Federated Data Quality Framework for Biomedical Data

Federated systems have long been promoted as alternatives to centralised architectures and single points of failure. However, decentralisation brings its own challenges—chiefly, the lack of consistent data-quality oversight. This poster presents our Federated Data Quality Framework, which enables quality assessment across federated nodes handling sensitive health data by embedding state-of-the-art privacy-preserving techniques such as differential privacy into a Security-by-Design approach.

Authors
Radovan Tomášik[1,2,3]
Simona Menšíková[1]

1. Masarykův onkologický ústav, Žlutý kopec 7, 656 53 Brno
2. Fakulta informatiky MU, Botanická 68A, 602 00 Brno-Královo Pole-Ponava
3. BBMRI-ERIC, Neue Stiftingtalstrasse 2/B/6, 8010 Graz, Austria

1 Motivation

Federated health-data networks keep each institution's records locally, which eliminates a single point of failure but also lacks any built-in, network-wide data-quality oversight. Without centralized monitoring, inconsistencies, missing fields, or duplicate records can remain hidden, jeopardising the reliability of downstream analyses.

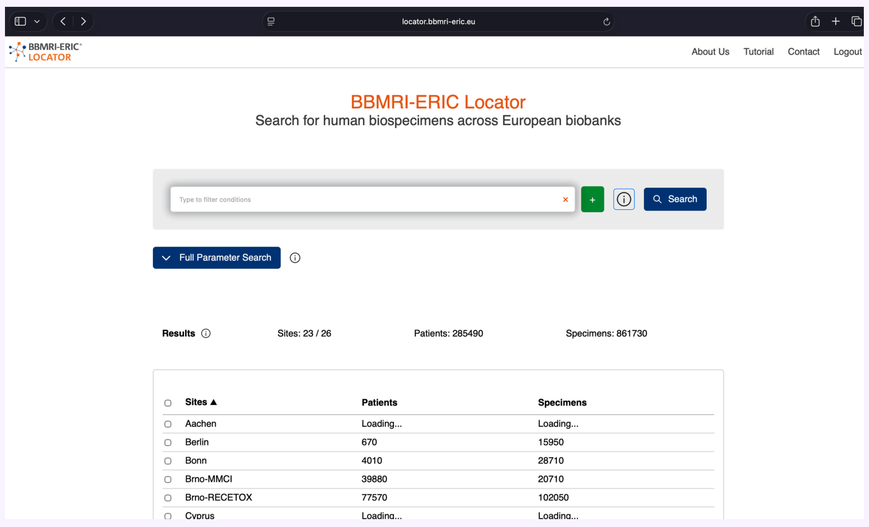

Figure 1. Motivation

2 Defining domain specific quality checks

ISO 8000 defines data quality as fitness for purpose, which means that achieving high-quality data requires domain-specific quality checks. In a federated system designed for searching biological specimens and assembling patient cohorts, a concrete example of such a check is: "How many patients have an invalid ICD-10 diagnosis associated with them?"

Description	Dimension
How many patients have an incompatible Diagnosis (e.g. Prostate Cancer for Female)	Accuracy
How many patients do not have a condition	Completness
Survival rate for patients stratified by gender	Accuracy
How many patients were last updated more than a year ago	Timeliness

Figure 2. Data Quality Checks

3 Creating machine actionable representation

Once a human-readable data-quality rule is defined, it must be translated manually into an executable query—typically a SQL statement or an HL7 CQL expression—that runs against the local data store.

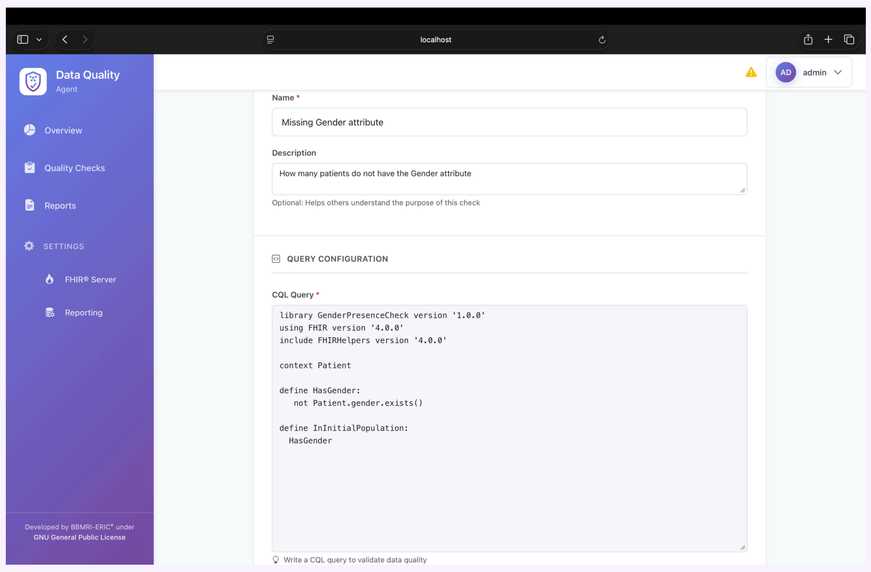

Figure 3. CQL Query example

4 Local data quality assessment

A local component (**Data Quality Agent**) is deployed with access to the database, where it executes the data quality checks and shows the results to the local data manager. By highlighting faulty records in the database it allows the local data manager to identify ETL issues.

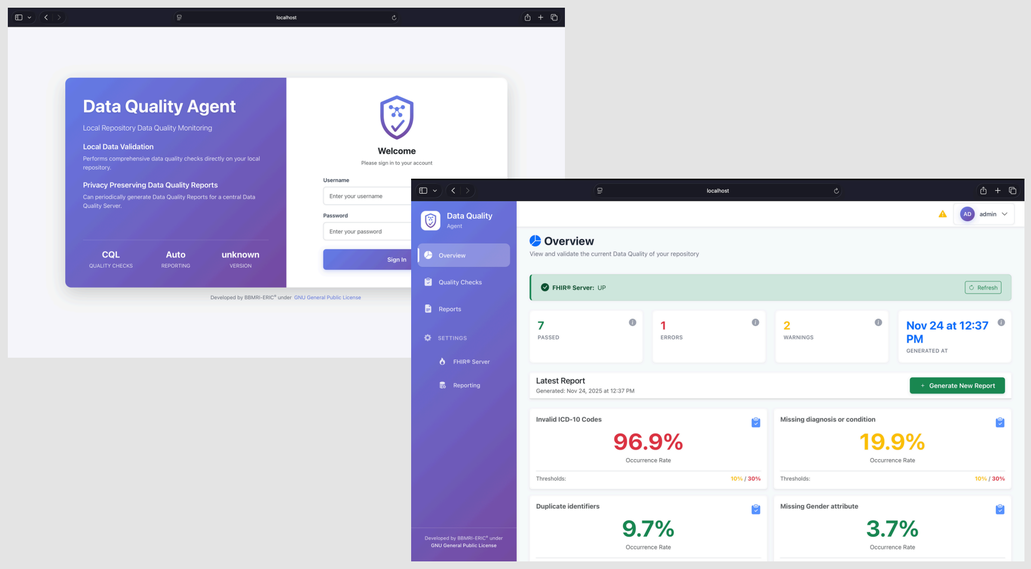

Figure 4. Data Quality Agent

5 Sharing the results

The results can then be reported to a central **Data Quality Server** in a privacy-preserving manner that preserves their statistical significance.

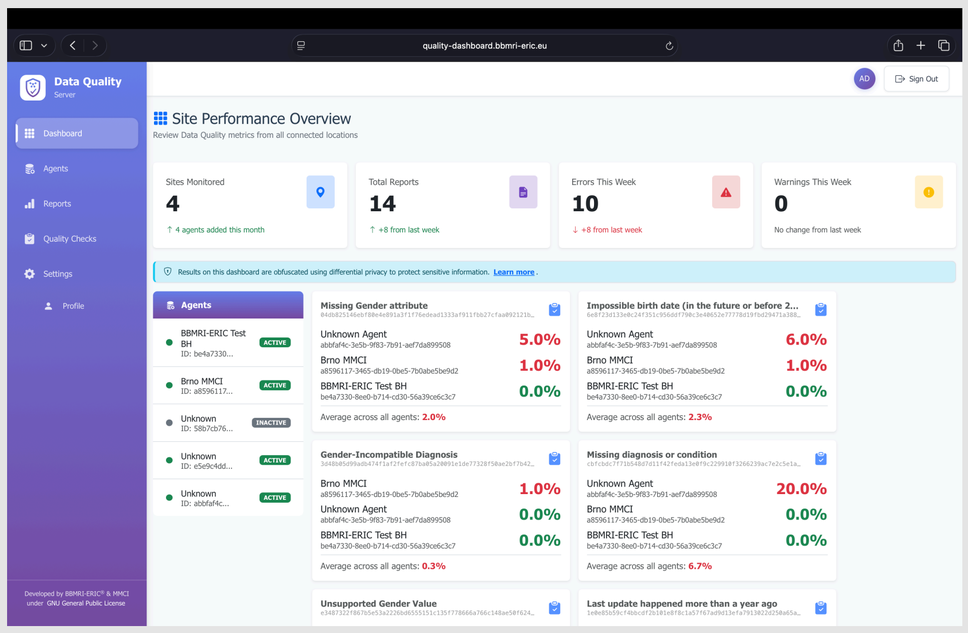

Figure 5. Data Quality Server

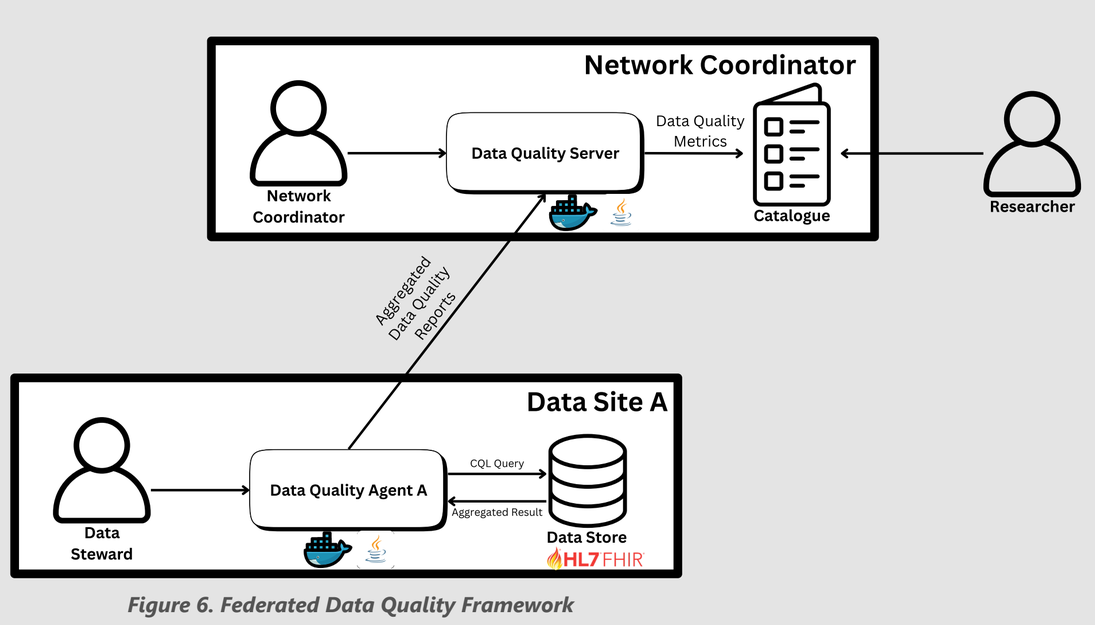

Figure 6. Federated Data Quality Framework