The NASA Constellation Program Ontologies - how they are supporting NASA Constellation Program Data Architecture and its applications.

NASA-Ontolog-KMWG OKMDS mini-series Session-05

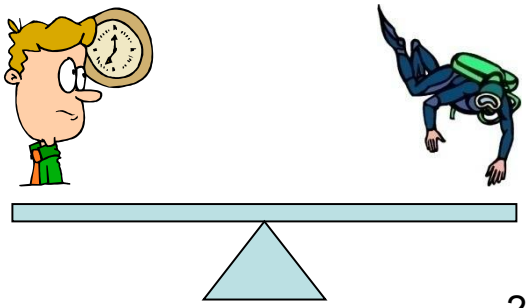
Ontolog,March 20, 2008

Ralph Hodgson, CTO, TopQuadrant, Inc.

http://ontolog.cim3.net/cgi-bin/wiki.pl?ConferenceCall_2008_03_20

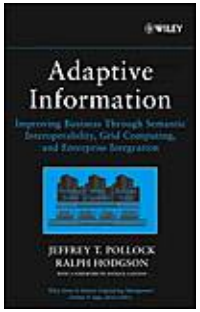
What there is to cover

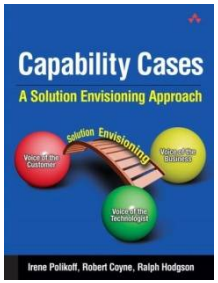
- Introductions
- Ontology Work in the Constellation Program
- Constellation Data Architecture (CxDA)
- NASA's CxDA Initiative
- NExIOM
- Examples from applications of the NExIOM ontologies
- Q & A

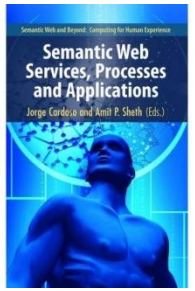


Introductions

- *Ralph Hodgson*
  - *co-founder and the CTO of TopQuadrant, Inc., a US-based company that specializes in semantic technology consulting, training, and tools.*
  - *Prior to starting TopQuadrant in 2001, held executive consulting positions at IBM Global Services where he was a founding member of Portal Practice and Object Technology Practice.*
  - *Prior to IBM, European Technology Director, founder, and Managing Director of Interactive Development Environments, which was an international CASE tools vendor.*
  - *Recent books he has co-authored are Adaptive Information, published by John Wiley in 2004, and Capability Cases: A Solution Envisioning Approach, published by Addison-Wesley in July 2005.*







What is a Data Architecture and the CxDA project

The Data Architecture (DA) provides guidance and a framework to a range of Constellation stakeholders that are data producers and consumers: engineers, managers, technicians, data architects, tool and process developers and users.

The CxDA project and its models, infrastructure, and services define consistent, unambiguous data representations and implement repeatable processes for data exchange in order to enable data sharing within and across Constellation Systems, Organizations, and Missions.

Ontology Work in the Constellation Program has explored many technologies and standards domaine

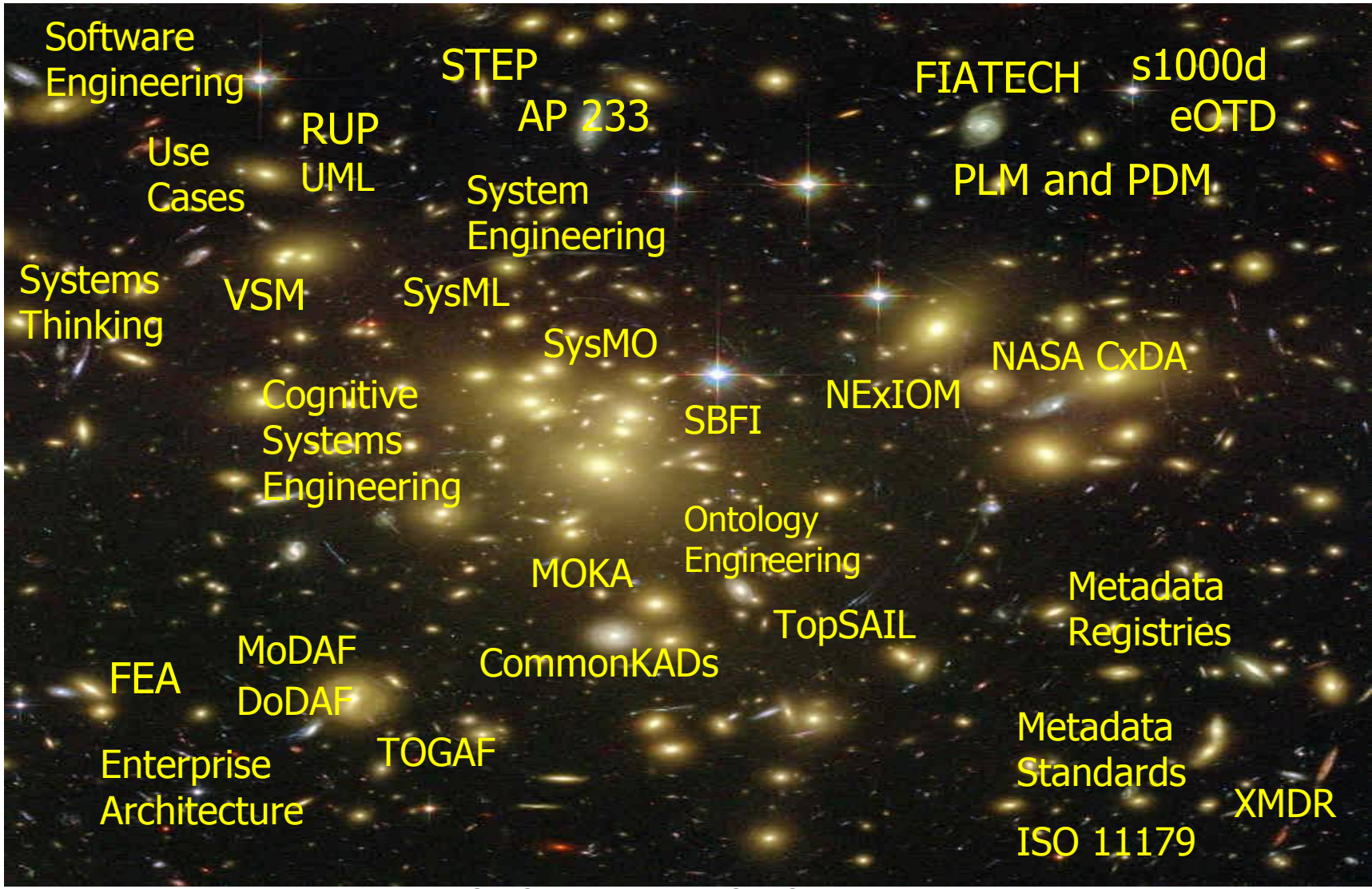


Image source: http://hubblesite.org/newscenter/archive/2003/01/ - Abell 1689 deep space image

A Data Architecture answers questions of the following kinds

- Is this piece of data of this type?
- How is the data from this part of the organization related to data from another part of the organization?
- What are the security constraints on this data resource?
- How do I translate the data from this system for another system?
- Am I registered as a recipient to receive a specific data exchange package if I am working in a given context?
- Can I share this data with a specific party?
- Does this person have the right to perform this operation on this data?
- What are our organization's obligations with respect to this data?
- Where did this data resource come from and what is its accreditation?
- Who owns this data and how many data exchange packages use it?
- Did this organization exchange data with this other organization?
- Are these tools accredited for this type of analysis

Sounds like Metadata, then what does it mean to say "Metadata is Data about Data"?

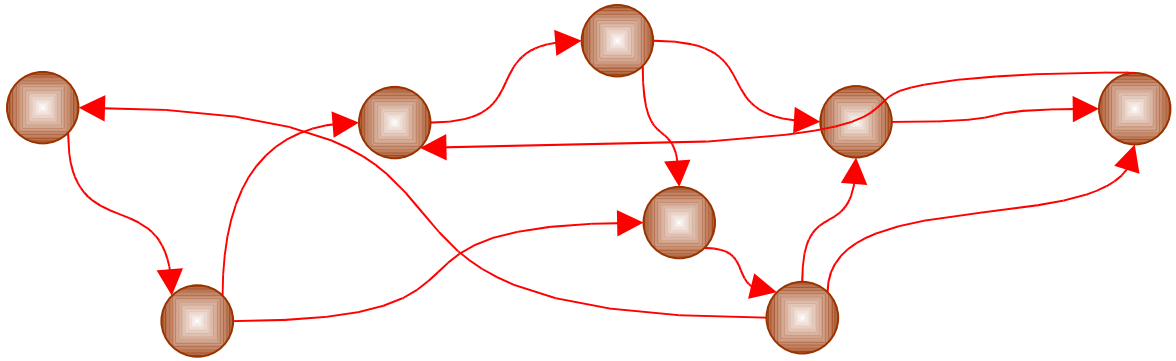
- Descriptive Metadata
  - Format, Data types, Value ranges, Units of Measure
- Provenance Metadata
  - Who produced it, how and when it was produced, how it can be trusted
- Relevance Metadata
  - Who uses it and what value it has
- Governance Metadata
  - Who approves, reviews, and has stewardship
  - Who can access it, confidentiality, licensing and rights
- Infrastructure Metadata
  - The resources needed to manage and control data
  - Long term preservation metadata

But someone's metadata is someone else's data.

- What is metadata, what are attributes depends on the context of use
  - For example, the metadata that states that something is ITAR restricted is not metadata to someone that needs to have a system that provides controls on how documents become available to different parties.
  - The ITAR restricted flag is operational data in the context of a governed content management system.
- Hard to say when Metadata is Non-Operational Data

What is Metadata Really Then?

- Is it
  - Data needed to make Data?
  - Data needed to manage Data?
  - Data needed to relate Data?
  - The language (ala metamodel) needed to express Data?
- Metadata quickly becomes ontology



A Data Maturity Model*

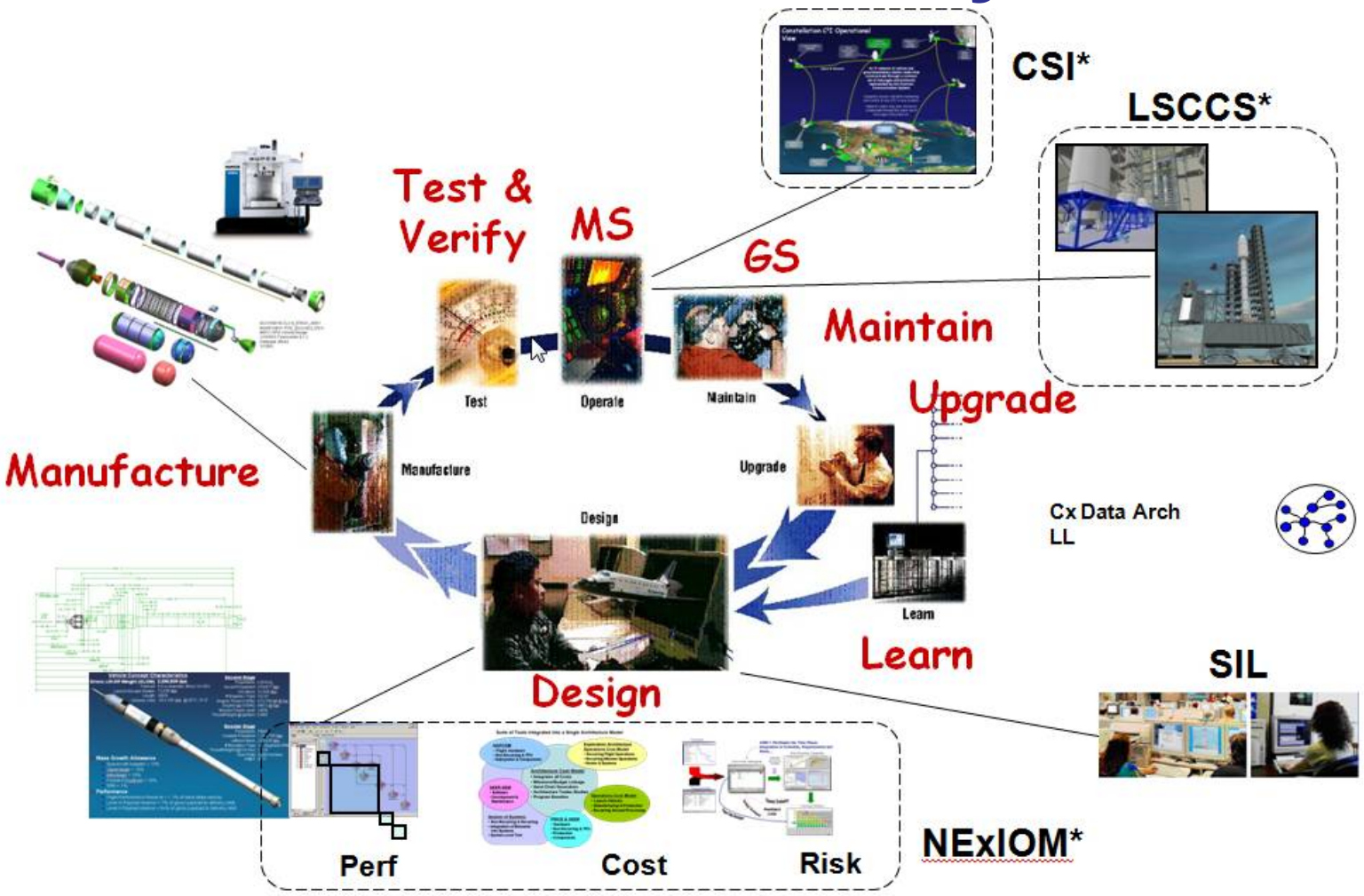
Data Maturity	Data Fidelity	Metadata Type	Data Instance	Data Organization	Metadata Content	Technology	Data Autonomy
5 Optimized Semantics	High Fidelity	Multi-layer Markup	Ontology with Brokering and Mediation	Ontology	Inferred Relationships	Semantic Technology	Smart Data
4 Managed Relationships		Entity & Relationship Markup	Taxonomies with horizontal integration of domain vocabularies	Ontology Taxonomy	Named Relationships	Data Independent	
3 Defined Content		Entity Markup	XML Documents using vertical domain vocabularies	Taxonomy	Metadata Content	Filtering, Clustering. Categorization	
2 Repeatable Structure		Structure Markup	Schemas & DTDs	Documents	Document Standards	Database Tools	
1 Chaotic	Low Fidelity	Un-structured	Text Documents & Database Records	Proprietary Application	Non-Aligned Properties	Search Tools	Dependent Data

* Adapted from: Joel A. Gladding, "Data Maturity Model", SAIC Lead, Defense Intelligence Data Architecture, Presentation April 2005

Metadata expressed in Ontologies

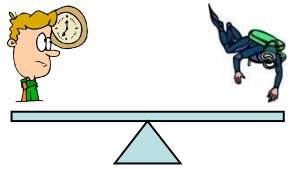
- Still serves as a Data Standard
  - defining attributes and properties of data
  - **Through precise specifications**
- Still helps to
  - develop software applications, databases
  - **By transformations**
- In the form of
  - Models of data and their metadata linked with general relationships, constraints, and rule
  - **As machine-intelligible representations**

CxDA in the CxP Lifecycle

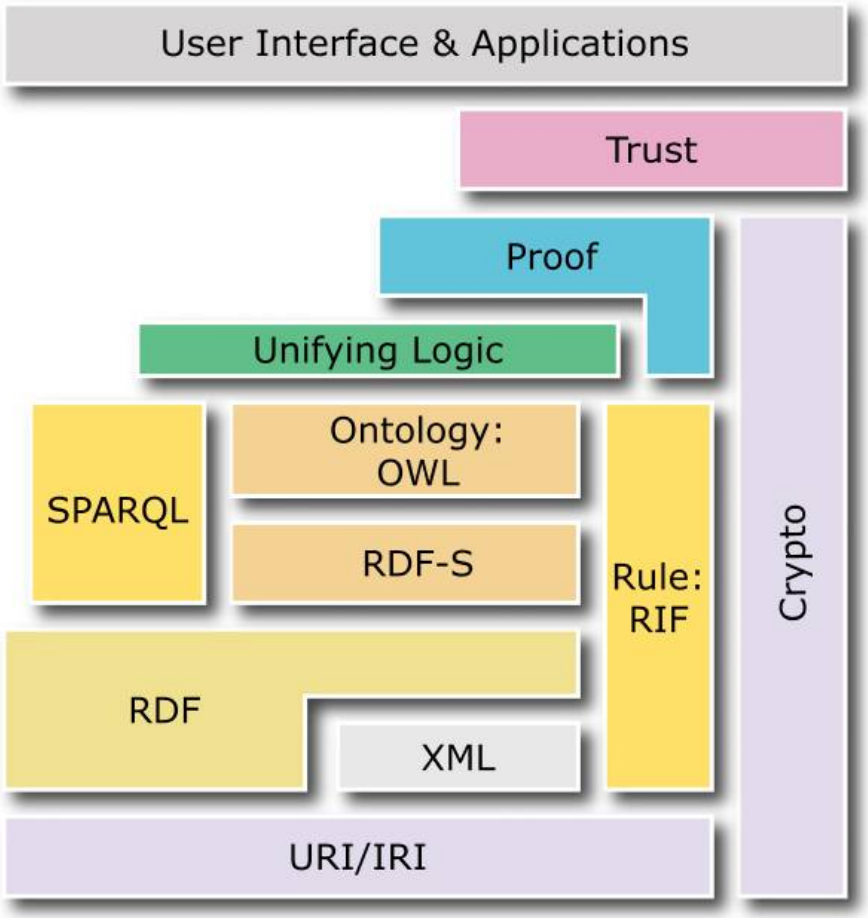


Cx Data Architecture integrates existing related DA work, fills in the gaps, provides overall DA guidance, generates common models, and develops common infrastructure

CxDA Metamodels are built using OWL – think of this as XML++



- OWL = Web Ontology Language
  - A language for describing the "what" of a domain of interest
  - Classes of things, properties of things, relationships between things
  - A standard defined by the World-Wide Web Consortium (W3C)
- How does it relate to XML?
  - OWL can be serialized using XML
  - OWL is built on the Resource Description Framework (RDF)
  - OWL constructs allow us to say things that XML Schema does not accomplish very flexibly



The Need for Different Kinds of Models

If you are doing these activities	*You need these Models*
Risk Analysis	Fault Trees FMEA Models PRA Models
Design	Architecture Models UML Models CAD Models
Decision Support Impact analysis Information Discovery Information Merging	Knowledge Models

The NASA NExIOM Ontologies connect Enterprise Architecture to Systems Engineering Models

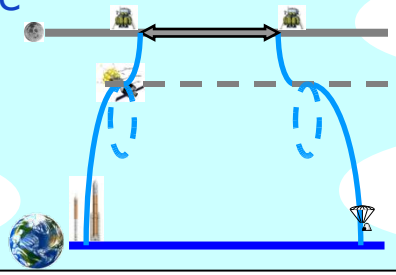
Organization has Policies
Organization has ProjectOffices
ProjectOffice has Goals
ProjectOffice uses System
Organization defines Mission
Mission has Phases

Process has Activities
Activity produces Workproduct
Workproduct isApprovedBy Role
Workproduct isProducedBy Tool
Tool analyzes MissionRisk
Tool calculates FigureOfMerit
FigureOfMerit isSpecifiedIn Document
Document isHeldIn System

semantics
+ metadata
+ Other representations
+ Part structures
+ Cx Element

Enterprise Knowledge

Business Concepts

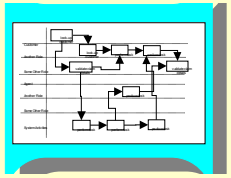


Mission

Organization

Process Knowledge

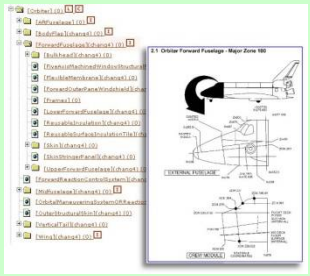
Process Concepts

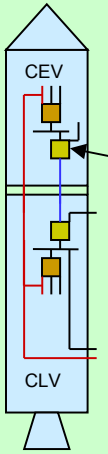


Lifecycle

Engineering Knowledge

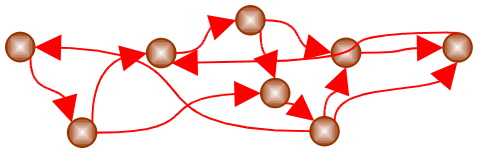
Engineering Concepts



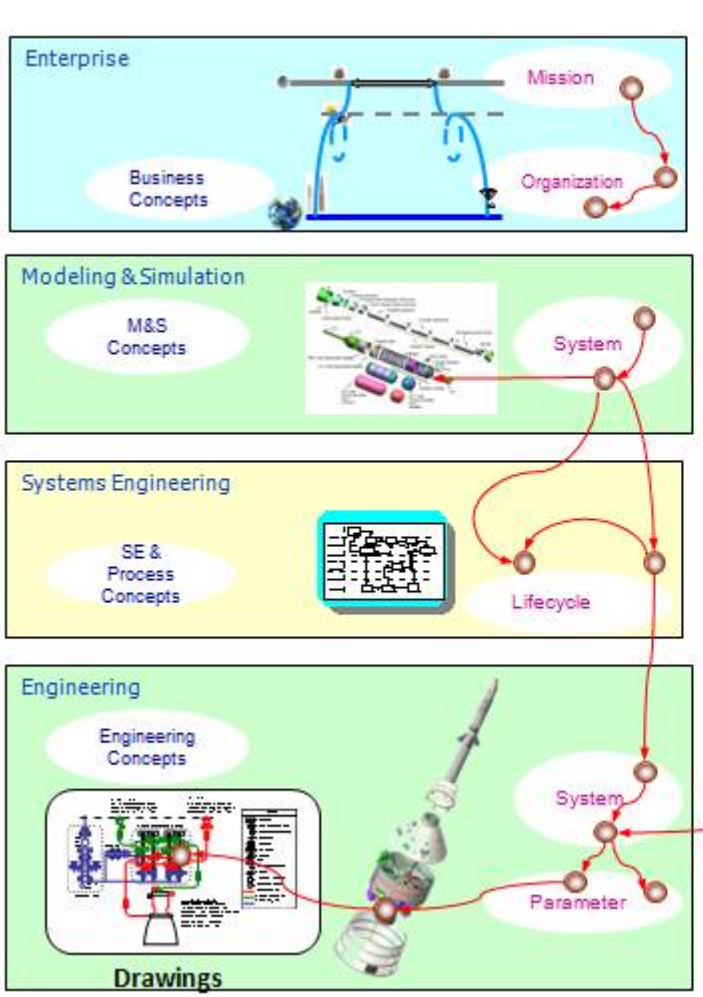


System

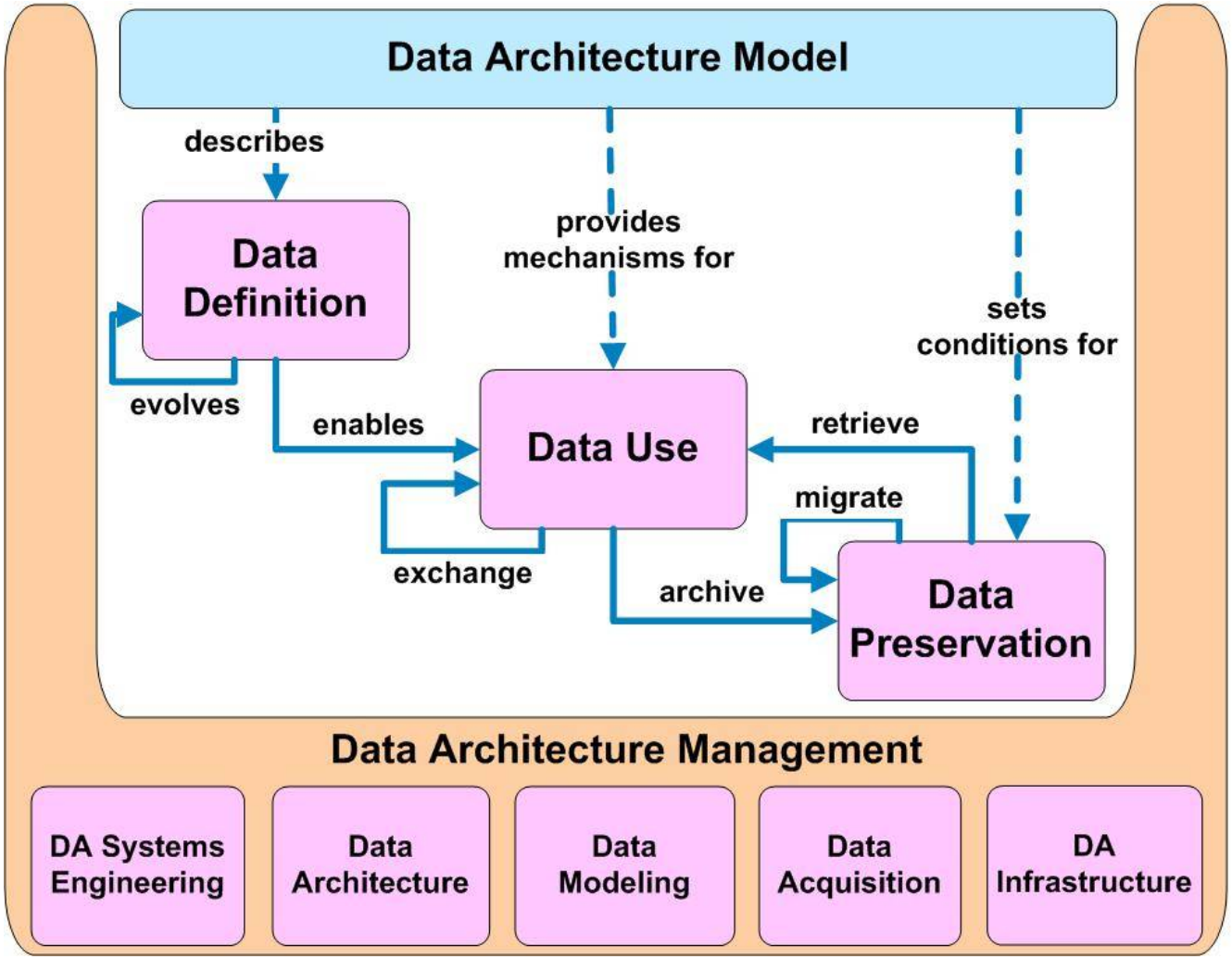
Parameter



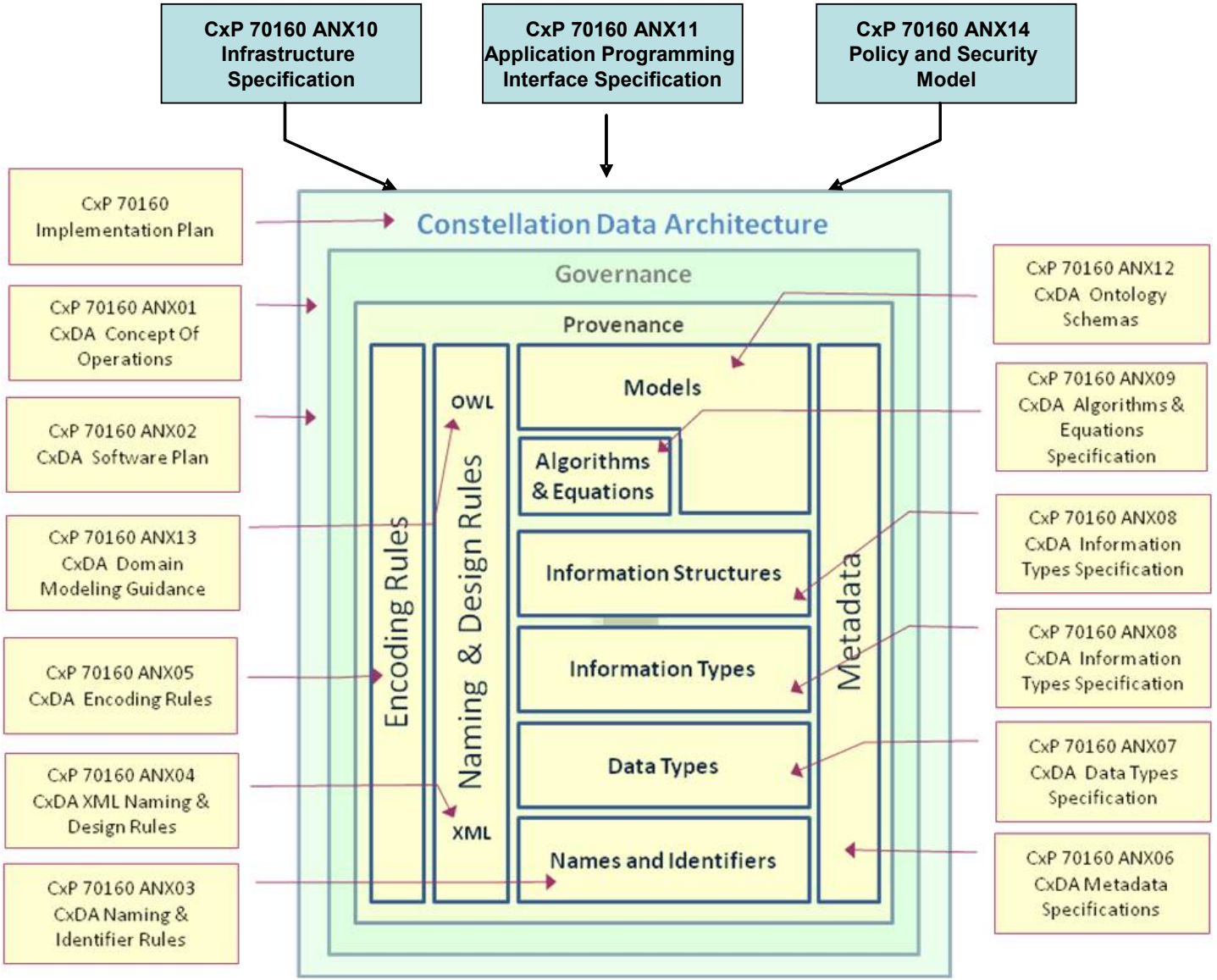
CxDA "connects the dots" across Information Objects



The Cx Data Architecture is a framework for the definition, usage, preservation and management of the constellation program data



NASA CxDA Framework



NExIOM Models Allow Traceability from Constellation Elements to Component, Parameters, Values and Units

- Hierarchical Identifier
  - Non-Arbitrary (from Names)
  - {Software+Human}-Oriented
  - *Which Thing + Navigation*

Constellation System : CEV+CLV

of *Crew Launch Vehicle*

CEV

of *First Stage*

Sub-System : Propulsion

of *Propulsion System*

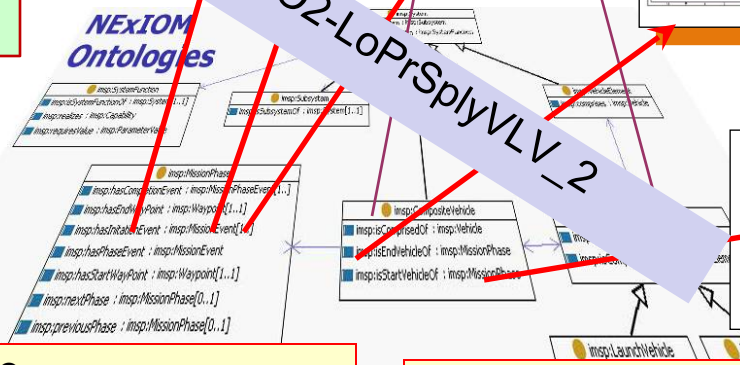
CLV

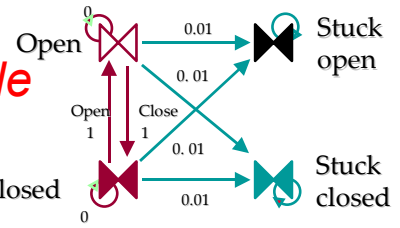
Resource : Fuel System

of *Fuel Subsystem*

CLV_1/FS_0/MPS_0/LO2-FuSYS_0/LO2-LoPrSplyVLV_2

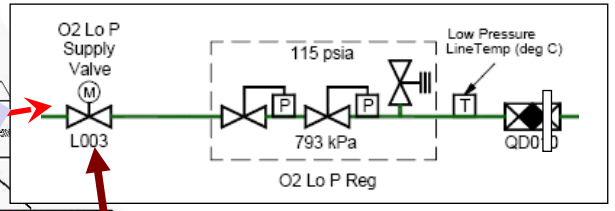
CxSID





Supply Valve

Device : O2 Lo P Supply Valve

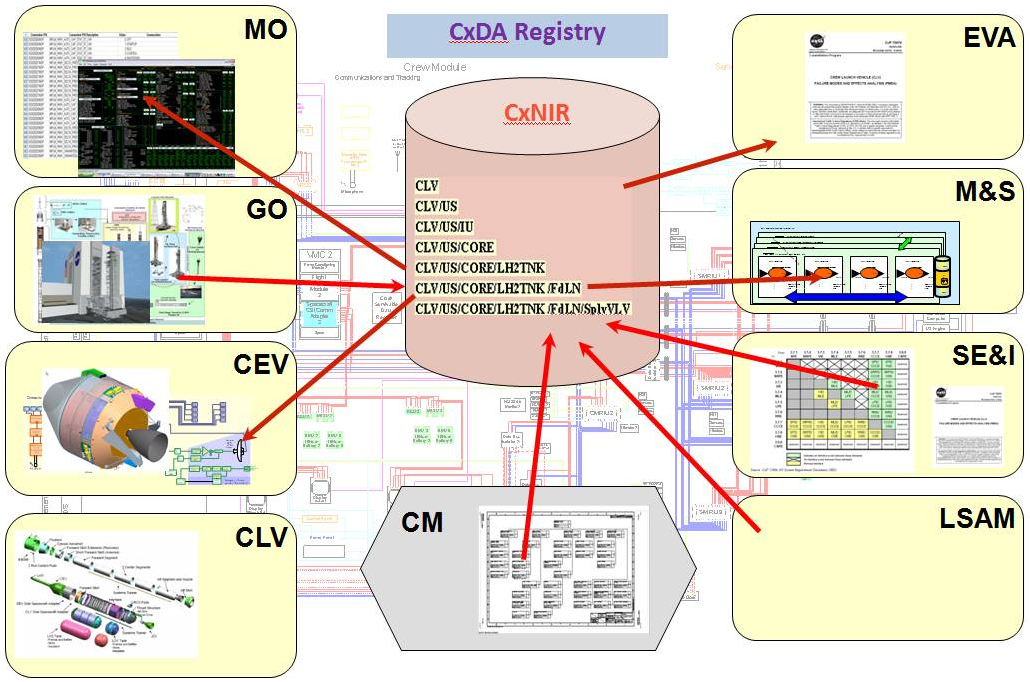


- Cx Name
  - Descriptive
  - Human-Oriented
  - *Which "Occurrence"*

- Industry use of Identifiers as designators
  - Arbitrary (e.g., P/N 0612375-766-9845)
  - Logic-Oriented (e.g., Software)
  - *Which "Instance"*

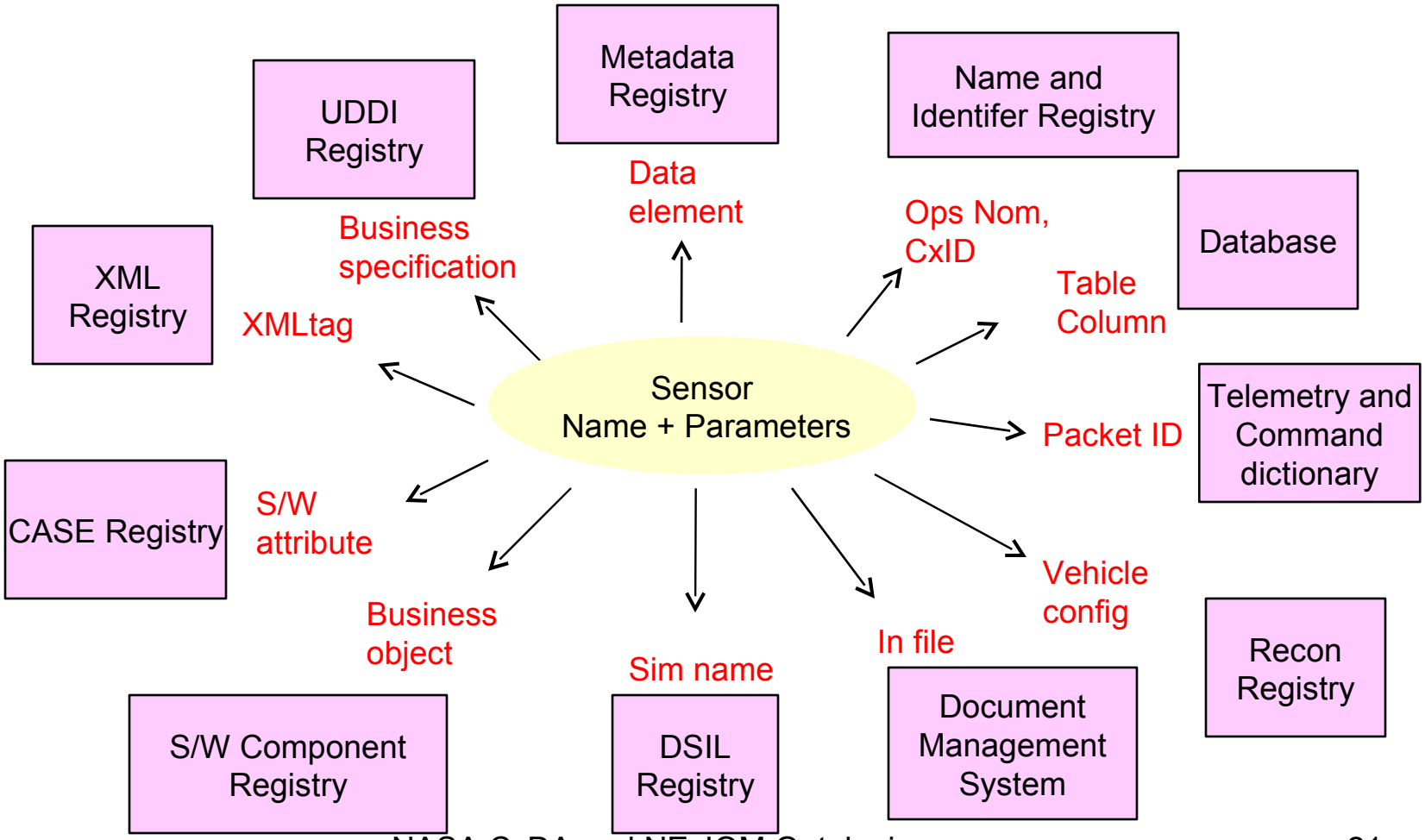
CxDA System of Registries (SoR)

- Provide consistent definitions of data
  - across time, between organizations, between processes.
- Connect "silos" of information
  - captured within applications or proprietary file formats, through the use of standardized data definitions
- Support the exchange of information
  - Using formats and protocols - XML and Web Services



The Roles of the CxDA SoR - 1

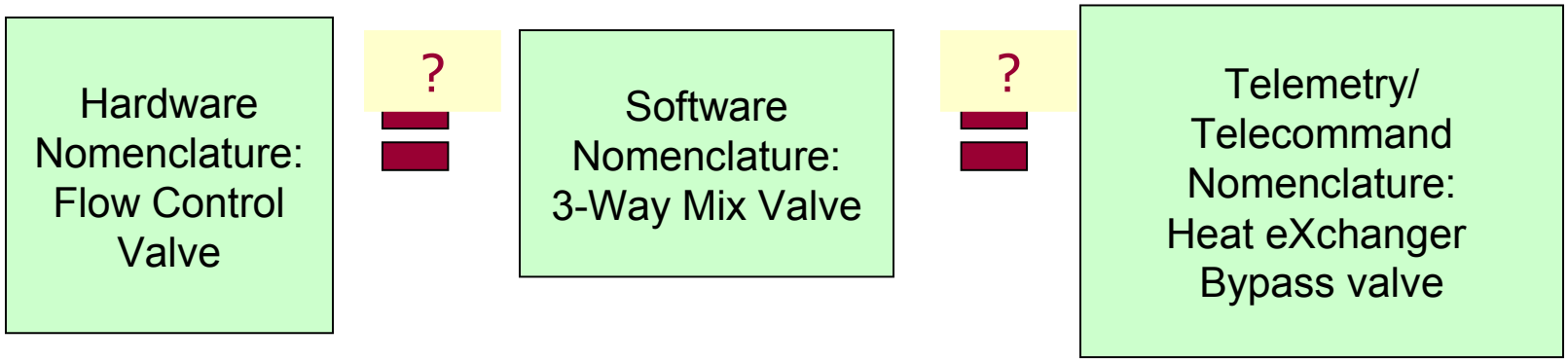
- A source of authoritative information
  - Nomenclature, names, identifiers, schemas, types, terms, protocols, definitions, etc. Instead of reading documents, expose granular details/specs of a file via model/schema -> allows querying by software and people



The Roles of the CxDA SoR - 2

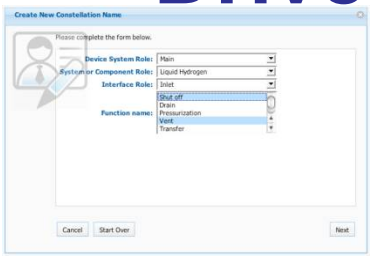
- A facility to relate information in multiple systems

Are these the same valves?



The Registries support answering these kinds of questions quickly, accurately, and provide a method of identifying/accessing the originating information

The CxDA Namer uses an Ontology-Driven Grammar Engine



Name Manager

Dialog Controller

Query Manager

CxDA Information Models

Query Engine

Inference Engines

Ontologies

Rules

Data Brokers

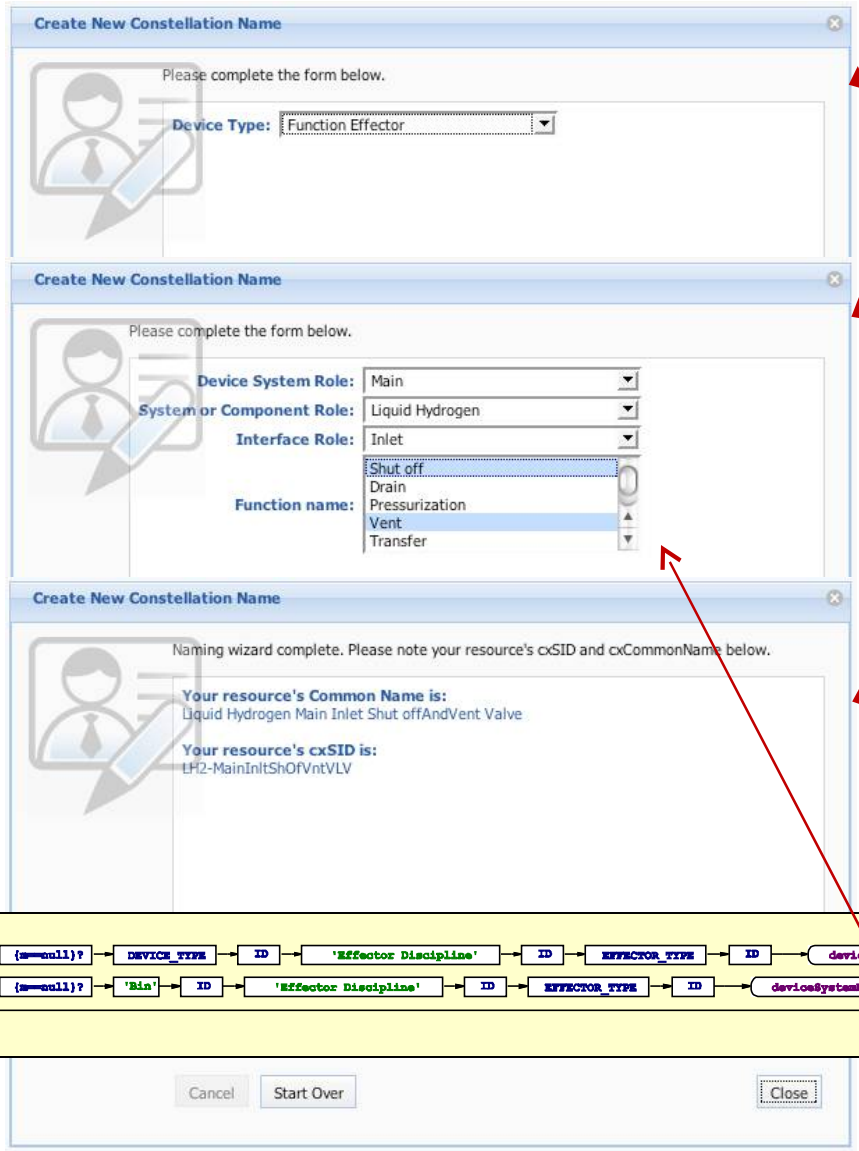
ANTLR Grammar

Lexer

Parser

CxDA Data Layer

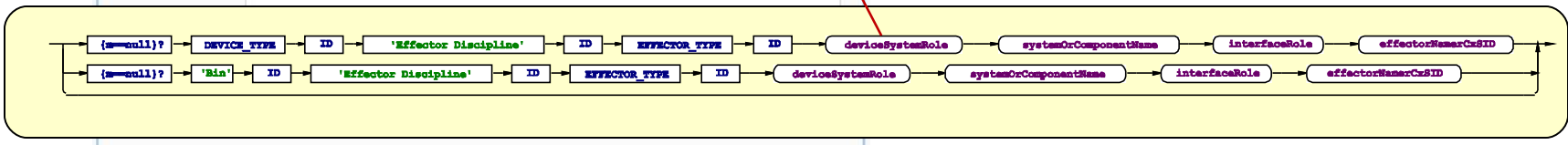
The Registry Name and Identifier Generator



I want to name a Hydraulics Valve

Ontology and Grammar control the Dialog with the user

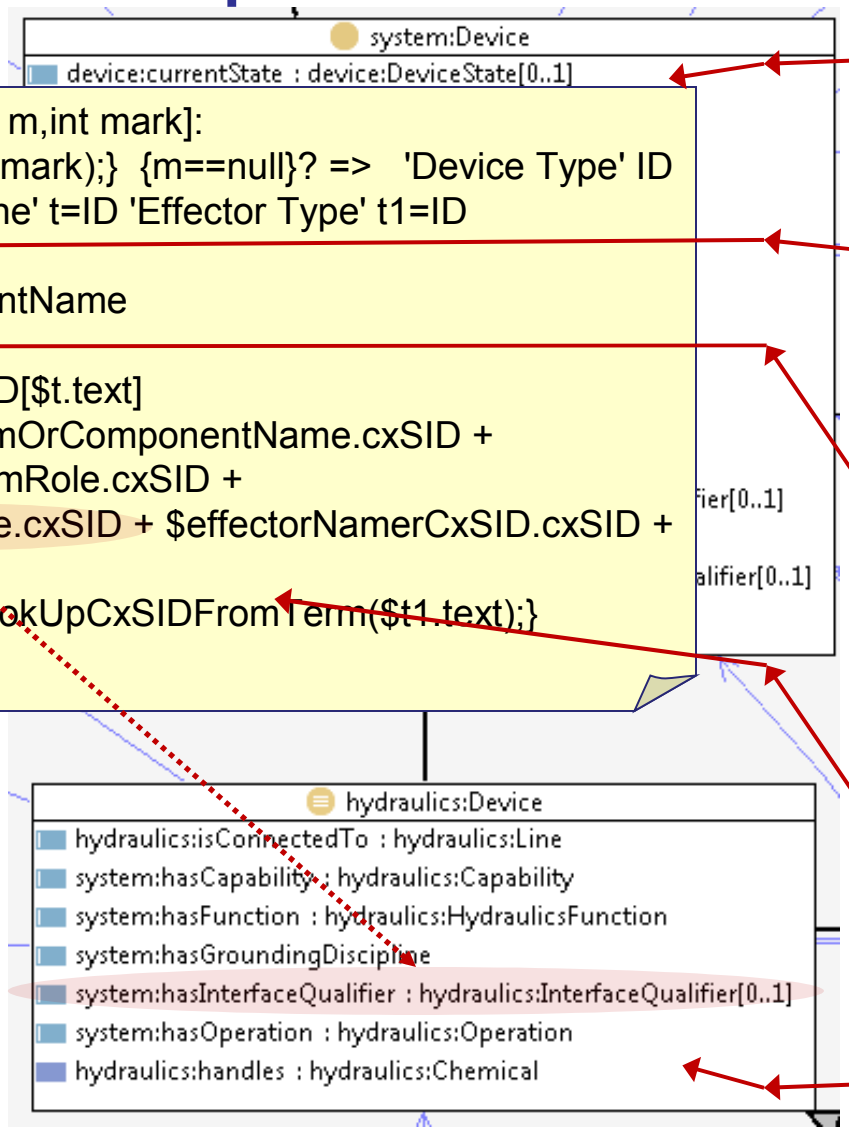
Constellation Names and Identifiers are generated



Namer uses Attribute Grammar with Java code making SPARQL queries on the Ontologies

```
computeCxSID[String m,int mark]:
    {System.out.println(mark);}  {m==null}? =>   'Device Type' ID
        'Effector Discipline' t=ID 'Effector Type' t1=ID
    deviceSystemRole
    systemOrComponentName
    interfaceRole
    effectorNamerCxSID[$t.text]
        {cxSID = $systemOrComponentName.cxSID +
            $deviceSystemRole.cxSID +
            $interfaceRole.cxSID + $effectorNamerCxSID.cxSID +

DeviceDataProvider.lookUpCxSIDFromTerm($t1.text);}
    |{ .. }
    |{input.consume();};
```



SPARQL queries

Grammar Rule

Device/System Role can be ‘main’, ‘auxiliary’, ‘primary’ or ‘secondary’

Interface Role depends on type of device. For a Valve can be ‘Inbound’ or ‘Outbound’

Consult the ontologies for vocabulary of terms associated with devices

Ontology of Devices

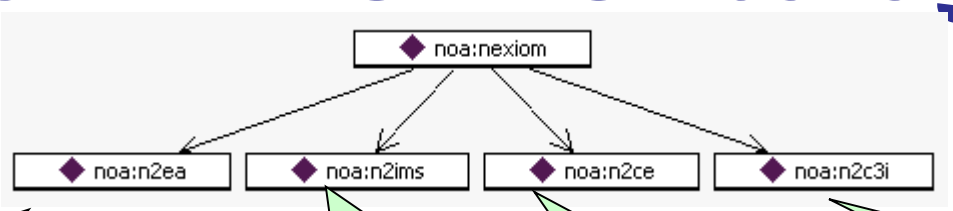
What is NExIOM

NExIOM, the *NASA Exploration Initiatives Ontology Models* formalize the way machines (and people) refer to NASA Elements, their Scientific and Engineering disciplines, related work activities, and their interrelationships in the Enterprise.

Through the use of agreed knowledge representations information become intelligible and actionable to machines, tools, and people. Information can be found, associated, aggregated and reasoned over to generate products and inform decisions within and across diverse organizational groups.

NExIOM consists of Models, a Semantic Infrastructure, and Services, integrated with operational tools and systems.

The NExIOM Ontologies

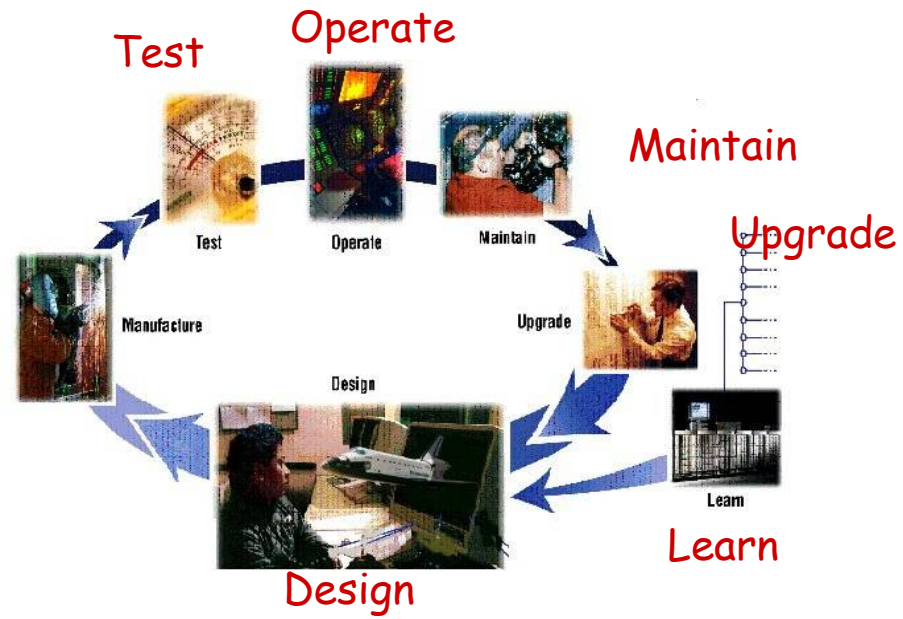


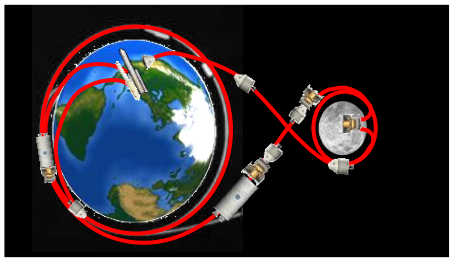
Level 2 Enterprise Architecture – databases, tools and systems used by organizations

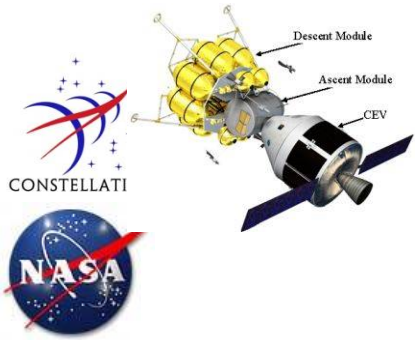
Level 2 Integrated Modeling and Simulation – lifecycle based

Level 2 Constellation Elements

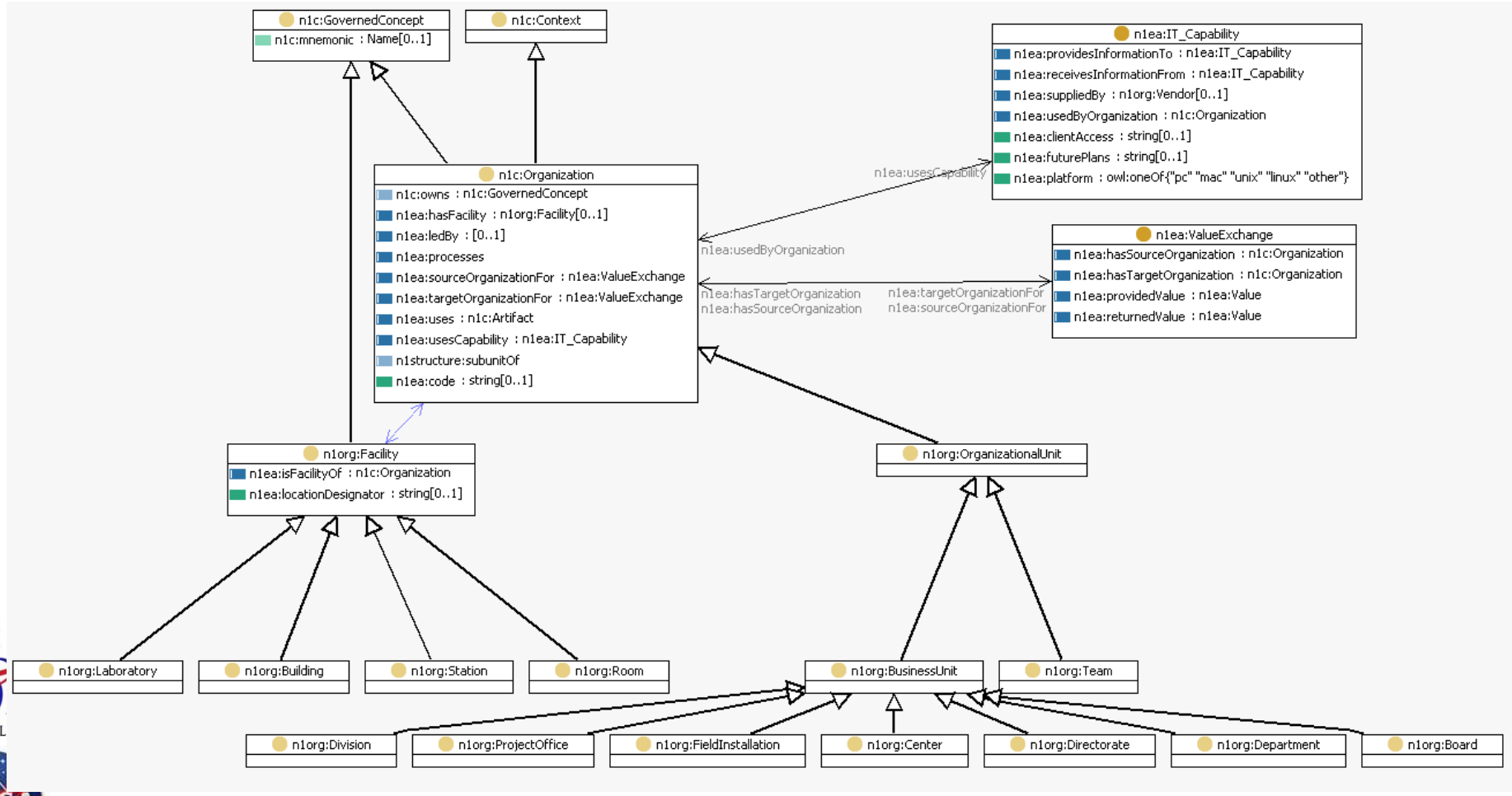
Level 2 C3I – Command, Control and Communications



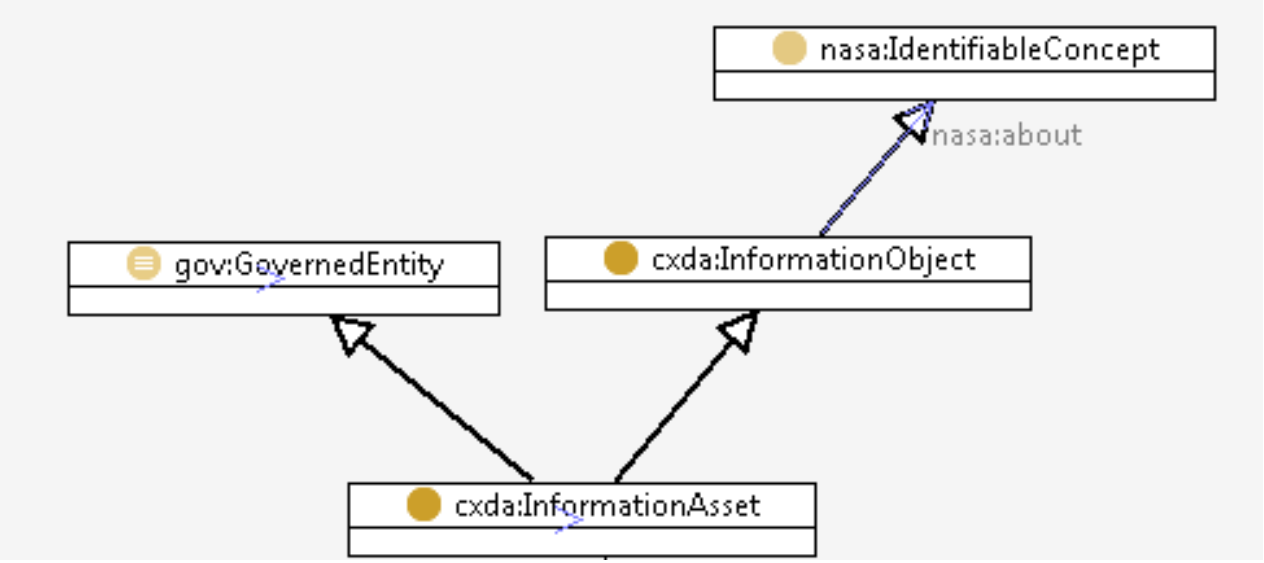




NASA Enterprise Architecture Ontology (extract)



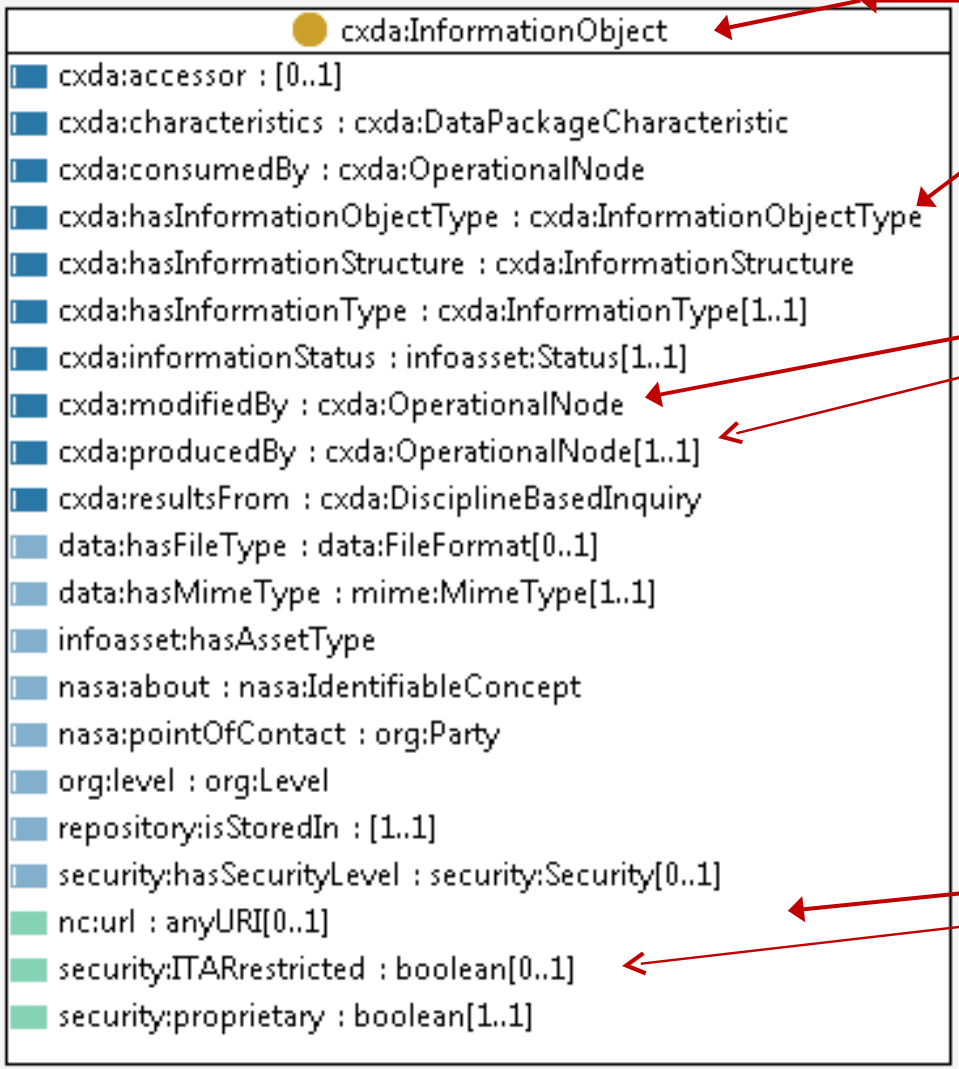
CxDA Information Assets



An 'Information Asset' is both a kind of 'Governed Entity' and an 'Information Object'.

An 'Information Object' is a kind of 'Identifiable Concept' – something that has names and identifiers

CxDA Information Object



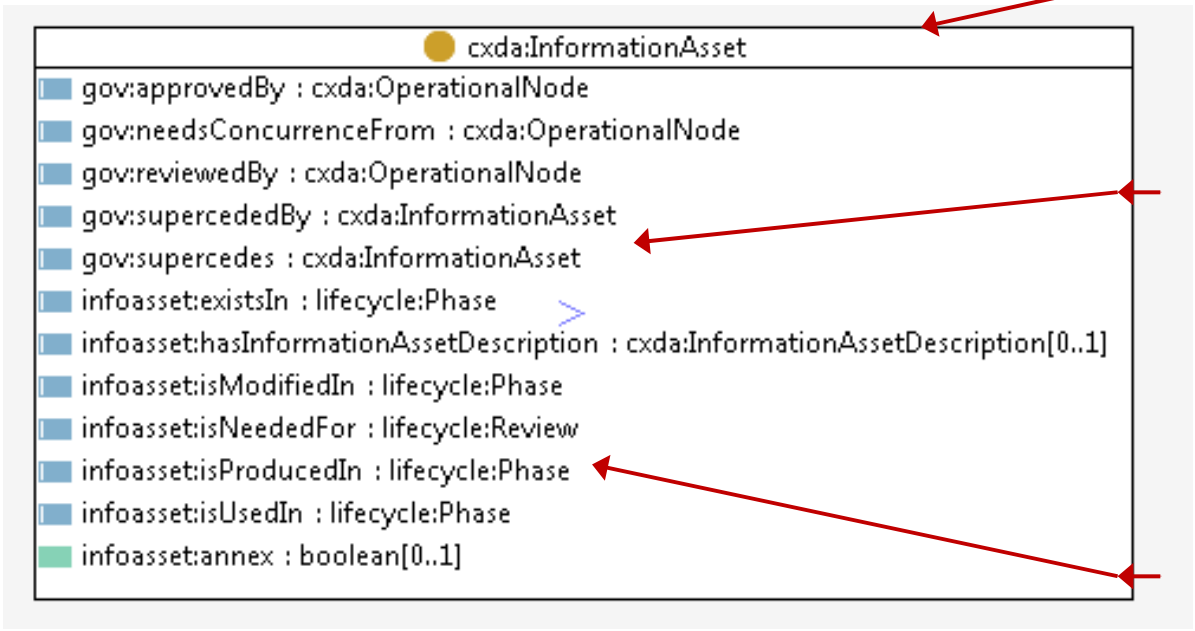
Information Object Class

Type Categories

Producers and Consumers are Operational Nodes

The CxDA Model has support for NASA's Security needs

CxDA Information Asset specializes CxDA Information Object with other Governance



Information Asset Class

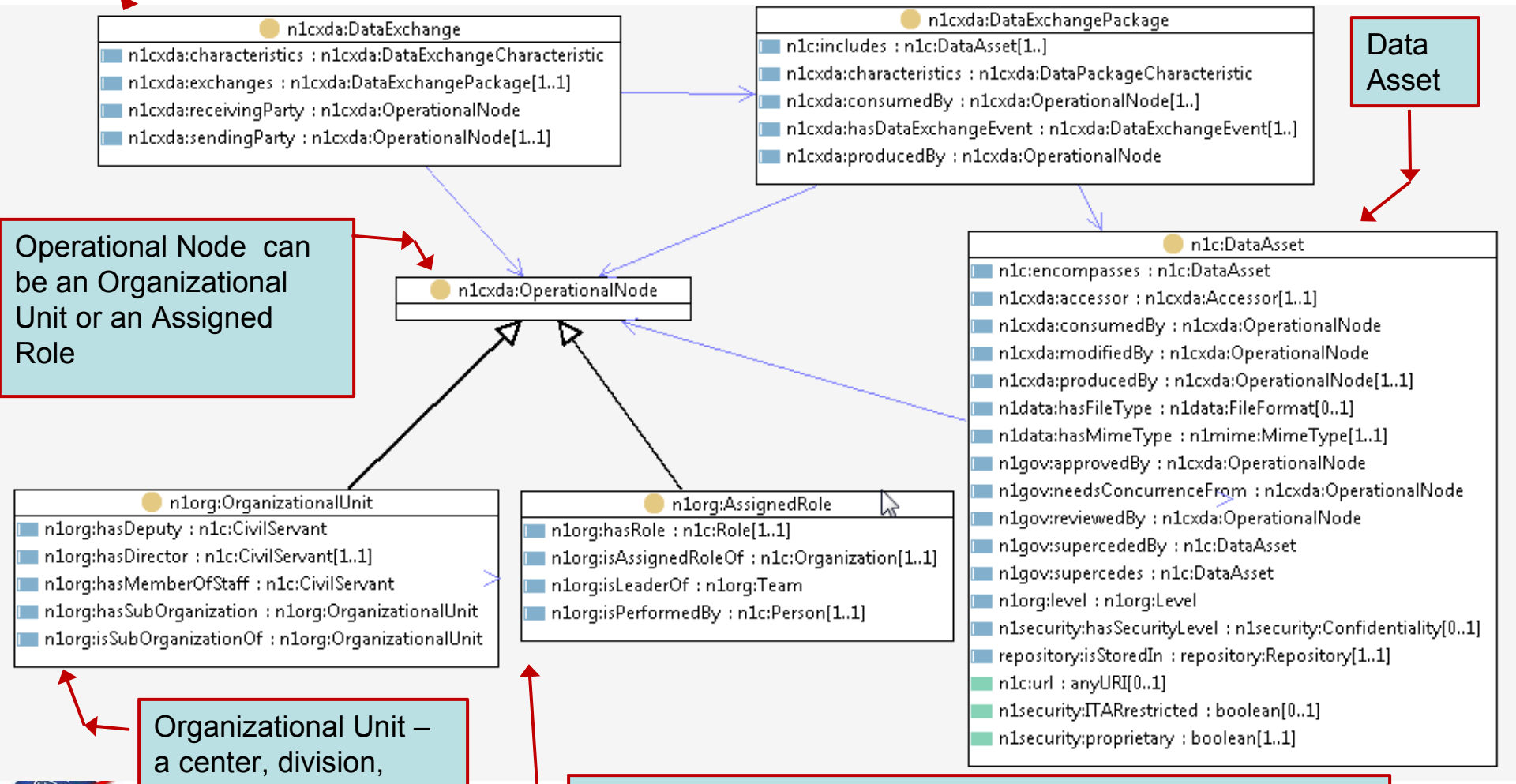
Provenance Metadata as associations to other inofrmation assets

Data Assets are also governed by NASA's processes for approvals, reviews and concurrence

Data Exchanges occur between operational nodes – Organizational Units and Assigned Roles

Data Exchange has characteristic properties and sending and receiving parties

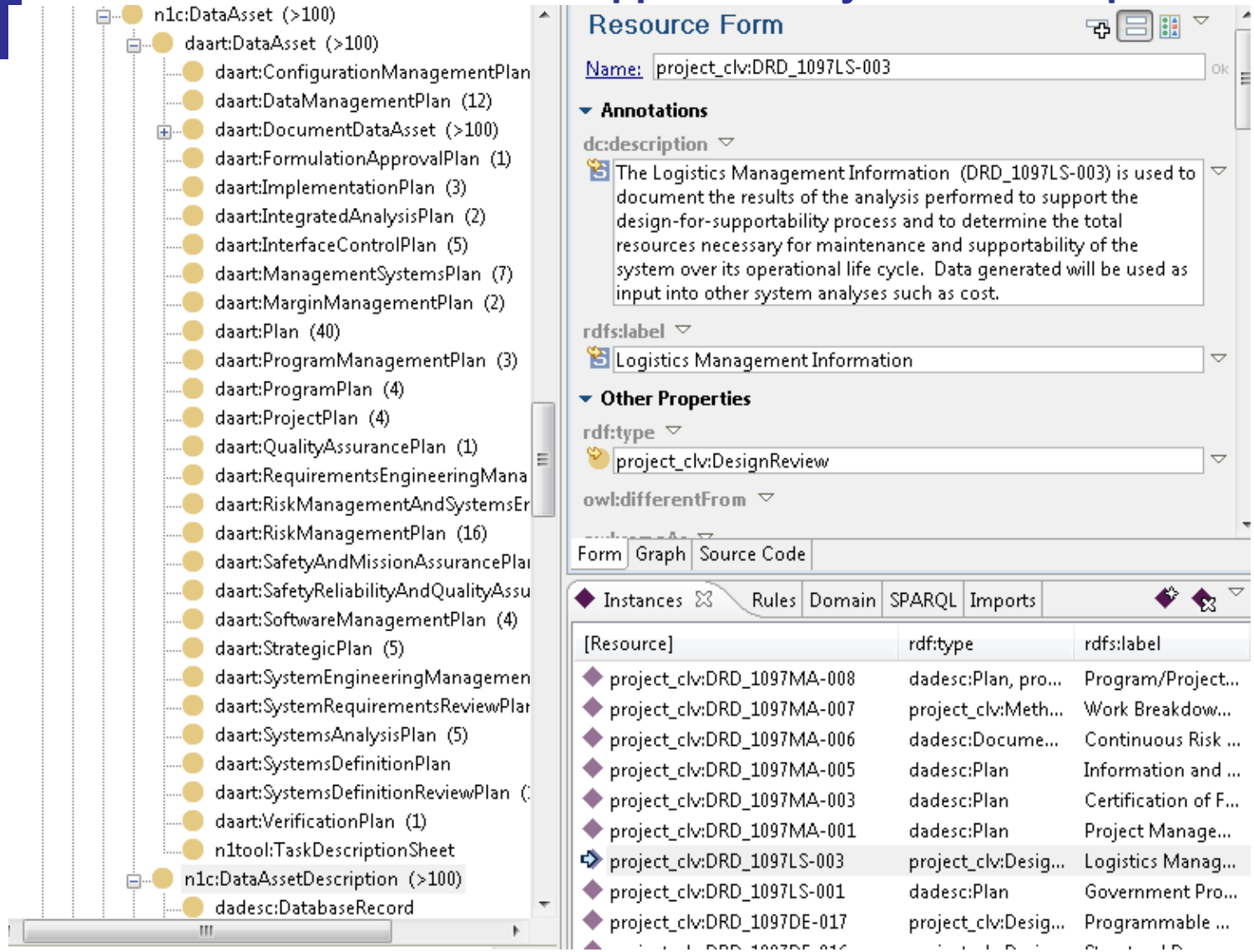
Data Exchange Package has producers and consumers



Organizational Unit – a center, division, office, etc.

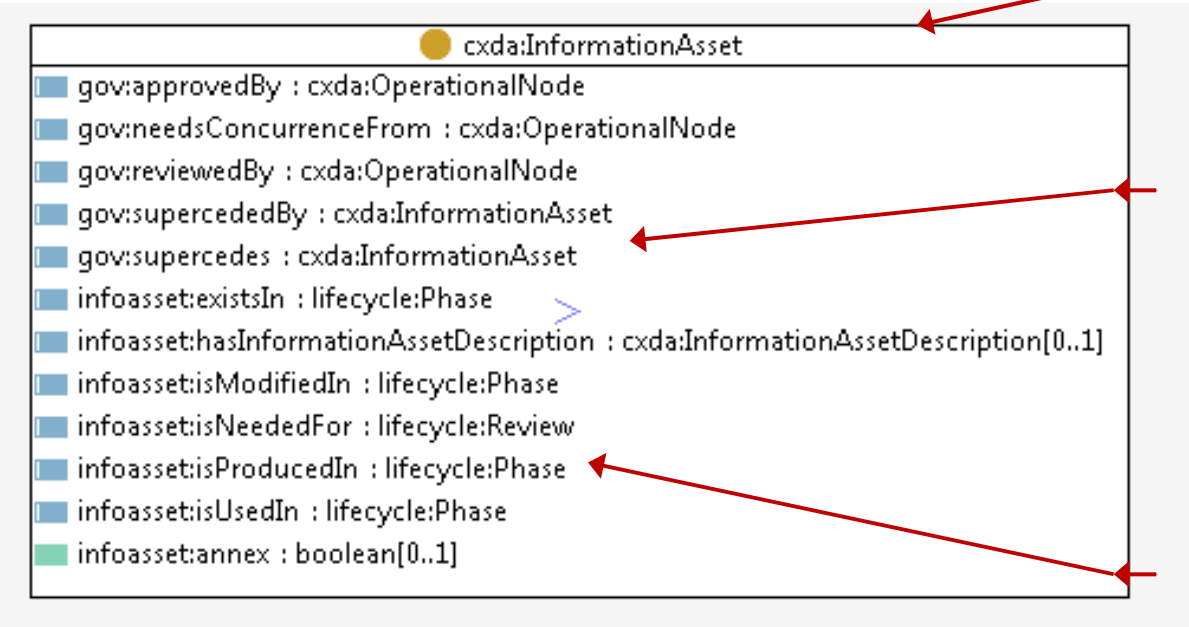
An Assigned Role is a person from an Organization performing a discipline (role)

Cx Data Assets are registered in the CxDA Registry Using a rich set of Data Models that support the key DoDAF viewpoints



A screenshot of the CxDA Data Capture and Modeling Environment

CxDA Information Asset specializes CxDA Information Object with other Governance

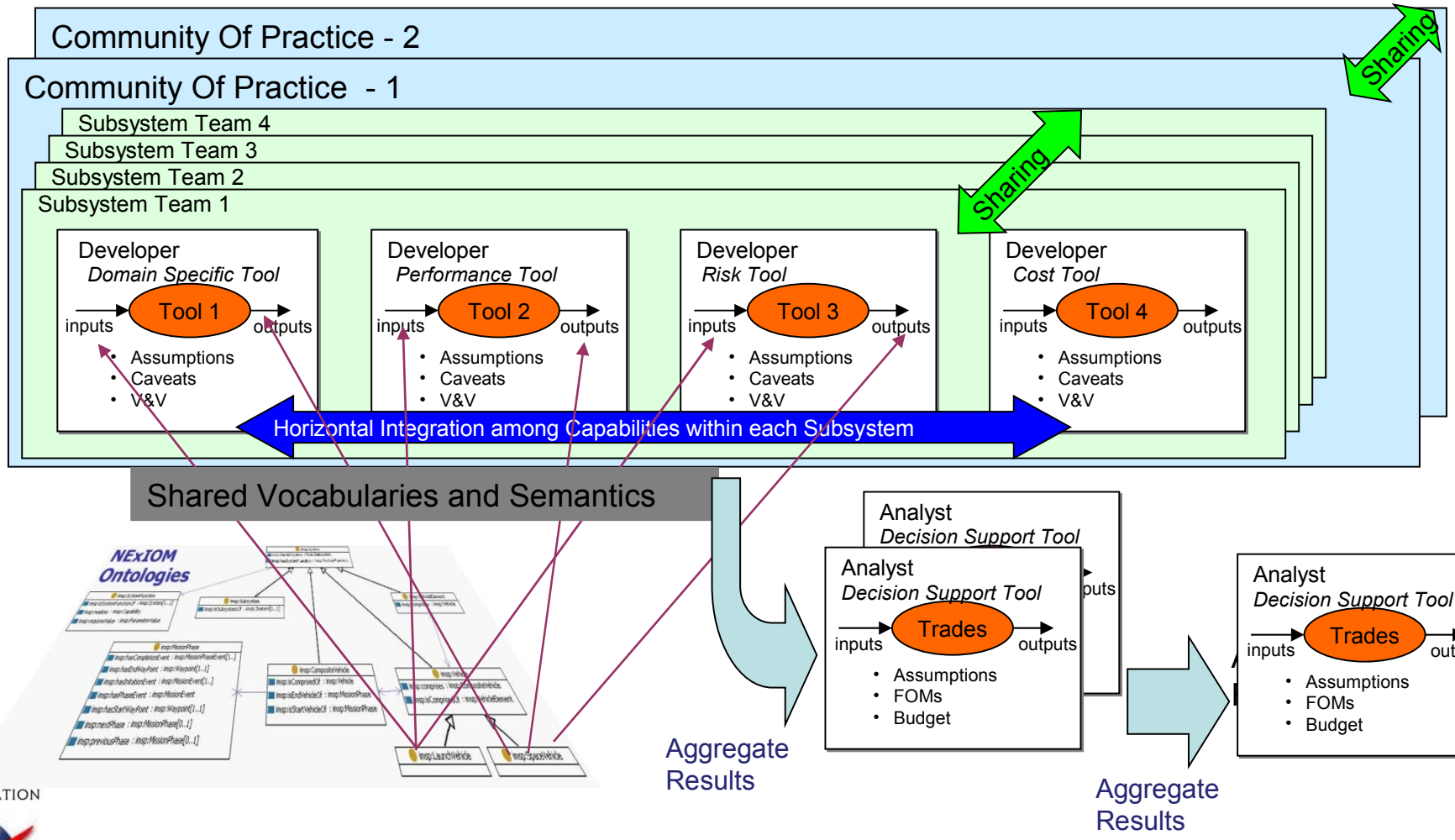


Information Asset Class

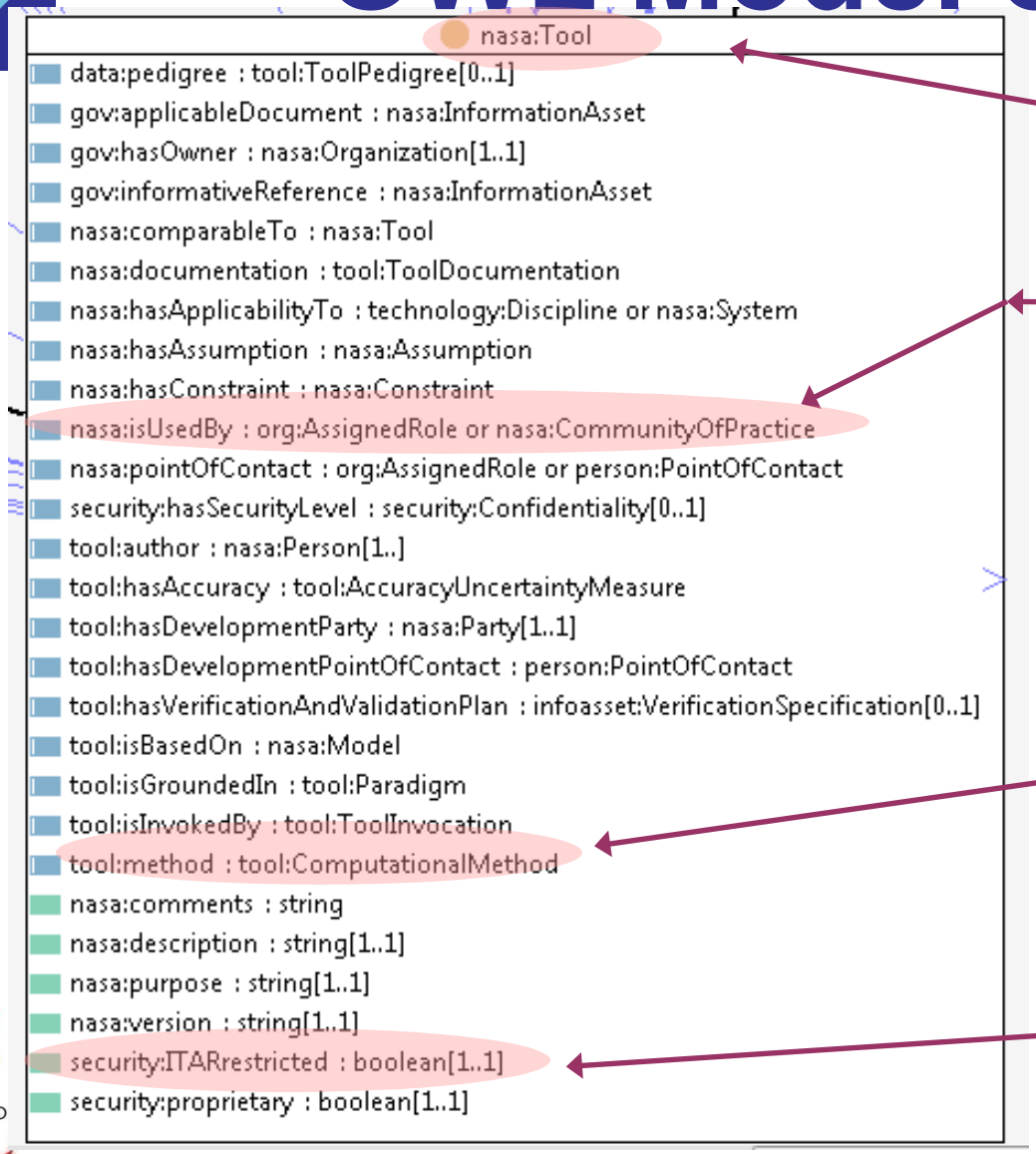
Provenance Metadata as associations to other information assets

Data Assets are also governed by NASA's processes for approvals, reviews and concurrence

NASA Modeling and Simulation Teams for different Constellation Elements use tools that need to interoperate



OWL Model of a Tool



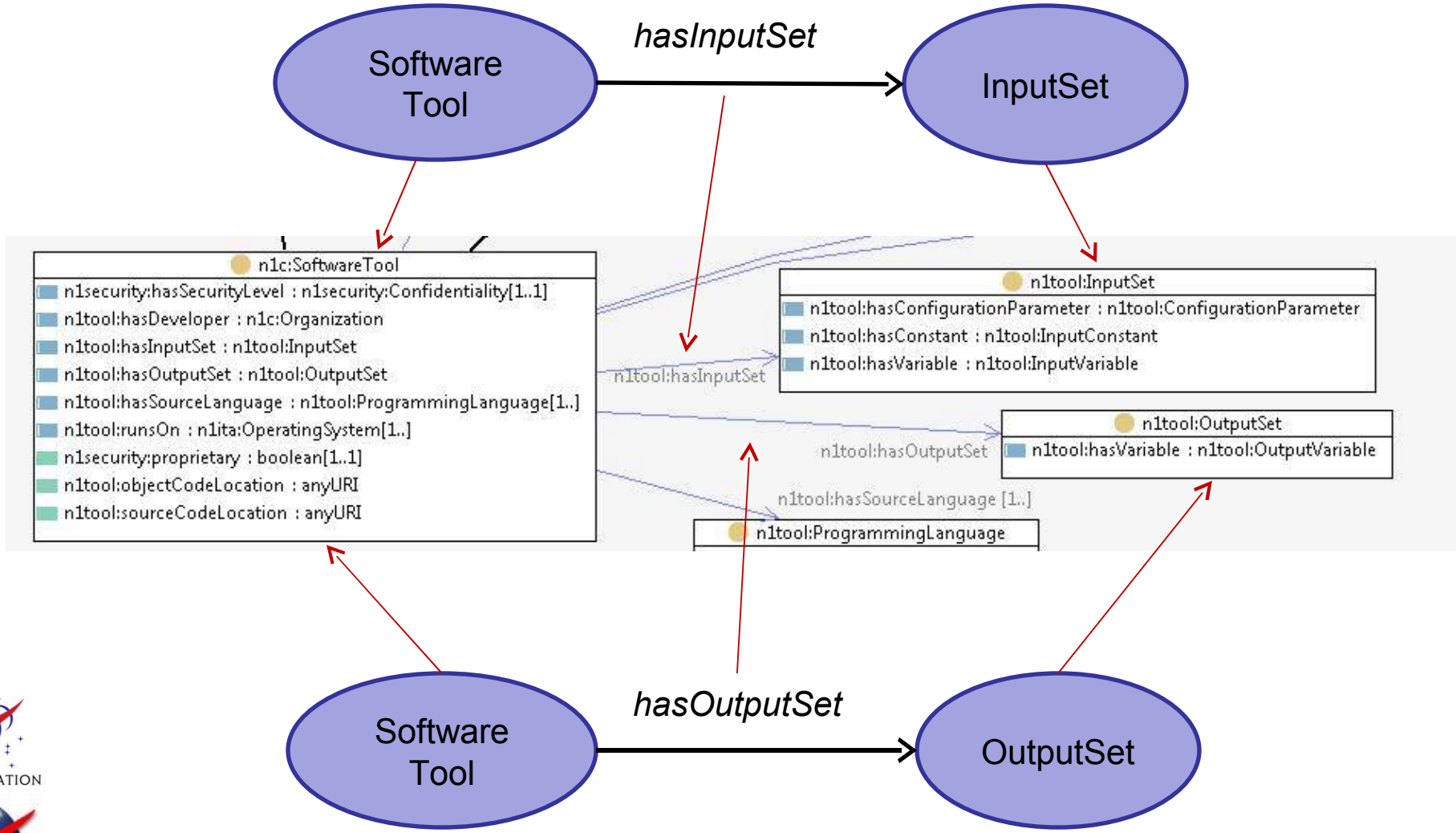
The class 'TOOL'

Object property 'isUsedBy' makes connection to 'AssignedRole' and/ or 'CommunityOfPractices'

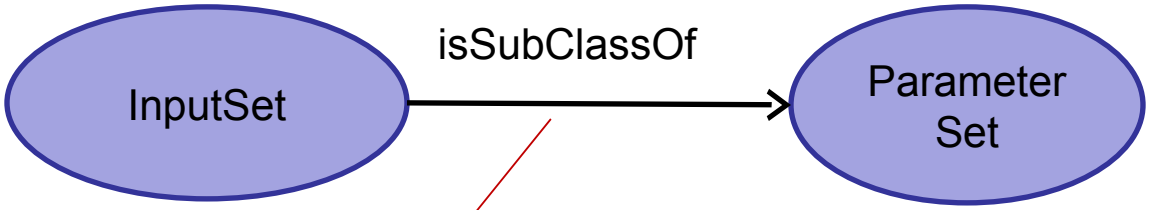
Object property for the tool's computational model

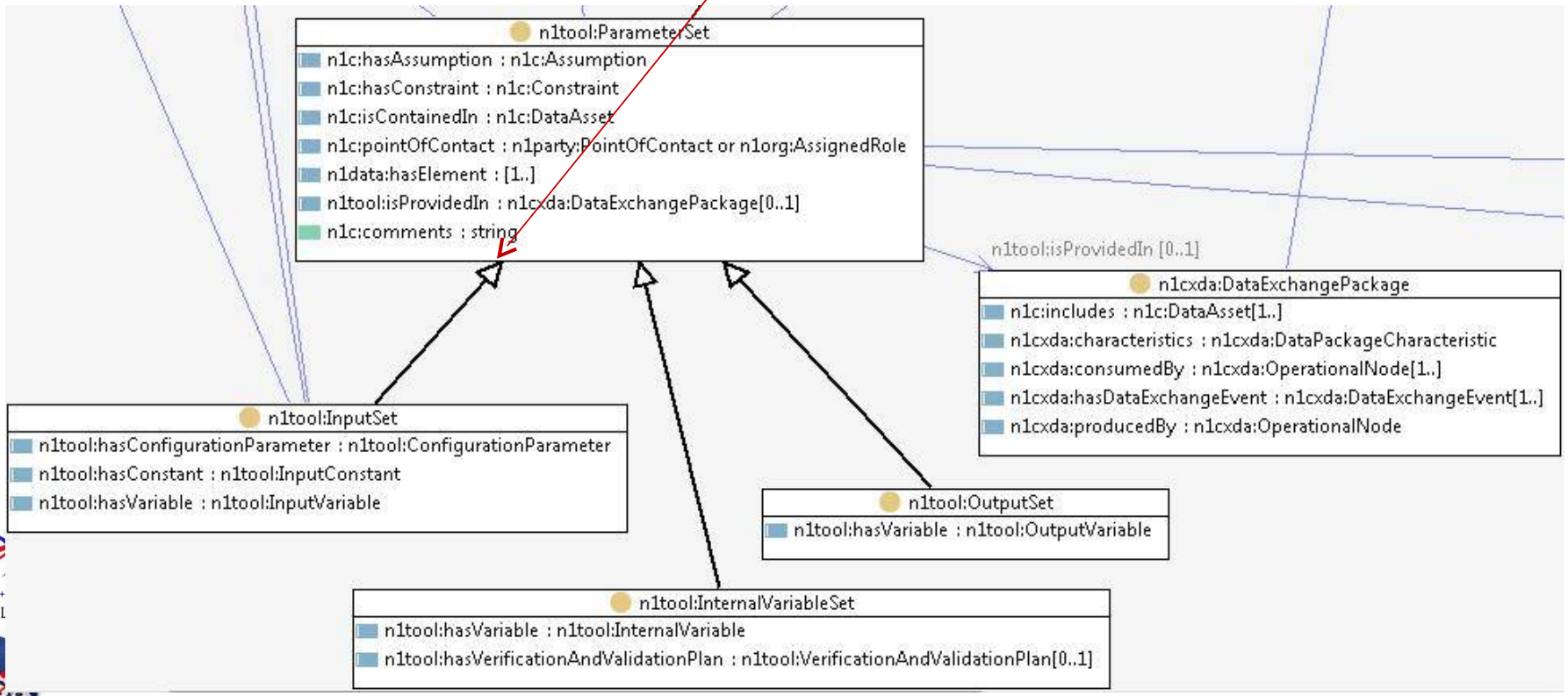
Data property attribute for 'ITAR Restriction'

A Software Tool has a Set of Inputs and a Set of Outputs

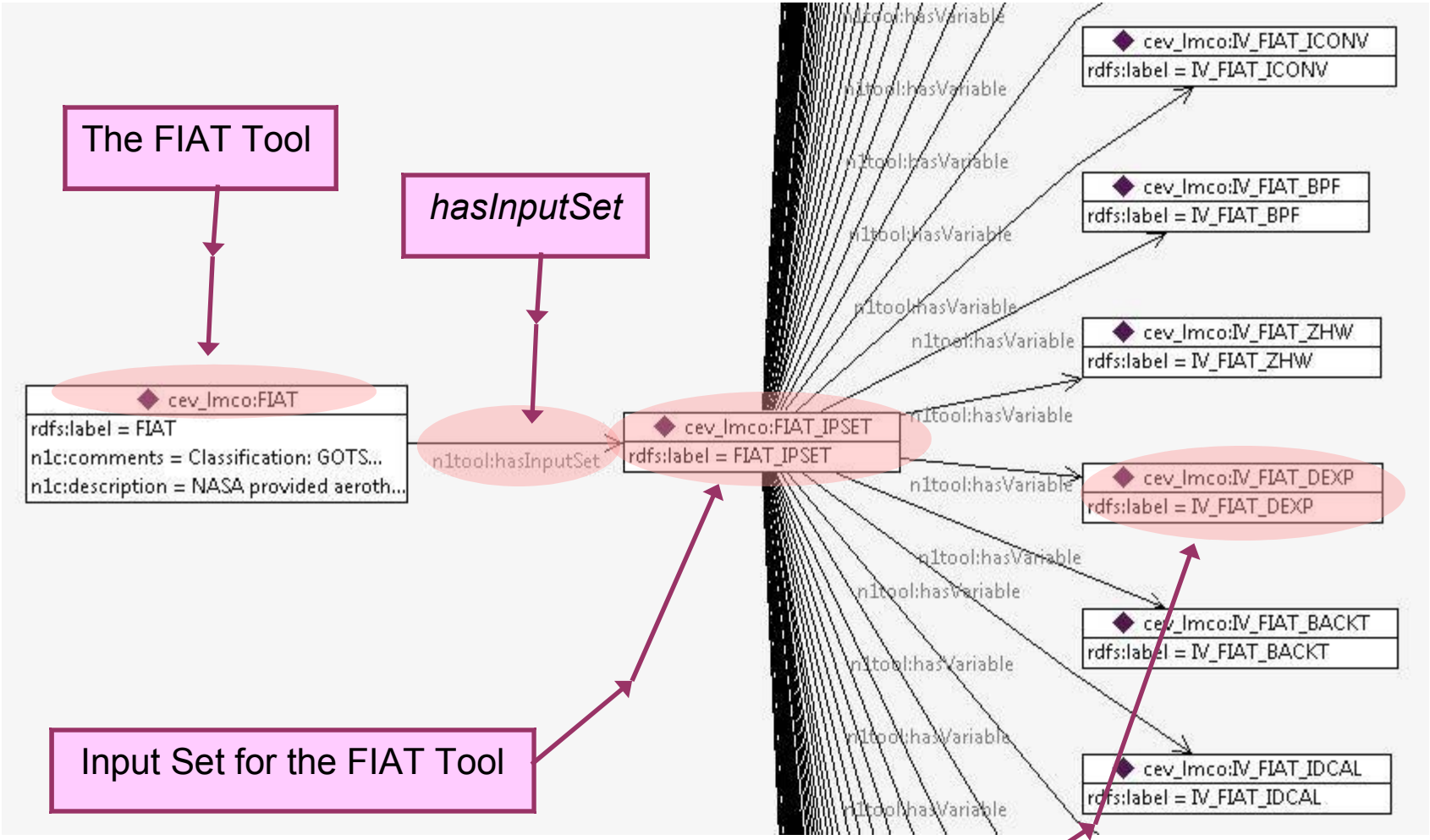


Input and Output Sets are 'ParameterSets' – a class that specifies common properties, e.g. assumptions and constraints



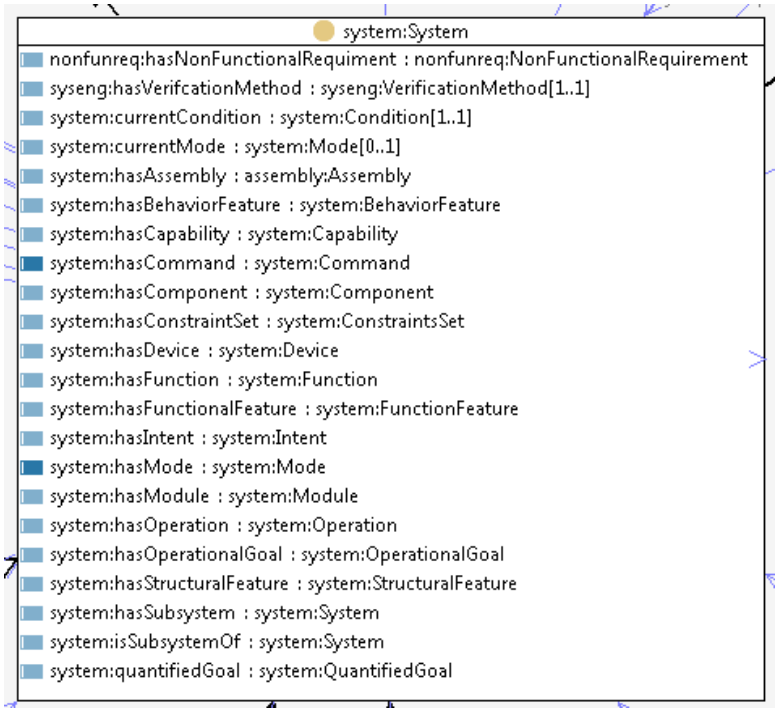


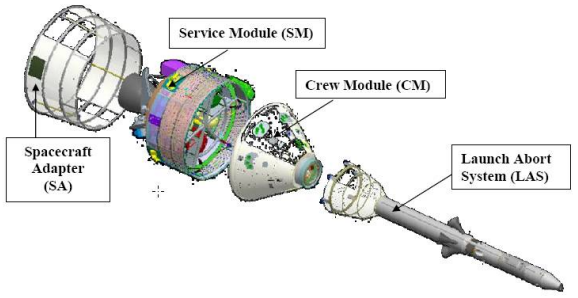
NExIOM CxDA Exchange Example: OWL Model showing explicit relationships

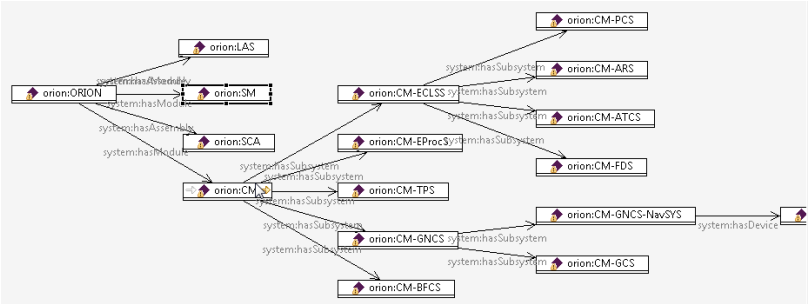


A System is modeled using an SBFI formalism

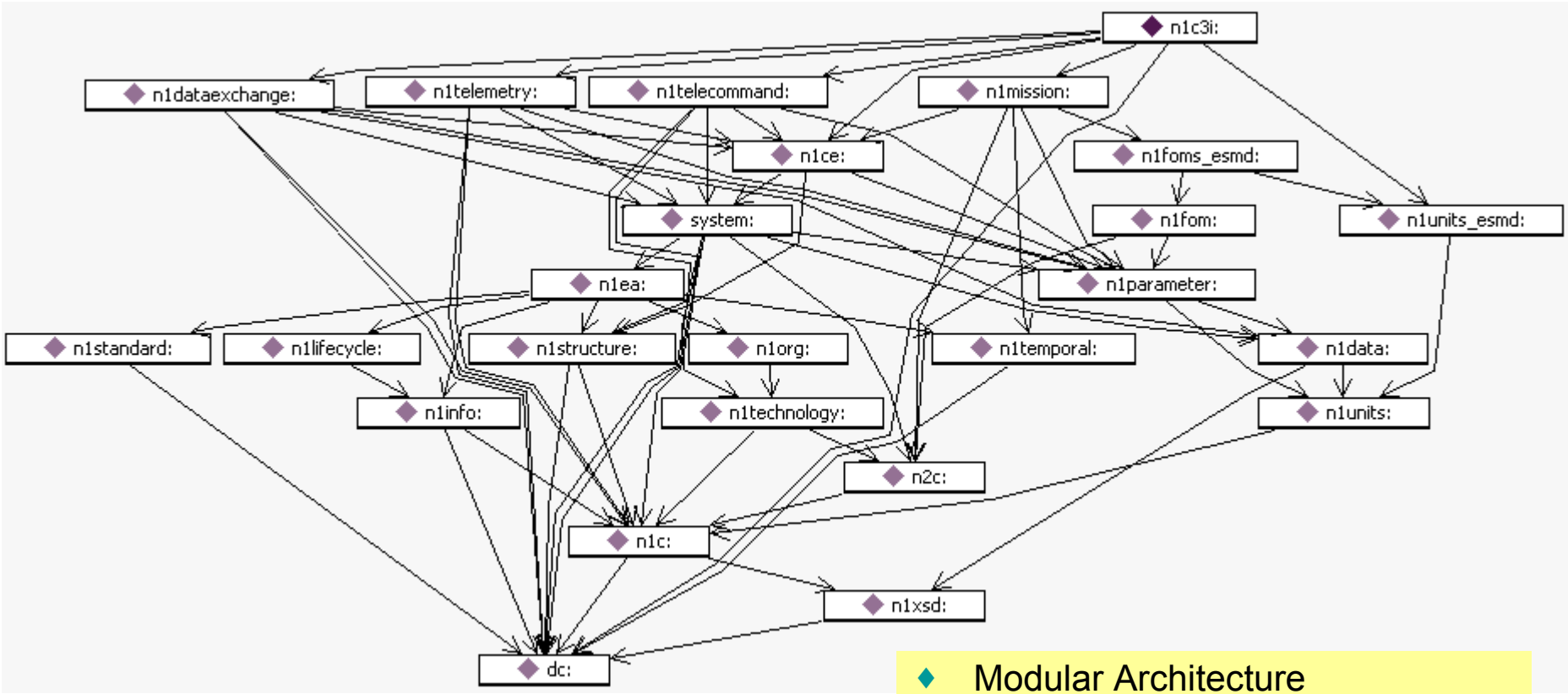
The NASA System Ontology extends SBF with other SE modeling constructs. Some SysML concepts are modeled directly others need to be “free-ed’ from their UML dependencies







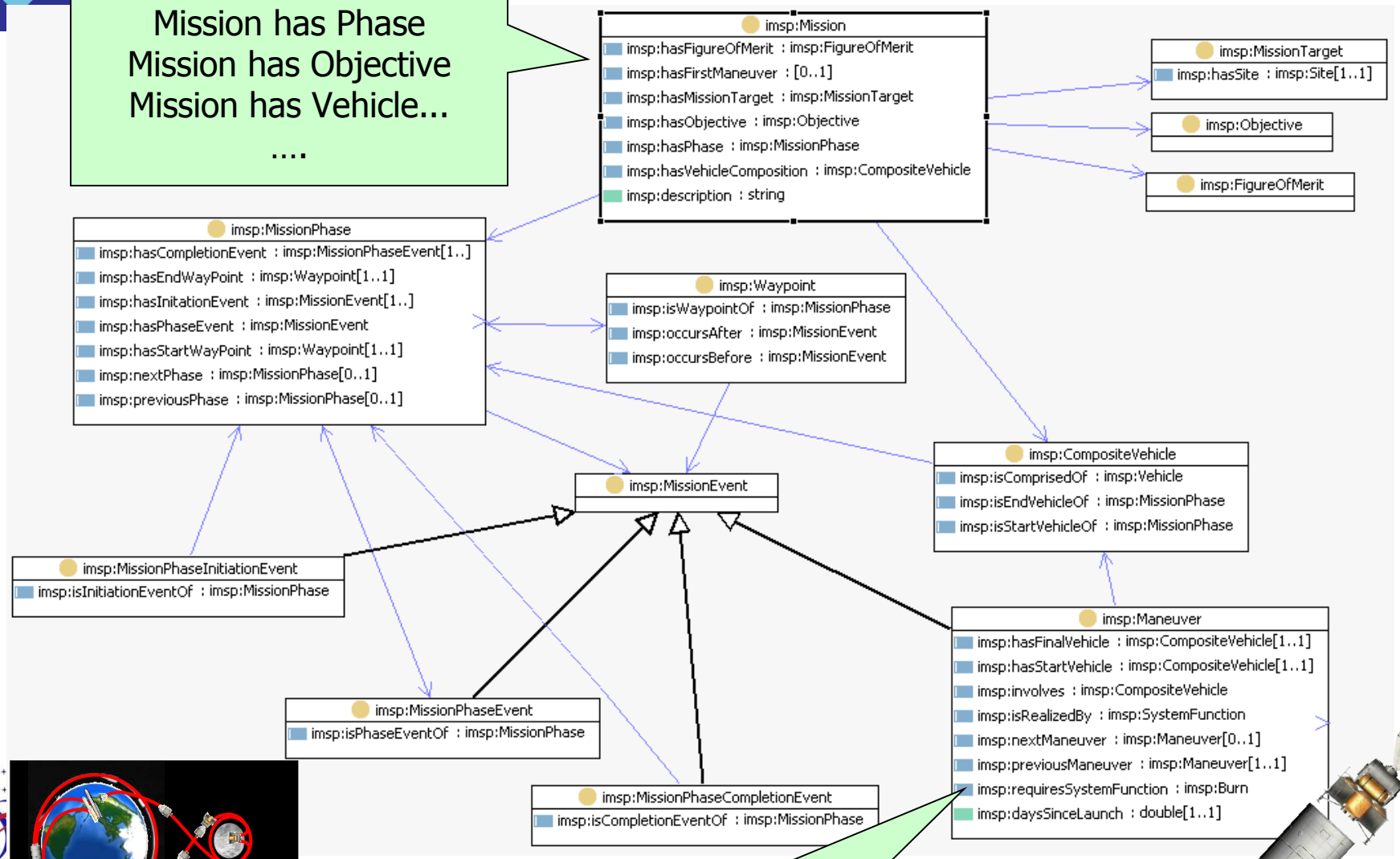
NExIOM Ontology Architecture for C3I



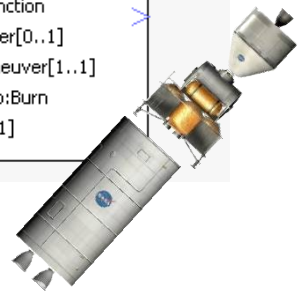
- Modular Architecture
- Reusable Models
- Different degrees of specificity
  - n1, n2, n3 … models

NExIOM models Missions and Maneuvers relating ConOps to Vehicle Elements, System Functions, Parameters and Units

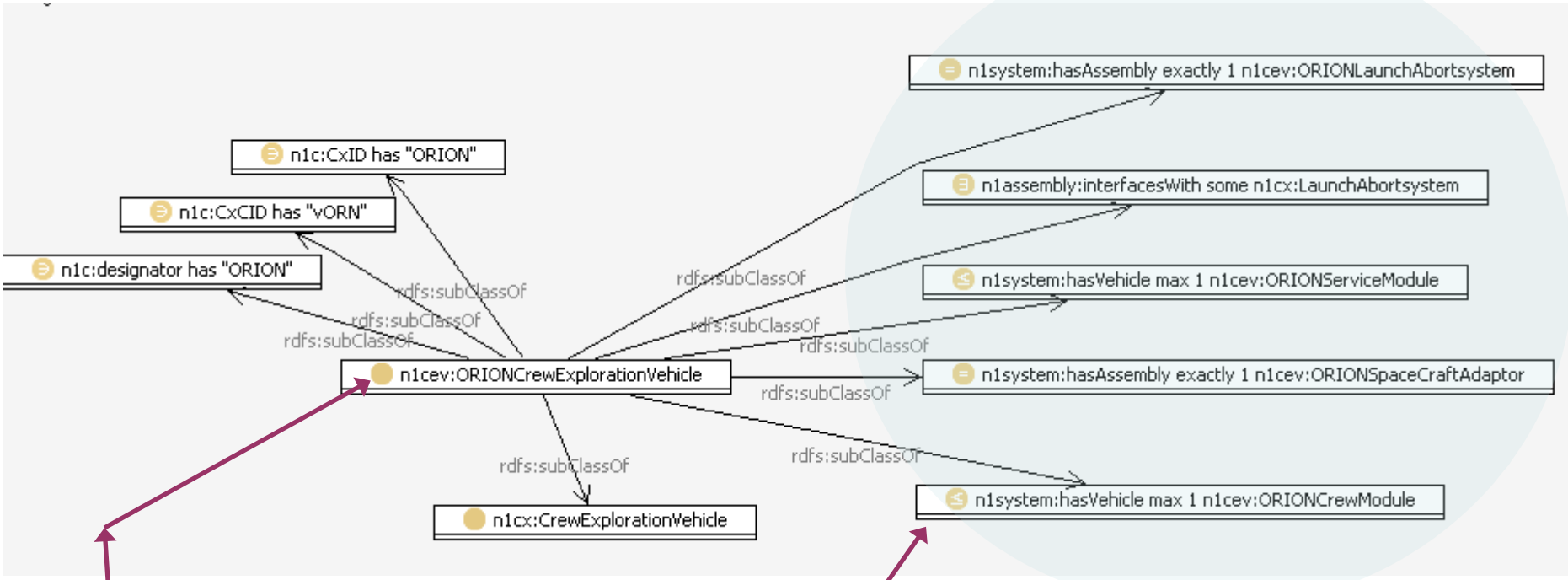
Mission has Phase
Mission has Objective
Mission has Vehicle...
....



Maneuver requires Burn

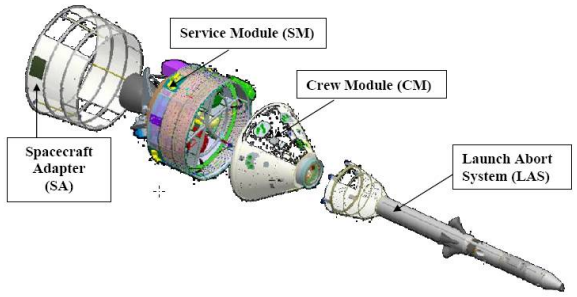


Part of the ORION Model

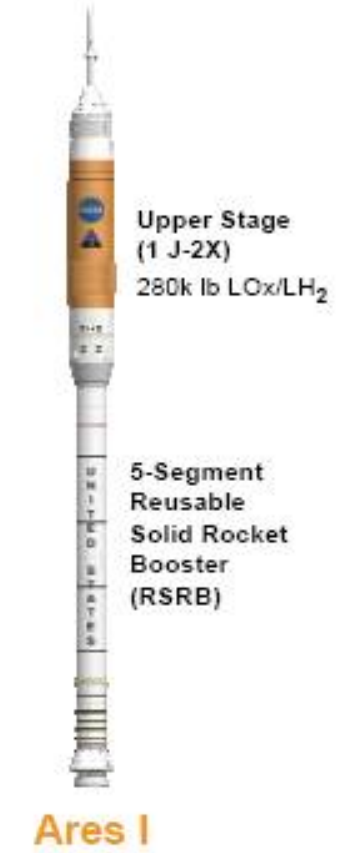


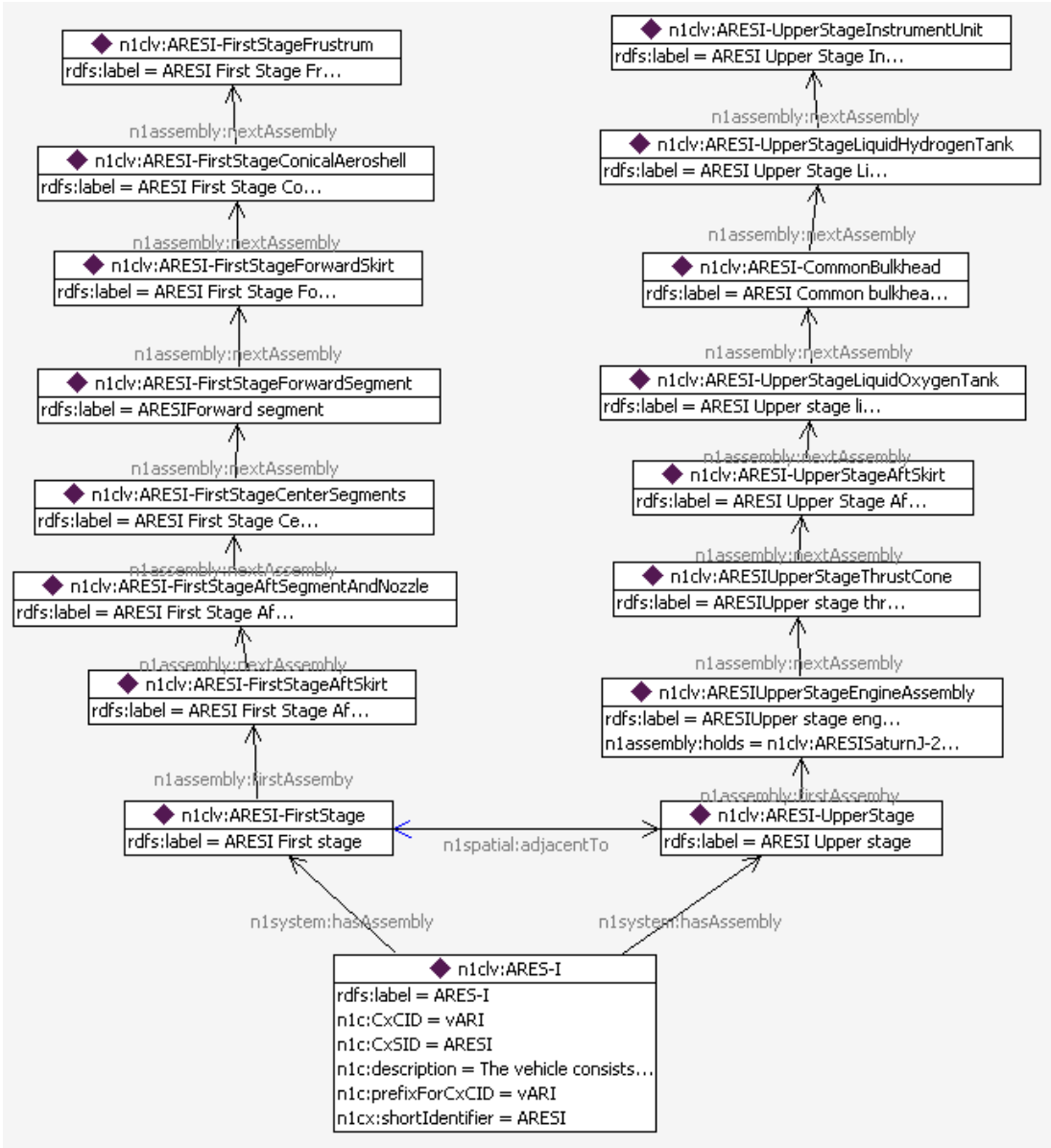
Crew Exploration Vehicle

Use of OWL restrictions to build SBFI models



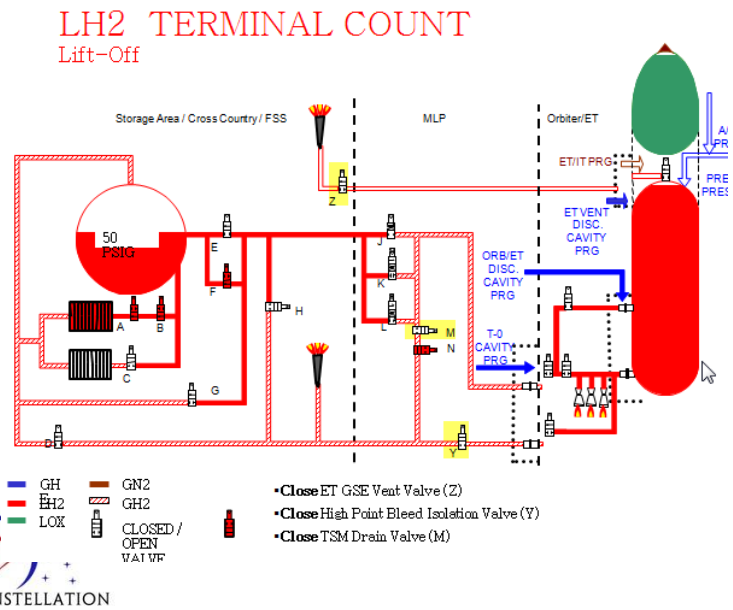
Crew Launch Vehicle - ARESI

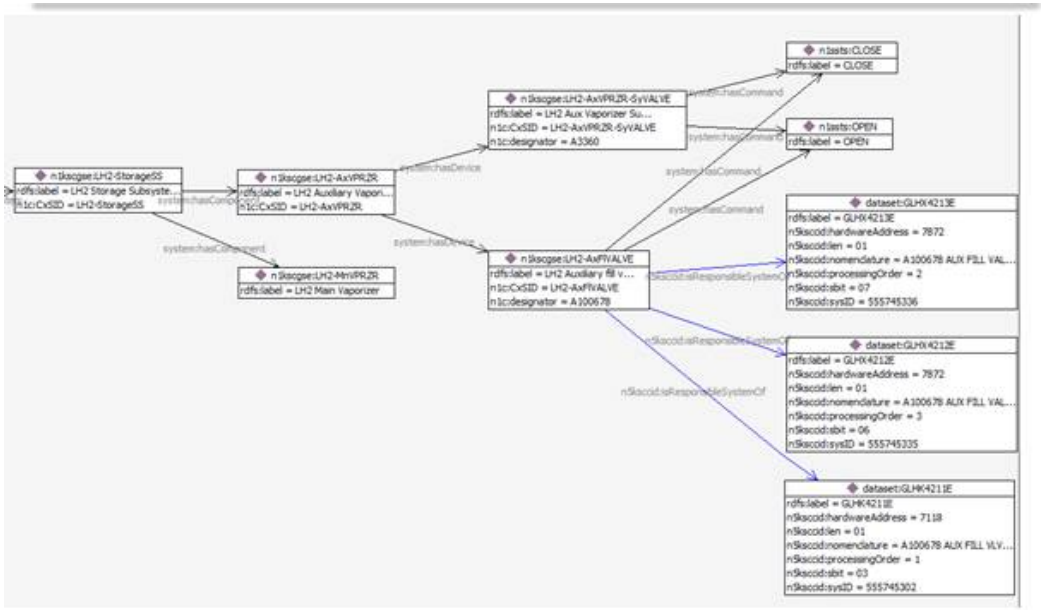




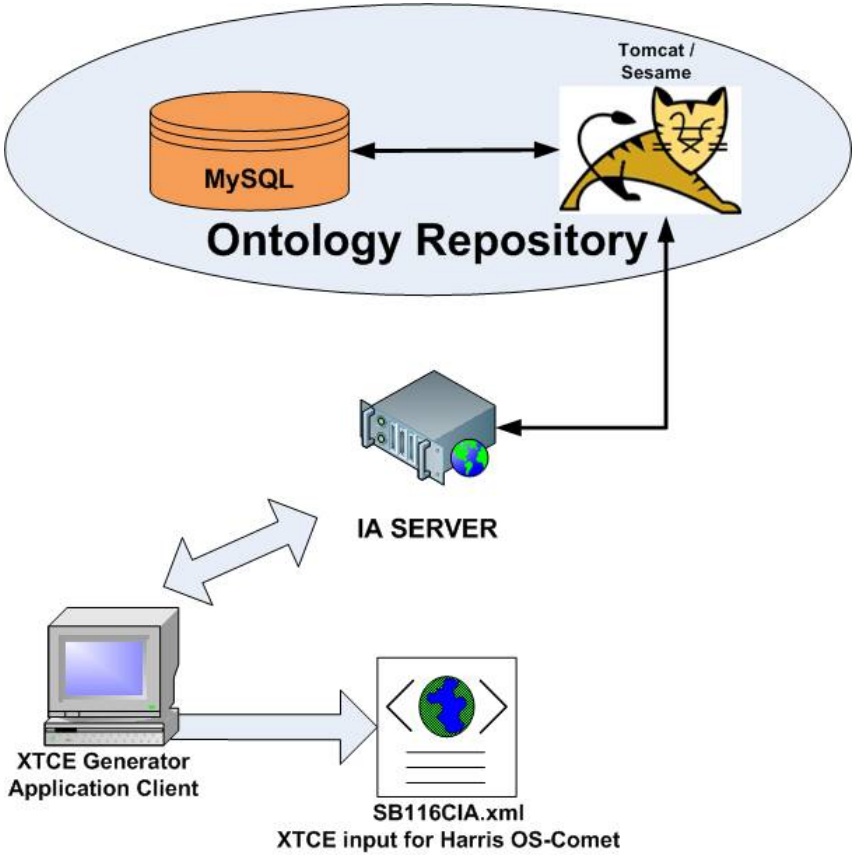
KSC Launch Control System

Ontologies model and locate Devices in their Functional Hierarchies for checking out of launch sequence operations





KSC Launch Control System: CxDA Ontologies generate XTCE for metadata exchange with ground systems



In Conclusion

1. Ontologies and Ontology-Based registries enable precise specifications of enterprise data and its use
2. Semantic web technologies provide flexiblity for interoperability and generation of work-products
3. Ontology-Based Grammar Engines enforce consistency of naming and identifier rules
4. “Connecting the dots” across the enterprise needs support for federation and governance
5. Federated Systems of Registries can be implemented using databases with Semantic Web Technologies

Thank You

Ralph Hodgson
E-mail: rhodgson@topquadrant.com