

Н. В. СЕРГИЕВИЧ

МГПУ им. И. П. Шамякина (г. Мозырь, Беларусь)

**О СТРУКТУРЕ БАЗЫ ДАННЫХ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ
ТЕСТИРОВАНИЯ «MASTERTEST»**

Процесс тестирования решения алгоритмической задачи на компьютере приблизительно можно описать следующим образом: составляется условие задачи, решить которую учащемуся требуется с помощью одного или нескольких изученных алгоритмов, причем в условии четко регламентируется результат, выдаваемый программой во всех возможных ситуациях, а также задаются области значений всех входных параметров. При этом стиль написания программы не учитывается, но строго отслеживается корректность ее работы [1].

Правильность работы программы оценивают по заранее подготовленному набору тестов, в котором каждый тест состоит из входных и выходных данных: программе даётся входной набор данных, а полученные на выходе результаты сравниваются с эталонными [2]. Способы сравнения могут варьироваться. Например, это может быть побайтное сравнение файлов либо подстановка ответа в условие задачи.

Обычно наборы составляются так, чтобы наиболее комплексно протестировать программу, проверив ее корректную работу на всех возможных наборах входных данных, удовлетворяющих условиям поставленной задачи. При этом можно также учитывать и некоторые дополнительные ограничения, такие, как время исполнения программы, объём используемой оперативной памяти и др. Такой способ оценки результатов может быть достаточно просто автоматизирован.

Для обеспечения возможности взаимодействия ядра и серверной части системы необходимо разработать соответствующий механизм. Т.к. ядро системы работает с очередью задач, а серверная часть активизируется по запросу от клиента, то прямой связи быть не может. Следовательно, нужно искать возможность обмена информацией без прямого соединения, но с использованием некоторого промежуточного «хранилища». Наилучшим вариантом для этого является использование базы данных.

База данных (БД) в данном случае является не только хранилищем данных, но и одним из основных механизмов взаимодействия автоматизированной системы тестирования и web-интерфейса. Во-первых, система тестирования является многопользовательской, следовательно, хранить в БД необходимо информацию о пользователях. Второй важной составляющей является информация о задачах, которые будут предоставлены учащимся для решения. Для работы с АСТ требуется также хранить сведения о поступивших на тестирование решениях и, соответственно, результатах тестирования предыдущих решений [3].

В результате оценки всех имеющихся требований спроектирована следующая структура БД, приведенная на рисунке. В данной БД отсутствует так называемая избыточность данных. Во всех таблицах есть первичные индексы (помечены флагом PK – primary key), необходимые для однозначного определения любой записи таблиц. В некоторых таблицах добавлены вторичные индексы (по одному или нескольким полям) для ускорения работы операций, связанных с выборкой данных (выделены «жирным» шрифтом). Вторичные индексы добавлялись только в тех случаях, когда по данному полю или группе полей осуществлялся отбор данных [4]. Следствием добавления вторичных индексов стала значительно возросшая скорость обработки некоторых запросов к БД.

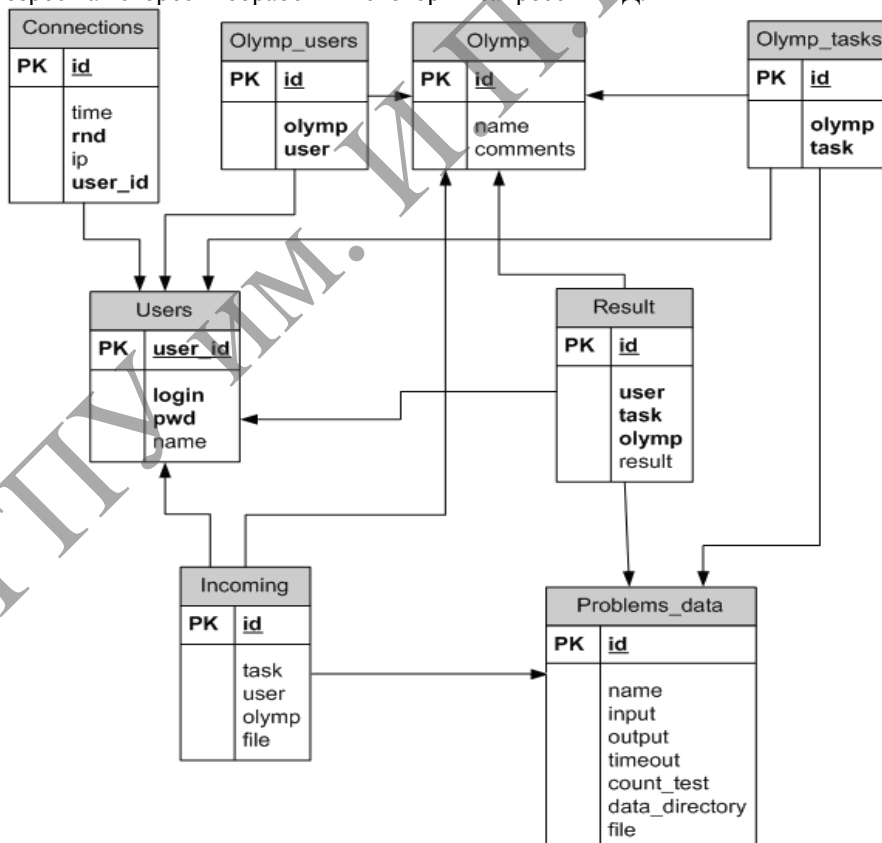


Рисунок – Структура базы данных АСТ «MasterTest»

Основными объектами нашей базы данных являются плоские таблицы. Каждая из таблиц либо хранит информацию о каком-либо множестве объектов (справочники), либо содержит сведения, характеризующие некоторую совокупность объектов, описанных в других таблицах (сводные таблицы).

В справочниках хранится информация об объектах, а также уникальный ключ, который позволяет однозначно идентифицировать объект. В нашей базе данных имеются три справочника: Users, Problems_data, Olymp. В них, соответственно, хранится информация о пользователях, задачах и курсах (см. таблицу 1). Курсы объединяют в себе подмножество различных задач комплекса предлагаемых для решения, и подмножество пользователей, которые будут их решать.

Таблица 1 – Основные таблицы базы данных

Название 1	Назначение 2	Структура 3
Users	Информация о пользователях	User_id – уникальный идентификатор пользователя, Login – имя пользователя, используемое для авторизации, Pwd – пароль, используемый для авторизации, Name – реальное имя пользователя, отображаемое в таблице результатов.
Problems_Data	Описание всех необходимых атрибутов задач, используемых в процессе тестирования решений.	Id – уникальный идентификатор задачи, Name – название задачи, Input – название входного файла, Output – название выходного файла, Timeout – ограничение времени исполнения пользовательского решения на одном тесте, Count_Test – количество тестов, Data_Directory – путь к каталогу с тестами, File – путь к файлу с условием задачи.
Olymp	Описание курса.	Id – уникальный идентификатор курса, Name – название курса, Comments – дополнительная информация.
Olymp_Users	Описание подмножества пользователей курсов.	Id – уникальный идентификатор связи пользователя и курса, Olymp – идентификатор курса, отражающий принадлежность пользователя к данному курсу, User – идентификатор пользователя.
Olymp_Tasks	Описание подмножества задач курсов.	Id – уникальный идентификатор связи курса и задачи, Olymp – идентификатор курса, отражающий принадлежность задачи к данному курсу, Task – идентификатор задачи.

Последние две таблицы, по сути, уже не относятся к справочникам, т.к. не содержат характеристик какого-либо объекта, а хранят информацию, которая относится к определенной связке объектов. Описание оставшихся таблиц приведено в таблице 2.

Таблица 2 – Дополнительные таблицы базы данных

Название	Назначение	Структура
Connections	Информация о подключении пользователей к системе	Id – уникальный идентификатор подключения; Rnd – ключ авторизации; Time – Время окончания сессии; Ip – IP адрес системы пользователя; User_id – идентификатор пользователя.
Incoming	Информация о поступивших на тестирование решениях	Id – уникальный идентификатор решения; Olymp – идентификатор курса; Task – идентификатор задачи; User – идентификатор пользователя; File – путь к файлу с решением; Intime – время отправки решения; Protocol – ссылка на протокол; Compiler – идентификатор компилятора.
Result	Таблица результатов	Id – уникальный идентификатор; Olymp – идентификатор курса; Task – идентификатор задачи; User – идентификатор пользователя; Result – результат тестирования.
Points	Информация о количестве баллов за пройденный тест	task – уникальный идентификатор задачи; test – номер теста; points – количество баллов за этот тест.

Забрав решение из очереди, ядро тестирует его и формирует соответствующий протокол, заносющийся в соответствующую таблицу Protocols. Эта таблица содержит лишь общие данные о результате тестирования. Более детализированные данные находятся в таблице ProtocolsD.

Таким образом, ядро тестирующей системы избавлено от необходимости заботиться о входящих данных [3]. Этим занимается серверная часть, предоставляя ядру решение пользователя. Именно она заботится о регистрации пользователей, подписке их на курсы, получении условий задач, обеспечении возможности отсылки решений.

ЛИТЕРАТУРА

1. Сергиевич, Н.В. Автоматизация проверки решений задач по программированию / Н.В. Сергиевич, М.И. Полоз // Сборник работ преподавателей физико-математического факультета. – Мозырь: УО МГПУ им. И.П. Шамякина, 2011.
2. Сергиевич, Н. В. О преподавании алгоритмизации и программирования в средней школе // Инновационные технологии обучения физико-математическим дисциплинам: материалы Междунар. науч.-практ. Интернет-конф., 27-31 окт. 2008 г., г. Мозырь / редкол.: В.В. Валегов (отв. ред.) [и др.]. – Мозырь: УО МГПУ им. И.П. Шамякина, 2008. – С. 144–147.
3. Лопато В. М. О разработке автоматизированной системы тестирования // Инновации-2004: Материалы XI Респ. студ. науч.-практ. конф., 22 апреля 2004 г., Мозырь: в 2 ч. – Мозырь: УО МГПУ, 2004 – Ч. 1. – С. 89.
4. Лещенко В. В. О подходе к реализации тестирующего модуля в автоматизированной системе тестирования // Инновации-2004: Материалы XI Респ. студ. науч.-практ. конф., 22 апреля 2004 г., Мозырь: в 2 ч. – Мозырь: УО МГПУ, 2004 – Ч. 1. – С. 89.

МГПУ им. И.П.Шамякина