

НАУЧНЫЕ ИССЛЕДОВАНИЯ

УДК 378.147
ББК 448-95.27

П. И. Алексеевский

Екатеринбург

РОЛЬ ОБЪЕКТНОГО МОДЕЛИРОВАНИЯ В ПРОЦЕССЕ ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ СТУДЕНТОВ ПЕДАГОГИЧЕСКИХ СПЕЦИАЛЬНОСТЕЙ

ГСНТИ 14.35.09
Код ВАК 13.00.02

КЛЮЧЕВЫЕ СЛОВА: обучение программированию; объектное моделирование; объектно-ориентированное программирование; проектирование программного обеспечения.

??АННОТАЦИЯ. Рассматривается курс программирования для студентов педагогических специальностей, который предполагает освоение объектно-ориентированной парадигмы программирования. Эффективность ее изучения повышается путем введения курса объектного моделирования с использованием языка UML в учебную программу на ранних стадиях процесса обучения. Язык UML может использоваться как студентами при выполнении практических заданий по программированию, так и преподавателями для наглядного представления информационных процессов.

P. I. Alekseebsky

Ekaterinburg

ROLE OF OBJECT MODELLING IN THE PROCESS OF TEACHING PROGRAMMING TO THE STUDENTS OF PEDAGOGICAL SPECIALIZATIONS

KEY WORDS: teaching programming; object modeling; object-oriented programming; projecting of software.

ABSTRACT. Programming course for students of pedagogical specialties requires mastering of the object-oriented programming paradigm. The effectiveness of its learning may be improved by the introduction of the object modeling course using UML in the curriculum at early stages of the learning process. The UML language may be used by students in the practical programming course as well as by teachers for the visual representation of information processes.

Курс программирования в высших учебных заведениях включает в себя изучение объектно-ориентированного программирования. Необходимость изучения объектно-ориентированной парадигмы для студентов педагогических специальностей

отражена в соответствующих стандартах (в частности, в стандарте специальности «050202. 65 (030100) Информатика»). В то же время для эффективного изучения программирования, а также для обучения программированию с использованием данной парадигмы требу-

© Алексеевский П. И., 2011

Педагогическое образование в России. 2011. № 3
E-mail: pedobraz@uspu.ru

ется дополнительная подготовка. Преподавание курса программирования начинается на первых стадиях процесса обучения по данной специальности, таким образом основы проектирования иерархии классов вводятся непосредственно в процессе обучения программированию. Данный подход имеет ряд недостатков. Механизмы, реализующие объектно-ориентированную парадигму, требуют дополнительного машинного времени для выполнения определенных операций над объектами, поэтому неоптимальная организация структуры программы может привести к заметному снижению производительности. Кроме того, избыточность этой структуры (лишние уровни абстракции и т. п.) затрудняет отладку программы. Для преподавателей это означает также сложность в проверке результатов выполнения студентами практических заданий.

В целях повышения качества программного кода, реализуемого студентами, требуется повысить их готовность к использованию технологий объектно-ориентированного программирования. Эта дополнительная подготовка предполагает введение в учебную программу курса объектного моделирования до изучения программирования либо одновременно с ним.

Такой подход применяется в рамках реализуемой в Уральском государственном педагогическом университете специализации «Компьютерные игровые технологии в образовании» [2]. Программа специализации предполагает изучение языков программирования С и С++, которому предшествует изучение курса объектного моделирования в рамках дисциплины «Игровые проекты». В качестве инструмента для объектного моделирования используется язык UML (Unified Modeling Language — унифицированный язык моделирования).

Язык UML является графическим языком моделирования, предоставляющим ряд абстракций для всесто-

ронного описания и анализа проекта. Проект при этом представляет собой совокупность моделей (объектная модель, компонентная модель, модель представления данных), каждая из которых реализуется набором диаграмм различного типа (диаграммы классов, объектов, прецедентов, деятельности, конечных автоматов и др.). Помимо графического представления, существует возможность сериализации моделей в различные текстовые форматы хранения данных, такие, как XML, YAML и др., что позволяет использовать их для автоматической генерации шаблонов программного кода, схем баз данных и технической документации. Некоторые средства проектирования, использующие язык UML, имеют также возможность анализа имеющегося программного кода и составления его объектной модели с целью анализа и оптимизации.

Основными факторами, обуславливающими выбор данного языка, являются: объектно-ориентированная направленность, универсальность, сравнительная простота, распространенность, отсутствие жесткой привязки к конкретному языку программирования, а также доступность поддерживающих его средств моделирования. Возможности языка UML позволяют применять его для глубокого анализа поставленной задачи и реализации программного кода на любом современном языке программирования, в том числе и на языках, не поддерживающих объектно-ориентированную парадигму, таких, как язык С (путем введения в программный код дополнительных алгоритмов с целью получения необходимых уровней абстракции и механизмов управления данными). Объектная модель программы, содержащая пояснения и требования к ее компонентам, позволяет также составлять наборы тестовых данных для проверки корректности работы реализованных алгоритмов для каждого из компонентов программы и подготовить техническую

документацию. Использование унифицированного языка моделирования при групповой работе студентов над проектом дает возможность эффективно обмениваться информацией о новых компонентах и их программных интерфейсах между участниками проекта [1. С. 1].

Современные тенденции развития ИТ-индустрии (облачные вычисления, параллельные вычисления, отказоустойчивые информационные системы) позволяют сформулировать ряд требований для будущих специалистов и учителей. Методика разработки систем параллельной обработки данных имеет ряд отличий от методики разработки локальных программ. В частности, требуется использовать какие-либо механизмы обмена данными между вычислительными узлами, в том числе сообщениями синхронизации. Язык UML предоставляет возможность описания в том числе и этих информационных процессов и потоков данных. Курс обучения использованию средств объектного моделирования позволяет студентам не только создавать модели программного обеспечения для упрощения их программной реализации, но и планировать дальнейшую деятельность по внедрению и сопровождению программы. Абстракции, предоставляемые языком UML, могут быть применены и для планирования любых других видов деятельности, помимо программирования.

Для практических занятий по изучению языка UML в рамках дисциплины «Игровые проекты» применяется среда Eclipse с установленными дополнениями для поддержки данного язы-

ка. Данный пакет является свободным программным обеспечением и по основным функциям соответствует коммерческому программному продукту IBM Rational Software Architect (также основанному на платформе Eclipse). Имеющийся функционал достаточен для изучения языка UML и применения его в учебной и преподавательской деятельности, при этом есть возможность в случае необходимости перейти на более мощную профессиональную платформу без полной переподготовки.

Опыт внедрения данного курса позволяет сделать следующие выводы: студенты, освоившие курс, более быстро и качественно выполняют практические задания по программированию с использованием объектно-ориентированной парадигмы — программный код, реализуемый ими, имеет более строгую классовую структуру без избыточных элементов; реализуемые алгоритмы содержат меньше семантических ошибок, поскольку их структура была проработана до реализации в программном коде; поиск допущенных студентами ошибок упрощен, поскольку происходящие в программе информационные процессы определенным образом отражены в заранее составленной модели.

Язык UML может быть использован и преподавателями в качестве инструмента для наглядного описания информационных процессов при объяснении учебного материала, что также может повысить степень усвоения материала студентами.

ЛИТЕРАТУРА

1. АЛЕКСЕЕВСКИЙ П. И. Применение технологий совместной разработки при обучении программированию // Математика. Компьютер. Образование : материалы XVI конференции. Пущино, 2009.
2. ЛАПЕНОК М. В., ГАЗЕЙКИНА А. И. Компьютерные игровые технологии в образовании // Повышение эффективности подготовки учителей физики и информатики: материалы междунар. науч.-практ. конф. / Уральский гос. пед. ун-т. Екатеринбург, 2006.

Статью рекомендует д-р пед. наук, проф. Т. Н. Шамало