From legacy systems to data pipelines modernization in fermentation process

Anaëlle Dessaignea, Amelie Brianea, Fayza Daboussia, Julien Cescuta , David Camilo Corralesa

aINRAE, UMS (1337) TWB, 135 Avenue de Rangueil, 31077 Toulouse, France

David-Camilo.Corrales-Munoz@inrae.fr

Abstract

Developments in biotechnology using high throughput systems are increasingly and consequently the creation and consumption of data continue to grow rapidly. Data migration is an essential part of legacy system modernization in bioprocess. Migration process involves transferring data from outdated platforms or unknown data schemas to more advanced and secure systems. Data migration can be represented through data pipelines including data extraction, transformation and loading (ETL). The data pipelines are implemented in order to increase the overall efficiency of data-flow from the source (raw data) to the knowledge generation (Mohanty et al., 2013). Legacy systems in fermentation generally occur in bioreactor components as sensors, protocols, software or databases. These issues can limit the integration with modern tools and systems as Process Analytical Technology (PAT) instruments (Gerzon et al., 2022), avoiding real-time data on process parameters and thereby fail to assist operators in maintain optimal conditions for cell growth and production. The aim of this research is to present a guided process for designing data pipelines in bioreactors legacy systems. We present as use case a set of 24 mini-bioreactors of 50 mL. We conducted unit testing for components of the ETL process in order to ensure the integration and migration process of the legacy DB.

**Keywords**: Extract, Transform, Load (ETL); Database (DB); Dashboard; Directed Acyclic Graph (DAG); Mini-bioreactor

* 1. Introduction

The propagation of Internet-enabled technologies; the capability to collect, manage, and use “big data”; as well as the evolution of refined analysis and predictive techniques are now facilitating and pushing towards the implementation of Industry 4.0 – also known as smart manufacturing (Oliveira, 2019). Industry 4.0 represents a new paradigm in our society since the industrial revolution. This new paradigm demands the integration of digital systems and devices to enable online processing of vast quantities of data (Gargalo et al. 2021). This is therefore leading progressively to the introduction and adoption of digital solutions to predict and optimize the behavior of the production process at each product life cycle stage in real time.

Currently, legacy systems and legacy databases are confronted by the demands of Industry 4.0, which emphasizes interconnectedness, real-time data analytics, and adaptability. The historical legacy systems has impeded the integration into dynamic data pipelines essential for Industry 4.0 advancements. Moreover, the integration challenges within the field of fermentation, marked by historical data capture methods and limited real-time monitoring capabilities; emphasize the need for a paradigm shift in data management strategies for the legacy data acquisition software that controls bioreactors.

In this paper, we propose a data pipeline solution for enhanced data management in a robotic bioprocessing platform that integrates a set of 24 mini-bioreactors. By introducing a comprehensive and adaptable data pipeline, our goal is overcome the obstacles concerned to legacy databases retrieval and improve the data accessibility and data integration. The rest of the paper is organized as follows. Section 2 describes the bioprocess platform. Section 3 presents the proposed data pipeline. Section 4 provides the results, Section 5 presents the conclusions, and future works.

* 1. Bioprocessing platform

The robotic bioprocessing platform built in 2013 integrates a set of 24-instrumented mini-bioreactors (Hamilton, 2015). These mini-bioreactors consist in a miniaturization of 50 mL reactor with measurement and control by mini-probes of the temperature, pH, agitation speed, airflow, and oxygen concentration. The set of 24-minibioreactors is distributed equally among three polyBLOCKs (PB). The polyBLOCK is responsible for heating, cooling and stirring. Each reactor has its own heating zone which is heated electrically.

The microorganism growing in the mini-bioreactors is monitored by ‘in-line’ turbidity probes. The platform is designed to perform automated sampling (up to one sampling every 20 min on the 24 mini-reactors). The pipetting robot collects a sample in a reactor and immediately places it in a microplate maintained at -20°C. The data acquisition software (DAS) allows to export the fermentation parameters (T°, pH, pO2, etc.).

* 1. Data pipeline architecture

Data pipelines transport raw data from database sources for use by analytics tools. This paper details the composition of the data pipeline for 24 mini-bioreactors mentioned before. The data pipeline is composed by a set of steps used to move the raw fermentation data from the legacy database to a more appropriate and modernized time series database. It involves the flow of data through a series of stages where data is extracted, transformed, and loaded efficiently. Figure 1 depicts the data pipeline architecture for 24 mini-bioreactors.



**Figure 1.** Data pipeline architecture for 24 mini-bioreactors

The key components of the data pipeline architecture include five steps:

* + 1. Data acquisition software (DAS)

It serves as the bridge between the physical mini-bioreactors and the digital side representing by the legacy DB. DAS communicates with sensors as pH, temperature, oxygen, agitation in order to collect information. The data acquisition software called WinISO provides an interface to configure, control, and receive data from mini-bioreactors. WinISO 2.3.3 is Windows based program provided by H.E.L Group (Martin, 2013). Currently, H.E.L Group launched a new DAS called labCONSOL, which replaces the older WinISO software.

* + 1. Legacy DB

The data generated by sensors of the mini-bioreactors are stored in a legacy database SQL Server 2019. This database is part of the WinISO software. The proprietary software imposes a limitation by not allowing direct access to the data persistence. Although the restriction to the access to the database is intended to protect the data and obviously create a long-term dependency on the proprietary software provider, this limitation restricts the integration with other tools. The main challenge to accessing the database was to understand its data schema, particularly in deciphering the complex relationships and dependencies between tables.

* + 1. Workflow management

Once the data schema of the legacy DB was deciphered, different workflows were coded in order to apply the ETL process. We used Apache Airflow as workflow management platform. Airflow allows creating, scheduling and monitoring workflows coded in Python.

The workflows are expressed as Directed Acyclic Graphs (DAGs), where each node represents a specific task. Each task in the DAG is a unit of work that can encompass running a Python script to executing a database query. DAGs are executed in parallel, enabling the concurrent execution of multiple DAG runs. Upon completion of a DAG run, Apache Airflow notifies its final status using either a "success" or "failed" tag.

We built three workflows, represented by three DAGs to manage ETL process from legacy DB to a time series DB (Figure 1). Each DAG is responsible for performing the ETL process for each PolyBLOCK. These DAGs contains four tasks:

* *Waiting\_data* checks whether new DoEs have been created. WinISO creates a Design of Experiment (DoE) and this is stored in the legacy database.
* *Creating\_time\_interval* has been implemented as function to trigger the execution of DAGs at regular intervals, specifically every five minutes.
* *Searching\_data* looks new data from DoE identified in *Waiting\_data* function
* *ETL* applies on detected data the ETL process:
	+ *Extract:* during data extraction, data of the mini-bioreactors is copied by batch from legacy DB to a staging area. The extraction process involved mixing SQL queries and code written in python due to the complexity of the data schema of the legacy DB.
	+ *Transform*: in the staging area, the raw fermentation data is preprocessed. Missing values were handled through imputation methods. Additionally, the names of the variables of the mini-bioreactors were harmonized due to inconsistencies or variations in nomenclature of the sensor names.
	+ *Load:* in this last step, the transformed data is moved from the staging area into a time series DB. The data that were already in the legacy DB were loaded. Data from running experiments are continuously loaded in periods of 5 minutes.
		1. Time series DB

We used InfluxDB as time series database. InfluxDB was designed for handling and managing time-stamped or time-series data. Time series data consists of sequences of data points (e.g., the temperature of one mini-bioreactor by minute), each associated with a specific timestamp, ordered chronologically. This type of database uses a tag-value data model which is well-suited for scenarios where data points are indexed based on time, and each data point is associated with additional metadata in the form of tag-value pairs.

* + 1. Dashboards

A dashboard provides a visual data representation in a consolidated and easy-to-digest-form. Dashboards in fermentation are useful for monitoring the key parameters (temperature, pH level), the bioreactor performance (agitation and aeration), the nutrient levels (e.g., glucose, nitrogen, glycerol, ethanol, etc.), and the biomass and product concentration. We built the dashboards for visualizing the processed data of the mini-bioreactors in Grafana open-source platform. Subsequently, we connected Grafana to InfluxDB for the visualization and analysis of mini-bioreactors data of previous experiments or running experiments. The dashboards were designed with an expert panel from fermentation service.

* 1. Results

In order to ensure the integration and migration process of the legacy DB to the time series database, we conducted unit testing for components of the ETL process, such as source extraction, data transformation, and target loading (Zhu et al., 1997). We used data from four experiments that were already in the legacy DB. Table 1 presents the dimensionality of those experiments characterized by average experiment length, the number of mini-bioreactors for each experiment, the number of variables of the mini-bioreactors and the data points for each variable.

Table 1. Dimensionality of experiments run in the mini-bioreactors.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Experiments** | **Average experiment length (hours)** | **Mini-bioreactors** | **Variables** | **Data points**  |
| RUN 1 | 48h25 | 8 | 557 | 581 |
| RUN 2 | 72h55 | 15 | 557 | 875 |
| RUN 3 | 75h | 16 | 557 | 300 |
| RUN 4 | 72h | 16 | 557 | 864 |

We employed the four experiments presented in Table 1 to conduct unit testing for the ETL components. These experiments correspond to the production of monoclonal antibodies. The application of the ETL process was conducted considering that the data of the four experiments are already stored in the database. Figure 2 presents the execution time for the components Extract, Test and Load.

In Run 1, the duration of the experiment was 48 hours and 25 minutes with 8 mini-bioreactors, 557 variables, and 581 data points, the Extract, Transform, and Load stages consumed times of approximately 37.63, 138.88, and 150.67 seconds, respectively. Run 2 encompasses an experiment duration of 72 hours and 55 minutes with 15 mini-bioreactors, 557 variables, and 875 data points, demonstrated increased processing times in all stages: 78.96, 459.76, and 334.02 seconds for Extract, Transform, and Load, respectively. Notably, Run 3, with 75 hours of experiment duration, 16 mini-bioreactors, 557 variables, and 300 data points, presented a reduction in the Extract and Load stages, with 49.26 and 225.37 seconds, while the Transform stage maintained a relatively high time of 315.88 seconds. The Run 4 with a 72 hours of experiment and featuring 16 mini-bioreactors, 557 variables, and 864 data points, the Extract, Transform, and Load stages displayed times of 114.19, 598.26, and 293.26 seconds, respectively.

**Figure 2.** Unit testing for ETL components based on four experiments stored in the legacy DB. Execution time expressed in seconds.

On average, the Transform stage exhibits the highest processing time, with an average of approximately 378.94 seconds, followed by the Load stage with an average time of around 250.83 seconds. On the other hand, the Extract stage shows the shortest average time at approximately 69.76 seconds. These findings indicate the Transform stage consumes a significant portion of the overall processing time due this stage involves the mix of SQL queries and code written in python due to the complexity of the data schema of the legacy DB.

* 1. Conclusions and future works

Accessibility to legacy databases presents a multitude of problems, primarily the outdated and often proprietary nature of these systems. The simplest solution is to pay for the new version of the data acquisition system; however, this incurs additional costs and in the future, will become a legacy system. In order to address these issues in the field of fermentation, we proposed a data pipeline for 24 mini-bioreactors with the objective to deal the data management of legacy software systems. The data pipeline proposes a set of steps to move the raw data from the legacy database to a more appropriate and modernized time series database through ETL process.

In summary, the Transform was the stage that demanded most processing time, primarily attributed to the legacy data schema. The complexity of the legacy database structure necessitates intricate transformations to align the data of bioreactors, variables (Temperature, pH, Oxygen, etc.) and values with the target schema, introducing additional processing steps with SQL queries and python scripts.

In conclusion, the implementation of the data pipeline for 24 mini-bioreactors improved the data management of the legacy databases by providing a structured and efficient mechanism for data integration, transformation, and migration. In general, the data pipelines can be an opportunity to facilitates the smooth transition from outdated bioreactor DAS, ensuring data consistency, improved accessibility, and enhanced interoperability with open source applications as Apache Airflow, InfluxDB and Grafana.

In addition, data pipelines play an important role in the construction of digital twins in bioprocess. These pipelines serve to collect, process, and integrate data in real-time, ensuring that the digital twin stays synchronized with its physical counterpart or the system it represents. As future work, we propose the construction of digital thread, an essential component to connect the physical objects with the digital objects, by facilitating measurement, sensing, monitoring, control, and communication about everything that is happening throughout the production process (Gargalo et al. 2021).

Acknowledgements

This work was funded (or co-funded) by the European Union under the Horizon Europe project Bioindustry 4.0, grant n. 101094287.

References

Gargalo, Carina L., Simoneta Caño de las Heras, Mark Nicholas Jones, Isuru Udugama, Seyed Soheil Mansouri, Ulrich Krühne, et Krist V. Gernaey. 2021. “Towards the Development of Digital Twins for the Bio-Manufacturing Industry”. P. 1‑34 in Digital Twins: Tools and Concepts for Smart Biomanufacturing, Advances in Biochemical Engineering/Biotechnology.

Gerzon, G., Sheng, Y., & Kirkitadze, M. (2022). Process Analytical Technologies – Advances in bioprocess integration and future perspectives. Journal of Pharmaceutical and Biomedical Analysis, 207, 114379. <https://doi.org/10.1016/j.jpba.2021.114379>

Hamilton Company. TWB and Hamilton Develop a Unique and Custom Made Microbial Culture Robot, Utilizing Cutting-Edge Technology for the Field of Industrial Biotechnology High Throughput Culture Platform. (2015). <https://www.hamiltoncompany.com/press-releases/twb-and-hamilton-develop-a-unique-and-custom-made-microbial-culture-robot-utilizing-cutting-edge-technology-for-the-field-of-industrial-biotechnology#top>.

Martin, David. 2013. WinISO Data Acquisition and Control Software (User Manual 2.0.1.). H.E.L Group.

Mohanty, S., Jagadeesh, M., & Srivatsa, H. (2013). “Big Data” in the Enterprise. In S. Mohanty, M. Jagadeesh, & H. Srivatsa (Éds.), Big Data Imperatives : Enterprise Big Data Warehouse, BI Implementations and Analytics (p. 1‑24). Apress. <https://doi.org/10.1007/978-1-4302-4873-6_1>

Oliveira, A. L. (2019). Biotechnology, Big Data and Artificial Intelligence. Biotechnology Journal, 14(8), 1800613. <https://doi.org/10.1002/biot.201800613>

Zhu, H., Hall, P. A., & May, J. H. (1997). Software unit test coverage and adequacy. Acm computing surveys (csur), 29(4), 366-427.