Методы сбора, хранения, обработки и анализа данных

Лекция 1

Проектирование баз данных

Организационное

- 36 часов лекций
- 18 часов лабораторных работ
- Задания на лабораторные работы: diskstation.belstu.by
- Для студентов ФИТ / Преподаватели / Блинова
- Доклад по согласованной теме
- Итоговый контроль – зачет
- Продолжение в следующем семестре

План курса

- Проектирование базы данных
- Специальные типы данных
- Расширенные возможности работы с данными
- Мониторинг и настройка
- Высокая доступность
- Информационная безопасность
- NoSQL решения
- Аналитика

ПРОЕКТИРОВАНИЕ БД

Требования к данным

- Многократное использование данных
- Простота и легкость использования
- Гибкость использования
- Быстрая обработка запросов на данные

OLTP и Хранилища данных

- OLTP – данные, получаемые в результате повседневных транзакций
- Основные принципы:
  - Хранение в единственном месте
  - Обеспечение транзакционной поддержки

OLTP и Хранилища данных

- Хранилища данных – поддержка принимаемых решений
- Основные принципы:
  - Предварительные вычисления
  - Частота обновления данных зависит от потребностей пользователей
  - Объединение данных из нескольких источников

Проектирование базы данных

1. Концептуальное проектирование
2. Логическое проектирование
3. Физическое проектирование
4. Физическая реализация
5. Оценка полученного результата

Команда

- Бизнес-аналитик
- Архитектор БД
- Системный архитектор
- Менеджер проекта

Бизнес-аналитик

- Излагает требования бизнеса в деталях
- Со стороны пользователя

Архитектор БД

- Разрабатывает техническую реализацию
- Проектирует размещение данных и доступ к ним
- Выбирает технологии хранения

Системный архитектор

- Обеспечивает проектирование полного интерфейса пользователей
- Отвечает за выбор промежуточных технологий

Менеджер проекта

- Координирует членов проекта
- Отвечает за сроки и деньги

Документация

- Перечень границ проекта
- Перечень отрицательного опыта пользователей
- Запросы пользователей
- Поддержка общедоступного хранилища документации с версионностью

Взаимодействие с заказчиком

- Внесение и одобрение изменений
- Опытные образцы
- Интервью с клиентами

Интервью с клиентами

- Кто будет использовать данные?
- Как данные будут использоваться?
- Что должно быть в отчетах?
- Где сейчас находятся данные?
- Сколько эти данные стоят?
- Интеграция новой БД с уже существующими данными?
- Правила, управляющие данными?
- Соглашения об уровне обслуживания?

Концептуальное проектирование

Цель – создание концептуальной модели данных исходя из представлений пользователей о предметной области

Концептуальное проектирование

- Отказ от выбора окончательной структуры на ранней стадии выполнения работ
- Определение сущностей
- Определение атрибутов и доменов
- Определение связей
- Определение бизнес-правил

Определение сущностей

- Человек
- Место
- Объект
- Идея
- Документ
- Другие сущности:
  - Протоколы или журналы
  - События

Определение сущностей

- После предварительного определения сущностей составить их список
- Сформировать список вопросов к клиенту по уточнению
- Уточнить список сущностей

Определение атрибутов и доменов

- Идентификаторы
- Описательная информация
- Указатели на расположение
- Связанная информация
- Допустимые значения

Определение связей

- Один-ко-многим
  - Имеет
  - Является
- Многие-ко многим

Определение бизнес-правил

- Набор правил, которые определяют деловое поведение заказчика
- Получение:
  - Изучение документации
  - Старый код
  - Интервью с заказчиком
- Идентификация основных процессов

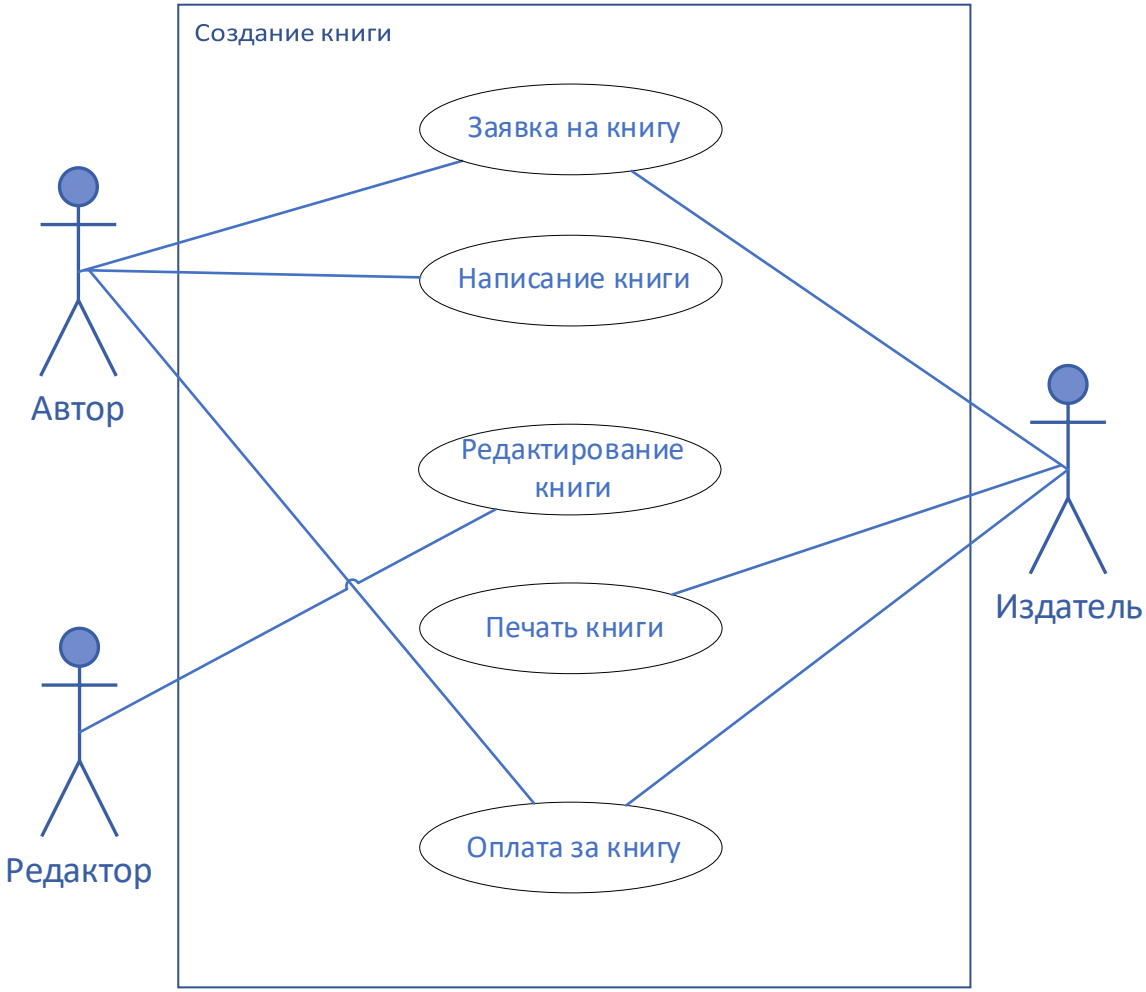
Моделирование данных

- UML – стандартная методология для определения и документирования программных систем
- IDEF1X – Integration Definition For Information Modelling – методология для реляционных данных
- IE – Informational Engineering – моделирование связей между таблицами

UML

- Модель описывает:
- Компоненты системы как **действия** (use-case)
- Пользователи системы как **исполнители** (actors)
- Отношения между исполнителями и действиями
- Действия могут быть связаны друг с другом:
  - Использует
  - Расширяет

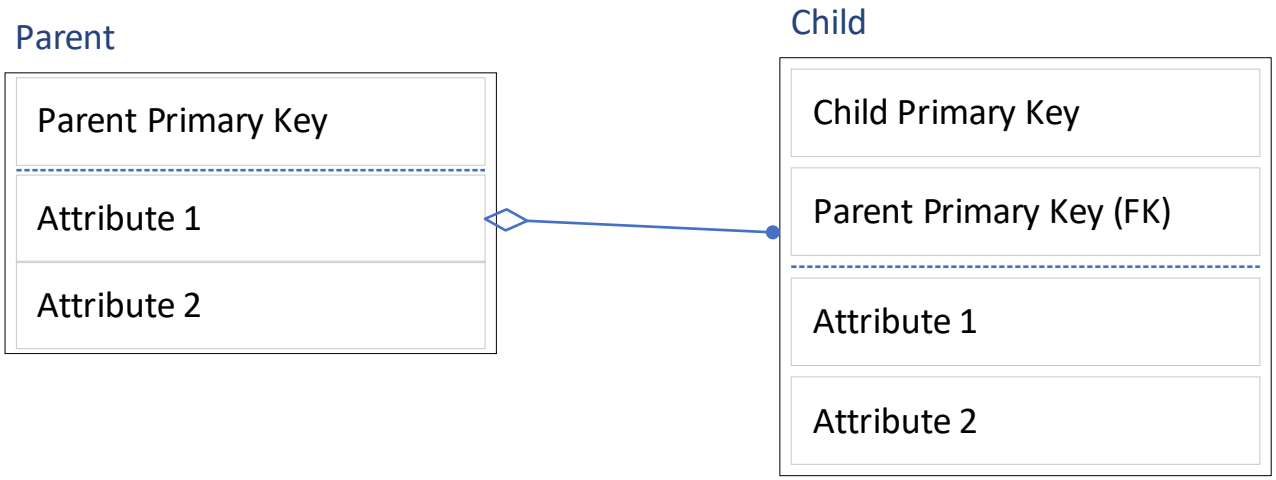
UML


Создание книги
Заявка на книгу
Написание книги
Редактирование книги
Печать книги
Оплата за книгу
Автор
Редактор
Издатель

IDEF1X

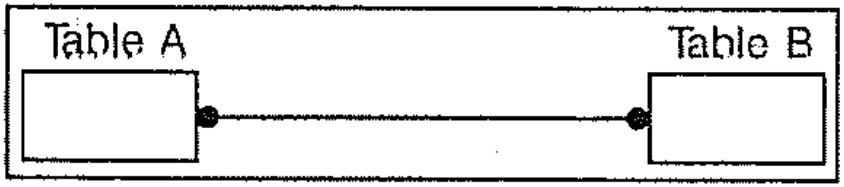
- Сущность и атрибуты
  - Зависимые и независимые сущности
  - Первичные ключи
  - Вторичные ключи
  - Внешние ключи
  - Домены
- Связи
  - Идентифицирующая
  - Неидентифицирующая
  - Необязательная
  - Рекурсия
  - Многие-ко-многим

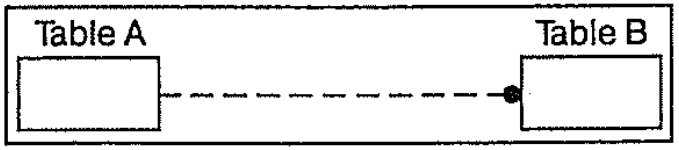
IDEF1X

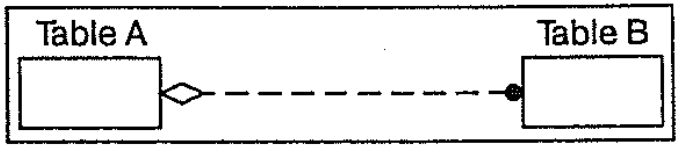

Parent
Parent Primary Key
Attribute 1
Attribute 2
Child
Child Primary Key
Parent Primary Key (FK)
Attribute 1
Attribute 2

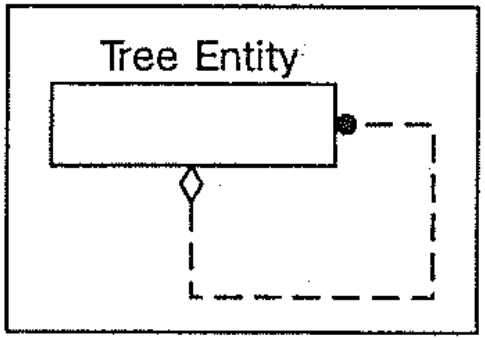
Связи

- Идентифицирующая – для определения зависимых сущностей
- Неидентифицирующая – более общий вид связи
- Обязательная или необязательная
- Рекурсия
- Многие-ко-многим

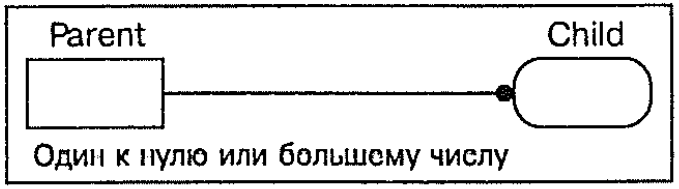




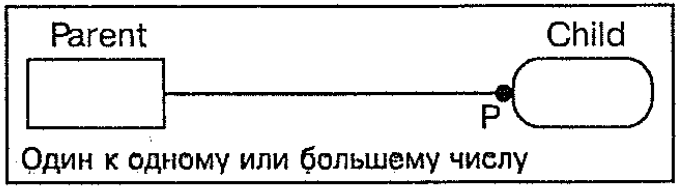




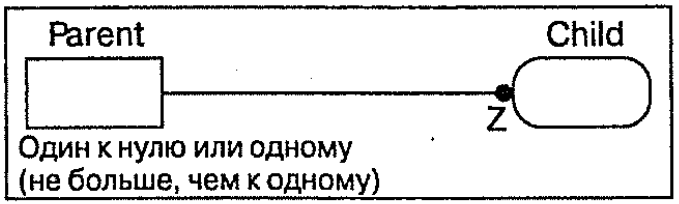
Мощность связи



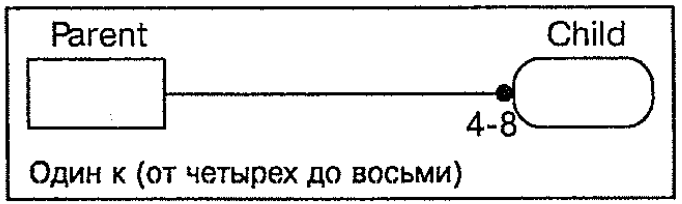
Один к нулю или большему числу



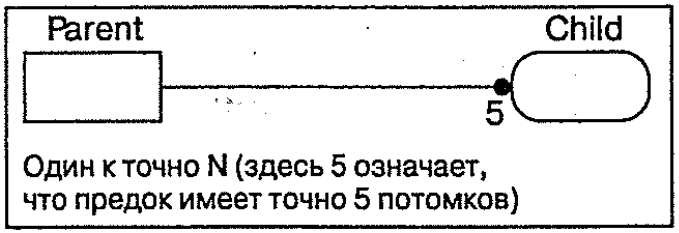
Один к одному или большему числу



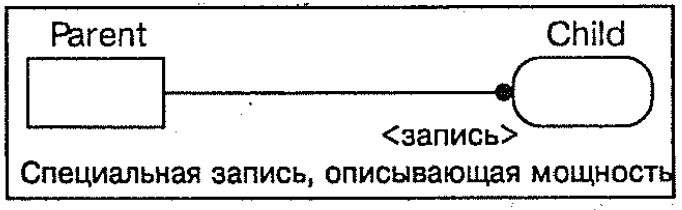
Один к нулю или одному
(не больше, чем к одному)



Один к (от четырех до восьми)

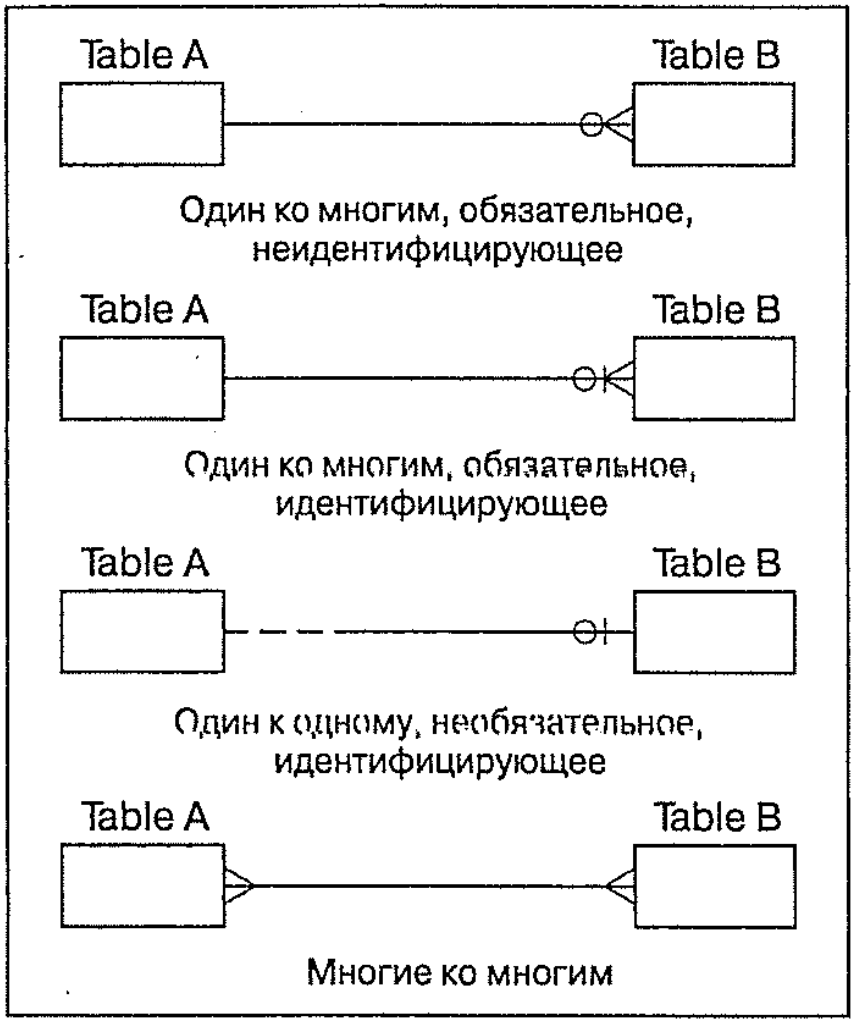


Один к точно N (здесь 5 означает,
что предок имеет точно 5 потомков)



Специальная запись, описывающая мощность

IE – Informational Engineering



Концептуальное проектирование

1. Определение сущностей и их документирование
2. Определение связей между сущностями и их документирование
3. Создание ER-модели предметной области
4. Определение атрибутов и их документирование:
   - имя атрибута и его описание;
   - домен атрибута;
   - тип и размерность значений;
   - значение, принимаемое для атрибута по умолчанию;
   - может ли атрибут иметь Null-значения;
   - является ли атрибут составным
5. Определение значений атрибутов и их документирование
6. Определение первичных ключей для сущностей и их документирование
7. Обсуждение концептуальной модели данных с конечными пользователями

Логическое проектирование

- Цель – преобразование концептуальной модели на основе выбранной модели данных в логическую модель, не зависимую от особенностей используемой в дальнейшем СУБД для физической реализации базы данных

Логическое проектирование

1. Выбор модели данных
2. Определение набора таблиц исходя из ER-модели и их документирование
3. Нормализация таблиц
4. Проверка логической модели данных на предмет возможности выполнения всех транзакций, предусмотренных пользователями
5. Определение требований поддержки целостности данных и их документирование:
   - обязательные данные
   - ограничения для значений атрибутов
   - целостность сущностей
   - ссылочная целостность
   - ограничения, накладываемые бизнес-правилами
6. Создание окончательного варианта логической модели данных и обсуждение его с пользователями

Нормализация

- Устранение NULL
- Устранение избыточности данных
- Устранение ненужного кодирования
- Максимизация кластерных индексов
- Уменьшение числа индексов на таблицу
- Хранение тонких таблиц

Нормализация

- 1 НФ:
- Все атрибуты должны быть элементарными
- Экземпляры сущности должны иметь одно и то же количество значений
- Все экземпляры сущности должны быть различны

Нормализация

- 2 НФ:
- Сущность должна соответствовать 1 НФ
- Каждый атрибут должен зависеть от ключа

- Пример (сущность – книга):

Book_ISBN Book_Title Id_Author Author_First_Name Author_Second_Name Author_Royalty	Book_ISBN Book_Title Id_Author Author_First_Name Author_Second_Name Book_ISBN Id_Author Author_Royalty

Нормализация

- 3 НФ:
- Сущность должна соответствовать 2 НФ
- Каждый атрибут должен зависеть только ключа

- Пример:

Book_ISBN Book_Title Book_Price Publisher_Name Publisher_City	Book_ISBN Book_Title Book_Price Id_Pablisher Id_Pablisher Publisher_Name Publisher_City

Нормализация

- НФ Бойса-Кодда:
  - Все атрибуты полностью зависят от ключа
  - Сущность находится в НФБК, если каждый детерминант – ключ
- **Детерминант** – любой атрибут или комбинация атрибутов, от которых функционально зависит любой другой атрибут или комбинация атрибутов
- Если набор столбцов является ключом, то необходимо внести ограничение уникальности

Нормализация

- 4 НФ:
  - Сущность находится в НФБК
  - Не должно быть больше одной зависимости с многими значениями, представленной в сущности
- Проблемы:
  - Тройные отношения
  - Скрытые атрибуты с несколькими величинами
  - Атрибуты с предыдущими значениями

Тройные отношения

- Проведение докладов:
  - Зал
  - Сессия
  - Докладчик
- Проблемы:
  - Если один доклад делают два докладчика?
  - Если докладчику надо более одного зала?

Скрытые атрибуты с несколькими величинами

- Контакты:
  - Имя
  - Адреса
  - Телефоны
- Проблемы:
  - Разные типы телефонов и адресов (рабочий, домашний, факс и т.д.)
  - Связь номера телефона и адреса
  - Несколько партнеров с одним адресом

Атрибуты с предыдущими значениями

- Использование оборудования:
  - Сотрудник
  - Оборудование
- Проблемы:
  - Сотрудник получает и сдает оборудование
  - Оборудование может последовательно находиться у разных сотрудников

Денормализация

Используется для улучшения работы:

- Вычисленные атрибуты (TotalSum)
- Преимущественные значения (PreferredPhoneNumber)
- Отметка изменений (LastUsage)

Логическое проектирование

- Логическая схема базы данных для курсовых и дипломных проектов (IDEF1x):
  - Изобразить сущности, каждой дать имя
  - Изобразить связи
  - Вначале – составляющие ключа, затем прочие атрибуты
  - Типы данных не указываются
  - PK и FK указываются

Использование данных

- Отчеты
- Ограничения
- Взаимодействие с внешними системами
- Планы преобразования данных

Отчеты

- Стандартные – отчеты, которые пользователь получает во время своей работы
- Специализированные – отчеты, результаты которых будут влиять на бизнес-процессы

Ограничения

- Тип приложения
- Стек технологий
- Количество внешних и внутренних пользователей
- Аппаратные средства – серверы, ПО
- Физические ограничения – облака, серверные, Интернет

Взаимодействие с внешними системами

- Определить внешние системы
- Описать взаимодействие
- Разработать промежуточный слой для взаимодействия
- Возможно, потребуется доработка внешних систем

Планы преобразования данных

- Средняя длина атрибутов
- % данных, заполненный для атрибута
- Первоначальное количество строк в таблице
- Скорость роста данных в таблицах
- Ожидаемый срок службы

Окончательный обзор проекта

- Планирование выполнения
  - Этапы (контрольные точки)
  - Список задач
  - Очередность задач
  - Время на каждую задачу
- Обзор и согласование документации

Физическое проектирование

- Цель – описание конкретной реализации базы данных
- Проблемы:
  - Размер и сложность данных
  - Поиск
  - Конкуренция за ресурс
  - Своевременность и частота отчетов
  - Бюджет

Физическое проектирование

- Цель – описание конкретной реализации базы данных
- Проблемы:
  - Размер данных
  - Сложность
  - Поиск
  - Конкуренция за ресурс
  - Своевременность и частота отчетов

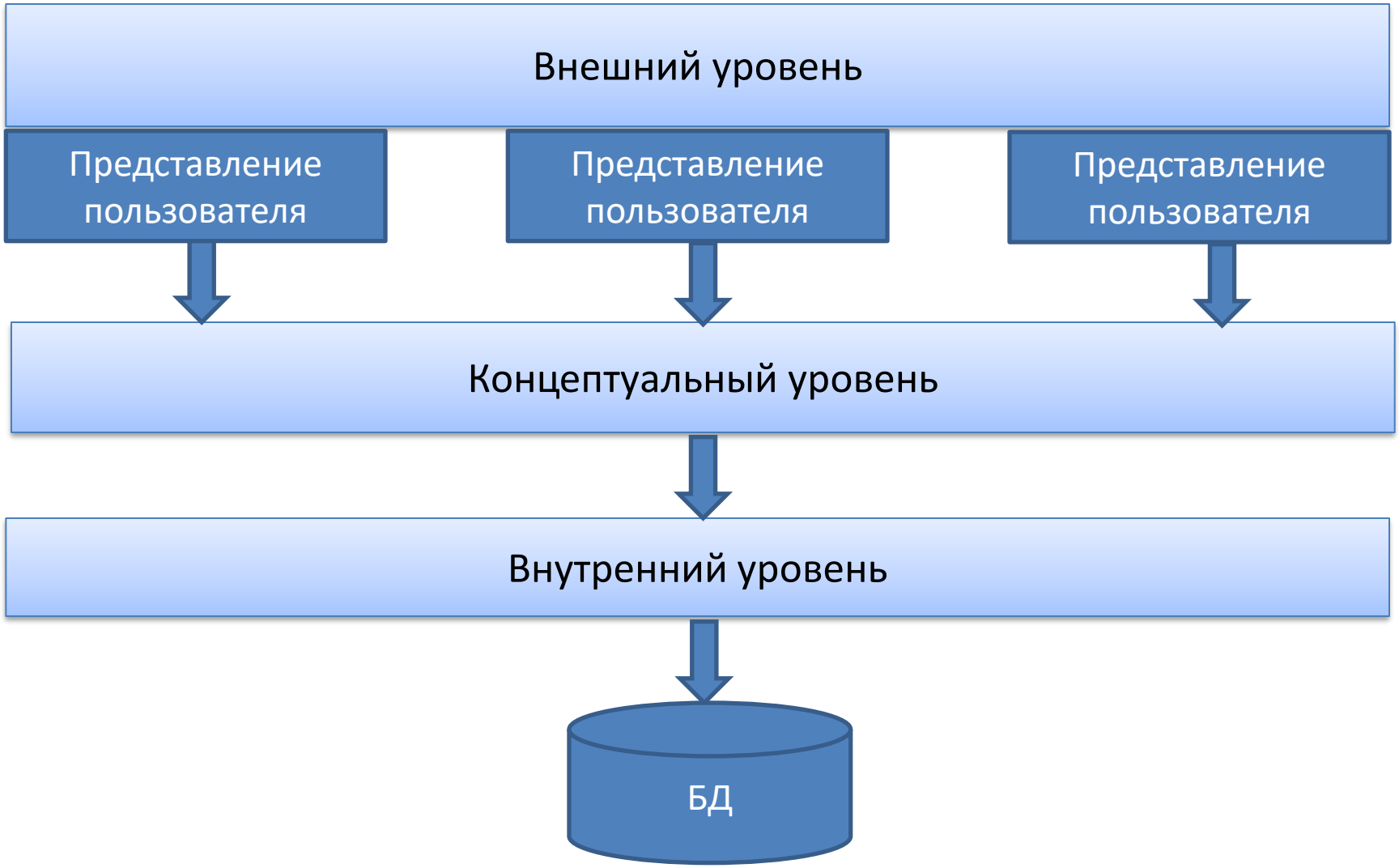
Физическое проектирование

1. Согласование архитектуры
2. Проектирование и разработка таблиц базы данных средствами выбранной СУБД
3. Реализация бизнес-правил в среде выбранной СУБД
4. Проектирование и реализация физической организации базы данных
5. Разработка стратегии защиты базы данных
6. Организация мониторинга функционирования базы данных и ее настройка

Физическое проектирование

- Физическая схема базы данных для курсовых и дипломных проектов (IDEF1x):
  - К логической добавляются типы данных через двоеточие
  - Указываются индексируемые поля

Модель организации баз данных



Модели данных

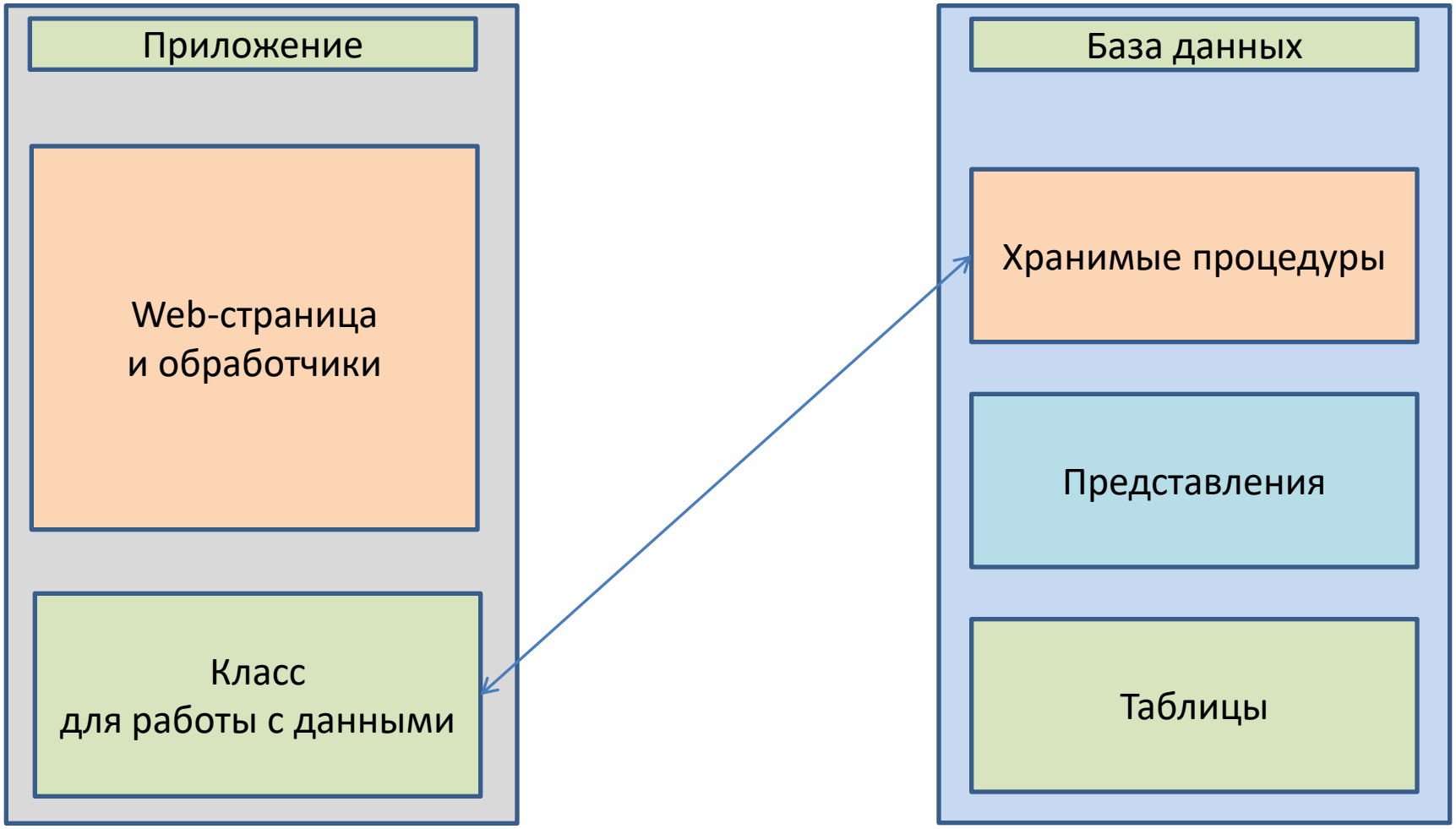
- Реляционная
- NoSQL-модели
  - Ключ – значение (Cassandra)
  - Сетевые (графовые, Gremlin)
  - Документные (MongoDB)
  - Табличные (с частичной поддержкой SQL)

ДОСТУП К ДАННЫМ ИЗ ПРИЛОЖЕНИЯ

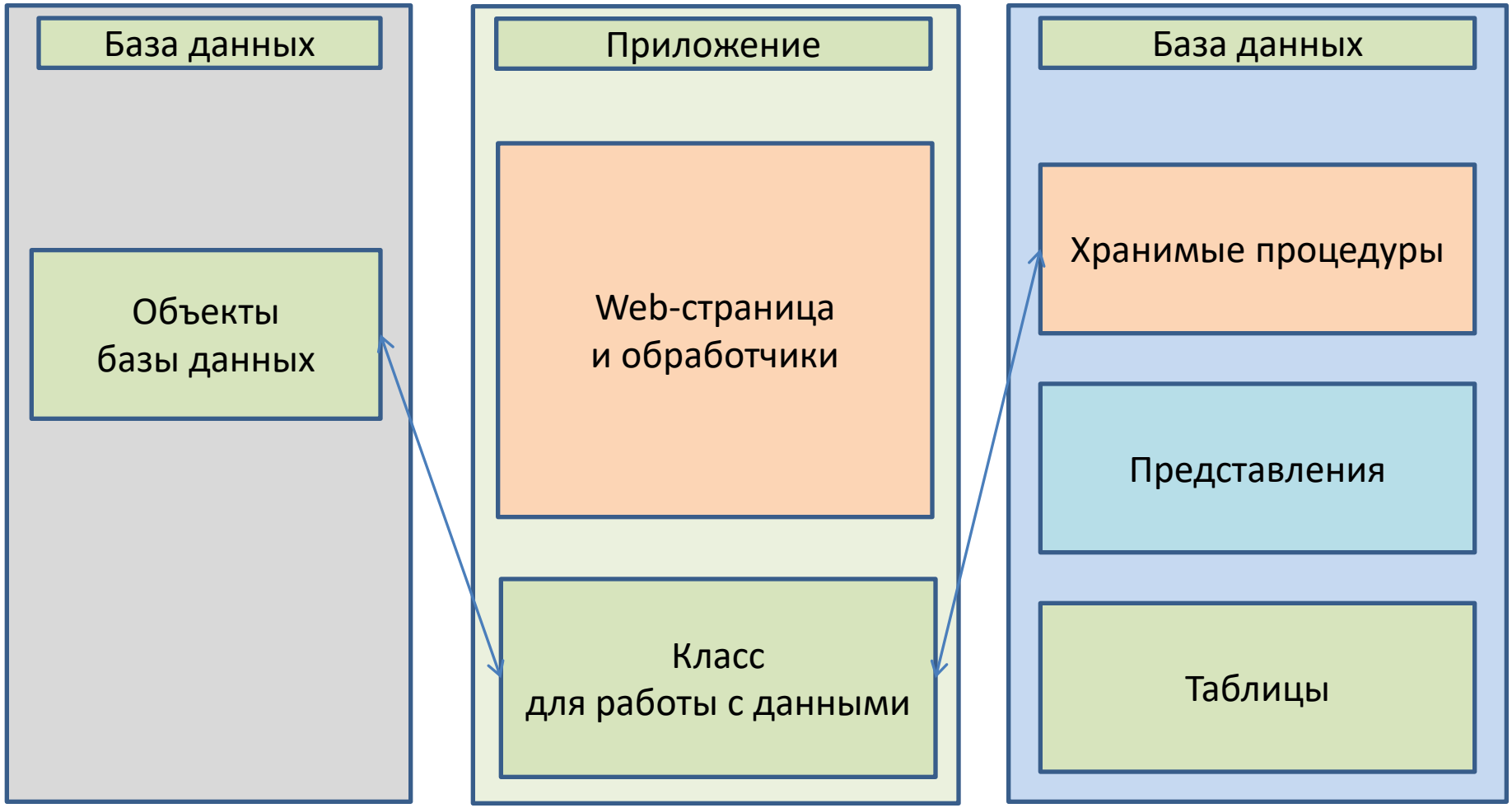
Модели приложений

- Приложение – база данных
- Приложение – несколько баз данных
- SOA

Модель приложения



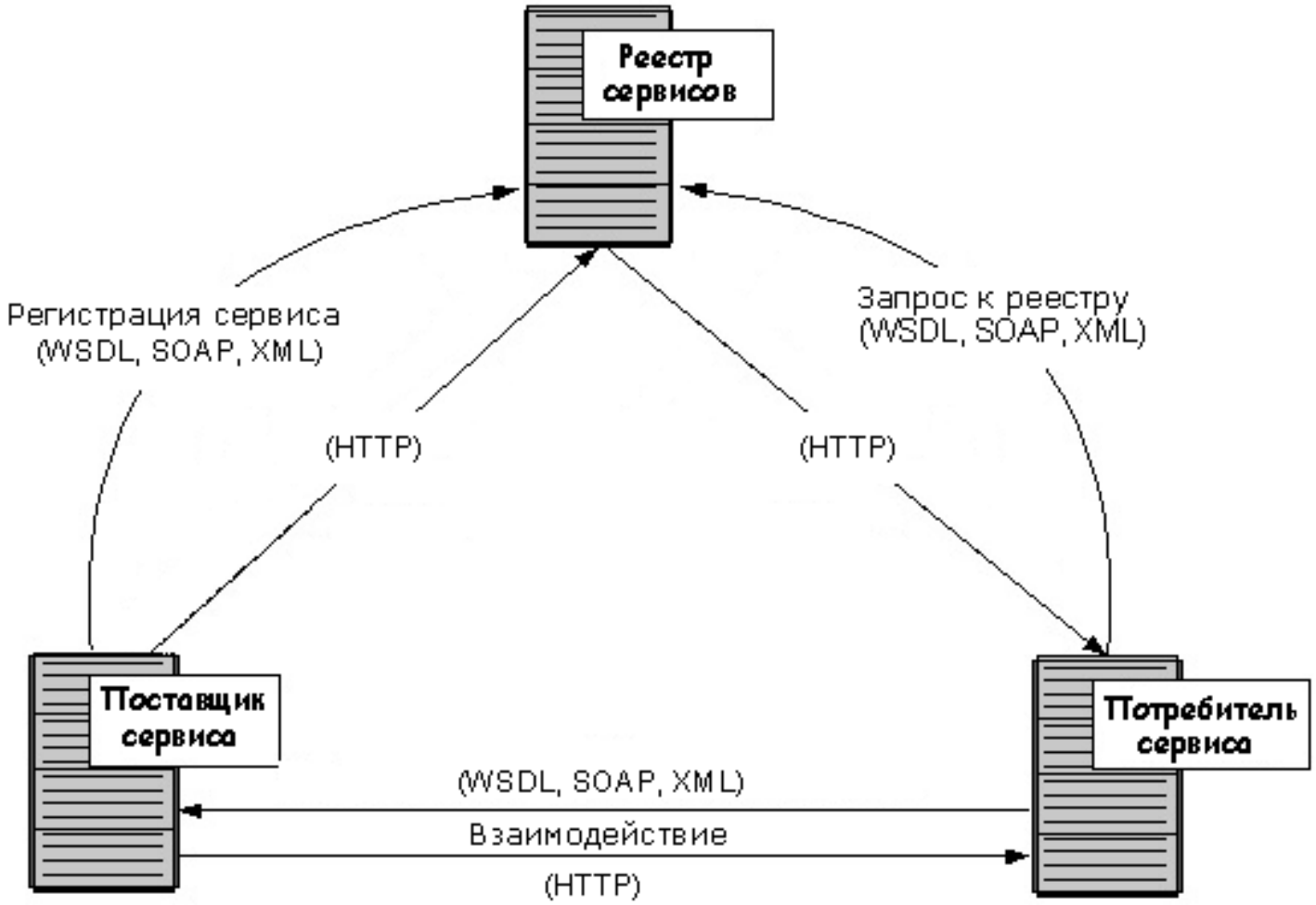
Модель приложения



SOA

- SOA – Service-Oriented Architecture – Сервис-ориентированная архитектура
- SOA — модульный подход к разработке программного обеспечения, основанный на использовании распределённых, слабо связанных заменяемых компонентов, оснащённых стандартизированными интерфейсами для взаимодействия по стандартизированным протоколам

SOA


Реестр сервисов
Регистрация сервиса (WSDL, SOAP, XML)
Запрос к реестру (WSDL, SOAP, XML)
(HTTP)
(HTTP)
Поставщик сервиса
Потребитель сервиса
(WSDL, SOAP, XML)
Взаимодействие
(HTTP)

API для доступа к данным

- Доступ к данным – прикладной программный интерфейс для СУБД
- Набор функций:
  - установление и закрытие соединения
  - обновление данных
  - передача запросов серверу
  - получение результатов выполнения запросов
  - получение кодов ошибок
  - характеристики структуры набора результата

Универсальный механизм доступа к данным

- Универсальный механизм доступа к данным обычно реализован в виде библиотек и дополнительных модулей – драйверов или провайдеров
- Библиотеки содержат стандартный набор функций или классов, подчиняющийся спецификации
- Дополнительные модули реализуют непосредственное обращение к функциям конкретных СУБД

Универсальный механизм доступа к данным

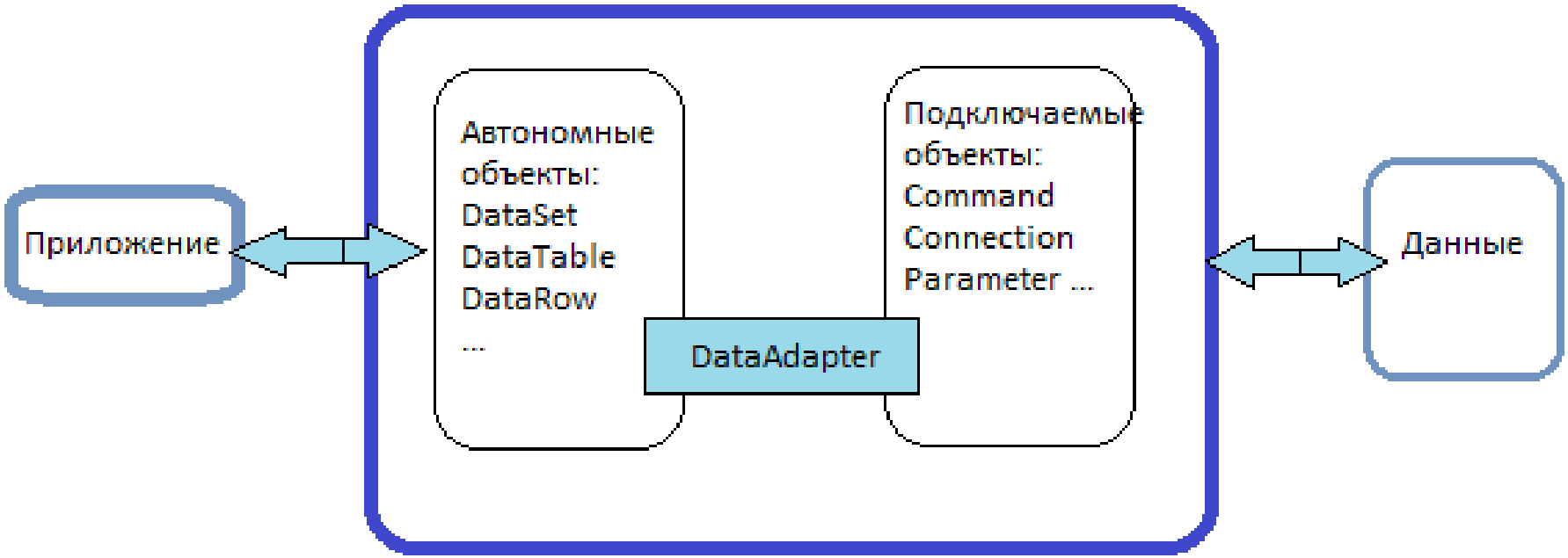
- Легко модифицировать, если необходима смена СУБД
- Изменяются только настройки доступа к данным
- Невозможность доступа к уникальной функциональности, специфичной для конкретной СУБД
- Снижение производительности приложений
- Усложнение процедуры поставки приложения

Microsoft Data Access Components

- ODBC – Open Database Connectivity
- OLE DB – Object Linking and Embedding Database
- ADO.NET

ADO.NET режимы работы с данными

- Постоянное подключение
- Отсоединенные данные



Постоянное подключение

- Установка соединения
- Подготовка и выполнение команды
- Работа с данными
  - чтение, запись
  - фильтрация, сортировка
  - тоже в пакетном режиме
  - блокировки, совместное использование
- Закрытие соединение и обработка ошибок

Постоянное подключение

- Работает в режиме удержания подключения к базе
- Обеспечивает максимальную гибкость и эффективность
- Обеспечивает минимальный расход оперативной памяти

Отсоединенные данные

- Загрузка данных с сервера
- Изменение данных в наборе на локальной машине
- Обновление данных на сервере на основе локальной копии

Отсоединенные данные

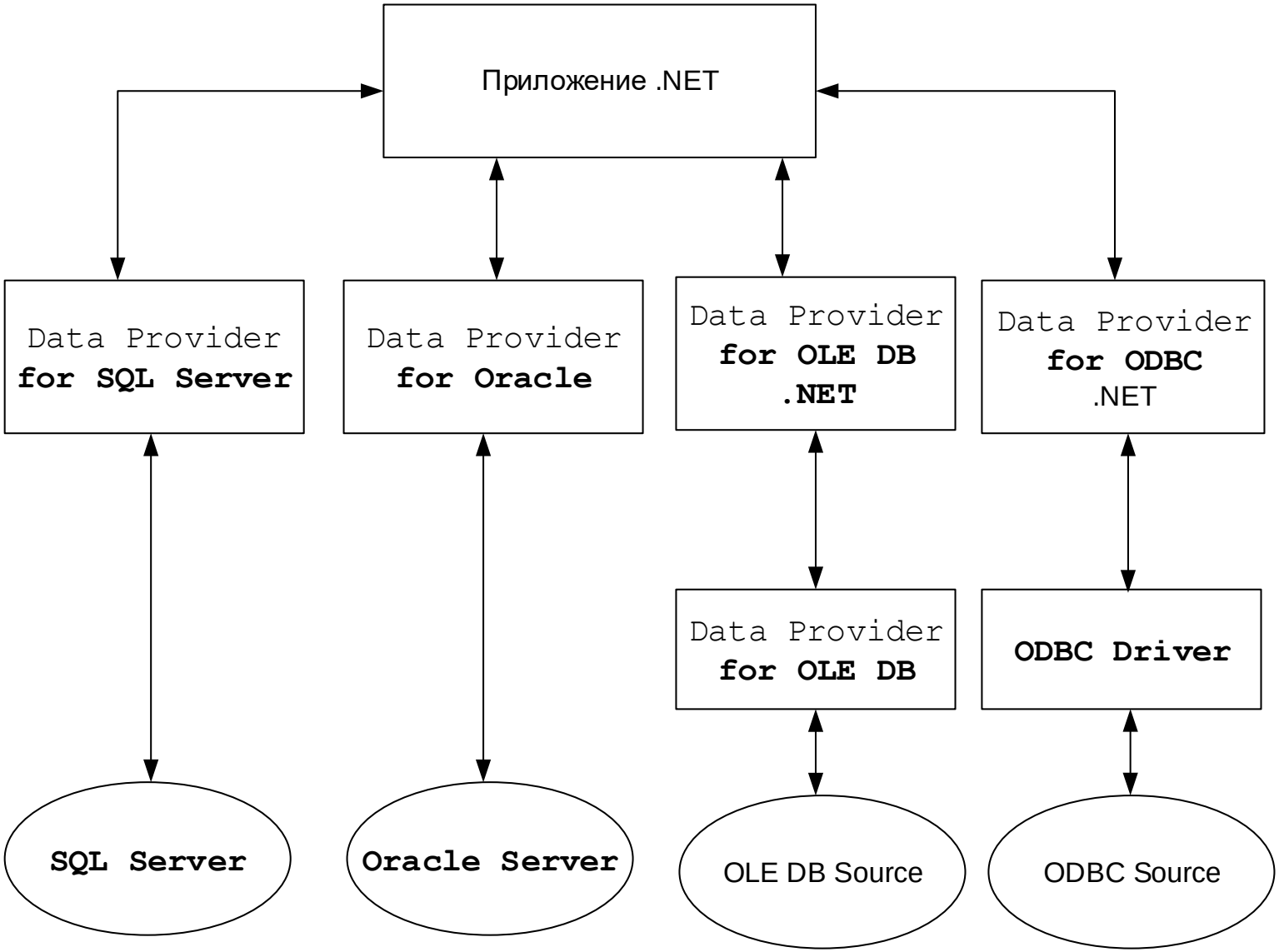
- Обеспечивает работу с данными в отсутствии подключения к БД
- Удобна для переноса данных по сети
- Расходует достаточно много памяти

Провайдеры данных

- Провайдеры (или поставщики) данных – извлечение данных из источника данных
  - SQL Server .NET Data Provider
  - Oracle Data Provider
  - ODBC.NET Data Provider
  - OleDB.NET Data Provider

```
''
using System;
using System.Data;
using System.Data.SqlClient;
using System.Data.OracleClient;
using System.Data.ODBC;
using System.Data.OleDb;
```

Провайдеры данных



Провайдеры данных

- Обеспечивают большую производительность
- Позволяют работать со специфическими типами данных для данной СУБД
- Лучше выполняют специфические для данной СУБД функции
- Можно использовать неспециализированного провайдера данных
- Можно определить набор провайдеров данных (модель ProviderBase)

Вопросы?