Policy-based Access Control for Task Computing Using Rei

Ryusuke Masuoka[1], Mohinder Chopra[1],
Yannis Labrou[1], Zhexuan Song[1],
Wei-lun Chen[1], Lalana Kagal[2], Tim Finin[2]

Fujitsu Labs of America[1] and UMBC[2]

May 10th, 2005

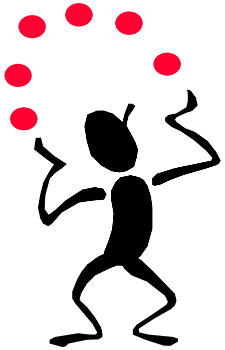
Outline

- Task Computing
- Rei
- Policy-based Access Control
- Summary

Task Computing (TC)

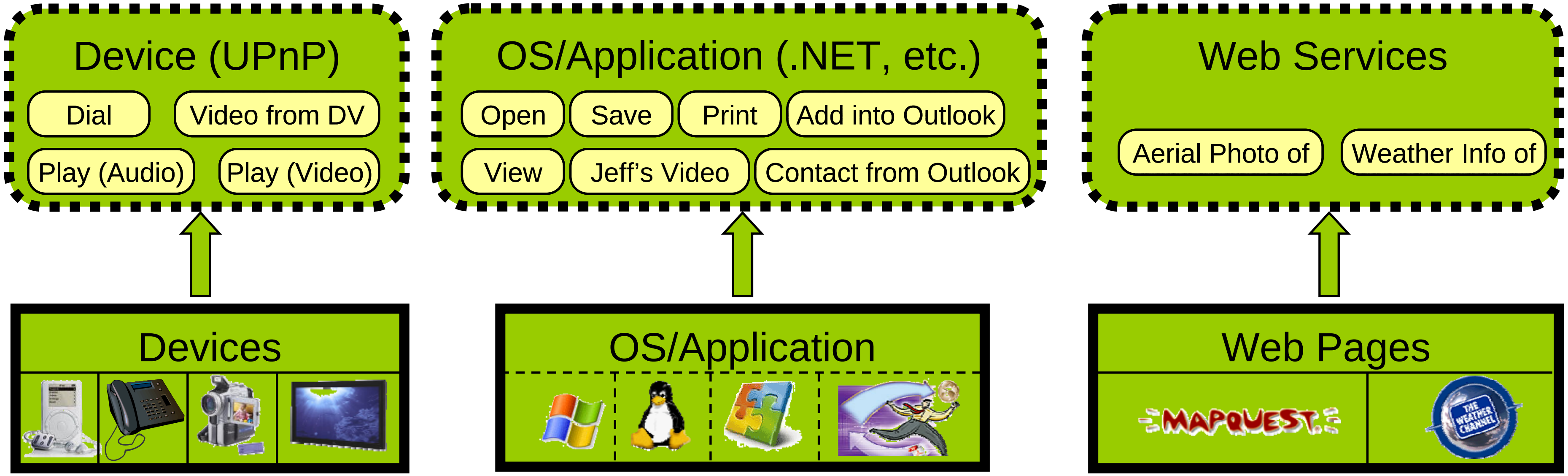
- Lets end-users accomplish complex tasks on the fly
  - With an open, dynamic, and distributed "universe of network-accessible resources" in ubiquitous computing environments and on the Internet
- Developed jointly Fujitsu Labs and MINDSwap of Univ. of Maryland and being productized by Fujitsu
- Based on Semantic Web technologies – OWL, OWL-S
- Many kinds of TC Clients
  - STEER-XT (Full client), Voice, Graphical, Gesture, etc.
  - Internationalized with eight languages
  - To accommodate many modalities with help of semantics
- Semantic Services – Building blocks for user's task
  - 50+ kinds of local, pervasive, and remote services implemented
  - Use of third-party Web Services from Amazon, Google, Yahoo
- TC Demo at the DevDay on Saturday (5/14)

How TC Works

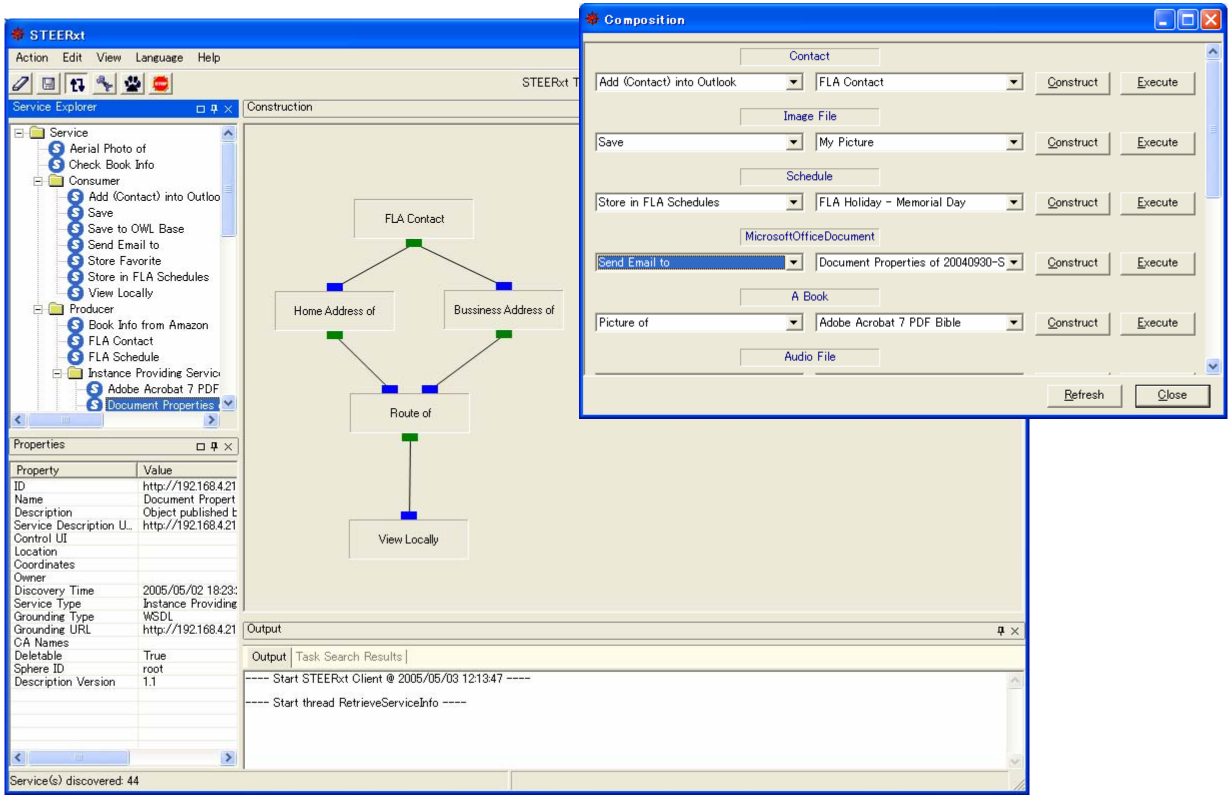


Play Jeff's Video
Dial Contact from Outlook
Weather Info of FLA, CP
…

- Service descriptions in OWL-S
- Found through discovery mechanisms
  - UPnP
  - Local file system
  - WS-based discovery
- Lets the end-users manipulate and execute tasks as service compositions



TC Clients



Property	Value
ID	http://192.168.4.21
Name	Document Propert
Description	Object published b
Service Description U...	http://192.168.4.21
Control UI	
Location	
Coordinates	
Owner	
Discovery Time	2005/05/02 18:23:
Service Type	Instance Providing
Grounding Type	WSDL
Grounding URL	http://192.168.4.21
CA Names	
Deletable	True
Sphere ID	root
Description Version	1.1

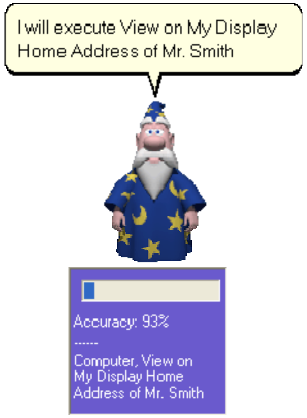
```
---- Start STEERxt Client @ 2005/05/03 12:13:47 ----

---- Start thread RetrieveServiceInfo ----
```

Service(s) discovered: 44

	Contact		
Add (Contact) into Outlook	FLA Contact	Construct	Execute
	Image File		
Save	My Picture	Construct	Execute
	Schedule		
Store in FLA Schedules	FLA Holiday – Memorial Day	Construct	Execute
	MicrosoftOfficeDocument		
Send Email to	Document Properties of 20040930-S	Construct	Execute
	A Book		
Picture of	Adobe Acrobat 7 PDF Bible	Construct	Execute
	Audio File		

STEER-XT Client

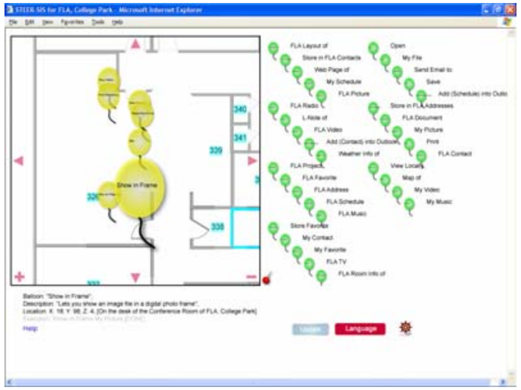


Accuracy: 93%

Computer, View on My Display Home Address of Mr. Smith

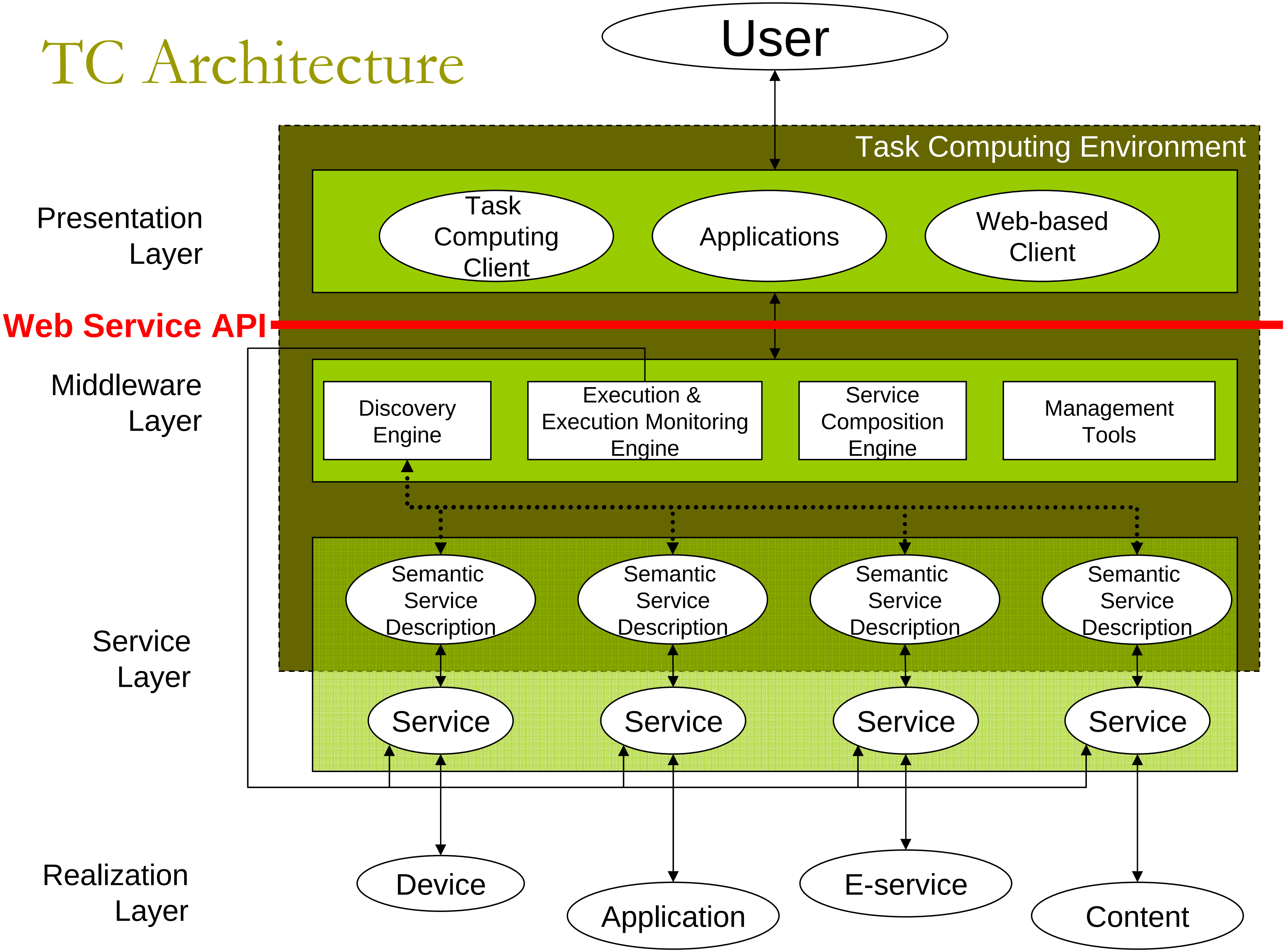
VoiceSTEER

Tasklet TCC



Graphical UI

TC Architecture



Policies for Task Computing

- We define policies as norms of behavior
  - Describe *ideal behavior* (security, privacy, management, etc.)
  - Positive and negative authorizations & obligations
  - Policies are defined over ‘classes’ of entities and actions defined by constraints on attributes of the action, actor, target, and the general context – not just on identities
- Useful for Task Computing
  - Presence of large number of resources
    - Policies provide high-level control of entities in the environment
  - Resources and clients not predetermined
    - Policies are based on attributes and not identities
  - Constantly evolving
    - Policies allow the behavior of entities to be dynamically modified

Rei Policy Spec Language

- A declarative policy specification language
  - Rules over permitted and obligated domain actions
- Represented in OWL-Lite + logical variables
  - Rule-based approach
  - Increased expressivity as it can express relations like role-value maps that are not currently possible in RDF or OWL
  - OWL extension is subset of SWRL
- Reasons over domain dependent information in RDF and OWL
  - F-OWL reasoner

Rei Policy Spec Language

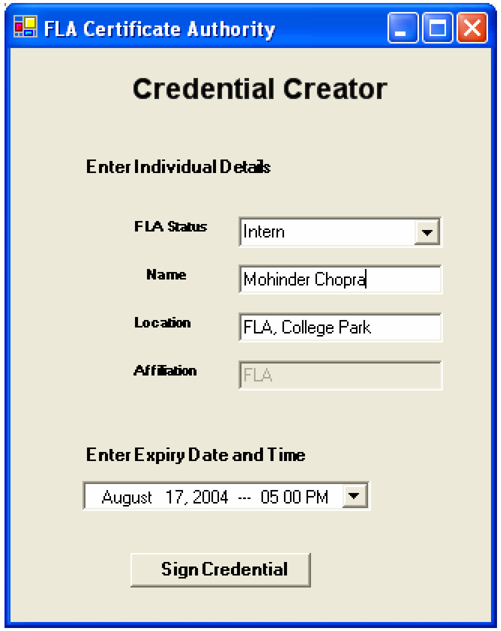
- Policy Engine
  - Answers queries about policies and domain knowledge
  - Example : Can X perform action Y on resource Z ? What are the current obligations of X ? What actions can X perform on resource/service Z ? ....
- Analysis tools
  - Verifying whether the given set of test cases is satisfied
  - Performing what-if analysis for testing the impact of changes to policies or domain knowledge
- Interface
  - Java API
  - Simple GUI in Protégé
  - GUI in Eclipse (under construction)

Motivations and Design Goals

- TC apparently needs access control
  - It made it very easy to use dynamically found resources
- Very dynamic and open ubiquitous environment requires:
  - Rule-based approach, not identity- nor role-based access control
- Design goals
  - Minimally obtrusive for users
    - Without spoiling TC user experience
  - Enable both centralized/distributed solutions
  - Allow multiple certificate authorities
  - Secure dynamic delegation

Check-in

- At the reception:



FLA Certificate Authority

Credential Creator

Enter Individual Details

FLA Status: Intern

Name: Mohinder Chopra

Location: FLA, College Park

Affiliation: FLA

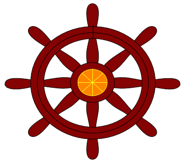
Enter Expiry Date and Time

August 17, 2004 -- 05:00 PM

<!– Facts about the Person (Credential) -->

```
<rdf:RDF …>
  <rdfs:label lang=en>Mohinder Chorpa</rdfs:label>
  <flaonto:Name  …>Mohinder Chorpa</flaonto:Name>
  <flaonto:Expiry …>2004-08-23T23:05:28Z</flaonto:Expiry>
  <flaonto:Status …>&flaonto;FLACPVisitor</flaonto:Status>
  <flaonto:Affiliation …>UMBC</flaonto:Affiliation>
  <Signature xmlns="h           w3.org/2000/09/xmldsig#">
    <SignedInfo>
      …
    </SignedInfo>
    <SignatureValue>ZrbEVA7JWWGNbpqc…Jo6dDw=
    </SignatureValue>
  </Signature>
</rdf:RDF>
```

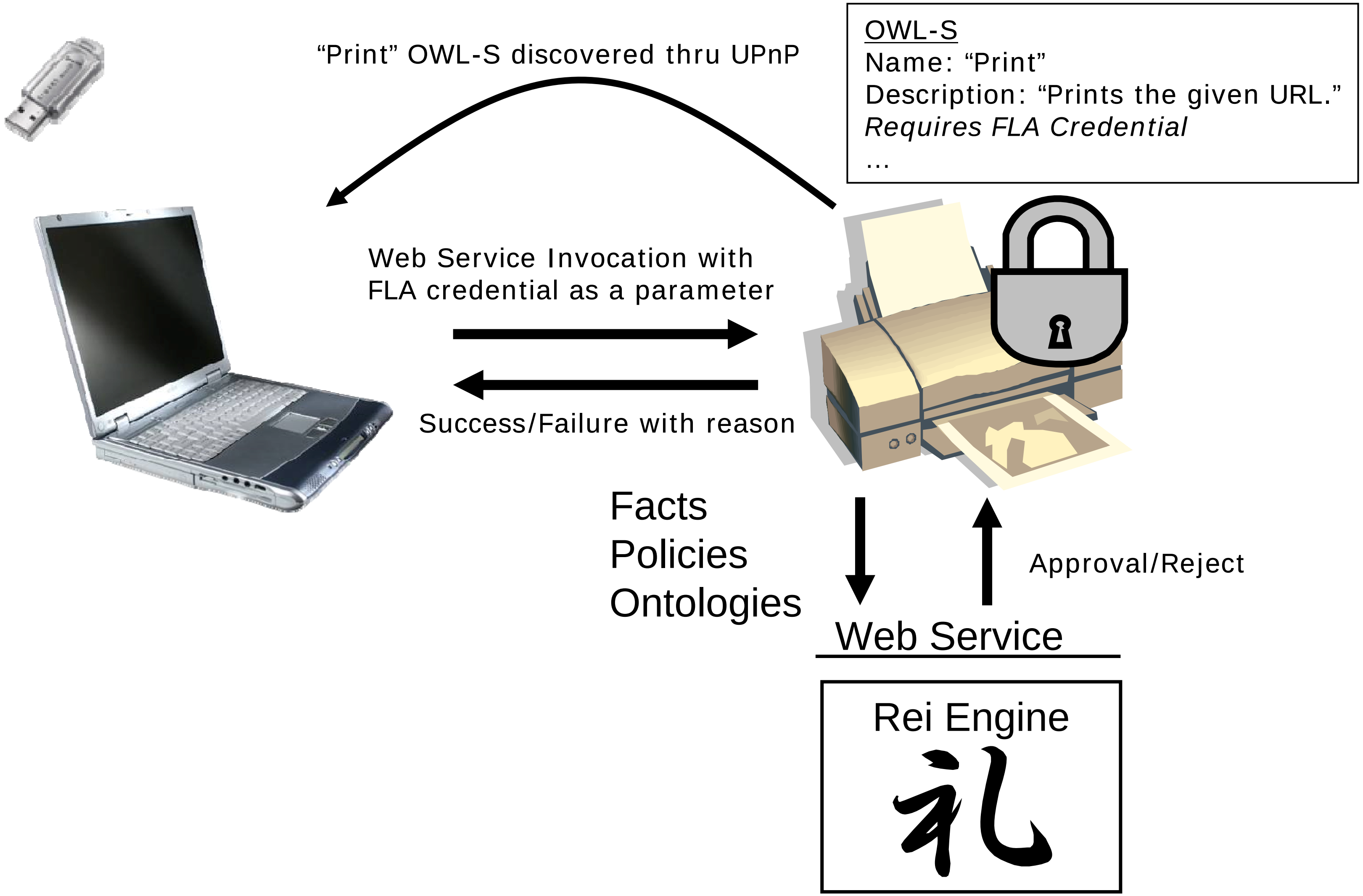
Digital Signaure



STEER + Credential

STEER-Stick

Discovery, Invocation, Authentication



"Print" OWL-S discovered thru UPnP

OWL-S
Name: "Print"
Description: "Prints the given URL."
Requires FLA Credential
…

Web Service Invocation with
FLA credential as a parameter

Success/Failure with reason

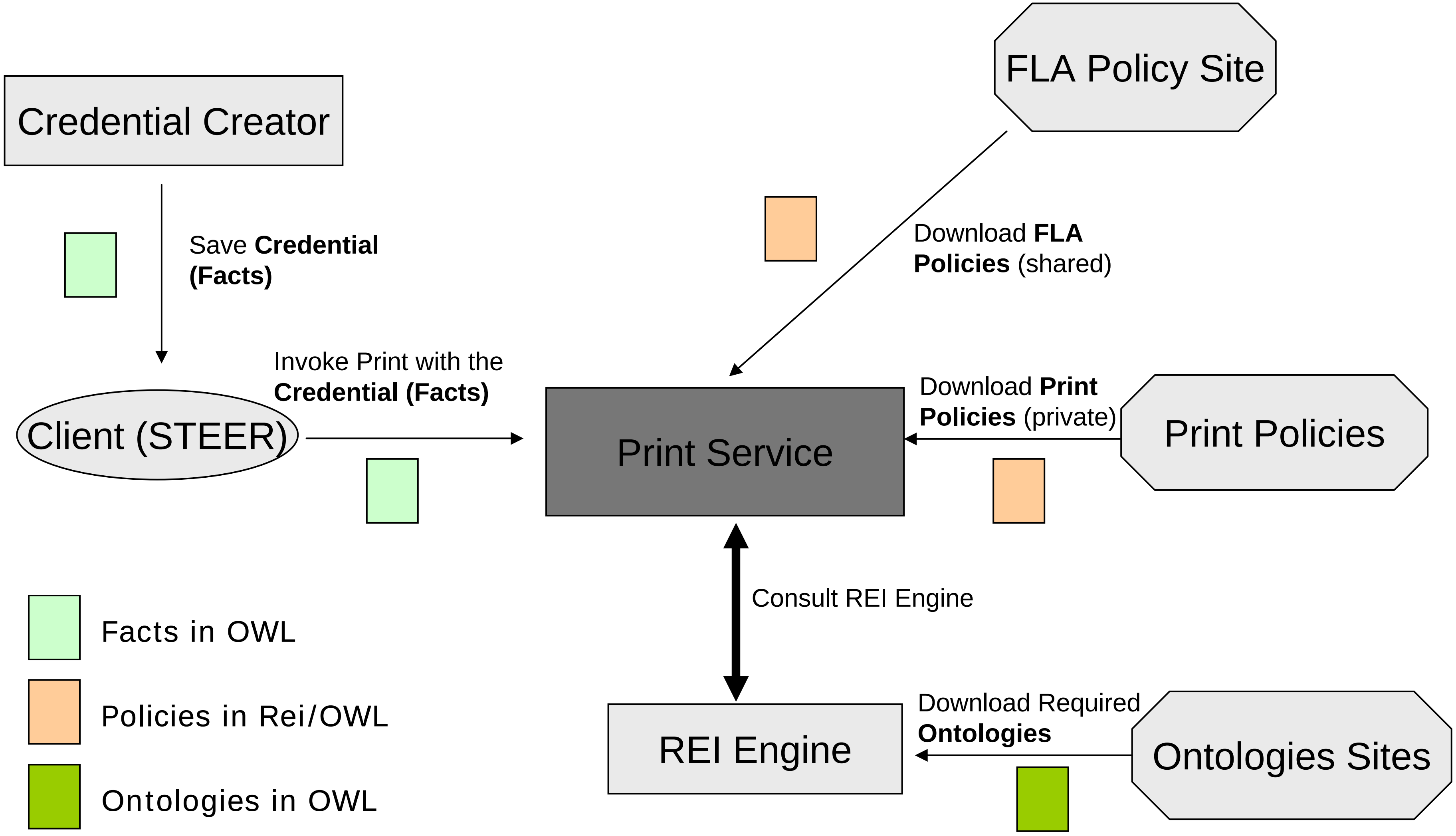
Facts
Policies
Ontologies

Approval/Reject

Web Service

Rei Engine
礼

Mix and Match at the Service



Credential Creator

Save **Credential (Facts)**

Client (STEER)

Invoke Print with the **Credential (Facts)**

FLA Policy Site

Download **FLA Policies** (shared)

Print Service

Download **Print Policies** (private)

Print Policies

Consult REI Engine

REI Engine

Download Required **Ontologies**

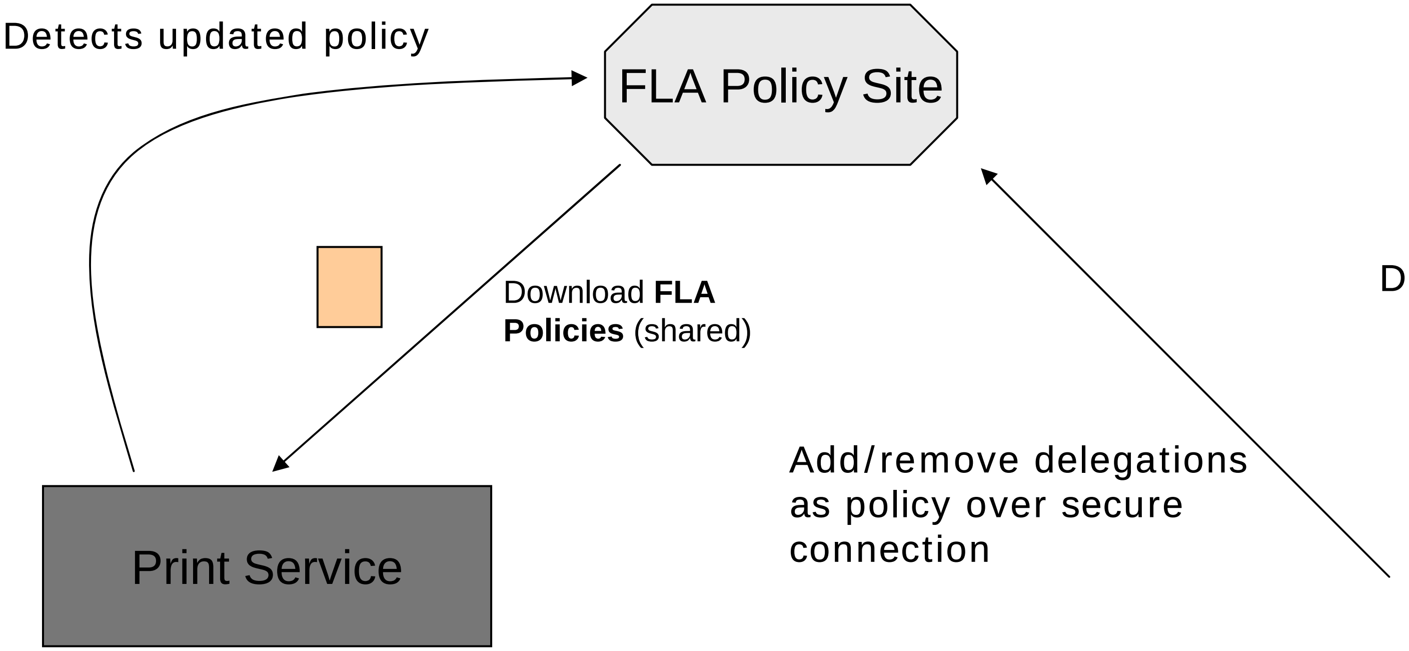
Ontologies Sites

Facts in OWL

Policies in Rei/OWL

Ontologies in OWL

Delegation



Detects updated policy

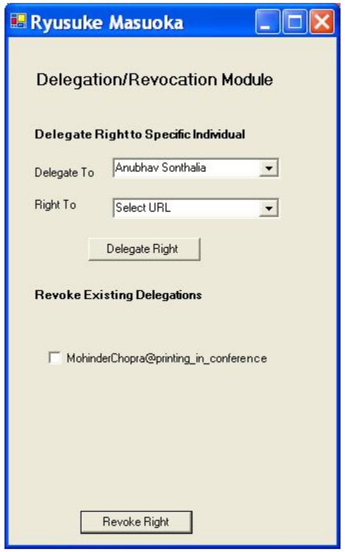
FLA Policy Site

Download **FLA Policies** (shared)

Print Service

Add/remove delegations as policy over secure connection

Delegation Manager



Ryusuke Masuoka

Delegation/Revocation Module

Delegate Right to Specific Individual

Delegate To: Anubhav Sonthalia

Right To: Select URL

Revoke Existing Delegations

□ MohinderChopra@printing_in_conference

Facts, Policies, Ontologies, Queries

- Facts:
  - Mohinder is a FLACP Visitor
- Policies (Private)
  - An employee can print
- Policies (Shared)
  - A senior employee can delegate the right to print (delegation)
  - Ryu delegates Mohinder the right to print
- Ontology
  - Ryu is a research fellow
  - A research fellow is a senior employee
- Queries
  - Can Mohinder print?

```
<!– Fact from Task Computing client -->
<rdf:RDF …>
  <rdfs:label lang=en>Mohinder Chopra</rdfs:label>
  <flaonto:Name  …>Mohinder Chopra</flaonto:Name>
  <flaonto:Expiry …>2004-08-23T23:05:28Z</flaonto:Expiry>
  <flaonto:Status …>&flaonto;FLACPVisitor</flaonto:Status>
  <flaonto:Affiliation …>UMBC</flaonto:Affiliation>
  <Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
    <SignedInfo>
      …
    </SignedInfo>
    <SignatureValue>ZrbEVA7JWWGNbpqc…Jo6dDw=</SignatureValue>
  </Signature>
</rdf:RDF>

<!– Printer Private Policy -->
…
<deontic:Permission rdf:about="&flapolicy;right_to_be_printed_on“
  policy:desc="All senior employees have the right to print">
  <deontic:actor rdf:resource="&flapolicy;var1"/>
  <deontic:action rdf:resource="&flapolicy;printing_in_conference"/>
  <deontic:constraint rdf:resource="&flapolicy;preOrSenior"/>
</deontic:Permission>
…

<!– Delegation Inserted (and Removed) in Shared Policy-->
<action:Delegation
  rdf:ID=“Delegation2004-08-23T19:32:19ZRyusukeMasuoka">
  <action:sender rdf:resource="&inst;RyusukeMasuoka"/>
  <action:receiver rdf:resource="&inst;MohinderChorpa"/>
  <action:content>
    <deontic:Permission>
      <deontic:action rdf:resource="&inst;ASeniorEmployeePrintingAction"/>
    </deontic:Permission>
  </action:content>
</action:Delegation>
```

Other Scenarios

- A senior employee gives to a class of users, the right to use a class of resources.
  - User class: Ex. all visitors from UMBC on Jan 31st
  - Resource class: Ex. all devices in the conference room
- Service policy check by client prior to invocation
  - Service policy in the OWL-S file
- Multiple CA's
  - Multiple CA's listed in the OWL-S file
  - Client have multiple credentials

Design Goals Revisited

- Minimally obtrusive for users
- Enable both centralized/distributed solutions
- Allow multiple certificate authorities
- Secure dynamic delegation

Summary

- Unobtrusive and flexible access control for Task Computing is implemented using Rei policy engine
- Future work
  - Discovery security
  - Service authentication by client
    - Service facts in the OWL-S file
  - Explanation and negotiation