BPMN 2.0 Primitives and Semantic Technology
Proof of Concept

Linus Chow
Principal BPM Champion, Public Sector
AIIM Ambassador
WfMC Public Sector Chair
13 July 2011
Why Semantic Enabled Business Models

• Business Challenges*
  • DoD spends more than $6.0B annually on building and maintaining over 2,000 systems and services
  • Systems and Processes are poorly integrated
  • Many redundant capabilities without a Holistic Enterprise View

• Combining Standards-Based Semantic and BPM Technology:
  • Supports Business Enterprise Architecture and DoDAF
  • Enables Federated Sharing of Process and Analytical Information
  • Helps Standardize Business Processes and Best Practices
  • Provides Transparency and Collaboration
  • Agile Development Life-Cycle Accelerating Capabilities to Market

*Reference: Memorandum for Secretaries of the Military Departments
Subject: Use of End-to-End Business Models and Ontology in DoD Business Architectures
Elizabeth A. McGrath, April 4 2011
Goals/ Objectives/ Deliverables

**Phase 1** *(POC)*
- Use Primitives BPMN 2.0 to construct a business process
  - Integrate Process with RDF Triplestore
  - Enable Run-Time Primitives BPMN 2.0 Process (with no conversion)
- Have the business process interact with OWL model and output Triples
- Query Triple Store to confirm results
- Store Oracle BPMN 2.0 Primitives in OWL
- Prove COTS BI Queries using SPARQL

**Phase 2** *(ongoing)*
- What is the delta between Oracle BPMN 2.0 Metadata and the Metadata needed to store full capability in OWL
- What implications to the future BPMN standard and other technology standards
Semantic E2E Architecture

Multiple RDF Sources

Enterprise Data Stores

Regional Data Stores

External Data

Common Vocabulary

Business Analytics

Standard Process
Transparency

*Source Business Mission Area CTO/CA Office of the DCMO
Proof of Concept: BPMN 2.0 Primitives Process
BPMN 2.0 Ontology

Design Principles

- Keep it simple but complete
- Close to BPMN 2.0 UML Metamodel
- Use RDF-S subClassOf for conceptual subclasses
- Object and Data Properties with meaningful names, domains and ranges
- BPMN 2.0 UML classes that got introduced for technical reasons (for containment and/or sub-classing) are not mapped to OWL
- No OWL model that is cluttered with entities from an automatic conversion from BPMN 2.0 XML-Schema or XMI.
- SPARQL queries on the model must be expressive and easy understood by both, Semantic Web AND BPM community
BPMN 2.0 Ontology

• Current Status
  • All BPMN 2.0 Flow Elements mapped to OWL classes
  • Verified by manually creating Triples for a sample BPMN process
  • Deployed Triples to Oracle Database and performed sample SPARQL Queries

• Next Steps
  • Automatic creation of (and underlying BPMN 2.0 process models, WSDL, XSD etc.)
  • Map missing BPMN 2.0 concepts to OWL classes
    • Data, Input, Output, Data Associations etc.
    • Service Model, Messages, Message Flow and Correlation
  • Provide extensions for Human Workflow, Organizational Data, Business Rules etc.
Example Triples from BPMN Process

- Sequence Flow between BPMN 2.0 Exclusive Gateway and User Task
- Shows object property for ‘inLane’
SPARQL Example I: Select User Tasks in Lanes

```
SELECT ?usertask ?lane
WHERE
```

Oracle SPARQL Endpoint Query Results

<table>
<thead>
<tr>
<th>usertask</th>
<th>lane</th>
</tr>
</thead>
</table>
SPARQL Example II:
Select all Flow Elements from Lane DCMO

```
SELECT ?flowelement
WHERE
{ ?flowelement rdf:type bpmn:Flow . ?flowelement bpmn:inLane p1:DCMO }
```
SPARQL Example III:
Select all Flow Elements that have sequence flow to "Approve Metric for Release"

```
SELECT ?source ?target
WHERE
  ?flowelement bpmn:hasTarget p1:Approve_Metric_for_Release }
```

Oracle SPARQL Endpoint Query Results

<table>
<thead>
<tr>
<th>source</th>
<th>target</th>
</tr>
</thead>
</table>
Executing the BPMN 2.0 Primitives Process
Instantiate the Process
Running Primitives BPMN 2.0
Human Interaction with the BPMN 2.0 Primitives Process
By-Directional Integration Between Runtime BPMN 2.0 Primitives and RDF
Executing Process Updates RDF
# Query RDF Triplestore

## Oracle SPARQL Service Endpoint using Joseki

Simple query test interface. You can put in your own query. The queries are submitted against the dataset specified in Joseki's configuration file.

### SELECT - get variables (apply XSLT stylesheet)

```sparql
PREFIX do: <http://purl.org/dc/elements/1.1/>
PREFIX dcf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX dfa: <http://www.w3.org/2008/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX fn: <http://www.w3.org/2005/xpath-functions#>
PREFIX jena: <http://www.apache.org/jena/jena-core/1.0.2/jena-core.jar#>
PREFIX jena-api: <http://www.apache.org/jena/jena-api/1.0.2/jena-api.jar#>
PREFIX jena-pretty-print: <http://www.apache.org/jena/jena-pretty-print/1.0.2/jena-pretty-print.jar#>
PREFIX jena-util: <http://www.apache.org/jena/jena-util/1.0.2/jena-util.jar#>
PREFIX oracle: <http://oracle.com/jena/jena-adaptors/ext/extension3#>
PREFIX oracle-jena-sparql-endpoint: <http://oracle.com/jena/jena-sparql-endpoint#>

SELECT ?agency_name ?abbr ?budget_amount ?quarter
WHERE {
  ?agency_name a jena:Resource .
  ?agency_name jena:hasAbbreviation ?abbr .
  ?agency_name jena:hasBudgetAmount ?budget_amount .
  ?agency_name jena:hasQuarter ?quarter .
}
```

### Oracle SPARQL Endpoint Query Results

<table>
<thead>
<tr>
<th>agency_name</th>
<th>abbr</th>
<th>budget_amount</th>
<th>quarter</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="http://localhost/Defense_Advanced_Reasearch_Projects_Agency4">http://localhost/Defense_Advanced_Reasearch_Projects_Agency4</a></td>
<td>&quot;DEF. ADV. RESEARCH PROJ.&quot;</td>
<td>&quot;4630383&quot;</td>
<td>&quot;FY2010Q4&quot;</td>
</tr>
<tr>
<td><a href="http://localhost/United_States_Navy">http://localhost/United_States_Navy</a></td>
<td>&quot;UNITED STATES NAVY&quot;</td>
<td>&quot;1083024&quot;</td>
<td>&quot;FY2010Q4&quot;</td>
</tr>
<tr>
<td><a href="http://localhost/United_States_Navy">http://localhost/United_States_Navy</a></td>
<td>&quot;UNITED STATES NAVY&quot;</td>
<td>&quot;392679&quot;</td>
<td>&quot;FY2010Q4&quot;</td>
</tr>
<tr>
<td><a href="http://localhost/United_States_Navy">http://localhost/United_States_Navy</a></td>
<td>&quot;UNITED STATES NAVY&quot;</td>
<td>&quot;235980&quot;</td>
<td>&quot;FY2010Q4&quot;</td>
</tr>
<tr>
<td><a href="http://localhost/United_States_Navy">http://localhost/United_States_Navy</a></td>
<td>&quot;UNITED STATES NAVY&quot;</td>
<td>&quot;3645.06&quot;</td>
<td>&quot;FY2010Q4&quot;</td>
</tr>
<tr>
<td><a href="http://localhost/United_States_Navy">http://localhost/United_States_Navy</a></td>
<td>&quot;UNITED STATES NAVY&quot;</td>
<td>&quot;0&quot;</td>
<td>&quot;FY2010Q4&quot;</td>
</tr>
<tr>
<td><a href="http://localhost/United_States_Navy">http://localhost/United_States_Navy</a></td>
<td>&quot;UNITED STATES NAVY&quot;</td>
<td>&quot;1397107&quot;</td>
<td>&quot;FY2010Q4&quot;</td>
</tr>
<tr>
<td><a href="http://localhost/Defense_Business_Transformation_Agency4">http://localhost/Defense_Business_Transformation_Agency4</a></td>
<td>&quot;BUSINESS TRANSFORMATION AGENCY&quot;</td>
<td>&quot;3259.01&quot;</td>
<td>&quot;FY2010Q4&quot;</td>
</tr>
<tr>
<td><a href="http://localhost/Defense_Business_Transformation_Agency4">http://localhost/Defense_Business_Transformation_Agency4</a></td>
<td>&quot;BUSINESS TRANSFORMATION AGENCY&quot;</td>
<td>&quot;0&quot;</td>
<td>&quot;FY2010Q4&quot;</td>
</tr>
<tr>
<td><a href="http://localhost/Defense_Business_Transformation_Agency4">http://localhost/Defense_Business_Transformation_Agency4</a></td>
<td>&quot;BUSINESS TRANSFORMATION AGENCY&quot;</td>
<td>&quot;4492335&quot;</td>
<td>&quot;FY2010Q4&quot;</td>
</tr>
</tbody>
</table>
```
Business Analytics Combining Process and Operational Data
SPARQL Gateway

Oracle Specific

OBI EE (BI Server XML Gateway)

HTTP

SPARQL Gateway

Standard Compliant

SPARQL Protocol

SPARQL Endpoint

XSLT
SPARQL Gateway Data Flow

- BI Server XML Gateway sends HTTP request to SPARQL GATEWAY
  - http://<sparql_gateway>/sparqlgateway/sg?query=...

- SPARQL Gateway
  - retrieves the query body
  - makes a call out to a remote SPARQL endpoint
  - receives SPARQL query results in XML
  - performs necessary XSLT transformation
  - serializes XML

- BI Server XML Gateway receives XML metadata
DOD XML Source Summary

Budget Amount

<table>
<thead>
<tr>
<th>Agency Name</th>
<th>Budget Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Defence Advanced Research Projects Agency</td>
<td>4,603,883</td>
</tr>
<tr>
<td>Defense_Business_Transformation_Agency</td>
<td>0</td>
</tr>
<tr>
<td>United_States_Navy</td>
<td>3,645</td>
</tr>
<tr>
<td>United_States_Strategic_Command</td>
<td>615,825</td>
</tr>
<tr>
<td>Grand Total</td>
<td>15,117,076</td>
</tr>
<tr>
<td>Agency Name</td>
<td>Abbreviation</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>Defense Advanced Research Projects</td>
<td>DEF. ADV. RESEARCH PROJ.</td>
</tr>
<tr>
<td>Total</td>
<td><strong>DEF. ADV. RESEARCH PROJ. Total</strong></td>
</tr>
<tr>
<td>Defense Business Transformation</td>
<td>BUSINESS TRANSFORMATION AGENCY</td>
</tr>
<tr>
<td>Agency</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td><strong>BUSINESS TRANSFORMATION AGENCY Total</strong></td>
</tr>
<tr>
<td>United States Navy</td>
<td>UNITED STATES NAVY</td>
</tr>
<tr>
<td>Total</td>
<td><strong>UNITED STATES NAVY Total</strong></td>
</tr>
<tr>
<td>United States Strategic Command</td>
<td>STRATEGIC COMMAND</td>
</tr>
<tr>
<td>Total</td>
<td><strong>STRATEGIC COMMAND Total</strong></td>
</tr>
<tr>
<td>Grand Total</td>
<td></td>
</tr>
</tbody>
</table>

**Budget Amount**

**Summary Pie Chart**

- **BUSINESS TRANSFORMATION AGENCY**: 29.74%
- **DEF. ADV. RESEARCH PROJ.**: 30.63%
- **STRATEGIC COMMAND**: 20.52%
- **UNITED STATES NAVY**: 19.10%

**FY2010Q4 Budget Amount**

- **BUSINESS TRANSFORMATION AGENCY**
- **DEF. ADV. RESEARCH PROJ.**
- **STRATEGIC COMMAND**
- **UNITED STATES NAVY**

[Edit - Refresh]
Connection Pool - DOD XML2

Connection properties:

- URL refresh interval: Infinite
- URL loading time-out: 5 minutes
- Maximum connections: 10

Query output format:

XML
### Session Information

**Sessions**

Total number of sessions: 6

<table>
<thead>
<tr>
<th>User ID</th>
<th>Host Address</th>
<th>Session ID</th>
<th>Browser Info</th>
<th>Logged On</th>
<th>Last Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>weblogic</td>
<td>67.189.127.252 &amp; comcast.net 67.189.127.252:3defbe.13</td>
<td>127.0.0.1.220c70b</td>
<td>Mozilla/5.0 (X11, Linux x86_64; rv:2.0.1) Gecko/20100101 Firefox/4.0.1</td>
<td>6/8/2011 7:05:57 PM GMT-05:00</td>
<td>6/8/2011 7:18:03 PM GMT-05:00</td>
</tr>
<tr>
<td>weblogic</td>
<td>67.189.127.252 &amp; comcast.net 67.189.127.252:3defbe.13</td>
<td>127.0.0.1.220c70b</td>
<td>Mozilla/5.0 (X11, Linux x86_64; rv:2.0.1) Gecko/20100101 Firefox/4.0.1</td>
<td>6/8/2011 7:05:57 PM GMT-05:00</td>
<td>6/8/2011 7:18:03 PM GMT-05:00</td>
</tr>
<tr>
<td>weblogic</td>
<td>67.189.127.252 &amp; comcast.net 67.189.127.252:3defbe.13</td>
<td>127.0.0.1.220c70b</td>
<td>Mozilla/5.0 (X11, Linux x86_64; rv:2.0.1) Gecko/20100101 Firefox/4.0.1</td>
<td>6/8/2011 7:05:57 PM GMT-05:00</td>
<td>6/8/2011 7:18:03 PM GMT-05:00</td>
</tr>
<tr>
<td>weblogic</td>
<td>67.189.127.252 &amp; comcast.net 67.189.127.252:3defbe.13</td>
<td>127.0.0.1.220c70b</td>
<td>Mozilla/5.0 (X11, Linux x86_64; rv:2.0.1) Gecko/20100101 Firefox/4.0.1</td>
<td>6/8/2011 7:05:57 PM GMT-05:00</td>
<td>6/8/2011 7:18:03 PM GMT-05:00</td>
</tr>
<tr>
<td>weblogic</td>
<td>67.189.127.252 &amp; comcast.net 67.189.127.252:3defbe.13</td>
<td>127.0.0.1.220c70b</td>
<td>Mozilla/5.0 (X11, Linux x86_64; rv:2.0.1) Gecko/20100101 Firefox/4.0.1</td>
<td>6/8/2011 7:05:57 PM GMT-05:00</td>
<td>6/8/2011 7:18:03 PM GMT-05:00</td>
</tr>
<tr>
<td>weblogic</td>
<td>67.189.127.252 &amp; comcast.net 67.189.127.252:3defbe.13</td>
<td>127.0.0.1.220c70b</td>
<td>Mozilla/5.0 (X11, Linux x86_64; rv:2.0.1) Gecko/20100101 Firefox/4.0.1</td>
<td>6/8/2011 7:05:57 PM GMT-05:00</td>
<td>6/8/2011 7:18:03 PM GMT-05:00</td>
</tr>
</tbody>
</table>

### Cursor Cache

<table>
<thead>
<tr>
<th>ID</th>
<th>User</th>
<th>Refs</th>
<th>Status</th>
<th>Time Action</th>
<th>Last Accessed</th>
<th>Statement</th>
</tr>
</thead>
</table>
SET VARIABLE QUERY_SRC='Workflow: SAW_DASHBOARD='Saw_DASHBOARD+'Saw_DASHBOARD_P='Saw_DASHBOARD_P+'Saw_DASHBOARD_P='Saw_DASHBOARD_P+'SELECT s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_10, s_11 FROM |

DOD XML Source.'Agency Name.'Abbreviation' s_1,
DOD XML Source.'Agency Name.'Agency Name' s_2,
DOD XML Source.'Measures.'ID' s_3,
DOD XML Source.'Quarter.'Quarter' s_4,
DOD XML Source.'Budget Amount' s_5,
REPORT_SUM 'DOD XML Source.'Measures.'Budget Amount' BY 'DOD XML Source.'Agency Name.'Abbreviation'.'DOD XML Source.'Quarter.'Quarter' s_6,
REPORT_SUM 'DOD XML Source.'Measures.'Budget Amount' BY 'DOD XML Source.'Agency Name.'Abbreviation'.'DOD XML Source.'Quarter.'Quarter'.'DOD XML Source.'Measures.'ID' s_7,
REPORT_SUM 'DOD XML Source.'Measures.'Budget Amount' BY 'DOD XML Source.'Agency Name.'Abbreviation'.'DOD XML Source.'Quarter.'Quarter'.'DOD XML Source.'Measures.'ID' s_8,
REPORT_SUM 'DOD XML Source.'Measures.'Budget Amount' BY 'DOD XML Source.'Agency Name.'Abbreviation'.'DOD XML Source.'Quarter.'Quarter'.'DOD XML Source.'Measures.'ID' s_9,
REPORT_SUM 'DOD XML Source.'Measures.'Budget Amount' BY 'DOD XML Source.'Agency Name.'Abbreviation'.'DOD XML Source.'Quarter.'Quarter'.'DOD XML Source.'Measures.'ID' s_10,
REPORT_SUM 'DOD XML Source.'Measures.'Budget Amount' BY | s_11
FROM 'DOD XML Source'.

ORDER BY 1, 2 ASC NULLS LAST, 3 ASC NULLS LAST

------------- General Query Info
Repository: SAW_DB玺 Presentation: DOD XML Source

------------- Logical Request [beta]
Query: 0 as c1 GB,
Agency Name.Agency Name as c2 GB,
Agency Name.Agency Name as c3 GB,
Measures.ID as c4 GB,
Quarter.Quarter as c5 GB,
Report_Sum.Budget Amount as [ Agency Name.Abbreviation, Agency Name.Agency Name, Quarter, Quarter, Measures.ID ] as c6 GB,
Report_Sum.Budget Amount as [ Agency Name.Abbreviation, Agency Name.Agency Name, Quarter, Quarter, Measures.ID ] as c7 GB,
Report_Sum.Budget Amount as [ Agency Name.Abbreviation, Agency Name.Agency Name, Quarter, Quarter, Measures.ID ] as c8 GB,
Report_Sum.Budget Amount as [ Agency Name.Abbreviation, Agency Name.Agency Name, Quarter, Quarter, Measures.ID ] as c9 GB,
Report_Sum.Budget Amount as [ Agency Name.Abbreviation, Agency Name.Agency Name, Quarter, Quarter, Measures.ID ] as c10 GB,

------------- The logical query plan
Plan: <1252> [for database 0:0:0]
DL c1 as c1 [for database 0:0:0],
DL c2 as c2 [for database 0:0:0],
DL c3 as c3 [for database 0:0:0],
DL c4 as c4 [for database 0:0:0],
DL c5 as c5 [for database 0:0:0],
DL c6 as c6 [for database 0:0:0],
DL c7 as c7 [for database 0:0:0],
DL c8 as c8 [for database 0:0:0],
DL c9 as c9 [for database 0:0:0],
sum_502999106 by [ DL c8, DL c4, DL c5 ] at distinct [ DL c9, DL c4, DL c5, DL c2 ] as c10 [for database 0:0:0],
DL c11 as c11 [for database 0:0:0],
DL c12 as c12 [for database 0:0:0]
Summary and Next Steps

• Completed Basic Executable Process
• Test Process and Validate Results
  • Process Executes Routes
  • Data Stored Correctly (Operational) to RDF
• Finalize / Discuss Target Ontology for BPMN 2.0
• In Parallel
  • BPMN 2.0 Primitives Conversion using Ontology
  • Business Intelligence Integrated using SPARQL
• BPMN 2.0 Primitives / Triplestore Research
• Implications on Other Technologies
  • Business Intelligence
  • Web 2.0 / Enterprise 2.0
  • Other
Purchasing Review Process

BPMN 2.0 Stored Semantically
For BPMN Primitives, where does RDF come into play?

• Dependency Analysis
  • Mostly: “Given an asset Foo (variable) in the SOA catalog, what is the impact to other assets if I want to modify Foo?” (we can not answer this today)
  • Envisioning a “BPM Ontology” for the assets in the SOA catalog (*)
  • Continuous refinement and extensibility of the semantic model

• Find out new things
  • Enable graph merge with other structured information available on the Web (FOAF, dbpedia, calendars, etc.)

• Semantic Search rather then stupid text search
• Enabler to share assets rather then duplicating assets
  • Today we have a large degree of redundancy and proliferation of metadata all over the place

(*) Related Work: SOA Ontology
BPM Ontology – The Big Picture
Examples

- proc:LoanRequestProcess rdf:type proc:BusinessProcess
- task:HomeLoanTask rdf:type task:UserTask
- org:ScottTiger rdf:type org:User
- org:ScottTiger org:performs task:ApproveHomeLoanTask
- proc:LoanRequestProcess proc:executes task:ApproveHomeLoanTask
- proc:LoanRequestProcess proc:executes svc:CreditCheckService
Demonstration Video

• BPMN 2.0 Executable
• Query Triplestore
• Use Business Intelligence
  • Show it is using SPARQL
• Show BPMN 2.0 Conversion to Triplestore
• Query Stored BPMN 2.0 Triplestore
• Q & A

Provided with the Presentation through the Conference Site
Questions