

Rapid COI Infospaces Creation and Deployment Using KAoS and COE

Andrzej Uszok, Jeffrey M. Bradshaw, Tom Eskridge,
James Lott, Niranjan Suri

Florida Institute for Human and Machine Cognition (IHMC)

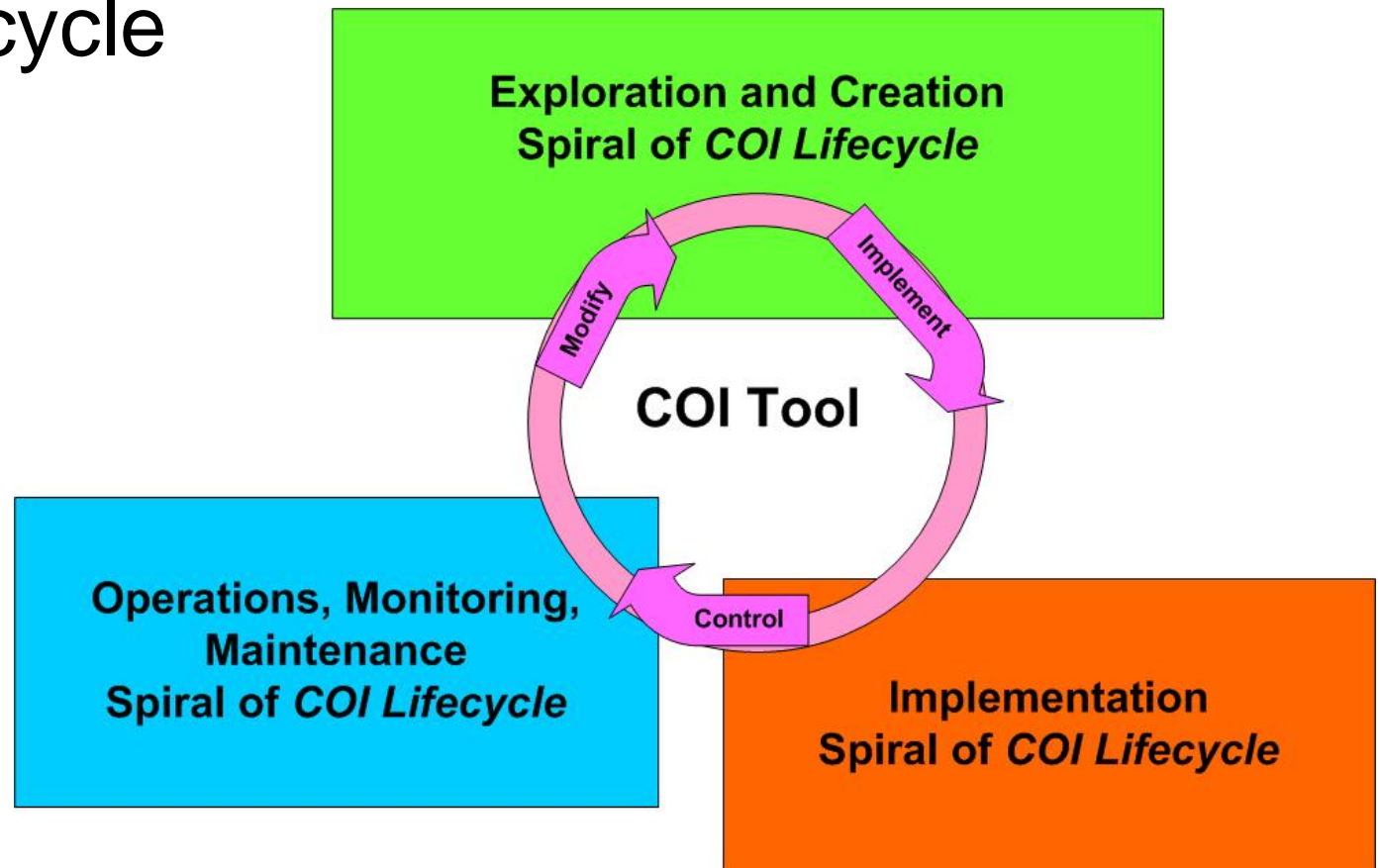
40 So. Alcaniz St., Pensacola, FL 32502

+1 (850) 202-4462

{auszok, jbradshaw, teskridge, jlott, nsuri}@ihmc.us

Problem Addressed in Project

Lack of methodologies and software tools to effectively support Communities of Interest (COI) lifecycle spirals.



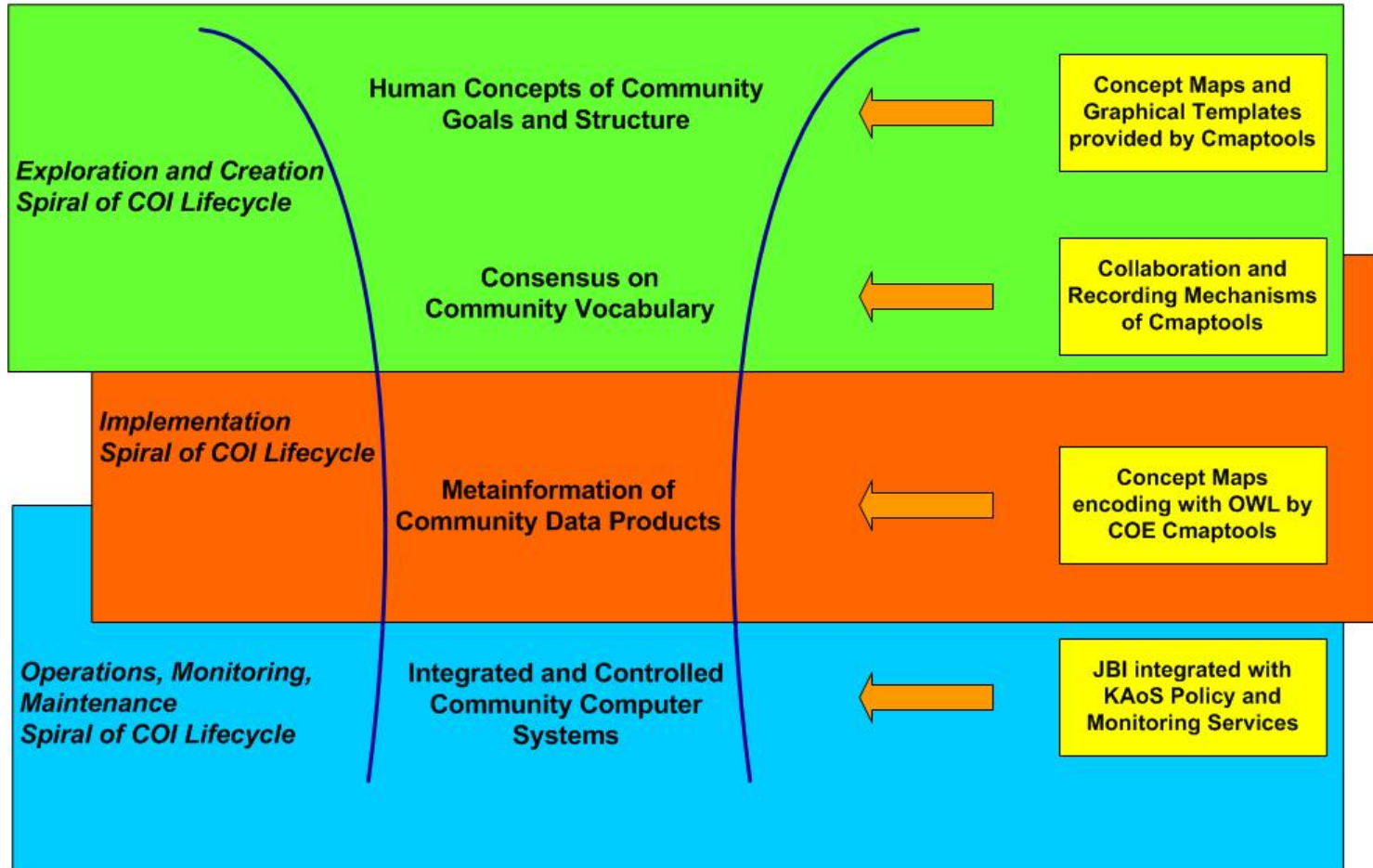
COI Concepts

- Assets and partnerships situated in the context of the mission they are supporting.
- Specific producers, consumers, data product and policies
- Many types of information must be captured in an easy-to-understand form:
 - information needs – COI scope,
 - types of information
 - types of consumers
 - infospace managers
 - applications used by consumers
 - degree of information integration
 - information security activities
 - consensus set of vocabulary terms and definitions
- Allow to determine whether the aggregated assets are adequate for performing the mission

COI-Tool Ideas

COI Lifecycle Needs

Solutions



Requirements-Features Overview

COI Lifecycle Needs

Solutions

Exploration and Creation

Easy-to-understand formal models of COI information requirements

Cmap Ontology Editor (COE)

Support for collaborative COI development

COE recording and model-sharing features

Ease of reuse

COE graphical templates

Implementation

Link abstract COI model of producers/consumers to actual assets and data resources

KAoS-COE model-mapping and automatic stub-generation features, links to meta-info

Data product policies

KAoS policy services

Harmonization of vocabulary

Simple semantic translation

Operations, Monitoring, Maintenance

Monitoring configuration, activity state, and policy compliance

KAoS activity and obligation monitoring features

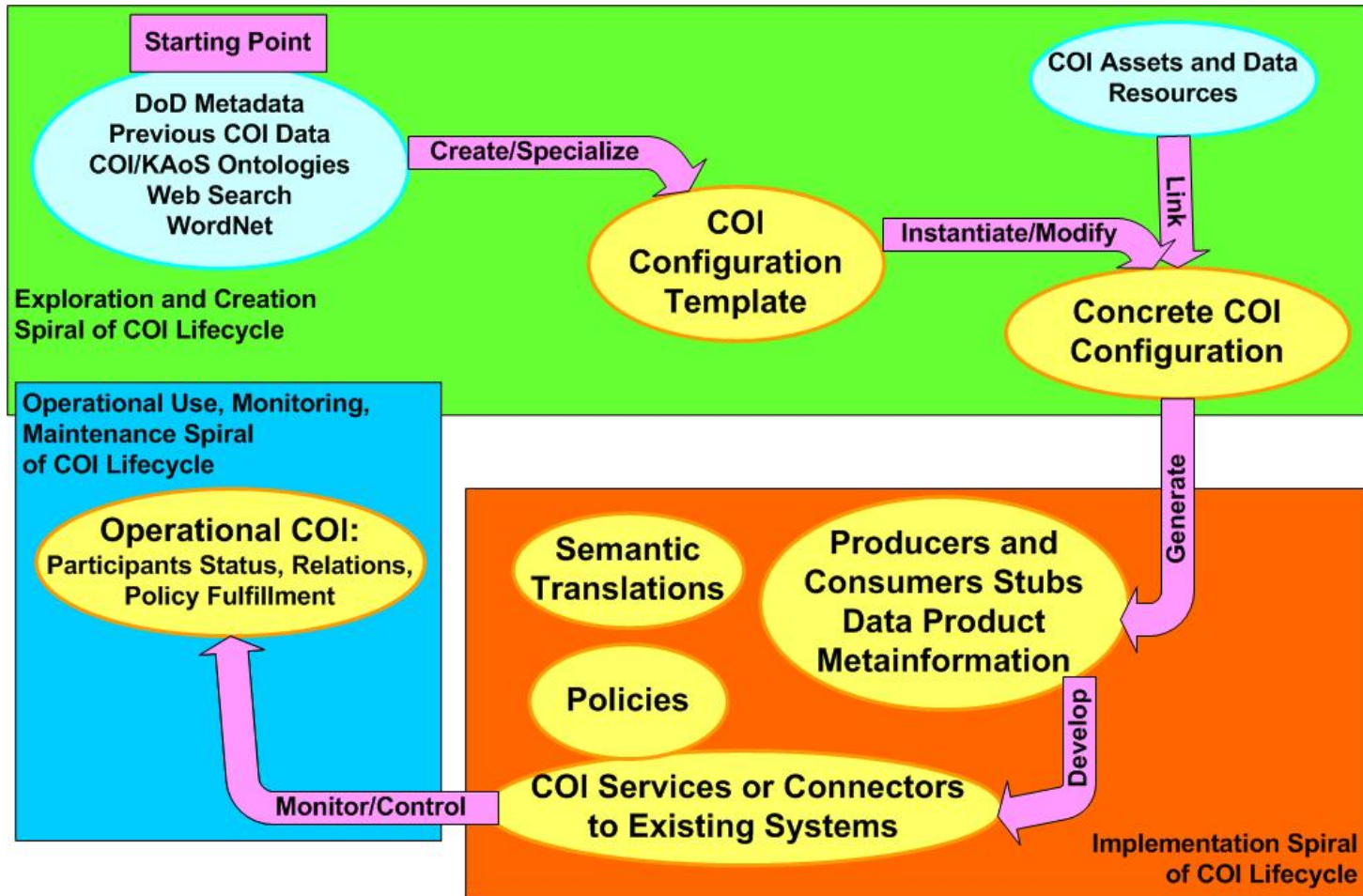
Monitoring producer/consumer/info object relationships

Additional KAoS monitoring features

Overall history and statistics

Recording mechanisms

COI-Tool Dataflow



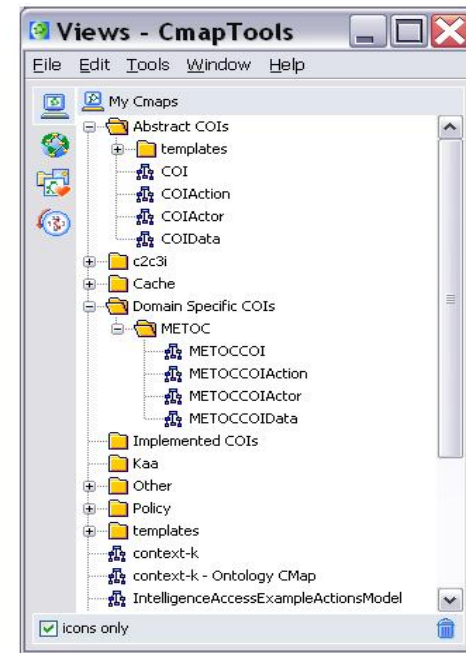
COI Demo Development

- Selected Korea METOC COI use case for testing and demonstration, which is a specialization of the generic METOC COI template
 - Based on concrete documents: the Korea METOC OPlan as well as AIR AND SPACE WEATHER OPERATIONS
- JBI RI clients simulate partners of this COI

Functionality Outline: Exploration Phase

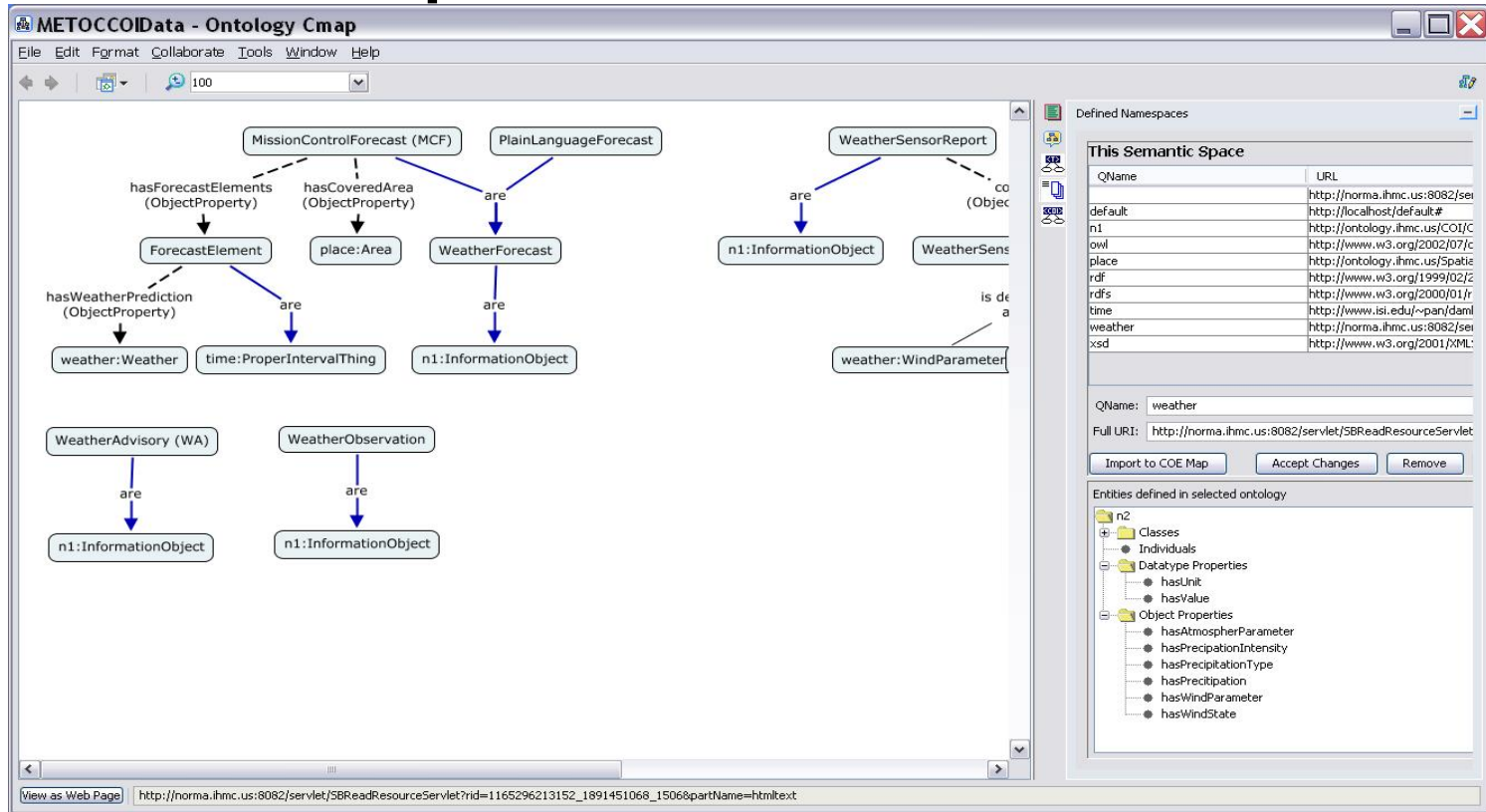
- Creation of a new COI
- Definition of COI properties, Roles and Data Product
- Usage of Web Search and Wordnet
- Definition of COI Roles relations to COI Products
- Usage of Concept Map Templates for map elements
- *Usage of DoD Metainformation Repository (not yet ready)*
- Usage of Collaboration Functionality
- Addition of links and multimedia resources
- Access to Map historical versions
- Creation of a new implemented COI Configuration

Creation of a new COI



- METOC COI creates a new COI configuration placeholder for definitions of METOC partners (roles/actors), data products, *classes of actions*, properties like managers, applications,, etc.

Definition of COI properties, roles, products, etc.



METOC COI manager graphically defines concepts and their relations. Uses namespace panel to access concept defined in other concept maps (ontologies) such as the weather ontology when defining weather forecast products.

Usage of Web Search and Wordnet

The screenshot displays the METOCCOIData - Ontology Cmap application. The main window shows an ontology graph with nodes like MissionControlForecast (MCF), PlainLanguageForecast, WeatherSensorReport, ForecastElement, weather:Weather, WeatherAdvisory (WA), and n1:InformationObject. A 'Dictionary and Thesaurus' window is open, showing the word 'weather' and its definitions and related words. The 'Definitions' section lists 'Nouns' and 'Verbs'. The 'Related Words' section lists 'atmospheric phenomenon', 'cloud', 'air pocket', 'aurora', 'condensation', 'condensate', 'front', 'greenhouse effect', and 'double click to see definition'. The 'Replace with' field contains 'cloud'. The 'Import to COE Map' button is visible in the bottom right of the dictionary window.

Defined Namespaces

QName	URL
default	http://norma.ihmc.us:8082/ser
n1	http://localhost/default#
n1	http://ontology.ihmc.us/COI/C
owl	http://www.w3.org/2002/07/c
place	http://ontology.ihmc.us/Spatia
df	http://www.w3.org/1999/02/z
dfs	http://www.w3.org/2000/01/r
ime	http://www.isi.edu/~pan/dam
weather	http://norma.ihmc.us:8082/ser
sd	http://www.w3.org/2001/XML:

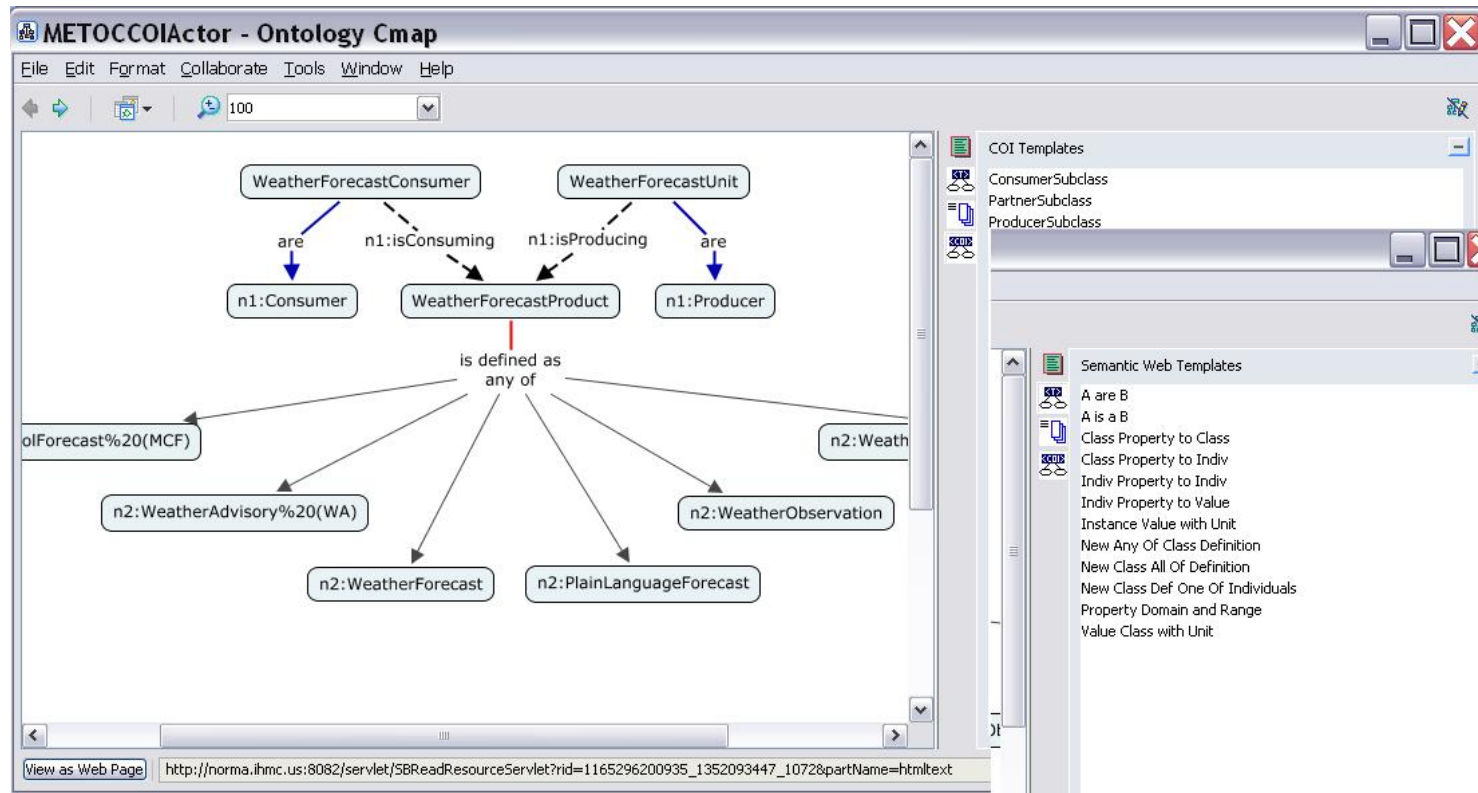
QName: weather
Full URI: http://norma.ihmc.us:8082/ser/vlet/SBReadResourceServlet

Entities defined in selected ontology

- n2
 - Classes
 - Individuals
 - Datatype Properties
 - hasUnit
 - hasValue
 - Object Properties
 - hasAtmospherParameter
 - hasPrecipitationIntensity
 - hasPrecipitationType
 - hasPrecipitation
 - hasWindParameter
 - hasWindState

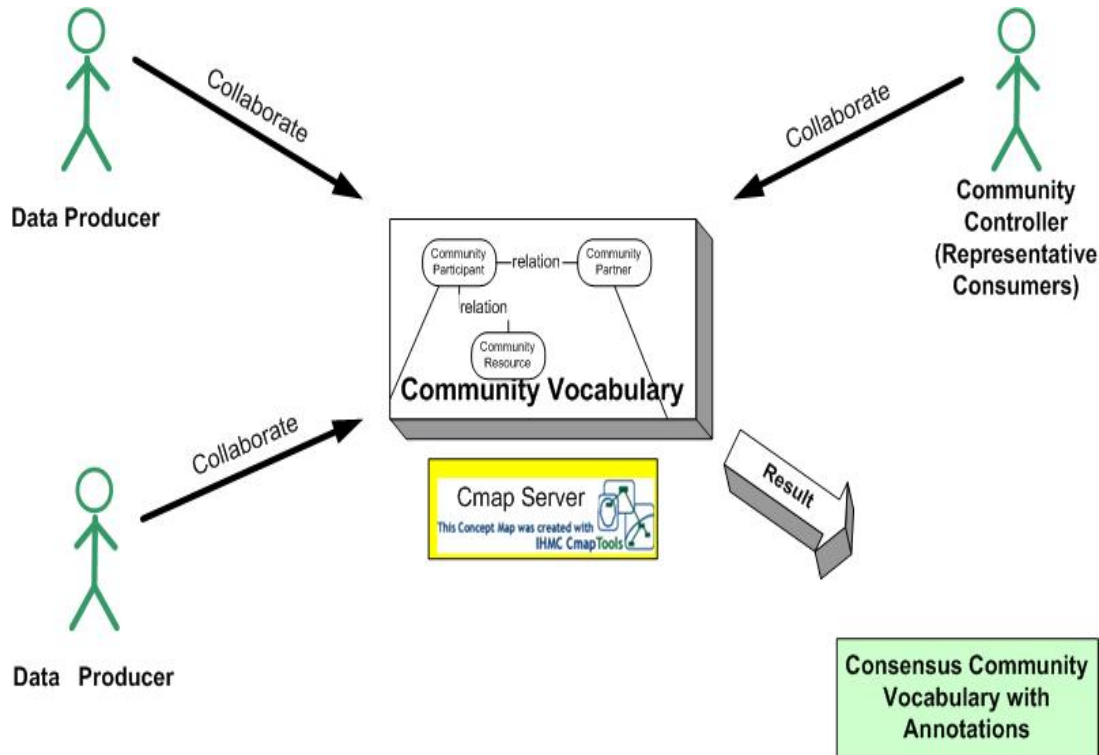
METOC COI manager uses Web resources to get weather definition and related concepts. Can easily create a new map node using the button from this window

Usage of Concept Map Templates for map elements



METOC COI manager uses Cmap Templates to start definition of specific COI roles of consumers of weather forecasts, producers of weather sensor data, etc.

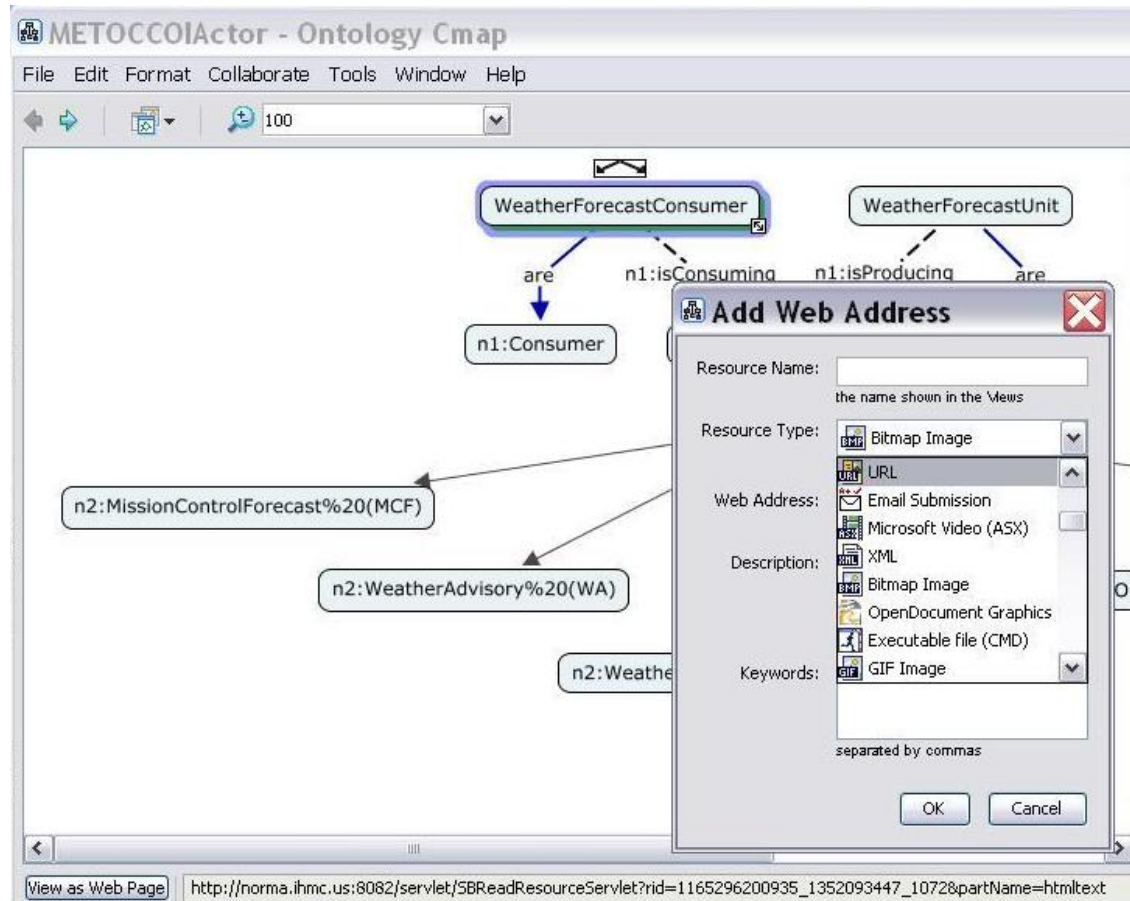
Usage of Collaboration Functionality



Partners are able to simultaneously open and edit a concept map with Community Vocabulary, they see other edits and annotations and are able to send chat messages to themselves.

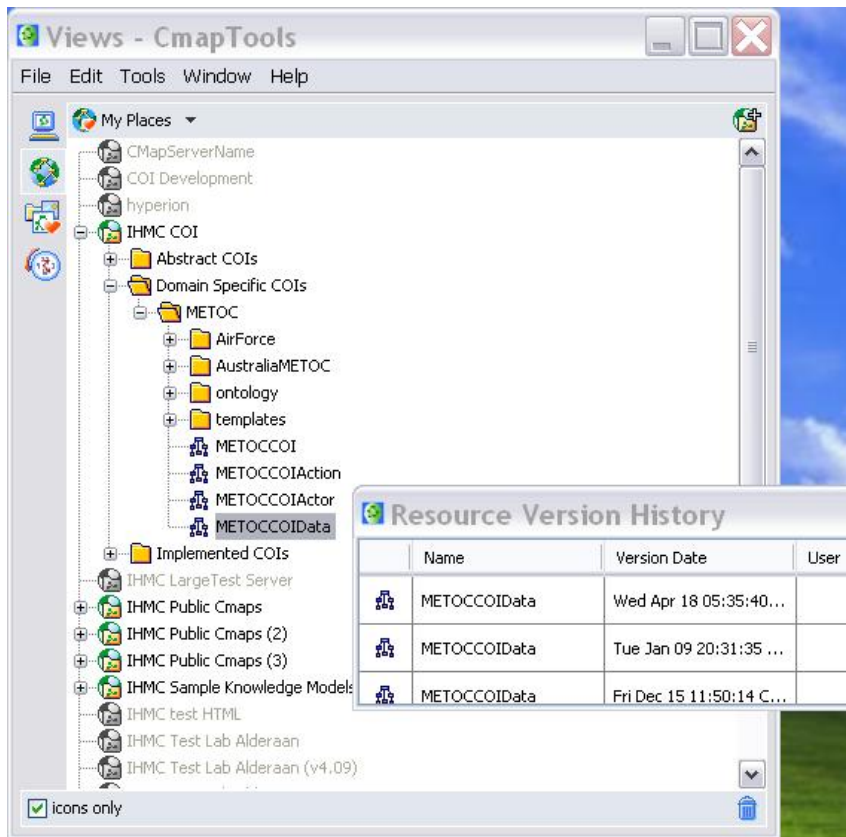
METOC COI partners are able to archive **Consensus** on weather concepts definitions, weather sensor product definitions, etc.

Addition of links and multimedia resources



METOC COI manager adds urls to weather services, links to documents and maps to COI concept in order to facilitate their human understanding.

Access to Map historical versions

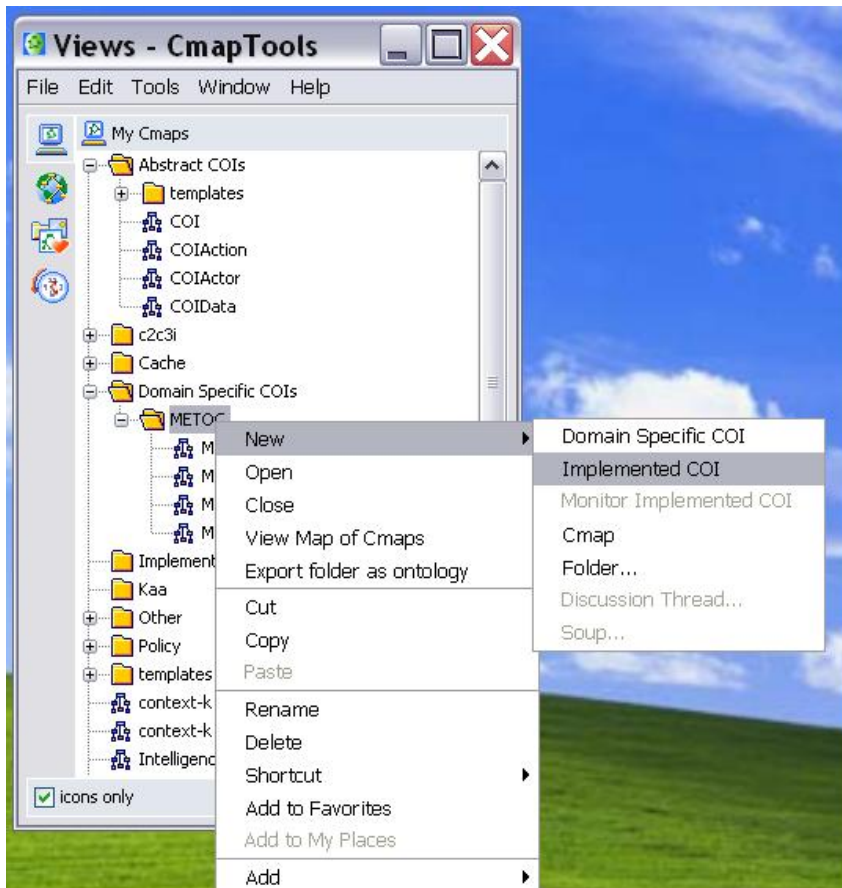


The screenshot shows the 'Views - CmapTools' window. The left pane displays a hierarchical tree structure under 'My Places'. The tree includes folders like 'Abstract COIs', 'Domain Specific COIs', and 'Implemented COIs'. Under 'Domain Specific COIs', there is a 'METOC' folder containing sub-folders like 'AirForce', 'AustraliaMETOC', 'ontology', and 'templates'. Below these are files like 'METOCCOI', 'METOCCOIAction', 'METOCCOIActor', and 'METOCCOIData'. The 'Implemented COIs' folder contains 'IHMC LargeTest Server', 'IHMC Public Cmaps', 'IHMC Public Cmaps (2)', 'IHMC Public Cmaps (3)', 'IHMC Sample Knowledge Models', 'IHMC test HTML', 'IHMC Test Lab Alderaan', and 'IHMC Test Lab Alderaan (v4.09)'. The 'METOCCOIData' file is selected. A 'Resource Version History' dialog is open in the foreground, showing a table of version history for 'METOCCOIData'.

	Name	Version Date	User
	METOCCOIData	Wed Apr 18 05:35:40...	
	METOCCOIData	Tue Jan 09 20:31:35 ...	
	METOCCOIData	Fri Dec 15 11:50:14 C...	

COI manager can access previous versions of the developed map from the integrated Subversion repository.

Creation of a new implemented COI Configuration

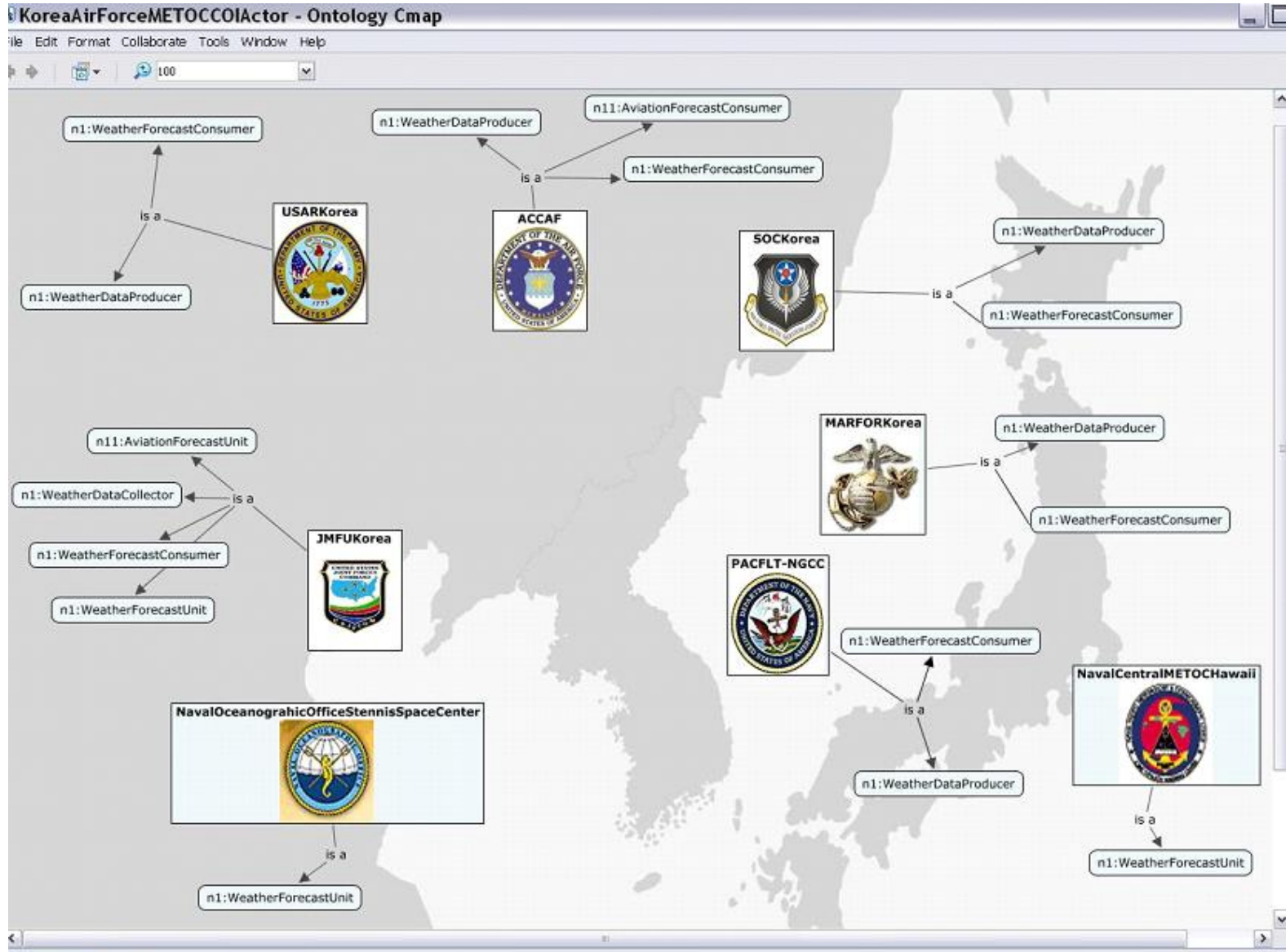


Koreas METOC COI manager creates a implementation of the METOC COI for the Korea military area.

It will assign COI roles to resources in his disposer e.g.: a specific military unit will have a role of a Forecast Provider; AirForce, Army, etc. units will be responsible for weather data collection, etc.

The resource can be any physical assets e.g.: military unit, UAV, databases able to fulfill the given COI role.

Korea METOC COI Configuration



Functionality Outline: Implementation Phase

- Generation of bootstrap files and code skeletons
- Integration between KAoS COI layer and JBI client
- Definition of policies
- Definition of semantic translation

Generation of bootstrap files and code skeletons



Korea METOC COI manger generates KAoS bootstrap files (*source code stubs*) for each of the COI partner and send them to the developers in the particular unit.

- Contained only the relevant information for this partner,
- Based on it developers will implement IMS JBI client providing and consuming given metainformation types and participating in the COI,
- Integrates the client with the KAoS Policy Service.

Example KAoS Bootstrap File

JMFUKorea

```
(java-agent-services
(key-prefix javax.agent.service)
(agent-configuration
(id http://norma.ihmc.us:8082/servlet/SBReadResourceServlet?rid=1165343712962_72793303_2751&partName=owl#JMFUKorea)
(domains
(http://norma.ihmc.us:8082/servlet/SBReadResourceServlet?rid=1165343714072_19188942_2985&partName=owl#KoreaAirForceMETOC-COI 1)
)
(ontological-types
(http://norma.ihmc.us:8082/servlet/SBReadResourceServlet?rid=1165296200935_1352093447_1072&partName=owl#WeatherForecastConsumer 1)
(http://norma.ihmc.us:8082/servlet/SBReadResourceServlet?rid=1165343277127_693030123_1812&partName=owl#AviationForecastUnit 2)
(http://norma.ihmc.us:8082/servlet/SBReadResourceServlet?rid=1165296200935_1352093447_1072&partName=owl#WeatherDataCollector 3)
(http://norma.ihmc.us:8082/servlet/SBReadResourceServlet?rid=1165296200935_1352093447_1072&partName=owl#WeatherForecastUnit 4)
)
(OntologicalOperations
(http://ontology.ihmc.us/COI/COIActor.owl#isProducing
http://norma.ihmc.us:8082/servlet/SBReadResourceServlet?rid=1165343275317_500653606_1495&partName=owl#AviationOperationsForecast)
(http://ontology.ihmc.us/COI/COIActor.owl#isProducing
http://norma.ihmc.us:8082/servlet/SBReadResourceServlet?rid=1165296200935_1352093447_1072&partName=owl#WeatherForecastUnitProducts)
(http://ontology.ihmc.us/COI/COIActor.owl#isProducing
http://norma.ihmc.us:8082/servlet/SBReadResourceServlet?rid=1165343275317_500653606_1495&partName=owl#FlightWeatherBrifings)
(http://ontology.ihmc.us/COI/COIActor.owl#isProducing
http://norma.ihmc.us:8082/servlet/SBReadResourceServlet?rid=1165343275317_500653606_1495&partName=owl#AviationOperationsObservation)
(http://ontology.ihmc.us/COI/COIActor.owl#isConsuming
http://norma.ihmc.us:8082/servlet/SBReadResourceServlet?rid=1165296200935_1352093447_1072&partName=owl#WeatherForecastUnitProducts)
)
)
)
```

COI Client Stub

- Generation of Java code opening of publish/subscription channels code specific to a given partner JBI client
 - *weatherSensorReportSub =
initSubscriberChannel(OntologyConcepts.WEATHER_SENSORREPORT_JBI_TYPE, "1.0", null);*
 - *WeatherWarningChecker callback = new
WeatherWarningChecker(weatherSensorReportSub,
OntologyConcepts.WEATHER_SENSORREPORT_CLASS);*
 - *weatherSensorReportSub.setSequenceCallback(callback);*
- Being develop generation of:
 - XMLSchema for metainformation
 - ant script generating necessary mappings of ontology to Java constances, necessary to build partner data product metainformation
 - ant script starting a given partner JBI client with the necessary configuration set-up

Integration between KAoS COI layer and JBI client

- Provided the following functionality:
 - Partner activation and relation **monitoring**
 - Reports when partner is activated and what data products and from whom it received
 - **Statistical** information about the relation
 - First/last time of data consumption, frequency of consumption, average time between consumption, etc.
 - **Authorization** policy checking
 - **Obligation** policy monitoring
 - Semantic **Translation**
- is provided as interceptors for JBI, to activate:
 - Rename client.properties file to clientOriginal.properties and copy the provided client.properties file from COI-Tool distribution

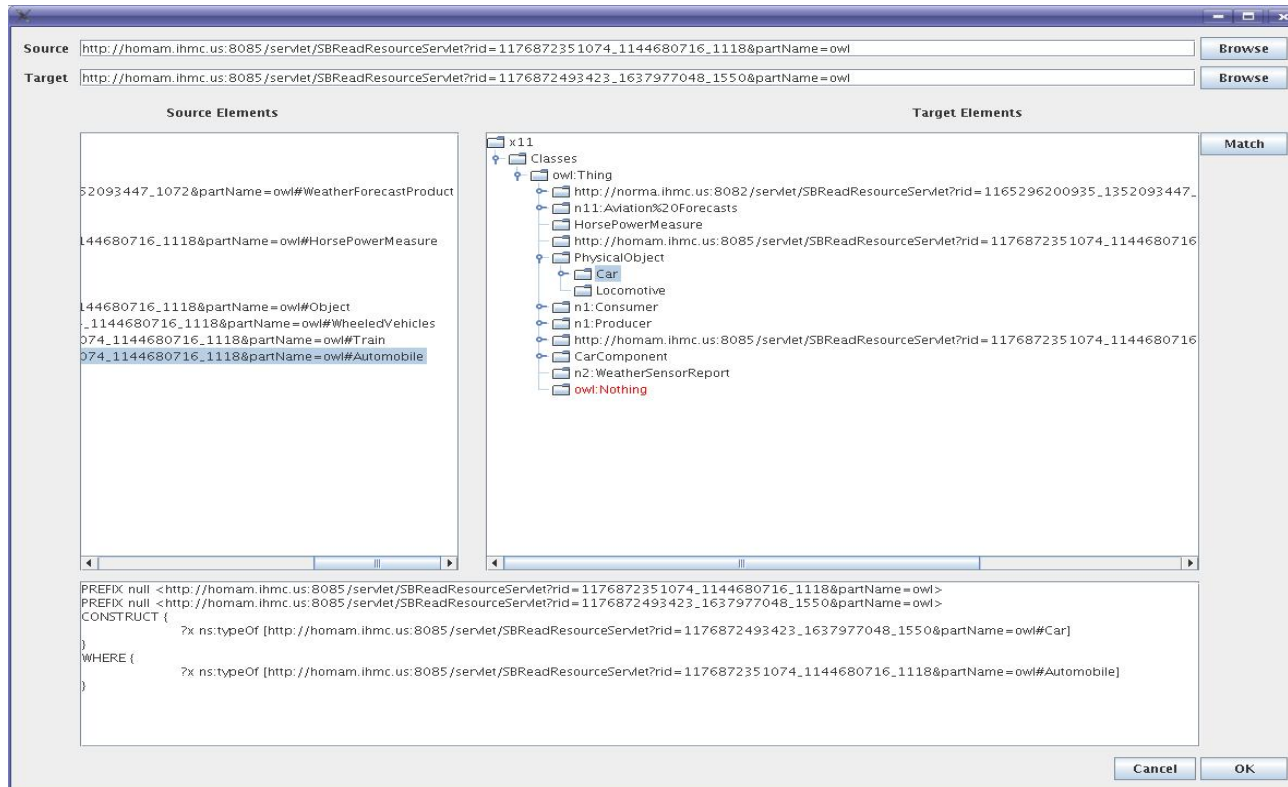
KAoS-COI client.properties

```
<capi.implementation>  
  <sequences>  
    <subscription>coi.jbi.interceptors.KAoSSubscriberSequence</subscription>  
    <publication>coi.jbi.interceptors.KAoSPublisherSequence</publication>  
    <query>coi.jbi.interceptors.KAoSQuerySequence</query>  
  </sequences>  
  <connections>  
    <connection.manager>  
      coi.jbi.interceptors.KAoSConnectionManager</connection.manager>  
    <connection>coi.jbi.interceptors.KAoSConnection</connection>  
  </connections>  
  <repositories>  
    <mdr>mil.af.rl.im.capi.client.core.plugin.j2ee.typemgt.MetadataRepository</mdr>  
  </repositories>  
</capi.implementation>
```

Definition of COI policies

- OWL representation of COI configuration defined in COI-Tool can be directly used as ontology vocabulary to define COI policies in KAoS
- For instance METOC manger can define policies:
 - preventing publication of certain data products; Korea METOC COI can dynamically activate these policy whenever necessary
 - obliging to produce certain data produce with a given frequency, e.g. weather data sensor has to be produced every 2 hours.
- Policies can be defined using KPAT, during runtime a list of policies applicable to a given partner can be also accessed from COI-Tool

Definition of Semantic Translation



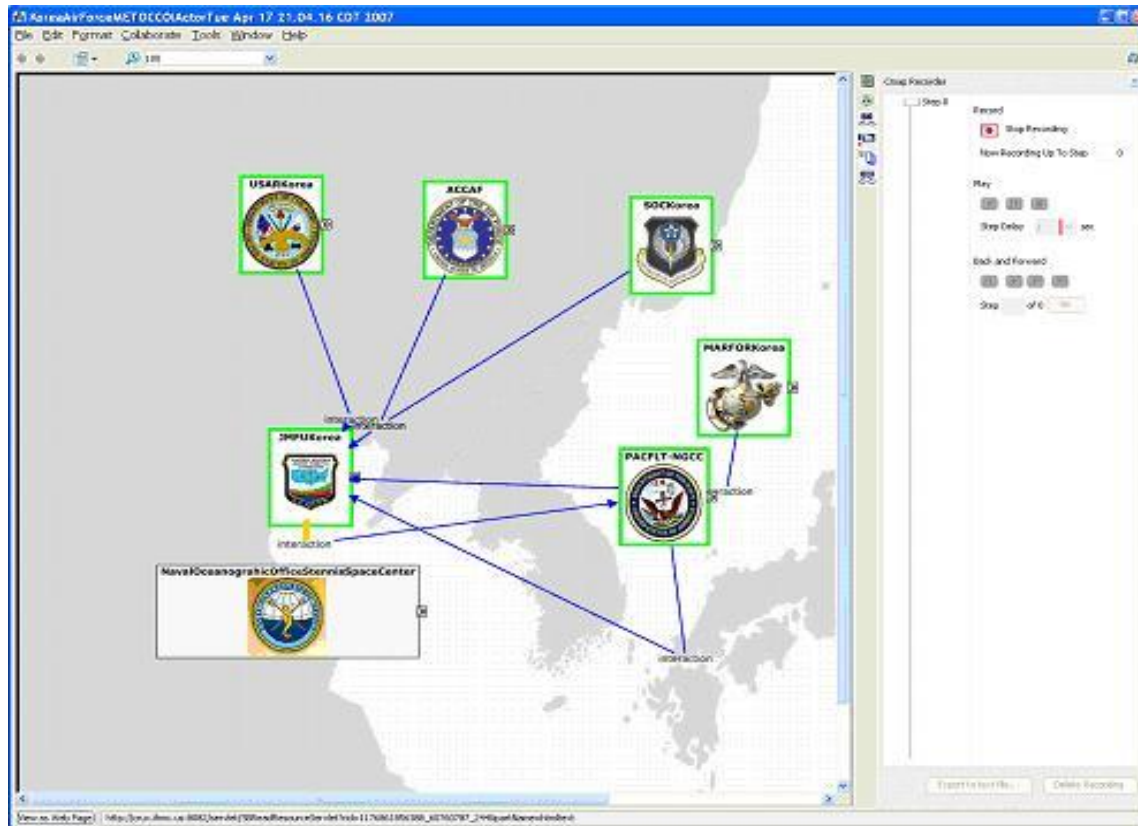
In reality, when implementing COI; partners will use different then agreed in the COI representations of data, for instance an Australian ship joining the Korea METOC COI is likely to use different weather data sensor format.

Korea COI manager can use COI-Tool to define alignment between the COI sensor data and the Australian sensor data representation.

Functionality Outline: Operational Phase

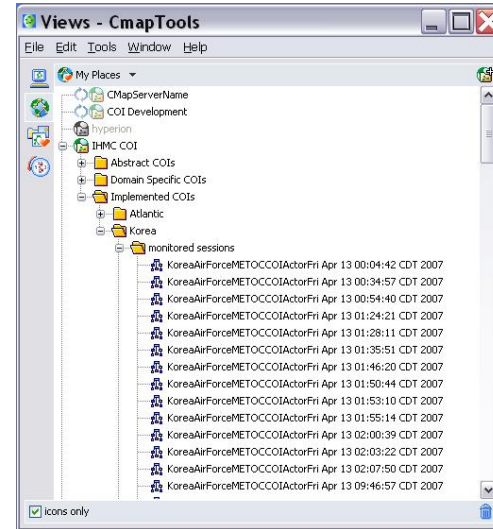
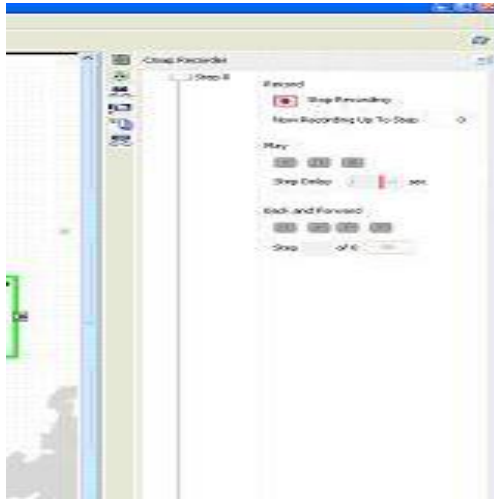
- Monitoring of partners activation and relations
- Access to statistical information about the activities on a given relation
- Enforcement of authorization policies
- Monitoring of obligation policies
- Information about new partner and incompatibilities
- Recording of monitoring session and access to recorded session

Monitoring of partners activation and relations



Korea METOC COI can connect to the KAOs controlling the COI and see active partners and relations between them.

Recording of monitoring session and access to recorded session



The recorded history of monitored session is stored in a special folder.

The session can be later replayed.

COI-Tool Project Web Site

- The project web site:
<http://ontology.ihmc.us/COI/>
- The web site contains:
 - general COI resources
 - METOC COI references
 - COI-Tool distribution and installation guide
 - link to Cmap Server with the demo METOC COI Configuration concept maps and ontologies
 - Video tutorials

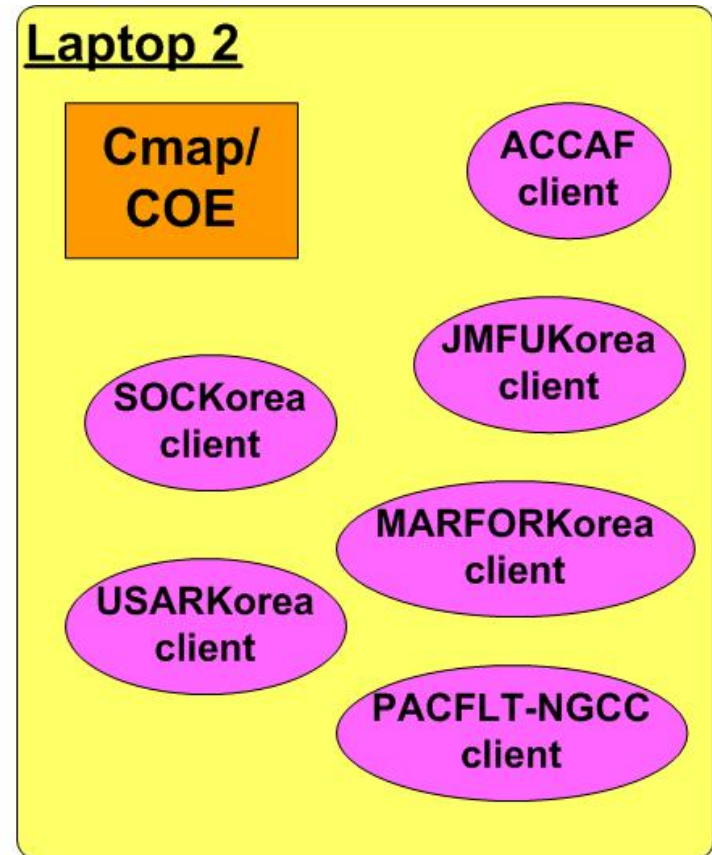
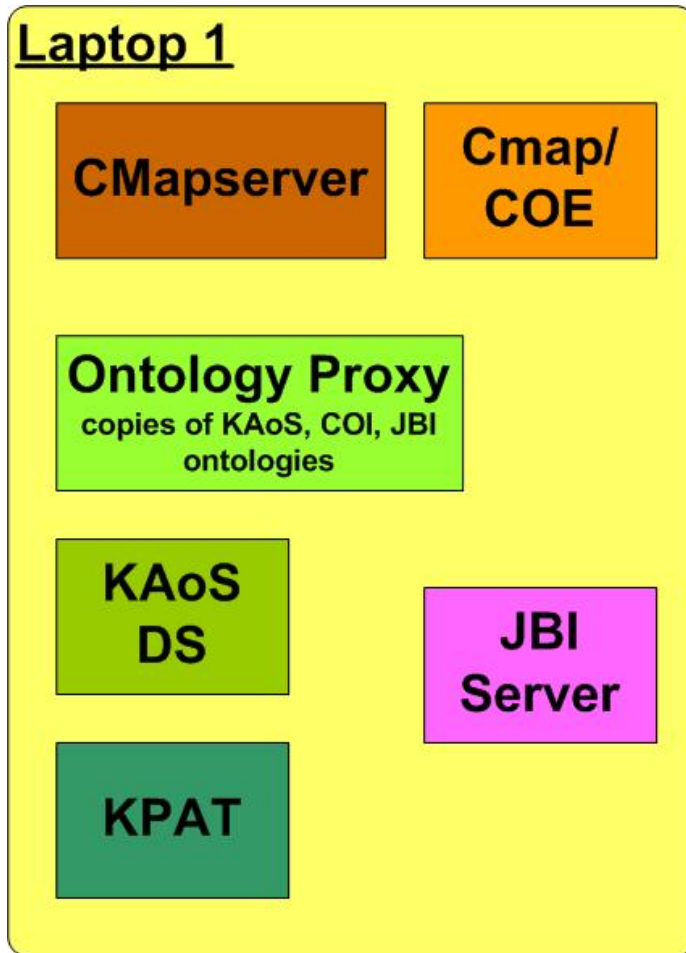
Possible COI-Tool Extensions

- Integration with DoD Metadata Registry through the Web Service interface
 - to browse and reuse existing Metadata and Taxonomies
 - to contribute developed COI Taxonomies and Schemas (when the DoD Metadata Registry Web Service interface has this functionality)
- Integration with IMS Metainformation catalog to automatically submit COI products metainformation schemas
- Exploration of support for COI workflow/actions
- Better support for dynamic communities

COI-Tool Summery

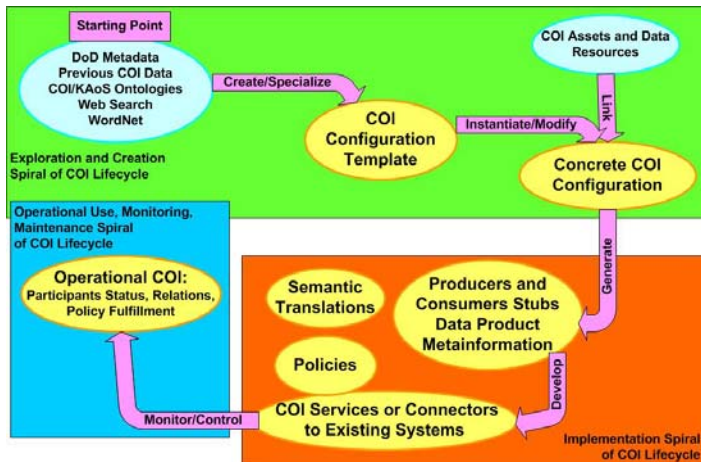
- Capture and share COI configurations in in two synchronized representations:
 - Graphical concept maps – easy to use by human
 - Transparent OWL encoding – computer processable
- Unique user-friendly Cmap environment with integrated Web Search, simultaneous collaboration and version control
- Facilitation of the COI implementation through integration with IMS/JBI RI and KAoS
- Reuse of COI models

Korea METOC COI Demo Setup





Rapid COI Infospaces Creation and Deployment using KAoS and CMaps



ID	Task Name	Start	Finish	Duration	2006												2007	
					Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb		
1	Ontology Development	3/16/2006	1/31/2007	46w	[Progress Bar]													
2	Design and create a generic COI ontology	3/16/2006	7/4/2006	15.8w	[Progress Bar]													
3	Design and create a more specific ontology for COI experimentation	6/7/2006	1/31/2007	34.2w	[Progress Bar]													
4	JB1 and KAoS Integration	4/18/2006	12/25/2006	37.2w	[Progress Bar]													
5	Wrap JB1 client to enable and enforce semantically-rich client descriptions	4/10/2006	6/30/2006	12w	[Progress Bar]													
6	Extend KAoS reasoning mechanisms to exploit the sophistication of the COI ontologies	6/27/2006	12/25/2006	26w	[Progress Bar]													
7	Integrate the KAoS Registration Service with JB1	5/18/2006	9/1/2006	15.4w	[Progress Bar]													
8	KAoS and CmapTools Integration	4/11/2006	12/13/2006	35.4w	[Progress Bar]													
9	Develop a new of CmapTools module allowing to observe KAoS registered COI resources, their relations and ontology (in COE)	4/11/2006	10/24/2006	28.2w	[Progress Bar]													
10	Extend CmapTools with graphical templates for the generic COI ontology	6/1/2006	12/13/2006	28w	[Progress Bar]													
11	COI policies development	9/4/2006	1/31/2007	21.6w	[Progress Bar]													
12	Develop and demonstrate policies to govern COI management by the IMS personnel	9/4/2006	1/31/2007	21.6w	[Progress Bar]													
13	Perform any necessary extensions to the KAoS framework to support policy specification for COI	9/4/2006	11/1/2006	8.6w	[Progress Bar]													
14	COI Demonstration Development	9/4/2006	2/15/2007	23.8w	[Progress Bar]													
15	Kickoff Meeting	3/31/2006	3/31/2006	.2w	[Progress Bar]													
16	First Quarterly Review Meeting	7/6/2006	7/6/2006	.2w	[Progress Bar]													
17	Second Quarterly Review Meeting First Project Demonstration	10/2/2006	10/2/2006	.2w	[Progress Bar]													
18	Third Quarterly Review meeting	12/14/2006	12/14/2006	.2w	[Progress Bar]													
19	Final Review Meeting - AFRL RRS Third Project Demonstration	2/28/2007	2/28/2007	.2w	[Progress Bar]													

Benefits to the War Fighter

- Capture and share COI configurations in in two synchronized representations:
 - Graphical concept maps – easy to use by human
 - Transparent OWL encoding – computer processable
- Unique user-friendly Cmap environment with integrated Web Search, collaboration and version control
- Facilitation of the COI implementation through integration with IMS/JB1 RI and KAoS
- Reuse of COI models

Technical Challenges

- Definition of common COI ontology
- Enrichment of semantic descriptions of COI infospace
- Comprehensive integration of three systems: KAoS, CMap and JB1
- Design of the data translation mechanisms
- Development of intuitive ontology templates for COI

R&D Effort Description

The objective of this effort is to investigate:

- (a) methodology and techniques for creating and maintaining COI infospaces;
- (b) mechanisms to leverage the KAoS Services framework and CmapTools in order to create a prototype enabling management of semantically-rich COI infospaces;
- (c) how these requirements could be supported by demonstration within a JB1 testbed