

Проектирование баз данных. CASE-технологии проектирования

Быстренина Ирина Евгеньевна,
кандидат педагогических наук, доцент
кафедры прикладной информатики
РГАУ-МСХА им. К.А. Тимирязева



Концептуальная модель БД

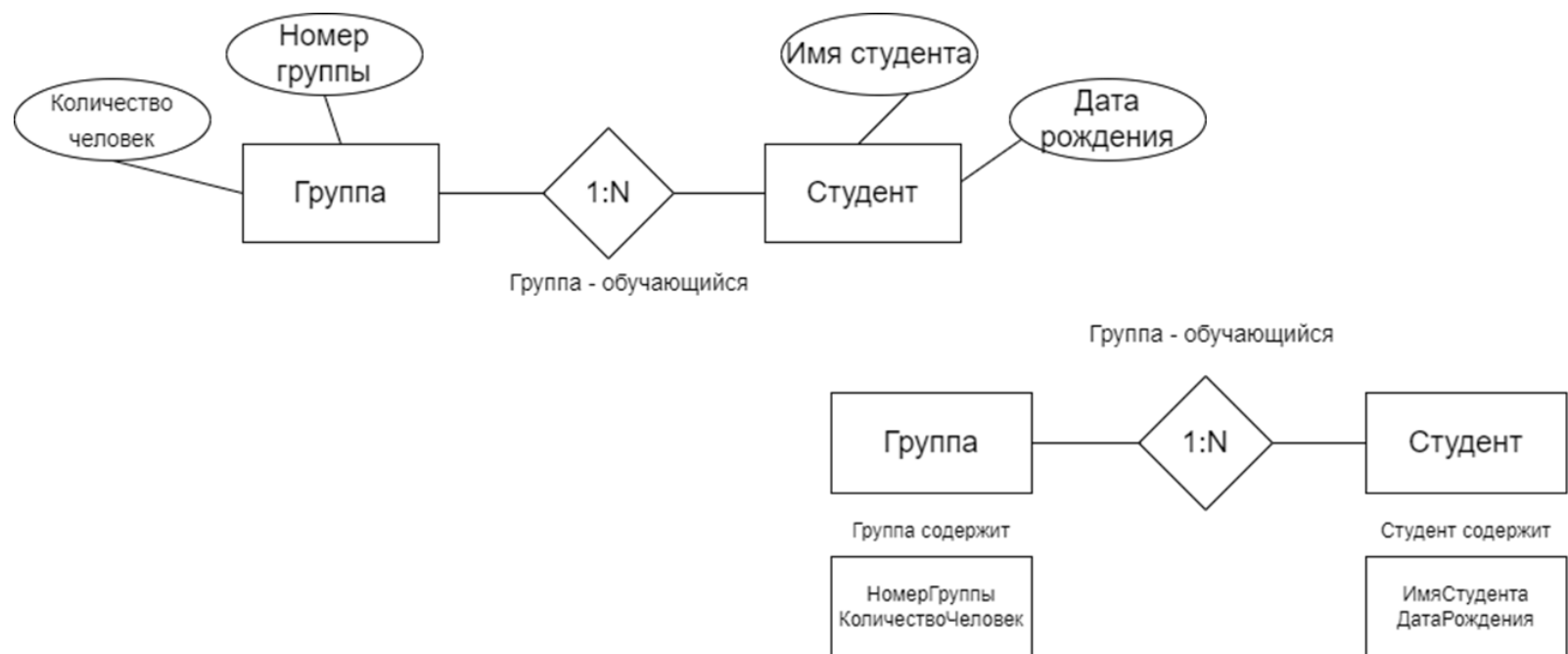


Рисунок 1 – ER-диаграмма с использованием нотации Питера Чена



Диаграмма «сущность-связь» в стиле UML

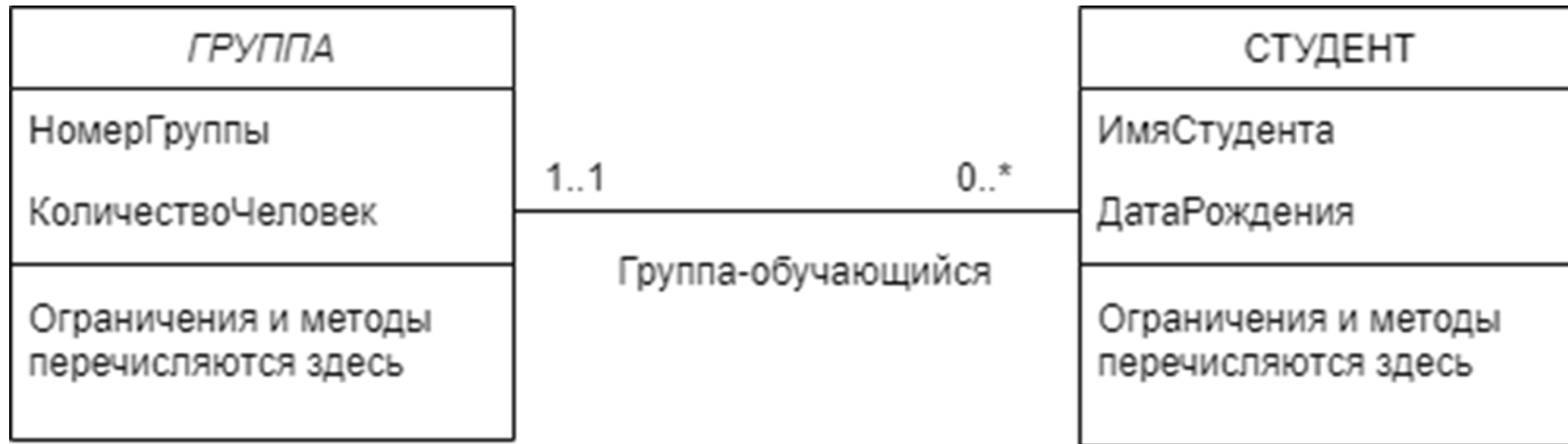
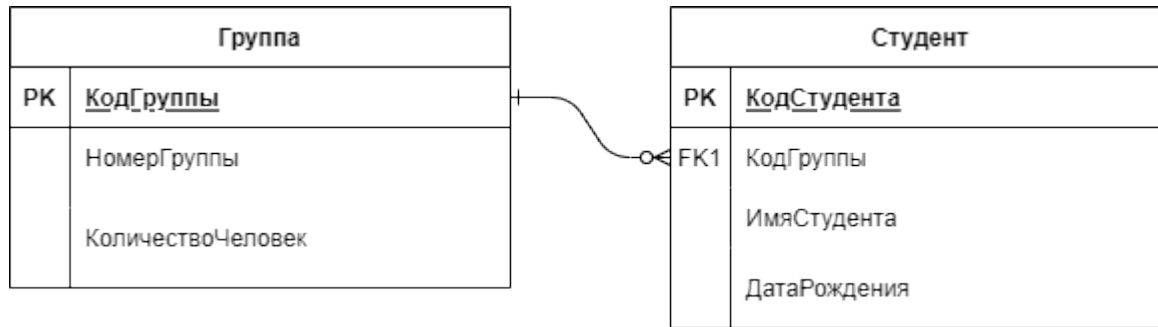


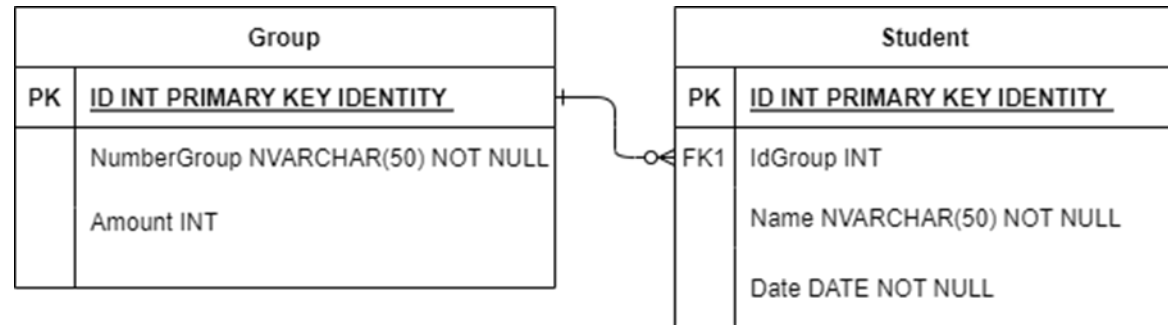
Рисунок 2 – ER-диаграмма с использованием нотации UML



Логическая модель БД



Физическая модель БД





CASE-средства – программные средства, поддерживающие процессы создания и/или сопровождения информационных систем, такие как: анализ и формулировка требований, проектирование без данных и приложений, генерация кода, тестирование, обеспечения качества, управление конфигурацией и проектом.

Обычно к CASE-средствам относят любое программное средство, автоматизирующее ту или иную совокупность процессов жизненного цикла программного обеспечения.



Case-система – набор Case-средств, имеющих определенное функциональное предназначение и выполненных в рамках единого программного продукта.

Case-технология – совокупность методологий анализа, проектирования, разработки и сопровождения сложных систем.



Основная цель CASE-систем и средств – отделить проектирование программного обеспечения от его кодирования и последующих этапов разработки (тестирования, документирования и пр.), а также автоматизировать весь процесс создания программных систем, или инжиниринг (от англ. engineering— разработка).



Различают **прямое проектирование (forward-engineering)** — процесс получения структуры базы данных для выбранной целевой СУБД на основе построенной ER-модели и **обратное проектирование (reverse-engineering — реверс-инжиниринг)** — когда ER-модель получается на основе существующей базы данных.

CASE-средства обычно поддерживают оба эти процесса.

Основные характерные особенности, которыми должны обладать CASE-средства



Мощные графические средства для описания и документирования информационных систем, обеспечивающие удобный интерфейс с разработчиком и развивающие его творческие возможности.

CASE-технологии обеспечивают всех участников проекта, включая заказчиков, единым строгим, наглядным и интуитивно понятным графическим языком, позволяющим получать обозримые компоненты с простой и ясной структурой.

Основные характерные особенности, которыми должны обладать CASE-средства



Использование специальным образом организованного хранилища проектных метаданных (репозитория).

Основа Case-технологий –использование базы данных проекта для хранения всей информации о проекте, которая может разделяться между разработчиками в соответствии с их правами доступа. Содержимое репозитория включает не только информационные объекты разных типов, но и отношения между их компонентами, а также правила использования или обработки этих компонентов.

Основные характерные особенности, которыми должны обладать CASE-средства



Интеграция отдельных компонент CASE-средств, обеспечивающая управляемость процессом разработки информационных систем.

На основе репозитория осуществляется интеграция CASE- средств и разделение системной информации между разработчиками. При этом возможности репозитория обеспечивают несколько уровней интеграции:

- общий пользовательский интерфейс по всем средствам;
- передачу данных между средствами;
- интеграцию этапов разработки через единую систему представления фаз жизненного цикла;
- передачу данных и средств между различными платформами.

Основные характерные особенности, которыми должны обладать CASE-средства



Поддержка коллективной разработки и управления проектом.

Case-технология поддерживает групповую работу над проектом, обеспечивая возможность работы в сети, экспорт и импорт любых фрагментов проекта для их развития и/или модификации, а также планирование, контроль, руководство и взаимодействие. Эти функции также реализуются на основе репозитория.

Основные характерные особенности, которыми должны обладать CASE-средства



Макетирование (прототипирование) — возможность быстро строить макеты (прототипы) будущей системы, что позволяет заказчику на ранних этапах разработки оценить степень соответствия ее собственным требованиям.



Классификация CASE-технологий

По ориентации на этапы жизненного цикла:

- Средства анализа, предназначенные для построения и анализа моделей предметной области;
- Средства анализа и проектирования, обеспечивающие создание проектных спецификаций;
- Средства проектирования баз данных, обеспечивающие моделирование данных и разработку схем баз данных для основных СУБД;
- Средства разработки приложений.



Классификация CASE-технологий

По функциональной полноте:

- Системы, предназначенные для решения частных задач на одном или нескольких этапах жизненного цикла;
- Интегрированные системы, поддерживающие весь жизненный цикл информационных систем и связанные с общим репозиторием.



Классификация CASE-технологий

По типу используемых моделей:

- Структурные – основаны на методах структурного и модульного программирования, структурного анализа и синтеза;
- Объектно-ориентированные – получили массовое использование с начала 1990-х гг, позволяют сократить сроки разработки, а также повысить надежность и эффективность функционирования информационных систем;
- Комбинированные – поддерживают одновременно структурные и объектно-ориентированные методы.



Классификация CASE-технологий

По степени независимости от СУБД:

- Независимые системы — поставляются в виде автономных систем, не входящих в состав конкретной СУБД;
- Встроенные в СУБД — обычно поддерживают главным образом формат баз данных СУБД, в состав которой они входят (возможна поддержка и других форматов баз данных).



AllFusion Erwin Data Modeler

- Предназначен для проектирования и документирования баз данных;
- Предназначен для всех компаний, разрабатывающих и использующих базы данных, администраторов баз данных, системных аналитиков, проектировщиков баз данных, разработчиков, руководителей проектов;
- Удобная в использовании графическая среда упрощает разработку базы данных и автоматизирует множество трудоемких задач.



Power Designer компании Sybase содержит модули:

- Process Analyst – средство для функционального моделирования.
- Data Analyst - инструмент для построения модели «сущность-связь», автоматической генерации на ее основе реляционной структуры.
- Application Modeler - инструмент для автоматической генерации прототипов программ обработки данных на основе реляционных моделей, построенных в DataAnalyst.



Open ModelSphere

это инструмент моделирования данных, процессов и UML, написанный на Java и распространяемый как бесплатное программное обеспечение под лицензией GPL. Он обеспечивает поддержку прямого и обратного проектирования между UML и реляционными схемами.

Процесс моделирования в Open ModelSphere базируется на методологии проектирования реляционных баз данных IDEF1X, которая определяет стандарты терминологии и графического изображения типовых элементов на ER-диаграммах.

Open ModelSphere

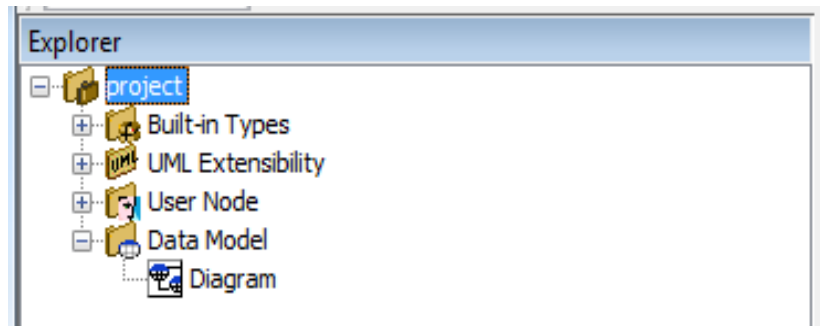


Рисунок 3 – Окно проекта

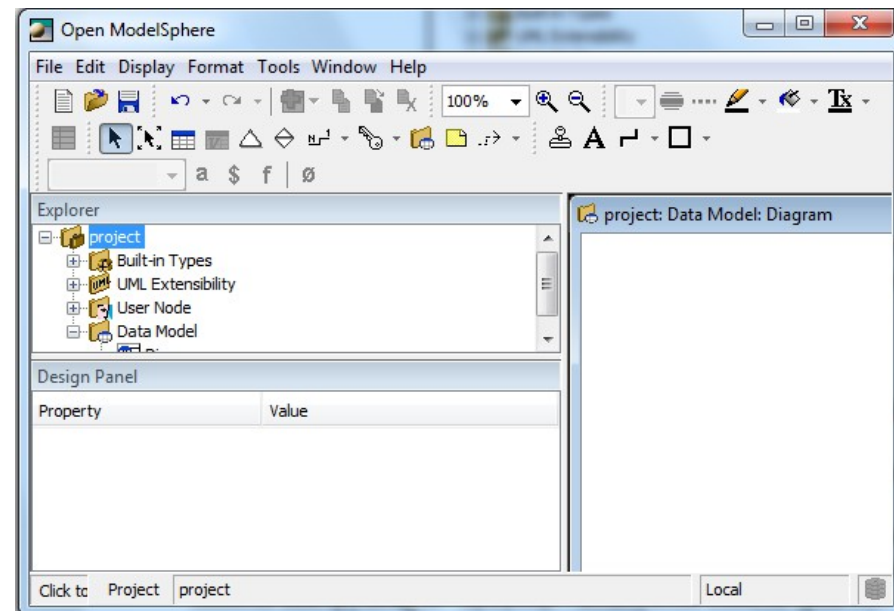


Рисунок 4 – Рабочее пространство Open ModelSphere

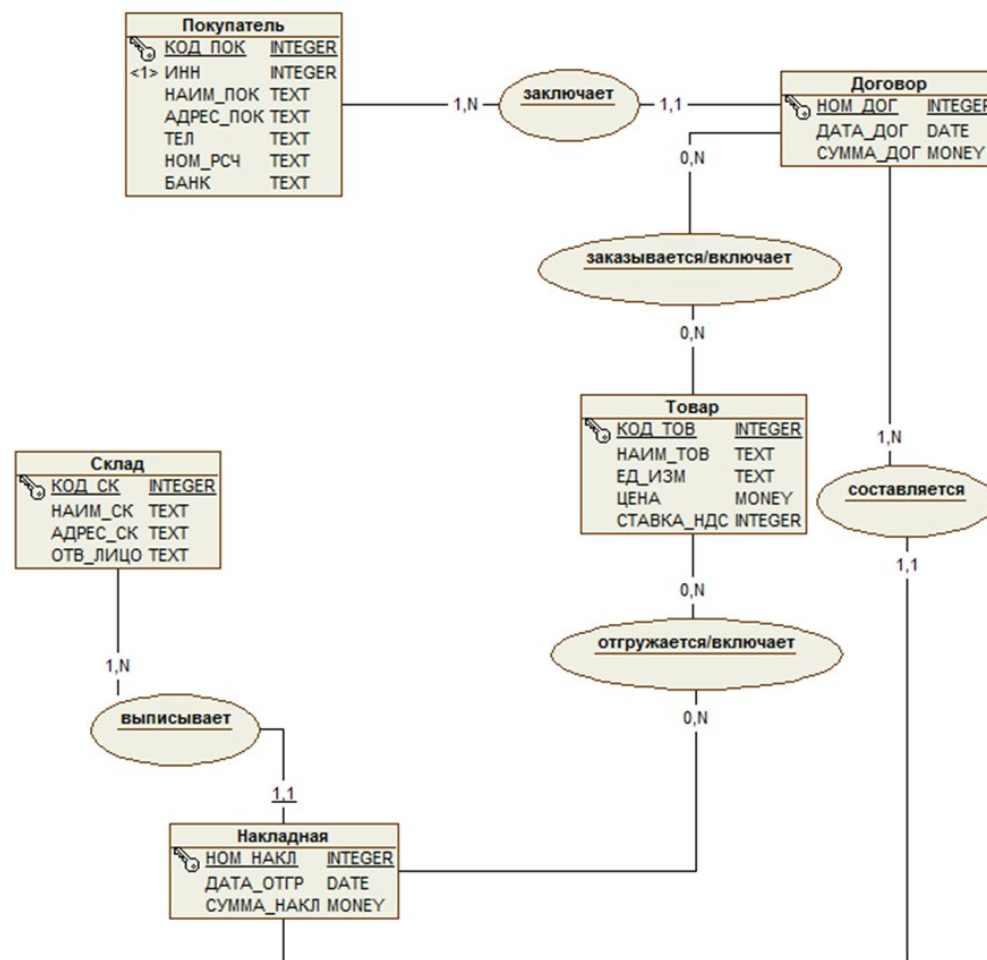


Рисунок 5 – Концептуальная модель данных

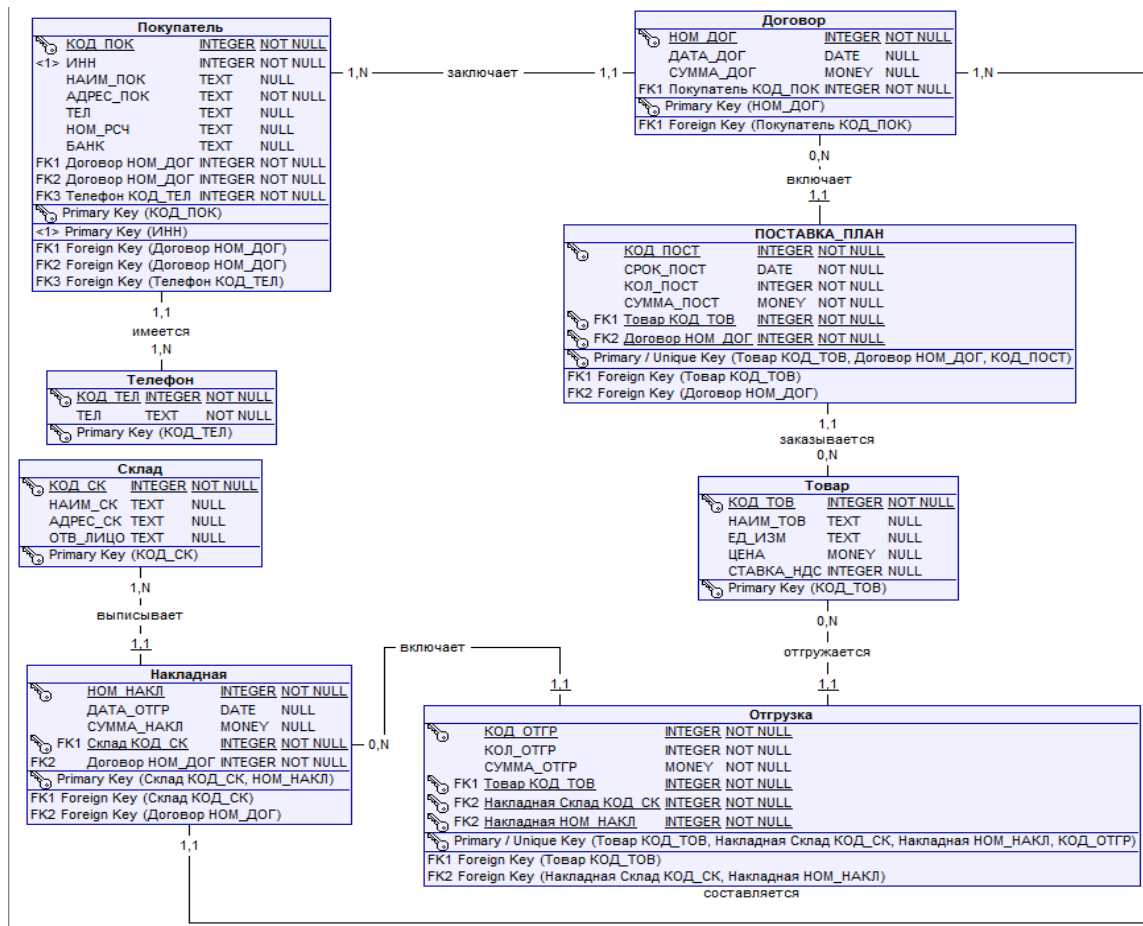


Рисунок 6 – Логическая модель данных

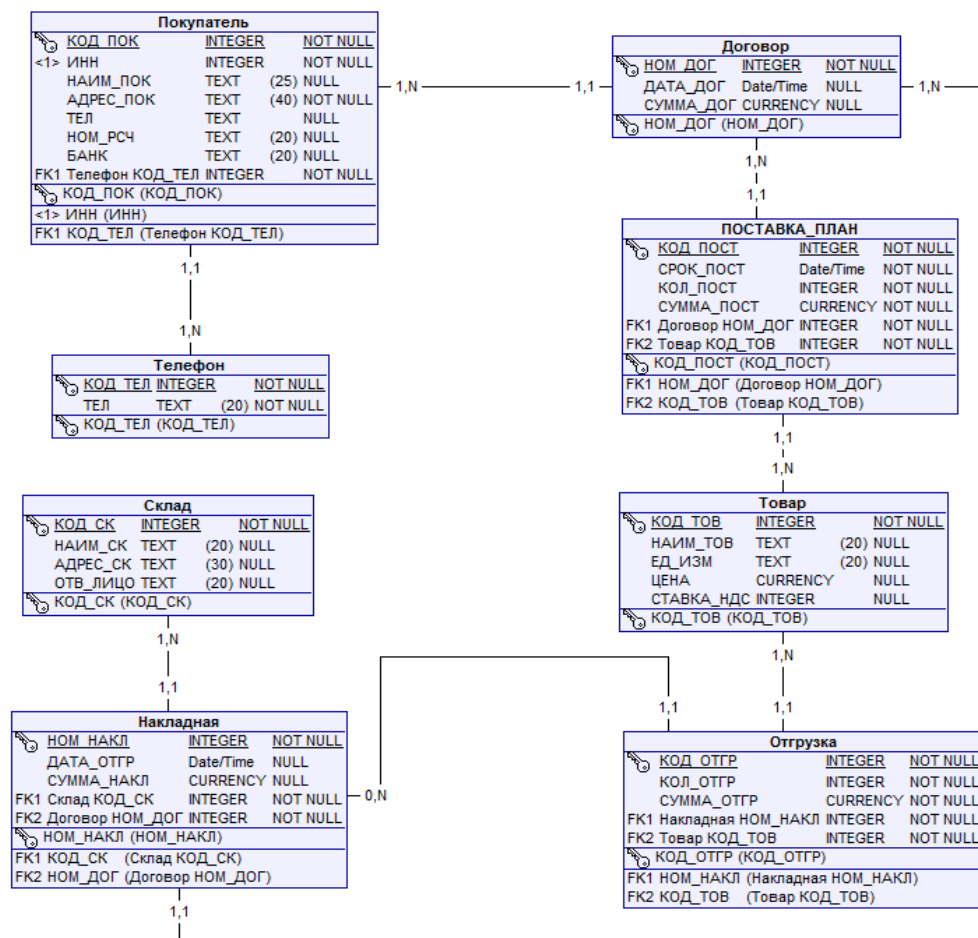


Рисунок 7 – Физическая модель данных

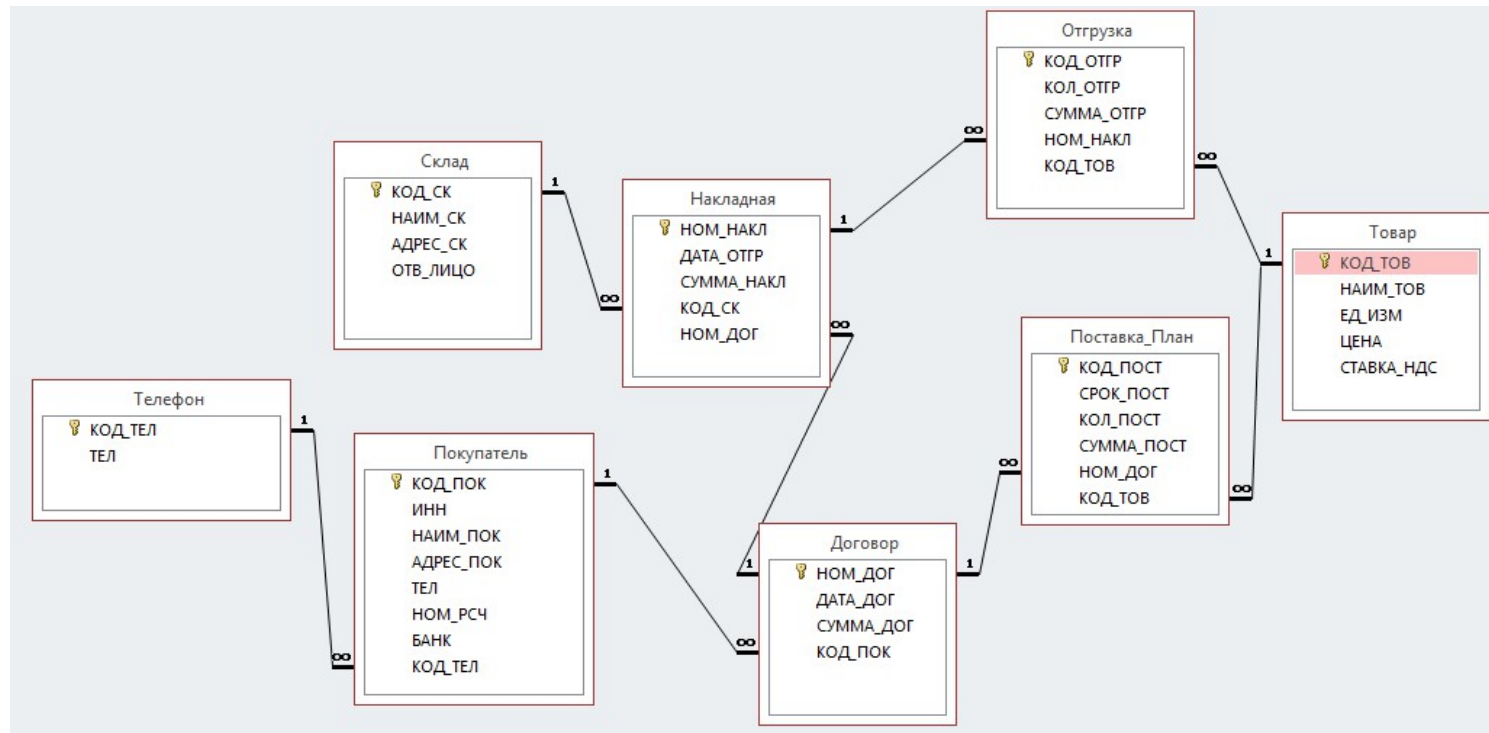


Рисунок 8 – Схема базы данных



Спасибо за внимание!