Policy Based Data Management

Reagan W. Moore
Arcot Rajasekar
Mike Wan
Wayne Schroeder
Mike Conway
Jason Coposky
{moore,sekar,mwan, schroeder}@diceresearch.org
michael_conway@unc.edu
http://irods.diceresearch.org

Policy-based Data Environments

- *Purpose* - reason a collection is assembled
- *Properties* - attributes needed to ensure the **purpose**
- *Policies* - controls for enforcing desired **properties,**
- **mapped to computer actionable rules**
- *Procedures* - functions that implement the **policies**
- **mapped to computer actionable workflows**
- *State information* - results of applying the **procedures**
- **mapped to system metadata**
- *Assessment criteria* - validation that **state information** conforms to the desired **purpose**
- **mapped to periodically executed policies**

2

Digital Library

Texas Digital Library

French National Library

Data Processing Pipeline

Ocean Observatories Initiative

Large Synoptic Survey Telescope

Data Grid

Teragrid

Temporal Dynamics of Learning Center

Australian Research Collaboration Service

Preservation Environment

NARA Transcontinental Persistent Archive Prototype

Taiwan National Archive

Carolina Digital Repository

Generic Infrastructure

- Clients – specific to discipline and life cycle state
- Policies – specific to discipline
- Procedures – specific to discipline
- Remaining infrastructure is generic
  - Network transport
  - Authentication / Authorization
  - Distributed storage access
  - Remote execution
  - Metadata management
  - Message passing
  - Rule engine

Applications

iRODS User Group Meeting

DataNet Proposal

DOE Collaborations

NASA Paper

Federated Observational and Simulation Data in the NASA Center for Climate Simulation Data Management System Project

John L. Schnase, Glenn Tamkin,

David Fladung, Scott Sinno, and Roger Gill

https://www.irods.org/index.php/iRODS_User_Group_Meeting_2011

NASA Center for Climate Simulation

- Observational data
  - MODIS data set, 1 PB, 54 million files, 300 million attributes
  - Created script to control registration of MODIS data into iRODS and an associated database to monitor the submission process
  - Provided access to data products
- Simulation data
  - Added auditing extensions to track status of products
  - New lookup and historical tables
  - Added policy enforcement points

NASA Center for Climate Simulation

- Integrated data grids with Earth System Grid
- Used FUSE file system interface
  - Effective for read operations on remote data grid

- Integrated with NASA Cloud Services
  - Amazon EC2 cloud model

KEK Paper

IRODS in an Neutrino Experiment

Adil Hasan

for

Francesca Di Lodovico (QMUL), Yoshimi Iida (KEK), Takashi Sasaki (KEK)

https://www.irods.org/index.php/iRODS_User_Group_Meeting_2011

iRODS in an Neutrino Experiment

- Tokai to Kamioka data grid in Japan
  - Provide access to global collaborators
  - Must aggregate files for storage in HPSS in 1-GB containers
  - File sizes ranged from kiloBytes to MegaBytes
- Created policies to:
  - Automate bundling of files
  - Replicate containers into HPSS
  - Purge cache and backup resources

Rule to Bundle Files

```
acKEKBundle(*collPath, *bundlePath, *cacheRes, *compRes, *archive,
*threshold)||
        msiCheckCollSize(*collPath, *cacheRes, *threshold,
                *aboveThreshold, *status)##
        ifExec(*aboveThreshold == 1,
                msiWriteRodsLog("Creating bundle", *status)##
                msiPhyBundleColl(*collPath, *compRes,*status)##
                msiWriteRodsLog("Finished bundling, starting to
replicate",
                        *status)##
                msiCollRepl(*bundlePath,
verifyChksum++++backupRescName                    =*archive,
*status)##
                msiWriteRodsLog("Finished replicating bundle", *status),
        nop##nop##nop##nop##nop, nop, nop) |nop##nop
```

iRODS Rule to Replicate Files

```
acKEKReplicate(*collPath, *cacheRes, *archive, *threshold)||
msiCheckCollSize(*collPath, *cacheRes, *threshold, *aboveThreshold, *status)##
ifExec(*aboveThreshold == 1, nop, nop,
        msiWriteRodsLog("Starting to backup files", *status)##
        acGetIcatResults(list, COLL_NAME LIKE '*collPath', *List)##
        forEachExec(*List, msiGetValByKey(*List, DATA_NAME, *Data)##
                msiGetValByKey(*List, COLL_NAME, *Coll)##
                msiGetValByKey(*List, DATA_RESC_NAME, *dataRes)##
                ifExec(*dataRes == *cacheRes,
                        msiWriteRodsLog("Replicating file *Coll/*Data",
*status)##
                        msiDataObjRepl(*Coll/*Data,
verifyChksum++++backupRescName=
                                *archive,  *status)##
                        msiWriteRodsLog("Completed replicating file
*Coll/*Data",
                                *status)
```

iRODS Rule to Trim Replicas

```
acKEKTrimData(*collPath, *cacheRes)||
acGetIcatResults(list, COLL_NAME LIKE '*collPath', *List)##
forEachExec(*List, msiGetValByKey(*List, DATA_NAME, *Data)##
        msiGetValByKey(*List, COLL_NAME, *Coll)##
        msiGetValByKey(*List, DATA_RESC_NAME, *DataResc)##
        msiGetValByKey(*List, DATA_REPL_NUM, *DataRepl)##
        ifExec(*DataResc == *cacheRes,
                msiWriteRodsLog("About to trim file *Coll/*Data",
*status)##
                msiDataObjTrim(*Coll/*Data, *cacheRes, *DataRepl, 1,
                        IRODS_ADMIN_KW=irodsAdmin, *status)##
                msiWriteRodsLog("Completed trimming replicas of
*Coll/*Data",
                        *status),
nop##nop##nop, nop, nop), nop##nop##nop##nop##nop) |nop##nop
```

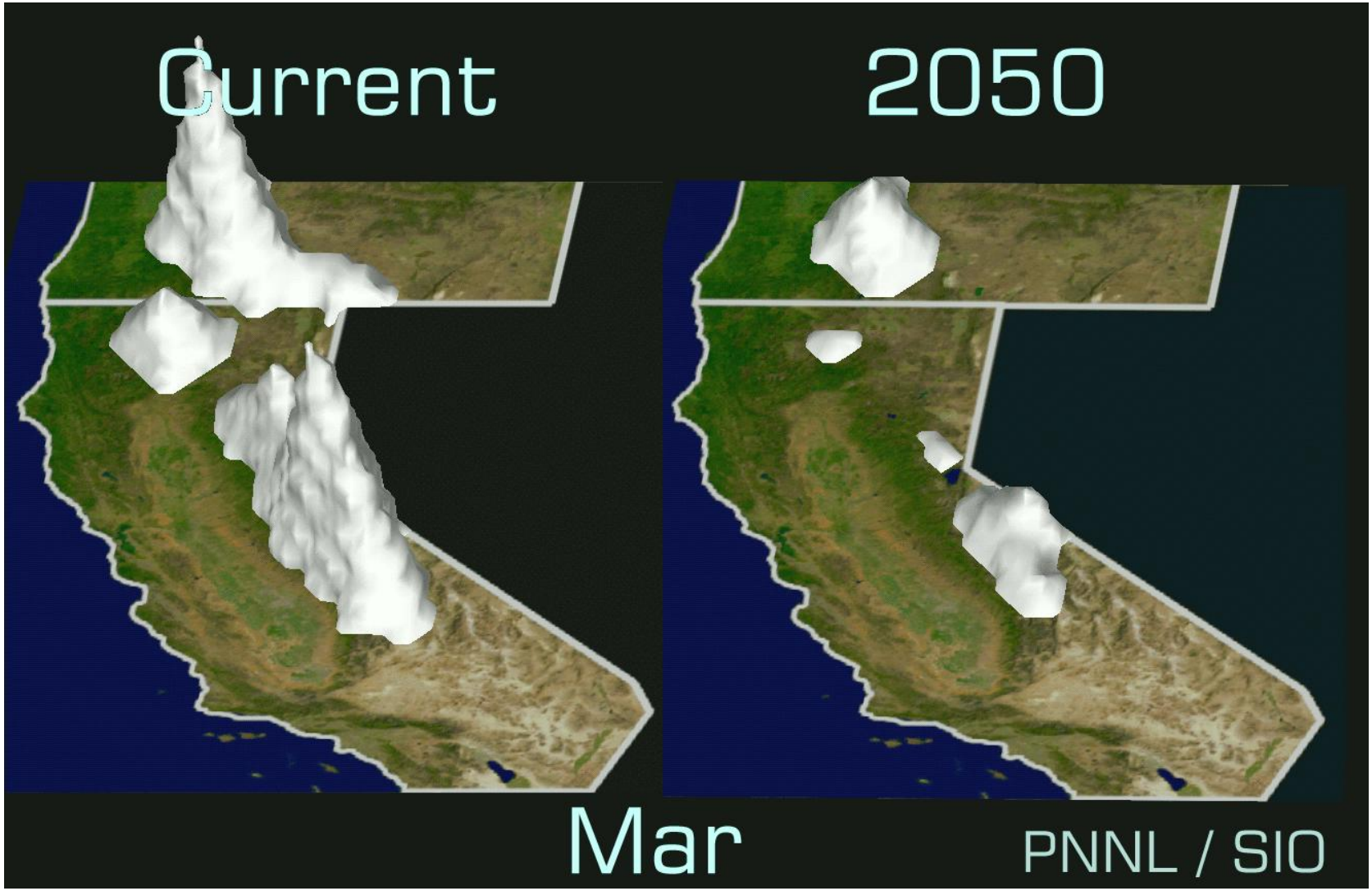
DOE Collaborations

LUSciD-LLNL UCSD/SIO Scientific Data Project:

Climate Studies

SIO	*LLNL*	*SDSC*
Tim Barnett	Doug Rotman	Reagan Moore
David Pierce	Dave Bader	Leesa Brieger
Dan Cayan	Ben Santer	Amit Chourasia
Hugo Hidalgo	Peter Gleckler	
Mary Tyree	Krishna AcutaRao	

Change in California Snowfall



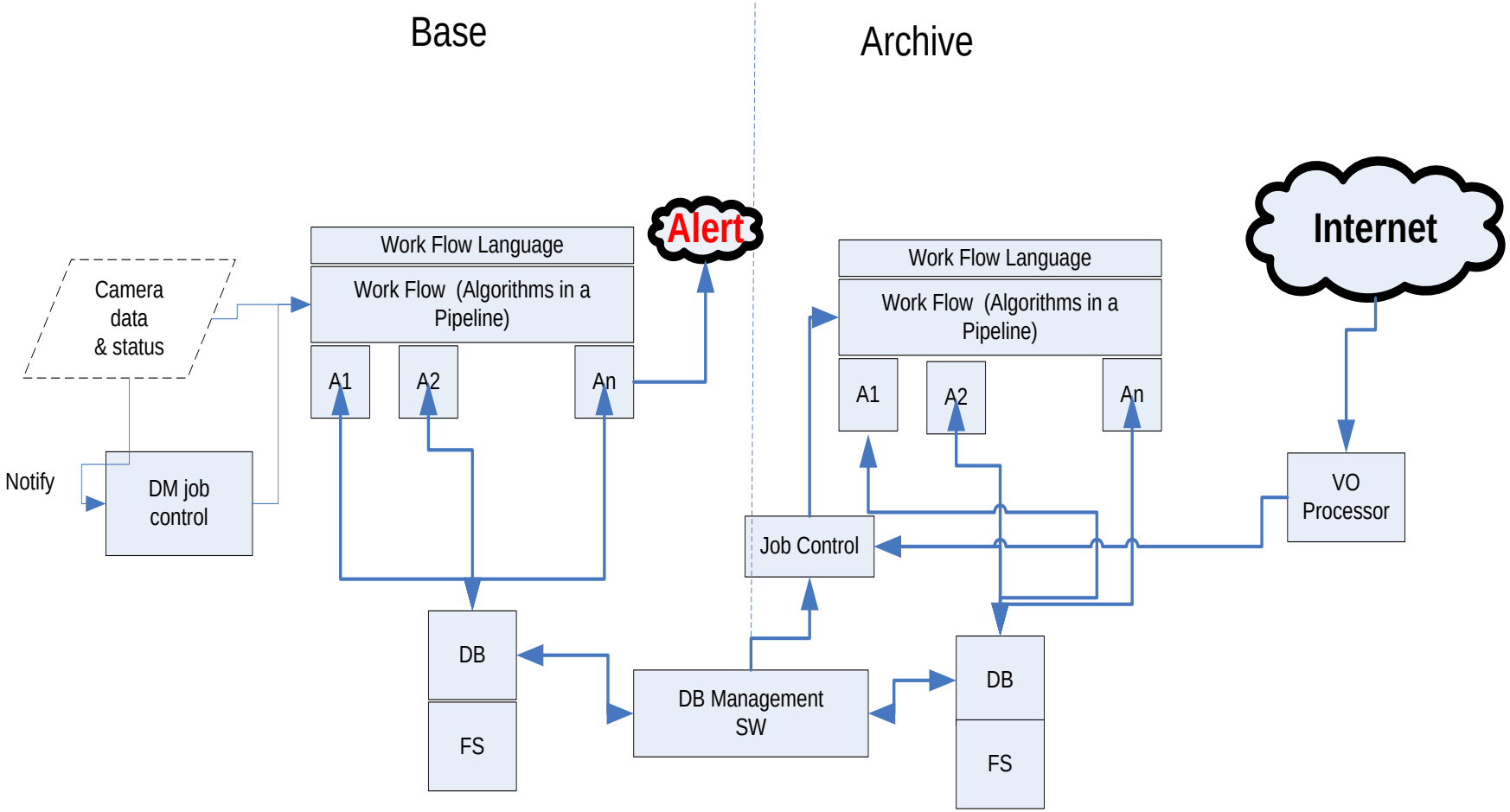
SDSC-LLNL data management collaboration

Ghaleb Abdulla

Center for Applied Scientific Computing

Lawrence Livermore National Lab

The LSST infrastructure challenge

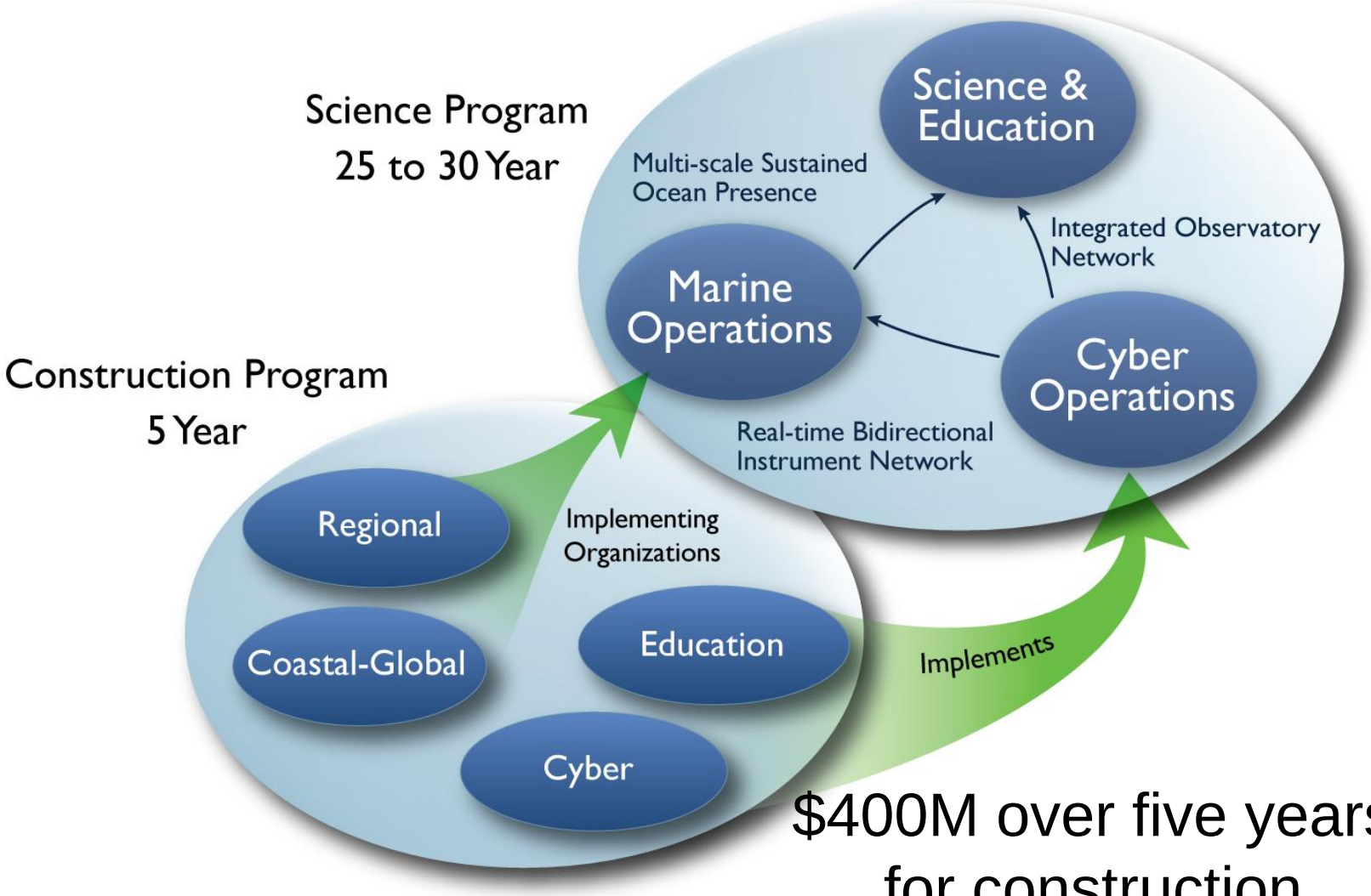


DataNet Proposal

Ocean Observatories Initiative

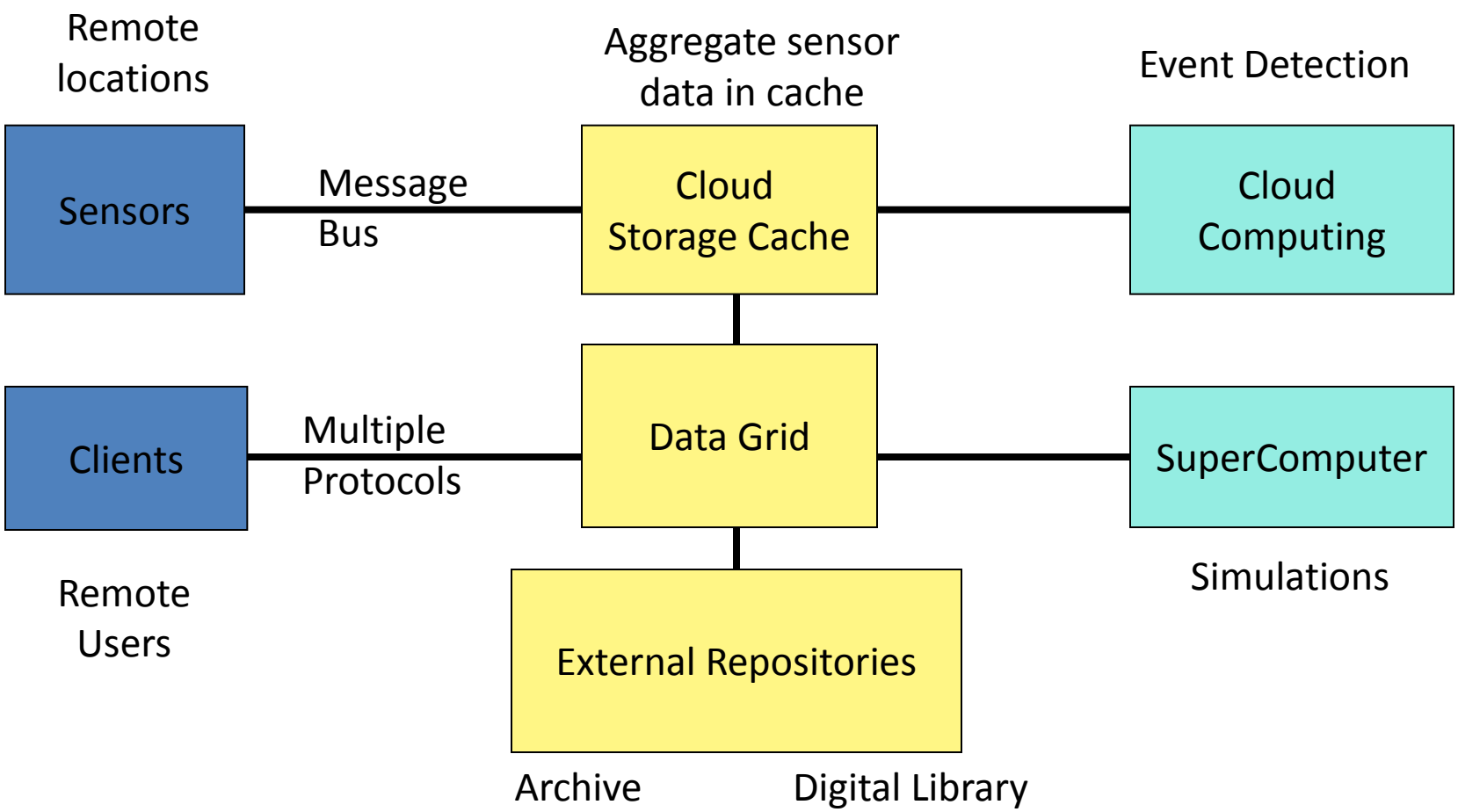
Hydrology Data Grid

OOI Will Be Linked by a Common Instrument, Infrastructure, and Information Management System

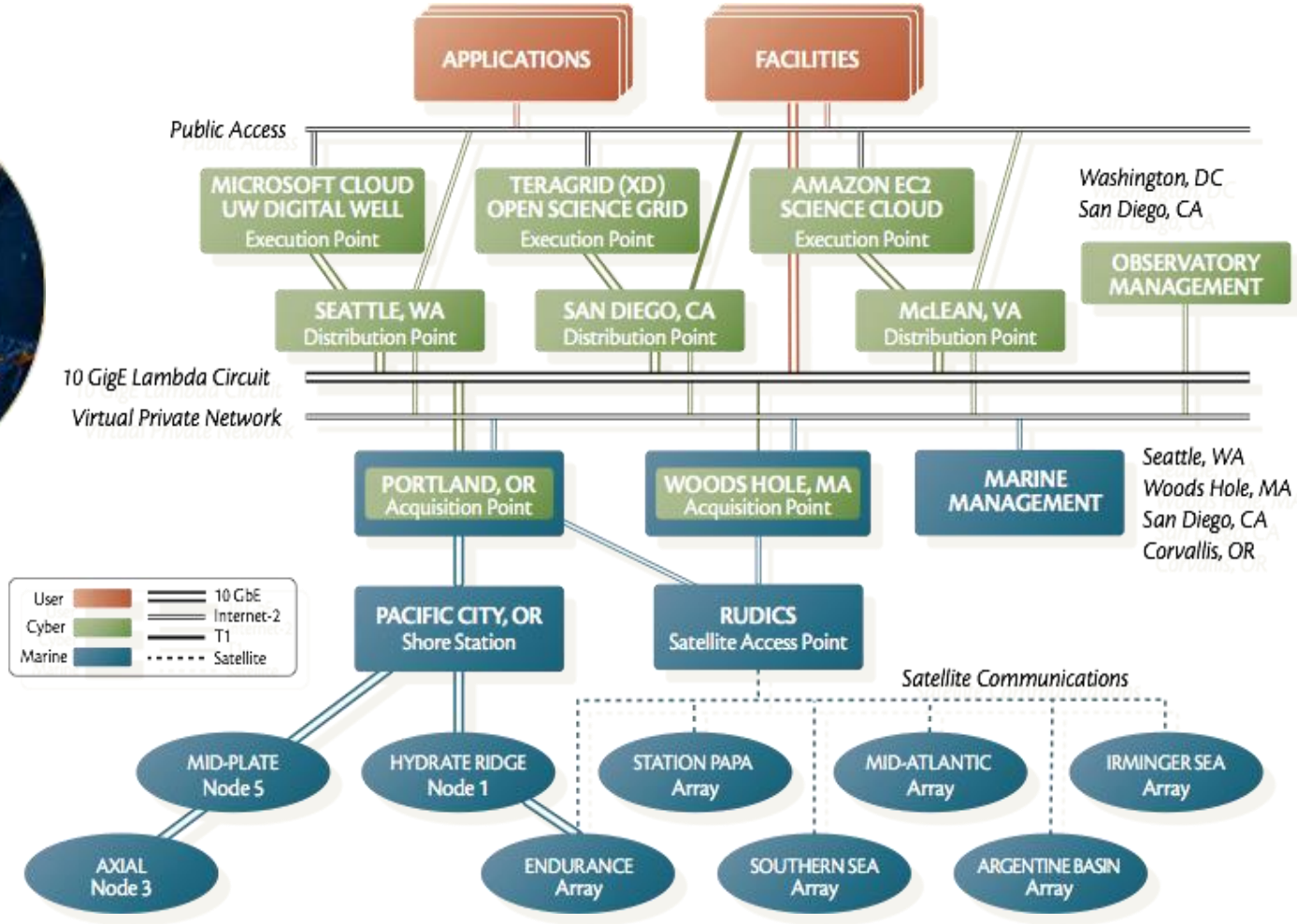


$400M over five years for construction

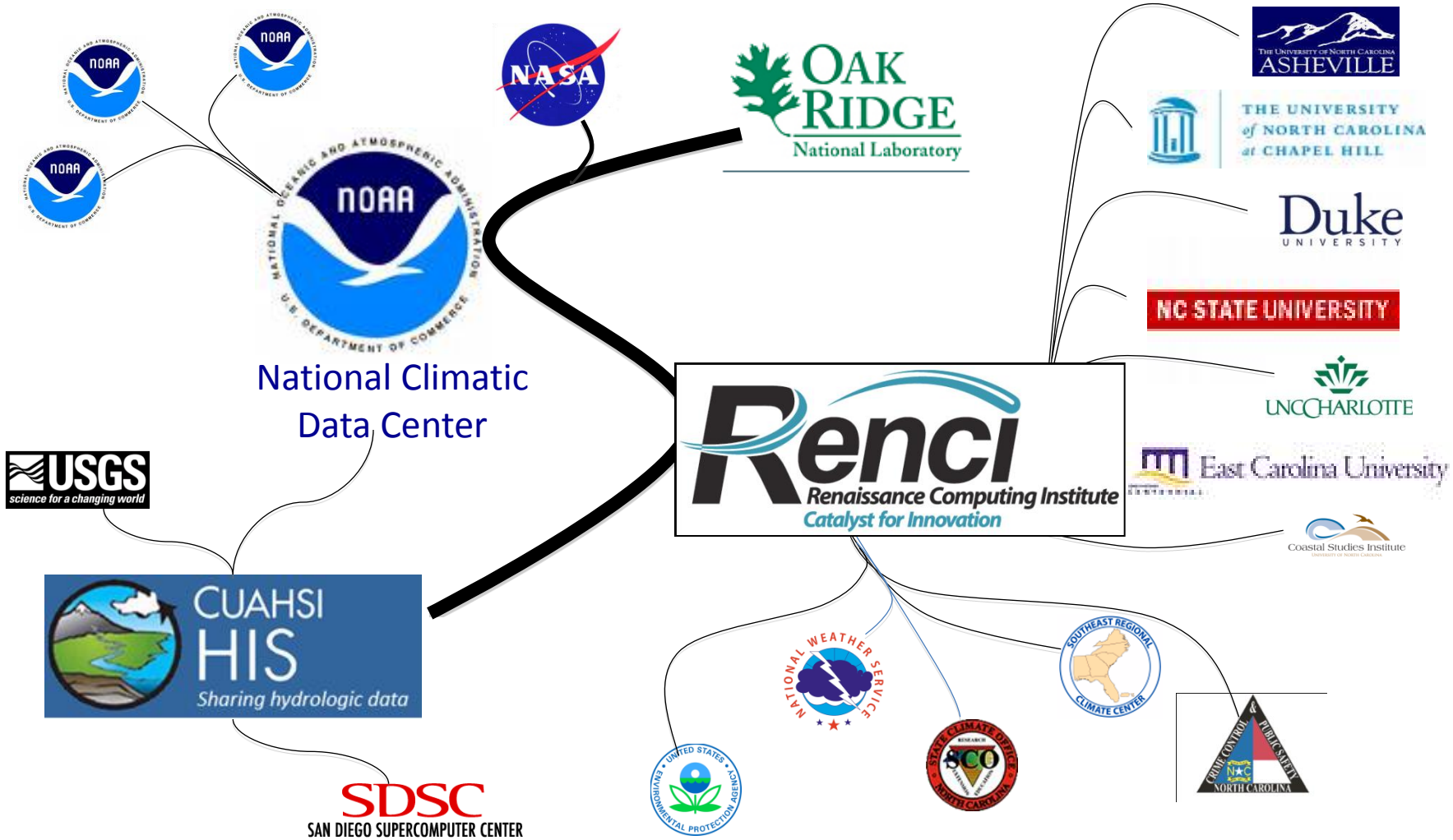
Ocean Observatories Initiative



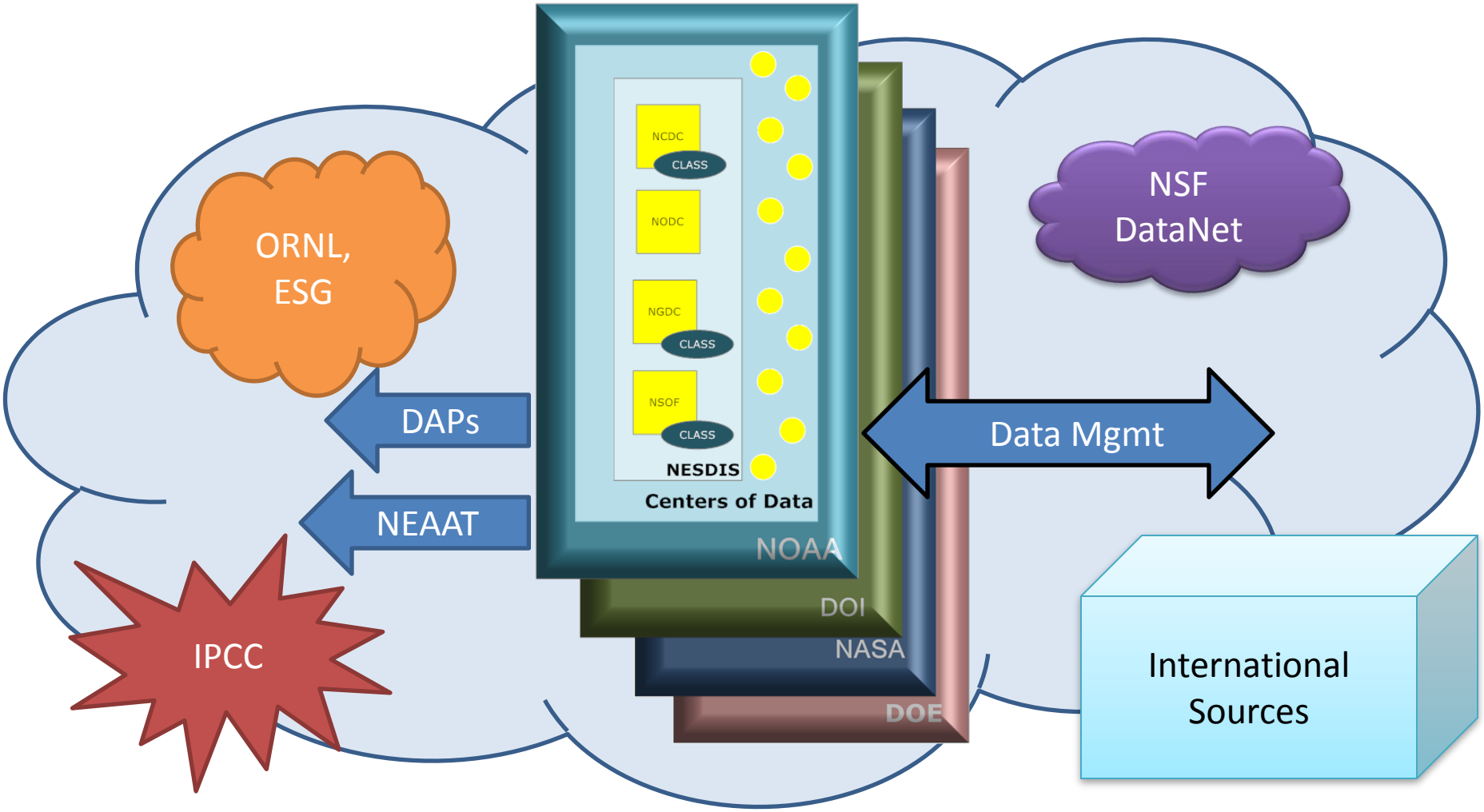
Network Deployment



Collaborations and Datagrids



NOAA's Data Centers Will Function in a Wider Information Landscape



Sustainability

DataDirect™
NETWORKS

Web Object Scaler

WOS

WOS and IRODS Data Grid

Dave Fellinger
dfellinger@ddn.com

A match made in…

iRODS

- Ease of Access
  - 48 clients
- Secure access controls
- Policy based data management

WOS

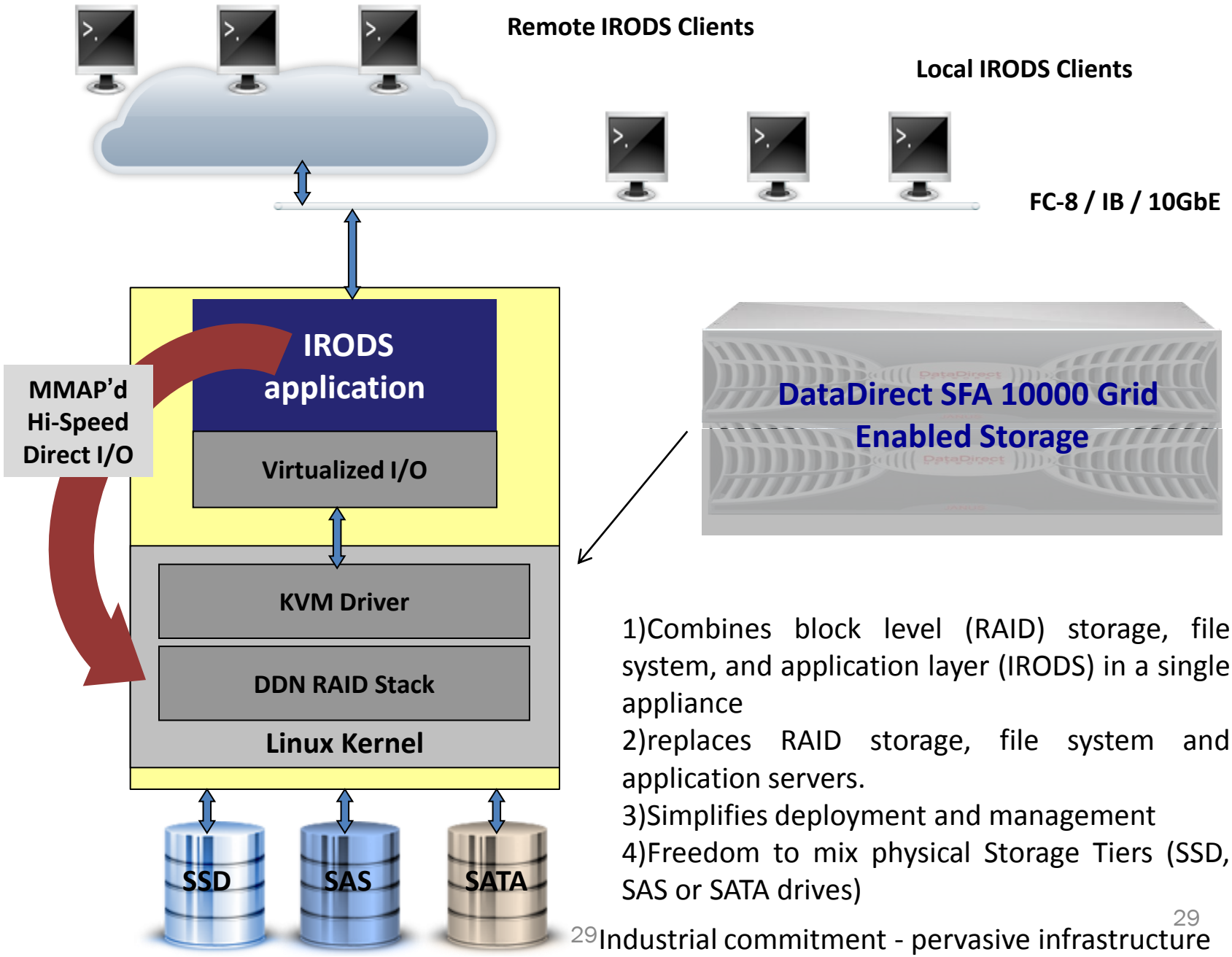
- Simplicity
- Scalability
- Reliability with Replication
- Policy based deterministic data placement and redundancy

iRODS Integrated Rule-Oriented Data System + WOS = Managed Collaboration

WOS is the simple solution for IRODS environments

- WOS is a flat, addressable, low latency data structure.
- WOS creates a “trusted” environment with automated replication.
- WOS is not an extents based file system with layers of V-nodes and I-nodes.
- IRODS is the ideal complement to WOS allowing multiple client access and an incorporation of an efficient DB for metadata search activities.

Technological Sustainability



1)Combines block level (RAID) storage, file system, and application layer (IRODS) in a single appliance
2)replaces RAID storage, file system and application servers.
3)Simplifies deployment and management
4)Freedom to mix physical Storage Tiers (SSD, SAS or SATA drives)

29
Industrial commitment - pervasive infrastructure

iRODS at RENCI

Leesa Brieger, Jason Coposky,

Vijay Dantuluri, Kevin Gamiel, Ray Idaszak, Oleg Kapeljushnik, Nassib Nassar, Jason Reilly, Michael Stealey, Lisa Stillwell

https://www.irods.org/index.php/iRODS_User_Group_Meeting_2011

irods@renci

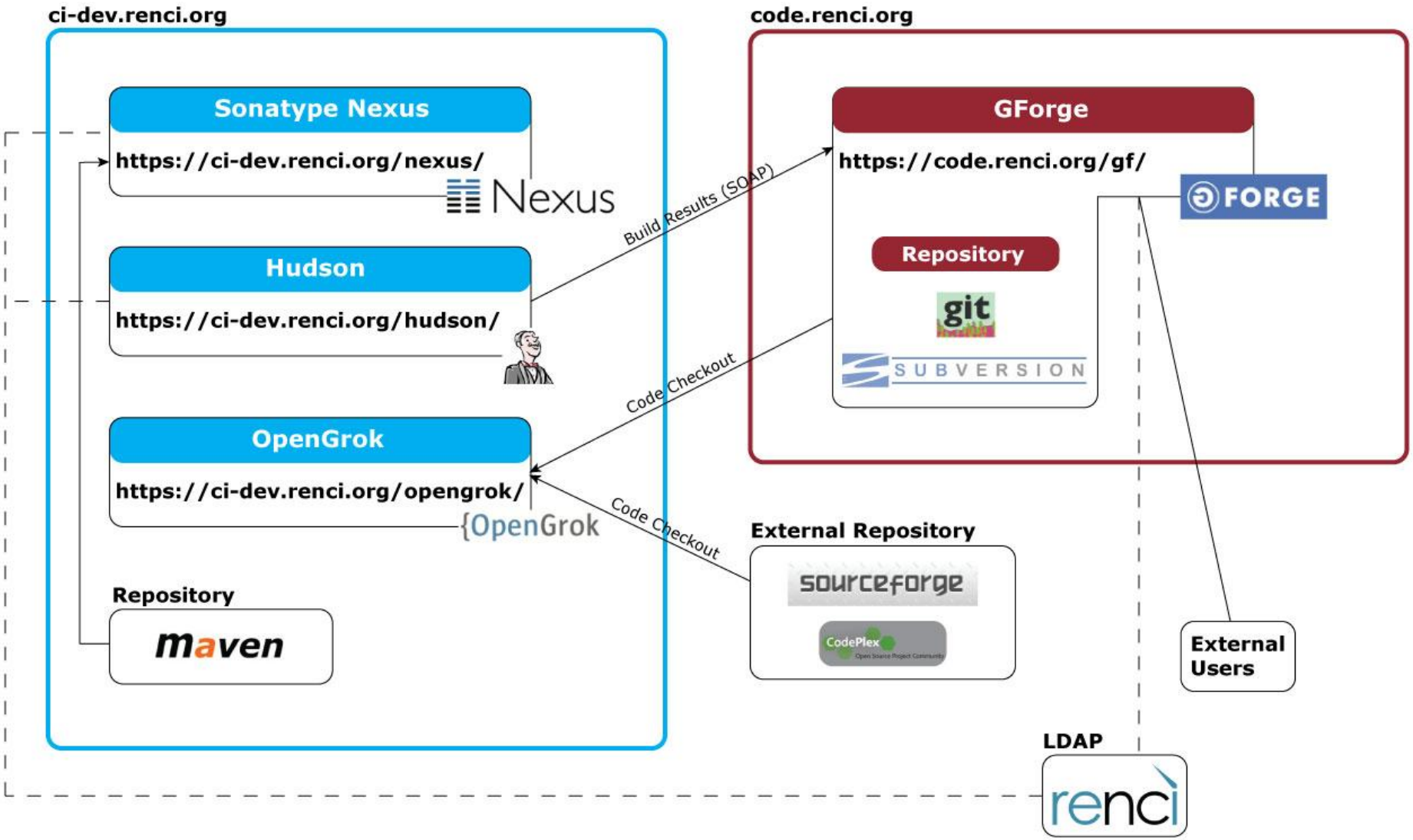
- A new initiative at the Renaissance Computing Institute (RENCI), a research unit of UNC
- An investment by UNC
- A step-up of the collaboration with DICE, already administratively tied to RENCI:
  - DICE-UCSD: Institute of Neural Computing (INC)
  - DICE-UNC: RENCI and the School of Information and Library Science (SILS)
- Stepping toward long-term sustainability

Collaborative Development Environment

- Git – distributed revision control system
- GForge – project and software development management system:
  - hosting & version control
  - bug-tracking
  - messaging
- Hudson – continuous integration environment: incremental quality control
- Nexus – Maven repository that tracks dependencies and bundles for check-out (Java)

Infrastructure Overview

Supports community-based software development



Common Infrastructure

- Data grids to support collaborations
  - Shared collections
- Data processing pipelines to support data mining
  - Data-driven science based on data mining
  - Detect significant events
  - Generate standard products and statistics (vary input)
- Digital libraries to support publication within a discipline
  - Provide services for use of standard data products
- Preservation Environments to support reference collections
  - Digital holdings on which future research is based

Cloud Storage

amazon web services™

MOSSO the hosting cloud

GO GRID

Institutional Repositories

Carolina Digital Repository

Texas Digital Library

Federal Repositories

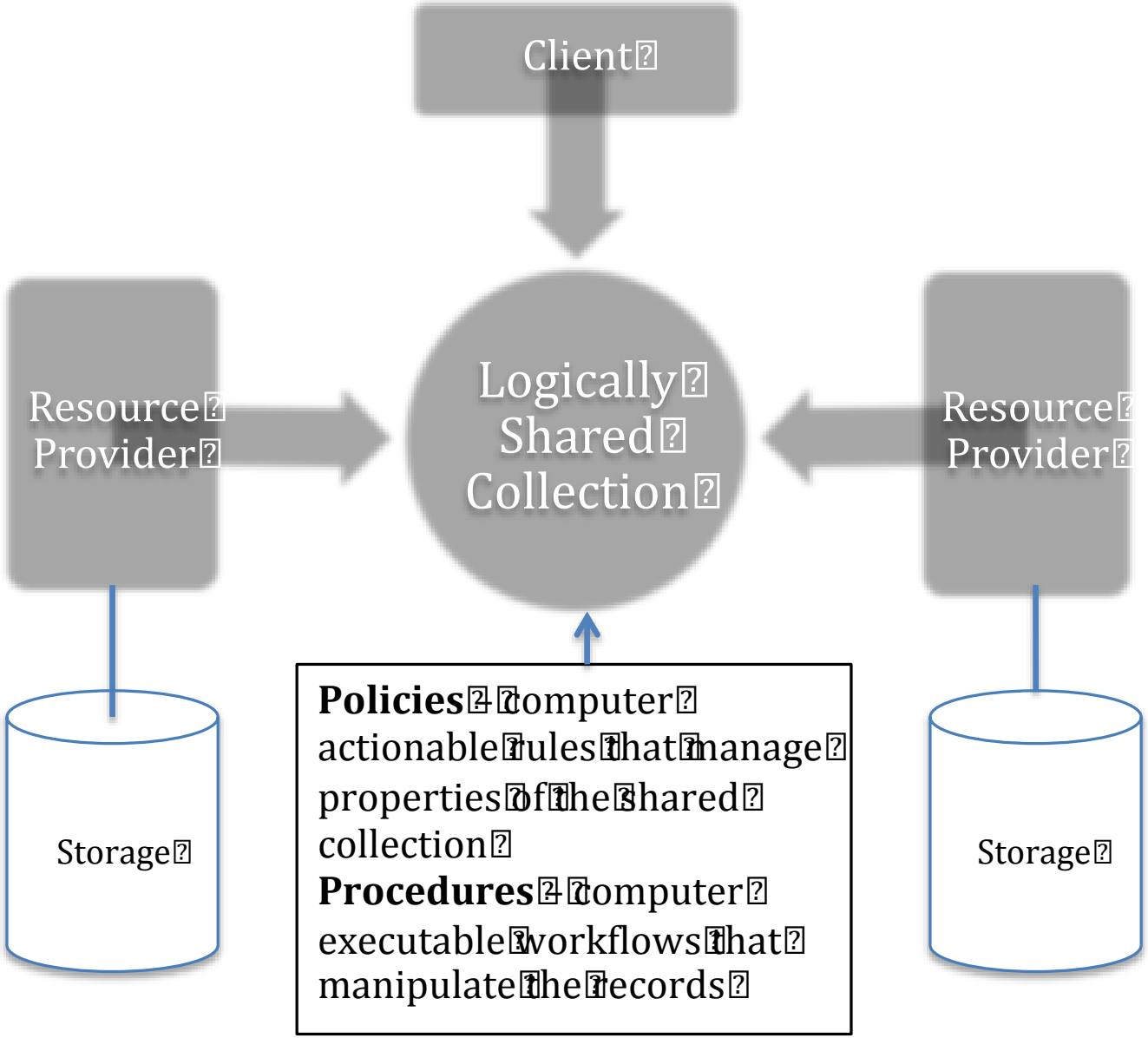
National Climatic Data Center

National Optical Astronomy Observatory

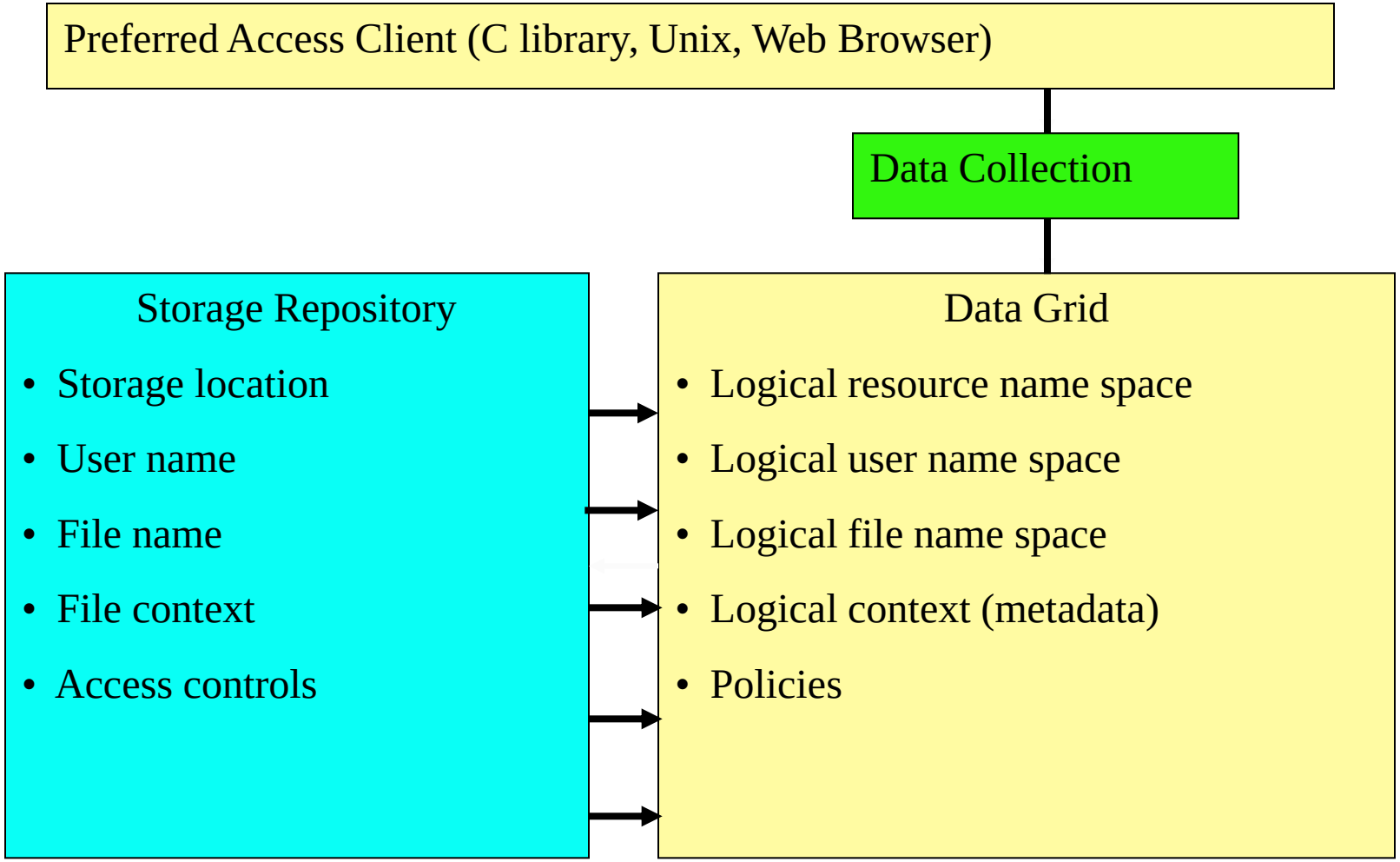
Build Shared Collections

- Enable data processing pipeline that spans multiple institutions
- Enable collaborative research on a shared collection
- Enable formation of a digital library that incorporates material from multiple sources
- Enable creation of a preservation environment that incorporates a deep archive

Policy Based Interoperability



Data Virtualization



Data is organized as a shared collection

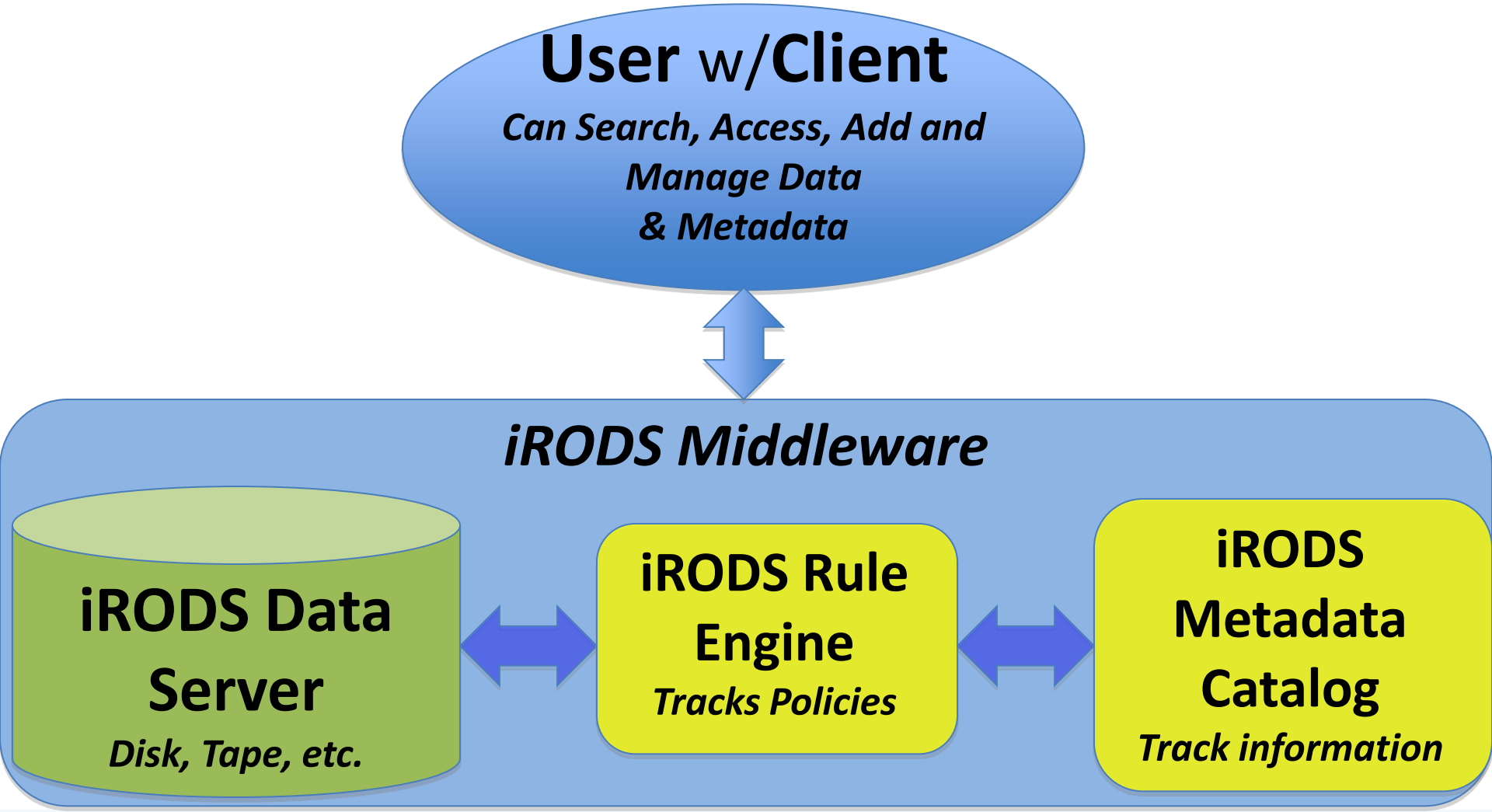
Data Life Cycle

The driving purpose changes at each stage of the data life cycle

Project Collection	Data Grid	Data Processing Pipeline	Digital Library	Reference Collection	Federation
Private	Shared	Analyzed	Published	Preserved	Sustained
Local Policy	Distribution Policy	Service Policy	Description Policy	Representation Policy	Re-purposing Policy

Stages correspond to addition of new policies for a broader community
Virtualize the stages of the data life cycle through policy evolution

Overview of iRODS Architecture



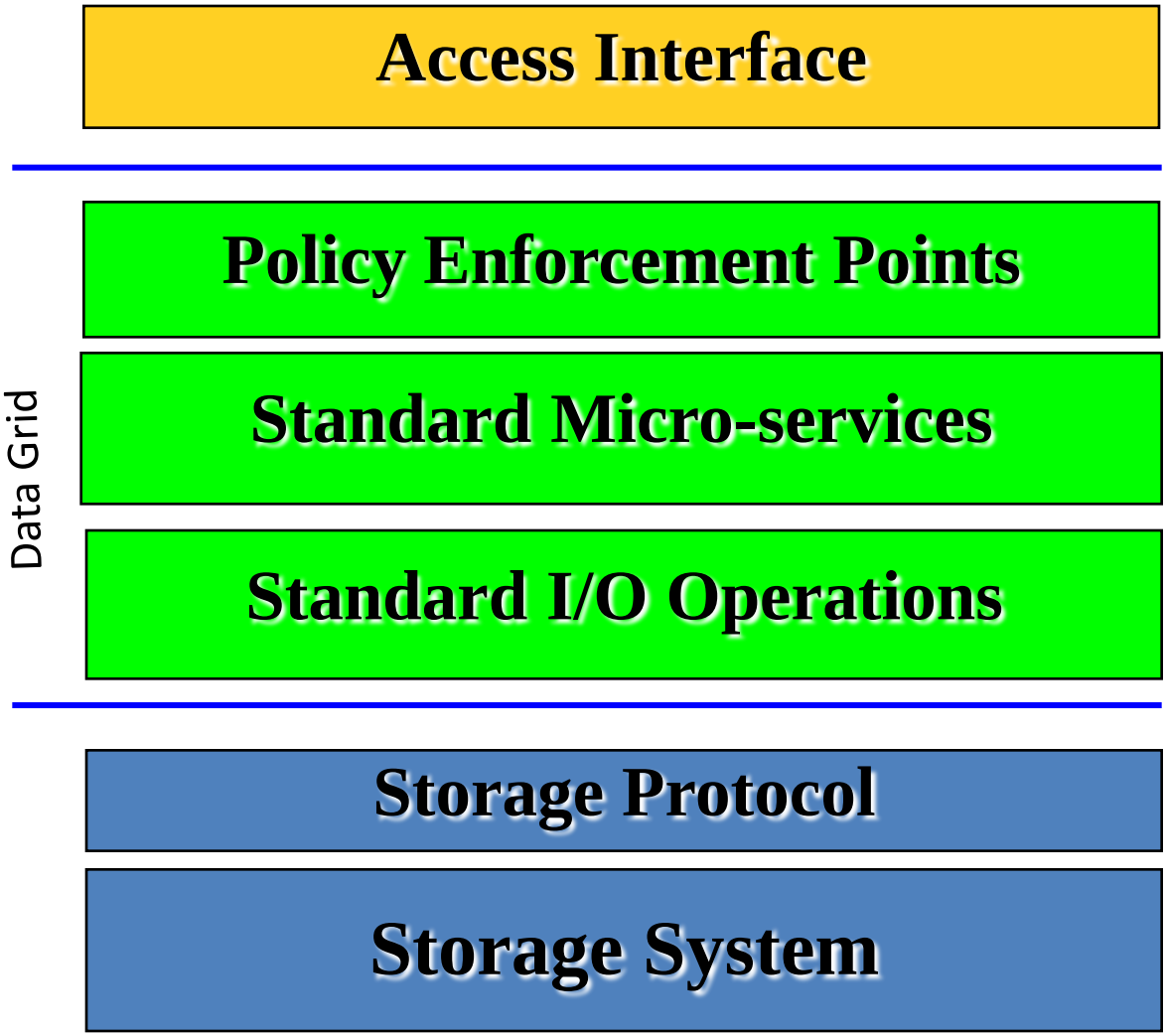
Access distributed data with Web-based Browser or iRODS GUI or Command Line clients.

iRODS - Policy-based Data Management

- Turn policies into computer actionable rules
  - Constrain application of policies by user group, storage resource, file type, file size, processing flag, system property, time dependence
- Compose rules by chaining standard operations
  - Standard operations (micro-services) executed at the remote storage location
- Manage state information as attributes on namespaces:
  - Files / collections /users / resources / rules
- Validate assessment criteria
  - Queries on state information, parsing of audit trails
- Automate administrative functions
  - Minimize labor costs

41

Data Virtualization



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 standard I/O operations to the protocol supported by the storage system

Policy Enforcement Points

- Locations within iRODS framework where policies are checked.
  - Each action may involve multiple policy enforcements points
- Policy enforcement points
  - Pre-action policy (selection of storage location)
  - Policy execution (file deletion control)
  - Post-action policy (derived data products)

Policy Enforcement Points (71)

ACTION
acCreateUser
acDeleteUser
acGetUserbyDN
acTrashPolicy
acAclPolicy
acSetCreateConditions
acDataDeletePolicy
acRenameLocalZone
acSetRescSchemeForCreate
acRescQuotaPolicy
acSetMultiReplPerResc
acSetNumThreads
acVacuum
acSetResourceList
acSetCopyNumber
acVerifyChecksum
acCreateUserZoneCollections
acDeleteUserZoneCollections
acPurgeFiles
acRegisterData
acGetIcatResults
acSetPublicUserPolicy
acCreateDefaultCollections
acDeleteDefaultCollections

PRE-ACTION POLICY
acPreProcForCreateUser
acPreProcForDeleteUser
acPreProcForModifyUser
acPreProcForModifyUserGroup
acChkHostAccessControl
acPreProcForCollCreate
acPreProcForRmColl
acPreProcForModifyAVUMetadata
acPreProcForModifyCollMeta
acPreProcForModifyDataObjMeta
acPreProcForModifyAccessControl
acPreprocForDataObjOpen
acPreProcForObjRename
acPreProcForCreateResource
acPreProcForDeleteResource
acPreProcForModifyResource
acPreProcForModifyResourceGroup
acPreProcForCreateToken
acPreProcForDeleteToken
acNoChkFilePathPerm
acPreProcForGenQuery
acSetReServerNumProc
acSetVaultPathPolicy

POST-ACTION POLICY
acPostProcForCreateUser
acPostProcForDeleteUser
acPostProcForModifyUser
acPostProcForModifyUserGroup
acPostProcForDelete
acPostProcForCollCreate
acPostProcForRmColl
acPostProcForModifyAVUMetadata
acPostProcForModifyCollMeta
acPostProcForModifyDataObjMeta
acPostProcForModifyAccessControl
acPostProcForOpen
acPostProcForObjRename
acPostProcForCreateResource
acPostProcForDeleteResource
acPostProcForModifyResource
acPostProcForModifyResourceGroup
acPostProcForCreateToken
acPostProcForDeleteToken
acPostProcForFilePathReg
acPostProcForGenQuery
acPostProcForPut
acPostProcForCopy
acPostProcForCreate

Policy Invocation

iput ../src/irm.c **checks 10 policy hooks**

srbbrick14:10900:ApplyRule#116:: acChkHostAccessControl
srbbrick14:10900:GotRule#117:: acChkHostAccessControl
srbbrick14:10900:ApplyRule#118:: acSetPublicUserPolicy
srbbrick14:10900:GotRule#119:: acSetPublicUserPolicy
srbbrick14:10900:ApplyRule#120:: acAclPolicy
srbbrick14:10900:GotRule#121:: acAclPolicy
srbbrick14:10900:ApplyRule#122:: acSetRescSchemeForCreate
srbbrick14:10900:GotRule#123:: acSetRescSchemeForCreate
srbbrick14:10900:execMicroSrvc#124:: msiSetDefaultResc(demoResc,null)
srbbrick14:10900:ApplyRule#125:: acRescQuotaPolicy
srbbrick14:10900:GotRule#126:: acRescQuotaPolicy
srbbrick14:10900:execMicroSrvc#127:: msiSetRescQuotaPolicy(off)

Policy Invocations (Cont.)

srbbrick14:10900:ApplyRule#128:: acSetVaultPathPolicy
srbbrick14:10900:GotRule#129:: acSetVaultPathPolicy
srbbrick14:10900:execMicroSrvc#130:: msiSetGraftPathScheme(no,1)
srbbrick14:10900:ApplyRule#131:: acPreProcForModifyDataObjMeta
srbbrick14:10900:GotRule#132:: acPreProcForModifyDataObjMeta
srbbrick14:10900:ApplyRule#133:: acPostProcForModifyDataObjMeta
srbbrick14:10900:GotRule#134:: acPostProcForModifyDataObjMeta
srbbrick14:10900:ApplyRule#135:: acPostProcForCreate
srbbrick14:10900:GotRule#136:: acPostProcForCreate
srbbrick14:10900:ApplyRule#137:: acPostProcForPut
srbbrick14:10900:GotRule#138:: acPostProcForPut
srbbrick14:10900:GotRule#139:: acPostProcForPut
srbbrick14:10900:GotRule#140:: acPostProcForPut

Data Grid Clients (48)

API	Client	Developer	Language
Browser			
	DCAPE	UNC	
	iExplore	RENCI-Oleg	C++
	JUX	IN2P3	Jargon
	Peta Web browser	PetaShare	
	iDrop web browser	Mike Conway	Java
	Davis web interface	ARCS	
	Rich web client	Lisa Stillwell - RENCI	
Digital Library			
	Akubra/iRODS	DICE	Jargon
	Dspace	MIT	
	Fedora on Fuse	IN2P3	FUSE
	Fedora/iRODS module	DICE	Jargon
	Islandora	DICE	Jargon
	Curators Workbench	CDR-UNC-CH	Jargon
File System			
	Davis - Webdav	ARCS	Jargon
	Dropbox / **iDrop**	DICE-Mike Conway	Jargon
	FUSE	IN2P3, DICE,	FUSE
	FUSE optimization	PetaShare	FUSE
	OpenDAP	ARCS	
	PetaFS (Fuse)	Petashare - LSU	
	Petashell (Parrot)	PetaShare	

iRODS Clients (Cont.)

Grid			
	GridFTP - Griffin	ARCS	
	Jsaga	IN2P3	Jargon
	Parrot	UND - Doug Thain	
	SRM	Academia Sinica	
	Saga	KEK	
I/O Libraries			
	PRODS - PHP	Renci - Lisa Stillwell	
	C API	DICE-Mike Wan	C
	C I/O library	DICE-Wayne Schroeder	C
	Fortran	Schroeder	C
	Eclipse file system	CDR - UNC-CH	Jargon
	Jargon	DICE-Mike Conway	Jargon
	Pyrods - Python	SHAMAN-Jerome Fusillier	Python
Portal			
	EnginFrame	NICE / RENCI	Jargon
	Petashare Portal	LSU	Jargon
Tools			
	Archive tools-NOAO	NOAO	
	Big Board visualization	RENCI	
	iFile	GA Tech	
	i-commands	DICE	
	Pcommands	PetaShare	
	Resource Monitoring	IN2P3	
	Sync-package	Academica Sinica	
	URSpace	Teldap - Academica Sinica	
Web Service			
	VOSpace	IVOA	
	Shibboleth	King's College	
Workflows			
	Kepler - actor	DICE	Jargon
	Stork - interoperability	LSU	
	Workflow Virtualization	LSU	
	Taverna - actor	RENCI	

Rules

Distributed Rule Engine

Distributed Rule Base – core.irb

Computer Actionable Policies

- Retention, disposition, distribution, arrangement
- Authenticity, provenance, description
- Integrity, replication, synchronization
- Deletion, trash cans, versioning
- Archiving, staging, caching
- Authentication, authorization, redaction
- Access, approval, IRB, audit trails, report generation
- Assessment criteria, validation
- Derived data product generation, format parsing
- Federation

KEK Paper

IRODS in an Neutrino Experiment

Adil Hasan

for

Francesca Di Lodovico (QMUL), Yoshimi Iida (KEK), Takashi Sasaki (KEK)

https://www.irods.org/index.php/iRODS_User_Group_Meeting_2011

iRODS in an Neutrino Experiment

- Tokai to Kamioka data grid in Japan
  - Provide access to global collaborators
  - Must aggregate files for storage in HPSS in 1-GB containers
  - File sizes ranged from kiloBytes to MegaBytes
- Created policies to:
  - Automate bundling of files
  - Replicate containers into HPSS
  - Purge cache and backup resources

Rule to Bundle Files

```
acKEKBundle(*collPath, *bundlePath, *cacheRes, *compRes, *archive,
*threshold)||
        msiCheckCollSize(*collPath, *cacheRes, *threshold,
                *aboveThreshold, *status)##
        ifExec(*aboveThreshold == 1,
                msiWriteRodsLog("Creating bundle", *status)##
                msiPhyBundleColl(*collPath, *compRes,*status)##
                msiWriteRodsLog("Finished bundling, starting to
replicate",
                        *status)##
                msiCollRepl(*bundlePath,
verifyChksum++++backupRescName                    =*archive,
*status)##
                msiWriteRodsLog("Finished replicating bundle", *status),
        nop##nop##nop##nop##nop, nop, nop) |nop##nop
```

Micro-Services

Functional Units for Building Procedures

Micro-Services

- Functions written in C
- Provided with the iRODS server code
- Provide:
  - Standard operations
  - Queries on metadata catalog
  - Interaction with web services
  - Invocation of external applications
  - Workflow constructs (loops, conditionals, exit)
  - Remote and delayed execution control

Micro-services - How many are needed?

print_hello_arg
msiVacuum
msiQuota
msiGoodFailure
msiSetResource
msiCheckPermission
msiCheckOwner
msiCreateUser
msiCreateCollByAdmin
msiSendMail
recover_print_hello
msiCommit
msiRollback
msiDeleteCollByAdmin
msiDeleteUser
msiAddUserToGroup
msiSetDefaultResc
msiSetRescSortScheme
msiSysReplDataObj
msiStageDataObj
msiSetDataObjPreferredResc
msiSetDataObjAvoidResc
msiSortDataObj
msiSysChksumDataObj
msiSetDataTypeFromExt
msiSetNoDirectRescInp
msiSetNumThreads
msiDeleteDisallowed
msiOprDisallowed

msiDataObjCreate
msiDataObjOpen
msiDataObjClose
msiDataObjLseek
msiDataObjRead
msiDataObjWrite
msiDataObjUnlink
msiDataObjRepl
msiDataObjCopy
msiExtractNaraMetadata
msiSetMultiReplPerResc
msiAdmChangeCoreIRB
msiAdmShowIRB
msiAdmShowDVM
msiAdmShowFNM
msiAdmAppendToTopOfCoreIRB
msiAdmClearAppRuleStruct
msiAdmAddAppRuleStruct
msiGetObjType
msiAssociateKeyValuePairsToObj
msiExtractTemplateMDFromBuf
msiReadMDTemplateIntoTagStruct
msiDataObjPut
msiDataObjGet
msiDataObjChksum
msiDataObjPhymv
msiDataObjRename
msiDataObjTrim
msiCollCreate

msiRmColl
msiReplColl
msiCollRepl
msiPhyPathReg
msiObjStat
msiDataObjRsync
msiFreeBuffer
msiNoChkFilePathPerm
msiNoTrashCan
msiSetPublicUserOpr
whileExec
forExec
delayExec
remoteExec
forEachExec
msiSleep
writeString
writeLine
writeBytesBuf
writePosInt
writeKeyValPairs
msiGetDiffTime
msiGetSystemTime
msiHumanToSystemTime
msiStrToBytesBuf
msiApplyDCMetadataTemplate
msiListEnabledMS
msiSendStdoutAsEmail
msiPrintKeyValPair

msiGetValByKey
msiAddKeyVal
assign
ifExec
break
applyAllRules
msiExecStrCondQuery
msiExecStrCondQueryWithOptions
msiExecGenQuery
msiMakeQuery
msiMakeGenQuery
msiGetMoreRows
msiAddSelectFieldToGenQuery
msiAddConditionToGenQuery
msiPrintGenQueryOutToBuffer
msiExecCmd
msiSetGraftPathScheme
msiSetRandomScheme
msiCheckHostAccessControl
msiGetIcatTime
msiGetTaggedValueFromString
msiXmsgServerConnect
msiXmsgCreateStream
msiCreateXmsgInp
msiSendXmsg
msiRcvXmsg
msiXmsgServerDisConnect
msiString2KeyValPair
msiStrArray2String
msiRdaToStdout

Micro-services (229)

msiRdaToDataObj
msiRdaNoResults
msiRdaCommit
msiAW1
msiRdaRollback
msiRenameLocalZone
msiRenameCollection
msiAclPolicy
msiRemoveKeyValuePairsFromObj
msiDataObjPutWithOptions
msiDataObjReplWithOptions
msiDataObjChksumWithOptions
msiDataObjGetWithOptions
msiSetReServerNumProc
msiGetStdoutInExecCmdOut
msiGetStderrInExecCmdOut
msiAddKeyValToMspStr
msiPrintGenQueryInp
msiTarFileExtract
msiTarFileCreate
msiPhyBundleColl
msiWriteRodsLog
msiServerMonPerf
msiFlushMonStat
msiDigestMonStat
msiSplitPath
msiGetSessionVarValue
msiAutoReplicateService
msiDataObjAutoMove
msiGetContInxFromGenQueryOut
msiSetACL
msiSetRescQuotaPolicy
msiPropertiesNew
msiPropertiesClear
msiPropertiesClone
msiPropertiesAdd
msiPropertiesRemove
msiPropertiesGet
msiPropertiesSet
msiPropertiesExists
msiPropertiesToString
msiPropertiesFromString
msiRecursiveCollCopy
msiGetDataObjACL
msiGetCollectionACL
msiGetDataObjAVUs
msiGetDataObjPSmeta
msiGetCollectionPSmeta
msiGetDataObjAIP
msiLoadMetadataFromDataObj
msiExportRecursiveCollMeta
msiCopyAVUMetadata
msiGetUserInfo
msiGetUserACL
msiCreateUserAccountsFromDataObj
msiLoadUserModsFromDataObj
msiDeleteUsersFromDataObj
msiLoadACLFromDataObj
msiGetAuditTrailInfoByUserID
msiGetAuditTrailInfoByObjectID
msiGetAuditTrailInfoByActionID
msiGetAuditTrailInfoByKeywords
msiGetAuditTrailInfoByTimeStamp
msiSetDataType
msiGuessDataType
msiMergeDataCopies
msiIsColl
msiIsData
msiGetCollectionContentsReport
msiGetCollectionSize
msiStructFileBundle
msiCollectionSpider
msiFlagDataObjwithAVU
msiFlagInfectedObjs

State Information - How Many?

ZONE_ID
ZONE_NAME
ZONE_TYPE
ZONE_CONNECTION
ZONE_COMMENT
ZONE_CREATE_TIME
ZONE_MODIFY_TIME
USER_ID
USER_NAME
USER_TYPE
USER_ZONE
USER_DN
USER_INFO
USER_COMMENT
USER_CREATE_TIME
USER_MODIFY_TIME
RESC_ID
RESC_NAME
RESC_ZONE_NAME
RESC_TYPE_NAME
RESC_CLASS_NAME
RESC_LOC
RESC_VAULT_PATH
RESC_FREE_SPACE
RESC_FREE_SPACE_TIME

RESC_INFO
RESC_COMMENT
RESC_CREATE_TIME
RESC_MODIFY_TIME
RESC_STATUS
DATA_ID
DATA_COLL_ID
DATA_NAME
DATA_REPL_NUM
DATA_VERSION
DATA_TYPE_NAME
DATA_SIZE
DATA_RESC_GROUP_NAME
DATA_RESC_NAME
DATA_PATH
DATA_OWNER_NAME
DATA_OWNER_ZONE
DATA_REPL_STATUS
DATA_STATUS
DATA_CHECKSUM
DATA_EXPIRY
DATA_MAP_ID
DATA_COMMENTS
DATA_CREATE_TIME
DATA_MODIFY_TIME

DATA_ACCESS_TYPE
DATA_ACCESS_NAME
DATA_TOKEN_NAMESPACE
DATA_ACCESS_USER_ID
DATA_ACCESS_DATA_ID
COLL_ID
COLL_NAME
COLL_PARENT_NAME
COLL_OWNER_NAME
COLL_OWNER_ZONE
COLL_MAP_ID
COLL_INHERITANCE
COLL_COMMENTS
COLL_CREATE_TIME
COLL_MODIFY_TIME
COLL_ACCESS_TYPE
COLL_ACCESS_NAME
COLL_TOKEN_NAMESPACE
COLL_ACCESS_USER_ID
COLL_ACCESS_COLL_ID
META_DATA_ATTR_NAME
META_DATA_ATTR_VALUE
META_DATA_ATTR_UNITS
META_DATA_ATTR_ID
META_DATA_CREATE_TIME

State Information (112)

META_DATA_MODIFY_TIME
META_COLL_ATTR_NAME
META_COLL_ATTR_VALUE
META_COLL_ATTR_UNITS
META_COLL_ATTR_ID
META_NAMESPACE_COLL
META_NAMESPACE_DATA
META_NAMESPACE_RESC
META_NAMESPACE_USER
META_RESC_ATTR_NAME
META_RESC_ATTR_VALUE
META_RESC_ATTR_UNITS
META_RESC_ATTR_ID
META_USER_ATTR_NAME
META_USER_ATTR_VALUE
META_USER_ATTR_UNITS
META_USER_ATTR_ID
RESC_GROUP_RESC_ID
RESC_GROUP_NAME
USER_GROUP_ID
USER_GROUP_NAME
RULE_EXEC_ID
RULE_EXEC_NAME
RULE_EXEC_REI_FILE_PATH
RULE_EXEC_USER_NAME
RULE_EXEC_ADDRESS
RULE_EXEC_TIME
RULE_EXEC_FREQUENCY
RULE_EXEC_PRIORITY
RULE_EXEC_ESTIMATED_EXE_TIME
RULE_EXEC_NOTIFICATION_ADDR
RULE_EXEC_LAST_EXE_TIME
RULE_EXEC_STATUS
TOKEN_NAMESPACE
TOKEN_ID
TOKEN_NAME
TOKEN_VALUE
TOKEN_VALUE2
TOKEN_VALUE3
TOKEN_COMMENT
AUDIT_OBJ_ID
AUDIT_USER_ID
AUDIT_ACTION_ID
AUDIT_COMMENT
AUDIT_CREATE_TIME
AUDIT_MODIFY_TIME
SL_HOST_NAME
SL_RESC_NAME
SL_CPU_USED
SL_MEM_USED
SL_SWAP_USED
SL_RUNQ_LOAD
SL_DISK_SPACE
SL_NET_INPUT
SL_NET_OUTPUT
SL_CREATE_TIME
SLD_RESC_NAME
SLD_LOAD_FACTOR
SLD_CREATE_TIME

NASA Paper

Federated Observational and Simulation Data in the NASA Center for Climate Simulation Data Management System Project

John L. Schnase, Glenn Tamkin,

David Fladung, Scott Sinno, and Roger Gill

https://www.irods.org/index.php/iRODS_User_Group_Meeting_2011

NASA Center for Climate Simulation

- Observational data
  - MODIS data set, 1 PB, 54 million files, 300 million attributes
  - Created script to control registration of MODIS data into iRODS and an associated database to monitor the submission process
  - Provided access to data products
- Simulation data
  - Added auditing extensions to track status of products
  - New lookup and historical tables
  - Added policy enforcement points

Storage System Drivers

Compound Resource

Supported Storage Systems

- File systems - Windows, Linux, Mac
- Tape archives - HPSS, Sam-QFS
- Repositories - Flickr, Web sites
- Cloud storage- Amazon S3, EC2
- Relational database - PostgreSQL, Oracle, mySQL
- Table driven resources

WOS Driver

- Implemented as a compound resource with a unix disk cache in front of the “get – put – delete” WOS interface
- Create WOS resource type
  - iadmin at resc_type wos
- Create compound resource
  - iadmin mkresc wosResc wos compound “IP-address-iRODS-server” “IP-address-WOS-Resource”/”WOSPolicy-file”

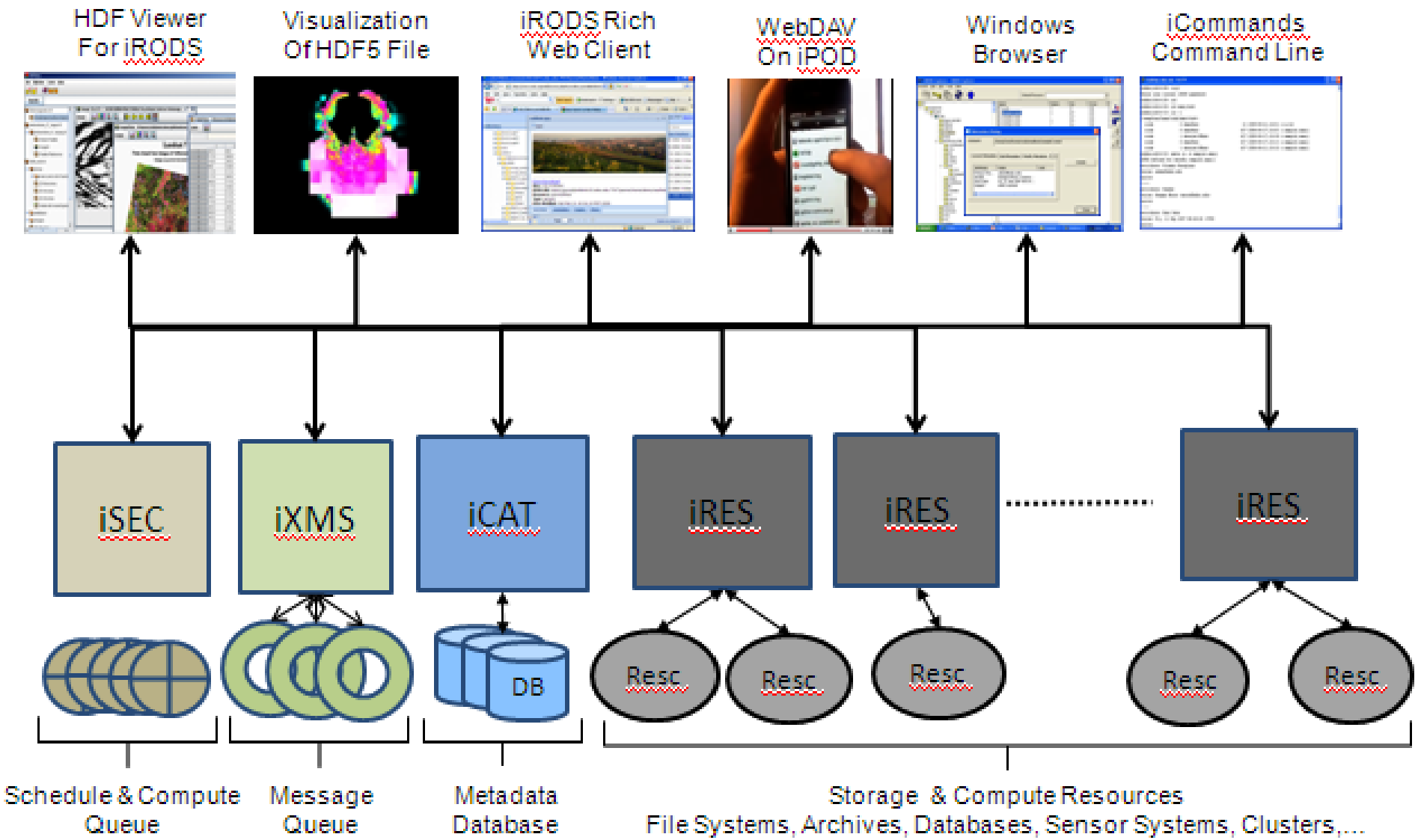
HPSS Drivers

- Implemented as compound resources
- Native HPSS driver
  - Supports parallel I/O
- Universal Mass Storage System driver
  - Developed by IN2P3
- File system interface to HPSS
  - Integration through GPFS

Control Levels

- Policy controls which sites are allowed to connect to the data grid
- Users are authenticated
  - GSI, Kerberos, Challenge-response
- Access control lists on every file
  - User ACLs, User Group ACLs
- All actions are authorized
  - Condition on the execution of every procedure
- Separate rule base for every storage system
  - Data access must comply with local policy

iRODS Distributed Data Management



Open Source Software

- Community driven software development
  - Focus on features required by user communities
  - Focus on bug-free software
  - Focus on highly reliable software
  - Focus on highly extensible software
  - Approximately 3-4 software releases per year
- Distributed under a BSD license
  - International collaborations on software development
  - IN2P3 (France), SHAMAN (UK), ARCS (Australia), Academia Sinica (Taiwan)

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 2011 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"