

Глава 19

Фреймворки прикладной области

В этой главе

- Введение
- Фреймворк организационной модели
- Инфраструктура модели продукта
- Инфраструктура ресурса операции
- Инфраструктура финансовых аналитик
- Инфраструктура учета
- Инфраструктура документа-источника

Введение

Microsoft Dynamics AX 2012 включает в себя набор новых прикладных фреймворков, увеличивающих уровень повторного использования кода в модулях приложения, что уменьшает потребность в создании искусственных межмодульных ролевых отношений и сокращает число тесно связанных взаимозависимостей между модулями. Например:

- роль операционного ресурса, назначаемая людям и рабочим центрам фреймворком операционных ресурсов, устраняет необходимость в искусственном назначении ролей в рабочих центрах сотрудникам и независимым подрядчикам, при этом они могут участвовать в планировании производства и календарном планировании.
- мощный механизм аналитик, ставший расширением фреймворка аналитик, журнал субкниг двойного учета, предоставляемый бухгалтерским фреймворком, и абстракция исходного документа, поддерживаемая фреймворком исходных документов, отвязывает операционные действия, такие как приобретение, получение и оприходование от учетных операций, таких как бюджетный контроль и финансовая отчетность.

В главе дается краткое описание концептуальных основ некоторых ключевых фреймворков прикладной области наряду со ссылками на технические описания, в которых можно найти более подробное руководство по реализации с примерами кода. Глава не содержит исчерпывающего списка всех прикладных фреймворков Microsoft Dynamics AX 2012. Напротив, она призвана дать лишь общий обзор важной новой функциональности с советами по ее использованию. За информацией по остальным прикладным фреймворкам обратитесь к документу «White Papers for Developers» по адресу: <http://technet.microsoft.com/EN-US/library/hh272882>. Эта страница часто обновляется по мере выхода новых описаний.



Примечание. В этой главе названия сущностей в концептуальных моделях прикладной области даются с большой буквы курсивом.

Фреймворк организационной модели

Microsoft Dynamics AX 2012 представляет новую инфраструктуру организационной модели. Она разработана для моделирования ключевых сценариев, требующихся в правительственных организациях и корпорациях, работающих в глобальном масштабе и имеющих разделение между официальной и операционной структурой. Новый фреймворк расширяет понятие компании, использовавшееся в Microsoft Dynamics AX 2009 и более ранних версиях.

С новыми типами организаций, появившимися в Microsoft Dynamics AX 2012, вы можете моделировать организации так, чтобы это отвечало их реальной операционной деятельности без необходимости изменения самого приложения. Большинство предприятий используют итеративный цикл мониторинга, измерения, анализа и улучшения. Результатом фазы анализа являются новые бизнес-правила и нормы, стратегические и операционные инициативы, нацеленные на улучшение результатов деятельности организации. С помощью организационного фреймворка вы можете создавать иерархические структуры, которые позволяют применять этот цикл повышения показателей.

Как работает фреймворк организационной модели

Организационная модель имеет две основных компоненты: *Типы организаций* и *Организационные иерархии*.

Последующие разделы описывают эти компоненты.

Типы организаций

Организационная модель в Microsoft Dynamics AX 2012 представляет два новых типа организации: *Юридическое лицо* и *Операционная единица*.

- **Юридическое лицо.** Организация с зарегистрированной или законодательно установленной структурой, имеющая право заключения контрактов и обязанная готовить отчетность о результатах своей деятельности. Юридическое лицо и компания в Microsoft Dynamics AX 2012 семантически означает одно и то же. Однако некоторые функциональные области приложения остаются основанными на модели данных, использующей концепцию компании (или *DataArea*). Эти области могут иметь ограничения, аналогичные Microsoft Dynamics AX 2009, и задавать неявные границы безопасности для данных.
- **Операционная единица.** Организация, разделяющая управление экономическими ресурсами и операционными процессами между людьми, в чьи обязанности входит максимизация использования ресурсов, в целях улучшения бизнес-процессов и контроля их результативности.

Microsoft Dynamics AX 2012 включает несколько типов операционных единиц.

- **Бизнес-единица.** Полуавтономная *Операционная единица*, созданная для достижения стратегических бизнес-целей.
- **Центр затрат.** Тип *Операционной единицы*, описывающий организацию, созданную для отслеживания затрат или расходов. Центр затрат является аккумулятором затрат и используется для управления издержками.
- **Подразделение.** Тип *Операционной единицы*, который может нести ответственность за прибыли и убытки, и сам может состоять из нескольких групп центров затрат. Подразделения также часто создаются на основе функциональных обязанностей или профессий, как, например, продажи и маркетинг.
- **Поток создания ценности.** Тип *Операционной единицы*, обычно используемый в бережливом производстве. В этой методологии поток создания ценности имеет один или более потоков производства, описывающий действия и потоки, требующиеся для поставки продукта, изделия или услуги покупателям продукта.

- **Канал розничной торговли.** Тип *Операционной единицы*, используемый в розничной торговле для представления канала продаж.

Команда – также тип *Операционной единицы*, но это неформальная группа людей, создаваемая обычно для достижения некоторой цели на короткий промежуток времени. Команды могут быть созданы для конкретных проектов или услуг. Остальные типы организационных единиц, описанные здесь, являются более постоянными, но и в них могут быть частые мелкие обновления или сильные изменения вследствие реструктуризации предприятия.

Эти типы операционных единиц поддерживают функциональность приложения в Microsoft Dynamics AX 2012. Тем не менее, каждая отрасль промышленности и предприятие имеет уникальные требования для своих операционных единиц и может по-разному их называть. Кроме того, в своих целях организации могут создавать и собственные типы операционных единиц. Больше об этом можно узнать далее в разделе «Создание нового типа операционных единиц».

Когда вы выстраиваете юридические лица и операционные единицы в иерархии и используете их для агрегированной отчетности, разделения доступа к данным и реализации политик предприятия, они помогают вам в реализации внутреннего управления организацией.

Организационные иерархии

Организация – это группа людей, которые работают вместе, выполняя оперативные или административные процессы. *Организационные иерархии* представляют собой соотношения между *Организациями*, составляющими предприятие или государственное учреждение. В предыдущих версиях системы компании не могли быть организованы в иерархию, представляющую структуру организации. В реальности организации обычно имеют иерархическую структуру по причинам, рассмотренным в предыдущем разделе.

Фреймворк организационной модели в Microsoft Dynamics AX 2012 поддерживает создание множества иерархий, действительных на некоторые даты. Это полезно при проведении реструктуризации, когда вы хотите, чтобы обновленная иерархия стала действительной, начиная с какой-либо даты в будущем. Фреймворк также поддерживает иерархии различного назначения.

Цель определяет то, как *Организационная иерархия* будет использована в сценариях приложения. *Цель*, которую вы выберете, определяет

типы организаций, которые могут быть включены в иерархии. В табл. 19-1 показаны типы организаций, которые вы можете использовать в целях иерархий, доступные в Microsoft Dynamics AX 2012.

Табл. 19-1. Цели и типы организаций

Цель	Описание	Типы организаций
Внутренний контроль закупаемой продукции	Используйте эту цель для задания политик управления закупочных процессов	Все
Внутренний контроль расходов	Используйте эту цель для задания политик отчетов о расходах	Все
Организационная диаграмма	Используйте эту цель в управлении персоналом для задания отношений для отчетности	Все
Внутренний контроль права подписи	Используйте эту цель для задания политик лимитов одобрения. Эти политики управляют ограничениями сумм трат и одобрений, назначенными сотрудникам	Все
Внутренний контроль платежей поставщику	Используйте эту цель для задания политик платежей поставщику	Юридические лица
Внутренний контроль аудита	Используйте эту цель для задания политик указания документов для аудита	Юридические лица
Централизованные платежи	Используйте эту цель для осуществления платежей одним юридическим лицом от имени другого юридического лица	Юридические лица
Контроль доступа	Используйте эту цель для определения безопасности доступа к данным в организациях	Все
Розничный ассортимент	Используйте эту цель для определения ассортимента в розничных каналах	Все
Пополнение запасов для розничной торговли	Используйте эту цель для определения конфигураций пополнения запасов	Все
Отчетность по розничной торговле	Используйте эту цель при определении аналитик для кубов отчетов по рознице	Все

Организационная модель имеет большое влияние на реализацию Microsoft Dynamics AX 2012 и на реализуемые в ней бизнес-процессы. В определении организационных структур должны участвовать руководящие работники и старшие менеджеры из различных функциональных областей, таких как финансы и учет, кадровая служба, операционный отдел, продажи и маркетинг.

На рис. 19-1 показана общая концептуальная модель фреймворка организационной модели.

Когда использовать фреймворк организационной модели

Фреймворк организационной модели используется для моделирования того, как работает предприятие. Вы можете использовать его двояко: используя встроенную интеграцию с другими фреймворками, приложениями и существующими модулями Microsoft Dynamics AX или же для моделирования собственных сценариев, подходящих именно вашей организации.

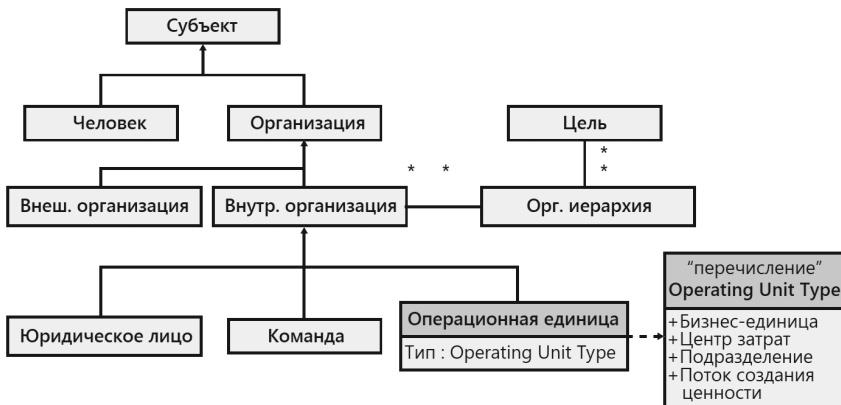


Рис. 19-1. Фреймворк организационной модели

Интеграция с модулями приложения из других фреймворков

Инфраструктура организационной модели неразрывно связана с некоторыми другими фреймворками и модулями в Microsoft Dynamics AX.

- **Адресная книга.** Все внутренние организации – юридические лица, операционные единицы и команды – являются типами сущности *Субъект*. Это означает, что эти организации могут использовать воз-

возможности адресной книги для хранения адресов и контактной организации. Больше информации об этом см. в техническом описании «Implementing the Global Address Book Framework» по адресу: <http://technet.microsoft.com/en-us/library/hh272867.aspx>.

- **Финансовые аналитики.** Вы можете использовать юридические лица и операционные единицы для задания финансовых аналитик и затем применения их в структурах счетов. Используя организации как финансовые аналитики, предприятие или государственное учреждение может проводить анализ финансовых показателей организации. Если два типа организации используются как различные финансовые аналитики в структуре счета, то отношения между организациями описываются через иерархии, играющие роль ограничений. Подробнее об этом смотрите в разделе «Инфраструктура аналитик» далее в этой главе.
- **Фреймворк политик.** Вы можете использовать инфраструктуру политик для задания внутренней политики управления для организации. Инфраструктура политик может быть использована для задания их в расходных отчетах, заявках на закупку, аудиторского контроля документов и платежей поставщику. Фреймворк политик предлагает поддержку для изменения политик и стандартное поведение политик по умолчанию для организаций на основе их иерархий, дает возможность внутреннего аудита для облегчения управления издержками, обнаружения мошеннических действий, улучшения операционной эффективности и повышения результативности предприятия в целом. Больше информации можно узнать в техническом описании «Using the Policy Framework» по адресу: <http://technet.microsoft.com/en-us/library/hh272869.aspx>.
- **Гибкая модель безопасности данных.** Новый гибкий фреймворк безопасности данных позволяет использовать возможности защиты данных, основываясь на каком-либо условии. Доступ к организациям может быть определен на основе иерархий. Больше об этом написано в главе 11.

Фреймворк организационной модели может быть использован в следующих модулях приложения.

- **Закупки и источники.** Строки заявок на закупку могут быть созданы для определенного закупающего юридического лица, и они будут получены операционной единицей, такой как центр затрат или подраз-

деление. Фреймворк организационной модели допускает различные сценарии, предоставляя доступ к просмотру или созданию заявок на закупку для любых закупающих юридических лиц и получающих операционных единиц, в которых вы имеете доступ на создание заявок на закупку.

- **Управление персоналом.** В управлении персоналом работники заключают договор трудоустройства с юридическим лицом и занимают некоторую позицию в подразделении. Все операционные сценарии в управлении персоналом используют эти концепции для выборки изменения данных.
- **Командировки и расходы.** Отчеты о расходах и их строки связываются с юридическим лицом, которое должно понести расход по строке расхода согласно установленному положению, а также связываются с операционной единицей для формирования внутренней отчетности.

Моделирование собственных функциональных сценариев

Вы можете использовать фреймворк организационных моделей для моделирования собственных сценариев. Общепринятым сценарием для реализации безопасности данных является фильтрация данных приложения на основе пользовательских ролей и членства во внутренних организациях. Например, организации могут ограничивать конкретного операциониста заказами на продажу на основе их географии, давая ему право видеть только заказы на продажу, созданные для его региона.

Вы можете создать новый сценарий или изменить существующий, выполнив следующие шаги.

1. Создайте или измените модель данных.
2. Создайте новую таблицу со свойством *SaveDataPerCompany*, установленным в *No*. Если вы работаете с существующими таблицами, которые хранят данные с разделением их по компаниям, измените значение свойства *SaveDataPerCompany* с *Yes* на *No*.
3. Создайте ссылку на организации как внешний ключ в таблице. Вам, возможно, придется сделать ссылки на операционную единицу и юридическое лицо, если ссылка на юридическое лицо не может быть получена через организационные иерархии юридических лиц или операционных единиц.

4. Если таблица включает избыточные данные в поле *Legal Entity*, задайте иерархические ограничения между юридическими лицами и операционными единицами для поддержания целостности данных.
5. Создайте новую форму (например, страницу списка) для ваших сценариев или измените существующий интерфейс пользователя для просмотра и работы с данными. Вы можете использовать собственные фильтры для обеспечения доступа пользователям к данным различных организаций.
6. Включите организации по умолчанию в таблицу финансовых аналитик, добавив их в структуры счетов.
7. Создайте расширяемую политику безопасности данных, основанную на организациях, к которым принадлежит или имеет доступ пользователь.
8. Используйте инфраструктуру политик для задания правильных политик доступа к данным для пользователей в этом сценарии.

Расширение фреймворка организационной модели

Вы можете изменить фреймворк организационной модели, создав собственные типы операционных единиц, собственные новые организационные цели или же модифицировав конструктор иерархий, поставляемый с Microsoft Dynamics AX 2012.

Создание нового типа операционных единиц

Ключевым сценарием расширения является адаптация организационной модели для соответствия требованиям конкретной вертикальной индустрии. Например, школьный регион (branch), школа (school) и школьный округ (school district), **несомненно, являются организационными концепциями**, так что вы можете смоделировать их как новые типы операционных единиц.

Предположим, что вы хотите создать операционную единицу *Branch*. Это можно сделать в три шага.

1. Создайте новый элемент перечисления для нового *Типа операционной единицы*.
2. Создайте представление.
3. Создайте пункт меню для нового типа операционных единиц.

В разделах ниже эти шаги описываются более подробно.

Создание нового элемента перечисления. Создайте новое значение в перечислении *OMOperatingUnitType*, соответствующее новому типу операционной единицы.

1. В дереве объектов приложения (AOT) перейдите к *Data Dictionary\Base Enums\OMOperatingUnitType*.
2. Щелкните правой кнопкой на перечислении *OMOperatingUnitType*, щелкните Создать Element и затем добавьте элемент с названием *Branch*.
3. В окне свойств для этого нового элемента установите для обеих свойств *Name* и *Label* значение *Branch*. Измените значение по умолчанию свойства *EnumValue* на такое целое значение, которое предупредит возможное пересечение вашего нового элемента с будущими элементами, которые будут добавляться командой Microsoft Dynamics AX.

Создание представления. Определите представление *DimAttributeBranchView*, подобное другим представлениям, созданным для других типов операционных единиц. Например, представление *DimAttributeOMBusinessUnit* создано для бизнес-единиц. Представление *DimAttributeOMBusinessUnit* содержит поля, позволяющие таблице *OMOperatingUnit* быть использованной в качестве финансовой аналитики для всех указываемых бизнес-единиц.

1. В AOT перейдите к узлу *Data Dictionary\Views* и затем найдите *DimAttributeOMBusinessUnit*.
2. Сдублируйте *DimAttributeOMBusinessUnit*: щелкните правой кнопкой на узле *DimAttributeOMBusinessUnit* и выберите Дублировать. AOT создаст узел с именем *CopyOfDimAttributeOMBusinessUnit*.
3. В окне свойств вновь созданного представления установите значения свойств, как показано в таблице ниже.

Свойство	Значение
<i>Name</i>	<i>DimAttributeBranchView</i>
<i>Label</i>	<i>Branches</i>
<i>SingularLabel</i>	<i>Branch</i>
<i>DeveloperDocumentation</i>	Впишите сюда описание представления. Например: <i>DimAttributeBranchView</i> содержит все записи из таблицы <i>OMOperatingUnit</i> , указанные как регионы

4. Под вновь созданным представлением *DimAttributeBranchView* найдите критерий *OMOperatingUnitTypeOMBusinessUnit*. Он находится в узле *Metadata\Data Sources\BackingEntity(OMOperatingUnit)\Ranges*. В окне свойств для критерия *OMOperatingUnitTypeOMBusinessUnit* установите свойства, как показано в таблице ниже.

Свойство	Значение
<i>Name</i>	<i>OperatingUnitTypeBranch</i>
<i>Field</i>	<i>OMOperatingUnitType</i>
<i>Value</i>	<i>Branch</i>

Создание пункта меню. И в конце создайте пункт меню для нового типа операционной единицы следующим образом.

1. В меню *Item\Display* создайте пункт с именем *BranchMenuItem*.
2. В окне *Properties* для *BranchMenuItem* установите свойства так:

Свойство	Значение
<i>Name</i>	<i>BranchMenuItem</i>
<i>Label</i>	<i>Branches</i>
<i>Object</i>	<i>OMOperatingUnit</i>
<i>EnumTypeParameter</i>	<i>OMOperatingUnitType</i>
<i>EnumParameter</i>	<i>Branch</i>

Теперь новый тип операционных единиц появится в списке типов операционных единиц, доступных при создании новой операционной единицы администратором системы.

Создание новой цели

Вы можете расширить организационную модель Microsoft Dynamics AX и создать в ней собственную цель. Цель задает то, как организационная иерархия используется в сценариях приложения. Предположим, вы хотите создать новую цель, называемую *Sales*. Для этого проделайте следующие шаги.

1. Создайте новый элемент перечисления для новой цели.
2. Создайте метод, добавляющий новую цель, и затем вызовите его, чтобы добавить цель в таблицу *HierarchyPurposeTable*.

В разделах ниже эти шаги описываются детальнее.

Создание нового элемента перечисления. Вначале создайте для новой цели соответствующий ей элемент перечисления. Для этого сделайте следующее.

1. В АОТ перейдите к узлу `Data Dictionary\Base Enums\HierarchyPurpose`.
2. Щелкните правой кнопкой на перечислении *HierarchyPurpose*, щелкните Создать Element и затем добавьте элемент с названием Sales.
3. В окне свойств для этого нового элемента установите для обеих свойств *Name* и *Label* значение *Sales*. Измените значение по умолчанию свойства *EnumValue* на такое целое значение, которое предупредит возможное пересечение вашего нового элемента с будущими элементами, которые будут добавляться Microsoft или независимыми поставщиками ПО.

Создание и вызов метода, добавляющего новую цель. Далее создайте метод, добавляющий новую цель, и вызовите его для добавления цели в таблицу.

1. В узле *Classes* АОТ найдите *OMHierarchyPurposeTableClass*.
2. Сдублируйте метод *addSecurityPurpose*: щелкните правой кнопкой на узле *addSecurityPurpose* и затем выберите Дублировать. АОТ создаст метод с названием *CopyOfaddSecurityPurpose*.
3. Замените код метода *CopyOfaddSecurityPurpose* следующим кодом. Этот код переименует метод.

```
private static void addSalesPurpose()
{
    OMHierPurposeOrgTypeMap omHPOTP;
    select ReclId from omHPOTP
    where omHPOTP.HierarchyPurpose == HierarchyPurpose::Sales;
    if (omHPOTP.ReclId <= 0)
    {
        omHPOTP.clear();
        omHPOTP.HierarchyPurpose = HierarchyPurpose::Sales;
        omHPOTP.OperatingUnitType = OMOperatingUnitType::OMAnyOU;
        omHPOTP.IsLegalEntityAllowed = NoYes::No;
        omHPOTP.write();
        omHPOTP.clear();
        omHPOTP.HierarchyPurpose = HierarchyPurpose::Sales;
    }
}
```

```

        omHPOTP.OperatingUnitType = 0;
        omHPOTP.IsLegalEntityAllowed = NoYes::Yes;
        omHPOTP.write();
    }
}

```

Представленный код подобен коду большинства методов класса *OMHierarchyPurposeTableClass*. Изменения в коде сделаны только в местах, где идет ссылка на значение перечисления *HierarchyPurpose*. В коде вы можете увидеть три упоминания *HierarchyPurpose::Sales*.

4. В классе *OMHierarchyPurposeTableClass* измените метод *populateHierarchyPurposeTable*, чтобы вызвать новый, созданный вами метод, добавив следующую строку кода:

```
OMHierarchyPurposeTableClass::addSalesPurpose();
```

Приведенный ниже код показывает изменения в методе *populateHierarchyPurposeTable*.

```

public static void populateHierarchyPurposeTable()
{
    OMHierPurposeOrgTypeMap omHPOTP;
    if (omHPOTP.ReclId <= 0)
    {
        ttsbegin;
        OMHierarchyPurposeTableClass::AddOrganizationChartPurpose();
        OMHierarchyPurposeTableClass::AddInvoiceControlPurpose();
        OMHierarchyPurposeTableClass::AddExpenseControlPurpose();
        OMHierarchyPurposeTableClass::AddPurchaseControlPurpose();
        OMHierarchyPurposeTableClass::AddSigningLimitsPurpose();
        OMHierarchyPurposeTableClass::AddAuditInternalControlPurpose();
        OMHierarchyPurposeTableClass::AddCentralizedPaymentPurpose();
        OMHierarchyPurposeTableClass::addSecurityPurpose();
        //Добавьте нижеследующую строку.
        OMHierarchyPurposeTableClass::addSalesPurpose();
        ttscommit;
    }
}

```

После завершению этих шагов новая цель появится в Управление организацией > Настройка > Организация > Цели организационной иерархии.

Расширение конструктора иерархий

Администраторы системы могут просматривать или изменять организационные иерархии, используя конструктор иерархий. Эта форма доступна через Управление организацией > Настройка > Организация > Организационные иерархии.

Разработчики могут изменять конструктор иерархий несколькими путями. Управляющий элемент конструктора иерархий может быть настроен путем изменения четырех параметров узлов организации внутри иерархии: цвет границы, картинка узла, цвет верхнего градиента и цвет нижнего градиента. Подробнее см. в техническом описании «Implementing and Extending the Organization Model in Microsoft Dynamics AX 2012» по адресу: http://download.microsoft.com/download/4/E/3/4E36B655-568E-4D4A-B161-152B28BAAF30/Implementing_and_extending_the_organization_model_in_Microsoft_Dynamics_AX_2012.pdf.

Инфраструктура модели продукта

Microsoft Dynamics AX 2012 **предлагает гибкую инфраструктуру управления** данными о продуктах, поддерживая как централизованное, так и разделенное по юридическим лицам управление информацией о продуктах, представляющих собой номенклатуру или услугу, получаемую в результате экономической деятельности.

Как работает инфраструктура модели продукта

В модели продукта информация о нем сгруппирована вокруг концепции *Продукта*, который концентрирует в себе информацию, общую для всей организационной структуры, а также концепции *Используемого продукта*, который содержит информацию, специфичную для отдельного юридического лица. Рис. 19-2 показывает общую концептуальную модель инфраструктуры модели продукта.

Типы и подтипы продукта

Продукт может быть *Номенклатурой*, воплощающей физическую сущность, для которой могут быть отслежены уровни складских запасов (готовая продукция или сырье), или же *Услугой*, воплощающей нематериальную сущность, для которой не существует складских запасов (такая, как предоставляемые консультационные услуги). *Продукт* может быть разделен по подтипам: *Продукт*, который может быть однозначно определен и ис-

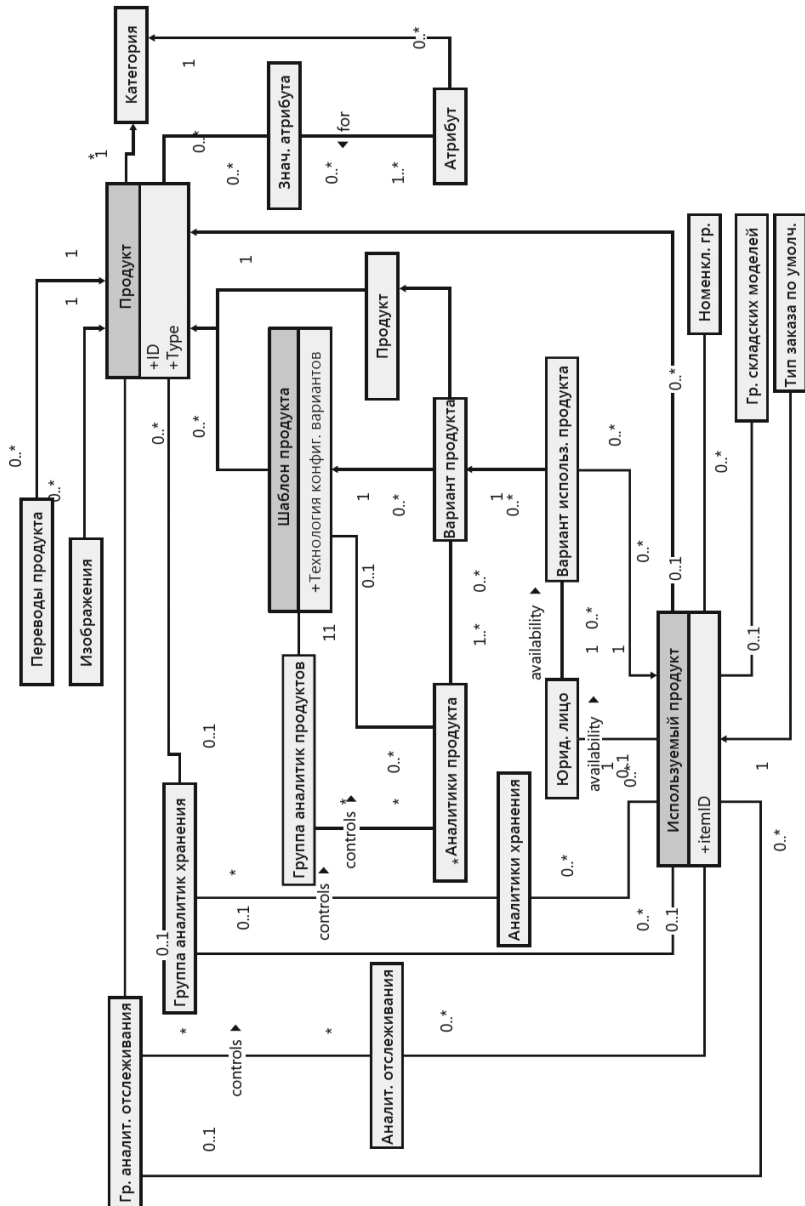


Рис. 19.2. Инфраструктура модели продуктов

пользован в экономических действиях, или же *Шаблон продукта*. *Шаблон продукта* – это стандарт или функциональное представление продукта, служащее основой для настройки различных вариантов продукта. В этом случае *Вариант продукта* – это уникально определяемый *Продукт*, который может быть использован в экономических действиях. Однако, так как все *Варианты продукта* связаны с *Шаблоном продукта*, значительная часть информации, определенную в *Шаблоне продукта*, у них общая.

Рассмотрим производственную компанию **Contoso, продающую** различные модели велосипедов, которые могут быть разного цвета и размера, и аксессуаров, такие как велосипедные фонари. В этом случае каждая модель велосипеда может быть определена как *Шаблон продукта*, а каждая комбинация цвета и размера задаст свой *Вариант продукта*. Фонари не имеют различных конфигураций, так что они могут быть созданы как отдельные *Продукты*.

Аналитики продуктов

Вариант продукта задается с использованием *Аналитик продуктов*, активных для данного *Шаблона продукта*. *Аналитики продуктов* – это характеристики, уникально идентифицирующие продукт. Microsoft Dynamics AX 2012 включает четыре *Аналитики продуктов*: Конфигурация, Размер, Цвет и Стил (вы можете их переименовать).

Группа аналитик продуктов, которая является обязательной для *Шаблона продукта*, включает в себе информацию об активных *Аналитиках продукта* и может быть использована для задания того, какие *Аналитики продукта* должны быть учтены в расчете цены в ценовых соглашениях. Так как *Аналитики продуктов* задают уникальные *Продукты*, то *Группа аналитик продуктов* должна быть назначена *Шаблону продукта*, после чего она станет общей для всей организационной структуры.

Например, Contoso продает две модели велосипедов. Каждая модель доступна в трех размерах (маленький, средний и большой) и в трех цветах (черный, красный и белый). Contoso может смоделировать этот ассортимент, создав *Группу аналитик продуктов* «Велосипеды», с активными *Аналитиками* Размер и Цвет, присвоив эту *Группу аналитик продуктов* двум *Шаблонам продукта*, представляющим две модели велосипедов. Поскольку Contoso продает еще и шлемы, доступные в одном цвете, но разных размеров, фирма может создать вторую *Группу аналитик продуктов* с названием Шлемы, которая имеет только одну активную *Аналитику*: Размер.

Аналитики хранения и отслеживания

Помимо *Аналитик продуктов*, есть еще два типа аналитик: *Хранения* и *Отслеживания*. *Группа аналитик хранения* определяет уровень детализации, используемый для определения физического местоположения товаров с такими возможными аналитиками, как *Сайт*, *Склад*, *Местоположение* и *Палета*. *Группа аналитик отслеживания*, с доступными аналитиками *Партия* и *Серийный номер* определяет, как будут отслеживаться конкретные экземпляры одного и того же *Используемого продукта*. Эти группы аналитик также задают политики относительно того, как конкретные значения аналитики влияют на действия на предприятии. Поскольку эти политики могут различаться в разных юридических лицах, то для одного и того же продукта в разных юридических лицах могут быть назначены различные аналитики хранения и отслеживания.

Например, в некоторых странах или регионах закон может требовать от Contoso отслеживать номера партий для велосипедных тормозов. Так что в случае обнаружения в них дефекта, влияющего на безопасность, вся партия может быть легко отозвана. Этого требования в других странах или регионах может не быть, так что Contoso в этом случае не должна отслеживать для тормозов номера партий.

Используемые продукты

Вся информация, приводившаяся до сих пор, относилась к концепции *Продукта* и касалась информации, общей для всей организационной структуры. Для того чтобы использовать продукт в юридическом лице, этой информации недостаточно. Чтобы это стало возможным, продукт вместе с относящимися к нему вариантами продукта должен быть использован для юридического лица. На *Используемом продукте* вы можете задать различные политики, которые определяют, как эти продукты могут быть использованы в юридическом лице. Эти политики включают *Группы аналитик хранения и отслеживания* (если они заданы в общей информации о продукте), *Группу складских моделей* и *Номенклатурную группу*, а также различные складские политики и политики себестоимости для этого продукта. Как только эти политики будут заданы, продукт может быть использован в операциях в юридическом лице.

Доступность продукта для юридических лиц может быть задана как на уровне продукта, так и на уровне варианта продукта. Например, конкретная модель велосипеда может быть использована только в Contoso U.S., так как компания хочет продавать ее только на этом рынке. Другая

модель может быть использована в Contoso U.S. только в черном и красном цвете, так как было принято решение, что продавать белые велосипеды в Соединенных Штатах не стоит.

Дополнительная информация о продукте

Помимо необходимой информации, для продукта могут быть заданы дополнительные детали, чтобы задать какие-либо ссылки либо для управления взаимодействием с другими компонентами системы. Например, вы можете задать для продукта его *Переводы*, чтобы определить название продукта или описание, отображаемое в различных контекстах. Точно так же вы можете присоединить *Изображения продукта*, чтобы дать о нем графическое представление. Примером информации, используемой для управления другими компонентами, может быть *Тип заказа по умолчанию*, который может установить юридическое лицо для определения, какой тип заказа должен быть создан для покрытия потребности в конкретном продукте. Например, *Contoso Italy, имеющая собственное швейное производство*, для велоодежды может установить *Тип заказа по умолчанию в Производство*, а для велосипедов, которые она закупает у внешних поставщиков, в *Заказ на покупку*.

Категории и атрибуты продукта

Microsoft Dynamics AX 2012 вводит концепцию *Атрибутов* продукта. Создаваемые пользователями *Атрибуты* могут быть связаны с *Категорией продукта* для описания характеристик, общих для всех продуктов этой *Категории*. Атрибуты продукта могут быть различными типами данных. Каждый *Атрибут* может иметь значение по умолчанию внутри *Категории*. Как только продукт добавляется в категорию, он наследует все атрибуты продукта внутри этой категории, вместе с их значениями по умолчанию. Значения атрибутов продукта могут быть изменены для каждого продукта отдельно.

Например, компания из пищевой отрасли имеет атрибут продукта, называемый *Жирность*, связанный с закупочной категорией молочных продуктов. Значение по умолчанию – 1%. Эта категория содержит продукт «Обезжиренное молоко», который наследует значение по умолчанию – 1% из категории. Категория также содержит продукт «Жирное молоко», который также наследует значение атрибута *Жирность*. Однако его значение для жирного молока было перезаписано значением 3%. Менеджер по закупкам может найти все продукты, соответствующие какому-либо кри-

торию, например, найти все молочные продукты со значением *Жирность* между 1% и 3%.



Примечание. Атрибуты продукта в основном представляют дополнительные описания продуктов, использующиеся для поиска. Вы не можете использовать атрибуты продукта для отслеживания складских остатков. Для отслеживания физического и финансового склада система использует только аналитики *Продукта*, *Отслеживания* и *Хранения*.

Технология получения конфигурации вариантов продукта

Определение шаблона продукта имеет обязательную технологию конфигурирования вариантов, используемую для создания стратегии конфигурации *Шаблона продукта*. Эта стратегия конфигурации указывает метод, использующийся для создания нового *Варианта продукта*, соответствующего потребностям клиента. Как результат конфигурирования продукта создается его новый вариант в общем хранилище продуктов. И при запуске конфигурации продукта происходит его автоматический выпуск для юридического лица.

Например, в среде настройки по заказу производственная компания предлагает набор предопределенных моделей продукта с различными ограничениями. Если существующие модели не отвечают ожиданиям какого-либо клиента, система создает новый уникальный вариант продукта для получения желаемой клиентом вариации продукта.

Технология конфигурирования, основанная на ограничениях. С помощью нового модуля *Конфигурация продукта* в Microsoft Dynamics AX 2012 вы можете описать общую модель продукта в терминах структуры продукта, его составляющих, атрибутов и ограничений. После создания модели продукта вы можете связать ее с определением шаблона продукта, подчиняющегося определенной стратегии конфигурирования, основанной на ограничениях. Эта продвинутая технология конфигурирования позволяет моделировать сложные структуры продуктов.

Например, компания, производящая систему домашнего кинотеатра, содержащую 10 компонентов с некоторым числом предопределенных ограничений на различные характеристики компонентов (атрибутов). Во время ввода заказа на продажу система отображает богатый пользователь-

ский интерфейс, проводящий пользователя через процесс конфигурирования для успешного создания правильного варианта продукта.

Технология конфигурирования, основанная на аналитиках. В Microsoft Dynamics AX 2012 вы можете настроить конфигурационную группу и правила конфигурации как часть общей настройки Управления запасами и складами. Эта информация может быть использована в определении спецификации. Вы можете использовать правила конфигурации для предварительного задания конфигураций продукта для использования как части конкретной конфигурации спецификации.

Например, производственная компания выпускает игровые устройства в нескольких конфигурациях. Для производства этих продуктов компания использует спецификацию, содержащую сырьевой компонент, поступающий в различных конфигурациях. С конфигурационными группами и правилами конфигурации система может ограничить включение в определенную конфигурацию игрового устройства только заданного типа сырья.

Эта функциональность предлагает легкий подход к конфигурированию продуктов в строке заказа, имеющих сравнительно простую структуры спецификаций.

Технология конфигурирования, основанная на заданных вариантах. Предварительное задание вариантов продукта – это наиболее простая стратегия конфигурирования в Microsoft Dynamics AX 2012. Различные варианты продуктов могут быть созданы автоматически или вручную, на основе значений аналитик продукта, связанных с шаблоном продукта.

Например, шаблон продукта может иметь активные аналитики Размер и Цвет. Каждый раз при добавлении пользователем нового значения цвета или размера система автоматически создает все возможные варианты продукта. Эта функциональность особенно ценна в розничной сфере, где компании предлагают широкий спектр продуктов различных видов, цветов и размеров.

Когда использовать инфраструктуру моделей продуктов

Вы можете расширить модель продукта для приведения в соответствие хранимой информации о продукте и его поведении с используемыми основными практиками ключевых организационных данных. Эти практики могут включать такие процессы, как управление данными, централизованный контроль основных данных или же продуктовые политики. Эти политики могут влиять на жизненный цикл продукта или же особое их

поведение в бизнес-процессах, например в закупке, производстве, логистике запасов и т. д.

К примеру, розничное предприятие, работающее в различных странах/регионах, может требовать осуществления процесса определения общей для всей организации политики, такой как цена продажи продукта. После установки новой цены продажи она должна быть установлена во всех розничных магазинах.

Расширения инфраструктуры модели продукта

В Microsoft Dynamics AX 2012 вы можете изменить модель продукта, модифицируя таблицы и классы с префиксом *EcoRes*. Например, вы можете начать с реализации политики розничной цены из предыдущего примера, добавив новое поле в таблицу *EcoResProduct* для отражения цены продажи в валюте, выбранной в качестве основной валюты в настройке параметров Главной книги юридического лица. Это поле будет автоматически унаследовано всеми подтипами продукта, такими как собственно продукты, шаблоны продуктов и варианты продуктов, что означает возможность назначать отдельную цену для вариантов продукта. Как только эта политика будет реализована, вам нужно будет вывести эту информацию в пользовательском интерфейсе, в модуле Управление сведениями о продукте, чтобы дать доступ пользователям к управлению ценами продажи.

Следующим шагом будет корректировка процесса использования продукта для распространения цены в используемые продукты. Этого можно достичь модификацией класса *EcoResProductReleaseManager*, ответственного за создание используемых продуктов. А именно, вам нужно будет установить нужное значение в таблице *InventTableModule*, хранящей цены продажи, покупки и производства по умолчанию для используемых продуктов этого юридического лица.

Если общая цена продажи продукта будет изменена, новая цена продажи должна распространиться по всем юридическим лицам, для которых использован этот продукт. Один из путей достижения этого – добавить данную логику в метод *update* таблицы *EcoResProduct*.

Возможная концептуальная модель показана на рис. 19-3

Более подробно об инфраструктуре модели продукта рассказывается в техническом описании «Implementing the Item-Product Data Management Framework», которое можно загрузить по адресу: <http://technet.microsoft.com/EN-US/library/hh272877>.

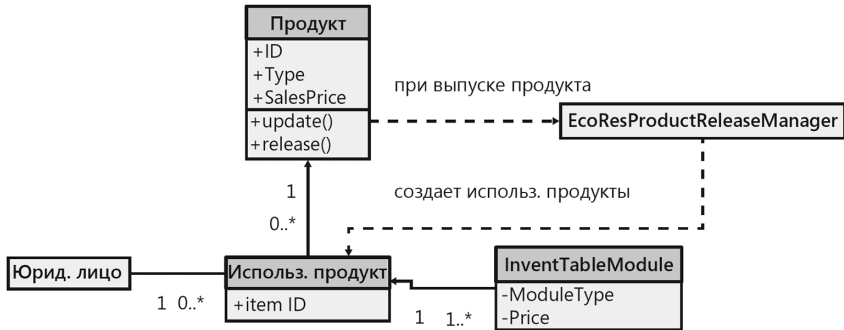


Рис. 19-3. Модель изменений

Инфраструктура ресурса операции

Проектирование производственного процесса в рамках ERP-системы традиционно выполнялось в виде описания того, какая деятельность должна быть осуществлена и кто должен это делать. Для этого проектировщик процесса должен знать не только, как создается продукт, но и какие ресурсы доступны для его создания. В Microsoft Dynamics AX 2012 реализована новая модель, позволяющая отделить процесс от ресурсов, чтобы можно было описывать процесс без необходимости ссылаться на конкретные ресурсы.

Как работает инфраструктура ресурса операций

Основной сущностью в модели ресурса операций является *Ресурс*, который определяется как что-либо, используемое для создания, производства или доставки товара или услуги и не являющееся материалами, которые потребляются в этом процессе. Существует множество типов ресурсов: Инструмент, Оборудование, Персонал, Местоположение и Поставщик.

Ресурс может быть членом *Группы ресурсов*, и *Членство в группе ресурсов* может со временем изменяться. *Группу ресурсов* можно рассматривать как средство организации *Ресурсов*. *Группа ресурсов* располагается в определенном сайте. *Ресурс* может входить только в одну *Группу ресурсов* в каждый момент времени.

Ресурс не обязан входить в *Группу ресурсов*, однако *Ресурс* рассматривается для планирования только в тот период или периоды, когда он связан с *Группой ресурсов*. Концептуальная модель предметной области для *Ресурсов* и *Групп ресурсов* показана на рис. 19-4.



Рис. 19-4. Ресурсы и Группы ресурсов

Возможности

Возможность – это способность ресурса выполнять заданную деятельность, такую как сварка, прессовка или мойка полов. *Ресурс* в одно и то же время может иметь множество *Возможностей*. Для каждого назначения вы можете устанавливать приоритет и уровень, с которыми *Возможность* может выполняться, например штамповка с давлением 4 тонны. Концептуальная модель предметной области для *Ресурсов* и *Возможностей* показана на рис. 19-5.

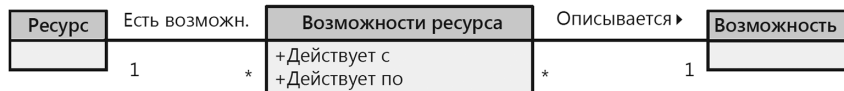


Рис. 19-5. Ресурсы и Возможности

Возможность может быть назначена любому *Ресурсу* независимо от его типа. Если *Ресурс* относится к типу *Персонал*, который ассоциирован с информацией о рабочем, навыках, курсах, сертификатах и названии должности из модуля *Управления персоналом*, то вы можете использовать эту информацию в дополнение к *Возможности* для определения компетенций ресурса.

Мероприятия и требования

Мероприятие – это обобщенная абстракция единицы работы, выполняемой одним или несколькими *Ресурсами*. Сущность *Мероприятия* сама по себе не видна пользователю, но используется внутри Microsoft Dynamics AX для обобщенного представления следующих бизнес-сущностей: *Часовой прогноз* (Проект), *Маршрут производства*, *Связь операции*, *Связь операции с моделью продукта* и *Связь операции с конфигуратором продукции*. Концептуальная модель предметной области для *Мероприятия* показана на рис. 19-6.



Рис. 19-6. Модель *Мероприятие*

Каждое *Мероприятие* может иметь набор *Требований мероприятия*, которые указывают, как много *Ресурсов* необходимо для *Мероприятия* и какими возможностями *Ресурсы* должны обладать для участия в *Мероприятии*. Несколько *Требований мероприятия* могут быть заключены в *Набор требований мероприятия*. Для того чтобы *Ресурс* подходил для *Мероприятия*, он должен отвечать всем требованиям в *Наборе требований мероприятия*. Для каждого *Требования мероприятия* вы можете указать, должно ли *Требование* учитываться при планировании операций или планировании заданий. Концептуальная модель предметной области для *Мероприятия*, *Наборов требований мероприятия* и *Требования мероприятия* показана на рис. 19-7.

Определение подходящих ресурсов

Поиск *Ресурсов*, которые подходят для *Мероприятия*, требует, чтобы была известна как минимум следующая информация.

- Дата, на которую выполняется поиск, поскольку информация о *Ресурсе* и его членстве может изменяться со временем.

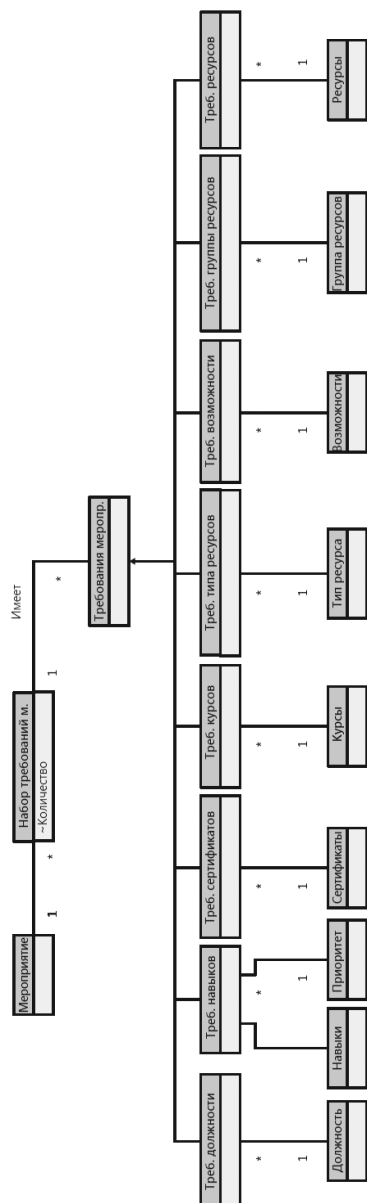


Рис. 19–7. Мероприятия, Наборы требований мероприятия и Требования мероприятия

- Контекст сайта, поскольку в большинстве случаев сайт будет ограничивающим фактором, так как ресурсы должны быть членами ресурсной группы на том сайте, где осуществляется производство.
- Метод планирования, поскольку *Требование мероприятия* может быть применимо либо к планированию операций, либо к планированию заданий, либо к обоим.

Концептуально определить подходящий *Ресурс* просто – это лишь вопрос перебора всех *Ресурсов* и выполнения проверки того, соответствуют ли навыки, возможности, тип *Ресурса* и т.д. тем, которые заявлены в требованиях, а также проверка того, что *Ресурс* не связан с *Группой ресурсов*, которая помечена как ячейка бережливого производства.

В коде вы можете найти *Ресурсы*, подходящие для *Мероприятия*, с помощью одного из двух основных прикладных интерфейсов (API), предлагаемых для набора требований мероприятия:

- ***applicableResourcesList*** возвращает идентификаторы всех подходящих ресурсов в виде списка;
- ***applicableResourcesQuery*** возвращает объект запроса с таблицей *WrkCtrTable* в качестве основного источника данных.

Когда *Мероприятие* спланировано движком планирования, выбранный *Ресурс* (или *Ресурсы*) может быть найден с помощью запроса резервирования мощностей.

Когда использовать инфраструктуру ресурса операций

Вы можете использовать инфраструктуру ресурса операций для любого мероприятия, которому требуется один или более ресурсов. Инфраструктура обеспечивает хорошую интеграцию с движком планирования, который может выполнять отбор ресурсов и выделение времени в соответствии с требованиями и приоритетами.

Расширение инфраструктуры ресурса операций

Вы можете расширить инфраструктуру ресурса операций как минимум в двух направлениях: за счет добавления новых классов мероприятий и за счет добавления новых классов требований мероприятия. В следующем разделе дается подробная информация о точках интеграции и приводятся некоторые соображения, которые следует учитывать.

Добавление нового класса мероприятия

Чтобы добавить новый класс мероприятия, сначала необходима основная таблица (в примере называется *X*), содержащая определение мероприятия, включая задачи, которые необходимо выполнить, длительность, ссылки на другие мероприятия и т.д. Для связи этой таблицы с обобщенным мероприятием создайте новую таблицу *WrkCtrXActivity*, имеющую связь 0-1 с таблицей *WrkCtrActivity* и связь 1-1 с таблицей *X*. Имея такую готовую структуру данных, можно создать форму, в которой пользователи смогут заполнять требования мероприятия для вашей таблицы *X* и переходить далее к отображению получившихся подходящих ресурсов. Форма *ProdRoute* является хорошим примером того, как можно реализовать подобную форму и какая логика необходима для управления *WrkCtrActivityRequirement* и связанных источников данных.

Если мероприятие должно быть спланировано, то это можно сделать с помощью того же движка планирования ресурсов, который используется для сводного планирования, заказов в производство и проектов. Основной класс – это *WrkCtrScheduler*. Для каждого типа мероприятий должен быть свой производный класс, такой как *WrkCtrScheduler_Proj*, который содержит информацию о том, как работать с данным типом. В минимальном виде в классе должны быть реализованы следующие методы.

- **loadData.** Подает в движок информацию о количестве мероприятий, которые необходимо спланировать, длительности использования подходящих ресурсов, зависимостях, связях между мероприятиями и т.д.
- **saveData.** Проходит по результатам, полученным из основного движка, сохраняя начальное и конечное время в мероприятии и создавая резервирования мощностей в таблице *WrkCtrCapRes*. Рекомендуется добавить новое значение в перечисление *WrkCtrCapRefType* и использовать это значение при сохранении резервирований мощностей. За счет этого будет проще определять, кому принадлежит резерв.

Добавление нового класса требования мероприятия

Если существует информация, относящаяся или напрямую с ресурсом операции, или с поставщиком, который связан с ресурсом, и эта информация определяет возможности ресурсов по реализации мероприятия, то вы можете включить эту информацию в процесс выбора ресурсов.

Если информация, относящаяся к ресурсу, хранится в таблице *Y*, то сначала создайте новое значение в перечислении *WrkCtrActivityRequire-*

mentType для представления сущности *Y*. Затем добавьте новую таблицу с названием *WrkCtrActivityYRequirement*, содержащую внешние ключи для связи с таблицей *WrkCtrActivity* и таблицей *Y*. Поскольку большая часть логики, связанной с требованиями к ресурсам, полагается на механизмы отражения, на новой таблице должны быть реализованы определенные методы. Используйте таблицу *WrkCtrActivityPersonTitleRequirement* в качестве примера того, какие методы необходимы.

После того как таблица требований реализована, необходимо изменить приложение в нескольких местах, чтобы учитывать новую таблицу. Лучше всего найти такие места с помощью перекрестных ссылок на одну из существующих таблиц, такую как *WrkCtrActivityPersonTitleRequirement*, и затем добавить обработку новой таблицы аналогичным образом.

Из соображений производительности движок планирования выполняет сопоставление требований к ресурсам для мероприятия и реальных возможностей ресурса за счет преобразования возможностей, навыков, сертификатов и т.д. в обобщенное свойство, которое можно сравнивать с требованиями за счет простого сравнения строк. Это преобразование выполняется методами *computeResourceCapabilities* и *computeResourceGroupCapabilities* класса *WrkCtrSchedulingInteropDataProvider*, которые также должны принимать во внимание информацию из таблицы *Y*. Нужно взвешенно подойти к решению, должно ли новое требование быть доступно для планирования заданий и операций. Если требование должно быть доступно для планирования операций, то необходимо сохранять и поддерживать используемую мощность для группы с учетом свойства *Y*, а также резервирование мощности, чтобы избежать избыточного резервирования. Эта возможность во время планирования требует больших затрат с точки зрения производительности.

Префиксы названий элементов модели MorphX для инфраструктуры ресурсов операций

Все элементы, которые связаны с моделью ресурсов операций, называются *WrkCtr**. Большинство названий схожи с концептуальными названиями, за исключением сущности *Ресурса*, которая по историческим причинам хранится в таблице *WrkCtrTable*. Более подробную информацию об инфраструктуре модели ресурса операций вы можете найти в следующих документах из серии «Core Concepts» на сайте Microsoft Dynamics InformationSource (<http://informationsource.dynamics.com>):

- Allocating resources based on resource requirements (распределение ресурсов на основе требований к ресурсам);
- Operations scheduling based on capabilities (планирование операций на основе возможностей).

Для доступа к этим документам зайдите под своей учетной записью на сайт **Microsoft Dynamics InformationSource**, щелкните **Library** и затем введите название документа в окне поиска.

Инфраструктура финансовых аналитик

Инфраструктура финансовых аналитик предоставляет способ отслеживания дополнительных элементов информации, таких как подразделение, центр затрат или цель, применительно ко всем документам в приложении. Эта информация может использоваться в финансовом учете для категоризации информации.

Как работает инфраструктура финансовых аналитик

Атрибут аналитики – это тип информации, отслеживаемой инфраструктурой финансовых аналитик. Множество значений атрибута аналитики определяется экземплярами бизнес-сущностей, которые существуют для соответствующего типа бизнес-сущности. К примеру, таблица *OMOperatingUnit* может предоставить список значений для аналитики «организационная единица». *Атрибуты аналитики* могут быть помещены в *Иерархию аналитик* для указания порядка следования. Например, одной из конкретизаций в *Иерархии аналитик* является *Структура счета*. *Атрибуты аналитики* могут быть сгруппированы в *Набор атрибутов аналитики*, используемый в некоторых *Настройках данных* для указания *Атрибутов аналитики*, применимых в определенных ситуациях, например, для флажка рядом с каждой аналитикой на форме Распределения ГК.

Концептуальная модель предметной области для инфраструктуры финансовых аналитик показана на рис. 19-8.

Существуют четыре основных шаблона хранения для показа и отслеживания информации об аналитиках.

- *Финансовые аналитики* – это упорядоченные наборы *Значений атрибутов аналитики*, которые ограничены структурой счета и дополнительными правилами учета, например, *Sales-11005-NorthAmerica-*

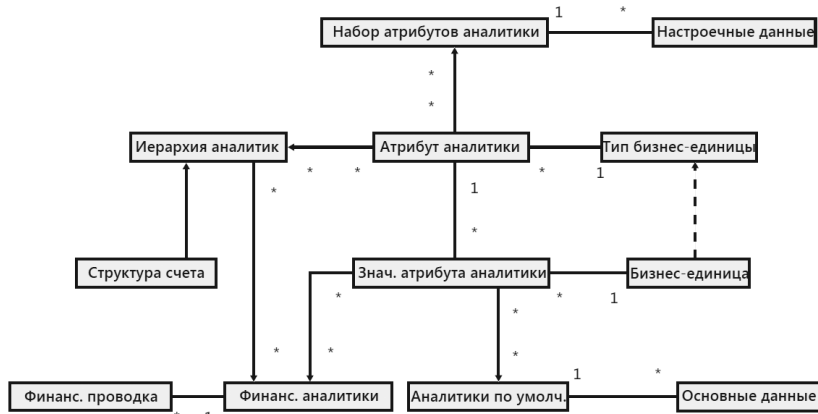


Рис. 19-8. Инфраструктура финансовых аналитик

Xbox-70004. Этот шаблон обычно используется в финансовых данных, таких как строки журналов.

- *Аналитики по умолчанию* – это неупорядоченные неограниченные наборы *Значений атрибутов аналитики*. Например, для записи в *Cust-Table* может быть установлено *SalesRegion=NorthAmerica*. Это значение затем будет использоваться по умолчанию в качестве *Финансовой аналитики*, когда в заказе на продажу будет использоваться запись клиента. Когда *Аналитики по умолчанию* отображаются на форме, то отображаются все аналитики, используемые в плане счетов для текущего юридического лица.
- *Наборы атрибутов аналитики* – это неупорядоченные наборы *Атрибутов аналитики*, где с каждым *Атрибутом аналитики* связано значение перечисления. Например, в процессе распределения пользователь может отметить, какие *Атрибуты аналитики* должны браться по умолчанию из исходной проводки, а какие должны принять указанное значение. Этот шаблон используется нечасто.
- *Наборы аналитик* – это упорядоченные наборы значений аналитик, похожие на *Финансовые аналитики*, но без требования, что они должны содержать счет ГК. Они используются, главным образом, в отчетности и отслеживании баланса. Например, чтобы отобразить страницу списка пробного баланса для счета ГК и подразделения, нужно создать набор аналитик, содержащий эти две аналитики.

Четыре основных шаблона далее конкретизируются в десятках специфических способов использования. Вот некоторые из наиболее часто употребляемых.

- Шаблон *Счет по умолчанию* является конкретизацией шаблона хранения *Финансовые аналитики*. Экземпляры шаблона *Счет по умолчанию* хранятся как стандартный счет ГК, но содержат лишь значение *Атрибута аналитики* *Счет ГК*. Этот шаблон используется в таких областях, как профили разности, для указания, какой счет ГК необходимо использовать, когда финансовыми обработками создается *Финансовая аналитика*.
- Шаблон *Динамический счет* – это конкретизация шаблона хранения *Финансовая аналитика*. Данный шаблон используется в журналах, где доступно поле *Тип счета*. Когда *Тип счета* установлен в *Счет ГК*, он ведет себя как обычный счет *Финансовой аналитики*. Когда же тип счета установлен во что-то другое, например *Клиент*, он ведет себя как поле выбора счетов клиентов. При использовании типа счета, отличного от *Счета ГК*, используется предопределенный скрытый *Атрибут аналитики*, чтобы обозначить клиента, поставщика, номенклатуру или иную сущность, соответствующую типу счета.

Кроме этого, шаблоны учета отражаются в разнообразных шаблонах бюджетирования.

Ограничение комбинации значений

Вы можете ограничить комбинации значений, которые допустимы в *Финансовых аналитиках*, двумя способами. Если ограничения настроены в дереве на форме *Настройка структур счета* (Главная книга > Настройка > Финансовые аналитики > Настройка структур счета), эти ограничения хранятся в таблицах *DimensionConstraintNode* и *DimensionConstraintNodeCriteria*. Поскольку структура данных в этих таблицах очень сложная, для выполнения проверок намного проще не читать их напрямую, а использовать метод *DimensionValidation::validateByTree*. Метод *validateByTree* проверяет, что *Финансовая аналитика* соответствует ограничениям, указанным в этих таблицах.

Другой способ ограничения значений – это нажать кнопку *Отношения* на панели операций формы *Настройка структур счета* и затем использовать форму *Выберите отношения*, чтобы указать отношения, которые вы хотите применить к структуре счета. Эта форма показывает все иерархии организационной модели, которые содержат типы организационной моде-

ли, используемые в качестве сущностей, на основе которых заданы *Атрибуты аналитики* в текущей иерархии. Например, если структура счета содержит подразделения и центры затрат и существует организационная модель, где соотносятся подразделения и центры затрат, то эта информация появится на данной форме. Информация будет отображена дважды – с сущностями **А** и **В**, **поменявшимися местами. Это позволяет системно** администратору указать, должны ли допустимые подразделения быть родителями или подчиненными для данного центра затрат. Схожим образом эти ограничения организационной модели применяются, когда вы используете методы класса.

Создание значений

Вы можете программно создавать *Финансовые аналитики* двумя способами. Для их явного создания используйте класс *DimensionStorage*. Он дает возможность добавить несколько иерархий и значений. Когда вы вызываете метод *save*, класс пытается найти существующие комбинации и, если нужной комбинации нет, то создает новую. *Финансовые аналитики* являются постоянными, и для каждой комбинации существует только одна, поэтому если один и тот же счет использован дважды, то этот метод гарантирует, что только один экземпляр будет создан в базе данных. При работе с существующими счетами по умолчанию и финансовыми аналитиками вы можете использовать класс *DimensionDefaultingService* для объединения значений в новые комбинации. Например, метод *DimensionDefaultingService::serviceCreateLedgerDimension* принимает счет и одну или более *Аналитик по умолчанию*, а затем объединяет их для формирования полной *Финансовой аналитики*.

Расширение инфраструктуры финансовых аналитик

Наиболее частой модификацией инфраструктуры финансовых аналитик является добавление нового типа сущности. Microsoft Dynamics AX 2012 содержит порядка 30 сущностей. Единственным требованием при добавлении нового типа является наличие для сущности естественного ключа, состоящего из одного уникального строкового поля длиной 30 символов или менее. Чтобы добавить новый тип сущности, создайте представление, которое будет являться оберткой для сущности и удовлетворять следующим условиям.

- Название представления должно быть *DimAttribute[название сущности]*. Например, *DimAttributeCustTable*.

- Представление должно содержать корневой источник данных, называющийся *BackingEntity* и связанный с таблицей, которая содержит суррогатный ключ и естественный ключ.
- Представление может при необходимости содержать дополнительные связанные источники данных для поддержки наследования и получения дополнительных полей, например *DirPartyTable*.
- Представление должно содержать следующие поля, называемые именно так, как указано.
 - ▶ **Key.** Поле должно быть суррогатным ключом связанной сущности, например, поле *Int64 RecId*.
 - ▶ **Value.** Поле должно быть естественным ключом связанной сущности, например, поле *str30 AccountNum*.
 - ▶ **Name.** Поле должно указывать на дополнительное описание для сущности, например, поле *str60 Description*.

Если представление удовлетворяет этим критериям, сущность автоматически станет доступна для использования в качестве источника значений аналитики.

Поскольку список типов сущностей, связанных с аналитиками, кэшируется как на клиенте, так и на сервере, новый тип сущности не появится в списке имеющихся сущностей, пока не будет сделан вызов для сброса кэшей либо пока сервер и клиенты не будут перезапущены. Чтобы новая сущность появилась сразу же, сбросьте кэши с помощью меню Сервис > Кэши в рабочей области разработки.

Получение данных

Атрибуты аналитики являются данными и могут добавляться или удаляться пользователем. Это означает, что конкретные аналитики не должны использоваться напрямую в коде, поскольку нет никакой гарантии, что та или иная аналитика существует; вместо этого аналитики следует рассматривать как настраиваемые данные. Исключением из этого правила является *Атрибут аналитики Счет ГК*. Во всех инсталляциях гарантированно есть лишь один *Атрибут аналитики*, который связан со *Счетами ГК*. Для получения этого *Атрибута аналитики* используйте метод *DimensionAttribute::getMainAccountDimensionAttribute*.

Получение информации об аналитиках зависит от используемого шаблона. В случае *Финансовой аналитики* может использоваться как полная

комбинация, так и отдельные составные части. Для получения полной комбинации создайте объединение с таблицей *DimensionAttributeValueCombination*, как показано в следующем примере.

```
GeneralJournalAccountEntry    gjae;
DimensionAttributeValueCombination  davc;
```

```
select gjae join DisplayValue from davc where
    davc.ReclId == gjae.LedgerDimension;
```

Для получения составной части *Финансовой аналитики* вы можете использовать абстракцию *DimensionAttributeLevelValueView*, чтобы скрыть определенные сложности модели аналитик.

```
GeneralJournalAccountEntry    gjae;
DimensionAttributeLevelValueView  dalvv;
DimensionAttribute              department;
```

```
department = DimensionAttribute::findByName('Department');
```

```
select gjae join DisplayValue from dalvv where
    dalvv.ValueCombinationReclId == gjae.LedgerDimension &&
    dalvv.DimensionAttribute == department.ReclId;
```

Атрибут аналитики Счет ГК является особым случаем. Он был денормализован в таблице *DimensionAttributeValueCombination* для оптимизации производительности запросов его значения, поскольку используется наиболее часто.

```
GeneralJournalAccountEntry    gjae;
DimensionAttributeValueCombination  davc;
MainAccount                    mainAccount;
```

```
select gjae
    join MainAccount from davc where
        davc.ReclId == gjae.LedgerDimension
    join Name from mainAccount where
        mainAccount.ReclId == davc.MainAccount;
```

Выборка значений *Аналитик по умолчанию* делается аналогично *Финансовым аналитикам*, однако у *Аналитик по умолчанию* нет сцепленного представления в виде одной строки, поскольку они являются неупорядоченными наборами.

Абстракция *DimensionAttributeValueSetItemView* для упрощения запроса объединяет таблицы *DimensionAttributeValueSetItem* и *DimensionAttributeValue*.

```

CustTable                                custTable;
DimensionAttributeValueSetItemView       davsiv;
DimensionAttribute                       department;

department = DimensionAttribute::findByName('Department');

select custTable
    join DisplayValue from davsiv where
        davsiv.DimensionAttributeValueSet == custTable.DefaultDimension &&
        davsiv.DimensionAttribute == department.ReclId;

```

Физические ссылки на таблицы

В табл. 19-2 приводится соответствие названий концептов в концептуальной модели предметной области и названий физических таблиц, которые реализуют эти концепты в приложении, где названия могут отличаться.

Табл. 19-2. Соответствие концептов и физических таблиц

Название концепта	Физические таблицы
<i>Финансовая аналитика</i>	DimensionAttributeValueCombination DimensionAttributeValueGroupCombination DimensionAttributeValueGroup DimensionAttributeLevelValue
<i>Аналитика по умолчанию</i>	DimensionAttributeValueSet, DimensionAttributeValueSetItem
<i>Набор атрибутов аналитики</i>	DimensionAttributeSet

Более подробную информацию об инфраструктуре финансовых аналитик вы можете найти в следующих документах:

- Securing Data by Dimension Value by Using Extensible Data Security (XDS) (<http://www.microsoft.com/download/en/details.aspx?id=26921>);
- Implementing the Account and Financial Dimensions Framework (<http://technet.microsoft.com/en-us/library/hh272858.aspx>).

Инфраструктура учета

Инфраструктура учета использует политики и правила для выведения требований учета к суммам и бизнес-событиям, которые документируются в строках документа-источника. Эти политики и правила абстрагировано представлены в виде пяти категорий.

- **Учетная политика.** Используется для определения того, применим ли учет к комбинации бизнес-событий и валютной суммы.
- **Правило вывода счетов ГК.** Используется для определения значений счетов ГК.
- **Провайдер списка счетов ГК.** Используется для предоставления списка комбинаций счетов ГК и того, где они должны использовать (в дебете или кредите).
- **Правило вывода аналитик.** Используется для определения значений аналитик.
- **Правило журнализации учета.** Используется для определения того, какой провайдер списка счетов ГК должен использоваться, и для определения параметров журнализации, таких как тип разnosки.

Инфраструктура учета также отвечает за перенос записей *Журнала субкниги* в *Общий журнал*. Правила переноса *Журнала субкниги* указываются для юридического лица и типа документа-источника и определяют, когда *Журнал субкниги* переносится в *Общий журнал* и должно ли при переносе происходить обобщение.

Как работает инфраструктура учета

Процесс *Распределения по бухгалтерским счетам* создает как минимум одно *Учетное событие*. Группы *Учетных событий* – это набор распределений, зависящий от их учетной даты. Когда заголовок *Документа-источника* передается в *Процессор* для обработки и *Процессор* переводит документ из состояния *В работе* в состояние *Завершено*, вызывается *Процессор журнализации* (журнализатор). Журнализатор обрабатывает все *Учетные события*, связанные с документом и находящиеся в состоянии процесса *Начато*, и переводит их в состояние процесса *Журнализовано*. *Учетная политика* определяет, нужно ли производить учет для сумм и бизнес-событий, задокументированных в *Строке документа-источник*. Если данная *Учетная политика* указывает, что нужно, то журнализатор использует *Правила журнализации*, *Правила вывода счетов ГК*, *Правила вывода аналитик* и *Провайдер списка счетов ГК* для определения комбинации счета ГК и аналитик, которые нужно использовать при создании записей *Журнала субкниги*.

Концептуальная модель предметной области для инфраструктуры учета показана на рис. 19-9.

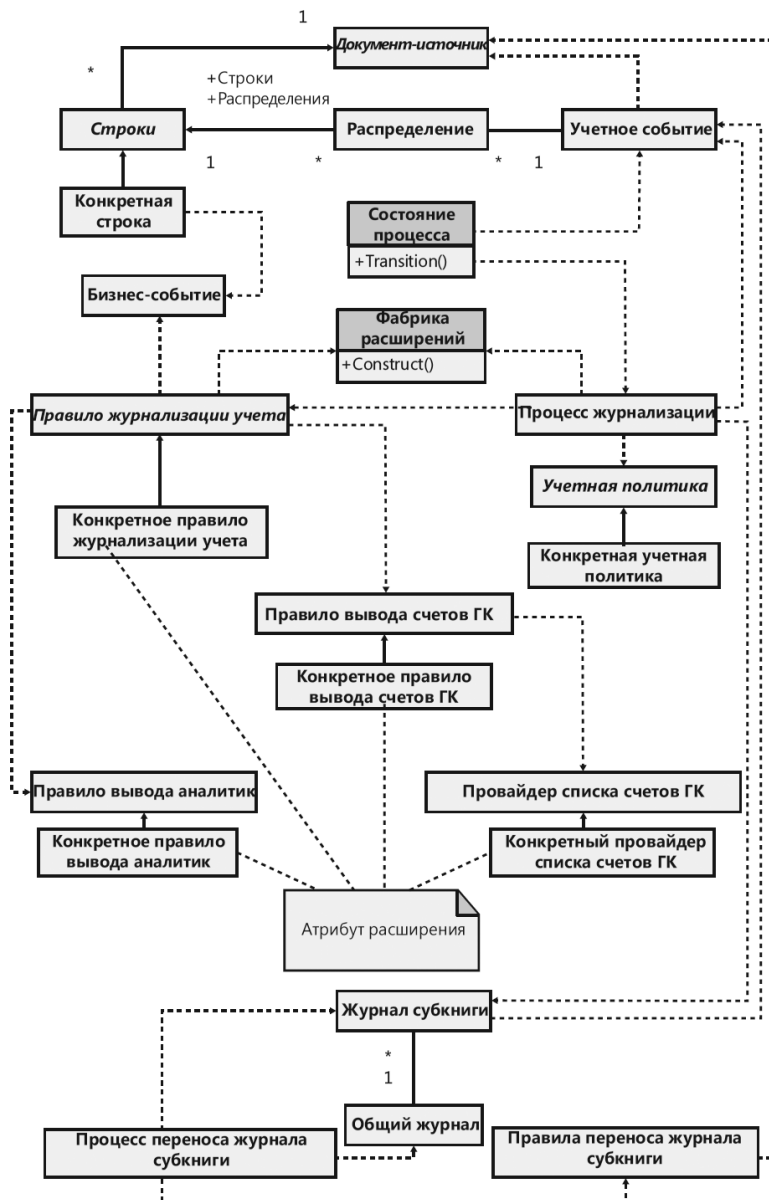


Рис. 19–9. Инфраструктура учета

Правила переноса журнала субкниги, показанные на рис. 19-10, указывают, когда должен происходить перенос (синхронно, асинхронно или в спланированном пакетном задании) и надо ли при переносе в общий журнал «схлопывать» суммы для одинаковых комбинаций счета ГК и аналитик.



Рис. 19-10. Применение правил в процессе учета

Когда использовать инфраструктуру учета

Расширяйте инфраструктуру учета для создания конкретных реализаций учетных политик, правил журнализации, выведения счетов ГК, провайдеров списков счетов ГК и правил выведения аналитик, чтобы поддерживать реализации новых документов-источников. В Microsoft Dynamics AX 2012 инфраструктура учета была расширена для создания конкретных реализаций учетных политик, правил журнализации, выведения счетов ГК и аналитик, чтобы генерировать записи журналов субкниги для таких документов-источников, как заявки на покупку, заказы на покупку, поступление продуктов, накладные поставщика, заявки на командировку, отчеты о расходах и накладные на услуги.

Расширение инфраструктуры учета

Примером расширения компонентов инфраструктуры документа-источника в Microsoft Dynamics AX 2012 являются заявки на покупку (префикс *PurReqSourceDocument*). Учетная политика *AccPolicyCommitFundsExpensedProd* и правило вывода аналитик *AccJourRuleCommitFundsForExpProdExtPrice* являются расширениями инфраструктуры учета, реализующими требования учета для документа заявки на покупку. Это – примеры расширения инфраструктуры учета.

Состояния процесса инфраструктуры учета

Состояния процесса для процесса учета приведены на рис. 19-11. В каждом состоянии обработки выполняется действие и обновляется статус обрабатываемого учетного события. В табл. 19-3 описаны состояния процесса.



Рис. 19-11. Модель состояний для учетного процесса

Табл. 19-3. Состояния процесса для инфраструктуры учета

Состояние	Процесс	Описание
<i>Начато</i>	Процесс распределения по бухгалтерским счетам	Учетное событие создано и ожидает журнализации
<i>Журнализовано</i>	Процесс журнализации субкниги	Были созданы записи журнала субкниги, отразив воздействие с точки зрения учета тех распределений, которые связаны с учетным событием. Записи журнала субкниги не были перенесены в общий журнал
<i>Завершено</i>	Процесс переноса субкниги	Записи журнала субкниги, которые связаны с учетным событием, были перенесены в общий журнал

Префиксы названий элементов модели MorphX для инфраструктуры учета

В табл. 19-4 приведено соответствие названий концептов в концептуальной модели предметной области и префиксов элементов в модели MorphX, которые реализуют эти концепты в приложении.

Табл. 19-4. Соответствие концептов инфраструктуры учета и префиксов элементов модели MorphX

Концепт	Префикс элемента модели MorphX
Учетная политика	<i>AccPolicy</i>
Процесс переноса журнала субкниги	<i>SubledgerJournalTransfer</i>
Учетное событие	<i>AccountingEvent</i>
Учетное правило журнализации	<i>AccJourRule</i>
Правило вывода аналитик	<i>DimensionDerivationRule</i>
Правило вывода счета ГК	<i>MainAccountDerivationRule</i>
Провайдер списка счетов ГК	<i>MainAccountDimensionListProvider</i>
Процесс журнализации	<i>SubledgerJournalization</i> <i>SubledgerJournalizer</i>
Журнал субкниги	<i>SubledgerJournal</i>
Общий журнал	<i>GeneralJournal</i>
Правило переноса журнала субкниги	<i>SubledgerJournalTransferRule</i>

Инфраструктура документа-источника

Документ-источник – это исходная запись, которая документирует факт происхождения одного или более *Бизнес-событий* в учетной системе. *Конкретные документы-источники*, такие как заказы на покупку, поступление продуктов и накладные поставщика, вводятся в учетную систему, а она записывает, классифицирует, отслеживает и строит отчеты относительно качества и стоимости экономических ресурсов, которыми обмениваются или которые вверяют для обмена при выполнении мероприятий, идентифицируемых *Бизнес-событиями*, такими как покупка, получение продуктов и запрос на оплату.

Как работает инфраструктура документов-источников

Инфраструктура документов-источников формирует проекцию конкретного документа-источника для процесса, изменяющего статусы документа-источника, чтобы отразить текущее состояние этого процесса. На рис. 19-12 показана модель предметной области для инфраструктуры документов-источников. Microsoft Dynamics AX передает запись заголовка или строки *Документа-источник* для обработки в *Процессор*, когда пользователь подтверждает, что были выполнены требования по документированию бизнес-события и внутреннему контролю процесса. *Процессор* – это конечный автомат, который переводит обработку документа-источника и его строк из одного *Состояния процесса* в другое. *Процессор* создает объект *Состояния процесса*, который соответствует статусу документа-источника или его строки и затем указывает *Состоянию процесса* перевести процесс в следующее состояние.

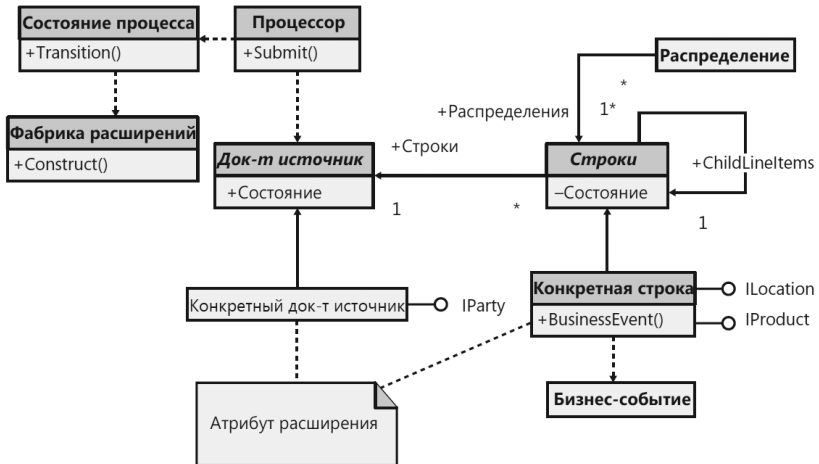


Рис. 19-12. Модель прикладной области для документов-источников

Состояние процесса сначала создает *Конкретный документ-источник* или *Конкретную строку* из переданной записи заголовка или строки *Документа-источника* с помощью фабрики расширений. Фабрика расширений использует тип документа-источника и идентификатор таблицы *Конкретного документа-источника* или *Конкретной строки*, взятые из записи заголовка или строки, чтобы найти соответствующий класс конкретного документа-источника. Соответствующий класс должен иметь

атрибут класса, который *Фабрика расширений* распознает как *Атрибут расширения* и которому также в качестве параметров указаны нужный тип документа-источника и идентификатор таблицы.

Состояние процесса осуществляет доступ к проекции данных *Конкретного документа-источника* и *Строки конкретного документа-источника*, выполняет действие, переводит процесс в новое состояние и обновляет статус процесса *Конкретного документа-источника* или *Строки конкретного документа-источника*, соответственно. Проекция данных *Конкретного документа-источника* и *Строки конкретного документа-источника* определяются одним или более интерфейсами. К примеру, реализация интерфейса *IParty* предоставляет *Состоянию процесса*, которое осуществляет доступ к данным, код счета субъекта, а реализация интерфейса *IProduct* предоставляет код номенклатуры и категорию производства.

Когда использовать инфраструктуру документов-источников

Реализуйте расширения инфраструктуры документов-источников для реализации конкретных документов-источников, которые документируют бизнес-события, чьи финансовые последствия записываются в журналах субкниги. В Microsoft Dynamics AX 2012 инфраструктура документов-источников была расширена для реализации документов-источников заявки на покупку, заказа на покупку, получения продуктов, заявки на командировку, отчета о расходах и накладной на услуги.

Расширение инфраструктуры документов-источников

Накладная на услуги в Microsoft Dynamics AX 2012 (префикс *CustInvoiceSourceDocument*) – это простейшее расширение компонентов инфраструктуры документов-источников. Читатели, не знакомые с этой инфраструктурой, для ее изучения в первую очередь должны обратить внимание на этот пример ее расширения. Каждый конкретный документ-источник и каждая его строка реализуют только те интерфейсы проекции документа-источника, которые требуются обработчиками распределения по бухгалтерским счетам и журнализации субкниг.

Состояния процесса журнализации субкниг и распределения по бухгалтерским счетам приведены на рис. 19-13. В каждом состоянии обработки осуществляется действие и обновляется статус строки докумен-

та-источника, участвующей в процессе. В табл. 19-5 описаны состояния процесса журнализации субкниг и процесса распределения по бухгалтерским счетам.

Табл. 19-5. Состояния журнализации субкниг и распределения по бухгалтерским счетам

Состояние	Процесс	Описание
<i>В обработке</i>	Журнализация субкниг	Состояние достигается в момент создания и изменения документа-источника. Процесс журнализации субкниг переходит в состояние Завершено при подтверждении оригинального или измененного документа-источника
<i>Завершено</i>	Журнализация субкниг	Состояние достигается при подтверждении документа-источника, и документированные последствия бизнес-событий помещены в журнал. Процесс журнализации субкниг переходит в состояние В обработке при изменении документа-источника или же окончании его обработки и запрете дальнейшего его изменения
<i>Окончено</i>	Журнализация субкниг	Состояние достигается, когда документ-источник не может быть более изменен. При окончании его обработки процесс журнализации субкниг балансирует любые открытые счета
<i>Черновик</i>	Распределение по бухгалтерским счетам	Состояние достигается при создании строки документа-источника, или же когда удаляются все распределения по бухгалтерским счетам, относящиеся к строке документа-источника
<i>Полностью распределенный</i>	Распределение по бухгалтерским счетам	Состояние достигается, когда распределение по бухгалтерским счетам впервые используется для распределения суммы, указанной в строке документа-источника, например скидки. Переход в это состояние также происходит при создании распределения по счетам или его порождения из сумм, указанных в строке документа-источника, например суммы без учета скидок и наценок. Кроме того, переход в это состояние происходит, когда сумма значений распределения равна самому распределяемому значению или же был изменен документ-источник

Табл. 19–5. Состояния журнализации субкниг и распределения по бухгалтерским счетам (окончание)

Состояние	Процесс	Описание
Частично распределенный	Распределение по бухгалтерским счетам	Состояние достигается, когда сумма распределяемых значений не равна распределяемому значению
Отменено	Журнализация субкниг Распределение по бухгалтерским счетам	Состояние достигается, когда документ-источник или его строка отменяется. Все распределения по бухгалтерским счетам реверсируются, и следствия отмененного бизнес-события попадают в журнал перед переходом в это состояние

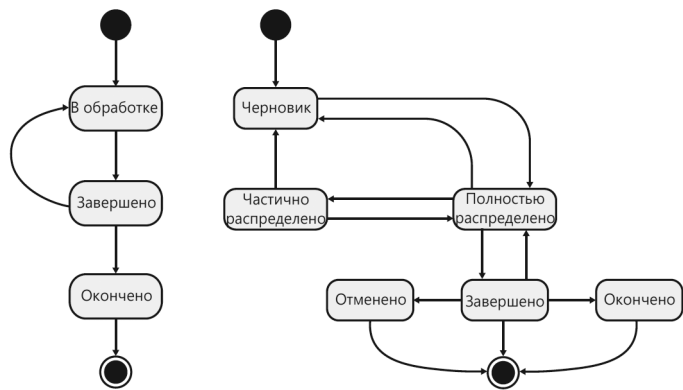


Рис. 19–13. Диаграмма состояний для процесса обработки документов, процесса распределения по бухгалтерским счетам и журнализации субкниг

Префиксы названий элементов модели MorphX для инфраструктуры документов-источников

В табл. 19-6 приведено соответствие названий концептов в концептуальной модели предметной области для инфраструктуры документов-источников и префиксов элементов в модели MorphX, которые реализуют эти концепты в приложении.

Табл. 19–6. Соответствие концептов инфраструктуры документов-источников и префиксов элементов модели MorphX

Концепт	Префикс элемента модели MorphX
Документ-источник	<i>SourceDocument</i>
Строка документа	<i>SourceDocumentLineItem</i>
Процессор	<i>SourceDocumentProcessor</i>
Состояние процесса	<i>SourceDocumentState</i> <i>SourceDocumentLineState</i>
Бизнес-событие	<i>BusinessEvent</i>
Распределение	<i>AccountingDistribution</i>
Фабрика расширений	<i>SysExtension</i>