CI Compass

NSF Major Facilities Cloud Use Cases and Considerations

January 12, 2024
Version 1.0
CI Compass (https://ci-compass.org/)

Preferred Citation: G. Bruce Berriman, Brian Dobbins, Jeremy Fischer, Bob Flynn *et al.*, NSF Major Facilities Cloud Use Cases and Considerations, Version 1, January 12, 2024, DOI: 10.5281/zenodo.10481410

Authors:

G. Bruce Berriman- IPAC-NASA Exoplanet Science Institute, California Institute of Technology
Brian Dobbins - National Institute of Atmospheric Research
Jeremy Fischer - Jetstream, Indiana University Bloomington
Bob Flynn - Internet2
Jeffrey Glatstein - Ocean Observatories Initiative, Woods Hole Oceanographic Institution
Rajiv Mayani - Information Sciences Institute, University of Southern California
Loïc Pottier - Information Sciences Institute, University of Southern California
Craig Risien - Ocean Observatories Initiative, Oregon State University
Benedikt Riedel - IceCube Neutrino Observatory, Wisconsin IceCube Particle Astrophysics Center
Mats Rynge - Information Sciences Institute, University of Southern California
Erik Scott - RENCI, University of Northern Carolina - Chapel Hill
Tyson Swetnam - CyVerse, University of Arizona
Amanda Tan - Internet2
Chad Trabant - SAGE/GAGE, EarthScope Consortium
Karan Vahi - Information Sciences Institute, University of Southern California
Don Brower - Center for Research Computing, University of Notre Dame
Charles Vardeman - Center for Research Computing, University of Notre Dame

This project is supported by the National Science Foundation Office of Advanced Cyberinfrastructure in the Directorate for Computer Information Science under Grant #2127548. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the funding agency.

Table of Contents

Executive Summary **1**
1. Introduction **2**
2. Cloud Opportunities for Major Facilities **5**
2.1 Availability 6
2.2 Scalability, Capacity, and Flexibility 6
2.3 Cost-Transparency and Efficiency 8
3. Cloud Providers **10**
3.1 Commercial Cloud Providers 10
3.2 Academic Cloud Providers 11
4. The Spectrum of Cloud Deployment Options **14**
4.1 Case Study: NCAR 14
4.2 Case Study: IceCube 15
4.3 Case Study: SAGE/GAGE 16
5. Cost Considerations **18**
5.1 Managing the Economics of Cloud Platforms 18
5.2 How Are Cloud Costs Estimated? 19
5.3 Storage and Transfers 20
Storage Classes 20
Egress Costs 20
Strategies for Mitigating Storage and Egress Costs 21
5.4 Processing 22
Range of compute instances 22
On-Demand vs. Reserved vs. Spot Instances vs. Savings Plans 22
Cost Drivers 22
5.5 Other Costs 23
5.6 Managing Costs 24
General Strategies 24
Housekeeping and Maintenance Strategies and Practices 25
Cost Savings Through Negotiation 25
5.7 Cost-Benefit Analysis and Return on Investment 26
5.8 Case Study: Ocean Observatories Initiative (OOI) 27
5.9 Case Study: IceCube 28
6. Navigating Vendor Lock-in **31**
6.1 Why Would an Organization Want to Switch Vendors? 31
6.2 What Leads to Vendor Lock-in? 32

Data Transfer Risk 32
Application Transfer Risk 32
Human Resources Factor 32
Authorization Lock In / Security Policies 32
Cost Risk 32
6.3 Strategies for Avoiding Vendor Lock-in 33
Evaluation of Cloud Vendors 33
Ensure Data Movement 33
Application Design 33
Use Principles of Reproducibility 34
6.4 MF Perspectives on Vendor Lock-in 34
7. Training: Cloud-ready Workforce 35
7.1 Cloud Learning Pathways 35
Foundational/Basic Cloud Training 35
Advanced Cloud Training 36
7.2 Cloud Workforce Development 36
7.3 Peer Cohort Asynchronous Learning Method (PCALM) 37
8. Summary 38
9. Acknowledgments 39
Appendix A: Cloud Building Blocks 39
A.1 Storage 40
A.1.1 Archival 40
A.1.2 Cloud Native Formats and Higher Level Data Services 41
A.2 Data Access / Dissemination 41
A.2.1 FAIR Data Principles 41
A.2.2 Democratization of Access 42
A.2.3 Egress 42
A.3 Infrastructure as Code 42
A.4 Processing 43
A.4.1 Streaming Data and Workflow Management Systems for the Cloud 43
A.4.2 High-performance and High-throughput Computing 44
A.4.3 Serverless Computing 45
Appendix B: Glossary 47
References 55

Executive Summary

The National Science Foundation (NSF) supports over 25 Major Facilities (MFs) that serve as cornerstones for the science community. These MFs, characterized by their continuous operations, large-scale and sophisticated data collection, and broad user communities, represent long-term investments intended for multi-decade operations.

Cloud computing presents a wealth of possibilities for these MFs. It offers an extensive range of services, including data storage, archival, processing, and sophisticated data access. Scalability is a distinct advantage of cloud platforms, allowing facilities to transition smoothly from initial development stages to high-demand phases. This shift, however, does not have to be all-or-nothing. MFs can leverage the cloud selectively, optimizing their resources without resorting to an all-in approach.

Case studies, such as those of the National Center for Atmospheric Research (NCAR), IceCube Neutrino Observatory (IceCube), and the combined Seismological Facility for the Advancement of Geoscience (SAGE) and Geodetic Facility for the advancement of Geoscience (GAGE) facilities, operated by the EarthScope Consortium, offer valuable insights. They underline the benefits of the cloud while also highlighting challenges, particularly in comparison to traditional on-premises (on-prem) infrastructure.

When considering cloud adoption, cost emerges as a pivotal factor. Costs are incurred for data storage, egress, compute resources, serverless functions, and application programming interface (API) requests. It is critical for MFs to understand these costs and factor them into operating budgets and planning.

A significant challenge that MFs might encounter is "vendor lock-in." Transitioning between vendors can become costly, primarily because of differing APIs, tools, and services offered by various cloud providers. MFs adopting cloud tools should employ strategies to mitigate vendor lock-in.

As the cloud becomes more integral to operations, training assumes paramount importance. A workforce adept at navigating cloud technologies will be crucial for MFs to harness the full potential of these platforms.

In essence, cloud computing offers a transformative opportunity for NSF MFs. While it introduces myriad complexities, its advantages are manifold. By viewing cloud adoption as a flexible, case-by-case decision rather than an absolute “all-or-nothing” option, MFs can make informed choices about the level and extent of cloud adoption, which has the potential to significantly amplify their capabilities, enhance research quality, and ensure efficient resource utilization. This report encourages MFs to educate themselves on the fundamentals of cloud computing in order to better understand and leverage the cloud tools and computing model where they best fit MF’s specific needs, optimizing benefits while effectively managing challenges and complexities.

1. Introduction

"The cloud" is a broad term, encompassing a large variety of technologies, designs, and players, both commercial and academic. Cloud technologies have built upon the advances in virtualization in the past few decades, and have enabled the decoupling of an organization's physical infrastructure with where compute resources and storage reside. It represents a transformation of information technology in multiple ways: cloud providers offer vast, configurable resources, enabling new approaches to address compute & data challenges. Cloud philosophies encourage rethinking of traditional IT use cases in modern and flexible ways. Cloud tools introduce new powerful and efficient approaches to leveraging technology for research. *With science being so intrinsically tied to computing and data management strategies, the cloud revolution raises big questions about where, when, and how major scientific facilities can make the best use of these technologies and approaches.*

The National Science Foundation (NSF) funds approximately 25 Major Facilities (MFs) managed by its Research Infrastructure Office (RIO). "*Major Facilities are defined as shared-use infrastructure, instrumentation and equipment that are accessible to a broad community of researchers and/or educators. These facilities are generally intended to serve the science community that is critical to supporting innovation across the nation"*[1] . These facilities are diverse, spanning a great number of science domains including astronomy, climate research, ecology, natural hazard engineering research, ocean science, physics, and seismology as well as leadership-class computing systems. A central mission of most of these facilities is to manage and operate large-scale scientific data collection equipment such as distributed sensor arrays, telescopes, and advanced detectors, and make this data available to their respective science communities. Some defining characteristics of a MF are:

- **Lifespan**: MFs are a long-term science infrastructure investment and are designed to be operated for at least a decade. Most operate for longer than their design lifetime. MFs may maintain large cyberinfrastructure (CI) that is outside the scope of any single research project.

- **Large Data Repositories:** Almost all of the MFs are continuously or regularly collecting data, resulting in raw and derived data products. The total storage needs for all MFs exceeds several exabytes, with some individual cases expected to reach several tens of exabytes. These datasets then have to be appropriately managed, cataloged, and made available to the larger scientific user community.

- **Continuous Science Operations:** When the instruments and detectors are online, the MFs are in a mode of continuous data capture and processing operations. As a result, they often have a complex and a distributed processing system underpinning their operations. Even when the main science instrument may not be online, the analysis and science operations are essentially a continuous operation.

- **Large User Communities**: MFs have much larger user communities than typical NSF research awards. MFs are often multi-institutional, with staff, researchers, and principal investigators (PIs) spread across multiple institutions. The MFs have a large-scale user base that is not just limited to users and personnel at participating institutions, but often encompasses the wider community, including private entities. For example, the Vera C. Rubin Observatory [2] serves the entire optical astronomy community. Similarly, the Laser Interferometer Gravitational-Wave Observatory (LIGO) [3] Scientific Community (LSC) has 1400-plus members from 127 institutions in 19 countries. In addition, with the advent of multi-messenger astronomy, LIGO also has users from other astronomy fields that are not part of the official LSC.

The above characteristics represent some of the challenges for MFs to adopt cloud technologies. In addition, the long project life spans increase the risk of technical debt accumulating over years of operation. Operations have often been designed with traditional data centers, the existing national cyberinfrastructure, and NSF distributed CI in mind. Using the cloud effectively means re-architecting to better leverage and more effectively use the myriad of services and capabilities offered by the various cloud providers. An important and often overlooked area when talking about the adoption of cloud technologies is the area of training existing staff to use the cloud effectively, supplementing existing staff with new hires conversant with cloud technologies and development operations (DevOps). Another nontechnical impediment is that MFs operate under a fixed budget; cloud usage can be expensive, and the costs can spiral out of control, especially in terms of data aspects if staff are not trained to properly architect and manage its use.

This report is primarily geared toward leadership at the NSF-funded MFs and aims to provide a high-level overview of the potential benefits, risks, and costs of incorporating the cloud for their operations. As such, the report addresses data dissemination, storage, and computing considerations, which almost all of the MFs uniformly face. Wherever possible, the report includes anecdotes provided by MFs that have incorporated or explored the use of cloud computing for their operations. The report is not a deep technical dive into cloud computing nor does it attempt to answer the question “How do you perform cloud computing?” Instead, it focuses on breadth to provide stakeholders with enough information and a starting point for making informed decisions about their organizations. Cloud service providers offer a staggering array of resources and services, with many services intricately layered and interwoven to create comprehensive solutions. Appendix A delves into the multitude of fundamental building blocks that the Cloud provides that may be of interest to MFs for their operations and usage patterns. In addition to storage offerings, this section also covers services supporting data dissemination and findability, processing of datasets in the cloud, and deployment of infrastructure in the cloud. The actual spectrum of services offered is considerably broader, designed to cater to diverse needs and requirements of diverse customers.

A secondary audience for this report is the agencies that fund MFs. The report hopefully provides insights into the complexity of the work that MFs face when they are considering moving all or part of their operations to the cloud. A key area of concern is the NSF funding for data repositories. The MF’s data repositories only grow over time. During the lifetime of the MF, this causes issues as an increasing fraction of the operational funds have to be spent in keeping the data available to researchers and continuously archived. After the MF completes its data taking or operations, archival of the data becomes a significant burden. Currently funded data repositories are agency[1]-, facility[2]-, field[3]-, research area-, or

[1] https://docs.nersc.gov/filesystems/archive/, https://sharing.nih.gov/data-management-and-sharing-policy/sharing-scientific-data/repositories-for-sharing-scientific-data
[2] https://computing.fnal.gov/data-storage-and-handling/
[3] https://www.nsf.gov/geo/oce/oce-data-sample-repository-list.jsp

university[4]-specific. This creates a difficult situation for MFs when it comes to archiving the data. It either has to be funded through the MFs operations, rely on external resource providers, or try to fit into the existing resources. There is currently no clear path for long-term MF data storage and archival.

Though focused on MFs, this report may also be useful to other scientific facilities, collaborations, or research organizations that are looking to leverage cloud computing for their needs. Personnel from various universities' research computing organizations may also benefit from this, as the report focuses on common research computing models, such as computing in the cloud and hosting data in the cloud long term.

[4] https://sdr.library.stanford.edu/

2. Cloud Opportunities for Major Facilities

Appendix A (Cloud Building Blocks) touches on key cloud building blocks that MFs can consider when leveraging cloud infrastructures and solutions. MFs should definitely consider the cloud, but it's important to recognize that using the cloud *does not have to be an all-or-nothing decision*. It's critical to evaluate each aspect of MF-specific infrastructure needs and related cloud offerings, and independently decide to what degree, if any, cloud solutions are applicable. For example, if an MF needs to serve large, in-cloud datasets with minimal compute, it could very easily make the most sense to use cloud-based computing for that task. This would avoid the potential slow and costly data transfers as well as the need for storage to hold a copy of the data on prem. On the other hand, a low-intensity compute task that generates a lot of data kept on prem is likely to be a poor fit for the cloud, due to data transfer speeds and costs.

Some scenarios for considering leveraging cloud tools and solutions:

- Avoiding long queue times and/or maintenance windows on local resources.
- Reducing the need for local infrastructure (racks, cooling, etc.) and physical maintenance staff.
- Accessing the newest hardware (e.g., GPUs, HBM CPUs, ARM CPUs, etc.).
- Accessing simplified, managed services higher in the stack (e.g., Kubernetes, SageMaker, functions).
- Paying only for what you need; unused resources can be shut off.
- Leveraging dynamic resource scalability.
- Providing distributed access for users outside your organization.
- Mitigating risk via off-site data / backups.
- Executing large, independent workloads or batch workflows.

Cloud technologies represent a set of *tools*, not all-or-nothing solutions. Understanding how and where they work well can help determine their best application. And just as some scenarios tend to favor cloud adoption, the balance can shift with other use cases. This balance is determined by both technical and non-technical factors. An example of some scenarios where cloud tools and solutions may not be the first choice include:

- The existing resources and tools currently meet all project needs.
- Other systems can provide the needed resources (eg, ACCESS [4] or PATh [5])
- The MF already runs large-scale resources/data centers with high efficiency and does not need the tooling available in the cloud.
- There are software licensing limitations.
- There are internal funding/subsidy policies that make cloud pricing less competitive.
- The MF has specialized or specific hardware requirements, e.g. tightly-interconnected networking.
- The MF has low-latency application requirements for connected instrumentation and data sources.

In addition to the above, let's examine three of the most commonly advertised benefits for using the cloud, especially related to computing, and give some examples of where they may and may not apply to MFs. Once again, the cloud is not all-or-nothing; it is simply a set of tools and resources that should be used when appropriate.

2.1 Availability

In traditional IT, "availability" has often been a measure of the resilience of services-critical operations that are made "highly-available" with failover. Systems, even large supercomputers at MFs, typically have well over 99% "availability," meaning they're down for maintenance and thus less accessible just 1% or less of the time. This also applies to cloud resources, though the nature of on-demand resources enables cyberinfrastructure professionals to architect systems in ways that are even more resilient than on-prem facilities. A local power outage can, for example, take down unprotected parts of a local data center, whereas cloud resources can be spun up in multiple geographical locations to mitigate such risks.

This can be a powerful use case for the cloud, but like all things, it's not universal. It may be trivial to launch resources to replace common small-scale tasks in the cloud, but novel or larger-scale tasks can still be a challenge. Even cloud resources are finite, and depending on one's need, users can sometimes run into capacity limitations, often without insight into why. One recent example comes from NCAR, where a research team had been using 60 cloud-based HPC nodes on a daily basis for nearly a month and a half without any availability issues, only to struggle to get more than 15 of those same nodes during a live workshop with students. That is, the cloud improves availability on the whole, but it shouldn't be viewed as limitless or automatic.

The cloud also offers a new definition of "availability"—that of offering new kinds of hardware to users, which not only can save costs or improve performance but also enables development on new technologies, which are often adopted by cloud providers on a faster cycle than by MFs. To use another NCAR example, the cloud has been used to test and develop the Community Earth System Model (CESM) [6] on Advanced Reduced Instruction Set Computing (RISC) Machines (ARM)-based "Graviton" processors, which appear promising for the future, and in general, more cost-efficient, HPC systems. Some new platforms being developed by NVIDIA [7] also fall in the same category. Similarly, the latest GPUs have been used for performance testing of other community models, enabling researchers to test the models' development, software environments, and the performance of multiple kinds of hardware to help determine focus areas for future on-prem procurements. This is a powerful and unique offering that the cloud provides, though sometimes requests for these resources take multiple days to provision (in contrast to mere minutes for resources that are less in-demand).

2.2 Scalability, Capacity, and Flexibility

Scalability, capacity, and flexibility of resources is one of the strongest selling points for the cloud. Rather than having to deal with multiple vendors to purchase hardware, deciding what hardware to buy, how to deploy the purchased hardware, etc; the cloud provides a plethora of choices in terms of choosing and deploying hardware. The cloud provides a means to easily scale from early development to heavy use more seamlessly. A facility doesn't need to worry about data center space, cooling infrastructure, or outgrowing any given physical footprint. This is particularly important considering that an MFs computational and data needs tend to grow with time.

Cloud services can potentially enable MFs to scale out their resources on demand. An example of this scalability and flexibility of the cloud is “autoscaling,” i.e., provisioning or de-provisioning resources automatically according to demand-based rules. This is particularly useful when there is a need to respond to high or low demand fluctuations, e.g. during a conference or when processing is time-critical.

Scalability, capacity, and flexibility are particularly attractive during the construction and initial operation phases of an MF. During these phases, the needs and requirements for the CI are fluid. Scientists are still developing data reduction and analysis algorithms to handle the new data and the requirements evolve quickly. For example, during the construction phase of the IceCube [8] detector, additional funding was secured that grew the size of the detector and the size of the data being collected and opened up new science capabilities for the experiment.

It’s important to note that traditional MF computational needs may not scale efficiently across all cloud vendors; one has to differentiate between scaling resources and scaling applications, particularly in the context of HPC. Scaling resources is a strong selling point for cloud vendors; a facility can begin a large-scale project with small-scale infrastructure, growing their data storage or compute resources easily, as needed.

Application scalability is a little more complex, as it depends on the nature of the application. In one example, IceCube and the San Diego Supercomputer Center (SDSC) demonstrated cloud scalability that could meet their needs in 2019 [9].They successfully completed a computational experiment as part of a multi-institution collaboration that marshaled all globally-available for sale GPUs (over 51,000) across AWS, Microsoft Azure, and GCP to perform Monte Carlo simulations of the detector. This showed that the IceCube infrastructure could readily be expanded into the cloud with existing technology and that the experiment could consume these resources readily. Another demonstration of the immense processing capabilities provided by the cloud is the effort by researchers at Clemson who used about 2.1 million virtual CPUs (vCPU) from Google Cloud to improve disaster planning and evacuation plans in coastal areas threatened by climate disasters [10]. These demonstrations also serve as examples of the large capacity that cloud resources can provide with the right architecting. This can be done while leveraging a single or multiple cloud providers.

However, for “tightly-coupled” HPC applications, the cloud is not a monolith, and vendor-specific differences can make a large difference. Climate and earth-system models, for example, are often dependent on low-latency communications across nodes, so differences in networks available in the cloud, and on-premise systems can result in significant performance degradation. Azure and Oracle Cloud Infrastructure provide HPC cloud instances with “industry-standard” high-performance networking hardware (Infiniband). AWS has their own high performance network stack (Elastic Fabric Adapter) [11]. GCP only supports ethernet at up to 100G.

The capacity and scalability of cloud resources is also appealing for collaborations such as those in Multi-Messenger Astrophysics (MMA), where various groups of scientists want to analyze data in a time-bound fashion when an interesting discovery is made by experiments such as LIGO and IceCube. For example, scientists may consider leveraging cloud technologies to detect optical counterparts using the Large Synoptic Survey Telescope (LSST) and other telescopes. In such cases, time is of critical importance, and cloud resources can enable the spin up of resources on demand without waiting in scheduling queues as is typical of academic clusters.

The cloud provides a high level of flexibility by enabling the definition of CI with code (or configuration files), i.e. Infrastructure as Code (IaC). Rather than having to rely on a comparatively static resource pool that on-premise resources provide, using the cloud allows for expanding the resource pool or changing the resource composition by changing a handful lines of code without having to wait for the purchase of new equipment.

IaC in combination with containerization of scientific software allows scientists to reproduce their workflows from the CI up, track changes to the code and infrastructure in version control systems, and perform fine-grained analysis of how changes in infrastructure can affect scientific results and vice versa. In the traditional CI ecosystem, where systems are configured and maintained by center staff and have a limited lifespan, full reproducibility is only possible for a limited time period.

Ultimately, the scalability of the cloud can be used to reduce the time to result. From an MF perspective, this means that increased computational capabilities provided by the cloud could reduce the time needed to analyze datasets or run simulations. Accelerating these computing tasks reduces the time to result or potentially improves scientific results.

2.3 Cost-Transparency and Efficiency

The cloud touts greater cost-transparency and efficiency compared to on-prem CI. Whether the cloud is actually cheaper than on-prem CI is still up for debate. There have been recent analyses and experiments that have revealed mixed results. Some enterprises have seen significant savings from moving from the cloud to on-prem [12]–[15]. On-prem CI can appear cheaper compared to the cloud when considering just storage and compute. Proper cost estimates require well-defined and understood CI needs in order to do accurate comparisons. Few academic enterprises undertake a full Total Cost of Ownership (TCO) evaluation of their systems for the purposes of these comparisons [16]. The cloud changes the expenditure model for CI from a more traditional split between capital expenses (CapEx), i.e. hardware and ancillary physical infrastructure, and operating expenses (OpEx), i.e. human effort, power, cooling, building maintenance, to an OpEx-only model, i.e. human effort, computing time, storage, etc. Accounting practices for federal funds at university funded or federally funded research and development centers (FFRDCs)-based MFs make CapEx exempt from indirect cost. Additionally the OpEx for the hardware, i.e. power, cooling, and building maintenance are generally covered by the hosting institution. This reduces the OpEx of CI for MFs significantly compared to the cloud, where MFs pay these OpEx. Another accounting practice that can cause wide variation in cloud cost estimates for MFs is whether the MF-hosting institution charges indirect cost on cloud expenditures. For example, the University of Washington and the University of California systems do not charge indirect cost on cloud computing, while the University of Wisconsin system charges a reduced indirect cost rate. Most institutions do not differentiate on indirects. Other supporting infrastructure that one has to pay for in the cloud, e.g. networking, is also covered by hosting institutions or other federally funding agencies, e.g. ESNet [17]. Thus the overall cost to the funding body of on-premise CI for MFs is hard to quantitatively estimate.

The cloud also provides a plethora of different services that can reduce the overall cost from both an operational CI and science perspectives. From the operational perspective, using

Kubernetes-as-a-Service (or any other as-a-service offering), for example, reduces the overall time needed to deploy and optimize a Kubernetes cluster, which in part brings savings in personnel time. Similarly, data analysts could use Machine Learning-as-a-Service to reduce their time to solution. Whether the extra cost of using these services is equivalent to the human time saved is difficult to estimate.

Overall, the cloud provides significantly higher cost-transparency compared to on-prem. The question whether there are cost savings from moving into the cloud remains open. If we purely consider the hosting cost for the MF, i.e. hardware and support infrastructure, on-prem is often cheaper than the cloud. There are, however, significant hidden costs for the hosting institution and the funding agency such as having to set up infrastructure from scratch and scale it to MF needs, all of which are hard to quantify.

The cloud, with its elasticity, is an ideal computing model for quickly changing and growing environments, e.g. construction phase of an MF. Once the baseline computing requirements and needs have been established, e.g. during the MF operations phase, hosting these resources on-prem may reduce cost. However, the cloud remains relevant during the MF's operations phase as it can supplement the existing on-prem CI in case of high demand or for time critical processing.

3. Cloud Providers

3.1 Commercial Cloud Providers

The major commercial cloud providers are Amazon, Google, and Microsoft. Oracle has made efforts to get traction in the research space, but adoption remains limited. Some federal research projects are using IBM as well.

Amazon - Amazon Web Services (AWS) [18] is the oldest commercial cloud service provider, and it currently provides the widest range of cloud services. It is characterized by a deep and varied set of tools which allow users to develop creative solutions limited largely by their imagination alone.

Google - Google Cloud Platform (GCP) [19] is known for strong AI/ML tools, impressive experience with image processing, and a powerful and easy-to-use data warehousing solution. It is seen as more cost-effective, and its pricing model is considered easier to understand. GCP offers a subscription program for predictable pricing on computing projects.

IBM - IBM Cloud [20] has a strong focus on hybrid compute, from bare metal servers to containers on Red Hat OpenShift or IBM Cloud Kubernetes services. They do not, however, have a hybrid storage solution. While traditionally relying on proprietary solutions, they have embraced open source solutions like Apache OpenWhisk for serverless compute. Since 2019, IBM has gone to great lengths to make sure its industry-leading AI platform Watson is multi-cloud cloud-friendly.

Microsoft - Microsoft Azure (Azure) [21] is widely used in the business community due to its deep history with enterprise productivity tools. Azure has strong hybrid cloud support. Whereas AWS has many discreet tools, Azure encapsulates processes for greater ease of use at the expense of some flexibility.

Oracle - Oracle Cloud Infrastructure (OCI) [22] leverages the company's history with database management, enterprise applications, and data processing in its cloud infrastructure offerings.

The commercial vendors are constantly innovating in an attempt to get or stay ahead. There are multiple approaches to deciding where to invest your time and resources:

- There are a number of sites online that do a comparison of the relative strengths of the various cloud vendors in a given area. One should check both the business motivation of the authors and timestamp of the document to judge bias and relevance. It is always a good practice to review more than one source.
- Examine what the cloud vendors do with their own technology. Two simple examples are:
    - Through Google Earth [23], Google has developed deep expertise in image processing. If you are a researcher using something like photogrammetry, you might consider talking with Google.
    - Amazon's commercial "bookstore" business has given them tremendous experience with database processing at speed and at scale.

- Each of the cloud vendors has engineers and architects dedicated to assisting researchers. It is very instructive to sit down with them to talk through your workflows and get their recommendation on how they would natively architect it on their platform. Are they able to accommodate the tools that you are familiar with? What are the benefits of any trade offs they might ask you to make? What kind of cost projections can they give you?
- If you find that multiple platform options work for your pipeline and you are unsure which one to choose, consider which vendor you or your support team are most familiar and comfortable with.

When considering using commercial clouds for your operations, it is also worthwhile to consider what type of data is being used. If datasets in question have commercial applications, then you might tilt towards commercial clouds, as they have incentive of providing computing time against those datasets.

3.2 Academic Cloud Providers

Academic clouds, both regional and national, reduce barriers to researchers adopting and utilizing the cloud. Specifically, academic clouds generally benefit institutions and researchers alike which lack the capital to support operations on a commercial cloud or the technical expertise to buy hardware and operate on-prem. Commercial cloud offers substantial scalability, but at a price. The overall operational savings and the cost-benefit ratio for academic clouds are yet to be determined [24]–[26]. Academic cloud providers include the NSF supported Chameleon [27], CloudLab [28], CyVerse [29], [30], Jetstream2 [31]. Most academic clouds use OpenStack, a free open source cloud computing platform. One feature of public cloud resources is an emphasis on usability for users who are not experienced with clouds.

Cost containment for commercial clouds is a major issue for researchers [26], [32], [33]. The NSF supported CloudBank [34] is intended to help funded research projects manage their commercial cloud costs and academic cost overheads.

The NSF funds several cloud resources for research and development purposes. These range from the experimental Chameleon [27] and CloudLab [28] projects that focus on computer science applications to the production Jetstream2 [31] cloud for general research and engineering across all disciplines. These efforts are often of limited value for MFs due to scope (e.g. Chameleon and CloudLab are primarily for NSF CISE researchers) and often have uncertain futures (e.g. initial grant length for Jetstream (Jestream2's predecessor) was five years with no promise of a follow-on grant). Jetstream2, which came online after the MFs began assembling this report, has a grant cycle of five years and a five-year follow-on system and may be of more interest and potential use to this audience.

The XSEDE [35] allocations process for Jetstream had also been a potential stumbling block since MFs could only plan year to year with allocations. That said, there have always been options in the form of discretionary allocations to help accommodate projects. With ACCESS, the follow-on to XSEDE, multi-year allocations are going to be standard-fare for infrastructure allocations and others that can justify the need. This should help alleviate a potential pain point for MFs. Additionally, the review process with ACCESS is changing to have tiers that have either service provider only reviews or much more limited review than the XSEDE review panels of the past. The work that the ACCESS Allocations [36] team has done (and is doing) should make academic research clouds like Jetstream2 a viable cloud option for MFs.

While the capabilities offered from academic clouds will never be able to rival commercial clouds, the following are resources built by teams that already support researchers and educators. Academic providers such as ACCESS typically have additional funding [37] to provide support to users and researchers. Dedicated support staff are available that can help researchers get started and, in some cases, even help with prototyping. They often understand the challenges presented by the research environments and are in excellent positions to support MFs and their research workloads. Support for these academic clouds is generally prompt, responsive, and can often be more flexible with finding solutions for researchers and infrastructure providers. Additionally, while academic clouds cannot provide all of the resources MFs may need, they can provide reliable resources for portions of their infrastructure and are an excellent place to explore and experiment with new workflows with little to no risk.

Table 1: Publicly available academic cloud service providers. Some projects allocate resources based on "standard units," which typically convert to CPU/GPU hours or storage space (GiBs) of arbitrary value (e.g., 1 SU = 1 CPU hour).

Platform Name	**Type**	**Service Provider**	**Availability**	**Allocation Type**	**Funding Source**
Chameleon[27]	Bare metal cloud, OpenStack cloud	University of Chicago, Texas Advanced Computing Center	By Request	Service Units (SU)	NSF
CloudLab[28]	Bare metal cloud	Clemson, University of Utah, University of Wisconsin	By Request	N/A	NSF
Jetstream2[31]	OpenStack cloud	Indiana University	Allocation via ACCESS-CI	Service Units (SU)	NSF
National Research Platform[38]	Federated Kubernetes	SDSC, Nebraska, Massachusetts Green High Performance Computing Center	By Request	N/A	NSF
CyVerse[39]	OpenStack Cloud, Kubernetes, HTCondor	University of Arizona, TACC	Free (basic account), Subscription, Contracts	Core hours and storage space ($)	NSF, USDA, Arizona Board of Regents, Subscriptions

4. The Spectrum of Cloud Deployment Options

We often hear of organizations "moving to the cloud," but it's important to realize that a successful CI strategy is rarely an "all-or-nothing" approach. Understanding what parts of the infrastructure map well to cloud providers and what is best in terms of economics, usability and maintainability is a large, complex, and organization-specific task. For example, an organization with very sporadic compute needs may benefit from moving those to the cloud vs. using dedicated resources that may end up being idle most of the time; however, the decision could still depend on factors such as the volume of data required for computation, the time it takes to upload data to the cloud when needed, and the associated storage costs. At the same time, a compute-focused HPC facility operating at peak load would likely not benefit from moving *that* compute capacity to the cloud, but *could* benefit from offloading suitable "burst" or high-priority compute to the cloud [40] in order to meet deadlines, albeit at a cost higher than using local resources. Another consideration for offloading to the cloud, is dealing with computations that need to be done in a time bound manner, e.g., in multi-messenger astronomy to perform analysis to detect optical counterparts of a gravitational wave detection by LIGO. Outside of the science use cases, administrative considerations like disaster recovery & continuity policies seem like a strong fit for clouds due to the availability of multiple vendors with resources in an ever-growing number of locations.

The key is to evaluate the best tool for each part of the CI infrastructure, and make decisions accordingly. In some cases, this may result in a complete move to a single cloud provider. In others, it could be the development of a hybrid approach balancing on-prem facilities with multi-cloud capabilities. And in others, this may result in learning lessons from clouds and then applying them to on-prem design decisions. Fundamental to all of these options is the understanding of the near and long term needs of CI personnel and users, and a realization that a cloud transition is an investment and a process, and isn't likely to bring immediate benefits. One recent article discussed that challenges in implementing cloud solutions in industry have resulted in many chief information officers (CIOs) still awaiting a return on the initial investment [32].

4.1 Case Study: NCAR

The National Center for Atmospheric Research (NCAR) [41] makes use of the cloud in multiple ways, each illustrative of the concepts above. In terms of compute, NCAR's on-prem facilities include a large supercomputer and local, fast file systems. An internal analysis of costs and performance showed that moving to cloud-based resources for these capabilities would be prohibitively expensive, and thus an ineffective use of fixed or limited budgets. However, the cloud does offer benefits when training new users on running earth system models. The cloud has a lot of ancillary services built-in (e.g. account management, two-factor security configuration, and data retention planning) and by its on-demand, ephemeral nature, simplifies the setup process for training needs compared to on-prem. This is quickly becoming a standard training approach for NCAR, and also has a synergistic effect of enabling cloud use for the wider research community that NCAR supports.

A different example from NCAR is the relationship built with AWS's OpenData [42] program for free, public access to open scientific datasets. Many datasets are available from on-prem file servers at NCAR, but this still means that *all* accesses to these datasets go to NCAR's servers - multiple requests are constrained by limited bandwidth, especially as data continually grows larger in size. In some early coordination with AWS, NCAR placed copies of popular datasets, like the CESM LENS [43], [44] data, in the OpenData program. This improved its accessibility to the scientific community (since it is now in two places), and the public cloud offers co-located compute and AI/ML resources with which to analyze it. In a newer effort, NCAR has also arranged with AWS to place some data necessary for running models in two geographically distributed OpenData buckets. This is a unique capability the cloud offers that a MF with a single on-prem facility cannot, and ensures that research teams across slow, international links can benefit from fast access to data. Since NCAR research teams have access to this data locally, it doesn't make sense to *move* this data to the cloud, but *copying* it and having it freely available is a win for everyone. Some larger MF's such as LIGO and Atlas have built distributed data serving infrastructures that provide similar geographical access as the Cloud.

4.2 Case Study: IceCube

The IceCube Neutrino Observatory (IceCube) [45] is a neutrino telescope that is embedded in the Antarctic ice sheet at the South Pole. The telescopes instrumented a cubic kilometer of the ice sheet with 5160 individual sensors and 324 sensors on the surface for Cosmic Ray studies and vetoing Cosmic Ray air showers. IceCube [45] has made use of the cloud in several different ways. While most of these uses have been exploratory or experimental, they have been able to reveal the advantages and disadvantages of the cloud compared to on-premise or national cyberinfrastructure.

The most informative of the cloud uses were the "cloudburst" experiments, which tested: cloud scaling, cost-effectiveness of the cloud, and network bandwidth between cloud and research networks.

- **Cloud scaling:** IceCube was able to aggregate over 51,000 GPUs across three cloud vendors [9] (AWS, Azure, GCP) and 28 different cloud regions to generate simulation datasets for the experiment. This allowed IceCube to see how many resources are available in the cloud, test whether their infrastructure was capable of utilizing this massive amount of resources, and whether the cloud provides the scale required to run IceCube simulation workflows.
- **Cost-effectiveness of the cloud:** From the pricing and performance data from the cloud scaling experiment, IceCube was able to determine the most cost-effective GPU instances. IceCube used this information to run a second experiment to accumulate an ExaFLOP-hr worth of GPU computing to perform Monte Carlo simulations of the detector, in particular propagate photons through the South Polar ice the detector is embedded in.
- **Network bandwidth between cloud and research networks:** The two previous experiments were performed with the input and output data staged in and out of the respective cloud region the workload was running in. The third experiment was used to determine the network bandwidth between cloud providers and research universities via existing research networks. IceCube was able to rent enough direct links between cloud providers and research networks to saturate the 100G link between UW-Madison and Internet2.

The combination of these three experiments informed IceCube that the cloud could be an alternative for IceCube's CI needs. However, it also revealed that the cost would be significantly (up to a factor 10) higher compared to the current on-premise and distributed infrastructure used today.

IceCube has also used the cloud as a way to test different GPU sharing technologies, in particular NVIDIA's Multi-Instance GPU and time-sharing of GPUs through Kubernetes (K8s) features in GCP. This has yielded insight into how to more effectively use the existing GPU resources and potential ways to improve GPU usage, particularly with newer generations of GPUs.

IceCube also tested using GCP's K8s-as-a-service offering (GKE) for their multi-messenger astrophysics workloads to host IceCube's infrastructure and using GKE's autoscaling feature to grow the computing resources as needed. This test was successful and IceCube is considering using GKE as the baseline compute infrastructure for most time-sensitive MMA workloads.

A potential other use for the cloud IceCube has been exploring is deploying the SSH gateways to the IceCube South Pole infrastructure in the cloud. The current cybersecurity settings with the Antarctic Service Contractor require a constant IP address for these gateways. Hosting said the IP address and VM in the cloud would half the number of gateways needed to run (1 at the University of Wisconsin and another at the University of Maryland). The much higher cost (2-10x depending on the details) compared to the on-prem hosting has prevented IceCube from moving forward with this cloud use. IceCube is still considering renting an IP address in the cloud to ensure it has a third option for disaster recovery.

In production, IceCube uses the cloud for ancillary services, in particular programmatic DNS updates for services run in local K8s. The cloud has provided features that were not available at the time through the existing DNS service provided by University of Wisconsin central IT. This gave IceCube the ability to iterate much more rapidly when it came to network-exposed K8s hosted services. The ease of use compared to and cost for these ancillary services are low compared to creating a DNS automation service.

4.3 Case Study: SAGE/GAGE

The combined Seismological Facility for the Advancement of Geoscience (SAGE) and Geodetic Facility for the advancement of Geoscience (GAGE) facilities, operated by the EarthScope Consortium [46], is taking an "all-in" approach to cloud adoption. The new, unified cloud-based platform (in development) will replace geographically distributed on-premise data centers that have operated for many decades and could not be easily combined given their specialization and siloing of data pipelines. Data operations supported by this new platform include data capture from field stations and from other agencies and institutions, derivative data product generation, archiving and curation, and distribution to researchers and the public. These activities require a significant amount of data flow, processing, storage, and distribution. Importantly, while there are many constantly running processes in this platform, there are also large volume and compute "bursts" requiring temporary increases in resource utilization.

Some characteristics of cloud services are not an ideal fit for such operations, in particular relatively high storage and data egress costs. Despite these disadvantages, the organization chose to pursue an all-in approach as they deemed the advantages to be greater. The advantages include: first, the opportunity to combine the operations of multiple on-premise data centers into a single platform gaining operational simplicity and efficiency. Second, the cloud provides the opportunity for substantial compute capacity adjacent to the data, eliminating the need to transfer data over the internet, and satisfying a goal that was very difficult with the self-managed on-prem centers. Third, the expandable nature of the cloud allows the platform to grow and accommodate anticipated large data sets much more quickly and easily than planning equipment purchases for, often necessarily, over-specified systems. Fourth, the cloud allows for temporary use of relatively cheap compute and scratch storage to handle episodic data capture and curation activities. There are many other advantages, such as the ability to use incredibly robust and capable services that would cost prohibitive to replicate on-premise such as AWS's S3 storage system.

Recognizing the non-ideal fit of some cloud services, the organization will pursue cost optimization and other activities to mitigate the disadvantages. For example, the use of intelligently tiered storage allows the migration of less-often accessed data to cheaper tiers saving cost. Also planned is the use of a non-cloud or alternate cloud storage system to store a copy of the primary data, which will provide both a component disaster recovery and an alternate data access point to mitigate egress costs.

5. Cost Considerations

5.1 Managing the Economics of Cloud Platforms

Cloud providers have fine-grained cost models, with pricing for almost all services and some associated actions. Some services have a free tier beyond which users start getting charged. Others incur costs at any level of usage. Examples include:

- **Ingress: The cost of uploading data into the cloud**. Commercial cloud vendors do not charge for upload via the internet. Bulk data migration services, such as AWS Snow Family [47], which provide secure bulk transfer at scale, do have associated fees.
- **Storage: The cost to store data in the cloud.** Commercial providers have object, file and block storage options, which are highly scalable and vary in cost based on the desired data durability, access frequency and performance. There are wide variations in cost for various storage options. Storage options are tiered, with rates decreasing as access latency increases.
- **Egress: The cost incurred to transfer data out of the cloud to the internet.** The commercial cloud service providers charge a fee for moving data out of its host region. The data host may choose to make the requester pay the egress fee. Upon request, most research and education organizations are eligible for waivers to offset a percentage of their egress costs from commercial cloud providers.
- **Compute:** General resources, such as virtual machines or high level services such as Kubernetes, have time-based charging, with a variety of cost models, based on machine/services and usage type.
- **Serverless/Functions:** A cloud computing model in which the cloud provider manages the server infrastructure and dynamically allocates resources as needed. Serverless is used for event-driven applications, backend processes, Internet of Things (IoT) applications and other loosely coupled computing needs. It is priced on a per-call basis.
- **API Requests:** The cost is based on the number of API requests made for listing, uploading and downloading objects.

This pricing model provides flexibility in the design and operation of cloud-based services, but requires staff to understand what generates costs and learn to financially engineer their applications and systems. Best practices recommend architecting proof-of-concept versions of workloads to estimate costs before deployment. Regular monitoring of costs after deployment is an essential component of learning cloud cost management. This section emphasizes strategies for assessing and managing costs, mapped onto the use cases described earlier. We will quote cost examples to illustrate strategies, but they should be considered as snapshots applicable only at the time of quotation.

Many institutions levy indirect costs on cloud services but not on capital expenses such as on-prem hardware. This has a significant impact on the cloud cost compared to the on-prem. This increases the cost of cloud services by up to 70% above what is advertised on the cloud vendor website. Additionally, there is generally no to little cost in terms of electricity or cooling for on-prem hardware, which produces an additional operational cost saving to on-prem hardware compared to the cloud. This practice varies from institution to institution, and some now waive the overhead altogether. Eligible NSF investigators may also apply to CloudBank [34] to use cloud allocations without indirects. Cloudbank is limited to the CISE directorate researchers at the moment.

The EDUCAUSE Total Cost of Ownership document [48] gives a thorough summary of the issues to be considered in costing on-premise services for comparison to cloud services, including a particularly useful summary of hidden costs. It ends with a worksheet that can be used to estimate and compare costs.

5.2 How Are Cloud Costs Estimated?

In a traditional on-prem data center, an organization procures servers, storage and networking equipment, intended to last five-plus years, through a large upfront Capital Expenditure (CapEx) that is considered a long-term investment. Generally, these CapEx's are based on the estimated maximum usage over the life of the hardware, and are therefore oversized and often under-utilized much of the time. While the upfront costs are large, the costs of owning, maintaining and operating (usually through a service contract or hosted at the recipient institution) are largely predictable and therefore easy to manage.

By contrast, commercial cloud computing services represent Operational Expenditure (OpEx), which incurs only usage costs for hardware maintained by the provider and included in the usage costs. The provider continually updates their hardware, so centers can avoid the technical debt that can be incurred with the use of aging equipment. The cloud model offers considerable flexibility to architects and developers creating new and powerful services. These benefits must be balanced against the fact that ongoing costs can be much harder to predict, plan and manage than one-time capital expenses. Costs must be continually monitored.

A useful starting point in assessing costs is to assume that all actions can in principle incur their own charge—transfer of data, network actions, computing, and storage. In practice, none of the major cloud providers now charge for ingress of data over the internet, nor do they charge for transfer of data between data centers in a given region. They do, however, charge for egress of data back out to the internet or between cloud regions. All organizations should inquire as to their eligibility for a data egress waiver. The major commercial cloud service providers offer limited data egress waivers to research and education organizations. These typically provide a waiver of egress costs equivalent to a percentage of the organization's overall cloud expenditure. While large organizations can often offset the majority of their egress costs, the data transfer volume of MFs is exceptional and should be discussed with the provider's architects and monitored for ongoing coverage. Providers offer limited free access to new users, and credits for educational and research use. While these options may offer limited access, they can be very useful for verifying cloud approaches and prototypes. While free, uploading large volumes of data over the Internet can be very slow, and the use of "wide pipe," such as that provided by Internet2, may be warranted.

Whether cloud services are more or less expensive to deploy than their on-prem equivalents depends on the scale and type of service - whether the service is simply moved to the cloud or re-architected to be cloud native - and the larger architectural needs of the system. There is no hard rule and each use case should be considered separately. Computing power, storage required in the moment, and their pricing models themselves offer far more options than is seen in traditional on-prem systems. These allow the creation of highly tailored, and therefore more efficient, choices, but require a greater learning curve and navigating more decision points. Some of these decisions may well include moving some services to the cloud while leaving others on-prem.

The following sections describe the tiers of service with reference to AWS; similar remarks apply to other large commercial clouds (e.g., Azure, Google Cloud, Oracle Cloud etc.).

5.3 Storage and Transfers

For the majority of MFs, making the collected data available to their user communities is one of their primary mandates. These datasets have become exabyte-sized in some cases. Storing data in the cloud can be attractive because of the associated gains in durability and availability of data. The cloud providers provide multi-tiered storage services, with the cost varying widely, generally falling as latency increases and frequency of access decreases. However, it is critical to analyze and project the storage costs upfront based on how the data are to be made available to users.

Storage Classes

There are three broad classes of storage that all major cloud providers offer:

Standard Storage for data that are most frequently accessed and downloaded by users, and require low latency and high throughput, e.g. science data products created by processing pipelines, or newly acquired high profile data collected from a sensor. Generally, standard object storage such as Amazon S3 are good examples for serving this data.

Midline Storage for data that are accessed less frequently, but requires rapid access when needed. For example, science data that is slated to be reprocessed, simulation data specific to a certain science question, or data that has been accessed recently. Generally the cheaper tiers of standard object storage services such as Amazon S3 Standard-Infrequent Access (S3 Standard-IA) [49] would be good choices for these datasets.

Coldline Storage for data that are accessed infrequently, and are mainly inward facing to a MF. For example, the original raw data that a MF retrieves from their instruments, and deep backups are classes of data that would be stored in coldline storage. Services such as Amazon Glacier [50] and its competitor variants are good candidates for storing these datasets.

Egress Costs

While most commercial providers do not charge for ingress, they do charge for transfers out of the provider to the internet or between their hosting regions. Egress charges are based on a number of factors, including the region the data resides in, the services being used, and, naturally, the amount of data. The costs can be high, and underscore the need for understanding the cloud financial model and configuration options before opening your data for download. From the perspective of MFs, the egress costs are often the hardest to estimate and most important to architect correctly because of the MF's goal of enabling access to their data for their communities. These communities are generally unaccustomed to paying the egress costs. Some MFs already have petabytes of data collected over the years and have thousands of users downloading data. The entire Large Hadron Collider (LHC) [51], [52] has in excess of an exabyte of archived data . Starting around 2027, they expect to produce about an exabyte of new data a year. IPAC [53], a NASA archiving facility, has 1.5 PB of public data archived on-prem which are made available freely to the users without authentication and receives several million queries per year to retrieve data. Another example is the CyVerse iRODS data store [54] is currently at ~6.5PB in user data growing at ~1PB/yr. They have ~400TB downloaded every month, and ~100TB uploaded. The SAGE facility manages a primary data set of approximately 1PB and distributes more than 1PB per year through more than a million requests per day. If their data store were on a commercial cloud without intelligent architectural decisions, they estimate the hosting and associated egress costs to be as large as their annual operating budget.

If storage is all that is required, rather than access to services and frameworks, there are providers who specialize in providing S3-like storage at low cost and with no or few tiers, and generally low or no egress costs, and also often no API request costs. One example of this is Wasabi [55], whose cost is generally lower than major cloud providers. MFs do need to ensure that the reliability and recovery times are acceptable for their needs.

In general, network transfers into the cloud are free, but transfers via devices that are mailed incur a charge. Transfers within a cloud region are generally free, though transfers across regions or across providers, as well as from the cloud to on-premise storage incur charges.

Strategies for Mitigating Storage and Egress Costs

Aside from exploiting tiering of data, there are other effective strategies for mitigating storage costs. Among them are:

- **Requesting a Data Egress Fee Waiver** or leveraging a contract with one. All major cloud providers have a provision to provide waivers for academic researchers and institutions [56]–[58].
- **Leveraging auto-tiering** services when available such as Amazon S3 Intelligent-Tiering Storage Class [59].
- **Open Datasets:** The major commercial cloud providers offer open dataset programs, whereby an organization can apply to make public datasets publicly available free of charge, generally for a negotiated period of time (usually two years), which can be renewed. Providers do, however, require that such data sets are of broad scientific value. An inherent risk with this approach is that if the agreements are not renewed, the MFs must find another way of financing storage costs.
- **Evolving users behaviors:** Those configuring storage buckets must be trained on how to configure and monitor the services to prevent misuse. Users will require training on how to responsibly download the data. Users should be encouraged to take their compute to the data in order to mitigate excessive egress charges in cases where the data sizes are large.
- **Guard Users Against Overuse:** MFs can consider providing their own interfaces to the users to manage data downloads from the cloud. These can keep track of downloaded datasets and warn users if they are downloading large volumes of data or downloading data already downloaded. In addition, interfaces developed must be made robust against security defects that can be exploited by bad actors who can then initiate massive downloads.
- **Requester Pays Model:** AWS and GCP offer a variety of configurations where datasets can be downloaded under a requester pays model [60], [61]. Services such as these allow MFs to pass the egress costs onto the end users, and enable the end users to subscribe to multiple data sources with the convenience of centralized billing.

5.4 Processing

Range of compute instances

The major providers generally offer the following classes of instances, optimized for different usage: general purpose, compute-optimized, memory-optimized, accelerated computing (such as GPUs), and storage-optimized instances. Amazon currently offers over 50 types of instances across all five classes, and all are charged at separate rates. Cloud providers' offerings are generally most suitable for applications that might be best described as high-throughput computing rather than for tightly coupled workloads. Creation and release of cloud instances can be greatly simplified by use of infrastructure-as-code tools (such as Terraform [62]).

On-Demand vs. Reserved vs. Spot Instances vs. Savings Plans

The most expensive instances are those described as **On-Demand** [63], often referred to as "pay as you go." Users launch, stop and terminate them as needed, and are charged for time used. They are best used for high-burst applications where processing is time-critical, but not for running operations 24x7 in the cloud. Fortunately, there are various options and cost models for making processing more economical, as discussed in what follows.

Reserved instances [64], best provisioned when fully utilized, offer a billing discount, with payment in advance for the use of a particular on-demand instance for a fixed term. The term commitments are usually one year or three years for a particular instance type. These are good options to use when compute cost and its duration can be estimated upfront and can be fully utilized for the contract period: there is no reimbursement or credit for unused resources.

Savings Plans [65] offer a flexible pricing model that can reduce bills by up to 72% compared to on-demand prices, in exchange for a one- or three-year hourly spend commitment. AWS, for example, offers Compute Savings Plans [65]. They are most effective when usage can be predicted beforehand. The article [66] has a helpful comparison of Reserved Instances and Savings Plans.

Spot (preemptible) instances [67] are the cheapest instances. Users take advantage of idle capacity at a steep discount (up to 90%) off the on-demand price, although spot prices fluctuate. The instances can be terminated without notice when demand rises. Users can specify the range of prices they are willing to pay. Codes are checkpointed on termination and can be restarted later. These instances are good for running batch jobs, and jobs that are short lived or can be easily checkpointed. They are not suited for time-critical applications.

Cost Drivers

This section summarizes cost drivers for various types of workloads and scenarios. The cost drivers are:

1. **Compute Cost:** Cost of the VMs and hardware, e.g. RAM, CPU-hours
2. **Storage Cost:** Cost to store your data, at least temporarily, in the cloud
3. **Network Egress Cost:** Cost incurred when transferring data in/out of the cloud. Ingress charges are typically waived and egress charges can be waived as well if overall cost is < 15% of total organizational cloud bill (if you have secured a waiver).

4. **Interconnect Cost:** Workloads that require tightly-connected worker nodes, e.g. weather simulations or molecular dynamics, will pay a premium for worker nodes with these capabilities. At the moment the only cloud providers that provide these services are Microsoft Azure and Oracle.

The predominant cost components depend on the compute model. For example:

- In data-intensive workloads, the biggest cost components are the compute and network or storage cost. Network egress costs are waived if the data stays within the same cloud region or is < 15% of the total cost, or reduced if they stay within the cloud provider. We are exchanging networking for storage costs by staying within the cloud environment.
- Workloads with specific hardware requirements also drive the price. Different generations of CPUs, types of instructional sets (ARM vs. x86), network connectivity (25G to 100 Gbps ethernet, Infiniband, vendor-specific networking, local VM storage type (NVME vs. SSD), etc. have different costs in the cloud. With proper planning, this can decrease overall cost. At the same time, requiring or relying on certain CPU features (e.g. AVX512 extensions), high memory footprint, etc. will increase overall cost.
- Workloads that are able to checkpoint and/or have short run-times (O(1 hr)) can take advantage of spot/preemptible instances. These can be a small fraction of the cost of fully reserved instances.
- Tightly coupled workloads will pay a premium (1.5-3x) for using Infiniband or other high-end network devices.

5.5 Other Costs

Although the discussions about the cost of commercial cloud usage are usually focused on the price of cloud compute and storage usage, there are many additional aspects to the cost. Some of these aspects apply to usage of non-commercial cloud resources (e.g., agency-funded national resources).

These costs exist when even a small number of researchers at an institution use cloud resources and generally need to be covered by the institution, even if the cloud infrastructure usage is not charged (e.g., if the cloud usage is covered by NSF credit awards, if a cloud vendor provides free credits, or if the resource allocation is provided at no cost by a federally-funded cloud platform project such as JetStream2). These costs can be roughly broken down into a number of types (in no specific order):

- **Financial processes and effort** - for example, the ability to pay for cloud resources via research grant accounts rather than a researcher's personal credit card, or reimbursement processes for personal credit card spending.
- **Legal processes** - information and support for researchers signing end-user agreements that put their institution and (university-owned) IP at risk, the need to provide institution-wide business associate agreements for computing on some types of data (HIPAA-aligned, for example), the ongoing need to support institutional agreements with commercial cloud vendors, etc.
- **Software licensing** - the need to enable researchers to use the licensed software they use on on-premise resources when they move their computing to a commercial cloud platform that does not include software.
- **Technical processes, resources, and ongoing maintenance of the processes** - enabling campus researchers to use single sign-on to access off-campus cloud resources, allowing and

enabling the transfer of data to and from those clouds, creating template frameworks for researchers who are not cloud experts to transfer their computing as seamlessly as possible to cloud platforms, without vendor lock-in, storage resources on-campus to enable researchers to maintain their data after their cloud usage is completed, etc.

- **Training** - this goes beyond training researchers on using the cloud resources. This includes the need to train the personnel supporting the researchers in their use of cloud resources. They need to constantly keep up with changes in commercial cloud offerings as well as federally-funded cloud platforms, be aware of (and advocate for) on-campus resources to support cloud usage, and understand research needs so as to match researchers with the correct cloud resources while being able to communicate needs to institution administration so that the appropriate institutional resources, processes, and knowledge are available for cloud users.

5.6 Managing Costs

Regardless of the use cases, diligent attention to a number of procedures can avoid unnecessary costs and prevent financial headaches:

General Strategies

- Invite the cloud vendor's engineers/architects to suggest an architecture and cost model for your application. Make them work for you to ensure you are building an optimized workflow.
- Learn how to use cost calculators offered by providers. With some practice, they yield estimates that are accurate to about 80%.
- Do a pilot project on the cloud infrastructure if possible, to understand the data and resources needed and costs incurred. Often it is very hard to figure out upfront what sort of costs one may occur.
- Consider requesting research credits to defray part of the costs. While the cloud service provider will rarely fund an entire project, they will often provide support and credits for the proof-of-concept so that you do not waste funds figuring out how to begin. They will sometimes help fund research when there is a pressing public good (e.g., COVID-19 research) or when it will result in positive press for all parties involved.
- Understand processing and storage expectations and how they translate to cloud services. How long will you need cloud services? Do you need to re-process your data periodically?
- When eligible, take advantage of third-party services, such as CloudBank to eliminate indirect costs charged by US institutions on services.,
- Optimization takes work, intention and skill. If done properly, it can pay big dividends.
- Be aware that prices can change without notice and different services do not change at the same rate.
- Understand the use cases and requirements thoroughly in selecting from multi-tiered services (e.g. storage options, compute instances).
- Organizations should explore existing consortium contracts they may be eligible for before considering negotiating a direct contract with a commercial cloud provider, particularly if this is done out of cost concerns. Consortium contracts generally have pre-negotiated discounts that are greater than a single organization could get on their own unless they are prepared to lock into a substantial annual spending commitment. Such pre-commits only make financial sense when the cloud usage is understood. Some MFs may be eligible for federal contracts, others may consider procurement co-ops like E&I [68], or community consortia like Internet2, where members of the

research and education community join together to negotiate legal and financial terms favorable to that community.

Housekeeping and Maintenance Strategies and Practices

- Strategies
    - Realize that cost calculators, while useful, only give *estimates* of costs.
    - Run a test/prototype first and examine the changes in costs before scaling up.
    - Leverage cloud's scalability rather than building infrastructure for the peak of the project. Use standard instances, the smallest needed for the job, before customizing. Customize or scale up only when necessary.
    - Think carefully about the storage costs. Leverage storage tiers and dynamic scaling.
    - Automate the daily shutdown of any infrastructure not in use.
    - Take full advantage of IaC to make efficient infrastructure that can be shut down or deleted entirely when not in use.
    - Understand the long tail of costs; avoid having jobs run longer than necessary.
    - Download only the data you need, and consider which data can be replicated elsewhere if needed. If you are downloading everything, re-think your cloud approach. Downloading large volumes of data can be very expensive
- Keeping costs under control
    - Monitor costs daily.
    - Set up the available billing alarms.
    - Automate shutdown of jobs once a spending cap is reached.

Cost Savings Through Negotiation

Any organization wishing to make use of the cloud should consider negotiating a contract with a cloud provider. Cost estimates, once complete, provide a basis for negotiations with providers. This is best done by a dedicated cost management team familiar with the technical needs as well as business and financial management, and by engaging directly with providers [69]. For example, the Vera Rubin Observatory (VRO) [2] and the Space Telescope Science Institute (STScI) [70] went through this exercise. After thorough prototyping experiments with Google Cloud and AWS, the Rubin Observatory set up an interim data facility with Google Cloud as the provider for three years.

STScI used AWS through a third-party provider [71] until 2018, and then engaged in major negotiations to establish an enterprise agreement with AWS. The best practices that emerged in negotiating discounts included: an upfront payment discount ("savings plan") of 50-60% (best for predictable workloads and a plan for storage costs upfront with commitment for a 10-20% discount. The option to add credits can be added to discounts and options to waive some egress costs—typically 15% of annual spending (this is now provided as standard by Google to Internet2 members) can be negotiated. NOTE: All of these discount and savings strategies should be available regardless of whether you are on a direct agreement or through a reseller. The Rubin Observatory negotiated a similar arrangement with Google Cloud [71].

5.7 Cost-Benefit Analysis and Return on Investment

The survey reported by Lifka *et al*. [72] showed that cost management was one of the major considerations for researchers and facilities planning to use cloud platforms, and the same remains true today. Cost-benefit analysis and Return on Investment (ROI) have been explored by several working groups and MFs in the last 10 years. Broekema *et al.* [73], for example, developed a conceptual model that assesses the cost and lifetime science value of a compute system. Makhlouf *et al*. [74] showed how to apply transaction cost theory to cloud computing. Even with these public studies, it is difficult to ascertain the actual ROI for academic research of national cyberinfrastructure resources, and evaluate public cloud services versus commercial cloud services. Geva *et al.* [75] surveyed academic institutions and found on-prem resources were rated as having higher ROI for the majority or all of the institutions' research computing needs. Stewart *et al.* [24], [25] noted that investing in local facilities is generally more cost effective than utilizing commercial clouds for academic research, but that it was possible to get commercial cloud costs within the actual costs of maintaining local facilities. Importantly, Stewart *et al.* [24], [25] noted that there is no direct comparison between public and commercial cloud service providers. The CASC Return on Investment Working Group [76] studied the current state of academic cloud and data center usage [26] , including ways in which ROI is analyzed in making usage choices.

"*From mid-2013 through 2017, Big Red II delivered 637,874,648 core hours to the IU community. The acquisition cost (AC) of the system was $7.5M; including system administration, power, cooling, space, and extended system maintenance, the total investment by IU over that period was $10,132,097. If one were to procure the same number of comparable instance hours within AWS (c4.large), the total value TV AWS would be $24,877,111–$37,634,604, assuming reserved up-front payments in 3-year or 1-year terms, respectively. In other words, a lower bound on the ROI of investment in local resources for Big Red II ranges from 2.5 to 3.7.*"[77]

The challenge of calculating ROI for commercial cloud usage and the need for accurate ROI accounting to justify funding resource procurement and leasing, as well as for hiring and retention of skilled personnel to manage resources on commercial cloud, requires more attention [26]. Relevant cost components for commercial cloud include cost of an appropriate number and size of instances versus utilization rate, as well as storage costs and egress costs for data from cloud resources. Hidden costs for commercial cloud include contract and billing management and the effort and personnel time required to move or optimize usage to create the greatest discounts in billing [25]. Investment in local facilities is a more cost effective tactic in general than using commercial cloud for research and development [75]. However, it is possible to bring commercial cloud costs within that of on-premise deployments: *"a comparison between AWS spot instances and Fermilab HEPCloud for analysis of CMS experiment data showed that it is possible to get costs of commercial cloud services close to the actual costs of local facilities. After using over 15 million hours, the steady-state cost of AWS came to 1.4±12% cents per core-hour, which is not much larger than the estimated 0.9±25% cents per core-hour for the Fermilab data center.*" [24], [78]

There have been a number of cost-benefit analyses published for a range of scales and use cases. These can provide guidance in performing cost-benefit analyses for MFs. Specifically within the astronomy community, there have been a variety of cost studies on processing data from various telescopes. For example, processing of imaging of Square Kilometer Away (SKA) scale data in different environments demonstrated *"the potential for low cost computing provided by cloud facilities, but also the potential for cost blow out if large amounts of data are kept long term in high availability storage"* [79]. For HPC applications, a NASA report from 2018 [80] concluded that *"commercial clouds do not offer a viable,* *cost-effective approach for replacing in-house HPC resources at NASA."* Additionally, they pointed out that the lack of high speed interconnect availability on AWS does hamper performance of HPC applications. It is worth noting that Azure [81] makes computing resources available with high speed interconnects. Also it is important to note that cost is not the only consideration in the cost-benefit analysis. For example, in the case of Very Long Baseline Interferometry (VLBI) correlation, Gill *et al.* [82] highlights the potential of the cloud *"to significantly reduce data processing times and allow the processing of more science experiments in a given year for the petabyte-scale data sets increasingly common in both astronomy and geodesy VLBI applications*." It also emphasizes the benefits of democratizing science where the cloud gives the opportunities to researchers that don't have access to large compute clusters to leverage the compute and storage capabilities of the cloud for a short period of time.

5.8 Case Study: Ocean Observatories Initiative (OOI)

The Ocean Observatories Initiative (OOI) [83] is a science-driven ocean observing network that delivers real-time data from more than 800 instruments to address critical science questions regarding the world's oceans. OOI data are freely available online to anyone with an Internet connection. OOI collects a significant amount of data consistently throughout the year, on a daily basis. The program is expected to collect data for a 25 year period. As this data builds over time, the cost of processing, storage and delivery will increase as well.

As of March 2022, the amount of data collected and delivered by OOI can be measured as follows:

Data Collected
• Cassandra 28 Nodes totaling 30TB
• Postgres database size of 350GB
• 112 billion rows of numerical data to date with new data ingress every second
• Raw data 1.1PB to date with expected growth rate of doubling every 3 years
• HD video (8,000 hrs), digital still pictures (900,000), bio-acoustic sonar, hydrophone acoustic sampling (280,000 hrs)

Data Delivered
• 25 million data requests per month
• These include 5.6 million data requests with 1000 rows or greater of data returned
• External real-time systems interrogate OOI API every second to every 30 seconds.

In May 2022, personnel from OOI attempted to roughly estimate the costs to run OOI production environment in the cloud. The numbers in the table below were computed using the Dell EMC Live Optics application and cloud provider cost estimation webpages. Their estimates show that it would cost approximately $4.3M annually to operate the OOI production environment in the cloud. It is important to note that OOI also operates 3 development environments that are not included in these estimates. This represents a 2x factor cost difference between an on-prem hardware architecture and a cloud implementation with cloud being the higher cost.

Table 2: OOI annual cloud cost estimates (as of May 6, 2022) based on liveoptics.com and AWS, Azure, and GCP cost estimation webpages.

1yr commitment VMs running 24/7	Prod Uframe VMs		Prod Cassandra VMs		7PB HA Raw Data Storage (eg S3 Standard)	Egress Fees (assuming 10 TB / month)	F&A Costs	Total Annual Cost
	Compute	Storage	Compute	Storage				
Amazon Web Services (AWS)	$248,636	$74,601	$621,906	$262,080	$1,764,000	$10,800	$1,446,281	$4,428,304
Google Cloud Platform (GCP)	$216,192	$93,250	$573,522	$327,600	$1,932,000	$10,200	$1,529,091	$4,681,855
Microsoft Azure	$218,162	$78,797	$498,174	$206,438	$1,428,000	$10,500	$1,183,434	$3,623,505

5.9 Case Study: IceCube

IceCube has been collecting data during the construction phase with a partial detector since 2006 and was completed in late 2010. IceCube is currently undergoing an iterative upgrade program that will extend the lifetime of the project to the 2050s. The first stage (IceCube Upgrade) is focused on neutrino oscillation and physics studies, as well as improving the calibration of the detector. IceCube Upgrade is slated for deployment in 2026. The second stage (IceCube Gen2) is focused on high energy neutrino astrophysics and is currently in the research and development phase. The current plan is for IceCube Gen2 to be constructed in the 2030s.

As of March 2023, IceCube collects data at the following rates (roughly):

Data

- Raw data rate: 1 TB/day
- Satellite data rate: 70 GB/day
- Offline data rate: 140 GB/day

There are higher and final level data products. At the moment, IceCube stores onsite:

- Analysis and User Data: 3 PB
- Experimental and Simulation Data: 7 PB

IceCube has two offsite data archives and a single onsite archive. A copy of the raw unprocessed data is stored at NERSC's tape facility. The Deutsches Elektronen-Synchrotron holds a copy of the satellite and offline data. The onsite archive of the raw data consists of hard drives that were transported back from the South Pole.

- DESY Archive: 1.1 PB
- NERSC Archive: 6.1 PB

From our experience, having more than one archival copy is essential. During IceCube's most recent data reprocessing campaign, a small fraction of the archive was corrupted and we had to use the data from the secondary copy. We only keep a single copy of the processed data because they can be regenerated from the raw data.

The processing of raw detector data consumes 300 CPUs at the South Pole. The offline processing uses an additional 220k CPU-hours per year. Overall, IceCube uses roughly 50M CPU-hours for user data analysis. These workloads typically require 4 to 6 GB of RAM per core.

IceCube analysis requires simulating the detector response to known signals to be able to differentiate the background from the signal and know how the detector responds to a given signal. Overall, the aim is to have the simulation's lifespan 10x longer than the data's lifespan. Given IceCube's high data rate and diverse science goals, IceCube has only been able to produce 1x the detector lifetime. One of the limitations to generating the desired amount of simulation are the required CI to do so. IceCube simulation consumes roughly 200M CPU-hours and 5M GPU-hours. For a cloud cost analysis, IceCube will also have to consider its network utilization, assuming IceCube either does not fit into a single region or single cloud provider. At the moment, IceCube transfers roughly 175 TB/day to and from Madison.

One thing to note is that IceCube members have additional local computing resources. The following table lists the expected costs of operating IceCube in the different clouds[5].

Table 3: IceCube Cost Analysis across various commercial Cloud providers

1yr commitment VMs running 24/7	**Data Analysis VMs**	**Simulation VMs**		**Accessible Data Storage (eg S3 Standard) 11 PB**	**Egress Fees (assuming 100 TB / month)**	**Total Annual Cost per year**
	Compute	Compute	GPU			
Amazon Web Services (AWS)	$1,128,463	$4,553,448	$5,192,928	$2,851,766	$95,846	$13,821,551
Google Cloud Platform (GCP)	$1,310,708.16	$4,935,221	$2,182,990	$2,774,481	$82,327	$11,285,727
Microsoft Azure*	$958,694.40	$4,899,994	$2,492,605	$2,266,553	$76,728	$12,795,574

* Reserved for 1 year

Several caveats:
- GPUs are the most cost-effective for the IceCube workload.
- Storage cost could be optimized with tiering, at the moment this is not done, so this is a 1:1 comparison

[5] May 2023 prices

For comparison, the total operations budget, including program administration, detector operations, CI, etc., is $38,380,300 for the period of April 2021 through March 2026 (five years)[6]. In addition, the collaboration as a whole spends on average ~$500,000 per year on hardware.

[6] https://www.nsf.gov/awardsearch/showAward?AWD_ID=2042807&HistoricalAwards=false

6. Navigating Vendor Lock-in

Vendor lock-in refers to the predicament that MFs may find themselves in, where the cost of switching to a different vendor is prohibitively high, i.e., enough to dissuade them from changing vendors. Each of the cloud vendors (academic or commercial) have their own API's, tools, and services that enable users to deploy their operations in their respective cloud environments. Once an major facility decides to move part or all of their operations to the cloud, they are faced with a choice on how to set up their tooling, i.e. whether to use the provider API and services directly or use third-party tools such as Terraform [62], Pulumi [84], etc., that allows users to set up their operations in a cloud agnostic manner. Another aspect of this is data lock-in, where the commercial cloud providers make it easy and free to upload data into their cloud while making it expensive and slow to download data from the cloud. Due to financial pressures, an insufficient workforce, or the need to avoid interruptions to business operations, the customer is "locked in" to what may be an inferior product or service [85]. This section attempts to list out and summarize findings from various online articles [85], [86], [87],[88] that talk about vendor lock-in, associated risks, and mitigation strategies that one can employ to avoid vendor lock-in.

6.1 Why Would an Organization Want to Switch Vendors?

Organizations often shortlist cloud vendors based on multiple criteria. These evaluations are based on a variety of factors ranging from an initial evaluation experience of the product, marketing materials provided by the cloud providers, and often experiences from other customers. However, these evaluations do not always result in an accurate picture of the vendors' capabilities. As a result, there is a risk that MFs can experience buyer's remorse if the resulting cloud deployment fails to live up to its promises of improved performance, reliability, scalability and costs.

A perceived risk is that a cloud provider may choose to discontinue unsuccessful or unprofitable offerings, or may choose to break backward compatibility and introduce newer related capabilities. In reality, as far as the major cloud providers are concerned, this risk is minimal and they generally go to great lengths to provide backward compatibility.

A bigger risk exists when using a small cloud vendor or provider. Most of the cloud agreements are often subscription-oriented (i.e., customers are billed monthly) and can often be terminated at the providers discretion. New, smaller vendors, or niche vendors always run the risk of going out of business. This risk is not strictly limited to a small company. For example, in 2016, the large enterprise company Hewlett Packard decided to quit the public cloud business and focus on private and hybrid cloud offerings [89].

A lot of cloud vendors provide attractive introductory pricing, as discussed in Section 5. There is always a risk of the cloud provider changing the pricing significantly at the end of the negotiated duration, leading to significant cost increases for an MF that may force it to look at alternatives.

6.2 What Leads to Vendor Lock-in?

Once an MF starts using the cloud for operations, it may find itself locked-in to the cloud provider. It is important to remember that not all causes of vendor lock-ins are engineered by the vendors. Often, vendor lock-in arises from the convenience of using the cloud provider API's and services directly. We list some considerations for you to be aware of that can lead to a vendor lock-in, as discussed in various articles [85]–[87].

Data Transfer Risk

Cloud vendors tend to incentivize customers to use their products by pricing features asymmetrically. For example, vendors can make it convenient and cheap to move data into their platforms, but asymmetrically price moving data out of their platforms. Cloud vendors could use proprietary formats for storing data making it difficult for MFs to migrate to other vendors.

Application Transfer Risk

Cloud vendors provide many proprietary services and APIs that make it difficult to switch providers. Some of these APIs are adopted as *de facto* standards, like S3 API, but most APIs remain specific to the vendors. Thus, changing cloud vendors would require re-engineering the solution to using the new vendors APIs, or even to use open-source solutions to reduce lock-in risks.

Human Resources Factor

Though cloud vendors provide low-level infrastructure and open-source software in a service form, most of the cloud ecosystem depends on proprietary aspects. For example, authorization features are usually non-standard and designed to tailor to the vendor's needs. Due to significant levels of proprietary features, it is difficult to expect MF employees to be proficient in multiple cloud vendors. Switching cloud providers can result in significant costs in retraining or restaffing the team.

Authorization Lock In / Security Policies

Systems commonly provide ways to delegate authentication to different authentication providers. For example, a system administrator can configure AWS to delegate authentication using identities defined in Auth0, which is a SaaS based identity provider. However, no such well-known mechanism to delegate authorization to an authorization provider exists. Essentially, cloud vendors provide ways to delegate authentication, but users are always required to use the vendors' authorization mechanisms, like identity and access management (IAM) for AWS, etc., leading to authorization lock-in. Changing vendors would require translating security policies defined in one system into differently defined policies of a different vendor. Currently, neither an authorization provider to whom authorization can be delegated to exists, nor any well-known tool to translate policies from different cloud vendors exists. The only approach is to custom-build a solution similar to an API gateway, which would be tedious and expensive.

Cost Risk

Cloud vendors incentivize users to use their services by providing introductory or negotiated pricing, but these have an end date. Cloud pricing can vary depending on availability, customer demand, and other external factors. Vendors can also use usage data to customize prices for the specific user, just enough to prevent you from switching vendors.

6.3 Strategies for Avoiding Vendor Lock-in

MFs choosing to run computations on commercial clouds may do so based on different criteria of feasibility. The most obvious criterion is the cost. One strategy to avoid ballooning costs when introductory pricing expires is for MFs to use self-managed, open-source software run in the cloud, or use cloud native versions of the open-source software. For example, you may choose to install and run a MySQL database server on an Elastic Compute Cloud (EC2) instance, or use a native AWS database offering like the AWS Relational Database Service (RDS). Such an approach enables MFs to migrate away from one cloud vendor to another, move services back on-prem, or adopt a hybrid approach. However, there may be performance degradation and more work on the part of the MF IT team.

Evaluation of Cloud Vendors

While evaluating cloud vendors, you should develop a lock-in metric and use it as a primary metric in the evaluation criteria. An example of a lock-in metric could be the data egress cost, i.e., evaluators could compare egress costs across the vendors and compute the expected cost of moving to a different vendor based on past estimates of amount of data generated by science runs. Identifying the metric to use will depend on the MFs mission. MFs who have a mandate to perpetually make the data available to the world would give higher importance to the egress costs, whereas MFs who perform compute intensive tasks would give higher importance to a VM pricing metric.

Ensure Data Movement

MFs should evaluate the use of open-source data formats over vendor-specific proprietary formats. MFs should develop applications with application-specific backup and restore capabilities with offsite or off-vendor backups.

Application Design

The most important strategy to avoid lock-in is to develop a loosely coupled application over a tightly coupled application.

You should consider using tools such as Terraform and Kubernetes for account provisioning and container orchestration wherever possible. Terraform is an open-source infrastructure as a code software tool that enables you to safely and predictably create, change, and improve infrastructure and Kubernetes is an open-source system for automating deployment, scaling, and management of containerized applications. Such tools have been around for a while and are withstanding the test of time. Also, many of these tools are developed by the parent companies of the cloud vendor themselves ensuring long-term support, and availability of experienced personnel.

While developing a loosely coupled application having appropriate abstractions is the key. For example, as outlined in this blogpost [90] , as far as possible *"depend on Open Source Software with hosted cloud versions, not proprietary cloud-vendor-only software,"* and design "*abstractions that allow swappable* *implementations anytime you have to talk to a cloud provider API directly"* are some things to keep in mind.

Use Principles of Reproducibility

The use of containers greatly aids in making computations reproducible, since they only require servers capable of running containers and researchers need not worry about the code, its dependencies, and their corresponding versions, environments, etc. Often, MFs run computations to conduct cutting-edge research. The research may be under intense scrutiny, especially for new discoveries. For these purposes, the results need to be peer-reviewed and all data and computations need to be made available so the findings can be reproduced. Containers can help in this case as well. . All cloud vendors provide native services which support containers. For example, GCP Google Kubernetes Engine, AWS Elastic Container Registry, etc. Also, cloud vendors provide higher-level services based on containers, like AWS Batch, Kubernetes, Container Registry, etc.

In addition to packaging applications as a container that can be deployed on a server in the cloud, it is also important to keep in mind principles of "infrastructure as code". IaC refers to the management of compute, storage, and network resources through software, abstracting away the physical details of the hardware. Declarative templates, generated with for example Terraform or Pulumi, allow for reproducibly deploying composable infrastructure across different cloud providers.

6.4 MF Perspectives on Vendor Lock-in

Avoiding vendor lock-in is a goal that should be pursued as much as practically possible. With cloud provider-agnostic tools (e.g., Terraform) and fundamental services (e.g.,object storage) having more and more compatibility between providers, the possibility of building cloud-agnostic data systems continues to improve. For small data systems that utilize only basic services, such as storage and VM-based compute, achieving cloud-agnostic deployment is reasonably feasible without much added complexity. As the data systems become more complex, the need to use higher-level services (such as serverless compute, integrated messaging systems, process tracking, etc.) become much more important in order to achieve efficient and scalable operation. There are not always equivalents to these newer, higher-level services between providers, and when there are equivalents, they are usually not compatible. Additionally, new services are nearly constantly being introduced with improvements that a facility may wish to use For large data systems utilizing these newer services, the burden of maintaining the capability to deploy to multiple cloud provider systems increases dramatically and, in many cases, is not possible without making choices that use less than ideal services.

As an illustration, SAGE and GAGE have been engaged in a multi-year effort to merge operations and move to the cloud. This complex undertaking started with a goal of being cloud agnostic and planned to use recommended tools such as Terraform to avoid being locked-in. However, as the effort progressed, they realized that there are subtle to significant differences between cloud vendors that make it virtually impossible to have the same platform deployment description for multiple vendors. Instead, they opted for a strategy of service adoption case-by-case to mitigate the degree of vendor lock-in. For example, using Kafka for data streams and PostgreSQL systems and alternate-vendor services such Auth0 (now Okta) when vendor-provided solutions showed no significant advantage.

7. Training: Cloud-ready Workforce

Training is essential to using cloud technologies effectively and efficiently. These technologies represent a fundamentally different paradigm in computing. They allow one to architect the technology to fit the problem, rather than allowing the technology to dictate how you structure your question. Decisions about when, where, and how to optimally leverage cloud computing must be based on a thorough understanding of the tools, techniques, and best practices, and how they differ from traditional computing models.

The breadth of tooling, flexibility of configuration, and granularity of fee-generating actions all speak to a need for comprehensive training for architects, developers, and operators in order to use these powerful tools with fiscal and technical effectiveness.

There are many cloud training options available. However, the needs of the research and scientific community differ from that of industry in general. Often, the architects and developers at MFs may need to be well-versed in multi-cloud infrastructure and understand how to translate research needs and existing infrastructure into sustainable, extensible, and cost-friendly architecture.

7.1 Cloud Learning Pathways

There are multiple tiers of training available, from beginner to advanced. Many training courses are freely available through the cloud providers and lead to certifications (e.g AWS Solutions Architect Associate and GCP Professional Cloud Architect certification, etc.). The learning pathways to these certifications are useful training tools even if the end goal is not a certificate. For example, a foundational learning pathway is useful in validating a broad knowledge of cloud concepts and the products, services, tools, features, benefits, and use cases of a particular cloud provider. Once users or learners have a base level understanding of cloud services, they can proceed to develop fundamental skills to deploy and maintain cloud projects, and, finally, design, implement and manage cloud services at their facilities.

Foundational/Basic Cloud Training

Foundational cloud training is used to introduce CI personnel to the building blocks of using the public cloud. Any potential CI user or builder considering cloud usage should have some fundamental level of cloud knowledge. Topics that should be covered in a foundational cloud course include:

- Introduction to cloud terminology (e.g. EC2 on AWS, Compute on Azure and GCP, etc.)
- Why and when to use the public cloud (see Section 2)
- Identity and Access Management (IAM), User Management
- Compute and Cloud Data Storage options
- Networking basics
- Basic cost management including using cloud calculators and best practices

Many resources exist, including public cloud provider training:

- Resources: Public cloud provider trainings, CLASS, Cloudbank

- Internet2's CLASS Essentials program [91]
- Google:
    - Google Cloud Skills Boost [92]
- Azure:
    - Azure Fundamentals (AZ-900) [93]
    - Azure Machine Learning Fundamentals (AI-900) [94]
- AWS
    - AWS Training and Certification [95]
    - Cloud Essentials Learning Path [96]

Advanced Cloud Training

In order to train a future workforce of CI professionals capable of implementing cloud technologies, advanced cloud training is necessary. The target audience for this type of training are those who will build, develop, administer and maintain cloud infrastructure. Topics that should be covered:

- Data archival/compliance, networking and data movement
- Containerization
- Microservices
- Architecting in the cloud
- Multi-cloud deployment using infrastructure-as-code (IaC / infrastructure-as-software (IaS) tools
- Hybrid cloud architecture
- Resources:
    - Cloud Design Patterns [97]
    - Internet2's CLASS Advanced [98]
    - Google Cloud Skills Boost Learning Pathways [92]
    - Azure Learning Paths for IT Pros, Data and IT Professionals and Developers [99]

7.2 Cloud Workforce Development

There are several "levels" of training and outreach targeted to different communities of cyberinfrastructure professionals: the builders/enablers, the users, and the administrative personnel. For the purpose of this report, we focus on workforce development of CI builders defined as those who deploy, build, manage, and support CI. For MFs, the objectives of training are to enable staff to (1) obtain sufficient training to make decisions around cloud solutions architecture and suitability for adoption, (2) understand cloud costing models, and (3) develop and build cost-efficient cloud infrastructure.

Working on the cloud is a paradigm shift that requires users of the system to understand multiple aspects of the problem. Often, a user of a public cloud platform must also act as the administrator of the account, manage costs, and be knowledgeable in a range of cloud-related services and management tools such as security protocols, Identity and Access Management (IAM), networking, well-architected frameworks, and continuous integration and deployment of code.

Continual workforce development is critical for any IT team to stay informed about the constantly emerging technologies. All the MFs have professional technical staff that have often spent the entirety of their careers at the MFs transitioning from a student/postdoc role to full-time staff. While they are tasked with maintaining the current systems, they need to upgrade their skills to understand and work in a cloud environment. This staff retraining is often overlooked. The lack of knowledge of new technologies can be a stumbling block when trying to make an informed decision about making the transition to cloud. It is important to start this process early, especially for long-term staff members. Providing avenues for training is critical in ensuring these staff do not feel left behind, and are able to make a successful transition to their evolving responsibilities.

Cloud technology covers a vast array of topics, tools, and practices. It is important to prioritize which cloud topics a particular staff member needs to devote time to. Training programs should start with generic cloud concepts before diving deep into any particular specialized tool or technique. MFs should identify early on staff or new hires that have cloud experience and can serve as mentors to the rest of the technical staff.

As an illustration, for SAGE/GAGE's multi-year transition to a cloud deployment, AWS training materials and LinkedIn courses have especially proved useful in training their staff. The cloud vendors often provide staff training as part of enterprise discount packages. Additionally, they bought some seats in Kode Kloud [100] for providing hands-on training for their staff. In their experience, hands-on training is more beneficial when linked to a real work project.

7.3 Peer Cohort Asynchronous Learning Method (PCALM)

Internet2 has developed a peer cohort training model [91] that combines asynchronous lessons with regular synchronous meetings to discuss the technology and its applicability to the cohort's context. Their primary audience is cloud-enablement teams in higher education institutions. The cohort is led by an experienced mentor from the research and education community who can help those new to the cloud tie the material back to the common challenges of providing IT infrastructure in these organizations. The cohort, typically no more than 15 participants, is made up of IT professionals from multiple institutions across the research and education community, working together on a specific certification or defined learning path. The participants go through the online material from Cloud Academy [101] on their own over the course of a week, soundboarding questions off their peers and the mentor via Slack. They then gather weekly to discuss the material and share ideas of how they might use what they have learned in the context of their institution. There are many benefits to this approach. It provides flexibility for the participants to go through the material at their own pace and on their schedule. The meetings allow them to form a broader context for the material that will help them apply it to their own work. They develop a professional network of peers who have shared the experience and whose work they understand and relate to. This model could easily be adapted for the specific needs and cultures of MFs and deliver the same valuable results.

8. Summary

Cloud computing, when examined through the lens of NSF Major Facilities, emerges as a nuanced and multi-layered subject, brimming with variables that demand careful deliberation. It can seem daunting at a first glance, but it's essential to bear in mind the wide-ranging benefits and cutting-edge capabilities facilitated by cloud tools and services. Regardless of the inherent intricacies, the prospective advantages of clouds are substantial. They provide on-demand accessibility to data storage and computing resources, equipping organizations with notable flexibility, adaptability, and scalability.

A major take-away from this report should be that the cloud does not have to be an all-or-nothing decision. We recommend that MFs look at cloud as another tool in their toolbox, and apply the tool where appropriate.

An important aspect that invariably accompanies discussions on cloud technology pertains to cost. Throughout this report, we have showcased various cost computations and experiences related to MFs. While some of these figures might appear staggering, they are understandable considering the substantial data and computational demands of MFs, which naturally result in significant expenses. It is important to bear in mind that the hyper-transparency and granularity of charges in the cloud, while initially challenging, present an opportunity to achieve a level of efficiency and total cost control that is difficult to do on-prem.

In conclusion, it's crucial to recognize that the shift to cloud computing is more than just a technical or financial decision - it signifies a fundamental cultural transformation. Consequently, it brings with it a wave of anticipation and possibility. By exploring cloud computing, NSF MFs can unlock a vibrant and perpetually evolving arena, abundant with novel tools and capabilities poised to substantially augment the research and discovery journey.

9. Acknowledgments

The authors would like to acknowledge Ewa Deelman, Anirban Mandal, Nicole Virdone, Christina Clark, and Brett Beasley for their diligent edit passes on the report and general feedback. The report reads much better for this. We would also like to thank Wendy Whitcup, Jeff De La Beaujardiere, and Mark Miesch for participating in the initial discussions as working group members that helped shape the outline of the report.

Appendix A: Cloud Building Blocks

Most of the NSF MFs are tasked with collecting large amounts of data and disseminating the data to their user communities. These datasets have been collected since the MFs have been in operations and, in some cases, even before or during construction, and have grown over the years (in some cases, up to an order of exabytes [51]). Storing this data and making it available to the users is one of the key operations that many MFs are involved in. Most MFs are mandated to archive their datasets with very long-term, multi-decade retention policies. In some cases, MFs have been mandated to hold onto their data indefinitely and make it available to the community.

With some careful considerations and fine tuning (such as categorizing datasets into different storage tiers based on usage), there is potential for MFs to leverage a "*cloud also"* approach, where they decide to host part of their data in the cloud for various reasons. As part of this approach, an MF may opt to strategically place some of the datasets in multiple global locations, with the aim of providing their customers faster access to these datasets. This approach also lowers the barrier for users from the scientific community who are not part of the Major Facility to compute in the cloud, and thereby reducing the load on MFs' existing data servers that can then be used to serve on-prem users. Such an approach is being actively considered by NCAR [44] for hosting some of the common climate model data required for running various supported climate models.

Cloud service providers offer a staggering array of resources and services, with many services intricately layered and interwoven to create comprehensive solutions. This section delves into some of these fundamental building blocks, though it is important to note that this is not an exhaustive list. The report in general is geared towards NSF MFs and, hence, attempts to cover things relevant to their operations and usage patterns. In addition to storage offerings, this section also covers services supporting data dissemination and findability, processing of datasets in the cloud, and deployment of infrastructure in the cloud. The actual spectrum of services offered is considerably broader, designed to cater to diverse needs and requirements across various domains.

A.1 Storage

Collecting, storing, and disseminating data is a key unifying theme across all of the MFs. The cloud provides a plethora of options for storing data, archiving of data, higher level services for accessing data, and even cloud native formats to efficiently access and use large datasets in the cloud.

The most common data storage option in the cloud are object stores, using services such as AWS S3 [102], Google Cloud Storage [103], and Azure Blob Storage [104], which enable scalable, configurable storage options, including public accessibility by end users. Wasabi [55] is another commercial cloud provider that exclusively focuses on storage solutions and has been used in the past by National Ecological Observatory Network (NEON) [105] to store all their datasets. All cloud providers provide HTTP endpoints for users to download datasets from the Object Stores. The NSF funded Open Storage Network (OSN) [106] is a similar service made available to academic users. OSN is a distributed, data-sharing and transfer service intended to facilitate exchanges of active scientific data sets between research organizations, communities and projects, providing easy access and high bandwidth delivery of large data sets to researchers. All these services (whether commercial or academic) are connected to the internet by high-speed networks. It should be noted that moving data, particularly out of commercial clouds, incurs a cost. In the long run, it might be more beneficial to move your MFs computing and workflows to the data rather than the data to compute. Storage options do change over time: AWS on S3's 17th anniversary recently added seven more S3 options [107].

In addition to object storage, all commercial clouds provide additional types of storage (such as block storage, network-attached storage (NAS)), which are more aligned to the type of storage that gets associated with compute instances in the cloud. Those storage options come into consideration when architecting data processing systems for the cloud [108].

A.1.1 Archival

Depending on the type of data being collected by MFs, clouds can be considered for hosting an archival copy of the data. Archiving in the cloud brings some advantages in terms of the reliability and availability provided especially by the commercial cloud providers. These characteristics can be challenging and difficult to match by the MFs in-house infrastructure. All commercial cloud providers provide special archival services, in case of AWS it is Glacier [108]. While these services are cheaper than object storage-based services, the time to retrieve/recover data can be relatively long; in some cases as long as a couple of days. However, they can prove to be an attractive solution for disaster recovery. Commercial clouds provide a myriad of options for data archiving, which allows for tuning of the number of copies and specifying locations of data centers where archives are kept. For example, MFs such as IceCube[8] do offsite backup at National Energy Research Scientific Computing Center (NERSC)[109]. An inherent risk of this choice is that NERSC is situated close to major earthquake fault lines.

The archival storage solutions provided by various commercial cloud providers eliminates the expense of on-prem backup management, and essentially eliminates on-site infrastructure (i.e., no tapes, etc.). Whether the recovery times are an acceptable consequence of such cost savings depends upon a facility's recovery requirements and expectations [71]. Moreover, recovery also incurs egress costs, which can be substantial for large data transfers and is an important consideration.

A.1.2 Cloud Native Formats and Higher Level Data Services

Storing data in the cloud gives MFs an opportunity to rethink formats in which their data is stored. For example, MFs can consider converting their data to a format optimized for transfers and computing such as Zarr [110] or Kerchunk [111]. Two common use cases are data that is frequently updated in the cloud, and data in which a subset is desired instead of a whole file. Objects stored in an Object Store are usually immutable, and changes to an object result in a new copy to be made, so dividing files into ‘chunks’ via formats like Zarr enable partial updates. TileDB is another format that is optimized for cloud storage with built-in support for compression and encryption and also follows a multi-file data format that allows concurrent updates [112]. Zarr format is being widely adopted by the Earth Sciences Community to read files from cloud storage backends [113], [114], [115]. Similar efforts are underway to provide data in traditional NetCDF format via “Kerchunk” metadata to also streamline reads in a similar fashion to Zarr by minimizing data transfers to only the portion of the data requested.

Commercial clouds provide a lot of tooling such as Google Cloud Platform’s (GCP) BigTable [116], and AWS’s DynamoDB [117] to which data can be stored in a way that makes it easier for end-users to work with datasets.

A.2 Data Access / Dissemination

MFs are not only tasked with preserving long term data, but also with disseminating that data to researchers and other stakeholders. Dissemination includes providing access to the raw data as well as curated, processed, or combined data. The major cloud vendors provide services covering most of these use cases. For example, access to existing data products can be done with direct, access-controlled retrieval from object stores. Another example we have encountered is using cloud provided built-in services like AWS Lambda serverless services for access to databases, repackaging, and light processing such as formatting. When a user searches on a data portal for data or retrieves data from an object store such as S3, a serverless function can be used to process the data before being returned to the calling application. Like most functionality provided by the cloud, similar data access/dissemination functionality could be built on-prem. However, using the cloud provided services frees up developer and system administration time and enables the IT personnel to focus on the higher level access/dissemination, instead of on the underlying service details and maintenance.

A.2.1 FAIR Data Principles

Cloud services can be leveraged to meet funding agency and user community expectations of satisfying FAIR (Findable, Accessible, Interoperable, Re-usable) [118] data principles. FAIR data can be achieved without the cloud, but the wide spectrum of existing data services provided by the cloud helps in architecting FAIR solutions. Metadata can be stored in relational databases, document databases, or graph stores. Search portals can make use of full-text indexing services such as AWS OpenSearch [119], and these services can also provide faceting and filtering. For example, the advanced access controls provided by AWS S3, can be used for fine-grained access control and data access logging. Cloud services make it easy to set up internal systems for tracking data, especially provenance data, that is critical to FAIR [120].

The most common dataset vocabularies are schema.org [121] and the W3C Data Catalog Vocabulary [122], both of which are used as the foundation of data set publication [123] for United States Government Agencies. The United States Federal Enterprise Data Resources maintains a resource catalog [124] which can be helpful as a starting point when considering implementing FAIR principles. For example, NEON has data management documentation pages [125] that describe the cloud data processing pipeline, standards, and policies for data collected and published by the observatory.

A.2.2 Democratization of Access

A common data use case for MFs is to produce data sets, which are then used by scientists in their own analysis so that they can generate additional products and insights. These models often require significant compute resources for processing and therefore their usage can be limited to people who have access to high-performance resources. Hosting such models and datasets in the cloud can serve the important purpose of democratizing access, where anybody can create temporary processing clusters in the cloud, and perform computing close to the data source. One particular advantage of this approach is that the user does not have to wait for data to download, and only pays for computational resources and does not have to pay for any egress costs, which they otherwise might have to do if they download datasets to their own institutions. MFs can also consider a hybrid approach by making a separate, semi-temporary copy of selected data that can be cached in the cloud, with the focus on computing in place rather than downloading. They can opt to continue serving data from their data servers, to enable users who want to download and compute on their local compute resources rather than the cloud.

A.2.3 Egress

Egress refers to the process of transferring data from inside the cloud environment to resources external to that cloud using the general Internet or specific cloud data transfer services. Egress costs are a major challenge for MFs, since they have large datasets and that the community wants to access. On the technical side, cloud services can provide excellent solutions. However, the technical benefits have to be contrasted with the non-trivial cost structure. The latter is discussed in detail under the Cost Consideration (Section 6) below.

A.3 Infrastructure as Code

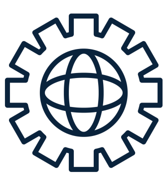
When considering processing in the cloud, it is equally important to consider how the underlying infrastructure needed to support the science mission is set up in the cloud. “Infrastructure-as-code” (IaC) refers to the management of compute, storage, and network resources by software, abstracting away the physical details of the hardware. Declarative templates, generated with Terraform [62], Pulumi [84], or similar tools, allow for deploying infrastructure across different cloud providers in a reproducible way. Publicly shared templates for a variety of IaC, such as JupyterHub for educational courses [126], scalable compute clusters with lightweight Kubernetes (K3s), and full Kubernetes for workbenches [127] have the potential to lower the barriers to adopting multi-cloud deployable services. CACAO [128] is an event-driven multi-cloud service from Cyverse [30] that enables researchers and educators to effortlessly manage, scale and share their tools and workflows to any research-capable cloud using declarative templates that allow you to describe your infrastructure that you want to spin-up in the cloud. It retrieves templates directly from Git and automates any changes to your infrastructure deployment reloads changes depending on template changes. CACAO is currently deployed for Indiana University's (IU) Jetstream2 [39].

IaC is key to understanding some of the most important and fundamental concepts of cloud infrastructure – scalability, elasticity, and disposability. All infrastructure should be designed to fail *and recover from that failure automatically.* The traditional practices of building oversized servers, robust enough to withstand unexpected demands, give way to composing a set of instructions that will spin up fully-configured server instances when one fails or when extra capacity is needed.

These days, with the adoption of microservices architecture, containers are the preferred building blocks of organizations' infrastructure. One example of IaC is container orchestration, which can be achieved using Kubernetes. A major Kubernetes feature is autoscaling based on resource demand, saving human resources from having to manually add new resources to deployments. An additional benefit of the commercial Kubernetes service platforms is that the providers keep the underlying infrastructure patched and secure. Kubernetes can also provide a little bit of cloud abstraction and allow projects to move between cloud vendors.

IceCube has recently used Google Kubernetes Engine (GKE) [129] to great effect, to gain access to graphics processing unit (GPU) resources that are not widely available on academic computing platforms such as the Open Science Grid (OSG) [9]. Using GKE, IceCube was able to provision clusters of various machine types [130] (e.g. central processing unit (CPU) + GPU and on-demand + Spot). GKE's in-built support of features such as auto-provisioning, auto-scaling, dynamic scheduling, orchestrated maintenance, job API and fault tolerance makes it possible for MF's to spin up such clusters on demand and efficiently run their workloads.

A.4 Processing



The majority of MFs have some kind of data processing needs, from filtering data coming from sensors to more classical large-scale computation (compute intensive workloads operating on large amounts of data). Processing requirements vary between the MFs in many dimensions and priorities. Some prioritize processing times, time sensitivity, and time to completion. Others focus on computation types, which can include stream processing, high-throughput computing (i.e., many small independent jobs), and high-performance computing workloads (i.e., few large jobs requiring tightly coupled servers). The most efficient way of processing data strongly depends on how these data are stored, accessed, and distributed.

Below are some key potential benefits that can be realized by moving your computations to the cloud.

A.4.1 Streaming Data and Workflow Management Systems for the Cloud

MFs perform data capture using different kinds of sensors and detectors. There are multiple models of processing, from many small devices (e.g. sensors) communicating with each other in a very distributed manner and sending data to a data center for further processing to several massive sources of data such as LIGO detectors producing massive amounts of data that need to be ingested and processed continuously. For data streaming in from various sources, Apache Kafka [131] is a popular choice for handling the data in a scalable and real time manner. A data streaming model provides low-latency for real-time data streams compared to batch based processing. Using a streaming model one can easily scale applications with minimal load on the processing infrastructure as data is evenly spread out and ingested over time. This is in contrast to a batch processing based system, where data may be aggregated and submitted in bursts. The data streaming model also allows for real-time stream processing, providing continuous data metrics that give the ability to diagnose data problems in real-time. An MF has an option of deploying it itself or using any of the managed instances provided by cloud vendors such as Amazon Kinesis [132], Confluent Cloud [133], [134] .

Some MFs rely on workflow management systems to orchestrate their computations with several recent efforts to better manage workload with streaming data. One representative example of such a trend is Pachyderm [135]. Pachyderm follows a Git-based model where data is committed to repositories (i.e., folders) and as soon as data is added to a particular repository, a particular workflow is triggered by Pachyderm. In that approach, data is driving the workflow execution [136]. Some other MFs have been looking at Dagster [137], a cloud native solution that is data centric and has support for unit testing and development lifecycle in-built. Both Dagster [138] and Pachyderm [139] support integrations with Kafka that allow for processing of data from multiple Kafka topics. In addition to using workflow systems for processing streaming data from Kafka, Apache Flink [140] is a relatively recent open source system that combines streaming data processing and batch processing in one system, and is rapidly gaining traction.

A.4.2 High-performance and High-throughput Computing

Many MFs have existed for decades and have legacy codes. It is likely that most of their workloads have not been developed with cloud computing in mind. For example, some heavy computational workloads such as physics simulations might use specialized software stacks (e.g., MPI, Slurm, HTCondor). Recently, most major cloud providers have developed high-performance computing (HPC) toolkits to easily and automatically deploy Slurm and MPI on their resources. Several providers (Azure, AWS, GCP) also provide batch submission services which mimic traditional batch processing in HPC [141]–[143]. Additionally, Azure and Oracle are currently the only providers offering dense, RDMA networking [81], [144] between nodes. These nodes are charged at rates that are substantially higher than for CPU instances. All the major commercial clouds provide an option of setting up fast, fully-managed parallel file systems such as Lustre for traditional HPC applications, and also provide options to users to dial up and down the performance of the filesystem, including easily shutting them off that results in the data being migrated back to the underlying object store [145].

Cloud providers also propose other computing solutions such as spot instances. Spot instances utilize idle cloud capacity that is currently not being used for on-demand computing by customers. These are usually made available at a steep discount with the caveat that they can be killed anytime by the provider. A key to using these effectively is to use checkpointing, where a user application state is regularly saved on the filesystem and the application has the ability to recover on a retry from that state. A drawback of checkpointing, especially for HPC applications, is the increased demands on the storage filesystem and, thus,must be carefully evaluated.

High throughput computing (HTC) refers to the scenario where a complex application does not need tightly coupled resources and can easily be broken down into either a set of independent jobs or dependent jobs with a DAG structure parallel invocations with little or no effort. For such types of computing, spot instances are well suited and cost-effective. Some MFs such as IceCube and LHC have large numbers of HTC jobs in their workloads. In a subset of cases, where the individual jobs are short running (not more than 10-15 minutes), the jobs can also be invoked as a Function as a Service (FaaS).

Another illustration of use of clouds for HTC is processing against datasets that are made available as part of the various cloud providers' Open Data-set programs. For example, in 2020 AWS made available the entire recent NCBI/NIH Sequence Read Archive (SRA) on their S3 servers [146] as an Open Dataset which allows for an unprecedented rate of access to the petabytes of raw data. Serratus[146] is a cloud computing infrastructure that has been developed for genomics that utilizes spot instances for processing against this dataset.

A.4.3 Serverless Computing

Serverless computing, of which Function as a Service (FaaS) [147] is a commonly used example, provides resources (mainly compute) which are charged only for the duration of the use and removes the need for infrastructure management. In the pay-per-use model, users do not need to reserve resources or pay when idle. Serverless computing offloads the responsibility of managing (e.g. spinning up the resources) the infrastructure and base software to the service providers. Tasks such as installing the operating system, applying security patches, and scaling up or down resources are done by the service providers, leaving developers to simply focus on writing the business logic in functions. Functions can be packaged and run in containers when invoked manually or through event triggers. AWS Lambda [148], Azure Functions [149] are examples of popular serverless computing provided by commercial cloud vendors.

Serverless computing has simplified, pay-per-call pricing and is well suited for short duration tasks with unknown or unpredictable traffic i.e. the cadence at which incoming events are received can sharply vary. This is useful for tasks which are triggered by user actions, also known as "events". For example, a user signup triggers an event to send a welcome email and adds user information to the user database. These two tasks (sending the email and adding a user) trigger two different functions within a serverless setup.

Serverless computing can also be used for asynchronous tasks, such as transcoding videos uploaded by a user into different formats for different devices, or triggering build, test, or packaging tasks when a developer pushes new code to a repository. More use cases for serverless computing can be found here [150].

MFs can use serverless computing for different tasks in their data lifecycles. For example, MFs can validate data received from sensors deployed in the field. The validated data can then be aggregated with pre-existing data to populate metrics for dashboards, etc.

Serverless computing does come with more restrictions compared to managing one's infrastructure. Only a handful of computing languages and operating systems are supported natively (see AWS Lambda Runtimes [148]). This can lead to issues if software relies on external packages or certain features in an operating system (OS). One can deploy custom runtimes, but will still be restricted to the pre-containerized runtimes from the cloud vendor rather than on one's own. Similarly, certain Instruction Set Architecture (ISA) features required by MF applications are not fully supported, again limiting the number of MF applications that can use serverless compute. These capabilities are constantly being expanded by the cloud providers and you should check their documentation for the latest capabilities.

Users only pay for the duration their codes run; this saves them money by not paying for time when otherwise dedicated resources are idle. Conversely, users should be alert about the duration their tasks run since long running tasks could nullify any cost advantages.

Appendix B: Glossary

A

- **ACCESS:** Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support, the NSF national research cyberinfrastructure
- **Agile:** a popular software development framework for teams
- **AWS:** Amazon Web Services commercial cloud service provider
- **Azure:** Microsoft's commercial cloud service provider

B

- **Biocontainer:** a community-driven project that provides the infrastructure and basic guidelines to create, manage and distribute bioinformatics packages (e.g conda) and containers (e.g docker, singularity)
- **Bioconda:** a channel for the conda package manager specializing in bioinformatics software

---

C

- **CARE:** Collective Benefit, Authority to Control, Responsibility and Ethics Indigenous data principles.
- **Chameleon:** NSF research cloud
- **Cloud:** (disambiguation) on-demand access, via the internet, to computing resources—applications, servers, data storage, development tools, networking capabilities, and more—hosted at a remote data center managed by a cloud services provider (or CSP).
- **Cloud Bank:** NSF funded provider for managing NSF funds to be spent on commercial cloud.
- **Cloud Native:** an application or data designed to reside in the cloud
- **Cloud Native Computing Foundation:** is the open source, vendor-neutral hub of cloud native computing
- **Composable:** allows computing resources to be provisioned with code, see IaC
- **Composable infrastructure:** see Infrastructure as Code
- **Conda:** an installation type of the Anaconda data science platform. Command line application for managing packages and environments
- **Container:** virtualization of an operating system run within an isolated, replicable user space

- **Continuous Integration:** (CI) is testing automation to check that the application is not broken whenever new commits are integrated into the main branch
- **Continuous Delivery:** is an extension of 'continuous integration' to make sure that you can release new changes in a sustainable way
- **Continuous Deployment:** a step further than 'continuous delivery,' every change that passes all stages of your production pipeline is released
- **Continuous Development:** a process for iterative software development and is an umbrella over several other processes including 'continuous integration,' 'continuous testing,' 'continuous delivery,' and 'continuous deployment'
- **Continuous Testing:** a process of testing and automating software development
- **CyVerse:** NSF supported cyberinfrastructure for Life Sciences (BIO directorate)

D

- **DevOps** Software *Dev*elopment and information technology *Op*erations techniques for shortening the time to change software in relation to CI/CD
- **Docker:** https://www.docker.com/ is an open source software platform to create, deploy and manage virtualized application containers on a common operating system (OS), with an ecosystem of allied tools. A program that runs and handles life-cycle of containers and images
- **DockerHub:** an official registry of docker containers, operated by Docker. https://hub.docker.com/
- **DOI:** a digital object identifier. A persistent identifier number, managed by the doi.org https://www.doi.org/
- **Dockerfile:** a text document that contains all the commands you would normally execute manually in order to build a Docker image. Docker can build images automatically by reading the instructions from a Dockerfile

---

E

- **Elastic:** (disambiguation) the ability of a cloud service rapidly scale the usage of resources such as storage
- **ElasticSearch:** is a search engine based on the Lucene library. also see OpenSearch
- **Egress:** the action of moving data out of a commercial cloud
- **Environment:** software that includes operating system, database system, specific tools for analysis

- **Entrypoint:** In a Dockerfile, an ENTRYPOINT is an optional definition for the first part of the command to be run

---

F

- **FAIR:** Findable, Accessible, Interoperable, Reusable data principles
- **FOSS:** (1) Free and Open Source Software https://en.wikipedia.org/wiki/Free_and_open-source_software

---

G

- **Git:** a version control system software
- **GitHub:** a website for hosting git repositories -- owned by Microsoft https://github.com
- **GitLab:** a website for hosting git repositories https://gitlab.com
- **GitOps:** using git framework as a means of deploying infrastructure on cloud using Kubernetes
- **Google Cloud:** Google's commercial cloud
- **GPU:** graphic processing unit

H

- **HPC:** high performance computer, for large synchronous computation
- **HTC:** high throughput computer, for many parallel tasks
- **HTCondor:** is an open-source high-throughput computing software framework for coarse-grained distributed parallelization of computationally intensive tasks
- **Hyperscale computing:** the ability of an architecture to scale appropriately as increased demand is added to the system

---

I

- **IaC:** Infrastructure as Code is the process of managing and provisioning computer data centers through machine-readable definition files
- **IaaS:** Infrastructure as a Service https://en.wikipedia.org/wiki/Infrastructure_as_a_service. online services that provide APIs
- **iCommands:** command line application for iRODS https://docs.irods.org/master/icommands/user/ for accessing iRODS Data Store
- **IDE:** integrated development environment, typically a graphical interface for working with code language or packages
- **Instance:** a single virtual machine
- **Image:** self-contained, read-only 'snapshot' of your applications and packages, with all their dependencies
- **iRODS:** an open source integrated Rule-Oriented Data Management System https://irods.org/

J

- **Java:** programming language, class-based, object-oriented
- **JavaScript:** programming language
- **JSON:** Java Script Object Notation, data interchange format that uses human-readable text
- **Jupyter(Hub,Lab,Notebooks):** an IDE, originally the iPythonNotebook, operates in the browser https://jupyter.org/

K

- **Kerchunk:** is a library that provides a unified way to represent a variety of chunked, compressed data formats
- **Kubernetes:** an open source container orchestration platform created by Google Kubernetes https://kubernetes.io/ is often referred to as "K8s"

L

M

- **Markdown:** a lightweight markup language with plain text formatting syntax
- **Metadata::** data about data, useful for searching and querying
- **Multi-thread:** a process which runs on more than one CPU or GPU core at the same time
- **MF:** major facility

N

- **National Research Platform:** a federated network of computational resources available for the public education and research community.
- **NEON:** National Ecological Observatory Network, NSF funded observatory for ecology
- **NSF:** National Science Foundation.

O

- **Object Storage:** computer data storage that manages data as objects
- **On-demand:** cloud instances that are always available “on-demand”, most expensive.
- **Open Science Grid (OSG):** national, distributed computing partnership for data-intensive research https://opensciencegrid.org/
- **OpenSearch:** a Lucene-based search engine that started as a fork of the Elasticsearch
- **ORCID:** Open Researcher and Contributor ID https://orcid.org/, a persistent digital identifier that distinguishes you from every other researcher

---

P

- **PaaS:** Platform as a Service, run and manage applications in cloud without complexity of developing it yourself

---

Q

- **QUAY.io:** private Docker registry https://quay.io

R

- **Recipe file:** a file with installation scripts used for building software such as containers, e.g. Dockerfile
- **Registry:** a storage and content delivery system, such as that used by Docker
- **RST:** ReStructuredText, a markdown type file
- **ReadTheDocs:** a web service for rendering documentation (that this website uses) https://readthedocs.org and readthedocs.com https://readthedocs.com/

S

- **S3:** Simple Storage Service (Amazon S3) is an object storage service
- **SaaS:** Software as a Service https://en.wikipedia.org/wiki/Software_as_a_service web based platform for using software
- **Schema:** a metadata standard for labeling, tagging or coding for recording & cataloging information or structuring descriptive records
- **Scrum:** daily set of tasks and evaluations as part of a sprint.
- **Serverless:** a method of providing backend services on an as-used basis
- **Singularity:** a container software, used widely on HPC, created by SyLabs
- **SLACK:** Searchable Log of All Conversation and Knowledge, a team communication tool
- **SLURM:** Simple Linux Utility for Resource Management, open source job scheduler
- **Spot instance:** a virtual machine that uses 'spare' compute capacity for less money than an 'on-demand' instance
- **Sprint:** set period of time during which specific work has to be completed and made ready for review
- **Stage:** environment that is as similar to the production environment as can be for final testing

---

T

- **Tensor:** algebraic object that describes a linear mapping from one set of algebraic objects to another
- **Thread:** a CPU process or a series of linked messages in a discussion board
- **TPU:** tensor processing unit
- **Travis:** Travis-CI https://travis-ci.org/, a continuous integration software

---

U

V

- **Vendor:** a commercial cloud provider, hardware, or software seller or retailer
- **VICE:** Visual Interactive Computing Environment https://learning.cyverse.org/projects/vice/en/latest
- **Virtual machine:** is a software computer that, like a physical computer, runs an operating system and applications

---

W

- **Waterfall:** software development broken into linear sequential phases, similar to a Gantt chart
- **W3C:** World Wide Web Consortium
- **Worker node:** A cluster typically has one or more nodes, which are the worker machines that run your containerized applications and other workloads. Each node is managed from the master, which receives updates on each node's self-reported status.

X

- **XSEDE:** the Extreme Science and Engineering Discovery Environment (retired), see ACCESS

- **XML:** Extensible Markup Language, data interchange format that uses human-readable text

---

Y

- **YAML:** YAML Ain't Markup Language, data interchange format that uses human-readable text

---

Z

- **Zarr:** a format for the storage of chunked, compressed, N-dimensional arrays
- **ZenHub:** team collaboration solution built directly into GitHub that uses kanban style boards
- **Zenodo:** general-purpose open-access repository developed under the European OpenAIRE program and operated by CERN
- **Zip:** a compressed file format

References

[1] “nsf21107 Research Infrastructure Guide (RIG) (December 2021) | NSF - National Science Foundation.” Available: https://www.nsf.gov/publications/pub_summ.jsp?ods_key=nsf21107&org=NSF. [Accessed: May 30, 2023]

[2] Ž. Ivezić *et al.*, “LSST: From Science Drivers to Reference Design and Anticipated Data Products,” *\apj*, vol. 873, p. 111, Mar. 2019, doi: 10.3847/1538-4357/ab042c

[3] B. P. Abbott *et al.*, “LIGO: the Laser Interferometer Gravitational-Wave Observatory,” *Rep. Prog. Phys.*, vol. 72, no. 7, p. 076901, Jun. 2009, doi: 10.1088/0034-4885/72/7/076901

[4] “ACCESS CI,” *Access*, Aug. 23, 2023. Available: https://access-ci.org/. [Accessed: Aug. 28, 2023]

[5] “Partnership to Advance Throughput Computing,” *PATh*. Available: https://path-cc.io. [Accessed: Aug. 28, 2023]

[6] “Community Earth System Model.” Available: https://www.cesm.ucar.edu/. [Accessed: Aug. 28, 2023]

[7] “World Leader in AI Computing,” *NVIDIA*. Available: https://www.nvidia.com/en-us/. [Accessed: Dec. 15, 2023]

[8] IceCube Collaboration *et al.*, “The IceCube Neutrino Observatory: Instrumentation and Online Systems,” 2016, doi: 10.48550/ARXIV.1612.05093. Available: https://arxiv.org/abs/1612.05093. [Accessed: Aug. 25, 2023]

[9] staff, “SDSC Conducts 50,000+ GPU Cloudburst Experiment with Wisconsin IceCube Particle Astrophysics Center,” *High-Performance Computing News Analysis | insideHPC*, Nov. 25, 2019. Available: https://insidehpc.com/2019/11/sdsc-conducts-50000-gpu-cloudburst-experiment-with-wisconsin-icecube-particle-astrophysics-center/. [Accessed: May 30, 2023]

[10] “Clemson experiment uses 2.1 million VCPUs on Google Cloud,” *Google Cloud Blog*. Available: https://cloud.google.com/blog/topics/hpc/clemson-experiment-uses-2-1-million-vcpus-on-google-cloud. [Accessed: May 30, 2023]

[11] “Elastic Fabric Adapter - Amazon Elastic Compute Cloud.” Available: https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/efa.html. [Accessed: May 30, 2023]

[12] P. Bendor-Samuel, “Why Is Cloud Migration Reversing From Public To On-Premises Private Clouds?,” *Forbes*. Available: https://www.forbes.com/sites/peterbendorsamuel/2021/08/10/why-is-cloud-migration-reversing-from-public-to-on-premises-private-clouds/. [Accessed: Aug. 29, 2023]

[13] “Why we’re leaving the cloud.” Available: https://world.hey.com/dhh/why-we-re-leaving-the-cloud-654b47e0. [Accessed: May 30, 2023]

[14] “Our cloud spend in 2022,” *37signals Dev*, Jan. 13, 2023. Available: https://dev.37signals.com/our-cloud-spend-in-2022/. [Accessed: May 30, 2023]

[15] DHH [@dhh], “Contrast that with just this one example of insanely powerful iron you can buy from Dell. The first R6525s have 256GB RAM, 3TB NVM, 2x10G net, 2x AMD EPYC 7513. Second, same, but 2x AMD EPYC 7443. So that’s a total of 288 vCPU, 15 TB NVM, 1.3TB RAM for $1,287/month over 3 years! https://t.co/l1qapdRF6T,” *Twitter*, Jan. 12, 2023. Available: https://twitter.com/dhh/status/1613558939760689153. [Accessed: May 30, 2023]

[16] “TCO for Cloud Services: A Framework,” Apr. 24, 2015. Available: https://library.educause.edu/resources/2015/4/tco-for-cloud-services-a-framework. [Accessed: May 30, 2023]

[17] “Our Mission,” *ESnet*. Available: https://www.es.net/about/our-mission/. [Accessed: Dec. 15, 2023]

[18] “What is AWS,” *Amazon Web Services, Inc.* Available: https://aws.amazon.com/what-is-aws/. [Accessed: Jul. 21, 2023]

[19] “Google Cloud overview | Overview,” *Google Cloud*. Available: https://cloud.google.com/docs/overview. [Accessed: Jul. 21, 2023]

[20] “IBM Cloud.” Available: https://www.ibm.com/cloud. [Accessed: Nov. 27, 2023]

[21] “What is Azure—Microsoft Cloud Services | Microsoft Azure.” Available: https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-azure/.

[Accessed: Jul. 21, 2023]
[22] "Welcome to Oracle Cloud Infrastructure." Available: https://docs.oracle.com/en-us/iaas/Content/GSG/Concepts/baremetalintro.htm. [Accessed: Jul. 21, 2023]
[23] "Google Earth." Available: https://earth.google.com/web/@0,0,0a,22251752.77375655d,35y,0h,0t,0r. [Accessed: Jul. 21, 2023]
[24] C. A. Stewart *et al.*, "Assessment of financial returns on investments in cyberinfrastructure facilities: A survey of current methods," in *Proceedings of the Practice and Experience in Advanced Research Computing on Rise of the Machines (learning)*, in PEARC '19. New York, NY, USA: Association for Computing Machinery, Jul. 2019, pp. 1–8. doi: 10.1145/3332186.3332228. Available: https://dl.acm.org/doi/10.1145/3332186.3332228. [Accessed: Jun. 02, 2023]
[25] C. A. Stewart *et al.*, "Assessment of non-financial returns on cyberinfrastructure: A survey of current methods," in *Proceedings of the Humans in the Loop: Enabling and Facilitating Research on Cloud Computing*, in HARC '19. New York, NY, USA: Association for Computing Machinery, Jul. 2019, pp. 1–10. doi: 10.1145/3355738.3355749. Available: https://doi.org/10.1145/3355738.3355749. [Accessed: Jun. 02, 2023]
[26] A. Chalker, C. W. Hillegas, A. Sill, S. Broude Geva, and C. A. Stewart, "Cloud and on-premises data center usage, expenditures, and approaches to return on investment: A survey of academic research computing organizations," in *Practice and Experience in Advanced Research Computing*, Portland OR USA: ACM, Jul. 2020, pp. 26–33. doi: 10.1145/3311790.3396642. Available: https://dl.acm.org/doi/10.1145/3311790.3396642. [Accessed: Jun. 02, 2023]
[27] K. Keahey *et al.*, "Lessons learned from the Chameleon testbed," in *Proceedings of the 2020 USENIX Conference on Usenix Annual Technical Conference*, in USENIX ATC'20. USA: USENIX Association, Jul. 2020, pp. 219–233.
[28] D. Duplyakin *et al.*, "The Design and Operation of CloudLab," in *Proceedings of the USENIX Annual Technical Conference (ATC)*, Jul. 2019, pp. 1–14. Available: https://www.flux.utah.edu/paper/duplyakin-atc19
[29] S. Goff *et al.*, "The iPlant Collaborative: Cyberinfrastructure for Plant Biology," *Front. Plant Sci.*, vol. 2, 2011, Available: https://www.frontiersin.org/articles/10.3389/fpls.2011.00034. [Accessed: Jun. 02, 2023]
[30] N. Merchant *et al.*, "The iPlant Collaborative: Cyberinfrastructure for Enabling Data to Discovery for the Life Sciences," *PLOS Biol.*, vol. 14, no. 1, p. e1002342, Jan. 2016, doi: 10.1371/journal.pbio.1002342
[31] "Jetstream2: Accelerating cloud computing via Jetstream | Practice and Experience in Advanced Research Computing." Available: https://dl.acm.org/doi/10.1145/3437359.3465565. [Accessed: Jun. 02, 2023]
[32] A. Loten and I. Bousquette, "CIOs Still Waiting for Cloud Investments to Pay Off," *Wall Street Journal*, Sep. 29, 2022. Available: https://www.wsj.com/articles/cios-still-waiting-for-cloud-investments-to-pay-off-11664449203. [Accessed: Jun. 02, 2023]
[33] "Wanclouds Releases its 2022 Cloud-Native Trends Outlook." Available: https://wanclouds.net/press_releases/wanclouds-releases-its-2022-cloud-native-trends-outlook.html. [Accessed: Jun. 02, 2023]
[34] M. Norman *et al.*, "CloudBank: Managed Services to Simplify Cloud Access for Computer Science Research and Education," in *Practice and Experience in Advanced Research Computing*, Boston MA USA: ACM, Jul. 2021, pp. 1–4. doi: 10.1145/3437359.3465586. Available: https://dl.acm.org/doi/10.1145/3437359.3465586. [Accessed: Aug. 28, 2023]
[35] J. Towns *et al.*, "XSEDE: Accelerating Scientific Discovery," *Comput. Sci. Eng.*, vol. 16, no. 5, pp. 62–74, Sep. 2014, doi: 10.1109/MCSE.2014.80
[36] "ACCESS Allocations: ACCESS." Available: https://allocations.access-ci.org/. [Accessed: Aug. 28, 2023]
[37] S. L. Knuth *et al.*, "The Multi-Tier Assistance, Training, and Computational Help (MATCH) Project, a Track 2 NSF ACCESS Initiative," *J. Comput. Sci. Educ.*, vol. 13, no. 2, Dec. 2022, doi: 10.22369/issn.2153-4136/13/2/4. Available: https://par.nsf.gov/biblio/10404263-multi-tier-assistance-training-computational-help-match-project-

track-nsf-access-initiative. [Accessed: Aug. 07, 2023]
[38] L. Smarr *et al.*, “The Pacific Research Platform: Making High-Speed Networking a Reality for the Scientist,” in *Proceedings of the Practice and Experience on Advanced Research Computing*, in PEARC ’18. New York, NY, USA: Association for Computing Machinery, Jul. 2018, pp. 1–8. doi: 10.1145/3219104.3219108. Available: https://doi.org/10.1145/3219104.3219108. [Accessed: Oct. 10, 2023]
[39] “CyVerse Cacao | Jetstream2.” Available: https://cacao.jetstream-cloud.org/. [Accessed: Jun. 02, 2023]
[40] “Slurm Workload Manager - Cloud Scheduling Guide.” Available: https://slurm.schedmd.com/elastic_computing.html. [Accessed: Jun. 02, 2023]
[41] “Who we are | National Center for Atmospheric Research.” Available: https://ncar.ucar.edu/who-we-are. [Accessed: Jul. 31, 2023]
[42] “Open Data on AWS,” *Amazon Web Services, Inc.* Available: https://aws.amazon.com/opendata/. [Accessed: Aug. 28, 2023]
[43] J. E. Kay *et al.*, “The Community Earth System Model (CESM) Large Ensemble Project: A Community Resource for Studying Climate Change in the Presence of Internal Climate Variability,” *Bull. Am. Meteorol. Soc.*, vol. 96, no. 8, pp. 1333–1349, Aug. 2015, doi: 10.1175/BAMS-D-13-00255.1
[44] J. de la Beaujardière, A. Banihirwe, C.-F. Shi, K. Paul, and J. Hamman, “NCAR CESM LENS Cloud-Optimized Subset.” UCAR/NCAR - Computational and Information Systems Lab, 2019. doi: 10.26024/WT24-5J82. Available: http://ncar-aws-www.s3-website-us-west-2.amazonaws.com/CESM_LENS_on_AWS.htm. [Accessed: Aug. 01, 2023]
[45] “IceCube Overview,” *IceCube*. Available: https://icecube.wisc.edu/about-us/overview/. [Accessed: Jul. 31, 2023]
[46] “About EarthScope,” *EarthScope Consortium*. Available: https://www.earthscope.org/about/earthscope/. [Accessed: Jul. 31, 2023]
[47] “AWS Snow Family – terabyte migration and edge compute devices – Amazon Web Services,” *Amazon Web Services, Inc.* Available: https://aws.amazon.com/snow/. [Accessed: Aug. 28, 2023]
[48] Abbo, Terri, Dugas, Tom, and Gardener, Lisa, “TCO for Cloud Services: A Framework,” Apr. 24, 2015. Available: https://library.educause.edu/resources/2015/4/tco-for-cloud-services-a-framework. [Accessed: Jul. 31, 2023]
[49] “Storage classes for infrequently accessed objects (S3 Standard-IA).” Available: https://docs.aws.amazon.com/AmazonS3/latest/userguide/storage-class-intro.html#sc-infreq-data-access. [Accessed: Jul. 31, 2023]
[50] “Amazon S3 Glacier Storage Classes | AWS,” *Amazon Web Services, Inc.* Available: https://aws.amazon.com/s3/storage-classes/glacier/. [Accessed: Aug. 29, 2023]
[51] A. J. Peters and L. Janyst, “Exabyte Scale Storage at CERN,” *J. Phys. Conf. Ser.*, vol. 331, no. 5, p. 052015, Dec. 2011, doi: 10.1088/1742-6596/331/5/052015
[52] L. Evans and P. Bryant, “LHC Machine,” *J. Instrum.*, vol. 3, no. 08, p. S08001, Aug. 2008, doi: 10.1088/1748-0221/3/08/S08001
[53] “IPAC,” *IPAC*. Available: https://www.ipac.caltech.edu/. [Accessed: Aug. 28, 2023]
[54] “Data Store | CyVerse.” Available: https://cyverse.org/data-store. [Accessed: Oct. 10, 2023]
[55] “Fast, Affordable Cloud Storage & Secure Data Protection - Wasabi.” Available: https://wasabi.com/. [Accessed: Aug. 29, 2023]
[56] B. Hillger, “Azure egress fee waiver for the academic community | Azure Blog | Microsoft Azure,” *Azure Blog*, May 02, 2016. Available: https://azure.microsoft.com/en-us/blog/azure-egress-fee-waiver-for-the-academic-community/. [Accessed: Jun. 02, 2023]
[57] “Get an egress discount for research and education | Cloud Billing,” *Google Cloud*. Available: https://cloud.google.com/billing/docs/how-to/egress-waiver. [Accessed: Jun. 02, 2023]
[58] “Data egress waiver available for eligible researchers and institutions | AWS Public Sector Blog,” Jun. 10, 2021. Available: https://aws.amazon.com/blogs/publicsector/data-egress-waiver-available-for-eligible-researchers-and-institutions/. [Accessed: Jun. 02, 2023]
[59] “Amazon S3 Intelligent-Tiering Storage Class | AWS.” Available:

https://aws.amazon.com/s3/storage-classes/intelligent-tiering/. [Accessed: Jun. 02, 2023]
[60] "Using Requester Pays buckets for storage transfers and usage - Amazon Simple Storage Service." Available: https://docs.aws.amazon.com/AmazonS3/latest/userguide/RequesterPaysBuckets.html. [Accessed: Jun. 02, 2023]
[61] "Requester Pays | Cloud Storage," *Google Cloud*. Available: https://cloud.google.com/storage/docs/requester-pays. [Accessed: Aug. 29, 2023]
[62] "Terraform by HashiCorp," *Terraform by HashiCorp*. Available: https://www.terraform.io/. [Accessed: Jun. 02, 2023]
[63] "EC2 On-Demand Instance Pricing – Amazon Web Services," *Amazon Web Services, Inc.* Available: https://aws.amazon.com/ec2/pricing/on-demand/. [Accessed: Jul. 21, 2023]
[64] "Amazon EC2 - Secure and resizable compute capacity – Amazon Web Services," *Amazon Web Services, Inc.* Available: https://aws.amazon.com/ec2/pricing/reserved-instances/. [Accessed: Jul. 21, 2023]
[65] "Cloud Cost Savings – Savings Plans – Amazon Web Services," *Amazon Web Services, Inc.* Available: https://aws.amazon.com/savingsplans/. [Accessed: Jul. 21, 2023]
[66] C. Team, "AWS Savings Plans Vs. Reserved Instances: When To Use Each." Available: https://www.cloudzero.com/blog/savings-plans-vs-reserved-instances. [Accessed: Jul. 21, 2023]
[67] "Amazon EC2 Spot Instances Pricing," *Amazon Web Services, Inc.* Available: https://aws.amazon.com/ec2/spot/pricing/. [Accessed: Jul. 21, 2023]
[68] "E&I Cooperative Services," *E&I Cooperative Services*. Available: https://www.eandi.org/. [Accessed: Aug. 29, 2023]
[69] "Rubin Observatory Partners with Google Cloud on Interim Data Facility | Vera C. Rubin Observatory Project." Available: https://project.lsst.org/rubin-observatory-partners-google-cloud-interim-data-facility. [Accessed: Jul. 31, 2023]
[70] "STScI Home," *STScI*. Available: https://www.stsci.edu/home. [Accessed: Oct. 10, 2023]
[71] Berriman, G. Bruce; O'Mullane, William; Mitschang, Arik; and Momcheva, Ivelina, "Cost Management on Commercial Cloud Platforms," in *Astronomical Data Analysis Software and Systems XXX. , Proceedings of a virtual conference held 8-12 November 202*, ASP Conference Series, 2022, p. 611.
[72] Lifka,D ; Foster, Ian; Mehringer, Susan; Parashar, Manish; Redfern, Paul; Stewart, Craig; Tuecke, Steve, "XSEDE Cloud Survey Report," 2013. Available: https://hdl.handle.net/2022/25307
[73] P. C. Broekema, V. Allan, R. V. van Nieuwpoort, and H. E. Bal, "On optimising cost and value in compute systems for radio astronomy," *Astron. Comput.*, vol. 30, p. 100337, Jan. 2020, doi: 10.1016/j.ascom.2019.100337
[74] R. Makhlouf, "Cloudy transaction costs: a dive into cloud computing economics," *J. Cloud Comput.*, vol. 9, no. 1, p. 1, Jan. 2020, doi: 10.1186/s13677-019-0149-4
[75] S. Broude Geva, A. Chalker, C. Hillegas, D. Petravick, A. Sill, and C. Stewart, "Results from a second longitudinal survey of academic research computing and data center usage: expenditures, utilization patterns, and approaches to return on investment," in *Practice and Experience in Advanced Research Computing*, in PEARC '21. New York, NY, USA: Association for Computing Machinery, Jul. 2021, pp. 1–4. doi: 10.1145/3437359.3465589. Available: https://dl.acm.org/doi/10.1145/3437359.3465589. [Accessed: Jul. 31, 2023]
[76] "Return on Investment," *Coalition for Academic Scientific Computation*, Oct. 25, 2022. Available: https://casc.org/researchpub/return-on-investment/. [Accessed: Jun. 02, 2023]
[77] C. A. Stewart et al., "Return on Investment for Three Cyberinfrastructure Facilities: A Local Campus Supercomputer, the NSF-Funded Jetstream Cloud System, and XSEDE (the eXtreme Science and Engineering Discovery Environment)," in *EEE/ACM 11th International Conference on Utility and Cloud Computing (UCC)*, Zurich: ACM, 2018, pp. 223–236.
[78] Holzman, B., Bauerdick, L.A.T., Bockelman, B. et al., "HEPCloud, a New Paradigm for HEP Facilities: CMS Amazon Web Services Investigation," *Comput Softw Big Sc*, vol. 1, p. 1, 2017, doi: https://doi.org/10.1007/s41781-017-0001-9
[79] R. Dodson, K. Vinsen, C. Wu, A. Popping, M. Meyer, A. Wicenec, P. Quinn, J. van Gorkom, E. Momjian, "Imaging SKA-scale data in three different computing environments," 2016. Astronomy and Computing," *Astron. Comput.*, vol. 14, pp. 8–22, 2016, doi:

https://doi.org/10.1016/j.ascom.2015.10.007
[80] Y.-T. Chang *et al.*, “Evaluating the Suitability of Commercial Clouds for NASA’s High Performance Computing Applications: A Trade Study.” May 01, 2018. Available: https://ntrs.nasa.gov/citations/20190002611. [Accessed: Jul. 31, 2023]
[81] T. Karmarkar, “Availability of Linux RDMA on Microsoft Azure | Azure Blog | Microsoft Azure,” *Azure Blog*, Jul. 09, 2015. Available: http://azure.microsoft.com/es-es/blog/azure-linux-rdma-hpc-available/. [Accessed: Jul. 21, 2023]
[82] Ajay Gill et al, “Prospects for Wideband VLBI Correlation in the Cloud,” *PASP*, vol. 131, p. 12406.
[83] “Ocean Observatories Initiative,” *Ocean Observatories Initiative*. Available: http://oceanobservatories.org/
[84] “Pulumi - Infrastructure as Code in any Programming Language,” *pulumi*. Available: https://www.pulumi.com/. [Accessed: Jun. 02, 2023]
[85] “What is ‘vendor lock-in’ and how can you avoid it? – Joon Solutions,” Jan. 05, 2022. Available: https://joonsolutions.com/what-is-vendor-lock-in/. [Accessed: Jun. 02, 2023]
[86] “What is vendor lock-in? | Vendor lock-in and cloud computing,” *Cloudflare*. Available: https://www.cloudflare.com/learning/cloud/what-is-vendor-lock-in/. [Accessed: Jun. 02, 2023]
[87] J. Opara-Martins, R. Sahandi, and F. Tian, “Critical analysis of vendor lock-in and its impact on cloud computing migration: a business perspective,” *J. Cloud Comput.*, vol. 5, no. 1, p. 4, Apr. 2016, doi: 10.1186/s13677-016-0054-z
[88] K. Mok, “Should we really be worried about vendor lock-in in 2020? - Protocol.” Available: https://www.protocol.com/manuals/new-enterprise/vendor-lockin-cloud-saas. [Accessed: Jul. 31, 2023]
[89] “HP just dropped out of the public cloud – now what? | Network World.” Available: https://www.networkworld.com/article/2996536/hp-just-dropped-out-of-the-public-cloud-now-what.html. [Accessed: Jun. 02, 2023]
[90] Y. Panda, “Freedoms for Open Source Software users in the Cloud,” Aug. 15, 2018. Available: https://words.yuvi.in/post/oss-in-the-cloud/. [Accessed: Jul. 31, 2023]
[91] “Learning Modules | Internet2 CLASS.” Available: https://internet2-class.github.io/learningmodules. [Accessed: Aug. 28, 2023]
[92] Qwiklabs, “Google Cloud Skills Boost,” *Qwiklabs*. Available: https://www.cloudskillsboost.google. [Accessed: Aug. 28, 2023]
[93] bipach, “Exam AZ-900: Microsoft Azure Fundamentals - Certifications.” Available: https://learn.microsoft.com/en-us/certifications/exams/az-900/. [Accessed: Aug. 28, 2023]
[94] bipach, “Exam AI-900: Microsoft Azure AI Fundamentals - Certifications.” Available: https://learn.microsoft.com/en-us/certifications/exams/ai-900/. [Accessed: Aug. 28, 2023]
[95] “Cloud Computing Training & Classes - Training and Certification - AWS,” *Amazon Web Services, Inc.* Available: https://aws.amazon.com/training/. [Accessed: Aug. 28, 2023]
[96] “AWS Skill Builder.” Available: https://explore.skillbuilder.aws/learn. [Accessed: Aug. 28, 2023]
[97] martinekuan, “Cloud design patterns - Azure Architecture Center,” Apr. 13, 2023. Available: https://learn.microsoft.com/en-us/azure/architecture/patterns/. [Accessed: Dec. 18, 2023]
[98] “CLASS Advanced Cohort,” *Internet2*. Available: https://internet2.edu/cloud/cloud-learning-and-skills-sessions/modules/class-advanced-cohort/. [Accessed: Dec. 18, 2023]
[99] “Azure Cloud Skills—Trainings and Certifications | Microsoft Azure.” Available: https://azure.microsoft.com/en-us/resources/training-and-certifications. [Accessed: Dec. 18, 2023]
[100] “60+ DevOps Courses | 400+ Labs | Learn-by-Doing.” Available: https://kodekloud.com/. [Accessed: Jun. 13, 2023]
[101] “Cloud Academy - Skills Readiness by Cloud Academy,” *Cloud Academy*. Available: https://cloudacademy.com. [Accessed: Jul. 31, 2023]
[102] “Cloud Object Storage – Amazon S3 – Amazon Web Services,” *Amazon Web Services, Inc.* Available: https://aws.amazon.com/s3/. [Accessed: Aug. 28, 2023]
[103] “Google Cloud Storage,” *Google Cloud*. Available: https://cloud.google.com/storage. [Accessed: Aug. 28, 2023]
[104] “Azure Blob Storage | Microsoft Azure.” Available: https://azure.microsoft.com/en-us/products/storage/blobs. [Accessed: Aug. 28, 2023]
[105] J. A. Teeri and P. H. Raven, “A National Ecological Observatory Network,” *Science*, vol. 298, no.

5600, pp. 1893–1893, Dec. 2002, doi: 10.1126/science.298.5600.1893
[106] “The Open Storage Network.” Available: https://www.openstoragenetwork.org/. [Accessed: Jun. 02, 2023]
[107] C. Mellor, “AWS S3: You can check out but can never leave,” *Blocks and Files*, Mar. 15, 2023. Available: https://blocksandfiles.com/2023/03/15/aws-s3-update/. [Accessed: May 30, 2023]
[108] “Storage - Overview of Amazon Web Services.” Available: https://docs.aws.amazon.com/whitepapers/latest/aws-overview/storage-services.html. [Accessed: Aug. 01, 2023]
[109] “National Energy Research Scientific Computing Center,” *NERSC*. Available: https://www.nersc.gov/. [Accessed: Aug. 28, 2023]
[110] “Zarr,” *Zarr*. Available: https://zarr.dev/. [Accessed: Jun. 02, 2023]
[111] “kerchunk.” python filesystem spec, Jun. 01, 2023. Available: https://github.com/fsspec/kerchunk. [Accessed: Jun. 02, 2023]
[112] “Key Concepts & Data Format.” Available: https://docs.tiledb.com/main/background/key-concepts-and-data-format. [Accessed: Jun. 02, 2023]
[113] R. Signell, “Cloud-Performant NetCDF4/HDF5 with Zarr, Fsspec and Intake,” *pangeo*, Dec. 18, 2020. Available: https://medium.com/pangeo/cloud-performant-netcdf4-hdf5-with-zarr-fsspec-and-intake-3d3a3e7cb935. [Accessed: Jun. 02, 2023]
[114] https://techcommunity.microsoft.com/t5/user/viewprofilepage/user-id/363616, “Moving HPC applications to object storage: a first exercise,” *TECHCOMMUNITY.MICROSOFT.COM*, Nov. 21, 2022. Available: https://techcommunity.microsoft.com/t5/azure-high-performance-computing/moving-hpc-applications-to-object-storage-a-first-exercise/ba-p/3678395. [Accessed: Jun. 02, 2023]
[115] R. P. Abernathey *et al.*, “Cloud-Native Repositories for Big Scientific Data,” *Comput. Sci. Eng.*, vol. 23, no. 2, pp. 26–35, Mar. 2021, doi: 10.1109/MCSE.2021.3059437
[116] “Cloud Bigtable: HBase-compatible, NoSQL database,” *Google Cloud*. Available: https://cloud.google.com/bigtable. [Accessed: Aug. 29, 2023]
[117] “Fast NoSQL Key-Value Database – Amazon DynamoDB – Amazon Web Services.” Available: https://aws.amazon.com/dynamodb/. [Accessed: Aug. 29, 2023]
[118] M. D. Wilkinson *et al.*, “The FAIR Guiding Principles for scientific data management and stewardship,” *Sci. Data*, vol. 3, no. 1, p. 160018, Mar. 2016, doi: 10.1038/sdata.2016.18
[119] “OpenSearch: What is it and why it’s important - AWS,” *Amazon Web Services, Inc.* Available: https://aws.amazon.com/what-is/opensearch/. [Accessed: Aug. 28, 2023]
[120] Vardeman, Charles, “Tech Notes: Making the Major Facilities Data Lifecycle FAIR,” Jan. 2022, doi: 10.5281/ZENODO.7007677. Available: https://zenodo.org/record/7007677. [Accessed: Jun. 02, 2023]
[121] “Schema.org - Schema.org.” Available: https://schema.org/. [Accessed: Jun. 02, 2023]
[122] “Data Catalog Vocabulary (DCAT) - Version 3,” *Data Catalog Vocabulary (DCAT) - Version 3*. Available: https://www.w3.org/TR/2023/WD-vocab-dcat-3-20230307/
[123] “DCAT-US Schema v1.1 (Project Open Data Metadata Schema).” Available: https://resources.data.gov/resources/dcat-us/
[124] “Data standards | resources.data.gov.” Available: https://resources.data.gov/standards/. [Accessed: Jun. 02, 2023]
[125] “Data Management | NSF NEON | Open Data to Understand our Ecosystems.” Available: https://www.neonscience.org/data-samples/data-management. [Accessed: Jun. 02, 2023]
[126] C. Holdgraf, “2i2c,” *2i2c*. Available: https://2i2c.org/. [Accessed: Jun. 02, 2023]
[127] “Discovery Environment.” Available: https://de.cyverse.org/. [Accessed: Jun. 02, 2023]
[128] “What is CACAO - Jetstream2 Documentation.” Available: https://docs.jetstream-cloud.org/ui/cacao/intro/. [Accessed: Jun. 02, 2023]
[129] “Google Kubernetes Engine (GKE),” *Google Cloud*. Available: https://cloud.google.com/kubernetes-engine. [Accessed: Aug. 29, 2023]
[130] “GKE GPU sharing helps scientists’ quest for neutrinos,” *Google Cloud Blog*. Available: https://cloud.google.com/blog/products/containers-kubernetes/gke-gpu-sharing-helps-scientists-quest-for-neutrinos. [Accessed: Jun. 02, 2023]
[131] J. Kreps, N. Narkhede, J. Rao, and others, “Kafka: A distributed messaging system for log

processing," in *Proceedings of the NetDB*, Athens, Greece, 2011, pp. 1–7.
[132] "Data Stream Processing - Amazon Kinesis - AWS," *Amazon Web Services, Inc.* Available: https://aws.amazon.com/kinesis/. [Accessed: Dec. 19, 2023]
[133] "Confluent Cloud: Fully Managed Kafka as Service," *Confluent*. Available: https://www.confluent.io/confluent-cloud/. [Accessed: Dec. 19, 2023]
[134] "Introducing Confluent Cloud for Apache Flink," *Confluent*. Available: https://www.confluent.io/blog/introducing-flink-on-confluent-cloud/. [Accessed: Dec. 19, 2023]
[135] "Pachyderm Home Page," *Pachyderm*, Jul. 06, 2022. Available: https://www.pachyderm.com/. [Accessed: Jun. 02, 2023]
[136] R. Mitchell *et al.*, "Exploration of Workflow Management Systems Emerging Features from Users Perspectives," in *2019 IEEE International Conference on Big Data (Big Data)*, Dec. 2019, pp. 4537–4544. doi: 10.1109/BigData47090.2019.9005494
[137] "Welcome to Dagster! | Dagster Docs." Available: https://docs.dagster.io. [Accessed: Aug. 01, 2023]
[138] "Dagster Kafka Integration Guide." Available: https://www.restack.io/docs/dagster-knowledge-dagster-kafka-integration. [Accessed: Dec. 19, 2023]
[139] "Spout PPS." Available: https://docs.pachyderm.com/latest/build-dags/pipeline-spec/spout/. [Accessed: Dec. 19, 2023]
[140] P. Carbone, A. Katsifodimos, S. Ewen, V. Markl, S. Haridi, and K. Tzoumas, "Apache Flink™: Stream and Batch Processing in a Single Engine," *IEEE Data Eng Bull*, vol. 38, no. 4, pp. 28–38, 2015.
[141] "Get started with Batch," *Google Cloud*. Available: https://cloud.google.com/batch/docs/get-started. [Accessed: Aug. 01, 2023]
[142] "What Is AWS Batch? - AWS Batch." Available: https://docs.aws.amazon.com/batch/latest/userguide/what-is-batch.html. [Accessed: Aug. 01, 2023]
[143] "Batch - Compute job scheduling service | Microsoft Azure." Available: https://azure.microsoft.com/en-us/products/batch. [Accessed: Aug. 01, 2023]
[144] J. Brar, "First principles: Superclusters with RDMA—Ultra-high performance at massive scale," Dec. 13, 2022. Available: https://blogs.oracle.com/cloud-infrastructure/post/superclusters-rdma-high-performance
[145] https://techcommunity.microsoft.com/t5/user/viewprofilepage/user-id/1463476, "Azure Managed Lustre: not your grandparents' parallel file system," *TECHCOMMUNITY.MICROSOFT.COM*. Available: https://techcommunity.microsoft.com/t5/azure-high-performance-computing/azure-managed-lustre-not-your-grandparents-parallel-file-system/ba-p/3889946. [Accessed: Aug. 04, 2023]
[146] R. C. Edgar *et al.*, "Petabase-scale sequence alignment catalyses viral discovery," *Nature*, vol. 602, no. 7895, Art. no. 7895, Feb. 2022, doi: 10.1038/s41586-021-04332-2
[147] "What is FaaS?" Available: https://www.redhat.com/en/topics/cloud-native-apps/what-is-faas. [Accessed: Jun. 02, 2023]
[148] "What is AWS Lambda? - AWS Lambda." Available: https://docs.aws.amazon.com/lambda/latest/dg/welcome.html. [Accessed: Jun. 02, 2023]
[149] "Azure Functions – Serverless Functions in Computing | Microsoft Azure." Available: https://azure.microsoft.com/en-us/products/functions. [Accessed: Jun. 02, 2023]
[150] Datadog, "Serverless Architecture: What It Is & How It Works | Datadog," *Serverless Architecture: What It Is & How It Works*. Available: https://www.datadoghq.com/knowledge-center/serverless-architecture/. [Accessed: Jun. 02, 2023]