

ОБЪЕКТНО-ОРИЕНТИРОВАННЫЙ АНАЛИЗ, ПРОЕКТИРОВАНИЕ И ПРОГРАММИРОВАНИЕ ИНФОРМАЦИОННОЙ СИСТЕМЫ УНИВЕРСИТЕТА

Попов А. В., Григорьева А. Л., Лошманов А. Ю.

ФГБОУ ВПО «Комсомольский-на-Амуре государственный технический университет», Комсомольск-на-Амуре, Россия (681013, г. Комсомольск-на-Амуре, пр. Ленина, 27), e-mail: loshmanov@kmscom.ru

Существует два основных подхода к разработке информационных систем (ИС), отличающихся критериями декомпозиции: функционально-модульный (или структурный) и объектно-ориентированный. Объектно-ориентированная технология проектирования ИС предоставляет мощную, гибкую, универсальную концептуальную основу для конструирования информационно-управляющих систем в различных областях хозяйственной деятельности и управления, сочетающую использование моделей современной логистики, объектного подхода к компонентам предметной области, современных инструментальных средств визуального программирования и СУБД с SQL-интерфейсом. В работе показано, что объектно-ориентированный подход может быть применен на всех стадиях создания информационной системы университета: анализ, проектирование и программирование. Это связано с тем, что объектная технология приобретает все более широкое распространение при разработке программного обеспечения. А значит, использование объектного подхода и на начальных стадиях создания ИС представляется авторам данной статьи более целесообразным.

Ключевые слова: информационная система, университет, объектно-ориентированный анализ и проектирование.

AN OBJECT-ORIENTED ANALYSIS, DESIGN AND PROGRAMMING OF INFORMATIONAL UNIVERSITY SYSTEM

Popov A. V., Grigoryeva A. L., Loshmanov A. Y.

Federal State-financed Educational Institution of Higher Professional Learning «Komsomolsk-na-Amure State Technical University», Komsomolsk-na-Amure, Russia (681013, Komsomolsk-na-Amure, Lenina, 27), e-mail: loshmanov@kmscom.ru

There are two main approaches to the development of information systems (IS). They differ in criteria of decomposition: functional-modular or structural and object-oriented. Object-oriented technology designing of IS is powerful, flexible, universal conceptual framework for designing of information and control systems in various fields of economic and management activities, which combines use of modern logistics models, object-oriented approach to components of subject area, modern tools of visual programming and DBMS with SQL interface. In this work have been shown that object-oriented approach can be applied at all stages of IS's creation: analysis, design and programming. This is because of object technology has become more and more widespread in software development. And that means using the object approach and in the initial stages of the IS's development seems to be more reasonable for authors of this article.

Key words: information system, university, object-oriented analysis and design.

Введение

Одним из эффективных направлений развития производства и совершенствования управления является разработка и внедрение на предприятии передовых информационных технологий, включающих в себя:

- определение функций, которые должны быть решены с целью обеспечения служб предприятия надежной и качественной информацией для принятия решений;
- определение задач, которые необходимо решить с целью обеспечения решения функций, определенных на первом этапе;
- определение перечня количественных и качественных показателей информации, необходимых для решения задач, определенных на втором этапе;

- определение форм и методов, основываясь на которых и/или используя которые с помощью количественных и качественных показателей достигается решение требуемых задач и определение заданных функций для принятия необходимых решений [1].

Несмотря на продолжительную историю создания в вузах информационных систем (ИС) поддержки управления и примеры успешного решения этой задачи в отдельных вузах, она по-прежнему является одной из важнейших задач вузовской информатизации. Необходимость ее решения становится еще более актуальной в связи с внедрением в университетах стратегического планирования, создания системы управления качеством [2, 6].

Программные проекты зачастую достаточно сложны, и их декомпозиция (по принципу «разделяй и властвуй») – это основная стратегия борьбы со сложностью. Она состоит в разбиении проблемы на мелкие управляемые элементы. До появления объектно-ориентированного подхода к анализу и проектированию наиболее популярной методологией декомпозиции являлись *структурный анализ и проектирование*. Этот подход заключается в декомпозиции задачи на функции или процессы, приводящей к созданию иерархии процессов и подпроцессов.

Основная идея объектно-ориентированного анализа и проектирования информационных систем состоит в рассмотрении предметной области и логического решения задачи с точки зрения объектов (понятий или сущностей).

В процессе объектно-ориентированного анализа основное внимание уделяется определению и описанию объектов (или понятий) в терминах предметной области. Например, в случае информационной системы ВУЗа среди понятий должны присутствовать *Преподаватель (Lecturer)*, *Студент (Student)* и *Заведующий кафедрой (Head Of Chair)*.

В процессе объектно-ориентированного проектирования определяются логические программные объекты, которые будут реализованы средствами объектно-ориентированного языка программирования. Эти программные объекты включают в себя атрибуты и методы. Например, в информационной системе ВУЗа программный объект Lecturer может содержать атрибут *Имя (name)* и метод *Сформировать отчет по НИР (Generate Report)*.

И наконец, в процессе конструирования или объектно-ориентированного программирования обеспечивается реализация разработанных компонентов, таких как класс Lecturer на языке C++, C#, Java, Smalltalk или Visual Basic [3].

Успешность проектов информационных систем в различных отраслях и эффективность их внедрения обусловлены, прежде всего, достаточно строгим определением объектов управления (точнее, компонентов сложного процесса управления) и оптимизируемых бизнес-процессов. Не случайно создание информационной системы

производится в контексте общей оптимизации бизнеса и зачастую предполагает реинжиниринг бизнес-процессов предприятия [4].

Определение экономических процессов

В первую очередь, необходимо выделить экономические процессы, происходящие на предприятии: выплата заработной платы, разработка программного обеспечения, ведение занятий, публикация научных статей, участие в научных грантах и т. д.

В терминах объектно-ориентированного анализа и проектирования этот этап называется *анализом требований*. Он подразумевает выделение экономических процессов и требований и их формулировку в виде прецедентов. Прецедент – это текстовое описание процессов, происходящих в системе или на предприятии, следующего вида.

Прецедент	<i>Сформировать отчет по выполняемой НИР (Generate Report)</i>
Описание	Этот прецедент начинается с распоряжения заведующего кафедрой преподавателю собрать данные о выполненной им НИР за определенный период и представить в определенном виде

Идентификация процессов и их оформление в виде прецедентов на самом деле не являются видом деятельности с точки зрения объектно-ориентированного анализа. Они не имеют никакого отношения к объектам. Однако это достаточно важный и широко используемый начальный этап объектно-ориентированного анализа и проектирования. Кроме того, прецеденты являются частью языка UML.

Распределение видов деятельности в организации

Следующий шаг состоит в определении видов деятельности участников процесса: преподаватель, студент, декан, научный сотрудник, лаборант и т.д. В терминах объектно-ориентированного анализа и проектирования этот этап называется *объектно-ориентированным анализом предметной области* и состоит в составлении концептуальной модели. Концептуальная модель отражает различные категории элементов предметной области: не только виды деятельности участников, но и все относящиеся к делу понятия.

Взаимодействие сотрудников

После идентификации процессов и видов деятельности сотрудников предприятия (в нашем случае – университета), необходимо определить, как выполняются процессы. Этот вид деятельности называется ориентированным на решение или разработку. Здесь определяются обязанности сотрудников по решению задач, необходимых для выполнения процесса, а также способы их взаимодействия.

Эта деятельность аналогична *объектно-ориентированному проектированию*, при котором основное внимание сосредоточено на распределении обязанностей. *Распределение*

обязанностей означает выделение задач и обязанностей различных программных объектов в приложении подобно распределению функциональных обязанностей между сотрудниками. Программные объекты, как и люди, в процессе выполнения своих обязанностей обычно взаимодействуют между собой.

Обязанности объектов и их взаимодействие, как правило, отображаются с использованием *диаграмм классов* и *диаграмм взаимодействий*. На диаграммах отображаются классы и потоки сообщений между программными объектами [3].

Проектирование и программирование ИС

Рассмотрим кратко основные аспекты и сложившиеся подходы к реализации ИС. ИС принято разделять по масштабу выполняемых функций на одиночные, групповые и корпоративные.

Одиночные ИС реализуются на автономном компьютере (чаще всего ПК), могут содержать несколько простых приложений, рассчитаны на работу одного пользователя или группы пользователей, разделяющих по времени одно рабочее место. Подобные приложения создаются с помощью так называемых «настольных» СУБД или с помощью файловой системы и диалоговой оболочки для ввода, редактирования и обработки данных.

Групповые ИС ориентированы на коллективное использование информации членами обособленной рабочей группы, обычно строятся как локальная вычислительная сеть ПК или реже как многотерминальная вычислительная система. Однотипные или специализированные рабочие места обеспечивают вызов одного или нескольких приложений. Общий информационный ресурс представляет собой базу данных или совокупность файловых структур. При разработке таких систем используются «настольные» СУБД, серверы БД для рабочих групп и соответствующие инструменты разработки.

Корпоративные ИС ориентированы на использование в масштабе предприятия (организации) для различных рабочих групп, могут поддерживать территориально разнесенные узлы или сети. Отличительная особенность таких систем – обеспечение доступа из подразделений к центральной или распределенной БД предприятия (организации), а также к информационным ресурсам рабочей группы. Такие системы реализуются на основе архитектуры «клиент – сервер» со специализацией серверов. При этом используются корпоративные SQL-серверы и соответствующие инструментальные средства [5].

Основой разработки файл-серверных приложений для локальных сетей ПК являются инструментальные средства «персональных» СУБД, реализованные в виде диалоговой интегрирующей среды, предоставляющей три уровня доступа:

- программирование и создание приложений;

- создание и ведение структуры БД, а также интерактивная генерация макетного приложения и его компонентов (меню, формы или окна, отчеты, запросы и программные модули);

- использование диалоговой среды и генераторов конечным пользователем для создания, ведения и просмотра БД, а также формирования простых отчетов и запросов.

Среди инструментальных средств реализации приложений с архитектурой «клиент – сервер» выделяют следующие:

- среды разработки приложений для серверов баз данных;
- независимые от СУБД инструменты для создания приложений «клиент – сервер»;
- средства поддержки распределенных информационных приложений.

Среди этой группы следует выделить инструментальные средства быстрой разработки приложений RAD (RapidApplicationDevelopment), обеспечивающие реализацию удаленного доступа к СУБД по двухзвенной схеме «клиент – сервер»; связь клиентских приложений с серверами БД с помощью непроцедурного языка структурированных запросов SQL; целостность БД, включая целостность транзакций; поддержку хранимых процедур на серверах БД; реализацию клиентских и серверных триггеров-процедур; генерацию элементов диалогового интерфейса и отчетов.

Средства RAD базируются на объектно-ориентированном подходе, когда информационные объекты формируются как действующие модели и их функционирование согласовывается с пользователем. Разработка приложений на основе RAD ведется с использованием множества готовых объектов, хранимых в виде базы данных. Объектно-ориентированные инструменты RAD в среде GUI позволяют на основе набора стандартных объектов, для которых инкапсулированы атрибуты и внутренние процедуры, формировать простые приложения без написания кода программы. Использование в RAD визуального программирования позволяет еще более упростить и ускорить процесс создания информационных систем. Логика приложения, реализованного с помощью RAD, является событийно-ориентированной, что подразумевает наличие определенного набора событий: открытие и закрытие окон, нажатие клавиши клавиатуры, срабатывание системного таймера, получение и передача управления каждым элементом экрана, некоторые элементы управления базой данных.

Наиболее полным описанием процесса разработки программного обеспечения, включающим методики выполнения работ на каждой стадии жизненного цикла системы, является RationalUnifiedProcess (RUP), уникальность которого заключается в том, что это стандартизованный процесс разработки программного обеспечения, используемый многими

крупными компаниями по всему миру. RUP обладает следующими преимуществами по сравнению с другими процессами:

- обеспечивает четко организованный подход к назначению задач и требований в рамках организации разработки;

- основан на объектно-ориентированных технологиях разработки программного обеспечения и может использоваться для широкого круга проектов и организаций;

- является итеративным процессом, который допускает расширение проблемы и круга задач по мере последовательного усовершенствования модели и программного обеспечения, позволяя увеличить коэффициент эффективности на протяжении нескольких итераций, что дает большую гибкость в приспособлении к новым требованиям и допускает идентификацию и разрешение рисков разработки заранее;

- создает описание программного продукта, позволяющего восстановить процесс его разработки;

- осуществляет полную поддержку различными инструментальными средствами, позволяющими автоматизировать работы на всех стадиях жизненного цикла программы и сохранить артефакты разработки в электронном виде;

- предоставляет возможность гибкой и перенастраиваемой конфигурации, позволяющей использовать его как для малых групп разработчиков, так и для больших организаций.

В качестве графической нотации в RUP используется Unified Modeling Language (UML), являющийся стандартом для представления объектных моделей. В UML артефакты разработки представляются диаграммами, описывающими структуру программы и ее поведение.

Заключение

Объектная технология приобретает все более широкое распространение при разработке программного обеспечения. Этот процесс особенно активизировался с появлением языка Java, в связи с чем возрастает роль объектно-ориентированного анализа и проектирования, так как владение этими вопросами обеспечивает создание робастных и простых в поддержке объектно-ориентированных систем. Вдобавок к этому, унифицированный язык моделирования UML является признанным стандартом для описания моделей, который обеспечивает возможность общения между разработчиками. А идиомы и удачные проектные решения при создании объектно-ориентированных систем были сформулированы в виде шаблонов (шаблоны GRASP), которые эксперты предлагают применять при создании информационных систем [3].

Список литературы

1. Создание информационной системы. – URL: <http://xreferat.ru/37/24-1-sozдание-informacionnoiy-sistemy.html> (дата обращения: 26.11.2012).
2. Толстобров А. П. Развитие информационных систем управления вузом и их интеграция. – URL: <http://do.gendocs.ru/docs/index-53927.html> (дата обращения: 26.11. 2012).
3. Ларман К. Применение UML и шаблонов проектирования: пер. с англ: уч. пос. – М.: Издательский дом «Вильямс», 2001. – 496 с.: ил.
4. Глухих И. Н. Корпоративная информационная система университета на базе интернет / интернет-портала. – URL: <http://ecsocman.hse.ru/data/093/949/1219/68-76.pdf> (дата обращения: 01.12.2012).
5. Гвоздева Т. В., Баллод Б. А. Проектирование информационных систем. – Изд-во «Феникс», 2009. – 512 с.
6. Программа стратегического развития ФГБОУ ВПО «Комсомольский-на-Амуре государственный технический университет» на 2012 – 2016 годы. – URL: <http://www.knastu.ru/images/stories/News/2012/strat/programm-knastu.pdf> (дата обращения: 08.12.2012).

Рецензенты:

Биленко Сергей Владимирович, доктор технических наук, доцент, помощник ректора по информатизации учебного процесса ФГБОУ ВПО «Комсомольский-на-Амуре государственный технический университет», г. Комсомольск-на-Амуре.

Одинокое Валерий Иванович, доктор технических наук, профессор, заведующий кафедрой «Прикладная математика и информатика» ФГБОУ ВПО «Комсомольский-на-Амуре государственный технический университет», г. Комсомольск-на-Амуре.