1. **Почему появляются нереляционные модели данных.**

Реляционные модели данных являются на сегодняшний день стандартом представления баз данных. Основные характеристики реляционной модели – хранение данных в плоских таблицах (отношениях), удовлетворяющих условиям:

* Первая нормальная форма – все атрибуты представляются простыми типами данных;
* Отсутствуют записи-дубликаты;
* Нет необходимости в фиксации порядка записей;
* Нет необходимости в фиксации порядка атрибутов.

Дополнительно отслеживаются вопросы целостности – целостности сущности (отсутствие кортежей-дубликатов) и ссылочной целостности (реализация связи между таблицами посредством внедрения первичного ключа одной таблицы в другую).

Преимущества реляционных моделей:

* Персистентность – долговременное хранение данных;
* Параллельность – управление транзакциями для обеспечения одновременной работы множества пользователей;
* Интеграция – возможность хранения на одном сервере баз данных множества приложения или возможность хранения в одной базе данных информации, предназначенной для множества приложений.
* Почти стандартная модель плюс универсальный стандартный язык обращения к данным SQL, поддержка которого внедрена почти во все языки программирования.

Недостатки реляционных моделей:

* Главная проблема – потеря соответствия способа хранения данных (таблицы) и способа представления данных в программах (объектная модель). Частично проблема была решена с помощью технологии объектно-реляционного связывания (ORM), но в этом случае возникал промежуточный слой управления в виде контроллеров доступа к данным;
* Избыточная нормализация приводит к увеличению времени обработки запросов для ситуаций выбора из многих таблиц (посредством подзапросов или операций соединения таблиц). Частично в ряде приложений решилась в концепции хранилищ данных, в которых требования нормализации нет.
* Необходимость хранения больших объемов данных приводит модели распределенных баз данных (хранение информации на нескольких серверах – горизонтальное масштабирование). Проблема заключается в больших затратах на обеспечение целостности данных (например, при добавлении новой записи требуется проверить отсутствие заданного ключа на всех серверах). Эти проблемы можно решить с помощью специальных соглашений о ключах (включение идентификаторов сервера и пр.). Другая проблема – распределенное хранение связанной информации (при соединении таблиц возможно придется обращаться к нескольким серверах).

Поэтому постоянно рассматриваются вопросы, связанные с некоторым отходом от стандартов реляционной модели в целях обеспечения эффективности поиска данных.

На данный момент выделяются:

* Объектно-реляционные модели, в которых частично отсутствуют требования к таблицам, за счет чего появляются возможности программно работать в объектно-ориентированном стиле;
* Базы данных NoSQL (понятие введено в 2009 г. Йоханом Оскарссоном) – агрегатные базы данных: базы данных «ключ-значение», документные базы данных, семейства столбцов; графовые базы данных. Главная цель – хранить рядом те данные, которые связаны друг с другом. Еще один плюс – отсутствие фиксированной структуры данных, возможность ее смены в процессе использования.

Неряляционные подходы не могут обеспечить универсальность, но для частных случаев задач позволяют существенно улучшить эффективность поиска.