
AI Planning for Grid/Web Services Composition, Policy Analysis & Workflow

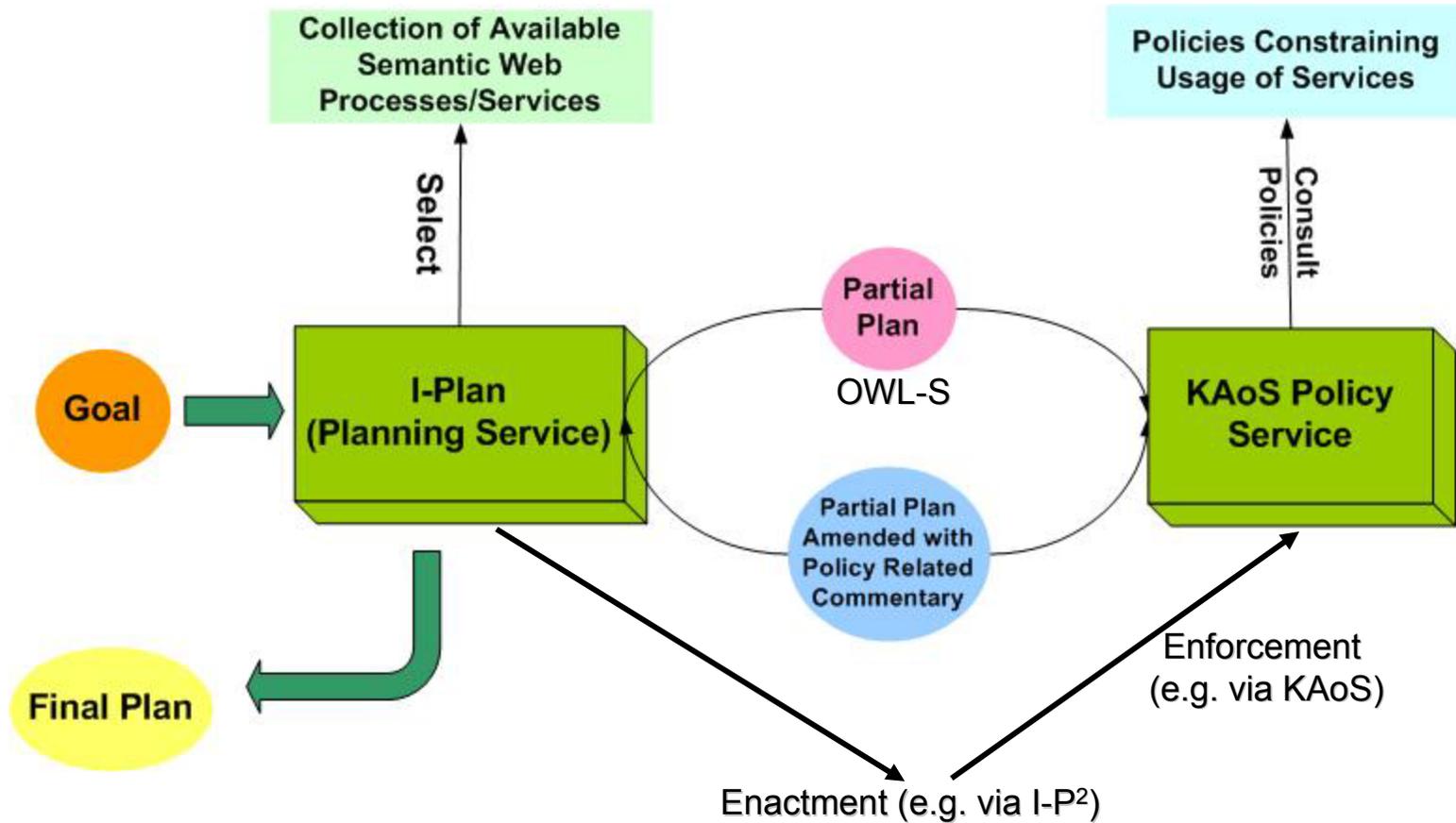
**Austin Tate & Jeff Dalton
AIAl, University of Edinburgh
Andrzej Uszok & Jeff Bradshaw
IHMC, Pensacola, FL**



Artificial Intelligence Applications Institute, University of Edinburgh, UK
Institute for Human and Machine Cognition, Pensacola, Florida



I-X/KAoS Composer (& Enactor)



Previous Relevant AIAI Work

- ***O-Plan***

- On-line web service exposing API via CGI scripts since 1994
- HTTP interface since 1997
- Simple - single user single-shot plan generator
- Mixed-initiative – multiple options, multiple users with multiple roles, long transactions, collaborative planning, execution and plan repair on failure
- Air Campaign Planning Workflow Aid - people and systems

- ***I-X***

- I-X supports the construction of mixed-initiative agents and systems which are intelligible to their users and to other systems and agents
- Dynamic workflow generation and reactive execution support
- I-Q query adaptor for OWL, OWL-S lookups via CMU Matchmaker, Semantic Web Queries via OWL and RDQL (AKTive Portal)
- I-Plan planning/re-planning tool

- ***CoAX and CoSAR-TS***

- Coalition Command and Control/Search and Rescue Task Support
- Use on CoABS Grid and with KAoS Domain and Policy Services



Previous Relevant IHMC Work

- **KAoS**

- Developed domain and policy services compatible with several popular agent (e.g., CoABS Grid, Cougar, Brahms, SFX) and distributed computing (e.g., CORBA, Grid Computing, Web Services) platforms
- Use of OWL to represent application domain concepts and instances, and policy information
- Analysis and policy disclosure algorithms built on top of Stanford's Java Theorem Prover

- **CoAX and CoSAR-TS**

- Use of KAoS to rapidly specify, deconflict, and enforce policies in coalition agents experiment (CoAX)
- Use of KAoS to define, deconflict, and enforce policies governing access to CMU Semantic Matchmaker information in conjunction with AIAI's I-X tool set (CoSAR-TS)



FY04 Progress

1. ***Initial exploration of the research agenda for using AI planners and workflow analysis capabilities as web service composition tools***
2. ***O-Plan Web Service experiments***
 - *Dealing with Inputs & Outputs*
 - *Recovering Dataflow from Plan Goal Structure*
 - *OWL-S Import & Export*
3. ***I-Plan***
 - *As a web service*
 - *As a Java planning tool (stand-alone and embedded)*
4. ***KAoS Policy Analysis of workflows***
 - *Translate instances of OWL-S processes into KAoS Action Classes to allow policies to be written about OWL-S processes*
 - *KAoS Policy Semantics extended for more sophisticated insertion of policy obligations into OWL-S composite processes*
 - *KAoS role-value-map extensions allow generation of richer OWL-S dataflow semantics*



FY04 Progress

5. ***Use KAoS Policy Analysis during I-Plan plan generation***
6. ***Scenarios***
 - ***Simple examples – e.g. document handling***
 - ***myGrid biochemistry scenario to identify tool requirements***
 - ***CoSAR scenario - Emerging web Interactive demo of all the integrated technology on CoSAR-TS scenario***

Explorations

- ***KAoS Workflow Policy Analyzer as a Web Service***
- ***Link to AKT work on OWL-S manual composition tool (SEdit)***



O-Plan/I-Plan OWL-S Importer

<http://ontology.ihmc.us/CoSAR-TS/CoSAR-TS-ServiceOntology.owl>

```
(refinement ArabelloCoastGuardCutter_Service (ArabelloCoastGuardCutter_Service ?hospitalLocation_In ?pickupLocation_In ?countryofHospital_In to ?injuryType_out ?person_out)
(variables ?hospitalLocation_In ?pickupLocation_In ?countryofHospital_In ?injuryType_out ?person_out)
(constraints
(world-state condition (type ?hospitalLocation_In) = Location)
(world-state condition (type ?pickupLocation_In) = Location)
(world-state condition (type ?countryofHospital_In) = Country)
(world-state effect (type ?injuryType_out) = Injury)
(world-state effect (type ?person_out) = Person))
(annotations
(output-objects = ((?injuryType_out Injury) (?person_out Person)))
(input-objects = ((?hospitalLocation_In Location) (?pickupLocation_In Location) (?countryofHospital_In Country))))))

(refinement GaoMarineHelicopter_Service (GaoMarineHelicopter_Service ?hospitalLocation_In ?pickupLocation_In ?countryofHospital_In to ?injuryType_out ?person_out)
(variables ?hospitalLocation_In ?pickupLocation_In ?countryofHospital_In ?injuryType_out ?person_out)
(constraints
(world-state condition (type ?hospitalLocation_In) = Location)
(world-state condition (type ?pickupLocation_In) = Location)
(world-state condition (type ?countryofHospital_In) = Country)
(world-state effect (type ?injuryType_out) = Injury)
(world-state effect (type ?person_out) = Person))
(annotations
(output-objects = ((?injuryType_out Injury) (?person_out Person)))
(input-objects = ((?hospitalLocation_In Location) (?pickupLocation_In Location) (?countryofHospital_In Country))))))

(refinement USArmyHelicopter_Service (USArmyHelicopter_Service ?hospitalLocation_In ?pickupLocation_In ?countryofHospital_In to ?injuryType_out ?person_out)
(variables ?hospitalLocation_In ?pickupLocation_In ?countryofHospital_In ?injuryType_out ?person_out)
(constraints
(world-state condition (type ?hospitalLocation_In) = Location)
(world-state condition (type ?pickupLocation_In) = Location)
(world-state condition (type ?countryofHospital_In) = Country)
(world-state effect (type ?injuryType_out) = Injury)
(world-state effect (type ?person_out) = Person))
(annotations
(output-objects = ((?injuryType_out Injury) (?person_out Person)))
(input-objects = ((?hospitalLocation_In Location) (?pickupLocation_In Location) (?countryofHospital_In Country))))))

(refinement USMarineHelicopter_Service (USMarineHelicopter_Service ?hospitalLocation_In ?pickupLocation_In ?countryofHospital_In to ?injuryType_out ?person_out)
(variables ?hospitalLocation_In ?pickupLocation_In ?countryofHospital_In ?injuryType_out ?person_out)
(constraints
(world-state condition (type ?hospitalLocation_In) = Location)
(world-state condition (type ?pickupLocation_In) = Location)
(world-state condition (type ?countryofHospital_In) = Country)
(world-state effect (type ?injuryType_out) = Injury)
(world-state effect (type ?person_out) = Person))
(annotations
(output-objects = ((?injuryType_out Injury) (?person_out Person)))
(input-objects = ((?hospitalLocation_In Location) (?pickupLocation_In Location) (?countryofHospital_In Country))))))
```

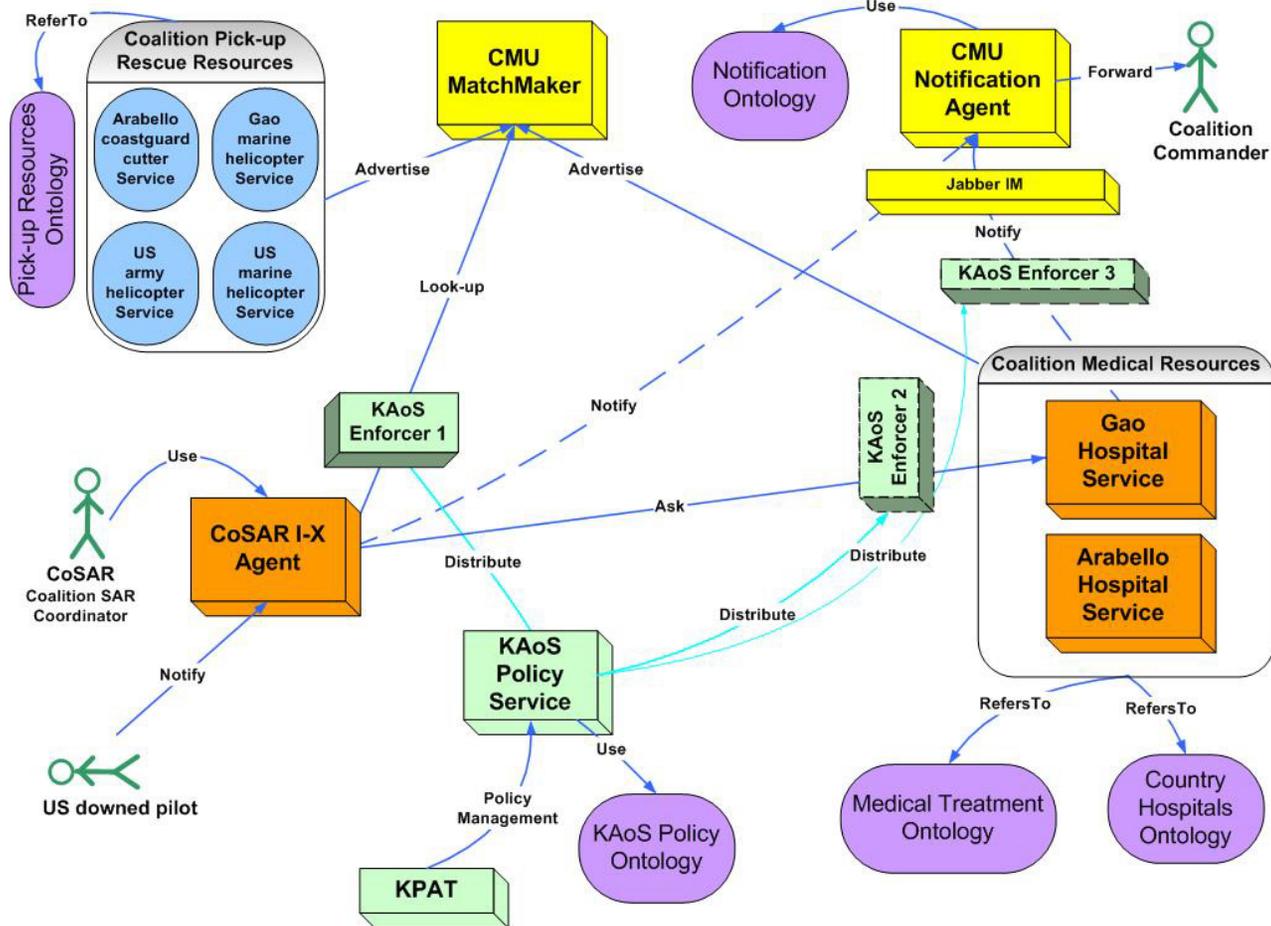
KAoS Policy about an OWL-S Process

Using vocabulary from CoSAR -TS OWL-S Process ontology policies

The screenshot displays the KPAT (KAoS Policy Administration Tool v2.0) interface. The window title is "KPAT [- KAoS Policy Administration Tool v2.0". The interface is divided into several sections:

- Domain View:** A tree view on the left showing a hierarchy of actor classes. The "Co-SARTS_Coordinator" class is selected and highlighted in blue.
- Template Information:** A section at the top right containing:
 - Name:** Generic OWL Editor
 - Description:** Generic editor for OWL policies
- Policy Editor:** A central area for defining a policy template:
 - Policy id:** urn:KAoS#policy-b0f08ea6-00fc-0000-8000-0000deadbeef
 - Policy name:** NotificationObligation
 - Description:** CoSAR-TS Coordinator is obliged to notify Coalition Commander
 - Priority:** 1
 - Policy Statement:** A dropdown menu shows "Co-SARTS_Coordinator" selected, followed by "is" and "obligated". Below this, it says "to perform NotificationProcess with properties:".
 - Table:** A table with columns "Role", "Restriction", "Complement", and "Value(s)". It is currently empty.
 - When Actor performs:** A section with a dropdown menu showing "PickUpPilot".
 - Properties List:** A list of properties is shown, including "countryOfHospital_In", "deliveryHospitalLocation_In", "destination_In", "hasContext", "hasObjectContext", "hospitalLocation_In", "injuryType_In", "injuryType_Out", "performedOn", "person_In", "person_Out", "pickUpLocation_In", and "Additional Properties". The "countryOfHospital_In" property is currently selected.
 - Buttons:** "Add" and "with properties" buttons are visible.
- Policy Changes:** A section at the bottom right with "Commit" and "Refresh" buttons.

COSAR-TS Web Interactive Demo



http://ontology.ihmc.us/CoSAR-TS/Demos/CoSAR-TS_Demo_Concept.htm

I-Plan Tool – CoSAR-TS Search & Rescue

Coalition Search and Rescue Coordinator

File New Tools Help Test

Issues

Description	Annotations	Priority	Action

Activities

Description	Annotations	Priority	Action
example		Normal	No Action
setup-initial-state		Normal	No Action
sea-rescue pilot-A red-sea burns		Normal	No Action
pick-up-and-transport pilot-A red-sea "Gahwad EI" burns		Normal	No Action
us-army-helicopter-service gahwad-el-lat-long red-sea Arabello to burns pi		Normal	No Action
treat-injury pilot-A "Gahwad EI" burns		Normal	No Action
arabello-hospital-service pilot-A burns to string-0		Normal	No Action

State

Pattern	Value
longitude Southampton	-1.404
maxSpeed USS_Michigan	70Km/h
weapon USS_California	"10 loaded torpedo"
weapons WMD	Biological
maxSpeed USS_California	52Km/h
maxSpeed GaoMarineHelicopter	120Km/h
type red-sea	location
type Heathrow	airport

Annotations

Key	Value

Coalition Search and Rescue C...

File

Planning statistics:
 Steps taken = 15
 Alternatives posted = 1
 Alternatives picked = 0
 Alternatives remaining = 1
 Number of nodes = 7
 Longest node-end path length = 13

Coalition Search and Rescue C...

File

```

Executing begin_of Item[Activity[arabello-hospital-s
Condition: p=v[(type pilot-A)=person]
Condition: p=v[(type burns)=injury]
Executing end_of Item[Activity[arabello-hospital-ser
Effect: p=v[(type string-0)=string]
Executing end_of Item[Activity[treat-injury pilot-A "Ga
Executing end_of Item[Activity[sea-rescue pilot-A re
Executing end_of Item[Activity[example]]
    
```

No problems found.

Final world state:
 (latitude AIA) = 55.944
 (longitude AIA) = -3.186
 (type Arabello) = country
 (latitude ArabelloCoastguardCutter) = 23.45
 (longitude ArabelloCoastguardCutter) = 23.55

CoSAR I-X Process Panels
Based on I-X Technology

I-K-C – CoSAR-TS Search & Rescue

Coalition Search and Rescue Coordinator

File New Tools Help Test

Issues

Description	Annotations	Priority	Action

Activities

Description	Annotations	Priority	Action
example		Normal	No Action
setup-initial-state		Normal	No Action
sea-rescue pilot-A red-sea burns		Normal	No Action
pick-up-and-transport pilot-A red-sea "Gahwad El" burns		Normal	No Action
gao-marine-helicopter-service gahwad-el-lat-long red-sea Arabello to burn		Normal	No Action
treat-injury pilot-A "Gahwad El" burns		Normal	No Action
arabello-hospital-service pilot-A burns to string-0		Normal	No Action

State

Pattern	Value
longitude Southampton	-1.404
maxSpeed USS_Michigan	70Km/h
weapon USS_California	"10 loaded torpedo"
weapons WMD	Biological
maxSpeed USS_California	52Km/h
maxSpeed GaoMarineHelicopter	120Km/h
type red-sea	location
type Heathrow	airport

Annotations

Key	Value

Coalition Search and Rescue C...

File

Planning statistics:
Steps taken = 13
Alternatives posted = 1
Alternatives picked = 1
Alternatives remaining = 1
Number of nodes = 7
Longest node-end path length = 13

Coalition Search and Rescue C...

File

Executing begin_of Item[Activity(sea-rescue pilot-A
Executing begin_of Item[Activity(pick-up-and-transp
Condition: p=v{(type "Gahwad El")=hospital]
Condition: p=v{(country "Gahwad El")=Arabello]
Condition: p=v{(location "Gahwad El")=gahwad-el-
Condition: p=v{(type red-sea)=location]
Executing begin_of Item[Activity(gao-marine-helicop
Condition: p=v{(type gahwad-el-lat-long)=location]
Condition: p=v{(type red-sea)=location]
Condition: p=v{(type Arabello)=country]
Executing end_of Item[Activity(gao-marine-helicopte
Effect: p=v{(type burns)=injury]
Effect: p=v{(type pilot-A)=person]
Executing end_of Item[Activity(pick-up-and-transpor
Executing begin_of Item[Activity(treat-injury pilot-A "C
Condition: p=v{(country "Gahwad El")=Arabello]
Executing begin_of Item[Activity(arabello-hospital-s

CoSAR I-X Process Panels
Based on I-X Technology



I-K-C – CoSAR-TS Search & Rescue

Coalition Search and Rescue Coordinator

File New Tools Help Test

Issues

ID	Description	Annotations	Priority	Action

Activities

ID	Description	Annotations	Priority	Action
	example		Normal	No Action
	setup-initial-state		Normal	No Action
	sea-rescue pilot-A red-sea burns		Normal	No Action
	pick-up-and-transport pilot-A red-sea "Gahwad E" burns		Normal	No Action
	us-marine-helicopter-service gahwad-el-lat-long red-sea Arabello to burns		Normal	No Action
	treat-injury pilot-A "Gahwad E" burns		Normal	No Action
	arabello-hospital-service pilot-A burns to string-0		Normal	No Action

State

Pattern	Value
longitude Southampton	-1.404
maxSpeed USS_Michigan	70Km/h
weapon USS_California	"10 loaded torpedo"
weapons WMD	Biological
maxSpeed USS_California	52Km/h
maxSpeed GaoMarineHelicopter	120Km/h
type red-sea	location
type Heathrow	airport

Annotations

Key	Value

Coalition Search and Rescue C...

File

Planning statistics:
Steps taken = 13
Alternatives posted = 0

Message

No plan was found

OK

Plan Replan Check Plan

Coalition Search and Rescue C...

File

No problems found.

Final world state:
(latitude AIA) = 55.944
(longitude AIA) = -3.186
(type Arabello) = country
(latitude ArabelloCoastguardCutter) = 23.45
(longitude ArabelloCoastguardCutter) = 38.55
(maxSpeed ArabelloCoastguardCutter) = 35Km/h
(type ArabelloCoastguardCutter) = cutter
(altitude Bandar_Airport) = 10
(latitude Bandar_Airport) = 19.25
(longitude Bandar_Airport) = 37.0
(type Bandar_Airport) = airport
(latitude Birmingham) = 52.45
(longitude Birmingham) = -1.75
(latitude Edinburgh Belfast Dublin London) = 55.2

Cancel

CoSAR I-X Process Panels
Based on I-X Technology



Some Features of the Approach

- 1. Planning using OWL-S Service Model IOPE Core**
- 2. Can easily extend to accommodate richer temporal, resource and performer constraints**
- 3. Policy analysis feedback during planning**
- 4. Should separate plan-time model from run-time enactment environment**
- 5. Single shot plan service with re-plan facility or richer “mixed-initiative” multiple-options mode**
- 6. Exploring links to a graphical web service editor**
- 7. Exploring seeking web service description information at planning or enactment time**
- 8. Can run as separate services or as embedded tools**



Continuing Issues

- 1. OWL-S input beyond primitives**
- 2. OWL-S output espec. wrt Preconditions/Effects**
- 3. Two way I-X \leftrightarrow KAoS rich interchange**
- 4. Widen scope of KAoS policy analysis**
- 5. Discrete vs. continuous analysis of workflows**
- 6. Mixed-initiative planning support, GUI**
- 7. Multiple option exploration, GUI**
- 8. Current service environment vs enactment model**
- 9. When to stop planning – how far to commit**
- 10. LOTS of planning power when we need it**



OWL-S Semantics Issues

- **OWL-S doesn't yet define a way to express preconditions and effects**
 - *The intention is to fix this in SWSL*
- **It is awkward to express the data-flow in a composite process that invokes the same service more than once**
 - *The intention is to fix this in OWL-S 1.1*
- **There are partial orders of service invocations and temporal constraints that the OWL-S control structures cannot express**
 - *The intention is to fix this in SWSL*

OWL-S Workflow Issues

- ***Current Process Model ontology is more suited to the purpose of defining internal structure of a single service***
- ***Need to attach Profile restrictions to a step of the workflow; used to find a Matchmaker-registered service that meets requirements during enactment***
- ***Composite processes are made up of non-unique instances of processes. We have not been able to find a way to add additional information to a particular step, for instance:***
 - ***Profile restrictions***
 - ***Policy analysis results***

OWL-S Deployment Issues

- *There doesn't seem to be an authoritative document that precisely defines the OWL-S semantics. Many questions aren't answered by the Technical Overview or by the OWL definitions of the OWL-S ontologies*
- *RDF is awkward to use and difficult to read, and OWL-S doesn't yet have an agreed alternative "surface syntax"*
- *There is currently no OWL-S editor*
- *Doing simple things with OWL-S requires lots of software (e.g. Jena2 and all that it requires or the OWL-S API which requires Jena2 and more)*



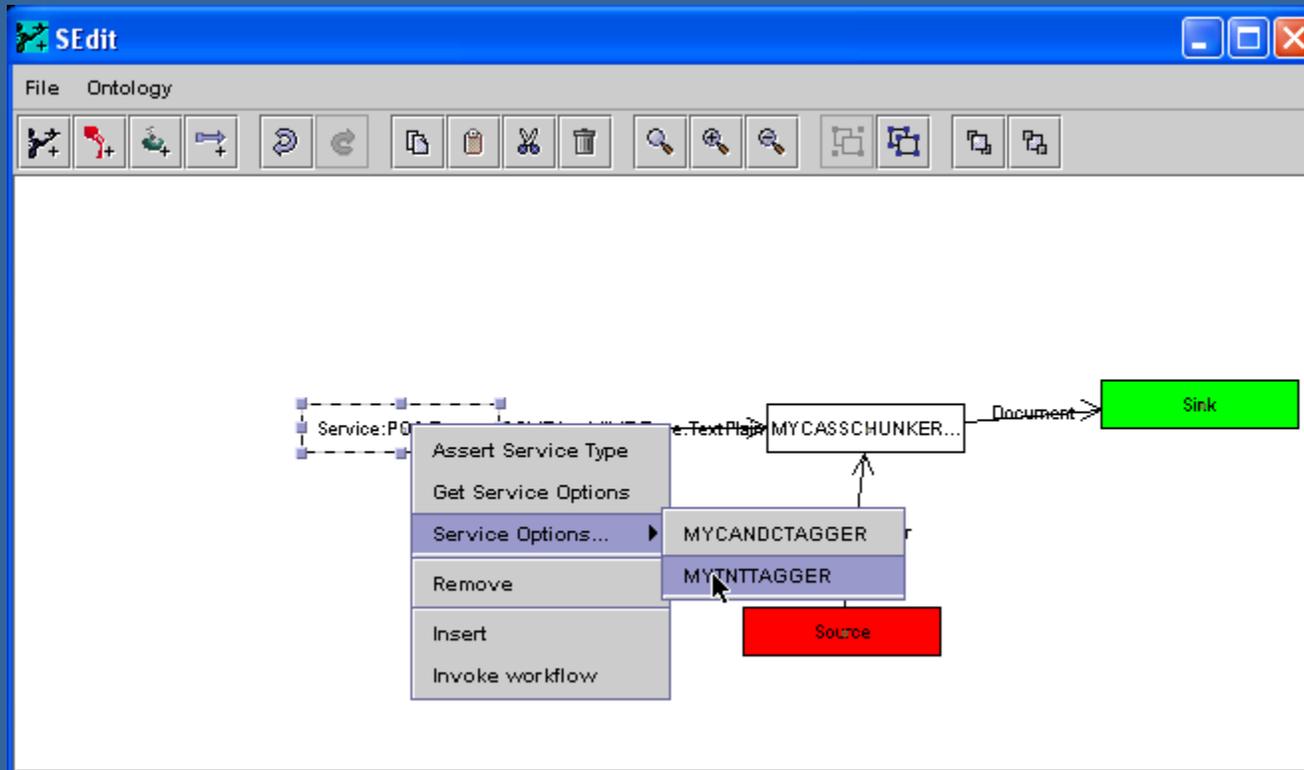
Continuing Work

- ***Complete integration of I-Plan Planner with KAoS policy analysis services***
 - *Also allow the use of WSDL workflow analyses*
- ***Java Web Start version of KPAT to obviate the need for prior installation on user's machine***
- ***Generic KAoS enforcer for OWL-S***
- ***Mixed-initiative planning, integration with AKT project graphical composition tool***
- ***Web-based demonstration integrating I-Plan, I-P², CMU Matchmaker, KAoS and servlets simulating services***



Semantic Web Service Workflow Composition Editor

AKT Project – Stephen Potter, AIAI



AIAI Summary Report

- **2003 Goal**

- Link I-X coordination and task support with KAoS agent, domain and policy services
- Demonstrate in a Search & Rescue scenario in TTCP Binni C2 Domain
- To be shown as AAI-2004 Intelligent Systems Demonstrator
<http://www.aiai.ed.ac.uk/project/cosar-ts/demo/isd/>

- **2004 Goal**

- Create a web service composition tool based on AI planning technology that can account for execution policy issues, requirements and constraints

- **Release Plans**

- Currently I-X version 3.3 and CoSAR demonstration are available via web for research use
- Open source I-X version 4.0 for research and US government use planned for September 2004. Tool based on this put on SemWebCentral soon after.

- **Plans to end of Project**

- Do our best to package the results (effort mostly used to date)
- Do our best to continue to participate in SWSL and W3C SWS-IG



IHMC Summary Report

- **2003 Goal**

- Provide KAoS domain and policy services to I-X
- Different from and complementary to CMU Matchmaker Policies and OWL-S security extensions
- Develop policies and enforcers for Search & Rescue scenario in TTCP Binni C2 Domain

- **2004 Goal**

- Provide policy analysis capability for OWL-S composite processes (next: WMSO)

- **Release Plans**

- Web hosting of KAoS and CoSAR demonstrations for research use
- Distribution of KAoS on SemWebCentral for research and US government use planned for October 2004

- **Plans to end of Project**

- Enrich policy analyses of OWL-S specified workflow
- Finish the live Web demonstration of integrated technology and CoSAR scenario by August 2004
- Collaborate with CMU on Matchmaker improvements and usage
- Develop generic policy enforcer for OWL-S services



Further Information

- <http://www.aiai.ed.ac.uk/project/cosar-ts/>
- <http://ontology.ihmc.us/>
- <http://i-x.info>

