

# Connecting Multilayer Semantic Networks to Data Lakes: The Representation of Data Uncertainty and Quality

E. Cayeux<sup>1\*</sup>, C. Damski<sup>2</sup>, J. Macpherson<sup>3</sup>, M. Laing<sup>4</sup>, P. Annaiyappa<sup>5</sup>, P. Harbidge<sup>6</sup>, M. Edwards<sup>7</sup>, and J. Carney<sup>8</sup>

<sup>1</sup>NORCE

<sup>2</sup>Genesis Petroleum

<sup>3</sup>Baker Hughes

<sup>4</sup>Halliburton

<sup>5</sup>Nabors

<sup>6</sup>PathControl

<sup>7</sup>Edwards Energy Innovation Consulting

<sup>8</sup>Schlumberger

## Summary

Drilling oil and gas wells is a complex process involving many disciplines and stakeholders. This process occurs in a context where some pieces of information are unknown, or are often incomplete, erroneous, or at least uncertain. Yet, during drilling engineering and construction of a well, drilling data quality and uncertainty are barely addressed in an auditable and scientific way. Currently, there are few or no placeholders in engineering and operational databases to document uncertainty and its propagation. The Society of Petroleum Engineers (SPE) has formed a cross-disciplinary technical subcommittee to investigate how to describe and propagate drilling data quality and uncertainty. The subcommittee is a cooperation between the drilling system automation, wellbore positioning, and drilling uncertainty prediction technical sections. As the topic is vast and complex, a systematic method was adopted, where multiple user stories or pain points were generated and ranked with the most compelling user story analyzed in detail. From this approach, a series of multidisciplinary workflows (drilling data generators) can now be captured and described in terms of data quality and propagation of uncertainty. The paper presents details of one user story focused on capturing the description of the quality and uncertainty of depth measurements. Multiple use cases have been extracted from this single user story exemplifying how multiple stakeholders and disciplines manage, communicate, and understand the notion of wellbore depth and its relative uncertainty. Current data stores have the main objective of recording the results of processes but have very limited capabilities to store how the interdisciplinary processes generated and cross-related these results. The study explores the use of semantic networks to capture those multidisciplinary data relationships. A minimum vocabulary has been created using just a few tens of concepts that has sufficient expressiveness to describe all the extracted use cases, showing that the semantic network method has the potential to describe a broad range of complex drilling-related processes. The study also demonstrates that use of a multilayered graph, employing other notions that do not expressly refer to the processes that generated the data, can capture the description of how uncertainty propagates between each of those concepts.

## Introduction

Decision-making in well operations requires a thorough understanding of risk—understanding the possibility that, at some time in the future, an actual “happening” will be different from a predicted “happening”. The Cambridge Advanced Learner’s Dictionary defines risk as “the possibility of something bad happening.” Understanding risk requires measurement of risk and the ability to weigh its consequences. Once understood, it is possible to manage, prevent, and mitigate risk.

In drilling data measurement systems, we are interested in quantifying the risk associated with some original or derived piece of data, for example, the measured depth (MD) of a well. There is some quantifiable level of uncertainty in this piece of data. When combined with other uncertain data, that uncertainty propagates through the interconnected measurement system and directly affects decision-making risk.

Errors in measurements, in the sense of statistics, can be caused by systematic errors (bias) and random errors (Taylor 1997). Systematic error is the quantified average difference between a set of measurements with a reference value: It characterizes the accuracy of the measurement. Random errors are a characteristic of the precision of the measurement and tell whether multiple measurements agree with each other. Uncertainty is the quantitative estimation of error present in data (i.e., how much a value may depart from its true value at a given confidence level). It is typically expressed as a standard uncertainty (i.e., the positive square root of the estimated variance Taylor and Kuyatt 1994), at least for normal probability distributions of measurement errors (Bailey 2017).

The concept of data quality is more diffuse than for data uncertainty, and it may have different meanings depending on perspectives (for example, consumer, business, or standards-based perspectives) or disciplines (for example, software engineering or control theory disciplines). Note that here, perspective and point of view are considered equivalent. In practice therefore, data quality is not a characteristic that exists independent of context; it must be evaluated from the point of view of the process or business. Moges et al. (2013) describes 17 different characteristics of data quality while Chu et al. (2001) uses 13 different properties. Here are some of the most common data quality characteristics:

\*Corresponding author; email: eric.cayeux@norceresearch.no

Copyright © 2023 The Authors.

Published by the Society of Petroleum Engineers. This paper is published under the terms of a Creative Commons Attribution License (CC-BY 4.0).

This paper (SPE 208754) was accepted for presentation at the IADC/SPE International Drilling Conference and Exhibition, Galveston, Texas, USA, 8–10 March 2022, and revised for publication. Original manuscript received for review 13 April 2022. Revised manuscript received for review 23 July 2022. Paper peer approved 15 August 2022.

- **Completeness:** Does the data set contain all the relevant information so that there is no need for additional information? (Pipino et al. 2002)
- **Validity:** Are the data valid with regards to the specific domain? (Askham et al. 2013)
- **Accuracy:** Does the data represent accurately the real-world and confirm verifiable sources? (Ballou and Pazer 1985)
- **Consistency:** Does information coming from multiple sources match? (Cong et al. 2007)
- **Timeliness:** Are the data used in a process or business logic up-to-date and still relevant? (Simard et al. 2019)
- **Reasonableness:** Can the data be used as part of a logic inferencing scheme? (Tayi and Ballou 1998)
- **Conformity:** Does the data conform to specifications or existing standards? (Wang et al. 1993)

The SPE industry group in 2020 created the subcommittee Drilling Data Quality and Uncertainty Description (DDQUD). Its members came from three SPE technical sections—the Drilling Systems Automation Technical Section (SPE-DSATS 2022), the Wellbore Positioning Technical Section (SPE-WPTS 2022; ISCWSA 2022), and the Drilling Uncertainty Prediction Technical Section (SPE-DUPTS 2022).

The scope of work of DDQUD is quite straightforward: Increase awareness and understanding in the drilling community of drilling data quality and uncertainty and, in essence, enable a consistent description of data quality and uncertainty of key drilling data.

The approach taken by DDQUD is inspired by the agile software development method (Beck 2000; Cohn 2004) and utilizes “user stories” to capture some specific requirements that shall be addressed. The utilized method was to (a) create a list of user stories along with their associated key drilling data, (b) rank the criticality (importance) of the user stories, (c) break down the most critical user stories into “use cases,” and (d) develop a method to describe uncertainty and quality for the use cases.

This last task delves into the very rich and fruitful topic of data modeling (e.g., data and knowledge representation, semantic networks, and multilayered graphs). This discussion leads to a description of the propagation of uncertainty in a data network. The use of propagation graphs and semantic networks could have far-reaching consequences on the way in which drilling data are accessed and stored.

For now, the mission is to develop a method that will result in increased confidence in understanding and describing data quality and uncertainty during well operations. This will result in improved overall decision-making related to the identification of risk and associated mitigation plans.

**Method to Select a User Story.** The task of describing exhaustively the quality and uncertainty of drilling data is overwhelming. Yet, if there is a generic way to solve the description of drilling data quality and uncertainty, it may not be necessary to examine all possible cases that are relevant in the drilling context. As described above, the approach was to collate a set of user stories with their key drilling data, rank them in terms of criticality, and select the most critical for further study. As each user story contain elements that overlap with other user stories, the methods developed will be applicable in a generic fashion.

**User Stories.** In the search of a generic solution to describe drilling data quality and uncertainty, it would be useful to have a relevant example that could facilitate making sense and communication of the otherwise rather dry and abstract concepts utilized in software engineering. Beck (2000) introduced the notion of a user story (Wikipedia and User-Story 2022) in the planning game when working on the extreme programming development method. In specifying a desired system, a user story is a description made by an end user, in natural language, of some important features of the system. In our case, the system is a constructed framework that can help in describing drilling data quality and uncertainty, and the end users are engineers involved in the drilling process, whether it is at the planning, execution, production, or abandonment phase. Multiple disciplines may be involved in the scenarios described by these user stories, including, but not limited to, drilling, geophysics, geology, reservoir engineering, production engineering, and completion engineering.

Many working groups, including this DDQUD group, have cooperated in capturing more than a hundred user stories. These collected user stories are in a document that is available to the drilling community on the website (de Wardt 2022) of the Drilling System Automation Body of Knowledge. This living document expands continuously with newly provided inputs.

**Definition of a Use Case.** A use case is a list of actions or events or steps typically defining the interactions between an actor and a system to achieve a goal. The concept of use cases in software development has been introduced by Jacobson to capture and specify the requirements of a system when utilizing object-oriented development methods (Jacobson 1993; Cockburn 2002), and it has been retained as an integral part of agile software development methods (Abrahamsson et al. 2017).

Use cases are a technique for capturing, modeling, and specifying the requirements of a system. A use case corresponds to a set of behaviors that the system may perform during interaction with its actors and which produces an observable result that contributes to its goals. Actors represent the role that human users or other systems have in the interaction.

The subject and goals define the scope of a use case. The subject identifies the system, subsystem, or component that will provide the interactions. Goals can be hierarchical, taking into account the organizational level interested in the goal and the decomposition of the user’s goal into subgoals. The decomposition of the goal is from the point of view of the users, and independent of the system, which differs from traditional functional decomposition. An example of the standard format of a use case is in **Table 1**.

Title (Goal)	The Goal Resulting from the Objectives
Primary actor	The primary actor defines the role the users and/or other systems have in the interaction.
Stakeholders	Roles affected by the goal.
Scope	The subject and its goals define the scope of each use case. In the example user story, the generic scope was the rounding of MD values (see next section), from which many use cases could be extracted, which could be further broken down into specific subgoals.
Level	Organizational level affected by the goal.
Brief	Short, informal description of what happens, with graphics if possible.

Table 1—A typical template to describe a use case.

By breaking down a user story into individual use cases, one is able to capture, model, and specify the requirements of a system. In the context of drilling data quality and uncertainty, the use case technique enables us to extract a set of behaviors the system may have and observe how actors, be they human or other systems, interact to produce an observable result that contributes to the set goal. From this characterization, we can pivot into a more abstract knowledge representation to demonstrate relationships in drilling data quality and uncertainty.

**An Example User Story.** The collected user stories have an incredibly variable content. Some are of a very general nature; others are very specific to a particular context. Therefore, the next question addressed was to select a user story that would have great potential in the context of DDQUD, and that would be not too difficult to solve.

The highest ranking user story selected to illustrate drilling data quality and uncertainty dealt with the rounding of MD values:

Tabulation of well survey data is typically to two decimal places when the uncertainty, at deeper depths, is in whole meters/feet and multiples thereof. This is misleading to the end user and subsurface modelling (especially cross correlation of pressures) suffers badly. A common representation of uncertainty on shared measured depth (MD) and true vertical depth (TVD) data will help ensure that end users take into account the real uncertainty in the data.

Note that this is User Story 20 in de Wardt (2022).

As implied by the user story, the uncertainty associated with MDs and TVDs are not always understood outside the directional drilling discipline. The multiple sources of uncertainties for MD have been analyzed (Brooks et al. 2005) while correlation methods for depth correction utilizing formation logging have been studied (Dashevskiy et al. 2008; Poedjono and Nwosu 2019). Note that Brooks and Wilson (1996) indicate possible errors in MD in the range of 4 to 8 m for a 5000-m long well. The uncertainty of TVD is directly related to wellbore position uncertainty evaluation methods. The estimations of wellbore position uncertainties have been developed over several years. Initial estimations made with the assumption of random measurement noise (Walstrom et al. 1969) have been superseded when realizing that systematic errors on the inclination and azimuth measurements lead to more realistic estimations of the wellbore position uncertainty (Wolff and de Wardt 1981). Multiple contributions (Grindrod and Wolff 1983; Dubrule and Nelson 1987; Thorogood 1990; Brooks and Wilson 1996; Ekseth 2000) have led to the current standard models for magnetic survey instruments (Williamson 2000) and for gyroscopic tools (Torkildsen et al. 2008). With magnetic surveys, the TVD uncertainty may be in the range of 20 m for an 8000-m long well (Williamson 2000).

This example of a user story breaks down into many use cases that express the variability in measurements of MD, how TVD is estimated, and how both are managed and communicated—and the relative uncertainty that may propagate from this point through any interconnected measurement system. Here is a list of 16 possible use cases, among others, generated from the user story:

1. Convey TVD uncertainty with reference to drillstring environmental corrections
2. Convey TVD uncertainty with reference to MD
3. Convey which datum is used for depth measurements
4. Convey TVD uncertainty in logging depth
5. Convey TVD uncertainty during wellbore intersection
6. Convey geological target boundaries as constraints for a surveying program
7. Convey requirements for wellbore position uncertainty associated with formation tops
8. Perform adequate quality assurance/quality control on wellbore surveying procedures
9. Convey wellbore position uncertainty relative to the last observed geological marker
10. Convey depth uncertainty when approaching a blowout well
11. Convey depth uncertainty with regard to potential collisions with other wells
12. Convey TVD depth uncertainty when surveying with an inclinometer
13. Correcting for the possible effect of time on certain depth data
14. Convey TVD uncertainty in reference to ground level
15. Convey TVD depth uncertainty for true vertical thickness from planned well path
16. Convey TVD depth uncertainty with associated geopressure gradients

The selected user story and some of its above listed use cases will serve to illustrate the derivation of a framework to describe the propagation of uncertainty in drilling data.

## Framework for Enabling Computer Interpretable Data Models

**Data and Knowledge Representation.** To fulfill the requirements of describing data quality and uncertainty, we need to look into a more general approach for knowledge representation. Data uncertainty and quality are meta-knowledge (Wikipedia and Meta-knowledge 2022). They describe the data and are not represented or stored at the same level as the primary data. They may also have some subjectivity, as the quality of the data depends to some extent on the purpose and use of the data.

To create meta-knowledge, we have to start defining entities (classes or objects as they are also known) which represent an abstraction of a small part of the primary data. Classes and objects, or instances of classes, are central concepts of object-oriented programming. The initial notions behind object-oriented languages were developed during the early days of research on artificial intelligence during the 1960s with programming languages such as LISP (McCarthy 1978) and Simula (Nygaard and Dahl 1978) and in the 1970s with Smalltalk (Kay 1996). Nowadays, most modern programming languages such as C++ (Stroustrup 2013), Java (Gosling et al. 2000), C# (Hejlsberg et al. 2008), and Python (Van Rossum and Drake 2010) use the concepts of object-oriented programming. This general approach is suitable for physical objects as well as conceptual properties. It fits well with our purpose for general knowledge representation and reasoning.

This investigation caused us to look more profoundly into epistemological issues in the drilling domain:

1. Source or nature of physical measurements  
For example, if a “bit depth” value is required, how do we actually measure it? What are the components of those measurements? How do we acquire the data?
2. The source of this knowledge  
Is the source of data or derived knowledge based on an instrument, calculation, observation, indirect measurement, etc.?

3. The structure or body of knowledge  
How do the components depend on each other? For example, how is the weight-on-bit measurement related to rock properties in the instantaneous rate-of-penetration calculations?
4. Subjectivity of this knowledge  
Regarding data quality, the source of data, primary or derived, can be subjective. It depends on how it was acquired or in which context it is used. The MD can be seen as derived data, from the driller's point of view, as it is based on a stand count, stand lengths, and current block position, or it can be seen as primary data in the perspective of the directional drilling engineer, when calculating the wellbore position using the minimum curvature method (Sawaryn and Thorogood 2005).

Other sections of this paper describe these issues. We identified these intertwined issues when we developed user stories as described in the section An Example User Story, which shows how the data flow between different systems and different experts.

In practical terms, we understand that the primary drilling data are stored in databases, files, data lakes (Fang 2015), eXtensible Markup Language (XML) stores (Bray et al. 2000), etc. This is the storage of what has been measured or calculated in the field, hereafter called “primary data.” As we needed to shift to a higher (abstract) level to discuss the quality and uncertainty of this primary data, we decided to use a hierarchical level of semantic network representation. This would allow us to represent and reason independently of how the primary data are stored.

For example, a trajectory of a well is an entity that is representable by a few attributes, such as MD, inclination, azimuth, TVD, and dogleg severity (DLS), etc. However, survey instruments measure a series of inclinations and azimuths which, when combined with the corresponding MD, are used to calculate the central axis of the wellbore, often referred to as a trajectory, and in which precisions and accuracies may be the cause of the uncertainty in those measurements. The information about the survey instrument might not be in the same data repository or not even recorded at all. The wellbore entity contains this trajectory, so it also inherits this uncertainty, as does the whole well. The wellbore position uncertainty might not belong to a set of critical data, unless there are other wellbores nearby and there is a need to assess the possible collision or interference between them. In this case, the purpose of data usage changes the need for measurement of uncertainty. Importantly, this uncertainty is not observable or measured unless we have a meta-knowledge to explain it.

Among many other examples, this shows the weakness of the current data-oriented representation to address data quality and uncertainty. In the next section, we will describe in detail the proposed framework to address this weakness.

**Semantic Network.** As explained earlier, we need to have a more flexible and abstract knowledge representation to address the relationship of drilling data uncertainty and quality. We chose a semantic network for this purpose. A semantic network is a graphical method, also known by other names in the literature such as knowledge graph, semantic web, etc. This method has been chosen to represent relationships in data, due to its application by Google, Facebook, Amazon, and many other companies who deal with unstructured and human-friendly information..

According to Wikipedia, a semantic network is “a network that represents semantic relationships between concepts. It is a directed or undirected graph consisting of vertices, which represent concepts, and edges, which represent semantic relations between concepts” (Wikipedia and Semantic-network 2022). The animal kingdom example in Fig. 1 shows characteristics similar to the drilling domain, where concepts can have many different relationships with one another in a nonhierarchical and multifaceted manner.

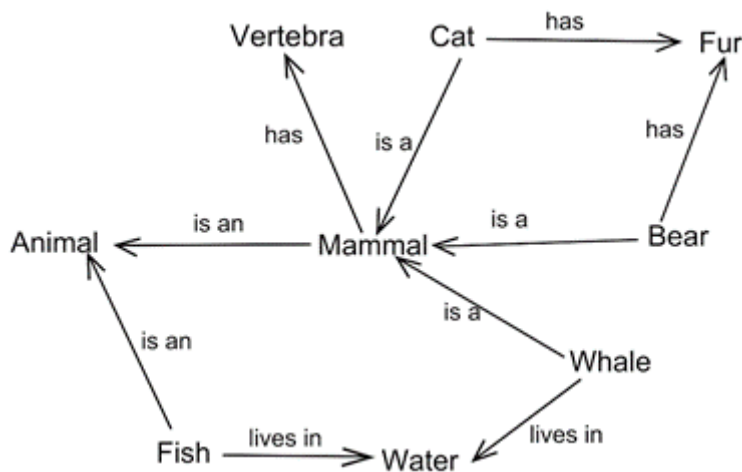


Fig. 1—Example of semantic network (Wikipedia and Semantic-network 2022).

One of the key aspects in building a semantic network is to identify and associate ontologies. In this paper, ontologies refer particularly to vocabularies. That is, classes and properties but not individual data points. In the example of Fig. 1, the entities cat, mammal, fish, etc., and their relationships “is a,” “is an,” “has,” “lives in,” are part of the ontology to describe the knowledge in this domain (Wikipedia and Ontology 2022). This ontology is the triplet “head-relationship-tail” as shown in Fig. 2, along with a drilling domain example where “head” and “tail” are concepts.

A note on terminology—within this document, we use triplets “head-relationship-tail” interchangeably with “subject-verb-object” or, more simply, concepts and relationships or even nodes and relationships. The semantic construct is the same and can be further extended as shown in Fig. 3.

The user stories built in the early stage of this project have produced a rich ontology. For example, the user story described earlier in the section An Example User Story—the rounding of MD values—has generated many use cases. One of them, number three in the list, describes confusion about the vertical depth reference (datum), as shown in Fig. 4. This use case was mapped and drafted in the semantic network shown in Fig. 5 for the case of a floating rig subject to tide and heave movements. During the process of building many semantic networks, we collected a number of concepts and their relationships. The partial ontology built so far is in Table 2.

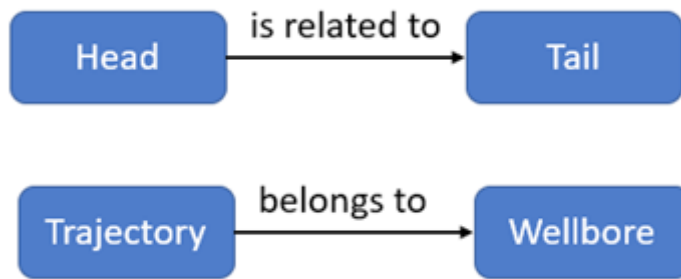


Fig. 2—Concepts and relationships.

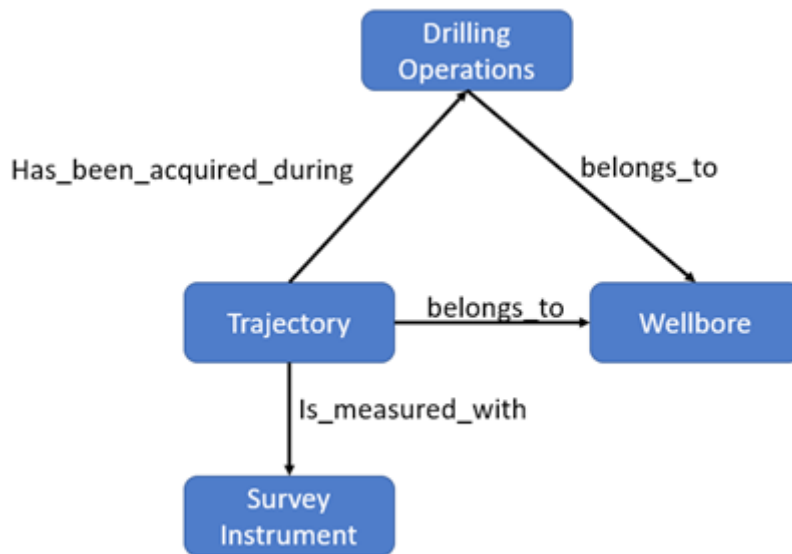


Fig. 3—A simple semantic network in the drilling domain.

### Use case #3 “Confusion around vertical datum”

- Use case (goal): convey which origin is used for depth measurements
- Primary actor: Driller
- Stakeholders: Directional driller, mud logger, LWD engineer, Operation geologist, completion engineer, etc.
- Scope: Drilling control system, DD Software, Mud logging software, formation evaluation interpretation software, Geological interpretation software, well design, well planning, completion design, completion operations, workover operations....
- Level: subfunction
- Brief: Different disciplines may use different depth datum.

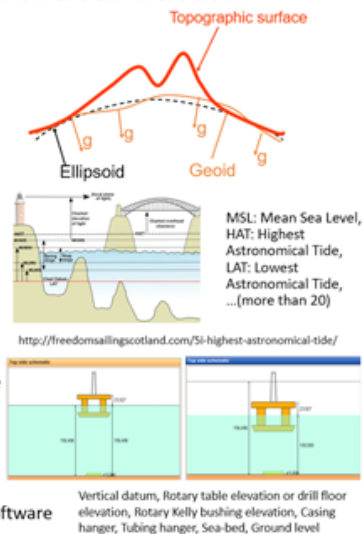
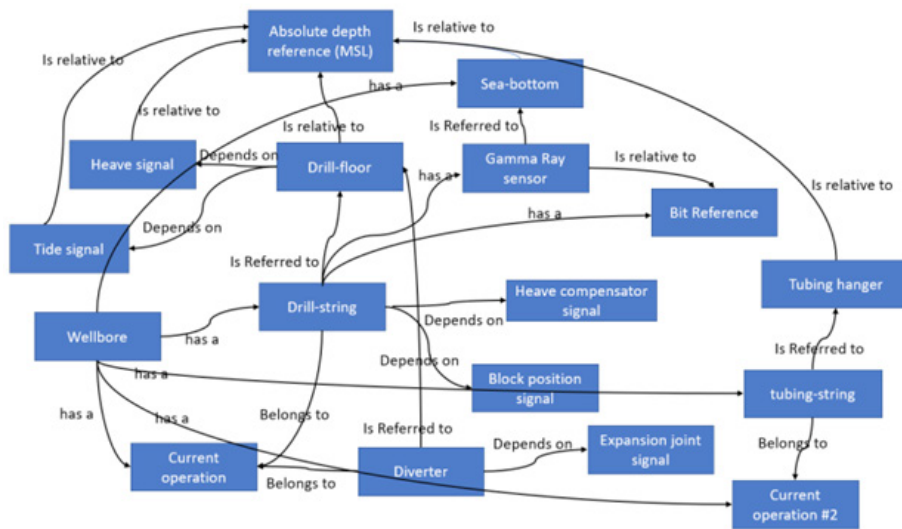


Fig. 4—Example of use case for vertical datum reference. Here, DD, LWD, and FE stand, respectively, for directional drilling, logging while drilling, and formation evaluation.





**Fig. 5—Semantic network based on use case “convey which origin is used for depth measurements,” described in Fig. 4. The use case is related to a floating rig subject to tide and heave movements.**

	Concept Categories		Relation Categories	
Bounds	Actual operation	Manual tally	Belongs to	Is relative to
Closed volume	Compensation	Measurement program	Constrains	Is reset using
Depth stack	Confidence factor	Operation event	Depends on	Is valid for
Formation	Correction	Planned operation	Extends to	Originates from
Formation top	Correction method	Real-time signal	Has a	Utilizes
Geomodel	Depth interval	Rig	Is a	Has top
Georeference	Distance	Rig element	Is associated with	Is subsurface of
Property cube 3D	Distribution	Survey file	Is caused by	Is corrected from
Rock volume	Diverter	Survey instrument	Is compensated using	Is corrected using
Seismic	Drillstring	Trajectory	Is described by	Is drilled using
Stratigraphy	Drillstring element	Uncertainty	Is estimated using	Is interpolated as
	Electronic tally	Wellbore	Is extended to	Is interpolated using
	Formation log	Well depth	Is higher vertical bound	
	Formation logging tool	Well path	Is lower vertical bound	
	Generator	Wireline	Is measured using	
	Interpolation scheme		Is referred to	

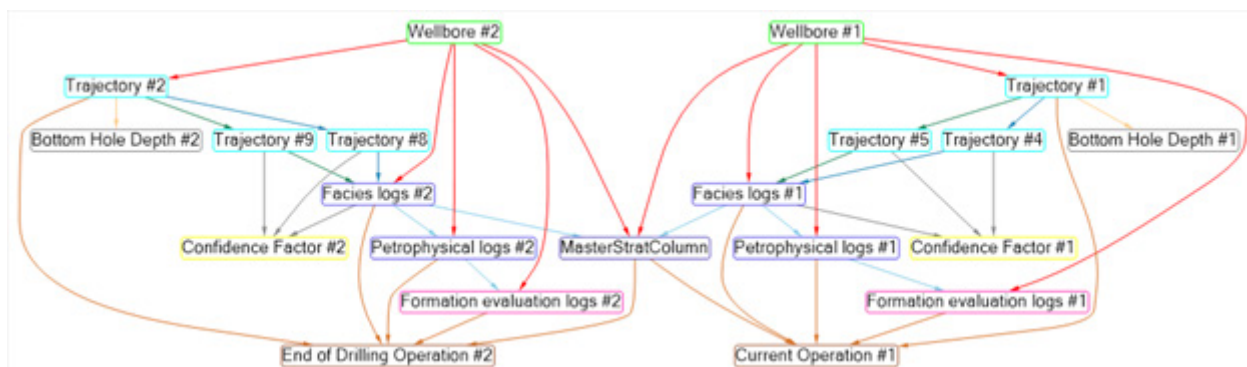
Table 2—List of concepts and relationships elicited from user stories.

When it is natural to work with multiple semantic networks, it can be useful to name and identify each of them. For instance, we have seen that the example user story selected through the ranking process has been broken down into 16 different use cases. It is natural to describe each of these use cases by its own semantic network.

However, these differently named semantic networks may refer to common concepts. It is therefore important when utilizing named semantic networks to have a mechanism for merging several semantic networks. A node in one named semantic network can be associated with another node in another named semantic graph through a relation of the type “is identical to.” The merged semantic network does not modify the original graphs. It simply supplements new facts about the similarity between nodes. This is in perfect agreement with the main idea of semantic networks (i.e., capturing information through statements). However, graphical user interfaces may take advantage of this similarity information and represent a simplified version of the merged graphs (Fig. 6).

So far, we have considered that we can express facts as a triplet head-relationship-tail and that any concept may be used for the head or the tail together with any relationship. This freedom may lead, however, to semantics that are meaningless, and rules must therefore assert the types of heads and tails that can be associated with any relationship.

For example, the statement “a rig”-“is a subsurface component of”-“a drillstring” is meaningless and yet the statement respects the structure “head-relationship-tail.” For that reason, we would like to supplement semantic networks with validation rules or constraints. For instance, it may be possible to define a rule that asserts that the head of the relationship “is a subsurface component of” must be a surface and that the tail of the relationship must be a volume. Then the relationship “is a subsurface component of” is applicable to heads



**Fig. 6—Two workflows for two different wellbores share the same stratigraphic column and therefore the two semantic networks are adjoined at the level of the node “MasterStratColumn” concept.**

of the type “formation top” and the tail can be of the type “closed volume,” “depth stack,” “formation,” “geo-model,” “rock volume,” “stratigraphy,” etc.

Rules also apply to the cardinality of relationships that arrive from or end at a node. For example, a “survey file” must have only a single “survey instrument” while a “trajectory” may be associated with several “depth intervals” connected to a unique “survey instrument.”

Rules can detect inconsistencies in the semantic network. When posting a new statement that contradicts an already defined statement, it is possible to detect an illogical relationship. For instance, such a rule can be that some relations are exclusive from each other (i.e., “relation 1” and “relation 2” cannot be applied simultaneously to the same pair of subject and object when the object is not null). For instance, the relations “is the higher vertical bound” and “is the lower vertical bound” are mutually exclusive when the vertical uncertainty is not null, because if they were not then it would mean that the vertical uncertainty is zero, which is in contradiction with the initial hypothesis that it was not null. So, if there is already a statement that says that “trajectory 1”-“is the higher vertical bound of”-“uncertainty 1,” and “uncertainty 1” is not null, then it will not be acceptable to post a new statement that states “trajectory 1”-“is the lowest vertical bound of”-“uncertainty 1”.

The set of rules that supplement the terminology helps to detect any inconsistencies when posting new statements in the semantic network. This set of rules needs to be rich to capture inconsistencies but not too restrictive, so that statements that have not been thought of at the design phase can still be expressed. This set of rules also captures domain knowledge.

An application can download a semantic network and can thereafter extract information. Browsing the semantic network to retrieve specific nodes or statements can be tedious and necessitate downloading all or at least a large part of the semantic network. If the semantic network is very large, this can be a rather time- and memory-consuming task. Alternatively, it is possible to query the server that hosts the semantic network to retrieve the desired information. In that case, the server performs the work and only the server needs to have the complete semantic network ready for browsing. Accessing data through queries allows reuse of the same queries in different contexts and the writing of applications that adapt themselves to variable semantic descriptions. A typical query language for semantic networks is SPARQL (Pérez et al. 2009), which is a recursive acronym for SPARQL Protocol and RDF Query Language, where RDF stands for Resource Description Framework (Lassila and Swick 1999). SPARQL utilizes a syntax that is inspired by SQL, Structured Query Language (Date 1989), which is used with relational databases. It is then possible to query the server regularly and retrieve updated information about the ever-evolving semantic network without having to download the semantic network.

Furthermore, graphs enable inferencing (reasoning). In the context of a general graph, inferencing utilizes the structure of the graph to find paths between nodes. Additionally, inferencing is used to assess the centrality of nodes in the graph (whether a node has many direct connections, is transitively connected to other nodes, can reach other nodes with few hops, or sits on the shortest path of many pairs of nodes). Semantic networks, perhaps not so obviously, allow us to take into account semantics (meanings): For instance, to consider the meaning of the relationships, perform taxonomic reasoning or use rule-based reasoning. It is possible to infer new statements from the existing semantic network even though they were not explicitly posted in the graph.

Building the semantic network enables:

- Basic reasoning of concepts
- Customization of specific conditions for end users and data providers implementation
- Mapping to multiple abstract layers to further improve the description and relation of the drilling knowledge
- Mapping into the primary data (described in the section Multilayered Perspectives)

To understand further the editing and processing capability of the semantic network, we have developed a computer system to implement these concepts (Fig. 7). We also make available a simple web interface to create and manipulate a semantic network for anyone who wants to become familiar with these concepts. It is accessible at <https://app.digiwells.no/DDQUDSemanticGraph/NamedGraphs>.

The semantic network is an abstract layer overlying the primary data. In the next sections, we will discuss how to express the primary data and how to integrate the primary data with the semantic network.

**The Data Lake Perspective.** It is legitimate to ask how capturing semantic information using semantic networks relates to storing data in a data lake. First, what is a data lake? A data lake is a system for storing raw copies of measurements, configuration data, or transformed data. It can support structured data storage (e.g., relational databases, semistructured, e.g., XML, JSON, CSV, LAS, unstructured data like documents, or binary data such as LIS, SEG Y, images, and videos).

With general-purpose data lake solutions, there is a risk that large amounts of data are stored and yet are difficult to use afterward unless metadata associated with the binary large objects help organize the data.

With structured data lakes, it is possible to organize and access data utilizing a predefined structure. Example of structured data lakes are WITSML (Wellsite Information Transfer Standard Markup Language) servers, the Open-Source Data Universe data platform (a forum of the OpenGroup), or Professional Petroleum Data Management Data Model. The intended residence of these repositories

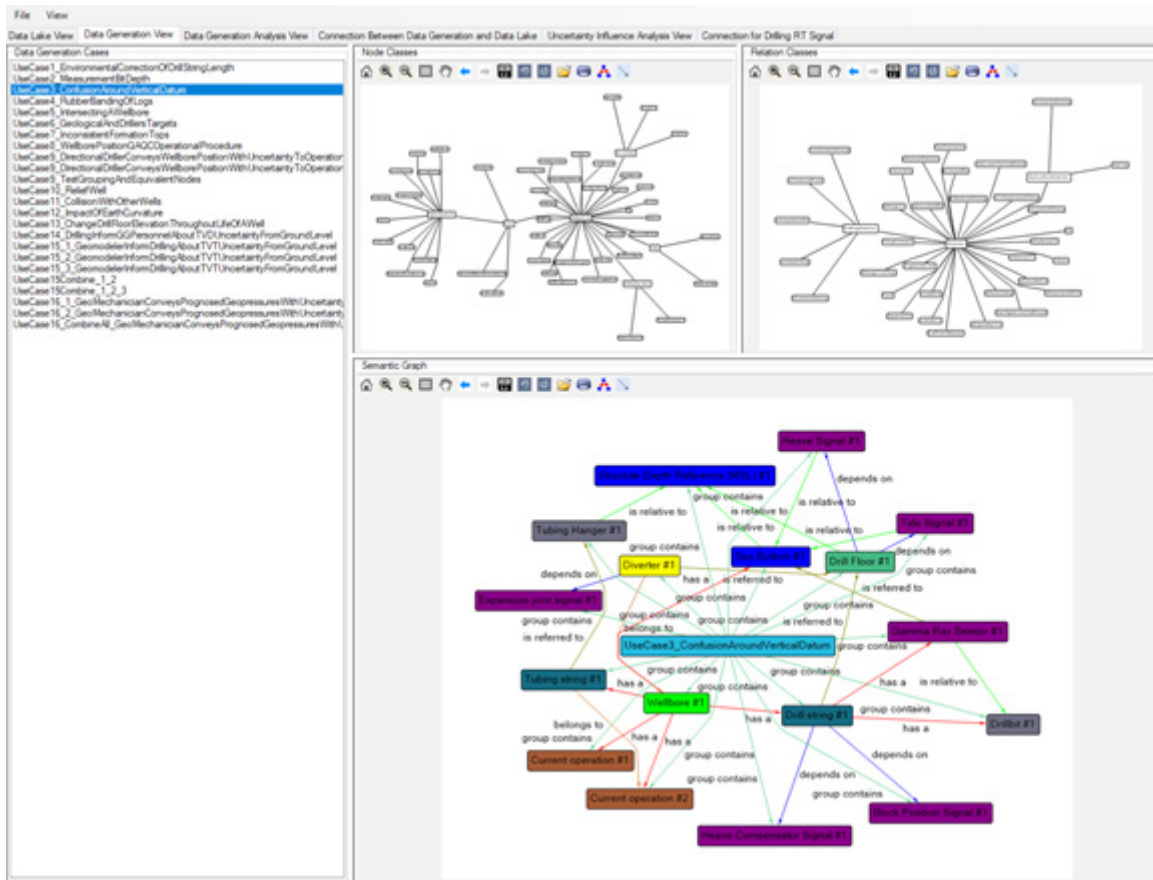


Fig. 7—Screen shots of a prototype application to manipulate semantic networks and their connection to primary data stores.

differs—Open-Source Data Universe in the cloud while the others reside at the edge and in corporate data centers. This being said, with the gain of importance of cloud solutions, it is more and more common to see WITSML servers being made available in the cloud (Cardoso Braga et al. 2021, Neri and Philo 2020).

Taking WITSML as an example of a structured data lake, a series of XML schema definition files defines the data model. The data schema defines classes, or in other words, the blueprints of different data characteristics (Wikipedia and Database-Schema 2022). A class can have several subclasses. A subclass is a specialization of the parent class, therefore defining a taxonomy or an inheritance tree, meaning that all the subclasses inherit the properties that exist at the level of a class. A property may be of one of the base types (e.g., numeric, character string, or it may be of a type defined as part of the data model). This allows, for instance, defining specific domain enumerations. A class may also define properties that refer to instances of other classes, thus allowing associations between instances of classes (Fig. 8).

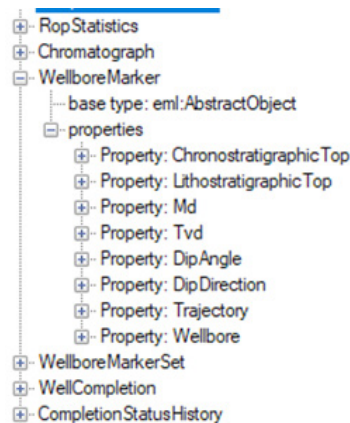


Fig. 8—Example of properties associated with a class, here WellboreMarker, in the WITSML data schema. Some of those properties are of complex types like stratigraphic top, trajectory, or wellbore. Other properties are of base types like MD, TVD, DipAngle, and DipDirection, all of which are floating point values.

Here, the choices for possible associations are static they are an integral part of the data model. To retrieve information, programs exploit these predefined associations, which facilitates the writing of data mining programs. When several programs adhere to the same



version of the data schema, they can exchange information because programmers (not the programs) understand the meaning of information. Fig. 9 illustrates that a wellbore, here named “WellBore 00,” has many subparts, including several trajectories, wellbore geometries, and logs. Recursively, wellbore geometries may have several geometry sections, etc.

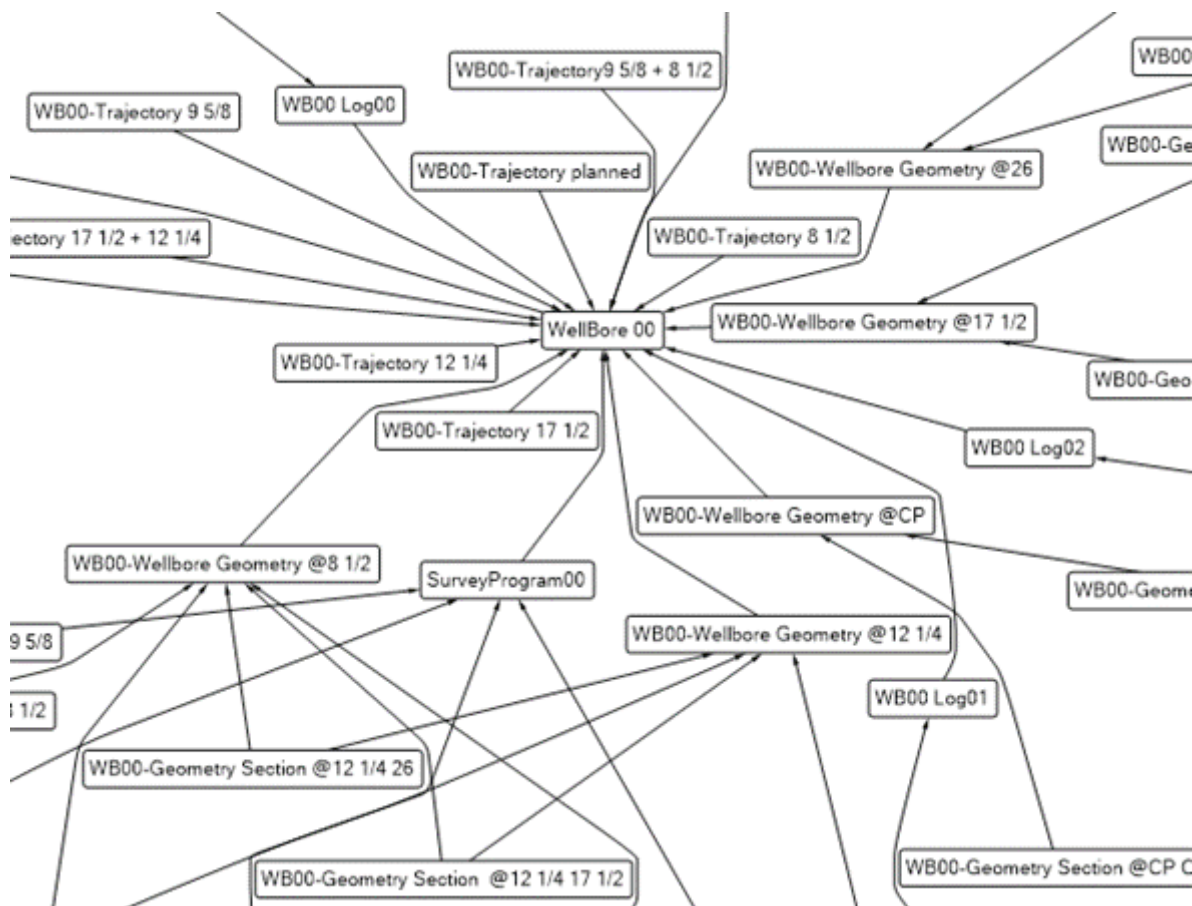


Fig. 9—Illustration of the connections between data objects (i.e., instances of classes) in a WITSML data store.

However, after discovering that some associations are missing in the data model, it is necessary to wait for the release of a new version of the data model and then for modification of all programs to use the new features of the data model. Managing new versions of the data model and upgrading the relevant programs may take several months or years.

The purpose of such structured data lakes is to store large amounts of data and to provide a way to retrieve information that relies on the programmers’ ability to understand the meaning of the data model. A structured data lake presents an efficient way to store raw and processed data, but usually does not provide many possibilities to capture the manner of generation of the processed data or the origin of the raw data. Such information would require much more flexibility than that available through static class hierarchy definitions.

For instance, in Fig. 9, there are multiple trajectories, named “WB00-Trajectory 12 ¼,” “WB00-Trajectory 17 ½,” “WB00-Trajectory 8 ½,” “WB00-Trajectory 9 5/8 + 8 ½,” etc. Even though a person may interpret their names, they are not sufficient for a computer program to understand the reasons for creating the different trajectories and their purpose, much less the acquisition or calculation of the data.

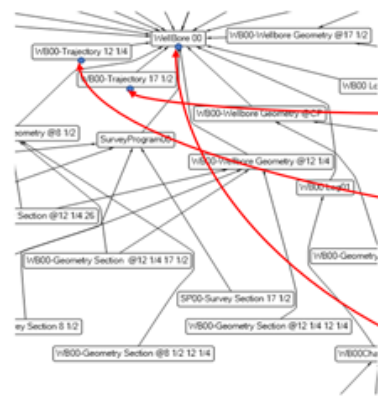
This is where semantic networks can help. Semantic networks can describe the meaning of the different trajectories by describing facts about those trajectories. Utilizing the semantic networks, an application can determine the meaning of the different trajectories and can, therefore, choose the one that is relevant for its purpose without necessitating the involvement of human beings in that decision. To make the connection between the semantic information captured in the semantic networks and the data stored in the data lake, a node in a semantic network can have a uniform resource identifier that describes where to access the actual (primary) data in the data lake.

Fig. 10 illustrates how a wellbore and two trajectories are associated between the data lake perspective and a semantic network that involves those objects. The red arrows show the connection between the two perspectives: (a) is the data lake perspective and (b) is the semantic network viewpoint.

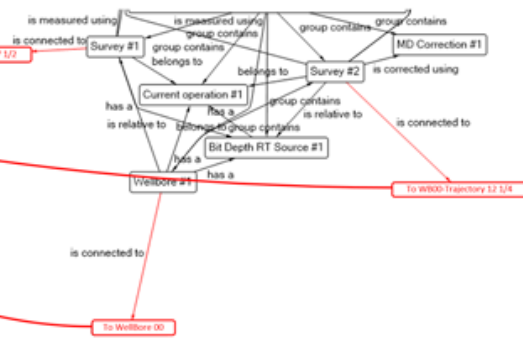
**Connection with the Semantic of Drilling Real-Time Signals.** Some of the semantic networks that capture the how and why of drilling data generation need to refer to drilling real-time signals. A method to represent the meaning of drilling real-time signal in a computer interpretable way can be found in Cayeux et al. (2019). The method uses semantic networks. However, it introduces specific concepts naturally linked to the description of the meaning of drilling real-time signals, and in the concepts and relations utilized to express statements on these signals. For instance, there are ways to describe the physical quantity of a signal or where a signal is logically associated in the drilling system, through possibly different perspectives such as a mechanical or a hydraulic point of view (Fig. 11). It is also possible to characterize the processing applied to the signal or the models that have been used to generate the real-time values.

It is possible to query the overall semantic network that contains the description of all the facts concerning drilling real-time signals. In that manner, it is possible to retrieve drilling real-time signals not only by their identification but also according to the specifications that are required for their usage in a workflow. This allows keeping workflow descriptions as generic as possible without having to

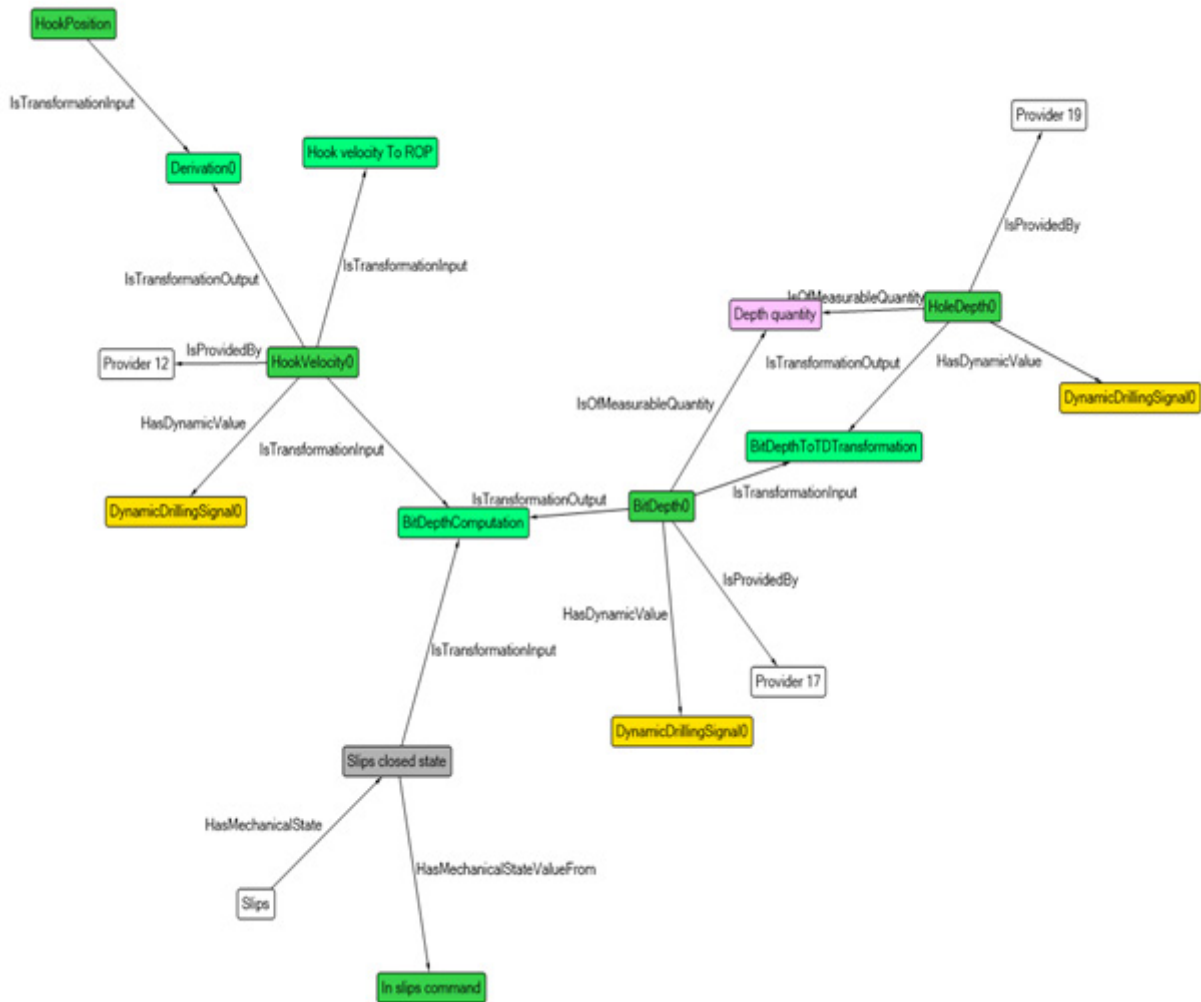
a) Data lake



b) Semantic graph



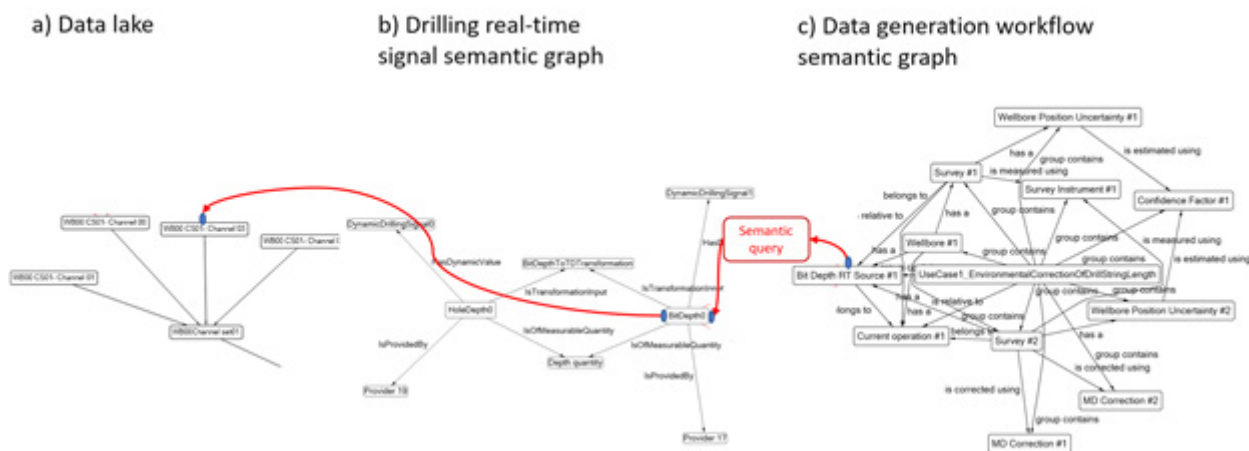
**Fig. 10**—Concepts in a semantic network can refer to objects in the data lake that actually contain the data. In this illustration, two different survey files referred in a semantic network and a wellbore have uniform resource identifiers that indicate where to find their actual content in the data lake.



**Fig. 11**—Example of a partial subgraph that describes facts associated with a signal “BitDepth0.” Note that the node colors correspond to which class they belong to.

recourse to statically defined connections. When based on a semantic query, the link to a real-time signal can change automatically if new information is available: It is possible to choose the most relevant signal at any time, even when signals become invalid or new signals become available.

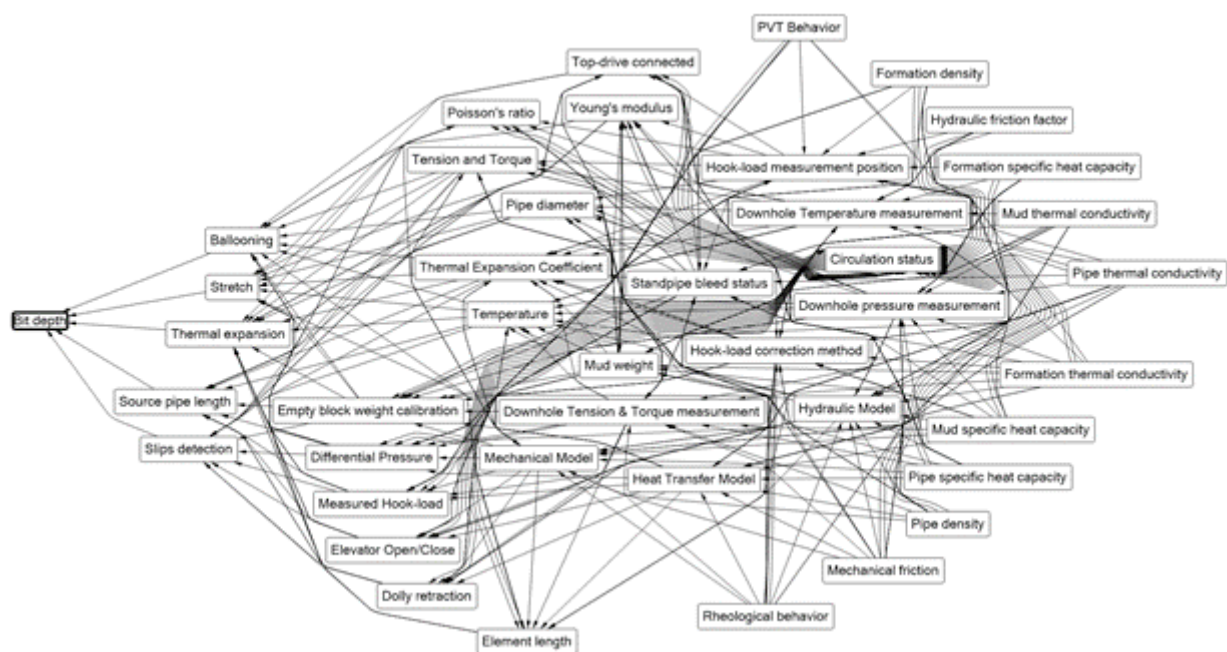
Of course, the value of the drilling real-time signals is stored into logs managed at the data lake level. A uniform resource identifier links the node of a signal belonging to the semantic network with its position in the data lake, as shown in Fig. 12.



**Fig. 12—A drilling real-time signal semantic network describes the meaning of signals. A signal node in this graph can connect to a position in time or depth-based log where the actual values are stored. A data generation workflow can query the drilling real-time signal semantic network to retrieve the most relevant signal.**

**Propagation of Uncertainty: Influence Diagrams.** Another crucial piece of information not often captured in drilling data lakes is quantified uncertainties of the stored data. Yet, the first lesson that every engineer learns in a physics class is that uncertainty accompanies any numerical value. Despite this common-sense rule, standard data models do not have any place to provide information on the precision and accuracy of numerical data.

In addition to providing numerical estimates of the uncertainty of stored data, it is also important to capture which sources of information influence the estimation of the uncertainty. Physical quantities influence other physical quantities, due to dependencies. Fig. 13 illustrates the possible associated effects that influence the estimation of MD. Note that one can refer to Bolt (2019) and Brooks et al. (2005) for more details about the topic of depth uncertainty. From these references, one can learn that “pipe ballooning” (Lubinski and Althouse 1962), “elasticity” (Brooks et al. 2005), and “thermal expansion” (Kyllingstad and Thoresen 2019), or the method used to get the length of each pipe, or the quality of the detection for whether the drillstring is in-slips or not, can all influence the estimation of bit depth. Recursively, each of these sources of influence depend on other quantities or models with their own sources of uncertainties. For instance, drillstring ballooning depends on the difference of pressure between the inside and outside of the drillstem, the Young’s modulus, and Poisson’s ratio of the materials (Landau et al. 1986) used by the drillstring components, the diameters and lengths of the pipes, and



**Fig. 13—Illustration of an influence graph for the uncertainty associated with the bit depth.**

so on. Additionally, either measuring or calculating the difference of pressure between the interior and exterior of the pipe brings other sources of uncertainty into play.

In this way, it is possible to create an influence graph between multiple sources of raw data and models and the quantifiable value of uncertainty. For such an influence graph, the relationship is the same for every edge of the graph, namely “influences the uncertainty of.” However, the relationship has a weight. This weight characterizes the degree of influence that the head of the relation has on its tail. The influence graph is related to Bayesian networks (Stephenson 2000) (i.e., a probabilistic graphical model that represents a set of concepts and their conditional dependencies using a directed acyclic graph—that is, a directed graph with no cycles) (Wikipedia and Bayesian-Network 2022). Then, it is possible to filter the influence graph for cutoff values of the edge weights and find which concepts and which paths between these concepts influences a particular value the most (Fig. 14).

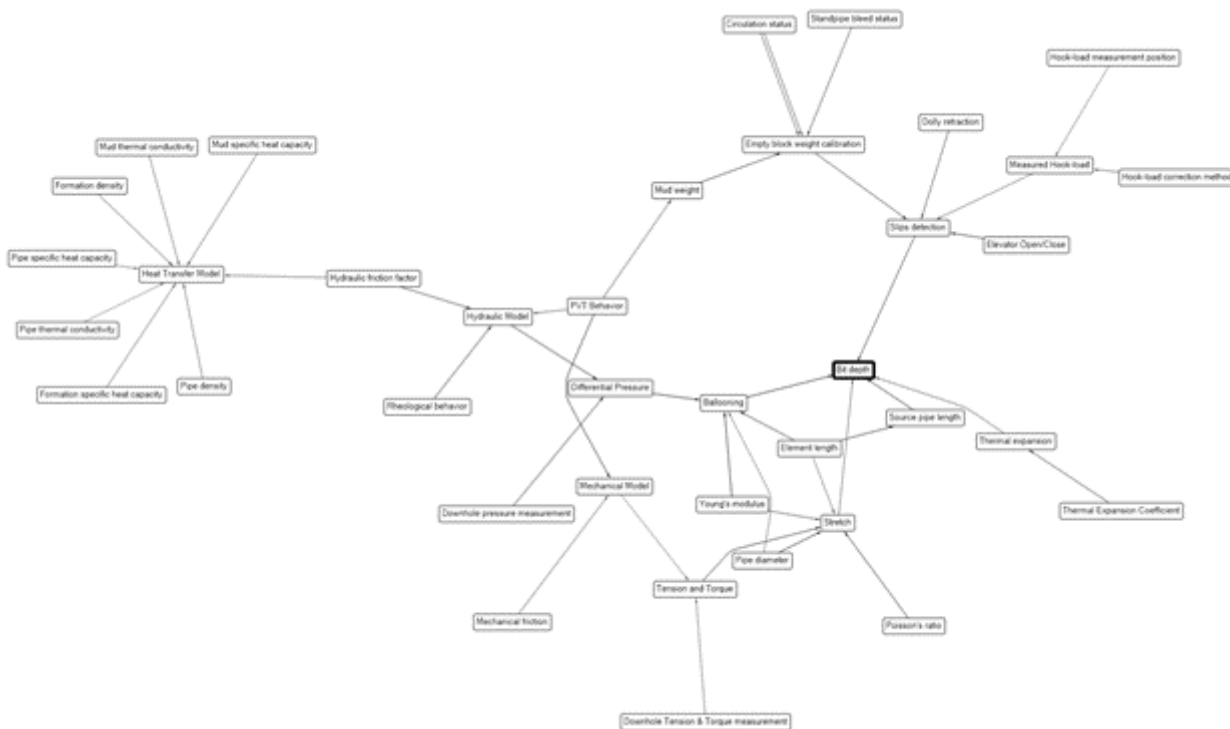


Fig. 14—Influence graph for the bit depth in a particular case after filtering out the relationships that have the less influences.

Note that an influence graph for a concept may be different from one case to another. For example, the influence graph for bit depth is different for a fixed platform compared to a floating rig. However, even for equivalent graphs in terms of nodes and edges, the weights may differ from one case to another case. For example, for two wells drilled from the same platform using the same type of equipment, the weights for the influence graph of the bit depth will depend on the trajectories, casing depths, etc.

**Multilayered Perspective.** We have seen that several types of graphs capture data and knowledge related to drilling programs and operations.

There is the data lake perspective, where actual data are stored and possibly structured. If one wants to capture the why and how of data creation, one can resort to semantic networks. The vocabulary is static, but there are no restrictions on the construction of the statements. This gives the same versatility as with a natural language: It is possible to express unplanned notions. However, it is important to define clearly the meaning of each concept and relationship to interpret properly the meaning of the statements.

A semantic network data representation is well suited to capture the wide range of processes and workflows that generate data. Multiple semantic networks capture the process of generating data through workflows. Combining these semantic networks creates an overall representation of all the facts involved during data generation. The semantic networks complement the data stored in the data lake, and concepts manipulated in semantic networks connect to actual data stored in the data lake.

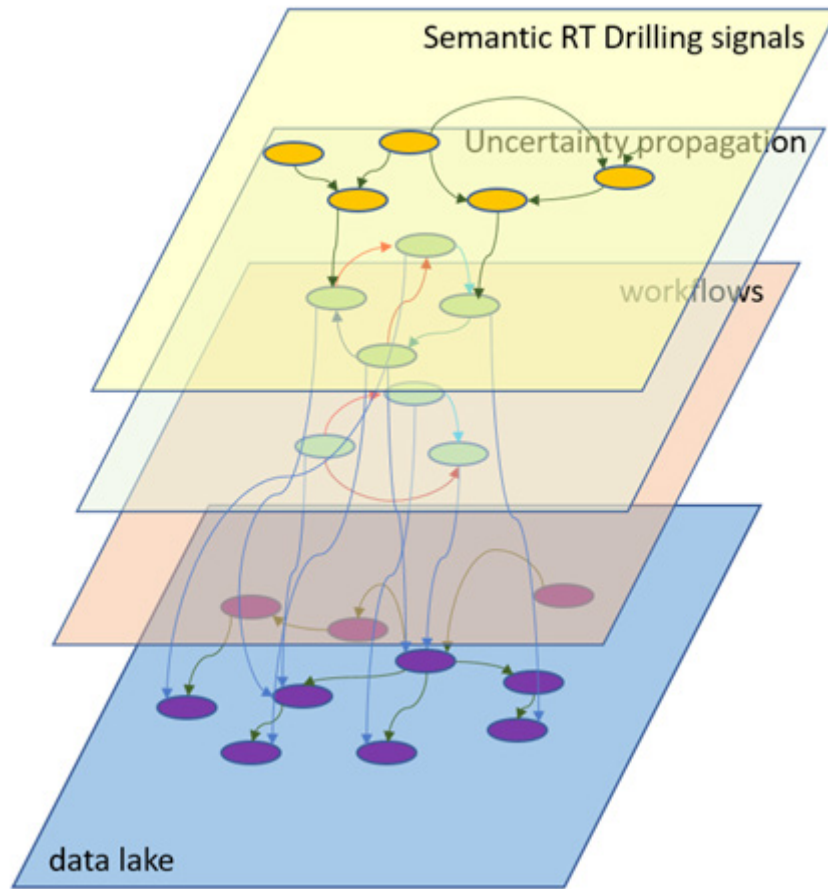
Similarly, the semantics of drilling real-time signals also defines other points of view that are in parallel to the data lake and the work process descriptions. For example, actual uncertainty information can occur in another perspective. Altogether, these interlinked perspectives define a multilayered graph (Boccaletti et al. 2014). The graphs used in the different perspectives are of different natures, as shown in Fig. 15. The purpose of the perspective dictates the choice of the type of graph used in the perspective. So far, we have considered class hierarchy-based data models, semantic networks, and Bayesian networks. Some of the interspersive links are static in definition; others are dynamic, for example, as the result of a semantic query.

This multilayered graph allows for different sorts of inferencing and learning. Discovery of new information not explicitly expressed in any of the different perspectives allows for synergies between all the different perspectives.

### Observations of the Practical Implications of a Generalized Data Modeling Perspective

The semantic network and multilayer graphs discussed above have many practical requirements for implementation. As previously discussed in this paper, it is necessary to collect and store a wealth of information for consumption by users and applications. To use the data effectively, it is essential to capture information about static relationships.





**Fig. 15—Data and knowledge are captured through different means. Each of these means provides a different perspective on the data and knowledge. Regardless of whether the data and knowledge are structured through static class hierarchy-based data models, semantic networks, or Bayesian graphs, each of these perspectives is interrelated with the others. The sum of data and knowledge represented by these different graphs is a multilayered graph. Note that here RT stands for real time.**

To assist with this, several commercial graph databases are available to optimize storage and retrieval of graphs. In addition to the object model (Wikipedia and Object-model 2022) and relationship storage, it is vital to retrieve time-indexed and depth-indexed data for each use case or workflow using this object model. In addition to the relationship information, there is metadata associated with each data element from its definition, acquisition, processing, and storage. Introduction of significant levels of data quality and uncertainty can occur across these phases due to a number of root causes, including the lack of access to metadata or access to incorrect metadata. The following subsections detail an illustrative sample of how one may go about accumulating the relevant metadata associated with each data element.

There are three levels of metadata in this case—the definition level, the deployment/acquisition level, and the processing and storage level.

**Definition Level.** The definition level describes the nature of the data element, minimum expected precision of the data, and user stories, workflows, and associated processes in which the data element is used. This type of information provides guidance to the implementation teams for minimum expectation and data requirements. It is also important to document what processes, workflow, or user stories will use this data, and if it is a required or an optional data item. This will help ensure that implementation teams enable a particular workflow for the required data and provide an alert if it is not relevant.

**Deployment Level.** The deployment level describes the data acquisition information, the specific sensor used to acquire the data, the make, model and suggested calibrated range of the sensor. Details are also included describing the sensing technology, analog to digital conversion frequency, and other associated sensor data requirements. The location of the sensor in the system is also noted. For an example, the sensor is the hookload sensor on the deadline, in the mast, on the topdrive. Also included is the calibration method and the last calibration timestamp. If the data are calculated or derived, then information about source data and its associated metadata will be required. The raw input scan frequency is critical information, particularly for high-frequency data. For example, data collected at 10 milliseconds, but with an analog to digital converter on the input card running at 15 milliseconds, will have a periodic stair-stepped appearance leading to artifacts. It is also essential to capture metadata about the sensor, the input card, and the scan frequency of the acquisition system, as these will yield information about the reliability of the data at full resolution.

**Process and Storage Level.** During the processing and storage phase, most of the sensed data on a drilling rig comes from the rig control system, instrumentation systems, service company skids, or all of these. Systems, either on the rig or in the cloud or both, then aggregate

this data. Transformation and derivation of new data occurs during aggregation. A critical issue in this process is timestamping of data. When the aggregation system receives data, it should retain the source timestamp; if not, its absence should be noted on all received data. See Annaiyappa et al. (2022) for more discussion on timestamping best practices. For derived data, documenting the derivation method is essential.

Very often, the aggregation system will scan the data at a higher sampling interval but only record and forward data at 1 second or 10 seconds. During this process, the value recorded is either the instantaneous value, the minimum, maximum, or average over the previous interval. Capturing this information as metadata, as additional information, is important. Often, this sampling function changes during the well. It is vital to record all changes for future reference.

## Future Work

It is a legitimate question to ask how, in practice, to capture all these statements, facts, uncertainty, and their associated propagation. If this was a manual operation by every stakeholder in a well design workflow or during a drilling operation, it is certain that the proposed framework would not be well populated nor maintained. However, it is possible to envision that the software used by the different stakeholders seamlessly captures the necessary information. Then, the required change is at the level of the software tool providers, not required by the engineering or operational personnel. This is still a huge leap, but the possible benefits of providing smarter functionalities that can use the meaning of data should help drilling software application vendors embrace this new opportunity. Indeed, it is feasible to conceive of new functionalities that take advantage of accessing information, when the meaning of this information is computer readable. A sequel article to this one will present how multiple software applications can make use of the concepts described in this paper to clarify notions that are used in different ways by the applications. As a result, seemingly inconsistent information can be shared between these applications and correctly converted across the various applications, without necessitating human interpretation of the different meanings used by each of the programs.

The work presented in this paper demonstrates connecting multilayered semantic networks to the data lakes to represent uncertainty and quality. In some sense, it is possible to extrapolate the overall combined framework to be a “semantic digital twin” of the well (regarding drilling) through the primary data and its interpretation. It is an opportunity to add relationships that are even more complex to represent other meanings beyond initial intentions. More importantly, it is possible to make inferences in this framework, as programmers know the concepts and relationships. Finally, it should be possible to integrate the created framework with similar ones from additional disciplines in the oil and gas domain. Parts of this future work are beyond the scope of this study.

While our mission was to develop an improved method that will result in increased confidence in understanding and describing data quality and uncertainty during well operation, it is projected that future work efforts in the area of “user integration” could benefit from further investigation. User interface design and integration are topics that can act together to capture overall data uncertainty and quality, thereby enhancing overall decision-making. Attention on the seamless human machine interface and overall user experience, and integration tools to visualize risk, are essential to improving overall decision-making. Multiple users, working together, can identify risk and associated mitigation plans in diverse ways. These differences can use visualization tools to help bring together different perspectives to complement and aid the overall decision-making process while unlocking new innovative ways of reducing uncertainty and improving data quality processes and tools overall. Tracking and providing digital tools for improving the overall user experience and decision-making process can also ensure incorporation of a continuous improvement process into the overall system.

As recent studies suggest (Bogert et al. 2021), humans have a tendency to place trust in data presented through computerized mediums, from self-driving cars to automated banking. As automation becomes more prevalent within well construction, the full qualification of uncertainty between systems therefore becomes critical to building trust in the outcomes that automated systems are delivering. Building a common model for describing and qualifying the propagation of uncertainty across the industry is therefore in everyone’s best interest.

## Conclusion

We have seen that primary data stores, also referred to as data lakes, are convenient for the storing and retrieval of raw and processed data. They typically use a static data model to facilitate the programming of data transfer functionalities. Usually, their data model focuses on what shall be stored, not on the why or how of data creation.

It is possible to supplement the primary data store with semantic information associated with the process of data generation. It is feasible to capture this additional information in a semantic network. Semantic networks are versatile data structures that are capable of describing information not conceived at the design level. This versatility is important when describing the origin of data, as there is virtually an infinite number of ways to generate data. Semantic networks can be associated with semantic rules to validate the consistency of the captured facts. They also allow for dynamic queries and inferencing.

Semantic networks can describe the meaning of real-time drilling signals. Combined with semantic queries, it is possible to provide an indirection between workflow description and the available signals that are available in the context of the drilling operation. This allows maintaining the generality of the description at a high level and reuse of the captured workflows in variable contexts.

In addition, primary data stores are often not well suited to capturing information about uncertainty and its propagation across related data and processes. An influence graph can describe data uncertainty. Such a graph is related to Bayesian networks, which describe the conditional probabilities of the propagated uncertainty between data sources and processing functions.

Connection of all these graphs is either through direct links or through indirectly using dynamic queries. This creates a multilayered graph in which it is possible to perform advanced reasoning operations.

With adoption of this framework, the capture of semantic information can take place directly in the various pieces of software used by the different stakeholders that participate in planning, operation, or analysis workflows, leading to a seamless integration of the semantic of data and its associated uncertainty across disciplines.

## Nomenclature

Bayesian network = a directed acyclic graph that uses Bayesian inference for probability computations of the graph nodes that are considered as probabilistic variables in Bayesian sense.

Data lake = a repository used to store and process structured and unstructured data, at any scale.

Data schema = an abstract design that represents how data are organized.

Meta-knowledge = knowledge about knowledge.

Object model = a logical view of a software or system that is modeled using an object-oriented method.

Ontology = in information science, an ontology represents the properties of a subject area and how they are related.

Use Case = in information science, a use case is a list of actions defining the interactions between different roles or actors.

User Story = in agile software development, a user story describes informally and in the terms of the end user, the end goal of an atomic functionality of the system to be developed.

## References

- Abrahamsson, P., Salo, O., Ronkainen, J. et al. 2017. Agile Software Development Methods: Review and Analysis. arXiv:1709.08439. <https://doi.org/10.48550/arXiv.1709.08439>(preprint; submitted 25 September 2017).
- Annaiyappa, P., Macpherson, J., and Cayeux, E. 2022. Best Practices to Improve Accurate Time Stamping of Data at the Well Site. Paper presented at the IADC/SPE International Drilling Conference and Exhibition, Galveston, Texas, USA, 8–10 March. SPE-208732-MS. <https://doi.org/10.2118/208732-MS>.
- Askham, N., Cook, D., Doyle, M. et al. 2013. The Six Primary Dimensions for Data Assessment. In *Defining Data Quality Dimensions*, 432–435. Bristol, UK: DAMA UK Working Group.
- Bailey, D. C. 2017. Not Normal: The Uncertainties of Scientific Measurements. *R Soc Open Sci* 4 (1): 160600. <https://doi.org/10.1098/rsos.160600>.
- Ballou, D. P. and Pazer, H. L. 1985. Modeling Data and Process Quality in Multi-Input, Multi-Output Information Systems. *Manage Sci* 31 (2): 150–162. <https://doi.org/10.1287/mnsc.31.2.150>.
- Beck, K. 2000. *Extreme Programming Explained: Embrace Change*. Boston, USA: Addison-Wesley Professional.
- Boccaletti, S., Bianconi, G., Criado, R. et al. 2014. The Structure and Dynamics of Multilayer Networks. *Phys Rep* 544 (1): 1–122. <https://doi.org/10.1016/j.physrep.2014.07.001>.
- Bogert, E., Schecter, A., and Watson, R. T. 2021. Humans Rely More on Algorithms than Social Influence as a Task Becomes More Difficult. *Sci Rep* 11 (1): 1–9. <https://doi.org/10.1038/s41598-021-87480-9>.
- Bolt, H. 2019. Increasing Value Through Reducing Differences Between LWD and Wireline Depths. Paper presented at the SPE Annual Technical Conference and Exhibition, Calgary, Alberta, Canada, 30 September–2 October. SPE-196035-MS. <https://doi.org/10.2118/196035-MS>.
- Bray, T., Paoli, J., Sperberg-McQueen, C. M. et al. 2000. Extensible Markup Language (XML) 1.0, W3C Recommendation October (Reprint).
- Brooks, A. G. and Wilson, H. 1996. An Improved Method for Computing Wellbore Position Uncertainty and Its Application to Collision and Target Intersection Probability Analysis. Paper presented at the European Petroleum Conference, Milan, Italy, 22–24 October. SPE-36863-MS. <https://doi.org/10.2118/36863-MS>.
- Brooks, A., Wilson, H., Jamieson, A. et al. 2005. Quantification of Depth Accuracy. Paper presented at the SPE Annual Technical Conference and Exhibition, Dallas, Texas, USA, 9–12 October. SPE-95611-MS. <https://doi.org/10.2118/95611-MS>.
- Cardoso Braga, D., Kamyab, M., Joshi, D. et al. 2021. Using Particle Swarm Optimization to Compute Hundreds of Possible Directional Paths to Get Back/Stay in the Drilling Window. Paper presented at the SPE Annual Technical Conference and Exhibition, Dubai, UAE, 21–23 September. SPE-206170-MS. <https://doi.org/10.2118/206170-MS>.
- Cayeux, E., Daireaux, B., Saadallah, N. et al. 2019. Toward Seamless Interoperability Between Real-Time Drilling Management and Control Applications. Paper presented at the SPE/IADC International Drilling Conference and Exhibition, The Hague, The Netherlands, 5–7 March. SPE-194110-MS. <https://doi.org/10.2118/194110-MS>.
- Chu, Y., Yang, S., and Yang, C. 2001. Enhancing Data Quality through Attribute-based Metadata and Cost Evaluation in Data Warehouse Environments. *J Chine Inst Eng* 24 (4): 497–507. <https://doi.org/10.1080/02533839.2001.9670646>.
- Cockburn, A. 2002. Use Cases, Ten Years Later. *Software Testing and Quality Engineering (STQE) Magazine* 4 (2): 37–40.
- Cohn, M. 2004. *User Stories Applied: For Agile Software Development*. Boston, USA: Addison-Wesley Professional.
- Cong, G., Fan, W., Geerts, F. et al. 2007. Improving Data Quality: Consistency And. Paper presented at the Title of Host Publication Proceedings of the 33rd International Conference on Very Large Data Bases, University of Vienna, 315–326, Vienna, Austria, 23–27 September.
- Dashevskiy, D., Dahl, T., Brooks, A. G. et al. 2008. Dynamic Depth Correction To Reduce Depth Uncertainty and Improve MWD/LWD Log Quality. *SPE Drill & Compl* 23 (1): 13–22. SPE-103094-PA. <https://doi.org/10.2118/103094-PA>.
- Date, C. J. 1989. *A Guide to the SQL Standard*. Boston, USA: Addison-Wesley Longman Publishing.
- de Wardt, J. 2022. Drilling System Automation Body of Knowledge. <https://dsabok.org/drilling-data-quality-uncertainty/> (accessed 27 May 2022).
- Dubrule, O. and Nelson, P. H. 1987. Evaluation of Directional Survey Errors at Prudhoe Bay. *SPE Drill Eng* 2 (3): 257–267. SPE-15462-PA. <https://doi.org/10.2118/15462-PA>.
- Ekseth, R. 2000. *Uncertainties in Connection with the Determination of Wellbore Positions*. PhD dissertation, Norwegian University of Science and Technology, Hogskoleringen, Trondheim, Norway.
- Fang, H. 2015. Managing Data Lakes in Big Data Era: What's a Data Lake and Why Has It Became Popular in Data Management Ecosystem. Paper presented at the 2015 IEEE International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER), Shenyang, China, 8–12 June. <https://doi.org/10.1109/CYBER.2015.7288049>.
- Gosling, J., Joy, B., Steele, G. et al. 2000. *The Java Language Specification*. Boston, Massachusetts, USA: Addison-Wesley Professional.
- Grindrod, S. J. and Wolff, J. M. 1983. Calculation of NMDC Length Required for Various Latitudes Developed From Field Measurements of Drill String Magnetisation. Paper presented at the IADC/SPE Drilling Conference, New Orleans, Louisiana, USA, 20–23 February. SPE-11382-MS. <https://doi.org/10.2118/11382-MS>.
- Hejlsberg, A., Torgersen, M., Wiltamuth, S. et al. 2008. *The C# Programming Language*. London, UK: Pearson Education.
- ISCWSA. 2022. Industry Steering Committee for Wellbore Survey Accuracy. <https://www.iscwsa.net/> (accessed 27 May 2022).
- Jacobson, I. 1993. *Object-Oriented Software Engineering: A Use Case Driven Approach*. Noida, India: Pearson Education India.
- Kay, A. C. 1996. The Early History of Smalltalk. In *History of Programming Languages---II*, 511–598. New York, USA: The Association for Computing Machinery.
- Kyllingstad, Å. and Thoresen, K. E. 2019. Improving Accuracy of Well Depth and ROP. Paper presented at the SPE/IADC International Drilling Conference and Exhibition, The Hague, The Netherlands, 5–7 March. SPE-194098-MS. <https://doi.org/10.2118/194098-MS>.
- Landau, L. D., Lifšic, E. M., Lifšitz, E. M. et al. 1986. *Theory of Elasticity*, 7 vols. Amsterdam, The Netherlands: Elsevier.
- Lassila, O. and Swick, R. R. 1999. Resource Description Framework (RDF): Model and Syntax Specification. Technical Report, W3C Recommendation 1999-02-22. <https://www.w3.org/TR/1999/REC-rdf-syntax-19990222>.
- Lubinski, A. and Althouse, W. S. 1962. Helical Buckling of Tubing Sealed in Packers. *J Pet Technol* 14 (6): 655–670. SPE-178-PA. <https://doi.org/10.2118/178-PA>.
- McCarthy, J. 1978. History of LISP. In *The First ACM SIGPLAN Conference*, 173–185. New York, USA: Association for Computing Machinery. <https://doi.org/10.1145/800025.808387>.

- Moges, H. T., Dejaeger, K., Lemahieu, W. et al. 2013. A Multidimensional Analysis of Data Quality for Credit Risk Management: New Insights and Challenges. *Inf & Manag* **50** (1): 43–58. <https://doi.org/10.1016/j.im.2012.10.001>.
- Neri, P. and Philo, R. 2020. Cross-Discipline Cloud-Based Platforms Require a Single Version of Truth Across All Data to Deliver Reliable Decisions. In *EAGE 2020 Annual Conference & Exhibition*, 1–5. The Netherlands: European Association of Geoscientists & Engineers. <https://doi.org/10.3997/2214-4609.202011760>.
- Nygaard, K. and Dahl, O. J. 1978. The Development of the SIMULA Languages. In *History of Programming Languages*, 439–480. New York, USA: The Association for Computing Machinery.
- Pérez, J., Arenas, M., and Gutierrez, C. 2009. Semantics and Complexity of SPARQL. *ACM Trans Database Syst* **34** (3): 1–45. <https://doi.org/10.1145/1567274.1567278>.
- Pipino, L. L., Lee, Y. W., and Wang, R. Y. 2002. Data Quality Assessment. *Commun ACM* **45** (4): 211–218. <https://doi.org/10.1145/505248.506010>.
- Poedjono, B., Nwosu, D. et al. 2019. Wellbore Positioning While Drilling With LWD Measurements. *Petrophysics* **60** (3): 450–465. SPWLA-2019-v60n3a8. <https://doi.org/10.30632/PJV60N3-2019a8>.
- Sawaryn, S. J. and Thorogood, J. L. 2005. A Compendium of Directional Calculations Based on the Minimum Curvature Method. *SPE Drill & Compl* **20** (1): 24–36. SPE-84246-PA. <https://doi.org/10.2118/84246-PA>.
- Simard, V., Rönnqvist, M., Lebel, L. et al. 2019. A General Framework for Data Uncertainty and Quality Classification. *IFAC-PapersOnLine* **52** (13): 277–282. <https://doi.org/10.1016/j.ifacol.2019.11.181>.
- SPE-DSATS. 2022. Drilling Systems Automation Technical Section. <https://connect.spe.org/dsats/home> (accessed 27 May 2022).
- SPE-DUPTS. 2022. Drilling Uncertainty and Prediction Technical Section. <https://connect.spe.org/dupts/home> (accessed 27 May 2022).
- SPE-WPTS. 2022. Wellbore Positioning Technical Section ISCWSA. <https://connect.spe.org/wellborepositioning/home> (accessed 27 May 2022).
- Stephenson, T. A. 2000. *An Introduction to Bayesian Network Theory and Usage*. Martigny, Switzerland: IDIAP.
- Stroustrup, B. 2013. *The C++ Programming Language*. London, United Kingdom: Pearson Education.
- Tayi, G. K. and Ballou, D. P. 1998. Examining Data Quality. *Commun ACM* **41** (2): 54–57. <https://doi.org/10.1145/269012.269021>.
- Taylor, J. 1997. *Introduction to Error Analysis, the Study of Uncertainties in Physical Measurements*. Sausalito, California, USA: University Science Books.
- Taylor, B. N. and Kuyatt, C. E. 1994. Guidelines for Evaluating and Expressing the Uncertainty of NIST Measurement Results. In *US Department of Commerce, Technology Administration*. Gaithersburg, Maryland, USA: National Institute of Standards and Technology.
- Thorogood, J. L. 1990. Instrument Performance Models and Their Application to Directional Surveying Operations. *SPE Drill Eng* **5** (4): 294–298. SPE-18051-PA. <https://doi.org/10.2118/18051-PA>.
- Torkildsen, T., Håvardstein, S. T., Weston, J. et al. 2008. Prediction of Wellbore Position Accuracy When Surveyed With Gyroscopic Tools. *SPE Drill & Compl* **23** (1): 5–12. SPE-90408-PA. <https://doi.org/10.2118/90408-PA>.
- Van Rossum, G. and Drake, F. L. 2010. *The Python Language Reference*. Amsterdam, The Netherlands: Python Software Foundation. <https://scicomp.ethz.ch/public/manual/Python/3.6.0/reference.pdf>.
- Walstrom, J. E., Brown, A. A., and Harvey, R. P. 1969. An Analysis of Uncertainty in Directional Surveying. *J Pet Technol* **21** (4): 515–523. SPE-2181-PA. <https://doi.org/10.2118/2181-PA>.
- Wang, R. Y., Kon, H. B., and Madnick, S. E. 1993. Data Quality Requirements Analysis and Modeling. In *IEEE 9th International Conference on Data Engineering*, 670–677. Piscataway, New Jersey, United States: IEEE. <https://doi.org/10.1109/ICDE.1993.344012>.
- Wikipedia and Bayesian-Network. 2022. Bayesian Network, 9 March 2022, at 16:32 (UTC). [https://en.wikipedia.org/wiki/Bayesian\\_network](https://en.wikipedia.org/wiki/Bayesian_network) (accessed 27 May 2022).
- Wikipedia and Database-Schema. 2022. Database Schema, 17 May 2022, at 02:26 (UTC). [https://en.wikipedia.org/wiki/Database\\_schema](https://en.wikipedia.org/wiki/Database_schema) (accessed 27 May 2022).
- Wikipedia and Meta-knowledge. 2022. Meta-Knowledge. <https://en.wikipedia.org/wiki/Meta-knowledge> (accessed 27 May 2022).
- Wikipedia and Object-model. 2022. Object Model, 26 February 2022, at 05:13 (UTC). [https://en.wikipedia.org/wiki/Object\\_model](https://en.wikipedia.org/wiki/Object_model) (accessed 27 May 2022).
- Wikipedia and Ontology. 2022. Ontology (Information Science), 2 May 2022, at 18:09 (UTC). [https://en.wikipedia.org/wiki/Ontology\\_\(information\\_science\)](https://en.wikipedia.org/wiki/Ontology_(information_science)) (accessed 27 May 2022).
- Wikipedia and Semantic-network. 2022. Semantic Network, 13 May 2022, at 16:12 (UTC). [https://en.wikipedia.org/wiki/Semantic\\_network](https://en.wikipedia.org/wiki/Semantic_network) (accessed 27 May 2022).
- Wikipedia and User-Story. 2022. User Story, 26 April 2022, at 10:49 (UTC). [https://en.wikipedia.org/wiki/User\\_story](https://en.wikipedia.org/wiki/User_story) (accessed 27 May 2022).
- Williamson, H. S. 2000. Accuracy Prediction for Directional Measurement While Drilling. *SPE Drill & Compl* **15** (4): 221–233. SPE-67616-PA. <https://doi.org/10.2118/67616-PA>.
- Wolff, C. J. M. de Wardt J. P. 1981. Borehole Position Uncertainty - Analysis of Measuring Methods and Derivation of Systematic Error Model. *J Pet Technol* **33** (12): 2338–2350. SPE-9223-PA. <https://doi.org/10.2118/9223-PA>.