Clouds: An Opportunity for Scientific Applications?

Ewa Deelman

USC Information Sciences Institute

Acknowledgements

- Yang-Suk Ki (former PostDoc, USC)
- Gurmeet Singh (former Ph.D. student, USC)
- Gideon Juve (Ph.D. student, USC)
- Tina Hoffa (Undergrad, Indiana University)
- Miron Livny (University of Wisconsin, Madison)
- Montage scientists: Bruce Berriman, John Good, and others
- Pegasus team: Gaurang Mehta, Karan Vahi, others

Outline

- Background
  - Science Applications
  - Workflow Systems
- The opportunity of the Cloud
  - Virtualization
  - On-demand availability
- Simulation study of an astronomy application on the Cloud
- Conclusions

Scientific Applications

- Complex
  - Involve many computational steps
  - Require many (possibly diverse resources)
  - Often require a custom execution environment

- Composed of individual application components
  - Components written by different individuals
  - Components require and generate large amounts of data
  - Components written in different languages

Ewa Deelman
deelman@isi.edu

Issues Critical to Scientists

- Reproducibility of scientific analyses and processes is at the core of the scientific method
- Scientists consider the "capture and generation of provenance information as a critical part of the <…> generated data"
- "Sharing <methods> is an essential element of education, and acceleration of knowledge dissemination."

NSF Workshop on the Challenges of Scientific Workflows, 2006, www.isi.edu/nsf-workflows06
Y. Gil, E. Deelman et al, Examining the Challenges of Scientific Workflows. IEEE Computer, 12/2007

Computational challenges faced by applications

- Be able to compose complex applications from smaller components
- Execute the computations reliably and efficiently
- Take advantage of any number/types of resources
- Cost is an issue
  - Cluster, Shared CyberInfrastructure (EGEE, Open Science Grid, TeraGrid), Cloud

Possible solution

- Structure an application as a workflow
  - Describe data and components in logical terms
  - Can be mapped onto a number of execution environments
  - Can be optimized and if faults occur the workflow management system can recover
- Use a workflow management system (Pegasus-WMS) to manage the application on a number of resources

Pegasus-Workflow Management System

- Leverages abstraction for workflow description to obtain ease of use, scalability, and portability
- Provides a compiler to map from high-level descriptions to executable workflows
  - Correct mapping
  - Performance enhanced mapping
- Provides a runtime engine to carry out the instructions (Condor DAGMan)
  - Scalable manner
  - Reliable manner
- Can execute on a number of resources: local machine, campus cluster, Grid, Cloud

Mapping Correctly

- Select where to run the computations
  - Apply a scheduling algorithm for computation tasks
  - Transform task nodes into nodes with executable descriptions
    - Execution location
    - Environment variables initializes
    - Appropriate command-line parameters set
- Select which data to access
  - Add stage-in nodes to move data to computations
  - Add stage-out nodes to transfer data out of remote sites to storage
  - Add data transfer nodes between computation nodes that execute on different resources

Additional Mapping Elements

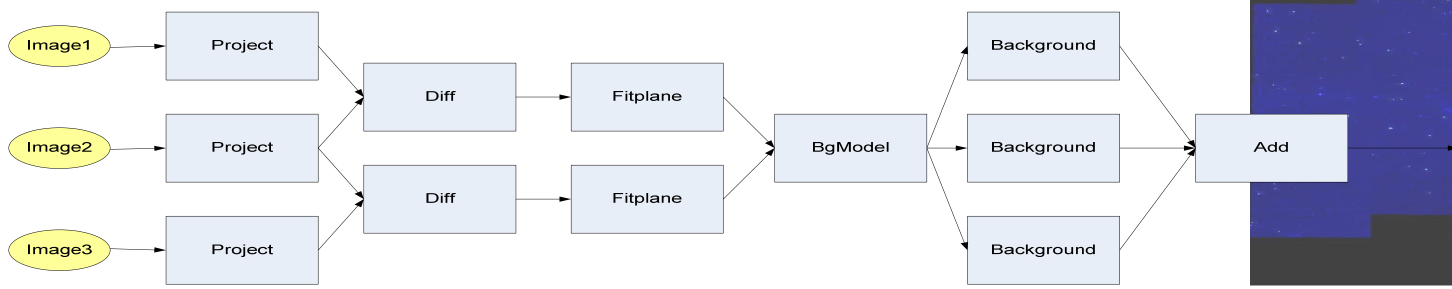
- Add data cleanup nodes to remove data from remote sites when no longer needed
  - reduces workflow data footprint
- Cluster compute nodes in small computational granularity applications
- Add nodes that register the newly-created data products
- Provide provenance capture steps
  - Information about source of data, executables invoked, environment variables, parameters, machines used, performance
- Scale matters--today we can handle:
  - 1 million tasks in the workflow instance (SCEC)
  - 10TB input data (LIGO)

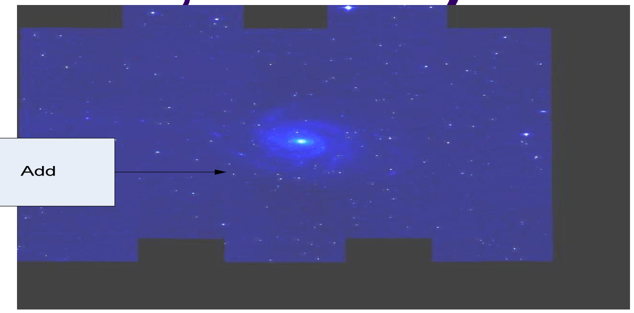
Science-grade Mosaic of the Sky

Point on the sky, area

Image Courtesy of IPAC, Caltech

Generating mosaics of the sky (Bruce Berriman, Caltech)





Size of the mosaic is degrees square*	Number of jobs	Number of input data files	Number of Intermediate files	Total data footprint	Approx. execution time (20 procs)
1	232	53	588	1.2GB	40 mins
2	1,444	212	3,906	5.5GB	49 mins
4	**4,856**	**747**	**13,061**	**20GB**	**1hr 46 mins**
6	8,586	1,444	22,850	38GB	2 hrs. 14 mins
10	20,652	3,722	54,434	97GB	6 hours

**The full moon is 0.5 deg. sq. when viewed form Earth, Full Sky is ~ 400,000 deg. sq.*

Types of Workflow Applications

- **Providing a service to a community (Montage project)**
  - Data and derived data products available to a broad range of users
  - A limited number of small computational requests can be handled locally
  - For large numbers of requests or large requests need to rely on shared cyberinfrastructure resources
  - On-the fly workflow generation, portable workflow definition
- **Supporting community-based analysis (SCEC project)**
  - Codes are collaboratively developed
  - Codes are "strung" together to model complex systems
  - Ability to correctly connect components, scalability
- **Processing large amounts of shared data on shared resources (LIGO project)**
  - Data captured by various instruments and cataloged in community data registries.
  - Amounts of data necessitate reaching out beyond local clusters
  - Automation, scalability and reliability
- **Automating the work of one scientist (Epigenomic project, USC)**
  - Data collected in a lab needs to be analyzed in several steps
  - Automation, efficiency, and flexibility (scripts age and are difficult to change)
  - Need to have a record of how data was produced

Outline

- Background
  - Science Applications
  - Workflow Systems
- The opportunity of the Cloud
  - Virtualization
  - Availability
- Simulation study of an astronomy application on the Cloud
- Conclusions

Clouds

- Originated in the business domain
- Outsourcing services to the Cloud
- Pay for what you use
- Provided by data centers that are built on compute and storage virtualization technologies.
- Scientific applications often have different requirements
  - MPI
  - Shared file system
  - Support for many dependent jobs

Container-based Data Center

Available Cloud Platforms

- Commercial Providers
  - Amazon EC2, Google, others
- Science Clouds
  - Nimbus (U. Chicago), Stratus (U. Florida)
  - Experimental
- Roll out your own using open source cloud management software
  - Virtual Workspaces (Argonne), Eucalyptus (UCSB), OpenNebula (C.U. Madrid)
- Many more to come

Cloud Benefits for Grid Applications

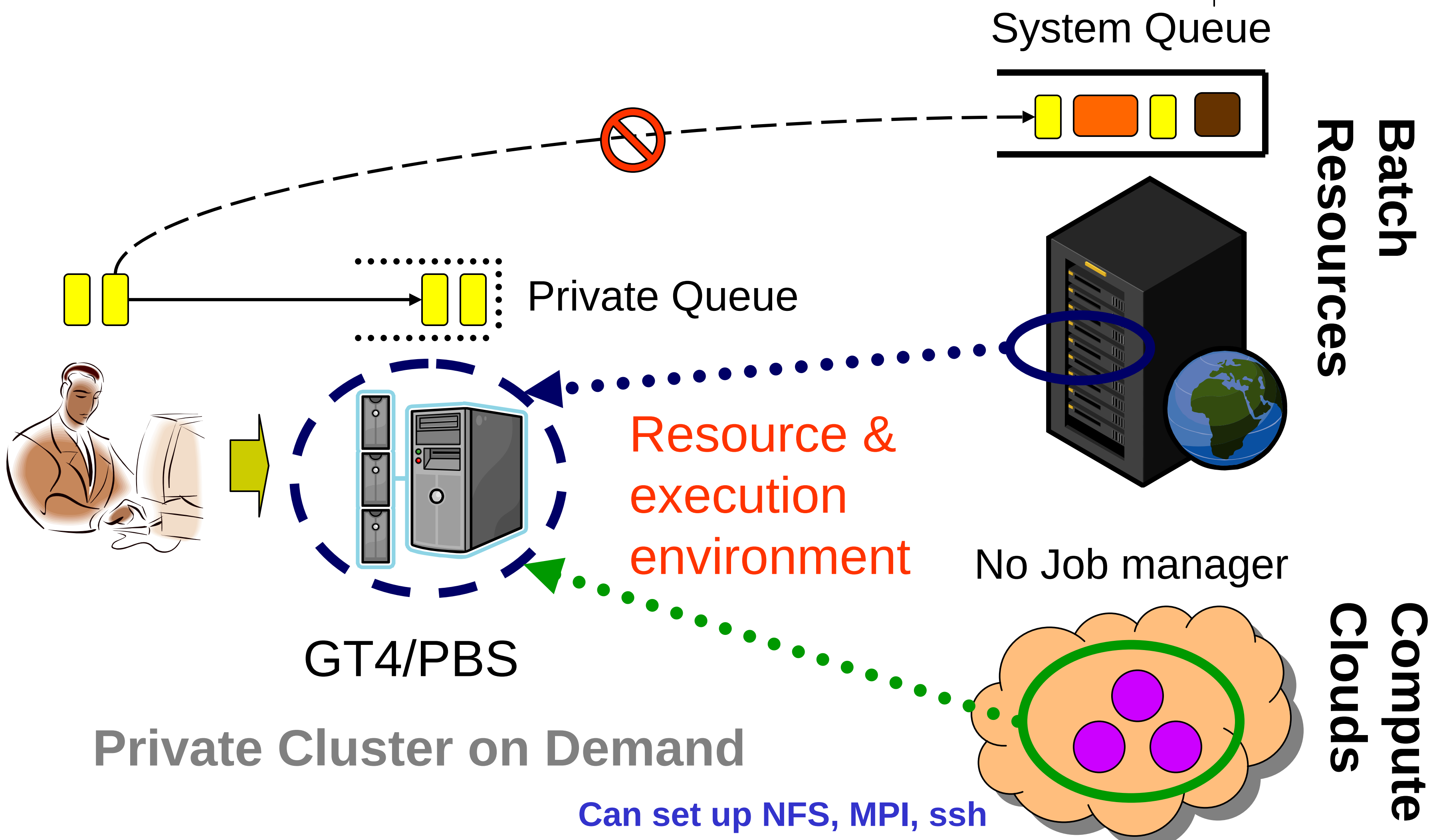
- Similar to the Grid
  - Provides access to shared cyberinfrastructure
  - Can recreate familiar grid and cluster architectures (with additional tools)
  - Can use existing grid software and tools
- Resource Provisioning
  - Resources can be leased for entire application instead of individual jobs
  - Enables more efficient execution of workflows
- Customized Execution Environments
  - User specifies all software components including OS
  - Administration performed by user instead of resource provider (good [user control] and bad [extra work])

Amazon EC2 Virtualization

- Virtual Nodes
  - You can request a certain class of machine
  - Previous research suggests 10% performance hit
  - Multiple virtual hosts on a single physical host
  - You have to communicate over a wide-area network
- Virtual Clusters (additional software needed)
  - Create cluster out of virtual resources
  - Use any resource manager (PBS, SGE, Condor)
  - Dynamic configuration is the key issue

Personal Cluster

Work by Yang-Suk Kee at USC



Private Cluster on Demand

Can set up NFS, MPI, ssh

EC2 Software Environment

- Specified using disk images
  - OS snapshot that can be started on virtualized hosts
  - Provides portable execution environment for applications
  - Helps with reproducibility for scientific applications
- Images for a workflow application can contain:
  - Application Codes
  - Workflow Tools
    - Pegasus, DAGMan
  - Grid Tools
    - Globus Gatekeeper, GridFTP
  - Resource Manager
    - Condor, PBS, SGE, etc.

EC2 Storage Options

- Local Storage
  - Each EC2 node has 100-300 GB of local storage
  - Used for image too
- Amazon S3
  - Simple put/get/delete operations
  - Currently no interface to grid/workflow software
- Amazon EBS
  - Network accessible block-based storage volumes (c.f. SAN)
  - Cannot be mounted on multiple workers
- NFS
  - Dedicated node exports local storage, other nodes mount
- Parallel File Systems (Lustre, PVFS, HDFS)
  - Combine local storage into a single, parallel file system
  - Dynamic configuration may be difficult

Montage/IPAC Situation

- Provides a service to the community
  - Delivers data to the community
  - Delivers a service to the community (mosaics)
- Have their own computing infrastructure
  - Invests ~ $75K for computing (over 3 years)
  - Appropriates ~ $50K in human resources every year
- Expects to need additional resources to deliver services
- Wants fast responses to user requests

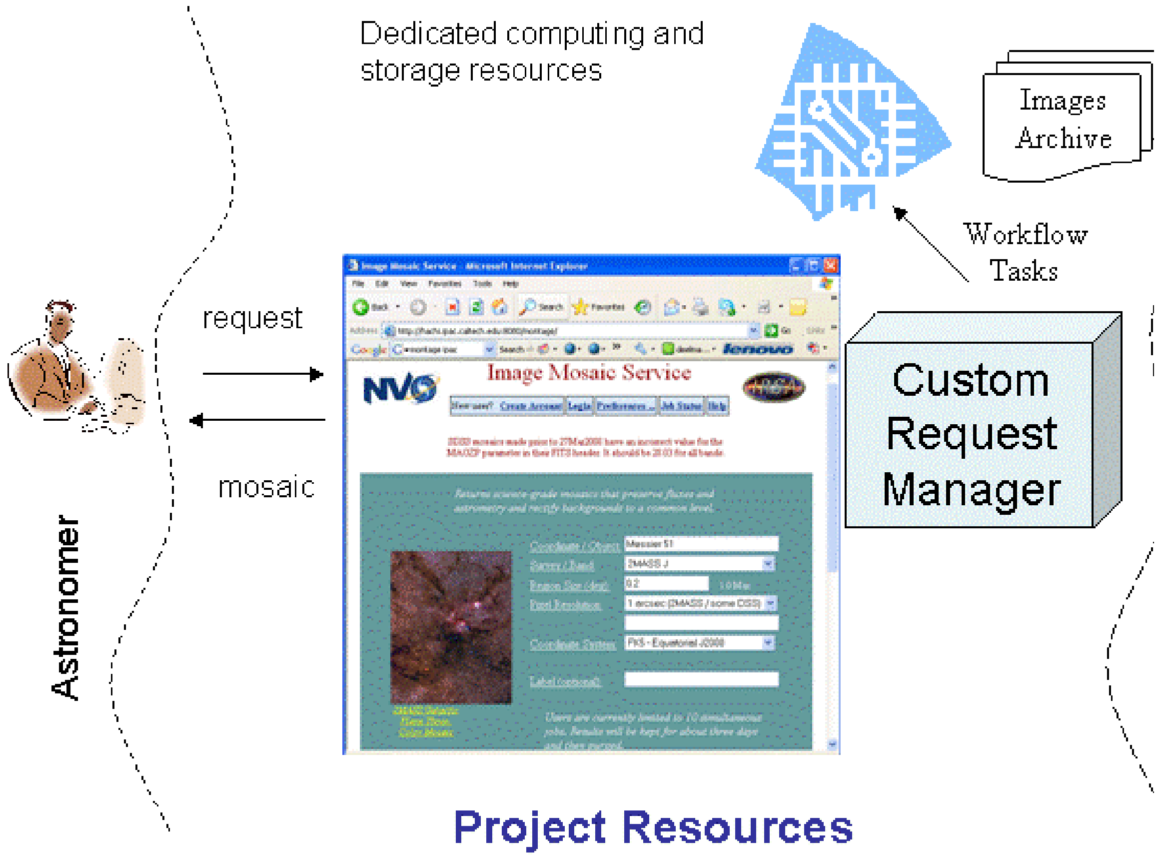
Cloudy Questions

- Applications are asking:
  - What are Clouds?
  - How do I run on them?
- How do I make good use of the cloud so that I use my funds wisely?
  - And how do I explain Cloud computing to the purchasing people?
- How many resources do I allocate for my computation or my service?
- How do I manage data transfer in my cloud applications?
- How do I manage data storage—where do I store the input and output data?

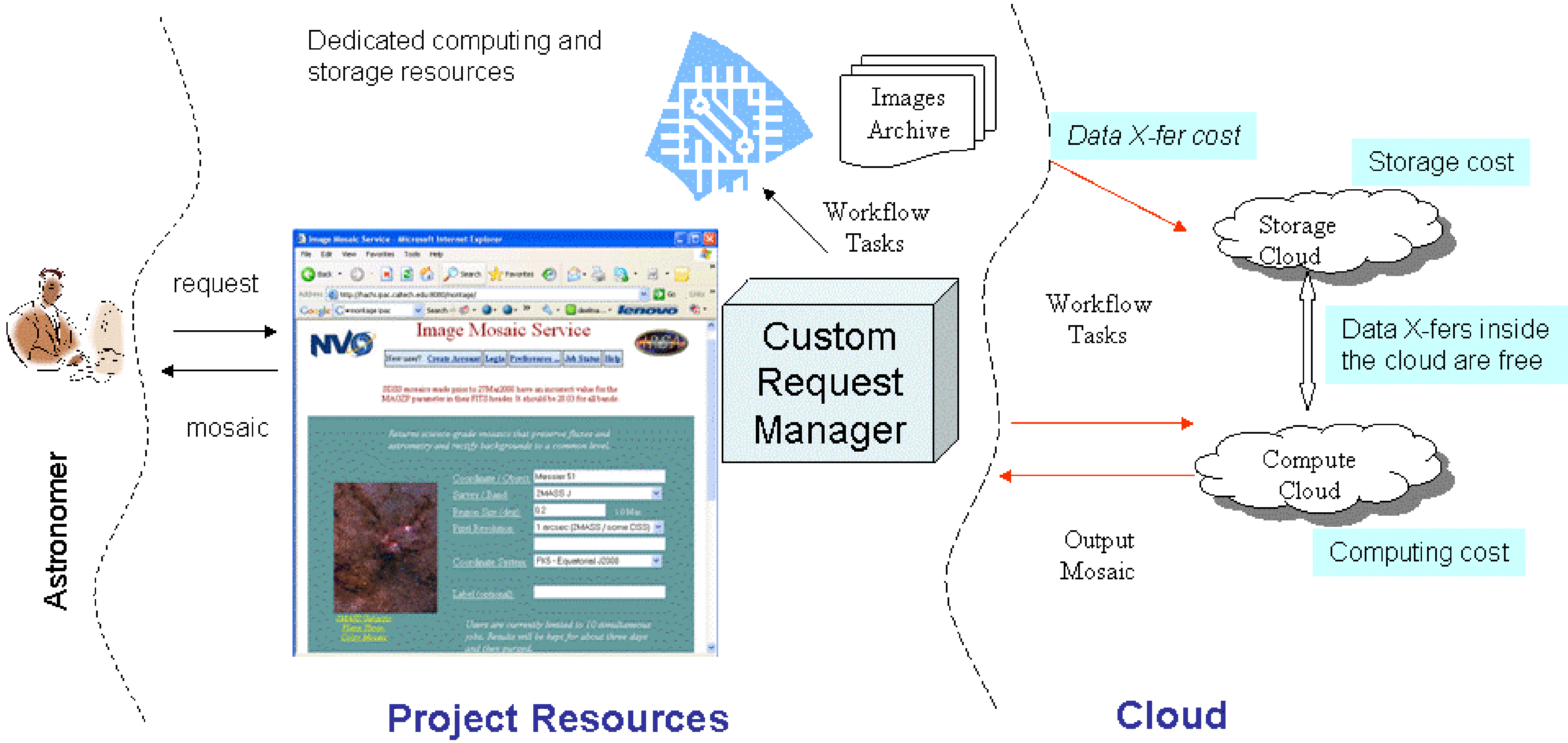
Outline

- Background
  - Science Applications
  - Workflow Systems
- The opportunity of the Cloud
  - Virtualization
  - Availability
- Simulation study of an astronomy application on the Cloud
- Conclusions

Montage Infrastructure



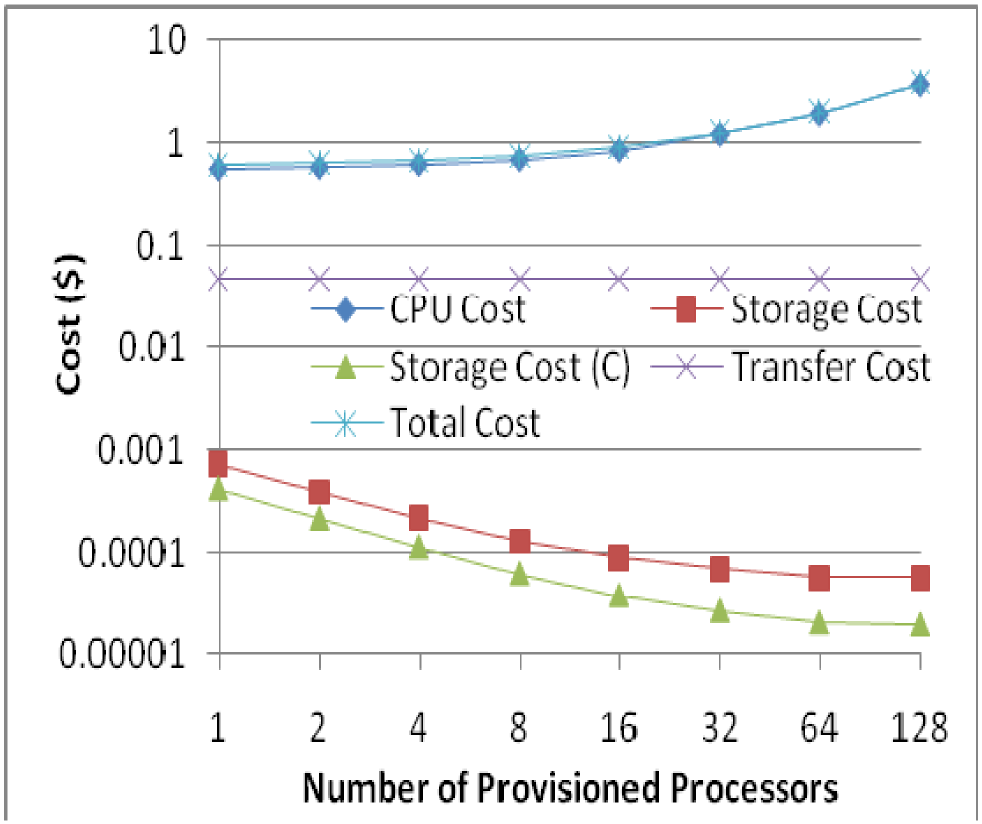
Montage Infrastructure

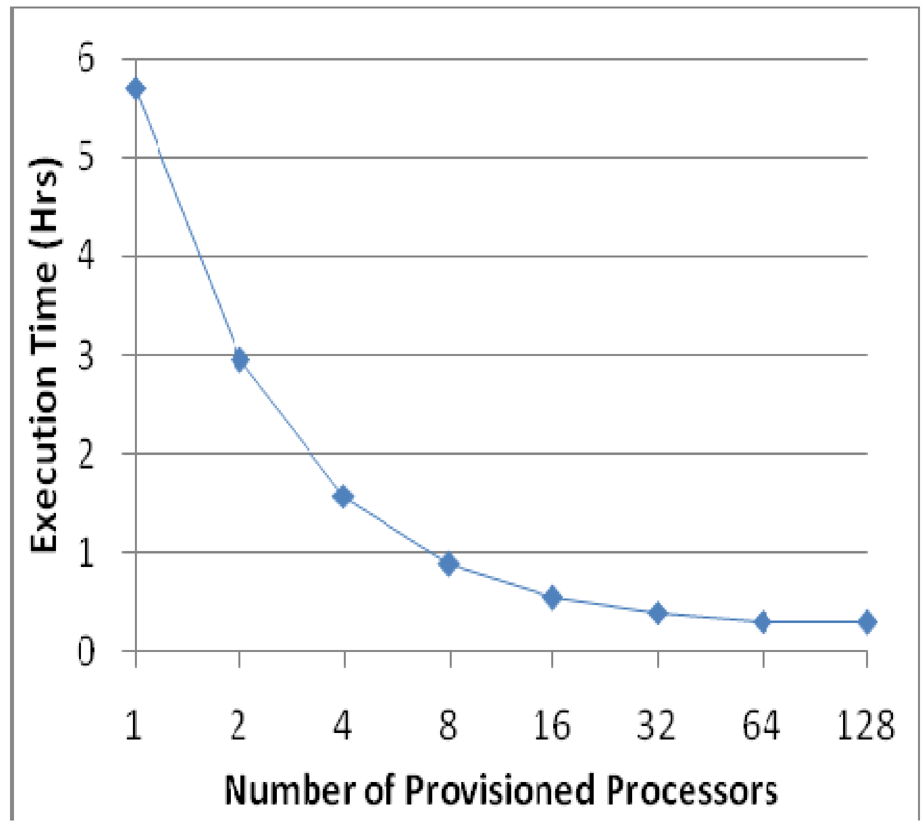


Computational Model

- Based on Amazon’s fee structure
  - $0.15 per GB-Month for storage resources
  - $0.1 per GB for transferring data into its storage system
  - $0.16 per GB for transferring data out of its storage system
  - $0.1 per CPU-hour for the use of its compute resources
- Normalized to cost per second
- Does not include the cost of building and deploying an image
- Simulations done using a modified Gridsim

How many resources to provision?



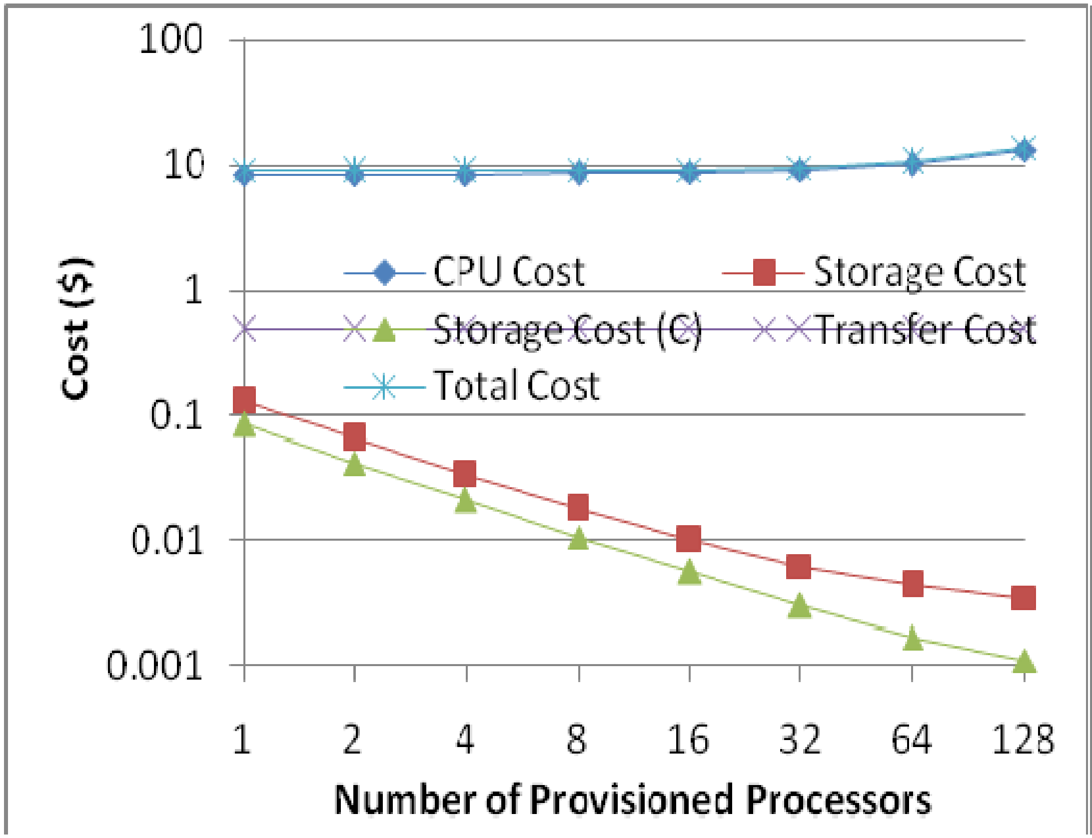


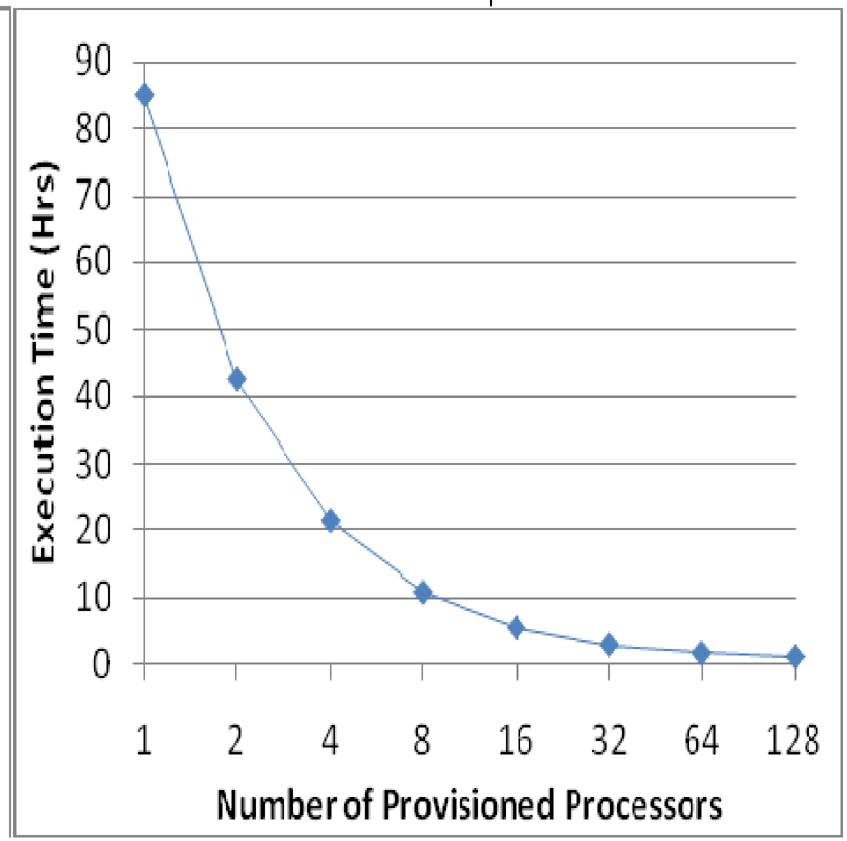
Montage 1 Degree Workflow 203 Tasks

60 cents for the 1 processor computation versus almost $4 with 128 processors, 5.5 hours versus 18 minutes

4 Degree Montage

pegasus



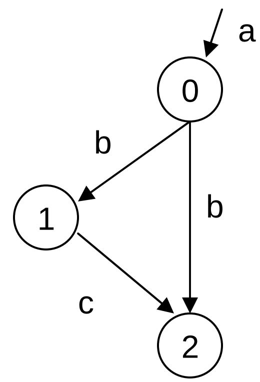


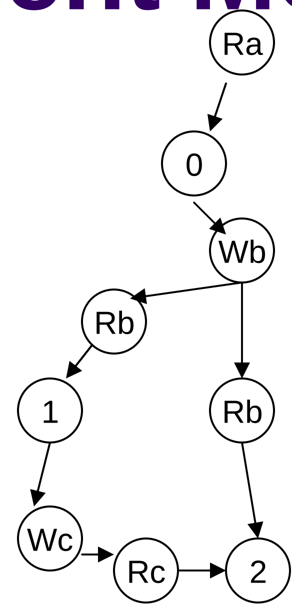
3,027 application tasks

1 processor $9, 85 hours; 128 processors, 1 hour with and $14.

Data Management Modes

- Remote I/O



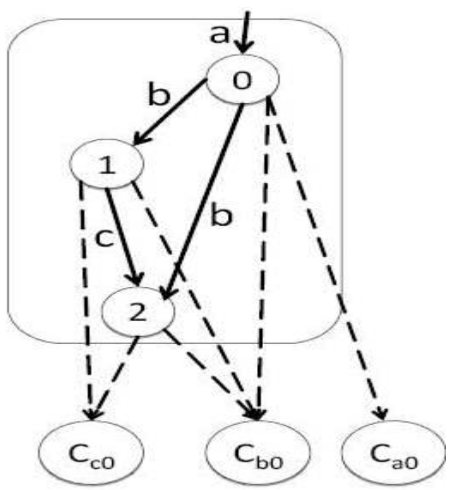


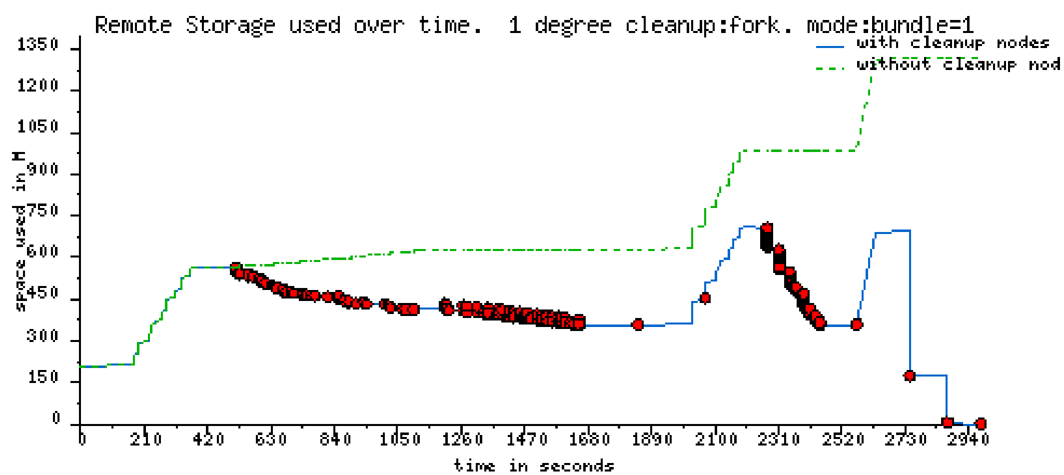
Good for non-shared file systems

- Regular

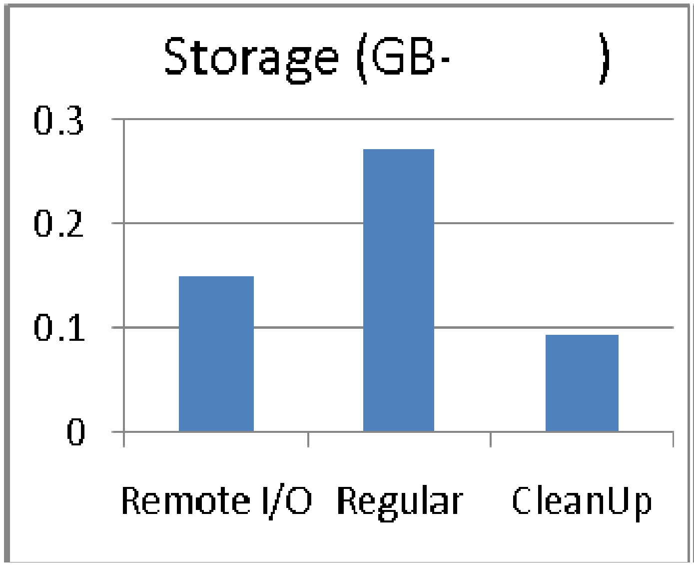
1.25GB versus 4.5 GB

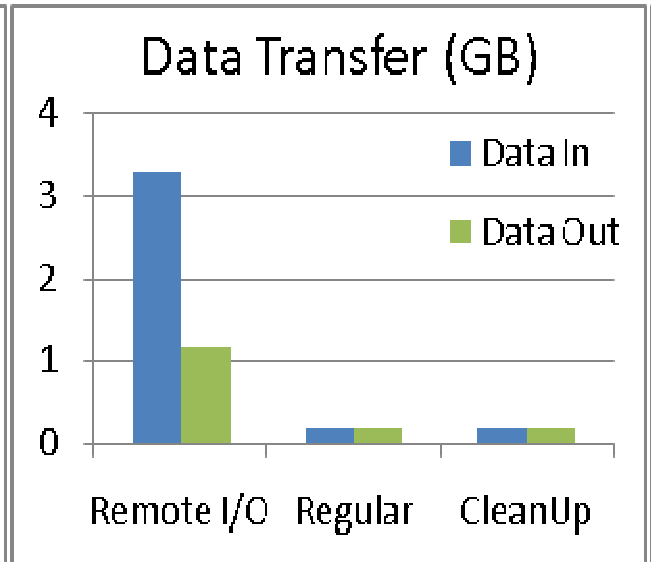
- Cleanup

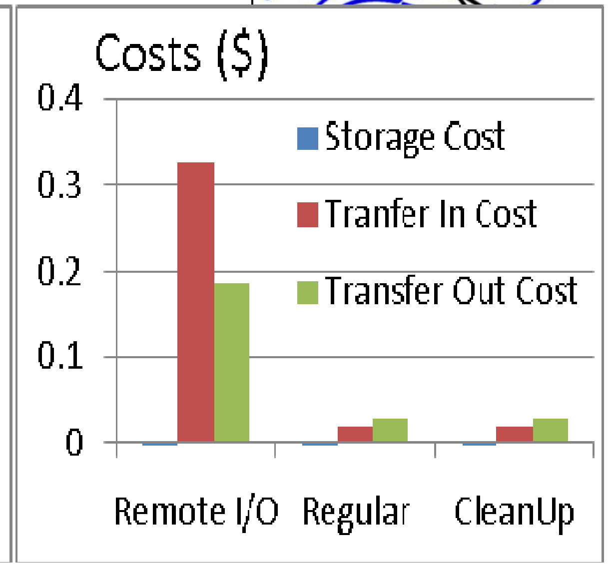




How to manage data?

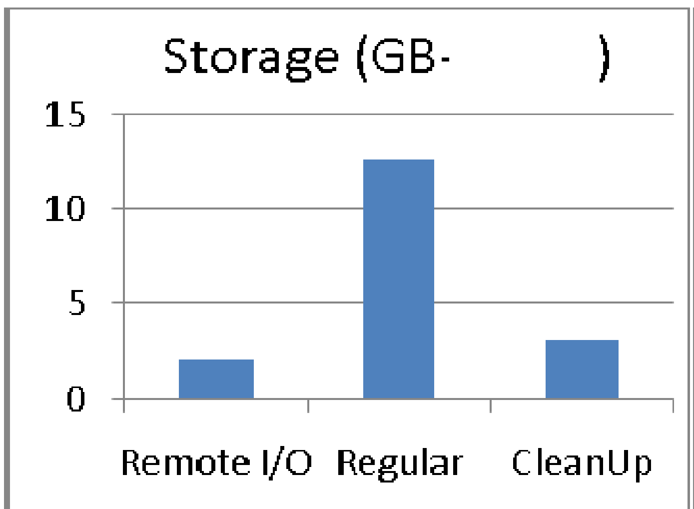


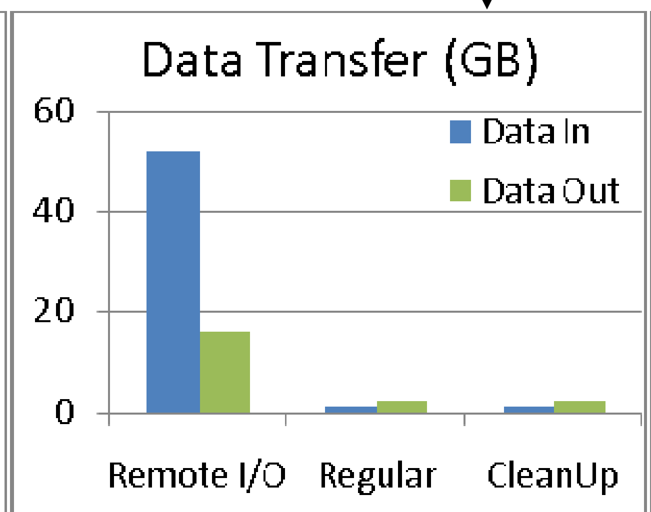


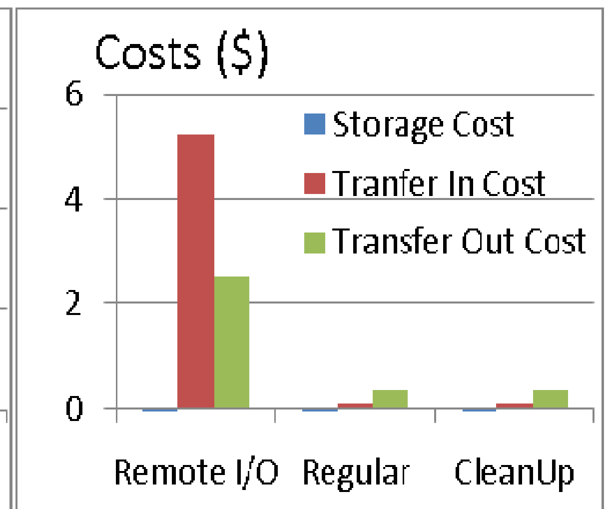


1 Degree Montage ↑

↓ 4 Degree Montage



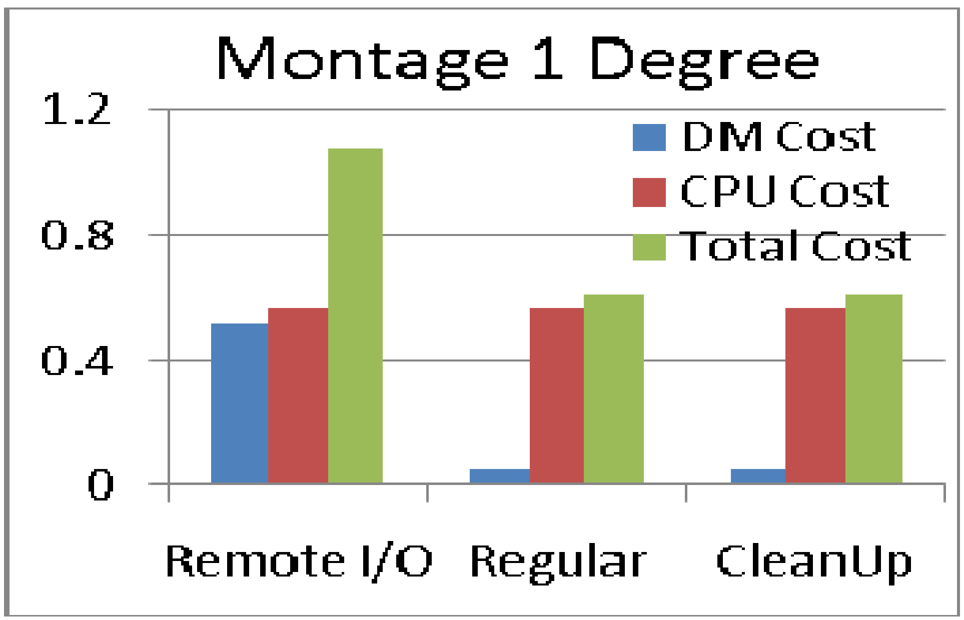


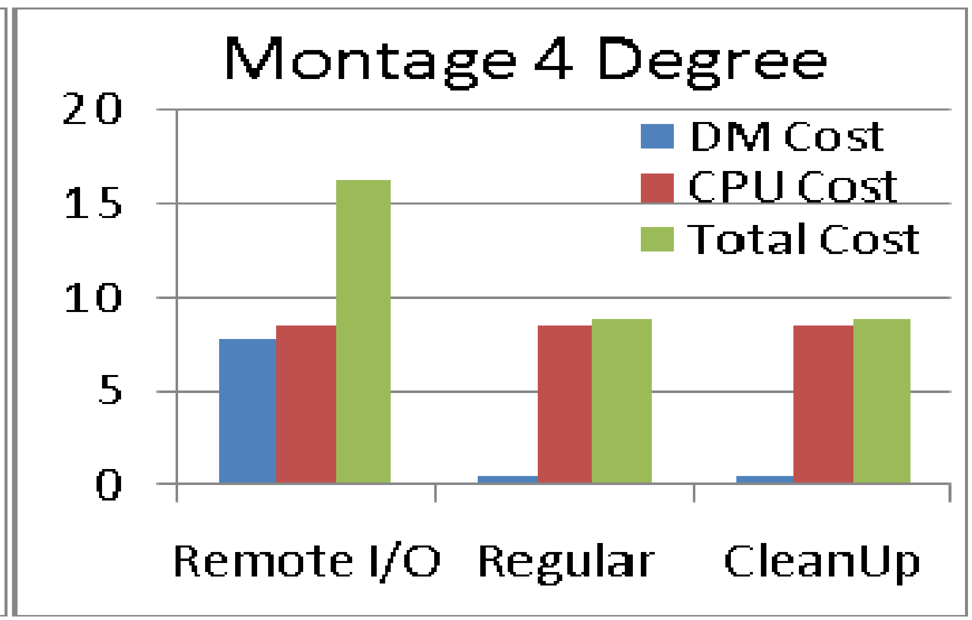


How do data cost affect total cost?

- Data stored outside the cloud
- Computations run at full parallelism
- Paying only for what you use
  - Assume you have enough requests to make use of all provisioned resources

Cost in $





Where to keep the data?

- Storing all of 2 Mass data
  - 12 TB of data → $1,800 per month on the Cloud
- Calculating a 1 degree mosaic and delivering it to the user $2.22 (with data outside the cloud)
- Same mosaic but data inside the cloud: $2.12
- To overcome the storage costs, users would need to request at least $1,800/($2.22-$2.12) = 18,000 mosaics per month
- Does not include the initial cost of transferring the data to the cloud, which would be an additional $1,200
- Is $1,800 per month reasonable?
  - ~$65K over 3 years (does not include data access costs from outside the cloud)
  - Cost of 12TB to be hosted at Caltech $15K over 3 years for hardware

The cost of doing science

- Computing a mosaic of the entire sky (3,900 4-degree-square mosaics)
  - 3,900 x $8.88 = $34,632
- How long it makes sense to store a mosaic?
  - Storage vs computation costs

	Cost of generation	Mosaic size	Length of time to save
1 degree^2	$0.56	173MB	21.52 months
2 degree^2	$2.03	558MB	24.25 months
4 degree^2	$8.40	2.3GB	25.12 months

Summary

- We started asking the question of how can a scientific workflow best make use of clouds
- Assumed a simple cost model based on the Amazon fee structure
- Conducted simulations
  - Need to find balance between cost and performance
  - Computational cost outweighs storage costs
- Storing data on the Cloud is expensive
- Did not explore issues of data security and privacy, reliability, availability, ease of use, etc

Will scientific applications move into clouds?

- There is interest in the technology from applications
- They often don't understand what are the implications
- Need tools to manage the cloud
  - Build and deploy images
  - Request the right number of resources
  - Manage costs for individual computations
  - Manage project costs
- Projects need to perform cost/benefit analysis

Issues Critical to Scientists

- Reproducibility – yes—maybe--through virtual images, if we package the entire environment, the application and the VMs behave
- Provenance – still need tools to capture what happened
- Sharing – can be easier to share entire images and data
  - Data could be part of the image

Relevant Links

- Amazon Cloud: http://aws.amazon.com/ec2/
- Pegasus-WMS: pegasus.isi.edu
- DAGMan: www.cs.wisc.edu/condor/dagman

- Gil, Y., E. Deelman, et al. *Examining the Challenges of Scientific Workflows.* IEEE Computer, 2007.
- Workflows for e-Science, Taylor, I.J.; Deelman, E.; Gannon, D.B.; Shields, M. (Eds.), Dec. 2006

- LIGO: www.ligo.caltech.edu/
- SCEC: www.scec.org
- Montage: montage.ipac.caltech.edu/
- Condor: www.cs.wisc.edu/condor/