



Math-Net.Ru

Общероссийский математический портал

М. А. Антонов, В. Л. Шабанов, Разработка программного комплекса для проведения олимпиад по программированию в Internet, *Матем. моделирование и краев. задачи*, 2008, часть 4, 12–16

Использование Общероссийского математического портала Math-Net.Ru подразумевает, что вы прочитали и согласны с пользовательским соглашением

<http://www.mathnet.ru/rus/agreement>

Параметры загрузки:

IP: 18.97.14.82

8 февраля 2025 г., 15:29:14



менеджеры сообщений определённых типов. Каждый такой менеджер получает сообщение, выполняет некоторые операции, обращаясь к определённым ботам (эту информацию можно получить из сообщения), и при необходимости возвращает некоторые результаты пользователям.

Преимущество такой схемы заключается в том, что вся низкоуровневая часть работы с Jabber и общая организация обработки сообщений уже реализована и инкапсулирована в соответствующих частях системы. Программисту, желающему добавить в систему бота, достаточно достать исходники (которые распространяются свободно), реализовать *один* класс — собственно описать бота с требуемой функциональностью, и зарегистрировать нового бота у нужного менеджера событий.

При реализации в системе достаточного количества ботов, она может быть полезна многим пользователям сети Jabber.

Самарский государственный технический университет, г. Самара
olorinbant@gmail.com

УДК 50.41.25

М. А. Антонов, В. Л. Шабанов
**РАЗРАБОТКА ПРОГРАММНОГО КОМПЛЕКСА
ДЛЯ ПРОВЕДЕНИЯ ОЛИМПИАД
ПО ПРОГРАММИРОВАНИЮ В INTERNET**

Введение. Олимпиада по программированию — интеллектуальное соревнование по решению различных задач, для решения которых необходимо придумать или применить какой-либо алгоритм и/или реализовать данный алгоритм на одном из языков программирования. Существуют и постоянно развиваются новые методы эффективной и надёжной проверки решений задач на этих олимпиадах.

Ранее программы участников проверялись вручную, посредством анализа алгоритма решения, после чего жюри принимало решение о начислении баллов. Этот способ проверки занимал много времени и был малоэффективен, поскольку не проводилась

проверка алгоритма на конкретных входных данных. При такой проверке невозможно выявить оптимальность решения, т. к. алгоритм, реализованный участником, мог не совпадать с алгоритмом, разработанным жюри при составлении задачи.

Следующим способом проверки стала проверка решений с помощью ЭВМ: жюри вручную запускало алгоритм на наборе тестов, составленных с учётом возможных ошибок участников, а баллы выставлялись в зависимости от количества тестов, на которых решение участника дало правильный ответ. Такой способ проверки выявлял наиболее типичные ошибки, а также позволял абстрагироваться от алгоритма участника (т. к. более не было необходимости просматривать исходный код решения вручную), однако время работы и размер использованной оперативной памяти можно было оценить лишь приблизительно.

Далее для проверки стали применяться программные комплексы, призванные облегчить задачу многократного запуска решений на одном и том же наборе тестов. Такие комплексы могут самостоятельно проверять ответы, выводимые решениями участников, и ограничивать время работы этих решений. Одновременно с развитием локальных сетей стала возможным доставка решения до физического места проверки, отправка и обработка результатов без вмешательства жюри. Участнику необходимо лишь отправить исходный текст своего решения по сети на сервер и, спустя несколько секунд, получить ответ проверяющей системы.

Обзор разработанной авторами системы. Система Dudge (Distributed Judge System) предоставляет завершённую инфраструктуру для проведения online-олимпиад по программированию — управление контентом (задачи, проводимые соревнования, пользователи), собственно проверку решений, веб-интерфейс для взаимодействия с пользователями. Архитектурно Dudge реализован как Enterprise-приложение, работающее в Internet в режиме 24/365, и удовлетворяет как специальным требованиям автоматизированной системы проверки решений — поддержка регистрации, отправки решений на сервер, проверку их на сервер, отсылку пользователю его результатов, обработка статистики, так и общим требованиям, предъявляемым к системам такого рода — высокая переносимость, отказоустойчивость, масштабируемость, защищённость.

Архитектура Dudge. Программный комплекс состоит из сле-

дующих компонентов.

Веб-интерфейс. Конечный пользователь системы взаимодействует с ней через web-интерфейс, разделенный физически на серверную и клиентскую части. Клиентская часть отвечает за корректное отображение интерфейса в браузере на компьютере пользователя, используя для этого общеиспользуемые веб-технологии представления данных. HTML отвечает за организацию представления информации на страницах, CSS — за единообразие в оформлении различных частей интерфейса, и JavaScript — за интерактивность.

Описанная клиентская часть динамически генерируется серверной частью в ответ на запрос пользователем определённого ресурса и отсылается клиенту (браузеру). В свою очередь, серверная часть разделяется на две логические части — генератор клиентской части и контроллер web-приложения.

Генератор представляет собой множество JSP-страниц (Java Server Pages, <http://java.sun.com/products/jsp/>), каждая из которых соответствует одной HTML-странице. При обращении клиента по соответствующему HTTP-адресу JSP-страница создает HTML-страницу и возвращает её клиенту.

Контроллер web-приложения является модулем разделения обязанностей в соответствии с паттерном MVC (Model—View—Controller, Модель—Представление—Контроллер). Этот паттерн позволяет архитектурно разделять крупные приложения (или их отдельные части, как в нашем случае) на три отдельных компонента (модель данных приложения, пользовательский интерфейс и управляющая логика), так, что модификация одного из компонентов оказывает минимальное воздействие на другие компоненты.

Слой бизнес-логики. В этом слое происходит обработка пользовательских запросов, это «сердце» приложения. Компонент бизнес-логики выполняет две основные задачи.

1. Выполняет обработку запросов на действия с данными (например, регистрация в системе нового пользователя или получение информации о нужной задаче).
2. Управляет проверкой присланных участниками решений.

Для реализации используется технология EJB (Enterprise Java Beans, <http://java.sun.com/products/ejb>). В качестве входной точки используются несколько EJB-компонентов, из которых про-

исходит обращение к другим модулям системы.

Модуль проверки решений. Модуль проверки решений реализован на C++, поскольку в нём требуется прямой доступ к низкоуровневому API операционной системы, и связывается с остальной частью системы (написанной на Java) с помощью технологии JNI (Java Native Interface — стандартный механизм Java, с помощью которого Java-код может взаимодействовать с собственным (платформно-зависимым) системным или прикладным кодом (вызывать его и вызываться им), написанным на языках, подобных Си, Си++ или ассемблера, скомпилированным и работающим непосредственно под управлением той или иной операционной системы). Требуется это для полного контроля над процессом проверки решения.

База данных. Хранит в себе всю постоянную информацию, обрабатываемую остальными частями системы.

Средства разработки. В качестве основной IDE используется *NetBeans* 6.0 (<http://www.netbeans.org/>). Это свободная интегрированная среда разработки приложений на языке программирования Java, Ruby, C++ и ряде других, поддерживающая рефакторинг, профилирование, выделение синтаксических конструкций цветом, автодополнение набираемых конструкций на лету, множество predefined шаблонов кода и др.

Для управления версиями программы использовалась открытая система контроля версий *Subversion* (<http://subversion.tigris.org/>), являющаяся стандартом де-факто для Интернет-репозитариев исходного кода. У *NetBeans* существует специальный SVN-плагин, позволяющий управлять версиями программы непосредственно из среды разработки.

В качестве СУБД был выбран свободный сервер баз данных *PostgreSQL*. Для администрирования и разработки базы данных применялся *pgAdmin3* (<http://www.pgadmin.org/>) — открытое приложение для работы с базами данных под управлением *PostgreSQL*. Эта утилита поддерживает администрирование сервера и БД *PostgreSQL*, редактор кода, браузер объектов базы данных, браузер запросов и многое другое.

Область применения. Разработанный программный комплекс может применяться для проведения олимпиад по программированию различного уровня в среде Internet.

В настоящее время он применяется в СамГТУ как для прове-

дения олимпиад по программирования уровня ВУЗа (в частности, весенняя вузовская олимпиада 2008 г. была проведена с помощью системы Dudge), так и для проведения регулярных тренировок по спортивному программированию, в которых участвуют студенты различного уровня подготовки и из которых в дальнейшем формируются сборные СамГТУ, представляющие ВУЗ на олимпиадах более высокого уровня.

Самарский государственный технический университет, г. Самара

olorinbant@gmail.com; vir1@glint.ru

УДК 532.5:519.8

Р. А. Байкенов, М. Ж. Мукумбеков

ВЫЧИСЛИТЕЛЬНЫЕ ТЕХНОЛОГИИ В ОСВОЕНИИ ЗАЛЕЖЕЙ НЕФТИ

Рост темпа производительности нефтегазодобывающей отрасли ставит новые задачи рационального освоения нефтегазовых месторождений, связанные с адекватным математическим моделированием, решением задач фильтрации, выбором эффективных вычислительных технологий, проведением расчётов технологических показателей для данных месторождений [1–3].

Как известно, месторождения по своему геологическому типу развития бывают различными. Это, в свою очередь, даёт разные постановки математических моделей для описания внутрипластового течения жидкости, и, соответственно, алгоритмы их решения. Поэтому в данной работе авторами была поставлена цель объединения различных математических моделей фильтрации в пакет прикладных программ. Пакет прикладных программ по моделированию процессов разработки состоит из нескольких основных модулей. Остановимся только на одном из них — модуле моделирования разработки нефтегазового месторождения нагнетанием пара в системе скважин.

В этом модуле рассматривается плановая задача многофазной фильтрации для различных вариантов разработки месторождения путём нагнетания пара. Математическая модель данной задачи описывается следующими уравнениями (в безразмерном виде):