Engineer

30 Tage Stromdaten in 180 ms

Varianten zur stündlichen Aggregation des Stromverbrauchs in Python, SQL und Bash + Jq entwickelt; der Python-Benchmark verarbeitete einen 30-Tage-JSONL-Datensatz in 180 ms.

2025

180 ms

für die Aggregation eines Monats an Zählerständen

Situation. Das Unternehmen verarbeitet große Mengen heterogener Stromdaten aus mehreren Quellen mit unterschiedlichen Meldeintervallen — von stündlichen bis zu 15-minütigen Abständen, teils mit unregelmäßigen Zeitstempeln. Diese Unterschiede erschweren die Erstellung eines zusammenhängenden, vergleichbaren Datensatzes. Für präzise Energieanalysen müssen die Daten auf einheitliche stündliche Verbrauchswerte normalisiert und je Zone aggregiert werden.

Aufgabe. Rohe Ereignisdaten aus einem historischen Datensatz im JSONL-Format sollten in stundengenau ausgerichtete Stromverbrauchswerte umgewandelt werden, mit einer Zeile je Stunde und Zone. Die Aufgabe umfasste insbesondere:

- Daten mit Zeitstempel an exakten Stundenintervallen ausrichten,
- die gesamte Stromproduktion innerhalb jedes Intervalls durch Summieren der Mix-Werte aggregieren,
- grenzüberschreitenden Austausch durch Addition der Importe und Abzug der Exporte berücksichtigen,
- die Endwerte in einem strukturierten, skalierbaren Format für weitere Analysen ausgeben.
- Die Lösung musste zudem effizient genug für große Zeitfenster von mindestens 30 Tagen und mehrere Länder beziehungsweise Zonen sein.

Massnahmen. Drei Lösungsvarianten mit Python, JQ für JSON-Verarbeitung an der Kommandozeile und SQL wurden umgesetzt und jeweils für unterschiedliche Einsatzbedingungen optimiert:

Python wurde wegen seiner Flexibilität, einfachen Datenbearbeitung und effizienten Transformationen im Arbeitsspeicher gewählt. Mit reinen Python-Funktionen entstand eine Pipeline, die:

- die JSONL-Dateien in Dataframes einlas,
- die Zeitreihen auf Stundenintervalle umrechnete,
- die Produktion aggregierte und den Nettostromverbrauch berechnete (Produktion + Importe – Exporte),
- die Ergebnisse als CSV exportierte oder zur Prüfung in eine schlanke SQLite-Datenbank lud.

JQ diente als schnelle Lösung mit wenigen Abhängigkeiten für Kommandozeilenumgebungen und wurde als Kette von JQ-Filtern umgesetzt.

In PostgreSQL wurden die JSONL-Daten importiert, normalisierte Tabellen angelegt und eine Reihe von SQL-Abfragen geschrieben.

Zur Performancebewertung wurden alle Lösungen mit historischen Datensätzen für 1 Tag und 30 Tage verglichen.

Ergebnis. Die Python-Implementierung erwies sich als schnellste und am besten skalierbare Lösung mit folgenden Laufzeiten:

- Verarbeitung der Daten für 1 Tag in nur 34 ms,
- Verarbeitung des Datensatzes für 30 Tage in 180 ms.

Damit war ihre Eignung für größere Zeiträume bei Laufzeiten unter einer Sekunde belegt. Der Code blieb außerdem klar und wartbar und ließ sich leicht um die Verarbeitung mehrerer Zonen erweitern oder in eine ETL-Pipeline integrieren.

Insgesamt zeigte der Einsatz mehrerer Werkzeuge einen flexiblen Umgang mit Technologien, gezielte Performanceoptimierung und eine zuverlässige Bewältigung von Zeitraster- und Aggregationsproblemen in

realen Stromdatenpipelines.

Diese Erfahrung für Ihr Projekt nutzen

Nutzen Sie „30 Tage Stromdaten in 180 ms“ als Ausgangspunkt für eine ähnliche Herausforderung in Ihrer Organisation. Die Leistungen hinter dieser Arbeit stehen auch für Ihr Projekt zur Verfügung. Die Abfragen und fehlenden Indizes hinter einem von Jahr zu Jahr schwerfälligeren System finden und die entscheidenden Probleme beheben.

Unsere Leistungen entdecken Kontakt aufnehmen →

Hinter das Ergebnis schauen

Lesen Sie die Situation, die Entscheidungen und das Ergebnis dieses Beispiels. Wir können besprechen, welche Teile sich auf Ihre Rahmenbedingungen übertragen lassen und welche einen anderen Ansatz erfordern. 30 Tage Stromdaten in 180 ms — 180 ms für die Aggregation eines Monats an Zählerständen

Unsere Leistungen entdecken Kontakt aufnehmen →

Brauchen Sie schnelle, effiziente Datenpipelines?

Besprechen Sie Ihr Projekt mit der Ingenieurin, die für die Umsetzung verantwortlich ist. Wir vereinbaren Umfang, Preisgrundlage und Übergabe schriftlich, bevor die Arbeit beginnt.

Rollenkontext: Plattform- und Dateningenieurin. Die Arbeit baut auf Datenbanken, PostgreSQL, SQL, Data Engineering, Datenpipelines (ETL/ELT), Datenanalyse, Automatisierung und CI/CD, Performanceoptimierung und Python. Und wenn Sie schon hier sind: Wir sind auch ziemlich gut in Datenbank-Performanceoptimierung, Entwicklung von Datenpipelines (ETL/ELT) und Datenanalyse und BI-Dashboards.

Kontakt aufnehmen Unsere Leistungen entdecken

Ein Teil einer **längeren Arbeitsbilanz** — vollständig auf dieser Website zu finden.

Jeder Eintrag folgt demselben Aufbau: die Situation, die Aufgabe, unser Vorgehen und die Veränderungen.

Alle Arbeiten ansehen

Die benötigte Verbesserung definieren

Bringen Sie einen repräsentativen Datensatz, die Fragen, die er beantworten muss, und den aktuellen Engpass mit. Wir können vereinbaren, wie Qualität, Durchsatz und Wiederherstellung gemessen werden.

Preise ansehen Kontakt aufnehmen →

Brauchen Sie schnelle, effiziente Datenpipelines?

Fragen Sie nach den Entscheidungen hinter „30 Tage Stromdaten in 180 ms“. Sie sprechen mit den Menschen, die die Arbeit erklären und Ihr eigenes Projekt einschätzen können.

Unsere Leistungen entdecken

welcome@engineer.company +45 50 10 61 84 Kopenhagen, Dänemark

Nehmen Sie diese Seite mit

Kontaktkarte

Mit Sorgfalt und Liebe gestaltet von D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – heute · Engineer ApS · Alle Rechte vorbehalten.

Gedruckt von https://engineer.company/de/portfolio/a-month-of-meter-readings-in-180-milliseconds/