



*Иванов Денис Юрьевич,  
Новиков Фёдор Александрович*

## МОДЕЛИРОВАНИЕ НА UML: ОПЫТ ПРЕПОДАВАНИЯ В ИНТЕРНЕТ

### Аннотация

В статье рассматривается опыт подготовки учебных материалов, включая тексты, видео-презентации, интерактивные дистанционные тесты, и опыт преподавания в сети Интернет авторского курса «Моделирование на UML».

**Ключевые слова:** дистанционное обучение, моделирование, UML.

### ВВЕДЕНИЕ

Унифицированный язык моделирования UML появился немногим более 10 лет тому назад и за это время фактически стал общепринятым стандартом описания архитектуры, проектных решений и других артефактов, возникающих при разработке программного обеспечения. В последние годы область применения UML расширилась, язык все чаще применяют не только при разработке приложений, но и как средство анализа и моделирования бизнес-процессов, средство описания и документирования различных стандартов, протоколов, образцов проектирования и так далее. Другими словами, язык UML в настоящее время – это *lingua franca* для квалифицированных инженеров, системных архитекторов и менеджеров среднего звена в области информационных технологий.

В настоящее время различные курсы по языку UML представлены в учебных планах большинства высших и некоторых средних учебных заведений, выпускающих программистов. Таким образом, молодые люди, начинающие свою карьеру сейчас, имеют

представление о языке со студенческой или даже школьной скамьи. Однако для большой массы не начинающих, а опытных инженеров, которым этот язык нужнее всего, ситуация не такая благоприятная. Десять лет назад, когда они учились, в их учебных программах нужного курса по языку, скорее всего, не было. Таким образом, существует значительная аудитория специалистов, которым необходимо освоить UML самостоятельно или в той или иной форме дополнительного образования «выше высшего». Об этом также свидетельствует приведенная ниже на рис. 3 гистограмма, показывающая заинтересованность в изучении UML в зависимости от возраста.

Чуть менее 10 лет тому назад мы начали разработку и преподавание авторского курса по языку UML. Первоначально этот курс назывался «Анализ и проектирование на UML» и реализовывался, в основном, в форме традиционных лекций для студентов или же очных корпоративных тренингов для инженеров программирующих организаций. Многократное преподавание курса в группах высоко квалифицированных и мотивированных слушателей позволяло вносить постоянные улучшения как в содер-

---

© Д.Ю. Иванов, Ф.А Новиков, 2009

жание, так и в форму представления курса. В настоящее время курс называется «Моделирование на UML» и подкреплен большим объемом разнообразных материалов: книга, набор видео-презентаций, слайдов, тестов и другое. Мы практикуем несколько разных форм преподавания: обычные лекции, интерактивные он-лайн тренинги, обсуждения в режиме вопрос-ответ на специальных форумах, публичных чатах, коллективное обсуждение и анализ работ учащихся в сети Интернет и другое.

Предметом статьи является обсуждение и анализ нашего опыта и наблюдений, полученных в процессе подготовки материалов курса и в процессе его преподавания в различных формах. План статьи следующий. В первом разделе мы обсуждаем сам предмет – унифицированный язык моделирования UML – и то, какие дидактические особенности имеет этот предмет. Во втором разделе излагается наша основная методическая идея по структурированию содержания и формы курса. В третьем разделе собраны наши фактические наблюдения по опыту реализации основной идеи. В заключении представлены наши выводы из сделанных наблюдений.

#### **ЧТО ТАКОЕ UML С ПРЕПОДАВАТЕЛЬСКОЙ ТОЧКИ ЗРЕНИЯ**

Обсуждаемый предмет обозначается идентификатором UML, который является аббревиатурой полного названия Unified Modeling Language. Правильный перевод этого названия на русский язык – унифицированный язык моделирования. Таким образом, обсуждаемый предмет характеризуется тремя словами, каждое из которых является точным термином.

Главным словом в этом сочетании является слово «язык». Язык – это знаковая система для хранения и передачи информации. Различаются языки формальные, правила употребления которых строго и явно определены, и неформальные, употребление которых основано на сложившейся практике. Различаются также языки естественные, появляющиеся как бы сами со-

бой в результате неперсонифицированных усилий массы людей, и языки искусственные, являющиеся плодом видимых усилий определенных лиц. Филологи, наверное, смогут назвать еще дюжину различных характеристик: нормативный и ненормативный язык, живой и мертвый, синтетический и аналитический и т. д. Например, язык, на котором написана эта статья является неформальным и естественным. С другой стороны, подавляющее большинство языков программирования являются формальными и искусственными. Встречаются и другие комбинации: например, язык алгебраических формул мы считаем формальным и естественным, а эсперанто – неформальным и искусственным.

Так вот, UML можно охарактеризовать как формальный искусственный язык, хотя и не в такой степени, как многие распространенные языки программирования. Признаком искусственности служит наличие трех общепризнанных авторов – Гради Буча, Ивара Якобсона и Джеймса Рамбо. Для описания формальных искусственных языков (в частности, для описания языков программирования) придумано и используется множество различных способов. Однако на практике сложилась общепринятая структура таких описаний. Считается, что формальный искусственный язык описан должным образом, если это описание содержит, по меньшей мере, следующие части:

- Синтаксис (syntax), то есть определение правил составления конструкций языка.
- Семантика (semantics), то есть определение правил приписывания смысла конструкциям языка.
- Прагматика (pragmatics), то есть определение правил использования конструкций языка для достижения определенных целей.

Как формальный искусственный язык UML имеет синтаксис, семантику и прагматику, хотя эти части названы в некоторых случаях иначе и описаны по другому, нежели это принято в текстовых языках программирования, поскольку, во-первых, UML язык графический, а не текстовый, а во-вторых, UML язык моделирования, а не программирования. Это порождает некото-

рые особенности, которые необходимо учесть при преподавании языка. Использовать формальные описания синтаксиса и семантики из стандарта (так называемые метамодели) на первых этапах обучения бесполезно. Мы многократно убеждались, что обучаемые испытывают серьезные затруднения при чтении метамodelей, и им не удается даже прочитать формальное определение, не говоря уже о том, чтобы понять, что это определение означает. Таким образом, первая проблема преподавания UML, которую необходимо решить – это выбор адекватных средств описания языка.

Второе слово «моделирование», входящее в название UML, имеет множество смысловых оттенков и сложившихся способов употребления. В частности, английские слова *modeling* и *simulation* оба переводятся словом «моделирование», хотя означают разные вещи. В первом случае речь идет о составлении модели, которая используется только для описания моделируемого объекта или явления. Во втором случае подразумевается составление модели, которая может быть использована для получения существенной информации о моделируемом объекте или явлении. При этом во втором случае обычно добавляется уточняющее прилагательное: численное моделирование, математическое моделирование и др. UML является языком моделирования в первом смысле, а обучаемые по умолчанию ожидают второй смысл слова «моделирование», и это обстоятельство необходимо учитывать при изложении материала с самого начала. В отношении разработки программного обеспечения так сложилось, что результаты фаз анализа и проектирования, оформленные средствами определенного языка, принято называть *моделью*.<sup>1</sup> Деятельность по составлению моделей естественно назвать *моделированием*. Именно в этом смысле UML является языком моделирования. К сожалению, это употребление слова расходится с устоявшейся отечественной традицией. В частности, названия профессий: модель, модельер, модельщик и модельист не имеют отношения к предмету. Те,

кто составляет модели UML, по-русски никак не называются, а должность, которую они занимают, носит название «архитектор» или «системный архитектор». Таким образом, вторая проблема преподавания UML, которую необходимо решить – это преодоление имеющегося у обучаемого неправильного понимания предмета и формирование правильного понимания.

Наконец, UML – это унифицированный язык моделирования. Описывая историю создания UML, его авторы характеризуют эпоху до UML как период «войны методов». Пожалуй, «война» – это слишком сильно сказано, но, действительно, UML является отнюдь не первым языком моделирования. К моменту его появления насчитывались десятки других, различающихся системой обозначений, степенью универсальности, способами применения и т. д. Авторы языков и теоретики программирования препирались между собой, выясняя, чей подход лучше, а практические разработчики всю эту «войну методов» равнодушно игнорировали, поскольку ни один из методов не дотягивал до уровня индустриального стандарта.

Толчком к изменению ситуации послужили следующие обстоятельства. Во-первых, массовое распространение получил объектно-ориентированный подход к разработке программных систем, в результате чего возникла потребность в соответствующих средствах. Другими словами, появления чего-то подобного UML с нетерпением ждали практики. Во-вторых, три крупнейших специалиста в этой области, авторы наиболее популярных методов, решились объединить усилия именно с целью унификации своих (и не только своих) разработок в соответствии с социальным заказом.

Приложив заслуживающие уважения усилия, авторы UML при поддержке и содействии всей международной программистской общественности смогли свести воедино (унифицировать) большую часть того, что было известно и до них. В результате унификации получилась теоретически изящная и практически полезная вещь – UML.

<sup>1</sup> Что вполне согласуется с наиболее общим смыслом слова модель – абстрактное описание чего-либо.

Если попытаться проследить историю возникновения и развития элементов UML как на уровне основополагающих идей, так и на уровне технических деталей, то пришлось бы назвать сотни имен и десятки организаций. Мы не будем этого делать, и не только из экономии места, но и потому, что история развития UML отнюдь не завершена – язык постоянно совершенствуется, обогащается и расширяется. Мы полагаем достаточным привести на рис. 1 картинку, иллюстрирующую историю развития UML.

Как видно из рис. 1, на особом положении оказалась версия 1.5. Дело в том, что версия 1.5 была выпущена в тот момент, когда «моделирующая общественность» предвкушала появление обещанной версии 2.0. На самом деле версия 1.5 содержит некоторые элементы версии 2.0, в частности, набор элементарных действий, достаточно широкий, для того чтобы применять UML не только как язык моделирования, но и как язык программирования. Но «генеральная линия» развития инструментальных средств прошла мимо этого явления. Все крупные поставщики инструментов предпочли заявить о поддержке версии 2.0 (иногда даже не имея для этого достаточных оснований) и оставили без поддержки версию 1.5. Заметим, что все это время процедура унификации непрерывно продолжает-

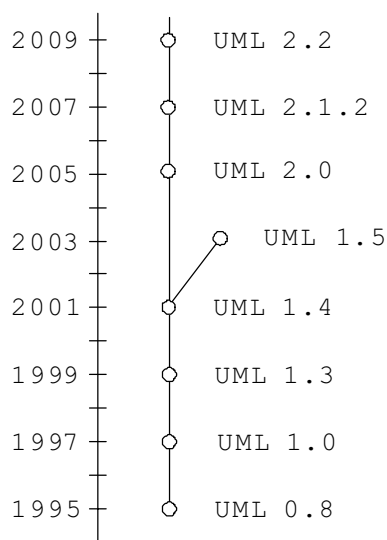


Рис. 1. История развития UML

ся. В каждой новой версии в UML унифицируются дополнительные средства, ранее оставшиеся за бортом языка.

Процедура унификации подразумевает, прежде всего, создание единой терминологии. Проблема терминологии является одной из самых болезненных при изучении языка UML.

Во-первых, авторы UML старались сделать язык независимым от конкретных языков программирования, моделей вычислимости и тому подобного. С этой целью они зачастую вводили новые термины для определяемых понятий, чтобы случайное совпадение со старым термином, уже занятым в какой-либо смежной области программирования, не вводило в заблуждение пользователя. Кроме того, в UML унифицированы многие различные подходы и методы, каждый со своей терминологической традицией, и их все необходимо было учесть и иногда пойти навстречу. Как всякий компромиссный вариант, терминология UML получилась довольно замысловатой и не всегда последовательной.

Во-вторых, обсуждаемый курс преподается по-русски и для русскоязычного слушателя. Стало быть, термины UML должны так или иначе быть переданы кириллическими буквосочетаниями. Язык UML сравнительно молод (но уже моден), бум публикаций в отечественной литературе начался только сейчас (с опозданием на десять лет, как обычно), поэтому устоявшейся терминологической традиции пока нет. Хотелось бы использовать хорошие русские слова – «эктор» и «персистентный» решительно отмечают. Словарь в таких случаях бесполезен (хотя им многие пытаются пользоваться). Остается фантазировать, опираясь на вышедшие из печати труды коллег, отечественные программистские традиции и собственный опыт. Основной критерий, который использован нами при выборе переводов терминов – как можно точнее передать *смысл* исходного термина (но не звучание, не морфологию и не буквальное значение). Таким образом, третья проблема преподавания UML – терминологическая.

### ТРИ ПРЕДСТАВЛЕНИЯ МОДЕЛИРОВАНИЯ НА UML

Наш способ решения проблем преподавания UML состоит в следующем.

Проблему сложности формального описания синтаксиса и семантики языка мы решаем тем, что описание каждой конструкции начинается с прагматики, с ответа на вопрос «зачем это нужно?». Изложение ведется на примерах. При этом мы избегаем «учебных» примеров, в которых все слишком упрощено и потому неубедительно. Мы рассматриваем один развернутый пример – информационная система отдела кадров, – и на этом примере демонстрируем все без исключения конструкции UML в условиях, «приближенных к боевым». Только после того как основная идея схвачена и слушатель понимает, зачем нужна эта конструкция и какая от нее бывает практическая польза, мы предлагаем сделать усилие и изучить формальные определения.

Вторая проблема, связанная с неадекватным исходным представлением о предмете, то есть фактически преодоление предубеждений, является самой сложной, по нашему мнению. Мы решаем эту проблему, структурируя материал не по частным различиям конструкций, не по версиям языка и не по типам диаграмм, а по способам применения языка.

Было бы очень соблазнительно иметь возможность строить модели любых систем для любых целей единообразно, придерживаясь, так сказать, одной универсальной точки зрения. Во многих ранних методологиях моделирования программных систем такие попытки (более или менее удачные) предпринимались. Как показывает практический опыт, не удается описать с единой точки зрения все без исключения аспекты моделируемой системы. Действительно, в модели нужно отразить множество вещей: интерфейсы для взаимодействия с внешним миром, внутреннюю логическую структуру программы, структуру хранимых данных, алгоритмы функционирования, состав артефактов, включаемых в поставку, и многое другое. Было бы самонадеянно утверж-

дать, что единое средство описания всех аспектов сразу в принципе невозможно, – просто пока мы не знаем такого средства. Отсюда следует вывод: моделировать сложную систему следует с нескольких различных точек зрения, каждый раз принимая во внимание один аспект моделируемой системы и абстрагируясь от остальных. Этот тезис является одним из основополагающих принципов UML, может быть самым важным принципом, предопределившим практический успех UML.

Идея состоит в том, что абстрактный граф модели, состоящий из множества разнотипных сущностей и отношений, не подлежит конструированию или изучению в целом. Каждый раз для визуализации, изменения или иных манипуляций из этого общего графа вычлняются только сущности и отношения, релевантные определенному аспекту моделируемой системы, а все остальные игнорируются. Такой вид с определенной точки зрения, можно сказать, проекцию модели, мы называем представлением (view). Можно сказать, что представление – это средство логического структурирования модели. Представления не являются нашим изобретением – в книге [1] приведено пять представлений, а в книге [2] – восемь.

Учитывая неформальный характер самого понятия представления и опираясь на собственный опыт использования UML, мы рискнули предложить свой вариант набора представлений. Их всего три.

**Представление использования.** Представление использования призвано отвечать на вопрос, что делает система полезного. Определяющим признаком для отнесения элементов модели к представлению использования является, по нашему мнению, явное сосредоточение внимания на факте наличия у системы внешних границ, то есть выделение внешних действующих лиц, взаимодействующих с системой, и внутренних вариантов использования, описывающих различные сценарии такого взаимодействия. Таким образом, единственным выразительным средством представления использования оказываются диаграммы использования.

**Представление структуры.** Представление структуры призвано отвечать (с разной степенью детализации) на вопрос, из чего состоит система. Определяющим признаком для отнесения элементов модели к представлению структуры является явное выделение структурных элементов – составных частей системы – и описания взаимосвязей между ними. Принципиальным является чисто статический характер описания, то есть отсутствие понятия времени в любой форме, в частности, в форме последовательности событий и/или действий. Представление структуры описывается прежде всего и главным образом диаграммами классов, а также, если нужно, диаграммами компонентов, размещения, внутренней структуры и, в редких случаях, диаграммами объектов.

**Представление поведения.** Представление поведения призвано отвечать на вопрос, как работает система. Определяющим признаком для отнесения элементов моде-

ли к представлению поведения является явное использование понятия времени, в частности, в форме описания последовательности событий/действий, то есть в форме алгоритма. Представление поведения описывается диаграммами автомата и деятельности, а также обзорной диаграммой взаимодействия, диаграммами коммуникации и последовательности. В редких случаях можно воспользоваться диаграммой синхронизации.

Такой набор представлений является ортогональным и согласованным с классификацией диаграмм. Более того, он во многом инспирирован личным опытом моделирования. По нашему мнению, процесс моделирования (независимо от назначения модели) является не линейным последовательным, а итеративным и параллельным, примерно таким, как показано на рис. 2.

Другими словами, процесс моделирования циклический, на каждом шаге может присутствовать уточнение представления использования, за которым следует параллельное моделирование структуры и поведения.

Исходя из сказанного, мы положили свой набор представлений в основу структуры курса и в основу решения второй проблемы. Представляя себе процесс, нарисованный на рис. 2, в целом, «с высоты птичьего полета», учащийся в меньшей степени тяготится неизбежными формальными определениями.

Решение же третьей проблемы – терминологической – очень просто и требует только силы воли. Нельзя позволять ни себе, ни учащимся использовать некорректные, неправильные переводы или транслитерации. Мы знаем, что противостоять негативным тенденциям уродования родного языка невероятно трудно, но мы убеждены, что это совершенно необходимо.

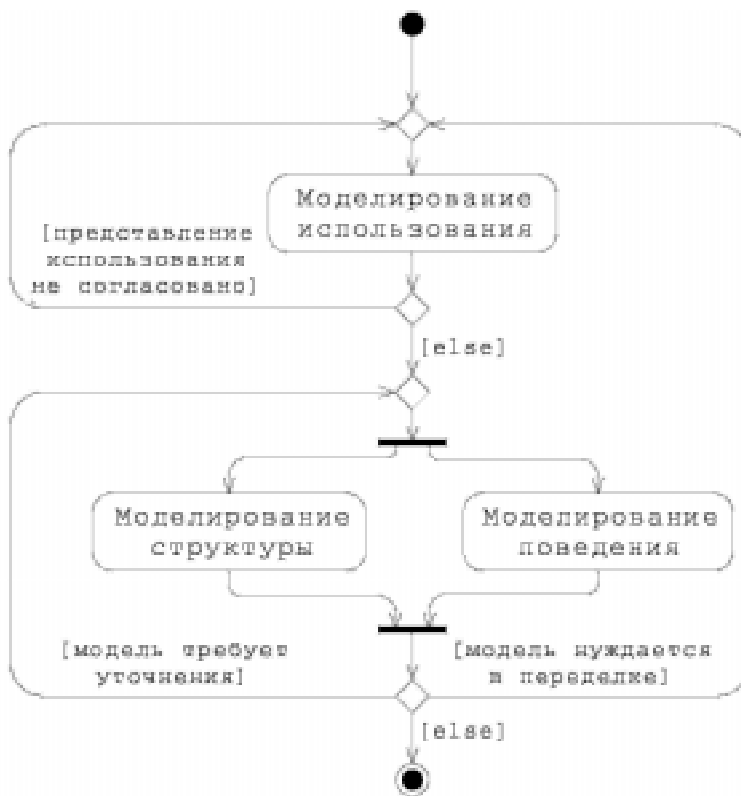


Рис. 2. Процесс моделирования

## НЕКОТОРЫЕ ДИДАКТИЧЕСКИЕ НАБЛЮДЕНИЯ

В 2009 году курс «Моделирование на UML» прошел всестороннюю апробацию.

Во-первых, мы собрали и систематизировали весь накопленный материал в книгу [3], которая имеет несколько областей применения, способов использования и целевых читательских аудиторий. С одной стороны, это учебник для начинающих, преимущественно студентов старших курсов технических университетов, проходящих обучение по направлениям подготовки, связанным с прикладной математикой, информатикой и программированием. Как учебник, книга содержит явные определения основных понятий, используемых при моделировании программного обеспечения, точные описания основных конструкций языка UML и многочисленные примеры их использования. Материал для начинающих составляет примерно половину объема книги. С другой стороны, это справочник и «поваренная книга» (cookbook) для опытных пользователей языка, желающих углубить свои знания и усовершенствовать навыки. Для достижения этой цели в книгу включен обширный справочный материал, включающий аннотированную библиографию, краткий, но полный толковый словарь терминов и сравнительный обзор инструментальных средств. Далее, изложение семантики конструкций UML проводится на сквозном примере – информационной системе отдела кадров. Результатом моделирования является набор семантически законченных моделей, объединенных одной предметной областью. Более того, в книге приводятся не только готовые диаграммы, но и описываются рассуждения и промежуточные шаги, приведшие авторов к тому или иному проектному решению. Тем самым мы включили в книгу свой опыт моделирования на UML, проверенный за время проведения многочисленных тренингов для множества организаций и групп разработчиков. Практические советы, рекомендации, рецепты и справки занимают примерно треть объема книги. С третьей стороны, в книгу включена «информация к размыш-

лению» для экспертов и знатоков. Мы имеем в виду многочисленные отступления, которые раскрывают взгляды авторов (часто апокрифические) на источники развития и перспективы применения UML, влияние UML на процесс разработки, связь UML с другими нотациями и тому подобное. Этот материал занимает немного места, но мы считаем его ценным и важным.

Самым важным с точки зрения преподавания является то обстоятельство, что мы имеем примерно вдвое больше материала в запасе, чем может быть изложено в каждом отдельном прогоне курса. Модульная структура материала позволяет варьировать уровень, сложность и предметную ориентацию изложения динамически, в соответствии с желаниями и возможностями аудитории, что очень важно в интерактивных формах обучения.

Во-вторых, как обычно, были проведены регулярные лекции с упражнениями для студентов. Лекции проводятся на основе обширной (более 500 единиц) коллекции слайдов, из которой собираются нужные презентации. Лекции подкреплены упражнениями, состоящими в разработке законченной модели не очень сложного приложения. Эта форма апробации традиционна, и мы на ней не останавливаемся.

В-третьих, мы провели бесплатные онлайн тренинги для двух групп слушателей. Поскольку эта форма сравнительно новая, участие в он-лайн тренингах сугубо добровольное, нам представляется весьма интересной и поучительной информация о том, кто является слушателями таких тренингов, то есть каковы особенности целевой аудитории обсуждаемого курса. На рис. 3 приведены две гистограммы – распределение слушателей по возрасту и распределение по должностям.

Мы видим, что потребителями курса «Моделирование на UML» являются высококвалифицированные инженеры и менеджеры наиболее продуктивного возраста. Большое наличие тестировщиков объясняется желанием последних получить дополнительные знания и переквалифицироваться в программистов.

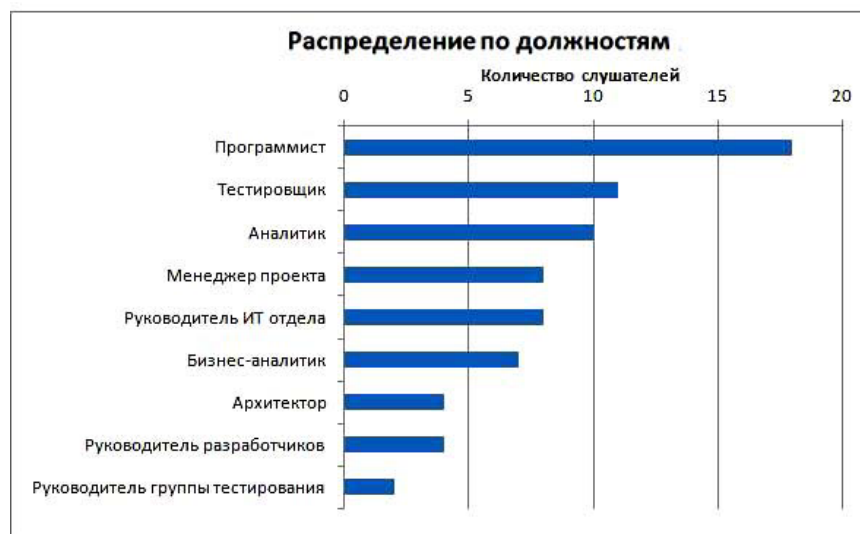


Рис. 3. Распределения слушателей по категориям

В-четвертых, опираясь на актив мотивированных и заинтересованных участников, собранный на он-лайн тренигах, мы поддерживаем сайт <http://www.uml3.ru>, на котором выложено в свободный доступ большое количество материалов, связанных с проектированием программного обеспечения (рис. 4).

Наконец, в-пятых, мы провели очередной очный корпоративный тренинг для наиболее подготовленных слушателей. Этот тренинг заснят на видео и прилагается к исходной книге. Таким образом, круг замкнулся.

### ЗАКЛЮЧЕНИЕ

Изменение жизненных реалий требует освоения и внедрения в практику новых форм обучения. Например, для получения

образования «выше высшего» проведение занятий он-лайн в Интернете является одним из наиболее эффективных способов. Но проведение таких занятий требует особым образом подготовленный материал. В то же время само обучение в новых формах вырабатывает такие материалы в качестве «побочного эффекта» (например аудио и видеозаписи занятий). Это означает наличие петли обратной связи: курс влияет на слушателей, а в ответ слушатели влияют на курс. Обратная связь есть и в традиционном способе обучения: студенты слушают курс, а через некоторое время, оставшись на кафедре, сами начинают преподавать. Отличие преподавания в Интернете состоит в том, что цикл обратной связи занимает недели или даже дни, а при традиционном способе этот цикл занимает годы.



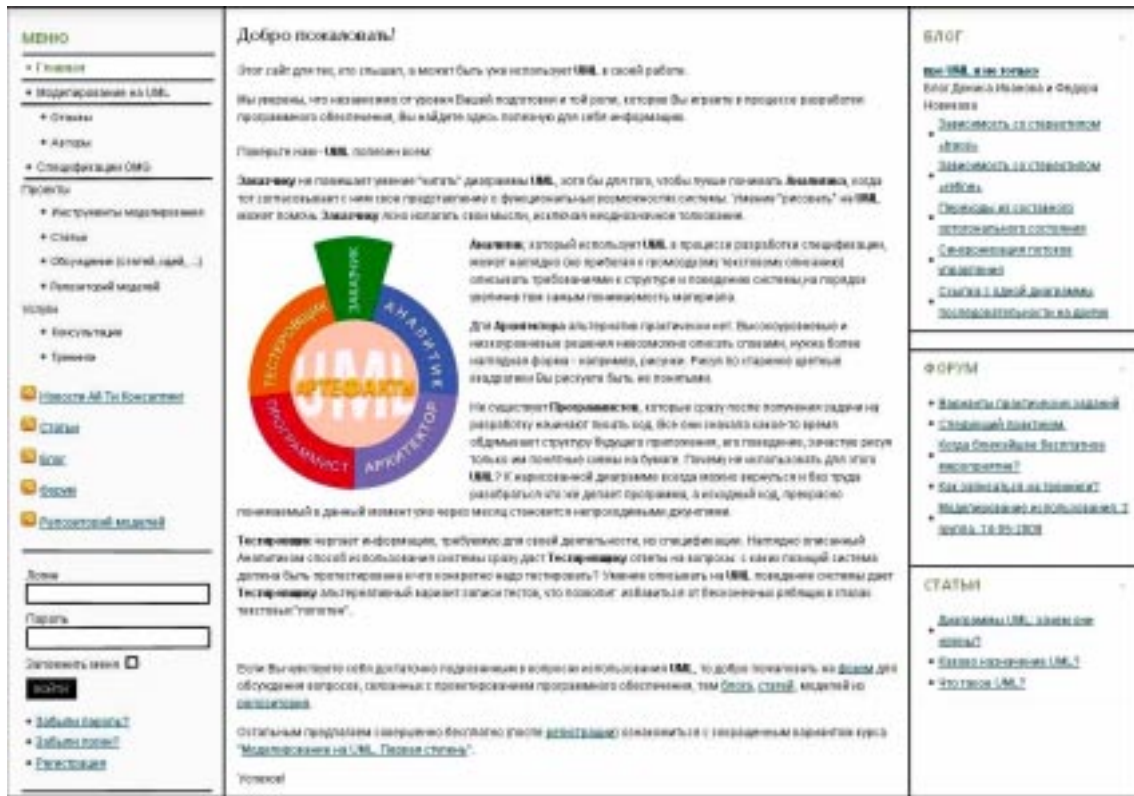


Рис. 4. Сайт UML 3

Мы не считаем, что обучение через Интернет должно вытеснить традиционные формы. Напротив, для приобретения фундаментальных знаний и навыков именно тра-

диционные формы наиболее адекватны. Но для обучения таким динамичным вещам, как моделирование на UML, как показывает наш опыт, Интернет подходит лучше всего.

**Литература**

1. Буч Г., Рамбо Д., Якобсон А. Язык UML. Руководство пользователя. Второе издание. М.: ДМК, 2006.
2. Буч Г., Якобсон А., Рамбо Д. UML. 2-е издание Классика CS. СПб.: Питер, 2005.
3. Иванов Д., Новиков Ф. Моделирование на UML. СПб.: Наука и техника, 2009.

**Abstract**

On-line Internet training experience on modeling with UML, including preparation of interactive textbooks, video presentations, remote tests is described.

*Иванов Денис Юрьевич,  
ИТ-консультант, Ай Ти Консалтинг,  
denis.ivanov@it-konsulting.spb.ru*

*Новиков Фёдор Александрович,  
кандидат физико-математических  
наук, заведующий лабораторией  
астрономического  
программирования ИПА РАН,  
fedornovikov@rambler.ru*

