Reference implementations

1 Data stations from the composable data stack

- Python and SQL(-like) languages as the *de facto* standard for analytical processing i.e. the most commonly used analytical scripting languages. Where necessary, using Intermediate Representations (IR), any analytical query can be transpiled to the target engine of choice
- Single-node compute capable of efficiently processing up to 1 TB of data within tens of seconds (polars, DuckDB), so we do away with distributed processing
- Open table formats (Iceberg, Hudi, Delta) and open file formats (parquet, AVRO)

2 Container-based trains as the most versatile solution

3 Bringing it all together for federated analytics & machine learning (FL)

- Local data stations are conceptualized as serverless lakehouses
  - Local ELT pipelines
  - Decentralized (pre-)processing, including quality control upon ingest
  - …
- For horizontally partioned data, we can apply FL techniques where only aggregated results are combined centrally
- For vertically partitioned data, we need an intermediate/temporary zone for linking the data
- For both horizontally and vertically partitioned data, we can choose to add PETs, most specifically MPC, as an extra security measure
  - Horizontally partitioned data: one-shot FL
  - Vertically partitioned data:
  - linkage in the blind
  - reversible pseudonimization
- standardized approach to mapppings [1]

Bibliography

1. Zhang S, Cornet R, Benis N (2024) Cross-Standard Health Data Harmonization Using Semantics of Data Elements. Scientific Data 11(1):1407. https://doi.org/10.1038/s41597-024-04168-1