Table of Content

1. Introduction

We adopt ATAM (Architecture Tradeoff Analysis Method) [1], a systematic architectural analysis method, to evaluate the architecture designed for the project “An Infrastructure Leveraging High-End Computing Capability (HECC) [2].” The motivation for this project is to allow scientists to access the supercomputing environment with minimal knowledge of its operational aspects.

Our project aims to provide an infrastructure, designed for integration with VisTrails [3] and potentially other scientific workflow management software, to seamlessly leverage HECC’s computing and storage resources. However, designing an infrastructure that interfaces with a large existing system is challenging and complex in nature because the design involves the consideration of and trade-offs between different system qualities. A systematic evaluation of the proposed architecture is therefore needed.

In this report, we evaluate our architecture design with ATAM. We document the architectural approaches incorporated in the design. We derive a collection of desired system qualities and scenarios from the stakeholder’s perspectives. With the identified qualities and scenarios, we further evaluate the appropriateness of the architecture and identify the tradeoffs involved in the fulfillment of the desired qualities.

The rest of the report is structured as follows. Section 2 describes the background of the HECC project together with a brief introduction to VisTrails. Section 3 presents our architecture design of the infrastructure. Section 4 details the application of ATAM to evaluate the architecture. Finally, section 5 concludes the report by highlighting what we have learned from the evaluation.

2. Background

The HECC Project, hosted by NASA, is a world-class computing and storage environment for conducting large-scale scientific research experiments to support NASA’s missions. Scientists access the environment by logging on to a front-end node and issue jobs to compute nodes.

The NASA Earth Exchange (NEX) [4] facility assists earth scientists in their research by collecting global and high-resolution satellite data and providing high computing systems to share with the geoscience community. Geoscientists use these environmental data sets in conducting their research, such as creating models to predict natural phenomena. Because the information is high-density, comes from all around the world, and encompasses many years, scientists require high computing power to complete their experiments in a reasonable amount of time.

NEX uses the Pleiades supercomputer provided by the HECC project. An overview of the computing node specifications is as follows [5]:

Node Type	Number of Nodes	Processors per Node	Processor Speed	Memory Size per Core
Sandy Bridge	1,728	2 eight-core processors	2.6 GHz	2 GB
Westmere	4,608	2 six-core processors	2.93 GHz or 3.06 GHz	2 GB
Nehalem	1,280	2 quad-core processors	2.93 GHz	3 GB
Harpertown	4,096	2 quad-core processors	3 GHz	1 GB

Pleiades contains the Portable Batch System (PBS) for job submission, monitoring, and management. PBS adopts job queues to manage pending work and acts as a scheduler. It dispatches jobs to be run on one or more compute nodes, based on factors such as mission shares (a certain percentage of CPU's on Pleiades are allocated to each NASA mission directorate), job priority, queue priority, and job size. Pleiades includes 2 PBS servers.

Pleiades has 16 front-end nodes (PFEs) to allow users to login to submit jobs to PBS, edit files, and debug. Front-end nodes known as "bridge" nodes have higher memory, and are best used for pre/post processing and file transfers.

VisTrails is a scientific workflow management software package used in data-related research. Value is gained by creating a solution to allow VisTrails to submit workflows as jobs to be processed in the Pleiades system, since high computing power is necessary in processing the scientists' data-intensive requests.

3. Architecture and Design

Please refer to the document "Architecture Proposal" for a detailed overview of the architecture and design.

4. Architecture Evaluation

4.1. Business and Mission Drivers

The project goal is to design a secure, reliable and easy-to-use interface for earth scientists to HECC. The approach aims on building a high-performing and reliable architecture system layer between scientists and Pleiades, the flag supercomputer of NASA's high-end computing infrastructure. The primary business drivers of the project are:

- Users can dispatch and schedule workflows to execute on computing platform automatically requiring minimal knowledge of the HPC systems.
- Different existing workflow software packages can be used.
- Private workflows and data cannot be accessed by unauthorized people.
- The system will degrade gracefully on system failure.
- Future changes to the system will be inexpensive and will maintain backwards compatibility.

To fulfill the business drivers, we adopt the following strategies:

- Focus on ease-of-use of the system
- Leverage existing scientific workflow software packages
- Incorporate proven security measures
- Use redundant system components to reduce risk of system failure
- Adopt known software design patterns to increase software modularity

The important driving quality attributes for the architecture are listed below:

Quality Attribute	Description
Security	The system is protected from unauthorized access and data confidentiality is ensured.
Reliability	The system is stable and data integrity is guaranteed.
Availability	The system service is available on a 24/7 basis.
Usability	The system is easy to configure and use.
Performance	The system effectively utilizes computing resources
Scalability	The system can accommodate a large number of users and data.

Extensibility	The system's functionalities are easy to extend and enhance.
Interoperability	The system can interface with different scientific workflow management software and commodity hardware.

These quality attributes, their concerns, and concrete scenarios are further explored in the section "Utility Tree."

4.2. Architectural Approaches

This section captures the list of architecture approaches to address the concerns about the quality of the project. The main concerns are: 1) the secure and reliable operation of Pleiades, and 2) the seamless user experience for domain scientists.

The strategy to address the first concern is to apply secure login methods to prevent access from unauthorized users. An architectural design pattern that can help accomplish this is client-server. By separating responsibilities and placing authentication on the server, the system will have more control on the authentication. Moreover, we will adopt concurrency patterns such as active object and lock to ensure the consistency of data shared by multiple compute nodes. If one compute node goes down, the failure will not affect the other ongoing tasks.

To address the user experience concern, our solution should provide an intuitive user interface and 24/7 service for scientists. The Scheduler Server should periodically report the status of a submitted job to the user. The Scheduler Service should find the most available PFE node to handle the requests, which will allow scientists to receive results quicker. The combination of these services will help provide a smooth user experience.

Other quality attributes such as scalability, extensibility, and interoperability, will be raised when the number of users grows. To handle a large number of users and data, the master Scheduler Server should distribute requests to multiple slave Scheduler Servers. The Scheduler Server should be extensible, able to adapt to new software and hardware design. Also, the service should be partitioned into layers, each containing modules independent to each other. Each module should be encapsulated and have a unified interface to interoperate.

To fulfill the requirements mentioned above, the team proposes the architectural approaches below along with the quality attributes they address.

Approach	**Description**	**Quality Attributes**

Redundant system components	Provides replicas of the Scheduler Servers.	Availability
Load-balancer	Fairly distributes requests among multiple slave Scheduler Servers.	Availability, Scalability
Memory and processor analysis	Used to assign the most available nodes to execute code.	Performance
Multitenancy	Used to serve multiple types of users and access methods.	Usability, Reliability
Three-tiered architecture	Used to separate scientific workflow management software from Pleiades by providing a middle layer.	Interoperability
Publisher-subscriber	Used to notify the Scheduler Service about the status of PFE nodes, PBS servers, and compute nodes.	Extensibility
Client-Server style	Authenticates users on the server side.	Security
Context-awareness	Software behaves based on user scenarios.	Usability
Module	Used to group related functions into one entity.	Interoperability, Extensibility
Facade	Provides unified interface.	Interoperability, Extensibility
Scheduler	Distributes requests to PFE nodes	Performance
Active Object	Used to isolate the scheduler service from other modules	Reliability
Lock	Used to protect shared data	Reliability
Strategy	Serves to isolate the scheduling algorithm from other components	Extensibility

4.3. Utility Tree

Quality Attribute	Security
Attribute Concerns	Only authorized users can access the system.

Scenarios	● Users are prompted to provide identification during login, and the system checks and authorizes the connection. ● When an anonymous user tries to access, connection is automatically closed by the system.
Attribute Concerns	Data transfer and connections are encrypted.
Scenarios	● Users can only connect to the system using ssh or https (depends on service type) connection. ● File transfer can only be done by using sftp. ● Unencrypted connections are automatically closed.

Quality Attribute	Reliability
Attribute Concerns	The Scheduler Server does not crash.
Scenarios	● Invalid parameters for a compute job are sent to the Scheduler Server, and the system does not crash. ● Failure to process one compute job does not influence the execution of other ongoing compute jobs or cause the system to crash.
Attribute Concerns	Computation results are correct.
Scenarios	● Job crashes should not return incorrect results. An error should be returned, or the job should be resumed in another computing node. ● When parallelizing a compute job, MapReduce ensures that either (1) there's no shared data or (2) shared data are protected or synchronized.

Quality Attribute	Availability
Attribute Concerns	Available to accept new compute jobs 24/7 (unless internal or external hardware failures cause service interruption)
Scenarios	● When the master Scheduler Server receives a new compute job, its load-balancing module distributes the job to one of the slave Scheduler Servers. From the moment on, the HECC Adapter communicates with the slave directly, and the master releases the allocated resources to serve new clients.
Attribute Concerns	Graceful degradation in case of service interruption

Scenarios	● If the Pleiades front-end nodes are unavailable, the Scheduler Server will hold the compute job and notify the HECC Adapter. The HECC Adapter will then notify the user and allows the user to cancel the compute job or continue to wait for the service to resume.

Quality Attribute	Usability
Attribute Concerns	Easy to config
Scenarios	● Config wizard shows when open VisTrails for the first time. ● Help information can be found when users do not understand items to configure.
Attribute Concerns	Easy to deliver task
Scenarios	● The UI widget for task delivery is visible and obvious. ● The user delivers the task through GUI but not command line client.
Attribute Concerns	Monitor execution
Scenarios	● The current status of execution query results is readable. ● Users are notified when errors occur.

Quality Attribute	Performance
Attribute Concerns	Tasks are executed on best available computing resources.
Scenarios	● An HECC user does not specify the node type, the system automatically assigns the job to the most powerful node.
Attribute Concerns	Invoke MapReduce calculation whenever it is possible
Scenarios	● A user predefines map and reduce functions, and the system recognizes and invokes corresponding work items. ● A user does not provides map and reduce functions, and the system tries to identify and generate required functions and divide data for MapReduce execution.
Attribute Concerns	Reduce data transfer and Internet response time

Scenarios	• The system load-balances the connections between HECC users and the system. • On matching all criteria an user predefined, the system assigns a work item to the node with largest available bandwidth.

Quality Attribute	Scalability
Attribute Concerns	The Scheduler Service can handle a large group of users
Scenarios	• When a large group of user are sending workflows, the workload of each slave scheduler server, PFE nodes should be as balanced as possible.
Attribute Concerns	The Scheduler Service can handle big data and workflows.
	• When big data and workflows are sent in, the Scheduler Service should find resources to handle them.

Quality Attribute	Extensibility
Attribute Concerns	The Scheduler Service can distribute workflows to new PFE nodes.
Scenarios	• When NASA installs a new PFE node or upgrades an existing PFE node, the Scheduler Service will be notified of the changes. The Scheduler Service should be able to distribute workflows to the PFE node.
Attribute Concerns	Users can assign a new type of nodes to resolve workflows.
Scenarios	• When NASA sets up a new type of nodes, the VisTrails plug-in will be notified. The VisTrails plug-in should allow users to assign the new type of nodes to resolve workflows.
Attribute Concerns	Each component in the Schedule Service should be independent from the others.
Scenarios	• When a better scheduling algorithm comes up, the development team should be able to replace the scheduling algorithm without modifying the other modules.

Quality Attribute	Interoperability
Attribute Concerns	Different scientific software can be used by domain scientists.

Scenarios	● Scientific use software other than VisTrails and the scheduler server can still connect to the Pleiades system.
Attribute Concerns	Different HECC processor and hardware types can be used.
Scenarios	● NASA changes the types of compute nodes in the system, and the system continues to operate.
Attribute Concerns	Alternate login methods are introduced.
Scenarios	● NASA introduces new method to login aside from front-end nodes, and scientists can login continue to login through a dedicated login method.

4.4. Architectural Analysis

From the utility tree, we identified the following sensitivity points, tradeoff points, and risks of the proposed architecture:

Quality Attribute	Sensitivity Points
Security	● Using proven security measures to authorize users and connections increases security.
Reliability	● Validating system parameters before job execution increases reliability. ● Isolating job executions to independent compute nodes increases reliability. ● Using redundant system components to mitigate system failure increases reliability.
Availability	● Incorporating a load balancer to distribute workload increases availability.
Usability	● Prompting users for necessary configuration options increases usability. ● Easily accessible help information and job statuses increases usability.
Performance	● Choosing the most powerful node increases performance, if none is chosen to meet user demand.
Scalability	● Choosing PFE nodes based on their workload for the user increases scalability.
Extensibility	● Increasing code modularity increases software extensibility.

Interoperability	● Providing common interfaces for different types of scientific workflow software packages and login methods increases interoperability.

Quality Attribute	Tradeoff Points
Security	● Extra authentication steps to access the service increases security while decreasing usability.
Reliability	● Redundant system components increase reliability but also decrease scalability because more system resources have to be reserved for a compute job.
Performance	● Choosing the node type for the user increases performance and increases usability by allowing the user to let the system decide.
Scalability	● Choosing PFE nodes for the user increases scalability and increases usability because the user allows the system to choose the optimal PFE node to use. ● Automatically choosing PFE nodes increases scalability, but decreases performance by requiring extra effort in maintaining the statuses of PFE nodes.
Interoperability	● Providing a common interface increases interoperability and increases usability.

Quality Attribute	Risks
Reliability	● Redundant system components could all fail at the same time.
Performance	● Choosing the most powerful node type for the user will increase the cost to the user.
Extensibility	● Developers will need to spend more time to design and implement the system.
Interoperability	● The solution can't be customized for specific scientific workflow packages. ● Scientific workflow systems might not have methods to connect to the common interface.

The tradeoff decisions were made based on prioritized user needs. For example, although usability is decreased by including additional authentication steps, the tradeoff point is made because security is a high-priority quality and some usability can be sacrificed. There are some tradeoff points that clearly show the reasons why a particular architectural decision is made, such as in the cases where two quality attributes increase from the decision. For example, by developing a common interface, interoperability increases by allowing different scientific software packages to use the solution and usability increases by allowing more types of users to use the developed solution.

Besides the listed sensitivity points, tradeoffs, and risks, the team still has to be aware of certain issues. One issue involves the stakeholder's concern on the security in the development and deploy phase. The PBS should not allow access without authentication and authorization. Neither the web service nor the virtual machine is allowed to run in the platform. This causes the inconvenience and difficulty in system prototyping, integration, and testing.

5. Conclusion

The ATAM revealed stakeholder concerns, helping identify the architectural approaches in alleviating them. Key qualities of the system are security, reliability, availability, usability, performance, scalability, extensibility, and interoperability; identifying the scenarios in which these qualities apply helped create project awareness among the stakeholders. Overall, the ATAM evaluation was successful, because the results were beneficial in making architectural decisions that target quality attributes, consider quality tradeoffs, and identify risks through scenarios. Both time and cost will be saved with the help of the information gained from this evaluation.

6. References

[1] ATAM: http://www.sei.cmu.edu/architecture/tools/evaluate/atam.cfm
[2] NASA HECC project: http://www.nas.nasa.gov/hecc/
[3] VisTrails: http://www.vistrails.org/index.php/Main_Page
[4] NEX: https://c3.nasa.gov/nex/
[5] Pleiades node details: http://www.nas.nasa.gov/hecc/resources/pleiades.htm