УДК 004.658.2

ОГЛЯД ОСОБЛИВОСТЕЙ ТЕСТУВАННЯ ETL ПРОЦЕСУ БАЗ ДАНИХ

Лісова Д.М., студентка 5 курсу, група ІКМ-221в,
Матюшенко М.В., д.т.н., доцент,
Національний технічний університет
«Харківський політехнічний інститут»
(м. Харків, Україна)

Анотація *– у статті проаналізовано основні етапи створення архітектури бази даних та її тестування на етапі налаштування ETL. Актуальність полягає в необхідності поліпшення якості розробляємого продукту, а саме реляційної бази, адже дефекти в даних та некоректні калькуляції досить важко виявити на великому об'ємі інформації який обробляється та завантажується. В роботі враховані підходи до тестування не лише статичних даних, але і ETL, тобто процесу завантаження та обробки датасетів.*

Ключові слова *– реляційна база даних, тестування, СУБД, тестування, ETL, датасет.*

Постановка проблеми. ETL – абревіатура від Extract, Transform, Load, саме цей процес відповідає за наповнення бази інформацією, тобто за завантаження датасетів. Це не тільки процес перенесення даних з одного сховища до іншого, але й інструмент підготовки їх до аналізу.

Привести всі дані до єдиної системи значень та деталізації, забезпечивши їх якість та надійність є першою задачею ETL процесу.

Забезпечити аудиторський слід при перетворенні (Transform) даних, щоб після перетворення можна було зрозуміти, з яких вихідних даних і сум зібрався кожен рядок перетворених даних, можна віднести до другої мети [1].

Тестування при імлементації системи що вирішує ці задачі є вкрай необхідним, особливо якщо ви пишете ETL-процес вручну, або робите його з використанням фреймворків низької готовності, в яких не задана готова структура проміжних таблиць.

Легко пропустити друге завдання і мати багато проблем з пошуком причин помилок у трансформованих даних.

Тож ETL потребує особливого підходу до тестування, адже не потрапляє під стандартні техніки тест-дизайну. Саме основні опорні точки тест-плану для цього процесу будуть описані в даній статті.

Аналіз останніх досліджень. Джефф Теобальд виділяє, що при проведенні тестування сховища варто зосередити увагу на таких аспектах [2]:

- Повнота даних;
- Перетворення даних;
- Якість даних;
- Продуктивність та масштабованість;
- Тестування інтеграції;
- Оцінка ступеня прийняття програми користувачами (User-Acceptance Testing (UAT));

Однією з головних проблем є не лише результат процесу тестування, але і особливості його проведення. До важливого моменту відносять необхідність у синхронізації продуктового та тестового контурів [3].

Формулювання цілей статті. Мета публікації полягає у виділенні опорних підходів до створення тестової стратегії для ETL процесу.

Основна частина. Аналіз базується на прикладі реляційної бази даних. Для маніпуляцій з інформацією використано засоби SQL.

При тестуванні процесу завантаження даних необхідно пам'ятати такі не очевидні моменти, що:

- Потрібно враховувати вимоги бізнесу щодо тривалості всього процесу ітерацій ETL для більш точного відслідковування періодичності проведення тестів.

- Дані можуть завантажуватися хвилею, що набігає, - з послідовним оновленням даних одного і того ж періоду в майбутньому протягом декількох послідовних періодів. (наприклад: оновлення прогнозу закінчення року щомісяця). Тому крім довідника «Період» є необхідний технічний довідник «Період завантаження», який дозволить ізолювати процеси завантаження даних у різних періодах і не втратити історію зміни цифр, що є вдалим рішенням для перевірки.

- Необхідне імітування негативних тест-кейсів. Наприклад: зімітувати ситуацію, що файл не виклали на початок ETL; delimiter використовується у значенні ключового поля; використання некоректного лайауту.

Тож варто виділити прийоми тестування на практиці.

Одним з основних тестів повноти даних є перевірка того, щоб всі дані завантажені в сховище. Це включає в себе порівняння кількості записів між вихідними даними, даними, завантаженими у сховище, та відхиленими.

Перевірка того, чи коректно перетворюються дані відповідно до бізнес-правил, може бути складною частиною тестування ETL-додатків. Один з типових методів - це вибрати кілька записів і " дивитися їх і порівнювати " , тобто вручну перевіряти правильність перетворення [4]. Це може бути корисним, але вимагає наявності тестувальників, які розуміють логіку ETL-процесу. Поєднання автоматизованого профілювання даних та

автоматизованого перенесення даних є більш довгостроковою стратегією. Виділено кілька простих методів автоматизованого перенесення даних:

- Створення електронних таблиць з варіантами вхідних даних та очікуваних результатів, які потім показуються клієнтам для перевірки. Цей відмінний підхід до виявлення вимог при проектуванні, він також може бути використаний під час тестування.
- Створення тестових даних, яке включає всі сценарії.
- Використання результатів профілювання даних для того, щоб порівняти діапазон і розподіл значень у кожному полі між вихідними та цільовими даними.
- Перевірка відповідності типів даних у сховищі його архітектури та/або моделі даних.

В даному випадку доцільно використовувати SQL конструкцію Field-to-field, яка може виглядати таким чином:
sum(iff (ifnull (t1.value_from_db_table, '1900-12-31') <> ifnull (t2. value_from_qa_table, '1900-12-31') ,1,0)) as "test"

Тобто відбувається порівняння даних з тестової таблиці та вже завантажених даних. Така конструкція дозволяє вирахувати кількість записів, які не співпадають.

Для успіху в тестуванні якості даних слід включати максимально можливу кількість сценаріїв даних [5]. Зазвичай правила для забезпечення якості даних визначаються на стадії проектування, наприклад:

- Відхилити запис, якщо певне десяткове поле має нечислові дані.
- Підставити нульове значення, якщо певне десяткове поле має нечислове значення.
- Перевірити та за необхідності виправити поле «Штат» на основі поштового індексу.
- Порівняти код продукту з даними у довідковій таблиці, і якщо збігу немає, у будь-якому випадку зробити завантаження, але при цьому повідомити користувачів.

Оскільки обсяг даних у сховищі зростає, очікується збільшення часу завантаження та виконання запитів. Цей процес може бути прискорений за наявності надійної технічної архітектури та моделі ETL-процесу. Метою тестування є виявлення будь-яких потенційних недоліків в архітектурі ETL, таких як читання файлу кілька разів або створення непотрібних тимчасових файлів [6]. Наступні стратегії допоможуть виявити проблеми з продуктивністю:

- Завантаження бази даних з максимально очікуваним обсягом даних у режимі промислової експлуатації, щоб цей обсяг міг бути завантажений у рамках узгодженого вікна.
- Порівняння часу завантаження цих ETL-процесів із завантаженням при менших обсягах даних, щоб передбачити проблеми масштабованості.

- Виявлення часу відхилення процесу визначення обсягів відхилених даних.

Отже, на базі поданих вище тест-кейсів розглянуто варіанти підходу до тестування ETL.

Висновки. У даній статті наведено варіанти тестування ключових вимог до ETL. Застосування цих рекомендацій у процесі розробки та тестування сховищ даних гарантує якість продукту на виході та допоможе запобігти дорогим помилкам, які можуть виникнути в режимі промислової експлуатації системи.

Перспективою подальшого дослідження є автоматизація наведених підходів шляхом проектування автотестів на основі засобів pytest та Jenkins. Ця система призначена для забезпечення процесу безперервної інтеграції програмного забезпечення.

Крім цього, звичайно в реальних системах є ще сервісні процеси — авторизації, розмежування доступу до даних, автоматизоване узгодження змін, які також мають свої особливості тестування.

Також важливо буде приділити увагу тестуванню системної інтеграції, яке включає тестування компонентів системи окремо і подальшу інтеграцію модулів.

Бібліографічний список

1. Codd E. F. Is Your DBMS Really Relational? USA, ComputerWorld. 152 с.

2. Jeff Theobald Strategies for Testing Data Warehouse Applications. URL: https://www.information-management.com/news/strategies-for-testing-data-warehouse-applications

3. Семенова І.І. SQL СТАНДАРТ В СУБД MS SQL SERVER, ORACLE, VFP И ACCESS: МАНИПУЛЮВАННЯ ДАНИМИ. Омськ: СибАДИ, 2008. 57 с.

4. Ralph Kimball, Joe Caserta The Data Warehouse ETL Toolkit:Practical Techniques for Extracting, Cleanin. USA: Wiley, 2004. 528 с.

5. Рекс Блек Основні процеси тестування. Планування, підготовка, проведення, вдосконалення. Москва: Лори, 2011. 544 с.

6. Erik Thomsen OLAP Solutions: Building Multidimensional Information Systems. USA: Wiley, 2004. 608 с.