

МЕТАОПИСАНИЕ БАЗ ДАННЫХ КАК ОСНОВА ИНТЕГРАЦИИ ИНФОРМАЦИОННО-СПРАВОЧНЫХ СИСТЕМ И ГИС

И. В. БЫЧКОВ, Е. С. ФЕРЕФЕРОВ, А. Е. ХМЕЛЬНОВ

Институт динамики систем и теории управления СО РАН

Иркутск, Россия

e-mail: fereferov@icc.ru

The problem of integration of already existing databases (DBs) and geoinformation systems (GIS) is considered. The proposed solution is based on a metadata description technology developed in ISDCT SB RAS. The technology was applied to construct GIS-software (GeoArm) that could be adjusted to access existing DBs, extending the original software, which uses the DBs, with new GIS-functions. The GeoArm workbench has been applied to the following projects: “Register of metal garages at Irkutsk municipal territory”, “Register of advertisement constructions at Irkutsk municipal territory”.

После многолетних усилий по внедрению информационных технологий информационно-аналитическая среда крупных организаций и предприятий часто представляет собой набор автоматизированных рабочих мест (АРМ), реализованных в разных системах программирования и работающих с различными СУБД. Базы данных (БД) этих АРМ обычно содержат какие-либо сведения об объектах, имеющих пространственную привязку (почтовые адреса, координаты, описание расположения объектов и т. д.). Для анализа пространственной информации предназначен специальный класс программ — геоинформационные системы (ГИС). Как правило, применение ГИС повышает производительность, уровень автоматизации и качества получаемых решений, а также расширяет круг специалистов, использующих информационные технологии в своей производственной деятельности. Поэтому в том случае, когда в имеющихся АРМ взаимодействие с ГИС не предусмотрено, может быть поставлена задача расширения их функциональных возможностей при помощи интеграции с ГИС, чтобы привлечь современные геоинформационные технологии для решения специализированных задач.

Однако включение кода взаимодействия с ГИС в каждое из уже существующих АРМ — задача трудоемкая, требующая для своего решения обязательного наличия исходных текстов АРМ, освоения разработчиками внедренных АРМ геоинформационных технологий или, наоборот, изучения специалистами в области ГИС исходных текстов уже существующих систем.

Один из подходов к решению рассматриваемой задачи — разработка специализированных приложений на базе популярных ГИС, таких как ArcView [1] или MapInfo

[2]. В этих системах предусмотрена работа с таблицами БД наряду с семантическими таблицами, связанными со слоями карт, а также существует возможность построения запросов, связывающих между собой несколько таблиц и картографических слоев. Однако воспользоваться этими возможностями может лишь опытный пользователь рассматриваемых ГИС, при этом даже для привязки к карте небольшого набора таблиц необходимо выполнить много операций. Кроме того, в большинстве ГИС механизм создания пользовательских форм для отображения записей из БД или не предусмотрен вообще, или связан с еще более сложным программированием. Таким образом, имеющиеся в рассматриваемых ГИС возможности по привязке информации из БД к карте не позволяют решить поставленную задачу в полном объеме, без трудоемкой разработки специализированных приложений, работающих под управлением соответствующих ГИС.

В качестве возможной альтернативы полномасштабной разработке приложения на внутреннем языке ГИС рассмотрим ГИС “Карта-2000” [3]. В данную ГИС встроена подсистема управления пользовательской БД — “Проект БД”. Эта подсистема предназначена для работы с данными, хранящимися во внешних БД. Она позволяет описывать совокупность отдельных таблиц, способов их отображения и редактирования. Для построения пользовательских запросов в модуле “Проект БД” требуется знание встроеного в систему языка построения запросов (на уровне оператора SELECT языка SQL). Полученные в результате тексты запросов можно использовать при построении интерфейса системы (пользовательских форм). Для создания форм применяется визуальное программирование, реализация этого механизма практически полностью копирует технику визуального программирования в Delphi и C++ Builder, основное отличие — лишь в отсутствии возможности задания кода для обработки событий. При этом механизм настройки взаимодействия с БД также копирует работу в Delphi и C++ Builder: для отображения информации из таблицы или запроса в полях формы необходимо описать набор данных, настроить источник данных и связать источник данных с визуальными компонентами [4]. Кроме настраиваемых пользователем форм, в данной системе поддерживаются создаваемые автоматически стандартные формы, однако их можно применять лишь для простых таблиц, не содержащих ссылок на справочники, поскольку в стандартных формах не поддерживается отображение информации из справочников (работа с LookUp-полями). Таким образом, создание проекта для отображения информации из БД в модуле “Проект БД” по своей трудоемкости несколько не уступает задаче создания соответствующего приложения в Delphi и C++ Builder и требует достаточно высокой квалификации пользователя, сравнимой с уровнем профессионального программиста.

Учитывая вышесказанное, можно сделать вывод, что при использовании возможностей существующих ГИС задача интеграции с уже созданными системами, использующими БД, оказывается почти столь же трудоемкой, как и разработка нового АРМ с нуля.

1. Предлагаемое решение

Для решения задачи привязки к карте информации, хранящейся в уже существующих БД, была разработана система ГеоАРМ, позволяющая воспроизвести основные функции имеющихся АРМ на основе оригинальной технологии обработки метаданных, при этом обеспечивается взаимодействие с ГИС.

Предложенная технология обработки информации в БД с их метаописанием базируется на принципах, апробированных в ИДСТУ СО РАН при создании информационно-аналитических систем (ИАС) с использованием ГИС, функционирующих в среде Интернет/интранет, которые предназначены для публикации без возможности редактирования (в режиме “только чтение”) связанных между собой картографической информации (геоданных) и информации из БД [5]. Для создания ИАС был разработан модуль публикации в Интернет/интранет информации из БД, настраиваемый на работу с конкретными данными при помощи метаописаний их структуры и шаблонов пользовательских форм (страниц), в котором также предусмотрено взаимодействие с модулями публикации геоданных.

Впоследствии было реализовано настраиваемое АРМ, позволяющее не только просматривать, но и редактировать информацию, содержащуюся в БД [6]. В настраиваемом АРМ метаописания используются для автоматической генерации полнофункциональных форм редактирования данных. Автоматически генерируемый по метаописаниям интерфейс позволяет воспроизвести значительную часть возможностей типичного АРМ, предназначенного для работы с описанными данными.

Разработка настраиваемого АРМ позволила перейти к созданию системы ГеоАРМ, которая обеспечивает взаимодействие с наиболее распространенными СУБД и содержит основные средства для работы с геоданными. Система ГеоАРМ поддерживает операции поиска, просмотра и редактирования информации в существующих БД, обеспечивая при этом возможность отображения этой информации на карте.

В системе ГеоАРМ реализованы следующие функции:

- формирование и отображение электронной карты местности;
- поиск объектов на карте по почтовому адресу;
- просмотр и редактирование содержимого БД;
- связь БД с электронной картой;
- ведение тематических слоев на карте местности с привязкой к БД;
- автоматизация администрирования описаний БД и электронных карт.

Кратко рассмотрим особенности реализованных функций.

Система ГеоАРМ включает в себя картографическую подсистему, реализованную при помощи пакета GIS ToolKit [7] из ГИС Панорама. Для поиска на карте объектов в соответствии с почтовыми адресами поддерживается функция *Адресный план города*: окно отображения карты содержит поля ввода для выбора имеющихся на карте домов по наименованию улицы и номеру дома.

Система ГеоАРМ может быть настроена для работы с уже существующими АРМ на основе описаний структуры БД. При этом обеспечивается просмотр и редактирование содержимого БД как в табличном представлении, так и в виде автоматически сгенерированных взаимосвязанных форм. Также для данных из таблиц могут быть сделаны выборки при помощи формируемых пользователем запросов.

Записи из таблиц БД можно привязывать к карте по содержащимся в их полях почтовым адресам или по кодам объектов. Привязка по почтовым адресам осуществляется при помощи функции адресного плана. Привязка по целочисленным кодам (первичным ключам) выполняется при помощи запоминания в специальную таблицу привязки к карте информации о соответствии записи объектам карты. При наличии информации о привязке объектов к карте становится возможным находить на карте объекты, соответствующие записям таблицы, и, наоборот, находить связанную с объектом карты информацию в БД. Кроме того, реализована функция автоматического нанесения на

карту объектов по содержащимся в БД координатам.

На основную карту местности (топооснову) могут дополнительно накладываться тематические слои, которые доступны для редактирования пользователем. На таких слоях можно создавать объекты карты, соответствующие имеющейся в БД информации, для которой отсутствуют объекты на топооснове.

```
[TABLES] ;Раздел описания таблиц
...
SP_T_UL= ;В базе данных имеется таблица SP_T_UL (справочник типов улиц)
SP_S_UL= ;В базе данных имеется таблица SP_S_UL (старых названий улиц)
SP_UL= ;В базе данных имеется таблица SP_UL (справочник улиц)
SP_ADR= ;В базе данных имеется таблица SP_ADR (справочник адресов)
vSP_ADR=v.SP_ADR ;На базе таблицы SP_ADR строится представление vSP_ADR
...
[TBL_MENU] ;Раздел описания таблиц и представлений доступных пользователю
; для непосредственного выбора из меню
vSP_ADR= ;В качестве названия пункта меню для представления vSP_ADR
;используется имя самого представления
vGarMG=Металлические гаражи ;Пункт меню для таблицы vGarMG называется
;«Металлические гаражи»
vGarMGReg=Регистрация металлических гаражей

[TBLF_SP_S_UL] ;Раздел описание полей таблицы SP_S_UL
KOD_UL=P ;P - целочисленное поле, являющееся первичным ключом таблицы
DAT_PEREIM=S ;S - строковое поле
UL_STAR=S

[TBLF_SP_UL] ;Раздел описание полей таблицы SP_UL
KOD_UL=P
UL=N ;N - именованное поле, т.е. символьное поле,
;содержащее уникальные для таблицы значения, которое
;потом может использоваться для выбора записи
KOD_TIP_UL=^SP_T_UL ;^ - целочисленное поле, содержащее ссылку на
;первичный ключ таблицы, т.е. поле KOD_TIP_UL ссылается
;на первичный ключ таблицы SP_T_UL
KOD_NPUN=^SP_NPUN ;Поле KOD_NPUN ссылается на первичный ключ
;таблицы SP_NPUN

[VF_vSP_ADR] ;Раздел описания представления vSP_ADR
KOD_DOM_G= ; В качестве поля KOD_DOM_G в представление включается одноимённое
; поле базовой таблицы
NPUN=KOD_NPUN.NPUN ;В качестве поля NPUN в представление включается поле
NPUN
;из той таблицы (SP_NPUN), на которую указывает ссылка
;KOD_NPUN из базовой таблицы
NRAI=KOD_NRAI.RAION ;В качестве поля NRAI в представление включается поле
;RAION из той таблицы (SP_RAION), на которую указывает
;ссылка KOD_NRAI из базовой таблицы
TIP_UL=KOD_UL.KOD_TIP_UL.TIP_UL ;Поле TIP_UL включается в представление через
;два уровня связей
UL=KOD_UL.UL ;В качестве поля UL в представление включается поле UL из той
;таблицы (SP_UL), на которую указывает ссылка KOD_UL из базовой
;таблицы
...
```

Рис. 1. Фрагмент файла описания структуры БД

Система ГеоАРМ взаимодействует с БД, используя файлы описаний структуры БД, которые являются текстовыми INI-файлами (рис. 1) и могут быть созданы и изменены в любом текстовом редакторе. В систему включены формы, позволяющие создавать и редактировать основные элементы файлов описаний интерактивно.

В готовый файл описания структуры БД, для повышения уровня безопасности функционирования АРМ, может быть записан признак завершенности, после считывания которого все административные функции в системе блокируются.

Система ГеоАРМ состоит из двух основных подсистем: настраиваемого АРМ и картографического модуля. Рассмотрим более подробно функции каждой из подсистем.

2. Настраиваемое АРМ

Настраиваемое АРМ включает в себя модуль просмотра и редактирования содержимого базы данных. Система ГеоАРМ может быть настроена без перекомпиляции на работу с любыми БД, доступ к которым обеспечивается на основе BDE (Borland Database Engine) при условии, что эти БД имеют следующую структуру: в качестве первичных ключей основных таблиц должны использоваться целочисленные поля, а связи между таблицами должны определяться как список соответствий между отдельными полями таблиц. Для таких БД можно организовать работу системы ГеоАРМ непосредственно с той же базой, с которой работают уже внедренные АРМ. Если же структура существующей БД препятствует ее непосредственному использованию в системе, то можно использовать периодическое преобразование такой БД в структуру, соответствующую вышеперечисленным требованиям.

В основе настраиваемого АРМ лежит разработанная в ИДСТУ СО РАН технология метаописаний БД, которая позволяет единообразно решать различные задачи обработки данных с использованием описаний структуры БД.

Метаописания БД. В описание БД входят описания *таблиц* и *представлений*. В описании таблицы указывается, какие поля таблицы БД используются в системе (некоторые существующие поля могут быть исключены из описания, тогда они игнорируются системой), какими свойствами с точки зрения системы они обладают и какие связи существуют между данной таблицей и другими таблицами.

Под *связью*, или *ссылкой*, таблицы **A** и таблицы **B** понимается наличие в одном или нескольких полях таблицы **A** уникального для таблицы **B** набора значений одного или нескольких ее полей. При этом говорят о наличии *внешнего ключа* [8] в таблице **A**. Для того чтобы определить связь в описании, не обязательно существование соответствующего внешнего ключа (FOREIGN KEY) в БД. Таблицу **A** будем называть *исходной* таблицей связи, а таблицу **B** — *целевой*. Поля, по которым устанавливается связь из исходной таблицы, назовем *ссылочными* полями. Поля целевой таблицы, по которым устанавливается связь, должны удовлетворять свойству *уникальности*, т. е. в целевой таблице не должно существовать двух записей, содержащих одинаковый набор значений в этих полях. Поэтому, как правило, связь устанавливается по полям, входящим в *первичный ключ* целевой таблицы. При наличии ссылки каждой записи исходной таблицы сопоставляется не более одной записи целевой таблицы. В отличие от ограничений FOREIGN KEY, допускается наличие в полях исходной таблицы значений, которым не сопоставляется запись в целевой таблице. Данная возможность позволяет работать и в тех случаях, когда, например, в зависимости от значения поля **Т.А**, содержащего

информацию о типе ссылки, значение поля **Т.В** интерпретируется как код из соответствующей типу ссылки таблицы (рис. 2).

Доступная по ссылке таблица может содержать ссылки на другие таблицы, которые в свою очередь могут содержать ссылки на таблицы, и т. д. При этом будем говорить о доступе к полям таблиц через несколько уровней ссылок.

При описании поля таблицы указывается его вид, который связан с типом поля в БД но, кроме того, несет и дополнительную информацию о возможных способах использования данного поля в системе.

Различаются следующие основные виды полей:

целочисленное — поля всех целочисленных типов;

числовое — все типы с плавающей точкой и числовые типы, имеющие знаки после запятой;

строковое — обычные символьные поля, содержащие произвольный набор значений;

именованное — символьное поле, содержащее уникальные для таблицы значения, которое поэтому может использоваться для выбора записи;

списочное — символьное поле, содержащее значения из ограниченного набора;

логическое — поле, содержащее значения ИСТИНА или ЛОЖЬ;

ссылочное — целочисленное поле, содержащее ссылку на первичный ключ некоторой таблицы (более сложные способы задания ссылок — по нескольким полям, с использованием других видов полей и т. д. — также могут быть явно описаны в другой секции файла).

При этом, с одной стороны, разным типам полей может сопоставляться один и тот же вид. Например, все целочисленные типы полей, независимо от их размера, относятся к целочисленному виду. С другой стороны, разные поля одного типа могут быть отнесены к разным видам. Например, строковое поле может быть отнесено к строковому, списочному или именованному виду, в зависимости от его роли в БД.

Описание представления строится начиная с одной *базовой* таблицы, при этом в него могут быть включены поля других таблиц, связанных с базовой (может быть через несколько уровней ссылок). Метаописание содержит следующую информацию: на какой таблице основано данное представление и какие поля этой таблицы и таблиц, связанных с ней, включаются в это представление. Таким образом, можно считать, что представления в описании БД отличаются от обычных представлений (VIEW) тем, что их структура ограничена определенным видом, предназначенным для декодирования значений кодовых полей через справочники.

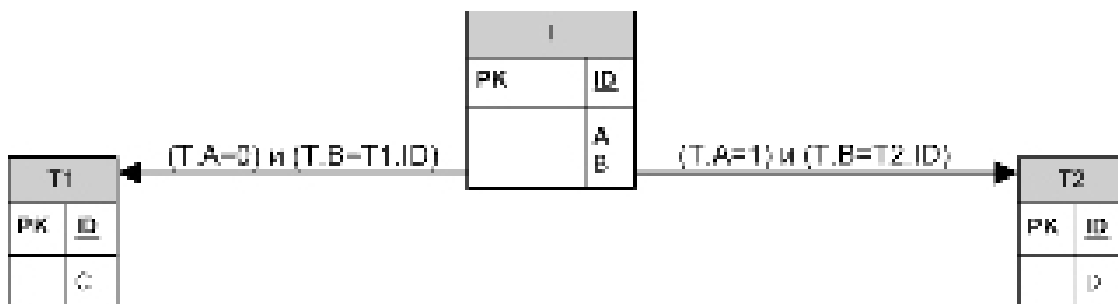


Рис. 2. Пример связей между таблицами, которые нельзя описать как ограничение FOREIGN KEY

После создания описания структуры БД можно получить доступ к табличным формам для просмотра и редактирования содержимого таблиц и представлений БД (рис. 3). При редактировании представления в табличной форме значения полей, получаемых из справочников, становятся LookUp-полями, т. е. позволяют изменять значения поля базовой таблицы с кодом-ссылкой на справочник посредством выбора из выпадающего списка соответствующего ему в справочнике значения (например, поле “Район” на рис. 3).

Кроме того, описанные таблицы и представления можно отобразить и редактировать в виде формы (рис. 4), при этом поля, включенные в представление из справочников, также отображаются как LookUp-поля, если они получены по ссылке непосредственно из базовой таблицы. Форма для редактирования таблицы или представления генерируется автоматически в соответствии с описанием. Поля, получаемые по общей ссылке, автоматически объединяются в группы (см., например, поля “Тип ул.” и “Улица” на

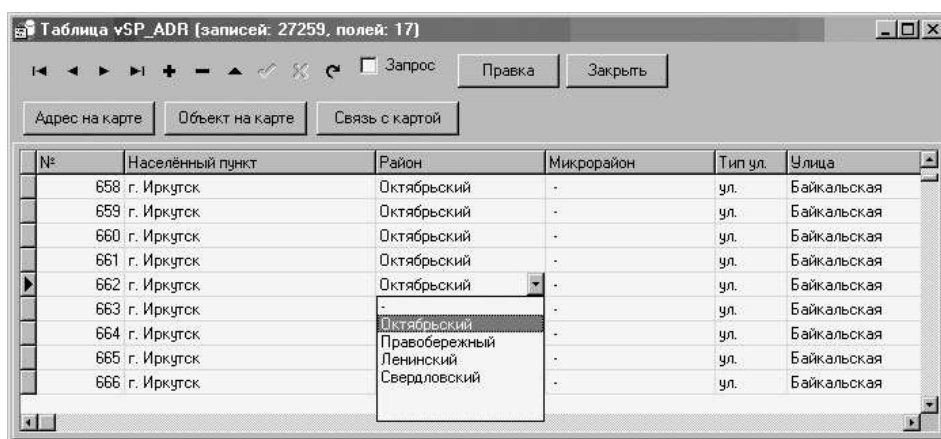


Рис. 3. Редактирование представления в табличной форме

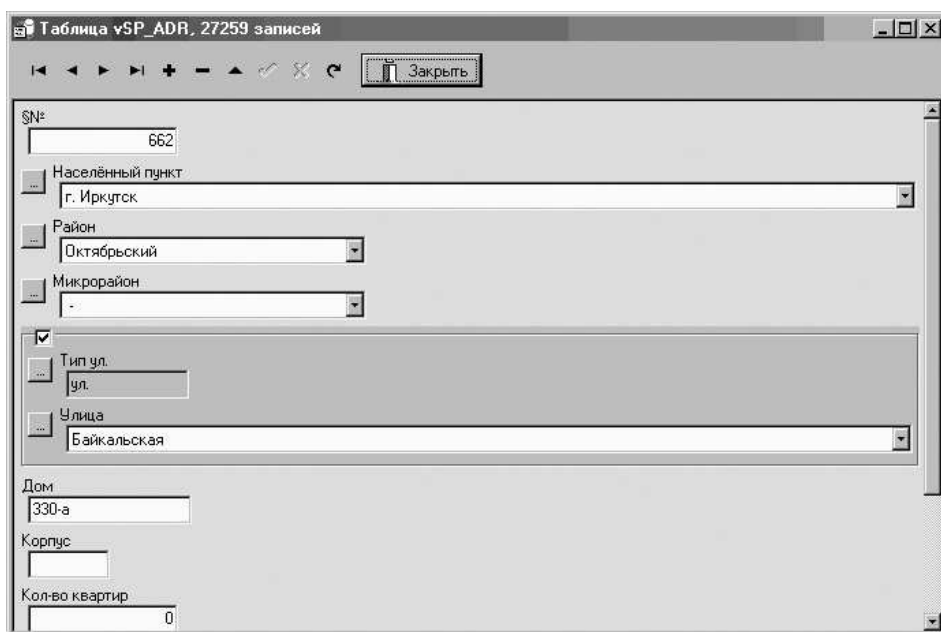


Рис. 4. Редактирование записи представления в виде формы

рис. 4), каждой ссылке соответствует кнопка перехода к форме для редактирования записи, на которую указывает ссылка.

Построитель запросов. В системе ГеоАРМ реализована возможность формирования пользовательских запросов. Пользователь может самостоятельно сформулировать условие запроса для отбора записей при просмотре таблицы или представления в табличной форме.

Редактор пользовательских запросов позволяет пользователю, не специалисту в программировании для баз данных, *интерактивно* сформулировать *произвольное* логическое условие для выбора записей таблицы или представления. При этом полностью *исключается* возможность создания запросов, содержащих ошибки с точки зрения организации данных.

В большинстве программных систем построитель произвольных запросов реализуется как текстовый редактор с набором элементов управления для вставки в текст имен полей, их значений и операторов. Место и последовательность вставки фрагментов текста при этом не контролируется системой, поэтому легко может быть создан текст, содержащий ошибки. При таком подходе надо учитывать, что любое сообщение об ошибке, возникшей при обработке запроса, способно поставить в тупик неподготовленного пользователя: он просто не поймет, что ему необходимо исправить, чтобы запрос отработал в соответствии с поставленным заданием. При предлагаемом интерактивном построении запросов все сформулированные пользователем запросы система будет способна выполнить. При этом у пользователя появляется возможность просмотреть результаты выполнения составленного им запроса, сравнить их с тем, что он ожидал получить, и, при наличии расхождений, попытаться их исправить.

В окне построителя запроса логическое выражение для условия запроса представлено в виде дерева разбора с корнем, расположенным слева (рис. 5.). Выражение представляет собой набор *операций сравнения*, связанных между собой логическими связками И, ИЛИ либо их отрицаниями. Под отрицанием логической связки понимается отрица-

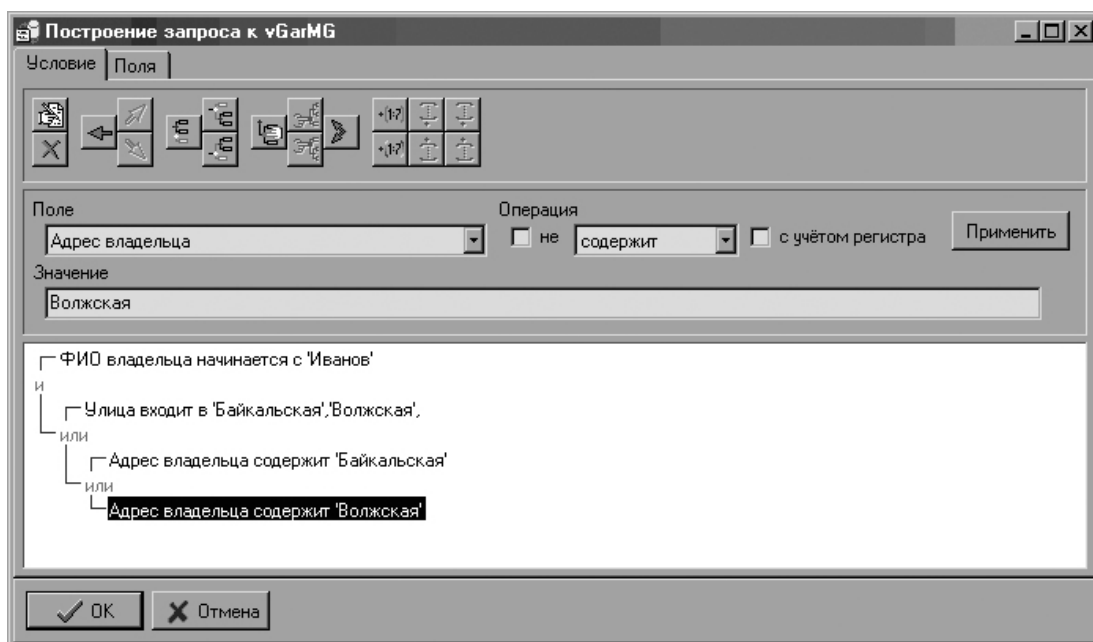


Рис. 5. Окно построителя запросов

ние результата вычисления со связкой без отрицания. Например, НЕ И(А,В) означает НЕ(А и В). Операции сравнения задают условия на отдельные поля таблицы или представления.

Чем выше в дереве разбора находится логическая операция, тем позже она применяется, т. е. сначала вычисляются значения для находящихся под некоторым узлом дерева поддеревьев, а затем к результатам вычислений применяется операция, стоящая в данном узле. Так, для дерева, представленного на рис. 5, сначала вычисляются условия на поле “Адрес владельца”, затем к результатам применяется логическая операция ИЛИ, потом вычисляется условие на поле “Улица”, к результатам применяется логическая операция ИЛИ и, наконец, вычисляется условие на поле “ФИО владельца” и к результатам применяется операция И. В конце вычисляется выражение:

(ФИО владельца начинается с 'Иванов') И
(Улица входит в 'Байкальская', 'Волжская' ИЛИ
(Адрес владельца содержит 'Байкальская' ИЛИ
Адрес владельца содержит 'Волжская'))

Таким образом, на рис. 5 представлен запрос для выбора всех владельцев гаражей с фамилией Иванов, проживающих на улице Байкальской или Волжской либо имеющих гаражи на этих улицах.

3. Картографический модуль

Данный модуль позволяет просматривать картографическую информацию, находить на карте объекты и адреса, информация о которых содержится в БД, и, наоборот, находить в БД информацию, связанную с объектом карты. Модуль реализован с использованием библиотек GIS Toolkit ГИС Панорама (рис. 6).

В картографическом модуле реализованы стандартные механизмы просмотра карты, такие как загрузка карты, масштабирование и сдвиг карты, нанесение и удаление объекта, получение информации об объекте.

Привязка БД к карте может быть двумя способами: через геокодирование, т. е. по полям, содержащим адреса домов, или привязкой к произвольным объектам карты через таблицу связей.

Для первого способа при описании структуры БД должны указываться поля таблиц, в которых содержатся почтовые адреса. При наличии такой информации на форме для редактирования таблицы отображается кнопка “Адрес на карте” (см. рис. 3), после нажатия на которую выполняется поиск адреса. При этом открывается окно карты, улица и номер дома заносятся в поля панели поиска адреса, и если такой адрес имеется на карте, то в окне выделяется соответствующее здание (рис. 6). Для использования второго способа в БД должна быть создана специальная таблица привязки объектов к карте MAP_LNK (одна на всю базу), позволяющая связать любой объект карты с данными из БД (рис. 7). Отметим, что создание такой таблицы — единственное необходимое изменение структуры исходной БД.

Поверх основной топоосновы в системе ГеоАРМ могут создаваться специализированные слои, предназначенные для нанесения на карту отсутствующих там объектов (например, слои “Гаражи”, “Реклама” и т. д.). При этом предполагается, что пользователь не имеет права редактировать исходную топооснову, но может вносить изменения в созданные им самим специализированные слои. При необходимости включения в клас-

сификатор карты новых типов объектов, такой классификатор должен быть создан или изменен средствами ГИС Панорама.

На специализированных слоях можно создать новые объекты. Предполагается, что при наличии в таблице адресов, характеризующих расположение объектов, пользователь будет находить фрагмент карты при помощи геокодирования, а затем выбирать уже имеющийся объект или создавать новый на специализированном слое и связывать этот объект с записью таблицы.

В том случае, когда в существующей БД имеется информация о координатах объектов, такие объекты могут быть автоматически нанесены на карту. Для этого в описании структуры БД должно быть указано, каким образом в ней хранятся координаты. После нанесения объекта на карту связь соответствующей записи с ним автоматически

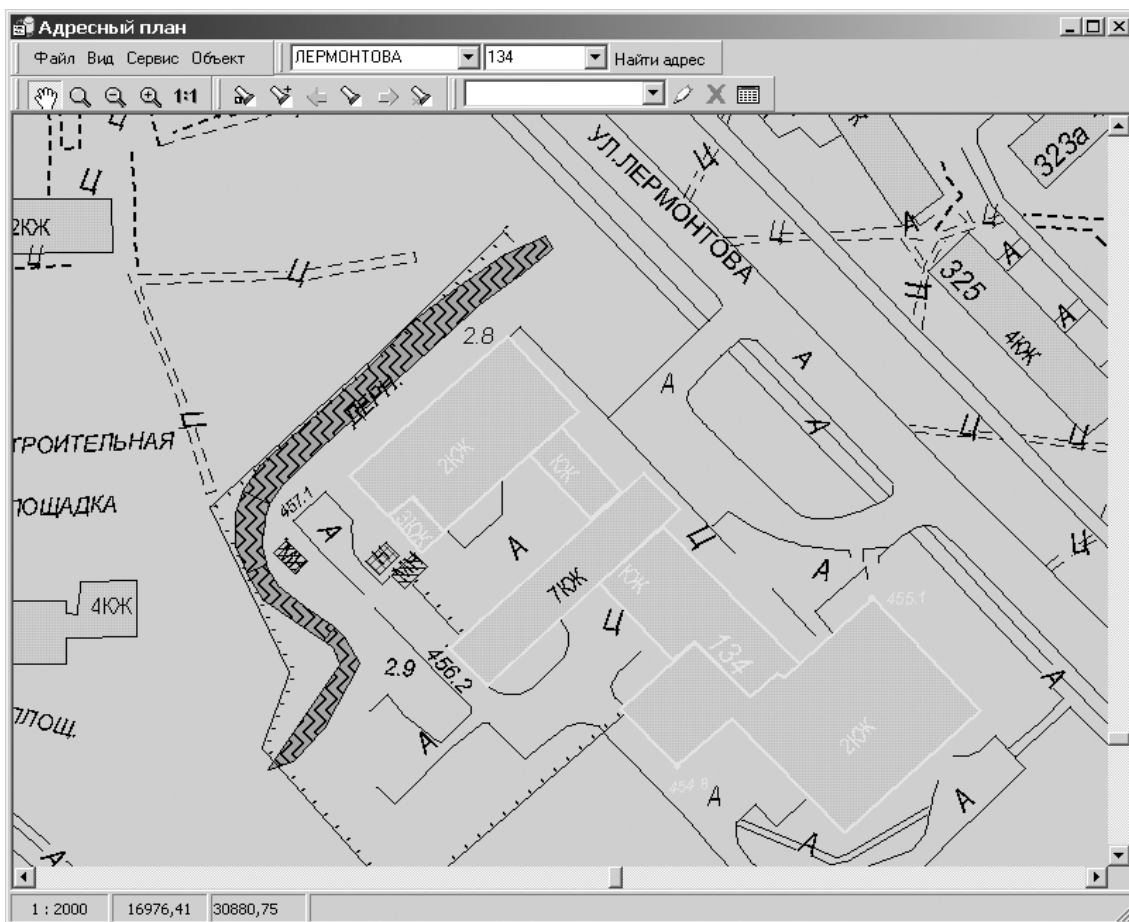


Рис. 6. Пример окна карты с найденным зданием по адресу: ул. Лермонтова, д. 134

MAP_LNK	
KIND	Код таблицы в БД
ID	Указатель на ключевое поле таблицы с данными из тематической БД (таблица определяется значением поля KIND)
ID_L	Номер слоя карты
ID_OBJ	Уникальный номер объекта на карте в слое ID_L

Рис. 7. Структура таблицы привязки объектов к карте

запоминается в таблице MAP_LNK.

Заключение

Созданная система ГеоАРМ позволяет организовывать работу с картой для баз данных, в которых изначально эта возможность не была предусмотрена. Настройку системы для работы с конкретной БД специалисты предметной области выполняют в достаточно сжатые сроки. После настройки системы ГеоАРМ становится альтернативой уже существующего АРМ и в случае простых АРМ может их полностью заменить, поддерживая при этом взаимодействие с ГИС.

В настоящее время система ГеоАРМ проходит опытную эксплуатацию в администрации Иркутска. Была проведена настройка системы для работы с рядом существующих БД. Также несколько информационных систем реализовано с нуля при помощи ГеоАРМ.

В дальнейшем планируется усовершенствовать построитель запросов, чтобы поддержать включение в запросы подзапросов и функций агрегирования по таблицам деталей. Также предполагается расширить возможности взаимодействия с картой, поддержав различные способы формирования тематических карт с использованием информации из БД.

Список литературы

- [1] ESRI GIS and Mapping Software. <http://esri.com>
- [2] MAPINFO Corporation. <http://mapinfo.com>
- [3] КБ ПАНОРАМА. <http://www.gisinfo.ru>
- [4] БОБРОВСКИЙ С. Delphi 7: Учебный курс. СПб.: Питер, 2003.
- [5] РУЖНИКОВ Г.М., МОИСЕЕВ А.Н., ОГЛОБЛИН В.А., БЫЧКОВ И.В., КУХАРЕНКО Е.Л., ХМЕЛЬНОВ А.Е. Геоинформационная система органов государственной власти Иркутской области. // Информационный бюллетень. 2001. № 2(29)–3(30). С. 30.
- [6] ХМЕЛЬНОВ А.Е., ФЕРЕФЕРОВ Е.С. ГеоАРМ — настраиваемое автоматизированное рабочее место с поддержкой работы с пространственной информацией // Тр. X Байкальской Всерос. конф. “Информационные и математические технологии в науке, технике и образовании”, 2005. Ч. 2. С. 154.
- [7] ПРИСЯЖНЮК С.П., ЖЕЛЕЗНЯКОВ А.В. Опыт применения инструментария GIS ToolKit в отечественных разработках // Информационный бюллетень, ГИС Ассоциация. 2001. № 3(30). <http://www.gisinfo.ru/item/8.htm>
- [8] ДЕЙТ К.ДЖ. Введение в системы баз данных. М.: Диалектика, 1998.

*Поступила в редакцию 14 апреля 2006 г.,
в переработанном виде — 19 июня 2007 г.*