The Planets Interoperability Framework:
An Infrastructure for Digital Preservation Actions

Ross King
AIT Austrian Institute of Technology GmbH
Outline

• Planets Project
• Architecture
• Service Interfaces
• Digital Objects
• Workflow
• Conclusions
The Planets Project

• “Permanent Long-term Access through NETworked Services”
• Addresses the problem of digital preservation
  – driven by National Libraries and Archives
• Project instrument: FP6 Integrated Project
• 5. IST Call
• Consortium: 16 organisations from 7 countries
• Duration: 48 months, June 2006 – May 2010
• Budget: 14 Million Euro
• http://www.planets-project.eu/
Interoperability Framework: Vision

- Integration of Planets results in a single downloadable package
- This package will be simple to
  - install
  - configure
  - administer
- When this package – the **Planets Software Suite** – is deployed
  - an administrator can
    - create and manage user accounts
    - deploy and browse services
    - browse registries
  - a preservation expert can
    - define service workflows
    - define and evaluate preservation plans
    - define and run experiments
  - a librarian or archivist can
    - define and test preservation plans
    - execute preservation processes on a repository
Interoperability Framework: Results

- Software Architecture
- Digital Object Model
- Preservation Service Interface Definitions
## Interoperability Framework: Architecture

<table>
<thead>
<tr>
<th>Presentation Layer</th>
<th>Computational Layer</th>
<th>Business Layer</th>
<th>Database Layer</th>
</tr>
</thead>
<tbody>
<tr>
<td>JSF</td>
<td>Administration</td>
<td>Workflow Engine</td>
<td>Object-Relational Mapping</td>
</tr>
<tr>
<td></td>
<td>Testbed</td>
<td>Data Registry</td>
<td>Relational Database Management System</td>
</tr>
<tr>
<td></td>
<td>PLATO</td>
<td>User Management</td>
<td></td>
</tr>
<tr>
<td>JAX-WS</td>
<td>Preservation</td>
<td>Notification</td>
<td></td>
</tr>
<tr>
<td>Java EE 5</td>
<td>Application Services</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Job Submission</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Service</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Registry</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Logging</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Logging</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Java Technologies</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Interoperability Framework: Architecture

- Multi-tiered Architecture
- Built on a number of Java standards
- Presentation Layer
  - Java Server Faces (JSF)
- Logic Layer
  - Java Enterprise Edition (Java EE 5)
  - Java API for XML Web Services (JAX-WS)
- Data Layer
  - Java Content Repository (JCR) API
  - Java Persistence API (JPA)
Interoperability Framework: Architecture
Interoperability Framework: Digital Objects

- **Events**: Type, Time, Agent, Service, Result, ...
- **Properties**: Creator, Title, Description, Format, ...
- **Metadata**: Embedded Data or Repository URL
- **Content**: Embedded Data or Repository URL
- **Digital Object**: contains_object
- **Tagged Uninterpreted Metadata Chunks**: Relationships (possibly associated with event)
Interoperability Framework: Digital Object Managers

Data Registry Service

Different data sources, repositories, interfaces, and protocols.
Interoperability Framework: Digital Object Managers
Interoperability Framework: Preservation Service Interfaces

- Define atomic preservation activities
  - Concentrates on low-level concepts and actions
    - Bit-stream operations, no data management
    - Designed to be light-weight and easy to implement
- Independent from a specific tool, language, or content type
  - e.g. Characterize, Migrate, Compare, CreateView
  - > 50 Tools wrapped/provided as Planets Services
- Provides the basic abstractions for assembling workflows.
Interoperability Framework: Preservation Service Registry

- **Presentation Layer**
  - JSF
  - Administration
  - Testbed
  - PLATO

- **Computational Layer**
  - JAX-WS
  - Java EE 5
  - Preservation Application Services
  - Job Submission Service

- **Business Layer**
  - Workflow Engine
  - Data Registry
  - User Management
  - Notification
  - Service Registry

- **Database Layer**
  - JCR
  - JPA
  - Object-Relational Mapping
  - Relational Database Management System

Java Technologies

planets
Interoperability Framework:
Preservation Service Registry
Interoperability Framework: Workflow Templates

• Separation of concerns:
  – Fragments of complex workflow logic are implemented by <<workflow developers>>
  – <<Experimenter>>s selected from predefined templates, configure them, and execute individual processes.

• Templates implement abstract and reusable processes definitions based on Preservation Service Interfaces and decision logic.

• Execute in trusted environment
  – handle digital objects in metadata repository and
  – basis for recording provenance and preservation information
Interoperability Framework: Workflow Templates

1. **Submission**
2. **Validate**
   - Validate the submission
3. **Identify**
   - Identify the digital objects in the submission package
4. **Characterize**
   - Characterize the valid digital objects
5. **Normalize**
   - Normalize the valid digital objects (e.g. to PDF/A)

SIP
Interoperability Framework:
Workflow Engine

Presentation Layer
- JSF
- Administration
- Testbed
- PLATO

Computational Layer
- JAX-WS
- Java EE 5
- Preservation Application Services
- Job Submission Service

Business Layer
- Workflow Engine
- Logging
- Data Registry
- Service Registry
- User Management
- Notification

Database Layer
- JCR
- JPA
- Object-Relational Mapping
- Relational Database Management System

Java Technologies

planets
Interoperability Framework: Workflow Engine

Workflow Control Panel

Edit an XML Workflow description

Services

Service ID | Type | Endpoint
--- | --- | ---
Identity | Identity | Select Endpoint...
Migrate | GraphicsMagicMigrate | Select Endpoint...

Parameters

Name/Value Pairs

Name | Value
--- | ---
compressionType | None, B@P, Fax, Group4, JPEG, LZW, RLE, Zip
imageQuality | 0-100

Add Parameter

Edit Parameters

Save Workflow | Clean Workflow
--- | ---
Export XML | Reset

Active Template

Active Workflow

Input Objects Selected

Submit Workflow

Error Messages

[Image of the European Union flag]
Interoperability Framework: Workflow Engine (next generation)

See http://www.trianacode.org/
Interoperability Framework: Conclusions and Future Work

• The Planets Software Suite is available for download in a platform-independent Java-based installation package:

http://gforge.planets-project.eu/gf/project/ifsp/

• In the fourth and final year of the project, the primary focus will not be on functionality but on architecture improvements. Clustering, scalability and robust workflows will be the main focus of our development.

• Finally, field tests of the Planets Software Suite will be carried out at partner institutions, demonstrating how the Interoperability Framework and the associated Planets applications and services can act as an added-value preservation action system for existing digital repositories at national libraries and archives.
Thank you for your attention!

Contact information:

Dr. Ross King
AIT Austrian Institute of Technology GmbH
ross.king@ait.ac.at