VIFI: Virtual Information Fabric Infrastructure for Data-Driven Discoveries From Distributed Earth Science Data

Ashit Talukder*, Mohammed Elshambakey*†, Sameer Wadkar‡, Huikyo Lee§, Luca Cinquini§
Shannon Schlueter*, Isaac Cho*, Wenwen Dou*, Daniel J. Crichton§
*CS Dept, University of North Carolina Charlotte, USA. {atalukde,melshamb,sschluet,icho1,wdou1}@uncc.edu
†AI Dept, IRI, SRTA-City, Egypt. mshambakey@srtacity.sci.eg
‡Axiomine, Washington D.C., USA. sameer@axiomine.com
§Jet Propulsion Laboratory/Caltech, USA. {Huikyo.Lee,luca.cinquini,Daniel.J.Crichton}@jpl.nasa.gov

***Abstract*—Traditional data analytics involves manually identifying and downloading relevant distributed datasets of interest to a common server/cluster where the analytics processes are executed. For very large distributed datasets, this slows down the analytics process, and for extremely large datasets it is often impractical to download such massive volumes due to bandwidth limitations. In such cases, data scientists need to be provided explicit access to the remote servers hosting the datasets, and possess detailed knowledge of the server infrastructure and environments, in order to send their analytics packages to the data owner. This alternative poses considerable challenges and has not been adequately addressed to date. In this paper, we describe a novel approach to addressing this challenge called Virtual Information Fabric Infrastructure (VIFI) which seamlessly allows users to conduct analytics-in-place by distributing analytics to the distributed repositories without moving the underlying datasets to a common location. By allowing automated analytics scripts to be sent to the data and orchestration of distributed infrastructure, VIFI allows users to conduct, execute and coordinate complex analytics activities in-place with the data on multiple data repositories. VIFI uses Docker containerization technology along with open-source workflow tool NIFI to achieve automated orchestration and distributed analytics without requiring users to posses detailed knowledge of the distributed repositories and their underlying infrastructure. We demonstrate and evaluate VIFI on a Earth Science use-case for evaluation of precipitation over the Great Plains involving analytics on massive distributed data repositories.**

I. Introduction

Data is increasingly used in science and technology, and other areas, to derive insights, initiate discoveries, and make decisions [1]. This is a consequence of improvement and convergence of many capabilities, including lower cost of storage, higher bandwidths, increased digitization capability through the prevalence of digital sensors and digital records (compared to paper-based storage), and parallel processing and compute capabilities. Social and behaviorial habits have changed in the last decade, making available online information about individual preferences, habits, events, and their opinions via social networks, blogs, twitter, and other avenues. This has immensely improved the potential of deriving knowledge, trends, patterns, and correlations through data, than was previously possible.

Many of the insights and discoveries that could be possible often reside not in a single dataset but in multiple datasets that are often distributed [2]. Therefore, the true power of data-driven insights can be leveraged when the disparate datasets are aggregated and combined, thereby allowing accurate and reliable extraction of complex correlations, patterns, and anomalies that may not be possible with a single data source alone. There are abundant examples of the value of aggregating data, ranging from science and technology to defense. In Earth science, climate predictions at a regional and global scale require the assimilation of multiple satellite and in-situ measurements, which is a tedious and non-trivial task since datasets are distributed across sites. In defense and security, insights into threats are often obtained by combining intelligence feeds from multiple sources, aggregated with publicly available open data including twitter and media. In biology, experimental data in one lab, while valuable, can be combined with those from other labs to yield insights that would not be possible by individual experiments.

In this context, data aggregation is often achieved by manual processes where the data scientist downloads all datasets of interest into a single repository, prior to doing analysis and discovery. Data aggregation is a tedious, time consuming process, and is increasingly posing a challenge as data volumes increase in size thereby making it impractical to move massive repositories across the network. Furthermore, as datasets increasingly exhibit streaming characteristics with dynamic, real-time updates, downloading the massive datasets from repositories for analysis becomes impractical and low-value due to the ”staleness” of downloaded information.

A complementary approach is the development of domain-specific, collaborative data-fabric solutions where teams of collaborators share data via common repositories or make their datasets available via interoperable solutions and APIs. This is often a tedious process, involving establishments of agreements, development of custom tools specific to the team, and data standards that make the collections of data amongst the teams interoperable with each other.

While the data fabric approach offers solutions for collaborative teams to mutually benefit from each other's datasets, such solutions are not generally scalable across domains. Furthermore, this often disadvantages data scientists and organizations that may not own large datasets of interest but may benefit from having access to the datasets owned by other institutions. In this paper, we offer a novel, complementary, Virtual Information Fabric Infrastructure (VIFI) that enables

978-1-5386-0435-9/17/$31.00 ©2017 IEEE

analytics at the data sources, when the datasets are distributed and do not need to be moved across the network. This novel capability allows for practical implementation of data science solutions when massive datasets make it difficult for data movement, and when the datasets change at a rate that makes regular downloads from separate sites tedious and infeasible. The remainder of the paper is organized as follows: a comparison to state of art is discussed in Section II, and the earth science use-case used for evaluation of our virtual information fabric is detailed in Section III. The VIFI proof of concept architecture and implementation is described in Section IV, followed by a discussion of the VIFI results on the earth science climate modeling evaluations in Section V. Ongoing and future planned enhancements to VIFI are discussed in Section VI.

II. Related Work

Current data-driven inquiry often involve identifying and assembling relevant data from disparate locations to one repository, prior to conducting analysis. To overcome this manual process, alternate solutions have been proposed involving data sharing using high-speed network solutions, and cloud-based approaches to host repositories, while others focus on providing shared computing resources. DataONE [3], [4] is a project focused on earth and environment sciences by providing easier access, search and discovery to multiple data repositories in these domains. The Open Science Grid [5], [6] enables scientific research by providing distributed computing resources. Few projects aim to develop research infrastructure. XSEDE (Extreme Science and Engineering Discovery Environment) [7] integrates data, supercomputers, and analytic tools thereby allowing end-users to share computing resources and data repositories. Globus [8], [9], [10] is software-as-a-service that makes it easy to discover, replicate, and access big data resources at different locations. Globus is used to deliver scalable research data management services in a secure manner to a variety of stakeholders. Some Globus features, like data publication and managed endpoints, needs Globus starter or standard subscription. SciServer [11] is a cyber-infrastructure system providing a suite of tools and services (including storage, access, query, and processing of Bigdata). SciServer aims to provide solutions for discovery, access, and query to data from different disciplines with different data formats and models. SciServer attempts to minimize data movement by collecting data from remote resources to the server that contains the majority of the required data. Finally, SciServer sends the analytics to the server using Jupyter Notebook [12]. In contrast to the existing solutions, VIFI aims to have "truly distributed analytics" where analytics are executed at data-owners' resources without data movement. SciServer adds new infrastructure for additional data and/or computational resources rather than integrating with existing data-owners' infrastructure. SciServer performs computations using Jupyter Notebook [12], whereas VIFI uses Docker [13] to provide more flexibility over the analytics tools and analytic environments that can be used by the data scientists in conducting

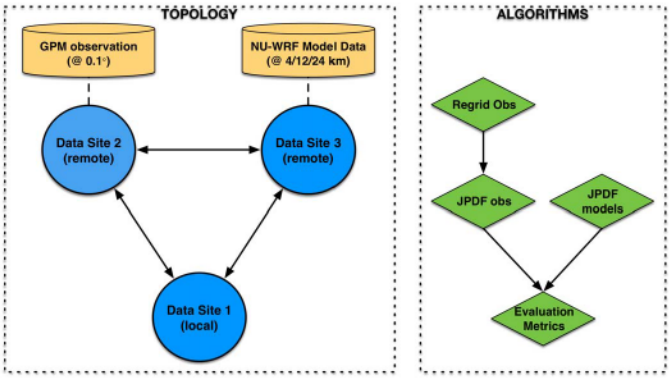

Fig. 1. Object model for the Earth Science use case: nodes, network, data and algorithms.

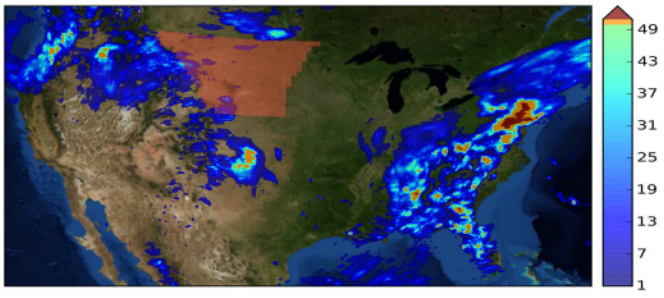

Fig. 2. Total rainfall estimates of GPM IMERG analysis for the contiguous US on June 1st, 2015.

data-driven inquiries (e.g., programming languages are not limited by what is provided by Jupyter). Additionally, some components of SciServer are not open-source, which may be a disadvantage in certain applications.

In contrast to previous projects, VIFI does not require data to be physically moved to a common location; rather, it provides a framework for analytics at the data source. VIFI will support authorization control for data and metadata access, and execution of models for end-users without exposing private datasets. VIFI aims to integrate with existing infrastructure of data owners and uses data owners' policies for resource access. Thus, VIFI considers two types of users: the data-scientist and the data-owner. VIFI aims to develop and use only open-source components (e.g., NIFI [14] and Docker Swarm [13]) and have free access to all its features (i.e., no subscription needed to use specific features of VIFI). This will also allow users to develope, reuse, and customize VIFI for their purpose as needed.

III. Earth Science Use Case: Evaluation of Precipitation over the Great Plains

As climate science continues to generate a massive amount of data, the process of climate model evaluation critically rests on data science and technology infrastructure, including storage, computation, networking, model evaluation tools and

metrics, and visualization capabilities. Both observational and model data are extremely distributed and managed by different institutions. Observational data are originated from a variety of sources, such as ground stations, flight missions, and satellites and independently stored and processed by different agencies. Climate models are developed and maintained by many countries around the world, and all of the models generate outputs in their own formats. The movement and analysis of these massive and distributed data, without any scalable data and computational infrastructure, is limiting the scientific yield that will result from the capture of these distributed datasets. Thus, the current Earth Science use case shows how VIFI provides shared cyber infrastructure with focused services, capabilities, and resources to facilitate data-driven research and decision making in climate science without moving massive datasets.

The Earth Science Use Case is illustrated in Figure 1. The observation data used in this study are from the Integrated Multi-satellite Retrievals for Global Precipitation Measurement (GPM IMERG, hereafter GPM [15]). By combining multiple satellites and ground gauge observations, GPM has provided a gridded precipitation over the globe every 30 minutes since 2014. Figure 2 displays daily total rainfall over the contiguous United States on June 1st, 2015. The GPM data for the 2015 summer (from June Through August) are stored at 0.1° resolution on a remote server (Data Site 2). The GPM data's size is about 11 Gigabytes. Climate simulation output from NASA Unified WRF (NU-WRF, [16]) at multiple resolutions (4km, 12km, and 24km) are stored on a different remote server (Data Site 3). For one summer, the size of NU-WRF output files with 24 km resolution is about 21 Gigabytes. A climate scientist user uses a local computer (Data Site 1) to evaluate simulated characteristics of precipitation in the simulations against observational data for the summer. This use case of simulated rainfall evaluation against GPM is illustrated in Figure 2. Our analysis domain shaded with red color in Figure 2 is the Northern Great Plains defined in [17]. Inspired by [18], we adapted the joint probability distribution function of rainfall intensity and duration (JPDF), which is applied independently to the observational and model data. This rainfall JPDF from observational data characterizes hourly precipitation data over the Great Plains and allows us to evaluate simulated rainfall in the models through the comparison of JPDFs between observational and model datasets. To quantify the performance of climate simulations, we defined the following evaluation metric, which measures the overlap ratio between the observed ($F_0(x,y)$) and simulated ($F_1(x,y)$) JPDFs.

$$overlap = \iint_{-\infty}^{+\infty} minimum(F_0(x,y), F_1(x,y))dxdy \quad (1)$$

The larger overlap is, the better performance of a simulation. Minimum and maximum values of the overlap are 0 and 100% respectively. The Earth Science Use Case in Section IV-C shows how VIFI will leverage data-driven discovery in climate science.

IV. VIFI: Virtual Information Fabric Infrastructure

VIFI will significantly reduce the time for data-driven inquiry by bringing the analytics (lightweight) to locations containing massive amounts of data. VIFI will provide an opportunity for aggregate analysis of a large number of operational datasets. Users will be able to perform emergent analysis as VIFI allows users to reuse data owned by others to perform an integrated analysis. VIFI allows virtual sharing of different types of testbeds that generate experimental data without necessarily moving the raw data that may be excessively large to move easily.

A. VIFI Architecture

Our VIFI architecture, as shown in Figure 4, is designed to address the *Distributed Orchestration of Federated Infrastructure and Portable Analytics* without requiring the user to possess detailed knowledge of the infrastructure, platforms, and environments at each data site or directly accessing and manually coordinating the analytics processes at each data site. Modern data science is inherently collaborative due to the now common practice of federating heterogeneous and disparate data to gain synergistic insight. At an extreme scale, the VIFI system proposes to orchestrate such collaborative federation for both data and analytics methodology. The VIFI Orchestrator, acting as a client/server interface between distributed partner infrastructures, will serve as the primary system component enabling federation functionality and managing distributed interactions. The VIFI Orchestrator is designed in an extensible manner composed of modular service components that provide functionality through a sustainable framework, thus allowing integration of new technologies as they become available. The proposed orchestrator extension will enable 1) distributed workflow optimization and coordination over widely distributed and loosely federated infrastructure. 2) Automated and dynamic provisioning of distributed processing infrastructure by providing a framework to provision guest processing on multiple local infrastructure hosting data via portable analytics containers (PACs).

B. Implementation

VIFI is an open-source platform that can be deployed on any infrastructure. While the final VIFI version will be deployed on open-source repository, we currently used Amazon EC2 [19], [20] infrastructure for VIFI initial development, testing and evaluation of proof of concept. Each AWS EC2 machine configuration is shown in Table I. The current VIFI proof-of-concept implementation has the following main components to allow automated analytics in-place on distributed data:

Portable Analytic Container (PAC): PAC is a containerized environment to execute users' analytic scripts. The analytics environment offers standardized, easy-to-use functionality for end-users to write, edit, and deploy analytics scripts to the distributed data repositories without assembling associated environments that are needed to run the analytics scripts. Each PAC is implemented as a Docker Image that contains the

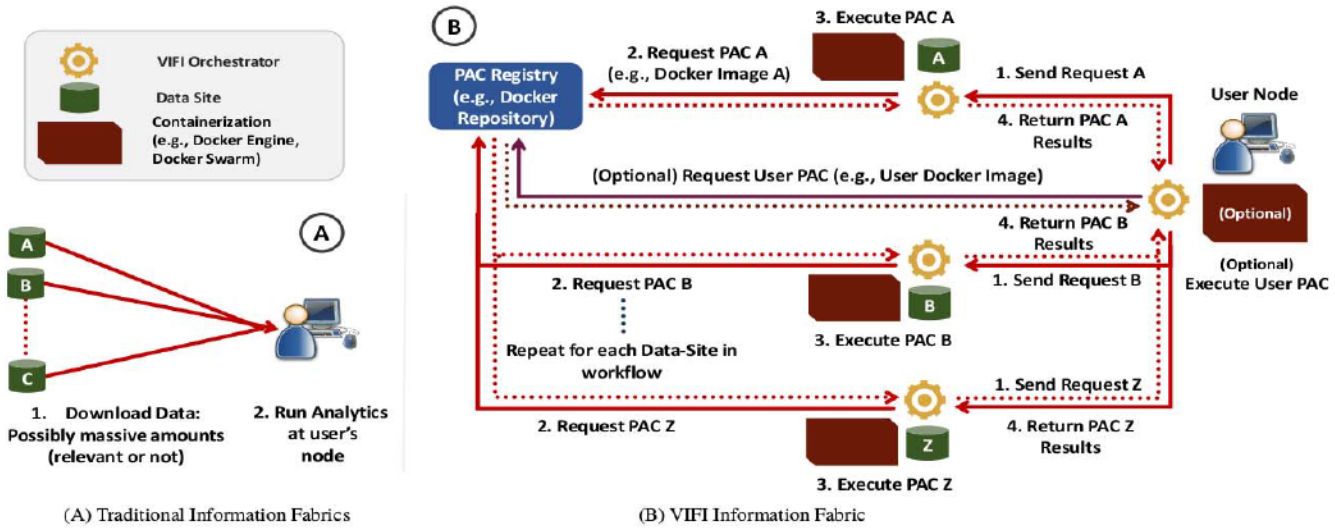

Fig. 3. Traditional Data Fabric compared to VIFI Data Fabric

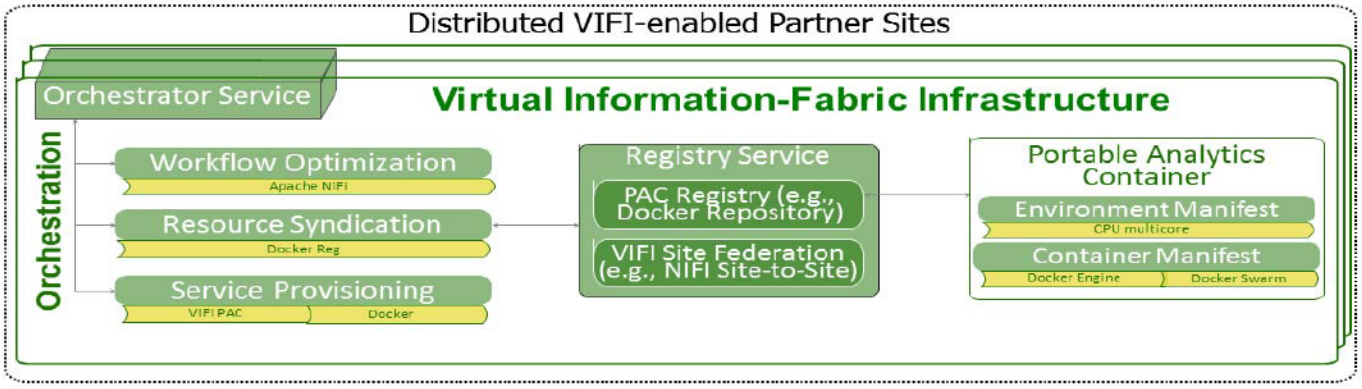

Fig. 4. VIFI architecture framework.

most common dependencies for running the required analytics (e.g., Python Docker Image with Open Climate Workbench libraries). This offers several advantages for distributed analytics: it facilitates lighweight containers that can be easily sent across the network, even under severe bandwidth limitations (since only the scripts, and not entire environments are transferred); and it makes it possible for novice users who may not possess in-depth data science knowledge or expertise to easily conduct data-driven experiments, since it absolves the need for the end-user to put together and install complex software environments in their local machines. We use Dockers for the PACs since it gives users more flexibility over the analytics tools and analytic environments that can be used by the data scientists in conducting data-driven inquiries, in contrast with Jupyter notebooks that have limited analytics environment support).

Registry Service: Registry is a repository that hosts different PACs to enable users to store, use and share PACs. In the current implementation, Docker repository [21] is used to implement the Registry Service. Future VIFI enhancements will include addition of distributed registery services to optimize download and upload time of PACs; this will enable greater efficiencies in creation and distribution of of PACs, especially as the size of VIFI users and application domains scales up.

Orchestrator: Orchestrator automates communication between different workflow components and orchestration of infrastructure at different sites. Each component can be as simple as a single process running on a host, or the host itself. Currently, NIFI [14] is used to implement orchestration and workflow design between workflow components. Workflow components can reside on the same host or on distributed hosts through NIFI site-to-site communication [14]. The VIFI orchestration services involves loosely coupled NIFI interaction and coordination between sites that enables fault tolerance and seamless scalability as the number of data-sites and users grow.

User Node: The user node is the access point for the user to interface with the VIFI system. The User node is a node with a user interface, network, and basic compute capabilities. It can be a PC or a server. NIFI is installed at each user node

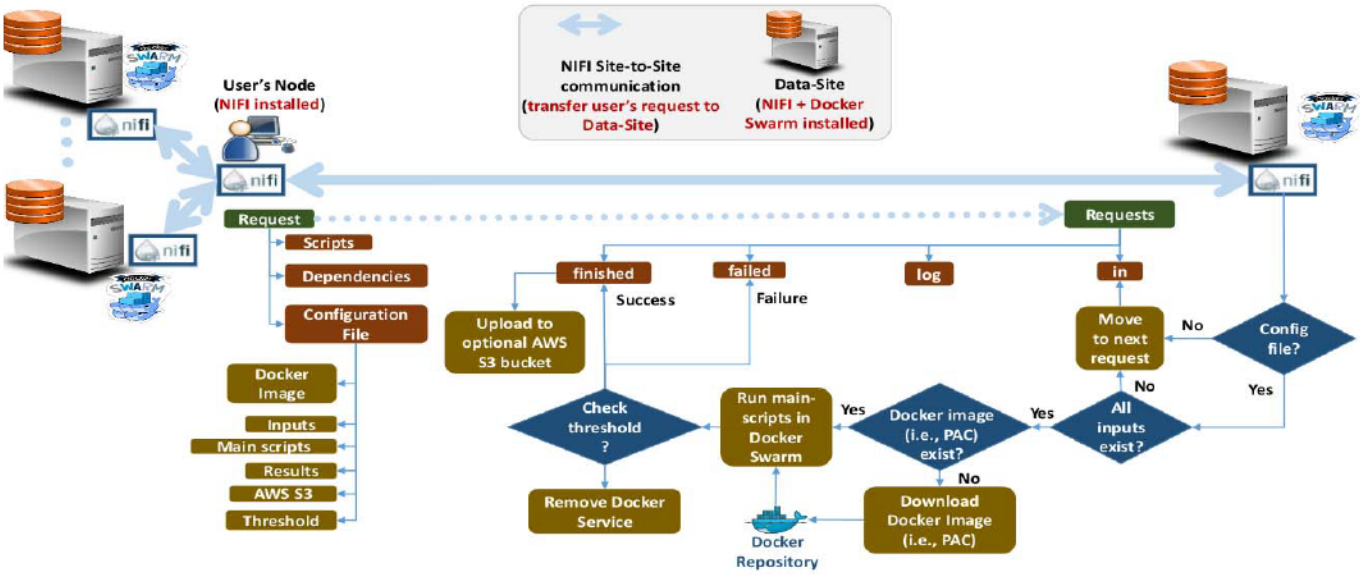

Fig. 5. Current VIFI implementation

to enable communication with other workflow components, as well as workflow initiation. Users prepare, edit, and submit analytic scripts, as well as other required inputs and interact with VIFI capabilities through a NIFI web interface and customized user-interfaces to edit, deploy PACs and review data analytics results returned after completion of the analytics processes from distributed repositories.

Data-Site: Partner Data-sites in the VIFI fabric host different aspects of heterogeneous data from various sources and models. Each VIFI-enabled data-site has NIFI [14] and Docker Swarm installed [13]. While NIFI enables distributed orchestration and communication between each data-site and the rest of workflow, Docker Swarm executes users' scripts using specified PACs as Docker services (e.g., user Python script run within Python Docker Image). Docker Swarm deploys a cluster at each data-site for parallel analytics execution automatically without requiring users to possess any knowledge of the infrastructure, platforms and environments at each Data-site.

In the traditional centric approach (Figure 3-A), required (massive) data must be downloaded to user's node first (step 1) before the user can run the required analytics at the user's node (step 2). In the VIFI approach (Figure 3-B), users submit their analytic requests to different data-sites (steps 1). Each data-site (and user node if some analytics will run locally) downloads required PAC from registery service (steps 2), then executes user's analytic request (step 3). Finally, each data-site sends analytic results back to user's node (steps 4). Figure 5 shows the current VIFI implementation. NIFI automates the workflow between user's node and different data-sites. Users submit their analytic requests, consisting of scripts, dependencies and configuration file through NIFI web interface. Users specify main scripts, dependencies, results, PAC, and other configuration parameters in the configuration file. Upon receiving a request, the data-site initially checks the configuration file to ensure all specified inputs exist. Otherwise, the data-site moves to the next request. Requests with incomplete inputs are checked periodically to ensure all inputs are satisfied. Data-site downloads nonexisting required PAC (i.e., Docker Image) from PAC registry (i.e., Docker repository). Users' main scripts are executed within the specified PAC as Docker Swarm services, then outputs results to AWS S3 bucket [20]. Future VIFI implementations will send results directly to user's node or another place specified by the user (e.g., FTP server). The Execution time of each analytic script is upper bounded by a specified threshold to ensure that the data-site is not completely consumed by one script. VIFI sets a default threshold value if one is not already specified by the user. The timestamp of requests' events (e.g., request received, processing started, succeeded, failed) are logged for monitoring. Finally, the Docker services configured for the analytics process are removed.

C. VIFI Proof-Of-Concept Implementation on Earth Science Use Case

To overcome the challenges raised by conducting analysis in the centralized architecture, VIFI will enable a new paradigm for executing scalable analytics optimized for distributed data systems similar to one in Figure 6, whereby an Orchestration Engine (deployed at each Node) is responsible for tasks on VIFI-enabled remote servers. The first steps in the centralized architecture (1 and 2 in Figure 6) are downloading

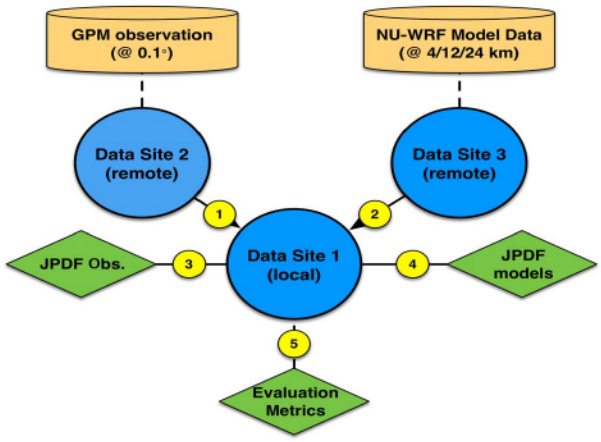

Fig. 6. Traditional execution of Rainfall Analytics use case, using a Centralized architecture.

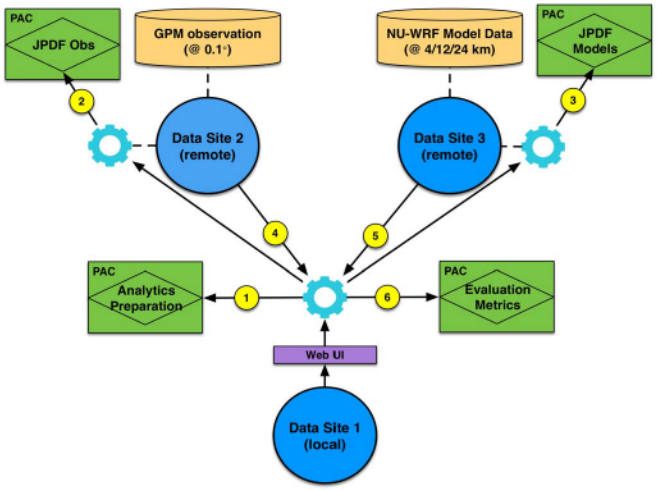

Fig. 7. VIFI-enabled distributed architecture for scientific analysis, leveraging Orchestration and Portable Analytic Containers.

observational and model data to a central location (User Node). After completing the downloading, the science algorithms to compare simulation output with observations, calculation of JPDFs and their overlap, are executed (steps 3, 4 and 5 in Figure 6). The workflow in the VIFI architecture (Figure 7) consists of the following steps: 1) VIFI Client running on User Node contacts the local VIFI Orchestrator to search and/or create suitable Portable Analytics Containers (PACs) to run user's scripts. 2) VIFI Client on User Node (i.e., Data Site 1) contacts the VIFI Orchestrator running on Obs Node (i.e., Data Site 2), asking to deploy and execute a Portable Analytics Container (PAC) that regrids the observational data to the desired model resolution (if necessary), and to calculate the observed JPDF using the newly regridded or original observational data. 3) VIFI Client on User Node contacts the VIFI Orchestrator on Model Node (i.e., Data Site 3), requesting to deploy a PAC to calculate simulated JPDFs with different spatial resolutions. 4&5) VIFI Client transfers the JPDF results from Obs Node and Model Node to User Node. 6) VIFI Client contacts the local VIFI Orchestrator to deploy a PAC on User Node that computes the overlap for comparing the observed and simulated JPDFs.

V. Results

Our focus for the VIFI framework in this paper is on evaluating the general-purpose VIFI POC architecture on a Earth science use-case for climate model prediction assessment. Specifically, comparing JPDFs between GPM and NU-WRF simulations improves our understanding of rainfall characteristics including duration and intensity of rain events. The main challenge in comparing regional precipitation characteristics using the JPDFs from observational (GPM) and model (NU-WRF) datasets is the massive size of datasets archived remotely on Data Sites 2 and 3, and slow data transfer speed between the servers and a user's local node (Data Site 1). For example, the NU-WRF simulation at 4 km horizontal resolution generates 756 gigabytes of two-dimensional hourly datasets over the contiguous United States for one season. Reducing the cost of archiving and transferring such a Big data is a key issue in the current study to be addressed by VIFI. As an alternative topology to the centralized case in Figure 4, we designed and implemented some experimental evaluation of precipitation simulated by NU-WRF with 24-km resolution in the distributed architecture. Figure 7 presents the distributed architecture implemented via VIFI. [22] simulated efficiency of the two workflows similar to those in Figure 1 and 6, and conclude that executing the model evaluation process in the distributed architecture offers a significant improvement in efficiency and run-time without compromising accuracy when evaluating climate models with high spatial and temporal resolutions. In this study, we take a step forward from simulating efficiency of the workflows in [22] by scheduling, executing, and monitoring a series of processes on local (Data Site 1) remote servers (Data Sites 2 and 3).

Figure 8 shows the VIFI results on climate model evaluation for precipitation; specifically, JPDFs difference between GPM and NU-WRF for the Northern Great Plains in summer. Overall, NU-WRF simulates reliable precipitation characteristics with the 24-km spatial resolution by showing the overlap ratio of 88 % for the two JPDFs from GPM and NU-WRF. The negative biases (brown-colored squares in Figure 8 indicate that short-duration downpour rain events whose peak rainfall rate is more intense than 5 mm/hour are less frequently simulated by NU-WRF compared to the GPM-observed precipitation.
There are several key benefits offered by the VIFI architecture. First, all phases of the scientific analysis lifecycle for the climate model evaluation (e.g., compute and data transfer) are executed by a single agent (the VIFI Orchestrator), without any manual intervention or apriori knowledge on the scientist's part. Second, the science algorithms including summarizing each dataset and calculating the evaluation metric are encapsulated in re-usable portable containers, which can be

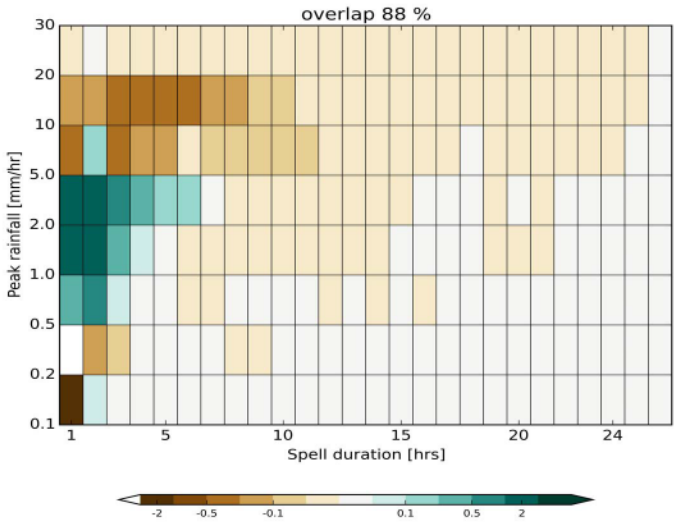

Fig. 8. The JPDF differences of NU-WRF with the 24-km resolution from GPM's JPDF.

seamlessly deployed and run on any VIFI-enabled Node. Third, computation can be efficiently distributed onto multiple servers, which have direct access to the data. Lastly, only a very small subset of the data (the JPDF results) whose sizes are less than 1 megabytes are transferred over the network, drastically reducing the data transfer time and saving storage capacity on the User Node.

In contrast, the traditional information fabric shown in Figure 3-A and 6 has many disadvantages including: 1) Large portions of time and network resources are taken up by transferring and downloading massive datasets (much of which may not be relevant to analysis), to the user's node. 2) High storage capabilities are required at the user's node to store the massive data size during analysis. 3) All computations are done at user's node which limits concurrency to user's node parallelism capability. 4) Users must install required algorithms and related dependencies on their machines. On the other hand, VIFI alleviates traditional data fabric problems and constraints as shown in Figures 7, 4, 3-B and 5.

Table II compares transmission and execution times between the traditional centric architecture (Figure 6) and the VIFI-enabled architecture (Figure 7) for the Earth Science use case when implemented on the AWS EC2 machines whose specifications are given in Table I.

As shown in Table II, downloading model and observational data from remote data sites (Data Sites 3 and 2 in Figure 6) to a user node (Data Site 1 in Figure 6) took 246.088 sec, and 126.934 sec, respectively, even with a high bandwidth connection between servers located in the same cloud network in our POC implementation. The download time of model data is about double the download time of the observation data, since the size of the model data ($\approxeq$ 21 GB) is about double the size of the observation data ($\approxeq$ 11 GB). The total data transfer time for model and observation data to a central user node is therefore 372 sec., even when data sites are within the same cloud network with a 750 MBps bandwidth interconnectivity. In most practical situations, data sites will be distributed across networks and regions (and even across countries) with significantly lower bandwidths, thereby resulting in slower data transfer rates between the sites.

In contrast to the traditional data-centric architecture in Figure 6, the VIFI-enabled architecture in Figure 7, and as explained in Section IV-B, requires the user to submit only the required scripts and configuration files from user node (Data Site 1 in Figure 7) to the remote observation and model nodes (Data Sites 2 and 3 in Figure 7). As shown in Table II, the transmission times of user's scripts and configuration files from user's node (Data Site 1 in Figure 7) to observation node (Data Site 2 in Figure 7) and model node (Data Site 3 in Figure 7) are only 0.183 sec and 0.145 sec, respectively, because of the small size of the user's scripts and configuration files ($\approxeq$ 8 KB). The evaluation script (i.e., comparison between observation and model data) executes at the user's node (Data Site 1 in Figure 7). Thus, the evaluation script does not need to be downloaded to user's node. Data Sites 1, 2 and 3 download the required PAC (i.e., Docker Image) from the PAC registery only if the required PAC does not already exist at Data Sites 1, 2 and 3. For fair comparison against the traditional centeric approach, we set up an AWS EC2 Container Registery (ECR) [23] to host PACs for our POC. PAC size is small (578.13 MB) compared to the model and observation data (21296 MB and 11373.308 MB, respectively). Thus, PAC download times to each of Data Sites 1, 2 and 3 in Figure 7 (0.156 sec, 0.128 sec, and 0.135 sec, respectively) are small compared to model and observation data download times to user's node (246.088 sec, and 126.934 sec, respectively). Most importantly, the PAC download only occurs when PAC does not already exist at the Data Site; therefore, the time taken to transfer a Docker image is a one-time cost and need not be taken into account for benchmarking evaluation purposes. Users and data scientists typically use common environments and therefore can resue existing PAC Docker images, with only the scripts being updated for new data analytics needs. **Therefore, the benchmarking evaluation using data transfer time metrics shows that VIFI is about three orders of magnitude faster than the solution using the traditional data-centric architecture (0.34 sec. vs 372 sec.)**, even when the data sites in our evaluations are in the same intranet with 750MBps bandwidth interconnectivity. As datasets get larger and distributed across networks, regions, and countries, the VIFI-enabled architecture is expected to show much faster improvements in efficiencies, in comparison to the traditional data-centric approach that require movement of large data repositories across networks.

VI. Conclusions and Future Work

This paper presents the *Virtual Information Fabric Infrastructure (VIFI) for Data-Driven Decisions from Distributed Data* . The paper discusses how VIFI avoids the problems inherent in traditional data fabric approaches e.g., inability to move massive amounts of data, impracticality to move con-

TABLE I
AWS EC2 MACHINES SPECIFICATIONS FOR VIFI POC

Model	m4.xlarge
CPU	4
Mem(GB)	16
SSD Storage	EBS-only
Storage(GB)	50
BW(Mbps)	750
OS	Ubuntu

Data sites are in the same cloud network for the POC with very high bandwidth interconnectivity (750MBps) between data sites; in most practical situations, data sites will be distributed with internet connectivity and significantly lower bandwidths, thereby resulting in slower data transfer rates between sites

TABLE II
COMPARISON OF TRANSMISSION TIMES BETWEEN TRADITIONAL CENTRALIZED AND VIFI-ENABLED ARCHITECTURES

Data_type	Size (MB)	Src	Dest	Time (Sec)
Traditional Centralized Architecture				
WRF24 (model)	21296	Data Site 3	Data Site 1	246.088
GPM (observation)	11373.308	Data Site 2	Data Site 1	126.934
VIFI-Enabled Distributed Architecture				
Docker Image *(one-time transfer)*	578.13	AWS ECR	Data Site 1	0.156
Docker Image *(one-time transfer)*	578.13	AWS ECR	Data Site 2	0.128
Docker Image *(one-time transfer)*	578.13	AWS ECR	Data Site 3	0.135
User script & config file	0.008	Data Site 1	Data Site 2	0.183
User script & config file	0.008	Data Site 1	Data Site 3	0.145
Result files	0.156	Data Site 2	Data Site 1	0.186
Result files	0.156	Data Site 3	Data Site 1	0.246

tinuously updated data by bringing analytics to data locations to enable information discovery that could not be previously explored. The paper also evaluates and provides benchmarks of VIFI impact on a real Earth Science use case. VIFI is expected to show much faster workflow runtimes as datasets get larger and more widely distributed. The overall VIFI architecture is immediately scalable to new data-sites by simply installing the VIFI components and registering required PACs.

Future enhancements to VIFI includes providing resource management and scheduling based on current and expected workloads to optimize concurrent request processing, extend logging and monitoring capabilities, improved user interfaces, data management, and security capabilities and encryption, and evaluations on other application domains. The Earth Science use case will be extended by including additional observational data or output models, integration with other teams like the Earth System Grid and Coupled Model Intercomparison Project (CMIP) Phase 6, and scaling data and computational approaches across distributed modeling and observational data centers.

ACKNOWLEDGMENT

Funding for this research was provided by the National Science Foundation (NSF) Data Infrastructure Building Blocks (DIBBs) Program under Award number 1640818. A portion of this work was performed by the Jet Propulsion Laboratory, California Institute of Technology under contract to the National Aeronautics and Space Administration. The authors also acknowledge Rick Hudson for his support on VIFI project planning and logistics.

REFERENCES

[1] J. Manyika, M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh, and A. H. Byers, "Big data: The next frontier for innovation, competition, and productivity," May 2011.

[2] "Geospatial big data: Challenges and opportunities," *Big Data Research*, vol. 2, no. 2, pp. 74 – 81, 2015, visions on Big Data.

[3] R. J. Sandusky, "Computational provenance: Dataone and implications for cultural heritage institutions," in *IEEE International Conference on Big Data (Big Data)*, Dec 2016, pp. 3266–3271.

[4] J. P. Cohn, "Dataone opens doors to scientists across disciplines," *BioScience*, vol. 62, pp. 1004 – 1004, 11/2012 2012.

[5] R. Pordes, D. Petravick, B. Kramer, D. Olson, M. Livny, A. Roy, P. Avery, K. Blackburn, T. Wenaus, F. Wrthwein, I. Foster, R. Gardner, M. Wilde, A. Blatecky, J. McGee, and R. Quick, "The open science grid," *Journal of Physics: Conference Series*, vol. 78, no. 1, p. 012057, 2007.

[6] I. Sfiligoi, D. C. Bradley, B. Holzman, P. Mhashilkar, S. Padhi, and F. Wurthwein, "The pilot way to grid resources using glideinwms," in *Proceedings of the 2009 WRI World Congress on Computer Science and Information Engineering - Volume 02*, ser. CSIE '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 428–432.

[7] J. Towns, T. Cockerill, M. Dahan, I. Foster, K. Gaither, A. Grimshaw, V. Hazlewood, S. Lathrop, D. Lifka, G. D. Peterson, R. Roskies, J. R. Scott, and N. Wilkins-Diehr, "Xsede: Accelerating scientific discovery," *Computing in Science Engineering*, vol. 16, no. 5, pp. 62–74, Sept 2014.

[8] I. Foster, "Globus online: Accelerating and democratizing science through cloud-based services," *IEEE Internet Computing*, vol. 15, no. 3, pp. 70–73, May 2011.

[9] K. Chard, S. Tuecke, and I. Foster, "Globus: Recent enhancements and future plans," in *Proceedings of the XSEDE16 Conference on Diversity, Big Data, and Science at Scale*, ser. XSEDE16. New York, NY, USA: ACM, 2016, pp. 27:1–27:8.

[10] B. Allen, J. Bresnahan, L. Childers, I. Foster, G. Kandaswamy, R. Kettimuthu, J. Kordas, M. Link, S. Martin, K. Pickett, and S. Tuecke, "Software as a service for data scientists," *Commun. ACM*, vol. 55, no. 2, pp. 81–88, Feb. 2012.

[11] D. Medvedev, G. Lemson, and M. Rippin, "Sciserver compute: Bringing analysis close to the data," in *Proceedings of the 28th International Conference on Scientific and Statistical Database Management*, ser. SSDBM '16. New York, NY, USA: ACM, 2016, pp. 27:1–27:4.

[12] http://jupyter.org/.

[13] https://docs.docker.com/engine/swarm/.

[14] https://nifi.apache.org/.

[15] G. J. Huffman, D. T. Bolvin, D. Braithwaite, K. Hsu, R. Joyce, C. Kidd, E. J. Nelkin, and X. P., "Nasa global precipitation measurement (gpm) integrated multi-satellite retrievals for gpm (imerg)," *ATBD Version 4.5*, 2015.

[16] C. D. Peters-Lidard, E. M. Kemp, T. Matsui, J. A. Santanello, S. V. Kumar, J. P. Jacob, T. Clune, W.-K. Tao, M. Chin, A. Hou, J. L. Case, D. Kim, K.-M. Kim, W. Lau, Y. Liu, J. Shi, D. Starr, Q. Tan, Z. Tao, B. F. Zaitchik, B. Zavodsky, S. Q. Zhang, and M. Zupanski, "Integrated modeling of aerosol, cloud, precipitation and land processes at satellite-resolved scales," *Environmental Modelling and Software*, vol. 67, pp. 149–159, 2015.

[17] M. S. Bukovsky, "Masks for the bukovsky regionalization of north america, regional integrated sciences collective," 2011.

[18] E. J. Kendon, N. M. Roberts, H. J. Fowler, M. J. Roberts, S. C. Chan, and C. A. Senior, "Heavier summer downpours with climate change revealed by weather forecast resolution model," *Nature Climate Change*, vol. 4, no. 7, pp. 570–576, 2014.

[19] A. Inc, *Amazon Elastic Compute Cloud (Amazon EC2)*. http://aws.amazon.com/ec2/#pricing: Amazon Inc., 2008.

[20] J. Murty, *Programming Amazon Web Services - S3, EC2, SQS, FPS, and SimpleDB*. Farnham: O'Reilly, 2008.

[21] https://hub.docker.com/.

[22] H. Lee, L. Cinquini, D. Crichton, and A. Braverman, "Optimization of system architecture for big data analysis in climate science," in *2015 IEEE International Conference on Big Data (Big Data)*, pp. 2169–2172.

[23] https://aws.amazon.com/ecr/.