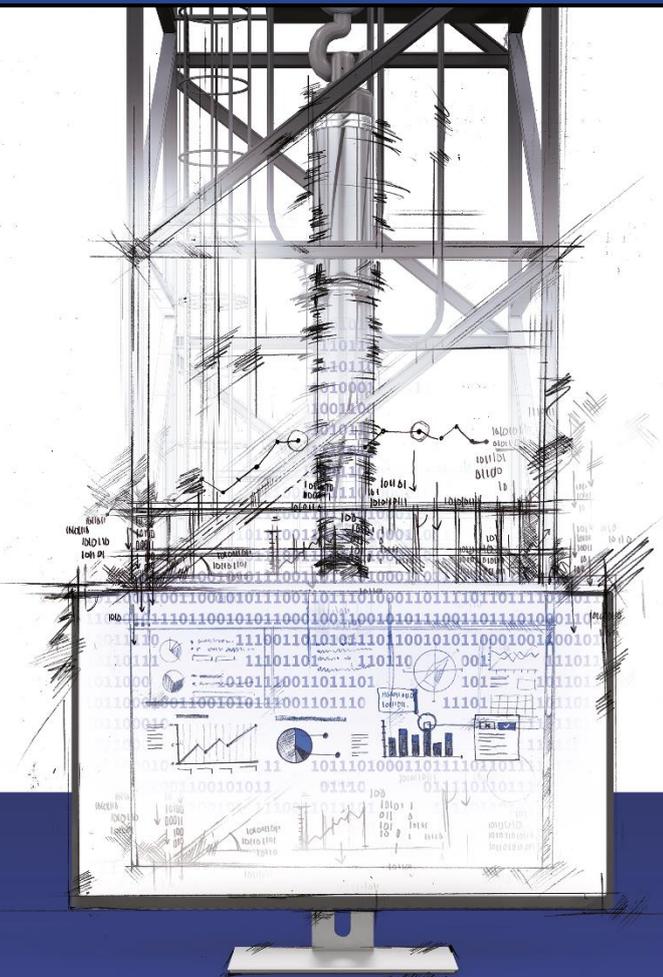


# WITSML™ Enabled Applications- Six Things to Think About



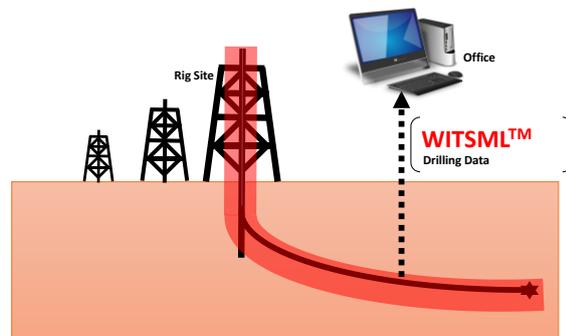
WITSMLstudio Briefing Note 17/001

# WITSML™ Enabled Applications- Six Things to Think About

WITSMLstudio is an online portal that provides access to WITSML technology and insights that help you take control of your drilling data.

WITSMLstudio Briefing Notes discuss practical issues that can confront anyone wanting to exploit WITSML standards.

This Briefing Note describes some of the issues around using a WITSML Datastore as the basis of an Enterprise Drilling Information System



The WITSML™ standards define formats and mechanisms for transmitting drilling, completions, and interventions data between organisations in the petroleum industry. Designed to address the shortcoming of the earlier WITS standard for wellsite data, WITSML first appeared in commercial software in 2002. Since then, the standard has matured and it has now become the most common way to exchange and store wellsite data.

Having operational drilling data captured, transmitted, and persisted in a standard manner offers significant benefits to drilling contractors and E&P Operators. Operational costs of data capture can be reduced, data can be more easily acquired and stored, and a greater range of applications can make use of the data.

However, the practical benefits of the WITSML standards only go so far and where there are gaps, or areas of ambiguity, careful consideration needs to be given to how to deal with them.

This WITSMLstudio paper highlights six data related issues that should be considered when contemplating the use of a WITSML as an enterprise store for drilling data.

Last Updated May 2017

**WITSMLstudio Briefing Note 17/001**

# **WITSML™ Enabled Applications- Six Things to Think About**

## **Executive Summary**

The E&P Industry collects data as it drills its wells. Sometimes this data is aggregated and transmitted to an operator so that they can perform surveillance and to optimise drilling activities. Sometimes it is left to drilling contractors to capture, use, and archive the data.

In recent years, operators have become increasingly more interested in exploiting the data that is captured during drilling operations and in using it for operational decision making, surveillance, and optimisation. As interest has swung from reporting to a greater emphasis on operational decision making, the demand for ever more real-time data has increased and the needs of real-time applications have placed greater demands on the behaviour, quality, consistency, and completeness of the real-time data.

WITSML standards have made significant contributions to the drilling information systems that are in place today. But as the exploitation of this data by operators become more ambitious, they begin to come up against issues that they might not have expected or have given any thought to. This paper identifies issues with respect to the organisation of data, the exchange of data in real time, the consistency of time and depth data, the need for additional data, rig and survey data. It comments on each of these and, where possible, offers options for addressing them.

The extent to which any of these issues impact an organisation will depend on individual circumstances and the flexibility of the vendor server and tooling options. Proactively considering these issues alongside the WITSML capabilities will save heartache, implementation/development costs, and unhappy users.

Contact [witsml@pds.nl](mailto:witsml@pds.nl) for more information

**WITSMLstudio Briefing Note 17/001**

# **WITSML™ Enabled Applications- Six Things to Think About**

## **The Organisation of Data**

A hierarchy of assets is used to breakdown a rig into its component systems, equipment, measuring instruments and measurement points/tags. At the same time the hierarchy can be used to group wells into assets, lines of business, geographical locations and so forth.

WITSML supports a basic asset hierarchy with respect to Wells, Wellbores and Data Objects such as Logs, Trajectories, and Rigs. Anything more sophisticated requires additional data to be separately maintained and merged with the WITSML structures as needed for navigation, reporting, or analysis.

WITSML server vendors generally provide tools that provide drilling-friendly asset management for Wells / Wellbores / Logs / Rigs. However, they may not offer support for elements not explicitly available as WITSML (sub)objects.

WITSML data objects are essentially containers for drilling data with standard elements and attributes to hold the data properties and values that are most commonly used. The standard provides a minimal hierarchy for grouping data together by Well and Wellbore, but it does not provide guidance on how to name or further organize the data.

Data providers may send WITSML data directly to operators following their own naming conventions. This applies to both the objects themselves as well as the log curve mnemonics used in the logs. Dashboard displays and calculation facilities often rely on a standardized naming convention for both objects and for mnemonics to find the data to display. Vendors may provide solutions to help automatically name or rename data objects and mnemonics, but manual work or custom solutions may also be required.

Grouping drilling data by hole section, data provider, data type (time or depth), or data source can be done in a variety of ways. Naming conventions can help users identify data by relevant concepts, but search and hierarchical data views may not be able to use these conventions to find or present the data. If the available standard elements on WITSML objects are populated, then this information can be used by search tools or in presenting the objects (e.g. in a tree or hierarchical view), but this will be dependent on the functionality provided in the vendor solution.

## **Exchanging Data in Real-time**

The persistence mechanism for storing and serving up drilling data needs to be able to store it as it comes in at a second/sub second frequency and to be able to serve it up fast enough to service its community of users. If client applications have demanding needs for interactivity on log plots or time critical applications, the frequency and latency associated with the implementation technology can present problems.

WITSML 1.3.1.1 and 1.4.1 have APIs that are based on SOAP Request/Response protocols these were intended to support 'near real-time' as opposed to 'real-time' exchange of data. WITSML 1.3.1.1 also had a Publish / Subscribe mechanism intended to support real-time exchange, but this is almost universally not used due to issues with getting through firewalls. To address these shortcomings and allow WITSML to be used in real-time scenarios, the Energistics Transfer Protocol (ETP) data exchange specification has been developed<sup>1</sup>.

Designed to replace TCP/IP WITS level 0 data transfers with a more efficient and simple-to-implement alternative, the key advantage of ETP is that data receivers do not have to poll for data and can receive new data as soon as they it becomes available from a data provider. ETP also relies on a

---

<sup>1</sup> <http://www.energistics.org/standards-portfolio/energistics-transfer-protocol>

**WITSMLstudio Briefing Note 17/001****WITSML™ Enabled Applications-  
Six Things to Think About**

persistent connection on traditional HTTP(S) ports (80/443) that can be initiated by either the server or the client to minimize the likelihood that firewalls will block the transfer.

ETP will initially focus on the transfer of data, between:

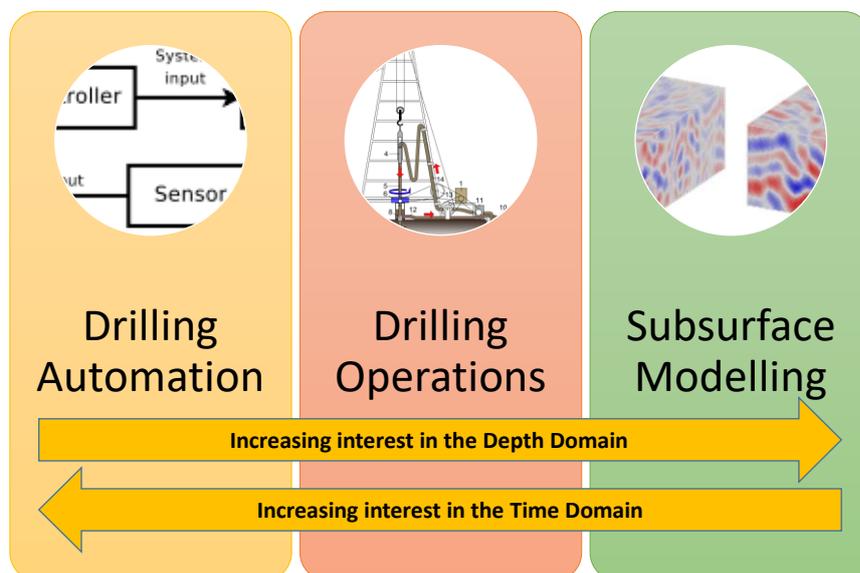
- Wellsite provider to a WITSML store (server),
- WITSML store to WITSML store (replication),
- WITSML store to client applications.

However, ETP is intended to support historical data queries in due course.

Vendors of WITSML Servers are in the process of adopting the ETP protocol in place of the SOAP protocol and capabilities based on the SOAP protocol should plan for this change.

**Consistency between Time and Depth Data**

Depending on the constituency of the user, drilling data is primarily used in the time domain, the depth domain (MD/TVD), or a combination of both.



*Figure 1: Spectrum of interest in access to drilling data by time and depth*

A WITSML server will generally store data against time and depth. However, the standard makes no stipulations as to how, data that can be accessed by time and depth, should be maintained to ensure consistency of operations (such as update or delete) between the two domains.

WITSML Server 1.3.1. and 1.4.1 implementations may store the data twice, once against time, and once against depth. And once stored, the parameter value against time may not be linked to the parameter value against depth. WITSML 2.0 allows for the two domains to be merged together as each channel can have multiple indexes: Time, Depth, Pressure, and more.

**WITSMLstudio Briefing Note 17/001**

# WITSML™ Enabled Applications- Six Things to Think About

If the relationship between time and depth is subsequently modified (to correct for errors (say, affecting MD) or to process new survey results (say, updating TVD)), the consistency of data across the two domains may be left to the user to sort out.

For depth data, WITSML servers do not typically record information about how the depth data was derived from time data. The actual derivation may vary by data provider and can vary by measurement type. If recording how the derivation was done or reproducing the depth data from the time data is important, agreements may need to be made with data providers to get the appropriate information or data.

## Storing Additional Data

Data in WITSML is exchanged in XML documents focussed on specific drilling concepts and WITSML schemas define standard information content for such items as Logs, Rigs, Wells, Wellbores.

An operator's implementation of a Drilling Information System based on WITSML, will likely need to store data that is not covered by the standard - such as application specific data fields. It may also "overload" existing objects to be able to introduce concepts, extend data footprint, and support functionality that is not supported by the standard. For example, the Rig schema might be overloaded to represent both the concept of rig with master reference data and the concept of an instance of a rig associated with operations on a specific wellbore. Another example might be the creation of 'special' wells to contain application settings.

A custom software solution, using a WITSML server as a primary data store, may also need to find a way to persist application data such as user preferences, security profiles, logs and configuration data.

Any vendor specific, application specific, or business specific data that does not fit into the standard schema attributes can be exchanged through an WITSML schema extension mechanism. However, using WITSML extension mechanisms to exchange (and in a sense have the WITSML Server 'store') non-standard data or the use of data overloading, reduces interoperability since the data can only be interpreted through the use of private knowledge. And if an appropriate schema does not exist, or the data is not handled well by the provided extension mechanisms, the data needs to be handled through special processing or stored elsewhere.

Supplementary tools in commercial offerings of WITSML implementations generally provide some support for custom attributes. Although some tooling may allow custom attributes to be inserted and retrieved, facilities to restrict the entry of data into custom attributes to specific data types, or to validate data entry are generally absent. This can create problems when the data is subsequently used in calculations or workflows.

Additionally, vendor solutions may store supplementary data specific to their solutions in the WITSML data store. Often this is only accessible using proprietary extensions to the standard or through the vendor's own tools. Giving custom or 3<sup>rd</sup> party tools access to this data requires cooperation from the vendor or may simply not be possible.

## Managing Rig Data

Rigs perform operations on different Wells/Wellbores, and Wells/Wellbores can be operated on by the same or different Rigs over time. An enterprise Drilling Information System needs to be able to navigate these changing relationships.

Being an exchange standard rather than a database design standard, WITSML is oriented to describe the current configuration and reality. The Rig is essentially treated as an attribute of a Wellbore and is

**WITSMLstudio Briefing Note 17/001**

# **WITSML™ Enabled Applications- Six Things to Think About**

intended to reflect the state of the Rig at the time it operated on the Wellbore. This means that master data about Rigs can be difficult to manage, information about the relationship between Rigs and Wellbores as it changes through time can be difficult to represent, and efficient access by Rig may be problematic.

A place for most Rig data can be made through the use of custom attributes – but with the attendant lack of support and interoperability issues.

## **Handling Survey Data**

Surveys are crucial to many aspects of the drilling process and WITSML has a native data object to describe surveys and specific rules and functionality for constructing and modifying them. However, the standard does not specify mechanisms for dynamically updating the relationship between MD and TVD as new survey stations are taken, or existing survey stations are modified or corrected.

Receiving survey data via WITS0 has particular limitations that need to be considered; specifically, that there is no guaranteed way to uniquely identify a previously received survey station that needs to be replaced with a corrected station, and there is no standard representation of the raw magnetic and gravitational readings used to generate the inclination and azimuth. Additionally, vendor tools for converting WITS0 data to WITSML may be limited to creating log data from WITS0 data. Populating a trajectory from survey data received via WITS0 may require additional tools from the vendor or a custom solution.

**WITSMLstudio Briefing Note 17/001**

# **WITSML™ Enabled Applications- Six Things to Think About Supplementary Notes**

## **What is WITSML?**

WITSML is an E&P community developed standard designed to facilitate “the ‘right-time’ seamless flow of well-site data between operators and service companies to speed and enhance decision making”<sup>2</sup>. Originally (October 2000) intended to replace WITS, and based on a use case scenario focussed on the communication of drilling data from the Rig to the Office, the WITSML standard has become a collection of XML schemas that define the content of messages containing different types of drilling data, and a protocol to exchange data in XML documents conforming to those schemas.

The WITSML standards are curated by the WITSML Special Interest Group which is facilitated by the Energistics Energy Standards organisation in Houston<sup>3</sup>.

A WITSML server (or WITSML store) is an implementation of a data store that supports Create, Read, Update, and Delete functionality using the WITSML protocols and schemas. It is a common assumption, but a false one, that a WITSML data store necessarily persists data in the form specified by the WITSML schemas. The WITSML specification standardises to access drilling data, not the mechanisms of its persistence.

The WITSML standard is not static, it evolves and matures. Version 1.3.1.1 is the most widely deployed, although the advantages of V1.4.1.1 are increasingly being recognised, and so support for it has been growing. Commercial WITSML stores will often offer support for multiple versions of the standard.

V2.0 of the WITSML standard was released in December 2016. As compared with V1.4.1.1, V2.0 has a revised set of data objects and replaces the Web services with the Energistics Transfer Protocol (ETP)<sup>4</sup>, which is part of the new Energistics Common Technical Architecture (CTA) and designed to replace the web service APIs for the RESQML and PRODML standards as well.

As the WITSML standard has matured, and as operators have promoted its adoption in drilling contracts, drilling contractors have offered more and more support for it over the years and it is fast achieving its goal of becoming the lingua franca for drilling data. Today:

- WITSML is widely understood and supported by the drilling community and its vendors.
- Several vendors offer competing end to end solutions based on WITSML.
- WITSML is considered to have the data footprint required for most of the uses to which drilling data may be put.
- 3<sup>rd</sup> party tools can be used in a plug and play scenario with a WITSML server - while an individual vendor’s solution may have proprietary infrastructure and processing, the data is accessed using standard mechanisms and can be easily exploited by 3<sup>rd</sup> party and/or internal applications.
- WITSML is a common mechanism for importing drilling data into subsurface applications, and it is the most common mechanism for getting real-time data into subsurface applications.

---

<sup>2</sup> <http://www.energistics.org/drilling-completions-interventions/reference-materials>: Understanding the WITSML Standards, Part 1, slide 4, Alan Doniger, Pemex, 21 August 2008

<sup>3</sup> <http://www.energistics.org/>

<sup>4</sup> <http://www.energistics.org/standards-portfolio/energistics-transfer-protocol>

**WITSMLstudio Briefing Note 17/001**

# **WITSML™ Enabled Applications- Six Things to Think About**

## **WITSML Sever Implementations**

As commoditised as the WITSML server market might be thought to have become, vendor implementations of the WITSML standards are not all equal. They don't all cover all the objects, they support different versions of the API, they may support different API operations, they are built on different implementation technology, and they offer different capabilities in supplementary tooling and bespoke functionality.

When choosing a WITSML store as the core repository for a drilling information solution, one can expect the vendor to provide solutions for many of the issues commented on in this paper. However, as this paper illustrates, additional effort will likely need to be spent on:

- Filling in gaps not adequately met, or not cost effectively met, by vendor solutions.
- Defining, implementing and maintaining data naming and organization conventions.
- Integrating with process control / drilling automation systems.
- Managing security.
- Managing data quality.
- Tooling to allow R&D engineers interact with the data to support their investigations/experiments.
- Handling data needs not covered by the WITSML standard (including the provision of maintenance and validation capabilities).
- Implementing Calculations / Calculation Engine with precise or complex behaviours.

In the sections that follow, this paper considers the following challenges that a WITSML based solution for managing drilling data needs to address:

- Storing non-standard data in WITSML can be a problem.
- The standard WITSML API has constraints.
- Using WITSML data in calculations can be problematic.
- Managing WITSML data requires effort.

The extent to which any of these issues is important to any particular implementation will depend on the scale of operations, the level of ambition, and the implementation of the selected WITSML server and associated tooling. Some of them will be addressed by vendor capabilities but others imply investment in custom software development and additional data storage capabilities.

**WITSMLstudio Briefing Note 17/001**

# **WITSML™ Enabled Applications- Six Things to Think About**

Petrotechnical Data Systems (PDS) is based in the Netherlands and since 1993 has been providing IT consulting, product development and research and development of innovative IT solutions for core business processes in the Exploration and Production (E&P) industry.

Since 2005 PDS' has operated a Centre of Expertise for Petrotechnical Software Technology in Rijswijk, The Netherlands. This facility houses 150 computer scientists and E&P specialists undertaking IT research, and the development, support and maintenance of subsurface and well applications and associated integration technologies for its Oil and Gas clients.

In 2009, PDS opened its near shoring centre in Sofia offering software development and quality assurance services to the rest of the PDS Group as well as to the local market.

In 2011, PDS opened its Houston facility which undertakes projects and consulting work for PDS' US based E&P clients as well as joint projects with other PDS centres. Houston is also the hub of PDS's open source product and custom development WITSML services.

[WWW.PDS.NL](http://WWW.PDS.NL)