

Структура документа

Данный документ предназначен для описания детальной архитектуры создаваемого программного обеспечения на уровне управляющих элементов и классов. Он должен состоять из заголовка, содержания и следующих разделов.

- **Введение**
В данном разделе указывается назначение данного документа (DDS).
- **Структура модулей клиента.**
Приводится описание форм и модулей клиентского уровня вплоть до отдельных управляющих элементов форм, объектов, классов и функций, диаграммы взаимодействия управляющих элементов для каждой формы ПИ.
- **Структура модулей сервера приложений.**
Детальное описание функциональных модулей сервера приложений.
- **Структура модулей сервера для хранения данных**
Структура и описание таблиц и запросов базы данных
- **Заключение**
В заключении необходимо привести информацию о количестве структурных элементов для детальной архитектуры.

----- **ПРИМЕР ДОКУМЕНТА** -----

Спецификация архитектуры на объектном уровне

(Detailed design specification)

Программный продукт «Erudite REG»

Содержание

Введение	4
Назначение спецификации	4
Структура архитектурных компонентов клиента	5
Объекты интерфейса авторизации	5
Объекты главного интерфейса	5
Объекты интерфейса для роли администратора	5
Редактирование таблицы «пользователи»	5
Предоставление информации о пользователях	5
Добавление нового пользователя	6
Дополнительные функции	6
Объекты интерфейса для роли клиента-оператора	7
Информация о книгах	7
Поиск книги по критериям	7
Вывод результата поиска книги	8
Оформление продажи	8
Структура модулей сервера приложений	10
Структура и описание таблиц БД	14
BOOK	14
INFO	14
CONTACT	14
RIGHT	14
USER	14
SALE	15
Заключение	16

Введение

Назначение спецификации

Данный документ предназначен для описания детальной архитектуры информационной системы, создаваемой для сотрудников книжного магазина «Эрудит».

Структура архитектурных компонентов клиента

Объекты интерфейса авторизации

- Поле для ввода логина (имя пользователя);
- Поле для ввода пароля;
- Графический логотип системы.

Объекты главного интерфейса

На главной форме приложения должны располагаться следующие элементы:

- Информационная строка, отображающая тип пользователя:
 - администратор;
 - клиент-оператор.
- Закладка «информация о предложениях»:
 - скидки;
 - новинки;
 - другое.
- Кнопка предоставления информации о книгах;
- Кнопка предоставления информации о продажах;
- Кнопка поиска книги по критериям;
- Кнопка оформления продажи;
- Графический логотип системы.

Для администратора форма имеет дополнительные атрибуты:

- Кнопка редактирование таблицы «пользователи».

Объекты интерфейса для роли администратора

Редактирование таблицы «пользователи»

- Кнопка для поиска пользователей;
- Кнопка для предоставления информации о пользователях;
- Кнопка редактирования информации о пользователях;
- Кнопка добавления пользователя;
- Кнопка дополнительных функций;
- Графический логотип системы.

Предоставление информации о пользователях

- Поле номер;
- Поле ФИО;

- Поле логин;
- Поле пароль доступа;
- Поле контактная информация:
 - Поле имя;
 - Поле телефон;
 - Поле e-mail;
 - Поле адрес;
 - Поле дополнительная информация;
 - Поле комментарии.
- Поле дополнительная информация;
- Кнопка для поиска пользователей;
- Кнопка редактирования информации о пользователях;
- Кнопка добавления пользователя;
- Кнопка дополнительных функций;
- Графический логотип системы.

Добавление нового пользователя

- Поле для ввода информации о пользователе:
- Поле ФИО;
- Поле логин;
- Поле пароль доступа;
- Поле контактная информация:
 - Поле имя;
 - Поле телефон;
 - Поле e-mail;
 - Поле адрес;
 - Поле дополнительная информация;
 - Поле комментарии.
- Поле дополнительная информация;
- Кнопка для поиска пользователей;
- Кнопка редактирования информации о пользователях;
- Кнопка дополнительных функций;
- Графический логотип системы.

Дополнительные функции

- Кнопка создания резервной копии БД;

- Поле имя файла сохранения;
 - Поле путь к файлу сохранения;
- Кнопка восстановление из резервной копии;
 - Поле путь к файлу восстановления.
- Кнопка обновление версии БД;
- Кнопка для поиска пользователей;
- Кнопка для предоставления информации о пользователях;
- Кнопка редактирования информации о пользователях;
- Кнопка добавления пользователя.

Объекты интерфейса для роли клиента-оператора

Информация о книгах

- Поля информации о книгах:
 - Поле номер;
 - Поле наименование;
 - Поле автор;
 - Поле жанр;
 - Поле тематика;
 - Поле штрих-код;
 - Поле дата поступления;
 - Поле дата издания;
 - Поле количество в наличии;
 - Поле цена;
 - Поле дополнительная информация;
 - Поле комментарии;
 - Поле расположение.
- Кнопка информации о схожих по тематике книгах;
- Кнопка предоставления информации о предложениях;
- Кнопка поиска книги по критериям;
- Кнопка оформления продажи;
- Графический логотип системы.

Поиск книги по критериям

- Поля критериев поиска:
 - Поле наименование;
 - Поле автор;

- Поле жанр;
- Поле тематика;
- Поле дата поступления;
- Поле дата издания;
- Поле количество в наличии;
- Поле цена;
- Поле расположение.
- Поле предоставления информации о предложениях;
- Кнопка поиска книги по критериям;
- Кнопка оформления продажи;
- Графический логотип системы.

Вывод результата поиска книги

- Поле номер;
- Поле наименование;
- Поле автор;
- Поле жанр;
- Поле тематика;
- Поле штрих-код;
- Поле дата поступления;
- Поле дата издания;
- Поле количество в наличии;
- Поле цена;
- Поле дополнительная информация;
- Поле комментарии;
- Поле расположение;
- Кнопка оформления продажи;
- Графический логотип системы.

Оформление продажи

- Поле для номера покупки;
- Поле наименование книги;
- Поле дополнительная информация;
- Поле цена;
- Поле количество;

- Поле дата;
- Поле скидка;
- Поле сумма к оплате;
- Поле продавец;
- Поле предоставления информации о предложениях;
- Поле предоставления информации о книгах;
- Поле поиска книги по критериям;
- Кнопка оформить покупку;
- Кнопка добавить покупку;
- Графический логотип системы.

Структура модулей сервера приложений

Функционально информационная система «Erudite REG» состоит из двух частей – клиентской и серверной (база данных) (рис.2).

Система управления базами данных – Microsoft Access.

Клиентская часть – приложение, реализованное в среде разработки Borland Delphi 6.

Работа с базой данных осуществляется посредством технологии ActiveX Data Objects(ADO). На машине клиента должен быть установлен провайдер данных. Взаимодействие компонентов ADO и провайдера осуществляется на основе универсальной для Windows технологии ActiveX, причем провайдер реализуется как COM – сервер, а ADO – компоненты – как COM-клиенты.

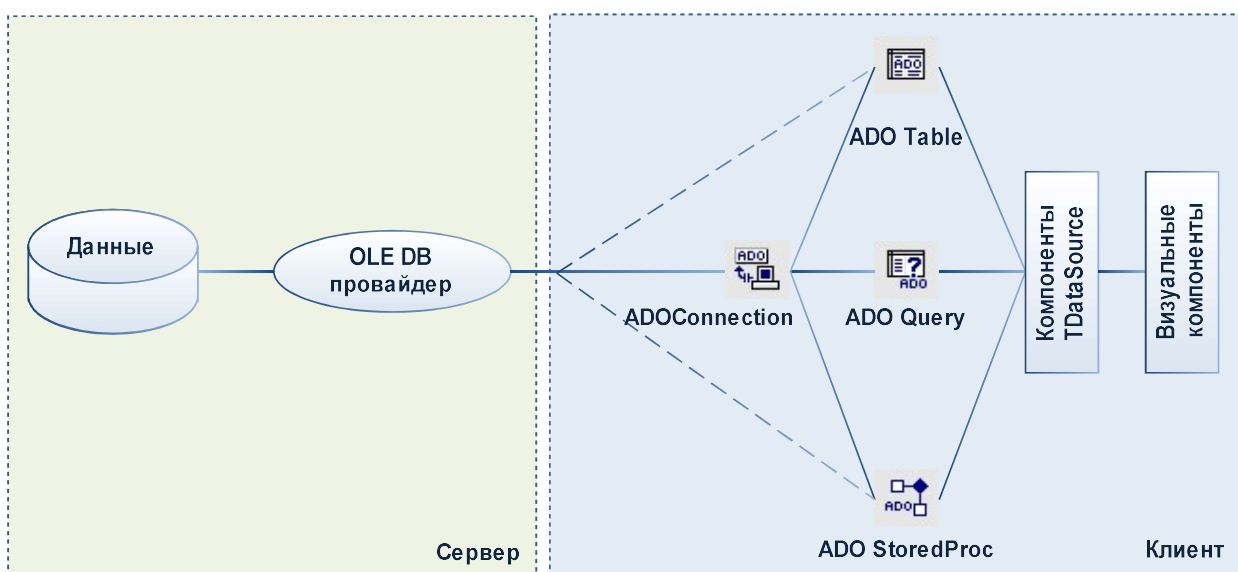


Рис.2 Технология взаимодействие клиента и сервера

Технология Microsoft ActiveX Data Objects обеспечивает универсальный доступ к источникам данных из приложений БД. Такую возможность предоставляют функции набора интерфейсов, созданные на основе общей модели объектов COM и описанные в спецификации OLE DB.

Технология ADO и интерфейсы OLE DB обеспечивают для приложений *единый способ доступа* к источникам данных различных типов (рис.3). Например, приложение, использующее ADO, может применять одинаково сложные операции и к данным, хранящимся на корпоративном сервере SQL, и к электронным таблицам, и локальным СУБД. Запрос SQL, направленный любому источнику данных через ADO, будет выполнен.

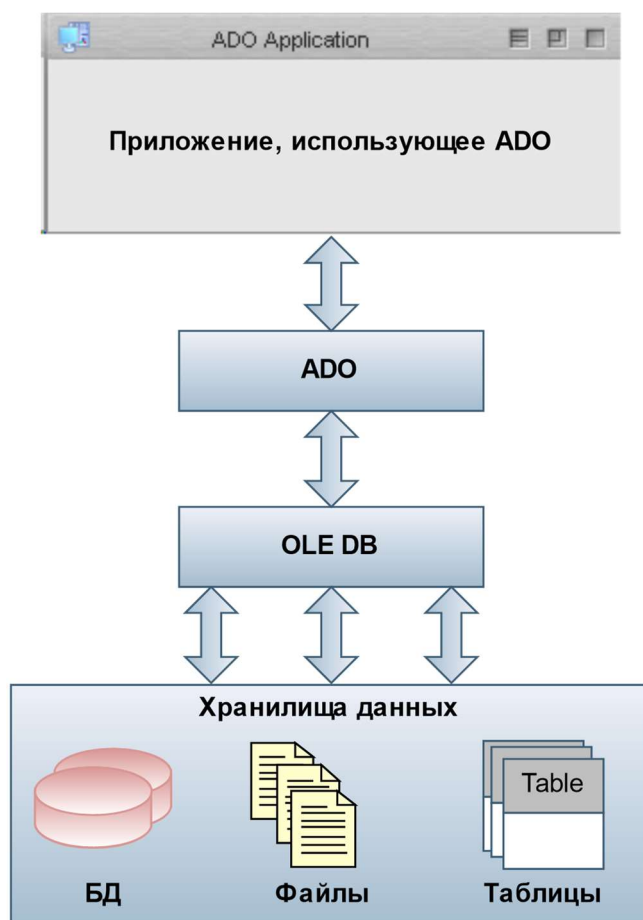


Рис. 3 Схема доступа к различным данным при помощи ADO

Согласно терминологии ADO, любой источник данных (база данных, электронная таблица, файл) называется *хранилищем данных*, с которым при помощи провайдера данных взаимодействует приложение. Минимальный набор компонентов приложения может включать объект соединения, объект набора данных, объект процессора запросов.

OLE DB представляет собой набор специализированных объектов COM, инкапсулирующих стандартные функции обработки данных, и специализированные функции конкретных источников данных и интерфейсов, обеспечивающих передачу данных между объектами. Другими словами, OLE DB — набор интерфейсов, основанных на COM, которые позволяют приложениям обращаться к данным, хранимым в разных источниках информации или хранилищах данных с помощью унифицированного доступа. Основные объекты OLE DB и их функции представлены на рис. 4.

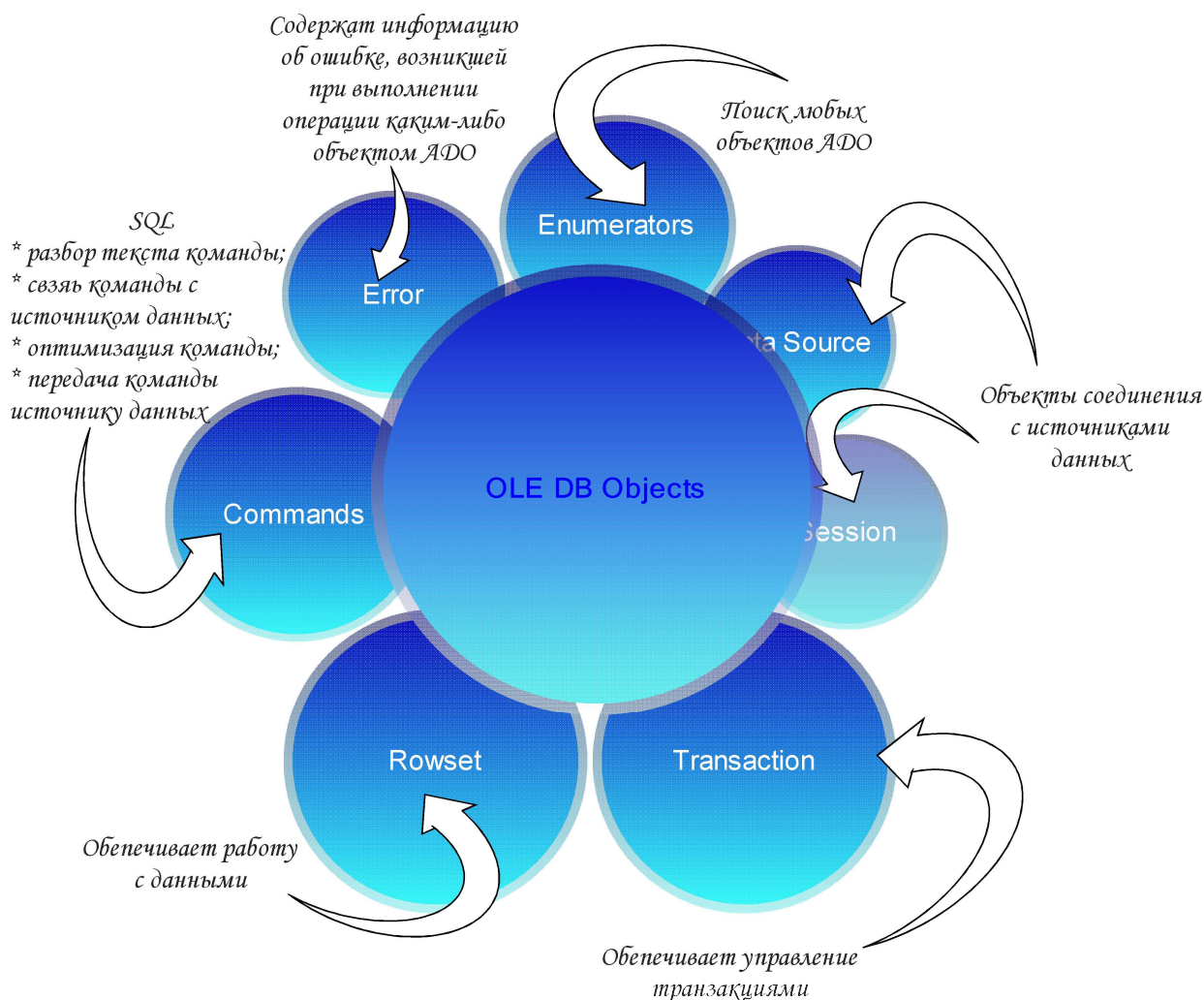


Рис. 4 Основные объекты OLE DB и их функции

Объекты OLE DB создаются и функционируют так же, как и другие объекты COM. Каждому объекту соответствует идентификатор класса CLSID, хранящийся в системном реестре. Для создания объекта используется метод CoCreateInstance и соответствующая фабрика класса. Объекту соответствует набор интерфейсов, к методам которых можно обращаться после создания объекта.

В результате приложение обращается не прямо к источнику данных, а к объекту OLE DB, который "умеет" представить данные (например, из файла электронной почты) в виде таблицы БД или результата выполнения запроса SQL.

Технология ADO в целом включает в себя не только сами объекты OLE DB, но и механизмы, обеспечивающие взаимодействие объектов с данными и приложениями. На этом уровне важнейшую роль играют провайдеры ADO, координирующие работу приложений с хранилищами данных различных типов.

Такая архитектура позволяет сделать набор объектов и интерфейсов открытым и расширяемым. Набор объектов и соответствующий провайдер может быть создан для любого хранилища данных без внесения изменений в исходную структуру ADO. При этом

существенно расширяется само понятие данных – ведь можно разработать набор объектов и интерфейсов и для нетрадиционных табличных данных. Например, это могут быть графические данные геоинформационных систем, древовидные структуры из системных реестров, данные CASE-инструментов и т. д.

Так как технология ADO основана на стандартных интерфейсах COM, которые являются системным механизмом Windows, это сокращает общий объем работающего программного кода и позволяет распространять приложения БД без вспомогательных программ и библиотек.

Структура и описание таблиц БД

BOOK

B_ID — уникальный идентификатор книги;

Name — наименование;

Author — автор;

Topic — краткое описание;

Theme — тематика;

Publisher — издательство;

PageCount — число страниц;

Quantity — количество;

DatePubl — дата издания;

Cost — цена;

Place — местонахождение.

INFO

I_ID — уникальный идентификатор информации;

Name — краткое наименование;

Date — срок действия предложения;

Info — информация.

CONTACT

C_ID — уникальный идентификатор контактной информации;

Town — город;

Street — улица;

House — номер дома;

Flat — номер квартиры;

Phone — телефон;

E-mail — адрес электронной почты;

Other — прочее.

RIGHT

R_ID — уникальный идентификатор прав;

Name — наименование;

Action — действие;

Description — описание.

USER

U_ID — уникальный идентификатор пользователя;

Login — логин;
 Password — пароль доступа;
 FIO — фамилия имя отчество;
 Other — дополнительная информация;
 C_ID — контакты;
 R_ID — права.

SALE

S_ID — уникальный идентификатор пользователя;
 B_ID — наименование товара;
 I_ID — наименование информации;
 U_ID — пользователь;
 Quantity — количество;
 Cost — цена;
 Date — дата.

Физическая модель базы данных, построенная с помощью CASE-инструмента Erwin4.0 показана на рис. 1.

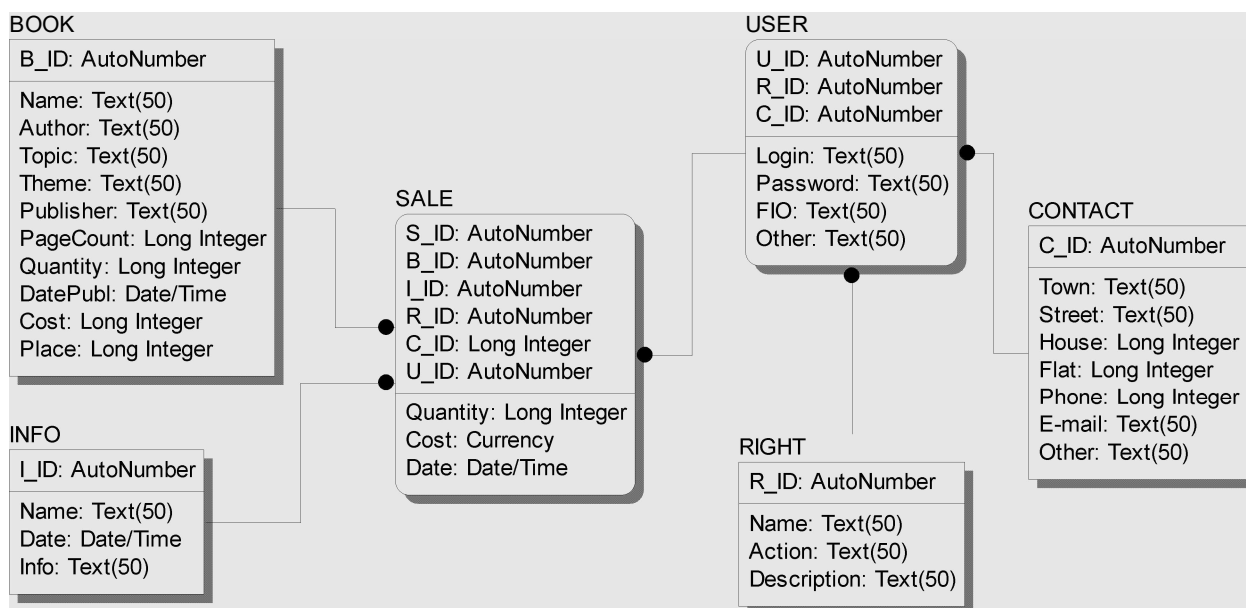


Рис. 1 Физическая модель базы данных

Заключение

Разрабатываемая информационная система будет состоять из 12 форм ПИ, включающих 84 управляющих элемента и функциональных объекта. Сервер приложений будет состоять из 5 функциональных модулей, включающих 47 функциональных объектов. База данных будет состоять из 6 таблиц включающих 43 поля.