

ДУДНИК О.В.

ЦИРКУН В.В.

ШАЛАГАН В.С.

ETL PIPELINE НА PYTHON ДЛЯ АНАЛИЗУ ДАНИХ АГРЕГАТОРІВ У СФЕРІ АВІАПЕРЕВЕЗЕНЬ

Вінницький національний технічний університет

Анотація

У сучасній авіаційній галузі, що стрімко розвивається, ефективне управління та аналіз даних є ключовими факторами успішної діяльності компаній–агрегаторів авіап перевезень. Ця стаття зосереджується на створенні та впровадженні ETL–пайплайну (Extract, Transform, Load) з використанням мови програмування Python для збору, обробки та інтеграції даних про авіарейси. Розглянуто методологію побудови ETL–процесів, включаючи витягування даних з SFTP–серверів, фільтрація, нарізання записів та знаходження помилок за допомогою бібліотеки Pandas, і подальше завантаження в базу даних для аналітики. Особлива увага приділена використанню Apache Airflow як інструменту для автоматизації робочих процесів, що забезпечує гнучке керування завданнями та масштабованість системи. У статті аналізуються переваги впровадження таких рішень для підвищення точності та актуальності даних, що дозволяє агрегаторам надавати клієнтам найкращі пропозиції в режимі реального часу. Описані практичні аспекти впровадження ETL–пайплайнів та їх вплив на підвищення ефективності бізнес–процесів у сфері авіап перевезень.

Ключові слова: ETL, Python, авіап перевезення, аналіз даних, Apache Airflow.

Abstract

In the fast-growing aviation industry, effective data management and analysis are key factors for the success of airline aggregator companies. This article focuses on creating and using an ETL pipeline (Extract, Transform, Load) with the Python programming language to collect, process, and integrate flight data. The article explains the method of building ETL processes, including extracting data from SFTP servers, filtering, splitting records, and finding errors using the Pandas library. After that, the data is loaded into a database for analysis. Special attention is given to Apache Airflow, a tool for automating workflows, which helps manage tasks flexibly and scale the system when needed. The article also looks at the advantages of using such solutions to improve the accuracy and freshness of data. This allows aggregators to offer the best options to clients in real time. The practical steps of setting up ETL pipelines and how they help make business processes more efficient in the aviation industry are described.

Keywords: ETL, Python, air transportation, data analysis, Apache Airflow.

Вступ

У висококонкурентному світі подорожей доступ до даних про рейси в реальному часі має вирішальне значення для агрегаторів рейсів, які прагнуть пропонувати своїм клієнтам найкращі пропозиції. Щодня злітають і приземляються мільйони рейсів, щоб не відставати від цін, наявності та

розкладів, що постійно змінюються, потрібні складні інструменти та стратегії. Тут у гру вступає аналіз даних польоту за допомогою передової технології вебзбирання.

Індустрія подорожей працює шаленими темпами. Ціни можуть коливатися протягом декількох хвилин залежно від попиту, наявності та інших факторів. Для агрегаторів рейсів надання актуальної інформації – це не просто зручність – це необхідність. Клієнти очікують побачити точні варіанти рейсів, і будь-які розбіжності можуть призвести до втрати продажів і завдати шкоди репутації бренду.

Аналіз даних про польоти передбачає збір, обробку та інтерпретацію даних із різних джерел, щоб отримати інформацію про розклад польотів, тенденції ціноутворення, наявність місць тощо. Аналізуючи ці дані в режимі реального часу, агрегатори можуть пропонувати своїм користувачам найбільш точні та конкурентоспроможні варіанти польотів.

Моніторинг даних за допомогою вебзбирання:

- збір даних: Інструменти вебзбирання автоматично витягують дані польоту з вебсайтів авіакомпаній, онлайн-туристичних агентств та інших джерел. Це містить інформацію про розклад рейсів, ціни, наявність місць і спеціальні пропозиції;
- агрегація даних: Зібрані дані збираються в режимі реального часу для створення повної бази даних доступних рейсів. Це дозволяє агрегаторам рейсів порівнювати різні варіанти та визначати найкращі пропозиції;
- аналіз даних: Після збору дані аналізуються за допомогою складних алгоритмів і інструментів аналізатора польоту. Цей процес передбачає визначення тенденцій, прогнозування змін цін і оптимізацію часу для рекомендацій щодо бронювання;
- оновлення в режимі реального часу: Найважливішою перевагою вебзбирання є його здатність надавати оновлення в режимі реального часу. Щойно ціна змінюється або рейс стає доступним, платформа агрегатора відображає цю інформацію, забезпечуючи клієнтам завжди доступ до найновіших даних [1].

Що таке ETL, процес побудови, переваги та недоліки

ETL означає Extract, Transform, Load, і це процес, який використовується в сховищах даних для вилучення даних із різних джерел, перетворення їх у формат, придатний для завантаження в сховище даних, а потім завантаження в сховище.

Процес ETL можна розбити на такі три етапи:

- витягнення : Перший етап у процесі ETL полягає у вилученні даних із різних джерел, таких як транзакційні системи, електронні таблиці та плоскі файли. Цей крок включає зчитування даних із вихідних систем і збереження їх у проміжній області;
- перетворення : На цьому етапі витягнуті дані перетворюються у формат, придатний для завантаження в сховище даних. Це може передбачати очищення та перевірку даних, перетворення типів даних, об'єднання даних із кількох джерел і створення нових полів даних;
- завантаження : Після перетворення даних вони завантажуються в сховище даних. Цей крок передбачає створення фізичних структур даних і завантаження даних у сховище.

Процес ETL — це ітераційний процес, який повторюється, коли нові дані додаються до сховища. Цей процес важливий, оскільки він гарантує, що дані в сховищі даних є точними, повними та актуальними. Це також допомагає переконатися, що дані мають формат, необхідний для аналізу даних і звітності (рис.1).

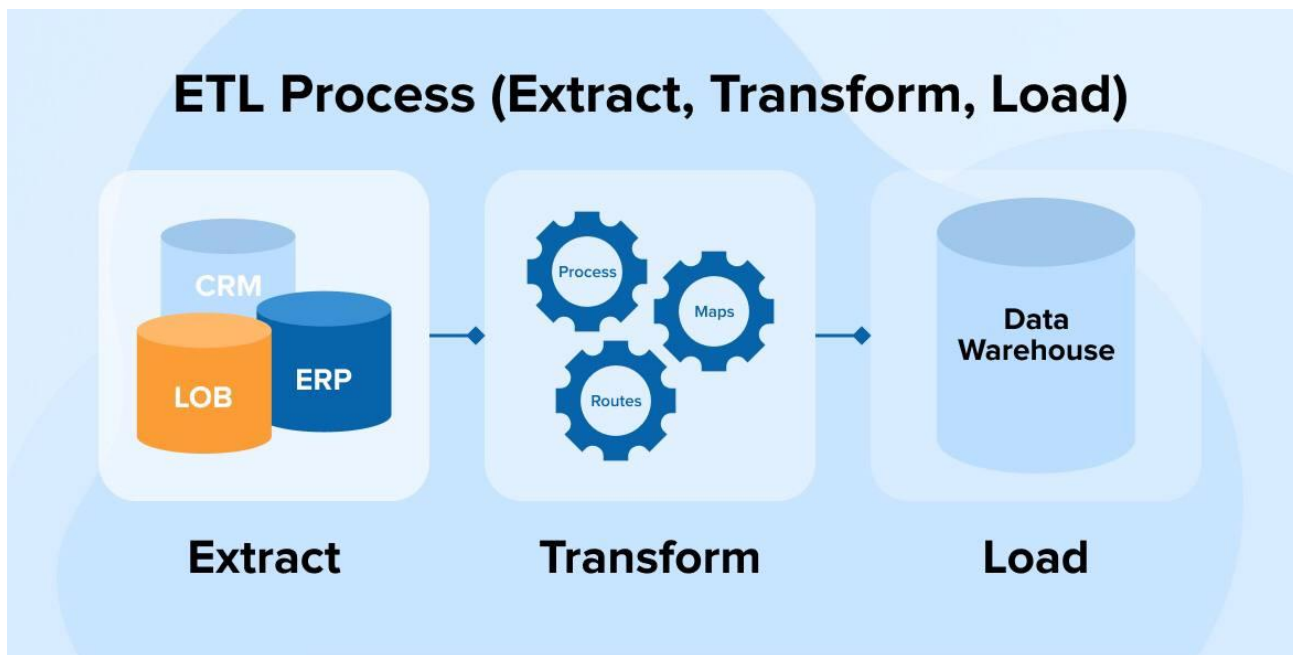


Рис.1 – ETL-процес

Етапи побудови ETL Pipeline

Витягування даних з SFTP-сервера. Агрегатори авіаквитків часто надають великі обсяги інформації (розклади, ціни, маршрути) через захищений протокол передачі файлів – SFTP.

Після витягування даних із SFTP-сервера наступним завданням є підготовка їх до подальшого аналізу. У реальних умовах обробляються великі обсяги даних – десятки або навіть сотні тисяч записів, що містять інформацію про рейси, перевізників, ціни, розклади тощо.

У таких випадках важливо не лише обробити дані, але й забезпечити їх якість (Data Quality). Наявність помилок у даних, таких як пропуски або некоректні формати, може призвести до неправильних висновків і, як наслідок, до помилкових бізнес-рішень.

Python з бібліотекою Pandas дозволяє ефективно працювати з великими наборами даних, здійснюючи нарізку і фільтрацію за необхідними полями та пошуком помилок.

Після завершення етапу трансформації, включаючи перевірку на помилки і нарізку даних, підготовлені файли готові до завантаження у сховище даних. На третьому етапі оброблені дані будуть інтегровані в базу даних (наприклад, PostgreSQL), що дозволяє використовувати їх для побудови звітів, аналітичних моделей і візуалізації.

Завдяки ретельній перевірці на етапі трансформації, забезпечується висока якість завантажених даних, що є основою для прийняття обґрунтованих бізнес-рішень.

Переваги процесу ETL у сховищах даних:

- покращена якість даних: Процес ETL гарантує, що дані в сховищі даних є точними, повними та актуальними;
- краща інтеграція даних: Процес ETL допомагає інтегрувати дані з багатьох джерел і систем, роблячи їх більш доступними та зручними для використання;
- підвищена безпека даних: Процес ETL може допомогти покращити безпеку даних, контролюючи доступ до сховища даних і забезпечуючи доступ до даних лише авторизованим користувачам;

- покращена масштабованість: Процес ETL може допомогти покращити масштабованість, надаючи спосіб керування та аналізу великих обсягів даних;
- підвищена автоматизація: Інструменти та технології ETL можуть автоматизувати та спростити процес ETL, зменшуючи час і зусилля, необхідні для завантаження та оновлення даних у сховищі.

Недоліки процесу ETL у сховищах даних:

- висока вартість: Процес ETL може бути дорогим для впровадження та підтримки, особливо для організацій з обмеженими ресурсами;
- складність : Процес ETL може бути складним у реалізації, особливо для організацій, яким бракує необхідного досвіду чи ресурсів;
- обмежена гнучкість: Процес ETL може бути обмежений з точки зору гнучкості, оскільки він може не мати змоги обробляти неструктуровані дані або потоки даних у реальному часі;
- обмежена масштабованість : Процес ETL може бути обмежений з точки зору масштабованості, оскільки він не зможе обробляти дуже великі обсяги даних;
- занепокоєння конфіденційністю даних : Процес ETL може викликати занепокоєння щодо конфіденційності даних, оскільки збираються, зберігаються та аналізуються великі обсяги даних [2].

Ефективне управління ETL–процесами за допомогою Apache Airflow

Для ефективного управління ETL–процесами, особливо коли мова йде про великі обсяги даних і регулярне оновлення інформації, важливо використовувати інструменти для автоматизації робочих процесів. Одним із таких потужних інструментів є Apache Airflow (рис.2).

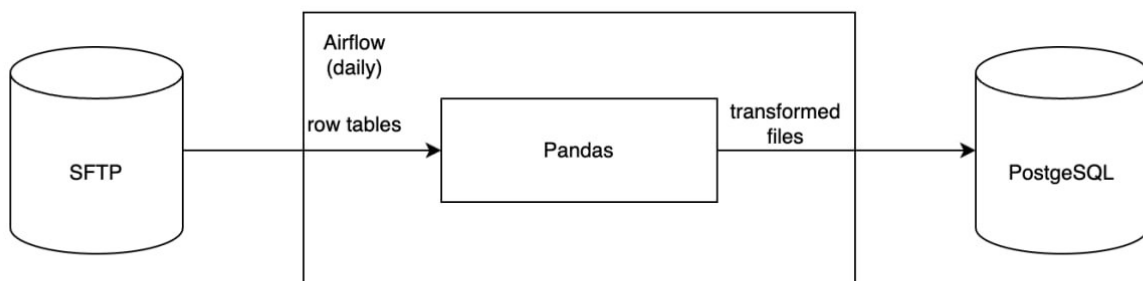


Рис.2 – Управління ETL–процесами з використанням Apache Airflow

Airflow — це платформа, яка дозволяє створювати та запускати робочі процеси. Робочий процес представлено як DAG (спрямований ациклічний графік) і містить окремі частини роботи, які називаються завданнями, упорядкованими з урахуванням залежностей і потоків даних.

DAG визначає залежності між завданнями, що визначає порядок виконання завдань. Завдання описують, що робити, будь то отримання даних, запуск аналізу, запуск інших систем тощо.

Airflow сам по собі не залежить від того, що ви використовуєте – він із задоволенням організує та запустить будь-що, або за підтримки високого рівня від одного з постачальників, або безпосередньо як команда за допомогою оболонки чи операторів Python.

Мінімальна установка Airflow складається з таких компонентів:

- планувальник, який керує як ініціюванням запланованих робочих процесів, так і надсиланням завдань виконавцю для виконання. Виконавець є властивістю конфігурації планувальника, а не окремим компонентом і виконується в процесі планувальника. Є кілька виконавців, доступних із коробки, і також можна написати свій власний;
- вебсервер, який представляє зручний інтерфейс користувача для перевірки, запуску та налагодження поведінки DAG і завдань;
- папка з файлами DAG, які зчитуються планувальником, щоб визначити, які завдання запускати та коли їх запускати;
- база даних метаданих, яку компоненти airflow використовують для зберігання стану робочих процесів і завдань [3].

Висновок: Впровадження ETL—пайплайнів на базі Python значно підвищує ефективність обробки великих обсягів даних в авіаційній галузі. Застосування бібліотеки Pandas дозволяє швидко та якісно трансформувати дані, тоді як Apache Airflow забезпечує автоматизацію процесів, підвищуючи їхню надійність і масштабованість. Завдяки таким рішенням агрегатори можуть надавати актуальні дані своїм клієнтам у режимі реального часу, що сприяє підвищенню конкурентоспроможності на ринку авіаперевезень.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Flight Data Analysis Through Web Scraping. URL: https://www-promptcloud-com.translate.goog/blog/flight-data-analysis-through-web-scraping/? x_tr_sl=en& x_tr_tl=uk& x_tr_hl=uk& x_tr_pto=sc (дата звернення: 07.02.2025).
2. ETL Process in Data Warehouse. URL: https://www-geeksforgeeks-org.translate.goog/etl-process-in-data-warehouse/? x_tr_sl=en& x_tr_tl=uk& x_tr_hl=uk& x_tr_pto=sc (дата звернення: 07.02.2025).
3. Apache Airflow Documentation. URL: https://airflow-apache-org.translate.goog/docs/apache-airflow/stable/core-concepts/overview.html? x_tr_sl=en& x_tr_tl=uk& x_tr_hl=uk& x_tr_pto=sc (дата звернення: 07.02.2025).

ДУДНИК Олександр Вікторович – к.т.н., доцент кафедри Обчислювальної техніки, Вінницький національний технічний університет, м. Вінниця;

ЦИРКУН Владислав Вікторович – аспірант кафедри Обчислювальної техніки, Вінницький національний технічний університет, м. Вінниця;

ШАЛАГАН Вікторія Сергіївна – студентка групи 2KI-23мс факультету Інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця.

DUDNYK Oleksandr Viktorovich – Ph.D., Associate Professor of the Department of Computer Engineering, Vinnytsia National Technical University, Vinnytsia;

TSYRKUN Vladyslav Viktorovich – Postgraduate Student of the Department of Computer Engineering, Vinnytsia National Technical University, Vinnytsia;

SHALAHAN Viktoriia Serhiivna – Student of Group 2KI-23ms, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia.