

## ФУНКЦИОНАЛЬНО-ОНТОЛОГИЧЕСКИЙ ПОДХОД К ПОСТРОЕНИЮ КОРПОРАТИВНОЙ ИНФОРМАЦИОННОЙ СРЕДЫ ВУЗА

*Владивосток, Владивостокский государственный университет экономики и сервиса*

### **1 Введение**

Основной задачей информатизации вуза является построение корпоративной информационной среды (КИС). Многие задачи в вузе не только решаются с использованием информационных технологий (ИТ), но и иницируются их развитием и внедрением. Информационная среда вуза из средства предоставления доступа к информации (независимо от цели, будь то получение информации, анализ данных или принятие решений) превратилась в настоящее время в обязательный компонент жизнедеятельности вуза на всех уровнях: управление, исполнение, обучение.

Рассматривая вуз как систему, можно выделить три основных "элемента", между которыми существует тесная взаимосвязь – это люди, процессы и данные. Эффективное управление бизнес-процессами, оперирующими данными, способствует эффективному функционированию вуза.

В общем случае процессом можно назвать совокупность взаимосвязанных и взаимодействующих видов деятельности, преобразующей входы в выходы. Когда речь идет о менеджменте организации, то процессы организации достаточно условно можно разделить на 4 вида: бизнес-процессы – процессы, которые иницируются за пределами организации, являются откликом организации на некоторое внешнее возбуждение, поддерживающие процессы, управляющие процессы и иные процессы.

В теории развития информационных систем (ИС), когда рассматриваются вопросы автоматизации, под бизнес-процессами понимают все процессы организации, независимо от того, где располагается источник возбуждения и каковы цели этих процессов. В этой работе мы будем рассматривать вопросы обеспечения управления любым процессом в вузе, который может быть автоматизирован, называя его бизнес-процессом в соответствии с укоренившейся в ИТ-литературе традицией.

В 2008 стандарт менеджмента качества ISO 9000 будет пересмотрен, и новый стандарт будет базироваться на японском стандарте JIS/TR Q 0005:2005, в котором прописаны 12 принципов менеджмента качества. Одними из двух новых принципов станут

принципы быстрой реакции на перемены и всеобщей оптимизации. Эти принципы означают, что, во-первых, постоянные изменения стали стандартом и это уже не форс-мажорные обстоятельства, а нормальный рабочий процесс любого предприятия. Во-вторых, требования к скорости реакции на изменения возросли, а следовательно, возросли и требования к скорости внесения изменений в ИС и модели данных, связанные с изменяющимися бизнес-процессами.

Определение. Определим понятие информационной среды как сложной информационной системы (или совокупности информационных систем), которая является активной проекцией реального мира на область ИТ, т.е. информационная среда не только отражает, но и влияет на системы реального мира.

В этом определении заложено фундаментальное понимание информационной среды как активной проекции деятельности вуза на область ИТ. Проекция будет эффективна тогда, когда она помогает повысить качество основных сфер деятельности вуза, а не тогда, когда она наиболее точно отражает деятельность и структуру высшего учебного заведения. Отличительной особенностью современных требований к КИС вуза является активное воздействие ИТ на процессы, которые уже сформированы в вузе, и эффективное участие в постановке новых процессов. Важным элементом определения КИС так же является понимание КИС как сложной системы. В сложной КИС поведение определяется не только поведением отдельных систем, но и их взаимодействием. Сложная КИС может быть неустойчивой, т.е. небольшие нарушения в одной составляющей КИС могут привести к катастрофическим нарушениям в другой. Сложной КИС является так же и потому что объединение ИС в КИС ведет не к суммированию функций, а к их возможно нелинейному увеличению.

Вуз – это инновационная организация, в которой бизнес-процессы изменяются постоянно, так что, программы, автоматизирующие некоторый бизнес-процесс, могут устареть еще до момента внедрения. КИС автоматизирует деятельность вуза, и чем большие сферы затрагивает автоматизация, тем сложнее процесс сопровождения и модификации ИС. Поэтому, часто, начиная с некоторого уровня развития автоматизации бизнес-процессов в вузе, разработчики занимаются в основном сопровождением уже существующих ИС и процесс развития КИС приостанавливается.

Таким образом, задачей развития КИС вуза становится создания методологии проектирования и развития КИС в сторону обеспечения простых способов сопровождения сложной среды, обеспечения быстрой реакции КИС на изменения в системах реального мира.

## 2 Методология проектирования КИС

Для того чтобы КИС быстро и адекватно реагировала на изменяющиеся бизнес-процессы вуза, чтобы была действительно необходимым инструментом работы сотрудников и обучения студентов, КИС должна быть управляема, настраиваема бизнес-аналитиками, т.е. специалистами предметниками, а не только разработчиками и администраторами. Для этого необходимо, чтобы КИС оперировала терминами предметных областей деятельности вуза (образовательной, научно-исследовательской, административного управления, инновационной и других).

В этом контексте КИС можно представить как *совокупность автоматизированных бизнес-процессов, реализующих задачи, стоящие перед вузом.*

Две основные методологии анализа предметной области и проектирования ИС – функционально-структурная и объектно-ориентированная – являются основными методологиями в последние десятилетия [1, 2]. Первая методология предполагает функциональную декомпозицию, выделяющую общие действия в функции, внутри которых, в свою очередь, могут быть выделены функциональные блоки, организованные в отдельные функции. Деление продолжается до элементарных неделимых функций, и программное приложение представляет собой последовательность вызова из одних функций – других функций.

Вторая методология подразумевает выделение объектов, определения поведения и свойств объектов в классах, выделение общего поведения и общих свойств в базовые классы и организация иерархии наследования между классами. Программное приложение представляет собой пересылку сообщений между объектами посредством вызовов методов соответствующих классов.

В последние годы акцент в проектировании ИС сместился в область проектирования предметной области и бизнес-процессов. В связи с этим большое внимание уделяется подходам, основанным на онтологиях [3, 4]. Использование онтологического моделирования стало популярным в связи с идеей семантического Web, для которого разработан и специальный язык OWL (Ontology Web Language ) [5]. Для графического изображения онтологий используется спецификация IDEF5.

Проектирование ИС начинаются с моделирования предметной области. Согласно онтологическому подходу предметная область, описываемая ИС, рассматривается как совокупность взаимосвязанных классов, имеющих некоторые атрибуты и свойства.

В представленной работе предлагается рассматривать предметную область, которая не только описывается, но и управляется информационной средой, как состояния бизнес-процессов, реализующих задачи предметной области в любой момент времени. Такой подход является прямым следствием определения КИС, как управляющего воздействия на предметную область. Управление определяется процессом, а не данными.

Предлагается применять методологии функционального, онтологического и объектно-ориентированного подхода для проектирования сложной КИС, используя функциональный подход для проектирования бизнес-процессов, онтологический – для описания предметной области, в том числе и бизнес-процессов, объектно-ориентированный – для реализации систем на языках программирования.

Таким образом, первой задачей проектирования от процессов, является определение цели. Определенная цель позволяет сформулировать задачи, которые требуется решить. Например, целью образовательного процесса является выпуск специалиста, умения, знания и компетенции которого удовлетворяют некоторым требованиям. Для достижения этой цели необходимо решить различные задачи: сформировать учебную программу, организовать прием студентов на программу, организовать учебный процесс и т.п.

Для решения задач выделяются бизнес-процессы верхнего уровня – создание учебных программ, организация приема студентов и т.п. Каждый бизнес-процесс верхнего уровня может быть разделен на более простые процессы.

Задачей проектирования является декомпозиция до бизнес-процессов, деление которых дальше на данном этапе невозможно. Дальнейшим этапом проектирования является анализ выделенных неделимых бизнес-процессов и их обобщение для создания базовых элементарных бизнес-процессов. В дальнейшем бизнес-процессы могут объединяться, создавая составные бизнес-процессы. Составные бизнес-процессы состоят из элементарных бизнес-процессов и других составных бизнес-процессов.

Для решения своей задачи элементарные бизнес-процессы должны оперировать некоторыми данными предметной области. После выделения элементарных бизнес-процессов, но не ранее, могут быть определены данные предметной области, которые в них задействованы. Выделение терминов предметной области, описание классов, их атрибутов и свойств здесь выполняется на основании определения элементарных бизнес-процессов, точнее их входов, выходов и условий.

Методика проектирования от целей определяется тем понимаем роли КИС, которую среда приобрела в последние годы. Ранее пользователей интересовали данные, ради которых строились ИС, обеспечивающие сбор, обработку и анализ данных (рис.1.а). Сейчас роль КИС состоит в достижении некоторых целей посредством автоматизации бизнес-процессов, которые используют некоторые данные и только те, которые необходимы (рис.1.б).

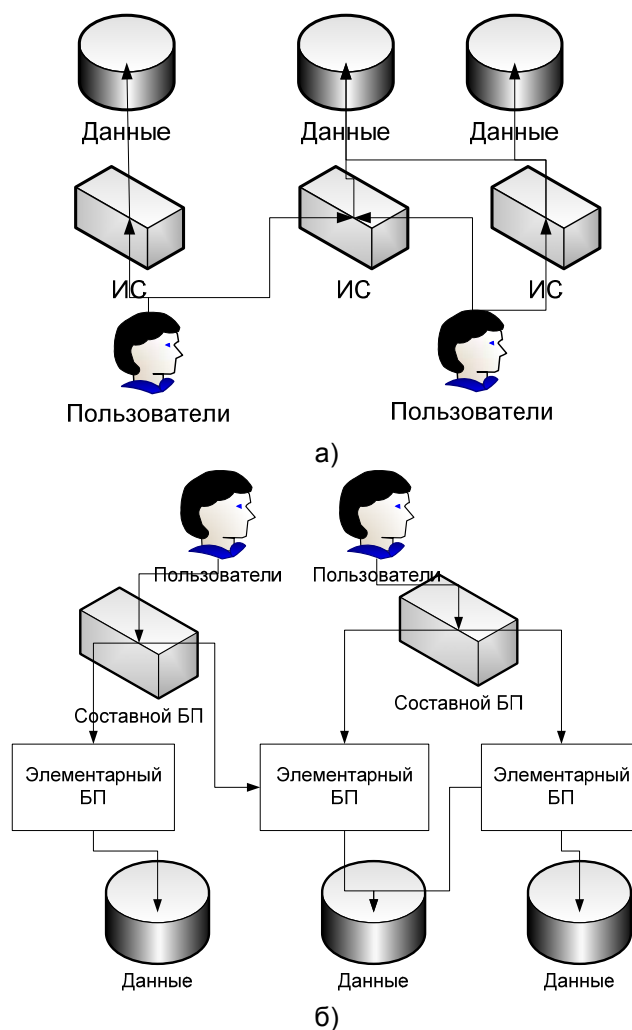


Рис.1. Сравнение двух подходов к разработке ИС

### 3 Обобщенный репозиторий метаданных КИС

Для хранения всех спроектированных процессов и классов в КИС используется Обобщенный репозиторий метаданных (ОРМД). ОРМД хранит описание всех понятий предметных областей, с которыми имеет дело КИС, включая области деятельности вуза, бизнес-процессы и ИТ-область. Условно можно выделить две основных области понятий – понятия предметной области и понятия ИТ-области. Область ИТ выделяется для реализации идеи отображения предметной области на область ИТ и для управления КИС.

К понятиям предметной области относятся такие понятия, например, как *Сотрудник, Организация, Подразделение, Адрес, Студент, Дисциплина, Информационный ресурс, Телефон, Аудитория, Здание, Пластиковая карта, Договор, Тест, Расписание, Бизнес-процесс, Событие* и т.п. Эти понятия используются специалистами-предметниками – учебным управлением, деканатами и кафедрами, управлением планирования и бухучета, управлением персоналом и т.д. Понятия области ИТ – это *Сервер, База данных, Пользователь, Проект, Роль, Таблица, Метод, Параметр, Компонент* и т.п. Эти понятия используются ИТ-специалистами.

#### 3.1 Классы

Каждому понятию соответствует класс, который имеет свойства. В простейшем случае свойства представлены атрибутами, т.е. свойствами, имеющими некоторый встроенный тип.

Между классами поддерживаются произвольные отношения. Эти отношения связывают классы через свойства. Класс **A** имеет произвольные отношения с классом **B**, при этом отношения могут быть представлены

- свойством класса **A** с указанием на класс **B**;
- классом **C**, который имеет отношения с классом **A** и классом **B**.

На самом деле в данном случае речь идет не об отношениях между классами, а об отношениях между экземплярами классов, а именно некоторый экземпляр класса **A** связан с экземпляром класса **B**. Чтобы проиллюстрировать разницу между отношениями между классами и отношениями между экземплярами классов рассмотрим отношения *Проекция классов* на ИТ-область.

Классы имеют проекцию на область базы данных. Проекция – это отношения, которые выражаются через отношения между классами (не экземплярами классов) предметной области и классом

*Источник данных.* Отношения *Проекция* в ОРМД описаны между классом – *базовый ко всем классам КИС* и классом *Источник данных*.

При наличии единой базы данных один класс предметной области имеет отношения проекции с одним экземпляром класса источника данных. В КИС большой организации база данных часто не единая, могут иметь место многочисленные реплики и использоваться различные базы данных. В этом случае определяются отношения классов с несколькими источниками данных. Отношения проекции могут иметь атрибуты, среди которых есть, например, первичность и полнота источника данных. Эти атрибуты участвуют в формировании процедур интеграции данных и приложений, как и собственно отношения Проекции.

Атрибуты класса обычно проецируются на переменные примитивных типов данных и хранятся в источнике данных класса, являясь одним из полей соответствующей таблицы. Свойства класса могут храниться в источнике данных класса, если отношения между классами, которые описывают эти свойства, представляют собой отношения один к одному. Отношения между классами типа многие ко многим или один ко многим всегда описываются свойством класса и хранятся как класс в ином источнике данных.

Классы отличаются друг от друга по имени, в рамках КИС имя класса должно быть уникальным (это не означает уникальность названия таблиц или представлений во всем пространстве данных КИ). Описание отношений между классами так же хранится в ОРМД. В онтологическом подходе отношения всегда бинарные и определены над двумя классами. В КИС имеют место ситуация, когда отношения, являясь бинарными, связывают два класса, но второй класс является одним из ограниченного множества классов. Классическими конструкциями онтологического подхода такие отношения описать не удается, поэтому в ОРМД введено понятие условий, которое используется в различных аспектах описания: в описании отношений, в бизнес-процессах, в управлении событиями, в автоматизации репликаций.

Рассмотрим, например, класс *A - Пользователь КИС*, который имеет отношения «образ в КИС» с классами *B = {Студент, Сотрудник, Внешний пользователь, Родитель студента, Работодатель, Школьник, Проект, Серверная компонента}*, в зависимости от значений одного из атрибутов отношений ( $R_{AB} \rightarrow c$ ). Эта зависимость выражается условием.

$$A \Rightarrow \{B_i, R_{AB} \rightarrow c \in d_i, i = \overline{1, N}\},$$

где  $N$  - число классов в  $B$ ,  $d_i$  - домен значений атрибута  $R_{AB} \rightarrow c$ , при котором выбирается  $B_i$  класс.

Еще одним примером наличия условия в отношениях можно рассмотреть условие в отношениях *Проекция*. Класс Сотрудник может иметь проекции на источники данных, расположенные в филиалах вуза, при условии, что экземпляры класса Сотрудник имеют отношение к филиалу (там работают). Это означает, что в базе данных филиала будут размещены только работающие там сотрудники.

При наличии условий у отношений можно говорить о контекстно-зависимых отношениях.

Определение. Отношения между классом  $A$  и классами  $B$  или  $C$  называются контекстно-зависимыми, если эти отношения определяются в зависимости от условия, накладываемого на атрибуты (свойства) класса  $A$ .

Если между классами  $A$  и  $B$  существуют отношения, которые не зависят ни от одного из атрибутов (свойств) классов  $A$  или  $B$ , то такие отношения называются контекстно-независимыми. Примером таких отношений могут быть отношения агрегации между *институтами* и *кафедрами* или отношения *читается на* между *дисциплинами* и *кафедрами*.

КИС, будучи сложной системой, строится на основании концепции интеграции [6], и имеет свойство, согласно которому любая область автоматизации такой КИС имеет непустое множество пересечений по крайней мере с одной из оставшихся областей автоматизации. Если это так, то значит и классы имеют связи не только внутри области, но и между областями, что совершенно не противоречит требованиям, выдвигаемым к онтологиям быть разделяемыми в различных направлениях.

Внутри одной области автоматизации между классами существует отношения, которые могут связывать классы непосредственно, или косвенно, через свойства других классов. Между любыми двумя классами в интегрированной КИС существуют отношения либо как свойства одного из этих классов, либо через свойства других классов. Если существуют два класса, между которыми нет связи, то либо речь идет не об интегрированной среде, либо такая связь не описана в настоящее время в КИС.

Наличие связей между классами позволяет выполнять поиск, основанный на косвенных связях, но здесь мы сталкиваемся с проблемой неоднозначности выводов.

Рассмотрим пример. Пусть описаны классы Студент, Кафедра, Дисциплина, Образовательная программа, Учебный план,



Преподаватель. Между ними существуют следующие отношения. Студент *учится на* Образовательной программе, Образовательная программа *ведется* Кафедрой, Учебный план *по* Образовательной программе, Учебный план *включает* Дисциплины, Преподаватель *ведет* Дисциплину, Преподаватель *работает на* Кафедре.

В соответствии с этими связями между Студентом и Преподавателем возможны два типа отношений:

1. студент имеет отношения к тем преподавателям, которые ведут дисциплины его учебного плана (т.е. преподаватели, которые учат конкретного студента);

2. студент имеет отношение к тем преподавателям, которые работают на кафедре, отвечающей за обучение студента по образовательной программе (т.е. преподаватели, которые работают на выпускающей студента кафедре).

Таким образом, если мы захотим найти всех преподавателей, связанных с некоторым студентом, то будет получено два пересекающихся, но несовпадающих множества преподавателей. Поскольку все классы в интегрированной среде связаны друг с другом, то таких вариантов будет значительно больше.

Одно из решений этой неоднозначности состоит во введении новых отношений, которые не только описывают связи между двумя классам, но и реализуют эти отношения через цепочку связей. Например, в ОРМД определены такие отношения Преподавателей, которые *учат* Студента как

Студент->Образовательная программа ->Учебный план->Дисциплина->Преподаватель

С другой стороны могут быть так же определены отношения между Преподавателем, которые *работает на* кафедре, отвечающей за выпуск Студента

Студент->Образовательная программа->Кафедра->Преподаватель

Для таких отношений не требуется наличие экземпляров класса отношений. Для них достаточно определение цепочки связи и наличие экземпляров соответствующих классов в цепочке. Т.е. для таких отношений достаточно описание цепочки классов без необходимости хранить экземпляры. Эти отношения могут затем использоваться в организации высокоуровневых запросов и поиске экземпляров классов.

### **3.2 Атрибуты и свойства**

Свойство может иметь ограничения по допустимости значений. Если свойство имеет ограничения некоторым доменом, то в

большинстве случаев мы имеем дело не с атрибутом, а с отношениями. Свойства могут иметь ограничения и по значению, не связанному с доменом, например, по диапазону, или какому-то регулярному выражению. Например, сотрудник может иметь ограниченный по типу строка атрибут - фамилию, ограниченный по типу целого и по диапазону значения - год рождения, и ограниченный по домену - уровень образования. Атрибут уровень образования ограничен доменом, состоящим из нескольких значений (начальное профессиональное / среднее общее / среднее профессиональное / высшее профессиональное / аспирантура). Для такого ограничения используется отдельное понятие Уровень образования, которое имеет соответствующий класс с атрибутом название. Классы *сотрудник* и *уровень образования* имеют отношения, выраженные свойством класса *сотрудник* *Уровень образования сотрудника*.

Разделение атрибутов и свойств класса в некоторых случаях не очевидно. Например, выше приведен пример с уровнем образования сотрудника, которое определено как свойство, на том основании, что уровень образования представляет собой отдельное понятие и может быть связано с сотрудником на основании отношений.

Атрибутом класса в ОРМД мы предлагаем считать такую характеристику класса, которая либо не имеет допустимых значений (но может иметь ограничения по типу), либо имеет в качестве ограничений домен, который не используется для идентификации экземпляра класса с заданным атрибутом. Такое дополнение требуется в связи с тем, что мы используем онтологический подход для описания метаданных КИС, в которой данные хранятся в базах данных.

Атрибуты классов имеют ограничения на тип. Кроме этого ограничения атрибуты в ОРМД имеют дополнительные ограничения на значения, не связанные с другим классом. В отношениях проекции атрибуты транслируются в поля таблиц баз данных. Ограничения атрибутов могут быть выражены

1. точным или максимальным размером множества;
2. допустимым множеством или диапазоном значений;
3. регулярным выражением, которое может быть составлено из других атрибутов класса.

Атрибуты могут иметь также значение по умолчанию. Такое значение присваивается атрибуту, если его значение не задано. Значение атрибута по умолчанию может быть константой или регулярным выражением, составленным из атрибутов классов КИС. Значение атрибута по умолчанию используется при организации процедур репликации данных и поддержки качества данных.

В отношении *Проекция* определены соответствия для ограничений

1. ограничение на точную мощность множества = 1 соответствует обязательному присутствию атрибута со значением not null;
2. ограничение на максимальную мощность множества =1 соответствует атрибуту, возможно со значением null;
3. ограничение на точную или максимальную мощность множества > 1 соответствует созданию новой или использованию дополнительной таблицы с полями идентификатор экземпляров класса и значения атрибута;
4. допустимое множество значений для атрибутов, диапазон значений, регулярные выражения транслируются в ограничения на поля таблицы (CHECK).

Так как в ОРМД отношения проекции создаются на основании уже существующих таблиц (идею автоматической генерации таблиц на основании описания в ОРМД мы здесь не рассматриваем) и представлений, то все необходимые ограничения на поля таблиц должны быть описаны в ОРМД (описание может выполняться в полуавтоматическом режиме). Необходимость описания определяется использованием классов в различных проектах и обеспечение процедур качества данных. Например, в процедуре автоматической интеграции данных используются значения по умолчанию для свойства, в случае, когда в реплике отсутствует экземпляр класса, с которым это свойство определяет связь. При отсутствии описания значения по умолчанию в этом случае автоматическая репликация данных не выполнится.

Если для свойств отношения проекции определена максимальная или точная мощность множества, равная 1, то источники данных класса  $x$  и отношения  $P(x,y)$  совпадают. Это означает, что одному экземпляру класса  $x$  соответствует единственный экземпляр класса  $y$  и это описано в той же таблице, что и  $x$ . Если определено только максимальная мощность множества отношений как 1, то это означает, что экземпляр класса  $x$  сопоставляется не более, чем с одним экземпляром класса  $y$ , т.е. для атрибута источника данных допускается равенство null. Если для отношений определены максимальное или точное ограничение > 1 или определено минимальное ограничение  $\geq 1$ , то источник данных отношений отличается от источника данных класса. Описание мощности множества позволяет, например, контролировать качество данных – критерии полноты, достоверности, корректности данных используя ограничения на мощность множества.

#### 4 Управление бизнес-процессами

Изменения реального мира требует изменения бизнес-процессов, что в свою очередь ведет к необходимости изменения процедур их автоматизации. Онтологии позволяют обеспечивать быстрые изменения в процедурах автоматизации, так как являются фактически переводчиками (адаптерами) между реальным миром и ИС. Использование онтологий в реализации бизнес-процессов позволяют контролировать их корректность, эффективность, оптимальность.

Целью подхода управления бизнес-процессов на основе онтологий является создание инструмента, который бы позволил бизнес-аналитикам, специалистам предметникам самим формировать бизнес-процессы, определять порядок и условия следования действий в бизнес-процессе. Такой подход позволяет своевременно реагировать на изменения бизнес-процесса в системах их управления, а так же позволяет разработчикам КИС оперативно создавать новые ИС.

Для управления бизнес-процессами необходимо разработать схему управления на основании классов и атрибутов. ОРМД содержит описания всех необходимых классов, которые используются в управлении бизнес-процессов, и описания бизнес-процессов, а так же схем их функционирования. Рассмотрим классы, обеспечивающие управление бизнес-процессами в КИС.

Элементарные бизнес-процессы реализуют логику, которая реализована в программах, процедурах, методах серверных компонентов, по средством регулярных выражений или условий, использующих атрибуты классов КИС.

Пользователи КИС могут формировать составные бизнес-процессы из элементарных и других составных бизнес-процессов. Понятие «состоять из» означает, что внутри составного бизнес-процесса элементарные и другие составные бизнес-процессы могут выполняться последовательно или параллельно.

Последовательность действий управляется либо последовательной связью между бизнес-процессами, либо блоком условий. Условия могут комбинироваться по «И», «ИЛИ» или «Не». В условиях используются классы КИС и их атрибуты. Собственно условия так же являются классом КИС и именно тем классом, который мы рассматривали выше в формировании контекстно-зависимой связи. В условиях атрибуты классов могут быть равными, меньше или больше некоторой величины.

Класс элементарный бизнес-процесс имеет отношения «реализуется» с классом серверной компоненты (Рис. 2). Серверная

компонента – это компонента КИС, которая реализует некоторую логику работы, имеет интерфейс, входные параметры и в отдельных случаях выходные параметры. В серверные компоненты выносятся взаимодействие с базами данных и вычислительные операции. Например, процедура аутентификации и авторизации пользователей КИС выполняется во всех системах КИС. Для реализации этой процедуры используется серверная компонента, имеющая несколько методов, позволяющих выполнять аутентификацию или авторизацию пользователя, получать информацию о пользователях с некоторой ролью и т.п. [6]. Серверные компоненты реализуются с использованием различных технологий – веб-службы, CORBA, DCOM, хранимые процедуры и функции баз данных, веб-приложения. Серверные компоненты описаны в ОРМД и сами и их методы. При описании методов серверных компонентов в ОРМД используются файлы – описания, например, WSDL или IDL. В этом случае описание выполняется автоматически (вручную вносится семантическое описание методов). При использовании в качестве методов – хранимых процедур или функций используются описания, хранимые в метаданных базы данных. В отдельных случаях (для веб-приложений) описания выполняются полностью вручную.

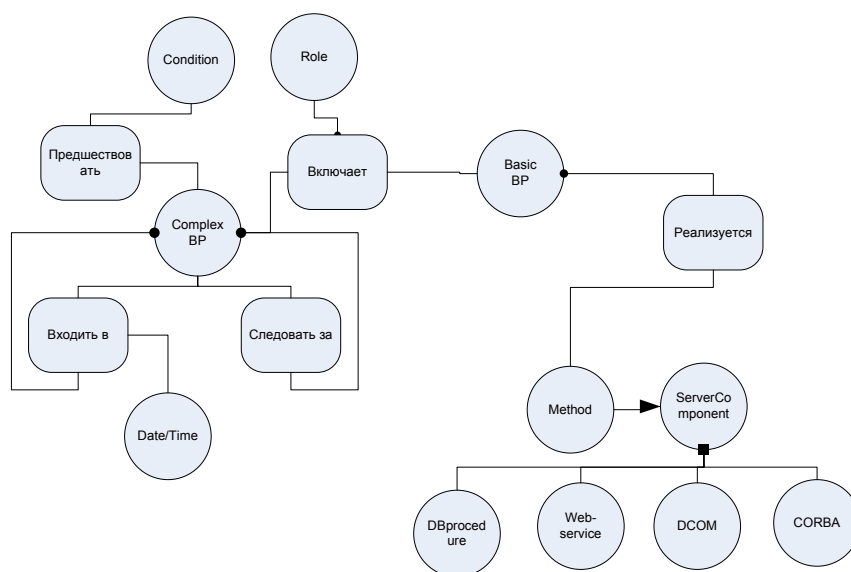


Рис.2. Классы и отношения бизнес-процессов

Класс Метод серверных компонентов имеет отношения «является частью» с классом серверная компонента. Класс серверная компонента является базовым к классам CORBA-компонента, DCOM-компонента, веб-служба, база данных, веб-приложение.

Доступ к элементарным бизнес-процессам внутри составного бизнес-процесса определены для ролей пользователей проекта (модель управления правами пользователей КИС рассматривалась в работе [7]). Это означает, что пользователь, который имеет роль, может или должен выполнить действие, определенное элементарным бизнес-процессом. Здесь под ролью понимается не набор функциональности в рамках проекта, а способ выделения подмножества пользователей, которым в системе управления правами (а не в проекте) задается набор функциональности. Это позволяет значительно более гибко подходить к разработке ИС.

Класс составной бизнес-процесс имеет отношения «следовать за» с самим собой, что позволяет организовать маршрут процессов, «входить в» с самим собой, что позволяет определять составной бизнес-процесс и отношения «включает» с классом элементарный бизнес-процесс, что позволяет детализировать составной бизнес-процесс до элементарного бизнес-процесса.

Отношения «входить в» имеют атрибутами дату и время начала и окончания процесса, периодичность, способ извещения о невыполнении процесса. Отношения «включает» имеют атрибутом связь с ролью. При связи с ролью существуют ограничения на область видимости, что позволяет пользователям выполнять элементарные бизнес-процессы над некоторым подмножеством данных, доступных в этом процессе.

Когда некоторый бизнес-процесс вуза изменяется должны быть изменены и составные бизнес-процессы. В некоторых случаях требуется разработка новых элементарных бизнес-процессов. Новые элементарные бизнес-процессы описываются в ОРМД. В определенных случаях изменения бизнес-процессов приводит к необходимости вводить новые классы, тогда от администратора КИС требуется описание этих новых понятий в ОРМД с определением всех отношений, в том числе отношений проекция между классами и экземплярами класса источник данных.

Реализация элементарных бизнес-процессов выполняется как разработчиками, так и в отдельных случаях бизнес-аналитиками. Использование бизнес-процессов, создание составных бизнес-процессов, определение условий выполняется бизнес-аналитиком.

#### **4.1 Контроль выполнения на основании событий**

Рассмотрим проблемы, связанные с вопросами контроля за выполнением бизнес-процессов. Первое, что представляет здесь интерес - это информирование заинтересованных участников процесса в необходимости выполнения ими назначенных им элементарных бизнес-процессов. Информирование заинтересованных пользователей может быть выполнено уведомлением по почте, по SMS или появлением доступа в системе, которая управляет бизнес-процессами.

Появление доступа в системе управления бизнес-процессами не представляет сложностей, так как система при каждом входе пользователя проверяет все доступные пользователю элементарные бизнес-процессы и дает доступ к тем, которые назначены пользователю, и разрешены с точки зрения предыдущих процессов.

Вопрос об извещении по почте несколько более сложный. Проблема связана с тем, что в данном случае речь идет не о модели запрос-ответ, а о модели событий, когда сервер извещает клиента о некотором событии. События, являясь классом КИС, используются для организации интеграции данных (репликаций и обеспечения качества), интеграции приложений и контроля за выполнением бизнес-процессов.

Событие определяется как значимое отражение изменения в состоянии некоторого ресурса и оно генерируется, чтобы известить об этом изменении [3]. С точки зрения онтологического подхода определим класс событие как отражение значимого изменения в состоянии некоторого экземпляра класса КИС или в связи с появлением или удалением экземпляра класса, а так же с ограничением на число экземпляров класса.

Рассмотрим вопрос контроля на примере процесса создания дисциплины. Для реализации процесса создания дисциплины используется класс Дисциплина. Класс событий, который описывает события, происходящие в предметной области и в КИС имеет атрибут, описывающий тип изменения состояния объекта и связь с классами КИС, объекты которых изменяются (создаются/изменяются/удаляются). В частности, один из объектов класса событий связан с классом дисциплин и имеет атрибут типа события = создание объекта. В правилах генерации событий (т.е. связи между событием и условиями) определены условия генерации этого события: событие генерируется при создании любой дисциплины, без ограничений на атрибуты.

Действия, которые связаны с событием, определяются возможностью извещения по электронной почте, sms, вызов метода серверной компоненты. Для рассматриваемого примера используем

способ отправки сообщения по почте со стандартным текстом, где некоторые поля являются динамическими, т.е. определяются атрибутами класса виновника события (класса Дисциплины в данном случае).

На основе события, генерируемого на создание объекта класса Дисциплина, и отношения проекции между классами Дисциплина и Источник данных создается триггер, связанный с таблицей источника данных. Итак, рассмотрим алгоритм генерации и обработки события создание дисциплины.

1. Выполнилось создание объекта класса Дисциплины.
2. Триггер, который сгенерирован как реакция на событие предметной области, генерирует в классе списка событий – объект, который содержит связи между типом событий, и имеет атрибутами идентификатор объекта, создание которого привело к возникновению события, даты и времени возникновения события.
3. Со списком событий связан отношением проекции триггер, который обеспечивает вызов метода серверной компоненты, обеспечивающего выполнения реакции на событие.
4. В методе специализированной серверной компоненты определяются необходимые действия, как ответ на событие и в случае отправки извещений (почтой или sms) выбираются получатели. Правило выбора получателя мы рассмотрим ниже.
5. Определив получателя и сообщение, метод серверной компоненты посылает сообщение получателям и меняет статус сообщения на обработанный с указанием даты и времени.

Определение получателя такого сообщения может быть нетривиальной задачей. При описании бизнес-процесса устанавливается связь между бизнес-процессом и событием. При этом есть возможность при установке связи определить одно и то же событие как результирующее для одного процесса, так и входное для другого.

Между элементарными бизнес-процессами и пользователями так же определена связь в системе управления правами. Но эта связь может быть определена в режиме связи с ролью и областью видимости.

Здесь возникают две основные проблемы:

1. как определить, какой элементарный бизнес-процесс следует за процессом создания дисциплины, так как перед следующими процессами может располагаться блок условий, определяющий дальнейший маршрут в составном процессе;
2. как определить, кому должно быть послано сообщение, так как роль, связывающая элементарный бизнес-процесс и



пользователей, дана различным пользователям, в том числе тем, кто не имеет отношения к конкретной вновь созданной дисциплине.

Для того, чтобы определить следующий элементарный бизнес-процесс после создания дисциплины, необходимо проанализировать составной бизнес-процесс, в который входит элементарный. Проверка одного или более условий позволяет определить, какой именно элементарный бизнес-процесс будет следующим в маршруте обработки созданной дисциплины.

В системе управления правами пользователей определена связь между элементарными бизнес-процессом и ролью пользователя. Если роль пользователя не имеет областей видимости, то всем пользователям с этой ролью будет отправлено уведомление о создании новой дисциплины. Если у роли есть область видимости, то необходимо выделить тех пользователей, которые должны получить уведомление.

Область видимости представляет собой какой-то класс предметной области, описанный в ОРМД. Необходимо найти связь между классом, который породил событие и классом – областью видимости. Такая связь может существовать или напрямую, или через отношения наследования между классами.

Например, согласовывать дисциплины от кафедры имеет право заведующий кафедрой, т.е. класс кафедр будет областью видимости для роли. Дисциплины связаны отношением *читается на* с классом подразделений, который является базовым по отношению к классу кафедры. На основании связей между дисциплинами и подразделениями можно выделить те кафедры, которые читают данную дисциплину. И в списке областей видимости выделить нужную кафедру, а затем и пользователей, которые должны утверждать дисциплину от кафедры. Именно этим пользователям и будет отослано сообщение.

#### **4.2 Реализация элементарного бизнес-процесса**

Можно выделить два основных подхода в автоматизации бизнес-процесса в рамках рассматриваемого в работе решения. Во-первых, мы рассматриваем составной бизнес-процесс как маршрут работы, единицей которой выступает элементарный бизнес-процесс. В этом подходе мы считаем элементарный бизнес-процесс неделимой частью, фактически черным ящиком, реализованным различными средствами – методами серверных компонентов, веб-приложениями, процедурами и функциями баз данных.

Второй подход предполагает реализацию элементарного бизнес-процесса средствами специализированного сервера приложений КИС. Реализация бизнес-процесса в этом случае описана в ОРМД.

Этот подход предполагает, что элементарный бизнес-процесс включает изменение состояния некоторого объекта на основании регулярного выражения. Операторами регулярного выражения могут выступать константы и атрибуты классов, описанных в ОРМД.

Использование того или иного регулярного выражения при реализации элементарного бизнес-процесса определяется условием, накладываемым на атрибуты. Применение регулярного выражения порождает результат, форма которого записана в ОРМД.

Элементарный бизнес-процесс, который реализуется с помощью регулярных выражений, может определять операции

1. создания объекта некоторого класса, описанного в ОРМД;
2. удаление объекта некоторого класса, описанного в ОРМД;
3. изменение атрибутов объекта некоторого класса, описанного в ОРМД;

Бизнес-процессы характеризуются входными и выходными данными, а так же временем выполнения (ограничения на длительность, время начала, время завершения и т.п.). Во многих случаях выход бизнес-процесса определяется внутри бизнес-процесса и сохраняется в базе данных или отображается на экране пользователя без необходимости описания его на уровне ОРМД. Но так же часто требуется последовательное выполнение бизнес-процессов, которые передают результат друг другу, так что входами в один бизнес-процесс, являются выходы другого бизнес-процесса.

Для реализации передачи – приема параметров между элементарными бизнес-процессами в ОРМД хранятся описания классов, которые связаны с выходными – входными параметрами процессов. В ОРМД хранятся описания входных/выходных параметров методов серверных компонентов. Эти параметры имеют связь с классами, описывающими входы и выходы процессов. Если один и тот же класс является одновременно выходом одного элементарного процесса и входом другого и эти процессы следуют друг за другом в некотором маршруте, то система управления бизнес-процессами после вызова метода для реализации одного бизнес-процесса, вызывает метод, реализующий следующий бизнес-процесс. При этом выходные параметра через память системы управления бизнес-процессами передаются на вход следующего бизнес-процесса.

Еще одной проблемой выполнения элементарных бизнес-процессов является необходимость циклической обработки. При описании элементарного бизнес-процесса может быть указана

необходимость обработки в цикле некоторой последовательности действий. Цикл может быть организован либо по экземплярам класса, который является входным параметром для бизнес-процесса, либо, в случае, если входной класс определяет отношения между другими классами по любому из объектов связанных классов.

Для сохранения результатов могут использоваться стандартные для ORMД метод сохранения нового экземпляра класса.

Конечно, создание такого элементарного бизнес-процесса требует от бизнес-аналитика знания понятия цикл, присваивание, регулярное выражение. Но эти знания не выходят за рамки школьной программы по информатике и не требуют навыков программирования.

## **5 Реализация функционально-онтологического подхода**

Во Владивостокском государственном университете экономики и сервиса (ВГУЭС) в течение последних 7 лет ведутся работы по созданию и развитию КИС. Первоначально это были отдельные несвязанные друг с другом проекты. Затем в качестве основной концепции развития КИС выбрана концепция интеграции, которая позволила объединить проекты на основании данных, интегрируя не только различные СУБД, но и объединяя системы управления сетевой инфраструктурой и прикладным программным обеспечением в единое целое. Одним из результатов интеграции является создание корпоративного портала ВГУЭС, системы единой регистрации управления правами пользователей [6, 7].

Дальнейшее развитие КИС привело к необходимости исследования вопросов интеграции приложений, которое осуществлено с использованием прикладных и управляющих серверных компонентов [6]. Этот подход позволил объединить различные технологии, ранее существующие независимо в КИС. Разработка систем выполнялась значительно быстрее, в силу того, что многие функции уже были решены в системах управления правами, в системе управления справочниками и в системе формирования отчетов.

В последние несколько лет развитие проектов в КИС ВГУЭС идет по направлению автоматизации бизнес-процессов. Системы управления учебным процессом (планирования учебного процесса, расчет нагрузки, автоматизация организации сессии), система расчета стипендии вуза построены на базе функционально-онтологического подхода с описанием бизнес-процессов, маршрутов, классов предметной области, условий, взаимоотношений между классами. На основании этого же подхода разработано хранилище полнотекстовых материалов.

В настоящее время на базе функционально-онтологического подхода ведется разработка систем документооборота, рассылки уведомления родителям студентов, обеспечения качества данных, обеспечения безопасности КИС посредством автоматического управления серверами КИС и телекоммуникационным оборудованием.

Например, в ОРМД хранятся описания классов – компьютеры, серверы, коммуникационное оборудование, виртуальные подсети, а так же описаны связи между этими классами. Каждый компьютер имеет отношение «соединен с» с коммуникационным устройством и «входит в» с виртуальными сетями. Отношение «соединен с» является транзитивным и симметричным. Поэтому для составления общей картины взаимодействия компьютеров нет необходимости в полном описании всех связей.

Класс виртуальных подсетей имеет отношение «доступ разрешен» с самим собой. Это позволяет для каждой подсети определить те подсети, доступ к которым разрешен из подсети. Отношения «доступ разрешен» для виртуальных сетей являются симметричными, но не транзитивными.

На основании этих данных (описания коммуникационного оборудования и политик доступа виртуальных подсетей) программы управления корпоративной сетью могут настраивать коммуникационное оборудование в соответствии с установленными правилами.

Для предотвращения несанкционированного вторжения в корпоративную сеть снаружи используется межсетевой экран, который может управляться так же средствами автоматической установки прав на основании описания в ОРМД.

Опыт разработки и внедрения сложной КИС показал эффективность применения функционально-онтологического подхода

## **6 Литература**

- [1] Буч Г., Рамбо Д., Джекобсон А. UML. Руководство пользователя. – М: ДМК, 2000. – 432с.
- [2] Вендеров А.М. Проектирование программного обеспечения экономических информационных систем: Учебник.- М.: Финансы и статистика.–2000.- 254 с.
- [3] Stojanovic L., Schneider J., Maedche A., Libischer S., Studer R., Lumpp Th., Abecker A., Breiter G., Dinger J.. The role of ontologies in autonomic computing systems//IBM Research Journal. - 2004.-vol.43.-№4.
- [4] Клещев А.С., Шалфеева Е.А. Классификация свойств онтологий. Онтологии и их классификации. Препринт, Владивосток, ИАПУ ДВО РАН, 2005, 19с.
- [5] OWL Web Ontology Language. Overview// <http://www.w3.org/TR/owl-features/>
- [6] Крюков В.В., Шахгельдян К.И. Корпоративная информационная среда вуза: методология, модели, решения. Монография//Дальнаука.- 2007. - 308 с.
- [7] Шахгельдян К.И., Крюков В.В., Гмарь Д.В. Система автоматического управления правами доступа к информационным ресурсам вуза//Информационные технологии. - 2006.- №2.- с.19-29.