Making Easy Things Easy and Lowering the Barrier to Entry in caBIG with Semantic Web Technologies

David Hau MD, MS
Acting Director of Semantic Infrastructure
NCI Center for Biomedical Informatics and Information Technology
NCBO Webinar, 12/2010
Making Easy Things Easy - Motivations

- There are *multiple ways of expressing semantics* in the community.
  - Static (informational) semantics:
    - UML
    - OWL ontology
    - HL7 MIF
    - XSD
    - caDSR – ordered lists of NCI Thesaurus concepts for specifying object class and property for each Common Data Element (CDE).
  - Dynamic (behavioral) semantics:
    - UML state machine
    - BPMN
    - WSDL / WADL
    - HL7 Behavioral Framework
    - OMG SoaML
Making Easy Things Easy - Motivations

- There is a **wide spectrum of interoperability needs** in the community.

- **Example:** Dr. Smith creates a new sequence alignment algorithm. Implements it as a service.
  - Lab-wide interoperability – share with other researchers in the lab
    - Possibly syntactic metadata is all that’s needed (what operations, what arguments, datatypes, order…)
  - If algorithm is deemed not useful, no further interoperability is needed.
  - If algorithm gains traction, may want to advance to **Institution-wide interoperability**. Semantics would be needed to capture information model, use case, etc.
  - Ultimately, the most used and most community-vetted services would naturally bubble up to community-level or **Global interoperability**. Fuller semantics may be needed to **make all interoperability aspects explicit**.
Lowering the Barrier to Entry in caBIG

- Support expressing semantics in ways most familiar and easy to the user.

- Support a level of interoperability consistent with a user’s needs - “Just enough semantics”

- **Value added by Semantic Infrastructure 2.0:**
  - Transformation between different formats of semantics
  - Leverage any semantics to realize useful interoperability capabilities, e.g. workflow composition – “A little semantics go a long way”
  - Provide tooling to guide user in providing more semantics to achieve a higher level of interoperability, *only if the need arises.*
Service Aware Interoperability Framework

Idea: Define a grammar / framework to explicitly specify all aspects of a service that have to do with “interoperability”.
Semantic Infrastructure 2.0

... is a repository for storing such interoperability-related info from each service, and allowing retrieval of this info at run-time.
caGrid 2.0

... is a platform which leverages the info from Semantic Infrastructure 2.0 to realize interoperability, with security.
Service Aware Interoperability Framework

• 5 by 3 matrix.
• 5 viewpoints:
  • Enterprise viewpoint – use case
  • Informational viewpoint – static semantics (models, concepts)
  • Computational viewpoint – behavioral semantics (operations, inputs, outputs)
  • (Engineering viewpoint)
  • (Technology viewpoint)
• 3 layers of abstraction:
  • Conceptual
  • Platform independent
  • Platform specific

This completely specifies the interoperability aspects of a service!
A set of services relevant to the goal of a “Learning Healthcare System” for Personalized Medicine.
### SAIF

<table>
<thead>
<tr>
<th></th>
<th>Enterprise Viewpoint</th>
<th>Informational Viewpoint</th>
<th>Computational Viewpoint</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conceptual</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Platform Independent</td>
<td></td>
<td>Trace Up</td>
<td></td>
</tr>
<tr>
<td>Platform Specific</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Conforms** (via Enterprise Conformance and Compliance Framework – ECCF)

**Implementation**
Lower Barrier to Entry for caBIG community

- NCI is fully specifying all enterprise services. This provides a framework of *reusable semantics* for the community.
Lower Barrier to Entry for caBIG community

• Community members will not need to specify their services on all 3 layers – they start with Platform Specific Layer only
  • Owners of caGrid 1.0 services
  • Owners of existing services, for example:
    • NCBI Entrez Utilities Web Service
    • MIT Broad Institute GenePattern modules
  • Owners of existing applications, for example:
    • CanReg5 tool from International Agency for Research on Cancer (IARC), World Health Organization (WHO).
Generate Platform Specific Model

<table>
<thead>
<tr>
<th>Enterprise Viewpoint</th>
<th>Informational Viewpoint</th>
<th>Computational Viewpoint</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conceptual (from existing NCI enterprise service)</td>
<td>Trace Up</td>
<td></td>
</tr>
<tr>
<td>Platform Independent (from existing NCI enterprise service)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Platform Specific</td>
<td>Description of use case</td>
<td>Information model annotated with community ontologies</td>
</tr>
</tbody>
</table>
Start participating in caBIG easily

- **Model Data semantically (LoD)**
  - find ontologies describing your domain
  - decide how data will be expressed in terms of these ontologies
- **Model your services I/O semantically**
  - decide how to describe operations in terms of domain ontologies
- **Populate an instance of the Service Metamodel**
  - name, description, scope, classification, etc
- **Goal:**
  - ~ 1 hr to get a semantically aware service up and running
## GenePattern modules

<table>
<thead>
<tr>
<th>Module Name</th>
<th>Description</th>
<th>Version</th>
<th>Module Repository</th>
<th>Public Server Only</th>
</tr>
</thead>
<tbody>
<tr>
<td>COPA (Cancer Outlier Profile Analysis)</td>
<td>Performs cancer outlier profile analysis</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CopyNumberDivideByNormals</td>
<td>Determines the copy number of a target SNP</td>
<td>3</td>
<td>SNP Analysis</td>
<td>Module Repository</td>
</tr>
<tr>
<td>CoxRegression</td>
<td>Cox regression using clinical data and .dss file</td>
<td>1</td>
<td>Survival Analysis</td>
<td>Module Repository</td>
</tr>
<tr>
<td>CytoscapeViewer</td>
<td>Visualizes a gene network</td>
<td>2</td>
<td>Visualizer</td>
<td>Module Repository</td>
</tr>
<tr>
<td>DownloadURL</td>
<td>Downloads a file from a URL</td>
<td>1</td>
<td>Preprocess &amp; Utilities</td>
<td>Module Repository</td>
</tr>
<tr>
<td>ESPPredictor</td>
<td>Determines which tryptic peptides will generate the highest signal in ESI-MS</td>
<td>1</td>
<td>Proteomics</td>
<td></td>
</tr>
<tr>
<td>ExpressionFileCreator</td>
<td>Creates a RDS or GCT file from a set of Affymetrix CEL files</td>
<td>1</td>
<td>Preprocess &amp; Utilities</td>
<td>Module Repository</td>
</tr>
<tr>
<td>ExtractColumnNames</td>
<td>Lists the sample descriptions from a .res file</td>
<td>2</td>
<td>Preprocess &amp; Utilities</td>
<td>Module Repository</td>
</tr>
<tr>
<td>ExtractComparativeMarkerResults</td>
<td>Creates a derived dataset and feature list file from the results of ComparativeMarkerSelection</td>
<td>3</td>
<td>Gene List Selection</td>
<td>Module Repository</td>
</tr>
<tr>
<td>ExtractRowNames</td>
<td>Extracts the row names from a .res, .gct, or .dss file</td>
<td>3</td>
<td>Preprocess &amp; Utilities</td>
<td>Module Repository</td>
</tr>
<tr>
<td>FeatureSummary/Viewer</td>
<td>Displays a summary of features</td>
<td></td>
<td>Visualizer</td>
<td>Module Repository</td>
</tr>
</tbody>
</table>
Short description file for each module

CoxRegression

Module name: CoxRegression
Description: Cox regression using clinical data and .cls file
Author: Yujin Hoshida (Broad Institute) gp-help@broad.mit.edu
Date: 5/20/2008
Release: 1.0

Summary: Once a genomics marker or predictive model for clinical outcome is built, the next step is to test whether it really associates with outcome of new cohorts, or whether it has better predictive ability compared to already known clinical markers/predictive models. "Survival analysis" modules perform such analyses seamlessly after clustering/marker gene/prediction analyses.

Cox proportional hazard modeling (Cox regression) is commonly used to assess association of variable(s) of interest with time-to-event data (e.g., death, tumor recurrence,...) [1-3]. For the CoxRegression module, a .cls file identifies the phenotype label for each sample and a separate tab-delimited text file defines all other clinical variables. The module generates a result file similar to the one shown below. Interpreting the results requires basic knowledge of biostatistics and survival data analysis.

```
# Cox regression results
tie data handling: Efron approximation
variable selection: none
Call:
```
## Input parameters for each module


### Parameters:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Choices</th>
</tr>
</thead>
<tbody>
<tr>
<td>input surv data filename</td>
<td>clinical data containing survival time and censor status - .txt Format shown below.</td>
<td></td>
</tr>
<tr>
<td>input cls file</td>
<td>class information file - .cls Format described in GenePattern File Formats.</td>
<td></td>
</tr>
<tr>
<td>output file</td>
<td>Name for the output file.</td>
<td></td>
</tr>
<tr>
<td>time</td>
<td>Column name for the survival time field. Must be a column in the input surv data file.</td>
<td>Default: time</td>
</tr>
<tr>
<td>status</td>
<td>Column name for the censor field (event=&quot;1&quot;, no event=&quot;0&quot;). Must be a column in the input surv data file.</td>
<td>Default: status</td>
</tr>
<tr>
<td>variable continuous</td>
<td>Comma separated list of the continuous dependent variables (e.g. age) to use for the analysis. Each variable must be a column in the input surv data file.</td>
<td></td>
</tr>
<tr>
<td>variable category</td>
<td>Comma separated list of the categorical dependent variables (e.g. gender) to use for the analysis. Each variable must</td>
<td></td>
</tr>
<tr>
<td>Conceptual (existing from NES – Outcome Service)</td>
<td>Enterprise Viewpoint</td>
<td>Informational Viewpoint</td>
</tr>
<tr>
<td>-------------------------------------------------</td>
<td>----------------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>Platform Independent (existing from NES – Outcome Service)</td>
<td>Trace Up</td>
<td></td>
</tr>
<tr>
<td>Platform Specific</td>
<td>Description of use case</td>
<td>Information model annotated with community ontologies</td>
</tr>
</tbody>
</table>
Service Hierarchy

Conceptual

Platform Independent
- Cox Regression
- Logistic Regression

Platform Specific
- GenePattern module
- (another implementation)
- GenePattern module
pubmed_100101.dtd
NCBI Entrez Utilities Service (PubMed eFetch WSDL)

```xml
<?xml version="1.0"?>
<wsdl:definitions
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:soap="http://schemas.xmlsoap.org/soap/"
    xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  </soap:targetNamespace>
  <message name="efetchRequest_m">
    <part name="input" element="nsf:efetchRequest"/>
  </message>
  <message name="efetchResponse_m">
    <part name="output" element="nsf:efetchResult"/>
  </message>
  <portType name="eUtilsServiceSoap">
    <operation name="run_efetch">
      <input message=""/>
      <output message=""/>
    </operation>
  </portType>
  </wsdl:definitions>
```

efetch_pubmed.wsdl
CanReg5 tool from IARC (Lyon, France - Morten Ervik, chief developer)

- Open-source tool to input, store, check and analyze cancer registry data
- Functionalities include:
  - data entry
  - quality control
  - consistency checks
  - basic analysis of the data.
- Includes a tool to set up a new database or modify an existing database by adding new variables, tailoring data entry forms etc.
- Uses ICD-O-3 for cancer typography and morphology, and local terminologies for patient demographics etc.
Options for participation in caBIG

- Expose a web service interface for the different modules.
- Create Platform Specific Model (PSM) on SI 2.0.

OR

- Expose relational database as a SPARQL endpoint.
  - E.g. possibly via W3C work on direct mapping of a relational data to RDF, and R2RML: RDB to RDF mapping language – exploratory stage.

- Possibly trace PSM up to semantic and functional profiles in an existing NCI enterprise service – Patient Outcome Service.
- Use / Reuse forms stored in SI 2.0. Use LexEVS 6.0 for value set management / validation, and terminology mapping.
| CanReg5 |
|-------------------------|---------------------|---------------------|
| **Enterprise Viewpoint** | **Informational Viewpoint** | **Computational Viewpoint** |
| Conceptual (existing from NES – Outcome Service) | Trace Up | |
| Platform Independent (existing from NES – Outcome Service) | Description of use case | Information model annotated with common data elements or ontologies |
| Platform Specific | | WSDL / WADL / SPDL / SAWSDL |
Wide spectrum of interoperability needs

- Interoperability needs depend on deployment context:
  - Within lab (~ syntactic interoperability ~ PSM)
  - Within institution (~ semantic interoperability ~ PSM/PIM)
  - Global (~ semantic interoperability ~ PSM/PIM/CIM)
Google Search
A doctor in an oncology outpatient clinic searches for a treatment “personalized” for a patient.

Female patient, age 50, Hispanic, breast cancer, (staging info), (grading info), (pointer to pathology images), (pointer to radiology images), ER+, PR-, HER2+, (other marker info), (other info such as family history, smoking history).

What is the treatment with minimum mortality and maximum quality of life?
Input:

• **Demographics / Family History / Social History:**
  - Female, age 50, hispanic, family history, smoking history

• **Pathology:**
  - Pointer to pathology images, staging info, grading info

• **Radiology:**
  - Pointer to radiology images

• **Lab / Genomic info:**
  - ER+, PR-, HER2+, other marker info
Output:

Treatment with

• Minimum mortality
• Maximum quality of life
Workflow

- Set up a workflow to answer the question. Components include:
  - Clinical trial / Clinical outcomes services
  - Pathology imaging services
  - Radiology imaging services
  - Genomics related services

Question: How can we specify each service so that this workflow composition can be fully or partly automated?
SI 2.0 Conceptual Framework

MOF, M3
- SAIF / ECCF Metamodel
- (Modeling language independent)
- Harmonized / mapped to RM-ODP

MOF, M2
- SAIF / ECCF UML profile
- BPMN, OWL ontology, MIF, WSDL, XSD …

MOF, M1
- Service description (e.g. Patient Outcome Service)

MOF, M0
- Service data and operations
SAIF Metamodel

- Modeling language independent

- Harmonized / mapped to RM-ODP (ISO/IEC 10746)

- Foundational metamodel provides the basis for
  - a repository which embodies all ECCF/SAIF concepts in a modeling language neutral form
  - the concrete foundation upon which multiple modeling languages (UML, HL7/MIF) can consistently apply the ECCF/SAIF concepts
SAIF / ECCF Metamodel
UML Profile for ECCF

- Embodies Semantic Framework for ECCF
- Stereotype: UML Model Element to ECCF Concept;
- Tags: details for Service Specification Documents
- Traceability to SAIF Frameworks, RM-ODP,
- SoaML
- OCL: Structural, Semantic enforcement

Credit: Tom Digre, SemanticBits, LLC.
CBIIT Goals Enabled by ECCF Profile

ECCF Specification Stack

- CIM
- PIM
- PSM

Consistent Documents
- Model Driven Technology Artifacts
- ECCF Service Registry

Computable Semantic interoperability
- validation
- Traceability

reuse

Credit: Tom Digre, SemanticBits, LLC.
Semantic Infrastructure 2.0
Service Development Process

Credit: Tom Digre, SemanticBits, LLC.

Scope
Analysis
Refinement
Technology Binding
PSM and other Target Architecture Artifacts

UML Service Model: an ECCF Specification Stack

import

Standard ECCF Specification Stacks
- BRIDG
- LS-DAM
- ISO Data Types
- etc.

import

HL7 MIF Models

ECCF Registry

Service Specifications and other Docs
Service Specifications and other Document Artifacts

Consistent DITA Artifacts Based on caCIS conventions And specifications

Consistent Target Documents Across Model Platforms

Credit: Tom Digre, SemanticBits, LLC.
Eclipse Modeling Framework UML2 Modeling Tool - Papyrus
caBIG® Platform Independent Model and Service Specification

RC213 Referral Consult Service

0.0.8

Mon Nov 15 05:32:09 EST 2010
1 Introduction
   1.1 Overview
   1.2 Relationship to Standards
   1.3 Relation to the RC213 Referral Consult Service Computationally Independent Model and Service Specification
       1.3.1 Conformance and Compliance

2 Platform Independent Model and Service Specification
   2.1 Overview and Architecture
   2.2 Implementation Considerations
       2.2.1 Assumptions
       2.2.2 Deployment Considerations
       2.2.3 Jurisdictional Domains
   2.3 Information Model
       2.3.1 <Primary Information Object>: Type
       2.3.2 <Primary Information Object>: State Model
       2.3.3 Implementation model
       2.3.4 Error Implementation Model
   2.4 Control Data Type Definitions
       2.4.1 Status/Return Values/Exceptions
       2.4.2 Error Codes
       2.4.3 Query Parameters and Result Sets

3 Interfaces
   3.1 UML Model of Interfaces
   3.2 Operations (Enumeration)
   3.3 Operation Behavior Descriptions
       3.3.1 <Interface>

4 Dynamic Model

5 Profiles
   5.1 Functional Profiles
   5.2 Semantic Profiles

6 Relationship with other services

7 Conformance Statements

8 Appendix A - References

9 Appendix B - Glossary
Generated PIM for Referral Consult Service
Generated PIM for Molecular Annotation Service

caBIG® Platform Independent Model and Service Specification

Molecular Annotation Service

0.1.2

Thu Oct 28 04:32:55 EDT 2010
# TABLE OF CONTENTS

1 Introduction  
  1.1 Overview  
  1.2 Relationship to Standards  
  1.3 Relation to the Molecular Annotation Service Computationally Independent Model and Service Specification  
    1.3.1 Conformance and Compliance  

2 Platform Independent Model and Service Specification  
  2.1 Overview and Architecture  
  2.2 Implementation Considerations  
    2.2.1 Assumptions  
    2.2.2 Deployment Considerations  
    2.2.3 Jurisdictional Domains  
  2.3 Information Model  
    2.3.1 <Primary Information Object> Type  
    2.3.2 <Primary Information Object> State Model  
    2.3.3 Implementation model  
    2.3.4 Error Implementation Model  
  2.4 Control Data Type Definitions  
    2.4.1 Status/Return Values/Exceptions  
    2.4.2 Error Codes  
    2.4.3 Query Parameters and Result Sets  

3 Interfaces  
  3.1 UML Model of Interfaces  
  3.2 Operations (Enumeration)  
  3.3 Operation Behavior Descriptions  
    3.3.1 <Interface>  

4 Dynamic Model  

5 Profiles  
  5.1 Functional Profiles  
  5.2 Semantic Profiles  

6 Relationship with other services  

7 Conformance Statements  

8 Appendix A - References  

9 Appendix B - Glossary
3.3 Operation Behavior Descriptions

3.3.1 MAPathwayAnnotationQuery

3.3.1 MAGeneAnnotationQuery

getFunctionalAssociations

Returns annotations describing a gene's molecular function

- redefines EA_Model::MAS CIM::CIM Computational Viewpoint::ComputationalService::MA-FP1: Gene Annotation Query Profile::getFunctionalAssociations

| Behavior Description | • Client supplies a GeneSearchCriteria instance with a gene symbol or alias and an Organism to search within
|                      | • The case of the symbol or alias is ignored
|                      | • If the Organism is null then all Organisms are searched
|                      | • The system returns the matching Gene objects associated AgentAssociation objects, if any. |

| Pre-Conditions       | |
| Security Pre-Conditions | |

| Inputs | • GeneSearchCriteria
|        | • Description:
|        | • Type:GeneSearchCriteria[1..1]
|        | • Type Description: |

| Outputs | • (return)
|         | • Description: |
ECCF Specification Stack as Model Composition

Credit: Tom Digre, SemanticBits, LLC.
Semantic Workflow Composition

- Collaboration with Eric Prud’hommeaux from W3C Semantic Web Healthcare and Life Sciences Interest Group, and SemanticBits, LLC.

- Expands on “SPARQL Annotations in WSDL” (SPDL) project.

**Assumptions:**
- Web services are described using WSDL
- Web services have their input and output messages annotated with SPAT Annotations.
- A Web service is a tuple $a = (pre, post)$ where both $pre$ and $post$ are sets of RDF triples (RDF graph).

- [http://esw.w3.org/HCLS/ClinicalObservationsInteroperability/caBIGSemanticWorkflowsPrototyping](http://esw.w3.org/HCLS/ClinicalObservationsInteroperability/caBIGSemanticWorkflowsPrototyping)
Semantic Workflow Composition - Framework

- Given a user goal \( u \) and a pool of web services \( WS \), a composition is a sequence of web services \((a_1,a_2,\ldots,a_n)\) where \( a_i = (pre_i,post_i) \) in \( WS \) such that
  - \( S_1 = \text{input} \) entails \( pre_1 \).
  - \( S_2 = S_1 \text{ merge post}_1 \) entails \( pre_2 \).
  - \( S_3 = S_2 \text{ merge post}_2 \) entails \( pre_3 \).
  - ...
  - \( S_n = S\{n-1\} \text{ merge post}\{n-1\} \) entails \( pre_n \).
  - \( S\{n+1\} = S_n \text{ merge post}_n \) entails \( goal \).
Semantic Workflow Composition – Future Iterations

- Introduction of SAWSDL
- Parallelization of services in compositions (only sequences get generated in Iteration 1)
- More intricate Semantic Web Service models, for example WSMO with more intricate pre- and postconditions in WSML or other logical languages
- Adding heuristic functions and filtering techniques to reduce the search space
- Taking into account background/domain ontologies and reasoning (basically amounts to the calculation of implicit triples at each search state); turns fast exponential, thus restricted ontology languages have to be considered
NCI Semantic Infrastructure 2.0
Inception Team

Larry Brem
Raghu Chintalapati
Brian Davis
Tom Digre
Stijn Heymans
Cecil Lynch
Ravi Madduri
Harsh Marwaha
Jim McCusker
Charlie Mead

Zoran Milosevic
Scott Oster
Josh Phillips
Eric Prud'hommeaux
Dianne Reeves
Eric Schripsema
Mukesh Sharma
Harold Solbrig
Stuart Turner