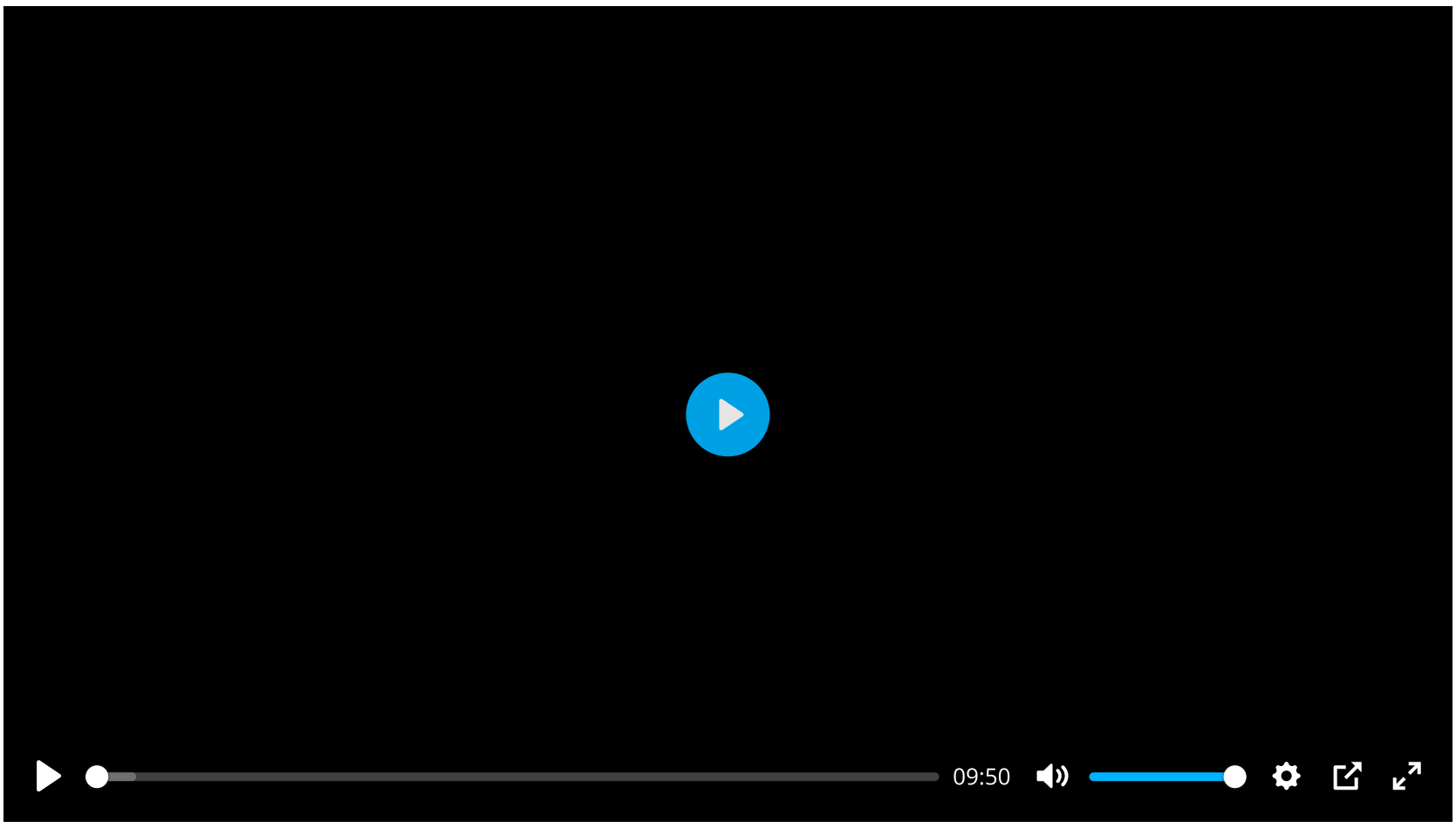
5.5 ORM, события и история



Текстовая расшифровка видео:

ХРАНЕНИЕ ИСТОРИЧЕСКИХ ДАННЫХ

План:

- Идем от бизнеса;
- Значения, которые не меняются никогда;
- Метод активной записи: SCD2;
- Новая колонка: SCD3;
- Табличка с историей: SCD4.

Идем от бизнеса

Изначально все вопросы на эту тему исходят от бизнеса.

Вопрос: за какой период мы в принципе храним данные? И за какой период в прошлое нам могут прилететь обновления?.

Ответ: предположим, что мы останавливаемся на промежутке в год, в течение которого бизнесу могут понадобиться данные. Когда данные прилетают от бизнеса, они могут быть в неотсортированном порядке. Возможна ситуация, в которой нам прилетят изменения в данных, запись которых была проведена сравнительно давно. В дизайне структуры нашей базы мы можем это как-то учитывать.

Вопрос: широкие денормализованные таблицы удобнее для аналитики, но как избежать траты места?

Ответ: если мы используем широкое хранилище (ClickHouse, Druid) в качестве DWH, храним там широкие базы, то нам важен промежуток времени, за который мы их храним. От этого зависит настройка retention. База может начать замедляться из-за переполненности данными. Подобные базы умеют работать на больших объемах (от терабайтов до петабайтов). Современные исследования доказали, что большинство бизнесов хранят данные с большим запасом, однако 99% всех аналитических запросов, выполняющихся на этих данных, затрагивают последние несколько процентов.
Вопрос: для каких значений нам в принципе нужен time travel?

Ответ: далеко не для всех мы можем отслеживать историю. Например, мы можем отслеживать историю для клиентов (история взаимодействия клиента с компанией). Есть случаи, когда time travel вовсе не нужен. Например, когда мы раз в год выгружаем справочник государственных кодов. Нам все равно, как они менялись, нам нужна лишь информация, которую мы можем достать из этих кодов.

Значения, которые не меняются никогда

Иногда могут быть значения, которые мы принимаем неизменяющимися никогда. Например, тот же справочник, который содержит неизменную, по нашему мнению, информацию. Если же изменения выходят, то старый справочник можно удалить и залить новый, перезаписав все.

Для того, чтобы описывать подобные данные, существует набор практик «**SCD**» (Slowly Changing Dimensions). Это подход, при котором мы либо датасет никогда не трогаем, либо перезаписываем целиком:

- **SCD Type 0** – подход, при котором мы не трогаем датасет.
- **SCD Type 1** – подход, при котором перезаписываем датасет целиком.

Метод активной записи: SCD2

Чаще всего мы сталкиваемся с другими типами SCD:

- SCD2;
- SCD3.

SCD2 – это всеобъемлющий вариант, где есть специальные колонки, по которым можно отслеживать исторические изменения.

Добавляется специальная отдельная колонка, например, с **номером версии** объекта, хранящегося в базе, или **временем действия**. Также может быть поле «**Effective date**» (на какой момент состояние объекта считалось эффективным) и «**Current flag**» (эффективен ли объект до сих пор или уже нет):

Supplier_Key	Supplier_Code	Supplier_Name	Supplier_State	Version
123	ABC	Acme Supply Co	CA	0
124	ABC	Acme Supply Co	IL	1

Supplier_Key	Supplier_Code	Supplier_Name	Supplier_State	Start_Date	End_Date
123	ABC	Acme Supply Co	CA	2000-01-01T00:00:00	2004-12-22T00:00:00
124	ABC	Acme Supply Co	IL	2004-12-22T00:00:00	NULL

Supplier_Key	Supplier_Code	Supplier_Name	Supplier_State	Effective_Date	Current_Flag
123	ABC	Acme Supply Co	CA	2000-01-01T00:00:00	N
124	ABC	Acme Supply Co	IL	2004-12-22T00:00:00	Y

Новая колонка: SCD3

Существуют и другие варианты. Существует **SCD3**, в котором каждый раз может создаваться новое поле, хранящее предыдущее состояние. В рамках одной строчки мы храним текущую версию объекта и предыдущую. Такой подход называют «**Alternate Reality**», когда объект существует одновременно в нескольких версиях:

Supplier_Key	Supplier_Code	Supplier_Name	Original_Supplier_State	Effective_Date	Current_Supplier_State
123	ABC	Acme Supply Co	CA	2004-12-22T00:00:00	IL

Табличка с историей: SCD4

Существует и **SCD4**.

В целом, существует множество SCD (около двенадцати).

SCD4 – часто использующийся подход, когда есть табличка с объектами, но при этом есть и отдельная табличка, в которой хранится информация о том, когда какой объект стал актуальным, когда по нему изменилась информация.

Подобный подход используют при проектировании хранилищ «**Data Vault**».

Когда создается отдельная табличка, она называется Point-in-Time Tables. В ней хранятся исторические изменения разных объектов:

Supplier

Supplier_Key	Supplier_Code	Supplier_Name	Supplier_State
124	ABC	Acme & Johnson Supply Co	IL

Supplier_History

Supplier_Key	Supplier_Code	Supplier_Name	Supplier_State	Create_Date
123	ABC	Acme Supply Co	CA	2003-06-14T00:00:00
124	ABC	Acme & Johnson Supply Co	IL	2004-12-22T00:00:00

Как вам урок?

Изучил, далее