АСКОУ DWH

# Принципы хранения данных в DWH

Хранилище делится на несколько баз данных / схем, каждая из которых соответствует модулю транзакционной части АСКОУ:

1. CORE – HR
2. ASSETS – inventory
3. MAINT – infrastructure
4. TRACKING – tracking
5. IOT – iot

Основой DWH является Postgresql с расширением TimescaleDB, предназначенном для оптимизации хранения данных с привязкой ко времени.

Вторым расширением является Citus, который позволяет хранить данные в таблицах в по-колонковом порядке, что дает прирост в производительности при аналитических обработках.

# Типы сущностей для хранения в DWH

1. По типу факт/размерность.
   1. Факты – события, которые происходят в системе: бизнес-транзакция, считывание показаний датчиков. Как правило, данные таких таблиц переносятся из OLTP базы в DWH, дельта переноса обычно определяется временем хранения данных в OLTP.
   2. Измерения – описание фактов: единица измерения, валюта, дата документа. Как правило, такие таблицы синхронизируются с OLTP системой. Синхронизация проводится по принципу выгрузки дельты изменений из OLTP в DWH.
2. По типу хранения (таблицы фактов):
   1. Архивные данные. Хранятся определенный период в качестве источника данных для непредвиденных восстановлений информации в других типах. Или согласно регламентированных сроков хранения данных.
   2. Данные для аналитических отчетов и других выборок.
   3. Данные для машинного обучения.
3. По типу изменяемости:
   1. Неизменяемые.
   2. Периодически изменяемые. В этих таблицах ведется специальная колонка «ActiveFrom».
   3. Последнее актуальное значение.
4. По версионности:
   1. Версионность ведется.
   2. Версионность не ведется.

# DB META

Здесь хранится информация по внутренним операциям ETL/DWH.

## Таблица LastExportedRecords

Таблица используется для определения дельты переноса для таблиц измерений.

Структура:

1. Id – int serial.
2. TableName – text – название таблицы для обработки.
3. LastExportedTime – timestamp – дата+время последней выгруженной записи ии OLTP.
4. PeriodToStore – int – количество дней хранения записей.

## Таблица EtlJournal

Таблица предназначена для регистрации операций выгрузки данных в DWH.

Структура:

1. Id – int – serial.
2. Name – text – название операции ETL.
3. BeginTime – timestamp – время начала операции.
4. EndTime – timestamp – время завершения операции.
5. ResultCode – int – код ошибки завершения. 0 – успешно.

# DB IoT

## Таблица IOT.Controllers

### Общее описание

Справочник контроллеров датчиков.

Таблица измерений. Хранится последнее актуальное значение. Версионность не ведется.

Структура:

1. Id – UUID
2. Description – text / string
3. TypeDescription – text / string

### ETL

Периодичность выгрузки: 1D.

Время выгрузки: 1AM.

Правила:

* 1. Источник выгрузки:
     1. iot.Controllers
     2. iot.ControllerTypes.
  2. Выгружаются все записи, которые были обновлены после последнего времени выгрузки по таблице meta.LastExportedRecords.
  3. Выгрузка через join по связи Controllers.ControllerTypeId = ControllerTypes.Id.

1. Mapping:
   1. Id = Controllers.Id.
   2. Description = Controllers.Description.
   3. TypeDescription = ControllerTypes.Description.

## Таблица IOT.IODevices

Справочник датчиков.

Таблица измерений. Хранятся последние значения. Версионность не ведется.

Структура:

1. Id – UUID – ключ записи.
2. Description – text / string.
3. ControllerId – ссылка на таблицу IOT.Controllers.
4. TypeDescription – text / string.
5. Model – text / string.
6. SerialNumber – text / string.

### ETL

Периодичность выгрузки: 1D.

Время выгрузки: 1AM.

Правила:

* 1. Источник выгрузки (транзакционная система):
     1. iot.Controllers
     2. iot.IODevices
  2. Выгружаются все записи, которые были обновлены после последнего времени выгрузки по таблице meta.LastExportedRecords. Колонка времени обновления:
  3. Выгрузка через join по связи Controllers.Id = IODevices.ControllerId.

1. Mapping:
   1. Id – UUID – IODevices.Id
   2. Description – IODevices.Description
   3. ControllerId – Controllers.Id
   4. TypeDescription – IODevices.Type
   5. Model – IODevices.Model
   6. SerialNumber – IODevices.SerialNumber

## Таблица IODeviceReadings

Архив показаний датчиков.

Таблица фактов. Хранятся последние значения показаний датчиков. Версионность не ведется.

Структура:

1. ControllerId – UUID – ссылка на IOT.Controllers.Id
2. IODeviceId – UUID – ссылка на IOT.IODevices.Id
3. ReadingValue – double – показатель датчика
4. ExtendedValue – jsonb – расширенное значение показателя датчика
5. ValueTimestamp – timestamp – время замера показателя.

### ETL

#### Таблица iot.MqttLogs

Периодичность выгрузки: 1D.

Время выгрузки: 1AM.

Правила:

* 1. Источник выгрузки:
     1. iot.MqttLogs
     2. iot.Controllers
  2. Выгружаются все записи, которые были обновлены после последнего времени выгрузки по таблице meta.LastExportedRecords. Значение для обновления — ValueTimestamp и Timestamp.
  3. Выгрузка через join по связи Controllers.MacAddress = MqttLogs.UserDefined

1. Mapping:
   1. ControllerId = IOT.Controllers.Id
   2. IODeviceId = null
   3. ReadingValue = MqttLogs.PeakRssi
   4. ExtendedValue = jsonb object:
      1. TagId = MqttLogs.IdHex
      2. Rssi = MqttLogs.PeakRssi
      3. ReadNumber = MqttLogs.Reads
   5. ValueTimestamp – MqttLogs.Timestamp