Sustainable Software

Reagan W. Moore
Arcot Rajasekar
Mike Wan
{moore,sekar,mwan}@diceresearch.org
http://irods.diceresearch.org

Sponsored by the executive office of the president national coordination office for networking and information technology research and development and by NARA
## Data Life Cycle

Each data life cycle stage re-purposes the original collection

<table>
<thead>
<tr>
<th>Stage</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Collection</td>
<td></td>
</tr>
<tr>
<td>Private</td>
<td></td>
</tr>
<tr>
<td>Local Policy</td>
<td></td>
</tr>
<tr>
<td>Data Grid</td>
<td></td>
</tr>
<tr>
<td>Shared</td>
<td></td>
</tr>
<tr>
<td>Distribution Policy</td>
<td></td>
</tr>
<tr>
<td>Data Processing Pipeline</td>
<td></td>
</tr>
<tr>
<td>Analyzed</td>
<td></td>
</tr>
<tr>
<td>Service Policy</td>
<td></td>
</tr>
<tr>
<td>Data Grid</td>
<td></td>
</tr>
<tr>
<td>Distributed Library</td>
<td></td>
</tr>
<tr>
<td>Published</td>
<td></td>
</tr>
<tr>
<td>Description Policy</td>
<td></td>
</tr>
<tr>
<td>Reference Collection</td>
<td></td>
</tr>
<tr>
<td>Preserved</td>
<td></td>
</tr>
<tr>
<td>Representation Policy</td>
<td></td>
</tr>
<tr>
<td>Federation</td>
<td></td>
</tr>
<tr>
<td>Sustained</td>
<td></td>
</tr>
<tr>
<td>Re-purposing Policy</td>
<td></td>
</tr>
</tbody>
</table>

Stages correspond to addition of new policies for a broader community

Virtualize the stages of the data life cycle through policy evolution
Map from the actions requested by the access method to a standard set of micro-services.

The standard microservices are mapped to standard operations.

The standard operations are mapped to the protocol supported by the storage system.
iRODS Distributed Data Management
Sustainable Software

• Bugzilla
  – Bug fixes are highest priority
  – Access to SVN and new features in development
• Discussion list
  – Community support for problem resolution
• International development
  – Expertise also resides outside development team
• Rapid Prototyping
  – Generate new releases with new feature requests
• Production evaluation
  – Feedback on robustness, performance, features
  – Iterate
What is your recommendation for how much of the Track1 and Track 2 money should be spent on software development by award winners, by others, and why?

- **Storage Resource Broker middleware development costs**
  - 300,000 lines of code
  - Six year development / ten year deployment
  - 10-15 professional software engineers

- **Total cost ~ $15,000,000**
  - $17 / line for design, development, testing, documentation, bug fixes
  - $14 / line for interoperability (clients)
  - $12 / line for application use support
  - $7 / line for management / administration
  - Total cost ~ $50 / line

- **Development funded by:**
  - NSF / NARA / DARPA / DoE / NASA / NIH / IMLS / NHPRC / LoC / DoD
  - More than 20 funded projects to sustain development
  - International collaborations on use, development, bug fixes, support
What evaluation criteria should be put in place for software development and maintenance? Consider properties that software needs for sustainability:

• **Strongly believe in generic infrastructure that supports all data management applications**
  – Identify generic properties required by all applications
  – Identify generic mechanisms to enable each community to tune the software

• **Strongly believe in highly extensible infrastructure**
  – Focus on framework for interoperability with legacy systems, new clients

• **Requires input from as many communities as possible on usage models, aggregated over 10-years of user input**
  – Science and engineering disciplines
  – Digital library community
  – Preservation community
  – Data processing pipelines

• **Requires validation of the technology in production use across communities**
  – Feedback on performance, robustness, bug fixes, new extensions
  – Implies rapid prototyping, iterative development cycle to track evolving requirements
Is there a (virtual) payment system that would allow other NSF awardees to indicate their desire for continued support of well-utilized software that they depend upon but do not develop themselves?

- **Collaboration software models:**
  - Explicit collaborations with multiple communities to prove software can be tuned to support community specific feature development
  - Multiple funding sources to ensure requirements gathered from all user communities

- **Peer models**
  - Usage statistics on communities successfully using technology
  - Migration of technology into commercial products

- **Standards models:**
  - Migration of technology specifications into commercial standards
  - Migration of middleware into disk controllers

- **Federation models:**
  - Creation of collaborations between disciplines that build upon common software
  - Migrate technology forward into the next project
How does standardization of software interfaces (including evaluation/funding) fit to enable interoperability among NSF funded software systems?

• Each research community has a different required architecture / protocol / interface:
  – Workflows, web services, portals, message bus, data grid

• Interoperability requires a highly extensible framework
  – Support mapping between protocols
  – Support evolution of the framework itself (policies, procedures, protocols, state information)

• Standardization is a point-in-time solution
  – Need interoperability mechanisms to enable use of the next generation protocols with last generation protocols
  – Example is virtualization of
    • Collections / trust / state information / procedures / policies