаще откладываются.

Еще одно преимущество сетки данных — четко определенное владение данными непосредственно с момента их создания. Этот подход упрощает структуру управления данными, оставляя лишь тонкий слой команды федеративного управления, причем ее деятельность ограничивается тем, чтобы согласовывать стандарты в рамках автономных доменов.

Разработка ускоряется также из-за того, что расширяются возможности команд внедрения. Поскольку они теперь отвечают за то, чтобы создавать и поддерживать продукты данных, скорость изменений больше не ограничивается пулом задач одной центральной команды по интеграции. Поэтому продукт данных быстрее развивается и избавляется от недостатков — особенно это касается

исправления ошибок и простоев. Кроме того, команда, которая владеет продуктом данных, способна быстрее реагировать на изменения, потому что ей не требуется переключать контекст, как единой центральной команде.

Еще один фактор, о котором стоит упомянуть, — это стабильность среды данных. Поскольку продукты данных предоставляют доступ к сокращенным версиям своих наборов данных, построенные на их основе конвейеры оказываются намного надежнее и требуют гораздо меньше усилий по сопровождению.

1.3. СЛУЧАЙ ИЗ ПРАКТИКИ: СНЕГОУБОРОЧНЫЙ БИЗНЕС

Кэндис управляет предприятием по уборке снега. Эта бизнес-леди начала собственное дело с того, что каждую зиму сама убирала снег. Через пару лет она расширила свое предприятие. Она сосредоточилась на логистике, бухгалтерии и ценообразовании и наняла трех сотрудников: Адама, который чистит дома в Сосновом проезде, Еву, которая обслуживает Дубовую улицу, и Боба, который заказывает новые лопаты, потому что они постоянно ломаются.

Задача Адама и Евы — не только убирать снег, но и привлекать новых клиентов на своих улицах. В конце концов, они проводят довольно много времени среди местных жителей, каждую зиму расчищая снег! Но Кэндис недовольна. На рис. 1.4 показано, как Кэндис поначалу управляла бизнесом.

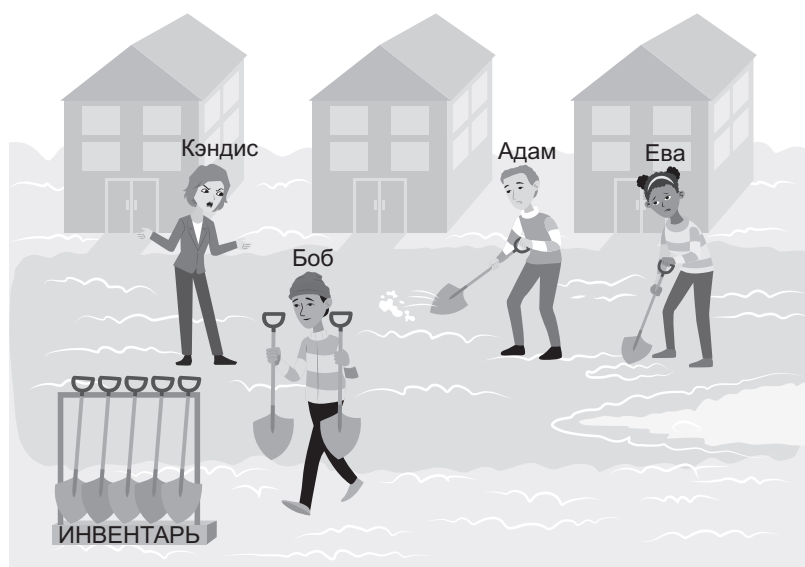


Рис. 1.4. В снегоуборочном бизнесе Кэндис, Адам и Ева выполняют свою работу, а Боб обеспечивает запас инвентаря, замораживая капитал

В прошлом году Кэндис попросила Адама и Еву записывать время работы и количество обслуженных домов. Она поместила эти данные в импровизированный файл Microsoft Excel, чтобы произвести некоторые расчеты и установить цены. Соответствующий поток данных показан на рис. 1.5.

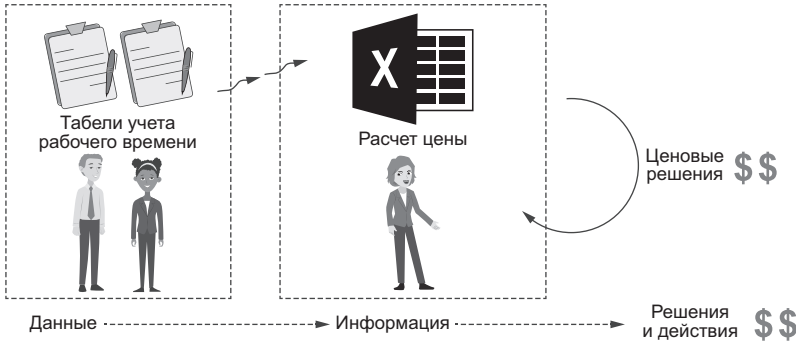


Рис. 1.5. Централизованный поток данных от Адама и Евы к Кэндис для принятия решений

Таким образом, можно сказать, что Кэндис превратила данные, полученные от Адама и Евы, в *информацию* в своем файле Excel, а затем преобразовала ее в *решения*, благодаря которым появилась бизнес-ценность. Кроме того, она попросила Адама и Еву приблизительно оценить, сколько лопат им понадобится. На рис. 1.6 показан поток данных, направленный к Бобу.

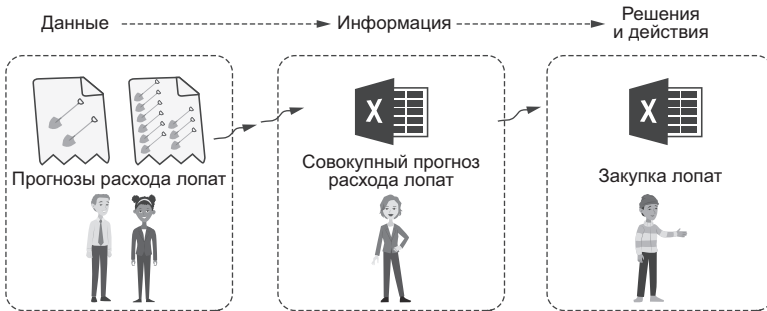


Рис. 1.6. Централизованный поток данных от Адама и Евы через Кэндис к Бобу

Можно сказать, что Адам и Ева предоставили Кэндис необработанные данные, а она агрегировала их в информацию, которую передала Бобу, чтобы тот превратил данные в решения и бизнес-ценность. Обратите внимание, что здесь мы снова видим *конвейер* — последовательность шагов, которые превращают данные в ценность. В этом конвейере есть два этапа, на которых устанавливаются цены (Адам и Ева собирают данные в своих табелях, а Кэндис агрегирует эти данные

и принимает решения на их основе), и три этапа, относящиеся к закупке Бобом лопат (Адам и Ева производят данные с помощью своих прогнозов расхода лопат, Кэндис агрегирует эти данные, а Боб принимает решения на их основе).

Но как мы уже отмечали, Кэндис недовольна сложившейся ситуацией. Прибыль не очень высока, а лопаты, похоже, закупаются неэффективно: иногда накапливается слишком много инвентаря, который не используется, а иногда Адаму и Еве приходится откладывать свою работу, потому что у них заканчиваются лопаты.

В этом году, прочитав книгу «Data mesh в действии», Кэндис решила провести эксперимент. Она собирается *создать основу для сетки данных*, как описано в главе 3 этой книги. Знания из главы 4 помогают Кэндис определить границы доменов и передать владение данными основным владельцам домена уборки снега, то есть Адаму и Еве. Это означает следующее:

- Кэндис прекращает вести учет времени и вместо этого поручает Адаму и Еве записывать время самостоятельно любым удобным для них способом.
- Адам и Ева будут устанавливать цены на своих улицах.

Затем происходит нечто любопытное: Ева решает повысить цены на Дубовой улице, а Адам снижает цены в Сосновом проезде. Оказывается, они просто знают о своих районах больше, чем Кэндис. На Дубовой улице к домам ведут длинные подъездные дорожки, поэтому Еве имеет смысл брать больше за уборку снега. А в Сосновом проезде местный паренек убирает снег дешевле, поэтому, чтобы оставаться конкурентоспособным, Адам вынужден снизить свои расценки. На рис. 1.7 показан децентрализованный поток данных на текущий момент времени.

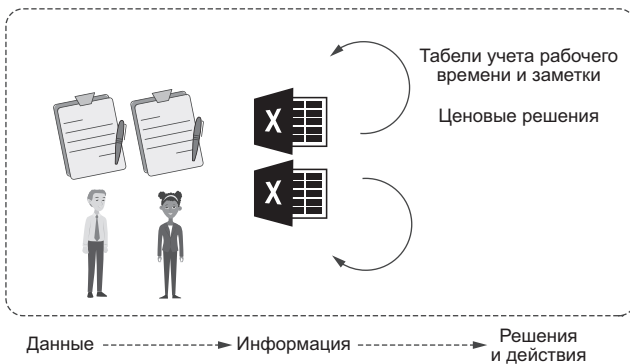


Рис. 1.7. Децентрализованный поток данных внутри доменов Адама и Евы

Если теперь посмотреть на поток данных, то можно убедиться, что он полностью остается у Адама и Евы: от данных к информации и, наконец, к решению — к модели ценообразования. По мере того как растет прибыль, Кэндис решает пойти еще дальше и на второй год эксперимента берется за проблему закупок. Чтобы ее решить, она поручает Адаму и Еве напрямую сообщать Бобу о том, сколько лопат им, скорее всего, понадобится.