

На правах рукописи



**Редькина Александра Васильевна**

**ПРОГРАММНЫЕ СРЕДСТВА ПОСТРОЕНИЯ СИСТЕМ  
ПОДДЕРЖКИ ОБУЧЕНИЯ РЕШЕНИЮ ЗАДАЧ**

05.13.11 – Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей

**АВТОРЕФЕРАТ**  
диссертации на соискание ученой степени  
кандидата технических наук

Красноярск 2010

Работа выполнена в Федеральном государственном образовательном учреждении высшего профессионального образования «Сибирский федеральный университет», г. Красноярск

- Научный руководитель: доктор технических наук,  
старший научный сотрудник  
Карпов Леонид Евгеньевич
- Официальные оппоненты: доктор технических наук,  
профессор  
Доррер Георгий Алексеевич
- кандидат физико-математических наук,  
доцент  
Волкова Ирина Анатольевна
- Ведущая организация: ГОУ ВПО «Сибирский государственный  
аэрокосмический университет имени  
академика М.Ф. Решетнева»

Защита состоится 18 июня 2010 года в 16.00 часов на заседании диссертационного совета ДМ 212.099.05 при ФГОУ ВПО «Сибирский федеральный университет» по адресу:

660074, г. Красноярск, ул. акад. Киренского, 26, УЛК, ауд. 1-15

С диссертацией можно ознакомиться в библиотеке Сибирского федерального университета по адресу: г. Красноярск, ул. ак. Киренского, 26, ауд. Г 274.

Автореферат разослан 18 мая 2010 г.

Ученый секретарь диссертационного совета



Непомнящий О.В.

## ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

**Актуальность проблемы.** В настоящее время процессы информатизации в обществе базируются на идее интеграции существующих и вновь разрабатываемых информационных систем, обеспечении массового доступа к информационным ресурсам. Информационные обучающие системы – одно из направлений внедрения и непосредственного применения новых информационных технологий.

Исследования в области компьютерной поддержки учебного процесса имеют более чем 40-летнюю историю. В последние годы интенсивно развиваются технологии обучения с использованием возможностей глобальной сети. Основными функциями систем поддержки обучения являются публикация и демонстрация учебного материала, а также тестирование, ограничивающее реакцию пользователя рамками заранее подготовленных для выбора альтернатив, либо конструируемых из них ответов. В настоящее время создано множество программных систем для организации поддержки обучения, их количество стремительно растет, но значительно влияния на традиционный процесс обучения они не оказывают. В них отсутствует поддержка формирования таких базовых навыков, как решение задач, являющихся в системе инженерного образования наиболее значимой и результативной основой естественно-научной, общей профессиональной и специальной подготовки. Большинство форм проверки приобретенных навыков отнимают у преподавателя много времени, и до сих пор они не автоматизированы даже частично. Публикация в единой информационно-образовательной среде теоретического материала и размещение поддержки в решении задач представляется одной из важных целей разработки обучающих систем. Разгрузка преподавателя от рутинной работы по проверке задач, интенсификация процесса обучения за счет быстрого получения ответа, подтверждающего правильность или ошибочность решения, являются актуальными задачами.

Основные трудности на пути разработки программ поддержки в обучении решению задач связаны с тем, что не существует единого алгоритма решения задач по различным дисциплинам, поэтому реализация такой поддержки обучения определяется методами решения задач заданной предметной области. Представляется актуальным исследование и разработка подходов к построению инструментальных средств, обеспечивающих поддержку разработчика систем обучения решению типовых задач, и интеграции этих средств в современные системы, включающие стандартную функциональность. Использование функциональности существующих систем обеспечит единую среду для обучения, позволит упростить разработку и снизить затраты за счет части работ по созданию приложений.

Возрастающая сложность создаваемых программных комплексов требует использования технологий, позволяющих объединять разнородные

системы, выстраивать связи и обмен оперативной информацией между программными компонентами. Построение систем обучения с гибкой структурой, подстраивающейся под предметную область, лежит в области развития технологий, связанных с появлением сервисно–ориентированных архитектур. Эта концепция использует модульный подход к разработке программного обеспечения, строгое разделения интерфейса и реализации, необходимое для постепенного наращивания функциональности. Практическое использование данных подходов требует удобных средств инструментальной поддержки. На сегодняшний день в области поддержки образования еще не предлагается достаточно гибких и пригодных для всех стандартных решений, поэтому задача их разработки и апробации является актуальной.

**Целью диссертационной работы** является разработка и исследование методов построения программных инструментальных средств создания компьютерных обучающих систем, обеспечивающих расширение функциональных возможностей современных средств поддержки образовательного процесса и ускорение сроков их разработки.

Для достижения указанной цели решаются **следующие задачи**:

1 Анализ основных характеристик современных инструментальных средств поддержки прикладных программных систем, в частности, средств поддержки систем обучения.

2 Разработка принципов построения программного инструментария для создания систем поддержки обучения в решении задач путем выявления характерных и специфических конфигураций, задаваемых предметной областью.

3 Разработка функциональной модели и базовой архитектуры инструментальных средств построения систем поддержки обучения решению задач на основе открытых систем.

4 Проведение экспериментальных исследований предложенных моделей, методов и решений для подтверждения работоспособности и эффективности заложенных подходов путем исследования свойств системы обучения решению задач по программированию, построенной с применением разработанных инструментов.

**Методы исследования.** В работе использованы понятия и методы теории графов, теории алгоритмов, формальных грамматик, теории языков программирования, теории систем, методы структурного анализа и проектирования.

**Научная новизна и положения, выносимые на защиту:**

1 Предложен подход к построению систем поддержки процесса обучения, основанный на разделении функциональности, порождающий различные конфигурационные аспекты программного комплекса, обеспечивающий эволюцию его функций.

2 Разработана модель инструментальных средств для построения систем поддержки обучения решению задач, описывающая базовые и проблемно-ориентированные составляющие реализуемой функциональности, расширяющей интеллектуальные возможности обучающих систем.

3 Предложена стратегия использования инструментальных средств поддержки в обучении решению задач, инвариантная к изучаемой дисциплине, отличающаяся свойством динамической адаптации к уровню знаний обучаемого.

4 Разработана архитектура инструментальных средств построения систем поддержки обучения решению задач на основе открытых систем.

### **Практическая ценность работы**

1 Проведенные автором исследования позволили создать инструментальную среду «Algorithm», реализующую проблемно-ориентированную часть архитектуры инструментальных средств, которая связана со специфическими особенностями дисциплины, и способствующую формированию практических навыков решения задач по основам программирования. Эффективность среды «Algorithm» подтверждена в процессе практической эксплуатации при обучении студентов политехнического института Сибирского Федерального университета.

2 Реализованы визуализаторы «ListView» и «GraphView», которые интегрированы в систему управления обучением для исследования возможностей их взаимодействия. Визуализаторы позволяют в интерактивном режиме продемонстрировать алгоритмы обработки динамических списков и поиска путей в графах, что способствует осмыслению и закреплению теоретического материала.

3 Визуализаторы «ListView» и «GraphView» внедрены в учебный процесс в Сибирском федеральном университете, интегрированы в ИОС Moodle в курсе «Программирование на языке высокого уровня».

Применение разработанных в диссертации программно-инструментальных средств позволяет с меньшими затратами времени создавать семейства обучающих систем различной предметной ориентации.

**Достоверность** научных положений, выводов и практических рекомендаций, полученных в диссертации, подтверждается экспериментальными исследованиями и сравнительным анализом существующих подходов к решению поставленной задачи, точной формулировкой критериев, компьютерным моделированием, а также результатами использования методов и средств, предложенных в диссертации, на практике.

**Апробация диссертации.** Основные положения диссертационной работы представлялись на XII международной конференции «Современное образование: содержание, технологии, качество» (С-Петербург, 2006), Всероссийской научно-практической конференции «Повышение качества непрерывного профессионального образования» (Красноярск, 2005), на научно-практической конференции «Совершенствование качества подготов-

ки специалистов» (Красноярск, 2002), на Всероссийской научно-практической конференции «Достижения науки и техники – развитию сибирских регионов» (Красноярск, 2003), на Всероссийской научной конференции «Наука и образование» (Белово, 2003), на научно-практической конференции «Управление качеством в новых информационных технологиях, системах и образовании» (Москва, 2003), на семинаре в институте вычислительного моделирования СО РАН.

**Сведения о внедрении.** Визуализаторы «GraphView» и «ListView» внедрены в учебный процесс, о чем есть соответствующие акты.

**Публикации.** По теме диссертации опубликовано 12 работ, в том числе, четыре статьи, из которых 2 статьи в изданиях по списку ВАК, 7 работ в материалах всероссийских и международных научно-практических конференций, одно свидетельство о регистрации программ для ЭВМ.

**Личный вклад автора.** Автору принадлежат идеи работы, определение цели и постановка задачи, модели и архитектура инструментальных средств поддержки обучения решению задач. Основные результаты, изложенные в работе, получены либо непосредственно автором, либо с его участием.

**Структура диссертации.** Диссертация содержит 149 страниц и состоит из введения, четырех глав, заключения и трех приложений. В списке использованных источников указано 101 наименование. Работа иллюстрирована 31 рисунком и 4 таблицами.

### **Содержание работы**

**Во введении** обоснована актуальность темы диссертационной работы, сформулированы цель и задачи работы, научная новизна, практическая значимость, перечислены основные результаты работы. Также приводится краткое содержание работы.

**В первой главе** дается обзор существующих инструментальных средств поддержки обучения. Показаны характерные особенности и свойства современных систем поддержки обучения, рассмотрены современные тенденции развития информационных систем и подходы к построению инструментальных средств разработчика систем обучения. Проанализированы принципы построения распределенных систем, рассмотрены проблемы взаимодействия и интеграции прикладных систем. Основное внимание уделено сервисно-ориентированной архитектуре, эта концепция обеспечивает модульный подход к разработке распределенного программного обеспечения, основанный на использовании сервисов со стандартизированными интерфейсами.

Анализ современного состояния систем поддержки обучения показал, что основная тенденция их развития направлена на создание систем

управления обучением. Они поддерживают средства публикации теоретических сведений в единой образовательной среде и контроля полученных знаний путем тестирования, но в них отсутствует функциональность, требующая организации учебной деятельности, основанной на специфических особенностях представляемых предметных областей. Это приводит к упрощению процесса поддержки обучения, сводя его к обучению декларативной части знания, что позволяет использовать типовые программные компоненты, инвариантные к изучаемой дисциплине. Существующие многочисленные специализированные инструменты поддержки обучения решению задач являются разрозненными и отделены от систем управления обучением. Предлагаемые инструментальные средства повышают активность пользователя, следовательно, возникает необходимость поиска новых типов взаимодействия с пользователем, что ведет к усложнению процесса анализа этого взаимодействия.

Возрастающая сложность систем, недостаточная разработанность вопросов, связанных с созданием инструментальных средств поддержки в обучении решению задач, требует исследования подходов к моделированию и проектированию типовых систем такой поддержки. Для разработки оптимальной структуры системы, позволяющей реализовать совокупность функций с определенным запасом, проанализированы подходы к построению систем обучения. С точки зрения функционально-структурного подхода при построении систем исходят из их поведения, определяющего назначение системы, отвлекаясь от их внутреннего содержания, которое может быть разнородно по составу. Объектно-ориентированный подход рассматривает систему как совокупность взаимосвязанных объектов, между которыми происходит информационный обмен, чтобы обеспечить поведение системы, соответствующее назначению системы. Таким образом, функционально-структурный подход является предпочтительным при создании развивающихся систем, представителями которых являются системы поддержки обучения. Объектно-ориентированный подход не противоречит принципам функционально-структурного подхода, поэтому на первоначальных этапах формируется совокупность реализуемых функций, а на этапах разработки деталей системы используется объектно-ориентированный подход.

Следование принципам распределенных систем и стандартам открытости снимает наиболее острые вопросы интеграции приложений, облегчает решение проблем взаимодействия прикладных программ, обеспечивает эволюционную разработку систем. Анализ способов взаимодействия приложений в распределенных системах показал, что применение сервисно-ориентированных технологий способствует быстрой реорганизации и адаптации систем, позволяет повысить эффективность процесса разработки приложений за счет применения готовых программных компонентов, обеспечивает эволюционное развитие информационных систем. Реализа-

ция систем поддержки обучения с гибкой структурой, учитывающих специфические особенности предметной области, представляется возможной на основе технологий распределенной обработки информации.

Во **второй главе** изложены предлагаемые автором подходы к решению поставленных задач. Проведено исследование поведения разрабатываемой системы с точки зрения реализуемых функций, на основе которого построено дерево функций. Дерево функций инструментальной поддержки в обучении позволило разделить функциональность между существующими и разрабатываемыми системами поддержки обучения и определить, какие дополнительные функциональные возможности необходимо реализовать для поддержки в обучении решению задач. На базе дерева функций системы осуществляется формирование структуры системы.

Анализ предметной области показал влияние специфических особенностей изучаемых дисциплин на конфигурацию программного инструментария для реализации систем обучения решению задач, выявлены базовые и проблемно-ориентированные компоненты инструментальных средств, их взаимосвязь и логическая совместимость. При построении инструментальных средств процесс поддержки обучения рассматривается как управляемый и контролируемый процесс решения задач, который можно представить его как тройку:

$$M = \langle \text{УП, МО, Р} \rangle,$$

где УП – структура управления процессом поддержки обучения, МО – модель обучаемого и Р – ресурсы, которыми располагает среда для обучения.

Структура управления определяет процесс взаимодействия системы с объектом управления, цель которого перевод объекта управления в состояние обученности, соответствующее заданному уровню знаний, который оценивается по умению решать задачи определенной степени сложности. Обучение включает в себя два взаимосвязанных процесса: организацию деятельности обучаемого и контроль этой деятельности. Организация определенной деятельности требует конкретного способа регистрации этой деятельности и адекватной формы контроля: в зависимости от полученных результатов формируется корректирующее воздействие. Для получения информации об объекте управления синтезируется модель обучаемого, позволяющая измерять параметры реального обучаемого. Модель обучаемого задается набором характеристик, измеряемых во время работы системы с обучаемым, и методами их обработки. Существующие модели обучаемого базируются только на уровне декларативных знаний, контролируемых с помощью тестов. В данной работе модель обучаемого опирается на способности обучаемого решать задачи определенной сложности и представляется набором шаблонов (рисунок 1), которые выстроены в соответствии с этапами познавательной деятельности теории познания (восприятие-осмысление-овладение-исследование).



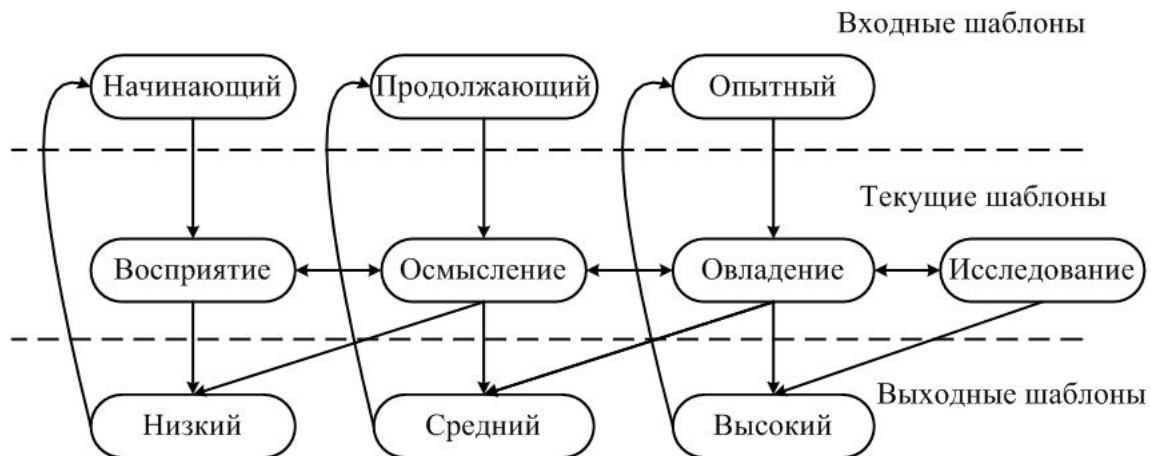


Рисунок 1 – Модель обучаемого

При первом входе в систему идентификация уровня знаний ведется на основании собственной оценки обучаемого. Для идентификации по текущим шаблонам разработана шкала оценивания, учитывающая степень сложности решаемых задач и сущность совершаемых ошибок, а также, что осознание общности в построении решений происходит в среднем при столкновении с третьей задачей. Каждая задача характеризуется шестеркой вида:

$$X = \langle t, o, s, R, P, d \rangle,$$

где  $t$  – показатель самостоятельности в решении задачи, определяемый уровнем подсказок (от частичного до полного решения);

$o$  – уровень сложности задачи;

$s$  – рубрика, характеризующая соответствующее уникальное подмножество стратегий решения;

$R$  – множество тестовых наборов данных, подготовленных для анализа решения задачи;

$P$  – множество подсказок и комментариев для данной задачи;

$d$  – показатель оптимальности, отражающий рациональность решения задачи обучаемым относительно решения эксперта.

Выходные шаблоны базируются на текущих шаблонах и определяются с использованием элементов нечеткой логики на основании степени принадлежности к тому или иному текущему шаблону. В конечном счете, модель обучаемого позволяет не задавать заранее определенную траекторию обучения, а выстраивать ее в зависимости от сложившейся ситуации. Предлагаемая модель обучаемого является динамической, отражает уровень знаний, служит основой для организации управления и инвариантна к предметной области.

Эффективная поддержка процесса обучения зависит не только от самого обучаемого и выделенных ресурсов, но и от стратегии обучения, которая, по сути, является алгоритмом управления обучением. Так как в качестве преподавателя выступают инструментальные средства, то рассмотрена стратегия их использования с точки зрения реализации механизма управления обучением, опирающегося на динамическую модель обучаемого, адаптирующегося к его уровню знаний. Предложено выстраивать алгоритм управления согласно психологически обоснованной последовательности этапов познавательной деятельности: восприятие-осмысление-овладение-исследование. Такой алгоритм обучения естественным образом коррелируется с текущими шаблонами модели обучаемого, позволяя избежать путаницы и сохранить взаимосвязь и согласованность в управлении.

Ресурсы – это информационные и вычислительные возможности системы, используемые для поддержки процесса обучения. Под ресурсами понимается не только имеющийся объем сведений, которыми располагает среда для обучения, но и функциональные возможности, учитывающие специфические особенности предметной области, в том числе и способы организации и контроля деятельности. В первую очередь – это способы представления решения задач и механизм проверки их корректности.

Под способом представления решения задач понимается, каким образом обучаемый сообщит системе свое решение, а это далеко не всегда разрешимая проблема. Для организации деятельности обучаемого система предоставляет библиотеку инструментов, реализующих элементарные операции, примитивы, из которых складывается решение. Пользователь с помощью библиотеки инструментов создает ответ-решение. Каждое действие пользователя регистрируется, чтобы в дальнейшем на основе полученного протокола, отфильтровав лишние действия, построить дерево решения. Проектирование набора инструментов на основе предметной области является одной из ключевых проблем, решение которой позволит пользователям при работе с программной системой не менять существенно свой стиль работы. Процесс проектирования сводится к описанию решения на некотором формальном языке, поддерживающим систему знаков, однозначно понимаемую как человеком, так и компьютером, следовательно, имеет свои синтаксические, семантические или прагматические аспекты. Введение формального языка приводит к необходимости однозначного перевода описания на одном языке в описание на другом и их эквивалентной интерпретации.

Не менее сложной является проблема анализа переданного решения. Способ представления решения во многом определяет и механизм проверки его корректности. Ресурсы системы, предоставляя инструменты для предъявления знаний, фиксируют действия пользователя, выполняют их первичную обработку и преобразуют в промежуточное представление, пригодное для дальнейшей обработки. Многообразие форм и способов

представления решений не позволяет найти универсальный подход, это ведет к необходимости ограничить предметные области и классы задач, для которых возможен анализ корректности решений. К таким представлениям можно относиться решения, сформулированные на конечном множестве объектов, а также где уже используется достаточно формализованное представление решения задач. Это – «Языки программирования», «Исчисление высказываний», некоторые разделы химии и алгебры и др. В качестве подобных ресурсов могут выступать тематические редакторы, включающие библиотеку инструментов, отражающих заданную предметную область. Информацию, поступающую в редактор надо зафиксировать, понять семантику, проанализировать и однозначно интерпретировать. На основе полученного представления строится механизм контроля корректности решений, опирающийся на специфические особенности предметной области. На данном этапе выявляются общие требования к тематическим редакторам с целью их дальнейшей интеграции в систему поддержки обучения решению задач, учитывая, что корректность решения задачи определяется не только по конечному результату, но и на основе анализа решения.

Рассмотрены следующие стратегии проверки корректности решений:

1 Сравнение уже имеющегося ответа с ответом пользователя. Этот способ не позволяет анализировать ход решения задачи.

2 Сравнение уже имеющегося описания решения с описанием решения пользователя. Так как может существовать несколько вариантов верных способов решения одной задачи, как и способов описания решений, то такая стратегия применима для узкого класса задач при ограниченном языке описания решений.

3 Сравнение уже имеющегося описания решения с описанием решения пользователя в контрольных точках. Данная стратегия предоставляет большую свободу, чем предыдущая, позволяя балансировать относительно заданного хода решения с обязательным прохождением контрольных точек.

4 Построение некоторого дерева решения и определение его эквивалентности с уже имеющимся деревом описания решения. Суть построения сводится к отслеживанию, перечислению и фильтрации примитивов, используемых пользователем в представлении ответа-решения, а также отражению связи между примитивами.

Дерево решений – граф, вершины и ребра которого представляют описание примитивов на основе регистрации действий пользователя. Количество видов вершин определяется набором инструментов тематического редактора. Ребра графа отражают ход решения задачи. Переход на следующую вершину графа определяется атрибутами, хранящимися в описании вершины. В качестве атрибутов, описывающих элементарные операции, используются как количественные признаки, так и поведенческие. Одним

из таких вариантов определения эквивалентности служит автоматическое построение ответа по имеющемуся дереву решения задачи с последующим сравнением его с ответом пользователя, построенного также автоматически по его дереву решения.

Выделены базовые и проблемно-ориентированные компоненты инструментальных средств поддержки процесса обучения решению задач. Инвариантной частью системы является подсистема управления, которая и является ядром системы. Она использует предложенную модель обучаемого для формирования реакции инструментальных средств на действия пользователя, и может быть использована для построения адаптивных обучающих систем по широкому спектру дисциплин. Вариативные компоненты определяются специфическими особенностями предметной области и требуют разработки коллекции тематических редакторов и анализаторов представленных решений задач. Согласно технологии открытых систем представляется перспективным хранение информации, полученной после первичной обработки редактором в формате XML, что способствует унификации интерфейсов интеграции с системой управления обучением.

**Третья глава** посвящена разработке функциональной модели и архитектуры инструментальных средств поддержки построения систем обучения решению задач, предлагается модель базы знаний и структура основных блоков системы.

Учитывая необходимость единой централизованной стратегии управления процессом поддержки обучения и децентрализованный характер оказания поддержки в решении задач, механизм интеграции существующих систем управления обучением и разрабатываемой инструментальной поддержки в решении задач становится неоспоримым преимуществом.

Существующие инструментальные системы управления обучением представляют поддержку в построении систем обучения декларативной части знания, организации процесса взаимодействия обучаемого и преподавателя, используя возможности локальных и глобальных вычислительных сетей. Построение системы поддержки обучения процедурной части знания возлагается на новую функциональность и реализуется в виде инструментальных средств поддержки в обучении решению задач, интегрированных в систему управления обучением. Таким образом, используется базовая функциональность системы управления обучением, обеспечивающая единый механизм регистрации и ведения статистики, единое информационное пространство и регламент подачи материала (рисунок 2), что является явным преимуществом при эксплуатации системы. Реализация подобной поддержки требует для каждой предметной области индивидуального подхода к разработке программного обеспечения, так как нет единого способа представления и контроля решений задач по различным дисциплинам. Фактически это приводит к необходимости расширить функ-

циональность обучающей системы всякий раз, как только возникает потребность в размещении новой дисциплины.



Рисунок 2 – Интеграция инструментальной поддержки в обучении решению задач в систему управления обучением

На основе разработанного дерева функций и стратегии поведения системы предложена архитектура инструментальных средств поддержки построения систем обучения решению задач. Строгая логическая структура разделяет систему на базовые и проблемно-ориентированные компоненты, перестраивающиеся под требования предметной области. Проблемно-ориентированные компоненты характеризуются схожими форматами хранения данных и основаны на однотипных технологических решениях. Для обмена информацией и хранения данных используется формат XML, гибко представляющий сложные структуры данных. Выполнение некоторых специфических компонент в виде сервисов позволяет разрабатывать и тестировать их независимо друг от друга, обеспечивает механизмы адаптации системы к вновь появляющимся задачам. Инструментальная поддержка является относительно автономной системой за счет взаимодействия слабосвязанных сервисов. Возможность отображения компонентов системы в модули системы сервис-ориентированной архитектуры создает предпосылки для включения новых функциональностей, внесения изменений в используемые сервисы (рисунок 3).

В качестве проблемно-ориентированных компонент выступают тематические редакторы и механизм проверки корректности решения. Унификация программных интерфейсов этих компонент обеспечит единую схему взаимодействия, упростит их разработку. Реестр сервисов – централизованный репозиторий, содержащий справочник всех сервисов, доступных в данном программном инструментарии. С учетом разделения функциональностей разработана функциональная модель инструментальной среды, представленная совокупностью диаграмм, использующих методы структурного анализа и проектирования.

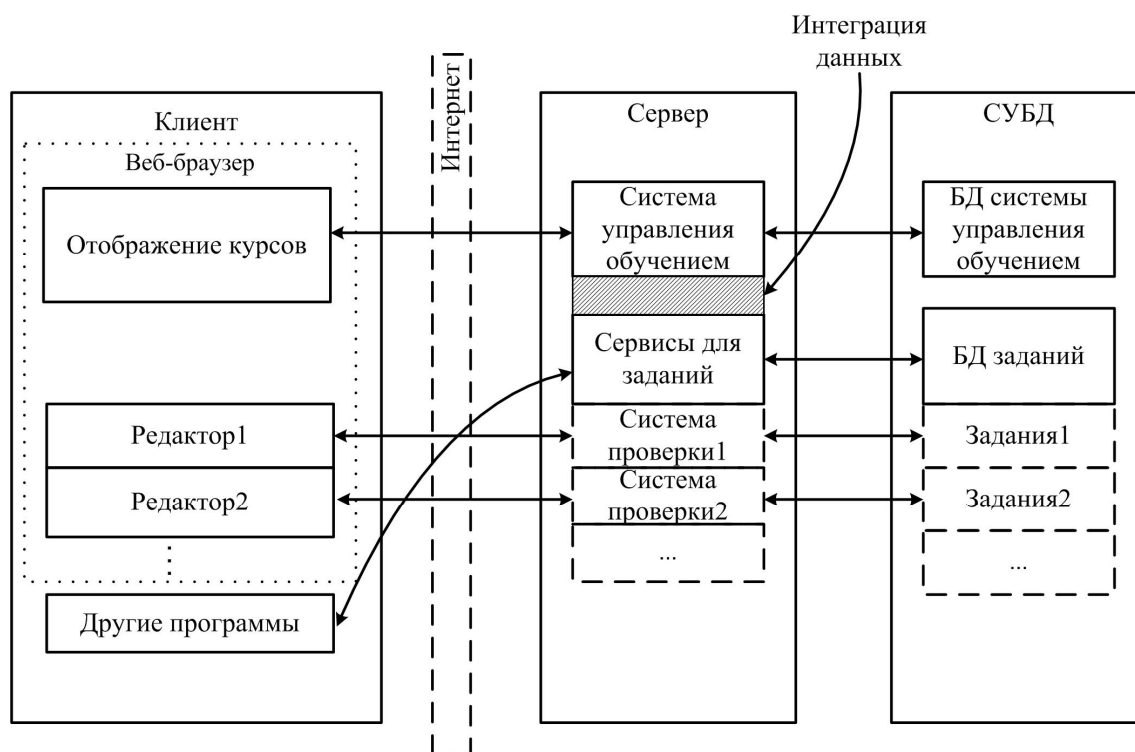


Рисунок 3 – Внутренняя организация на уровне программ

Предложенная логическая структура позволяет перенести нагрузку по подготовке решения задачи на сторону клиента, разгрузив таким образом сервер. Выполнение проблемно-ориентированных компонент в виде сервисов позволяет разнести источники информации, необходимые для обучения, в пространстве и во времени. На рисунке 4 приведена структурная схема инструментальной среды.

Блок управления осуществляет координацию взаимодействия всех модулей, реализует алгоритм управления. Он обеспечивает взаимодействие с пользователем, учитывает его привилегии. Блок определения уровня знаний идентифицирует уровень знаний обучаемого по правилам, заданной моделью обучаемого, в соответствии со сложностью решаемых им за-

дач. Блок решения задач включает тематический редактор, блок первичной обработки, механизм анализа и проверки корректности решения.

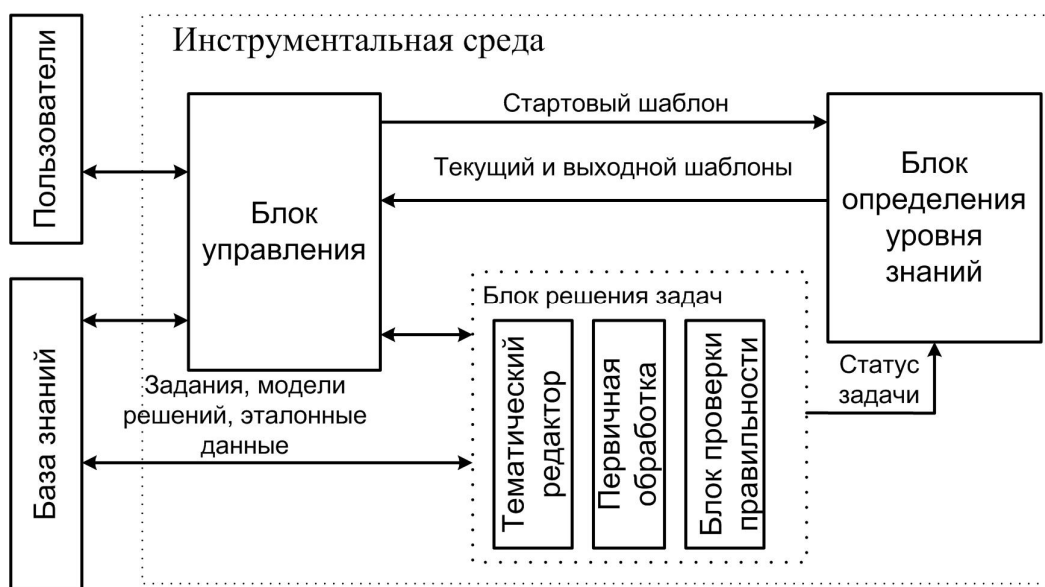


Рисунок 4 – Архитектура инструментальных средств поддержки в решении задач

Как показал анализ подходов к построению систем поддержки обучения в решении задач, функционально-структурная организация разрабатываемой системы перестраивается под требования предметной области. Предлагаемая инструментальная поддержка не является монолитной, а представляет набор динамических взаимодействующих в соответствии с деловой стратегией приложений, построенных на сервисной ориентации - средстве эффективной интеграции разнообразных систем. Основное преимущество предложенной архитектуры состоит в построении гибких адаптивных обучающих систем, способных к функциональным перестройкам и постоянному развитию.

**Четвертая глава** посвящена реализации программного обеспечения, разработанного на основе предложенных методов.

Реализация предложенной архитектуры программных средств построения систем обучения рассматривалась на примере построения инструментальной поддержки обучения решению задач по основам программирования. Это позволило на действующем прототипе инструментальной среды оценить ее возможности, работоспособность и эффективность. Исследования велись в двух направлениях:

1 Для исследования работоспособности и эффективности инструментальных средств поддержки в решении задач разработана локальная версия инструментальной среды «Algorithm», предназначенная для под-

держки обучения решению задач по основам программирования и реализующая проблемно-ориентированную часть архитектуры инструментальной средств, связанную со специфическими особенностями дисциплины. Среда содержит графический редактор блок-схем, интерпретатор, визуализатор и модуль анализа и проверки корректности представленных решений. Она позволяет экспериментально исследовать эффективность применения инструментальной среды в учебном процессе.

2 Для исследования возможностей интеграции инструментальной поддержки в информационно-образовательную среду локализована система управления обучением MOODLE, в которой разработан курс «Программирование». В курс «Программирование» интегрирована инструментальная среда «GraphView», поддерживающая редактор графов и визуализацию поиска путей в графах, имеющая функциональные возможности, аналогичные тематическим редакторам инструментальных средств поддержки построения систем обучения в решении задач. В ходе работы были реализованы визуализаторы-демонстраторы «ListView» и «GraphView», позволяющие в интерактивном режиме показать алгоритмы обработки динамических списков и поиска путей в графах. Визуализаторы «ListView» и «GraphView» внедрены в учебный процесс в политехническом институте Сибирского федерального университета.

Апробация инструментальной среды «Algorithm» проводилась на примере усвоения курса «Программирование на языке высокого уровня» в политехническом институте Сибирского Федерального университета. Целью эксперимента было определить качественные изменения уровня усвоения учебного материала, возникающие при применении инструментальной среды «Algorithm». В эксперименте принимало участие 8 групп общей численностью 210 человек. Во всех группах основы алгоритмизации изучались с использованием блок-схем. Студенты первых четырех групп (104 человека) проходили обучение по обычной форме подготовки (без применения среды «Algorithm») – контрольный поток. Студенты других четырех групп (106 человек) обучались с использованием среды «Algorithm» – экспериментальный поток.

По окончании курса обучения алгоритмизации студентам было выдано итоговое задание, по результатам которого можно судить о полученных навыках. Итоговое задание выполнялось письменно и содержало по две задачи на алгоритмизацию. Если студент не справлялся с итоговым заданием, то ему предоставлялись две дополнительные попытки. Результаты экспериментов (таблица 1) подтверждают эффективность применения инструментальной среды. Экспериментальная группа, использовавшая в обучение среды «Algorithm», справилась с итоговым заданием, сократив на 19% количество попыток. Уровень подготовки обучаемых повысился, а время, затрачиваемое преподавателем на рутинную работу по проверке решений задач, на подборку тестов сократилось.



Таблица 1 Анализ результатов выполнения итогового задания

Показатели	Контрольный поток				Экспериментальный поток			
	Группа 1	Группа 2	Группа 3	Группа 4	Группа 1	Группа 2	Группа 3	Группа 4
Число студентов	26	25	26	27	27	27	25	26
1 попытка	10	8	10	12	15	15	13	14
2 попытка	9	10	8	8	10	9	10	9
3 попытка	7	7	8	7	2	3	2	3
Среднее количество попыток по группам	1,88	1,96	1,92	1,89	1,51	1,55	1,56	1,58
Среднее количество попыток по потоку	1,91				1,55			

Следует отметить, что при использовании инструментальной поддержки задачи проверяются оперативно и качественно, так как тестовые данные заранее подготовлены и предусматривают все варианты исхода решения задач, включая крайние и исключительные случаи, с большим объемом входных и выходных данных. В обычной практике преподаватель не имеет такой возможности. Все обучаемые находятся в равных условиях: их задачи проверяются в одинаковых условиях, исключается эмоциональная составляющая взаимоотношений с преподавателем.

Исследование методов взаимодействия инструментальной среды «GraphView» с системой MOODLE показало, что разработка инструментальной среды для построения систем поддержки обучения в решении задач, основанной на сервисах, позволит упростить разработку программного обеспечения за счет выявления общих принципов построения подобных систем и взаимодействия их компонентов.

**В заключении** формулируются основные результаты работы.

Приложения содержат акты о внедрении, алгоритмы управления поддержкой процесса обучения решению задач по этапам, фрагменты программ, разработанных для экспериментальных исследований инструментальной поддержки.

## ОСНОВНЫЕ РЕЗУЛЬТАТЫ РАБОТЫ

1 Предложен подход к построению систем поддержки процесса обучения, основанный на разделении функциональности, порождающий различные конфигурационные аспекты программного комплекса, обеспечивающий эволюцию его функций.

2 Разработана модель инструментальных средств для построения систем поддержки обучения в решении задач, описывающая базовые и проблемно-ориентированные составляющие реализуемой функциональности, расширяющей интеллектуальные возможности обучающих систем.

3 Предложена стратегия использования инструментальных средств поддержки обучения решению задач, инвариантная к изучаемой дисциплине, отличающаяся свойством динамической адаптации к уровню знаний обучаемого.

4 Разработана архитектура инструментальных средств для построения систем поддержки обучения в решении задач на основе открытых систем.

5 Разработана инструментальная среда «Algorithm», позволяющая формировать практические навыки решения задач по основам программирования. В результате экспериментальных исследований показана эффективность применения инструментальной поддержки в обучении решению задач. Реализованы визуализаторы «ListView» и «GraphView», которые интегрированы в систему управления обучением для исследования возможностей их взаимодействия.

### Результаты работы отражены в следующих публикациях:

**1 Редькина, А.В. Обучение синтезу алгоритмов / А.В. Редькина // Вестник СибГАУ. – 2008. – №1(18) – С. 30-34.**

**2 Редькина, А.В. Автоматизированная система обучения синтезу алгоритмов / А.В. Редькина // Системы управления и информационные технологии. – 2007. – №2.2(28). - С. 280-284**

3 Редькина, А.В. Информационные технологии для решения дидактических задач в курсе программирования / А.В. Редькина // Информационные технологии моделирования и управления. Научно-технический журнал. Воронеж: Изд-во «Научная книга». – 2007. – Вып. 5(39) – С. 523-528.

4 Редькина, А.В. Визуализация поиска путей в графе «GraphView» / А.В. Редькина, Малявко С.Н. // Свидетельство об официальной регистрации программы для ЭВМ №2007613730 / Федеральная служба по интеллектуальной собственности, патентам и товарным знакам. – 31.08.2007

5 Редькина, А.В. Открытые технологии в непрерывном образовании / А.В. Редькина, Покидышева Л. И., Редькин А. В // Современное образование: содержание, технологии, качество. Материалы XII международная конференция. С-Петербург: Изд-во СПбГЭТУ «ЛЭТИ». – 2006, Т1 – С. 65-67.

6 Редькина, А. В. Подходы к построению инструментальных средств поддержки систем обучения решению задач / А. В. Редькина // Теоретические основы проектирования и разработки распределенных информационных систем: Сб.ст. VIII Всероссийской научно-практической конференции с международным участием, ПРИС 2010. – Красноярск: Изд-во «Экспресс-Офсет» –2010. – С. 108 – 112.

7 Редькина, А.В. Система проведения дистанционных конкурсов по программированию / А.В. Редькина, Л. И. Покидышева, А. В. Редькин // Наука и образование. Материалы Всероссийской научной конференции. Белово. – 2003. – С. 185-186.

8 Редькина, А.В. Подсистема тестирования лабораторных работ по программированию/ А.В. Редькина, А. В. Редькин // Управление качеством в новых информационных технологиях, системах и образовании. Материалы Всероссийской научной конференции. Москва. – 2003. С. 125-126.

9 Редькина, А.В. Тесты в системе дистанционного образования по курсу «Алгоритмические языки и программирование» / А.В. Редькина, А. В. Редькин // Достижения науки и техники – развитию сибирских регионов. Материалы Всероссийской научно-практической конференции. Красноярск: Изд-во КГТУ. – 2003. – С. 159-160.

10 Редькина, А.В. Система дистанционного образования по курсу «Алгоритмические языки и программирование» / А.В. Редькина, А. В. Редькин // Вестник КГТУ, Вып. 33, Математические методы и моделирование. Красноярск: Изд-во КГТУ. – 2004. – С. 249-256.

11 Редькина, А.В. Программная среда для обучения Moodle / А.В. Редькина, А. В. Редькин // Повышение качества непрерывного профессионального образования. Материалы Всероссийской научно-практической конференции. Красноярск: Изд-во КГТУ. – 2005. – С.236-239.

12 Редькина, А.В. Обучающие системы / А.В. Редькина // Проблемы информатизации региона (ПИР-2000). Материалы Шестой Всероссийской научно-практической конференции: Красноярск: КГТУ. – 2000. – С. 174.

Редькина Александра Васильевна  
Программные средства построения систем поддержки обучения решению задач  
Автореф. дисс. на соискание ученой степени кандидата технических наук  
Подписано в печать 14.05.2010. Заказ №  
Формат 60x90/16. Усл. печ. л. 1,3. Тираж 100 экз.  
Отпечатано в типографии ИПК Сибирского федерального университета  
660041, г. Красноярск, пр. Свободный, 82а