Инженерия данных. Профессия инженера данных

Обзор экосистемы инженерии данных

Из чего состоит экосистема инженера данных:

Инфраструктура, инструменты, фреймворки, процессы для реализации следующих этапов работы с данными:

- извлечения данных из разрозненных источников;
- проектирования и управления пайплайнами данных;
- проектирования и управления хранилищами данных;
- автоматизации и оптимизации воркфлоу и потоков данных между системами;
- разработки ПО для реализации процессов обработки данных.

На этапе извлечения данных из источника нужно оценить, с какими данными предстоит работать, чтобы спроектировать архитектуру целевой системы.

Какие бывают типы данных, и как тип данных влияет на их сбор, хранение и обработку:

По наличию структуры выделяют следующие типы данных:

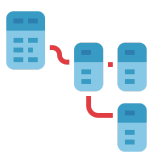
1. Структурированные;
2. Неструктурированные;
3. Слабоструктурированные.

Структурированные данные

Они имеют фиксированный формат (структуру) и могут быть представлены в виде совокупности строк и столбцов.

- Обычно содержатся в таблицах БД, электронных таблицах;
- Чаще всего относятся к машинно-генерированным данным и потребляются также машинами;
- Их удобно собирать и хранить в реляционных базах данных.

Типичные источники структурированных данных:



Реляционные базы данных

Данные датчиков

OLTP системы

Интернет вещей и промышленный интернет вещей

Электронные таблицы

Сетевые логи

Онлайн-формы

Логи веб-серверов

Неструктурированные данные

Не могут быть представлены в виде совокупности строк и столбцов, не содержат легко идентифицируемой при анализе структуры.

- Чаще всего создаются человеком и потребляются также человеком;
- Их хранение невозможно организовать в виде набора строк и столбцов в реляционных БД;
- Обычно хранятся в виде файлов и документов, также для их хранения используют базы данных NOSQL.

К неструктурированным данным относятся:

Содержимое текстовых документов

Видеопотоки

Тело электронных писем

Аудиозаписи

Изображения

Слабоструктурированные (полуструктурированные) данные

Частично могут быть представлены в виде совокупности строк и столбцов с неструктурированными включениями.

Представляют собой смесь структурированных и неструктурированных данных.

- Имеют некоторые общие признаки и свойства, но не имеют фиксированной структуры;
- Содержат теги и метаданные, используемые для группировки и организации данных в иерархические структуры;
- Для хранения и обмена слабоструктурированными данными используются языки разметки данных.

Примеры слабоструктурированных данных

Логи

- Частично содержат жесткую структуру – метку времени, имя машины, код ошибки;
- Описание ошибки не структурировано.

Электронные письма

- Заголовок письма – структурирован;
- Тело письма – не структурировано.

Некоторые источники слабоструктурированных данных


Электронные письма


TCP/IP (сетевые) пакеты

XML, JSON и другие языки разметки

Сжатые файлы (.zip)

Исполняемые файлы

Транзакционные и аналитические системы

Транзакционные системы

- Системы оперативной обработки транзакций (OLTP);
- Много транзакций;
- Небольшие порции данных;
- Чувствительны к задержкам обновления данных;
- Сбалансированное чтение/запись либо бОльшие нагрузки по записи;
- Предназначены для хранения больших объемов повседневных операционных данных;
- Обычно являются реляционными, но могут быть нереляционными.

Аналитические системы

- Системы оперативной аналитической обработки (OLAP);
- Немного транзакций;

- Большие порции данных;
- Чувствительны к пропускной способности;
- Большие нагрузки по чтению (включая полное сканирование таблиц);
- Оптимизированы для проведения комплексного анализа данных;
- К ним относятся реляционные и нереляционные базы данных, хранилища данных, витрины данных, озера данных и хранилища больших данных.

Пайплайн (конвейер) данных

- Набор инструментов и процессов, которые охватывают весь путь данных от их источника до конечного пользователя.

Для интеграции с источниками данных используются процессы:

- ETL (Extract-Transform-Load)
- ELT (Extract-Load-Transform)

Языки запросов и программирования, которые полезно знать инженеру данных

Языки, используемые для работы с данными:

1. Языки запросов

Предназначены для запросов и манипуляции данными в базах данных.

С помощью SQL можно:

- Добавлять, удалять и обновлять записи в БД;
- Создавать новые БД, таблицы, представления;
- Писать хранимые процедуры и вызывать их при необходимости.

2. Языки программирования

Предназначены для разработки приложений. работы с данными.

Преимущества Python:

- Быстрое прототипирование;
- Поддержка многих парадигм программирования;
- Готовые библиотеки для решения распространённых задач, в т.ч.:
  - очистки данных – Pandas
  - статистического анализа – Numpy, Scipy
  - веб скрэпинга – Scrapy, BeautifulSoap
  - визуального представления данных – Matplotlib, Seaborn
  - обработки изображений – OpenCV

Преимущества Java:

- Идеально подходит для написания быстровыполняемого кода;
- На Java написаны многие BigData-фреймворки и инструменты: Hadoop, Hive, Spark;
- Используется во многих процессах анализа данных, включая:

- очистку
- статистический анализ
- импорт и экспорт
- визуализацию

3. Скриптовые языки

Предназначены для автоматизации рутинных задач

Возможности Unix Shell:

- Манипуляции файлами в файловой системе;
- Запуск и остановка программ, процессов, задач;
- Резервное копирование, запрос логов о работе системы;
- Установочные скрипты для сложных программ и систем;
- Запуск задач по расписанию.

Примеры типовых задач, которые решает инженер данных:

1. Извлечение сырых данных из БД (SQL)
2. Разработка приложений для преобразования данных (Python)
3. Написание shell-скриптов для повторяющихся операционных задач (Unix Shell)
4. Предоставление и обслуживание BI-инструментов (дашборды)

Как вам урок?

Далее