Formal Ontology in Information Systems
N. Aussenac-Gilles et al. (Eds.)

doi:10.3233/FAIA231140

Ontology Explorer: An Ontology-Based Visual Analytics System for Exploring Time Series Data in Oil and Gas

Nicolau O. SANTOS [a,1], Jonathan C. RIVERA [a], Rafael H. PETRY [a],
Fabricio H. RODRIGUES [a], Givanildo S. NASCIMENTO [a,b], João L. D. COMBA [a,2]
and Mara ABEL [a]

[a] *Instituto de Informática - UFRGS*
[b] *Petróleo Brasileiro S.A. (Petrobras)*
ORCiD ID: Nicolau O. Santos https://orcid.org/0000-0003-0901-2465, João L. D. Comba https://orcid.org/0000-0003-2921-2130

Abstract. Data analytics is the best approach for extracting hidden patterns and trends from time series data. One strong limitation that restrains the use of the method is the difficulty in selecting the appropriate data when a large variety of records come from several providers. Domain ontologies can organize and offer a uniform view for data analysis without plastering the data in a rigid format. This work describes an innovative visualization platform for data analysis of petroleum production time series that allows the user to explore the semantics of the oil well data with the support of a well-founded domain ontology. The O3PO ontology represents a production plant's installation assets, describing the equipment's relevant properties, such as position, relationship with other assets, and sensor measurements. The visualization platform takes advantage of the ontology to assist the user in locating the installation component, equipment, and collected measurements, bringing the time series data to the analytic tool for analysis by various groups. The ontology provides a taxonomy for navigating between classes and subsequently selecting instances, components, and properties. We describe the use of the visualization tool in real-case data from an offshore oilfield in Brazilian Pre-salt. This work contributes to the conception of the next generation of digital twins for the oil and gas industry.

Keywords. applied ontology, data visualization, time-series data, digital twins, oil and gas, visual analytics

1. Introduction

Well-founded ontologies are a powerful tool to make apparent the semantics of the data that supports the decision-making process in surveillance environments. The monitoring and optimization of petroleum-plant production is a complex example of this environment. Optimizing oil and gas production requires the right-time analysis of a large

---

[1] Corresponding Author: Nicolau O. Santos, e-mail=nicolau.santos at inf.ufrgs.br
[2] Corresponding Author: João L. D. Comba, e-mail=comba at inf.ufrgs.br

amount of data produced by several distinct data providers that adopt their conventions and formats. The data refers to several types of physical properties (pressure, temperature, *etc.*) and the state of valves placed at different heights in the production tubing of wells with distinct functionalities (*e.g.,* production, injection, or observer well). An extensive set of sensors attached to the pieces of equipment collect such data at several different intervals, employing various methods and units of measurement, and store it in tabulated file formats. The usual approach for monitoring petroleum wells is to label each incoming data set with a tag that summarizes the origin, kind, quantity, measurement unit, position, and many other data characteristics, according to the view and convention of the engineering staff. Although experienced engineers can manage to select and group the relevant data, as the type of measures available increases it is more challenging to integrate the required information at the right time. Also, decoding the data tags is a significant barrier for any new professional team member.

Ontologies can play a significant role in this scenario as it offers an application-independent tool to raise and clarify the semantics of the data. A domain ontology can express the essential properties of the plant facility to characterize each piece of equipment unequivocally. In addition, it explicitly describes the relevant dependent ontological properties and their domains and axioms to characterize the entities to which the files with measurement data refer, as well as the spatial and functional relations of the pieces of equipment in the complex organization of a petroleum production plant. Navigating the petroleum-plant ontological model allows engineers to understand the configuration and dependence of the sensor measurements and to select the data for analysis.

Our work explores the frontiers of technology for facilitating the intelligent data analytics of petroleum production data under the requirements of a real offshore production plant. The study is part of the PeTwin project[3], which aims to define the system architecture and application requirements for the new generation of digital twins (DTs) for petroleum production plants. In this paper, we describe the system architecture of a tool for selecting, visualizing, and exploring data with the support of the O3PO[4] domain ontology [18].

In summary, the main contributions introduced in this paper are:

- Ontology Explorer, a new Visual Analytics system that leverages the concept of Ontologies in the interactive exploration of time series in digital twins;
- A case study that demonstrates an application of the Ontology Explorer in the oil and gas digital twins.

We organize the paper as follows. Section 2 summarizes past literature related to the research problem. Section 3 summarizes the requirements formulated during the development of the prototype. Section 4 presents the O3PO ontology used in this work. Section 5 describes the Ontology Explorer architecture, which includes the user interface, the Ontology API that supports the queries over the ontology issued by the Ontology Explorer user interface, the back end that supports queries to the ontology API, and the data representation module. Section 6 shows a use case with production data from a Brazilian off-shore production plant, followed by Section 7 that presents an evaluation of the proposal. Section 8 concludes the paper with a discussion of limitations and future steps of the project.

---

[3] www.inf.ufrgs.br/petwin/

[4] https://github.com/BDI-UFRGS/O3POntology

2. Related work

The oil production industry is a complex process that involves various stages, including exploration, drilling, production, and transportation. Time-series visualization tools can be used to monitor and analyze the data generated during these stages to improve the efficiency and safety of the offshore oil production industry [5].

Time-series visualization tools have the potential to effectively connect temporal data with the properties and components of an offshore oil-producing plant. However, there are limitations to these tools. For example, the optimization of oil production operations in offshore fields is based mainly on heuristic rules and simulation-like tools with limited optimization features [11]. Time-series visualization with the support of ontologies can potentially improve the data access and integration in the oil and gas industry. Ontology mapping and reasoning in semantic time series processing have been investigated, and a general time series ontology has been proposed to integrate new domain ontologies [3]. In addition, Semantic Web technologies have been used in the oil and gas industry to address problems such as information integration and knowledge management [2]. Visual computing technologies are also essential for software in the oil and gas industry, allowing users to gain insights and actionable information when dealing with increasingly complex, multidisciplinary datasets and processes [23].

An ontology-based visualization tool can offer several benefits to offshore oil production. Firstly, it can enhance the understanding of complex systems and relationships between various components involved in oil production. This is particularly important in offshore oil production, where the environment is harsh and complex, and the interactions between different components can be challenging to comprehend. Secondly, the tool can improve collaboration and communication between different stakeholders involved in oil production. The tool facilitates communication and enables stakeholders to work together more effectively by providing a shared understanding of the system and its components. Thirdly, an ontology-based visualization tool can aid decision-making by providing a comprehensive view of the system's components and interdependencies, helping to identify potential risks and opportunities and enabling stakeholders to make informed decisions regarding the production process.

Ontologies can be used in offshore oil production for data integration and knowledge management by providing a shared understanding of the concepts and relationships within the industry. For example, an ontology of offshore oil production processes and equipment can be used to facilitate data integration and knowledge management across different stages of the production process [18].

Other ontologies, such as ISO 15926 [6] or OntoCAPE [15], can provide a common language for stakeholders in the offshore oil production industry but focus on something other than time-series data visualization. By leveraging the domain's semantics through O3PO [18], the Ontology Explorer enables users to filter data based on business-related criteria. This eliminates the need for users to understand the data storage conventions when conducting a textual search. With the Ontology Explorer, users can instead navigate through the ontology taxonomy to locate the desired equipment type, select specific equipment components, and choose the properties they need to analyze. Additionally, the Ontology Explorer supports features such as comparing data across equipment and selecting a specific time interval for data analysis. Overall, this approach reduces cognitive load and streamlines the data retrieval process for offshore oil production industry users.

In our scenario, domain experts mainly use professional tools for their day-to-day needs. However, an ontology-based visualization tool in the offshore oil production industry provides a semantic approach to time-series data visualization. This approach allows the properties of the time-series data to be linked to the equipment and its components without needing to search for arbitrary file names. By providing a semantic representation of the data, the tool enables domain experts to understand the relationships between the data and the equipment and components that it represents.

3. Visual Analytics Requirements

Time series data frequently appear in DTs as collections of temporal data sampled from sensors. There is a vast literature on algorithms for analyzing time series data [1]. In particular, time series data plays a vital role in the petroleum industry in tasks such as oil forecasting [19] and anomaly detection in reservoir data [22], among others. The challenge that surfaces in DTs is coordinating access to an extensive collection of time series data corresponding to several types of physical properties placed at different locations and labeled with different tags.

While several applications in the oil and gas industry leverage the concept of ontology [16, 17], the focus is on data representation. We propose in this work that ontologies can be employed to ease the interactive exploration of DTs, which are heavily based on time series data. The motivation to access time series data by navigating through an ontology is the possibility of following semantically consistent classes, instances, parts, and properties instead of looking for files with arbitrary names in a myriad of data. Industry professionals have described this pain which is reasonably known in the industry [10].

The visual analytics interface of Ontology Explorer also has to enable accessing and visualizing time-series data without knowing the names of the files that store the data. For example, a user could start by selecting a particular well or equipment class, followed by the instance of interest or one or more particular components, and choose one or more properties for that artifact. While many works describe more elaborate ways to visually interact with ontologies [7, 12, 14, 20, 21, 24], in this work, we strive for a simple design that can be used to demonstrate the value of using an ontological model with data and defer to using more sophisticated interfaces in the future. Our architecture allows the evolution and expansion of the domain ontology while the interface dynamically reflects its actual state in associating the meaning to the data.

To support the goals outlined above, our research team realized design sessions and validated the prototype with domain experts participating in the PeTwin project. We enumerate below seven requirements (**R1**-**R7**) of the prototype Ontology Explorer:

- **R1**. Display the time series associated with sensor data as line graphs, supporting individual plots, or composed plots that allow comparisons;
- **R2**. Allow the user to define temporal filters to narrow the analysis to specific periods (months, days, and hours);
- **R3**. Display the taxonomy of the ontology, allowing the user to hide, expand and inspect certain paths from the root of the taxonomy;
- **R4**. Support a text query from the user that highlights in the taxonomy the class nodes that match the input query, summarizing at each node the count of children nodes returned by the query;

- **R5**. Support selection of class nodes and inspection of their instances;
- **R6**. Support navigation through semantically non-hierarchical associations such as components or properties;
- **R7**. Support selection of one or more properties and the display of composed line plots associated with the time series.

4. The O3PO ontology

The purpose of O3PO [18] is to provide a formal, standardized vocabulary for describing entities associated with an offshore petroleum production facility. The current ontology mainly covers wells and associated facilities, subsea installations, and all components that build the oil path between the reservoir and the platform. Thus, the ontology provides means to explicitly describe the meaning of the large volume of data inherent to modern offshore operations, which is collected by several companies during field production.

The main contribution of the ontology in petroleum data integration and interoperability comes from providing a backbone taxonomy for the domain composed of *rigid types* and *sortal types* (or *sortals*) [8]. A rigid type is a type that is essential for its instances (*i.e.,* an object cannot cease to be an instance of a rigid type without ceasing to exist). Therefore, the classification of objects into rigid types is immutable and represents the invariant aspects of the domain. A sortal is a type that provides a criterion of identity for its instances, i.e., a criterion for deciding, given two references to objects, whether they refer to the same or to distinct objects. Moreover, every object is an instance of a unique most general sortal that carries a criterion to assess its identity.

As a result, every domain element will be immutably classified in some category from this backbone taxonomy. This helps in anchoring and aligning the distinct models underlying data coming from various sources since any category in such underlying models will specialize or correspond to some category in the backbone taxonomy. For example, *well* would be a type in such backbone taxonomy. A well may be classified as a production well, an injection well, or an observer well, depending on the type of equipment that is installed on it and the function they execute. Wells may also be classified as active wells or shut-in wells, depending on the status of their gate valves. Then, we may have some data sources concerned with the functional character of wells, employing the former classification, and other sources concerned with the operational status of wells, using the latter classification. Those data sources rely on classifications that have no common category, which prevents the integration of their data, even though they implicitly describe entities of the same fundamental type, *i.e.,* wells. In this case, an ontology that provides a backbone taxonomy of the domain can help in the task, allowing it to make explicit that both data sources describe entities that are instances of the rigid sortal *well.*

O3PO supports the expressivity needed to account for routine tasks in the domain. It provides means to describe (1) the attributes that characterize an asset in the plant, (2) its functional decomposition, and (3) the hydraulic connections among pieces of equipment. Figure 1 exemplifies the use of these capabilities to model the instances and relationships that form the oil path in a generic offshore petroleum production facility. It depicts the structure that allows the flow of oil from a reservoir to a platform through a chain of fluid supply relationships (adapted from [13]). Moreover, it displays the partial decomposition of a well into some of its components and exemplifies some of the most common types of property that characterize the entities from the domain. This indicates the usefulness of O3PO to provide explicit semantics to the numerous similar configurations that may be described in the specifications of oil production environments.

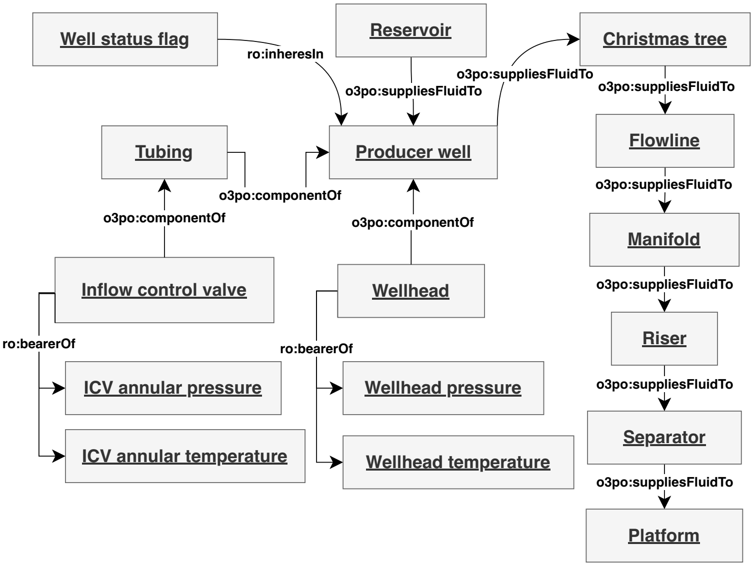


Figure 1. Diagram showing relations between instances in a subsea production system. Taken from [18].

Finally, to link time-series data to the properties of equipment in a plant, O3PO takes advantage of the Information Artifact Ontology (IAO) [4]. Each label, or tag, associated with time-series files is represented as an instance of the *IAO:Symbol* category and is linked via the *IAO:is about* relation to the property and the equipment to which the measurement data contained in the file refers. With that, the domain ontology is able to support the semantic search over the data, eliminating the need for a previous data transformation step or the engineer's deep knowledge about the data storage structure.

5. Ontology Explorer

A flexible web system with high usability is vital since business models are geared toward data-driven processes. We implemented the entire ecosystem of Ontology Explorerin an n-layer architecture, which effectively separates functionalities, responsibilities, and dependencies. This allowed better management of each component's modular change and maintainability. Each layer of this architecture runs on different servers, thus allowing better use of the machine's resources. Figure 2 illustrates the main components of the Ontology Explorerprototype: the user interface, backend, data module, and ontology module (described in the previous section). We detail each of these modules below.

5.1. User Interface

Ontology Exploreris implemented using the ReactJS library. It has been integrated with other data visualization libraries such as Nivo, D3.js, and HightCharts to allow users to

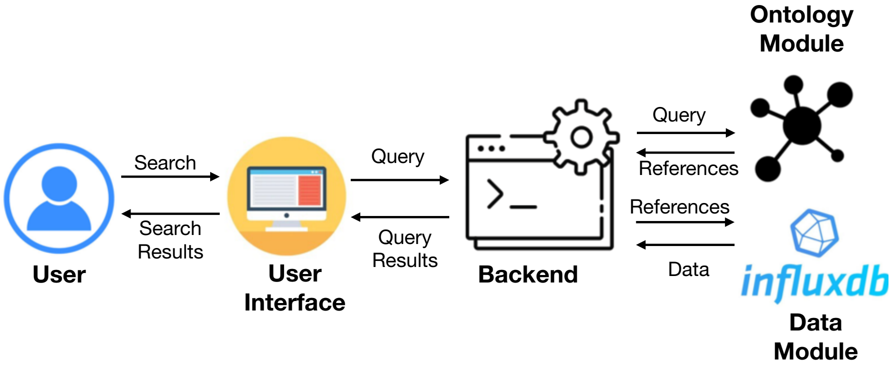


Figure 2. Ontology ExplorerArchitecture

interact according to their needs and explore time series data. The user interface design is guided by the seven requirements outlined in Section 3.

Figure 3 shows a screenshot of the system. The interface has multi-element containers (tabs) that support different views and interactions with the data. Figure 3(a) lists the tabs available, including a simple dashboard to display time series data, a tab to explore correlations and anomalies among the time series, a map explorer to show the geolocation of the wells, a tab to display soil properties such as porosity and permeability from Nuclear Magnetic Resonance (NMR) files, additional pressure values from observer wells, and finally the tab to interact with time series using the ontology. The contents in Figure 3(b)-(e) are specific to the Ontology Explorertab.

The interface is separated into four main areas. Figure 3(a) is the tab selector. Figure 3(b)(-d) is composed of interaction panels that use the ontology to explore the time series data. Figure 3(e) is an area dedicated to displaying and interacting with collections of time series. Finally, Figure 3(f) supports the specification of temporal filters.

The user interface supports the requirement **R1** of displaying the time series in other tabs of the interface and in the time series view of Figure 3(e). In this tab, miniatures of the time series are displayed, firstly grouped by the type of well (producer, injector, and observer) and secondly by the name of the well. When no filter is applied, a scrolling window shows the entire collection of time series miniatures. An enlarged time series plot is displayed by clicking on one of the miniatures plots. In this view, the user can also add other time series for comparison. Having aggregated analytical graphs gives users a more efficient alternative instead of doing it manually with non-specialized tools. The temporal filter box in the upper right corner of the interface (Figure 3(f)) supports requirement **R2** and allows narrowing the analysis to a specific time interval.

Performing comparisons between two or more time series within the ontology is a recurring need for analysts and one of the primary motivations for requirements **R3-R7**. The different views (Figure 3(b)-(d)) shows ways to interact with the Ontology API through the Owl-Ready2 library that represents the ontology and their relationships. The simplest way is to explore the hierarchical structure of the ontology taxonomy (Figure 3(c)), expanding or collapsing different branches until reaching relevant nodes (requirement **R3**). This is helpful when the user has prior knowledge of the taxonomy.

When the user is unfamiliar with the ontology or the ontology is too complex, the user can search instead for a concept by typing its textual form in a search box. We

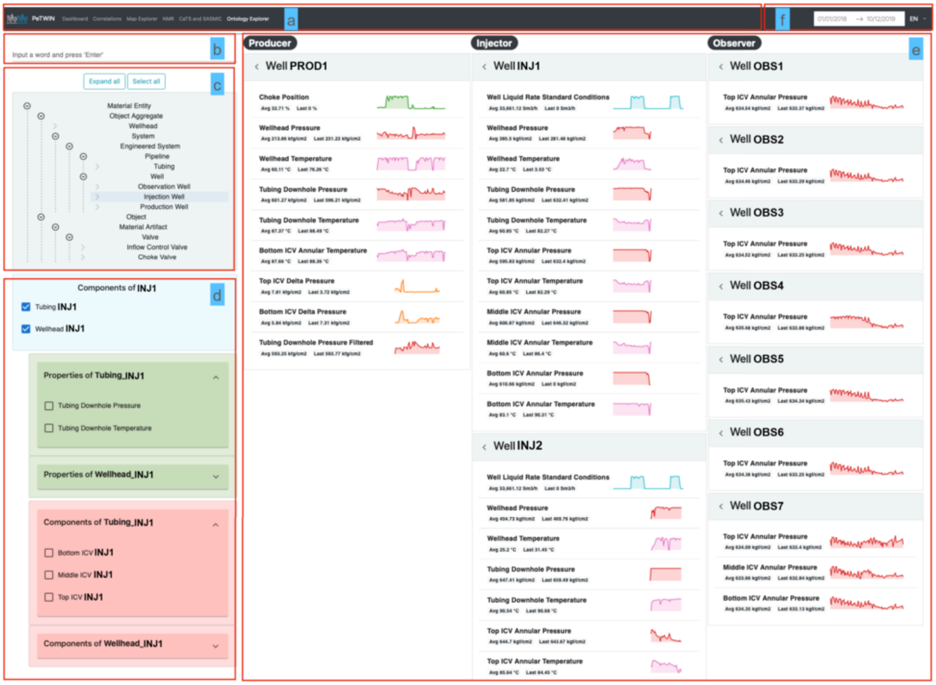


Figure 3. Ontology Explorer User Interface: (a) tab selector, (b) search box, (c) hierarchical navigation in the ontology taxonomy, (d) exploration of components and properties of selected instances, (e) time-series plots and (f) temporal filter.

illustrate this interaction in Figure 4. In this example, the user types the word *well* in the search box (Figure 4(a)). The taxonomy first appears collapsed at the root with the number of nodes that satisfy the query term (in this case, 5, as can be seen in Figure 4(b)). The following boxes (Figure 4(b)-(g)) show the results when the user clicks over nodes to open branches in the taxonomy until reaching the last level that exposes three types of wells (observation, injection, and production). This interaction satisfies requirement **R4**.

The next step in the exploration process is to select a class node and inspect its instances. Figure 5 shows a pop-up menu that shows the instances of a class node when the user hovers the mouse over the name of the class node in the ontology taxonomy. By clicking over the instance name, boxes appear below, showing the associated properties and components of the selected class node. This selection satisfies requirement **R5**.

The selection of an instance triggers a navigation process through the properties and components of class nodes. Figure 6 shows distinct navigation processes starting from instances of a producer, an injection, and an observation well. When an instance is selected, a box shows the list of its components. Each entry in this list has a checkbox that can be selected for further exploring the components and properties of the entry. In this example, we checked the Tubing component in each of the wells. An indented set of boxes shows the properties (green box) and components (red box) of the selected tubing component. This process is repeated to select the top and bottom ICV of PROD1, the bottom and middle ICV of INJ1, and the top ICV of OBS1. Checking the entries in the components list allows navigation through the non-hierarchical associations of

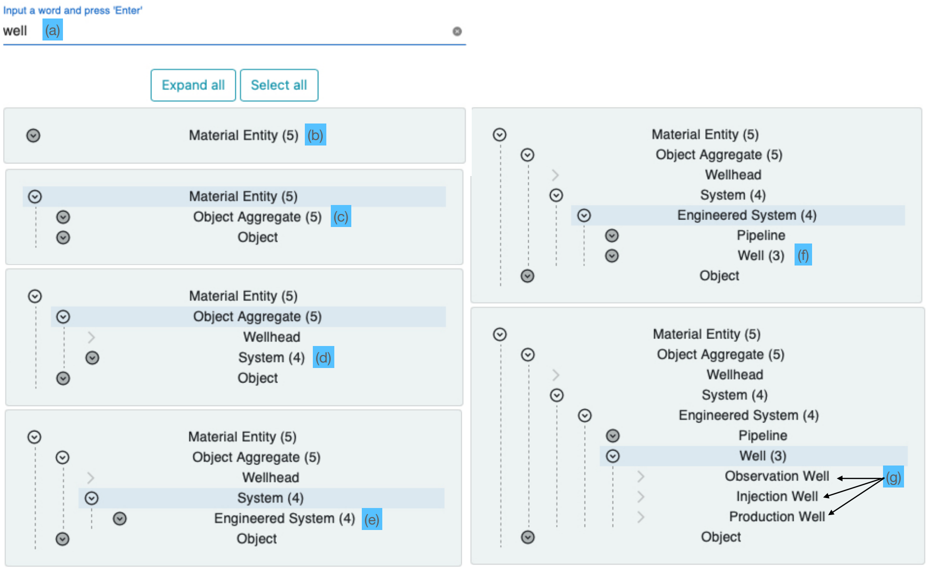


Figure 4. Finding relevant information using the ontology: (a) an input box allows the user to type the search term, in this case, *well*. The user expands the hierarchy following the locations where numbers indicate the existence of relevant results: (b) material entity, (c) object aggregate, (d) system, (e) engineered system, and (f) well. The last level (g) reveals the existence of three types of well to be explored (observation, injection, or production wells).

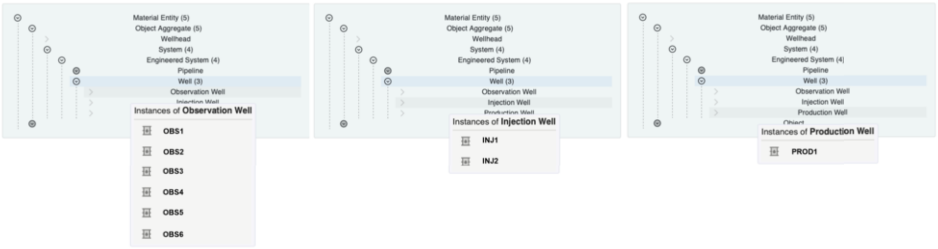
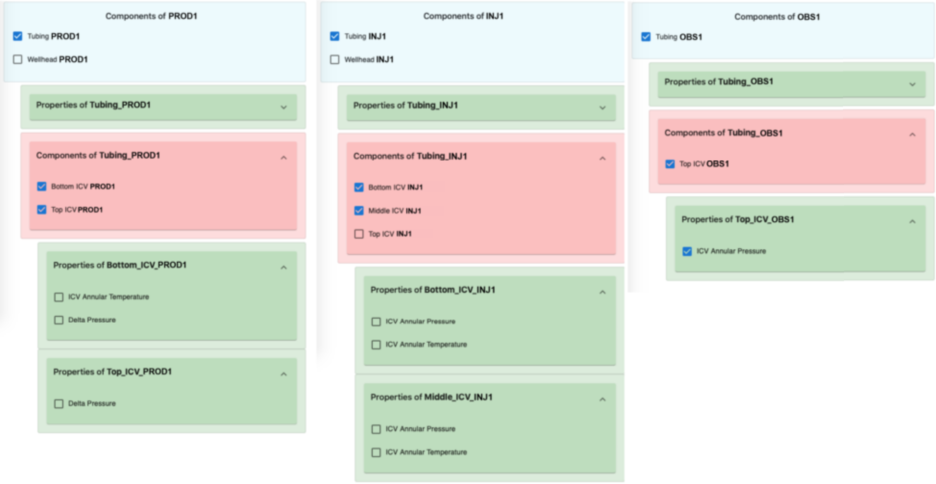


Figure 5. Instances associated with class nodes observation, injection, and production well.

the ontology (requirement **R6**). Subsequent boxes always appear below; the user can scroll the window to follow deeper navigation processes. When the boxes associated with one or more properties are selected, composed line plots are generated to show the corresponding time series, thus satisfying requirement **R7**.

5.2. *The Ontology API*

The Ontology API allows for semantic search on informational entities related to various components of a petroleum production facility by dynamically connecting the ontology with the analytic interface. The API can connect with any domain ontology with elements equivalent to BFO:material_entity, IAO:is_about, RO:has_quality, RO:quality_of and O3PO:has_component with minor adjustments.



Components of **PROD1**
☑ Tubing **PROD1**
☐ Wellhead **PROD1**
Properties of **Tubing_PROD1**
Components of **Tubing_PROD1**
☑ Bottom ICV **PROD1**
☑ Top ICV **PROD1**
Properties of **Bottom_ICV_PROD1**
☐ ICV Annular Temperature
☐ Delta Pressure
Properties of **Top_ICV_PROD1**
☐ Delta Pressure

Components of **INJ1**
☑ Tubing **INJ1**
☐ Wellhead **INJ1**
Properties of **Tubing_INJ1**
Components of **Tubing_INJ1**
☑ Bottom ICV **INJ1**
☑ Middle ICV **INJ1**
☐ Top ICV **INJ1**
Properties of **Bottom_ICV_INJ1**
☐ ICV Annular Pressure
☐ ICV Annular Temperature
Properties of **Middle_ICV_INJ1**
☐ ICV Annular Pressure
☐ ICV Annular Temperature

Components of **OBS1**
☑ Tubing **OBS1**
Properties of **Tubing_OBS1**
Components of **Tubing_OBS1**
☑ Top ICV **OBS1**
Properties of **Top_ICV_OBS1**
☑ ICV Annular Pressure

Figure 6. Navigation through properties and components of a given instance.

All individuals in the ontology, such as equipment, components, properties, and relations, are stored in an n-triples file, and specific time series are loaded when an instance is selected in the front end. The tags, which are names associated with the time-series files and information content entities, are manually mapped to instances of properties in particular equipment using the "is about" relation provided by IAO.

The backend uses the O3PO ontology and employs reasoning to provide access to both asserted and inferred data, allowing for complete data access. It is necessary to represent the production plant elements, their relationships, and tags in the N-Triples format. The ontology and particulars are used to infer information about the particulars.

To fulfill the requirements outlined before, the application includes these aspects:

- **Ontology Tree**: This component enables the entity's visualizations in hierarchical objects. This view gives the users information about the relations between objects in the hierarchy. Users can move over an individual entity, and a tool-tip control shows the instances of the entity on demand;
- **Components of instance**: This view shows components related to the instance selected in a tool-tip control of the (Fig. 4) and enables the capability of selecting one or more components simultaneously according to the users' need;
- **Properties of entity**: Properties of entities represent the attributes required for users and have specific information about the measurement (i.e., temperature, pressure, flow) in different well locations (i.e., top, middle, top). Users can select properties that satisfy their analysis, the information is obtained through the back-end service;
- **Components of component**: Some class instances have components that within their structure have other components; the back-end and front-end services support this structure. When the component is selected, a properties view is shown under that component;
- **Instances of entities**: Some classes have instances under them; This view allows for a quick search of specific instances; when the mouse hovers over the class name, the list of its instances is shown;

- **Information entities about an entity**: An information content entity is about another entity. Users can select an entity and retrieve the information content entities that refer to that specific instance;
- **Qualities of entity**: Qualities (`bfo:quality`) of entities represent the dependent attributes required for users and have specific information about an entity. Users can explore which qualities an entity has associated with it.

Using FastAPI [5], we created a prototype to offer endpoints for an interface application, OwlReady2, for ontological search. To explore the ontology, we used the OwlReady2 Python library[6] [9], as it supports the necessary operations for the semantic retrieval of information sources, providing the means to allow interactive navigation in the ontological structure for the end-user.

5.3. Data Warehouse and Backend Service

An essential component of a visual analytics ecosystem is storing information in a specialized data repository, allowing easy and fast access to the data in different formats. In this work, we use *InfluxDB*[7], a NoSQL database optimized for storing and processing time series information. It is an open-source tool, and its incorporation has been an essential factor for the Ontology ExplorerVisual Analytics platform because it supports efficient access to times series data through customized and specialized queries.

The client layer – the front end, publishes information to the user according to the queries or filters in the web platform. It also presents controls that allow the reduction or expansion of the scope of information required for each user's need. The backend layer is responsible for fetching information from the data warehouse by performing the necessary processing or business logic to be sent to the upper layer while supporting the user interaction with the front-end layer.

Currently, the techniques available to interact between a data warehouse and a web application have significantly evolved, allowing a quick and easy integration. Our backend service has business logic following what is required by users and thus exposes personalized information. The implementation follows recent technology that enables the fastest integration with many cloud or on-premise providers.

Pooling functions require a higher level of processing, and the backend services can offer this capability since these services are typically hosted on servers with high processing power in specialized computer architecture. All the services exposed in the backend service are processed by the authentication and authorization mechanism based on JWT technology, which requires that each request be authenticated.

6. Use case application of the Ontology Explorer

To demonstrate the usability of the Ontology Explorerin a practical scenario, we devised a use case that reflects common tasks in an engineer routine: the investigation of injection loss and the estimation of reservoir connectivity. Those cases were constructed based on a dataset[8] from an anticipated production system (SPA) in an offshore oil field in Brazil's

---

[5] https://fastapi.tiangolo.com/

[6] https://owlready2.readthedocs.io/en/v0.37/contact.html

[7] https://www.influxdata.com/

[8] The data comes from the PeTwin project –`https://www.inf.ufrgs.br/petwin/`

Pre-salt. The dataset consists of a collection of CSV files containing time-series data. They consist of measurements of physical properties such as pressure, temperature, and flow rates, colected by sensors placed in various locations and equipment that compose the plant. It mainly comprises data about production wells (*i.e.,* intended to produce hydrocarbons from a reservoir), injection wells (*i.e.,* intended to inject water or gas into the reservoir), and observation wells (*i.e.,* intended to monitor reservoir conditions and collect data). Due to confidentiality constraints, we cannot give more details of the data.

An important problem that arises in oil and gas exploration is *injection loss*, which refers to the event in which a volume of injected fluid does not reach the target zone in a reservoir but instead is lost in the surrounding rock formations. It can be caused by, among other things, low reservoir quality, flawed wellbore integrity, and inadequate fluid viscosity. It is a relevant event because minimizing injection loss is crucial for effective oil and gas production and can be accomplished by wellbore integrity management, enhanced fluid injection techniques, and numerical simulation models.

Now, suppose that an engineer wants to investigate if there is any injection loss for a specific well (INJ1). The particular interest is in the time interval between May 1st and 31st of 2018. After selecting the start and end dates, the tool permits navigating the taxonomy present in the ontology until he finds the injection well class (figure 4 shows the sequence of steps to achieve that). By selecting the desired instance of the injection well class (figure 5), the Ontology Explorershows the components of the well (as seen in figure 7). After choosing a particular component, the tool presents the list of properties that characterize the selected component as well as the list of objects that compose it. The user can navigate between the parts and properties of the selected instances.

To accomplish the task of investigating the injection loss, the engineer has to compare the annular pressures of all inflow control valves (ICVs) (i.e., valves used to regulate the flow that enters the well from the reservoir) in the well INJ1. Thus, the user chooses all ICVs that are part of INJ1 and selects the annular pressures associated with each of them. The interface automatically displays the time series corresponding to the desired properties. An aggregate plot is shown so the engineer can analyze the pressure behavior of the valves in the well. It's possible to select a shorter time interval (*e.g.,* May 10th to 13th) to investigate the pressure behavior of the valves in further detail. Figure 7 also shows an illustrative picture of the annular pressure aggregate plot.

7. Tool evaluation

Based on the presented requirements, the proposed time-series visualization tool that utilizes a domain ontology to connect data with equipment properties in an oil and gas plant is deemed to be satisfactory. The tool provides an intuitive and interactive interface for users to visualize time-series data and correlate it with the properties of equipment in the plant. It effectively utilizes the domain ontology to establish relationships between the data and equipment properties, enabling users to easily analyze and interpret the data. Furthermore, the tool allows for the customization of visualizations and provides features such as data filtering and drill-down, which enhance the user experience and enable them to gain insights into the data at a granular level. Overall, the proposed tool satisfies the requirements presented and is expected to be a valuable asset for data analysis and decision-making in the oil and gas industry.

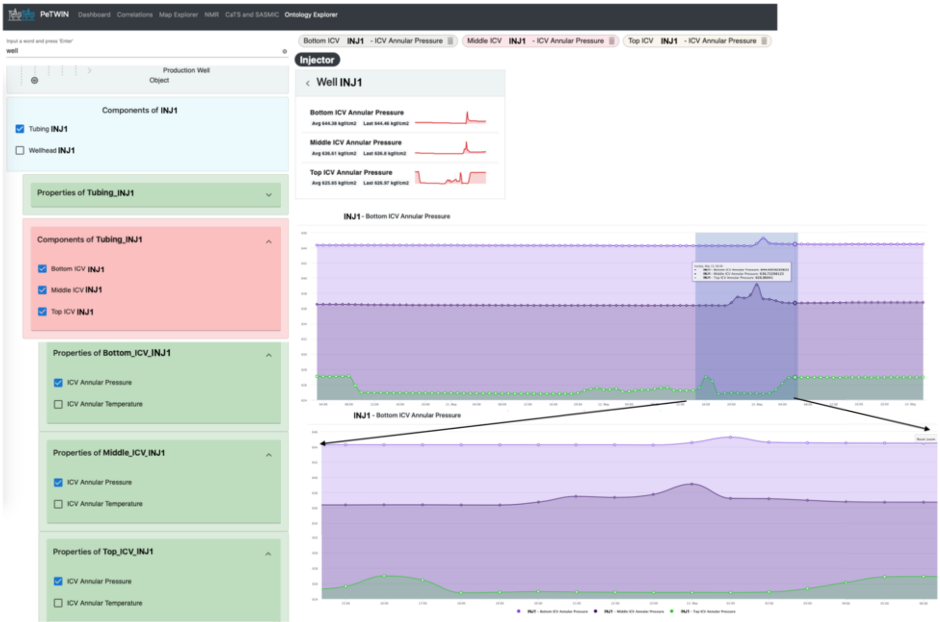


Figure 7. Exploring properties.

It should be noted that the current application interface requires users to have some understanding of ontological terms such as ”Material entity” and ”Object Aggregate.” However, the interface is still quite usable even for those who are not experts in ontology.

An important consideration for the tool’s future development is the need to redesign the interface to enable visualization of a larger number of different time series. Currently, the tool employs a list-based visualization approach that may not scale well for very large amounts of series elements. Therefore, it would be beneficial to explore other visualization techniques that can effectively display a higher volume of time-series data without overwhelming the user with too much information.

8. Conclusion

This paper presents Ontology Explorer, a visual analytics system for time series data in the oil and gas field, enhanced by the O3PO ontology to enable semantic interactive exploration of data.

Leveraging the semantics of the domain, the tool allows users to filter data in terms that closely reflect the core business, relieving the extra burden of thinking about the way information is stored.

In Ontology Explorerthe users do not have to figure out what might be the labeling convention used to describe data in order to retrieve it in a usual textual search. Instead, they can navigate through the taxonomy of the ontology to find the type of equipment in which they are interested, choose the particular pieces of equipment – or components of such equipment – they want to inspect and select the properties to analyze. The system offers other functionalities, such as comparing data from different equipment and choosing a particular time interval for data analysis.

OntologyExplorer was designed to integrate the development of a digital twin for the oil and gas domain in the context of the PeTwin project.

Given its modular architecture, the tool could fit other domains in which the analysis of time series data is required, basically requiring the replacement of O3PO by an ontology for the specific domain that uses BFO as a top-level ontology and has properties that the time series can be about.

Future work will focus on the ontological treatment of the types of events whose signature is typically implicit in the temporal data we are working on (*e.g.,* the event of injection loss), aiming to provide a similar semantic exploration of them. We plan to release a general version of the tool, allowing users to plug in their ontologies, provided they meet certain requirements, along with necessary documentation. Also, regarding the larger landscape of the PeTwin project, the prototype has been validated and is being installed for evaluation and improvement at the corporate partner. Additionally, we will work towards integrating the Ontology Explorerwith machine learning methods to extract patterns from production data.

Acknowledgments

This work was developed in the Petwin Project (PeTWIN.org) scope, supported by the Brazilian Federal Agency for Innovation (FINEP) and the Libra consortium for petroleum production. It is also partially financed by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

References

[1] Mohammed Ali, Ali Alqahtani, Mark W. Jones, and Xianghua Xie. Clustering and classification for time series data in visual analytics: A survey. *IEEE Access*, 7:181314–181338, 2019.

[2] Idoia Berges, Víctor Julio Ramírez-Durán, and Arantza Illarramendi. Facilitating data exploration in industry 4.0. volume 11787 LNCS, 2019.

[3] Bojan Božić and Werner Winiwarter. Ontology Mapping and Reasoning in Semantic Time Series Processing. In *Proceedings of International Conference on Information Integration and Web-based Applications & Services*, IIWAS '13, pages 443–452, New York, NY, USA, December 2013. Association for Computing Machinery.

[4] Werner Ceusters and Barry Smith. Aboutness: Towards foundations for the information artifact ontology. In *Proceedings of the Sixth International Conference on Biomedical Ontology*, pages 1–5. CEUR vol. 1515, 2015.

[5] Jaqueline B. Correia, Fabrício Rodrigues, Nicolau Santos, Mara Abel, and Karin Becker. Data Management in Digital Twins for the Oil and Gas Industry: Beyond the OSDU Data Platform. *Journal of Information and Data Management*, 13(3), September 2022.

[6] Leal D. Iso 15926 "life cycle data for process plant": an overview. *Oil & Gas Science and Technology - Rev. IFP*, 60:629–637, 2005.

[7] I. Siqueira da Silva and C. Dal Sasso Freitas. Using visualization for exploring relationships between concepts in ontologies. In *2011 15th International Conference on Information Visualisation (IV 2011)*, pages 317–322, Los Alamitos, CA, USA, jul 2011. IEEE Computer Society.

[8] Nicola Guarino and Christopher A Welty. An Overview of OntoClean. In Steffen Staab and Rudi Studer, editors, *Handbook on Ontologies*, pages 151–171. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[9] Lamy Jean-Baptiste. *Ontologies with Python: Programming OWL 2.0 Ontologies with Python and Owlready2*. Apress Berkeley, 2021.

[10] Evgeny Kharlamov, Martin Skjæveland, Dag Hovland, Theofilos Mailis, Ernesto Jimenez-Ruiz, Guohui Xiao, Ahmet Soylu, Ian Horrocks, and Arild Waaler. Finding data should be easier than finding oil. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 1747–1756. IEEE, 2018.

[11] Vassileios D. Kosmidis, John D. Perkins, and Efstratios N. Pistikopoulos. Optimization of well oil rate allocations in petroleum fields. *Industrial & Engineering Chemistry Research*, 43(14):3513–3527, 2004.

[12] Simone Kriglstein. Owl ontology visualization: Graphical representations of properties on the instance level. In *2010 14th International Conference Information Visualisation*, pages 92–97, 2010.

[13] Ville Kukkonen, Ali Kücükavci, Mikki Seidenschnur, Mads Holten Rasmussen, Kevin Michael Smith, and Christian Anker Hviid. An ontology to support flow system descriptions from design to operation of buildings. *Automation in Construction*, 134:104067, 2022.

[14] Vincent Link, Steffen Lohmann, Eduard Marbach, Stefan Negru, and Vitalis Wiens. WebVOWL: Web-based Visualization of Ontologies. http://vowl.visualdataweb.org/webvowl.html, last accessed July 2022.

[15] Wolfgang Marquardt, Jan Morbach, Andreas Wiesner, Aidong Yang, and Onto CAPE. *OntoCAPE - A Re-Usable Ontology for Chemical Process Engineering*. Springer-Verlag Berlin Heidelberg, 2010.

[16] Shastri L Nimmagadda, Andrew Ochan, Neel Mani, and Torsten Reiners. Big Data guided Digital Petroleum Ecosystems for Visual Analytics and Knowledge Management. In *Proc. Australasian Exploration Geoscience Conference (AEGC)*, 2021.

[17] Shastri L Nimmagadda, Andrew Ochan, Christine Namugenyi, and Torsten Reiners. Data geo-Science Approach for Modelling Unconventional Petroleum Ecosystems and their Visual Analytics. In *Proc. Australasian Exploration Geoscience Conference (AEGC)*, 2021.

[18] Nicolau O. Santos, Mara Abel, Fabrício Henrique Rodrigues, and Daniela Schmidt. Towards an ontology of offshore petroleum production equipment. In Emilio M. Sanfilippo, Mohamed-Hedi Karray, Dimitrios Kyritsis, and Arkopaul Sarkar, editors, *Proceedings of the 12th International Workshop on Formal Ontologies meet Industry (FOMI 2022) Co-located with workshops about the Industrial Ontology Foundry (IOF) and the European project OntoCommons (EU H2020 project), Tarbes, France, September 12-15, 2022*, volume 3240 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2022.

[19] Leonid B. Sheremetov, Arturo González-Sánchez, Itzamá López-Yáñez, and Andrew V. Ponomarev. Time series forecasting: Applications to the upstream oil and gas supply chain. *IFAC Proceedings Volumes*, 46(9):957–962, 2013. 7th IFAC Conference on Manufacturing Modelling, Management, and Control.

[20] Isabel Cristina Siqueira Silva, Giuseppe Santucci, and Carla Maria Dal Sasso Freitas. Visualization and analysis of schema and instances of ontologies for improving user tasks and knowledge discovery. *Journal of Computer Languages*, 51:28–47, 2019.

[21] Nine Points Solutions. OntoGraph. https://github.com/NinePts/OntoGraph, last accessed July 2022.

[22] Aurea Soriano-Vargas, Rafael Werneck, Renato Moura, Pedro Mendes Júnior, Raphael Prates, Manuel Castro, Maiara Gonçalves, Manzur Hossain, Marcelo Zampieri, Alexandre Ferreira, Alessandra Davólio, Bernd Hamann, Denis José Schiozer, and Anderson Rocha. A visual analytics approach to anomaly detection in hydrocarbon reservoir time series data. *Journal of Petroleum Science and Engineering*, 206:108988, 2021.

[23] Mario Costa Sousa, Emilio Vital Brazil, and Ehud Sharlin. Scalable and interactive visual computing in geosciences and reservoir engineering. *Geological Society, London, Special Publications*, 406(1):447–466, 2014.

[24] Y. Yang, M. Wybrow, Y. Li, T. Czauderna, and Y. He. Ontoplot: A novel visualisation for non-hierarchical associations in large ontologies. *IEEE Transactions on Visualization & Computer Graphics*, 26(01):1140–1150, jan 2020.