**Ваграменко Ярослав Андреевич,**

*ФГНУ «Институт информатизации образования» РАО,*

*заместитель директора по информационным образовательным ресурсам, д.т.н., профессор, ininforao@gmail.com*

**Яламов Георгий Юрьевич,**

*ФГНУ «Институт информатизации образования» РАО,*

*ведущий научный сотрудник, к.ф.-м.н.*

**Фанышев Роман Геннадьевич,**

*ФГНУ «Институт информатизации образования» РАО,*

*аспирант*

**ВЫБОР ИСТОЧНИКОВ ИНФОРМАЦИИ, ХАРАКТЕРА КОНТЕНТА, ОЦЕНКИ НАУЧНОЙ И СОЦИАЛЬНО ЗНАЧИМОЙ ИНФОРМАЦИИ ДЛЯ ПОДДЕРЖКИ САМООБРАЗОВАНИЯ**

**CHOICE OF SOURCES OF INFORMATION, CHARACTER
OF THE CONTENT, ASSESSMENT SCIENTIFIC AND SOCIALLY SIGNIFICANT INFORMATION FOR SELF-EDUCATION SUPPORT**

***Аннотация.*** В данной статье рассмотрены проблемы выбора источников информации из внешних информационных сред, характеристики социально значимой информации и процесс ориентации образовательного контента за счет использования программных компонентов среды программирования Visual Studio 2010 .Net и расширения языка C# – Knowledge.Net. Данные аспекты рассмотрены применительно к формированию базы знаний экспертной обучающей системы с точки зрения их интеграции со средой высокоуровневых языков программирования и хранения компонентов образовательного контента на языке представления знаний экспертной системы.

***Ключевые слова*:** образовательный контент; экспертная система; база знаний экспертной системы; язык представления знаний.

***Annotation.*** This article examines the problem of selecting the sources of information from external information environments, and social characteristics of meaningful information and process oriented learning content through the use of software component development environment Visual Studio 2010. Net language extensions and C # - Knowledge.Net. These aspects are considered in relation to the formation of a knowledge base of expert training system in terms of their integration with the environment of high-level programming languages and storage components of the educational content in the language of knowledge representation of expert system.

***Keywords:*** educational content; expert system; knowledge base of expert system; knowledge representation language.

Процессы формирования информационных ресурсов для целей самообучения в системе высшего образования происходят непрерывно и они параллельно связаны с традиционным образовательным процессом. С этой позиции можно сказать, что искомый образовательный информационный ресурс может быть несколько раз изменен и скорректирован в процессе самообучения с участием преподавателя и студента таким образом, чтобы его содержимое удовлетворяло требованиям учебного плана, в котором сосредоточены основные нормативы, качественные и количественные характеристики изучаемой предметной области. С точки зрения оценочных характеристик такого информационного ресурса можно судить о его социальной значимости в пределах преподаваемого курса и, как следствие, значимости получаемых умений и навыков предметной области будущей профессиональной деятельности студента.

При рассмотрении процесса ориентации информационного ресурса в процессе самообучения с точки зрения программной реализации можно выделить несколько основных пользователей и определить их форму участия в формировании базы знаний фрагментов образовательного контента.

Процесс формирования информационных моделей, методов и технических приемов создания баз знаний интеллектуальных информационных систем будем условно называть – инженерия знаний [1]. В данном процессе основными фигурами являются инженер знаний (преподаватель), специалист предметной области (преподаватель, специалист организационной части учебной деятельности – методист) и пользователь (студент).

Инженерия знаний образовательной предметной области в силу своей слабой структурированности является сложной задачей. Для ее решения необходимо рассмотреть и решить следующие проблемы: проблема выбора источников информации, проблема оценивания результирующего (извлеченного знания) образовательного контента в процессе самообучения и проблема интеграции компонентов приобретения знаний экспертной системы самообучения. Ниже более подробно рассмотрим данные проблемы, а так же методы и используемые технологии для их решения.

**Проблема выбора источников информации**

Для решения данной проблемы необходимо определить так называемую предметную область преподаваемых курсов. Рассматривая конкретные курсы, данная предметная область описывается учебным планом, который является центральным инструментом (вместе с рабочей программой) управления образовательным процессом, содержащим необходимые наборы разделов, тем, понятий, навыков и умений, которыми должен обладать в результате обучения студент. Однако, для обеспечения необходимого качественного уровня результирующих знаний, необходимо так же рассматривать внешние источники информации, которые могут быть сосредоточены в виде баз данных (БД), электронных каталогах, файловых и web-серверах вуза.

Как правило, данные внешние источники информационных ресурсов имеют не только различные форматы представления информации, но и методы их программной обработки. Основной задачей проблемы выбора источников информации является выбор такой технологии их интеграции (подключения) с существующей системой управления базой знаний, которая позволит реализовывать их поэтапный анализ, структурирование (приведение к общему формату) и ориентацию под конкретные задачи конкретного этапа обучения. К примеру, по учебному плану необходимо организовать тестовый контроль уровня знаний обучаемого по результатам изучения раздела, который выявит его особенности и подходы к освоению будущих разделов. В данном случае необходимо помимо использования собственных информационных ресурсов подключать внешние сервисы самообучения (банк вопросов на web-сервере), что становиться возможным, имея унифицированную внутреннюю среду информационных ресурсов с единым форматом и системой управления контентом (метасистемой).

Таким образом, выбор источников информации сводиться к поэтапному подключению необходимых внешних сервисов поддержки самообучения по заданному преподавателем алгоритму. Программным способом это реализовано в виде хранилища файлов – компонентов курса в едином мета – формате с соответствующей структурой XML дерева, которая позволила на любом этапе самообучения осуществить выбор необходимого фрагмента образовательного ресурса.

**Проблема оценивания результирующего (извлеченного знания) образовательного контента.**

Данная проблема при детальном рассмотрении образовательного процесса по основным этапам самообучения сводиться к оцениванию основных фрагментов образовательного контента, хранящегося в базе знаний экспертной системы и во внешних источниках.

Как и в случае с выбором источников информации центральная и определяющая роль оценивания отдается по прежнему инженеру знаний – преподавателю, а оценочные характеристики результирующего образовательного контента складываются из весовых коэффициентов составляющих единиц, которые математически представлены в виде узлов математического направленного графа. Программным способом реализация подобного графа осуществляется при помощи объекта TreeView среды Visual Studio 2010 .Net. Узлы (nodes) данного дерева связаны с полями таблиц реляционной базы данных. Ключевые поля таблиц БД определяют основные уровни иерархии представления образовательных единиц (понятие, термин, раздел, подтема и т.д.), а остальные поля таблиц БД формируются из описательной информации в виде кортежа данных и имеет формат, описанный ниже:

<*понятие предметной области нижнего уровня1> =<раздел предметной области верхнего уровня1 > + <оценочные характеристики образовательного контента1> + <базовые алгоритмы и условия поэтапного контекстного обучения1>*

В рабочей памяти экспертной системы данная запись, состоящая из кортежа (выборки) данных представляется в виде правила (продукции). Программным способом данный функционал реализован в виде объектно-ориентированного класса RulSet.

Используя данный подход к построению соответствующей модели образовательного контента в виде графа и дерева преподавателю достаточно на начальном этапе формирования информационного ресурса ввести весовые коэффициенты его фрагментов в соответствующие узлы дерева и сформировать логические продукции вывода рекомендаций, опираясь на фактические шкалы, сохранив их в базе знаний экспертной системы. В дальнейшем преподавателю необходимо контролировать корректность автоматических решений экспертной системы, редактировать, добавлять и удалять необходимые смысловые единицы в результирующий общий (general) граф, описывающий предметную область конкретного курса.

Таким образом, имею подобную гибкую систему описания образовательного контента с использованием моделей графа и дерева, можно организовать совокупные оценочные характеристики образовательного контента по всем смысловым параметрам и единицам электронного образовательного ресурса, а алгоритм оценивания реализовать автоматически путем хранения данных картежей в виде продукций экспертной системы. Причем каждая составная единица информационного ресурса четко определена и идентифицирована с точки зрения программной реализации и учебного плана за счет иерархии и декомпозиции.

**Проблема интеграции компонентов приобретения знаний экспертной системы самообучения.**

Данная проблема является следствием описанных выше проблем в силу отсутствия необходимого программного инструментария и интерфейсов преобразования компонентов образовательного контента на всех этапах его жизненного цикла внутри базы знаний экспертной системы. Для того чтобы ее решить опишем основные этапы жизненного цикла электронного образовательного ресурса:

1. Выбор источника информационного ресурса на конкретном этапе изучения – результат – подключение (интеграция) внешнего сервиса хранения.
2. Декомпозиция образовательного ресурса на фрагменты, имеющие смысловую суть в рамках интересующей предметной области – результат – построение математического графа – дерева с иерархией основных единиц предметной области.
3. Разработка правил логического вывода фрагментов образовательного ресурса в виде рекомендаций по самообучении. – результат – набор кортежей данных, описывающих предметную область с различных аспектов, весовых коэффициентов и степеней интеграции от верхнего уровня.
4. Предоставление доступа к базе знаний пользователям студентам – результат – интерфейс доступа к базе знаний предметной области.

Для решения данной проблемы необходимо разбить фрагмент образовательного ресурса на уровни его представления, начинаю с первого этапа и заканчивая его предоставлением в пользование. Последний этап жизненного цикла образовательного ресурса представляет в данном случае верхний уровень – уровень пользователя, а первый этап – нижний уровень – уровень языка представления знаний экспертной системы. Ниже опишем используемые технологии объектно-ориентированного программирования и функциональные особенности языка CLIPS для построения интерфейсов представления информации в базе знаний экспертной системы.

*1 этап.* Реализуется за счет стандартного инструментария среды Visual Studio 2010.Net DataSource, который предоставляет доступ к базам данных.

*2 этап.* Реализуется за счет использования языка представления знаний Knowledge.Net [1], в котором используется следующие модели представления знаний: продукционная, фреймовая, сетевая, иерархическая.

*3 этап.* Реализован с использованием средств и функционал языка CLIPS и библиотеки *Mommosoft.ExpertSystem*, в которой представлены классы основных функций программирования базы продукционных правил и реализованы основные компоненты работы классической экспертной системы, такие как: база знаний, рабочая база знаний, набор фактов, набор фреймов и т.д.

*4 этап.* Реализован с использованием портальных решений и клиент- серверных технологий доступа к базе знаний за счет использования машины логического вывода кортежей данных, описывающих этапы самообучения и выдачи рекомендаций.

Общий принцип работы программных компонентов экспертной системы представлен на рисунке ниже (рис. 1).

Рис. 1 Общий принцип работы

программных компонентов экспертной системы

Таким образом, язык представления знаний экспертной системы состоит из статических атрибутов и методов, которые предоставляют интерфейс для доступа к фреймам, концептам и наборам правил, позволяют сохранить и загрузить данные в/из базы знаний.

Для решения описанных проблем использовано сочетание методов программной инженерии и методов инженерии знаний с преобладанием первой.

**Обзор существующих информационных систем поддержки самостоятельной работы студентов**.

Ниже рассмотрим несколько систем для организации самостоятельной (тьюторской) поддержки самообучении, такие как: Moodle, «Прометей», «1С:Образование». С помощью системы дистанционного обучения (СДО) «Прометей» можно построить в Интернет или Интранет виртуальный университет. Эта система предоставляет множество функций для организации учебного процесса и общения пользователей. Основные функции системы рассчитаны на тестирование, обмен файлами, чат, форум, но данная система не предоставляет учебный материал. Еще несколько недостатков:

* требования к конфигурации сервера и базового программного обеспечения;
* необходимость установки программного обеспечения и базы данных на сервере заказчика;
* привязка к продуктам Microsoft.

С другой стороны, пользователю разрабатываемой информационной системы необходимо иметь только браузер и подключение к Интернету.

Среда дистанционного обучения Moodle является современной, прогрессивной, постоянно развивающейся средой. Разработчику учебно-методических комплексов она предоставляет возможности использовать все необходимые ресурсы и средства контроля. Большим достоинством является распространение системы по лицензии GPL, что позволяет, не нарушая ничьих авторских прав свободно использовать, распространять и модернизировать систему. Вместе с тем, Moodle имеет значительный недостаток: в системе не предусмотрены группы уровня сайта, что делает очень сложным учет студентов разных специальностей. Группы в Moodle существуют не для управления правами доступа к курсам, а для разделения групп слушателей в одном курсе. Чтобы одни слушатели не видели активность других. Группы создаются внутри курса и не могут быть перенесены в другие.

Кроме этого, оценками слушателя можно оперировать только внутри курса. Нет возможности составить итоговую ведомость, например, по всем дисциплинам семестра, да и само понятие семестра в базовой версии системы отсутствует. Из сказанного можно сделать вывод, что Moodle является системой, ориентированной на западную модель обучения.

«1С:Образование» является системой программ для поддержки и автоматизации образовательного процесса. С помощью системы программ «1С:Образование» можно создавать и использовать в учебном процессе различные образовательные комплексы. Образовательные комплексы могут содержать в себе разнообразные наглядные, справочные, тестовые и другие материалы. Данная система использует «Единую коллекцию цифровых образовательных ресурсов», отслеживает состояние работы учащихся в реальном времени, редактирование учебных материалов, организация общения внутри группы в реальном времени (чат) и обмен почтовыми сообщениями, контроль и самоконтроль учебной деятельности пользователей. Но «1С:Образование», в основном, рассчитана на организацию учебного процесса в школе.

Таким образом, хоть и существует множество систем для организации поддержки самостоятельной работы студентов, но ни одна из них не дает полного спектра функций для реализации экспертной оценки качества предоставляемого и формируемого знания, а так же алгоритмы и функционал, учитывающий индивидуальные особенности обучаемых.

Еще одним примеров решения проблем создания интеллектуальных обучающих систем является проект «IDEA» (создание экспертных систем в области обучения по различным предметным областям) [4]. Он был направлен на создание на основе автоматных моделей [2] моделей ученика и учителя, которые взаимодействуют друг с другом через пространство учебного материала, формализованного в виде, например, размеченных информационных деревьев или нагруженных графов более общего вида. Удалось построить удачные примеры обучающих систем в области изучения иностранных языков (в том числе и с применением экспертной системы), которые, однако, не были развиты до своего полного завершения из-за необходимости создания большого набора решающих правил, что требовало больших затрат ресурсов, которыми организаторы работ в то время не обладали.

Курс и система «IDEA» были продемонстрированы на выставке
CeBIT-93 (Ганновер, Германия) и получили хорошие отзывы специалистов. По результатам маркетинговых исследований было решено в первую очередь разрабатывать инструментальные средства для расширения возможностей проектирования дизайна курсов, реализации дополнительных презентационных возможностей и т.д.

Часть, связанная с экспертной системой, оказалась на тот момент невостребованной рынком образовательных услуг, финансирование этих разработок фирмой Link & Link GmbH было прекращено, однако в 1993—1995 гг. продолжались научные исследования в этой области, финансируемые программой «INTAS» (грант INTAS 94-0135). Эти исследования продолжаются и сейчас при поддержке РФФИ, федеральной целевой программы «Интеграция» и т. д. В то же время фирмой Link & Link GmbH с 1993 по 2001 г. было выпущено на рынок три различных версии «IDEA»; в настоящее время продается версия IDEA 4.0 Professional.

Проект «IDEA» возник в 1990 г. в результате научного сотрудничества сначала лаборатории проблем теоретической кибернетики, а затем и кафедры математической теории интеллектуальных систем механико-математического факультета МГУ им. М. В. Ломоносова (инициатор и руководитель работ—заведующий кафедрой академик, доктор физико-математических наук профессор В. Б. Кудрявцев) и Института русской и советской культуры Рурского университета (Бохум, Германия) (руководитель работ – заместитель директора института доктор К. Вашик) [5].

Проект был одним из первых российско-германских научных проектов в этой области, давший мощный толчок для исследований по проблемам компьютерных систем обучения на высоком междисциплинарном уровне. Позднее к проекту подключилась немецкая фирма Link & Link GmbH, которая и финансировала разработку. Первая версия инструментальной систем для разработки интеллектуальных обучающих систем появилась примерно в середине 1992 г. Она включала в себя авторскую систему (средства разработки собственно курсов), систему ученика (средства отображения на экране учебного материала) и средства для создания экспертной системы. Был разработан демонстрационный курс «Итальянский язык для немцев-туристов» с простыми демонстрациями возможностей экспертной обучающей системы [6].

В настоящее время существуют два основных направления в области встраивания знаний в к гипер-текст (ГТ): извлечение знаний из документов, уже введенных в систему; введение знаний в процессе построения самой системы (from scratch), В рамках первого направления существует целый спектр подходов, начиная от автоматизированного построения ГТ из линейного текста с помощью методов семантической индексации и заканчивая построением новых связей, как в процессе навигации, производимой пользователем, так и в зависимости от предыдущих действий пользователя, его конечных целей, в зависимости от контекста и условий его вызвавших и т.п.

Что касается второго направления, то здесь большое распространение получили экспертекстовые системы, использующие отдельные методы и процедуры ЭС для управления навигацией в ГТ, например, известные коммерческие системы Knowledge Pro, INTERNIST, TIES, Oxford System of Medicine и отечественная КРЕДО. Большинство из этих систем значительно ускоряют доступ к информации и увеличивают возможности манипулирования ею, однако не осуществляют настоящего логического вывода, т.к. приобретенные и используемые в них знания не формализованы, (исключение здесь составляют системы типа SATELIT, в которой ввод знаний в ГТ-систему осуществляется в виде формализма *концептуальных графов* Sowa) [5].

Другой аспект интеграционных процессов в искусственном интеллекте (ИИ) связан с технологией извлечения знаний из естественных языковых (ЕЯ) текстов, т.к. последовавший после возникновения Интернет информационный взрыв разнородной по содержанию и по форме поставляемой в Интернет информации, стимулировал интенсивные исследования в области трансформации пространства Web в *пространство знаний*. Поэтому одно из ведущих направлений в этом процессе принадлежит методам обработки ЕЯ, т.к. основная часть информации на Web представлена в виде ЕЯ-текстов, что является основанием для перехода к использованию методов и систем обработки ЕЯ в среде Интернет, в частности систем обработки *связных ЕЯ-текстов,* интерес к которым обусловлен значительным объемом ЕЯ-информации, циркулирующей в Интернет и Интранет.

В данной реализации инструментальной системы не предусмотрена возможность добавления в формальную XML-структуру новых концептов и связей между ними. Это позволило создать набор компонентов, каждый из которых предназначен объектно-ориентированного представления соответствующего концепта, регламентированного учебным планом. Такая реализация позволяет пользователю оперировать привычными понятиями «Учебный курс», «Специальность», «Учебная тема», «Учебный объект». Связи между концептами в формальной XML-структуре реализованы соответствующими атрибутами классов, хранящих множество ссылок на связанные объекты.

Большинство существующих экспертных систем используют в качестве базовых языков Prolog и Lisp. Такой подход мотивируется удобством использования данных языков в задачах искусственного интеллекта. На наш взгляд, такой выбор имеет ряд недостатков. Во-первых, нестандартность семантики этих языков требует специальной подготовки инженеров знаний. Например, на языке Prolog выполнение программы это не последовательное исполнение команд, а вывод некоторой переменной исходя из начальных значений и правил, а в языке Lisp используется обратная запись для представления арифметических выражений. Соответственно, в этом случае, инженеры знаний должны иметь специальную подготовку для создания экспертных систем (ЭС) на базовых языках такого рода. Другой недостаток данного подхода, это плохая приспособленность этих языков к задачам разработки графического интерфейса. Удобный, интуитивно понятный графический интерфейс – это важная составляющая качественной экспертной системы, т.к. ЭС, прежде всего, ориентированы на обычных пользователей, без навыков программирования. Язык представления знаний С# Expert основан на языке C#. Такое решение имеет ряд преимуществ. Прежде всего, при разработке ЭС на С# Expert’е может быть применен объектно-ориентированный подход привычный большинству программистов (инженеров знаний), более того, C# популярный язык, на котором легко могут работать также специалисты использующие C++ или Java. Все это делает возможным использование C# Expert широким кругом специалистов (инженеров знаний). C# – это язык платформы .NET. Таким образом, еще одно значительное преимущество выбранного подхода, это возможность разрабатывать ЭС под .NET. На сегодняшний день, .NET – это наиболее современная и перспективная платформа для коммерческих приложений.

Существует множество уже готовых решений под .NET. Следует
также отметить, что .NET обеспечивает хорошую межъязыковую совместимость и широкие возможности использования программных компонент (Assembly, COM, DLL).

Основная сложность при создании экспертных систем это представление знаний экспертов в базе знаний наиболее подходящим образом для решения задач в заданной области. Для обеспечения такой возможности, базовый язык ЭС должен иметь четкий, хорошо структурированный способ представления данных и знаний.

К примеру, хороший способ представления знаний обеспечивает ЭС GURU разработанная фирмой Micro Data Base Systems, США. Эта система ориентирована на разработку ЭС в области деловых расчетов. К полезным возможностям GURU следует отнести возможность описания массивов как элементов данных, поддержка работы с таблицами и базами данных. Более того, в отличие от многих других ЭС, GURU предоставляет интегрированный подход к обработке данных, позволяя совместное использование наборов правил (продукций) с таблицами и реляционными базами данных. В ЭС GURU реализована гибкая подсистема логического вывода, позволяющая осуществлять как прямой, так и обратный вывод на одних и тех же наборах правил, имеющая способ разрешения конфликтов правил (conflict resolution) с помощью указания их приоритетов и порядка выполнения. Также GURU позволяет работать с нечеткими знаниями и использовать нечеткий логический вывод на основе коэффициентов уверенности, поддерживается возможность использования различных формул для вычисления коэффициента уверенности. Тем не менее, в системе GURU содержится и ряд ограничений, к примеру, отсутствуют средства представления сложных структурированных объектов и понятий сложной структуры данных, не предусмотрена возможность описания процедурных знаний. По-видимому, эти ограничения объясняются спецификой области применения: деловые расчеты, а также, возможно, устарелостью системы в целом.

В качестве, примера другой удачной экспертной системы, можно упомянуть ЭС KEE (Knowledge Engineering Environment). Это фреймовая экспертная система, где основным элементом данных базы знаний является юнит (фрейм). Юниты состоят из слотов, а слоты в свою очередь могут содержать данные простых типов (число, строка и т.п.), таблицы, графику, указатели на другие юниты или процедурные знания, написанные на языке Lisp. В системе KEE также реализован механизм наследования, который позволяет организовывать юниты в иерархические структуры, обеспечивая логически связанное представление информации в базе знаний. Безусловно, фреймовая структура данных, реализованная в системе KEE, обеспечивает более широкие возможности представления данных, чем структуры данных ЭС GURU. Основным недостатком ЭС KEE является использование языка Lisp в качестве базового языка системы, и как следствие сложная семантика базового языка с достаточно нетрадиционной формой записи для большинства инженеров знаний [7].

При генерации результирующего кода слоты фрейма представляются в виде объектов класса Slot и содержаться в коллекции CSharpExpertAbstract.slots. Общая схема работы конвертора показана на рисунке ниже (рис. 2).

Точка входа в C# Expert конвертор находиться в методе ExpComp.Main(string[] arg); Метод получает имя исходного файла и запускает метод Parser.Parse() который осуществляет синтаксический анализ и генерацию промежуточного кода. После этого вызывается метод OutputTextGenerator.generateOutputProgram (StreamWriter s) который проводит семантический анализ промежуточного кода, замену идентификаторов доступа к слотам и генерацию результирующей программы на C#.

Рис. 2. Общая схема компиляции проекта на языке C# Expert

Приведенные технологии позволяет описывать и строить экспертные системы самообучения общего вида на основе математических графов с использованием современных технологий программирования. Описанные выше проблемы решены за счет использования принципов декомпозиции исходного образовательного ресурса. В результате чего инженер знаний имеет дело с обособленными (имеющими конечный смысл в контексте самообучения) понятиями предметной области учебного курса. Интеграция базы знаний экспертной системы с внешними источниками реализована за счет использования промежуточных программных интерфейсов поэтапного преобразования информации сначала в виде, понятном ядру экспертной системы, а затем в виде, понятном конечному пользователю.

Положительным при использовании подобных информационных технологий (типа экспертных систем) в образовании является повышение качества обучения за счет:

* большей адаптации обучаемого к учебному материалу с учетом собственных возможностей и способностей;
* возможности выбора более подходящего для обучаемого метода усвоения предмета;
* регулирования интенсивности обучения на различных этапах учебного процесса;
* самоконтроля;
* доступа к ранее недосягаемым образовательным ресурсам российского и мирового уровня;
* поддержки активных методов обучения;
* образной наглядной формы представления изучаемого материала;
* модульного принципа построения, позволяющего тиражировать отдельные составные части информационной технологии;
* развития самостоятельного обучения.

Итак, с учетом вышеизложенного определим основные задачи, решаемые экспертными системами в процессе самообучения:

* управление процессом обучения с учетом индивидуальной подготовленности обучаемого, его индивидуальных особенностей;
* диагностика и прогнозирование качества усвоения предметной информации и формирование изменений в последовательности представления учебного материала;
* поддержание профессионального уровня обучаемого в данной предметной области.

*Литература*

1. Гребнев А.Н. AtLeap – Java каркас с открытым исходным кодом для Web-приложений // Вестник ИжГТУ. 2006. №1. С. 64-68.
2. Интеграция методов инженерии знаний и инженерии программ: система управления знаниями Knowledge.NET / А.В. Новиков, В.О. Сафонов, М.В. Сигалин, А.Л. Смоляков // Компьютерные инструменты в образовании. 2005. № 5.
3. Применение Web-онтологий в задачах дистанционного образования. [Электронный ресурс] // Блог PhD Щербака Сергея: [сайт].
URL: <http://shcherbak.net/>
4. Сафонов В.О. Экспертные системы – интеллектуальные помощники специалистов. СПб.: Санкт-Петербургская организация общества «Знания» России, 2002. C. 196.
5. Управление знаниями: [сайт]. URL: https://sites.google.com/ site/upravlenieznaniami/inzeneria-znanij
6. Knowledge.NET ontology-based knowledge management toolkit for Microsoft.NET / V. Safonov, A. Novikov, A. Smolyakov, D. Cherepanov, M. Sigalin [Электронный ресурс] // Conference «.NET Technologies 2006»: [сайт]. URL: http://wscg.zcu.cz/Rotor/NET\_2006/Papers\_2006/poster/A89-full.pdf
7. Web-страницы проекта Knowledge.NET: [сайт].
URL: http://www.knowledge-net.ru

[Российский портал информатизации образования](http://rpio.ru/) [содержит: законодательные и нормативные правовые акты государственного регулирования информатизации образования, федеральные и региональные программы информатизации сферы образования, понятийный аппарат информатизации образования, библиографию по проблемам информатизации образования, по учебникам дисциплин цикла Информатика, научно-популярные, документальные видео материалы и фильмы, периодические издания по информатизации образования и многое другое.](http://rpio.ru/)

