**Engineer**

# 30 jours de données électriques traités en 180 ms

Conçu des variantes d'agrégation horaire de la consommation électrique en Python, SQL et Bash + Jq ; le benchmark Python a traité un jeu de données JSONL de 30 jours en 180 ms.

2025

## 180 ms

pour agréger un mois de relevés de compteurs

**Situation.** L'entreprise traite de grandes quantités de données électriques hétérogènes provenant de plusieurs sources, chacune avec sa fréquence de remontée : des intervalles horaires aux intervalles de 15 minutes, voire des horodatages irréguliers. Cette variabilité complique la constitution d'un jeu de données cohérent et comparable. Pour permettre des analyses énergétiques précises, les données doivent être normalisées en valeurs de consommation horaires homogènes, agrégées par zone.

**Tâche.** L'objectif était de convertir les événements bruts d'un jeu de données historique au format JSONL en chiffres de consommation électrique alignés sur l'heure, avec une ligne par heure et par zone. Il fallait précisément :

- Aligner les données horodatées sur des intervalles horaires stricts,
- Agréger la production totale d'électricité de chaque intervalle en additionnant les valeurs du mix,
- Intégrer les échanges transfrontaliers en ajoutant les importations et en soustrayant les exportations,
- Produire les valeurs finales dans un format structuré et évolutif, adapté aux analyses ultérieures.
- La solution devait également être assez efficace pour de longues périodes (30 jours ou plus) et plusieurs pays ou zones.

**Action.** Trois variantes ont été mises en œuvre avec Python, JQ pour le traitement JSON en ligne de commande, et SQL, chacune optimisée pour un contexte différent :

Python a été retenu pour sa souplesse, sa facilité de manipulation des données et son efficacité pour les transformations en mémoire. Des fonctions Python pures ont servi à construire un pipeline qui :

- Analysait les fichiers JSONL pour les convertir en tableaux de données
- Rééchantillonnait les séries temporelles sur des intervalles horaires
- Agrégeait la production et calculait la consommation électrique nette (production + importations – exportations)
- Exportait les résultats en CSV ou les chargeait dans une base SQLite légère pour inspection.

JQ a permis une solution rapide avec peu de dépendances pour les environnements en ligne de commande, sous forme d'une chaîne de filtres JQ.

Dans PostgreSQL, les données JSONL ont été importées, des tables normalisées créées et une série de requêtes SQL écrite.

Les performances de chaque solution ont été mesurées sur des historiques de 1 jour et de 30 jours.

**Résultat.** L'implémentation Python s'est révélée la plus rapide et la plus adaptée à la montée en charge, avec les temps suivants :

- Traitement de 1 jour de données en seulement 34 ms
- Jeu de données de 30 jours en 180 ms

Ces résultats ont confirmé son adéquation aux périodes plus longues tout en conservant un traitement inférieur à la seconde. Le code était aussi clair, maintenable et facile à étendre à plusieurs zones ou à intégrer dans un pipeline ETL.

L'approche à plusieurs outils a ainsi montré de la souplesse dans leur choix, une optimisation efficace des performances et une prise en charge solide des difficultés d'alignement temporel et d'agrégation des pipelines électriques réels.

### Mettez cette expérience au service de votre projet

Prenez « 30 jours de données électriques traités en 180 ms » comme point de départ pour un défi similaire dans votre organisation. Les prestations mobilisées pour cette réalisation sont disponibles pour votre projet. Identifier les requêtes et les index manquants dans un système alourdi au fil des ans, puis corriger ce qui compte.

**Découvrir nos prestations** Contactez-nous →

### Examinez ce qui a conduit au résultat

Découvrez la situation, les décisions et le résultat de cet exemple. Nous pouvons discuter des éléments transposables à vos contraintes et de ceux qui nécessitent une autre démarche. 30 jours de données électriques traités en 180 ms — 180 ms pour agréger un mois de relevés de compteurs

**Découvrir nos prestations** Contactez-nous →

## Besoin de pipelines de données rapides et efficaces ?

Discutez de votre projet avec l'ingénieure responsable de sa réalisation. Nous convenons par écrit du périmètre, de la base tarifaire et de la transmission avant le début des travaux.

Rôle exercé dans ce contexte : Ingénieure plateformes et données. Cette réalisation fait appel à Bases de données, PostgreSQL, SQL, Ingénierie des données, Pipelines de données (ETL/ELT), Analyse de données, Automatisation et CI/CD, Optimisation des performances et Python. Et puisque vous êtes ici, nous sommes aussi plutôt doués en Optimisation des performances de bases de données, Développement de pipelines de données (ETL/ELT) et Analyse de données et tableaux de bord BI.

**Contactez-nous** Découvrir nos prestations

## Un élément **d'un parcours plus long** — entièrement présenté sur ce site.

Chaque entrée suit la même structure : la situation, la tâche, ce que nous avons fait et ce qui a changé.

**Voir l'ensemble du parcours**

### Définissez l'amélioration nécessaire

Présentez un jeu de données représentatif, les questions auxquelles il doit répondre et le goulot d'étranglement actuel. Nous pouvons convenir de la manière de mesurer la qualité, le débit et la reprise.

**Consulter les tarifs** Contactez-nous →

# Besoin de pipelines de données rapides et efficaces ?

Interrogez-nous sur les décisions qui sous-tendent « 30 jours de données électriques traités en 180 ms ». Vous parlerez aux personnes capables d'expliquer le travail et d'évaluer votre propre projet.

**Découvrir nos prestations**

welcome@engineer.company +45 50 10 61 84 Copenhague, Danemark

Emportez cette page avec vous

Fiche de contact

**Créé avec soin et amour par D.**

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

**Droit d'auteur © 2025 – aujourd'hui · Engineer ApS · Tous droits réservés.**

**Imprimé depuis https://engineer.company/fr/portfolio/a-month-of-meter-readings-in-180-milliseconds/**