PANACEA Tutorial

MT Summit
3rd September 2013

Marc Poch, Universitat Pompeu Fabra
marc.pochriera@upf.edu

Antonio Toral, Dublin City University
atoral@computing.dcu.ie
Tutorial Objectives

• Become familiar with the web service paradigm and its advantages

• Become familiar with the PANACEA platform

• Learn how to search and use web services

• Learn how to chain web services to build more complex processing pipelines (workflows)

• Learn how to create and share your own web services
Tutorial outline

• Introduction to the PANACEA platform [30 min]
• Tour of the PANACEA webs [15 min]
• Find and try web services [45 min]
• Interoperability [10 min]

• Break [15 min]

• Chain web services: workflows [60 min]
• Be part of the platform: creating and sharing web services [30 min]
Introduction to the PANACEA platform [30 min]
Development of a platform (a space of interoperability defined by standardized protocols and common interfaces) for the easy integration of a variety of software components, tools and methodologies deployed as web services to configure a factory for the automation of acquisition, processing and annotation of language resources.
Tools to be shared remotely

Sharing workflows

Web Service wrapper

Workflow editor and engine

The Registry

Format Converters

Common Interfaces

From local tools to sharing workflows
Platform definition

The PANACEA platform is an interoperability space based on tools, guidelines, a Common Interface definition, and a “Travelling Object” specification.

Components:

- **Tools**: Taverna, BioCatalogue, myExperiment, Soaplab, storage system
- **Common Interface**: WS interoperability
- **Travelling Object**: XCES, GrAF, CoNLL, LMF
- **Documentation**
**Platform tools and portals**

**PANACEA Platform**: uses, adapts and improves myGrid tools for eScience (used in biology, social science, music, astronomy, multimedia and chemistry).

- **Share tools** (remotely run distributed tools)
- **Web Services**
- **SOAP or REST**
- **Soaplab Server**
  - JAX-WS, Axis, CXF, etc.

**PANACEA Registry**
- **registry.elda.org**

**Biocatalogue**
- **www.taverna.org.uk**
  - **Taverna**
  - **myExperiment**
  - **myExperiment**

**Call / chain Web Services**
- **Workflows**
- **Social Network**
- **Share and find workflows**
- **Share and find Web Services**
- **Share and find**
Technological option: Web Services

- Easy deployment of command line tools as WS. (Java, Python, C++, UIMA, etc.)
- Clients: Java, Python, Perl, Taverna, etc.
- No coding needed! Only metadata
- "Polling" techniques for long lasting tasks
- Web form to run the web services
- URL input / output ready
- PANACEA improvement for SOAP messaging (network usage and memory)
- PANACEA limit multiple users
Technological option: Registry and myExperiment

- User friendly GUI
- Free, open source, Continuously maintained
- Search function
- Users rating (users feedback)
- Service annotations and Language Categorization (PANACEA)
- Monitoring system (web service status and data results)
Technological option: Taverna

- User friendly GUI
- Free and open source
- Continuously maintained (v. 2.4)
- SOAP and REST web services
- Credentials manager (passwords, certificates, etc.)
- Multiple files processing ("lists")
- PANACEA Workflows, best practises, videos, etc.:  
  - Parallelization, Error recovery: "retries", Polling
- PANACEA collaboration: bug fixing and pre-release tests
Service providers

How can I share the tools on my server?

Where can I make my Web Services public?

Web Services → SOAP or REST → Soaplab Server: JAX-WS, Axis, CXF, etc.

Registry → Biocatalogue → PANACEA Registry: registry.elda.org
Users

Can I run tools without installing them?

Web Services

Where can I find Web Services?

Registry

How can I run Web Services?

Web clients:
- Java, Python, perl clients, etc.
- Web clients: Soaplab, Spinet

How can I chain WS?

-Java, Python, perl clients, etc.
- Taverna workflows

Workflows
Tour of the PANACEA webs [15 min]
The PANACEA web

• Main Page
• Link Buttons (Registry, myExperiment, tutorials and documentation)
• Tutorials Page / Videos
• Documentation
• Deliverables

http://panacea-lr.eu
The Registry

http://registry.elda.org

• The PANACEA Registry is a BioCatalogue instance (the source code has been used to deploy the registry on a server)

• Features:
  • Annotation capabilities and categorization
  • Search function
  • Automatic status check system for web services
myExperiment

• The PANACEA myExperiment is a myExperiment instance (the source code has been used to deploy it on a server)

• Features:
  • Annotation capabilities
  • Search function
  • “Services tab beta” added to PANACEA myExperiment. Users can list web services from the Registry and see in which workflows have been used.
Find and try web services [45 min]
Registry tutorial

- [http://registry.elda.org](http://registry.elda.org)
- Global view of the Registry
- Search engine
- Categorization System
- Metadata and documentation
- Monitoring System
- [http://vimeo.com/24790416](http://vimeo.com/24790416)
• Spinet web client (Soaplab web services)
• Taverna
• SOAP, WSDL. Examples (perl, python)
  – http://ws02.iula.upf.edu/panacea/examples/soaplab-clients/soaplab_clients.zip
• Soaplab command-line client
  sh $SOAPLAB_FOLDER/build/run/run-cmdline-client
  -protocol axis1
  -e http://srv-cngl.computing.dcu.ie/panacea-soaplab2-axis/services/panacea.europarl_lowercase
  -w -r input_direct_data "ASDA"
Spinet Tutorial

• Spinet is the Soaplab web client used to test and run WS deployed on a Soaplab Server.
• Every Service provider has (at least) a Soaplab Server

• the Demo...
  - Access Spinet directly from the Registry
    “Test Form Location (Spinet Web Client):”
  - Configure mandatory parameters and RUN the WS

  - 10 minutes to try to find and run some web services.

You can start from http://registry.elda.org
Twitter NLP + Registry

(3rd party tool)

- This web service is based on the Twitter NLP tool developed by Noah's ARK group.
- Noah's ARK group is Noah Smith's research group at the Language Technologies Institute, School of Computer Science, Carnegie Mellon University.

1. Search the WS in the Registry
2. Check monitoring system
3. Use web client with example data
WS advantages (for users)

• No installation

• No maintenance

• No machine resources

• Easily found on the Registry

• Usability

• Can be combined in workflows (share experiments)
Interoperability [10 min]
Interoperability

- Three levels of interoperability:
  - Communication protocols: SOAP, REST
  - Data
    Tool B does not “understand” format N!
    All tools understand the previous format
  - Parameters
Common Interface

- A Common Interface (CI) defines the mandatory parameters for every functionality, e.g. PoS tagging:

  PoS Tagger A
  - MANDATORY: input language
  - OPTIONALS: Param A

  PoS Tagger B
  - MANDATORY: input language
  - OPTIONALS: PARAM 1, PARAM 2

http://registry.elda.org
Travelling Object

- Travelling Object (TO): common data and metadata format used in PANACEA to make components interoperable

- TO1: minimal common vertical in-line format used by deployed tools (based on the XCES standard)
- TO2: stand-off format. Based on the GrAF standard, the XML serialization of LAF (ISO 24612, 2009)
- LMF: for lexical resources
- CONLL: for parsers
- Converters and adapted WS outputs
Format Converters

31 Format converters on the PANACEA Registry

- Freeling to TO. CNR
- KAF to TO. CNR
- Basic Xces to txt. CNR
- PoS tag. (Freeling treetagger) to GrAF. UPF
- Dependency parsing (Freeling) to GrAF. UPF
- Dependency CoNLL to GrAF. CNR
- Word doc to txt. UPF
- In-house mwe to LMF. CNR
- Pdf to text. UPF
- Multi. encodings converter (ISO, UTF, etc.). UPF
- Aligner to TO. DCU
- Sentence alignment to TMX. DCU
- Treetagger to MOSES (factored models). DCU
- UIMA to GrAF. ILSP

Providers are encouraged to provide converters for the formats they are interested on
3rd party tools integration

- PANACEA WS wrapper (Soaplab) and the CI make it easy for WS Providers to integrate 3rd party tools.
- ILSP tools are **UIMA** tools
- **Freeling**
- **Treetagger**
- **Twitter NLP**
- **MALT Parser**
- **DeSR**
- **MOSES, GIZA++, other aligners**
- **DELiC4MT, MT evaluation**
- **Berkeley** tagger, parser, aligner

UIMA
UPC
University of Stuttgart
Carnegie Mellon University
Uppsala University
Università di Pisa
Edinburgh, etc.
DCU
Berkeley University
Chain web services: workflows [60 min]
Workflows

• Once we can run WS...
• …it’s time to chain them

• Workflows are process chains that combine multiple WS and/or processors.

• We use Taverna 2.4 http://www.taverna.org.uk
  - Documentation:
    http://dev.mygrid.org.uk/wiki/display/taverna/Documentation+and+Videos
  Quick start guide, videos, etc.
Workflow tutorial

• Find and run a workflow
  • http://vimeo.com/28449833

• Building a workflow from scratch
  • http://vimeo.com/28450024
Web cleaner and anonymizer
http://myexperiment.elda.org/workflows/98

- Input: a list of URLs to process
  - Example: a web article from www.fifa.com

1. ILSP Web cleaner and text extractor WS
2. UPF Anonymizer WS
   - Internally calls Freeling NER WS (3rd party tool)
   
   Interoperability ✔

Video: http://ws02.iula.upf.edu/panacea/examples/videos/Panacea_web_cleaner_and_anonymization_v01.mp4
Creation of a bilingual dictionary (only FR-EN)

- [http://myexperiment.elda.org/workflows/93](http://myexperiment.elda.org/workflows/93)

- Input: Pairs of Basic Xces Documents

1. Sentence alignment: Hunalign (3rd party tool) Interoperability ✓
2. PoS tagging: Treetagger (3rd party tool) Interoperability ✓
3. Build phrase tables: Moses (3rd party tool) Interoperability ✓
4. Bilingual dictionary extractor

Video: [http://ws02.iula.upf.edu/panacea/examples/videos/Panacea_bilingual_dictionary_extraction_v01.mp4](http://ws02.iula.upf.edu/panacea/examples/videos/Panacea_bilingual_dictionary_extraction_v01.mp4)
Be part of the platform: creating and sharing web services [30 min]
Deploy your tools as WS

• There are multiple solutions:
  – Soaplab, CLAM, Apache Axis2, Apache CXF, Spring

http://soaplab.sourceforge.net/soaplab2

• PANACEA can provide tips on setting up a Soaplab2 server
Share your WS in the Registry

• Provide your web services publically: gain visibility, make your work useful for others


• **How to register WS**
Thank you

Questions?