

УДК 371.3

О. И. Гороховский, к. т. н., доц.; Т. И. Трояновская

ИНФОРМАЦИОННАЯ ТЕХНОЛОГИЯ ПОСТРОЕНИЯ АДАПТИВНОЙ СИСТЕМЫ ДИСТАНЦИОННОГО ОБУЧЕНИЯ

Статья посвящена описанию разработки информационной технологии построения адаптивных систем дистанционного обучения с учетом статической и динамической составляющих предметно-ориентированной доминанты студента, а также с учетом индивидуального временного графика и скорости восприятия материала студентом.

Ключевые слова: информационная технология, адаптивная система дистанционного обучения, предметно-ориентированная доминанта.

Введение

Для предоставления студенту дистанционной формы обучения (ДФО) возможности получать качественное дистанционное образование Г. Вебер [1] предлагает дополнить систему дистанционного обучения (СДО) адаптивностью. Адаптивная СДО (АСДО) “подстроится” к особенностям обучения отдельно взятого студента. Такое направление очень **актуально**, поскольку сложность дисциплин (особенно специализированных) возрастает и составить определенный усредненный план подачи материала для них очень сложно. Это, в свою очередь, снижает общий уровень качества образования и составляет значительную часть в перечне проблем современной образовательной системы.

Целью данной статьи является разработка информационной технологии построения АСДО, способной приспосабливаться к особенностям восприятия учебного материала студентом. Для **достижения поставленной цели** система должна содержать компоненты сбора, обработки, хранения данных и их использования при выдаче студенту материалов дистанционного курса (ДК). Для этого в АСДО включены такие модули: модуль сбора данных о деятельности студента (для сбора первичных данных по прямым и косвенным оценкам [2]); модуль выдачи методических материалов для непосредственного отображения лекционных и практических материалов; автоматический модуль обработки данных и выдачи материала (АМОД) для обработки первичных данных оценивания и подбора учебных материалов и контрольных элементов.

Технология разработки системы, включающей такие модули, состоит из следующих этапов: разработка структур данных; разработка АРМ преподавателя; разработка АМОД (подсистемы выдачи студенту материалов, подсистемы анализа данных и корректировки курса); разработка АРМ студента. Диаграмма соответствующего технологического процесса показана на рис. 1.

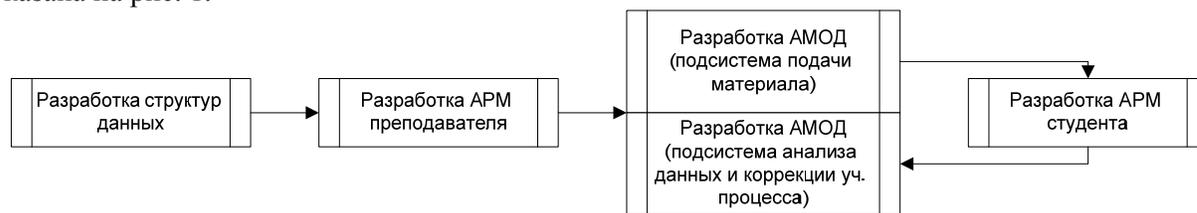


Рис. 1. Диаграмма технологического процесса разработки АСДО

Разработка АМОД и разработка АРМ студента показаны на данном рисунке как циклический процесс, поскольку они взаимосвязаны.

1. Разработка структур данных

В общем случае СДО базируется на представлении материалов, входящих в состав курсов, предлагаемых студентам. Анализируя современные электронные учебники (ЭУ), можно сказать, что описать ДК можно с помощью файла формального содержания. АСДО может использовать этот файл для определения текущего местонахождения студента в ДК и для отслеживания пути пользователя. Выделяют линейный и иерархический форматы, которые определяют содержание курса:

1. Линейный формат. Используется для реализации ДК по традиционной схеме ЭУ с линейной компоновкой материала (рисунок 2).
2. Иерархическое представление. Кроме элементов курса содержит вложенные категории (модули, темы, лекции), на которые логически разделен ДК (рисунок 3).



Рис. 2. Схема линейного представления

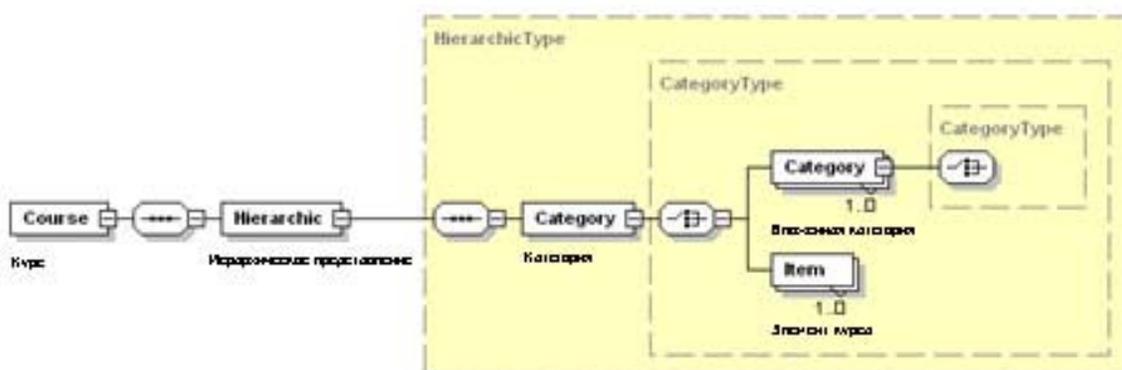


Рис. 3. Схема иерархического представления курса

Для организации ДК достаточно существования файлов одного из вышеназванных типов и гарантированного существования каждого элемента курса. Для обеспечения целостной передачи курса, разработанного в автономном режиме в АСДО, разработан модуль генерации ЭУ для перевода материала в необходимый формат – АРМ преподавателя.

2. АРМ преподавателя

Разработка ДК состоит в создании структурных единиц курса, из которых организован курс. В силу необходимости дозированной подачи студенту материала соответственно его уровню подготовки и хода его обучения, введем дополнительное определение «понятие» – логически завершённого фрагмента учебного материала.

Технологический процесс разработки ДК состоит в выполнении следующих действий: 1) разработки плана курса; 2) информационного наполнения плана курса; 3) дробления материала на отдельные понятия; 4) формирования последовательности структурных элементов курса из понятий соответственно созданному плану курса.

Для обеспечения этого технологического процесса необходимо выделить основные структурные элементы курса. Они могут соответствовать как традиционной (статической), так и динамической компоновке ЭУ: модуль, тема, лекция, понятие. В данной разработке используется динамическое компонование учебного материала, поскольку, чтобы

обеспечить студенту подачу материала соответственно его уровню, эта компоновка должна формироваться и корректироваться в процессе хода обучения.

Таким образом, обобщенная технологическая процедура выглядит так:

1. Разбивка массива информации по дисциплине на отдельные понятия.
2. Логическое группирование понятий.
3. Информационное наполнение понятия. Для каждого понятия должны быть предусмотрены различные варианты подачи материала.
4. Группирование понятий в лекции соответственно логическому группированию.
5. Построение логической последовательности лекций (проекта курса).
6. Группирование лекций в темы и модули.
7. Создание единого индекса-файла (содержание курса в форме, которая может быть выдана на АРМ студента). Сразу заметим, что этот файл может изменяться соответственно ходу обучения студента, в отличие от файла оглавления, который создается на этапе проектирования ДК.
8. Создание пакета курса (например, в виде Zip-файла по стандарту SCORM).

Для реализации такой технологической цепочки предназначен АРМ преподавателя, на вход которого подается набор материалов, а на выходе имеем логически организованный ДК, который можно разворачивать на сервере. Серверная часть АМОД принимает созданный курс и подает его студенту.

3. Разработка АМОД: подсистема выдачи материалов

АМОД разворачивается на сервере и является набором программ, обрабатывающих запросы пользователя, или других программ, обращающихся к нему. Общая функциональная модель работы АМОД показана на рисунке 4.



Рис. 4. Общая функциональная модель работы АМОД

Для реализации серверных программ обработки данных была избрана технология сервлетов [3]. Агент сбора данных принимает и присылает запросы в XML-формате. HTML-запросы могут быть двух вариантов – get-запрос и post-запрос (рисунок 5).

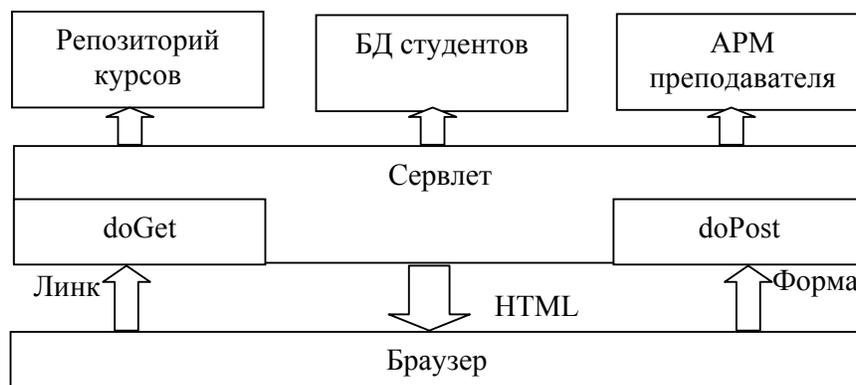


Рис. 5. Структурная модель АМОД, реализованного на сервлетах

Get-запрос используется для перенаправления студента на служебные страницы профиля

или для обеспечения работы агента контроля сессии. Post-запрос ссылается в виде набора значений «ключ-значение», это удобно для посылки результатов тестирования, например, где каждому вопросу соответствует условное имя, а значением является ответ студента [3, 4]. Сервлет имеет абсолютный доступ к ресурсам сервера.

Сервлеты отвечают за реализацию предметно-ориентированной доминанты (ПОД) студента [5], реализуя его динамическую часть. Первичные данные, собранные модулем сбора данных, через Post-запрос поступают к сервлету и БД студентов.

4. Разработка АРМ студента

Задача АРМ студента – сбор данных о деятельности студента в процессе обучения, которые являются входными данными для подсистемы корректирования курса и динамической частью ПОД [6]. Модуль сбора данных – автономная программа АРМ студента, передающая данные о ходе обучения студента на АМОД. На рисунке 6 приведена схема описания процесса работы модуля сбора данных. Sender – модуль активного компонента – отвечает за пересылку сигнала сервлету. Scanner следит за событиями на стороне пользователя. Формат данных, с помощью которого присылаются сигналы о возникновении определенного события на стороне пользователя, показан на рис. 7. Далее переданный сигнал подается на селектор, который считывает сигнал, декодирует его и определяет, какой именно сервлет отвечает за обработку этого сигнала по таблице соответствия.

Опишем сигналы, которые соответствуют событиям, произошедшим на стороне пользователя, и назначим им соответствующие сервлеты: Bound Start Time – осуществляет запись в БД стартовой пометки таймера при переходе на страницу или в начале работы с материалами лекции; Bound End Time – осуществляет запись в БД конечной пометки таймера; Perform Test – обрабатывает данные, полученные во время прохождения студентом тестовой задачи, осуществляет их анализ. Разные сигналы предусматривают внесение изменений в разные части БД и в динамическую часть ПОД.

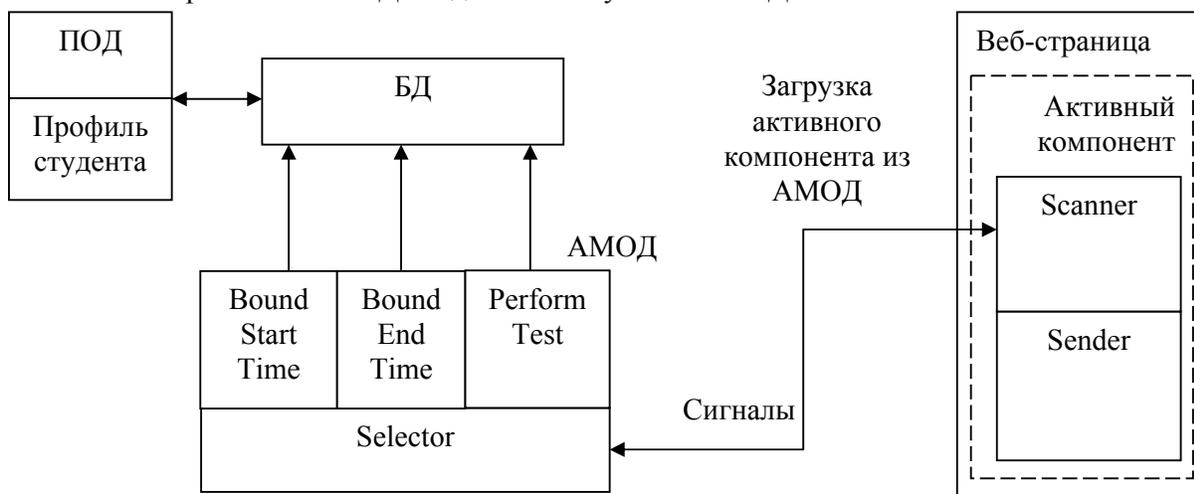


Рис. 6. Схема функционирования модуля сбора данных

По своей структуре апплет представляет собой фоновую нить процесса внутри браузера, которой выделяется отдельный фрагмент страницы, где размещен пользовательский интерфейс апплета. В языке разметки HTML для этого предусмотрен специализированный тег `applet`, который описывает расположение такого компонента [4, 7]. Запуск апплета происходит одновременно с загрузкой веб-страницы на рабочую станцию пользователя, а остановка процесса наступает после того, как страница закрывается или происходит переход на следующую. Во время пребывания пользователя на странице компонент апплета имеет неограниченный доступ к контенту страницы, может реагировать на действия пользователя

через обработчики событий и логическое имя средствами специального объекта JSObject.

Следующим этапом после снятия данных агентом сбора данных, является пересылки сообщений. С целью облегчения работы с посылкой и получением данных в качестве носителя целесообразно использовать формат XML. На рисунке 7 показана схема сообщения, которое присылается на АМОД.

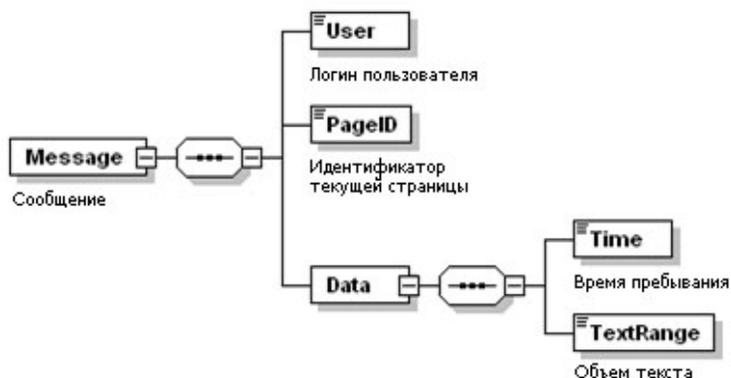


Рис. 7. Схема сообщения, которое присылается на АМОД

Этот формат можно расширить, изменив схему документа, здесь от начала заложена возможность эффективного добавления новых возможностей без дополнительных затрат.

Поскольку большую часть своей деятельности студент проводит в интернет-браузере, пользуясь им как основным инструментом работы с материалами ДК, очевидно, что АРМ студента должен быть реализован с помощью таких технологий, которые бы позволяли интегрировать все необходимое в интернет-браузер.

Статическая часть АРМ (профиль), соответствующая статической части ПОД, реализована в виде веб-страницы. Динамическая часть ПОД - частично возложена на АМОД, а частично реализована с помощью апплета. ПОД разделена на две части для обеспечения эффективности работы с данными о ходе обучения студента. Статические данные, являющиеся постоянными вдоль большого промежутка времени, сохраняются в отдельной защищенной таблице данных (на рис. 8 – БД пользователей), что однозначно идентифицирует студента в системе. Динамические данные сохраняются в БД, где они могут быть легко откорректированы соответственно ходу обучения.

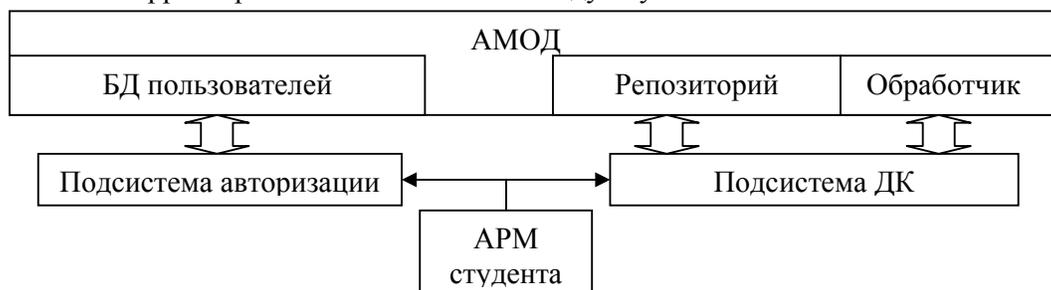


Рис. 8. Взаимодействие АРМ студента из АМОД

При регистрации студента в АСДО подсистема авторизации обращается к таблицам статической части ПОД и осуществляет идентификацию студента. При дальнейшей работе подсистема ДК, основываясь на установленной идентификации, выполняет задачи выдачи студенту учебного материала и обработки первичных данных непрямого оценивания (НО), которые являются первичными данными для динамической ПОД, что является механизмом корректирования учебного процесса.

5. Разработка АМОД: подсистемы анализа данных и корректирование курса

Для завершения технологического цикла разработки системы АСДО следует добавить к АМОД подсистему корректирования курса. На рисунке 9 представлена общая схема АСДО.

Архитектура АМОД, основанная на сервлетах, позволяет организовать модуль анализа данных и корректирование курса в виде сервлета, на вход которого подаются данные о ходе обучения студента. На основе таких сообщений формируется информация о пути студента и НО, которые вносятся в динамическую часть ПОД, и значит – влияют на способ подачи учебного материала студенту.

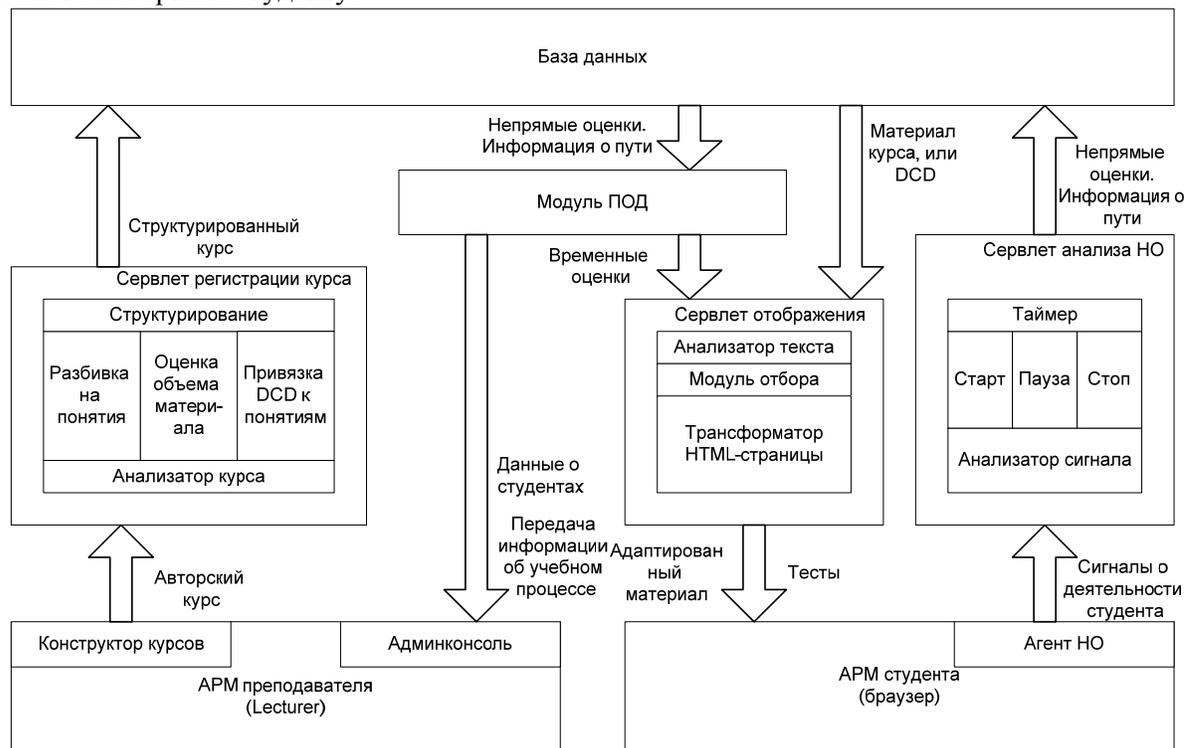


Рис. 9. Технологическая диаграмма АСДО

Выводы

В данной статье предложена информационная технология разработки АСДО, которая подает материал студентам с учетом особенностей хода их обучения при помощи использования технологического решения на основе сервлетов и агента сбора данных, расположенных на стороне пользователя. Объединяя приведенные в статье технологические решения составляющих, получена целостная технология построения АСДО, которая позволяет создавать адаптивную среду для предоставления студентам услуг ДФО.

Особенности предложенной информационной технологии:

1. Динамическое формирование наполнения курса осуществляется соответственно ходу обучения отдельного студента, благодаря разбивке учебного материала на отдельные понятия, которые потом группируются в структурные элементы курса.
2. Уменьшение нагрузки на пользовательскую сторону за счет использования сервлетов как основы реализации АСДО.
3. Универсальность данной системы за счет использования стандартов XML и HTML как основных носителей данных.

СПИСОК ЛИТЕРАТУРЫ

1. Gerhard W., Marcus S. User Modeling and Adaptive Navigation Support in WWW-Based Tutoring Systems // User Modeling: Processings of the Sixth International Conference. Vienna: CISM. 1997.
2. Гороховський О.І., Трояновська Т.І., Кисюк Д.В. Автоматизація роботи викладача дистанційної форми навчання за допомогою непрямих оцінок // «Наукові дослідження – теорія та експеримент». Полтава, 14 - 16 травня, 2007. С. – 127-131.
3. Hunter J. , Crawford W. Java Servlet Programming. O'Reilly. 1998.
4. Musciano C., Kennedy B. / HTML: The Definitive Guide, 3rd Edition. O'Reilly. 1998.
5. Трояновська Т.І. Розробка комп'ютерної підсистеми аналізу та формування предметно-орієнтованої домінанти студента системи дистанційного навчання // Вісник Черкаського державного технологічного університету. - 2007. - № 3-4. - С. 41-46.
6. Гороховський О.І., Трояновська Т.І., Снігур А.В. Динамічна складова предметно-орієнтованої домінанти студента дистанційної форми навчання // Електромашинобудування та електрообладнання. Міжвідомчий науково-технічний збірник. - 2008. - № 70. - Київ. : "Техніка". С. 28-32.
7. Гороховський О.І., Трояновська Т.І. Моделювання, створення та практика автоматизованих систем дистанційного навчання // Інформаційні технології та комп'ютерна інженерія. - 2007. - С. 235-239.

Гороховский Александр Иванович, к. т. н., доцент кафедры вычислительной техники тел.: (0432) 59-84-05, e-mail: goroh@vstu.vinnica.ua.

Трояновская Татьяна Ивановна, аспирант, тел.: (063) 209-10-30, e-mail: trtet@mail.ru.
Винницкий национальный технический университет.