

Глава 4

Основы XBRL

В этой главе...

- Более глубокое описание XBRL
- На что похож XBRL
- Техническая сторона XBRL
- Что должен знать каждый бизнес-пользователь

В этой главе мы рассмотрим физическое содержание языка XBRL и расскажем о тэгах и угловых скобках. Мы представляем XBRL в виде слоев или уровней, каждый из которых строится на основе предыдущего. Однако не обязательно глубоко понимать XBRL на каждом уровне. Бизнес-пользователи взаимодействуют с XBRL с помощью программного обеспечения, а оно должно скрывать технические аспекты XBRL. Мы не собираемся учить вас использовать программное обеспечение — мы даем твердые знания о XBRL. Таким образом, выбирая инструменты, вы будете достаточно много знать о XBRL, чтобы быстро освоить любой его инструмент.

В этой главе мы не пытаемся описывать все нюансы XBRL. Это было бы слишком утомительным для вас. Вместо этого мы сосредоточимся на важных деталях, которые действительно необходимо знать. Мы заложим прочный фундамент, который послужит основой вашего дальнейшего роста. Мы даже укажем на соответствующие ресурсы, которые можно использовать для более глубокого изучения деталей, если у вас появится такое желание.

Подготовка к фестивалю фанатов XBRL

В этой главе мы покажем фрагменты базовых таксономий и отчета XBRL, которые можно найти по следующим адресам.

- ✓ **Таксономия XBRL.** <http://xbrl.squarespace.com/storage/examples/Example.xsd>
- ✓ **Отчет XBRL.** <http://xbrl.squarespace.com/storage/examples/Example-instance.xml>



Для того чтобы избежать набора длинных адресов, зайдите на сайт <http://www.dummies.com/go/xbrl>. Вы окажетесь на домашней странице, где расположены все необходимые ссылки.



Мы хотим, чтобы фрагменты кода были минимальными и как можно более простыми. Фрагменты кода в этой главе должны помочь вам лучше разобраться в конкретной ситуации. Все фрагменты в этой главе взяты из базовых отчетов. Эти примеры должны дать вам представление об отчете и таксономии XBRL.

Структура XBRL

Спецификация XBRL не описывает логическую модель. Однако на высших уровнях XBRL можно предположить существование некой логической модели. Впрочем, поскольку мы планируем погрузиться в детали XBRL, нас в первую очередь будет интересовать физическая модель XBRL.

Ядро XBRL — это спецификации XBRL 2.1 <http://www.xbrl.org/SpecRecommendation>. На основе спецификации XBRL создаются модули, обеспечивающие дополнительную функциональность. В зависимости от своих потребностей вы можете выбирать тот или иной модуль XBRL. В этой главе мы придерживаемся базовой спецификации XBRL. (Другие модули рассматриваются в главе 16.)

Спецификация XBRL образует структуру, которая делит XBRL на две главные части.

- ✓ **Таксономии XBRL**, являющиеся схемами XML, определяют концепты и представляют собой управляемый словарь, используемый отчетами XBRL и базами ссылок XLink, которые обеспечивают дополнительную информацию о концептах.
- ✓ **Отчеты XBRL** содержат факты и контекстную информацию об этих фактах.

Мы разделим наше объяснение XBRL на две части: таксономии и отчеты XBRL. Таксономия XBRL содержит намного больше компонентов, чем отчет XBRL. Но таксономии и отчеты XBRL взаимодействуют друг с другом. Мы объединим две эти части структуры XBRL и объясним, как они взаимодействуют.

После изложения основных концепций XBRL мы погрузимся в детали, из которых состоят таксономия и отчет XBRL. Мы опишем только наиболее важные детали, не перегружая вас лишними подробностями, которые вам вряд ли понадобятся.

Важным свойством структуры XBRL является его гибкость, обеспечиваемая расширяемостью XBRL. Мы покажем, как таксономии XBRL обеспечивают дополнительные возможности, которые используются отчетами XBRL. После формулировки ключевой идеи мы погрузимся в ключевые концепты, чтобы расширить ваш уровень понимания в довольно специфичных областях. В заключение мы укажем другие главы, которые расширяют информацию, изложенную в этой главе.



XBRL для фанатов XML

Если вам посчастливилось освоить язык XML, то для вас будут полезными следующие советы.

- ✓ В XBRL существует XML Schema (см. www.xbrl.org/2003/xbrlinstance-2003-12-31.xsd), которая регламентирует создание таксономий и отчетов XBRL.
- ✓ Другая схема описывает, как создаются базы ссылок XBRL (см. <http://www.xbrl.org/2003/xbrl-link-base-2003-12-31.xsd>).
- ✓ Если вам известны традиционные подходы к созданию языков семейства XML и при этом вы впервые видите язык XBRL, то может возникнуть путаница. XBRL не просто строит традиционную схему XML, объясняющую, как создаются отчеты. XBRL строит слой над XML, который используется для создания метаязыка, который, в свою очередь, частично используется для создания схемы XML.
- ✓ В языке XBRL схема таксономии (представляющая собой часть таксономии XBRL) играет ту же роль, что и язык XML Schema. Схема таксономии описывает элементы и атрибуты XML, аналогично традиционному языку XML. Однако XBRL добавляет метаданные, необходимые для деловой отчетности, например, информацию о периоде, информацию о том, является ли концепт датой, и т.д. Эти дополнительные метаданные широко используются в деловой отчетности, поэтому XBRL добавляет их в схему. Схема таксономии — это язык XML Schema, выражающий способ создания отчетов XML в XBRL.
- ✓ Другой частью таксономии XBRL является база ссылок. Используя базы ссылок в таксономии XBRL можно добавлять метаданные. Хотя это и не обязательно, базы ссылок обеспечивают информацию, полезную для понимания таксономии XBRL и использования отчета XBRL. Базы ссылок позволяют выражать многие виды семантики.
- ✓ Концепты XBRL — это элементы языка XML Schema с конкретным значением атрибута `substitutionGroup`. Из-за особенностей атрибута `substitutionGroup` в языке XML Schema все концепты XBRL являются глобальными. Благодаря этому элементы схемы таксономии представляют собой простой список. С помощью баз ссылок в эту простую схему можно добавлять любое количество иерархий.
- ✓ Пользователь XBRL должен работать с помощью процессора XBRL, а не только синтаксического анализатора XML. В большинстве отчетов вы можете сойти с ума, пытаясь использовать XBRL только с синтаксическим анализатором XML.

Изложение основных принципов XML для пользователей XBRL

Поскольку XBRL — это язык семейства XML, мы должны немного рассказать о расширяемом языке разметки (eXtensible Markup Language — XML). XML предназначен для создания языка разметки, которым является XBRL. XBRL — это язык семейства XML. Язык XBRL использует другие части семейства XML, включая пространства имен, XML Schema, XML Base, XLink, XPath и XPointer.

Если хотите подробно узнать о XML, рекомендуем книгу Лючинды Дайкс (Lucinda Dykes) и Эда Титтеля (Ed Tittel) *XML для чайников* (Wiley).

Составные части таксономии XBRL

Таксономии XBRL состоят из различных физических частей и выражают концепты, ресурсы и взаимосвязи (см. главу 3). Эти взаимодействующие части, описанные в следующих разделах, обеспечивают функциональные возможности, необходимые для выражения смысла обмениваемой бизнес-информации.

Схемы таксономии и базы ссылок

Таксономии XBRL состоят из двух частей.

- ✓ **Схемы таксономии** — это часть языка XML Schema, входящая в таксономию XBRL. Схемы таксономии содержат определения концептов, принимающие форму элементов языка XML Schema. Например, бизнес-концепт Cash может быть элементом языка XML Schema, у которого есть значение атрибута Cash и другие атрибуты.
- ✓ **Базы ссылок** — это часть языка XLink, входящая в таксономию XBRL и представляющая собой XML-документы. Термин “база ссылок” (linkbase) является сокращением от названия “базы данных ссылок”. *Базы ссылок* — это физические сущности, используемые для выражения логических сущностей — сетей. Есть два типа сетей: ресурс и взаимосвязь. (Термины “сеть”, “ресурс” и “взаимосвязь” рассматриваются в разделе “Сети и расширенные ссылки”.) Сети ресурсов и взаимосвязей выражаются с помощью синтаксиса языка XLink в виде *расширенной ссылки* (extended link).



Расширенные ссылки — это контейнеры, в которых хранятся данные, содержащиеся в базах ссылок. Мы не собираемся углубляться в техническое обсуждение расширенных ссылок; если хотите, прочитайте спецификацию языка XLink по адресу <http://www.w3.org/TR/xlink>. Более подробно язык XLink и базы ссылок рассматриваются в главе 22.

Связанный набор таксономий

Отдельная таксономия XBRL может состоять из набора схем таксономий и баз ссылок. У этого набора есть конкретное и важное имя в языке XBRL: *связанный набор таксономии* (discoverable taxonomy set — DTS).

Набор DTS управляется правилами поиска, определенными в спецификации XBRL, которую понимают процессоры XBRL. DTS может содержать любое количество схем таксономии и/или баз ссылок и начинаться с любой схемы таксономии, базы ссылок или отчета XBRL.

Сети и расширенные ссылки

Сети — это логический аспект XBRL, физически выражаемый как набор баз ссылок. Базы ссылок существуют в пределах физической модели и являются коллекциями

расширенных ссылок. Расширенные ссылки в языке XLink работают немного иначе, чем в языке XBRL. В XLink все расширенные ссылки физически отделены друг от друга, а в XBRL к расширенной ссылке добавлен ролевой атрибут. Сеть — это коллекция всех расширенных ссылок определенного типа с одной и той же ролью расширенной ссылки (extended link role). Роль расширенной ссылки — это уникальный идентификатор, выраженный как атрибут роли расширенной ссылки.



Сети используются для разделения и организации ресурсов и взаимосвязей, имеют разные роли расширенных ссылок и разные уникальные идентификаторы, создающие разделение. Сетевые роли и роли расширенных ссылок — это одно и то же. Создатели таксономии XBRL определяют эти роли в пределах схем таксономии.

Ресурсы и взаимосвязи в пределах сетей

С физической точки зрения сети разделяются на два типа в зависимости от типов расширенных ссылок, которые они используют.

- ✓ *Сети ресурсов* (resource networks) предоставляют дополнительную информацию о концепте, которая имеет форму ресурса XLink. Из пяти стандартных типов баз ссылок ярлыки и ссылки представляют собой ресурсные базы ссылок, выражающие сети ресурсов. Ресурсы могут иметь разные роли, позволяющие разделять ресурсы на категории. Примером дополнительной информации является англоязычный ярлык Property, Plant and Equipment со стандартной ролью, которая связана с концептом имени PropertyPlantAndEquipmentwithin в схеме таксономии. Примером ссылки является указатель URL на руководство по бухгалтерскому учету или некоторая описательная информация о главе, разделе, подразделе или странице этого руководства.
- ✓ *Сети взаимосвязей* (relation network) выражают взаимосвязи между концептами, используя дуги XLink. Существует пять стандартных видов баз ссылок, из них базы ссылок презентаций, вычислений и определений образуют сети. У взаимосвязей (выраженных как дуга XLink) могут быть разные роли, позволяющие разделить их на категории. Примером сети взаимосвязей является сеть презентаций с сетевой ролью Balance Sheet (Бухгалтерский баланс), которая означает, что концепт PropertyPlantAndEquipmentis является частью данной роли и связан с концептом Assets (Активы).

Идентификация частей отчета XBRL

По сравнению с таксономией XBRL отчеты XBRL выглядят просто. Это отдельные физические файлы, которые содержат следующие части: ссылки на информацию DTS, факты, контексты, единицы измерения и сноски. Все эти компоненты помогают использовать эту бизнес-информацию, выраженную в ваших терминах.

Мы советуем прочитать главу 3. Объяснения каждой из этих частей приводятся в разделе “Отчеты XBRL”. Еще более подробному объяснению отчетов XBRL посвящена глава 18.

Обеспечение гибкого обмена бизнес-информацией

Таксономии и отчеты XBRL обеспечивают обмен бизнес-информацией — основную цель XBRL. Ключевые взаимосвязи между компонентами в этой главе имеют огромное значение для понимания XBRL, которое помогает провести разделительную линию между тем, как заставить XBRL удовлетворить ваши потребности, и борьбой с XBRL на каждом шаге на этом пути. В этом разделе объясняется как логический смысл основных идей, так и их физическая реализация.

Определение концептов и организация схем таксономии

Концепты определяются в схемах таксономии, которые на самом деле являются схемами XML. Концепты XBRL — это элементы языка XML Schema. В основном язык XBRL дополняет средства языка XML Schema средствами определения концептов XBRL. Кроме того, язык XBRL добавляет к элементам языка XML Schema несколько атрибутов.

Схемы таксономии могут ссылаться на другие схемы таксономии, определенные ранее. XBRL использует механизмы языка XML Schema (импорт и включение) для объединения наборов концептов, описанных в отдельных физических файлах. XBRL также обеспечивает механизмы (`schemaRef` и `linkbaseRef`) для объединения компонентов DTS. Все эти схемы таксономии становятся частью DTS. Эти механизмы позволяют строить модульные схемы таксономии. Другой механизм раскрытия схем таксономии основан на связях между базой ссылок и схемой таксономии.

Схемы таксономии могут также ссылаться на ресурс или сети взаимосвязей, которые также становятся частью DTS.

Использование сетей для разделения и организации наборов ресурсов

Сети ресурсов физически содержатся в базах ссылок XLink в форме расширенных ссылок, имеющих определенную роль. Ресурс, определенный в рамках этих сетей, позволяет добавлять информацию к концептам XBRL, определенным в схемах таксономии. XBRL обеспечивает два типа сетей ресурсов в спецификации XBRL: ярлыки (`labels`) и ссылки (`references`). (Дополнительная информация изложена ниже, в разделе “Сети ресурсов”.) Модули XBRL добавляют другие типы ресурсов, такие как информация о форматировании и формулы (см. главу 16). Можно также добавлять собственные типы ресурсов, используя спецификацию обобщенных баз ссылок (см. главу 16). Схемы таксономии или отчеты XBRL могут соединить сеть ресурсов с DTS.

Отделение и организация наборов взаимосвязей с помощью сетей

Базы ссылок XLink физически содержат сети взаимосвязей в форме расширенных ссылок с определенной ролью. Сети взаимосвязей позволяют связывать один концепт с другим разными способами для достижения разных целей. В спецификации XBRL предусмотрены три типа сетей взаимосвязей: презентаций, вычислений и определений. (Дополнительная информация изложена ниже, в разделе “Сети взаимосвязей”.) Можно также добавлять свои собственные типы взаимосвязей, используя спецификацию баз обобщенных ссылок (см. главу 16). Схемы таксономии или отчеты XBRL могут соединить сеть взаимосвязей с DTS.

Обмен фактами с помощью отчетов XBRL

Информация, являющаяся предметом обмена, содержится в отчетах XBRL и выражается в форме фактов. Каждый факт связан с концептом из таксономии XBRL, который выражает этот концепт и либо определяет ее, либо ссылается на определение концепта, внешнего по отношению к таксономии XBRL, с помощью одной или нескольких ссылок XBRL. Концепты связаны с отчетом XBRL, будучи частью DTS. Понять и использовать факты, содержащиеся в отчете XBRL, помогает дополнительная информация, содержащаяся в сетях, связанных с отчетом XBRL. С фактами также связывается контекстная информация, помогающая объяснить, с какой сущностью и каким периодом связан данный факт. Кроме того, числовая информация сопровождается дополнительной контекстной информацией о единицах измерения. Для сообщения дополнительной информации, такой как комментарии, можно использовать сноски XBRL (не путайте их с бухгалтерскими сносками).

Обеспечение гибкости посредством расширения

Способность выражать концепты, ресурсы и взаимосвязи — это, конечно, хорошо, но что делать, если таксономия XBRL неточно отражает ваши желания? Вы можете изменить ее. Именно из-за этой особенности язык XBRL называют расширяемым языком деловой отчетности. Одна из наиболее мощных характеристик XBRL — его динамичность в соответствующих ситуациях. Несмотря на возможность расширения, использовать ее не обязательно.

Необходимость расширять XBRL несколько запутывает синтаксис на физическом уровне. Все эти отдельные компоненты обеспечивают гибкость, необходимую для правильного расширения XBRL. Но есть и хорошие новости — процессоры XBRL скрывают большую часть этого физического синтаксиса от пользователя. С логическим представлением о XBRL работать намного легче.

Детали механизма расширяемости будут изложены в разделе “Обеспечение гибкости посредством расширения”, а пока просто помните, что он существует. Изменить чью-либо схему таксономии или базы ссылок невозможно, но можно создать собственные схемы таксономии или базы ссылок, добавляя или изменяя части DTS.

Можно даже аннулировать то, что существует в чьей-либо таксономии, обозначив ее как ненужную. Кроме того, можно изменить весь DTS, изменяя логически то, что невозможно изменить физически. Далее, пользователи информации, содержащейся в отчете XBRL, могут четко видеть изменения, которые внесли пользователи, потому что информация об этих изменениях сообщается в форме таксономии XBRL, которая является частью DTS.

Достижение совместимости посредством проверки на соответствие

Все это кажется сложным? Деловая отчетность действительно сложная, иногда невероятно сложная. Хорошо, что с этой сложностью сталкиваются компьютеры, а не вы. Программное обеспечение ведет вас через этот процесс, удостоверяется, что вы следуете правилам XBRL, и оказывает другую помощь в достижении ваших целей. Программное обеспечение удостоверяется в том, что все хорошо, проверяя, что ваш код не содержит ошибок. Этот процесс называется *проверкой соответствия* (validation). Существует множество способов совершать ошибки, поэтому существует много способов проверки. (Глава 12 касается ратификации более подробно.)

Срываем покров таинственности с DTS

XBRL требует, чтобы все в отчете XBRL было явно определено. Отчет XBRL должен прямо или косвенно физически ссылаться на все таксономии XBRL, концепты, ресурсы и взаимосвязи, которые влияют на его содержание. Первое, что делает процессор XBRL, сталкиваясь с отчетом XBRL, — пытается обнаружить все связанные с ним таксономии XBRL; если он не может это сделать, процесс останавливается!



Во многих случаях, ссылаясь на таксономии XBRL, люди на самом деле имеют в виду DTS.

Файлы схемы в XML и таксономии XBRL работают по-разному. Таксономии XBRL должны существовать и поддаваться обнаружению. XML может делать подсказки о схеме, или указывать на схему, или вообще не обеспечивать никакой схемы. К языку XBRL это не относится. Схемы, которые являются таксономиями XBRL, обязательно должны существовать, потому что таксономии XBRL позволяют понять информацию, содержащуюся в отчете XBRL. На рис. 4.1 показано, как многочисленные схемы таксономии и базы ссылок объединяются в набор DTS.

На рис. 4.1 отчет XBRL связан с двумя таксономиями XBRL, каждая из которых состоит из одной схемы таксономии и двух баз ссылок. DTS — это объединенный набор всех схем таксономии и баз ссылок обеих таксономий XBRL.



Обратите внимание на то, что отчет XBRL никогда не является частью DTS, но он может содержать ссылки на схемы таксономии, схемы XML и базы ссылок, которые являются частью DTS.

Принципы функционирования сетей

Когда мы говорим о сетях в этой книге, мы не говорим о компьютерных, кабельных или любых других сетях, представляющих собой средство распространения информации. Хотя очевидно, что сеть — это не бухгалтерский или бизнес-термин, это способ описания того, что происходит при обмене бизнес-информацией.

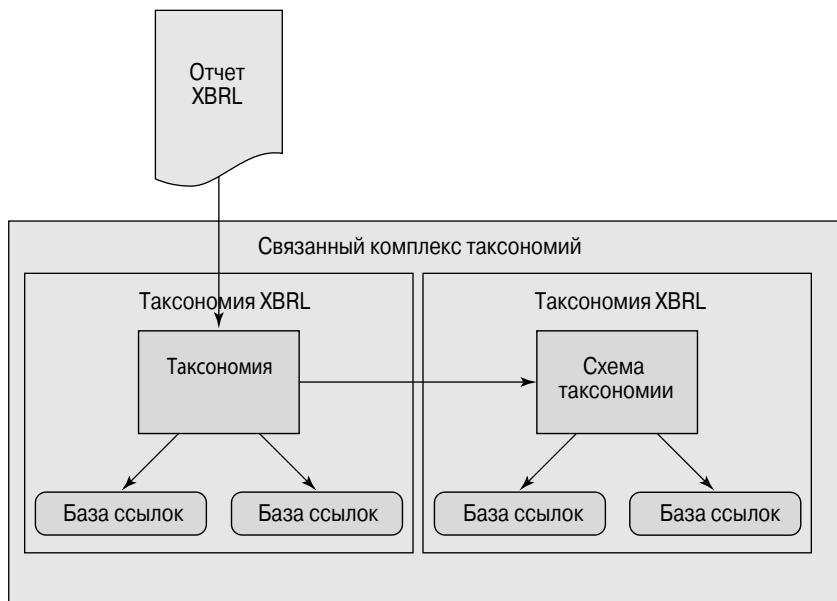


Рис. 4.1. У сетей есть два важных состояния

- ✓ **Неразрешенная, или неструктурированная, форма.** На жаргоне XBRL это состояние называют *базовым набором* (base set). Базовый набор — это реальные физические взаимосвязи, существующие в нечеткой, или неструктурированной, сети. Например, если между двумя концептами выражена взаимосвязь, а затем была создана другая взаимосвязь, которая должна отменить первую, то в базовом наборе будут существовать обе взаимосвязи. Эти две взаимосвязи являются скорее физической формой фактических взаимосвязей. Для того чтобы лучше понять, что мы имеем в виду, рассмотрим противоположную форму.
- ✓ **Разрешенная форма.** На жаргоне XBRL это состояние называют сетью или сетью взаимосвязей. Сеть — это разрешенная форма набора взаимосвязей. Если у вас есть две взаимосвязи — та, которая определяет некую взаимосвязь, и та, которая отменяет первую взаимосвязь, то в разрешенной форме этого набора не будет взаимосвязей. Разрешенная форма набора взаимосвязей — это скорее логический результат применения всех физических взаимосвязей.

Рассмотрим пример сети. Property, Plant and Equipment (Основные средства) — это бухгалтерский концепт, который часто включается в сети вычислений XBRL, чтобы обозначить математические бухгалтерские взаимосвязи, показанные на рис. 4.2.

Как следует из рис. 4.2, XBRL утверждает, что Property, Plant and Equipment — это сумма концептов Land (Земля), Buildings (Строения), Furniture and Fixtures (Мебель и движимое имущество) и др. Для того чтобы выразить взаимосвязи в таксономии XBRL, у вас должна быть по крайней мере одна сеть, потому что все ресурсы и взаимосвязи существуют в сети.

(=) Property, Plant and Equipment, Net
(+) Land
(+) Buildings
(+) Furniture and Fixtures
(+) Other

Рис. 4.2. Взаимосвязь, выраженная в виде сети, описывающей концепт Property, Plant and Equipment и его компоненты

Когда использовать сети

Когда вы строите таксономию XBRL, вы выражаете ресурсы и взаимосвязи в рамках сетей. Понимание, зачем использовать сеть, помогает понять, что они собой представляют. Взаимосвязи относят к разным сетям по двум главным причинам.

- ✓ **Удобство.** Создатель таксономии может создать несколько меньших сетей, а не одну большую. Например, в таксономии американских правил GAAP информация, содержащаяся в таксономии, разделена на многочисленные сети. Другой способ создания таксономии — объединение всех сетей в одну. Иногда легче работать с несколькими мелкими сетями, чем с одной большой. Сети допускают разделение на части при необходимости.
- ✓ **Необходимость.** Иногда создавать сети необходимо. Еще одна причина для создания сетей — устранение конфликтов, возникающих, когда у одного родительского концепта есть два или больше возможных набора дочерних концептов. Например, в одной сети было бы невозможно выразить три разных значения концепта Receivables, Net (Дебиторская задолженность, чистая).
 - Receivables, Net (Дебиторская задолженность, чистая) = Receivables, Gross (Дебиторская задолженность, Общая) минус Allowance for Bad Debts (Резерв на сомнительные долги)
 - Receivables, Net (Дебиторская задолженность, чистая) = Receivables, Net, Current (Дебиторская задолженность, чистая, текущая) минус Receivables, Net, Noncurrent (Дебиторская задолженность, чистая, долгосрочная).

- Receivables, Net (Дебиторская задолженность, чистая) = Trade Receivables, Net (Торговая дебиторская задолженность, чистая) плюс Financing Receivables, Net (Финансовая дебиторская задолженность, чистая) плюс Others Receivables (Другая дебиторская задолженность, чистая) и т.д.



Слева на рис. 4.3 представлены три способа вычисления концепта Receivables, Net. Люди могут увидеть это разделение; мы понимаем то, что эти способы различаются. Но как компьютер может узнать, что они различаются? В то же время на рис. 4.3, *справа*, можно увидеть, что сети осуществляют три физически разных вычисления? Компьютерам необходимо сообщить, что эти три вида вычислений отличаются друг от друга. Иначе произойдет коллизия. Такое физическое разделение обеспечивают сети, сообщая приложениям о разделении.

Conflicts would exist		Networks separate conflicting computations	
		Network (By Component)	
Receivables, Net		Receivables, Net	
	Trade Receivables, Net		Trade Receivables, Net
	Financing Receivables, Net		Financing Receivables, Net
	Other Receivables, Net		Other Receivables, Net
		Network (By Net/Gross)	
Receivables, Net		Receivables, Net	
	Receivables, Gross		Receivables, Gross
	Allowance for Bad Debts		Allowance for Bad Debts
		Network (By Current/Noncurrent)	
Receivables, Net		Receivables, Net	
	Receivables, Net, Current		Receivables, Net, Current
	Receivables, Net, Noncurrent		Receivables, Net, Noncurrent

Рис. 4.3. Сети помогают избежать конфликтов

Попытка выразить все три взаимосвязи в рамках одной сети закончилась бы коллизией. Для того чтобы обойти эту проблему, необходимо использовать три разных сети, выражающих разные взаимосвязи вычислений.

Другие важные аспекты сетей

Укажем некоторые другие важные аспекты сетей, которые следует иметь в виду.

- ✓ Невозможно физически изменять сети, созданные другими, но можно создавать свои собственные сети, скрывая исходные взаимосвязи, чтобы, по сути, аннулировать их и определить новые. Сети существуют в виде набора физических файлов. Невозможно изменить чей-либо физический файл. Вместо этого взаимосвязи, которые вы создаете, получают приоритет над существующими взаимосвязями, по

сути, образуя новые взаимосвязи. Вы можете создать новую взаимосвязь, аннулировав существующие, просто создать новую взаимосвязь или сделать и то и другое. В действительности этот процесс создает новый набор взаимосвязей и документов так, будто вы изменили то, что существовало ранее.

- ✓ Хотя вы не можете изменить набор физических файлов чьей-либо таксономии, вы не должны использовать файлы базы ссылок, которые содержат взаимосвязи. Если вы не обращаетесь к файлам базы ссылок для достижения практических целей, то этих взаимосвязей не существует в контексте DTS, потому что файл никогда не будет обнаружен. (Более подробно таксономии и отчеты XBRL обсуждаются в главах 17 и 18.)
- ✓ Вы не можете упорядочивать сеть в XBRL. Например, допустим, что у вас есть три сети: Balance Sheet (Бухгалтерский баланс), Income Statement (Отчет о прибылях и убытках) и Cash Flow Statement (Отчет о движении денежных средств). Пользователи отчета XBRL могут использовать их в таком порядке: Balance Sheet, Income Statement и Cash Flow Statement. Несмотря на то что в языке XBRL нет средств для того, чтобы задать определенный порядок, для этого можно использовать собственные подходы. Однако следует иметь в виду, что они являются закрытыми и на них не следует рассчитывать за пределами конкретной части программного обеспечения, которое поддерживает этот конкретный подход.
- ✓ Пользователи могут изготовить сеть по своему вкусу без участия создателя соответствующего отчета XBRL или с помощью механизма расширения XBRL. По этой причине документы XBRL иногда называют интерактивными данными. Пользователи могут использовать информацию, содержащуюся в отчете XBRL, так, как они считают целесообразным, без учета мнения создателя информации. Однако, когда пользователи информации действительно реорганизуют ее, они берут на себя ответственность за получение правильных взаимосвязей.

Схемы таксономии

Схема таксономии — это одна из двух составляющих частей таксономий XBRL (второй являются базы ссылок), предназначенная для выражения информационной модели отчетов XBRL. Схемы таксономий состоят из нескольких частей, каждая из которых помогает выразить значение релевантной информационной модели. Все ее компоненты использовать не обязательно, но при желании это возможно.

Важные части схем таксономий — концепты и указатели на другие концепты, ресурсы и взаимосвязи. (Мы объясним каждую часть схемы таксономии в следующих разделах.)



Префиксы пространств имен XML улучшают читабельность документов XML. Хотя эти приставки необязательные, они могут быть чем-угодно (конечно, в пределах надлежащего синтаксиса XML). Мы используем эти приставки пространств имен для демонстрации разметки XML.

- ✓ `xs:` пространство имен XML Schema.
- ✓ `xbrli:` пространство имен XBRL Instance.
- ✓ `link:` пространство имен XBRL Linkbase.
- ✓ `basic:` схема таксономии в нашем примере.

Концепты

По существу, схемы таксономии описывают концепты, моделируя подходящие аспекты коллекции бизнес-концептов (управляемый словарь), связанные с некоторым обменом бизнес-информацией. Отчеты XBRL используют концепты из схемы таксономии, чтобы выразить значения фактов с целью обмена этой информацией.

Концепты имеют следующие особенности, выражаемые атрибутами элемента `xs:element` языка XML Schema.

- ✓ **Имя.** У каждого концепта должен быть атрибут имени. Имя идентифицирует концепт в рамках схемы таксономии.
- ✓ **Идентификатор.** У каждого концепта может быть атрибут идентификации. Если вы используете базы ссылок, то вам нужен идентификатор, когда вы добавляете ресурсы или выражаете взаимосвязи, потому что этот идентификатор описывает, как данный ресурс или взаимосвязь связаны с концептом. Идентификатор имеет двойное предназначение. Во-первых, он однозначно идентифицирует концепт в рамках схемы таксономии. Во-вторых, взаимосвязи и ресурсы используют идентификатор (не имена) как уникальные ключи, чтобы создать ссылки между взаимосвязью или ресурсом и релевантным концептом.
- ✓ **Тип.** У каждого концепта должен быть атрибут типа. Возможно, вы уже слышали термин “тип данных”. Типы данных ограничивают содержание, например, строка, целое число или дата. В языке XBRL есть подобные примитивные типы, но, помимо этого, в нем также есть много более специализированных типов, характерных для бизнеса, например, деньги или акции. Можно также определять собственные типы.
- ✓ **Группа подстановки.** У каждого концепта должен быть атрибут `substitutionGroup`, а его значением, скорее всего, будет `xbrli:item`, который означает, что элемент является концептом XBRL.



Группы подстановки могут быть многочисленными. Просто помните, что их много, и будьте осторожны при их использовании. В девяносто девяти случаях из ста вы будете использовать очевидную группу подстановок `xbrli:item`.



- ✓ **Тип периода.** У каждого концепта должен быть атрибут `xbrli:periodType`, предоставляющий информацию о виде периода, к которому относится концепт. Обычно используются два значения этого атрибута: на дату и за период. Значение атрибута `xbrli:periodType` на дату означает определенный момент времени. Значение атрибута `xbrli:periodType` за период означает промежуток времени между двумя моментами. Например, бухгалтерский баланс отражает информацию об отдельном моменте времени, скажем, 31 декабря 2008 г., и, следовательно, атрибут `xbrli:periodType` имеет значение на дату. Отчет о прибылях и убытках отражает информацию о промежутке времени, например “В течение года, заканчивающегося 31 декабря 2008 г.” или за период с 1 января 2008 г. по 31 декабря 2008 г., и, таким образом, атрибут типа `periodType` имеет значение за период. Другое возможное значение атрибута типа `periodType` — `xbrli:forever`. Дополнительную информацию можно получить, прочитав спецификацию XBRL.
- ✓ **Баланс.** Концепты могут иметь необязательный атрибут `xbrli:balance`, который может иметь два значения: дебет или кредит. Если вы — бухгалтер, то работаете с дебетами и кредитами. В противном случае вы, вероятно, не должны иметь дело с дебетами и кредитами, или у вас есть бухгалтер, помогающий вам, поэтому спрашивайте у него о дебетах и кредитах.

Вот как выглядит концепт в схеме таксономии, выраженной в виде элемента языка XML Schema:

```
<xs:element name = "NetIncomeOrLoss"
  id = "basic_NetIncomeOrLoss"
  type = "xbrli:monetaryItemType"
  substitutionGroup = "xbrli:item"
  xbrli:periodType = "duration"
  xbrli:balance = "credit"
>
```

Предыдущий кодовый фрагмент выражает концепт `NetIncomeOrLoss` в форме элемента XML Schema `xs:element` в схеме таксономии. У элемента есть много атрибутов, большинство из которых обеспечены языком XML Schema, включая имя, идентификатор, тип и `substitutionGroup`. Атрибуты `xbrli:periodType` и `xbrli:balance` добавлены в XML Schema языком XBRL. Значения атрибутов довольно очевидны (просмотрите список перед фрагментом кода).



Иногда концепты называют элементами, потому что они существуют в пределах физической части таксономии XBRL, известной как схема таксономии, которая представляет собой язык XML Schema. Но не все элементы языка XML Schema в таксономии являются концептами. Атрибуты `substitutionGroup` фактов XBRL с единственным значением всегда имеют значение `xbrli:item`. Это означает, что они являются концептами XBRL.

Указатели на другие концепты, ресурсы или взаимосвязи

Вторую часть схемы таксономии образуют ссылки на другие схемы таксономии и базы ссылок. Создать ссылки можно с помощью следующих элементов.



- ✓ **Элемент `import`**. Для того чтобы сослаться на другую схему таксономии или файл XML Schema, можно использовать элемент `xs:import` языка XML Schema в файле схемы таксономии. Каждый импортированный элемент XML Schema имеет уникальный идентификатор пространства имен.
- ✓ **Элемент `include`**. Элемент `xs:include` языка XML Schema похож на элемент `xs:import`. Различие между ними состоит в том, что включаемые файлы внедряют элементы в пространство имен файла, содержащего включаемый элемент, и не содержат идентификатор пространства имен.
Элемент `include` используется редко, однако в языке XBRL GL он применяется действительно широко.
- ✓ **Элемент `linkbaseRef`**. Элемент `link:linkbaseRef` или ссылка базы ссылок ссылается на базу ссылок, содержащую сети или другие ресурсы и взаимосвязи.

Эти указатели позволяют использовать схемы таксономии, схемы XML или базы ссылок, созданные другими. Они позволяют строить свою таксономию из блоков, чтобы некоторые части вашей таксономии XBRL можно было использовать, не обращаясь к другим частям, или в виде физически изолированных файлов.

Сети ресурсов

Сети ресурсов позволяют выразить дополнительную информацию о концептах. Ярлыки и ссылки — это два типа ресурсов, которые предусмотрены стандартом XBRL. Спецификация XBRL Formula обеспечивает третий тип ресурсов. Рассмотрим эти ресурсы подробнее.

- ✓ **Ярлык (label)**. Этот вид ресурсов, как следует из его названия, позволяет авторам таксономий XBRL создавать ярлыки для каждого концепта в таксономии. В общем, они обеспечивают более простой в использовании ярлык вместо неудобных элементов имени в языке XML. Другое предназначение ярлыков — обеспечить поддержку нескольких языков и диалектов. Кроме того, ярлыки создают документацию концепта, например, понятное для человека определение концепта.
- ✓ **Ссылка (reference)**. Эти ресурсы позволяют авторам таксономий XBRL выражать ссылки на внешние источники (например, на абзац в руководстве), которые объясняют или уточняют концепт в терминах, понятных человеку. Ссылки — это указатели на ссылку, а не сами ссылки.

- ✓ **Формула** (formula). В базовой спецификации XBRL их нет, но в спецификацию XBRL Formula они добавлены. Формулы позволяют автору таксономии XBRL выражать разные типы бизнес-правил. Отчеты XBRL, использующие таксономию XBRL, которая содержит эти правила, должны подчиняться им. (Дополнительная информация о спецификации XBRL Formula приведена в главе 16.)

Можно выразить бесконечное количество разных типов сетей ресурсов. Можно даже определить собственный тип сетей ресурсов, используя спецификацию XBRL Generic Linkbase (см. главу 16).

Ресурсы связаны с концептом в схеме таксономии. Ресурсы могут иметь разные роли, многие из которых предписывает XBRL, но авторы таксономии могут также определять свои собственные роли ресурса. Эти роли позволяют авторам таксономии подразделять ресурсы, если потребуется. (Если вас интересует, как создать ресурсы, обратитесь к главе 22.)

Подразделять ресурсы на категории можно с помощью предыдущих ролей ресурса. Такая классификация помогает отделять один тип ресурса от другого. Роль ресурса отличается от сетевой роли (роли расширенной ссылки), которая содержит ресурсы. Наконец, запомните, что ресурсы связаны с идентификационным элементом концепта, выраженного в схеме таксономии.



Документация для каждого концепта, выраженного в таксономии XBRL, обычно реализуется в виде ярлыков с определенной ролью (см. <http://www.xbrl.org/2003/role/documentation>). Язык XML Schema предлагает документацию в форме элемента `xs:appinfo`, но этот способ не стандартизирован, не допускает расширение и вообще игнорируется большинством процессоров XBRL. Мы не можем запретить вам его использовать, просто помните, что это плохая идея.



Ярлыки, как правило, содержат понятный для человека текст, например `Assets`. Однако ярлыки могут содержать элементы разметки XHTML, которые позволяют более рационально использовать ярлыки. Хорошо ли это — другой вопрос. Лучшая практика использования элементов языка XHTML в ярлыках еще не согласована. Просто знайте об этой изящной особенности, но будьте осторожны!

Сети взаимосвязей

Для того чтобы выразить взаимосвязи между концептами, можно использовать следующие типы стандартных сетей взаимосвязей.

- ✓ **Презентация** (presentation). Эти взаимосвязи позволяют выразить простой тип отношения родитель–потомок, т.е. иерархию. Презентации прежде всего предназначены для того, чтобы помочь организовать таксономию XBRL. Презентации можно также использовать для того, чтобы создать понятную для человека визуализацию отчета XBRL.

- ✓ **Вычисление** (calculation). Эти взаимосвязи позволяют выражать определенные расчетные зависимости между концептами в рамках таксономии XBRL. Вычисления предусматривают только сложение и вычитание, а концепты должны находиться в одном и том же контексте XBRL. Во многих ситуациях это полезно, но иногда недостаточно. Если вы хотите производить более сложные вычисления, используйте спецификацию XBRL Formula (см. главу 16).
- ✓ **Определение** (definition). Эти взаимосвязи можно использовать для достижения многих целей. Как правило, определения позволяют выражать любой тип взаимосвязей. Вы можете описать любую роль дуги, объявив, какую взаимосвязь вы создали. Например, спецификация XBRL Dimensions использует определения, чтобы выразить типы данных с несколькими единицами измерения.

Можно выразить бесконечное количество разных типов сетей взаимосвязей. Кроме того, можно определить собственный тип сетей взаимосвязей, используя спецификацию XBRL Generic Linkbase (см. главу 16).

Взаимосвязь и концепт в схеме таксономии, к которой относится взаимосвязь, неразрывны. Процессоры XBRL понимают эту ссылку. У взаимосвязей могут быть разные роли дуг (arc roles), многие из которых предопределяет XBRL, но создатели таксономии могут также определять свои собственные роли дуг. При необходимости эти роли можно разделить на категории.

Мы не приводим здесь пример XML с базой ссылок, потому что с физической точки зрения XML слишком сложен и объяснение деталей принесло бы больше вреда, чем пользы. Такой пример есть в главе 22.

Отчеты XBRL

Отчеты XBRL — это физические документы, содержащие бизнес-информацию, которая была опубликована, передана или обменена как-то иначе. Отчеты XBRL состоят из следующих частей.

- ✓ Ссылка (ссылки) на таксономию (таксономии) XBRL.
- ✓ Контексты.
- ✓ Единицы измерения.
- ✓ Факты.
- ✓ Сноски XBRL (т.е. комментарии).
- ✓ Другие технические детали.

На рис. 4.4 показаны компоненты отчета XBRL и взаимосвязи с таксономиями XBRL, которые поддерживают этот отчет. Слева на рисунке показаны компоненты отчета XBRL и взаимосвязи между ними, а справа — ссылка между отчетом и таксономией XBRL, взаимосвязи между отчетом XBRL и таксономией XBRL, а также взаимосвязи внутри таксономии XBRL.

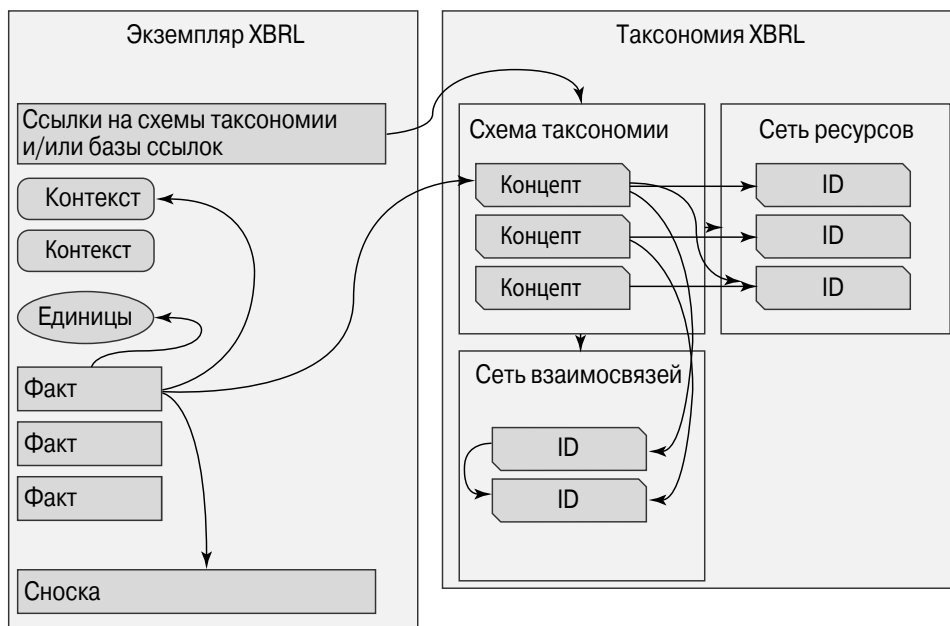


Рис. 4.4. У сетей есть два важных состояния

Ссылки на таксономии XBRL

Отчеты XBRL всегда ссылаются на таксономию или набор таксономий XBRL, известный как набор DTS. Поскольку таксономии XBRL образуют DTS, который определяет управляемый словарь и другие семантические значения, выражаемые таксономией XBRL, вам необходимо знать, где искать все эти части таксономии XBRL. Ссылки на эти физические файлы сообщают, где найти части таксономии XBRL, объясняющие управляемый словарь, использование отчета XBRL и все другие значения, выраженные для информационной модели DTS.

- ✓ **SchemaRef.** Элемент `link:schemaRef`, или ссылка схемы, указывает на файл схемы таксономии или файл XML Schema. Эти файлы могут содержать ссылки на еще большее количество файлов. Эти физические файлы могут быть компонентами, созданными другими, или компонентами, созданными вами, чтобы выразить новые концепты, взаимосвязи или ресурсы, используемые вашей информационной моделью, или изменить информацию другой таксономии XBRL, к которой вы обращаетесь. Такие модификации обычно называют *расширением* (extension).
- ✓ **LinkbaseRef.** Элемент `link:linkbaseRef`, или ссылка базы ссылок, указывает на базу ссылок, которая содержит взаимосвязи или ресурсы. Базы ссылок самостоятельно ссылаются на схемы таксономии, содержащие концепты, связанные с данной базой ссылок.



С помощью элемента `linkbaseRef` базы ссылок можно непосредственно соединить с отчетами XBRL. В качестве альтернативы можно использовать элемент `schemaRef`, соединяющий базы ссылок со схемой таксономии, которая ссылается на одну или несколько баз ссылок.

Процессор XBRL несет ответственность за охват всей таксономии XBRL, на которую есть ссылки и которые образуют набор DTS, правильно объединяя все части и сообщая о любых проблемах, касающихся этого процесса пользователю отчета XBRL.

Рассмотрим пример элемента `schemaRef`:

```
<link:schemaRef xlink:type = "simple" xlink:href = "Example.xsd"/>
```

Предыдущий фрагмент кода демонстрирует ссылку на схему таксономии с именем `Example.xsd`. Ссылка на базу ссылок выглядит аналогично, но элемент называется иначе.



Программное обеспечение может динамически добавлять базы ссылок в набор DTS, соединяя базу ссылок с отчетом XBRL. Например, рассмотрим отчет XBRL, относящийся к таксономии XBRL, в которой есть ярлыки только на английском языке. Предположим, что кто-то еще создает базу ссылок, которая обеспечивает для той же самой таксономии XBRL ярлыки на испанском языке. Далее, предположим, что ваш родной язык — испанский и вы предпочли бы использовать испанские ярлыки, но база ссылок физически не связана с отчетом XBRL. Нет проблем! Все, что необходимо сделать, — загрузить отчет XBRL, который автоматически загрузит DTS, на который указывает отчет XBRL. Тогда вы используете свое программное обеспечение, чтобы указать на базу ссылок с испанскими ярлыками. Вот! Можно использовать испанские ярлыки! Довольно удобно, не так ли? Именно поэтому документы XBRL иногда называют интерактивными данными.



Тот же самый подход можно использовать, чтобы реорганизовать способ, которым вы рассматриваете информацию, содержащуюся в отчете XBRL. Если вам не нравится способ создания отчета XBRL или то, как таксономия XBRL организовала информацию, можно просто реорганизовать информацию в отчете, создав собственное расширение таксономии XBRL. В этом случае вы можете использовать свою точку зрения, поддержанную вашей таксономией XBRL, вместо использования точки зрения создателя отчета, который содержит данную информацию.

Предоставление контекста фактов

Контекстная информация помогает пользователям информации, содержащейся в отчете XBRL, понимать контекст, в котором используется информация. Контекст обеспечивается элементом `xbrli:context`, который содержит следующие элементы.

- ✓ **Идентификатор.** У каждого контекста в отчете XBRL есть идентификатор, который однозначно идентифицирует контекст так, чтобы на него могли ссылаться одно или несколько значений факта.

Идентификационный атрибут контекста может выглядеть, например, так: x10B72009. У этих идентификаторов нет смысла. Они просто однозначно идентифицируют контекст и похожи на идентификационные номера. Значение выражается в элементах и атрибутах контекста.

- ✓ **Предприятие.** Элемент `xbrli:entity` — это часть контекста, которая состоит из двух компонентов: главного и дополнительного. Главный компонент — элемент `xbrli:identifier`. Схемный атрибут элемента `xbrli:identifier` сообщает, какая схема идентификации должна использоваться. Значение элемента `xbrli:identifier` — это реальный идентификатор из этой схемы. Например, американская комиссия SEC использует систему нумерации CIK. Отчеты XBRL, поступающие в комиссию SEC, используют указатель на эту схему, который может выглядеть вроде `http://www.sec.gov/CIK`. Значением идентификатора является фактическое число CIK, например 0000066740.

Вторая, необязательная часть информации — элемент `xbrli:segment`. В элемент `xbrli:segment` можно поместить любую информацию, какую пожелаете, чтобы помочь пользователям извлечь пользу из необходимой контекстной информации и понять значение факта. Отметим, что спецификация XBRL Dimensions описывает способ выражения данного типа контекстной информации, а использование информации, содержащейся в спецификации XBRL Dimensions, в элементе `xbrli:segment` является экспериментальной практикой. Однако, помимо языка XBRL, можно использовать любой диалект XML. (Спецификация XBRL Dimensions обсуждается в главе 16.) В сегменте можно выражать бесконечное количество элементов.

- ✓ **Период.** Элемент `xbrli:period` содержит контекстную информацию о периоде времени, к которому относится значение факта. Атрибут `xbrli:periodType` определен в схеме таксономии. Атрибут концепта `xbrli:periodType` в схеме таксономии должен соответствовать периоду, используемому в контексте этого концепта. Обычно используются два значения атрибута `xbrli:periodType`.
 - **На дату.** Если концепт таксономии XBRL определяет значение на дату, будет использоваться элемент `xbrli:instant`, содержащий единственную дату, например `<xbrli:instant>200 9-12-31</xbrli:instant>`
 - **За период.** Если концепт таксономии определяет значение за период, то в элементе `xbrli:period` будут использоваться два элемента: `xbrli:startDate` и `xbrli:endDate`. У каждого из них есть значения, обозначающие начало и конец диапазона дат. Дата начала может быть похожей на `<xbrli:startDate>2009-01-01</xbrli:startDate>`, а дата окончания — на `<xbrli:endDate>2009-12-31</xbrli:endDate>`.

С периодом связана еще одна возможность, но она используется редко: значение `xbrli:forever` означает, что никакая дата не указана и контекст применяется всегда.

- ✓ **Сценарий.** Элемент `xbrli:scenario` работает в манере, аналогичной элементу `xbrli:segment` контекста `xbrli:entity`, определенного ранее. Элемент `xbrli:scenario` является необязательным и может содержать любой текст XML, кроме XBRL. Один из способов выразить эту информацию — использовать спецификацию XBRL Dimensions. Этот подход в настоящее время становится лучшей практикой. В элемент `xbrli:scenario` можно поместить буквально любой тип контекстной информации, например, является ли информация актуальной, планируемой и т.д. В рамках сценария можно выразить бесконечное количество элементов. Соединяя эти части, приведем пример элемента `xbrli:context`.

```
<xbrli:context id = "D-2009">
  <xbrli:entity>
    <xbrli:identifier scheme="http://www.Sample.
    com%22%3eSAMP%3c/identifier">
      <xbrli:segment>
        >>>> информация о сегменте <<<<<
      </xbrli:segment>
    </xbrli:entity>
    <xbrli:period>
      <xbrli:startDate>2009-01-01</xbrli:startDate>
      <xbrli:endDate>2009-12-31</xbrli:endDate>
    </xbrli:period> <xbrli:scenario>
      >>>> информация о сценарии <<<<<
    </xbrli:scenario>
  </xbrli:context>
```

Предыдущий образец кода показывает компонентам элемент `xbrli:context` из отчета XBRL. У контекста есть значение идентификатора D-2009, которое совпадает со значением `contextRef` одного или нескольких фактов. Эту тему мы рассмотрим в разделе “Выражение значения фактов”. Элемент сущности `xbrli:identifier` имеет значение SAMP, которое поступает из схемы `http://www.Sample.com`. Элемент сегмента в нашем примере пуст (мы просто хотели показать, где он располагается). Он рассматривается в главе 22. Элемент `xbrli:period` имеет значение продолжительности (промежуток времени, начинающийся 01.01.2009 и заканчивающийся 31.12.2009). Как и значение элемента `xbrli:segment`, значение элемента `xbrli:scenario` не показано, чтобы не углубляться в объяснения раньше времени (см. главу 22).

Единицы измерения обеспечивают дополнительный контекст для числовых фактов

Для того чтобы обеспечить надлежащий контекст для числового значения, необходимо знать единицы измерения. Например, допустим, что у вас есть значение 1000. Что это: какая-то валюта, а если да, то доллары США, евро, иена или что-то другое? А может быть, это квадратные футы, численность персонала или что-то еще?

Отчет XBRL содержит единицы измерения всякий раз, когда появляются числовые значения факта. Элемент `xbrli:unit` содержит дополнительный контекст.

- ✓ **Идентификатор.** У каждого элемента `xbrli:unit` есть идентификационный атрибут, который однозначно идентифицирует единицу измерений так, чтобы числовые значения факта могли на нее ссылаться. Идентификатор единиц измерения работает точно так же, как идентификационный атрибут контекста.
- ✓ **Система измерений.** В принципе, у единиц измерения есть один элемент `xbrli:measure`, но он может принимать любые значения. Значение элемента `xbrli:measure` — это фактическая единица измерения значения факта — например, `iso4217:EUR`, который кодирует код валюты ISO для евро. Если у значения факта нет единицы измерения, такой как процент, то система измерения имела бы значение `xbrli:pure`, которое указывает на то, что значение представляет собой безразмерную величину.

Посмотрим, как можно выразить единицы измерения в отчете XBRL:

```
<xbrli:unit id = "U-Monetary">  
<xbrli:measure>iso4217:EUR</xbrli:measure>  
</xbrli:unit>
```

Этот пример кода демонстрирует элемент единицы измерения с идентификационным значением атрибута `U-Monetary`, который может соответствовать одному или нескольким значениям атрибута факта `unitRef` и связывает факт с этой единицей измерения. У элемента `xbrli:unit` есть элемент `xbrli:measure` со значением `iso4217:EUR`. Это означает, что используется единица измерения EUR, представляющая собой кодовое обозначение валюты в стандарте ISO 4217.

Выражение значений для фактов

Содержание отчета XBRL — это факт, который он передает. На что похож этот факт, зависит от того, имеет ли он числовую природу. Числовые факты должны указывать на единицы измерения, в которых они выражаются, а также количество десятичных разрядов.

Факт в отчете XBRL состоит из следующих компонентов.

- ✓ **Имя концепта.** Каждый факт — это данные для концепта, определенного в одной из схем таксономии в пределах комплекта DTS отчета XBRL. Например, имя концепта может быть чем-то вроде `gaap:CashAndCashEquivalents`. Имя концепта — это на самом деле не точное имя таксономии, а скорее составное (иногда называемое `QName`), которое состоит из конкатенации имени и того, что называют префиксом пространства имен, которому принадлежит схема таксономии. *Префикс пространства имен* (`namespace prefix`) — это сокращение *идентификатора пространства имен* (`namespace identifier`),

который идентифицирует схему таксономии. Да, мы знаем, что это скучно. Но технические специалисты внимательно изучают такую информацию и устанавливают ссылки незаметно для вас.

- ✓ **Идентификатор.** Факты могут иметь идентифицирующий атрибут, который однозначно идентифицирует факт, но это необязательно. Вообще, идентификатор факта является необязательным атрибутом.
- ✓ **Ссылка контекста.** У каждого факта есть ссылка на контекст или атрибут `contextRef`. Атрибут `contextRef` относится к одному из идентификационных атрибутов элемента контекста, определенного в вашем отчете XBRL. Элемент контекста содержит информацию, необходимую для понимания контекста, в котором используется значение факта.
- ✓ **Ссылка на единицу измерения.** У каждого факта с числовым концептом есть атрибут `unitRef` или атрибут ссылки на единицу измерения. Атрибут `unitRef` является одним из идентификационных атрибутов одного из элементов единицы измерения, определенного в вашем отчете XBRL. Такие концепты, как строки, даты и некоторые другие, не нуждаются в информации о единице измерения, следовательно, вы не обязаны обеспечивать их для фактов с такими типами. Ваше программное обеспечение может помочь вам разобраться, необходимы ли единицы измерения для факта.
- ✓ **Десятичные числа.** У каждого факта с числовым концептом должен быть атрибут, задающий количество десятичных знаков или точность. И десятичные знаки, и атрибут точности означают практически одно и то же. Большинство людей предпочитают задавать количество десятичных разрядов, потому что они легко их понимают. Вы можете задать количество десятичных знаков равным, скажем, 2, чтобы указать, что значение факта указано с точностью до двух десятичных разрядов, или 3, чтобы указать, что число выражается в тысячах. В принципе, приложения сами подсказывают правильный выбор.
- ✓ **Значение факта.** У фактов есть значение, которое называется *значением факта* (fact value). Значение факта может быть числом, например 1000, текстом, например LIFO, или любым другим типом данных, включая даже текст, состоящий из нескольких параграфов. Тип данных определяется и реализуется схемой. Тип данных определяется и задается схемой таксономии.

Вот пример числового факта:

```
<basic:NetIncomeOrLoss
  contextRef = "D-2009"
  unitRef = "U-Monetary"
  decimals="-3">5347000</basic:NetIncomeOrLoss>
```

Предыдущий фрагмент отчета XBRL демонстрирует факт со значением 5347000. Этот факт выражает информацию для концепта `basic:NetIncomeOrLoss`. Он указывает на контекст с атрибутом `contextRef`, равным D-2009, и значением атрибута единиц измерения `unitRef`, равным U-Monetary. (Если вас интересует дополнительная информация, обратитесь к разделам “Предоставление контекста фактов” и “Единицы измерения обеспечивают дополнительный контекст для числовых фактов”). Значение атрибута десятичных чисел, равное 3, означает, что числовой факт задается с точностью до трех десятичных разрядов (т.е. в тысячах).

Комментарий со сносками XBRL

Отчеты XBRL могут также содержать то, что является, в принципе, комментариями, но в XBRL называется *сносками* (footnotes). Сноски XBRL используются в отчете XBRL как комментарии или примечания, которые относятся к одному или нескольким значениям факта.



Сноски XBRL — эффективная сеть ресурсов, физически содержащаяся в отчете XBRL. Сеть сносок выражается как база ссылок и работает точно так же, как любой другой тип ресурса базы ссылок.

Сноски хороши и полезны в особых ситуациях, но обычно без них можно обойтись (см. главу 22).



Многие люди путают сноски XBRL с термином “сноска”, который относится к финансовым отчетам. Это не одно и то же. Функциональные возможности, которые предоставляют сноски XBRL, способ, которым люди их используют, и способ их реализации в программном обеспечении плохо согласованы друг с другом, поэтому использования сносок, если это возможно, необходимо избегать. Когда определенный регулирующий орган или кто-то еще настаивает на использовании сносок, приходится подчиниться, но при этом необходимо следовать руководству, предоставленному этим регулирующим органом.

Обеспечение гибкости с помощью расширения

Многие компании нуждаются в информационных моделях XBRL, чтобы стать динамичными (гибкими), и язык XBRL был создан, чтобы удовлетворить эту потребность. XBRL позволяет модифицировать таксономии XBRL, но это не обязательно.



Мы используем термин “модификация” в особом смысле. Эти модификации означают не изменение физических файлов, созданных другими людьми, а скорее добавление новых компонентов таксономии XBRL, которые могут виртуально изменить то, что создали другие. Они явно выражают способ этих изменений.

Укажем два условия, которые необходимо знать, чтобы понять, как работает расширение.

- ✓ **Базовая таксономия XBRL.** Базовую таксономию XBRL можно интерпретировать как таксономию XBRL, которую расширяет другая таксономия XBRL. Для описания таких таксономий используются также термины *стандартные таксономии* (standard taxonomies) или *якорные таксономии* (anchor taxonomies). Например, базовой может быть таксономия, навязанная регулирующим органом.
- ✓ **Расширенная таксономия XBRL.** *Расширенную таксономию* XBRL можно интерпретировать как таксономию XBRL, которая расширяет некоторую базовую таксономию XBRL. Для описания таких таксономий используются также термины *индивидуальная таксономия* (custom taxonomy) или *таксономия компании* (company taxonomy). Расширения таксономий XBRL представляют собой настройки, созданные над некоторой базовой таксономией XBRL для чьего-то определенного использования. Например, компания, которая обменивается данными с регулирующим органом, позволяющим использовать расширения, может создать расширение таксономии компании, которое дополняет базовую таксономию, созданную регулирующим органом.



С физической точки зрения базовая таксономия XBRL не отличается от расширенной. Фактически таксономия XBRL может быть и базовой, и расширенной. Например, таксономия XBRL может расширять другую таксономию XBRL, и затем кто-то еще может использовать эти таксономии XBRL в качестве основы для дополнительной таксономии XBRL.

У расширяемости XBRL есть два фундаментальных аспекта.

- ✓ **Расширение.** Добавление или изменение посредством расширения существующего DTS. Расширение означает создание нового расширения таксономии XBRL, которая ссылается (но непосредственно не воздействует или не заменяет) на оригинал, который, в свою очередь, часто называют базовой таксономией XBRL. Расширение таксономии XBRL добавляет концепты, ресурсы, взаимосвязи или сети ресурсов и взаимосвязей.
- ✓ **Запрет.** Удаление путем аннулирования взаимосвязи или ресурса из существующей таксономии с помощью расширения таксономии XBRL. Ключевой момент заключается в том, что вы избавляетесь от чего-то, добавляя что-то. Обратите внимание на то, что вы не можете запретить сами концепты. Запрещение взаимосвязи или ресурса влечет за собой создание одного или нескольких новых ресурсов или взаимосвязей в новом расширении таксономии XBRL, в которой они удаляют и аннулируют определенный ресурс или взаимосвязь из

существующей ранее базовой таксономии XBRL, на которую они ссылаются.



Никакой тип расширения не влияет на существующую ранее базовую таксономию XBRL. Многие организации могут одновременно расширить одну и ту же базовую таксономию XBRL совершенно разными способами, не воздействуя друг на друга или базовую таксономию XBRL. Однако расширения могут воздействовать на весь комплект DTS.

Понимание расширения

Предположим, что кто-то создал таксономию XBRL, которая включала в себя концепты, ресурсы и взаимосвязи. Расширение может

- ✓ добавить один или несколько новых концептов;
- ✓ запретить использование существующих ресурсов, фактически аннулируя их;
- ✓ запретить использование существующих взаимосвязей, фактически аннулируя их;
- ✓ добавить один или несколько новых ресурсов или взаимосвязей;
- ✓ добавить одну или несколько новых сетей ресурсов или взаимосвязей.

Пользователь, желающий расширить некоторую базовую таксономию XBRL, создает новую таксономию XBRL, погружает ее в свое расширение, ссылаясь на него, и таким образом изменяет набор DTS с помощью расширения и/или информации о запрете из дополнительно созданной таксономии XBRL. Итак, то, что вы делаете с расширением или с запретом, по существу, добавляет части (схему таксономии или базу ссылок) к набору, который включает DTS, таким образом изменяя весь DTS.

Форма — полная противоположность расширения, она статична. Человек, заполняющий форму, не может ее изменить, чтобы она стала соответствовать его потребностям. (Впрочем, автор формы может год от года обновлять форму, чтобы добавлять новые сущности.) Если ваша информация статична как форма, то особенности расширения XBRL могут не иметь для вас никакой пользы. Но другие особенности XBRL, такие как способность выражать семантику (смысл) формы, все еще могут оказаться полезными, например, для формы вычислений или других взаимосвязей. Главное, что формы статичны и не нуждаются в расширении, тогда как многие типы бизнес-информации не сводятся к формам; они носят динамичный характер.

Приведем два примера статичных и динамичных данных.

- ✓ **Статичные.** Большинство налоговых форм.
- ✓ **Динамичные.** Большинство финансовых отчетов американских акционерных обществ.

Причины для расширения

Перечислим три основные причины, по которым может возникнуть желание создать дополнительную таксономию XBRL.

- ✓ **Использование существующих информационных моделей.** При необходимости можно полностью переопределить взаимосвязи или создать собственную таксономию XBRL. Однако, если существует другая таксономия XBRL, которая лучше соответствует вашим потребностям, то вместо создания новой таксономии XBRL просто расширьте существующую и получите то, чего желаете. Примером является форма, которую вы хотели бы немного изменить. Допустим, вам нравится 90 процентов того, что существует в этой форме, но вам не нравятся другие 10 процентов и вы хотите добавить еще 5 процентов, а не начинать работу на пустом месте. Тогда просто создайте расширение XBRL. Таким образом, вы расширите форму, чтобы создать новую, измененную форму, основанную на другой форме.
- ✓ **Индивидуальный способ выражения информации.** Часто организации, даже работающие в одной и той же отрасли промышленности, используют разную терминологию — например, разные ярлыки для одного и того же концепта или разные способы вычисления значений. При необходимости можно воспользоваться этими моделями, если система, в которой вы действуете, позволяет это. Кроме того, можно просто добавить ярлыки на другом языке. Способ использования основной информационной модели таксономии XBRL ясно выражен в вашей дополнительной информационной модели таксономии XBRL, и те, кто используют ваш отчет XBRL, могут увидеть и понять, что вы сделали.
- ✓ **Индивидуальный способ представления информации.** Можно полностью реорганизовать чью-то таксономию, не изменяя значения информации, выраженной ею. Например, таксономия XBRL американских правил GAAP — это согласованная таксономия, созданная бухгалтерами. У аналитиков есть другое представление о точно такой же информации. Аналитики могли бы реорганизовать части таксономии XBRL американских правил GAAP или полностью реорганизовать всю таксономию, если бы посчитали целесообразным, совершенно не изменяя значение информационной модели.



То, что позволено относительно расширения таксономии XBRL, как правило, базируется на согласованной практике деловых партнеров или регулирующих органов, действующих в системе, с которой вы взаимодействуете.

Один из сценариев, в котором проявляется ценность расширений, — финансовая отчетность публично продаваемых компаний. Даже компании, работающие в одной и той же промышленности, не сообщают одну и ту же информацию одним и тем же

способом (особенно в США). Иначе говоря, финансовые отчеты — это не формы, которые должны быть заполнены; они носят более динамичный характер. Финансовые отчеты имеют много общего, но компании проявляют гибкость, сообщая финансовую информацию с учетом уникальных аспектов своего бизнеса. Гибкость в данном случае может означать включение концепта, который больше нигде не используют, добавление сущностей немного по-другому или перемещение сущностей из одного места в другое. Язык XBRL был разработан именно для того, чтобы справиться с такой гибкостью.