УДК 339.54.012

Козлов Вячеслав Васильевич, кандидат технических наук доцент кафедры «Информатика и вычислительная техника» Самарский государственный технический университет Молодогвардейская ул., 244, Самара
Мишин Максим Антонович, бакалавр 4 курса кафедры «Информатика и вычислительная техника» Самарский государственный технический университет Молодогвардейская ул., 244, Самара

ETL И ELT: ЧТО ВЫБРАТЬ ДЛЯ ВАШЕГО ХРАНИЛИЩА ДАННЫХ?

Аннотация: *В статье рассматриваются подходы к интеграции данных ETL (Extract, Transform, Load) и ELT (Extract, Load, Transform), используемые в современных хранилищах данных. Проведён сравнительный анализ данных методов, выявлены их основные преимущества и недостатки, а также особенности применения в различных условиях. Особое внимание уделено вопросам производительности, масштабируемости и экономической эффективности. На основе проведённого анализа сформулированы рекомендации по выбору оптимального подхода в зависимости от задач и инфраструктуры организации.*

Annotation: *This article examines two data integration approaches, ETL (Extract, Transform, Load) and ELT (Extract, Load, Transform), used in modern data warehouses. A comparative analysis of these methods is conducted, highlighting their key advantages, disadvantages, and application scenarios. Particular attention is paid to performance, scalability, and cost efficiency. Based on the analysis, practical recommendations are provided for*

selecting the optimal approach depending on the organization's tasks and infrastructure.

Ключевые слова: *интеграция данных, ETL, ELT, хранилища данных, обработка данных, большие данные, облачные технологии, трансформация данных, аналитика данных*

Keywords: *data integration, ETL, ELT, data warehouses, data processing, big data, cloud technologies, data transformation, data analytics*

Введение

Современные бизнес-процессы требуют эффективной работы с данными. Хранилища данных выступают ключевым инструментом для интеграции информации из множества источников, обеспечения её доступности и аналитики. Основные методы обработки данных в хранилищах

— это ETL и ELT.

Выбор между ETL и ELT влияет на скорость обработки данных, их качество и экономическую эффективность всей системы. В данной статье мы рассмотрим, что представляют собой оба подхода, их различия, преимущества и недостатки, а также предоставим рекомендации по выбору оптимального метода.

Хранилища данных играют важную роль в современном мире, где объемы информации постоянно растут. Компании ищут решения, которые позволяют быстро адаптироваться к изменениям, минимизировать затраты и повышать эффективность. Анализ методов ETL и ELT — ключ к оптимизации этих процессов.

1. Определение и принципы работы ETL и ELT

1.1. ETL (Extract, Transform, Load)

ETL (Extract, Transform, Load) – это классический подход к интеграции данных, при котором данные извлекаются из источников,

трансформируются в промежуточной среде, а затем загружаются в хранилище. Основные этапы:

1. **Extract (Извлечение):**
   - Данные извлекаются из различных источников: реляционных баз данных, API, файлов или потоковых источников.
   - На этом этапе важно обеспечить совместимость и синхронизацию данных.
2. **Transform (Трансформация):**
   - Преобразование данных включает очистку, фильтрацию, агрегирование и структуризацию.
   - Используются специализированные ETL-инструменты (например, Talend, Apache Nifi) или программные решения.
   - Этот этап требует мощных вычислительных ресурсов на стороне ETL-сервера.
3. **Load (Загрузка):**
   - Преобразованные данные загружаются в целевое хранилище в удобном формате для анализа.
   - Может быть реализована как одноразовая загрузка (batch) или потоковая загрузка (streaming).

Ключевые характеристики ETL:

- Контроль: Детальный контроль над процессами на каждом этапе.
- Гибкость: Возможность выполнять сложные трансформации до загрузки.
- Минусы: Высокие требования к инфраструктуре ETL-сервера и сложности масштабирования.

Формула процесса:

$$ETL = Extract(data) + Transform(logic) + Load(target)$$

1.2. ELT (Extract, Load, Transform)

ELT (Extract, Load, Transform) – это более современный подход, использующий вычислительные ресурсы хранилища данных для выполнения трансформаций. Основные этапы:

1. **Extract (Извлечение):**
   - Данные извлекаются так же, как в ETL, из разных источников.
2. **Load (Загрузка):**
   - Необработанные или минимально обработанные данные напрямую загружаются в хранилище (например, Snowflake, BigQuery).
   - Современные хранилища оптимизированы для быстрого чтения и записи больших объемов данных.
3. **Transform (Трансформация):**
   - Преобразование данных выполняется с использованием SQL-запросов или встроенных инструментов хранилища.
   - Это позволяет использовать ресурсы хранилища, что ускоряет обработку и снижает нагрузку на промежуточную инфраструктуру.

Ключевые характеристики ELT:

- Масштабируемость**:** Подходит для обработки больших данных.
- Эффективность**:** Использует мощности хранилищ, таких как облачные решения.
- Минусы: Меньший контроль над сложными трансформациями до загрузки.

Формула процесса:

$$ELT = Extract(data) + Load(target) + Transform(logic\_on\_target)$$

2. Сравнение ETL и ELT

Для удобства сравнения ниже представлена таблица с основными характеристиками:

Таблица 1. Основные характеристики ETL и ELT

Критерий	**ETL**	**ELT**
Этапы процесса	Трансформация до загрузки	Трансформация после загрузки
Скорость обработки	Зависит от ETL-инструмента	Быстрее за счёт хранилища
Объем данных	Подходит для меньших объёмов	Эффективен для больших данных
Использование ресурсов	Сервер ETL	Сервер хранилища
Гибкость	Ограниченная	Высокая
Затраты на внедрение	Выше (при сложной архитектуре)	Ниже (в облачных решениях)

Ключевые моменты:

- ETL чаще используется в проектах с небольшими объемами данных или там, где требуется сложная трансформация перед загрузкой [1].
- ELT актуален для работы с большими данными и облачными хранилищами, где мощность и гибкость инфраструктуры играют ключевую роль.
- Современные хранилища данных, такие как Snowflake, BigQuery или Amazon Redshift, оптимизированы для ELT, что делает его популярным в последние годы.

3. Когда выбирать ETL?

Подход ETL имеет свои сильные стороны, которые делают его предпочтительным выбором в определенных сценариях. Этот подход лучше всего подходит, когда данные требуют сложных преобразований до их

загрузки в хранилище, а также в случае, если объем данных относительно небольшой.

Примеры сценариев:

1. **Сложные трансформации:**

Если данные требуют глубокой очистки, агрегирования, перекодирования или преобразования форматов перед загрузкой, ETL позволяет выполнить это в контролируемой среде ETL-сервера.

Например, в случаях, когда данные из множества источников имеют разную структуру и формат, требуется унифицировать их до загрузки.

2. **Ограниченные возможности хранилища:**

Некоторые хранилища данных, особенно локальные, могут быть не оптимизированы для выполнения сложных вычислений или работы с сырыми данными. В таких случаях предварительная обработка данных перед загрузкой помогает снизить нагрузку на хранилище [2].

3. **Небольшие объемы данных:**

ETL подходит для обработки данных в компаниях с небольшим масштабом операций или тех, кто работает с данными, объем которых легко обрабатывается ETL-сервером.

Формула эффективности ETL:

$$Efficiency_{ETL} = \frac{Transform\_Time}{Load\_Time + Storage\_Cost}$$

Где:

- Transform_Time – время, затрачиваемое на обработку данных.
- Load_Time – время, необходимое для загрузки данных.
- Storage_Cost – затраты на использование хранилища.

4. Когда выбирать ELT?

Подход **ELT** ориентирован на современные потребности, такие как работа с большими объемами данных, использование облачных

технологий и необходимость быстрой загрузки данных для последующего анализа.

Примеры сценариев:

1. **Большие объемы данных:** ELT оптимально подходит для работы с "большими данными", где использование мощностей хранилища (особенно облачного) позволяет быстро обрабатывать огромные массивы информации.

Например, компании, собирающие данные из миллионов точек (сенсоров IoT, веб-аналитики), могут эффективно загружать их в облачное хранилище, такое как Snowflake или BigQuery [3].

2. **Облачные хранилища:** Современные облачные платформы обладают высокой вычислительной мощностью, что позволяет быстро выполнять трансформации данных внутри хранилища. Это снижает потребность в промежуточных серверах и делает процесс более гибким и масштабируемым.

3. **Высокая частота обновления данных:** ELT актуален для сценариев, где данные поступают в режиме реального времени или с высокой скоростью, а их анализ должен быть доступен практически сразу. Например, компании, занимающиеся обработкой потоковых данных (стриминг), могут извлекать и загружать данные, выполняя трансформации "на лету".

Формула эффективности ELT:

$$Efficiency_{ELT} = \frac{Compute\_Power}{Transform\_Complexity}$$

Где:

- Compute_Power – мощность хранилища, используемая для вычислений.
- Transform_Complexity – сложность операций по преобразованию данных.

5. Кейс-стади: Пример выбора подхода

Для лучшего понимания применения ETL и ELT рассмотрим бизнес-кей

с.

Описание кейса:

Компания занимается электронной коммерцией и собирает данные о транзакциях, поведении пользователей и маркетинговых кампаниях. Данные поступают из различных источников:

- Транзакции из базы данных SQL.
- Маркетинговая аналитика из сторонних API (Google Ads, Facebook Ads).
- Лог-файлы поведения пользователей на сайте.

Сравнение подходов:

1. ETL:

Если компания использует локальное хранилище, где важно предварительно очистить и унифицировать данные, то подход ETL будет предпочтительным [4]. Например:

- API может возвращать данные в разрозненных форматах, требующих сложной обработки.
- Данные о транзакциях могут содержать ошибки, требующие предварительной очистки.

2. ELT:

Если компания выбрала облачное хранилище, такое как Snowflake, она может загрузить все исходные данные сразу, а затем выполнить трансформации с помощью мощностей хранилища. Например:

- Лог-файлы можно загрузить без предварительной обработки, а затем анализировать их через SQL-запросы.
- Преобразование данных API и их объединение может быть выполнено уже в хранилище, где это быстрее и дешевле.

Итоговый выбор:

В данном случае, если компания ориентирована на масштабируемость и облачные решения, ELT будет оптимальным выбором. Однако, для традиционных решений с ограниченными ресурсами или сложными трансформациями до загрузки, предпочтителен подход ETL.

6. Рекомендации по выбору подхода

Выбор между ETL и ELT зависит от нескольких факторов, включая тип хранилища, объем данных, специфику задач и доступные ресурсы. Ниже приведены основные рекомендации, которые помогут принять оптимальное решение.

6.1. Привязка к типам хранилищ

- **Реляционные базы данных:**

  Если ваша инфраструктура основана на реляционных базах данных (например, PostgreSQL или Oracle), где вычислительные ресурсы ограничены, лучше использовать **ETL**. Такой подход минимизирует нагрузку на хранилище [5].

- **NoSQL-хранилища:**

  Для неструктурированных данных (например, MongoDB, Cassandra) лучше подойдет **ELT**, так как оно позволяет загружать данные в исходном виде и трансформировать их по мере необходимости.

- **Облачные хранилища:**

  Современные облачные платформы (Snowflake, BigQuery, Amazon Redshift) оптимизированы для **ELT**. Они предоставляют мощные вычислительные ресурсы, упрощая обработку данных непосредственно в хранилище.

6.2. Учет специфики задач

- **Аналитика и отчетность:**

Если основной задачей является создание отчетов, ETL может быть предпочтительным, так как он позволяет заранее подготовить данные в удобной форме [6].

- **Масштабируемость:**

  Для крупных организаций с большим объемом данных или ростом нагрузки рекомендуется ELT благодаря его гибкости и способности легко масштабироваться.

- **Частота обновления данных:**

  ELT подходит для сценариев с высокой частотой обновлений (например, потоковая обработка данных), в то время как ETL лучше для периодической обработки (batch processing).

6.3. Рекомендации по инструментам

Каждый подход имеет свои инструменты, которые можно выбрать в зависимости от потребностей:

- **Для ETL:**
- Talend – универсальный инструмент для управления данными.
- Informatica – мощная платформа для интеграции данных.
- Apache Nifi – удобное решение для потоковой обработки.
- **Для ELT:**
- BigQuery – облачное хранилище от Google с высокими вычислительными мощностями.
- Snowflake – гибкая платформа для хранения и обработки данных.
- Azure Data Factory – инструмент от Microsoft для управления потоками данных.

7. Заключение

В современном мире данные являются ключевым активом, а выбор подхода к их обработке может существенно повлиять на успех проекта.

Рассмотрим основные выводы:

Резюме сравнений:

- ETL остается классическим выбором для небольших проектов, где важна строгая обработка данных перед загрузкой.
- ELT набирает популярность благодаря своей гибкости, особенно в условиях облачных технологий и больших объемов данных.

Выводы:

- **Контекст – ключевой фактор:** Выбор между ETL и ELT должен основываться на задачах, инфраструктуре и объемах данных.
- **Долгосрочная перспектива:** Если компания планирует рост и увеличение данных, имеет смысл инвестировать в облачные решения и ELT.

Список литературы:

1. ETL против ELT: что лучше? Полное руководство (2024) // Астера [Электронный ресурс]. - Режим доступа: https://www.astera.com/ru/type/blog/etl-vs-elt-best-approach/. - Дата обращения: 16.12.2024.
2. Галанин В. ETL vs. ELT: что выбрать для вашего хранилища данных? // Хабр. URL: https://habr.com/ru/articles/695546/ (дата обращения: 16.12.2024). 3. Меньшикова Л. В., Исрафилов М. А. Анализ средств разработки динамических web-сайтов //Системный Анализ, Управление И Обработка Информации. – С. 202389.
4. Мытников А. Н. Сравнительный анализ ETL и ELT //Информатика и вычислительная техника. – 2020. – С. 139-145.

5. Рындина С. В. Технологии Анализа Больших Данных (Продвинутый Уровень): Управление И Руководство Данными, Хранение И Обработка Данных.
6. ЭТЛ или ЭЛТ: какой метод интеграции данных выбрать [Электронный ресурс] // FreeCodeCamp News : [сайт]. URL: https://tr-page.yandex.ru/translate?lang=en-ru&url=https%3A%2F%2Fwww.freecodecamp.org%2Fnews%2Fetl-vs-elt-which- data-integration-technique-should-you-choose%2F (дата обращения: 16.12.2024).