Описание процедуры ETL

Предварительно была создана база данных со следующей структурой:

|  |  |  |
| --- | --- | --- |
| № | Наименование таблицы | Краткое описание |
| 1 | fact\_flights | Таблица фактов, содержащая данные по совершенным перелетам |
| 2 | dim\_calendar | Таблица измерений по датам, наполняется данными на этапе создания таблицы, то есть еще до запуска ETL-процесса |
| 3 | dim\_passengers | Таблица измерений по пассажирам |
| 4 | dim\_aircrafts | Таблица измерений по самолетам |
| 5 | dim\_airports | Таблица измерений по аэропортам |
| 6 | dim\_tariffs | Таблица измерений по классам обслуживания |
| 7 | tickets\_passengers | Вспомогательная таблица, содержащая данные по пассажирам в связке с данными по билетам, используется для загрузки данных в таблицу фактов fact\_flights |
| 8 | tickets\_tariffs | Вспомогательная таблица, содержащая данные по классам обслуживания в связке с данными по билетам, используется для загрузки данных в таблицу фактов fact\_flights |
| 9 | fact\_flights\_rejected | Таблица для записей, предназначенных для таблицы fact\_flights, но не прошедших процедуру контроля качества данных |
| 10 | dim\_passengers\_rejected | Таблица для записей, предназначенных для таблицы dim\_passengers, но не прошедших процедуру контроля качества данных |
| 11 | dim\_aircrafts\_rejected | Таблица для записей, предназначенных для таблицы dim\_aircrafts, но не прошедших процедуру контроля качества данных |
| 12 | dim\_airports\_rejected | Таблица для записей, предназначенных для таблицы dim\_airports, но не прошедших процедуру контроля качества данных |
| 13 | dim\_tariffs\_rejected | Таблица для записей, предназначенных для таблицы dim\_tariffs, но не прошедших процедуру контроля качества данных |
| 14 | tickets\_passengers\_rejected | Таблица для записей, предназначенных для таблицы tickets\_passengers, но не прошедших процедуру контроля качества данных |
| 15 | tickets\_tariffs\_rejected | Таблица для записей, предназначенных для таблицы tickets\_tariffs, но не прошедших процедуру контроля качества данных |

SQL-скрипты, формирующие указанные таблицы, представлены в отдельных файлах.

Созданная процедура ETL с точки зрения структуры состоит из нескольких трансформаций, объединенных в одну задачу (job). Такая структура нужна для того, чтобы последовательно выполнялись следующие действия: отключение ограничений в базе данных (внешних ключей), загрузка данных в таблицы измерений, восстановление внешних ключей, загрузка данных в таблицу фактов. Таким образом, в ETL-процесс входят следующие трансформации:

|  |  |  |  |
| --- | --- | --- | --- |
| № | Наименование шага в задаче | Наименование файла трансформации | Краткое описание |
| 1 | Passengers data load | dim\_passengers\_transformation | Загрузка данных в таблицу dim\_passengers (dim\_passengers\_rejected в случае непрохождения проверок) |
| 2 | Aircrafts data load | dim\_aircrafts\_transformation | Загрузка данных в таблицу dim\_aircrafts (dim\_aircrafts\_rejected в случае непрохождения проверок) |
| 3 | Airports data load | dim\_airports\_transformation | Загрузка данных в таблицу dim\_airports (dim\_airports\_rejected в случае непрохождения проверок) |
| 4 | Tariffs data load | dim\_tariffs\_transformation | Загрузка данных в таблицу dim\_tariffs (dim\_tariffs\_rejected в случае непрохождения проверок) |
| 5 | Additional table 1 | tickets\_passengers\_transformation | Загрузка данных в таблицу tickets\_passengers (tickets\_passengers\_rejected в случае непрохождения проверок) |
| 6 | Additional table 2 | tickets\_tariffs\_transformation | Загрузка данных в таблицу tickets\_tariffs (tickets\_tariffs\_rejected в случае непрохождения проверок) |
| 7 | Fact flights load | fact\_flights\_transformation | Загрузка данных в таблицу fact\_flights (fact\_flights\_rejected в случае непрохождения проверок) |

Ниже приводится более подробное описание каждого шага.

1. **Passengers data load**

В качестве источника данных для наполнения таблицы dim\_passengers используется таблица tickets базы данных demo (схема bookings). Данные при загрузке проверяются по следующим трем правилам:

Если

1. и телефон, и электронная почта пассажира отсутствуют

или

1. номер телефона состоит не из 12 символов (включая символ «+»)

или

1. имеется адрес электронной почты, но он не соответствует шаблону %@%

то данные не проходят проверку и записываются в таблицу dim\_passengers\_rejected.

1. **Aircrafts data load**

В качестве источника данных для наполнения таблицы dim\_aircrafts используется представление aircrafts базы данных demo (схема bookings). Данные при загрузке проверяются по следующим трем правилам:

Если

1. максимальная дальность полета в километрах меньше 1 000 (для примера пусть существует такая техническая характеристика: предположим, что все современные самолеты имеют максимальную дальность полета не меньше 1 000, соответственно значения, меньшие 1 000, могли попасть в таблицу по ошибке)

или

1. максимальная дальность полета в километрах больше 20 000 (аналогично для примера пусть существует такая техническая характеристика: предположим, что все современные самолеты имеют максимальную дальность полета не больше 20 000, соответственно значения, большие 20 000, могли попасть в таблицу по ошибке)

или

1. код самолета содержит пробелы (возможное неаккуратное заполнение)

то данные не проходят проверку и записываются в таблицу dim\_aircrafts\_rejected.

1. **Airports data load**

В качестве источника данных для наполнения таблицы dim\_airports используется представление airports базы данных demo (схема bookings). Данные при загрузке проверяются по следующему правилу: если часовой пояс не соответствует шаблону %/% (например, в случае если использовался другой вариант записи, что-то типа «UTC+9:30»), то данные не проходят проверку и записываются в таблицу dim\_airports\_rejected.

1. **Tariffs data load**

В качестве источника данных для наполнения таблицы dim\_tariffs используется таблица seats базы данных demo (схема bookings). Данные проверяются по простому условию: обозначение класса обслуживания должно соответствовать «Business», «Comfort» или «Economy», иначе данные не проходят проверку и записываются в таблицу dim\_tariffs\_rejected.

Для данной таблицы суррогатный ключ создается внутри трансформации при помощи шага «Add sequence». Так, при обновлении таблиц, то есть при повторном запуске трансформации, значения суррогатного ключа останутся неизменными в отличие от того, что получилось бы, например, при формировании суррогатного ключа в таблице, указав тип serial: с каждым новым запуском трансформации значения будут увеличиваться.

1. **Additional table 1**

На данном шаге записываются данные во вспомогательную таблицу tickets\_passengers. В качестве источника данных используется таблица tickets базы данных demo (схема bookings). Данные проверяются по тем же правилам, которые использовались для проверки данных для загрузки в таблицу dim\_passengers.

1. **Additional table 2**

На данном шаге записываются данные во вспомогательную таблицу tickets\_tariffs. В качестве источника данных используется таблица ticket\_flights базы данных demo (схема bookings). Данные проверяются по тем же правилам, которые использовались для проверки данных для загрузки в таблицу dim\_tariffs.

Для поиска ключа класса обслуживания используется дополнительный шаг «Combination lookup/update», который ищет значение ключа на основе ранее созданной таблицы dim\_tariffs.

1. **Fact flights load**

Трансформация предназначена для загрузки данных в таблицу фактов fact\_flights.

Источниками данных являются следующие таблицы:

* таблица flights базы данных demo (схема bookings),
* dim\_calendar, созданная на более раннем этапе данного ETL-процесса,
* tickets\_tariffs, созданная на более раннем этапе данного ETL-процесса,
* tickets\_passengers, созданная на более раннем этапе данного ETL-процесса.

Структура данной трансформации представлена соединениями (join’ами) указанных таблиц. Такая структура позволяет легко собирать данные из таблиц, расположенных в разных базах данных, а также осуществлять более гибкий и детальный контроль качества данных: проверки могут быть добавлены на каждом этапе.

Примером такой проверки может быть следующий случай. Можно собирать в отдельные файлы или таблицы строки, которые «выпали» из потока на этапе того или иного join’а, так как им не нашлось пары – потенциальная проверка целостности, чтобы не было ситуаций, когда есть выполненный рейс, но нет данных по вариантам размещения на борту для данного рейса.

В рамках данной трансформации выполнены следующие проверки для обеспечения качества данных:

Если

1. фактические дата и время отправления находятся на временной оси после фактических даты и времени прибытия

или

1. фактические дата и время отправления находятся на временной оси после текущих даты и времени

или

1. фактические дата и время прибытия находятся на временной оси после текущих даты и времени

то данные не проходят проверку и записываются в таблицу fact\_flights\_rejected.