Data Science Initiatives

Reagan Moore
rwmoore@renci.org
University of North Carolina at Chapel Hill

• Data Intensive Cyber Environments Center
  – DataNet Federation Consortium
  – Research Data Alliance

• Renaissance Computing Institute
  – National Consortium for Data Science
  – iRODS Consortium

• School of Information and Library Science
  – LifeTime Library
  – Data science courses
Topics

1. DataNet Federation Consortium: federation across NSF cyber infrastructure projects
2. Progress towards a theory of data science: Basis for quantitative predictions about load and performance (Hao Xu)
3. Development of policies for managing protected data: pluggable rule engine in iRODS (Hao Xu)
4. iRODS Consortium: Industrial storage vendor provisioning of data management - packaging of iRODS with Seagate disk
5. Federation of repositories across federal agencies. Current applications of iRODS include NASA, NOAA, NOAO, and NSF awards.
DataNet Federation Consortium [1]

- University of North Carolina at Chapel Hill
  - Odum Institute - Dataverse (Social Science)
  - Institute for the Environment (Hydrology)
  - Renaissance Computing Institute - GENI (SDN Networks)
  - Data Intensive Cyber Environments Center (Data grids)

- University of California, San Diego
  - Science Observation Network – SciON (Sensor data)
  - Temporal Dynamics of Learning Center (Cognitive science)

- Arizona State University
  - Natural language processing (Computer science)

- University of Arizona
  - The iPlant Collaborative (Plant biology)

- University of Virginia
  - HydroShare (Hydrology)

- Drexel University
  - Semantic ontologies (Engineering)
Federation Mechanisms

• Claim three mechanisms are sufficient for federating existing data management systems

  – Tightly coupled federations
    • Shared name spaces
      – Federated data grids

  – Loosely coupled federations
    • Direct interaction using API of the remote system
      – Encapsulate published APIs in micro-services

  – Asynchronous federations
    • Indirect interaction
      – Communicate through a message bus
Federation of Data & Services

• Move data to remote service
  – Access remote service
  – Move data, apply service, and return results

• Move service to local data
  – Encapsulate service in virtual machine image (Docker)
  – Move service to local storage
  – Execute service as part of a local workflow
Theory of Data Science [2]

• A successful theory should support:
  – Characterization of a data management system through the changes to state information by operations
  – Identification of the assertions (conserved properties) maintained by the data management system
  – Prediction of the probability of success in maintaining the assertions in the presence of failure modes
  – Prediction of the sustainable workload
  – Identification of the assertions maintained across a federation
  – Prediction of the probability of success and sustainable workload of a federation
Required Infrastructure Components

• Policy-based system
  – Computer actionable rules
  – Computer executable procedures

• Persistent state information
  – Identification of the states changed by each operation

• Event tracking
  – For each operation, monitor the state changes
Policy Components - Conceptual Fundamentals
Policy-based Data Management Concept Graph for iRODS
Policy-based Data Management Concept Graph for iRODS

1. **Purpose**
   - Defines Collection
   - SubType: Sharing, Publication, Preservation

2. **Property**
   - Defines Policy

3. **Policy**
   - Has Digital Object
   - Control Updates

4. **Collection**
   - Has Property
   - Has Attribute

5. **Digital Object**
   - Updates Persistent State Information

6. **Attribute**
   - Isa

7. **Community Consensus**

8. **Computer Actionable Implementation**
Policy Components - Conceptual Fundamentals

Policy-based Data Management Concept Graph for iRODS

- Purpose
  - Defines Collection
  - SubType
    - Sharing
      - Publication
      - Preservation
    - Integrity
    - Authenticity
    - Access control
  - Completeness
  - Consensus
  - Correctness
  - Consistency

- Property
  - Defines Policy
  - Isa
    - Replication Policy
    - Checksum Policy
    - Quota Policy
    - Data Type Policy
  - HasFeature
    - Replication Policy
    - Checksum Policy
    - Quota Policy
    - Data Type Policy

- Policy
  - Defines Procedure
  - Controls
    - Updates Persistent State Information
  - Isa
    - Integrity
    - Authenticity
    - Access control

- Procedure
  - Updates Persistent State Information

- Collection
  - Has
    - Digital Object
    - Attribute
  - Isa
    - Digital Object

- Digital Object
  - Isa
    - Attribute

- Attribute
  - Isa
    - Persistent State Information
Policy Components - Conceptual Fundamentals

Policy-based Data Management Concept Graph for iRODS

- Purpose
- Sharing
- Publication
- Preservation
- Integrity
- Authenticity
- Access control
- Completeness
- Correctness
- Consensus
- Consistency
- SubType

Policy Concepts:
- Defines
- HasFeature
- Isa

Collection:
- Defines
- Has

Replication Policy
- Isa
- HasFeature

Checksum Policy
- Isa
- HasFeature

Quota Policy
- Isa
- HasFeature

Data Type Policy
- Isa
- HasFeature

Property
- Defines

Replication Policy
- Isa
- HasFeature

Checksum Policy
- Isa
- HasFeature

Quota Policy
- Isa
- HasFeature

Data Type Policy
- Isa
- HasFeature

Procedure
- Updates
- Isa

Workflow
- Chains
- Isa

Function
- Isa

Operation
- SysChksumDataObj
- GetDataObjRepl
- SetDataType
- SetQuota
- GetUserACL

Attribute
- Isa

Persistent State Information
Policy Components - Conceptual Fundamentals

Policy-based Data Management Concept Graph for iRODS

- **Purpose**
  - Defines
  - SubType: Sharing, Publication, Preservation

- **Property**
  - Defines
  - HasFeature: Completeness, Correctness, Consensus, Consistency

- **Collection**
  - Has
  - Isa: Replication Policy, Checksum Policy, Quota Policy, Data Type Policy
  - Updates

- **Digital Object**
  - Has
  - Isa: Persistent State Information

- **Attribute**
  - Isa: Function, Operation

- **Policy**
  - Defines
  - Isa: Policy Enforcement Point

- **Procedure**
  - Updates
  - Isa: Workflow Chains

- **SubType**
  - Isa: Function, Operation

- **Data ID**
  - Isa: DATA_ID

- **Data Repl Num**
  - Isa: DATA_REPL_NUM

- **Data Checksum**
  - Isa: DATA_CHECKSUM

- **Replication Policy**
  - Isa: IsA

- **Checksum Policy**
  - Isa: IsA

- **Quota Policy**
  - Isa: IsA

- **Data Type Policy**
  - Isa: IsA

- **Consistency**
  - Isa: IsA

- **Integrity**
  - Isa: IsA

- **Authenticity**
  - Isa: IsA

- **Access control**
  - Isa: IsA

- **Sharing**
  - Isa: IsA

- **Publication**
  - Isa: IsA

- **Preservation**
  - Isa: IsA

- **GetUserACL**
  - Isa: IsA

- **SetDataType**
  - Isa: IsA

- **SetQuota**
  - Isa: IsA

- **DataObjRepl**
  - Isa: IsA

- **SysChksumDataObj**
  - Isa: IsA
Data Science

• iRODS provides the required components
  – Operations defined in micro-services
  – State information updated on each operation
  – Policies control execution of the micro-services
    • Pre-process policy
    • Post-process policy

• Needed high performance tracking of events
  – Write rules in C++
  – Send event messages to external index
iRODS Pluggable Architecture

- Interactions with new technologies are encapsulated in plug-ins
  - API
  - Authentication systems (GSI, Kerberos)
  - Databases
  - Micro-services (curl)
  - Network
  - Storage systems (S3, WOS, HPSS)
  - Zonereport
Pluggable Architecture (Coposky)

- Plug-in new micro-services (iRODS 4.0)
  - Associate policy enforcement points with each plug-in
    - Pre-process policy
    - Post-process policy

- Pluggable Rule engine architecture (Hao Xu)
  - Support plug-ins for each rule language
    - iRODS, Python, JavaScript
  - Support meta-rule name spaces
    - Automatically add auditing rules to every micro-service invocation
Pluggable Rule Engine

• Supports 5 basic functions
  – Rule-engine-plugin-start
  – Rule-engine-plugin-stop
  – Rule-exists
  – Rule-execution
  – Callback

• Plugin typically is implemented in 100 lines of code
  – iRODS rule language, Python, Javascript, C++
iRODS Core

Plugin Architecture

RE_Plugin

Any plug-in

Policy Enforcement Point

Meta-Rule

Rules
Rule Name Space

- RE Plugin provides extended namespace support for the translation to the default semantics.
- `pep[...] = ns_{1}\text{preop}(args, env) \gg \ldots ns_{n}\text{preop}(args, env) \gg \text{op}(args, env) \gg \ldots ns_{n}\text{postop}(args, env) \gg \ldots ns_{1}\text{postop}(args, env)`
- By default, we have namespace `ns_{1} = ""`.
- We can add more namespaces. For example, for auditing `ns_{2} = "audit "` or indexing `ns_{3} = "index "` or security.
- For the audit namespace, pre and post file read PEPs:
  
  `audit_pep_resource_read_pre`
  `audit_pep_resource_read_read_post`
Policy Sets [3]

- Event auditing
- External indexing (events, metadata, text)
- Protected Data management
- Preservation
- Digital Library
- Data sharing
Protected Data

• Automated enforcement of 51 tasks, such as
  – Protected data type detection
  – Access approval flags
  – Encryption
  – Access control setting
  – Replication
  – Retention
  – Audit trail parsing for compliance
  – Verification of required access controls
  – Verification of integrity
  – Password constraints
iRODS User Group Meeting [4]

- iRODS Consortium – membership based sustainable infrastructure
  - Pharmaceutical companies
    - Genomics data grids
  - Storage vendors
    - Data management
    - Example – iRODS appliance integrates disk storage with pre-installed iRODS data grid (Seagate, DDN)
    - Can connect iRODS appliance to any data grid
    - Generalization of SAN technology
Federation [5]

• DataNet Federation Consortium pursuing federation across cyberinfrastructure projects and federal agencies
  – Data Infrastructure Building Blocks
    • GABBS – Geospatial Modeling and Analysis Building Blocks
    • Encapsulated service in Docker image
  – Data grids – NASA, NOAA, NSF, (EPA, NIH, NIEHS, NARA, LoC)
• Service federation
  – Through the Discovery Environment (iPlant) manage data movement and execution of encapsulated service on HPC resource (TACC)
  – Workflow structured objects – track provenance of workflows within iRODS
Virtualization of Data Flows

• Currently virtualize data collections and workflows
  – Manage their properties, access controls, provenance, naming, sharing, organization

• Can also virtualize data flows
  – Integration of Software Defined Networks and Policy-based data management
  – Enables re-execution of data flows, access controls on data flows, sharing, caching, assignment of I/O streams, event tracking, access by collection/file name
Development

• iRODS Consortium
  – iRODS release 4.1.3
  – Pluggable architecture – Jason Coposky

• DICE Center - DFC
  – Workflow structured objects – Arcot Rajasekar
  – Pluggable rule engine – Hao Xu

• RENCI
  – GENI – Shu Huang, Yufeng Xin

• Project support from:
  – NSF DataNet Federation Consortium Grant OCI 0940841
Federation Architecture

REST

Browse

I/O

Library

FUSE

iRODS Data Grid

Message

Bus

Storage

Database

Micro-

service

External

Indexing

Federated Data

Grid

Web

Service

Web

Service