Ph.D. course: Advanced Distributed Systems Development with Multiagent Systems

Dott. Daniela Briola
Prof.ssa Viviana Mascardi
Prof. Rafael Bordini

UNIVERSITÀ DEGLI STUDI DI MILANO BICOCCA

DIPARTIMENTO DI INFORMATICA SISTEMISTICA E COMUNICAZIONE

JADE: Ontologies

Daniela Briola

Daniela.briola@unimib.it

Ph.D. Course: Advanced Distributed Systems Development with Multiagent Systems

OBJ5 (based on OBJ4)

- It creates a Java version of the OWL ontology (with some limitations due to the different expressivity of them)
- A predefined ontology model the Jade foreseen Concepts
  - All the new classes must be subclasses of the Jade Concepts
- For each class, an interface is created
- Subclass relationship is mapped into ”extends” between interfaces
- Multiple properties are mapped into List
- Functional properties are mapped into simple fields
  - Cardinality is not translated
  - Object type is translated using the Java interfaces as types
- A Default implementation of each OWL Class is generated, implementing the corresponding interface
- A BeanOntology class is created to manage in JADE all the concepts and properties
- A jar to be used in Jade is created

Let's try

- Download the new version of the ontology we developed together
- Have a look to the new classes (C1… Services) and their properties (in particular, the "inverse of")
- Import (again) the Ontology of Jade (do not change its name, only reimport it) and save the project
- Under Project → Configure, add the Tab of OntologyBeanGenerator
- Export the ontology as in figure (next slide)
- Download the Mas2 from the course site
- Update the BuildPath to include the ontology

phdCourse Protégé 3.5 (file:\C:\Program%20Files\Protege_3.5\phdCourse.pprj, OWL / RDF Files)

File Edit Project OWL Reasoning Code Tools BioPortal Window Collaboration Help

Metadata(Ontology1579790741.owl) | OWLClasses | Properties | Individuals | Forms | Ontology Bean Generator 5.0

Ontology Bean Generator | Java Method Mapper

status:

```
 added slots:
   Integer p3
 created file: Application.java
 added slots:
   String hasApplicationName
Collection has_failures
Collection is_installed_on
 created file: BadBehaviour.java
 added slots:
   Application is_associated_to
 created file: Failure.java
 added slots:
   Application is_associated_to
 created file: Crash.java
 added slots:
   Application is_associated_to
 created file: Machine.java
 added slots:
Collection has_IP
Collection has_installed_app
Collection has_Name
 created file: GetFailureList.java
 added slots:
   String hasServiceName
 created file: GetApplicationLlst.java
 added slots:
   String hasServiceName
 created file: Service.java
 added slots:
   String hasServiceName
```

progress

Ontology Bean Generator 5.0 (for Jade 3.5)

package name (e.g. mypackage.onto)

PhdOnto2

location excl. package (e.g. /home/chris/projects/myproject/src/)

This directory will be created if it doesn't already exist

C:\Users\VM\Desktop\phd course\onto2

ontology domain (e.g. Newspaper)

PhdOnto2

Warning!! This procedure will overwrite all existing files in the previous package!

Generate Beans

example:

```
public interface CD extends Concept {
  public void setPrice(int value);
  public int getPrice();

  public void setTitle(String value);
  public String getTitle();
}

-------------------------------------------------------
```

- [x] generate jade ontology file
- [x] generate beans
- [ ] use JADE names when specified

- ○ J2SE JavaBean compatible [JADE]
- ◉ J2SE and Java 1.1 compatible [JADE, JAD
- ○ J2ME compatible [JADE-LEAP]

powered by Acklin agent based support

Media Lab Europe
The European Research Partner of the MIT Media Lab

JADE

- Have a look to the code:
  - See how the ontology is used to send and receive messages based on the ontology
  - See how agents register services and search for them
- Create 3 agents of type AgentService (call them as you wish)
- Start the sniffer, sniffing them
- Start a AgentRequester, and check the results on the sniffer
- Try to uncomment line 67 in AgentService…