

Модель базы знаний с возможностью интеграции внешних источников информации в системе «Аналитик»

И.П. Кузнецов, д.т.н., гл. науч. сотрудник, ИПИ РАН

Б.И. Рабинович, аспирант, ИПИ РАН

Аннотация

В данной работе рассматриваются методы, позволяющие расширить возможности логико-аналитической системы «Аналитик» в плане хранения и обработки информации. Для этого, во-первых, в качестве хранилища структур знаний предлагается использовать СУБД Oracle, которая допускает работу с большими объемами данных. И, во-вторых, рассматриваются варианты подключения к системе Аналитик в качестве дополнительных источников информации внешних баз данных (БД), что позволяет пользователю получать более полную информацию об интересующем объекте. В качестве примера рассматривается средства подключения БД МГТС.

Annotation

The article is devoted to analysis of a method that allows improving the work of the logical analytic system “Analytic” in case of storing and processing of information. In this purpose at first as storage of Knowledge Base is suggested to use the Data Base Management System (DBMS) “Oracle”, which provides working with a huge range of information. And, at second, the article deals with methods of linking external data bases to the system “Analytic”, what provides to user getting more important information about target object. The analysis given in the article is based on the example of working with the data base of MGTS.

1. Введение

Одной из актуальных проблем является обработка больших потоков неформализованной информации - текстов естественного языка (ЕЯ). Для этого в рамках плановых тем ИПИ РАН была разработана система «Аналитик», которая послужила основой создания других систем: ИКС, «Криминал» [7,10]. В этих системах организована автоматическая формализация текстов ЕЯ с построением структур, допускающих сложные виды логико-аналитической обработки – преобразование представлений, различные виды семантического поиска: поиск похожих, поиск по приметам, поиск по связям и др.

Особенность системы «Аналитик» - в наличии Базы Знаний (БЗ) и лингвистического процессора, отображающего тексты на структуры знаний. БЗ служит для хранения структур знаний и их обработки, которая осуществляется с помощью специализированного языка – Декл [10]. Следует отметить, что при работе с большими массивами текстовой информации (например, сводками происшествий или публикациями СМИ) возникают структуры знаний большого объема. Для работы с ними в системе «Аналитик» предложена двухуровневая организация БЗ. Различаются внешняя и активная БЗ. Для длительного хранения структур знаний большого объема служит внешняя БЗ. Для этого была разработана специализированная БД, основанная на плоских файлах. Упомянутые структуры из этой БД подкачиваются в оперативную память по мере необходимости, образуя активную часть (АЧ) БЗ, где и осуществляется обработка. Итак, БД играет роль хранилища знаний, т.е. внешней БЗ.

В тоже время возможности существующей БД не соответствуют потребностям рынка информационных услуг. Максимальное количество документов и структур, с которыми специализированная БД обеспечивает устойчивую работу, ограничено несколькими сотнями тысяч. В тоже время существующие потоки информации - это миллионы документов с объемами, исчисляемыми многими гигабайтами. Отсюда возникает необходимость использовать в качестве хранилища знаний современные СУБД, обеспечивающие работу с большими объемами информации (Oracle, MSSQL, MySQL).

Помимо сказанного, использование современных СУБД позволяет устранить ряд других недостатков специализированной БД: это сравнительно медленный поиск структур знаний при больших объемах данных, высокая трудоемкость удаления структур. При этом в СУБД уже решены такие важные задачи, как защита данных, 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", MSSQL, MySQL) и широко используемые в криминальной милиции. В этом случае, используя внешние БД, следователь-аналитик сможет получить наиболее полную информацию об интересующем его объекте. В тоже время перекачать всю эту информацию в БЗ не представляется возможным по разным причинам: большой объем, ограниченный доступ и др. Отсюда возникает необходимость организации взаимодействия внешних БД и БЗ.

Такое взаимодействие будет проиллюстрировано на примере БД МГТС. При этом 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.

2. Основные компоненты системы, основанной на технологии БЗ

В этой главе приведено описание основных компонентов системы. Для наглядности представим структурную схему системы (см. рис.1).

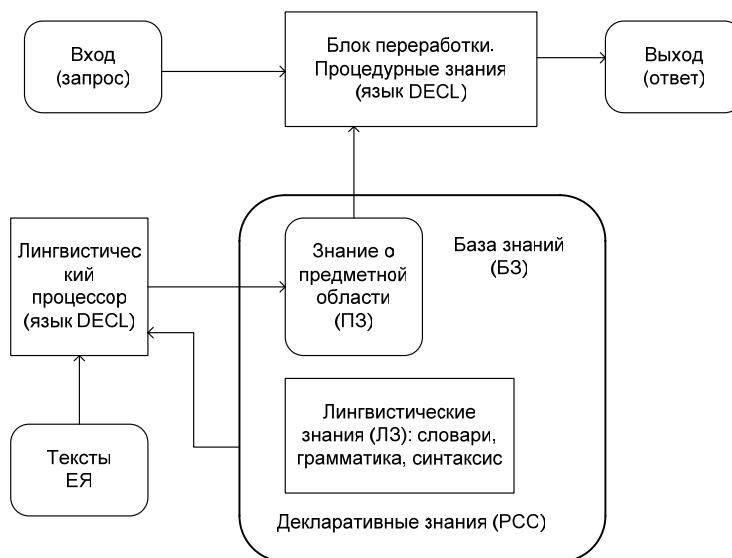


Рис. 1. Структурная схема системы.

2.1 Расширенные семантические сети.

Расширенные семантические сети (РСС) играют роль структур знаний в БЗ [1]. РСС состоят из однотипных *N*-арных фрагментов. В каждый из них введена вершина, называемая кодом фрагмента и соответствующая всей представленной в нем информации. Помимо этого, вводится множество "внутренних" вершин, которые порождает сама система по мере необходимости, и которые сопоставляются неименованным объектам.

РСС ориентированы на отображение особенностей семантики естественного языка (ЕЯ) - упоминаемых объектов, их связей. Коды вершин служат для интеграции множества связанных объектов в один объект, что выражается на ЕЯ в виде форм с отглагольными существительными. Понятие связи рассматривается в широком смысле. Это могут быть не только бинарные отношения, но и зависимости. Например, связанными считаются также объекты, участвующие в одном действии. Группа связанных объектов может быть связана с другой группой, что на ЕЯ выражается в виде глагольных форм с отглагольными существительными, или же в виде сложноподчиненных предложений [16]. Опыт использования языка РСС в системах «Диес», «ИКС», «Аналитик» для формализации текстов ЕЯ показывает возможность представления семантики предложений (объектов и связей) с высокой степенью точности.

Представляющие объекты и связи какого-либо документа РСС образуют так называемые содержательный портрет этого документа [7]. Такие портреты необходимы для обеспечения быстрого и качественного поиска информации по значимым компонентам и связям. Содержательные портреты (как и сами документы) помещаются в БД, ориентированную на большие потоки информации (в перспективе - сотни мегабайт) и обеспечивающую их быстрый выбор - за счет индексных файлов. Как уже говорилось, такие портреты подкачиваются в оперативную память (ОП) по мере необходимости, образуя активную часть БЗ.

При использовании технологии БЗ (в отличие от БД) структуры знаний не ограничиваются какими-либо схемами. Любая глагольная форма с ее обязательными и факультативными актантами навязывает свою схему, которая представляется в виде фрагмента РСС. Приведем пример.

Текст СМИ: "Багдад. 9 июня. ИНТЕРФАКС/АФП - Два акта саботажа были совершены на нефтепроводах на севере Ирака ..."
--

Содержательный портрет:

PLACE_(ГОРОД,БАГДАД/1+) ДАТА_(9,ИЮНЬ/2+) ОРГ_(ИНТЕРФАКС,АФП/3+) КОЛИЧ_(2,АКТ,САБОТАЖ/4+) PLACE_(СЕВЕР,ГОС-ВО,ИРАК/5+) СОВЕРШИТЬ(4-,НА,НЕФТЕПРОВОД,5-/6+)

Содержательный портрет представлен на языке РСС, см. п. 2.2. Каждая строка – это фрагмент РСС, который представляет информационный объект. Например, фрагмент ОРГ_(ИНТЕРФАКС,АФП/3+) представляет организацию ИНТЕРФАКС/АФП,

выделенную из текста. Этот фрагмент имеет код 3+, который необходим для представления связей между объектами, в том числе, связи за счет их участия в одном действии. Фрагменты с именем PLACE_ представляют место, КОЛИЧ_ - количество, ДАТА_ - дату. Во фрагментах все слова (кроме имен объектов) представлены в нормальной форме.

Содержательные портреты позволяют с достаточной точностью представлять объекты, связи и действия, выделенные из текстов ЕЯ. Например, с помощью последнего фрагмента представлено, что в действии СОВЕРШИТЬ имеется два объекта, соответствующие количеству "Два акта саботажа" (с кодом 4+, 4-) и месту "На севере Ирака" (с кодом 5+, 5-). Упомянутое действие (с кодом 6+, 6-) может быть связано с другими действиями. Образуются сложные структуры, которые далеко выходят за рамки типовых схем БД.

2.2 Язык Декл.

Язык был создан для обработки структур знаний - РСС [1,10]. Декл - это язык логического программирования, обладающий металогическим уровнем. Он состоит из фрагментов РСС, а также правил ЕСЛИ ... ТО (продукций), в левой и правой части которых находятся РСС с вершинами-переменными. Последние играют роль слотов, которые при применении продукций к структурам знаний заполняются вершинами-объектами. Помимо этого в языке Декл имеются встроенные фрагменты (предикаты), при выполнении которых идет обращение к внешним программам, выполняющим интерфейсные и системные функции.

Язык Декл предназначен для системных программистов и инженеров по знаниям. Он служит для построения систем, основанных на знаниях. Его правила обеспечивают преобразования РСС: одни структуры заменяются другими. Одни правила (в случае их применимости) могут вызывать другие. Наборы правил с указанием последовательности их вызовов образуют программы на языке Декл. Как показывает опыт, язык Декл является удобным средством построения различного рода оболочек экспертных систем, семантико-ориентированных лингвистических процессоров, аналитических систем, работающих со структурами знаний [7]. С помощью Декл удастся относительно быстро решать различные интеллектуальные задачи, реализация которых на других языках программирования требует существенно больших временных и умственных затрат.

Программы, написанные на языке Декл, при считывании с диска в ОП автоматически транслируются на РСС, где имеются спецфрагменты, отделяющие левую часть от правой. Далее осуществляется их выполнение программным ядром. Таким образом, поддерживается металогический уровень: одни правила могут создавать и изменять другие правила (и те, и другие - это РСС).

Отметим, что правила или продукции языка Декл определяются, прежде всего, удобством обработки структур знаний. В Декл условная часть записывается первой, т.е. слева (в отличие, например, от языка Пролог), а следствие - справа. Процедура применения правил управляется другими правилами и сводится к проверке условия с порождением следствий. Таким образом, реализуется как прямой, так и обратный вывод. Более того, в левой и правой части каждого правила языка Декл может быть множество фрагментов (предикатов). Процедура вычисления носит не "жесткий" характер. Правила при применении осматривают все знания, находящиеся в ОП, а не только предикаты запроса. Можно легко управлять такой процедурой, заставляя правила осматривать только нужную часть знаний [10]. Все сказанное определяет высокие возможности языка Декл для обработки структур знаний и решения логико-аналитических задач.

2.3 Лингвистический процессор.

Лингвистический процессор написан на языке Декл и является неотъемлемой частью системы, основанной на знаниях, т.е. с БЗ. Он обеспечивает автоматическое построение по текстам документов их содержательных портретов (РСС). Он включает в себя блоки лексического, морфологического и синтактико-семантического анализа. Блок лексического анализа обеспечивает автоматическое выделение из текстов предложения и слов. Блок морфологического анализа обеспечивает преобразование слов к единому виду (каноническому) и дает их признаки (часть речи, род, число, падеж и др.) Морфологический анализ необходим, чтобы избавиться от различных форм написания слов, что облегчает поиск. Синтактико-семантический анализ служит для выделения из документа значимых объектов и связей. Блок синтактико-семантического анализа выполняет следующие функции:

- по признакам и контексту выделяет значимые объекты (ФИО людей, организации и др.);
- для каждого выявленного значимого объекта находит в документе связанную информацию (для лиц это их год рождения, пол, адрес и др.);
- осуществляет идентификацию объектов по анафорическим ссылкам.

В результате строится РСС - содержательный портрет документа. К этому портрету добавляются фрагменты, формируемые блоком обработки полученных данных. Он осуществляет содержательный анализ информации на уровне РСС с автоматическим выявлением особенностей документа и его значимых объектов. Например, автоматическое выявление атрибутов лица, его словесного портрета, формирование по классификатору особенностей события или происшествия.

2.4. Логико-аналитическая обработка

Логико-аналитическая обработка осуществляется программами на языке Декл, где данными являются структуры знаний. В результате удается решать новые задачи, связанные с семантическим поиском, экспертными оценками, принятием оперативных решений. Например, в системе Криминал решаются следующие наукоемкие задачи:

- Поиск похожих происшествий (документов) с подсчетом степени сходства и объяснением причин.
- Поиск похожих фигурантов по приметам.
- Поиск информации фактографического характера по запросам на естественном языке.
- Поиск прямых и косвенных связей фигурантов (по совпадению ФИО, по упоминанию в одном происшествии и др.) с их отображением в графическом виде.
- Логический анализ документов с автоматическим выделением особенностей происшествия - способа проникновения, совершения, орудия преступления и др.

Качественное решение этих задач зависит от полноты информации, доступной системе. Отсюда необходимость использования современных БД.

3. Структура внешней Базы Знаний

Будем называть БД, в которой хранятся структуры знаний (содержательные портреты) и сами документы, внешней Базой Знаний Системы или просто БЗ. Следует

отличать ее от активной части БЗ, которая находится в оперативной памяти, куда подкачиваются структуры из внешней БЗ и где осуществляется их обработка.

Существующая (внешняя) БЗ представляет собой набор плоских файлов. Структура существующей БД представлена в Табл.1.

Табл. 1. Структура существующей БД.

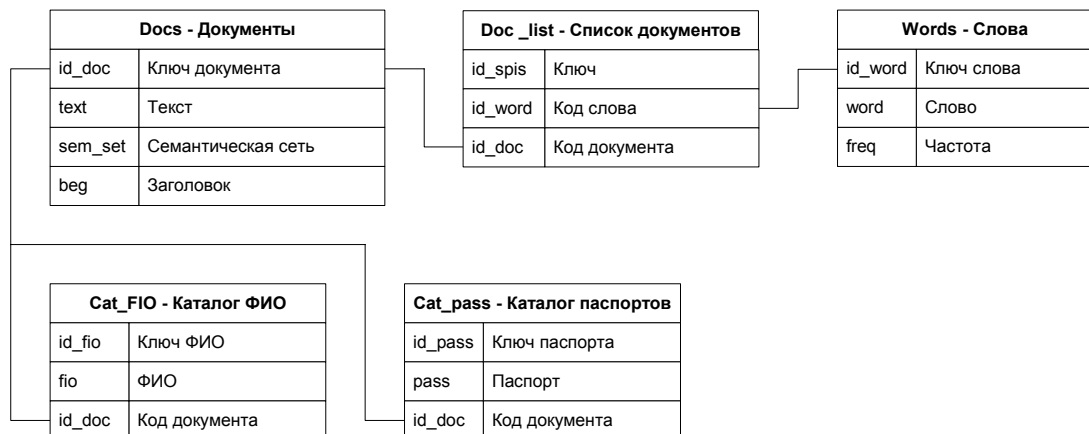
Файл	Содержание	Атрибуты
DB_TXT.db	Текст документа	Текст
		Длина
DB_TXT.dbi	Типизированный текст	Номер документа
		Смещение
		Размер
DB_ZZZ.db	Семантическая сеть документа	Семантическая сеть
		Длина
DB_ZZZ.dbi	Типизированная семантическая сеть	Номер семантической сети
		Смещение
		Размер
NET2.slv	Индексный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

4. 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.) [5]. Особенностью предлагаемого решения является отказ от хранения структур знаний в плоских файлах.

Поскольку быстрый поиск является основным преимуществом СУБД, 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, поскольку они являются характеристикой одной и той же сущности «Документ» (Табл. 2). Для ускорения работы следует отдельно хранить заголовок документа, поскольку 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 для того, 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, преобразовывая 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 на Табл. 2.

Табл 2. Схема БЗ в современной СУБД



Отметим, что все аналитические модули, написанные на Декл, также как и прежде, работают только с РСС, загруженными в активную часть БЗ. Изменения при переносе хранилища касаются только способа попадания РСС в активную часть. При хранении БЗ в плоских файлах подкачка РСС в эту часть БЗ осуществляется следующим образом:

- 1) Выбирается документ.
- 2) Из типизированного файла DB_ZZZ.dbi считывается информация о смещении начала РСС соответствующего документа в файле с семантическими сетями DB_ZZZ.db
- 3) Из файла DB_ZZZ.db, извлекается РСС этого документа.
- 4) РСС помещается в активную часть БЗ.

При увеличении размеров базы существенно возрастает время считывания информации из файлов. После переноса хранилища из плоских файлов в СУБД поиск РСС осуществляется при помощи простого запроса к таблице Docs. РСС всего документа хранится в одной записи таблицы в том же виде, что и в плоских файлах (см. п. 2.1). При этом для ускорения поиска применяется индексирование этой таблицы в СУБД. Из результата запроса содержимое поля sem_set - РСС документа, включающие индексы семантических сетей - загружается в активную часть БЗ. Эти индексы используются Деклом в аналитических режимах.

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 возникает возможность подключения к системе внешних баз данных. Обратим внимание на преимущества, которые дает нам применение 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 этого существенно повышается надежность системы в целом.

5. Взаимодействие с внешними БД

Выше рассматривалась задача организации внешнего хранилища знаний (внешней БЗ) в современной СУБД. Другая задача связана с подключением в качестве внешнего источника данных внешней БД (например, МГТС, ГИБДД, МТС и т.п.). Все базы, 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 подключение простой внешней базы базы МГТС.

Структура такой БД выглядит так:

Табл. 3. Структура таблицы БД МГТС с информацией по физическим лицам.

Телефон
Город
Улица
Дом
Буква
Дробь
Корпус
Строение
Квартира
Фамилия или название организации
Имя и Отчество (полностью)
Паспортные данные
Комментарии по абоненту
Комментарии по телефону

Табл. 4. Структура таблицы БД МГТС с информацией по юридическим лицам.

Номер телефона
Принадлежность к организации

Шаги, которые необходимо предпринять для подключения такой базы к внешней БЗ системы, реализованной в СУБД Oracle:

- 1) Загружаем БД МГТС в Oracle. В Приложение 1 описано несколько методов «заливания» в Oracle таких баз.
- 2) Обеспечиваем связь (шаблон) между объектами системы и объектами базы МГТС.
- 3) Обеспечиваем сохранение этих связей (шаблонов) в системе.
- 4) Реализуем механизм запроса информации по шаблонам во внешние базы.
- 5) Визуализируем результат поиска по запросу и сохраняем этот результат.

6. Подключение внешней БД на примере базы МГТС.

Рассмотрим перечисленные выше шаги на примере базы МГТС.

1) Имеющаяся в наличии база МГТС представляет собой один файл БД Access размером 1,5 гигабайта. В этом файле находится 8 таблиц, в каждой из которых примерно по 300 тысяч записей. Кроме этого некоторые из таблиц имеют разные структуры (Табл. 3, 4). Для того чтобы загрузить эту базу в Oracle и в дальнейшем использовать ее в системе, необходим конвертор, который позволил бы загрузить все таблицы из ACCESS в одну таблицу Oracle. Этот вариант является самым трудоемким, но, учитывая ценность данных базы МГТС, необходимо конвертировать данные, исходя из удобства дальнейшего использования.

Табл. 5. Структура базы МГТС в Oracle.

name	type
Id (PK)	integer
ind	varchar(20)
telefon	varchar(20)
city	varchar(30)
street	varchar(100)
house	varchar(5)
letter	varchar(5)
drob	varchar(5)
korpus	varchar(5)
building	varchar(5)
flat	varchar(5)
family_orgname	varchar(80)
fullname	varchar(100)
passdata	varchar(80)

2) Далее необходимо привести в соответствие объекты системы и поля БД МГТС. Как видно из структуры базы МГТС, одному объекту системы ФИО соответствует два поля базы МГТС, а одному объекту Адрес – восемь полей базы МГТС. Настройка шаблонов должна позволять ставить в соответствие одному объекту системы несколько полей внешней БД. Соответствие для базы МГТС приведено в табл. 6.

Табл. 6. Соответствие объектов системы полям БД МГТС

АНАЛИТИК	МГТС
Телефон	Телефон
Адрес	Город
	Улица
	Дом
	Буква
	Дробь
	Корпус
	Строение
	Квартира
ФИО	Фамилия или название организации
	Имя и Отчество (полностью)
Паспорт	Паспортные данные
Не используются	Комментарии по абоненту
	Комментарии по телефону

3) Далее встает вопрос сохранения введенного шаблона в системе. Для сохранения шаблона можно предложить два способа.

- Записывать определенную структуру, содержащую формат шаблона, в текстовый файл.
- Записывать определенную структуру в таблицу Oracle.

Второй способ предпочтительней по той причине, что одна из основных целей разработки является перенос хранилища данных из плоских файлов в СУБД. Структура соответствующей таблицы будет выглядеть следующим образом.

Табл.7. Таблица «Шаблоны».

name	type
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, для того, чтобы не давать пользователю заводить несколько шаблонов с одним и тем же именем.

4) В случае, когда у аналитика возникнет необходимость поиска важной информации по внешней базе МГТС, необходимо выполнить в следующую последовательность действий:

- Определение значения объекта системы, которое будет использоваться в качестве параметров запроса
- Определение объектов системы, значения которых необходимо узнать, используя внешнюю БД.
- создание запроса к внешней БД, при помощи ранее созданного шаблона.

Например, пусть известно значение объекта «Телефон». Оно равно «123-45-67». Необходимо узнать значение объекта «ФИО». Тогда, используя шаблон «MGTS», необходимо сформировать следующий запрос [6] 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 при 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, который необходим для того, чтобы осуществить связку между таблицами на уровне языка SQL. Реализация такой задачи представляется крайне сложной в исполнении и нецелесообразной. Гораздо проще позволить пользователю вводить запрос, который

связывал бы объекты системы и поля сложной БД на языке SQL самостоятельно. Именно для этого в таблицу «Шаблоны» было добавлено поле «Запрос».

5) После того, как требуемая информация найдена, необходимо ее визуализировать и сохранить. Визуализация объектов системе 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 сохранения информации в системе.

Еще раз отметим, что для создания шаблонов, обеспечивающих связь системы с внешними БД, используется специальная прослойка, которая называется редактор шаблонов. В этом редакторе происходит создание шаблонов соответствия между объектами системы и объектами внешних БД. При этом для одной внешней БД, в зависимости от сложности, может создаваться один или более шаблонов соответствия. Отметим так же, что найденная во внешних БД информация преобразовывается в РСС и подкачивается в ОП, дополняя активную часть БЗ. Эта информация в виде РСС сохраняется в БЗ системы. Таким образом, происходит увеличение пространства поиска системы в целом.

7. Способы сохранения в системе информации

После того, как на экране была отображена информация, найденная в результате поиска по внешней БД, встает вопрос, что делать с ней дальше. Ответ на этот вопрос напрямую связан с интенсивностью, с которой нам необходимо обращаться к той или иной внешней базе. Рассмотрим три возможных способа работы с найденной во внешних базах информацией.

1) Вся полученная (по запросу) информация в процессе анализа преобразуется в семантическую сеть (РСС). Возникает возможность добавлять найденную информацию во внешнюю БЗ системы, где она бы хранилась как обычный документ. Здесь вопрос об интенсивности использования базы особенно актуален. Если преобразовать базу МГТС в БЗ системы, то, с одной стороны, она будет занимать очень много места, с другой стороны, возможности и скорость поиска значительно возрастут. При этом нет необходимости каждый раз обращаться к внешней базе за информацией.

2) На всю требуемую и найденную информацию в системе проставляются ссылки на местонахождение этой информации. Такой способ позволит значительно сократить занимаемое результатами поиска место, при этом скорость поиска будет меньше, чем при использовании первого способа. Недостаток этого способа заключается в сложности разработки такого механизма. Перед разработчиками встанет вопрос о виде ссылок и о методе их хранения. Приходится отслеживать «битые» ссылки, которые неминуемо будут возникать при удалении данных из внешних баз, придется реализовывать интерфейсы и механизмы работы с этими ссылками (удаление, редактирование ссылок и т.п.). Такой метод хранения результатов поиска эффективен как в случае частого, так и в случае редкого использования внешней БД.

3) Каждый раз при новом запросе идет обращение по шаблону к внешней БД. Результаты запросов не сохраняются в системе. Положительная сторона этого решения заключается в экономии места на жестком диске. Этот способ не эффективен, если необходимо часто обращаться к внешней БД. Если же обращение идет редко, то тот факт, что результаты не сохраняются, имеет ряд положительных моментов: результаты запросов не дублируются во внешнюю Базу Знаний, не надо реализовывать механизмы работ с этими результатами.

В данный момент времени работа с внешними БД реализована по третьему способу. В ближайшее время планируется реализация первого способа, когда результаты запросов будут сохраняться в системе. Тогда у пользователя будет возможность самостоятельно принимать решения о том, сохранять или не сохранять результаты запросов в основную БД системы.

8. Заключение

В статье описываются методы модернизации функционала системы АНАЛИТИК в соответствии с основными требованиями к подобному классу современных программных продуктов. Показано: что использование современных СУБД позволяет повысить производительность и надежность системы, уменьшить ограничения на объем 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. Говорить о том, что была описана Система Управления Базами Знаний (СУБЗ) нельзя по той причине, что функцию управления Знаниями выполняет Декл, в то время как СУБД выполняет лишь роль хранилища данных.

Систему такого класса можно с уверенностью назвать логико-аналитической. Ее особенность: наличие лингвистического процессора, возможности автоматизированной обработки неструктурированной информации, возможности создания режимов аналитической обработки и отображения информации, а так же 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.

Приложение 1. Некоторые способы загрузки внешних источников данных в СУБД Oracle.

- 1) Если источник хранился в СУБД Oracle, загрузка осуществляется с помощью внутреннего механизма экспорта и импорта схем данных. Т.е. достаточно сделать экспорт необходимой таблицы из исходной схемы в схему системы Аналитик [2].
- 2) Если источник данных хранится в другой распространенной СУБД, существует возможность использования механизмов соединения баз данных Borland Database Engine или DBExpress, реализованные компанией Borland. С помощью этих механизмов можно, написав соответствующий конвертер, осуществить загрузку данных в СУБД Oracle.
- 3) Существует несколько механизмов соединения баз данных, реализованных в Windows: ADO (ActiveX Data Objects), здесь доступ к данным осуществляется при помощи механизма OLE DB (Object Linking and Embedding Data Base), или ODBC (Open Database Connectivity) [3].
- 4) Кроме этого, компании-разработчики СУБД создают специальные драйверы для прямого доступа к базам данных этих СУБД. Этот способ отличается от выше описанных способов соединения с БД более высокой скоростью соединения, надежностью, т.к. эти драйверы создаются специально для соединений с конкретной СУБД и учитывают все ее особенности.
- 5) Самым сложным является вариант, когда источник данных хранится в специфической БД, для которой не предусмотрен ни один из вышеописанных методов соединения. В этом случае необходимо механизмами этой БД реализовать импорт данных в текстовый файл, после этого осуществить импорт данных в Oracle с помощью одной из его утилит – sqlloader. Эта утилита обеспечивает «заливку» данных в Oracle из текстовых файлов с разделителями.
- 6) Если же представление данных в текстовом виде невозможно, то остается создать драйвер для одного из вышеописанных механизмов (например, для ODBC или для прямого доступа).

Литература.

1. Кузнецов И.П. Продукционный язык программирования Декл. Система обработки Декларативных структур знаний Деклар-2. – М.: Издательство «ИПИРАН» 1988 г.
2. Couchman J., Schwinn U., Oracle 8i Certified Professional DBA – М.: Издательство «Лори», 2002 г.
3. Хомоненко А.Д., Гофман В.Э., Мещеряков Е.В. и др., «Delphi 7, наиболее полное руководство» – Спб.: Издательство «ВНУ», - с. 488
4. Рабинович Б.И. Аналитическая система обработки и управления структурированной информацией. Интеллектуальные технологии и система. Выпуск 6. – М.: Издательство ООО «Эликс+», 2003 г, стр. 173-186.
5. КОАР Open Portal. К базам данных завтрашнего дня. [Электронный ресурс]/ Статья. 2000 г. Блэкфорд Д., Стрехлоу К. – Режим доступа: <http://koapp.narod.ru/tehlit/base/bd/06db.htm>; свободный – Загл. с экрана. – Яз. рус., англ.
6. Центр Информационных Технологий. Введение в стандарты языка баз данных SQL [Электронный ресурс]/ Статья.1998 г. Кузнецов С.Д – Режим доступа: <http://www.citforum.ru/database/sqlbook/index.shtml>; свободный – Загл. с экрана. – Яз. рус., англ.
7. Кузнецов И.П., Мацкевич А.Г. Особенности организации базы предметных и лингвистических знаний в системе АНАЛИТИК. Труды международной конференции Диалог'2003 - М.: Издательство «Наука», 2003.
8. Рабинович Б.И. Организация баз знаний в современных СУБД. Проблемы и методы информатики. II Научная сессия Института проблем информатики Российской академии наук. Москва, 18-22 апреля 2005 г. Тезисы докладов. – М.: Издательство «ИПИ РАН», 2005 г.
9. Григорьев Ю.А., Ревунков Г.И., Банки данных: Учеб. для вузов – М.: Издательство «МГТУ им. Н.Э. Баумана», 2002 г, стр. 126.
10. Шарнин М.М, Кузнецов И.П. Продукционный язык программирования Декл . В сб. «Система обработки Декларативных структур знаний Деклар-2» - М.: Издательство «ИПИРАН», 1988 г. – с. 134-152.
11. Литвак Б.Г. Экспертная информация. Методы получения и анализа. – М.: Издательство «Радио и Связь», 1982 г.
12. Гаврилова Т.А., Хорошевский В.Ф. Базы знаний интеллектуальных систем. – СпБ: издательство «Питер», 2000 г.
13. Попов Э.В. Общение с ЭВМ на естественном языке. – М.: Издательство «Наука», 1986 г.
14. А.Тей, П. Грибомон, Ж. Луи. Логический подход к искусственному интеллекту: от классической логики к логическому программированию: Пер с франц./Тейз А., Грибомон П., Луи Ж. И др. – М.: Издательство «Мир», 1990. – 432 с.
15. Маклаков С.В., «BPWIN и ERWIN: CASE-средства для разработки информационных систем». – М.: Издательство «Диалог-МИФИ», 1999 г., 256 стр.
16. Кузнецов И.П. Семантические представления. – М.: Издательство «Наука», 1986 г.
17. Кузнецов И.П., Мацкевич А.Г. Лингвистический процессор для автоматического выявления из текстов значимой информации с ее компоновкой в рамках указанных шаблонов. Труды межд. Семинара Диалог 2001 по комп. лингвистике и её приложениям: Т.2. – М.: Издательство «Наука» 2001 г. – с. 134-137.