СПИСОК ЛИТЕРАТУРЫ

1. Горбатов В. А. Основы дискретной математики. – М.: Высш. шк., 1986. – 311 с.
2. Балдин А. В., Брешенков А. В. Анализ проблемы проектирования реляционных баз данных на основе использования информации табличного вида и разработка модели методики проектирования. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2007. – 150 с.
3. Кузнецов О. П., Адельсон-Вельский Г. М. Дискретная математика для инженера. – 2-е изд. перераб. и доп. – М.: Энергоатомиздат, 1988. – 480 с.

Статья поступила в редакцию 23.01.2007

Александр Владимирович Брешенков родился в 1955 г., окончил МВТУ им. Н.Э. Баумана в 1982 г. Канд. техн. наук, доцент кафедры "Компьютерные системы, комплексы и сети" МГТУ им.Н.Э. Баумана. Автор 70 научных работ в области САПР ЭВМ и баз данных.

A.V. Breshenkov (b. 1955) graduated from the Bauman Moscow Higher Technical School in 1982. D. Sc. (Eng.), assoc. professor of "Computer Systems, Complexes and Networks" department of the Bauman Moscow State Technical University. Author of 70 publications in the field of systems of automated design and data bases.

УДК 681.3.01

Б. И. Рабинович

РЕДАКТОР ШАБЛОНОВ СОЕДИНЕНИЙ КАК СРЕДСТВО ИНТЕГРАЦИИ БАЗЫ ЗНАНИЙ СИСТЕМЫ "АНАЛИТИК" С ВНЕШНИМИ ИСТОЧНИКАМИ ДАННЫХ

Рассмотрены методы, позволяющие расширить возможности логико-аналитической системы "Аналитик" в плане хранения и обработки информации. В качестве хранилища структур знаний предложено использовать систему управления базами данных Oracle, которая допускает работу с большими объемами данных. Рассмотрены варианты подключения к системе "Аналитик" в качестве дополнительных источников информации внешних баз данных, что позволит пользователю получать более полную информацию об интересующем объекте.

Одной из актуальных проблем взаимодействия информационных потоков является обработка больших потоков неформализованной информации — текстов естественного языка. Для этого в рамках плановых тем Института проблем информатики РАН были разработаны

системы ИКС, “Аналитик”, “Криминал” [1, 2]. Их особенность заключается в наличии лингвистического процессора, отображающего тексты на структуры знаний, на уровне которых обрабатывается информация. Такие системы для хранения структур знаний используют свою внутреннюю базу данных (БД), основанную на плоских файлах, и подкачиваются по мере необходимости, образуя активную часть базы знаний (БЗ), где и осуществляется обработка. Итак, БД играет роль хранилища знаний, т.е. внешней БЗ. В то же время возможности существующей БД не соответствуют потребностям рынка информационных услуг. Максимальное число документов и структур, с которыми внутренняя БД устойчиво работает, ограничено несколькими сотнями тысяч. В то же время существующие потоки информации — это миллионы документов с объемами, исчисляемыми многими гигабайтами. Отсюда возникает необходимость использовать в качестве хранилища знаний современные системы управления БД (СУБД), обеспечивающие работу с большими объемами информации (Oracle, MSSQL, MySQL).

Помимо сказанного использование современных СУБД позволяет устранить ряд других недостатков внутренней БД, а именно сравнительно медленный поиск структур знаний при больших объемах данных и высокую трудоемкость удаления структур знаний. При этом в СУБД уже решены такие важные задачи, как защита данных, а также реализована возможность их обновления и управления.

Вторая проблема, которая особенно важна для системы “Криминал”, это использование внешних источников информации: телефонных справочников, адресных книг и других данных, введенных в соответствующие БД (“Кронос”, ГИБДД, МГТС) и широко используемых в криминалистике. В этом случае, используя внешние БД, следователь-аналитик получает наиболее полную информацию об интересующем его объекте. В то же время перекачать всю информацию в БЗ не представляется возможным по разным причинам: из-за большого объема, ограниченного доступа и т.д. Отсюда возникает необходимость организации взаимодействия внешних БД и БЗ.

Такое взаимодействие проиллюстрировано на примере БД МГТС. При этом взаимодействии в процессе обработки ряда объектов (телефонов, фигурантов) в рамках БЗ для получения недостающей информации формируется обращение к БД МГТС. Найденные данные преобразуются в структуры знаний, которые пополняют БЗ и таким образом увеличивают пространство поиска в аналитических режимах системы.

Структура внешней базы знаний. Будем называть БД, в которой хранятся структуры знаний (содержательные портреты) и сами документы, внешней БЗ системы или просто БЗ. Следует отличать ее от

активной части БЗ, находящейся в оперативной памяти, куда подкачиваются структуры из внешней БЗ, и где они обрабатываются.

Существующая (внешняя) БЗ представляет собой набор плоских файлов (табл. 1). В ней хранятся следующие данные: тексты загруженных в систему документов (DB_TXT.db), семантические сети (DB_ZZZ.db), каталоги и индексы, которые автоматически строятся на их основе. Кроме того, есть еще два типизированных файла (DB_TXT.dbi, DB_ZZZ.dbi), в которых хранится информация о том, на какой строке файлов БД с текстами и семантическими сетями начинаются документы, и какая у них длина. Это сделано для организации более быстрого доступа к данным [1].

Таблица 1

Структура существующей БД

Файл	Содержание	Атрибуты
DB_TXT.db	Текст документа	Текст
		Длина
DB_TXT.dbi	Типизированный текст	Номер документа
		Смещение
		Размер
DB_ZZZ.db	Семантическая сеть документа	Семантическая сеть
		Длина
DB_ZZZ.dbi	Типизированная семантическая сеть	Номер семантической сети
		Смещение
		Размер
NET2.slv	Индексный файл	Слово
		Частота
		Смещение
NET2.ind	Набор документов	Длина
		Набор документов

В БД есть два индексных файла. Первый (NET2.slv) — представляет собой перечень ключевых слов, найденных в семантических сетях, частоту их появления во всех документах и адрес (смещение) списка документов во втором файле (NET2.ind), в которых были найдены эти слова. Ключевыми являются слова, по которым осуществляется поиск. К ним не относятся предлоги, частицы, союзы и т.п.

На этапе загрузки строятся файлы с каталогами основных объектов, по которым проводится быстрый поиск по БД. Такими объектами могут быть, например, адрес, телефон, ФИО [3].

Работа с внешней БЗ осуществляется с помощью стандартных функций языка ДЕКЛ [4, 5]. Перечень основных функций и процедур — это запись семантической сети, запись текста загруженного файла, запись индексов и т.д.

Структура внешней БЗ в реляционной СУБД. В этом разделе предлагается структура внешней БЗ системы исходя из того, что

она будет реализована в современной реляционной СУБД (InterBase, MsSQL, MySQL, Oracle и т.п.) [6]. Особенностью предлагаемого решения является отказ от хранения структур знаний в плоских файлах [3].

Поскольку быстрый поиск — это основное преимущество СУБД, то можно отказаться от типизированных файлов DB_TXT.dbi и DB_ZZZ.dbi. Данные, которые хранятся в файлах DB_TXT.db и DB_ZZZ.db, можно объединить, так как они являются характеристикой одной и той же сущности "Документ". Для ускорения работы следует отдельно хранить заголовок документа, так как чаще всего отображаются не документы, а именно заголовки, которые представляют собой первые несколько слов документов. Файлы NET2.slv и NET2.ind представляют собой единую сущность "Слово", которая характеризуется самим словом, частотой и списком документов, в котором оно встретилось. Характеристика "частота" используется, чтобы показать, сколько раз то или иное слово встречается во внешней БЗ. Если в результате удаления документа эта характеристика становится равной нулю, это слово удаляется. Для создания экземпляра сущности "Слово" необходимо во время загрузки каждого документа разбирать его, преобразуя каждое слово в нормальную форму (единственное число, мужской род, именительный падеж для существительных), и уже после этого заполнять соответствующие таблицы. Для ускорения работы выделена отдельная сущность "Набор документов". Схема такой БЗ представлена в табл. 2.

Таблица 2

Структура БЗ в современной СУБД

Сущность	**Атрибут**
Документ	Номер документа
	Текст
	Семантическая сеть
	Заголовок
Слово	Номер слова
	Слово
	Частота
Набор документов	Номер набора
	Номер слова
	Номер документа
Каталог ФИО	Номер ФИО
	Номер документа
	ФИО
Каталог паспортов	Номер каталога паспортов
	Номер документа
	Серия и номер паспорта

Как видно из схемы, все данные, необходимые для того, чтобы система работала так же, как и с БЗ на плоских фалах, сохранены. После

переноса хранилища внешней БЗ в СУБД можно подключать внешние базы. Обратим внимание на преимущества нового способа хранения знаний: решается проблема удаления и изменения документов, увеличивается максимально возможный размер внешней БЗ, возрастает скорость поиска на больших объемах данных, значительно увеличивается уровень безопасности, появляется возможность восстановления данных и введения RAID-массивов. Кроме того, существенно повышается надежность системы в целом.

Взаимодействие с внешними БД. Ранее была рассмотрена задача организации внешнего хранилища знаний (внешней БЗ) в СУБД Oracle. Другая задача связана с подключением в качестве внешнего источника данных внешних БД (ВБД). Все ВБД в рамках решаемой задачи классифицируем как базы с простой и сложной структурами. Простая структура БД — это БД, состоящая из одной таблицы, сложная — из нескольких взаимосвязанных таблиц. Рассмотрим в качестве примера подключения простой ВБД базу МГТС.

Перечислим шаги, которые необходимо предпринять для подключения такой базы к ВБД системы:

1. Загрузить БД МГТС в СУБД Oracle [7, 8];

2. Обеспечить связь (шаблон) между объектами системы и объектами базы МГТС;

3. Сохранить шаблон в системе;

4. Настроить SQL запросы во внешние базы по шаблону.

5. По запросу визуализировать и сохранить результат поиска.

Подключение ВБД на примере базы МГТС. Рассмотрим перечисленные шаги на примере базы МГТС.

Имеющаяся в наличии база МГТС представляет собой один файл БД ACCESS размером 1,5 Гб с восьмью таблицами, в каждой из которых примерно по 300 тысяч записей. Таблицы имеют разную структуру (см. табл. 3, 4, 5). Чтобы загрузить эту базу в СУБД Oracle и в дальнейшем использовать ее в системе, необходим конвертор, который позволил бы загрузить все таблицы из ACCESS в одну таблицу Oracle.

Необходимо привести в соответствие объекты системы и поля БД МГТС (табл. 6). Как видно из структуры базы МГТС, одному объекту системы “ФИО” соответствует два поля базы МГТС, а одному объекту “Адрес” — четыре поля базы МГТС. Настройка шаблонов должна позволять ставить в соответствие одному объекту системы несколько полей внешней БД. В данном примере подобное соответствие иллюстрирует табл. 6.

Далее возникает вопрос о сохранении введенного шаблона в системе. Для этого можно предложить два способа:

Таблица 3

Фрагмент структуры таблицы БД МГТС с информацией о физических лицах

Телефон
Город
Улица
Дом
Квартира
Фамилия или название организации
Имя и отчество (полностью)

Таблица 4

Структура таблицы БД МГТС с информацией о юридических лицах

Номер телефона
Принадлежность к организации

Таблица 5

Фрагмент таблицы с базой МГТС

Table MGTS	
Id (PK)	integer
telefon	varchar(20)
City	varchar(30)
street	varchar(100)
house	varchar(5)
Flat	varchar(5)
fullname	varchar(100)

Таблица 6

Соответствие объектов системы полям БД МГТС

АНАЛИТИК	МГТС
Телефон	Телефон
Адрес	Город
	Улица
	Дом
	Квартира
ФИО	Фамилия или название организации
	Имя и отчество (полностью)

• записать определенную структуру, содержащую формат шаблона, в текстовый файл;

• записать определенную структуру в таблицу Oracle.

Второй способ предпочтительней, потому что одной из основных целей разработки является перенос хранилища данных из плоских файлов в СУБД. Структура соответствующей таблицы будет выглядеть следующим образом (табл. 7).

Таблица 7

Шаблоны

ID записи	Integer
Имя шаблона	Varchar2(20)
Объект системы	Varchar2(40)
Поле БД	Varchar2(80)
Запрос	Varchar2(200)

Такая структура позволяет хранить все возможные настройки шаблонов. Необходимо ввести уникальность по имени шаблона, чтобы не давать пользователю заводить несколько шаблонов с одним и тем же именем.

В случае когда возникнет необходимость поиска важной информации во внешней базе МГТС, необходимо выполнить следующую последовательность действий:

• определить значение объекта системы, которое будет использоваться в качестве параметра запроса;

• определить объекты системы, значения которых необходимо узнать, используя внешнюю БД;

• используя ранее созданный шаблон, автоматически создать запрос к внешней БД.

Например, пусть известно значение объекта “Телефон”, равное “123-45-67”. Необходимо узнать значение объекта “ФИО”. Тогда, используя шаблон “MGTS”, необходимо сформировать следующий запрос [9] к внешней БД “MGTS”.

```
Select mgts.family||’ ‘||mgts.fullname
From mgts
Where mgts.telefon=’123-45-67’.
```

В результате будет получено значение объекта “ФИО”, которое используется в последующей обработке.

Что касается работы со сложными БД, то здесь возникают дополнительные трудности. Для организации такой работы предлагается “вшить” все алгоритмы работы с ней в систему. Иначе (если необходимо визуализировать настройку шаблонов в виде удобного для пользователя интерфейса) потребуется организация сложного интерфейса,

необходимого, чтобы осуществить связку между таблицами на уровне языка SQL. Реализация такой задачи представляется крайне сложной и нецелесообразной. Гораздо проще позволить пользователю вводить запрос, который связывал бы объекты системы и поля сложной БД на языке SQL самостоятельно. Для этого в таблицу "Шаблоны" добавлено поле "Запрос".

После того как требуемая информация найдена, необходимо ее визуализировать и сохранить. Визуализация объектов системы реализована в виде графов. Каждый найденный объект отображается на графе и связывается дугой с объектом, по которому был совершен поиск. При этом можно предложить несколько вариантов хранения информации в системе.

Укажем на то, что для создания шаблонов, обеспечивающих связь системы с внешними БД, используется специальная прослойка, которая называется редактором шаблонов. В этом редакторе создаются шаблоны соответствия между объектами системы и объектами ВБД. При этом для одной ВБД в зависимости от сложности может создаваться один или более шаблонов соответствия. Отметим, что найденная в ВБД информация преобразуется в РСС и подкачивается в оперативную память, дополняя активную часть БЗ. Эта информация в виде РСС сохраняется в БЗ системы. Таким образом, происходит увеличение пространства поиска системы в целом.

Заключение. Описаны методы модернизации функционала системы "Аналитик" в соответствии с основными требованиями, предъявляемыми к подобному классу современных программных продуктов. Показано, что использование современных СУБД позволяет повысить производительность и надежность системы, уменьшить ограничения объема занимаемой физической памяти, увеличить степень безопасности системы. Хранилище знаний организуется на базе современной СУБД (взамен плоских файлов).

В статье также описан метод взаимодействия современной СУБД и активной части БЗ системы "Аналитик". Указывать на то, что была описана система управления БЗ, нельзя, так как функцию управления знаниями выполняет язык ДЕКЛ, в то время как СУБД выполняет лишь роль хранилища данных.

Систему такого класса можно с уверенностью назвать логико-аналитической. Ее особенности — это наличие лингвистического процессора, возможность автоматизированной обработки неструктурированной информации, возможность создания режимов аналитической обработки и отображения информации; наличие интерфейса соединения и извлечения информации из внешних БД и обеспечение функций

хранения и управления. Подобная система должна найти широкое применение в рамках комплексных систем, обеспечивающих различные виды поиска и аналитической обработки больших объемов информации (в том числе неформализованной) в разных прикладных областях.

СПИСОК ЛИТЕРАТУРЫ

1. Кузнецов И. П., Мацкевич А. Г. Особенности организации базы предметных и лингвистических знаний в системе АНАЛИТИК // Тр. междунар. конф. Диалог'2003. – М.: Наука, 2003.
2. Шарнин М. М., Кузнецов И. П. Продукционный язык программирования ДЕКЛ // В сб. "Система обработки декларативных структур знаний Деклар-2". – М.: Изд-во "ИПИРАН", 1988. – С. 134–152.
3. Рабинович Б. И. Аналитическая система обработки и управления структурированной информацией // Интеллектуальные технологии и системы. Вып. 6. – М.: Эликс+, 2003. – С. 173–186.
4. Кузнецов И. П. Продукционный язык программирования ДЕКЛ. Система обработки декларативных структур знаний Деклар-2. – М.: ИПИ РАН, 1988.
5. Рабинович Б. И. Организация баз знаний в современных СУБД. Проблемы и методы информатики // Тез. докл. II Науч. сессии ИПИ РАН. Москва, 18–22 апреля 2005 г. – М.: ИПИ РАН, 2005.
6. Блэкфорд Д., Стрехлоу К. KOAP Open Portal. К базам данных завтрашнего дня. [Электронный ресурс] / http://koapp.narod.ru/tehlit/base/bd/06db.htm.
7. Jason Couchman, Ulrike Schwinn. Oracle 8i Certified Professional DBA – М.: Лори, 2002.
8. Хомоненко А., Гофман В., Мещеряков Е., Никифоров В. Delphi 7, наиболее полное руководство. – Спб.: Изд-во "BHV". – 488 с.
9. Кузнецов С. Д. Центр Информационных Технологий. Введение в стандарты языка баз данных SQL [Электронный ресурс] / http://www.citforum.ru/database/sqlbook/index.shtml

Статья поступила в редакцию 25.04.2007

Борис Ильич Рабинович родился в 1983 г., окончил в 2005 г. МГТУ им. Н.Э. Баумана. Аспирант Института проблем информатики РАН. Автор 8 научных работ в области анализа данных.

B.I. Rabinovich (b. 1983) graduated from the Bauman Moscow State Technical University. Post-graduate of the Institute of Informatics Problems of the Russian Academy of Sciences. Author of 8 publications in the field of data mining.