Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 207 (2022) 3433–3442

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Knowledge Based and Intelligent Information and Engineering Systems, KES2022, 7-9 September 2022, Verona, Italy

# Towards Improved Visualization and Optimization of Aquaculture Production Process

Aya Saad[a,*], Oscar Nissen[a], Espen Eilertsen[a], Finn Olav Bjørnson[a], Tore Norheim Hagtun[b], Odd-Gunnar Aspaas[b], Alexia Artemis Baikas[b], Sveinung Johan Ohrem[a]

[a]*Aquaculture Department, Sintef Ocean AS, Trondheim, Norway*
[b]*Searis AS, Trondheim, Norway*

**Abstract**

Aquaculture is one of the largest, and fastest growing industries in Norway. Recently, the industry has experienced significant development in the daily operations acquiring new technologies and systems that capture data and automate the different processes. These emerging technologies enable the generation of enormous amounts of data from sensors in the fish cages, cameras, boats, and feeding control rooms. Additional information relevant to the aquaculture industry is based on e-mails, manual notes, or intrinsic experiences and knowledge exchanges. One of the critical aspects of successful fish farming operation management, which is yet not achieved, is to allow domain experts to gain insight into the interconnection between the broad spectrum of heterogeneous data currently realized.

This paper describes a framework for storing and retrieving critical information connected to fish farming based on a graph database approach. The overall architecture is presented with detailed illustrations of how data is visualized and interpreted through a user-friendly interface. Accordingly, this work demonstrates how aquaculture users can benefit from the system to identify possible connections in the data and reveal previously undiscovered causalities and correlations that suggest optimal actions. Further, studies and evaluations of the querying system are conducted, evaluating the capability of the proposed design to process complex relationships. This work showcases that the system helps fish farmers and aquaculture users gain knowledge, reveal hidden links in the data, and improve aquaculture operations.



*Keywords:* Knowledge representation; graph database; aquaculture production process

## 1. Introduction

The aquaculture industry, one of the most growing industries in Norway and the largest worldwide contributor to food production, generates enormous amounts of data every day. Data gathered comes from heterogeneous sensors in the fish cages, cameras, boats, feeding barges, e-mails, notes, and conversations between people working in production

* Corresponding author.
*E-mail address:* aya.saad@sintef.no


10.1016/j.procs.2022.09.531

sites [8]. Some tools exist for gathering and displaying the produced data. In addition, there are several initiatives for standardizing the data (e.g. the Aquacloud collaboration [1]), yet there exist many challenges with today's solutions related to finding and visualizing the interconnections between the gathered data. When the data quantity is high, the interconnections between data become more critical, complex, and hard to realize. Furthermore, it is difficult for humans to navigate through millions of data points and find these interconnections. To further the advancement of the fish farming operations, domain experts, e.g., site managers and fish farmers, need to gain insights into the interconnections between the broad spectrum of the gathered heterogeneous data. Such insightful understanding has not yet been realized. The Precision Fish Farming (PFF) framework [8] promotes the idea of building up smart data for the aquaculture industry to achieve optimal operations. PFF accordingly identifies four concepts that work together iteratively in a cycle: 1) Observe (collect data), 2) Interpret (analyze data), 3) Decide (decision making and support), and 4) Act (actions executed upon the decision taken). Making data smart necessitates the construction of detailed definitions and domain specifications that transform data into knowledge rather than intrinsic experiences. This shift is essential to the industry's digital transformation, where technology is used for innovation, simplification, and improvement. To achieve a digital transformation, there is an urgent need to put in place systems that provide more insights into the data and help end-users realize hidden patterns and trends for better decision making and process optimization [5].

To provide this digital transformation, this paper presents the design and evaluation of a 5 components framework based on a graph database approach. The contribution of this paper is fourfolds:

- Design and implement a domain data model for the aquaculture fish production process which is the core to achieving smart information.
- Build a knowledge graph and a graph database from the data model to facilitate easy and user-friendly information storage and retrieval.
- Illustrate how the different concepts and data gathered can be correlated to provide insightful meanings to the end-users allowing them to make better decisions that lead to fish production process optimization.
- Highlight the benefits and potential use of the system.

This work addresses digitalization and improved fish survival and welfare in the aquaculture industry by targeting data-driven production and providing an in-depth understanding of complex interconnections in the data. Graph databases are used as a basis for the implementation since they are considered good tools for storing relationships and connections between data [12]. Data related to fish farming operations may be a combination of groups of fish, treatment methods, feeding strategies, assets, sensor data, and observations made by aquaculture sites' personnel [8, 5]. As a first step, all these data are conceptualized in a domain-specific data model ontology targeting the aquaculture production process. A knowledge graph then uses this conceptual model to create individual instances representing part of the production life cycle. The graph database, in turn, stores the knowledge graph and the logic that describes the interconnections between the different entities. The implemented system is modular; it enables users to store data and relevant historical information. The system is also augmented by a visual interface that further allows the end-users to observe how a production cycle evolves and what are the main factors that affect the end-product quality. Moreover, this visual interface allows the end-users to input additional feedback quotes or narrative notes that are fed back to the graph database. In this way, the data is iteratively and continuously incremented.

This approach is a new way of visualizing aquaculture data that improves the understanding of the operations and highlights useful information that can reveal undiscovered causalities and correlations which can be used to suggest optimal decisions and actions for the fish farming production process. Moreover, when including historical information, the proposed system can provide fish farmers with support for long- and short-term decisions by suggesting actions or strategies that have been successful in the past. This paper demonstrates how the implemented system forms a good foundation for decision-making that can help optimize the planning and scheduling of aquaculture production processes.

## 2. Related work

Big Data analytics in the aquaculture industry are essential to support and utilize large-scale aquaculture data. Akerkar et al. [3] reported a need for specialized data architectures to gain further value from data integration and bring data-driven aquaculture practices to a new level of development. Furthermore, reports on the state-of-the-art data analytics within the industry have stated relational databases to be lacking the ability to meet modern applications in the form of more specified architectures [6]. One of the mentioned scalable architectures is graph databases. Among limited existing literature on graph databases used in the aquaculture industry, Tejaswini et al. showed using graph databases is useful in establishing relationships between fish species [20]. Their study focused on correlating the combination of fish species with suitable areas for fish farming. In contrast, our work focuses on the daily operations related only to salmon fish farming and how graph databases can be adopted to aid the decision-making that optimizes the production process.

The adaption of graph databases has been studied in many other research areas as well, including dairy farming [21] biology [24], medicine [17], manufacturing [15] and finance [23]. Yoon et al. [24] established a graph database that outperforms traditional relational databases both in terms of efficiency and potential capabilities related to the understanding of complex relationships between biological data. On the other hand, Yerashenia et al. [23] created a novel Bankruptcy Prediction Computation model with embedded machine learning to add further insight into a business. Lastly, Tomic et al. [21] presented an information management solution for dairy farming in the form of a knowledge graph, which refers to how a graph database organizes data. The work in this paper builds on previous research for developing efficient graph database models. It focuses on describing domain specifications that describe the fish farming production process.

## 3. Methodology

The main scope is to represent the aquaculture production process and operations to enhance end-user experience and better understand how operations are exerted and linked together. The gained insights enable optimal decision-making in the operations towards achieving better fish quality. The adopted method of an aquaculture production process representation should seamlessly allow the capture, store, and reuse of information systematically, linking low-level numerical sensor data with high-level domain expert (e.g., fish farmers) experiences. Accordingly, a knowledge-based system is incrementally constructed by building an architecture for storing and retrieving critical information connected to fish farming operations. The core component of this architecture is a graph database. Critical information includes fish farming assets, transportation equipment, fish batches, feeding targets, events, sensor data, and human observations. One of the main advantages of using a graph database is providing insight into interconnections between variables or entities through better data visualization. This visualization can reveal, for example, the root cause of undesired incidents, such as a high mortality rate observation.

In a graph database, all the information about an entity can be contained in a single node, and all related information is visualized through the edges connected to this node [18]. Each edge in the graph can be assigned a weight that holds information about the cost associated with it. Graph database systems exploit this particular structure in algorithms that, e.g., traverse the graph and find the shortest path or sub-graphs [10]. One well-known example of graph databases used in practice is social network models, e.g., Facebook. Graph databases are more scalable and adaptable than relational databases allowing faster query-answer mechanisms that explore the defined nodes (entities) [12]. In graph databases, the querying algorithm has a computational complexity of $O(m)$ where m is the number of steps (edges visited) to traverse the graph. This computational complexity in relational databases is estimated as $O(m \log n)$ where $n$ is the number of nodes (entities) defined in the graph [19].

## 4. Proposed framework

This work aims to provide more insights into the aquaculture production process and accordingly highlights the means to optimize them. Figure 1 depicts a block diagram showing the five underlying components of the proposed framework and how they are interrelated, making an iterative workflow cycle that incrementally builds up the database. The process follows a systematic approach that starts by gathering all the possible information to identify the key concepts to establish the domain model specifications of the aquaculture production process. Accordingly, the domain model is constructed in the form of a modular ontology that aims to create smart information that facilitates knowledge extraction and provides more insights into the data analysis. A knowledge graph then uses the ontology to create individual and specific instances considering an example scenario that represents part of the aquaculture production cycle. Next, a graph database [22] stores the knowledge graph data and the logic that describes their interconnections and context. The fifth component of the proposed framework is the user interface, where data is represented to the

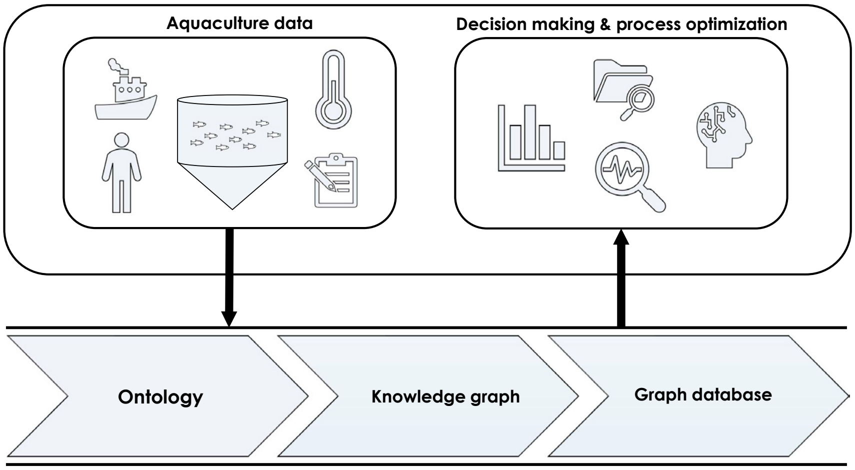


Fig. 1. Block diagram illustrating the different components of the proposed framework.

end-user, showing historical trends and information on how they are interrelated. From this interface, end-users can highlight additional observations and provide their feedback on them in the form of quotes or narrative notes that can be utilized to incrementally build the data that is fed back again into the knowledge graph. The novelty of this approach compared to the existing dominating systems in aquaculture is the use of the ontology and knowledge graph to create rich, interconnected and smart data sets.

### *4.1. Aquaculture production process data*

The aquaculture industry has a massive amount of data characterized as being heterogeneous. This data can be gathered from a broad spectrum of sensors attached, for example, to fish cages, boats, and feeding centrals. Other forms of data can be images or videos captured by cameras, and e-mails, quotes, notes and observations made by people working at the production sites, such as veterinarians, farmers, or managers.

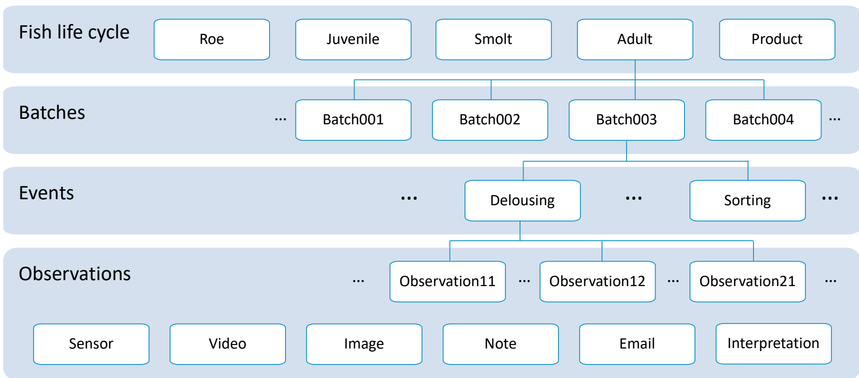


Fig. 2. Aquaculture production cycle

This work follows a bottom-up approach to graph database creation identifying the building blocks that support the development of the system. Some design considerations must be taken into account to facilitate the store, retrieve, and reuse of information. The raw data and information gathered from sensors, end-users, and domain experts are mostly observations and user feedback, which can be quantitative and qualitative. These observations are modelled as a chronologically ordered list to maintain and track their sequence of occurrence. Hence, reasoning about the sequence and interconnectivity of events is essential to support the tracking and the full understanding of the aquaculture production process from the start (roe fish) to the end (end-product). These individual observations in the data model need to be further grouped to provide a logical structure and maintain common metadata. The root element of this observation list tree is the event (e.g., delousing, feeding, sorting, transportation, vaccination) since an event holds a large amount of metadata about the context. As illustrated in Figure 2, events consist of an ordered sequence of observations within a specific time interval in which a fish batch is actively engaged or affected. It is worth noting that a fish batch is part of a fish life cycle (e.g., juvenile or adult). The fish farmer company decides how to allocate and categorize fish in batches and assign them a cage in its facility. In addition, batches can move from one fish cage to another, or be re-defined, as an effect of applying a specific aquaculture operation.

### 4.2. *Ontology formulation*

Creating and developing ontologies is a well-known approach used to represent concepts of the real world. Ontology creation is essential to designing and modelling a coherent knowledge management system from various sources and tools. Ontologies are part of the sense-making process for analyzing data and extracting knowledge [14]. The aquaculture production process ontology will act as the main promoter of understanding what is happening during the operations. The main goal is to demonstrate and convey information about how and where a task is performed and communicated, hence identifying the cause and effect of performing different operations during the fish production life cycle. To achieve this goal, the ontology model in this work elaborates on how observations and events with various representations and formats are logically grouped and related in a sequence.

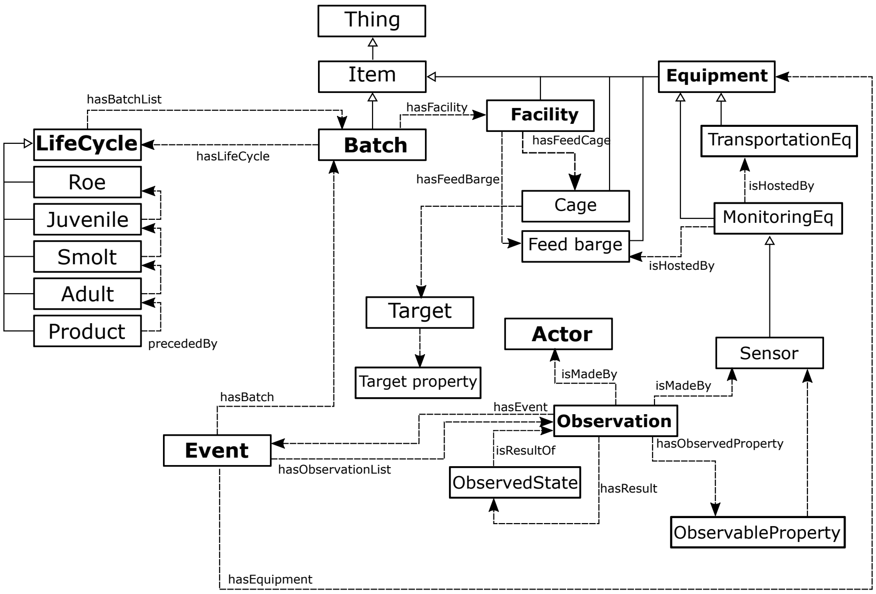


Fig. 3. A representation of the proposed ontology showing the different concepts of the data model related to the aquaculture operations.

In the ontology design, the domain meta-model defines the interface, making it possible to integrate different domain models uniformly. The domain model describes the real world in generic terms. It is linked to the data in two ways:

1. Domain concepts express the data relevance and elaborate on the domain model.
2. Model insights in the domain refer to data as part of an argument or rationale.

**Tags and metadata** are labels that link objects to the ontology. They describe the actors and their roles (e.g., supplier, veterinarian, fish farmer), events (e.g., delousing, sorting, transportation), equipment (e.g., sensors, wellboat), and locations (e.g., facility). There are two interpretations of these links:

i. They are objective observations, e.g., sensor-based measurements, which can reasonably be represented as properties of the data object.
ii. They are subjective interpretations stored, e.g., quotes or notes, allowing multiple expressions for conflicting views.

Both objective and subjective interpretations are valid for different links. Data items can be tagged by attaching a cross-reference, recording timestamp, and author (made by a human or a sensor). Figure 3 depicts the key concepts that define the data model of the aquaculture production process ontology. The ontology in this work is a modular Ontology Web Language (OWL)-full ontology based on OWL-Description Logic (DL) or OWL-DL with a combination of features from the Resource Description Framework (RDF) schema [11]. A key concern is establishing a precise semantic framework that links observations to domain models. The domain specifications are defined over three types of modeling primitives: concepts, individuals, and roles [14, Ch. 3], where an individual is a single element of the domain, a concept is an abstract representation for a set of individuals, and a role defines a relationship between individuals.

The ontology is modular and complies with the **OWL** ontology in the sense that the class **Item** is a subclass of the **OWL:Thing** and at the same time it is the superclass of all items, or concepts defined in the ontology and highlighted in bold in Figure 3 listed as **LifeCycle**, **Batch**, **Facility**, **Equipment**, **Event**, and **Observation**. The super class **Item** identifies common object properties over the class **Item** which are inherited by all of its subclasses. Those common properties are listed as:

- *hasTime*. A timestamp used to record the creation time or the time of entry into the database.
- *startTime*. Raw representation of the start time is inserted to simplify processing, avoiding using OWL time.
- *endTime*. Raw representation of the end time given when a time interval is used.
- *hasAuthor*. The person or equipment (e.g., sensor) who authored the observation.
- *hasItem*. The parent item in the hierarchy or, in other words, the item to which this item belongs.

The **Observation** class is a critical subclass of the class **Item** which models all types of observations either made by a human or a sensor. The *isMadeBy* relationship differentiates between reported objective (when the **Observation** is computed or provided by a sensor) and subjective entries (when a person makes them). The **Sensor** and the **Actor** data models are designed to link the **Observation** to its author. One fundamental data model is the **Event** model; it is used to record the different events that happen on a **Batch**. It is worth noting that each recorded **Observation** must be attached to an **Event**, hence keeping track of the operations that handle a specific **Batch**. The **LifeCycle** sub-ontology is based on the plant ontology detailed in [4] since it represents the different stages of the fish life cycle from roe to product. Each stage is related to its precedent with a *precededBy* relation. In fish farming, fish are grouped in batches to facilitate the handling of operations. A fish within a stage in its life cycle (i.e., adult) can be assigned to more than one batch. Accordingly, the **Batch** sub-ontology has a relation *hasLifeCycle* to define at which stage of the life cycle the batch is. **Facility**, and **Equipment** are other sub-ontologies that are explicitly built to meet the design purpose of the aquaculture process. The **Cage** model is a sub-class of **Equipment** which has some targets (e.g., feeding amount goal, growth rate). These targets were added to evaluate the fish farming operations' performance and gauge how the fish farming facilities follow their production plans.

### *4.3. Knowledge graph formulation*

Knowledge graphs use ontologies to create individual and specific instances. This Section elaborates on the knowledge graph formulation of a delousing operation that took place at the fish production facility on July $7^{th}$, 2021. The operation was performed on one of the adult batches (*Batch001*) for an entire workday from 7 AM to 6 PM. The batch was moved from one cage (*Merd02*) to another (*Merd01*), both located at the same production facility. To perform this operation, the production facility hired a supplier (*Supplier01*) who provides this service with its own well-boat (service equipment with id: *WellBoat01*). The well-boat has sensors and equipment attached to monitor the status, perform the delousing, and provide warnings in case the measurements reach certain thresholds or limits that need the operators to interfere and take proper actions. At the same time, the production facility company has additional sensors and cameras attached to the cages for the same monitoring purposes. One personnel at the fish farming company noticed at 1 PM from the videos captured by the camera attached to (*Merd02*) that fish in (*Batch001*) were pumped very harshly during the delousing operation. This observation has been noted and saved as (*NOBS001*) and considered subjective since a person made it. Figure 4 illustrates the knowledge graph formed from the ontology detailed in Section 4.2, which is related to this example scenario (delousing operation), highlighting the main instances and actors.

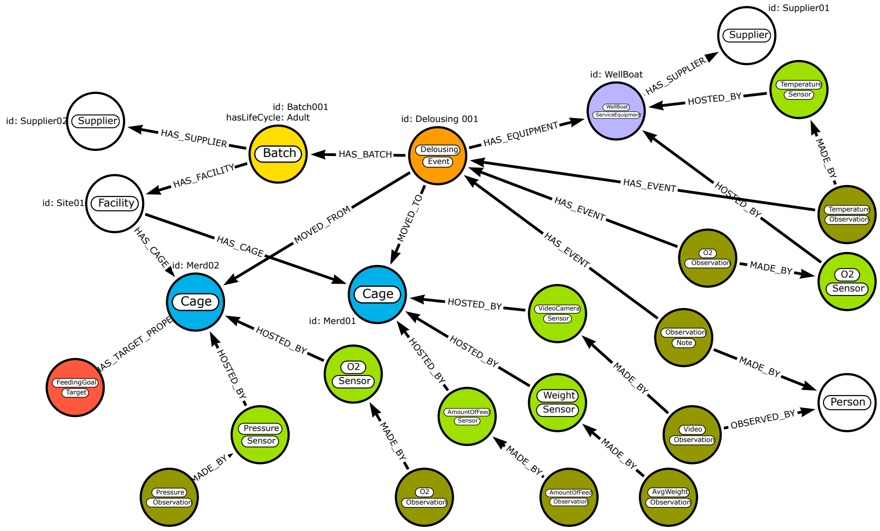


Fig. 4. A representative knowledge graph showing the different nodes involved in the delousing event operation.

### *4.4. Graph database formulation*

Graph databases, by definition, are not using the Structured Query Language (SQL), as such they are called NoSQL databases and store information in a graph-like structure defined by nodes, edges, and properties. They are more suited to applications, especially when data is densely interconnected but also not necessarily structured. Graph databases are used to store knowledge graph data and the logic that describes their interconnections and context. Graph operations are executed to manipulate the data and provide insights about them and their interconnectivity. Among the current graph database models such as AllegroGraph [2], DEX [16], HyperGraphDB [13], and Neo4j [9], the Neo4j is chosen for the graph database implementation in this project due to its efficient implementation [22, 7]. The Neo4j is an open-source java graph database that utilizes the Cypher query language to create and interact with the data.

One of the advantages of Neo4j is that it can store the formal representation and description of the model expressed by the ontology with a simple loss-less conversion. This conversion is a simple mapping from one graph format to another. It assigns each (class, property, and individuals) item in the ontology a node label. Annotations are mapped into properties (e.g., labels, id, descriptions) and superclass become edges. In this manner, the graph database stores the data and the logic behind their interconnectivity from the knowledge graph. It is worth noting that this conversion can be reversible, enabling the mapping of the graph components and exposing them as RDF, the formal ontology description. This fact facilitates an incremental and flexible integration of both the ontology and the graph database, ensuring the system integrity.

### *4.5. Data visualization and interpretability*

Part of the strength of graph data models from a data visualization point of view is that in many cases their structure is closely linked to how a domain expert would intuitively describe and understand their industrial reality and relations on a whiteboard. The intuitive and visual nature of the data structure opens up new possibilities for visualization and data exploration while increasing the interpretability of vast amounts of data for both human and machine use cases. As more data is being generated and more parts of the value chain are digitalized, a vital challenge for many industries is solving the interpretability of data for a workforce of varying backgrounds, skill levels, and roles. The graph approach can be a fundamental stepping stone to building more intuitive and powerful applied tools for the aquaculture ecosystem.

Building, enriching, and interacting with the graph data structure requires dynamic and interactive interfaces for both human and machine applications. The proposed approach builds on the work done by Clarify.io[1], a platform for gathering data throughout the aquaculture value chain and enriching it with human context and expertise through comments, labels, and annotations. The Clarify platform contributes with enriched streaming data and events to build and evolve the aquaculture knowledge graph over time, both through machine-to-machine communication APIs and by keeping humans in the loop in collaborative end-user tools. On top of the graph database, various tools for interaction and exploration may be built, supporting all aspects of decision-making from observation through action. From interacting directly with the graph structure to explore traceability and relations, to more advanced contextualized search, to visualization, modeling, and applied optimization tools that democratize access and usability of vast amounts of complex data for the whole workforce.

## 5. Results and discussion

Initial results in this work include a complete ontology for the aquaculture production data. Furthermore, a graph database representation of all assets that belong to one of the fish farms has been implemented in Neo4j. The graph database and graph visualization have been applied to new and historical data. Including historical and ongoing production data helps provide more information and knowledge about the performance of the fish farming operations during different production cycles and shows how they affect the end product (i.e., fish) quality.

### 5.1. Simple querying with a user-friendly interface

An example Cypher query that returns all events that happened on the batch with id *'Batch001'* is as follows:

```
MATCH path=(:Batch{id: 'Batch001'})-[:HAS_BATCH]-(e:Event) RETURN path
```

This example elaborates on which events from the operations affected the final product. The keyword *path* in the Cypher query listed above shows a visualization of the full path from the **Batch** and all the links that lead to the **Event**s that affected this **Batch** *id: 'Batch001'*. When the *path* keyword is removed, the query returns only the list of **Event** leaf nodes. Among the list of events that happened on the *id: 'Batch001'*, this query returns the delousing operation *id:'Delousing001'* that was performed on July the $7^{th}$, 2021 and detailed in Section 4.3 from the graph database.

### 5.2. Adding more functionalities by integrating sensor data in a single interface

One main advantage of adopting this framework is that it provides a single hub interface that allows the end-users to visualize heterogeneous data captured or gathered from various technologies. These technologies can be sensors that monitor environmental conditions, cameras that capture videos and images of the fish, but also e-mails, manually written notes, and quotes made by people working in the production sites. Having a single interface that integrates, for example, sensor data (e.g., temperature, oxygen, pressure, and current) to observe a single batch for its entire life cycle irrespective of its location is novel to the fish farming operation in the aquaculture industry. The proposed framework can provide this information and link it to the quality of the final product. This interface can help site managers or fish farmers find whether specific environmental conditions can affect the fish welfare hence the end-product quality.

### 5.3. Comparing the performance of two production life cycles

Generally, the same production facility site can produce fish batches with different quality grades. A significant benefit of the proposed framework is to allow end-users to check on joint events and observations that are tied to those events on different fish batches located in the same production facility site. Querying these joint events and their observations highlights the common reasons for 'good' or 'bad' batch quality grades. For example, a production facility that applies the same operation, e.g., delousing, on its batches, with the same service equipment, e.g., wellboat, from the same supplier and operation crew can realize a difference in its batch quality grade. The proposed framework having all the measurements and observations integrated into one interface can help query the system

[1] https://www.clarify.io/

about the reason for the difference in the end-product quality grade. For the example mentioned above, querying the knowledge graph shows that the delousing operation took several days. On the first day, observations show that the crew faced harsh weather conditions, which led to postponing the delousing operations for a couple of days for the rest of the batches. This delay urged the fish farmers to starve the fish for a couple of more days before applying the delousing operation to the rest of the batches. The query output infers that the higher quality end product is a result of stopping the feed for more days, making the fish more tolerant to the delousing process. To the best of our knowledge, these cause-and-effect realizations cannot be identified without having the system in place. These explanations can provide recommendations on how fish farming operations are exerted nowadays and how they can be optimized and customized for a specific aquaculture facility site.

### *5.4. Process optimization*

The proposed framework is a beneficial tool that can help end-users, e.g., site managers and fish farmers, improve their production plans. This tool can provide insightful meanings behind the captured data by querying the system and comparing the performance of different production cycles, including current and historical data. When starting a new production cycle, site managers at the aquaculture facility can use the tool as a recommender system. This recommender system can pinpoint the best practice for planning a production cycle, such as 1) the best plan for a given facility location, 2) the most suitable service provider companies who should be involved in the operations, 3) the best supplier of the roe fish, 4) the most suitable time of the year to produce the fish, and 5) the most appropriate schedule to perform the different operations. Moreover, the system can help the end-users find hidden relations that impact the production cycle. Some other potential use of the system can be answering the following questions:

- Why did a specific batch have more mortalities than other batches?
- Why was a particular batch a lower quality than another batch?
- What are the operations that are tied to each production cycle?
- Are there any anomalies and abnormal observations affecting the end-product quality?
- How many anomalies or abnormal observations are coupled to each fish life cycle stage?
- What are the equipment and vessels involved in operations performed on a batch?

## 6. Conclusions and further work

This paper proposes a novel framework as a new approach to visualizing aquaculture data that enables end-users, e.g., site managers and fish farmers, to learn more about the cause and effect of the decisions taken and actions exerted during the aquaculture production cycle. The framework is based on a graph database established by a conceptual ontology model that identifies a coherent knowledge management system from various sources and tools. The ontology of the aquaculture production process defines the logical grouping of the different concepts to realize better how the various concepts are sequenced and interrelated. This paper shows how the proposed framework can provide insightful meanings and correlations between the various captured data. The input data to the system is heterogeneous and includes sensor measurements that monitor the environmental conditions and manual observations provided by aquaculture facility personnel. The framework also comprises historical production cycle data to promote better tracking of fish throughout their production cycle, from roe to end-product. The graph database representation can lead to discovering new relationships that were previously underestimated or unknown to the fish farmers. This framework is an iterative process allowing a continuous evolution of data gathering and inclusion into the core graph database to enrich the quality of the data and their correlation. Furthermore, the work presented shows how this framework can be a foundation for decision making and process optimization by identifying the hidden relationships which were previously unknown.

This work paves the way for many future research directions. One of them can be studying the system's scalability to showcase how reliable and flexible it is when data from many production sites are integrated. Another future direction is augmenting the framework with Machine Learning capabilities to realize unknown relationships automatically and predict the effects these relationships may have on the end product. Last but not least, the system visualization tool can be further enhanced to understand the connection between the different observations. The proposed framework can be a beneficial visualization tool for aquaculture industry personnel. The work presented is a good foundation for decision-making that can improve fish production operations since it provides more insights that help optimize the planning and processes.


## Acknowledgements

This work has been funded by the Research Council of Norway through the project: Graph databases for visualisation and optimization of aquaculture production processes (RCN 321422). The project owner is Searis AS, Eide Fjordbruk AS is an industrial partner, and SINTEF Ocean AS serves as a research partner. The authors acknowledge the inputs from Erlend Eide of Eide Fjordbruk AS.


## References


[1] Industry collaboration enables big data analytics, Mar 2022. URL: https://aquacloud.ai/about/.
[2] Jans Aasman. Allegro graph: Rdf triple database. *Cidade: Oakland Franz Incorporated*, 17, 2006.
[3] Rajendra Akerkar and Minsung Hong. Big data in aquaculture. Technical report, Vestlandsforskning, 6, 2021.
[4] Shulamit Avraham, Chih-Wei Tung, Katica Ilic, Pankaj Jaiswal, Elizabeth A Kellogg, Susan McCouch, Anuradha Pujar, Leonore Reiser, Seung Y Rhee, Martin M Sachs, et al. The plant ontology database: a community resource for plant structure and developmental stages controlled vocabulary and annotations. *Nucleic acids research*, 36(suppl_1):D449–D454, 2008.
[5] Hans V Bjelland, Martin Føre, Pål Lader, David Kristiansen, Ingunn M Holmen, Arne Fredheim, Esten I Grøtli, Dariusz E Fathi, Frode Oppedal, Ingrid B Utne, et al. Exposed aquaculture in norway. In *OCEANS 2015-MTS/IEEE Washington*, pages 1–10. IEEE, 2015.
[6] João Costa, Matjaž Rihtar, Ioannis Zarifis, Antzoulatos Gerasimos, John McLaughlin, and Derek O'Keefe. Aquasmart: Aquaculture smart and open data analytics as a service, 2016. URL: https://ec.europa.eu/research/participants/documents/downloadPublic?documentIds=080166e5a5de4dd0&appId=PPGMS.
[7] Diogo Fernandes and Jorge Bernardino. Graph databases comparison: Allegrograph, arangodb, infinitegraph, neo4j, and orientdb. In *Data*, pages 373–380, 2018.
[8] Martin Føre, Kevin Frank, Tomas Norton, Eirik Svendsen, Jo Arve Alfredsen, Tim Dempster, Harkaitz Eguiraun, Win Watson, Annette Stahl, Leif Magne Sunde, et al. Precision fish farming: A new framework to improve production in aquaculture. *biosystems engineering*, 173:176–193, 2018.
[9] Faming Gong, Yuhui Ma, Wenjuan Gong, Xiaoran Li, Chantao Li, and Xiangbing Yuan. Neo4j graph database realizes efficient storage performance of oilfield ontology. *PloS one*, 13(11):e0207595, 2018.
[10] Ralf Hartmut Güting. Graphdb: Modeling and querying graphs in databases. In *Proceedings of the 20th International Conference on Very Large Data Bases*, pages 297–308, 1994.
[11] Matthew Horridge, Simon Jupp, Georgina Moulton, Alan Rector, Robert Stevens, and Chris Wroe. A practical guide to building owl ontologies using protégé 4 and co-ode tools edition1. 2. *The university of Manchester*, 107, 2009.
[12] Michael Hunger, Ryan Boyd, and William Lyon. The definitive guide to graph databases for the rdbms developer. *Neo Technology*, 2016.
[13] Borislav Iordanov. Hypergraphdb: a generalized graph database. In *International conference on web-age information management*, pages 25–36. Springer, 2010.
[14] C Maria Keet. An introduction to ontology engineering, 2018.
[15] Marlene Kuhn and Jörg Franke. Data continuity and traceability in complex manufacturing systems: a graph-based modeling approach. *International Journal of Computer Integrated Manufacturing*, 34(5):549–566, 2021.
[16] Norbert Martinez-Bazan, Sergio Gómez-Villamor, and Francesc Escale-Claveras. Dex: A high-performance graph database management system. In *2011 IEEE 27th International Conference on Data Engineering Workshops*, pages 124–127. IEEE, 2011.
[17] Safikureshi Mondal, Anwesha Basu, and Nandini Mukherjee. Building a trust-based doctor recommendation system on top of multilayer graph database. *Journal of Biomedical Informatics*, 110:103549, 2020.
[18] Jan Paredaens, Peter Peelman, and Letizia Tanca. G-log: A graph-based query language. *IEEE Transactions on Knowledge and Data Engineering*, 7(3):436–453, 1995.
[19] Ian Robinson, Jim Webber, and Emil Eifrem. *Graph databases: new opportunities for connected data.* " O'Reilly Media, Inc.", 2015.
[20] H Tejaswini, Manohara Pai MM, and Radhika M Pai. Knowledge graph for aquaculture recommendation system. In *2021 IEEE Mysore Sub Section International Conference (MysuruCon)*, pages 366–371. IEEE, 2021.
[21] Dana Tomic, Domagoj Drenjanac, Sandra Hoermann, and Wolfgang Auer. Experiences with creating a precision dairy farming ontology (dfo) and a knowledge graph for the data integration platform in agriopenlink. *Agrárinformatika/Journal of Agricultural Informatics*, 6(4):115–126, 2015.
[22] Jim Webber. A programmatic introduction to neo4j. In *Proceedings of the 3rd annual conference on Systems, programming, and applications: software for humanity*, pages 217–218, 2012.
[23] Natalia Yerashenia and Alexander Bolotov. Computational modelling for bankruptcy prediction: Semantic data analysis integrating graph database and financial ontology. In *2019 IEEE 21st Conference on Business Informatics (CBI)*, volume 1, pages 84–93. IEEE, 2019.
[24] Byoung-Ha Yoon, Seon-Kyu Kim, and Seon-Young Kim. Use of graph database for the integration of heterogeneous biological data. *Genomics & informatics*, 15(1):19, 2017.