frontiers | Frontiers in Chemical Engineering

TYPE Original Research
PUBLISHED 14 December 2022
DOI 10.3389/fceng.2022.1036867

OPEN ACCESS

EDITED BY
Fengqi You,
Cornell University, United States

REVIEWED BY
Irene Mei Leng Chew,
Monash University Malaysia, Malaysia
Xingsi Xue,
Fujian University of Technology, China

*CORRESPONDENCE
Carina L. Gargalo,
carlour@kt.dtu.dk

SPECIALTY SECTION
This article was submitted to Computational Methods in Chemical Engineering, a section of the journal Frontiers in Chemical Engineering

RECEIVED 05 September 2022
ACCEPTED 30 November 2022
PUBLISHED 14 December 2022

CITATION
Caño De Las Heras S, Gargalo CL, Caccavale F, Gernaey KV and Krühne U (2022), NyctiDB: A non-relational bioprocesses modeling database supported by an ontology. *Front. Chem. Eng.* 4:1036867. doi: 10.3389/fceng.2022.1036867

NyctiDB: A non-relational bioprocesses modeling database supported by an ontology

Simoneta Caño De Las Heras, Carina L. Gargalo*, Fiammetta Caccavale, Krist V. Gernaey and Ulrich Krühne

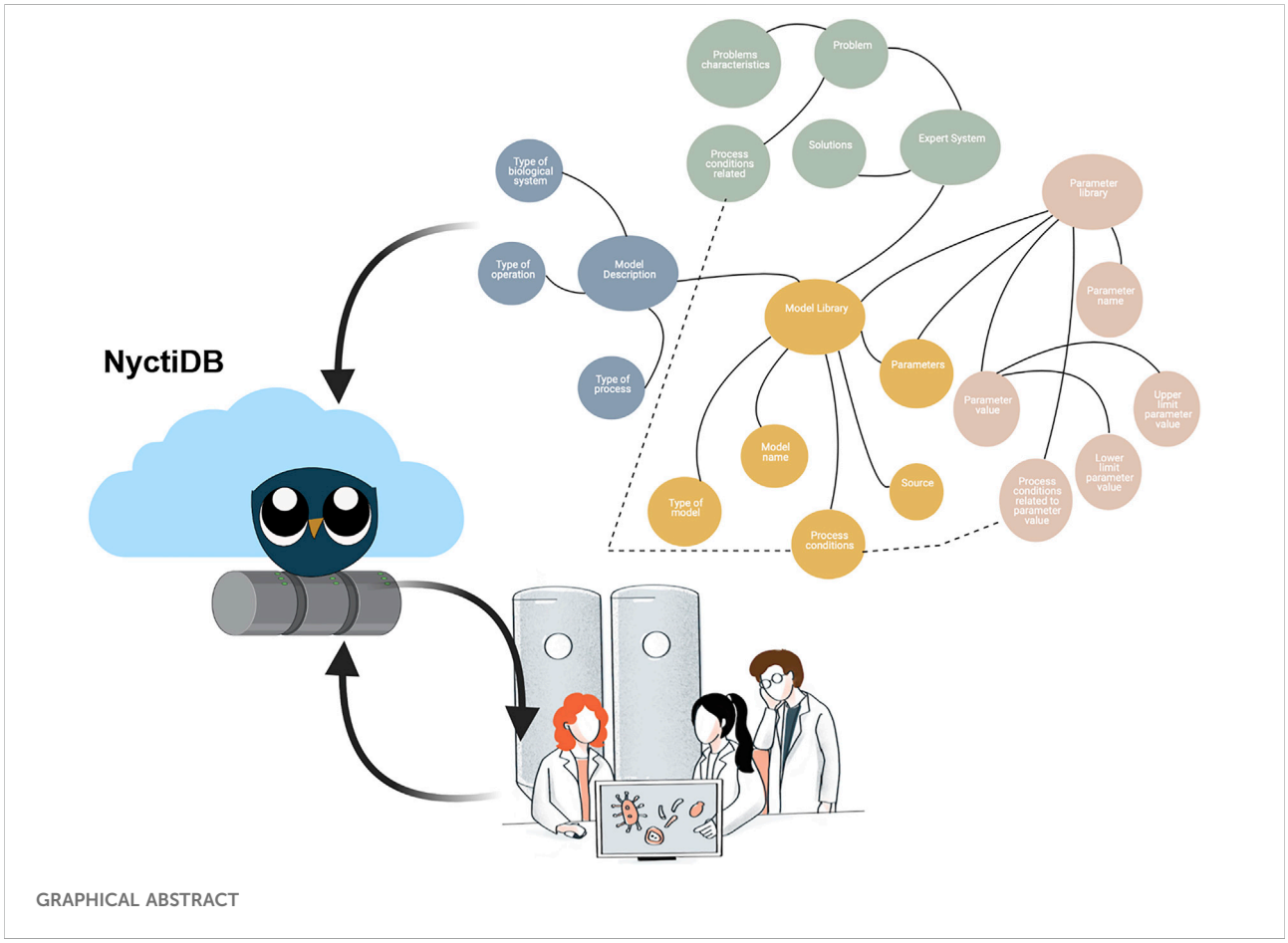
Process and Systems Engineering Center (PROSYS), Department of Chemical and Biochemical Engineering, Technical University of Denmark, Lyngby, Denmark

Strategies to exploit and enable the digitalization of industrial processes are on course to become game-changers in optimizing (bio)chemical facilities. To achieve this, these industries face an increasing need for process models and, as importantly, an efficient way to store the models and data/information. Therefore, this work proposes developing an online information storage system that can facilitate the reuse and expansion of process models and make them available to the digitalization cycle. This system is named *NyctiDB*, and it is a novel non-relational database coupled with a bioprocess ontology. The ontology supports the selection and classification of bioprocess models focused information, while the database is in charge of the online storage of said information. Through a series of online collections, *NyctiDB* contains essential knowledge for the design, monitoring, control, and optimization of a bioprocess based on its mathematical model. Once *NyctiDB* has been implemented, its applicability and usefulness are demonstrated through two applications. Application A shows how *NyctiDB* is integrated inside the software architecture of an online educational bioprocess simulator. This implies that *NyctiDB* provides the information for the visualization of different bioprocess behaviours and the modifications of the models in the software. Moreover, the information related to the parameters and conditions of each model is used to support the users' understanding of the process. Additionally, application B illustrates that *NyctiDB* can be used as AI enabler to further the research in this field through open-source and reliable data. This can, in fact, be used as the information source for the AI frameworks when developing, for example, hybrid models or smart expert systems for bioprocesses. Henceforth, this work aims to provide a blueprint on how to collect bioprocess modeling information and connect it to facilitate and empower the Internet-of-Things paradigm and the digitalization of the biomanufacturing industries.

KEYWORDS

non-relational database, bioprocesses, digitalization, modeling, ontology

Abbreviations: AI, Artificial intelligence; ES, Expert system; GMoP, Good modelling practices; IDE, Integrated Development Environment; IoT, Internet of Things; ISO, International Organization for Standardization; JSON, JavaScript Object Notation; NoSQL, No Structured Query Language; ODE, Ordinary differential equation; OOP, Object Oriented Programming; PAT, Process Analytical Technology; QbD, Quality by Design; SBML, Systems Biology Markup Language; SQL, Structured Query Language; UML, Unified Modeling Language; YAML, Ain't Markup Language; XML, Extensible Markup Language.



GRAPHICAL ABSTRACT

1 Introduction

In this period of rapid digital and technological transformation, the biomanufacturing industry faces many challenges, from the acquisition and processing of data Gargalo et al. (2020a) to the generation of process models Narayanan et al. (2020). This digital transformation aims to exploit the use of the internet to link and communicate information. This is commonly known as the Internet-of-Things paradigm. In the IoT, heterogeneous information is integrated into the cloud, and it undergoes a wide exchange of data between different systems. The availability of this data to be linked takes the system a step closer to fully understand the process as well as how to optimize it. Figure 1 shows a conceptual representation of the connection between data and the digitalization of a system.

While data import and processing methods have been widely studied Charaniya et al. (2008) and/or can be used from other disciplines Narayanan et al. (2020), the re-use and storage of information related to process models is a different and more complex topic that has been discussed for many years already. Process models are highly specific and costly to develop, especially regarding the time required for information gathering. Consequently, researchers rely on knowledge databases, literature reviews, and/or computer-aided tools as well as their previous experiences. When information is not available, researchers need to generate it through experimental work. Although a more accurate source, this option is very time and resources consuming and thus not a sustainable strategy to gather all levels and types of information. Therefore, searching for or acquiring the relevant information for and about process modeling can become a challenging task. A step towards solving this issue could be using an online process model database. A database is commonly defined as a collection of data in a computer system. There are several types of databases, and new systems for data collection inside computers are continuously being developed as new digitalization strategies are pushed forward. Hence, this work faces the question: *how should information about bioprocess modeling be stored so that it can be linked and exploited in the digitalization and IoT paradigms?*

This question is tackled here by developing *NyctiDB*, a non-relational database for bioprocesses that connects the information in such a way that the number of possibilities is maximized, i.e., reuse of model structure based on similarities of the bio-kinetic process. *NyctiDB* focuses not only on the kinetic models but also on the process conditions and characteristics for which they were developed. Moreover, the collected information should be easily understandable and open to fellow researchers and industrial partners, as *"If I have seen further it is by standing on the shoulders of Giants"*- Issac Newton; Steinbeck et al. (2003). We believe that *NyctiDB* can provide a structure and platform to help researchers and industries advance on the digitalization of bioprocesses. To do so, *NyctiDB* has been developed as an open-source, accessible, and readable database.

This article is organized as follows. Section 2 presents a comprehensive literature review of the current status of modeling and databases for biotech processes. Section 3 provides a contextual theoretical background about data storage strategies. In Section 4, *NyctiDB* and its functionalities are described, along with illustrative examples of each library. Potential applications are explored in Section 5. Finally, conclusions are presented in Section 6.

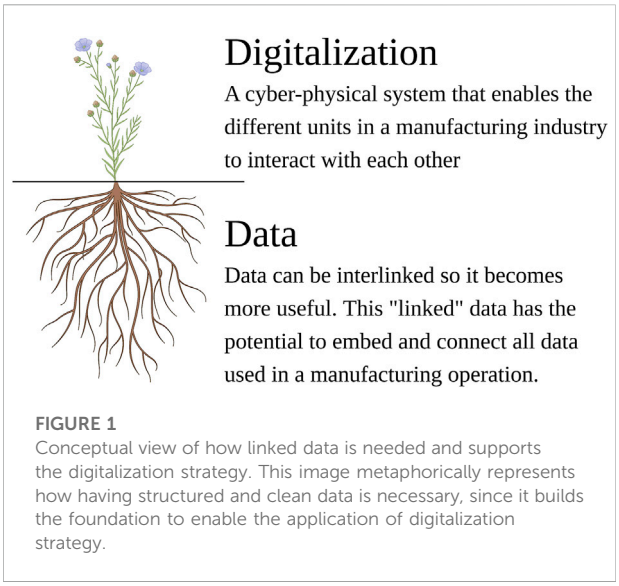


FIGURE 1
Conceptual view of how linked data is needed and supports the digitalization strategy. This image metaphorically represents how having structured and clean data is necessary, since it builds the foundation to enable the application of digitalization strategy.

2 Literature review

2.1 Previous efforts on database development for bioprocesses

Previously developed biological databases have mainly focused on the molecular biology and genetic information of the systems. Some examples of these biological databases are Swiss-Prot Bairoch and Apweiler (2000) and PIR Barker et al. (2000)for protein sequences, or Biofilms Structural Database for the different protein structures involved in biofilm formation, development, and virulence Magalhães et al. (2020), or GenBank Benson et al. (2012) and DDBJ Tateno et al. (2002)for genome sequence information, as well as Kanehisa (1998), BRENDA Schomburg et al. (2002) for enzyme information, or CDK for chemo- and bioinformatics Steinbeck et al. (2003). However, structured compilations of models built with an engineering approach are still scarce. Some compilations of models are embedded in commercial simulators, and therefore, their mathematical model and process conditions are not freely accessible. SuperPro Inc. (2017) or LABSTER ApS. (2018)are examples of such software with a biochemical modelling database without providing free access to the users. On the other hand, there are other bioprocess simulators that contain process models while allowing the inspection of the models. Some examples of these open-source simulators are BioSTEAM Cortes-Peña et al. (2020), PhotoBioLib Perez-Castro et al. (2017) for photobioreactors, and pyFOOMB Hemmerich et al. (2020). Non-etheless, the collection of models available in these software tools is still very limited, and more importantly, they do not provide a framework and structure on how to add new models.

The biggest (bio)model repository is BioModels Malik-Sheriff et al. (2020). BioModels provides an extended number of models and is supported by a curator group. Still, the models do not share a common structure or even a standard file format or programming language. This lack of standardization is not in line with the latest requirements established by the FAIR principles (Findability, Accessibility, Integrateability, Reusability) Wilkinson et al. (2016). These principles work towards improving digital assets at a global level tackling their findability, accessibility, interoperability, and reuse.The lack of standardization and integrated storage of process design conditions and specifications do not facilitate the reuse of the models within the biotech industry and scientific community. Previous efforts have been done within synthetic biology, such as with the development of SBML Caltech (2022). SBML, or Systems Biology Markup Language, is a markup language for communicating and storing computational models of biological processes. Furthermore, model standardization and reuse has been achieved to some extent in the wastewater field with their range of standard Activated Sludge Model (ASM) Henze et al. (2005) or the Benchmark Simulation Model (BSM) Nopens et al. (2009). Meanwhile, it is still essential to develop a bioprocess model's data storage system that considers the industry needs and the FAIR principles.

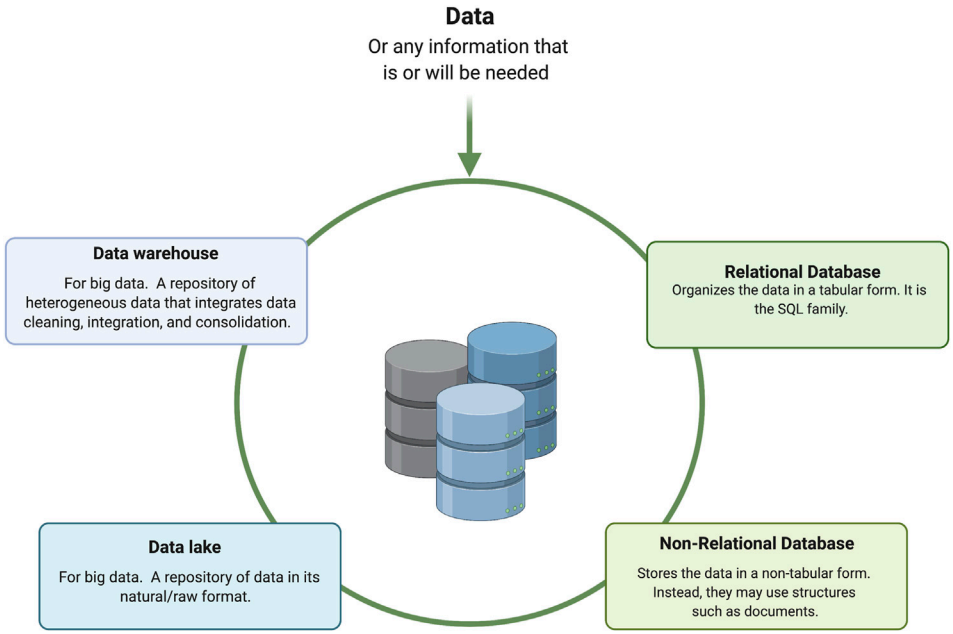


FIGURE 2
An overview of the most common data storage strategies.

2.2 Modeling and simulation of bioprocesses

Process models enclose information through a set of mathematical expressions that enable, for example, the design of equipment to predict the system's behavior and process optimization, among others Gernaey et al. (2010). Conventional bioprocess modelling depends significantly on unstructured mechanistic models (e.g., the Monod, Tessier, and Blackman equations) Gomez et al. (2016). Although mechanistic models yield better process understanding, there are several implied challenges. Some of these challenges are Tsopanoglou and del Val (2021): i) developing mechanistic models requires considerable experimental effort for model validation; ii) it is difficult to automate model assembly; ii) using mechanistic models is resource intensive to use in industry due to high-level expertise required; and, iv) the development of overparametrized models lead to the lack of robustness and universality. However, in recent years, modeling and simulation have shaped important strategies such as the Quality-By-Design (QbD) framework, the Process Analytical Technology (PAT) guidance or model-driven control Sin et al. (2009); Mears et al. (2017). Furthermore, modelling and simulation form the core of the most recent digitalization developments, like digital twins Gargalo et al. (2020a); Narayanan et al. (2020); Udugama et al. (2021). These current efforts support shaping a more sustainable future through the implementation of bioprocesses. This has been included in the Sustainable Development Goals (SDGs) by the United Nations United Nation Development Program (2014). Meanwhile, from an economic perspective, biomanufacturing is enduring a constant market growth due to an augmenting demand for vaccines, drugs, and enzymes 202 (2020), which also push forward strategies for the optimization of processes, such as model-driven methods.The scientific interest is further proven by the continuously increasing number of publications related to the field of mathematical modeling of bioprocesses and microbial growth Udugama et al. (2021), from 10,020 papers published on the topic in 2017 to 18,381 in 2020 Dimensions (2022). However, still, when designing a bioprocess, the critical challenge is usually the availability of suitable models Kroll et al. (2017). Although clearly needed, to the best of our knowledge, there is no record of a comprehensive and structured collection of bioprocess models coupled with the associated model information in the form of an open-source database.

3 Data storage strategies: Background

The creation of a data storage strategy is a complex process that requires numerous and well-thought decisions. Several data storage strategies, as well as data formats, can be selected to design and develop a database. This section reviews the different

TABLE 1 Main differences between relational and non-relational databases.

	Relational	Non-relational
Storage of information	Tabular format. The database is structured through tables composed of columns and rows	Non-tabular format. A possibility can be the use of documents
Application	When accuracy is crucial, and data does not change. For example, financial applications	It supports many different kinds of data and dynamics databases. Cisco, Google or the U.S. Immigration and Customs Enforcement Mongo (2022a) are examples of software used
Preferred application	Small and medium size applications Gyorödi et al. (2015)	Big applications Gyorödi et al. (2015)
Management system	The SQL family: MySQL, PostgreSQL, SQLite Kimelman et al. (2013)	IBM Cloud Database, MongoDB, Amazon Dynamo DB.

available options and the reasoning behind the choices made while developing *NyctiDB*. Furthermore, the use of an ontology is also described along with its benefits and deficiencies.

3.1 Common types of data storage strategies

New data storage strategies are continuously being developed to handle the recent large amounts of data, as well as the increase of variety and velocity in which the data is communicated Chen et al. (2014); Chen et al. (2013); Boiarkina et al. (2018). Some of the most common data storage strategies are presented in Figure 2.

Data lakes and data warehouses Big Data Blog, Oracle (2022)- among others - have been developed to store large amounts of data. Figure 2 introduces the significant differences between data lakes and warehouses. In this work, we focus on relational and non-relational databases since they are very flexible and suited for any user and are more organized for smaller or medium size amounts of data.

It is beneficial to proactively select a database structure that can later adapt to a "big data" strategy and with cloud compatibility.

A database, as previously mentioned, is defined as an organized collection of data typically stored in a computer system. Currently, a database is classified based on how it stores the data, and thus, a database can be relational or non-relational. Relational databases are well established, and they are the traditional choice for database design Kimelman et al. (2013). Nevertheless, non-relational databases, also known as NoSQL, commonly store data in a non-tabular form and are being increasingly used due to their advantages in big data and real-time web applications Paul (2022). Table 1 further illustrates some of the main differences between relational and non-relational databases.

Non-relational databases have been established as the favorite option for IoT, based on their flexibility in terms of their data structure and faster data retrieval compared to relational databases Gyorödi et al. (2015). Based on these characteristics, in this work, a non-relational database was selected as the basis for *NyctiDB* since it seems to be the best choice considering the expected needs of a digitalized biotech industry.

Mongo (2022a) is currently the most popular example of a non-relational database program Mahipal Nehra (2022); Gyorödi et al. (2015). Some examples of software using MongoDB are SEGA, BARCLAYS, or the tax platform for the UK government, among many others Mongo (2022b). MongoDB stores its data records in documents; these documents are subsequently grouped in collections. Documents in a collection do not need to share the same structure or set of fields, however, practically, they do share some similarities (e.g., a similar structure and common fields). Those documents are characterized by being available to store their data in a JSON data format. JSON is a data format with an easily understandable architecture JavaScript (2022) which improves readability. It allows for easy examination by non-programmers due to its self-evident structure. Therefore, it is an appropriate data format to store data inside a flexible, reusable, and expandable database. Furthermore, the JSON data format is also aligned with the FAIR principles (Section 2.1). A simple example of a JSON file is presented in Supplementary Material Section 9.1.

The documentation created for the database, and made available in (https://NyctiDB-sphinx.readthedocs.io/en/latest/), includes an easy tutorial about how to store data in JSON format. Other examples of data formats comparable to JSON (or classified as "human-readable data") are YAML Evans (2022), SBML Caltech (2022), and XMLConsortium World Wide Web (2022).Using a non-relational database implies a sacrifice: the explicit structure provided by relational databases is lost. However, this feature can be obtained by using an ontology that can support the non-relational database. This limits complexity and organizes the data based on the enclosed information and knowledge Wikipedia (2022b). Therefore, in this work, we propose using an ontology to provide additional data structure and facilitate future database expansion.

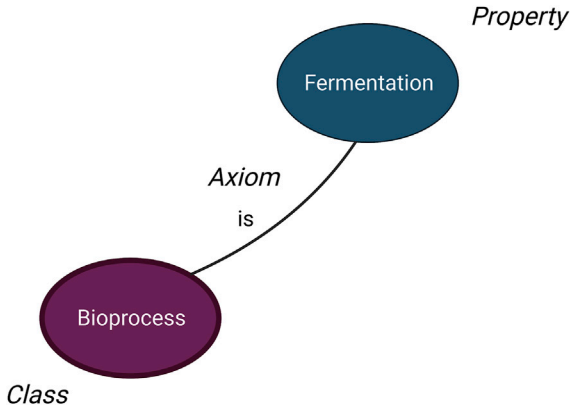


FIGURE 3
Schematic example of the different categories that compose an ontology. This figure shows that a fermentation (property) is (axiom) a bioprocess (class).

3.2 Ontology: A structural support for a non-relational database

What the saying *"it is a small world"* truly means is that *"it is a world full of linked data"*. Linked data is created through interlinked information by semantic queries, and due to its interconnected nature, the data increases its usefulness Wikipedia (2022a). The most common way to represent linked datasets is through ontologies. An ontology has been traditionally defined as a system of categories that aim to provide a representation of the world Guarino et al. (2009) based on specifications of a relational vocabulary. Practically, an ontology-based management system allows for the representation of linked datasets by enabling the conceptualization of explicit specifications. Conceptualization is defined as an abstract, simplified view of the world that we wish to represent for some specific purpose Gruber (1993). Therefore, the ontology development process transforms the initial need (i.e., to create reusable and shareable knowledge regarding bioprocesses) into a final product: the evaluated, documented ontology, codified in a formal language. This work uses the ontology life cycle developed by Fernandez-Lopez et al. Fernández-López et al. (1997). Hence, the ontology is accomplished by applying the following stages/steps: i) specification; ii) knowledge acquisition; iii) conceptualization; iv) integration; v) implementation; vi) evaluation; and, vii) documentation. This is achieved by schema mapping, entity resolution, and data fusion Devanand et al. (2020); Mandreoli and Montangero (2019). An ontology consists of classes, properties, individuals, and axioms Gruber (1993). Classes describe the categorization of individuals and their properties. Axioms are used to express basic statements in the ontology by reusing classes and properties. This is schematically represented in Figure 3. In Figure 3, the class of "bioprocess" contains a property "fermentation" connected through the axiom "is". Furthermore, the class "bioprocess" and the axiom "is" can contain other properties such as "bioremediation", "biological nutrient removal in wastewater", or "enzymatic hydrolysis".

Overall, ontologies can facilitate data reuse, integration, and maintenance as well as provide less ambiguous interpretation and reasoning Poveda-Villalón, 2020 and (Ontology Engineering Group–Universidad Politécnica de Madrid) 2). Based on the numerous benefits, several ontologies have been developed for life sciences, such as for biomedical terminology Beisswanger et al. (2008), and molecular biology Blake (2004); Aranguren et al. (2008). For example, OntoCAPE Morbach et al. (2009) is a re-usable ontology created for computer-aided chemical process engineering and has been successfully used in model generation, knowledge management, and data integration in this area. Another example is the ontology developed for searching and inferences of process monitoring and analysis tools Singh et al. (2010). In addition, the use of ontologies has now extended to cross-domain areas of engineering, such as, for example, the study of interactions in a biodiesel plant Devanand et al. (2020). However, most of these ontologies face issues [e.g., how to appropriately use them Soldatova and King (2005)]; thus, international standards, like ISO 19150-4:2019 Information/Geomatics (2019), have been pushed forward.When compared to databases, ontology engineering still has to deal with the same kind of issues that databases faced several years ago, such as mapping searches, schema alignment, matching conflicts, etc. Martinez-Cruz et al. (2012). Recent studies have shown that combining ontologies with databases is a promising approach to assist non-relational databases for structuring and query relaxation Gundla and Chen (2016). In other words, using an ontology as a foundation for database development can bring several benefits, which are explored in this work.As established in Section 1 and Section 2, specifically in the area of bioprocess modeling, there is a scientific and industrial need for an online, flexible, reusable, shareable, and expandable system to collect and access data.As previously mentioned, in this work, we propose a non-relational database–*NyctiDB*–coupled with an ontology. *NyctiDB* is characterized by its flexibility and its online nature, which assist the introduction of new information and its integration inside the IoT paradigm.

Based on the information presented here, the approach selected is the one that combines a non-relational database implemented in MongoDB with an ontology. Therefore, the next step is the further collection and structuring of the information in a comprehensive library of bioprocess modeling knowledge.

4 NyctiDB: A bioprocess modeling non-relational database

As previously stated, the main goal of this work is to create a computer-aided tool for the reuse and sharing of bioprocess modeling knowledge. Hence, a non-relational database for

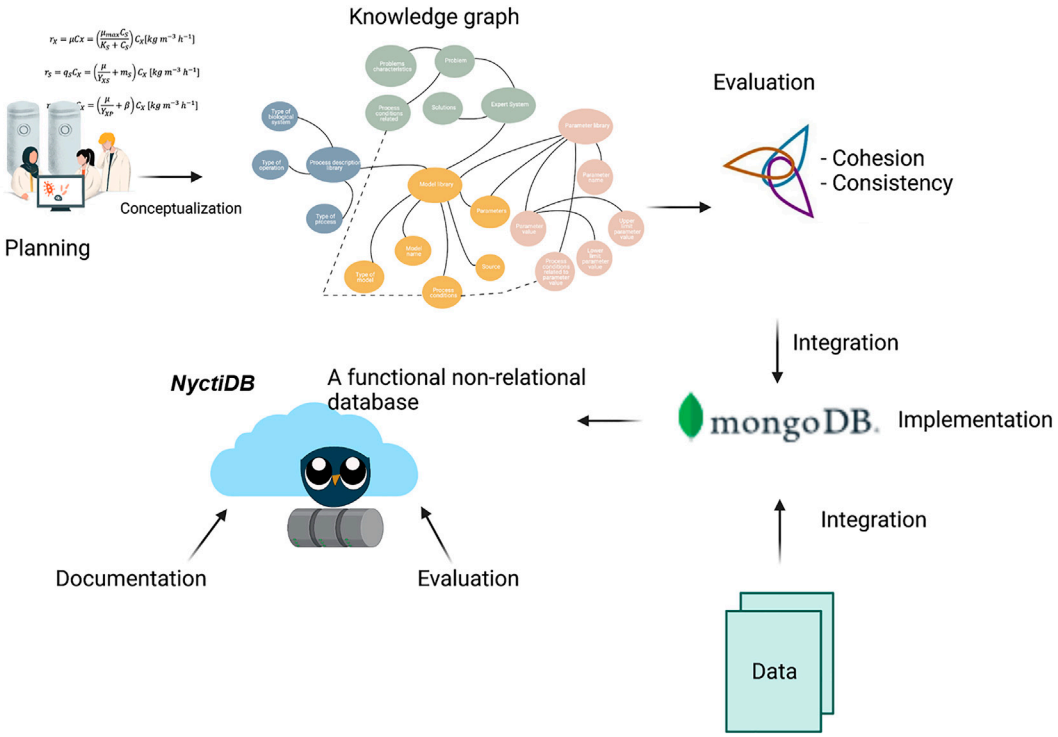


FIGURE 4
NyctiDB's workflow: the link between the ontology and the non-relational database. More information about the knowledge graph can be seen in Figure 5.

bioprocesses, supported by an ontology, has been developed and named *NyctiDB*. *NyctiDB* provides a frame for collecting and storing bioprocesses' information inside an online platform which is currently in the proof of concept stage (BioVL). Furthermore, the online nature of the proposed system facilitates its implementation in any computational tool. Thus, *NyctiDB* can be: i) easily integrated and become a valuable tool in the digitalization of the biotech industry, as well as b) a future high-value component of computer-aided bioprocess simulators. The developed approach is composed of two connected parts: an ontology and a non-relational database. The corresponding workflow is presented in Figure 4.

Figure 4 illustrates the different activities which are part of the workflow. The first activity is conceptualization of the domain knowledge of the specific bioprocess modeling task. In this step, one needs to describe the problem and its solution, which is to be integrated into the ontology. To do so, all useful and potentially usable domain knowledge is identified. Then, this is categorized and represented through a knowledge graph (Figure 5). The knowledge graph is built on four classes. These classes are:

1) A library of mathematical bioprocess models;
2) A library with theoretical knowledge related to the bioprocess description;
3) A library of common parameters in bioprocess models; and,
4) A problem and solutions library, also known as expert system.

This ontology structure aims to provide a complete overview of the bioprocess' characteristics based on its process model. However, the ontology is designed in such a way that, when found, new and relevant information is to be included. Thus, only partial completeness of the ontology can be proved, which corresponds to the inclusion of the terms, the stopover problem and the term's level of granularity Fernández-López et al. (1997). Apart from completeness, the ontology should guarantee its conciseness and consistency. Conciseness ensures that there are no duplicated and/or irrelevant terms, whereas consistency pertains to all terms and corresponding meaning being sound in the biotech modeling domain. In this work, Protégé Stanford University (2022) is the ontology editor used for the development and evaluation of the ontology. A two-step evaluation is recommended, with a first knowledge graph validation followed by a second validation once *NyctiDB* is populated with data (see Figure 4). In the current study, the knowledge graph has been tested and validated using the HermiT reasoner included in Protégé; the ontology was found to be consistent and coherent. The validated knowledge graph is used to guide the integration, implementation, and later maintenance of the data that is performed at the non-relational database level. Lastly,

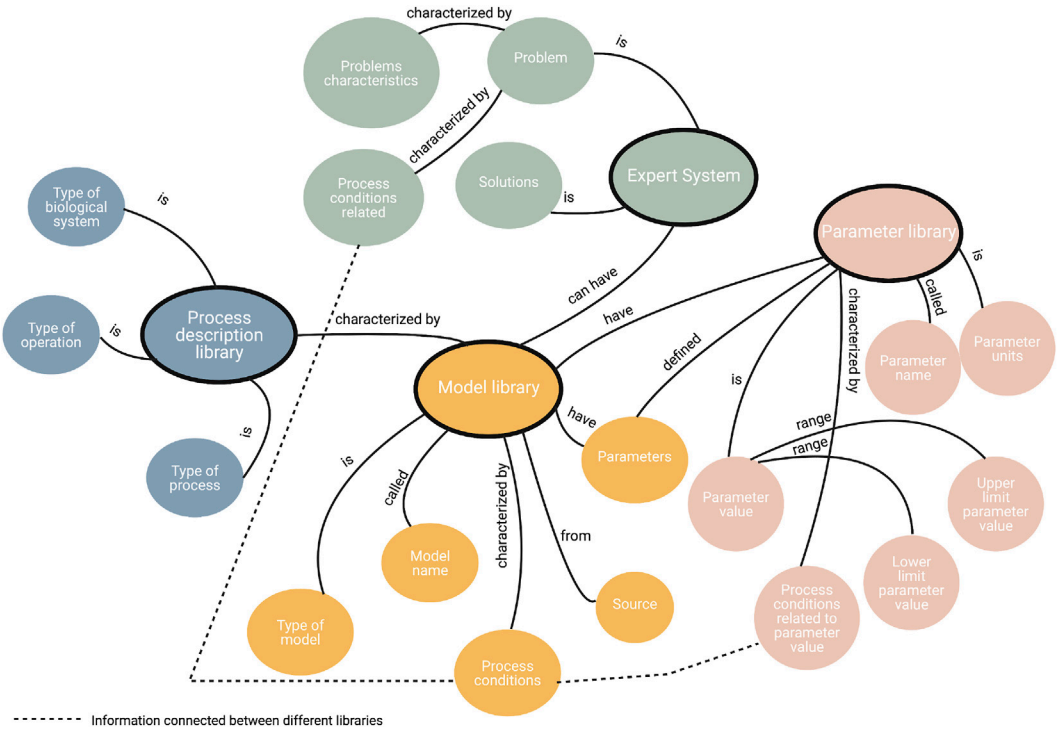


FIGURE 5
NyctiDB's UML-linked knowledge graph. Although included in this graph, the axioms do not have great relevance during the implementation of the ontology inside the database. The axioms are depicted for the sake of clarity and interpretability of the functional dependencies.

the structure is made available online as a non-relational MongoDB database based on a set of documents corresponding to the different classes, properties, and axioms.

In the next subsection, the integration and implementation of *NyctiDB* is illustrated through a case study on aerobic growth of *Corynebacterium glutamicum* under product inhibition conditions Khan et al. (2005). Other examples are available in *NyctiDB*'s GitHub repository as well as on its documentation webpage (https://github.com/BioVL/Pymongo-Minerva and https://minerva-sphinx.readthedocs.io/en/latest/)[1].

4.1 Library of mathematical bioprocess models

The mathematical models' library is the core of the database. As shown in Figure 5, this library contains information to fully identify the model. Thus, it contains the name, type of model (e.g., a Monod equation-based model with product inhibition), the source of the model, process conditions for which the model is developed, and its parameters. To highlight the usefulness of *NyctiDB*, an example of an implemented model and how it is embedded in the database is demonstrated. This example describes the aerobic growth of *Corynebacterium glutamicum* and can be found in Supplementary Material Section 9.3. Noteworthy is that the use of a non-relational database provides the flexibility to add or delete some information into a specific model without the need to restructure the database.

TABLE 2 Matrix presentation of a Monod-Herbert anaerobic biomass growth model. With variables: substrate concentration, C_S; product concentration, C_P; biomass concentration, X. The parameters in the matrix are: biomass yield on substrate, Y_{SX}; biomass yield on product, Y_{PX}; decay coefficient of biomass, k_d; maximum growth rate of biomass, μ_{max}; affinity coefficient for substrate, K_S. Based on Gernaey et al. (2010).

Components, n	C_1	C_2	C_3	Rates, ρ_1
Symbols	C_S	C_P	X	—
Units	$g \cdot m^{-3}$	$g \cdot m^{-3}$	$g \cdot m^{-3}$	$g \cdot m^{-3} \cdot h^{-1}$
Process, m	—	—	—	—
Growth	$-1/Y_{SX}$	$1/Y_{PX}$	1	$\mu_{max} \cdot \frac{C_S}{C_S+K_S} \cdot X$
Decay	0	0	-1	$k_d \cdot X$

1 These links will shortly be updated to https://github.com/BioVL/Pymongo-NyctiDB and https://NyctiDB-sphinx.readthedocs.io/en/latest/ in order to reflect the name of the database, respectively.

This library of mathematical process models contains first principles mechanistic models implemented in Python that can be freely and easily (re)used. Those models are implemented following the good modeling practices (GMoP). GMoP aims to provide the tools to (bravely) face the complexity of building mechanistic models for bioprocesses Sin et al. (2009). There are many GMoP workflows, such as the ones proposed in Mears et al. (2017); Kroll et al. (2017); Kell and Sonnleitner (1995). These previous works have focused on taking a deeper look into phenomena that occur in large-scale fermentation processes. Whereas in this work, the primary target is to provide a straightforward implementation template that other researchers can use to implement their models inside a computer-aided tool, following conservation principles and using matrix modeling notation.

4.1.1 GMoP: Matrix modeling notation

The matrix notation is a well-established system to mathematically describe complex models Henze et al. (2005); Noorman et al. (1991). It uses the linear relationships among net conversion rates to describe the conservation relationships inside a system. This method has been extensively applied to bioprocess models, for example to express the growth of *Streptomyces coelicolor* Sin et al. (2008) and *Saccharomyces cerevisiae* Lencastre Fernandes et al. (2012).

The description of a system through this methodology starts by defining the number of components n and the number of processes m in the system. This is the foundation for the stoichiometric matrix, S, the process rate vector, ρ, and finally, the component conversion rate vector r.

$$r_{nx1} = S_{mxn} \cdot \rho_{mx1} \tag{1}$$

The overall component conversion rate vector is then coupled to a general mass balance equation. Table 2 presents a simple matrix notation example of a Monod-Herbert anaerobic biomass growth model.

The matrix notation presented in Table 2 is mathematically translated into Eq. 2.

$$\begin{bmatrix} r_S \\ r_P \\ r_X \end{bmatrix} = \begin{bmatrix} -1/Y_{SX} & 0 \\ 1/Y_{PX} & 0 \\ 1 & -1 \end{bmatrix} \cdot \begin{bmatrix} \mu_{max} \cdot \dfrac{C_S}{C_S + K_S} \cdot X \\ k_d \cdot X \end{bmatrix} \tag{2}$$

The overall component conversion rate vector (Eq. 1) is then coupled to a mass balance which results in a set of ordinary differential equations. This is solved as a set of ordinary differential equations (ODEs) that contain information related to the behavior of the process. In this work, Python is used as the software tool to solve the ODEs. By using an open source programming language, we create a reproducible, open access, and easy-to-use model implementation template. An implementation example in the form of a *fill up* template can be found in Supplementary Material Section 9.2. In addition, the complete script regarding the modeling of the aerobic growth of *Corynebacterium glutamicum* under product inhibitionKhan et al. (2005) can be found in the supporting material.

The model implementation template benefits from using Python's object-oriented properties expressed as *classes*. Object-oriented programming (OOP) is characterized by its encapsulation, abstraction, inheritance, and polymorphism properties. To sum up, this type of programming allows to couple systems so that they can work as modules and be easily expandable without affecting the whole system. Some of its advantages are as follows Educative, Inc. (2022):

- it allows to embed model complexity into reproducible and simple structures;
- OOP can be used across programs or functions, such as in the combination of neural networks and in hybrid modeling;
- it is easy to debug as all the information is enclosed inside said program;
- it protects and secures information through encapsulation;
- it allows class-specific behavior through its polymorphism.

These properties have made OOP the most popular coding structure for programmers and is widely used for software development. Furthermore, it is increasingly applied when developing hybrid models for the biotech industry.

4.2 Process description library

NyctiDB, as previously mentioned, enables and facilitates the collection and storage of relevant process information for which the model has been created, such as process description, process conditions, and parameters. Knowledge about the process is fundamental when using a specific model; for example, when a model is selected, researchers will need to compare their process conditions to know if: the model can directly be used, cannot be used, or needs to be re-calibrated to the new process conditions. Therefore, *NyctiDB* includes this information in the form of a library. It contains: i) the type of biological system modeled (e.g., enzymes, mixed culture, yeast); ii) the type of process (e.g., aerobic or anaerobic conditions); and iii) the type of operation mode in which the data has been obtained (e.g., continuous, fed-batch, or batch). An example contained in this library in *NyctiDB* is presented in Supplementary Material Section 9.5.

One of *NyctiDB*'s main benefits, and the libraries within, is that it facilitates the re-use of models in the biotech community. The inclusion of process description information can have a positive impact related to the time and resources that it saves in process design and operation.

4.3 Library of process parameters

First-principles process models commonly consist of: i) a set of parameters; ii) a set of state variables (e.g., concentrations and

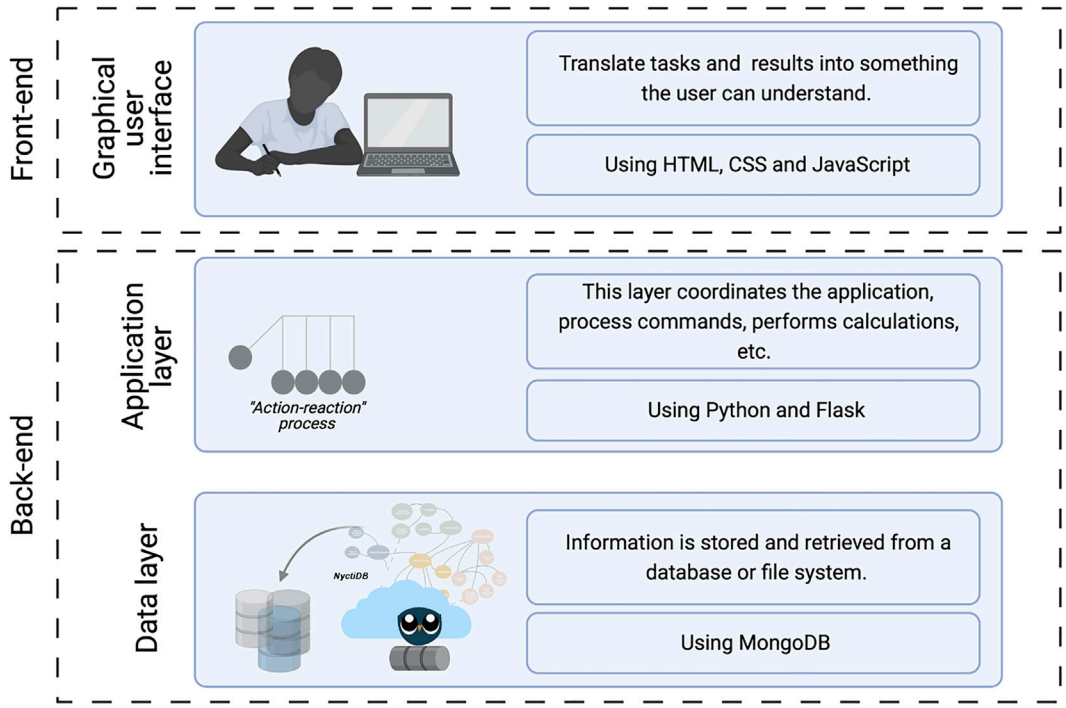


FIGURE 6
Three-tier software architecture of BioVL.

volume); and iii) a set of mathematical equations that represent the process rates. The set of parameters depends on the process for which data has been collected; therefore, the parameters may need to be re-calibrated for the new process conditions. This step is known as parameter estimation, and it is usually one of the most complicated and time-consuming steps in the development of a mo del Sin et al. (2009). Since it is computationally expensive and time demanding, a good starting point for the parameter estimation can benefit the overall process model development and application.

In summary, the library of process parameters integrated into *NyctiDB* includes: i) a collection of common model parameters, ii) the process conditions under which the model was collected, iii) the conditions that affect the parameters as well as upper and lower limits, and finally, iv) the settings that will determine or define these parameters. An example can be found in Supplementary Material Section 9.6.

4.4 The expert system library

The last library comprises possible operational problems based on the type of bioprocess and its characteristics. It includes a description of the problem, how it could or would affect the process model parameters, and the process conditions associated with the presence of such problems. Moreover, within this library, the problems are complemented with a list of possible solutions. An example can be found in the supporting material provided.

5 Application

Databases are valuable digital resources that can be used independently for different tasks. Therefore, in this section, two specific cases are presented of the application of *NyctiDB*. The first case, application A, is the integration of *NyctiDB* as part of the software architecture of an educational bioprocess simulator. The second example aims to explain the potential use of the stored information by using this non-relational database (*NyctiDB*) as an input for an artificial intelligence algorithm.

5.1 Application A: Software integration

A database is an intrinsic part of a well-designed simulator that divides its functions into layers or tiers 201 (2019). The process simulator can be expandable and customizable through its database without affecting the previously established functionalities. Of note is that the development of this tailored database, and other databases developed following the same strategy, is a clear step forward in the research community that focuses on developing a storage

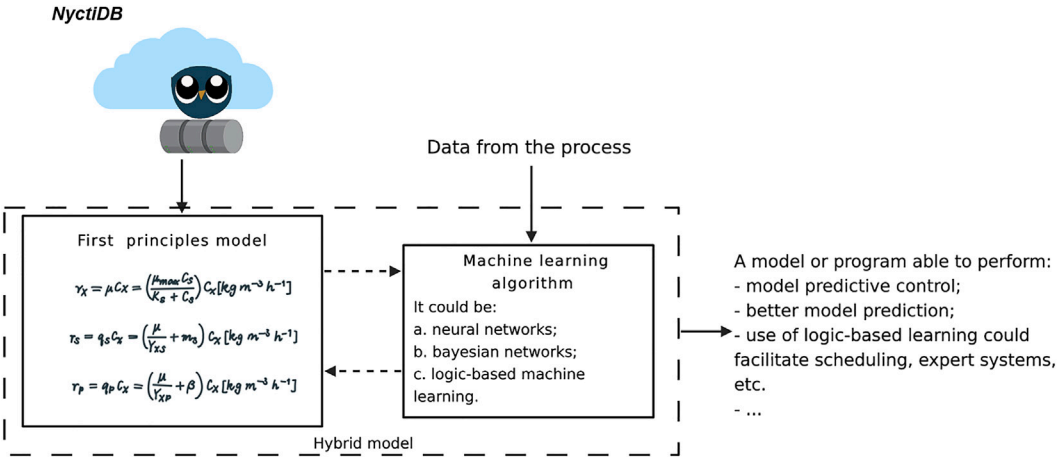


FIGURE 7
NyctiDB as first principles models provider for hybrid modeling. Based on Nielsen et al. (2020).

system, not only to collect mechanistic models for biochemical processes (the goal of this work) but also other models. Since knowledge can be shared and distributed, it saves resources. In this work, *NyctiDB* is integrated as the data layer (or data tier) inside the software architecture of BioVL (see Figure 6). This software architecture allows the independent use of the layers. As a consequence, the data layer used (*NyctiDB*) could be used for other software platforms. In this case study, *NyctiDB* is integrated inside a software called **Bio**process **V**irtual **L**aboratory (BioVL–www.biovl.com), previously developed by the authors de las Heras et al. (2021). Figure 6 shows the BioVL's software architecture.

BioVL has been designed as an educational bioprocess software for undergraduate and graduate students. It focuses on the modeling part of the discipline through:

1) The design of a learning goal based on Bloom's taxonomy Krathwohl (2002) for the use, reuse, and explanation of bioprocess models; and.
2) A learning design that combines game elements and an agile microlearning strategy. This involves an IDE for Python and a small exercise module (e.g., to practice how to implement a Monod-based equation). Additionally, there are theoretical and practical questions about selecting the model and which units correspond to a certain parameter.

BioVL, as documented in de las Heras et al. (2021), involves its future users through co-participatory design (e.g., chemical and biochemical engineering students). Currently, its functionalities include i) a chatbot to encourage collaborative learning de Las Heras et al. (2020); ii) learning content about the formulation of bioprocess models; iii) a simulator in which operational problems take place, and the student must propose a solution; and, iv) a library of mechanistic models. This can be explored in its prototype platform at www.biovl.com. Furthermore, we hope to encourage students to learn about GMoP implementation and become proactive contributors to *NyctiDB* by actively applying the proposed template (Section 4.1.1).

5.2 Application B: *NyctiDB* as AI enabler

An AI model built without discipline has a high probability of failing. Therefore, a solid frame on which AI operates is as important to provide as quality input data. Commonly, first-principles models have been integrated inside deep learning frameworks or other machine learning strategies, enhancing the fitting regression of process variables. Typically known as hybrid models, these systems have achieved promising results in improving models, implementing model predictive control strategies or dynamic optimization, etc. Some examples of such systems in bioprocesses can be found in Eikens et al. (1999); Gao et al. (2010); Nazemzadeh et al. (2020); Nielsen et al. (2020); Ning and You (2019); Cabaneros Lopez (2020). However, even if the combination of mechanistic models (also known as first-principle or deterministic models) and machine learning have demonstrated their benefits, the integration of hybrid modeling into the bio-manufacturing facilities is still more an exception than a rule. *NyctiDB*, with its open source and online nature as well as its OOP model structure, can be an exceptional tool to facilitate hybrid modeling implementation inside the different unit operations (Figure 7).

Nevertheless, deep learning cannot be straightforwardly applied when a) there is not enough data and/or b) when the problems involve multiple numbers of objects, and their

relationship is intrinsically interconnected. In these cases, other disciplines under the AI umbrella, such as logic-based machine learning, might be more suitable Utgoff et al. (2011); Raedt (2010). Logic-based machine learning aims to compute intelligible logical programs based upon given knowledge, being characterized by the use of first-order logic to represent hypotheses Muggleton and De Raedt (1994); Law et al. (2015).

Although it has proven its value in areas such as bio- and chemo-informatics Agrafiotis et al. (2007); Ando et al. (2006); Begam and Kumar (2012), natural language processing Muggleton (1993), or web mining Lisi (2007), inductive logic programming is still a relatively unexplored territory.

Therefore, the information in *NyctiDB*, as well as the logic connections embedded in its ontology, can be used as background knowledge for inductive-logic programming. An example could be the use of the problem and solutions library for the development of an expert system (ES). An ES is an interactive and reliable computer-based decision-making system that uses both facts and heuristics to solve complex decision-making problems. Hence, ES is generally composed of "if . . . , then . . . " statements obtained from process experts Lennox et al. (2002). Consequently, the combination of the information inside the libraries in *NyctiDB* can be used as the support system on which to build an automatized ES through AI. Such a system could be beneficial for the ES development, a traditionally very costly and time-consuming process. In summary, *NyctiDB* can be used to facilitate applications within the fields of both numerical and symbolic AI.

6 Limitations and future perspectives

Currently, *NyctiDB* includes a selection of mechanistic models as a demonstration. However, this database is under continuous development, where new data, models, and relationships are being included and integrated. These capabilities will enable researchers to access real-world data to train both mechanistic and data-driven models on an online platform. This possibility will empower research across fields and the use of this database within digitalization efforts, in which having a cloud-based data storage system, as presented in this work, is fundamental. For example, see Park et al. (2021) and Gargalo et al. (2020b) for details concerning the flow of information under the digitalization paradigm. Moreover, the data and models included in the database will also have a peer-review cycle allowing other users to edit the information currently available in the database and, at the same time, add new information. Noteworthy is that the authors will review this information. Hence, the goal behind this approach is twofold: i) it ensures the growth of the data stored in the database, which grows exponentially with the number of users, and ii) it guarantees the quality of the stored data through a peer-review process.

7 Concluding remarks

This work aims to investigate and propose a frame for collecting and storing information related to process models, specifically in the area of bioprocesses. This is particularly useful for developing an online repository compatible with digitalized biomanufacturing. To this end, *NyctiDB*, a non-relational database supported by an ontology, has been developed (https://minerva-sphinx.readthedocs.io/en/latest/)[2]. The data within *NyctiDB* is subdivided into libraries, and they are: i) a process models library; ii) a parameters library; iii) a process description library, and iv) an expert system library. These libraries can support the implementation of good modeling practices, potentially decrease the resources used on model development and contribute to a better understanding of bioprocesses and their models.*NyctiDB* is a digital asset applicable to different projects and functionalities. Its usefulness has been exemplified and detailed through two examples. In application A, the database is integrated as the data layer of a three-tier software architecture of an education bioprocess simulator (BioVL). Application B explores the use of the information collected in the database for supporting machine learning algorithms. It is furthermore explained how the relationships inside the ontology could be used to apply logic-based machine learning. Although full of potential, logic-machine learning (and inductive logic programming) is, to some extent, an unexplored AI branch in the biotech area.To conclude, we believe that *NyctiDB* can be an important tool assisting and enabling the biotech industry for an active transition towards implementing the Internet of Things and, consequently, empowering the digitalization of the bio-manufacturing industry.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: NyctiDB GitHub repository as well as on its documentation webpage https://github.com/BioVL/Pymongo-Minerva[3] and https://minerva-sphinx.readthedocs.io/en/latest/. Note that these links will shortly be updated to https://github.com/BioVL/Pymongo-NyctiDB and https://NyctiDB-

2 This link will shortly be changed to https://nyctidb-sphinx.readthedocs.io/en/latest/ in order to reflect the name of the database.

3 This link will shortly be changed to https://github.com/BioVL/NyctiDB in order to reflect the name of the database.

sphinx.readthedocs.io/en/latest/ in order to reflect the name of the database.

Author contributions

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

Acknowledgments

The authors acknowledge the Novo Nordisk Foundation for the financial support in the frame of the Accelerated Innovation in Manufacturing Biologics (AIMBio) project (grant number NNF19SA0035474).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Agrafiotis, D. K., Bandyopadhyay, D., Wegner, J. K., and Van Vlijmen, H. (2007). Recent advances in chemoinformatics. *J. Chem. Inf. Model.* 47, 1279–1293. doi:10.1021/ci700059g

Ando, H. Y., Dehaspe, L., Luyten, W., Van Craenenbroeck, E., Vandecasteele, H., and Van Meervelt, L. (2006). Discovering H-bonding rules in crystals with inductive logic programming. *Mol. Pharm.* 3, 665–674. doi:10.1021/mp060034z

Anonymous (2020). Global bio-manufacturing market 2020-2025: Cell line engineering, disposable manufacturing Technology, perfusion culture, in-silico modelling, modular factories gaining momentum. *Plant Autom. Tech.* Available at: https://www.plantautomation-technology.com/pressreleases/global-bio-manufacturing-market-2020-2025-cell-line-engineering-disposable-manufacturing-technology-perfusion-culture-in-silico-modelling-modular-factories-gaining-momentum

Anonymous (2019). Web application architecture: How the web works. *AltexSoft*. Available at: https://www.altexsoft.com/blog/engineering/web-application-architecture-how-the-web-works/

ApS, L. (2018). *Labster*.

Aranguren, M. E., Antezana, E., Kuiper, M., and Stevens, R. (2008). Ontology design patterns for bio-ontologies: A case study on the cell cycle ontology. *BMC Bioinforma.* 9, 1–13.

Bairoch, A., and Apweiler, R. (2000). The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000. *Nucleic acids Res.* 28, 45–48. doi:10.1093/nar/28.1.45

Barker, W. C., Garavelli, J. S., Huang, H., McGarvey, P. B., Orcutt, B. C., Srinivasarao, G. Y., et al. (2000). The protein information resource (PIR). *Nucleic acids Res.* 28, 41–44. doi:10.1093/nar/28.1.41

Begam, B. F., and Kumar, J. S. (2012). A study on cheminformatics and its applications on modern drug discovery. *Procedia Eng.* 38, 1264–1275. doi:10.1016/j.proeng.2012.06.156

Beisswanger, E., Schulz, S., Stenzhorn, H., and Hahn, U. (2008). BioTop: An upper domain ontology for the life sciences. *Appl. Ontol.* 3, 205–212. doi:10.3233/ao-2008-0057

Benson, D. A., Cavanaugh, M., Clark, K., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J., et al. (2012). GenBank. *Nucleic acids Res.* 41, D36–D42. doi:10.1093/nar/gks1195

Big data Blog Oracle (2022). Data lake, data warehouse and database. . .what's the difference? Available at: https://blogs.oracle.com/bigdata/data-lake-database-data-warehouse-difference.

Blake, J. (2004). Bio-ontologies—Fast and furious. *Nat. Biotechnol.* 22, 773–774. doi:10.1038/nbt0604-773

Boiarkina, I., Depree, N., Prince-Pike, A., Yu, W., Wilson, D. I., and Young, B. R. (2018). *Using Big Data in Industrial Milk Powder Process Systems*, 44. Elsevier Masson SAS. doi:10.1016/B978-0-444-64241-7.50377-3

Cabaneros Lopez, P. (2020). *Towards industry 4.0 in the bioprocessing industries: 'Real-time' monitoring and control of lignocellulosic ethanol fermentations*. Denmark: Technical University of Denmark.

Caltech (2022). The systems biology markup language. Available at: http://sbml.org/News.

Charaniya, S., Hu, W. S., and Karypis, G. (2008). Mining bioprocess data: Opportunities and challenges. *Trends Biotechnol.* 26, 690–699. doi:10.1016/j.tibtech.2008.09.003

Chen, J., Chen, Y., Du, X., Li, C., Lu, J., Zhao, S., et al. (2013). Big data challenge: A data management perspective. *Front. Comput. Sci.* 7, 157–164. doi:10.1007/s11704-013-3903-7

Chen, M., Mao, S., and Liu, Y. (2014). Big data: A survey. *Mob. Netw. Appl.* 19, 171–209. doi:10.1007/s11036-013-0489-0

Consortium World Wide Web (2022). Extensible markup language (xml). Available at: https://www.w3.org/XML/.

Cortes-Peña, Y., Kumar, D., Singh, V., and Guest, J. S. (2020). BioSTEAM: A fast and flexible platform for the design, simulation, and techno-economic analysis of biorefineries under uncertainty. *ACS Sustain. Chem. Eng.* 8, 3302–3310. doi:10.1021/acssuschemeng.9b07040

[Dataset] Mahipal Nehra (2022). Top 10 nosql databases in 2022- kernel description. Available at: https://www.decipherzone.com/blog-detail/nosql-databases.

de las Heras, S. C., Gargalo, C. L., Weitze, C. L., Mansouri, S. S., Gernaey, K. V., and Krühne, U. (2021). A framework for the development of Pedagogical Process Simulators (P2Si) using explanatory models and gamification. *Comput. Chem. Eng.* 1, 107350. doi:10.1016/j.compchemeng.2021.107350

de Las Heras, S. C., Jones, M. N., Gernaey, K. V., Kruhne, U., and Mansouri, S. S. (2020). An E-learning bot for bioprocess systems engineering. *Comput. Aided Chem. Eng.* 48, 2023–2028.

Devanand, A., Karmakar, G., Krdzavac, N., Rigo-Mariani, R., Eddy, Y. S. F., Karimi, I. A., et al. (2020). OntoPowSys: A power system ontology for cross domain interactions in an eco industrial park. *Energy AI* 1, 100008. doi:10.1016/j.egyai.2020.100008

Dimensions (2022). Dimensions: Virtual laboratory education. Available at: https://app.dimensions.ai/discover/publication?search_text=Virtuallaboratoryeducation&search_type=kws&search_field=full_search.

Educative, Inc. (2022). What is object oriented programming? Oop explained in depth. Available at: https://www.educative.io/blog/object-oriented-programming.

Eikens, B., Karim, M. N., and Simon, L. (1999). Neural networks and first principle models for bioprocesses. *IFAC Proc. Vol.* 32, 6974–6979. doi:10.1016/s1474-6670(17)57190-6

Evans, Clark (2022). Yaml ain't markup language. Available at: https://yaml.org.

Fernández-López, M., Gómez-Pérez, A., and Juristo, N. (1997). *Methontology: From ontological art towards ontological engineering*.

Gao, Y., Kipling, K., Glassey, J., Willis, M., Montague, G., Zhou, Y., et al. (2010). Application of agent-based system for bioprocess description and process improvement. *Biotechnol. Prog.* 26, 706–716. doi:10.1002/btpr.361

Gargalo, C. L., Heras, S. C. d. l., Jones, M. N., Udugama, I., Mansouri, S. S., Krühne, U., et al. (2020a). "Towards the development of digital twins for the bio-manufacturing industry," in *Digital twins* (Cham: Springer), 1–34.

Gargalo, C. L., Heras, S. C. d. l., Jones, M. N., Udugama, I., Mansouri, S. S., Krühne, U., et al. (2020b). *Towards the development of digital twins for the bio-manufacturing industry*.

Gernaey, K. V., Lantz, A. E., Tufvesson, P., Woodley, J. M., and Sin, G. (2010). Application of mechanistic models to fermentation and biocatalysis for next-generation processes. *Trends Biotechnol.* 28, 346–354. doi:10.1016/j.tibtech.2010.03.006

Gomez, J. A., Höffner, K., and Barton, P. I. (2016). Mathematical modeling of a raceway pond system for biofuels production. *Comput. Aided Chem. Eng.* 38, 2355–2360. doi:10.1016/B978-0-444-63428-3.50397-0

Gruber, T. R. (1993). A translation approach to portable ontology specifications. *Knowl. Acquis.* 5, 199–220. doi:10.1006/knac.1993.1008

Guarino, N., Oberle, D., and Staab, S. (2009). "What is an ontology?," in *Handbook on ontologies* (Cham: Springer), 1–17.

Gundla, N. K., and Chen, Z. (2016). Creating NoSQL biological databases with ontologies for query relaxation. *Procedia Comput. Sci.* 91, 460–469. doi:10.1016/j.procs.2016.07.120

Gyorödi, C., Gyorödi, R., and Sotoc, R. (2015). A comparative study of relational and non-relational database models in a web-based application. *ijacsa.* 6, 78–83. doi:10.14569/ijacsa.2015.061111

Hemmerich, J., Tenhaef, N., Wiechert, W., and Noack, S. (2020). *pyFOOMB: Python framework for object oriented modelling of bioprocesses*.

Henze, M., Gujer, W., Mino, T., and van Loosdrecht, M. C. M. (2005). *Activated sludge models ASM1, ASM2, ASM2d and ASM3* 9, 121. IWA publishing.

Information Geomatics (2019). *Information geomatics*. ISO 19150-4:2019.

JavaScript (2022). *Javascript object notation (json)*. Available at: https://www.json.org/json-en.html.

Kanehisa, M. (1998). Databases of biological information. *Trends Biotechnol.* 16, 24–26. doi:10.1016/S0167-7799(98)00133-4

Kell, D. B., and Sonnleitner, B. (1995). Gmp - good modelling practice: An essential component of good manufacturing practice. *Trends Biotechnol.* 13, 481–492. doi:10.1016/S0167-7799(00)89006-X

Khan, N. S., Mishra, I. M., Singh, R. P., and Prasad, B. (2005). Modeling the growth of Corynebacterium glutamicum under product inhibition in L-glutamic acid fermentation. *Biochem. Eng. J.* 25, 173–178. doi:10.1016/j.bej.2005.01.025

Kimelman, D., Perez, M., Kimelman, D., and Perez, M. (2013). Domino.Research.Ibm.Com. Available at: http://domino.research.ibm.com/library/cyberdig.nsf/papers/4AE4B674EEF2BE11 85257BFA0057FBB0/$File/rc25412.pdf. Domino.Research.Ibm.Com 25412/papers/4AE4B674EEF2BE1185257BFA0057FBB0/File/rc25412.pdf.

Krathwohl, D. R. (2002). A revision of bloom's taxonomy: An overview. *Theory into Pract.* 41, 212–218. doi:10.1207/s15430421tip4104_2

Kroll, P., Hofer, A., Stelzer, I. V., and Herwig, C. (2017). Workflow to set up substantial target-oriented mechanistic process models in bioprocess engineering. *Process Biochem.* 62, 24–36. doi:10.1016/j.procbio.2017.07.017

Law, M., Russo, A., and Broda, K. (2015). Learning weak constraints in answer set programming. *Theory Pract. Log. Program.* 15, 511–525. doi:10.1017/S1471068415000198

Lencastre Fernandes, R., Kishna Bodla, V., Carlquist, M., Heins, A.-L., Eliasson Lantz, A., Sin, G., et al. (2012). "Applying mechanistic models in bioprocess development,". *Measurement, monitoring, modelling and control of bioprocess. Advances in biochemical engineering/biotechnology*. Editor T.-H. N. Mandenius CF (Berlin, Heidelberg: Springer), 132, 137–165. doi:10.1007/10{_}2012{_}166

Lennox, B., Kipling, K., Glassey, J., Montague, G., Willis, M., and Hiden, H. (2002). Automated production support for the bioprocess industry. *Biotechnol. Prog.* 18, 269–275. doi:10.1021/bp0101839

Lisi, F. A. (2007). *Building rules on top of ontologies for the semantic web with inductive logic programming*.

Magalhães, R. P., Vieira, T. F., Fernandes, H. S., Melo, A., Simões, M., and Sousa, S. F. (2020). The biofilms structural database. *Trends Biotechnol.* 38, 937–940. doi:10.1016/j.tibtech.2020.04.002

Malik-Sheriff, R. S., Glont, M., Nguyen, T. V. N., Tiwari, K., Roberts, M. G., Xavier, A., et al. (2020). BioModels—15 years of sharing computational models in life science. *Nucleic Acids Res.* 48, D407–D415. doi:10.1093/nar/gkz1055

Mandreoli, F., and Montangero, M. (2019). Dealing with data heterogeneity in a data fusion perspective: Models, methodologies, and algorithms. *Data Handl. Sci. Technol.* 31, 235–270.

Martinez-Cruz, C., Blanco, I. J., and Vila, M. A. (2012). Ontologies versus relational databases: Are they so different? A comparison. *Artif. Intell. Rev.* 38, 271–290. doi:10.1007/s10462-011-9251-9

Mears, L., Stocks, S. M., Albaek, M. O., Sin, G., and Gernaey, K. V. (2017). Mechanistic fermentation models for process design, monitoring, and control. *Trends Biotechnol.* 35, 914–924. doi:10.1016/J.TIBTECH.2017.07.002

Mongo, D. B. (2022a). Mongodb. Available at: https://www.mongodb.com.

Mongo, D. B. (2022b). Who uses mongodb? Available at: https://www.mongodb.com/who-uses-mongodb.

Morbach, J., Wiesner, A., and Marquardt, W. (2009). OntoCAPE—a (re) usable ontology for computer-aided process engineering. *Comput. Chem. Eng.* 33, 1546–1556. doi:10.1016/j.compchemeng.2009.01.019

Muggleton, S., and De Raedt, L. (1994). Inductive logic programming: Theory and methods. *J. Log. Program.* 19, 629–679. doi:10.1016/0743-1066(94)90035-3

Muggleton, S. (1993). Inductive logic programming: Derivations, successes and shortcomings. *Lect. Notes Comput. Sci. Incl. Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)LNAI* 667, 21–37. doi:10.1007/3-540-56602-3{_}125

Narayanan, H., Luna, M. F., von Stosch, M., Cruz Bournazou, M. N., Polotti, G., Morbidelli, M., et al. (2020). Bioprocessing in the digital age: The role of process models. *Biotechnol. J.* 15, 1900172. doi:10.1002/biot.201900172

Nazemzadeh, N., Sillesen, L. W., Nielsen, R. F., Jones, M. N., Gernaey, K. V., Andersson, M. P., et al. (2020). *Integration of Computational Chemistry and Artificial Intelligence for Multi-scale Modeling of Bioprocesses*, 48. Elsevier Masson SAS. doi:10.1016/B978-0-12-823377-1.50050-1

Nielsen, R. F., Gernaey, K. V., and Mansouri, S. S. (2020). *A Hybrid Model Predictive Control Strategy using Neural Network Based Soft Sensors for Particle Processes*, 48. Elsevier Masson SAS. doi:10.1016/B978-0-12-823377-1.50197-X

Ning, C., and You, F. (2019). Optimization under uncertainty in the era of big data and deep learning: When machine learning meets mathematical programming. *Comput. Chem. Eng.* 125, 434–448. doi:10.1016/j.compchemeng.2019.03.034

Noorman, H. J., Heijnen, J. J., and Luyben, K. (1991). Linear relations in microbial reaction systems: A general overview of their origin, form, and use. *Biotechnol. Bioeng.* 38, 603–618. doi:10.1002/bit.260380606

Nopens, I., Batstone, D. J., Copp, J. B., Jeppsson, U., Volcke, E., Alex, J., et al. (2009). An ASM/ADM model interface for dynamic plant-wide simulation. *water Res.* 43, 1913–1923. doi:10.1016/j.watres.2009.01.012

Park, S. Y., Park, C. H., Choi, D. H., Hong, J. K., and Lee, D. Y. (2021). Bioprocess digital twins of mammalian cell culture for advanced biomanufacturing. *Curr. Opin. Chem. Eng.* 33, 100702. doi:10.1016/j.coche.2021.100702

Paul, A. (2022). Rdbms dominate the database market, but nosql systems are catching up. Available at: https://db-engines.com/en/blog_post/23.

Perez-Castro, A., Sanchez-Moreno, J., and Castilla, M. (2017). PhotoBioLib: A modelica library for modeling and simulation of large-scale photobioreactors. *Comput. Chem. Eng.* 98, 12–20. doi:10.1016/j.compchemeng.2016.12.002

Poveda-Villalón, M. (2020). *Introduction to linked (open) data and semantic web"*.Ontology engineering group - Universidad Politécnica de Madrid)

Raedt, L. D. (2010). "Inductive logic programming," in *Encyclopedia of machine learning*. Editors C. Sammut and G. I. Webb (Boston, MA: Springer US), 529–537. doi:10.1007/978-0-387-30164-8{_}396

Schomburg, I., Chang, A., and Schomburg, D. (2002). BRENDA, enzyme data and metabolic information. *Nucleic acids Res.* 30, 47–49. doi:10.1093/nar/30.1.47

Sin, G., Gernaey, K. V., and Lantz, A. E. (2009). Good modeling practice for PAT applications: Propagation of input uncertainty and sensitivity analysis. *Biotechnol. Prog.* 25, 1043–1053. doi:10.1002/btpr.166

Sin, G., Ödman, P., Petersen, N., Lantz, A. E., and Gernaey, K. V. (2008). Matrix notation for efficient development of first-principles models within PAT applications: Integrated modeling of antibiotic production with Streptomyces coelicolor. *Biotechnol. Bioeng.* 101, 153–171. doi:10.1002/bit.21869

Singh, R., Gernaey, K. V., and Gani, R. (2010). ICAS-PAT: A software for design, analysis and validation of PAT systems. *Comput. Chem. Eng.* 34, 1108–1136. doi:10.1016/j.compchemeng.2009.06.021

Soldatova, L. N., and King, R. D. (2005). Are the current ontologies in biology good ontologies? *Nat. Biotechnol.* 23, 1095–1098. doi:10.1038/nbt0905-1095

Stanford University (2022). Proégé. Available at: https://protege.stanford.edu.

Steinbeck, C., Han, Y., Kuhn, S., Horlacher, O., Luttmann, E., and Willighagen, E. (2003). The Chemistry Development Kit (CDK): An open-source Java library for

chemo- and bioinformatics. *J. Chem. Inf. Comput. Sci.* 43, 493–500. doi:10.1021/ci025584y

SuperPro Inc (2017). *SuperPro designer*.

Tateno, Y., Imanishi, T., Miyazaki, S., Fukami-Kobayashi, K., Saitou, N., Sugawara, H., et al. (2002). DNA Data Bank of Japan (DDBJ) for genome scale research in life science. *Nucleic acids Res.* 30, 27–30. doi:10.1093/nar/30.1.27

Tsopanoglou, A., and del Val, I. J. (2021). Moving towards an era of hybrid modelling: Advantages and challenges of coupling mechanistic and data-driven models for upstream pharmaceutical bioprocesses. *Curr. Opin. Chem. Eng.* 32, 100691–100697. doi:10.1016/j.coche.2021.100691

Udugama, I. A., Lopez, P. C., Gargalo, C. L., Li, X., Bayer, C., and Gernaey, K. V. (2021). Digital twin in biomanufacturing: Challenges and opportunities towards its implementation. *Syst. Microbiol. Biomanuf.* 1, 257–274. doi:10.1007/s43393-021-00024-0

United Nation Development Program (2014). Sustainable development goals: Improving human and planetary wellbeing. *Tech. Rep.* 82.

Utgoff, P. E., Cussens, J., Kramer, S., Jain, S., Stephan, F., Raedt, L. D., et al. (2011). Inductive transfer. *Encycl. Mach. Learn.* 545–548, 545–548. doi:10.1007/978-0-387-30164-8_401

Wikipedia (2022a). Linked data. https://en.wikipedia.org/wiki/Linked_data.

Wikipedia (2022b). *Ontology (information science.*

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., et al. (2016). The FAIR Guiding Principles for scientific data management and stewardship. *Sci. Data* 3, 160018–160019. doi:10.1038/sdata.2016.18