ROBIN (RuciO/BIgData Express/SENSE) A Next-Generation High-Performance Data Service Platform

Wenji Wu, Liang Zhang, Qiming Lu, Phil DeMar, Robert Illingworth
{wenji, liangz, qlu, demar, illingwo}@fnal.gov
Fermilab

Joe Mambretti, Se-young Yu, Jim Hao Chen
{j-mambretti, young.yu, jim-chen}@northwestern.edu
Northwstern University

Inder Monga, Xi Yang, Tom Lehman, Chin Guok, John MacAuley,
{imonga, xiyang, tlehman, chin, macauley}@es.net
ESnet

***Abstract*—Increasingly, scientific research discovery depends on the analysis of extremely large amounts of data. This data is gathered through very large scientific instruments, very large numbers of small devices, and mid-scale instruments. These devices generate exponentially increasing volumes of data. Such data is often obtained, analyzed, stored, visualized, and transferred at and among sites world-wide by large collaborations of science communities. Managing and moving extremely large volumes of data across today's networks is a special multidimensional challenge. Also, existing data management and movement services and tools often are inadequate to address all specific requirements. This paper describes the various components of this challenge and how those issues can be addressed by ROBIN, a unique comprehensive set of integrated services designed specifically for managing and moving extremely large amounts of data over long distances, e.g., thousands of miles around the globe. This paper also describes the results of initial experiments using that set of integrated services.**

Keywords—big data, data management, high-performance data transfer, DTN, high-speed networking, science workflow manager, programmable networking

I. INTRODUCTION

Within large-scale science, research discovery involves analysis of extremely large amounts of science data. Typically, that data is created by very large scientific instruments such as particle accelerators, telescopes, photon/x-ray sources, etc., or generated via simulations on HPC systems, or collected from large numbers of sensors [1]. Often, an experiment will require combining data from multiple types of these sources. In most cases, the raw data generated by these various sources needs to be archived, processed into a more suitable format for analysis, and finally made available to scientists.

Large-scale science is also characterized by global collaborations, which share not only in the analysis of experiment data, but also in its processing and archiving storage. Intelligent placement and efficient movement of experiment data across the collaboration, for both processing and analysis purposes, becomes a critical element in the outcome of the experiment. In such global collaborations, management and movement of experiment data normally involves multiple separate administrative domains, including independent end-sites that host subsets of the data, and interconnected network service providers that provide the transit services between the end-sites. The challenges with data management and movement across such multi-domain environments is compounded by the multidimensional nature of the task. Typically, the data management software that controls the cataloging and placement of experiment data/metadata functions independently of the data transfer tool(s) used for end-to-end data transfer, which in turn functions independently from the network services actually transporting the data. There is minimal coordination between the various data management and data transfer functions operating simultaneously at multiple levels. The high-level challenge becomes getting these layered functions to perform optimally as a whole.

This paper describes the many individual components of this challenge. It also describes how those issues can be addressed by a unique comprehensive set of integrated services designed specifically to satisfy the challenges involved with managing and moving extremely large amounts of data world-wide, including the administrative boundary issues inherent in multi-domain environments. The comprehensive approach used here is key. The services include Rucio [2], a data management service widely used by particle physics research communities; BigData Express [3], a schedulable, predictable, and high-performance data transfer service; and SENSE [4], a distributed network resource orchestrator. Each service is comprised of multiple subservices. Together, they combine to provide a high-performance data service platform for the management and movement of experiment data within large-scale science collaborations. Contributions of each service are described.

This paper discusses results of initial experiments using these integrated services. Testing is conducted on a large-scale international testbed extending from the Midwest in the US to the particle physics research center at CERN (Switzerland), host to the Large Hadron Collider (LHC). The LHC generates more data than any other science instrument, amounting to hundreds of petabytes per year, which must be distributed to collaboration sites on a global scale.

II. PRIMARY CHALLENGES AND OUR SOLUTION

As noted, large-scale, high volume, distributed data management and movement challenges have not yet been satisfactorily addressed because of the size and complexity of the task. To date no approach has attempted to provide a comprehensive service solution that incorporates:

- *Services designed for scientists*. A key issue is that the communities using these services are scientists, not network engineers. Scientists should be allowed to use services without having to address low level issues of network configurations, topologies, and protocols.
- *Scientific workflows.* Scientists use workflow managers to conduct their experiments. Consequently, large-scale data movement services should be processes that are integrated into those familiar tools, not low-level independent processes.
- *Data management*. One key high-level service is data management, which allows for identification and organization of files and data across highly distributed locations.
- *Large volume data transfer services.* For large scale data transfers multiple challenges need to be addressed. One challenge is a need for the highest possible performance for the transfers because of the volume of data. Another challenge is the need to address explicit or implicit time constraints determined by scientific applications [3]. Time-constraint categories include *real-time data transfer* (i.e., data transfer is on the critical path of a workflow with a specific deadline), *deadline bound data transfer* (data transfer is not on the critical path but does have an explicit deadline, and *background data transfer*, which has no explicit deadline.
- *Orchestration.* Scientific data transfers require large numbers of resources in many locations across multiple domains to be discovered, claimed, isolated, configured, utilized, monitored, and released after use.
- *Science DMZs.* Science DMZ architectures [6] with dedicated high-performance Data Transfer Nodes (DTNs) have been widely deployed. The hardware devices, software, configurations, and policies of Science DMZ need to be structured and optimized for high-performance data transfer.
- *High performance networking.* International research communities have deployed ultra-scalable networks based on 100-400 Gbps WAN technologies with corresponding LAN deployments in the research end-sites. Advanced techniques for dynamically deploying virtual paths, e.g., AutoGOLE/NSI [5], and ESnet's OSCARS [7] have also been deployed.

To address all these issues simultaneously, *ROBIN* (RuciO/BIgData Express/SENSE): a next-generation high-performance data service platform, has been designed, implemented, and tested. *ROBIN* is an integration of the three primary services, Rucio, BigData Express, and SENSE. Without end-to-end integration and orchestration of these resource components, the individual processes may not only function sub-optimally, they may actually contend with each other. The results are likely to be degraded or at least highly-variable performance.

III. Backgrounds and Related works

A. Rucio

Rucio is an open source science data manager that provides a smart namespace enabling scientists to organize files in datasets and containers, create virtual overlaps, distribute them by scope, and attach metadata. It also provides storage support, connecting existing storage and allowing extensions to additional storage. Rucio enables easy integration into existing applications and workflow managers because it provides open libraries and REST interfaces. Rucio provides authentication and authorization using a variety of techniques or combinations, including usernames and passwords, x509 certificates with proxy support, GSS/Kerberos, SSH public keys, OpenID Connect, and SAML. Monitoring is provided through ElasticSearch and Graphite. Rucio is open source Python code and can be deployed with pip or containers.

In the Rucio framework, data are organized using Data Identifiers (DIDs). Rucio Storage Element (RSE) provides a unified interface to the distributed data centers. Rucio associates actual locations of the DIDs with RSEs. These file locations are commonly called *replicas*. Replication rules are typically defined on the DIDs. A replication rule is a logical abstraction which defines the minimum number of replicas to be available on a list of RSEs.

B. BigData Express (BDE)

Fermilab has developed BigData Express (http://bigdataexpress.fnal.gov) to provide a schedulable, predictable, and high-performance data transfer service for big data science. Essentially, *BigData Express* is a middleware data transfer service with several key features:

- A data-transfer-centric architecture to seamlessly integrate and effectively coordinate computing resources in an end-to-end data transfer loop.
- A distributed peer-to-peer model for data transfer services, making it very flexible for the establishment of data transfer federations.
- A scalable software architecture. BigData Express makes use of MQTT as a message bus to support communication among its components.
- An extensible plugin framework to support different data transfer protocols, including mdtmFTP [8], GridFTP [11], and XrootD.
- An end-to-end data transfer model with fast provisioning of end-to-end network paths for guaranteed QoS. Specifically, the use of an SDN-enabled BigData-Express LANs and SDN-enabled WAN path services to reduce or eliminate network congestion.
- A high-performance data transfer engine. BigData Express uses mdtmFTP as its default data transfer engine. mdtmFTP is specifically designed for optimization of data transfer performance on multicore systems (DTNs).
- A rich set of REST APIs to support scientific workflows.

The BigData Express software is currently deployed and being evaluated at multiple research institutions, including UMD, StarLight, FNAL, KISTI, KSTAR, SURFnet, and Ciena. The BigData Express research team is collaborating with StarLight to deploy BigData Express on various research platforms, including Pacific Research Platform, National Research Platform, and Global Research Platform.

C. SENSE

The Software-defined network for End-to-end Networked Science at Exascale (SENSE) (http://sense.es.net) system is a model-driven, multi-domain automation and orchestration system which enables Layer 2 and Layer 3 services to be provisioned and managed across a variety of network and end-site resources. This provides a basis for innovation of smart network services to accelerate scientific discovery in the era of big data, cloud computing, machine learning and artificial intelligence. An intent based interface allows application workflow agents to express their high-level service requirements and intelligent orchestration and resource control systems allow for custom tailoring of end-to-end networked services based on individual application and infrastructure operator requirements. This allows the science applications to manage the network as a first-class schedulable resource as is the current practice for instruments, compute, and storage systems.

The SENSE services are enabled by some novel technologies, including hierarchical service-resource architecture, unified network and end-site resource modeling and computation, model-based real time control, and application-driven orchestration workflow. Work is underway to integrate real time telemetry data as part of the services computation and orchestration functions.

The SENSE system has been deployed on a mix of development and production resources on ESnet as well as multiple DOE Laboratories and Universities. Work is now underway to integrate this deployment with the AutoGOLE global infrastructure [5]. This will include deployment of SENSE resource control and orchestration functionality at multiple end sites which are part of the AutoGOLE global footprint.

D. High Capacity Networks

Common carriers, U.S. Research and Education networks and private networks have deployed 100 Gbps communication services as core infrastructure. Within data centers, multiple 100 Gbps interconnects are common, and 400 Gbps paths are being implemented. Currently, networks are beginning to implement paths with multiple 100 Gbps channels and 400 Gbps channels. 800 Gbps and Tbps paths are being planned. Many commercial cloud providers have Tbps paths among their primary data centers. Given the capacity of these core paths, key issues are at the network edge.

E. Science DMZs and Data Transfer Nodes(DTNs)

Recognizing that edge issues need to be addressed, the DOE's ESnet has designed a set of equipment, software, configurations, and policies, termed the Science DMZ to optimized high-performance data transfer. This approach has been implemented at DOE research labs. Also, through the National Science Foundation's Campus cyberinfrastructure program, it has been implemented on over 150 university campuses. Larger scale science DMZs have also been implemented as regional, nation, and international research platforms, e.g., the Global Research Platform.

A key resource for a Science DMZ is a Data Transfer Node (DTN), a computer system designed for and dedicated to high performance large scale wide-area data transfer. Most DTNs take advantage of scalability NUMA (non-uniform memory access) architecture, implementing multiple nodes within a server. Each node consists of cores, local memories, and/or I/O devices, usually 100 Gbps NICs.

F. Open Grid Forum (OGF) Network Services Interface (NSI)

In order to facilitate end-to-end dynamic service provisioning, several research and education (R&E) networks joined the Open Grid Forum Network Services Interface working group to develop protocols to support interdomain service requests. The OGF NSI working group published several standards, defining a connection service protocol [13], a document distribution service protocol [14], interdomain authentication and authorization [15], policy [16], path finding [17], and error handling [18]. These standards form the foundation for dynamic multi-domain layer 2 services used in R&E networks today to support large science collaborations such as the Large Hadron Collider (LHC) experiments, and the Legacy Survey of Space and Time (LSST).

IV. ROBIN: A Next-Generation High Performance Data Service Platform

A. Architecture

Figure 1 illustrates the architecture of ROBIN. Conceptually, ROBIN is based on a distributed architecture that can be decomposed into four layers:

- *Scientific applications layers*. Large scientific instruments (e.g., colliders, light sources, and telescopes) generate exponentially increasing volumes of data. To enable scientific discovery, science data must be collected, indexed, archived, shared, and analyzed, typically in a widely distributed, highly collaborative manner.
- *Rucio data management service,* which is responsible for science data organization, management, and access.
- *BigData Express data transfer service,* offering high-performance data transfer service.
- *SENSE smart network service,* which builds dynamic circuits, with service guarantees, across all of the networks within the end-to-end path.

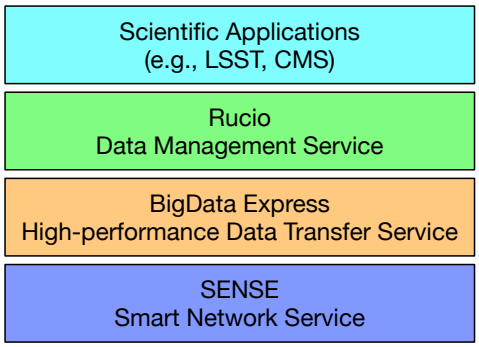

Figure 1 ROBIN layered architecture

Figure 2 illustrates the logic diagram of ROBIN. Each BigData Express site will register as an RSE with the Rucio server. In operations, the Rucio server decides which files to move, groups them in transfer requests, submits the transfer tasks to BigData Express. BigData Express schedules and assigns resources (DTNs, network) to execute the data transfer tasks. In particular, BigData Express intelligently programs network at runtime to suit data transfer requirements. It will call SENSE to provision WAN network paths with guaranteed QoS between sites. Then, BigData Express launches the data transfer tasks, monitors the progress of the transfers, retries in case of errors, and notifies the Rucio server upon completion. The Rucio server also monitors the progress of the transfers and handles failures through retries. Finally, the Rucio server notifies the client upon completion.

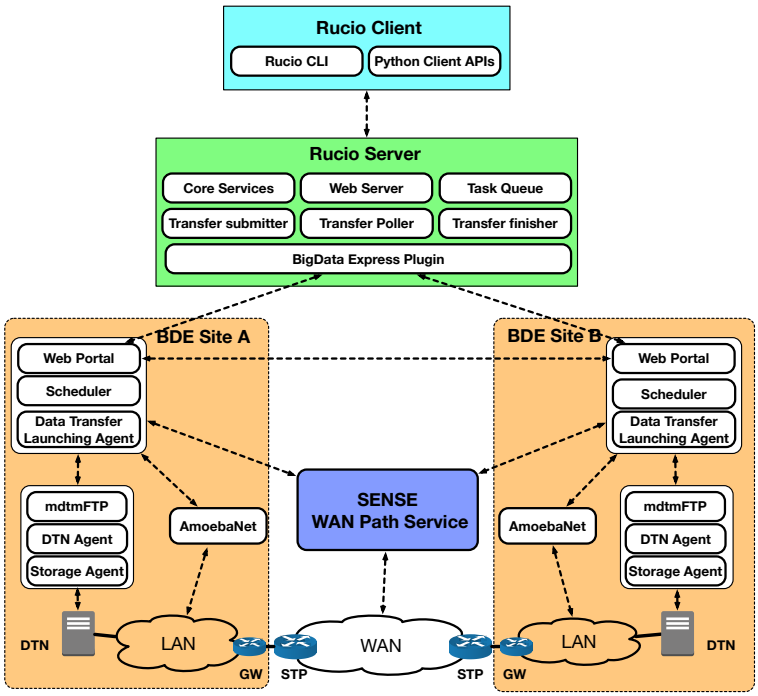

Figure 2 ROBIN logic diagram

B. *Site registration*

In the Rucio framework, a single RSE represents the minimal unit of globally addressable storage. *Site registration* will register a BigData Express site as an RSE with the Rucio server. Typically, the following information need to be passed to the Rucio server for registration:

- The RSE name.
- The information necessary to access the new RSE, including hostname, port, protocol, and local file system path.
- The distance metric between the new RSE and other RSEs. All connected RSEs have the notion of distance. Functional distance is always a non-zero value with increasing integer steps, and zero distance indicates no connection between RSEs. Distance influences the sorting of files when considering sources for transfers.

Rucio is transport protocol agnostic, meaning that the RSEs can accept transfers via multiple protocols. A new protocol *"bde"* is defined to support BDE-based data transfers. A BDE-based RSE is configured to use the "*bde"* protocol.

C. *Rucio/BDE job launching mechanism*

The Rucio server launches BigData Express to establish direct RSE to RSE transfers over the network. Launching the BigData Express data transfer mechanism starts with defining replication rules that request data transfer to an RSE, as well as protect data from being inappropriately deleted. A replication rule requires at least four parameters: *files it effects*, *RSEs where the replica can be placed*, *number of copies, and the lifetime of files*. Rucio's *rule engine* validates available quota, evaluates the RSE, creates transfer tasks for files not existing on the RSE, and creates replica locks to prevent files from being deleted.

BigData Express employs a distributed, peer-to-peer model. A logically centralized BigData Express scheduler coordinates all activities at each BigData Express site. BigData Express schedulers located at different sites negotiate and collaborate to execute data transfer tasks. Typically, the Rucio server sends data transfer tasks to source BigData Express sites. BigData Express provides a rich set of REST API calls for data transfer job submission and monitoring. Once authenticated, the Rucio server can directly call the BigData Express services to perform actions such as querying the resources, submission of a data transfer job, or polling the status of an ongoing job.

Typically, the Rucio/BDE job launching mechanism works as follows.

1) A Rucio client uses Rucio CLI to request replication on a destination RSE.
2) The client sends the replication request to the Rucio server.
3) The Rucio server creates a replication rule for the request and generates the corresponding data transfer tasks. The tasks are temporarily kept in a *task queue*.
4) The Rucio server (*Transfer-submitter daemon*) regularly pulls tasks from the queue. It ranks the sources for each task, selects the protocol "*bde*" for

source and destination RSEs, submits the tasks in groups to BigData Express.

5) Upon request, BigData Express schedules and assigns resources (DTNs, network) to execute the data transfer tasks. In particular, BigData Express calls SENSE to provision WAN paths with guaranteed QoS between sites. (See the next section for details).

6) After the DTNs and the paths have been successfully reserved, BigData Express launches the data transfer tasks, monitors the progress of the tasks, retries in case of errors, and notifies the Rucio server upon completion.

7) The Rucio server (*transfer-poller daemon*) closely monitors the status of the transfers. A failed data transfer will be resubmitted in the *task queue* for retries until the maximum retry limit is reached.

8) The Rucio server (*transfer-finisher daemon*) updates the internal states and notifies the client upon completion.

D. *On-Demand Provisioning of End-to-End Network Paths with Guaranteed QoS*

BigData Express intelligently programs network at run-time to suit data transfer requirements. It dynamically provisions end-to-end network paths with guaranteed QoS between DTNs. An end-to-end network path typically consists of LAN and WAN segments. In the BigData Express end-to-end data transfer model, LAN segments are provisioned and guaranteed by AmoebaNet [3][9], while WAN segments are provisioned via the SENSE service to provide the path between the data source and destination sites.

AmoebaNet applies SDN technologies [10] to provide "Application-aware" network service services in the local network environment. It offers several capabilities to support BigData Express operations. To support network programmability, AmoebaNet provides a rich set of network programming primitives to allow BigData Express to program the local area network at run-time. To support QoS guarantees, AmoebaNet provides two classes of services, priority and best-effort. Priority traffic flows are typically specified with designated rates or bandwidth. AmoebaNet uses QoS queues to differentiate priority and best-effort traffic at each SDN switch. Priority traffic is transmitted first, but metered to enforce rate control. In addition, AmoebaNet supports QoS-based routing and path selection. Finally, AmoebaNet supports fine-grained control of network traffic.

WAN QoS is provisioned and guaranteed by SENSE to reserve bandwidths between Service Termination Points (STPs), where AmoebaNet services end.

Typically, AmoebaNet gateways (GWs) are either logically, or physically connected to WAN STPs. VLAN popping, pushing, and/or swapping operations are performed at AmoebaNet gateways to concatenate WAN and LAN segments.

As illustrated in Figure 3, BigData Express typically performs the following operations to provision an end-to-end network path:

1) Estimate and calculate the DTN-to-DTN traffic matrix, and the related QoS requirements (e.g. throughput, delay).

2) Negotiate and broker network resources to determine the end-to-end rate for the path.

3) Call SENSE service to set up the site-to-site WAN path.

4) Call AmoebaNet at each site to program and configure the LAN paths.

5) Send ICMP ping traffic to verify a contiguous end-to-end network path has been successfully established.

A large data transfer job typically involves many DTNs, and a corresponding large number of data flows. To avoid the necessity of establishing many WAN paths between the source and destination sites, multiple LAN segments can be multiplexed/de-multiplexed to/from a single WAN path, which in turn is configured to support the aggregated bandwidth of its component paths. This strategy helps to reduce burden on WAN path services.

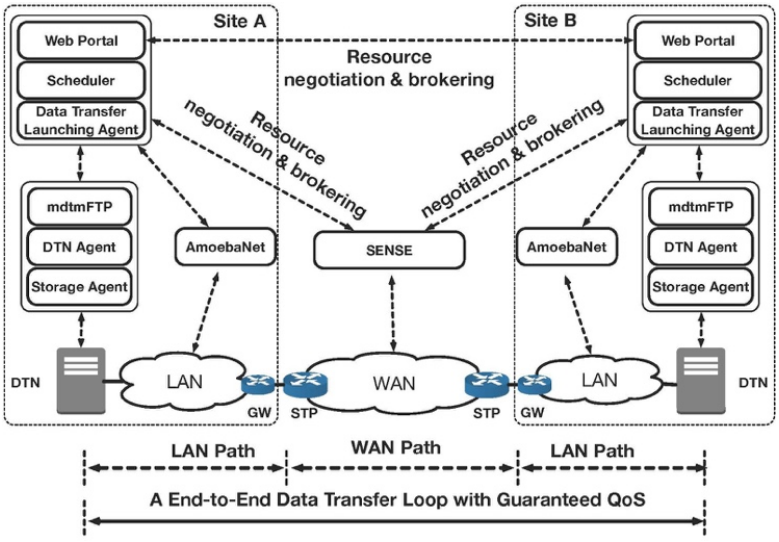

Figure 3 Provisioning of end-to-end path with guaranteed QoS

E. *Authentication and authorization*

Rucio, BigData Express, and SENSE each has its own security model, which has been implemented separately. Table 1 summarizes each system's authentication and authorization methods. In ROBIN, we keep each system's security intact, and execute a logic mapping between them to enforce security at all levels.

We create direct mappings between Rucio accounts and BigData Express accounts with X509 certificate delegation to seamlessly integrate the security models between the two systems:

- A Rucio user gets mapped to a BigData Express user when the user tries to access BigData Express services with a pre-configured credential. BigData Express provides multiple methods for exchanging and verifying credentials with Rucio, including authorized API key and proxy certificate.
- A valid X509 certificate must be presented or provided to initiate a data transfer task with BigData Express. If a Rucio user is authenticated with an X509 certificate, the module will delegate the user certificate and initiate the data transfer task. Or a user can separately request a CILogon [12] issued certificate for the purpose.
- A BigData Express account can be mapped to one or more Rucio user accounts (1:N mapping) in different setups and scenarios.

SENSE service supports OIDC (OpenID Connect [19]) authentication and authorization. Each BigData Express site, acting as a SENSE client, will first verify its identity to the authentication server with pre-configured client credential. A short-term access token will then be issued upon successful authentication that then can be used in subsequent steps to authorize the SENSE services through REST API calls.

Systems	Authentication/Authorization methods
Rucio	Username/password, X509 certificates, Kerberos tickets, SSH-RSA public key
BigData Express	Username/password, X509 certificates
SENSE	Username/password, OIDC

Table 1 Rucio, BigData Express, and SENSE Authentication/Authorization methods

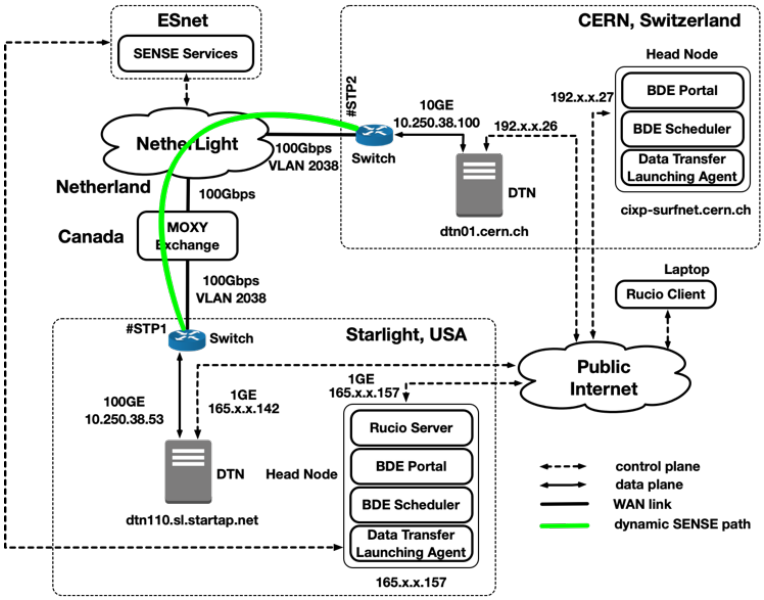

Figure 4 ROBIN Cross-Atlantic Testbed

V. Experiments

In this section, we discuss experiments conducted to evaluate ROBIN. We deployed and evaluated ROBIN on a trans-Atlantic international testbed to demonstrate its key features and capabilities, including site registration, Rucio/BDE job launching mechanism, and on-demand provisioning of WAN paths.

A. An International Testbed

The testbed (Figure 4) consisted of two administratively independent sites - the StarLight International/National Communication Exchange Facility in Chicago and the CERNLight Open Exchange in Switzerland. These two locations were connected with a dedicated layer-2 WAN circuit.

StarLight site:

- DTN: *dtn110.sl.startap.net*, with several Intel NVMe drives for data storage, a 100GE Mellanox NIC for data transfer, and a 1G NIC for control.
- Head node: *165.x.x.157,* with a 1G NIC for control.

CERN site:

- DTN: *dtn01.cern.ch*, with a rotational disk for data storage, a 10GE Mellanox NIC for data transfer, and 1G NIC for control.
- Head node: *cixp-urfnet.cern.ch,* with a 1G NIC for control.

The trans-Atlantic path was provisioned as a dedicated layer 2 WAN circuit between the two sites' border routers, which transits across *StarLight*, *MOXY Exchange*, *NetherLight*, and *CERN*. The circuit can be dynamically set up and torn down using the SENSE service. Because the DTN at each site was deployed using the Science DMZ architecture, directly connected to each site's border router, there was no need to deploy AmoebaNet to provision QoS in the end-site LANs.

A Rucio server was deployed at the head node of StarLight, which managed data in the testbed. BigData Express software ran at both sites. The Rucio server could access BigData Express services from either https://165.xx.157:5000 (BigData Express Web Portal @StarLight), or https://cixp-surfnet.cern.ch:5000 (BigData Express Web Portal @CERN), respectively. The SENSE service was run by ESnet. In addition, a Rucio client ran on a laptop to send requests and queries to the Rucio server.

Separate users were created to access the Rucio server and BigData Express, respectively. The Rucio user was mapped to the BigData Express user with X509 certificate delegation. And each BigData Express site had a pre-configured account to access the ESnet SENSE Orchestrator.

```
$rucio-cmd rucio list-file-replicas test:25g-1.bin

+---------+-----------+------------+-----------+-----------------------------------------------------------------------------------------+
| SCOPE   | NAME      | FILESIZE   | ADLER32   | RSE: REPLICA                                                                            |
|---------+-----------+------------+-----------+-----------------------------------------------------------------------------------------|
| test    | 25g-1.bin | 26.844 GB  | 49576448  | STARLIGHT-SITE: bde://165.124.33.157:5000/165.124.33.142/disk0//25g-1.bin |
+---------+-----------+------------+-----------+-----------------------------------------------------------------------------------------+

$rucio-cmd rucio list-rules test:25g-1.bin

ID                                ACCOUNT    SCOPE:NAME      STATE[OK/REPL/STUCK]    RSE_EXPRESSION      COPIES  EXPIRES (UTC)    CREATED (UTC)
--------------------------------  ---------  --------------  ----------------------  ----------------  --------  ---------------  -------------------
554acd9b7ddb4a319a37308a1285753f  root       test:25g-1.bin  OK[1/0/0]               STARLIGHT-SITE           1                   2020-08-31 04:04:32
```

Figure 5 The replica and the replication rule for *25g-1.bin*

```
$rucio-cmd rucio list-rules test:25g-1.bin

ID                                ACCOUNT    SCOPE:NAME      STATE[OK/REPL/STUCK]    RSE_EXPRESSION      COPIES  EXPIRES (UTC)    CREATED (UTC)
--------------------------------  ---------  --------------  ----------------------  ----------------  --------  ---------------  -------------------
c510e4cd53b44b77b6cdb2c367036766  root       test:25g-1.bin  REPLICATING[0/1/0]      CERN-SITE                1                   2020-08-31 04:09:34
554acd9b7ddb4a319a37308a1285753f  root       test:25g-1.bin  OK[1/0/0]               STARLIGHT-SITE           1                   2020-08-31 04:04:32
```

Figure 6 The replication rule created for the Rucio data replication request

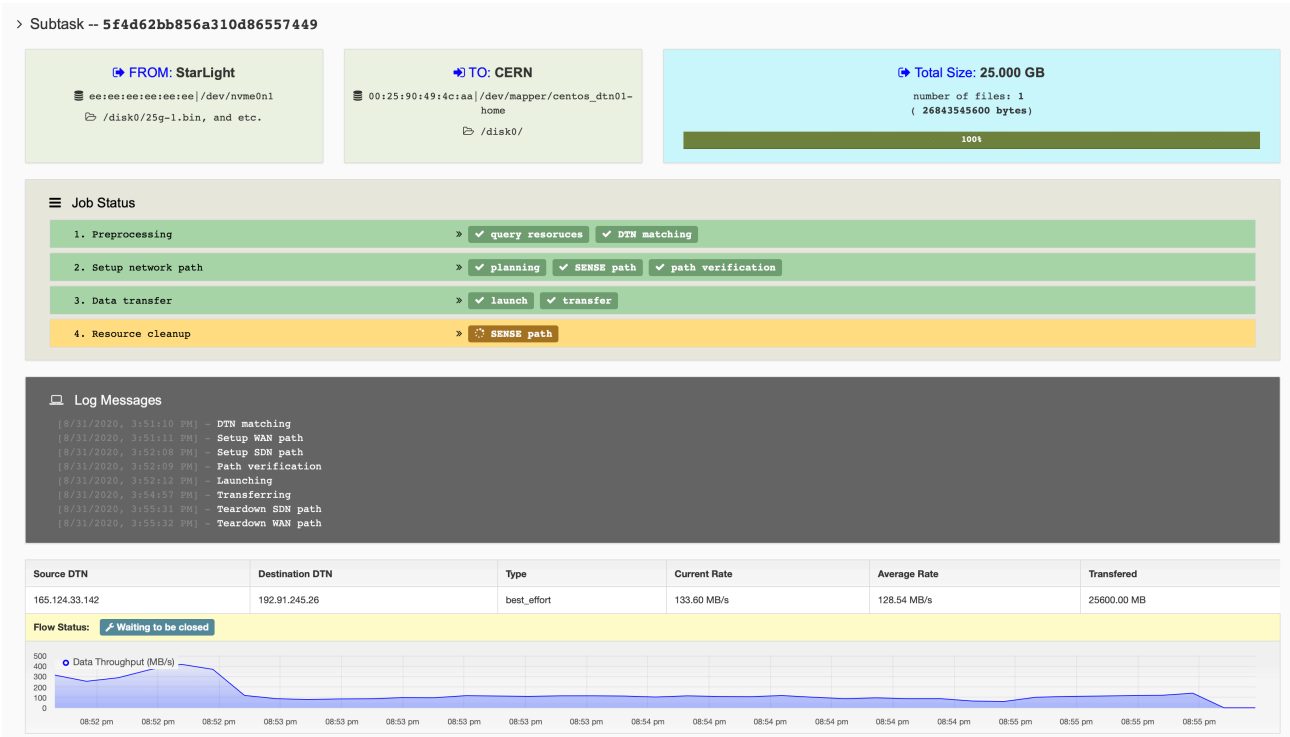

Figure 7 BigData Express data transfer process

B. *Evaluation*

The evaluation was designed to consist of multiple steps to demonstrate and test ROBIN's key features and capabilities. First, we registered each BigData Express site in the testbed as an RSE with the Rucio server. Then, we created an experiment file named *25g-1.bin* and registered the file with the Rucio server. Finally, we used the Rucio client to submit a request to the Rucio server to replicate the registered file from *StarLight* to *CERN*.

B.1 Site registration

We ran the script in Appendix 1 to register each site in the testbed as an RSE with the Rucio server. In particular, the two sites were registered as *STARLIGHT-SITE* and *CERN-SITE*, respectively. Please refer to Appendix 2 and 3 for the RSE details. After registration, the Rucio server could access the RSEs and launch BigData Express to establish direct RSE to RSE transfers over the network.

B.2 Experiment data preparation

To prepare for experiments in the next steps, we generated a file named *25g-1.bin* and manually registered it with the Rucio server at *STARLIGHT-SITE*.

When a new file enters Rucio, the system typically performs the following operations: (1) first, registering the file, (2) then registering the replica, (3) next actually uploading the file to storage, and (4) finally placing a replication rule on the file to secure the replica. Figure 5 illustrates the replica and the replication rule for the experiment file in Rucio.

B.3 Rucio data replication

We used the Rucio client to submit a request to the Rucio server to replicate the *25g-1.bin* from *STARLIGHT-SITE* to

BigData Express Control Event	*Transaction Time*	*SENSE Control Event*	*Transaction Time*
Service Negotiation	*5*	*Compute Service Intent (initial)*	*3*
		Re-Compute Service (negotiate)	*2*
Service Reservation	*9*	*Reserve with RMs*	*7*
Service Allocation	*94*	*Commit with RMs*	*34*
		Verify Service Model	*51*
Service Deallocation	*60*	*Release with RMs*	*4*
		Commit with RMs	*33*
		Verify Service Model	*12*

Table 2. BigData Express and SENSE transaction timing observations (event duration by seconds)

CERN-SITE. Once the Rucio server received the request, it first created a replication rule for the request (Figure 6), then generated the corresponding data transfer task, and finally submitted the task to BigData Express to launch data transfer.

We monitored the BigData Express data transfer process from BigData Express Web Portal @StarLight. The BigData Express data transfer successfully launched and ran to completion. Figure 7 illustrates the process, which is split into four stages:

1) *Prepossessing*. BigData Express schedules and assigns DTN to execute the data transfer tasks;
2) *Set up network path*. BigData Express calls SENSE to reserve and provision a 10Gbps WAN network path between StarLight and CERN;
3) *Data transfer*. BigData Express launches the data transfer tasks, monitors the progress of the tasks, and retries in case of errors;
4) *Resource cleanup*. BigData Express cleans up resources, including tearing down the WAN path.

Figure 7 also illustrates the throughput of the data transfer. It can be seen that the throughput of the first 40 seconds is higher, and subsequently stabilizes down to ~125MB/s. Due to time constraints, we installed and configured a low-end DTN at CERN, whose disk I/O speed was capped at ~125MB/s. In the first 40 seconds, the transferred data was temporarily cached in the DTN system memory, resulting in higher performance. Once the system memory was used up, subsequently transferred data must be saved in the disk. The throughput was ultimately constrained by the DTN's disk I/O performance.

In the experiments, we also collected events logging data between BigData Express and SENSE, and evaluated whether the timing of control processing and transactions satisfied the overall workflow requirement. We observed the interaction and recorded the timing of the following events during the WAN path provisioning.

- BigData Express discovers SENSE Service Termination Points and negotiates WAN path services with SENSE.
- BigData Express reserves and allocates SENSE service.
- BigData Express verifies service readiness with SENSE, retrieves manifest, and starts transfer.
- BigData Express stops transfer and deallocates service with SENSE.

Table 2 shows the processed data representing the average transaction timing of the above events. Through the experiments, we confirmed that the sequence and timing of all the interactions between BigData Express and SENSE satisfied the ROBIN service workflow requirements. It is worth noting that the BigData Express and SENSE interaction is scalable for a larger and more complex WAN environment, as demonstrated by the experiments. There are two major reasons for this:

- The SENSE solution architecture orchestrates multiple domains through distributed transactions, that both maintain resource integrity and parallelize the resource allocation;
- The intent-based interactions between BigData Express and SENSE normalize the information model and control mechanisms for end-to-end service orchestration. While underlying domains may use diverse types of network controllers like OpenNSA, OSCARS and OpenDaylight, that complexity is hidden for high level operations.

The results of these experiments validate the approach used, indicating that the major challenges of providing services in support of globally-distributed, multi-domain science research can be addressed by developing a comprehensive solution that integrates all resource elements involved in such transfers.

These initial successful results demonstrate opportunities for additional research using this approach, including exploring its utility for 100 Gbps international WAN paths, high-end DTNs, multiple site deployment, increased automation, and enhanced parameter analytics.

VI. Conclusion

This paper presents Robin, a unique comprehensive set of integrated services designed specifically for managing and moving extremely large amounts of data over long distances. The services include: Rucio, a data management service widely used by particle physics research communities; BigData Express, a schedulable, predictable, and high-performance data transfer service; and SENSE, a distributed resource network orchestrator.

References

[1] "Synergistic Challenges in Data-Intensive Science and Exascale Computing", DOE ASCR Data Subcommittee Report 2013.

[2] M. Barisits et al., "Rucio: Scientific Data Management." *Computing and Software for Big Science* 3, no. 1 (2019): 11.

[3] Q. Lu et al., "BigData Express: Toward Schedulable, Predictable, and High-Performance Data Transfer," 2018 IEEE/ACM Innovating the Network for Data-Intensive Science (INDIS), Dallas, TX, USA, 2018, pp. 75-84, doi: 10.1109/INDIS.2018.00011.

[4] Inder Monga, et al., "Software-Defined Network for End-to-end Networked Science at the Exascale", Future Generation Computer Systems, Volume 110, Pages 181-201, 2020.

[5] AutoGOLE Task Force Update, 2015. Available at: http://www.glif.is/meetings/2015/spring/vanmalenstein-autogole.pdf.

[6] Dart, Eli, et al. "The science dmz: A network design pattern for data-intensive science." Scientific Programming 22.2 (2014): 173-185.

[7] Guok, Chin, ESnet Network Engineer, and David Robertson. "ESnet On-Demand Secure Circuits and Advance Reservation System (OSCARS)." Internet2 Joint Techs Workshop, Salt Lake City, Utah. 2005.

[8] L. Zhang et al., "mdtmFTP and its evaluation on ESNET SDN testbed." Future Generation Comp. Syst. 79: 199-204 (2018)

[9] S. Shah et al., "AmoebaNet: An SDN-enabled network service for big data science." J. Network and Computer Applications 119: 70-82 (2018).

[10] McKeown, Nick, et al. "OpenFlow: enabling innovation in campus networks." ACM SIGCOMM Computer Communication Review 38.2 (2008): 69-74.

[11] W. Allcock, J. Bester, J. Bresnahan, A. Chervenak, L. Liming, and S. Tuecke, "GridFTP: Protocol Extension to FTP for the Grid," Grid Forum Internet-Draft, Mar. 2001.

[12] Jim Basney, Terry Fleury, and Jeff Gaynor, "CILogon: A Federated X.509 Certification Authority for CyberInfrastructure Logon," Concurrency and Computation: Practice and Experience, Volume 26, Issue 13, pages 2225-2239, September 2014.

[13] G. Roberts, J. MacAuley, T. Kudoh, C. Guok, "NSI Connection Service v2.1", OGF GFD.237, December 2019, < http://www.ogf.org/documents/GFD.237.pdf>

[14] J. MacAuley, G. Roberts, "Network Service Interface Document Distribution Service", OGF GFD.236, June 2019, < http://www.ogf.org/documents/GFD.236.pdf >.

[15] H. Trompert, J. MacAuley, "NSI Authentication and Authorization", OGF GFD.232, August 2017, < http://www.ogf.org/documents/GFD.232.pdf >.

[16] J. MacAuley, T. Kudoh, C. Guok, G. Roberts, "Applying Policy in the NSI Environment", OGF GFD.233, August 2017, < http://www.ogf.org/documents/GFD.233.pdf >.

[17] J. MacAuley, C. Guok, G. Roberts, "Network Service Interface Signaling and Path Finding", OGF GFD.234, August 2017, < http://www.ogf.org/documents/GFD.234.pdf >.

[18] J. MacAuley , T. Kudoh, C. Guok, "Error Handling in NSI CS 2.1", OGF GFD.235, August 2017, < http://www.ogf.org/documents/GFD.235.pdf >.

[19] https://openid.net/connect/

Appendix 1 Site Registration Script

```
# First, create the RSEs
rucio-cmd rucio-admin rse add --non-deterministic CERN-SITE
rucio-cmd rucio-admin rse add --non-deterministic STARLIGHT-SITE


# Add the protocol definitions for the storage servers
rucio-cmd rucio-admin rse add-protocol --hostname cixp-surfnet-dtn.cern.ch \
        --scheme bde --port 5000 \
        --prefix /192.91.245.26/disk0/ \
        --impl rucio.rse.protocols.bde.Default \
        --domain-json '{"wan": {"read": 1, "write": 1, "delete": 1, "third_party_copy": 1}, "lan": {"read": 1, "write": 1, "delete": 1}}' \
        --extended-attributes-json '{"bdeportal": {"url": "https://cixp-surfnet-dtn.cern.ch:5000", "apikey": \
                                        "ad7oiy51g07-0fzld5m3wai-qvr06qn2lx"}}' \
        CERN-SITE


rucio-cmd rucio-admin rse add-protocol --hostname 165.x.x.157 \
        --scheme bde --port 5000 \
        --prefix /165.x.x.142/disk0/ \
        --impl rucio.rse.protocols.bde.Default \
        --domain-json '{"wan": {"read": 1, "write": 1, "delete": 1, "third_party_copy": 1}, "lan": {"read": 1, "write": 1, "delete": 1}}' \
        --extended-attributes-json '{"bdeportal": {"url": "https://165.x.x.157:5000", "apikey":
                                        "a09ib4ba7dqsp-m3trq4xamzetm-fojrsx2u6jpmz"}}' \
       STARLIGHT-SITE


# Enable BDE
rucio-cmd rucio-admin rse set-attribute --rse CERN-SITE --key bde --value https://cixp-surfnet-dtn.cern.ch:5000
rucio-cmd rucio-admin rse set-attribute --rse STARLIGHT-SITE --key bde --value https://165.x.x.157:5000


# SET BDE name convention
rucio-cmd rucio-admin rse set-attribute --rse CERN-SITE --key naming_convention --value BDE
rucio-cmd rucio-admin rse set-attribute --rse STARLIGHT-SITE --key naming_convention --value BDE


# Fake a full mesh network
rucio-cmd rucio-admin rse add-distance --distance 1 --ranking 1 CERN-SITE STARLIGHT-SITE
rucio-cmd rucio-admin rse add-distance --distance 1 --ranking 1 STARLIGHT-SITE CERN-SITE


# Indefinite limits for root
rucio-cmd rucio-admin account set-limits root CERN-SITE -1
rucio-cmd rucio-admin account set-limits root STARLIGHT-SITE -1


# Create a default scope for testing
rucio-cmd rucio-admin scope add --account root --scope test
```

Appendix 2 RSE Information - STARLIGHT-SITE

$ rucio-cmd rucio-admin rse info STARLIGHT-SITE

```
 Settings:
==========
  third_party_copy_protocol: 1
  rse_type: DISK
  domain: [u'lan', u'wan']
  availability_delete: True
  delete_protocol: 1
  rse: STARLIGHT-SITE
  deterministic: False
  write_protocol: 1
  read_protocol: 1
  availability_read: True
  staging_area: False
  credentials: None
  availability_write: True
  lfn2pfn_algorithm: hash
  sign_url: None
  volatile: False
  verify_checksum: True
  id: be8e57b690ec4caeb16ce56ccb1db36f
Attributes:
============
  bde: https://165.x.x.157:5000
  naming_convention: BDE
  STARLIGHT-SITE: True
Protocols:
===========
  bde
    extended_attributes: {u'bdeportal': {u'url': u'https://165.x.x.157:5000', u'apikey': u'a09ib4ba7dqsp-m3trq4xamzetm-fojrsx2u6jpmz'}}
    hostname: 165.x.x.157
    prefix: /165.124.33.142/disk0/
    domains: {u'wan': {u'read': 1, u'write': 1, u'third_party_copy': 1, u'delete': 1}, u'lan': {u'read': 1, u'write': 1, u'delete': 1}}
    scheme: bde
    port: 5000
    impl: rucio.rse.protocols.bde.Default
Usage:
=======
  rucio
    files: 0
    used: 0
    rse: STARLIGHT-SITE
    updated_at: 2020-08-26 20:12:58
    free: None
    source: rucio
    total: 0
    rse_id: be8e57b690ec4caeb16ce56ccb1db36f
```

Appendix 3 RSE Information - CERN-SITE

```
$ rucio-cmd rucio-admin rse info CERN-
SITE


Settings:
==========
  third_party_copy_protocol: 1
  rse_type: DISK
  domain: [u'lan', u'wan']
  availability_delete: True
  delete_protocol: 1
  rse: CERN-SITE
  deterministic: False
  write_protocol: 1
  read_protocol: 1
  availability_read: True
  staging_area: False
  credentials: None
  availability_write: True
  lfn2pfn_algorithm: hash
  sign_url: None
  volatile: False
  verify_checksum: True
  id: e17d96435ec64dd0ae3cca7e93175a70
Attributes:
============
  bde: https://cixp-surfnet-dtn.cern.ch:5000
  CERN-SITE: True
  naming_convention: BDE
Protocols:
===========
  bde
    extended_attributes: {u'bdeportal': {u'url': u'https://cixp-surfnet-dtn.cern.ch:5000', u'apikey': u'ad7oiy51g07-0fzld5m3wai-qvr06qn2lx'}}
    hostname: cixp-surfnet-dtn.cern.ch
    prefix: /192.x.x.26/disk0/
    domains: {u'wan': {u'read': 1, u'write': 1, u'third_party_copy': 1, u'delete': 1}, u'lan': {u'read': 1, u'write': 1, u'delete': 1}}
    scheme: bde
    port: 5000
    impl: rucio.rse.protocols.bde.Default
Usage:
=======
  rucio
    files: 0
    used: 0
    rse: CERN-SITE
    updated_at: 2020-08-26 20:12:57
    free: None
    source: rucio
    total: 0
    rse_id: e17d96435ec64dd0ae3cca7e93175a70
```