

Task Computing: Pervasive Portals for Web Services

Bijan Parsia

Ryusuke Masuoka

Service Oriented Architecture (SOA)

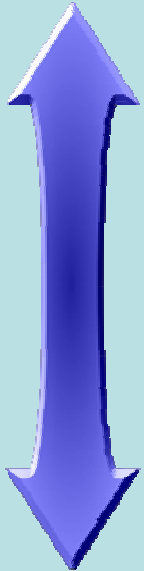
- New paradigm for distributed computing?
- Extreme message orientation
 - Was: messages more like method calls
 - Moving toward: messages as *documents*
 - Push version of the Web?
- Very loose coupling
 - Vertically: Language, OS, implementation independence
 - Horizontally: composable components

SOA on a Web Scale

- Web Pages and Web Services
- Ubiquitous Services
 - Web Clients everywhere
- Services abstract over agents
 - Agents offer services
 - Service orientation hides suppliers
 - Agents avail themselves of services
 - Web Browser (a “user agent”)
 - Web Crawlers
 - Dual Use!
- How to manage Services for Users?

Goal of Task Computing

- User wants to do “Tasks”



Filling the gap

- “Services” offer means
 - Web services, UPnP services etc.

Task Computing (at the technical level)

- “**Task Computing**” is computation to fill the **gap** between user’s **tasks** and the **available means**
- Three aspects:
 - Grasp the current environment
 - Find what’s available
 - Understand it
 - Connect it to user’s tasks
 - Organize the available
 - Suggest sensible possibilities
 - Manipulate the current environment
 - Enactment not the only interaction
 - Extend the available

Grasping the Environment

- User agent discovers services through UPnP
- Then gets the Semantic Service Description (SSD) of each found service through UPnP request
- Business card metaphor:
 - With a **business card** (SSD), you can put the **person** (service) in a **social context** (Web ontology)
 - Then you can interact with the person (service)
- Role & situation dependent filtering
 - Administrator vs. meeting participant



Presenting the environment

- Publish based Filtering
 - On user's devices (configured by user)
 - Remote (bookmarks, search engine)
 - Environmental (UPnP)
- Description
 - Use concepts of right granularity and “flow”
- Composition
 - Visible to the user
 - Hidden from the user

Semantic Service Compositions

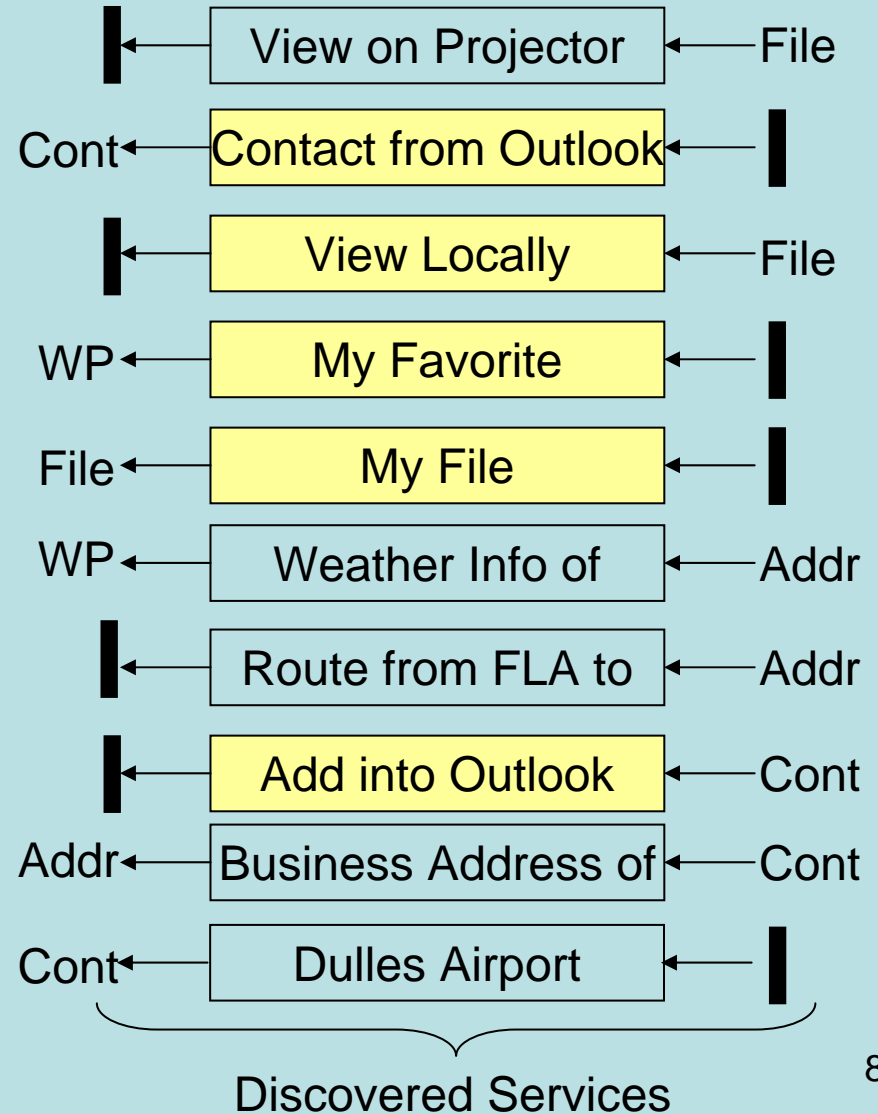
What to do with this big stack?!

How do we hook up to our tasks?

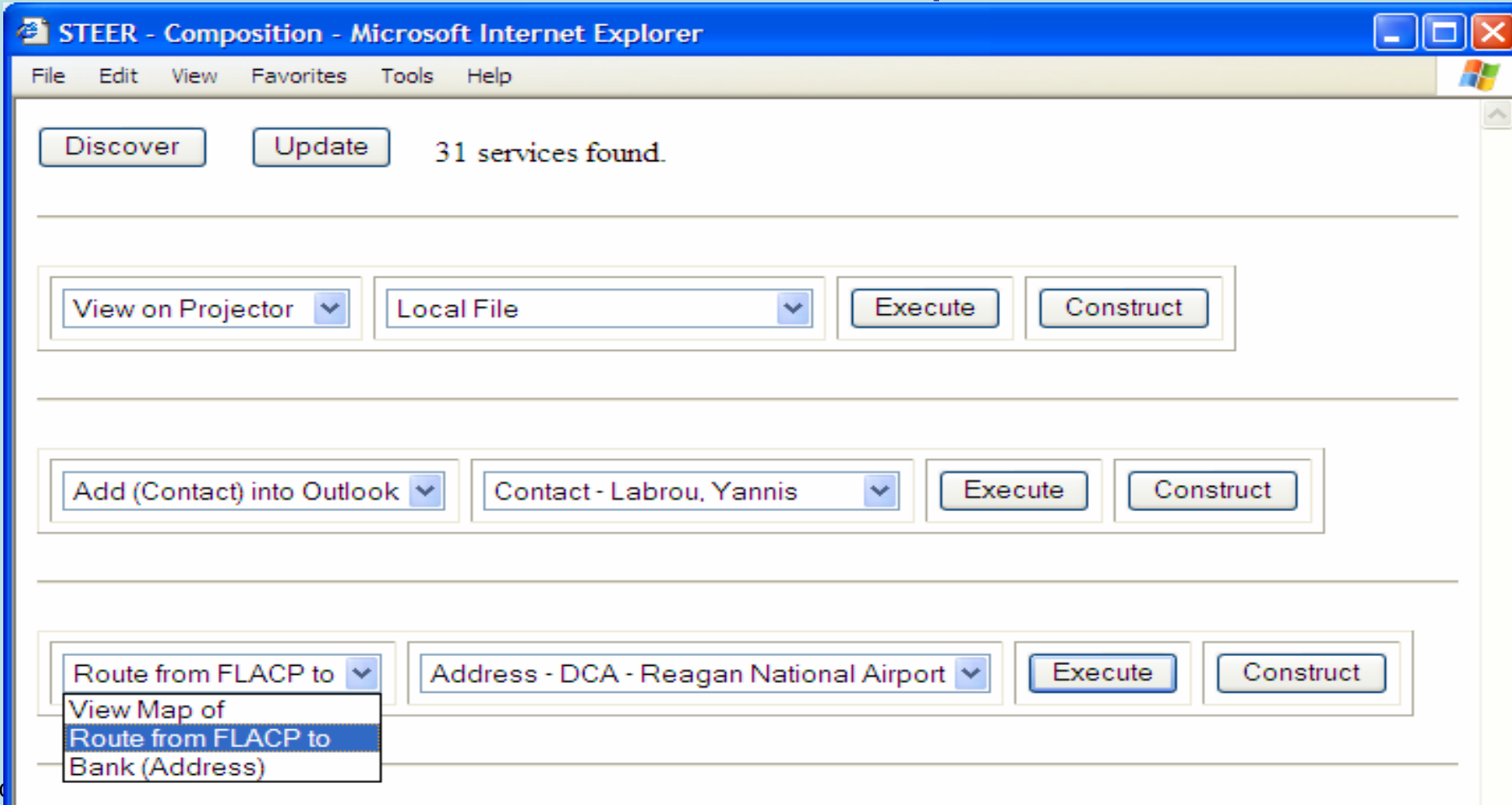
Grain:

Single services not useful
2-step compositions, useful

2-step compositions categorize the services in user-salient ways



Semantic Service Compositions



STEER - Composition - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Discover Update 31 services found.

View on Projector Local File Execute Construct

Add (Contact) into Outlook Contact - Labrou, Yannis Execute Construct

Route from FLACP to Address - DCA - Reagan National Airport Execute Construct

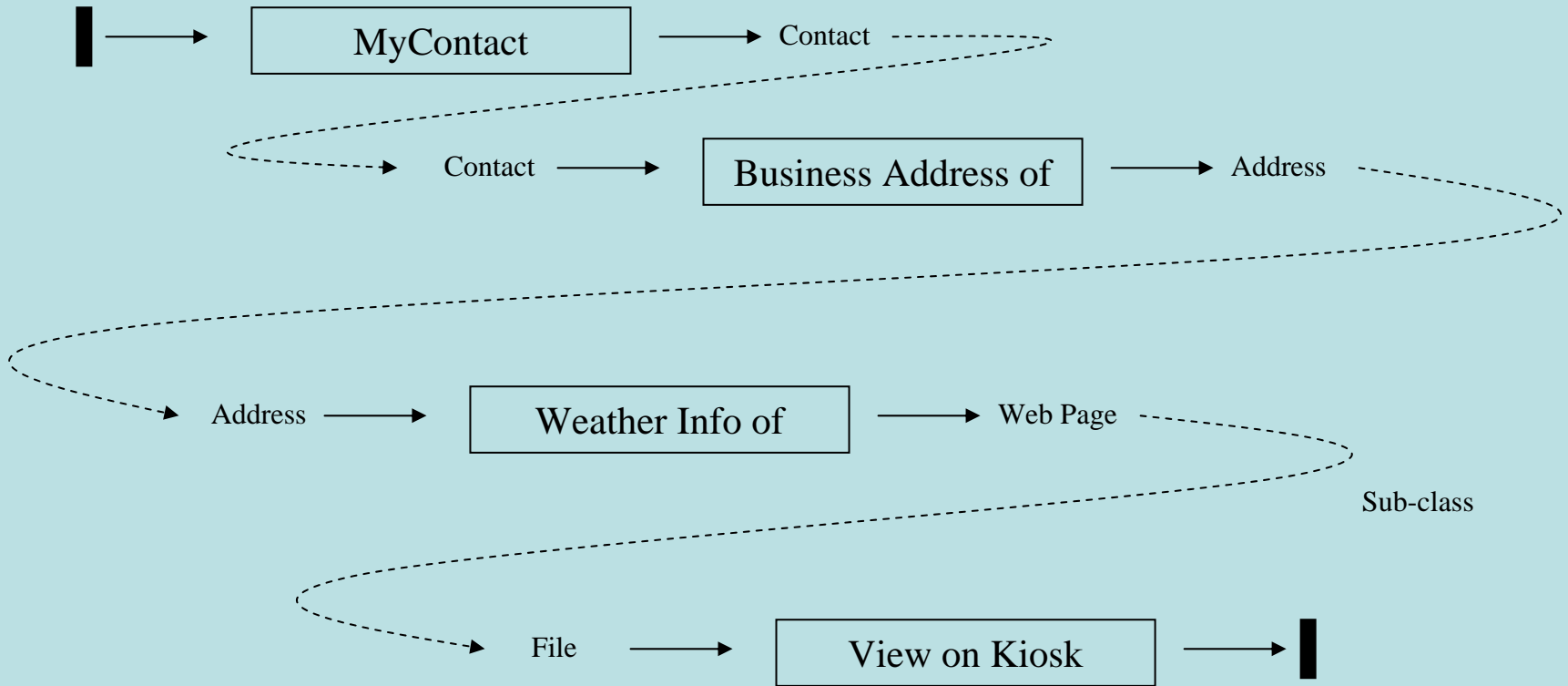
View Map of
Route from FLACP to
Bank (Address)

Ac Cont: Contact WP: Web Page 2004/12/13

Initial Compositions

Extending Compositions

View on Kiosk Weather Info of Business Address of My Contact



Pervasive Environment

- Choices of devices to interact are already good specifications of user's task
- Devices as embodiments of services
 - Devices with their physical embodiments have much simpler, straight-forward, and well-defined semantics.
- Oligopoly of device ontologies by manufactures will contribute to much less complexity.

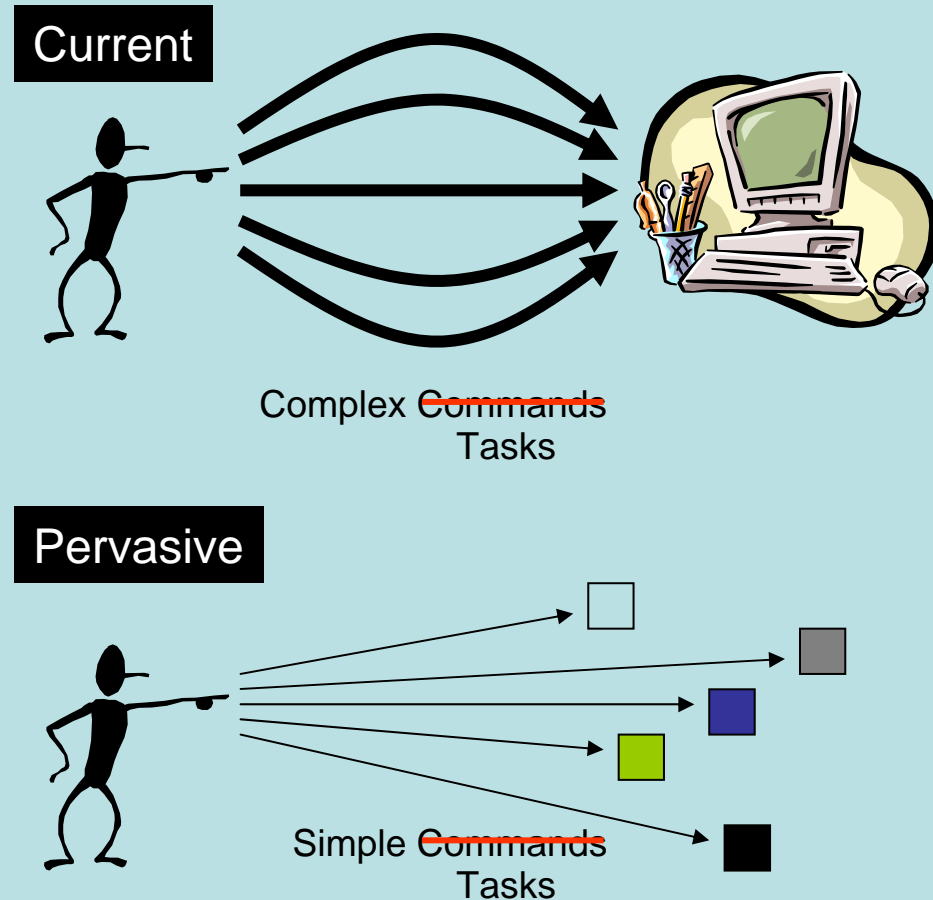


Figure adopted and modified from "Real Eye Communicator: An Eye-mediated Real World Printing Device," Y. Mitsudo and K. Mogi, Pervasive 2002

Service Granularity

- Not too big, not too small
- Granularity of word or phrase
 - The system can have smaller or larger components internally,
 - But it should be presented to the user only when its granularity is right, or
 - You should make its granularity right by consolidating or splitting up

Weather of

2004/12/13

Play

View on Projector

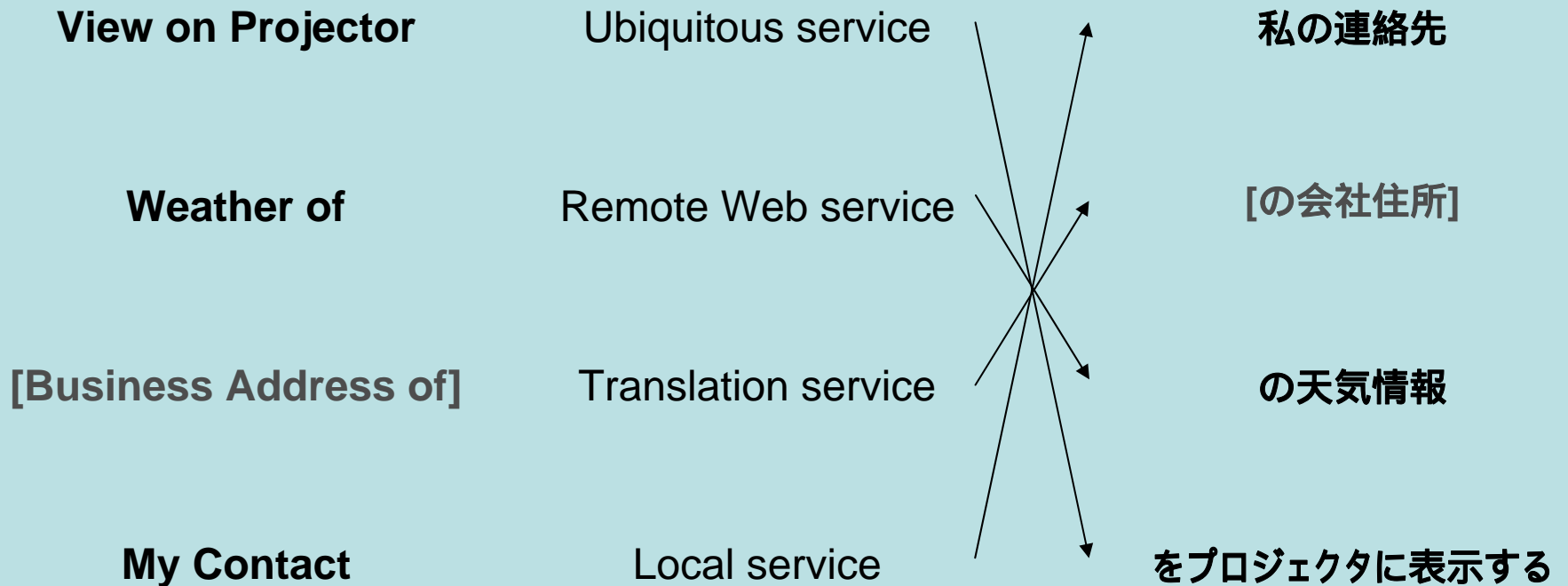
FLA, College Park

Business Address of

My Contact

Example

In Japanese



Hidden compositions

- Some steps the system composes for user
 - Translation services
 - Meaning preserving (at least, entailment)
 - Not of significance to the task
 - Enabler, not facilitator
 - Arbitrary number (depth and width)
 - Fuse appropriate translation chains
 - User sees more useful combinations
- Saved compositions
 - Describe complex workflow as a phrase
 - Future: connect with HTN planning
 - Not fully grounded compositions

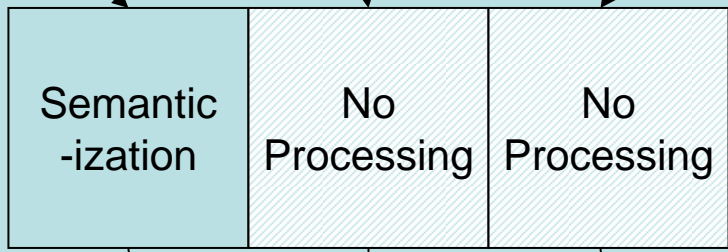
Manipulating the Environment

- The user “publishes” services from OS/App/OWL objects, and OWL-S
 - “Publish a service” =
Make the service’s OWL-S through UPnP
 - Virtual, augmented or specialized services
 - Compositions, partial specializations, information
- Numerous unique ways of publishing services
 - White hole, bank service, multimedia publisher, image generator, etc.

White Hole

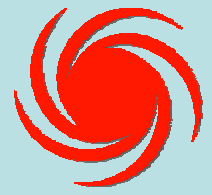
- Drag and drop acquisition
 - White hole “spews matter”
- Takes files, URIs, OWL “objects”, photos, service descriptions
 - If there is a service description, fine
 - Otherwise, generate appropriate one
 - “Object providing” services
- Service management with “PIPE”
 - Lifetime, access time, local vs. pervasive
- Other publishers work similarly

OS/Application objects Semantic objects Semantic Service Description



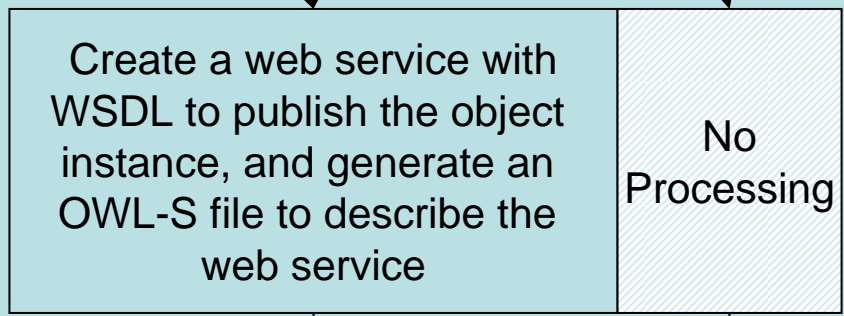
White Hole and PIPE

White Hole



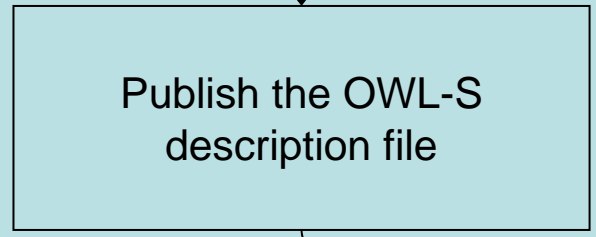
Semantic objects

Web Service API



PIPE

OWL-S service description with grounding to web service



PIPE - A Service Management Tool

Allow Task Computing Client to discover
2004/12/13

Concepts and Objects

- Like a MUD or MOO
 - Objects are given to the system by description
 - Users interact based on those descriptions
- Builtin description generation
 - Dragging a file on the white hole
 - Does something useful
 - By generating a default description
- Systematic service metaphor
 - Everything, even a contact, is a service

Pervasive Portals

- Environmental salience
 - Resident devices
 - Admin established service descriptions
 - Mobile devices and service descriptions
 - Hugely dynamic!
 - Publishing allows for more permanent effect
- Localize service *descriptions*
 - With network, all services always available
 - Manage services by managing descriptions
 - Locations are *categories*

Links, etc.

- To play:
 - <http://taskcomputing.org/>
- TC is joint research
 - Fujitsu Labs of America, College Park
 - The MIND Lab at U of Maryland, College Park
 - <http://mindswap.org>
- Many slides from Ryusuke Masuoka or derived from his slides
 - From: 4-6, 8-13, 15, 17