

Совместное использование методов бизнес-моделирования и объектно-ориентированной методологии при проектировании профилированного пользовательского интерфейса

Игнатова И.Г., Соколова Н.Ю.

Московский государственный институт электронной техники (технический университет)
124498, Москва, Зеленоград, проезд 4806, д.5, МИЭТ, МОЦНИТ

получена 10 03 2006

Аннотация

В статье рассматривается механизм совместного использования методов бизнес-моделирования и объектно-ориентированной методологии при конструировании профилированного пользовательского интерфейса. Обоснована возможность такого использования. Даны краткие характеристики основных компонентов DFD и UML диаграмм, а также их соответствие при переходе от одной диаграммы к другой. Приведен пример использования.

1. Введение

Конструирование пользовательского интерфейса, полностью удовлетворяющего функциональным требованиям, является важной задачей при разработке программных систем различного назначения. Особенно актуальна данная задача при разработке систем, в которых поддерживается работа различных категорий пользователей, чьи функциональные потребности могут быть как схожи, так и различны. Процессу разработки интерфейса, ориентированного на пользователя, посвящена книга [1], в которой подробно описывается процесс конструирования пользовательского интерфейса с использованием диаграммы прецедентов языка моделирования UML (Unified Modeling Language) объектно-ориентированной (ОО) методологии.

2. Объектно-ориентированная методология

Общепризнано, что ОО методология [1, 2, 3, 4, 5, 6] является наиболее перспективной при разработке программного обеспечения. Это обусловлено наличием свойства наследования классов, что упрощает внесение изменений путем добавления новых классов в библиотеки классов, разработанных в других проектах. Наиболее часто при проектировании систем используется язык UML, который позволяет в едином инструментарии представить и статическую составляющую модели программной системы, и динамическую составляющую этой модели, которая описывает поведение системы и данных [3, 4, 5]. Достоинством языка является то, что он позволяет описывать структуру базы данных (БД), в том числе и реляционной [4].

До применения ОО методологии в разработке программного обеспечения необходимо решить следующие задачи подготовительного этапа:

- 1) определить требования, предъявляемые к системе;
- 2) выделить пользователей системы, их роли, которые могут быть реализованы, решаемые ими задачи, а также процессы предметной области (ПрО), описав их в прецедентах;
- 3) идентифицировать понятия, атрибуты, ассоциации для построения диаграммы классов.

Как правило, эти задачи решаются путем взаимодействия с заказчиками, знающими о бизнес-процессах предприятия, протекающих потоках информации и документов. Таким образом, встает задача выбора способа представления бизнес-процессов и потоков информации в форме, которая, с одной стороны, понятна как специалистам ПрО (заказчикам), так и разработчикам ПО, а с другой стороны, может быть определенным способом трансформирована в диаграммы языка UML, соответствующие подготовительному этапу создания ОО модели.

Как известно из технологии разработки программного обеспечения (ПО) [7], первым этапом создания программных систем является анализ требований к системе, который позволяет выявить роли пользователей в системе, решаемые ими задачи и описать интерфейс пользователя набором функций и их содержанием. В понятиях ОО анализа и проектирования это означает идентификацию процессов, протекающих в предметной области, и описание их с помощью прецедентов (вариантов использования). Хотя прецеденты, являющиеся частью языка UML, и не относятся к ОО модели, но их выделение важно для построения

ОО модели с точки зрения определения требований, выдвигаемых к системе. Описание прецедента обязательно включает определение его инициатора, т.е. пользователя системы, с которым данный прецедент связан.

Следующим этапом создания ОО модели является декомпозиция ПрО на понятия, атрибуты, ассоциации в терминах данной ПрО. Необходимо отобразить практически для каждого прецедента - процесса, какие объекты (понятия) ПрО взаимодействуют между собой с помощью сообщений. Данное взаимодействие отображается с помощью диаграмм взаимодействий языка UML. Итогом ОО анализа является построение диаграммы классов, для чего выполняется анализ диаграмм взаимодействий с точки зрения идентификации классов объектов, их связей и определения методов.

3. Методология бизнес-моделирования

Другая методология, применяемая при анализе деятельности предприятий или организаций в современных методологиях бизнес-моделирования, каким является семейство стандартов IDEF, позволяет определить роли пользователей для реализации их в автоматизированной системе и выделить решаемые задачи, что является важным при разработке ОО модели программной системы. Поэтому для формирования требований к системе, определения бизнес-процессов, понятий предметной области, ролей пользователей в системе и решаемых ими задач предлагается использовать методологию бизнес-моделирования с использованием диаграмм DFD(Data Flow Diagrams). DFD диаграммы входят в состав стандарта IDEF0 и описывают процесс декомпозиции бизнес-процессов до уровня, когда выделенному процессу можно будет поставить в соответствие подпрограмму разрабатываемого ПО [2, 8, 9]. Диаграммы DFD наглядны и понятны специалистам ПрО. Особенностью DFD диаграмм является возможность представления процессов реальной ПрО, которые будут или должны будут происходить при использовании разрабатываемой информационной системы. В отличие от диаграмм UML, диаграммы DFD позволяют отобразить движение потоков информации и документов от одного процесса к другому.

4. Переход от DFD к UML

Основу DFD диаграмм составляют: процессы, внешние сущности, хранилища данных и связи [8, 9].

В состав диаграммы прецедентов языка UML [2-6] входят: варианты использования (прецеденты), спецификации прецедентов, актеры, связи.

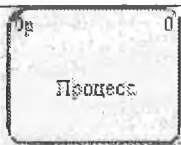

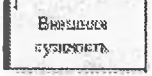

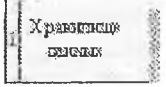


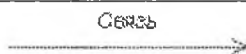
DFD	UML
	
	
	
	

Рис. 1. Соответствие элементов DFD UML

При переходе от DFD диаграмм к диаграммам UML в качестве актеров ОО модели могут быть задействованы внешние сущности, а в качестве претендентов на класс - хранилища данных. В результате анализа и проектирования с помощью DFD диаграмм получают иерархически связанные процессы и подпроцессы, в состав которых входит набор функций. Иными словами, в результате построения DFD

диаграмм все функции, подлежащие автоматизации в программном комплексе, уже систематизированы в логически связанные группы - подпроцессы, а следовательно, можно в качестве названия прецедента брать название подпроцесса, с учетом того, что данный подпроцесс взаимодействует с внешней сущностью системы. На рис. 1 представлено соответствие компонент DFD диаграмм компонентам диаграмм UML.

В качестве примера рассмотрим этап анализа при разработке программной системы поддержки учебного процесса, осуществляемого по дистанционной технологии. На данном этапе были построены DFD диаграммы, описывающие функционально-структурную модель предметной области. Пример DFD диаграммы первого уровня изображен на рис. 2. Полученные с помощью DFD диаграмм подпроцессы имеют иерархическую структуру, фрагмент которой представлен на рис. 3.

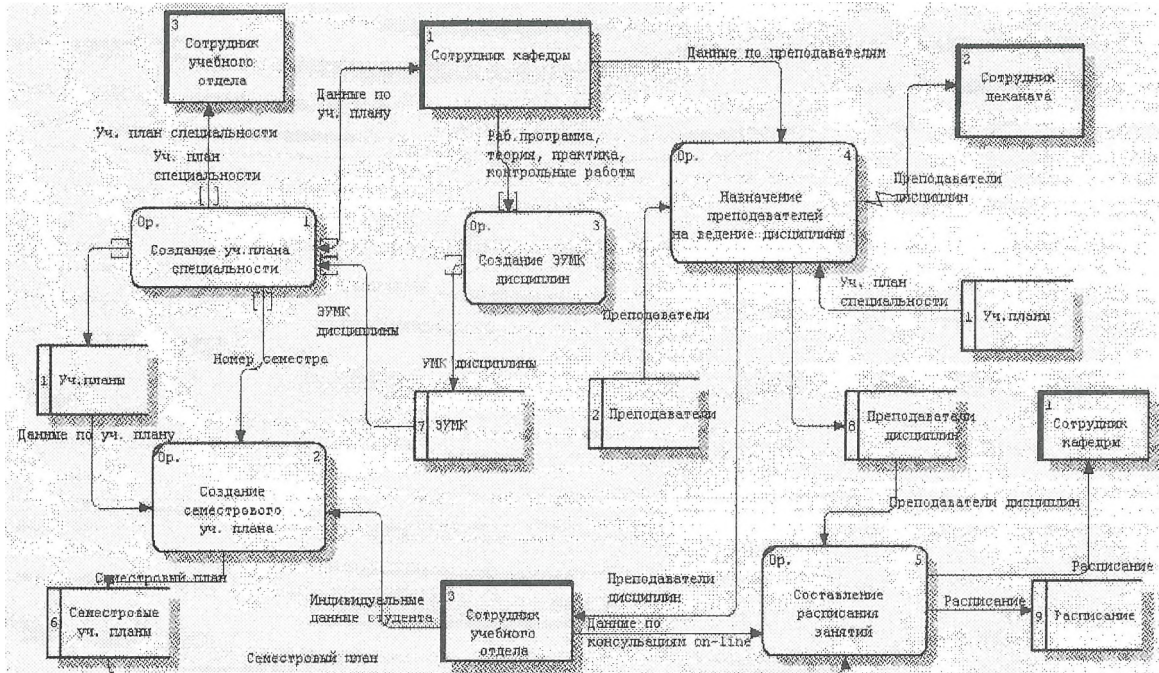


Рис. 2. DFD диаграмма процесса "Обучение с применением дистанционных технологий"

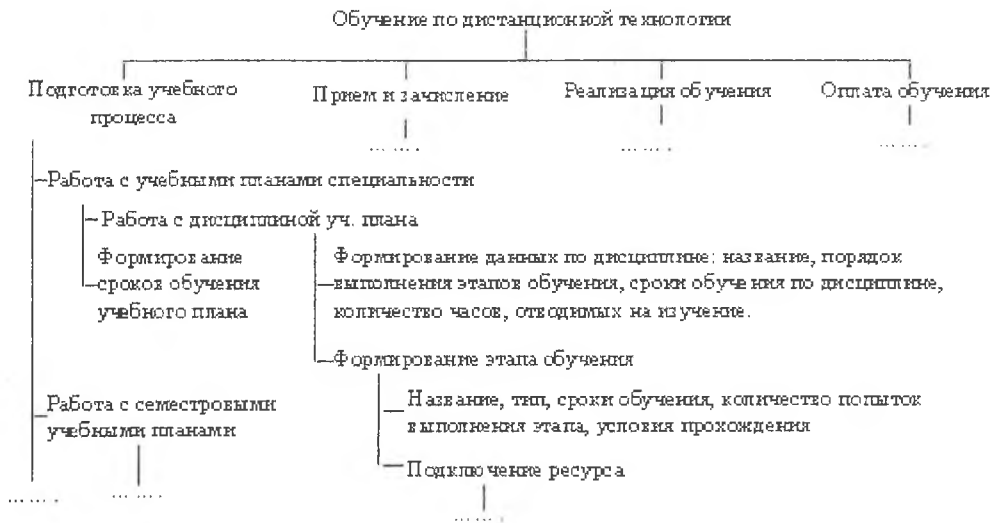


Рис. 3. DFD диаграмма подпроцесса "Обучение с применением дистанционных технологий"

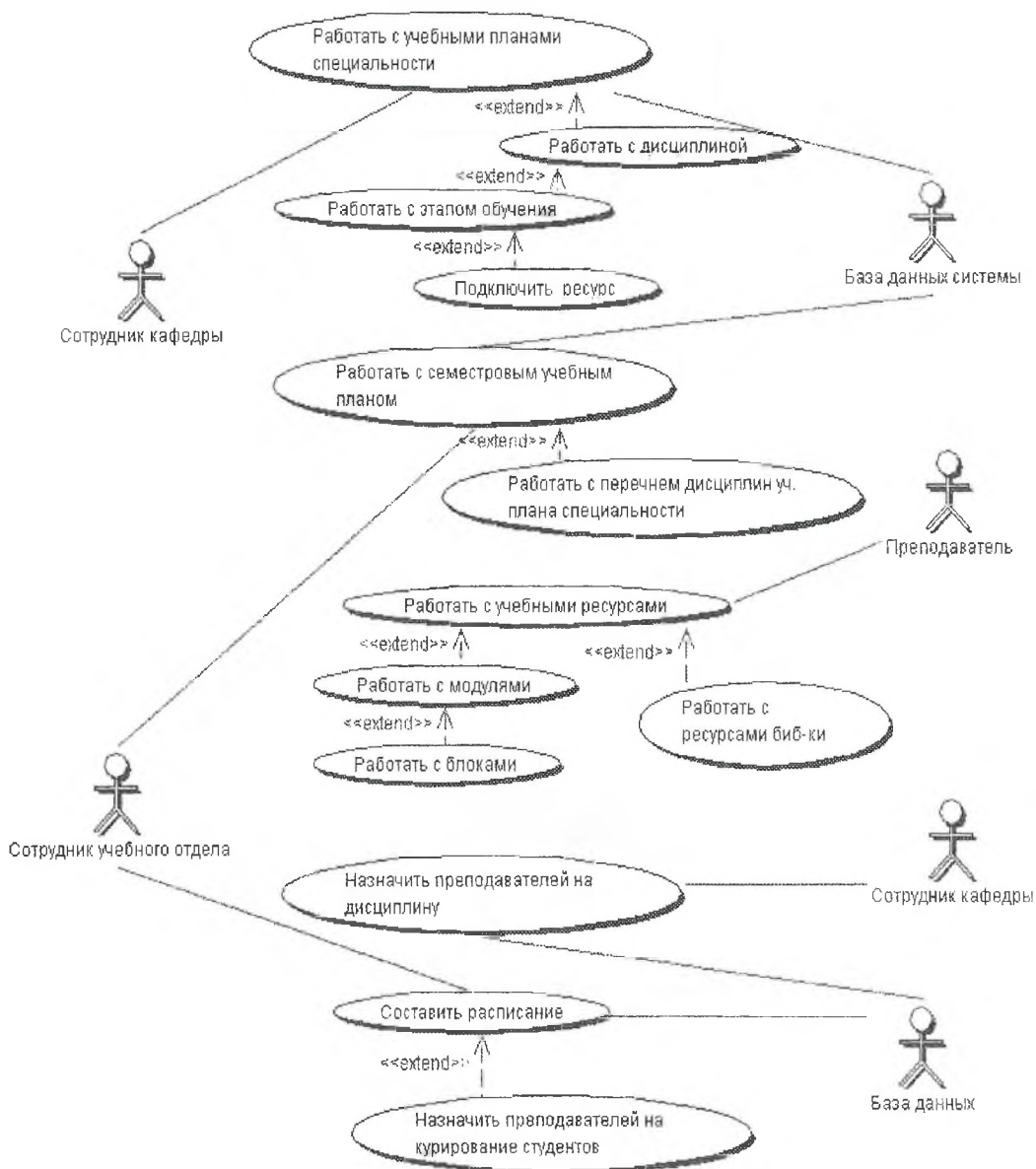


Рис. 4. Диаграмма прецедентов пакета "Подготовка учебного процесса"

DFD диаграммы позволили идентифицировать актеров диаграммы прецедентов, а именно: сотрудник деканата, сотрудник учебного отдела, сотрудник кафедры, преподаватель, студент, сотрудник финансового отдела. А также определить претендентов на классы, например, преподаватель, студент, учебный план, дисциплина и т.д. Анализ диаграмм с точки зрения выявления прецедентов позволил идентифицировать подпроцессы, которые могут быть выбраны в качестве прецедентов. В данном случае были выбраны подпроцессы диаграмм второго уровня. Кроме этого, была использована возможность языка UML организации прецедентов по пакетам. Было создано четыре пакета, каждый из которых соответствует подпроцессу DFD диаграммы первого уровня: подготовка учебного процесса, прием и зачисление, реализация обучения и оплата обучения. В данные пакеты включены прецеденты, соответствующие подпроцессам диаграмм второго уровня.

Анализ диаграммы прецедентов пакета "Подготовка учебного процесса", представленной на рис. 4, позволяет определить основные составляющие пользовательского интерфейса для каждой категории пользователей системы. Например, интерфейс сотрудника кафедры обязательно должен включать операции по

поддержке работы с электронными учебными планами специальности, дисциплинами, этапами обучения по дисциплине и подключению обучающего или контролирующего ресурса.

5. Заключение

Таким образом, при конструировании профилированного пользовательского интерфейса, являющегося одной из составляющих всего процесса анализа и проектирования сложного программного обеспечения, совместно могут быть использованы два разных подхода - бизнес-моделирования и объектно-ориентированного анализа и проектирования. Методы бизнес-моделирования позволяют сформировать требования к системе, определить бизнес-процессы, понятия предметной области, роли пользователей в системе, решаемые ими задачи, которые в дальнейшем, путем соотнесения элементов DFD и UML диаграмм (рисунок 1), могут быть использованы при разработке программной системы методами объектно-ориентированного программирования.

Список литературы

1. *Лари Константайн*. Разработка программного обеспечения. СПб.: Питер, 2004.
2. *Вендеров А.М.* Проектирование программного обеспечения экономических информационных систем. М.: Финансы и статистика, 2000.
3. *Буч Г., Рамбо Д., Джекобсон А.* Язык UML. Руководство пользователя. М.: ДМК, 2000.
4. *Леоненков А.В.* Самоучитель UML. СПб.: БХВ-Петербург, 2001.
5. *Фаулер М., Скотт К.* UML в кратком изложении. Применение стандартного языка объектного моделирования. М.: Мир, 1999.
6. *Ларман К.* Применение UML и шаблоны проектирования. М.: Издательский дом "Вильямс", 2001.
7. *Эрик Дж.Брауде.* Технология разработки программного обеспечения. СПб. Питер, 2004.
8. *Маклаков С.В.* Моделирование бизнес-процессов с BPwin 4.0. М.: ДИАЛОГМИФИ, 2002.
9. *Черемных С.В. и др.* Структурный анализ систем: IDEF - технологии. М.: Финансы и статистика, 2001.