

TeachExpert: СРЕДА ВИЗУАЛЬНОЙ РАЗРАБОТКИ ОБУЧАЮЩИХ СИСТЕМ

Проанализированы отечественные реализации инструментальных средств для разработки интеллектуальных обучающих систем и систем тестирования знаний обучаемых. Описаны инструментальные средства визуальной разработки TeachExpert, предназначенные для создания интеллектуальных обучающих систем.

Интеллектуализация является одним из современных направлений развития технологии компьютерных обучающих систем. Практическую реализацию в настоящее время активно получают два подхода: экспертно-обучающие (ЭОС) и гипертекстовые системы (ГТС). Роль контрольно-оценочного этапа процесса обучения определяет тот устойчивый интерес, который проявляется к разработке компьютерных систем тестирования знаний обучаемых. Проанализируем тенденции отечественных реализаций инструментальных средств для разработки интеллектуальных систем и систем тестирования знаний обучаемых.

Системы тестирования знаний. В инструментарии подготовки систем контроля знаний используют различные модели тестовых заданий (закрытые, на соответствие, на ранжирование, открытые, ситуационные [4]; с одиночным выбором ответа из множества возможных [5]). Авторы контролирующей системы имеют возможность создавать собственный сценарий теста [2] или автоматизировать этот процесс с помощью генератора [3]. Создание программы-теста выполняется с использованием различных технологий: текстовый файл исходных вопросов + компилятор вопросов + утилиты для объединения файлов вопросов в модули [2], тест «наполняется» самим автором [5].

Инструментарий разработки интеллектуальных систем. Прослеживается тенденция совмещения технологий экспертных систем с гипертекстом [1, 6] и мультимедиа [1, 3]. С помощью такого инструментария создаются обучающие системы различных классов: проблемно-ориентированные экспертные системы для обучения [1], электронные справочники [6], диалоговые обучающие системы [3]. Отмечается переход в среду Windows и использование коммерческих универсальных (но не специализированных) интегрированных сред для разработки Windows-приложений (Borland C++ 4.5 [1] и Delphi [3]).

На кафедре Прикладной математики и информатики Донецкого Государственного Технического Университета разработаны инструментальные средства TeachExpert для визуального создания экспертно-обучающих систем. Данный инструментарий использует технологию визуального проектирования приложений, что не требует от преподавателей-предметников высокого уровня программистской подготовки.

База Знаний системы TeachExpert

При проектировании экспертно-обучающей системы линейные знания конкретной предметной области (ПО) представляются в виде взвешенного графа. Отдельные, связанные логически, разделы ПО выделяются в главы, называемые листьями. Листья являются элементарным, минимальным источником информации о ПО, несут основную информационную нагрузку системы. Они объединяются в более общие разделы, называемые узлами. Связи имеют вес, определяющий важность главы в узле и учитывающийся при оценке знаний обучаемого. Узлы определяют маршрут обхода информационного дерева при освоении ПО.

Листья имеют сложную структуру, которую можно описать двойкой:

$$L_i = \{I, Q\},$$

где I - содержит информацию ПО, Q - информацию, необходимую для тестирования по данной главе.

Множество I определим следующим образом: $I = \{T_i, G_i, A_i, S_i, E_i, Gl_i\}$, где

- T_i - собственно информация, представленная в текстовом виде;
- G_i - графическая информация (список графических изображений);
- A_i - аудио-информация (список музыкальных, речевых, шумовых эффектов);
- S_i - список ссылок на другие главы ПО (т.е. другие листья информационного дерева: $S_i = \{L_j | j \in [1; N], j \neq i\}$, N - общее число листьев-глав);
- E_i - список ссылок на главы особого типа, главы-примеры, составляющие отдельную характеристику ПО (множество глав-примеров составляют решатель задач ПО, который углубляет знания обучаемого путем демонстрации методов решения задач ПО);
- Gl_i - список ссылок на слова из глоссария.

Глоссарий включает набор специальных слов-терминов ПО, т.о. $Gl_i = \{W_k\}$, где $W = \{W_n\}$ - глоссарий, представляющий собой множество слов с установлением каждому из них в соответствие некоторого объема информации, определяющего лексическое или семантическое значения слова-термина. Глоссарий позволяет углубить знания ПО путем уточнения терминологии.

Множество Q есть: $Q = \{Quest_i\}$, где **Quest_i** - список вопросов, позволяющих протестировать полученные знания и навыки при изучении информации данного листа.

Между информационной и тестирующей частью на этапе проектирования устанавливается взаимно однозначное соответствие. Каждый вопрос представлен объектом со сложной структурой, описываемой множеством:

Quest = {**Text, Time, Type, Rep**}, где

- **Text** - текст вопроса, составляющий его смысловую нагрузку
- **Time** - время в сек., отведенной на обдумывание данного вопроса.
- **Type** - тип вопроса; который принимает одно из трех возможных значений: YesNo - односложный ответа типа "Да/Нет", Select - выбор ответа из множества вариантов, MultiSelect - множественный выбор, т.е. возможность выбора нескольких ответов из множества предлагаемых.
- **Rep** - вариант ответа на вопрос, который задается парой: **Rep** = {**Answer, Ball**}, где: **Answer** - текст варианта ответа; **Ball** - балл, получаемый за выбор этого варианта. Т.о. за каждый ответ обучаемый получает определенный балл, что расширяет понятие ответа на вопрос от простого "верный ответ - неверный ответ" до более гибкой, нечеткой модели тестирования с принципом "более верный ответ - менее верный ответ".

Итак, если глава L_i содержит n вопросов и для каждого вопроса есть x_n вариантов ответов с весом каждого $v(x_n)$, то при выборе обучаемым некоторого варианта ответа, он получает за этот ответ балл, соответствующий весу выбранного варианта. Тогда общий вид функции $T(L)$ приобретает вид:

$$T(L_i) = \sum_n v(x_n)$$

Для получения оценки поддерева (узла), содержащего m листьев или уже имеющих оценку узлов, применяется модель строгой оценки с учетом веса каждого подузла:

$$T(U_j) = \min(T_m * a_m)$$

При последовательном изучении темы каждого листа с последующим тестированием получается иерархическое дерево оценок, представляющее модель знаний обучаемого о данной ПО. Дерево динамическое, т.к. система реагирует на недостаточное освоение отдельной главы рекомендацией повторения материала и нового тестирования с обновлением модели обучаемого.

Структура системы TeacherExpert.

Структурная схема системы состоит из трех базовых модулей: информационный модуль, тестирующий модуль, модуль анализа.

Информационный модуль позволяет воспроизводить текстовую, аудио и графическую информацию. Модуль тестирования предназначен для воспроизведения

тестов и приема ответов у обучаемого. Модуль анализа, используя логический вывод, определяет реакцию системы на знания обучаемого, строит модель, выбирает стратегию дальнейшего обучения, наиболее адекватную индивидуальным особенностям обучаемого.

Внутренними информационными потоками системы являются:

- Поток "*Информация - Тесты*" показывает запрос от информационной части на тестирование отдельной главы. При решении пользователя протестировать свои знания по отдельной главе, он может перейти от окна информационной части к окну тестирования, где ему будет предложен ряд вопросов по изученной главе.

- Поток "*Тесты - Анализ*" показывает, что результаты тестирования отправляются на анализ. После тестирования результаты опроса используются для построения модели обучаемого, которая в свою очередь используется на этапе анализа и выбора стратегии обучения.

- Поток "*Анализ - Информация*" определяет принятые решения о стратегии дальнейшего обучения. При этом система самостоятельно определяет общий уровень знаний обучаемого по данной ПО и выявляет слабые места обучаемого с выдачей рекомендации о повторении отдельных глав или целых разделов. Предусмотрена возможность поэтапного обучения, при котором обучаемый не допускается к следующему разделу без усвоения информации предыдущего, информация которого необходима для полного понимания данного, и обучения в произвольном порядке, при котором обучаемый рассматривает предложенные темы в выбранном им порядке, но не получает зачет по общей теме ПО без полного тестирования по всем ее составляющим.

Таким образом, среда визуальной разработки TeachExpert генерирует ЭОС, учитывающую индивидуальную методику автора и основанную на модели обучаемого.

1. Балик Н.Р., Иваськов И.С. Инструментальная экспертная система. Всеукраїнська конференція молодих науковців "Інформаційні технології у науці та освіті" - Черкаси, 1997. с.70.
2. Жульковский О.А., Изюмовский Д.В. Система тестового контроля знаний. Всеукраїнська конференція молодих науковців "Інформаційні технології у науці та освіті" - Черкаси, 1997. с.48.
3. Конишев С.М. "Глобус" - генератор диалоговых обучающих систем. Нові технології навчання. Вип.14. - К.:ІСД МО, 1995. с.129-130.
4. Цибанев В.И., Фоменко С.Б. Система "Контроль". Проблемы вищої школи. Вип. 80. - К.: Вища школа, 1994. - с.80-83.
5. Цурин О.Ф., Кравцова О.О. Инструментальные способы создания систем контроля. Нові технології навчання. Вип. 8. - К.: НМК ВО, 1992. с.43-44.
6. Шевляков А.С., Федоров В.К. Инструментальная система для разработки электронных справочников. Всеукраїнська конференція молодих науковців "Інформаційні технології у науці та освіті" - Черкаси, 1997. с.71.