Policy Based Preservation Environments

Reagan W. Moore
Arcot Rajasekar
Mike Wan
Mike Conway
Antoine de Torcy
Richard Marciano
Jewel Ward

{moore,sekar,mwan}@diceresearch.org
http://irods.diceresearch.org
Preservation Challenges

- **Infrastructure independence**
  - Manage properties of the archives independently of the choice of technology

- **Communication with the future**
  - Provide representation information for each record (provenance, descriptive metadata, data structure, parsing)
  - Provide representation information for the preservation environment (policies, procedures, state information)
  - Maintain ability to apply preservation procedures by migrating procedures onto new technology

- **Assessment of communication from the past**
  - Track application of all versions of preservation policies
  - Validate assertions about preservation properties
Policy-based Data Preservation
integrated Rule Oriented Data System

• **Purpose** - reason a preservation environment is assembled
• **Properties** - attributes needed to ensure the **purpose**
• **Policies** - controls for enforcing desired **properties**
• **Procedures** - functions that implement the **policies**
• **State information** - results of applying the **procedures**
• **Assessment criteria** - validation that **state information** conforms to the desired **purpose**
• **Federation** - controlled sharing of **logical name spaces**

These are the essential elements for policy-based data preservation.
Infrastructure Independence

Map from the actions requested by the client to multiple policy enforcement points.

Map from policy to standard micro-services.

Map from micro-services to standard Posix I/O operations.

Map from standard Posix I/O operations to the protocol supported by the storage system.
iput ../src/irm.c checks 10 policy enforcement points

srbbbrick14:10900:ApplyRule#116:: acChkHostAccessControl
srbbbrick14:10900:GotRule#117:: acChkHostAccessControl
srbbbrick14:10900:ApplyRule#118:: acSetPublicUserPolicy
srbbbrick14:10900:GotRule#119:: acSetPublicUserPolicy
srbbbrick14:10900:ApplyRule#120:: acAclPolicy
srbbbrick14:10900:GotRule#121:: acAclPolicy
srbbbrick14:10900:ApplyRule#122:: acSetRescSchemeForCreate
srbbbrick14:10900:GotRule#123:: acSetRescSchemeForCreate
srbbbrick14:10900:execMicroSrvc#124:: msiSetDefaultResc(demoResc,null)
srbbbrick14:10900:ApplyRule#125:: acRescQuotaPolicy
srbbbrick14:10900:GotRule#126:: acRescQuotaPolicy
srbbbrick14:10900:execMicroSrvc#127:: msiSetRescQuotaPolicy(off)
srbbbrick14:10900:ApplyRule#128:: acSetVaultPathPolicy
srbbbrick14:10900:GotRule#129:: acSetVaultPathPolicy
srbbbrick14:10900:execMicroSrvc#130:: msiSetGraftPathScheme(no,1)
srbbbrick14:10900:ApplyRule#131:: acPreProcForModifyDataObjMeta
srbbbrick14:10900:GotRule#132:: acPreProcForModifyDataObjMeta
srbbbrick14:10900:ApplyRule#133:: acPostProcForModifyDataObjMeta
srbbbrick14:10900:GotRule#134:: acPostProcForModifyDataObjMeta
srbbbrick14:10900:ApplyRule#135:: acPostProcForCreate
srbbbrick14:10900:GotRule#136:: acPostProcForCreate
srbbbrick14:10900:ApplyRule#137:: acPostProcForPut
srbbbrick14:10900:GotRule#138:: acPostProcForPut
srbbbrick14:10900:GotRule#139:: acPostProcForPut
srbbbrick14:10900:GotRule#140:: acPostProcForPut
Preservation Policies

• Arrangement
  – Organize records in series, and manage policies on each series

• Authenticity
  – For every record, record provenance metadata

• Chain of custody
  – For every record, manage an audit trail

• Integrity
  – For every record, manage two replicas and verify checksums
  – For every record, enforce retention and disposition

• Trustworthiness
  – For each series, validate ISO MOIMS-rac assessment criteria
Applying Policy Sets

• Maintain a repository of policies in iRODS
  – Policies stored as XML files with input parameters explicitly characterized as policy attributes

• Map policies to a record series (collection)
  – Set collection attributes from policy attributes

• On ingestion of a file:
  – Check record series for policy attributes
  – Invoke associated policy and apply preservation procedure
  – Decouples choice of access interface from specification of preservation policies
Arch - Defining a Policy

Add a Policy

Policy Name
demo policy with checksum

Select a Policy
Repository: saa demo policy repo 1

Description
This is a simple policy that does checksum and replication

Do objects need to go to a staging area first?

Is virus scanning required?

Is a checksum required?

How many replicas should be made?

How many days should items be retained?

Reset  Update
Adding Policies to a Record Series

**Add a Policy**

- **Policy Name:** demo policy with checksum
- **Select a Policy Repository:** saa demo policy repo 1
- **Description:** This is a simple policy that does checksum and replication
- **Do objects need to go to a staging area first?**
  - [ ]
- **Is virus scanning required?**
  - [x]
- **Is a checksum required?**
  - [x]
- **How many replicas should be made?**
  - [ ]
- **How many days should items be retained?**
  - [ ]

[Reset] [Update]
iDrop Client - Ingesting a Record
iRODS is a "coordinated NSF/OCI-Nat'l Archives research activity" under the auspices of the President's NITRD Program and is identified as among the priorities underlying the President's 2009 Budget Supplement in the area of Human and Computer Interaction Information Management technology research.

Reagan W. Moore
rwmoore@renci.org
http://irods.diceresearch.org

NSF OCI-0848296 “NARA Transcontinental Persistent Archives Prototype”
NSF SDCI-0721400 “Data Grids for Community Driven Applications”