Введение в Entity Framework

Объектно-реляционное отображение

(Object-relational mapping - ORM) технология программирования, предназначенная для автоматического сопоставления и преобразования данных между таблицами реляционных СУБД и объектами классов.

ORM - системы автоматически генерируют SQL запросы для выполнения операций над данными при выполнении вызовов.

Функционал ORM систем:

- Создание объектной модели по БД
- Создание схемы БД по объектной модели
- Выполнение запросов к БД с помощью LINQ

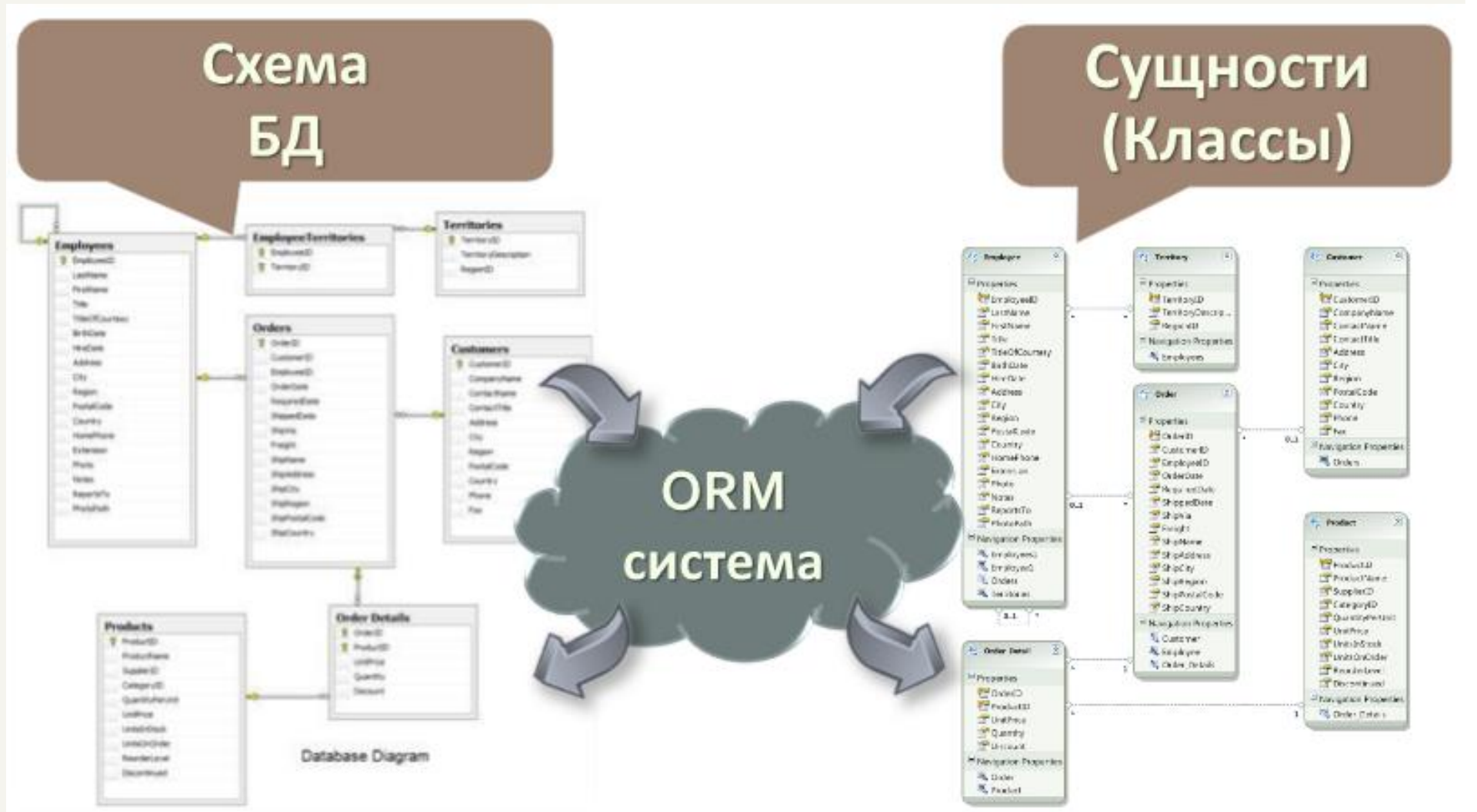


Схема
БД
Сущности
(Классы)
ORM
система
Database Diagram

Entity Framework (EF) - это объектно-реляционный модуль сопоставления (соответствия), позволяющий разработчикам .NET работать с реляционными данными с помощью специализированных объектов.

Entity Framework (EF):

- объектно-ориентированная технология доступа к данным;
- объектно-реляционное отображение (ORM) для .NET Framework от Microsoft.

Entity Framework - представляет собой «обёртку» для ADO.NET по взаимодействию с базой данных, базирующуюся на ORM

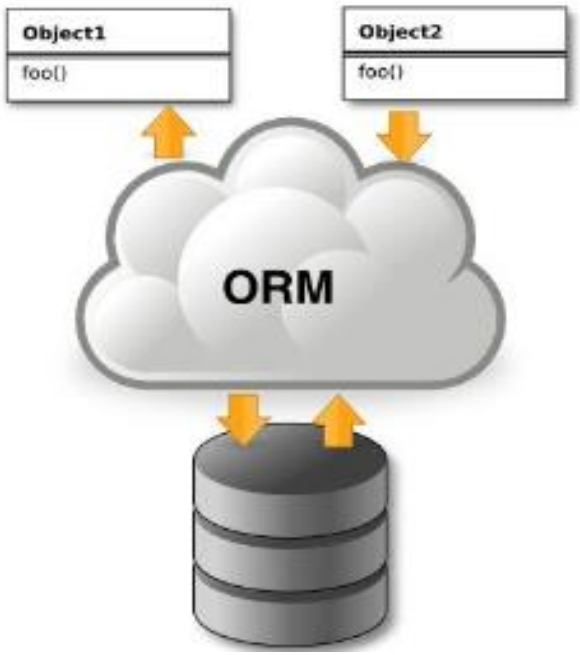
История:

- Первая версия **EntityFramework 1.0** вышла в 2008 году и представляла ограниченную функциональность, базовую поддержку **ORM** и один подход к взаимодействию с БД – **«DatabaseFirst»**.
- В **2010** году вышла **EntityFramework 4.0** версия и с этого времени EntityFramework **стал рекомендуемой технологией** для доступа к данным, а в Framework были введены новые возможности взаимодействия с БД - подходы **«ModelFirst»** и **«CodeFirst»**.
- Дополнительные улучшения функционала последовали с выходом версии **EntityFramework 5.0** в **2012** году.
- В **2013** году был выпущен **EntityFramework 6.0**, обладающий возможностью **асинхронного доступа** к данным.

История:


Visual Studio
2008 SP 1
Visual Studio
2010
Visual Studio
2012
Getting
Started
Entity Framework
4.0
Entity
Framework
Ver 5
EF 6
EF Version 1
EF Version 4
EF 4.1 - 4.3
EF Version 5
EF Version 6

Достоинства применения Framework



Основное достоинство:

- устраняет необходимость в написании большей части кода для доступа к данным, который обычно требуется разработчикам.

- Производительность (быстрота разработки)

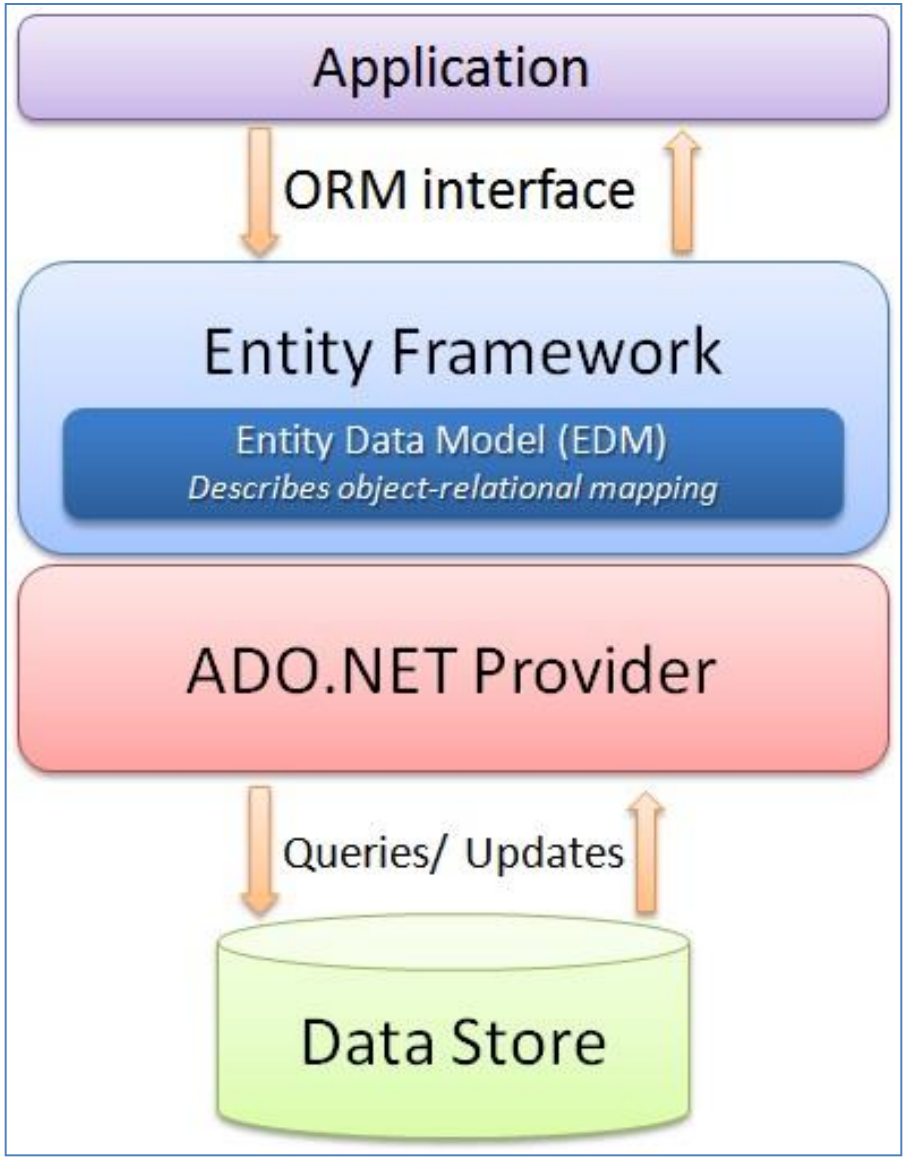
Microsoft любит того, у кого хорошее железо

- Дизайн приложения
- Повторное использование
- Сопровождаемость

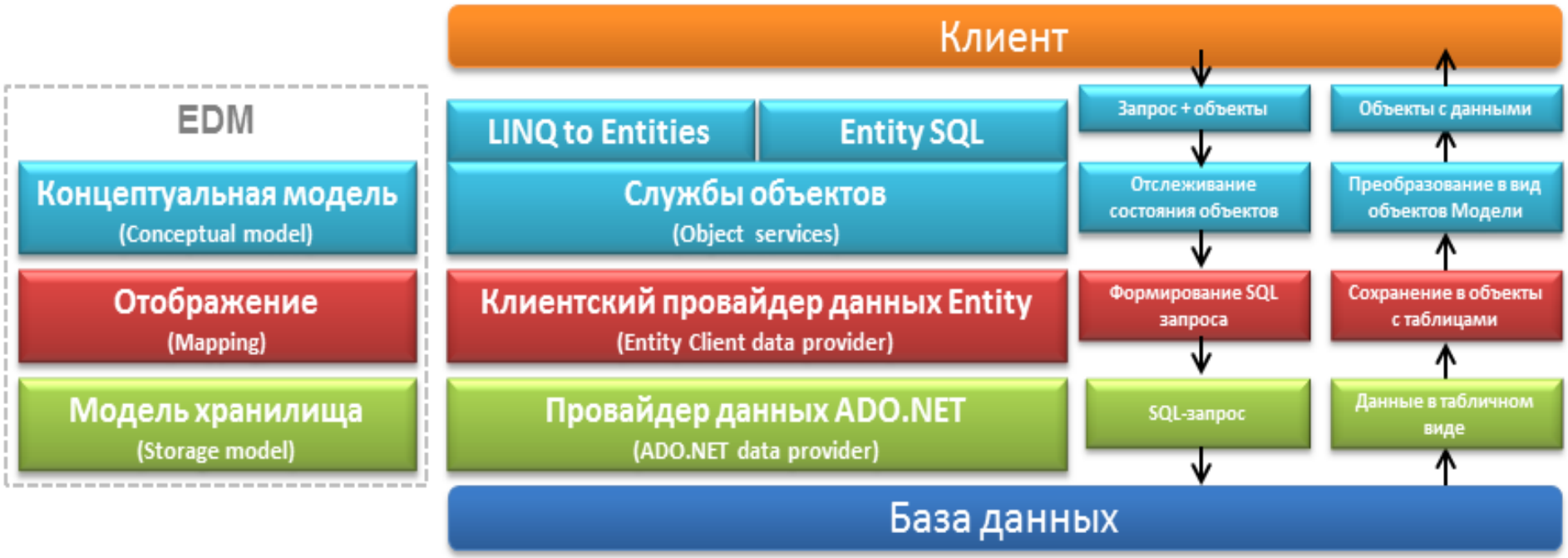
Основные возможности Entity Framework

- Сопоставление **таблиц базы данных** и связей между ними **в объекты .NET** и отношения между ними;
- Выполнение **запросов к базе данных** через работу с .NET объектами (**linq to entities**);
- Create / Read / Update / Delete
- Создание, изменение, удаление схемы базы данных.

Архитектура Entity Framework



Архитектура Entity Framework



Entity Data Model

Entity Data Model (EDM) представляет модель между объектной моделью (сущностями) и БД, согласно которой определяются правила соответствия объектов элементам базы данных

Строго типизированные классы, используемые для сопоставления базам данных, называются **сущностями**.

Entity Data Model

Основные составляющие EDM:

- **Концептуальная модель (Conceptual model)** определяет классы (сущности) и отношения между ними (при их наличии).
- **Отображение (Mapping)** описывает сопоставление между концептуальной моделью и моделью хранения
- **Модель хранилища (Storage model)** описывает таблицы, расположенные в реляционной базе данных.

Архитектура Entity Framework

Слой Службы объектов (Object Services)

Служба объектов управляет сущностями клиентской стороны при работе с ними в коде.

Служба объектов:

- отслеживают изменения, внесенные в сущность, управляет отношениями между сущностями в базе данных и сохраняет состояния сущности с помощью сериализации.

Служба объектов управляет любым классом за счет наследования от базового класса **EntityObject.**

Слой Клиентского провайдера данных (Entity Client data provider)

Клиентский провайдер данных Entity:

- организует работу с **поставщиком данных ADO.NET** для установки соединений с базой данных,
- генерирует необходимые SQL-операторы на основе состояния сущностей и запросов LINQ,
- отображает извлеченные данные из базы данных на сущности,
- другие внутренние операции.

Слой провайдера данных ADO.NET (ADO.NET data provider)

Слой провайдера данных ADO.NET:

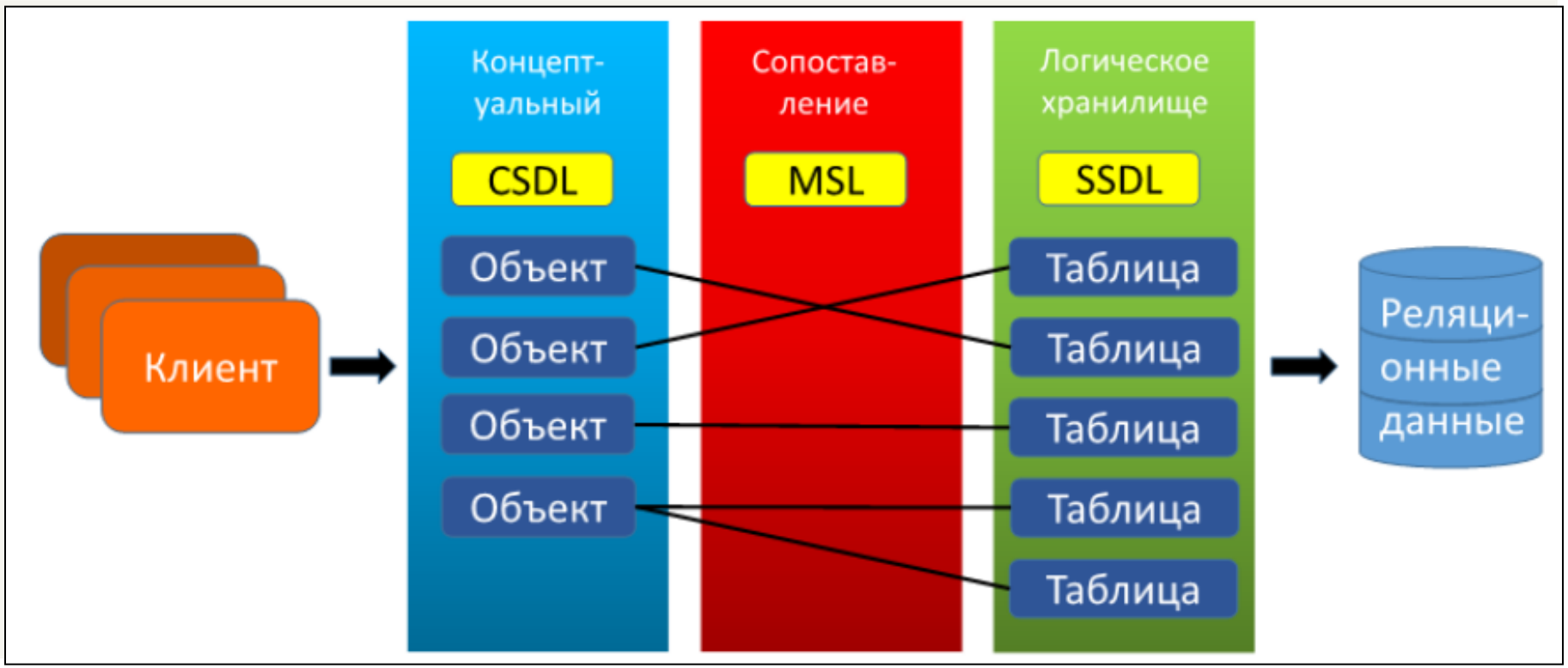
Провайдер данных ADO.NET, используется для непосредственного обращения к реляционной системе управления базами данных.

System. Data. EntityClient содержит классы EntityCommand и EntityConnection, EntityDataReader.

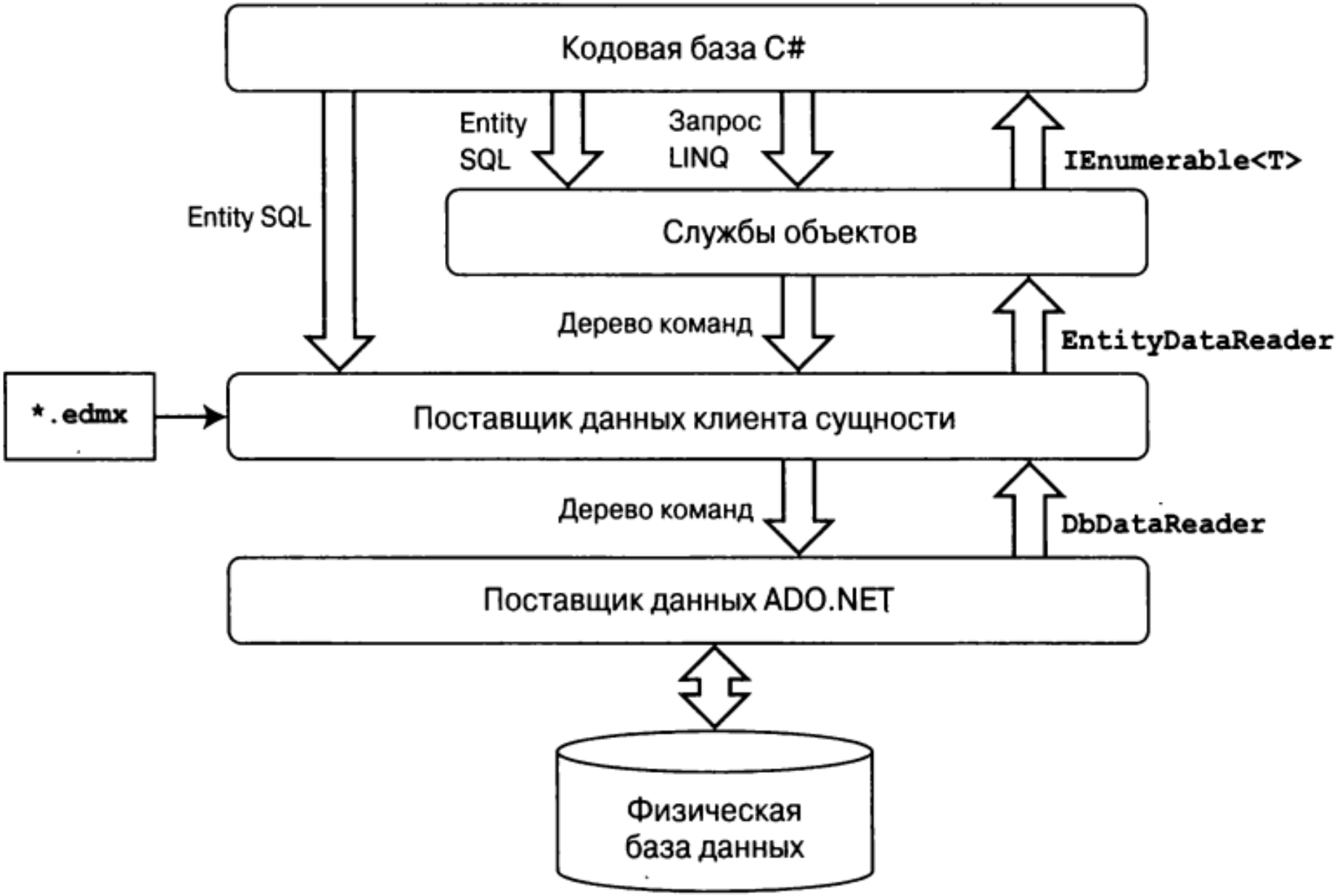
Архитектура модели EDM

Языки на основе XML, которые описывают концептуальную модель, модель хранения и сопоставление между этими моделями:

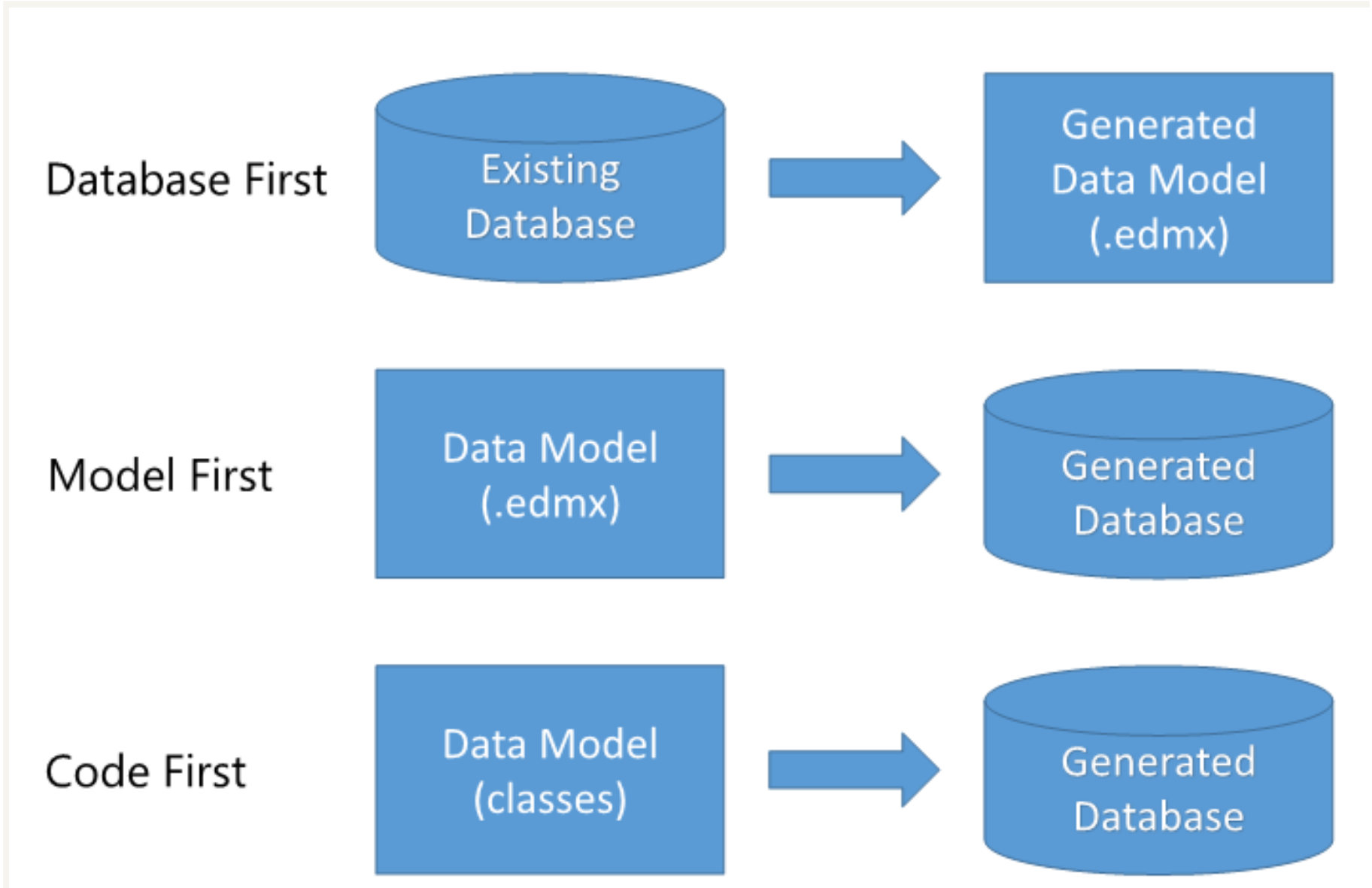
- язык определения концептуальной схемы (**CSDL**)
- язык определения схемы хранения (**SSDL**)
- язык определения сопоставлений (**MSL**)



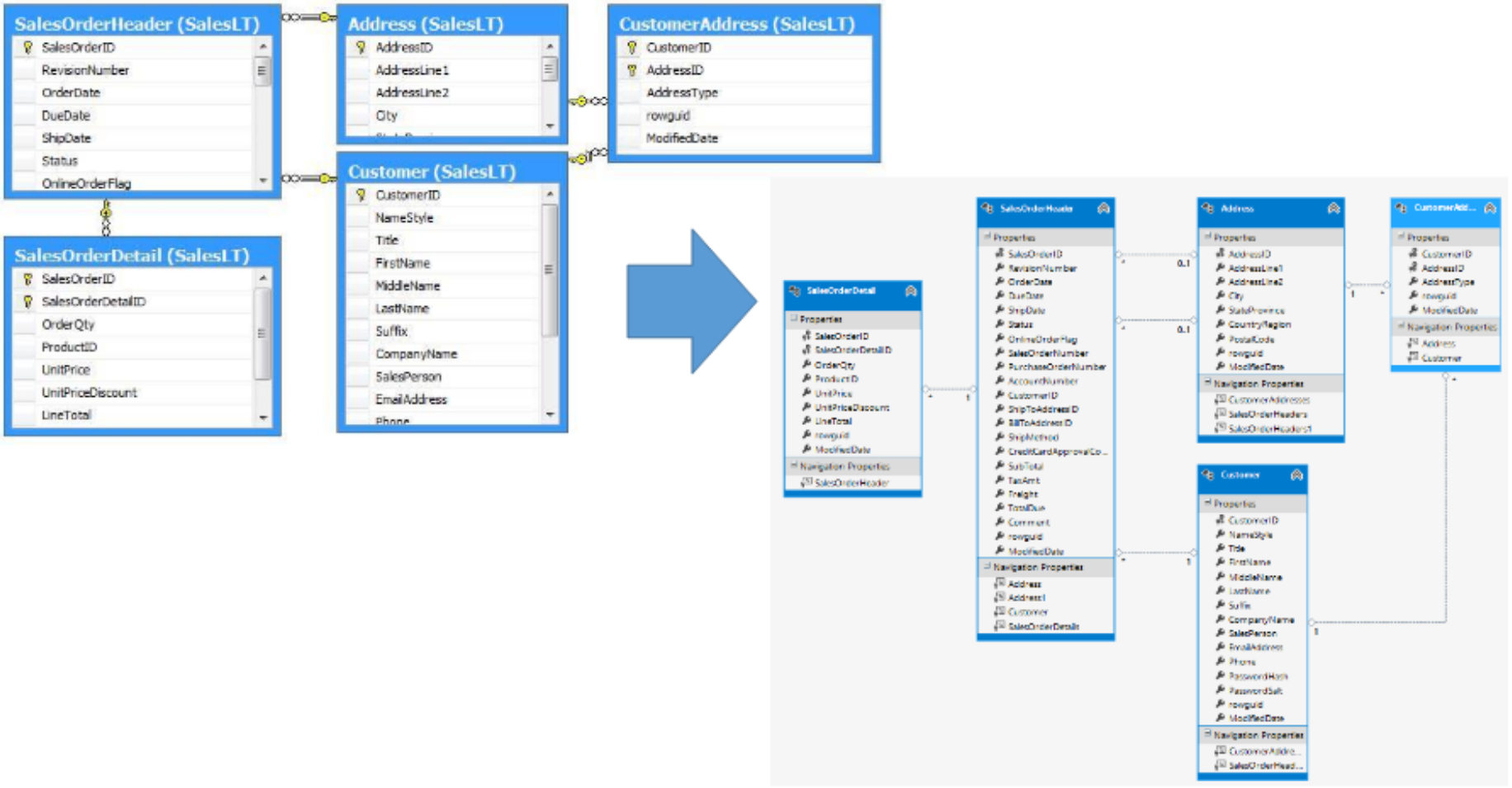

КЛИЕНТ
Запрос + объекты
Отслеживание состояния объектов
Формирование SQL запроса
SQL-запрос
Объекты с данными
Преобразование в вид объектов Модели
Сохранение в объекты с таблицами
Данные в табличном виде
БАЗА ДАННЫХ


Кодовая база C#
Entity SQL
Запрос LINQ
IEnumerable<T>
Entity SQL
Службы объектов
Дерево команд
EntityDataReader
*.edmx
Поставщик данных клиента сущности
Дерево команд
DbDataReader
Поставщик данных ADO.NET
Физическая база данных

Основные способы создания моделей

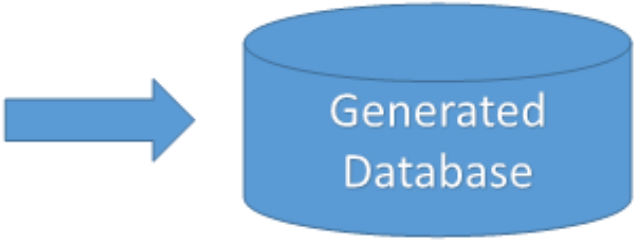


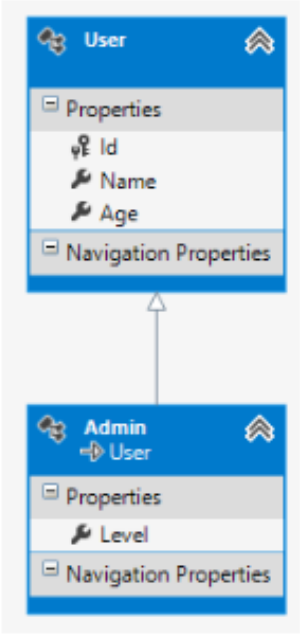
Database First



Model First

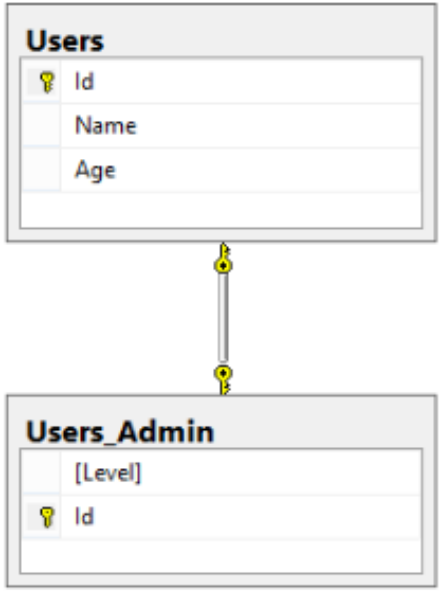
Model First






```
public partial class User
 {
      public int Id { get; set; }
      public string Name { get; set; }
      public DateTime? Age { get; set; }
 }

public partial class Admin : User
{
     public string Level { get; set; }
}
```



Демонстрация

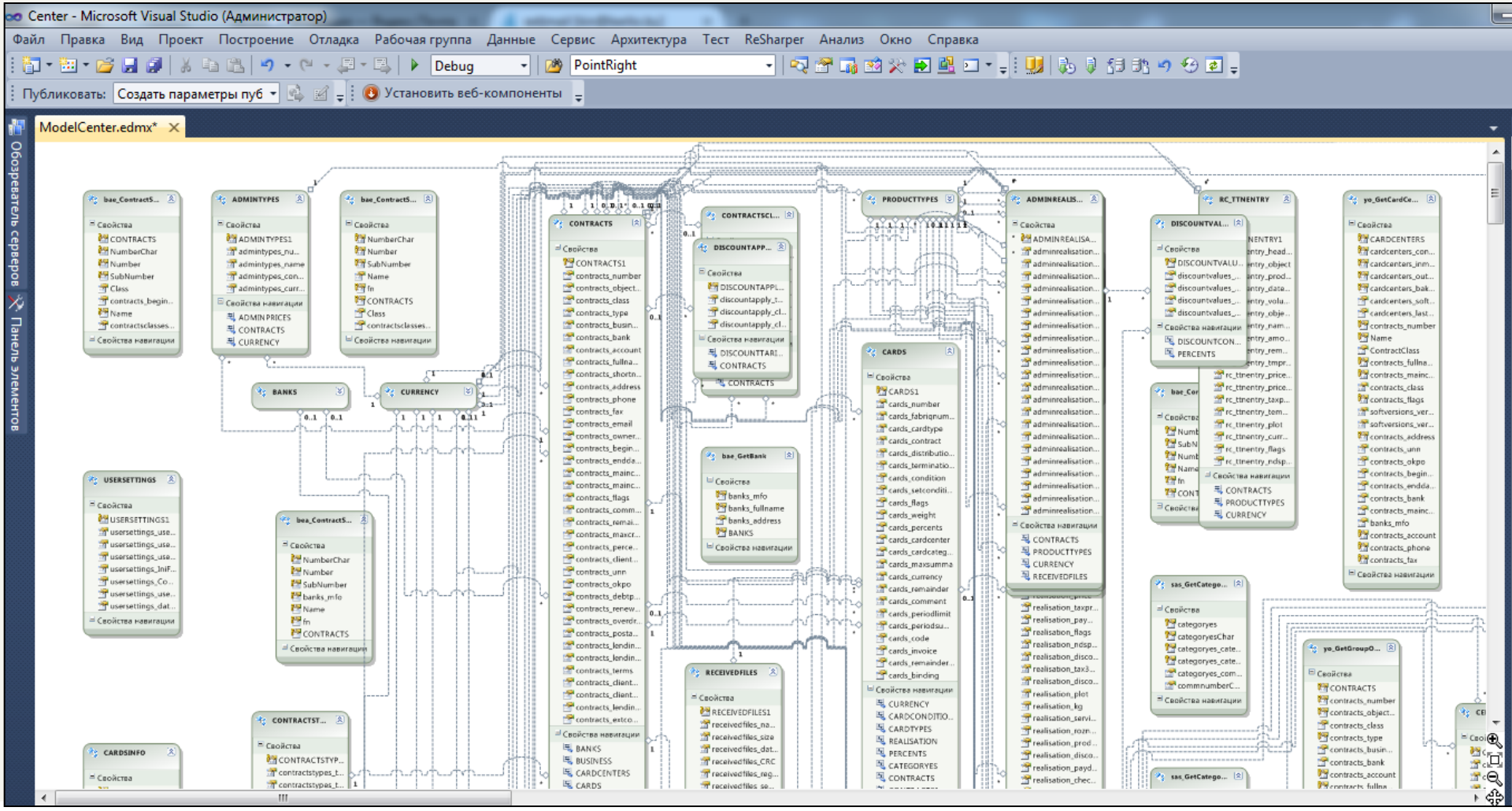
- DataBase First
- Model First

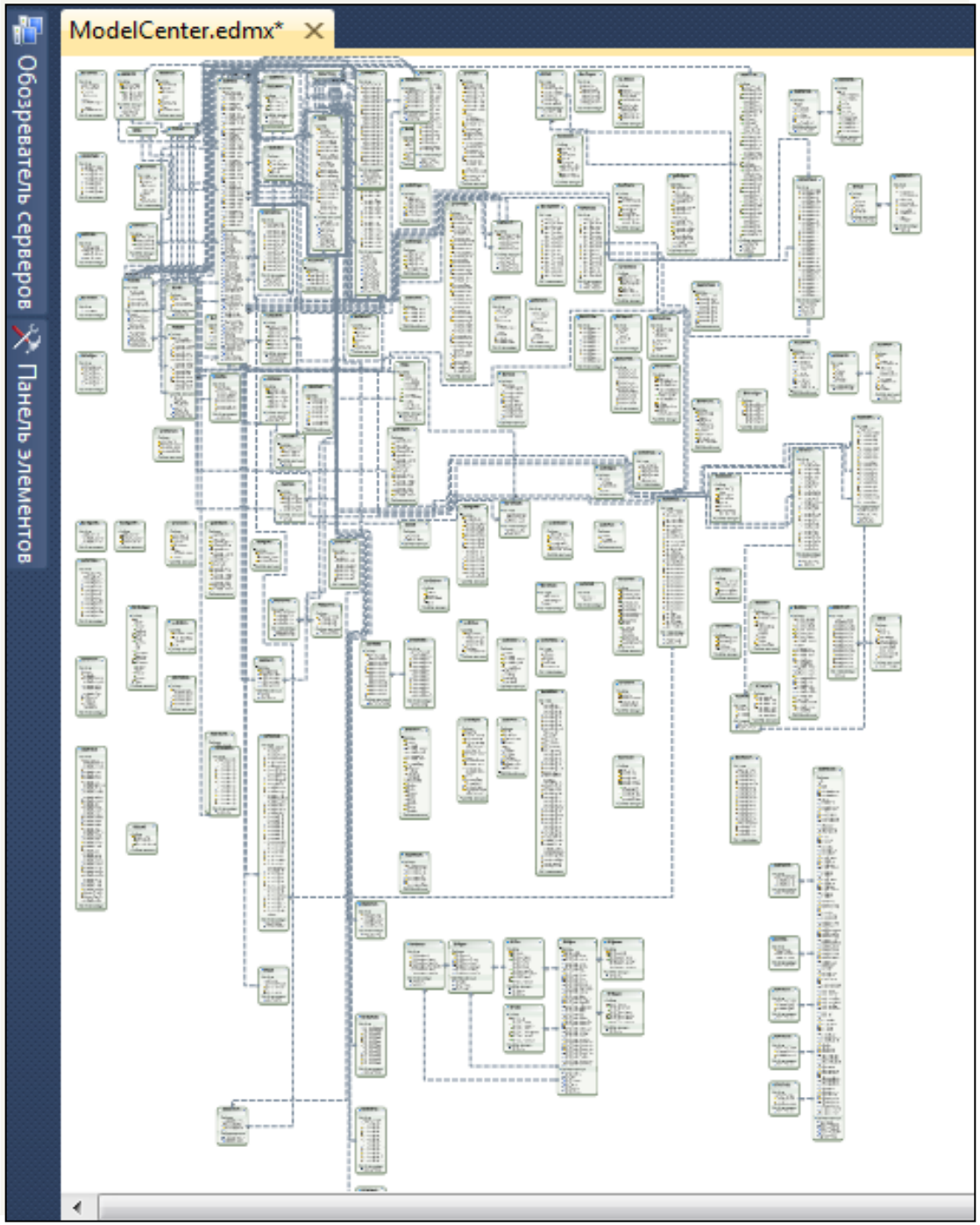
Основные операции для работы с данными

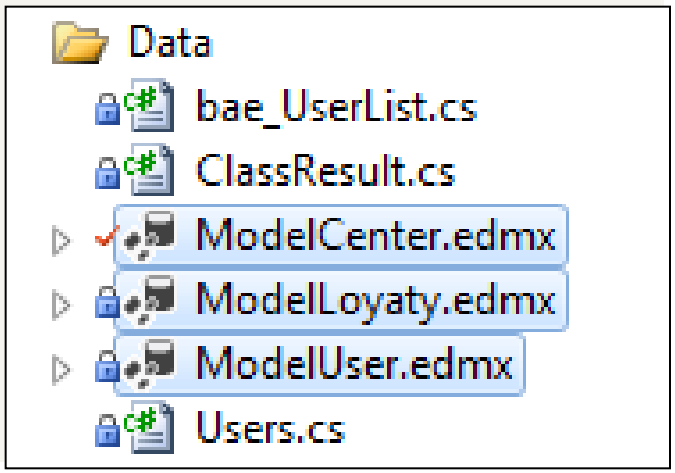
Создание, чтение, обновление и удаление данных.

- Create
- Read
- Update
- Delete

Entity Framework позволяет легко манипулировать данными благодаря следующим операциям:


Center - Microsoft Visual Studio (Администратор)
Файл Правка Вид Проект Построение Отладка Рабочая группа Данные Сервис Архитектура Тест ReSharper Анализ Окно Справка
Debug
PointRight
Публиковать: Создать параметры пуб
Установить веб-компоненты
ModelCenter.edmx*
Обозреватель серверов
Панель элементов
CONTRACTS
ADMINTYPES
BANKS
CURRENCY
USERSETTINGS
CARDSINFO
CONTRACTSTYPES
DISCOUNTAPPLIES
PRODUCTTYPES
CARDS
RECEIVEDFILES
ADMINREALISATIONS
DISCOUNTVALUES
CARDCENTERS


ModelCenter.edmx*
Обозреватель серверов
Панель элементов


Data
bae_UserList.cs
ClassResult.cs
ModelCenter.edmx
ModelLoyaty.edmx
ModelUser.edmx
Users.cs