Experiences from a Large Scale Ontology-Based Application Development

Ontology Summit 2012

David Price, TopQuadrant

Agenda

- Customer slides explaining EPIM ReportingHub
- TopQuadrant slides explaining ontology-related topics

EPIM

E&P Information Management Association

EPIM ReportingHub (ERH)

A new powerful knowledge sharing platform

ECIM , September, 2011

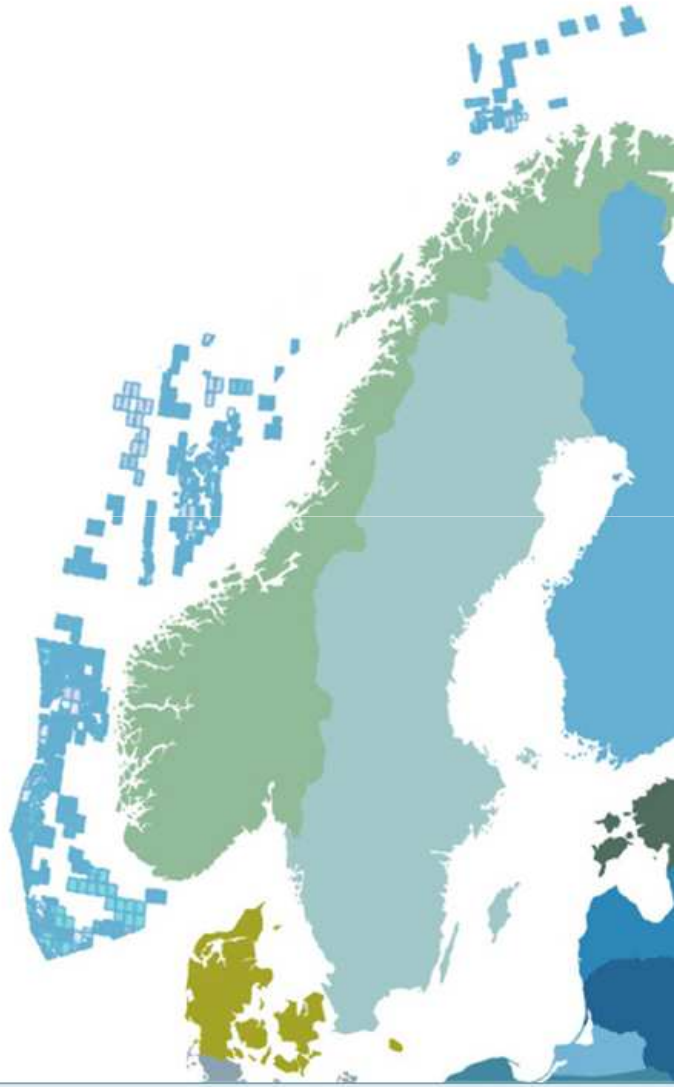
TopQuadrant™

EPIM

E&P Information Management Association

EPIM is the instrument for the operators on the Norwegian Continental Shelf to secure efficient information sharing among all relevant stakeholders by providing cost effective and user friendly common digital solutions based on international standards

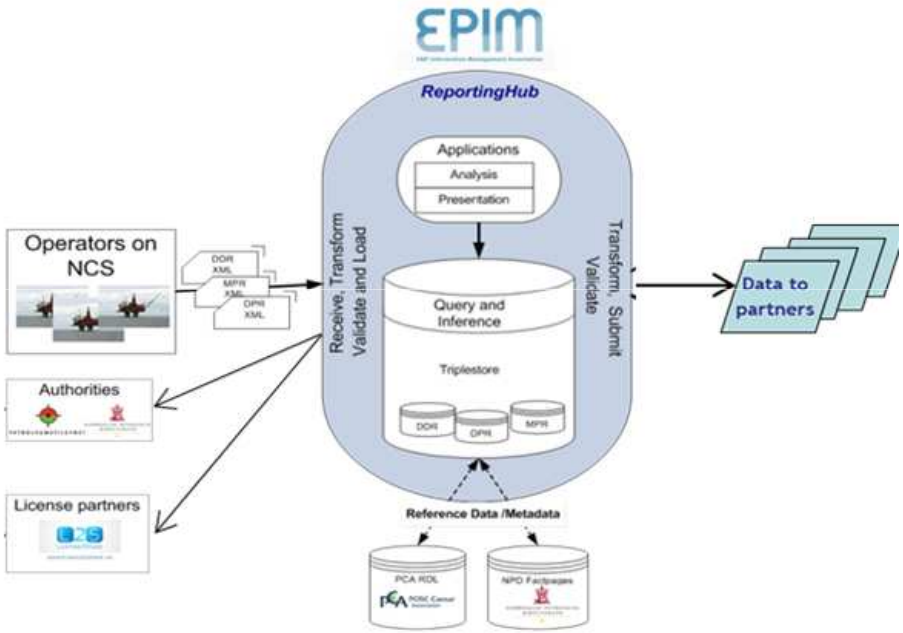
- License2Share (L2S)
- EqHub
- EnvironmentHub
- EPIM ReportingHub (ERH)



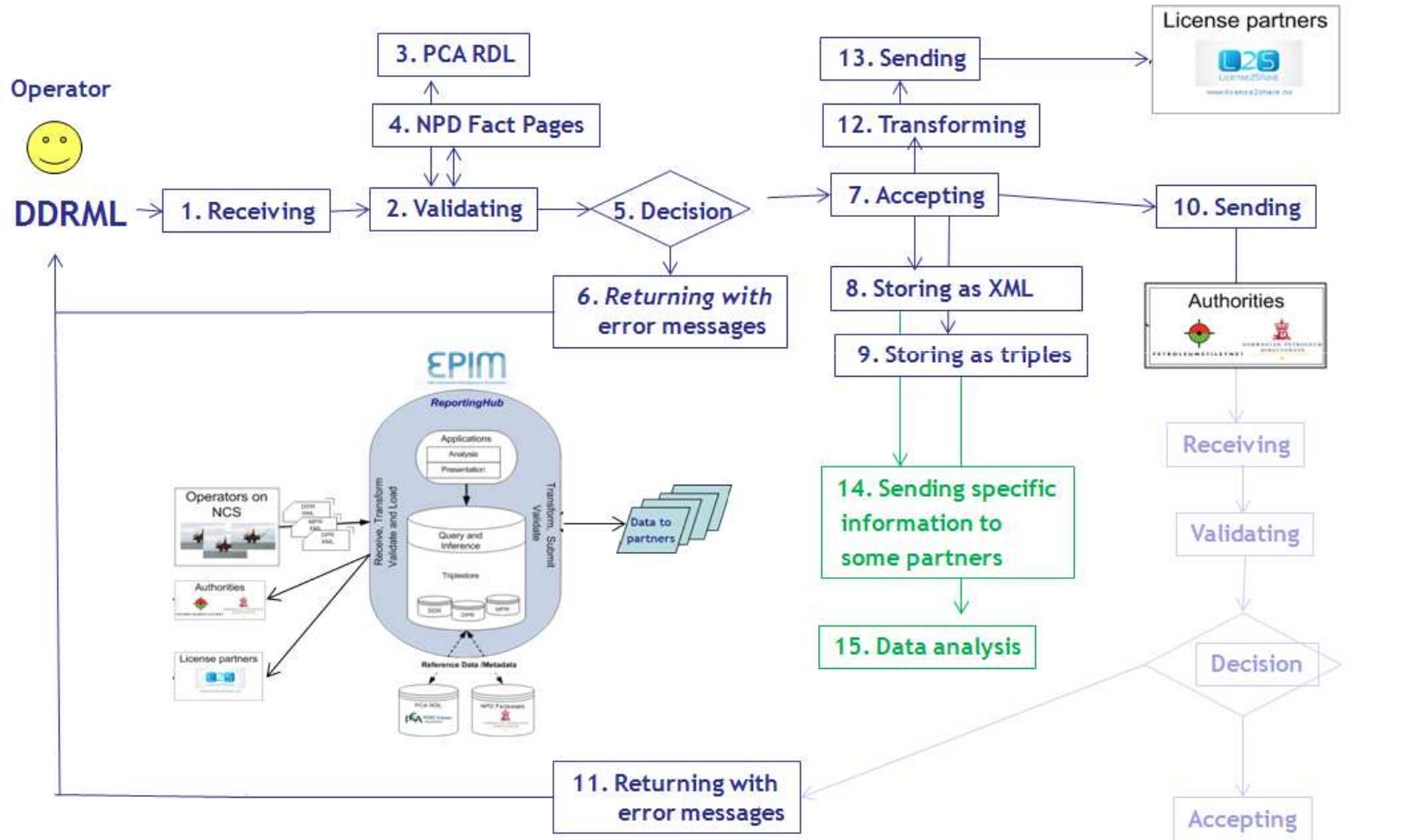
EPIM E&P Information Management Association

ERH – Main Objectives

- Shall receive drilling and production reports from operators on the NCS, validate, store and send them to authorities and partners
- Shall provide drilling and production data to the partners on request
- Shall be flexible for changes in the reports and easy to extend to new types of reports
- Shall be based on NPD Fact Pages and PCA E&P ontology



Key elements of the Daily Drilling Reporting (DDR)



EPIM E&P Information Management Association

ERH - Semantic Web

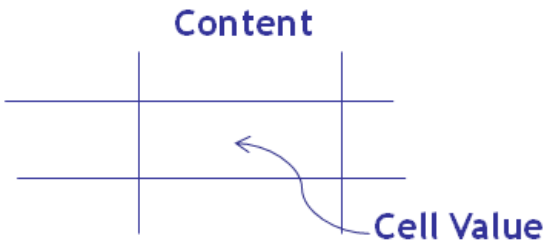
Evolution:

- Web 1.0 - Pages and documents
- Web 2.0 - Social networking
- Web 3.0 - Semantic Web
  - ✓ Resource[1] Description Framework (RDF)
    - RDF is a distributed data model on the Semantic Web consisting of a triple

  - ✓ Web Ontology Language (OWL)
    - OWL is very expressive language and provides bases for reasoning - all written as triples
  - ✓ Simple Protocol and RDF Query Language (SPARQL)
    - Query on RDF triples - similar to SQL for RDB

[1] A resource is anything that someone want to talk about and that has a unique address through a Uniform Resource Identifier (URI)

Table



Any table of data might be expressed as RDF triples where

- ✓ Subject is the row number
- ✓ Predicate is the column content
- ✓ Object is the cell value

Subject and Predicate are resources

Why ontologies?

The real world is complex and changing, you need a solution that can cope with that complexity and adapt with the changes.

That's what ontology does for you

ERH - Summing up

ERH is a powerful knowledge sharing platform which provides huge flexibility:

- The technology can model the complexity of the world and easily adapt to changes
- The reports can easily be updated (adding/deleting data) and the triplestore will handle such changes without any modification
- The ontologies allows us to do reasoning such as validation
- The data has now well defined meaning so they can be used outside ERH
- Triplestores can be build one at a time and query all of them as one database
- Easy to transfer data between triplestores without loss of information
- More and more public data is available as RDF triples (open linked data)

ERH - Summing up

EPIM is now in very good position for offering the operators enhanced services by building an integrated report system covering all reports from the operators to the authorities and the partners.

Some of these reports are:

- ✓ Monthly sales of gas
- ✓ Monthly economic reporting
- ✓ Monthly HSE reporting
- ✓ Revised national budget
- ✓ RiskNivå Norsk Petroleumsvirksomhet (RNNP)
- ✓ Yearly environment report
- ✓ Yearly reports to Statistics Norway (SSB)
- ✓ Yearly reports to Oil and Gas Producers (OGP)

This will greatly improve the partners ability to analyse all the data in the reports in an integrated holistic way.

Scale of EPIM ReportingHub

- 300 million triples in the next 4 years
  - Potential for many more if data sources added or historical data imported
- 40+ concurrent users
  - REST/SOAP services too
- Source XSDs have 2000-ish elements -> resulting ontologies have 900-ish classes and 900-ish properties
- Delivered on an SaaS basis with high availability Service Level Agreement
  - RDF database replication and warm fallover
  - Secondary app server running at all times

TopQuadrant Viewpoint

- Ontologies are software artifacts - part of an application that has a purpose
  - Purpose is not always to represent the real world, not always to enable reasoners
  - Creating and using ontologies has challenges, but it's not a lot harder than other software engineering tasks
- Applications that use ontologies often have competing and conflicting requirements and require trade-offs
  - E.g. How can my ontology 1) support change and aggregation using 4-dimensionalism, 2) support DL reasoning, 3) be understandable by software engineers and 4) have good query performance?
  - Answer … it cannot

Use of Ontologies

- We use ontologies for everything, literally, and we use RDF/OWL
  - E&P ontologies cover the domain and are based in ISO 15926
  - Proxy ontologies of XML Schemas for Daily Drilling, Daily Production, and Monthly Production allow XML-as-OWL data
  - Ontology of NPD CSV/spreadsheet allows CSV-as-OWL data
  - Ontology of SPARQL is part of SPIN W3C submission
  - Ontology of functions similar to software library of functions allows packaged SPARQL, Java or JavaScript code
  - Ontology of stored SPARQL as REST-like service returning JSON, XML, CSV, RDF

Use of SPIN/SPARQL

- These are critical ontology-related technologies … ontologies on their own do very little for an application
- SPIN is SPARQL Inference Notation, W3C member submission
  - Makes using SPARQL for constraints, transforms, etc. simpler
- We use SPIN/SPARQL for transformation and data validation
  - Norwegian Petroleum Directorate Facts as CSV -> SPIN/SPARQL Construct E&P data
  - Platform Operator Report XML -> SPIN/SPARQL Construct E&P data
  - Validation of input data against NPD Facts -> SPIN Constraint as SPARQL
- We use SPARQL Web Pages for report generation
  - E&P data -> SPARQL embedded in HTML Report for Partners in License

Challenge : The Ontology Team

- Disparate view, approach, background of ontologists working together
  - E.g. ‘zealots’ with respect to their flavor of ontology
    - can always do it better and cannot believe how little understanding followers of other ontology religions have
    - results often vary and have to be integrated by single ontology lead at the end
  - E.g. engineers turned ontologists often don't have the necessary background/skills
    - Early mistakes become legacy that has to be ‘worked around
- These were more of a problem wrt background requirements on the project, not as much within the team, but even then they impacted the project

Challenge : Ontology Cost

- Large scale ontology development is expensive
- Is also engineering so suffers from ‘At some point you just stop’ syndrome
  - Reuse is often hard because organizations don't have the money to complete the ontology work
  - things left in not-quite-good-enough state for future use when the money runs out
  - … and the money always runs out
  - yet organizations often insist on reuse of previous work to recoup investment … which costs them more

Challenge : Ontology as Software

- Ontology team meets Software team can be a clash of cultures
  - E.g. Software developers can prove their code works, ontologists cannot
  - It's hard to test an ontology (e.g. test data?)
  - Ontological purity not high priority for Java or XML Schema developers
  - In the end, the software development team owns the application, not the ontologist

Challenge : Practicalities

- Infrastructure supports software development far better than large-scale ontology development, yet ontology is ‘just another software artifact’ in large scale apps
  - E.g. diff utility helps software developers track change, but doesn’t work on ontology files
  - E.g. Cannot SPARQL with ‘label’ rather than URI so human readable URIs are an issue that tools are not going to handle
- Documentation of ontologies is often vague/ambiguous
  - It means whatever the author says it meant when you manage to find them and ask

Summary wrt EPIM ReportingHub

- Semantic analysis of XML schema to E&P/ISO 15926 ontologies is not a quick process
- We used Franz's AllegroGraph for this project – there are scalable, enterprise triplestores now
  - Supports ACID commit/rollback, warm standby, replication, etc.
- Transformations using SPIN/SPARQL make mappings and semantics visible to non-programmers
- SPIN/SPARQL is also how we will project from E&P ontology into DL-safe ontologies, full ISO 15926, etc.