A Workflow-based Network Advisor for Data Movement with End-to-end Performance Optimization

Patrick Brown and Mengxia (Michelle) Zhu
Department of Computer Science
Southern Illinois University
Carbondale, IL 62901, USA
Email: {patiek, mzhu}@cs.siu.edu

Qishi Wu and Daqing Yun
Department of Computer Science
University of Memphis
Memphis, TN 38152, USA
Email: {qishiwu, dyun}@memphis.edu

Jason Zurawski
Office of the CTO
Internet2
Washington, DC 20036, USA
Email: zurawski@internet2.edu

***Abstract*—Next-generation eScience applications often generate large amounts of simulation, experimental, or observational data that must be shared and managed by collaborative organizations. Advanced networking technologies and services have been rapidly developed and deployed to facilitate such massive data transfer. However, these technologies and services have not been fully utilized mainly because their use typically requires significant system- and network-related domain knowledge that most application users lack, and in many cases users may even not be aware of their existence. By leveraging the functionalities of an existing data movement advising utility, we propose a new Workflow-based Intelligent Network Data Movement Advisor (WINDMA) with end-to-end performance optimization. WINDMA integrates three major components: resource discovery, data movement, and status monitoring, and supports the sharing of common data movement workflows through account and database management. This system provides a web interface and interacts with existing data/space management and discovery services such as Storage Resource Management, transport methods such as GridFTP and GlobusOnline, and network resource provisioning brokers such as ION and OSCARS. We demonstrate the efficacy of the proposed transport-support WINDMA system in several use cases based on its implementation and deployment in wide-area networks.**

Keywords: High-performance networks; large data transfer; eScience applications

I. Introduction

Next-generation collaborative eScience applications often generate large amounts of simulation, experimental, or observational data on the order of terabytes or petabytes at present and exabytes in the predictable future. These data sets must be transferred to different geographic locations for various purposes such as data sharing, remote visualization, and distributed analysis. Due to the sheer volume, the data transfer at such a scale requires a controlled high-bandwidth network connection, which, unfortunately, is not readily available over shared IP networks. For example, the network bandwidth availability on the Internet is subject to constant changes due to time-varying concurrent cross traffics, therefore providing very little guarantee on transport throughput or dynamics for any data transfer.

To overcome these limitations, the current development of networking technologies has made it possible to provision dedicated connections with reserved bandwidth in high-performance networks. Recently, several high-performance networking projects are under way to develop such network services, which, however, have not found a large user base in broad science community as initially expected mainly because: i) the use of these services typically requires a considerable level of knowledge for network and host configurations that most scientific users often lack; ii) many users are even not aware of the existence of such advanced networking services and resources due to the communication gap between different technical domains. Of course, as domain experts with their own research missions, scientific users should not be expected to explore the availability of special or "hidden" networks and deal with the complexity of network/host configurations in the first place. Ideally, the application or the network itself in the case of software-defined networking (SDN) [1] should be smart enough to make a choice if the user is able to provide some simple information such as a target use case or the expected duration/amount of data flow. However, the reality of the current situation in the broad science community is that a substantial majority of scientific users are still relying on old-fashioned tools (e.g. HTTP, SCP, or FTP through a default IP path) that they are familiar with from their empirical studies for their daily data transfer needs.

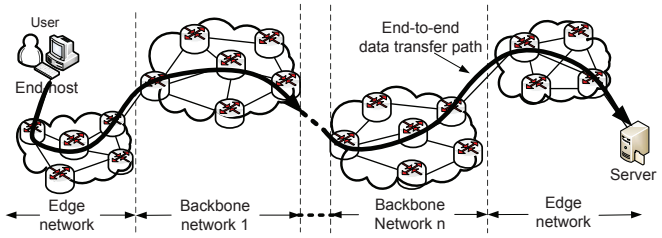


Fig. 1: The network infrastructure for wide-area bulk data transfer across multiple domains.

As illustrated in Fig. 1, driven by the continuously expanding scope of scientific collaboration, many eScience applications require wide-area bulk data transfer across multiple domains over different network segments such as edge and backbone networks from a source end node to a destination end node. However, in most cases, we only have insight into or control over a portion of the entire end-to-end path. To

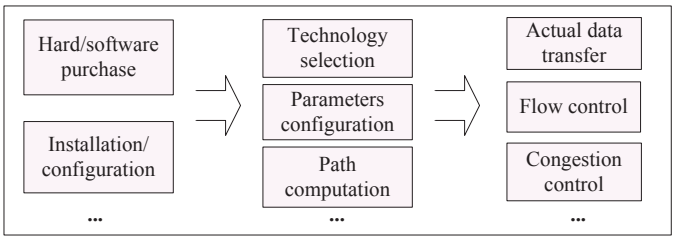


Fig. 2: Multiple steps in a data transport solution.

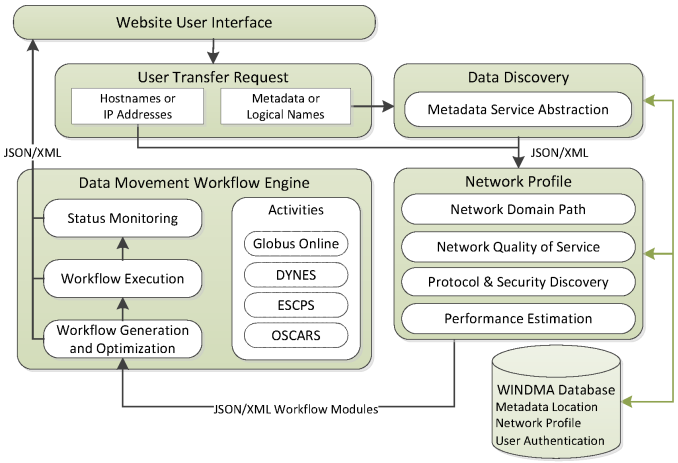


Fig. 3: The WINDMA framework: functional components and control flow.

meet a specific user request for data transfer, we have to take multiple steps to acquire and deploy the right system hardware/software, select the suitable technologies based on available resources, determine the best data transfer path, and perform the actual data movement, as shown in Fig. 2. Note that system and network resources vary significantly in their type, cost, performance, reliability, and security. For example, an end host might be equipped with multiple network interface cards (NIC) of different speed and cost; OSCARS in ESnet [2]–[4] and ION in Internet2 [5] provide different levels of bandwidth provisioning services at different cost and admission rate.

The goal of our work is to provide users an integrated solution for resource discovery, path composition, and data movement. By leveraging the functionalities of the existing Network-Aware Data Movement Advisor (NADMA) [6], we propose a transport-support workflow system to facilitate large data transfer with end-to-end performance optimization. This system employs a Workflow-based Intelligent Network Data Movement Advisor (WINDMA) utility and augments NADMA to interface with various network services to set up circuits and utilize appropriate transport methods for actual data transfer in different network environments. WINDMA inherits some basic functions from the previous NADMA for service discovery. However, a significant number of new features including workflow generation and management as well as performance estimation and optimization have been added to WINDMA to provide an additional level of intelligence in choosing an optimal set of network services. Our system operates at the application level and thus does not have the authority to decide the lower-level network routing path, which is typically determined by the underlying network infrastructure. While the current WINDMA implementation seeks to optimize the use of current-generation technologies such as dynamic circuit reservation, the modular design and cost model of WINDMA facilitate our future exploration into next-generation software-defined networking (SDN), particularly OpenFlow [1] technology, to define the dynamic path based on current traffic.

Within our WINDMA framework, the user only needs to submit a request that describes the data transfer requirements such as the service start and end time, the data source and destination nodes, a desirable bandwidth, or possibly a financial cost limit on the deployment and utility expenses*. These parameters are typical of the requirements found in dynamic circuit provisioning services and are useful in the construction of deadline-constrained transport solutions. Upon the receipt of such a request, our system first invokes the NADMA functionality to explore the available services and resources in end systems, edge segments, and backbone networks, models them as transport-support workflow modules with quantified parameters, and composes an optimal network path for end-to-end data transfer with different objectives such as cost (financial or technical), delay, and reliability. We implement and test the proposed transport solution in real network environments. The experimental results show that our method can achieve a reasonable accuracy in modeling existing services and improve the performance of bulk data transfer in wide-area networks.

The rest of the paper is organized as follows. Section II presents the WINDMA architecture and details its functional components. Section III presents the transport workflow optimization method. Section IV describes the system implementation and operation procedure. Section V presents the experimental results in several use cases. Section VI concludes our work.

II. The WINDMA Architecture

As shown in Fig. 3, the WINDMA framework consists of a group of distinct components written in Python that interact with users through a front-end web interface built on the Drupal [7] content management system. The Python components feature data discovery, network profiling, and workflow generation and optimization while remaining independent from each other, communicating input and output using XML and

*Even though most network services and resources are not free, their financial cost is quite minimal and is often negligible, especially in shared IP networks. Some advanced services such as OSCARS in ESnet and ION in Internet2 are currently provided to authorized users free of charge, but it is predictable that some pricing and accounting components will be integrated into these services in the future.

JSON format. A web-based front-end, enabled through Drupal, interprets the outputs and provides a simple web interface that abstracts the interaction of individual components. The independent nature of each component allows for WINDMA to operate as a library of functional components, which can be made available to other existing systems, or as an independent tool behind a front-end interface as presented in this paper.

Each component in the framework is sensitive to the context of its operation, allowing for flexible functionality in both a local and a remote context. The local context allows WINDMA to be distributed as a downloadable client-side tool or library that can operate in a trusted local context with deeper integration and utilization of third-party client-side tools such as data transfer client software. The remote context enables WINDMA to be used in a remote system, such as a website, which may have limited user trust. The automation abilities of WINDMA respect the context of operation and intelligently automate tasks only if the context permits it.

A user data transfer request may be defined as a physical or logical address together with some parameters that describe the user-desired services. The *Data Discovery* component translates any logical data into physical addresses before the request is interpreted by the *Network Profile*r. The Network Profiler discovers the network domain path, quality of service, protocols, and security mechanism capabilities of the transfer request by probing end host resources and networks while querying a central database for known network capabilities. WINDMA forwards such queried information to a *Data Movement Workflow Engine* component that provides the ability to construct, optimize, execute, and monitor a data movement workflow that is capable of performing the data transfer request. For the sake of completeness, we shall provide a brief overview of the Network Profiler, Data Discovery, and Database components of NADMA [6] that are inherited by WINDMA and provide a summary of the new Data Movement Workflow Engine of WINDMA while leaving the detailed discussion to Section III.

A. Network Profiler

To provide accurate advising for data movement in dynamic network environments, it is critical to collect and store status and resource information including network domain topology, provisioning services, and data management and movement protocols, which must be updated in a timely manner. For this purpose, we create and maintain a database that contains information enabling the construction of a network profile. This information includes organization references, host names and IP address subnets, network domain topology, and network technologies and capabilities of end sites.

The network quality of service is automatically discovered by interacting with the WINDMA database to determine if an end host has access to high-performance network infrastructure and necessary provisioning services that are capable of provisioning dedicated bandwidths. The system also supports scanning end hosts to discover system and network resources, including a variety of transport protocols, absent from the database. The end host resources and networking service technologies that are discovered by the Network Profiler component, such as ESnet OSCARS and Internet2 ION (as enabled by projects such as DYNES [8]), are organized into discrete modules and used by the Data Movement Workflow Engine during workflow construction and optimization.

B. Data Discovery

Since the physical location of a particular dataset may be unknown to the user, WINDMA provides a data discovery capability whereby the user can query third-party data services by keywords to locate a dataset of interest. This component interacts with metadata services using web service interfaces to query and discover the location of data sets as discussed in [6].

C. Database

The storage and retrieval of network domain topology, metadata location, authentication mechanisms, and known network capabilities are made possible through an SQL database. As discussed in [6], the database contains a collection of network and protocol information that has been automatically retrieved from known web service sources or manually entered into the database. Unlike NADMA, the database of WINDMA is transportable, enabling WINDMA to be used from both a client-side context as a local utility program and a server-side context as a library to an existing web portal such as Drupal. The SQL database supports both MySQL for server-side contexts as well as SQLite [9] for client-side contexts.

D. Data Movement Workflow Engine

The resources discovered by the Network Profiler and Data Discovery components are used to generate, optimize, execute, and monitor data movement workflows in the Data Movement Workflow Engine. The resources provided to the engine are modeled as data movement workflow modules. The self-contained engine uses these modules to generate data movement workflows by formulating a workflow optimization problem as discussed in Section III.

The data movement workflow consists of a dependency graph of tasks that must be completed for successful data movement using the subset of modules selected during workflow optimization. This task graph is transformed into a set of activities that fulfill the specific tasks to support workflow execution and status monitoring. The activities utilize client tools and third-party web services to support file transport protocols such as GridFTP and high-performance bandwidth reservation systems. Each activity implements a well-defined interface that allows for interchangeability while abstracting the specific, often messy, operation of the underlying tools and services.

III. Transport-Support Workflow Optimization

A. General Purpose

Emerging high-performance networking technologies (refer to Table I for a partial list of existing ones) have been rapidly

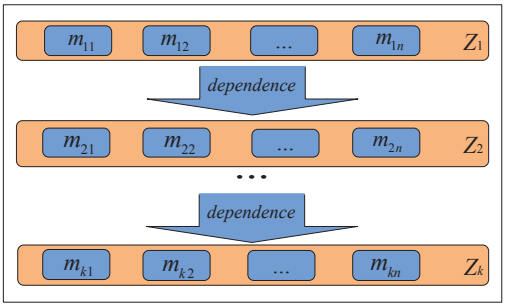


Fig. 4: A zone-based transport-support workflow structure.

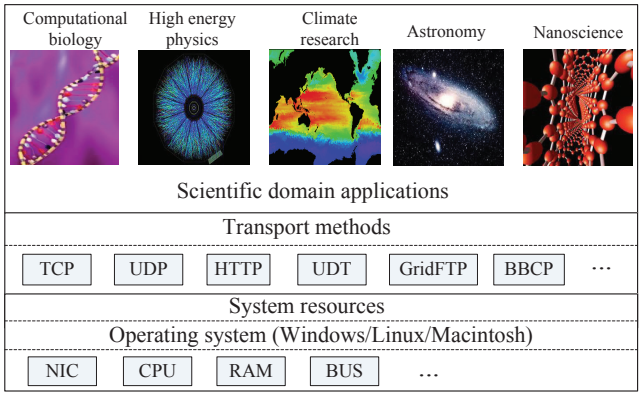


Fig. 5: A general structure of end host modules.

developed and deployed to support the transfer of large data sets produced by next-generation scientific applications for collaborative data processing, analysis, and storage. Unfortunately, due to the lack of computer and network knowledge, scientific users have not been able to fully utilize these resources. We model various types of resources discovered by NADMA in end systems, edge segments, and backbone networks, based on which, we formulate a Transport-Support Workflow Optimization Problem (TSWOP) that considers a comprehensive set of performance metrics and network parameters in different phases including device deployment, circuit setup, and data transfer. We propose an integrated solution to choose an appropriate set of technologies and services to compose the "best" transport-support workflow and provide end users with step-by-step advice to meet the user's data transfer requirements.

B. Math Models of Transport-Support Workflow Modules

We model the entire data transfer process with a zone-based structure as shown in Fig. 4. The data transfer process is divided into k zones in a logical space, and we categorize those workflow modules with execution dependencies into these zones. Each module represents a certain type of resource or task that must be used or performed for data transfer.

1) End Host Modules: At the end host, typically three steps across three zones would be invoked during the packet receiving process: (i) a data packet arrives at the NIC and generates an interrupt, (ii) the kernel traps the interrupt and reads the packet from the NIC's ring buffer to the transport protocol buffer, and (iii) the transport protocol processes and forwards the packet to the target user application. Accordingly, the end host modules are modeled within three zones, namely, system resource, transport method, and user application, as shown in Fig. 5.

- The modules in the system resource zone include both hardware units such as CPU, network interface card (NIC), and RAM, and system software such as operating systems;
- The modules in the transport method zone include application-layer transport protocols, kernel-level transport protocols, and other network services and resources. In Fig. 5, we model the application-layer protocol HTTP as a module that runs over the kernel-level transport protocol TCP, which is also modeled as a module. We place them in the same zone as both of them are transport protocols providing services to user applications. We would like to point out that our zone-based structure is flexible in that we can further divide the modules in each zone into more subzones as in the case of TCP/IP stack;
- The modules in the user application zone include all user applications that require data transfer services. These applications come from a wide range of disciplines spanning from climate research, nanoscience, astronomy, neutron sciences, high energy physics, computational materials, fusion simulation, to computational biology.

2) Networking Service Modules: A networking service module could be a technology, a mechanism, or a hardware/software system, which takes the users request as input, performs some predefined routines, and sends back to the user the resources and/or other relevant results under request. The common networking modules provide end users either a default IP or a network provisioning service with guaranteed bandwidth such as OSCARS in ESnet [2] and ION in Internet2 [5]. Several common networking service modules are listed in Table I [10]–[21]. Typically, some of these networking services utilize graph-based algorithms, taking into account the parameters and constraints specified in the Virtual Circuit reservation, such as source and destination endpoints, bandwidth, or VLAN tagging to compute a path for a reservation request.

Most of these services in public networks are not free. Even for those specialized services such as OSCARS in ESnet and ION in Internet2, a metering model could be predicted in the future. Therefore, with the specification of user desired services, we could calculate the financial cost according to the charging policies implemented by various service providers. In general, end users tend to be greedy in resource use, and thus the desired services may not always be available to all the users due to the competition between them or to a particular user due to the financial cost constraint, as commonly observed in high-performance network environments.

To compute the path for data transfer, the networking modules take the network topology with capacity information

TABLE I: Networking technologies/services/resources.

Modules	Remark
MPLS	label switching based link-level technology
VLAN	virtual LAN, regardless of the physical location
IP routing	the default IP path
OSCARS	bandwidth reservation within ESnet
ION	bandwidth reservation within Internet2
DYNES	edge network bandwidth reservation for Internet2
DRAGON	using MPLS technology
CHEETAH	circuit-switched
TeraPaths	end-to-end virtual path with bandwidth guarantees
ESCPS	provisioning end-to-end inter-domain dynamic circuits
UCLP	network resources treated as software objects
JGN(2/2plus)	fully-fledged next-generation testbed for research
Geant2	funded by NRENs and EC, testbed for research
HOPI	combining packet- and circuit-switching
GENI	virtual laboratory for exploring future Internet

and the current resource reservation as input, and return to the end user with desired services together with the financial cost or a simple service rejection message.

C. Technical Approach

1) User Request: A user request R specifies the desired data transfer service such as transfer start time t_s, transfer finish time t_f, source host address H_s, destination host address H_d, and transfer data size DS as well as some data transfer constraints and objectives such as the loss rate LR, required bandwidth BW and upper bound C_{net} of financial cost spent on the use of networking services. We use an n-tuple $R = (r_1, r_2, \ldots, r_n)$ to model a generic user request, where r_i $(1 \le i \le n)$ are user-specified parameters that describe desired services and constraints. Some user requests may only involve a subset of parameters in R, in which case, we can assign 0 or *null* to those parameters that are not under consideration.

2) Transport-Support Workflow Optimization Problem:

- **Calculation of Credit:**
  Depending on the modules' properties, the parameters r_i in the user request may be fulfilled at a different degree, which reflects the level of satisfaction for the data transfer requirement. Our goal is to select a subset of modules discovered by NADMA across all the zones to meet the user request R as much as possible.
  We define a 0-1 vector $X = (x_1, x_2, ..., x_n)$ for a specific workflow module w corresponding to the user request R, where $x_i(w)$ is 1 when module w is selected from its zone and it fulfills the user request parameter r_i; and $x_i(w)$ is 0 when w is selected but it cannot fulfill r_i. We associate each transport module with a vector X, which represents the *credit* of fulfilling the parameters of a user request R. Ideally, we wish to select modules that can satisfy all the objectives and meet all the constraints in the user request. However, scientific users, who are domain experts without sufficient network knowledge, may not always be able to provide reasonable or realistic requests. In many cases, due to the limited network resources and conflicting parameters, it is nontrivial to select a subset of modules to meet such user requests.

  For each selected module w, we calculate its credit $P(w)$ as follows:

$$P(w) = f(X(w)) = \sum_{i=1}^{n} (\alpha_i \cdot x_i(w)), \tag{1}$$

  where α_i is the weight for each parameter r_i in R, which is either decided by the end user according to the user's preference among all the parameters or automatically assigned by our model with the same value indicating the same level of importance.

- **Problem Definition**
  We formally define the Transport-Support Workflow Optimization Problem (TSWOP) as follows:
  Definition 1. *TSWOP: Given a user data transfer request $R = (r_1, r_2, ..., r_n)$, its corresponding weight vector $\alpha = (\alpha_1, \alpha_2, ..., \alpha_n)$, and a set of workflow modules categorized into k zones of the data transfer process, and a predefined 0-1 credit vector $X = (x_1, x_2, ..., x_n)$ for each module, we wish to select a subset of modules across the k zones such that the user request can be successfully met with the maximal credit computed as:*

$$\begin{aligned} \max(Credit) &= \max_{\substack{all\ possible \\ module\ selections}} \{\sum_{j=1}^{k} P(w_j)\} \\ &= \max\{\sum_{j=1}^{k}\sum_{i=1}^{n} (\alpha_i \cdot x_i(w_j))\} \end{aligned} \tag{2}$$

  and under the financial upper bound constraint Cost,

$$\sum_{j=1}^{k} C(w_j) \le Cost. \tag{3}$$

  In Eq. 3, w_j is the selected module from zone j, and $C(w_j)$ is the financial cost incurred by using the service provided by module w_j.

- **Estimation of Credit Vector X:**
  In our model, it is critical to predefine the parameters in X for a given module since these parameters affect the module selection and eventually the data transport performance. Some of these parameters may be straightforward while others may not. For example, it is relatively easier to determine if a module could provide a reliable data transfer service than to ensure that a certain failure rate or loss rate requirement be satisfied. For a better illustration, let us consider OSCARS as an example. Upon the receipt of a user request for a reserved bandwidth, OSCARS generates a base topology graph considering the parameters and constraints specified in the user's reservation request such as source and destination hosts, required bandwidth or VLAN tagging. The decision on such a request can be directly made by the implemented path computation algorithm. However, if the request specifies some requirements on failure rate or loss rate, the OSCARS service may not be able to determine if such requirements could be met. A feasible approach is to use historical data to estimate and predict the performance of a service, which is used to determine the corresponding parameters in X.

IV. System Implementation and Operation Procedure

The WINDMA transport-support workflow system is exposed through the browser by a web-based front-end built on a LAMP stack [22], utilizing the popular open-source Drupal content management system to provide routine administrative and user management functionalities. Queries are made against a SQL server to store and retrieve information about known advanced network resource storage and provisioning systems as well as determine network domain capabilities of end host networks. The website offers three ordered phases of Discovery, Transfer, and Monitoring that provide a general means for users to submit data movement requests based on various transport methods, discover network technologies of the end hosts and intermediate networks, and automatically construct and execute optimized data movement workflows. To better illustrate the operation of WINDMA, in each phase we will consider an example use case where the user desires to transfer data between a source host and a destination host on the Energy Sciences Network (ESnet), located at Brookhaven National Laboratory (BNL) in New York and Lawrence Berkeley National Laboratory (LBNL) in California, respectively.

A. Discovery Phase

WINDMA requires the entry of the source endpoint and destination endpoint, one of which may be the user's local computer, for the desired data transfer. The two endpoints are sufficient for WINDMA to provide a detailed analysis, but the user may optionally provide additional information about their network or endpoints including network capabilities and supported transfer protocols. Generic information about the nature of the user's end hosts and local area network capabilities are stored in the database as learned information that can be utilized in future discovery processes.

If an endpoint is unknown, the Data Discovery component may be used to locate the desired dataset. The Data Discovery component interacts with metadata services to locate and retrieve physical location information about the desired dataset. The user may query the target metadata service for a dataset of interest by keyword or a dataset identifier. For demonstration purposes, the Data Discovery of WINDMA currently supports the Earth System Grid (ESG) [23] metadata portals. The user submits a keyword query, selects from a list of matched datasets, and WINDMA automatically populates the appropriate endpoint and dataset information.

The endpoints given by the user or discovered by WINDMA as well as optional end host and LAN network information are processed by WINDMA to launch the discovery process. WINDMA uses this information together with the Network Profiler component to discover the network location and organization associated with the source and destination endpoints. The organization and network locations are used to build a network profile. The network domains and transport protocols supported along the possible paths from source to destination are determined in the Network Profiler component, yielding

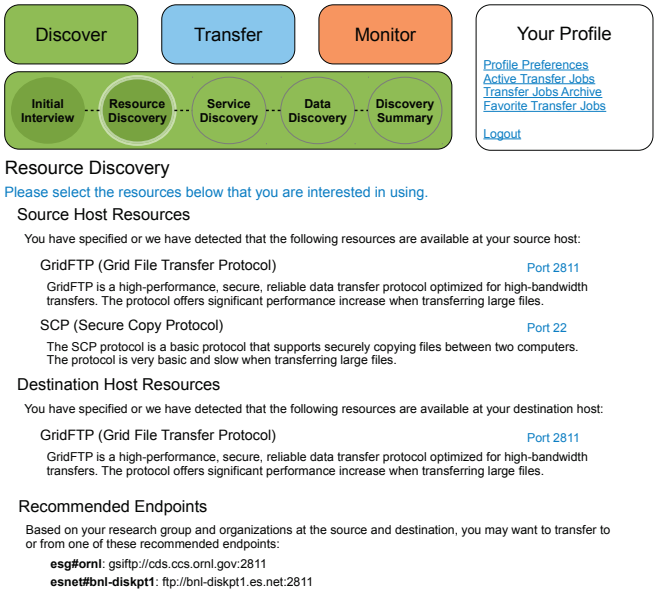


Fig. 6: WINDMA protocol discovery summary and selection.

a collection of End-host and Network Service modules that represent the capabilities of the end-host local networks as well as the wide-area networks between them. These capabilities, including the availability of high-performance networks and specific protocols, are presented to the user with accompanying information describing each specific technology. The user can use this information to decide which technologies they want to use to enable their data movement or they can allow WINDMA to generate a recommended data movement workflow using the techniques discussed in Section III.

In our example case, the user specifies the source host at Brookhaven National Laboratory and a destination host at Lawrence Berkeley National Laboratory. WINDMA uses the host information in conjunction with the Network Profiler component to discover specific information about each end-host. The combination of our protocol scanning and the known ports of the available protocols at the end hosts indicate the presence of SCP and GridFTP transfer protocols, as shown in Fig. 6. The organizations found in WINDMA's database and the characteristics of the physical sites found using WINDMA's network Quality of Service discovery reveal that high-performance networks are available at both the source and destination using ESnet OSCARS [2], as shown in Fig. 7.

Based on the discovered information, either the user or WINDMA selects a set of possible network technologies, including high-performance network services and specific protocols to use for composing a data movement workflow. This selection is submitted to WINDMA's Data Movement Workflow Engine and a data movement workflow is generated by selecting an optimized subset of modules as discussed in Section III. The workflow consists of a dependency graph of tasks that must be performed to enable desired network technologies and carry out the actual data transfer. This task

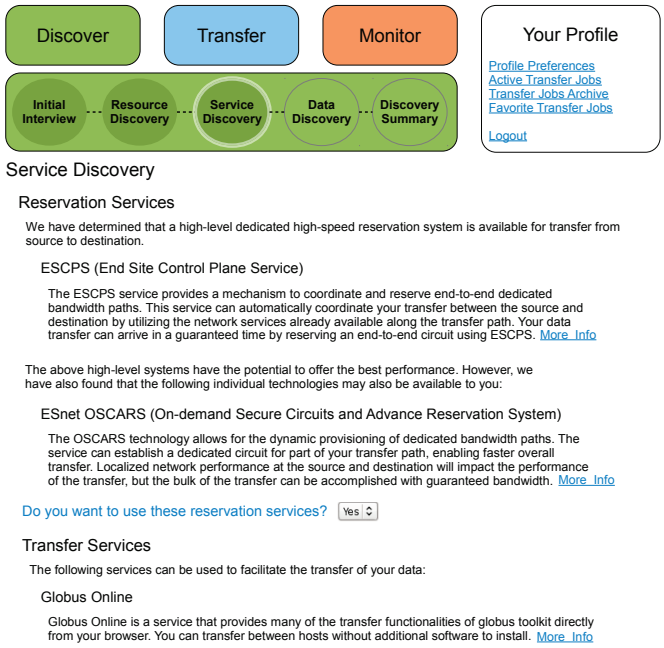


Fig. 7: WINDMA network and circuit reservation service discovery.

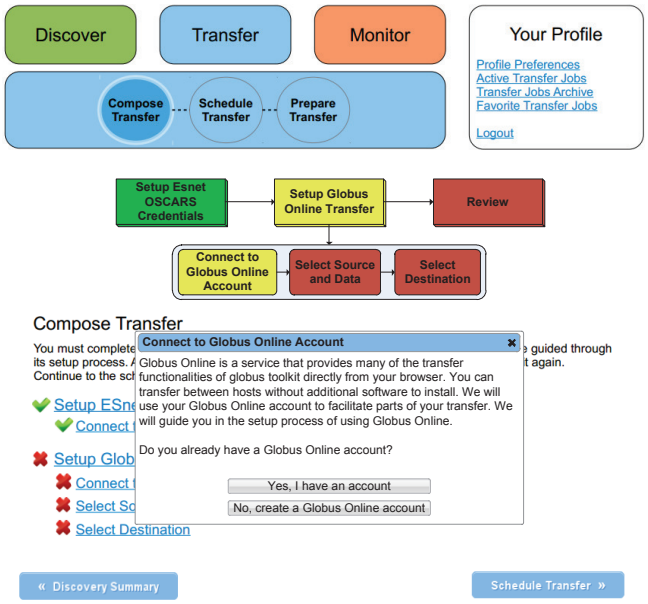


Fig. 8: WINDMA Transfer Phase showing precondition steps guided in a constructed workflow.

graph can be realized as both a manual set of instructions as well as a template for the automated workflow execution functionality offered by WINDMA in the Transfer Phase. In our example case, the discovered technologies and specific transfer constraints result in a distinct workflow that recommends using OSCARS instead of the default shared IP routing to facilitate a high-performance transfer over a dynamically provisioned virtual circuit using GridFTP as the transfer protocol.

B. Transfer Phase

The execution of data movement workflows can be performed directly from the WINDMA website using the Data Movement Workflow Engine component. The task graph workflow constructed in the Discovery phase is realized as a set of jobs to be executed by various transfer tools and third-party services available to WINDMA. A standard activity interface is implemented for different task classes and functions that allow for the execution of interchangeable components in the workflow graph. For example, the task of transferring data using GridFTP can be fulfilled by a Globus Online [24] implementation that exposes an activity interface for a GridFTP transfer. Activity implementations may also be refined by the network and host constraints present in the workflow, ensuring that only the activity implementations that can fulfill this specific workflow are considered.

The standard activity interface allows for implementations to feature both the use of standard client tools as well as third-party web services such as Globus Online for third-party transfer support. The web APIs of dynamic circuit provisioning systems found in high-performance networks also expose the ability to dynamically provision end-to-end dedicated circuits. Each activity uses one or more of these tools or web services to provide a function to fulfill a specific task. The activity itself may contain a series of preconditions that must be satisfied, such as obtaining necessary security credentials for a particular service. WINDMA guides the user in satisfying these conditions from the website.

Continuing the example case from the Discovery phase, the proposed transfer from Brookhaven National Laboratory to Lawrence Berkeley National Laboratory produced a task graph that requests an activity for fulfilling a dynamic circuit reservation for a transfer from an ESnet site to another ESnet site using OSCARS. Since this request can be fulfilled through ESnet OSCARS, the OSCARS reservation activity is utilized to expose OSCARS-enabled dynamic circuit reservation to WINDMA utilizing the OSCARS web service APIs. Similarly, the need for a GridFTP transfer can be fulfilled by a Globus Online activity implementation using a layer-3 reservation. Both of these implementations have preconditions of credential gathering and setup that must be completed by the user before workflow execution is possible. WINDMA presents these preconditions as steps in the transfer setup process, as shown in Fig. 8. The user is also provided the capability to specify the data transfer deadline, if needed, as shown in Fig. 9.

Besides credential management, steps including service authentication, dataset selection, and transfer preferences may also be necessary. The user is guided through these steps from the browser as WINDMA uses corresponding tools and web services to complete the preconditions. After satisfying the preconditions of the activities associated with the task graph, the user instructs WINDMA to execute the data movement

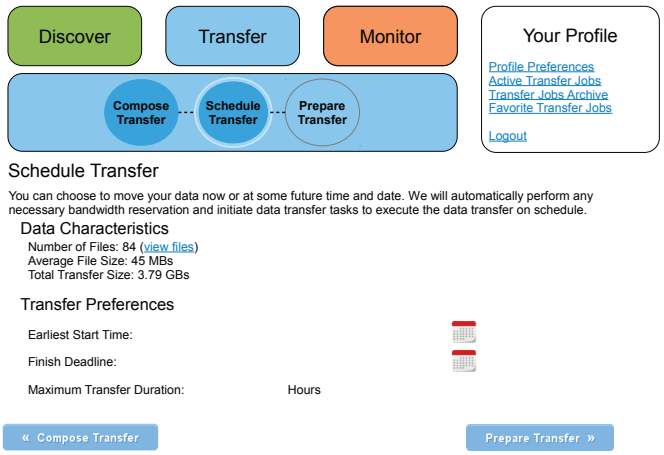


Fig. 9: Data transfer deadline specification in WINDMA.

workflow and the workflow is executed using the associated activities to satisfy the dependencies of the task graph. In our example case, the task graph indicates that file transfer is dependent on dynamic circuit provisioning. Therefore, WINDMA executes the OSCARS activity and waits for its success condition before executing the Globus Online activity.

C. Monitoring Phase

WINDMA monitors the execution of the workflow using the same activity implementations used to facilitate the Transfer Phase. Activities that support monitoring certain metrics such as transfer speed expose their data to WINDMA through a monitoring interface. The specific activity implements specific monitoring capabilities and WINDMA combines the metrics available from each activity associated with the task graph into a single report. The results are interpreted by WINDMA to extrapolate the performance of the current workflow and can be stored as historical indicators of the transport-support workflow module's performance parameters for future use as discussed in Section III.

The outcome of the workflow execution is similarly monitored by WINDMA. In the event of an error when executing the workflow, the activity that propagated the error determines if the system can be recovered. WINDMA will attempt to resolve the error automatically if possible, otherwise user intervention may be required. For example, the source dataset could be moved on the source machine during a transfer, causing subsequent file transfers to fail. The activity responsible for the file transfer may present this situation as a recoverable error with guided user interaction to recover and continue the execution of the workflow.

The monitoring of the example case is provided by both the OSCARS and Globus Online activities. The OSCARS activity contacts the OSCARS web service to reassure the user of the dynamic circuit provisioning while the Globus Online activity contacts the Globus Online web service to monitor the progress of individual file transfers. The results are appropriately stored in the database for historical reference and displayed to the user.

D. Collaboration

The web interface of WINDMA facilitates collaboration among the members of a distributed team. WINDMA takes advantage of the Drupal framework by allowing users to be organized into research groups, enabling collaboration among users and groups through the sharing of data movement workflows. Since the data movement workflows constructed by WINDMA can be represented as a graph of generic tasks, workflows can be shared between users without worry of credential exposure or unintended authorization. A user can share transport workflows as task graphs with others. Upon sharing, task graphs can be mapped into appropriate activities based on the user's preferences or system access capabilities.

In the example case, the user shares the existing data movement workflow stored as a task graph with a new user. WINDMA automatically associates the graph with new activities specific to the new user. If the new user indicates a different set of constraints for her transfer, such as lacking system access to a dynamic circuit provisioning system, WINDMA can appropriately choose the activities that fulfill the workflow using the new set of constraints. In this example, the OSCARS activity would not be used.

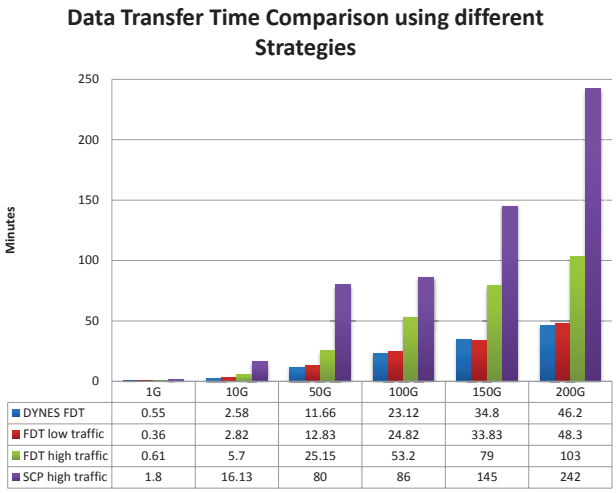
V. Experimental Results

To illustrate the efficacy of our WINDMA transport-support workflow system, we run several data transfer jobs with different file sizes and transfer strategies between two DYNES [8] endpoints that are capable of establishing dedicated end-to-end circuits.

We consider an example use case where the source host is located at John Hopkins University (JHU) and the destination host is located at the University of Michigan (UM). Each of the two end workstations features one quad-core Intel Xeon 2.4GHz CPU, 24GB of RAM and a high-performance RAID disk of 11TB. The sustained sequential read and write speed of the disk at both ends are around 1,100 MB/s in our I/O test, and both machines are equipped with a 10 GE network interface card (NIC), which ensures that the I/O performance will not be the bottleneck so that our experiments can show the impact of different transport solutions.

We compared the entire data transfer times using Fast Data Transfer (FDT) [25], a multi-threaded tuned TCP-based data movement application, and the regular SCP. The transport-support workflow constructed by WINDMA in the test case is $FDT \Rightarrow DYNES \Rightarrow ION \Rightarrow DYNES \Rightarrow FDT$, in comparison with the default one $SCP \Rightarrow TCP \Rightarrow IPv4 \Rightarrow TCP \Rightarrow SCP$.

The channel between these two end hosts is either a dedicated circuit with reserved bandwidth of 1Gbps or a default shared IP path. The measured transfer times for a set of different file sizes under different transfer scenarios, namely FDT over circuit, FDT over IP with low and high concurrent traffic, and SCP with high concurrent traffic, are plotted in Fig. 10 for comparison.



	1G	10G	50G	100G	150G	200G
DYNES FDT	0.55	2.58	11.66	23.12	34.8	46.2
FDT low traffic	0.36	2.82	12.83	24.82	33.83	48.3
FDT high traffic	0.61	5.7	25.15	53.2	79	103
SCP high traffic	1.8	16.13	80	86	145	242

Fig. 10: Transfer time comparison with different data sizes using different transfer strategies under different network conditions.

We observe that the transfer times over circuit- and non-circuit-based channels are very similar under a low level of concurrent traffic. Considering the significant overhead for setting up and tearing down the circuit (approximately 6 minutes in our test case), it is not worthwhile to establish a dedicated circuit if there is only light traffic in the network or the file size is small. Obviously, the larger the data set is, the more benefits the user can reap from utilizing a dynamically established dedicated circuit for reliable and fast data movement. However, if there is heavy concurrent traffic along the path, the non-circuit-based transfer job slows down drastically due to the increasing competition for shared bandwidth. On the other hand, the performance of the circuit-based transfer jobs are not perceivably affected by concurrent traffic.

If the users are not aware of or do not know how to utilize these advanced networking services that are available in their network domains, they may simply choose SCP over shared IP network for their data transfer (i.e. $SCP \Rightarrow TCP \Rightarrow IPv4 \Rightarrow TCP \Rightarrow SCP$). In the test case, the proposed WINDMA transport system has successfully discovered the available resources and computed the best data movement workflow to support the fastest data transfer. The workflow constructed for a regular data transfer job can be saved and reapplied to automate future data transfer tasks.

VI. Conclusion

We proposed a workflow-based transport solution, WINDMA, to support bulk data transfer in large-scale eScience applications. WINDMA inherited resource discovery functions from the existing NADMA tool. We constructed cost models for discovered resources and formulated path composition as an optimization problem. Actual data transfer is performed by running underlying transport methods or invoking existing data transfer services. Experimental results in real network environments show that the proposed transport solution is able to compose an optimal end-to-end path and carry out efficient data transfer for a given user request.

The parameter estimation in the transport workflow optimization model needs to be further investigated. It is of our future interest to refine this optimization model to better fit in real network environments. We will deploy a mature version of this system in production networks to support real-life scientific applications.

Acknowledgments

This research is sponsored by U.S. Department of Energy's Office of Science under Grant No. DE-SC0002078 with Southern Illinois University at Carbondale and Grant No. DE-SC0002400 with University of Memphis. The DYNES project is supported by the NSF Grant No. 0958998.

References

[1] Open Networking Foundation, https://www.opennetworking.org/.
[2] OSCARS: On-demand Secure Circuits and Advance Reservation System. http://www.es.net/oscars.
[3] C. Guok, D. Robertson, M. Thompson, J. Lee, B. Tierney, and W. Johnston, "Intra and interdomain circuit provisioning using the OSCARS reservation system," in Proc. of the BROADNETS, San Jose, CA, Oct. 1-5 2006, pp. 1–8.
[4] Energy Sciences Network. http://www.es.net.
[5] Internet2 Interoperable On-Demand Network (ION) Service. http://www.internet2.edu/ion.
[6] P. Brown, M. Zhu, Q. Wu, and X. Lu, "Network-aware data movement advisor," in Proc. of the 1st Int. Workshop on Network-aware Data Management, in conjunction with the Supercomputing Conference, Seattle, CA, USA, Nov. 14 2011.
[7] Drupal. http://drupal.org.
[8] J. Zurawski, E. Boyd, T. Lehman, S. McKee, A. Mughal, H. Newman, P. Sheldon, S. Wolff, and X. Yang, "Scientific data movement enabled by the DYNES instrument," in Proc. of the 1st Int. Workshop on Network-aware Data Management, in conjunction with the Supercomputing Conference, Seattle, CA, USA, Nov. 14 2011, pp. 41–48.
[9] SQLite. http://www.sqlite.org.
[10] ANI: Advance Network Initiative. http://www.es.net/RandD/advanced-networking-initiative.
[11] UCLP: User Controlled LightPath Provisioning. http://www.uclp.ca.
[12] DRAGON: Dynamic Resource Allocation via GMPLS Optical Networks. http://dragon.maxgigapop.net.
[13] JGN II: Advanced Network Testbed for Research and Development. http://www.jgn.nict.go.jp.
[14] Geant2. http://www.geant2.net.
[15] ESCPS: End Site Control Plane Service. https://plone3.fnal.gov/P0/ESCPS/.
[16] GridFTP. http://www.globus.org/grid_software/data/gridftp.php.
[17] GENI: Global Environment for Network Innivations. http://www.geni.net.
[18] BeStMan: Berkeley Storage Manager. Advance Network Initiative. https://sdm.lbl.gov/bestman.
[19] A. Patil, B. Belter, A. Polyrakis, T. Rodwell, M. Przybylski, and M. Grammatikou, "The GEANT2 advance multi-domain provisioning system," in Proc. of TERENA Net. Conf., Catania, Italy, May 15-18 2006.
[20] N. Rao, W. Wing, , S. Carter, and Q. Wu, "Ultrascience net: Network testbed for large-scale science applications," IEEE Communications Magazine, vol. 43, no. 11, pp. s12–s17, 2005, an expanded version available at www.csm.ornl.gov/ultranet.
[21] W. Allcock, J. Bresnahan, R. Kettimuthu, M. Link, C. Dumitrescu, I. Raicu, and I. Foster, "The globus striped GridFTP framework and server," in Proc. of Supercomputing, 2005.
[22] J. Lee and B. Ware, Open Source Development with LAMP: Using Linux, Apache, MySQL, Perl and PHP. Addison-Welsey, 2002.
[23] Earth System Grid (ESG). http://www.earthsystemgrid.org.
[24] Globus Online. https://www.globusonline.org.
[25] FDT: Fast Data Transfer. http://monalisa.cern.ch/FDT.