Managing Data-Intensive Scientific Workflows in Distributed Resources

Ewa Deelman
USC Information Sciences Institute
Marina Del Rey, CA 90292

PPAM 2007
Gdansk, Poland

http://www.isi.edu/~deelman
deelman@isi.edu

Acknowledgments

- Gurmeet Singh, Karan Vahi, Arun Ramakrishnan, Gaurang Mehta
  - **USC Information Science Institute, Pegasus**
- Henan Zhao, Rizos Sakellariou
  - **University of Manchester, UK**
- Kent Blackburn, Duncan Brown, Stephen Fairhurst, David Meyers
  - **LIGO, Caltech, USA**
- G. Bruce Berriman, John Good, Daniel S. Katz
  - **Montage, Caltech and LSU, USA**

Outline

- Motivation: LIGO gravitational-wave applications and requirements
- Pegasus workflow mapping system
- Reducing the workflow data footprint
- Data-space aware workflow scheduling
- Evaluation of the approach in simulation and on the grid
- Conclusions

LIGO: (Laser Interferometer Gravitational-Wave Observatory)

- Aims to detect gravitational waves predicted by Einstein's theory of relativity.
- Can be used to detect
  - binary pulsars
  - mergers of black holes
  - "starquakes" in neutron stars
- Two installations: in Louisiana (Livingston) and Washington State
  - Other projects: Virgo (Italy), GEO (Germany), Tama (Japan)
- Instruments are designed to measure the effect of gravitational waves on test masses suspended in vacuum.
- Data collected during experiments is a collection of time series (multi-channel)

LIGO: (Laser Interferometer Gravitational-Wave Observatory)

- Aims to de
  theory of re
- Can be use
  - binary p
  - mergers
  - “starqua
- Two install
  Washingto
  - Other pr
- Instrument
  gravitational waves on test masses suspended in vacuum.
- Data collected during experiments is a collection of time series (multi-channel)

LIGO Livingston

LIGO’s computations

- Binary inspiral analysis
- Size of analysis for meaningful results
  - at least 221 GBytes of gravitational-wave data
  - approximately 70,000 computational tasks
- Desired analysis:
  - Data from November 2005--November 2006
    - 10TB of input data
  - Approximately 185,000 computations edges
    - 1 Tb of output data

LIGO’s computational resources

- LIGO Data Grid
  - Condor clusters managed by the collaboration
  - ~ 6,000 CPUs
- Open Science Grid
  - A US cyberinfrastructure shared by many applications
  - ~ 20 Virtual Organizations
  - ~ 258 GB of shared scratch disk space on OSG sites

Problem

- How to automate the execution of thousands of tasks?
  - Use a workflow structure for the application
  - Use Pegasus workflow manager to map high-level workflows onto available resources
  - Use Condor DAGMan for workflow execution
- How to “fit” the computations onto the OSG
  - Take into account intermediate data products
  - Minimize the data footprint of the workflow
  - Schedule the workflow tasks in a disk-space aware fashion

Workflow Building Blocks

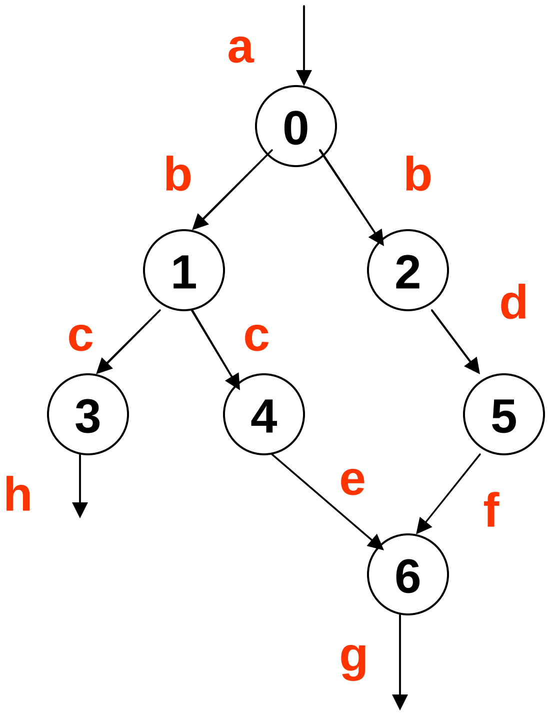
- Standalone computations
- Data transfers
- Result (final and intermediate) registration in catalogs (*optional*)

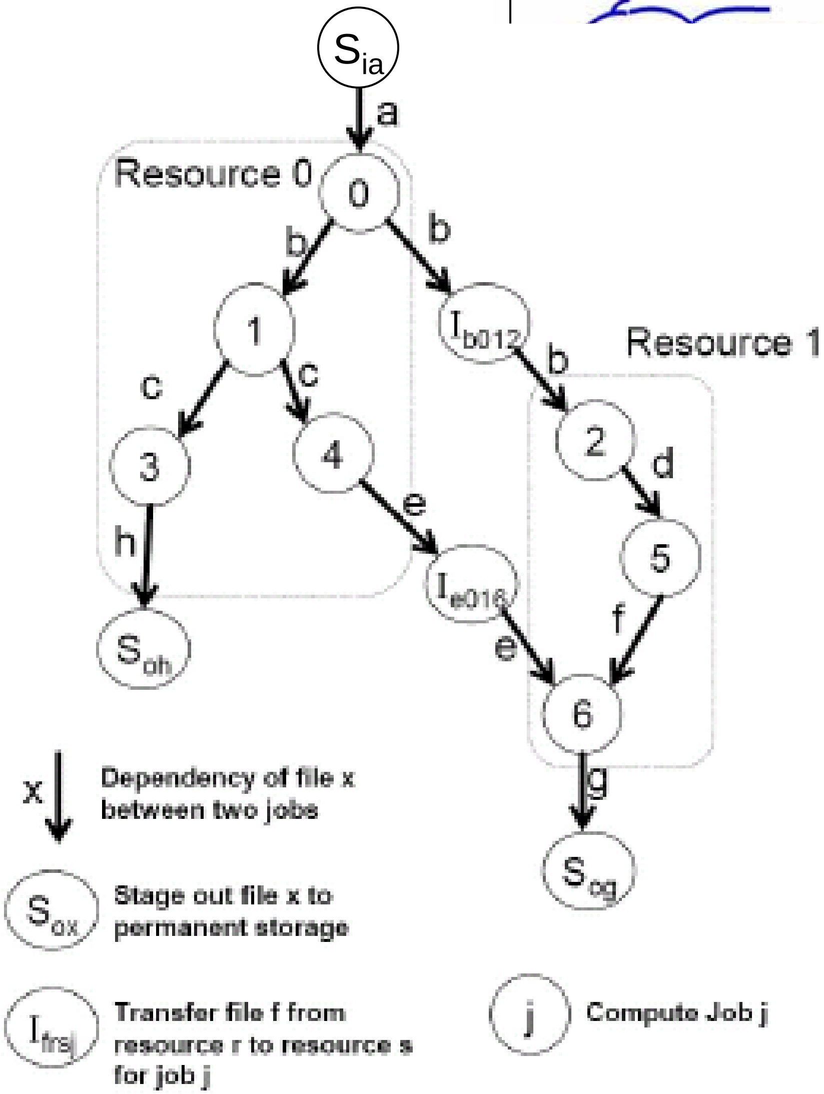
- In distributed environments there are many choices of compute and data resources
- In many cases data movement depends on the scheduling of the computation

Pegasus est. 2001

- Based on the programming language principles
  - Leverage abstraction for workflow description to obtain ease of use, scalability, and portability
  - Provide a “compiler” to map from high-level descriptions to executable workflows
    - Correct mapping
      - Uses information services available on the grid
      - Infers data transfer and registration
    - Performance enhanced mapping
    - Data-space conscious mapping
  - Rely on a runtime engine to carry out the instructions—Condor DAGMan
    - Scalable manner
    - Reliable manner

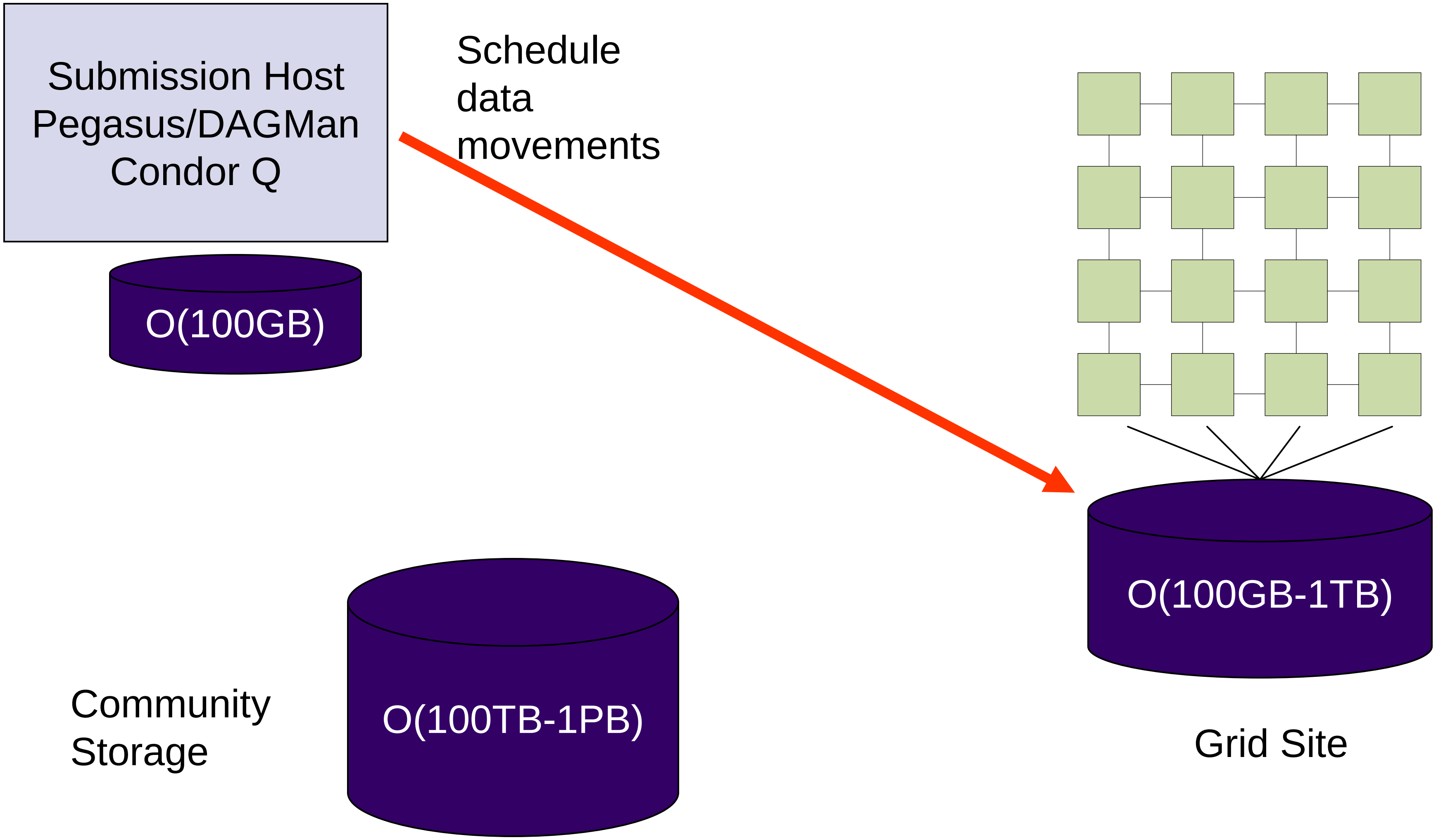
Pegasus mapping





- Select compute resources
- Select data sources
- Add data stage-in and data stage-out nodes
- Originally: data cleaned up once all execution done

Storage on the Grid



Submission Host
Pegasus/DAGMan
Condor Q

O(100GB)

Schedule
data
movements

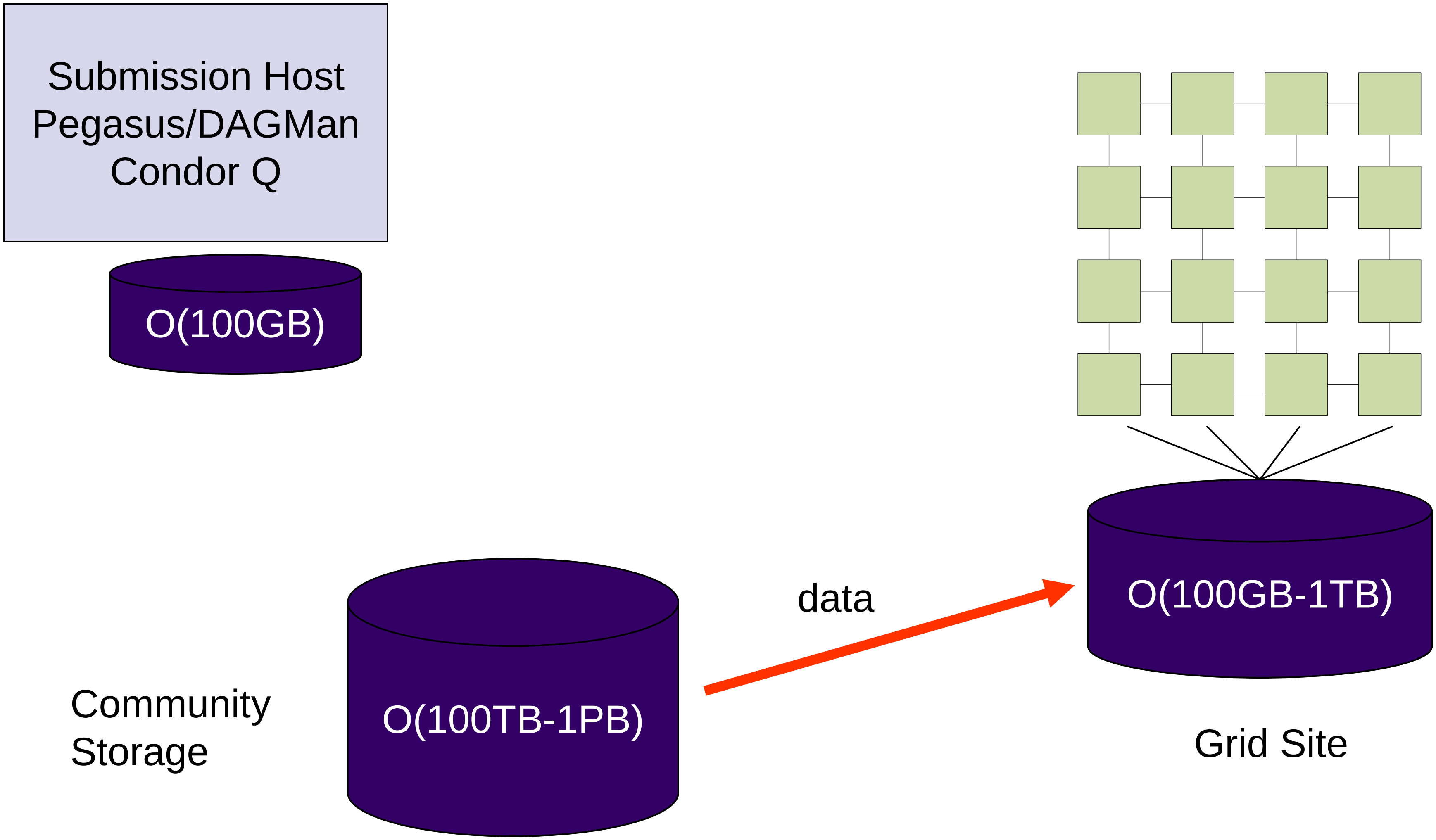
O(100GB-1TB)

Grid Site

Community
Storage

O(100TB-1PB)

Storage on the Grid



Submission Host
Pegasus/DAGMan
Condor Q

O(100GB)

data

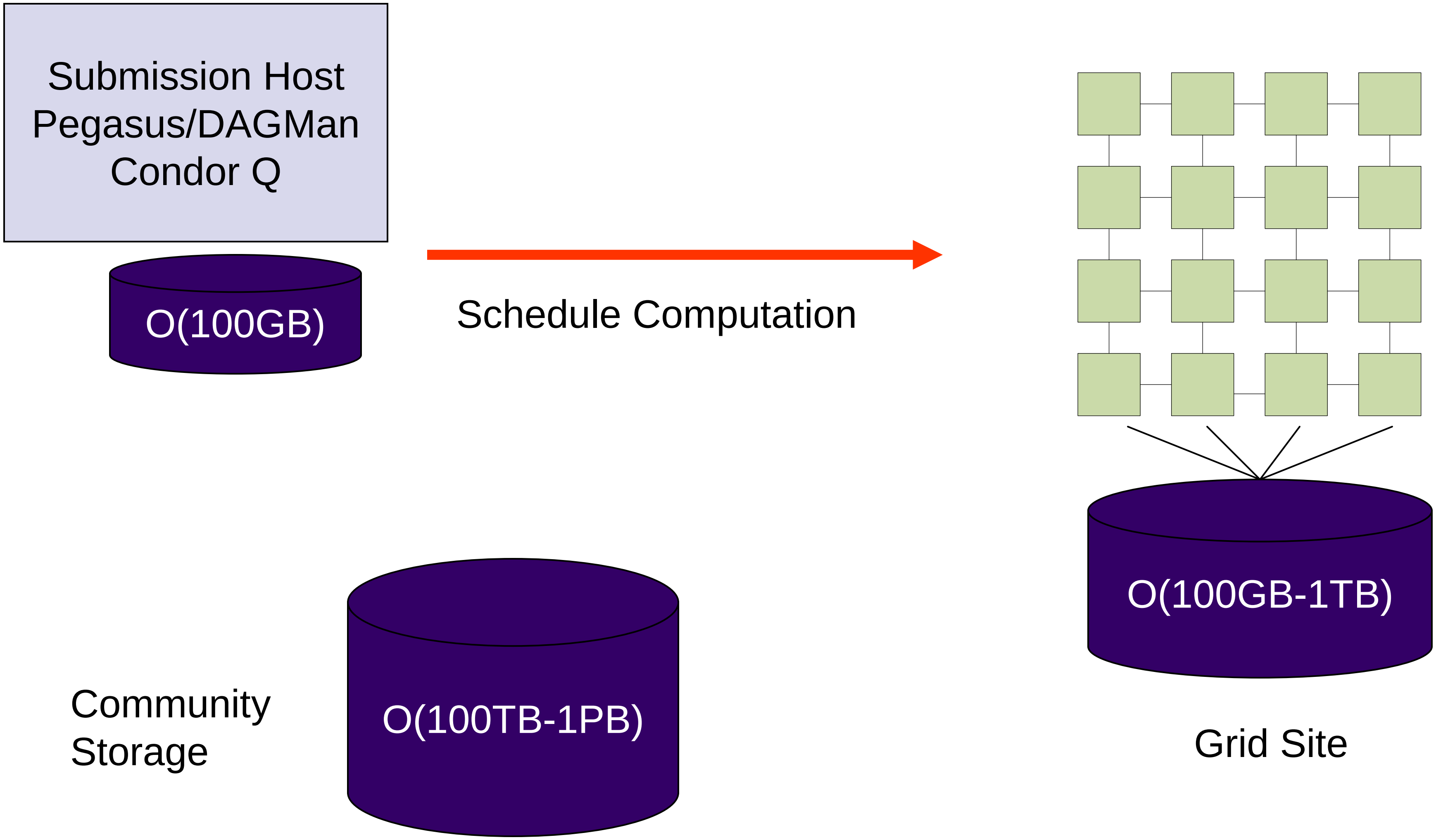
O(100GB-1TB)

Community
Storage

O(100TB-1PB)

Grid Site

Storage on the Grid



Submission Host
Pegasus/DAGMan
Condor Q

O(100GB)

Schedule Computation

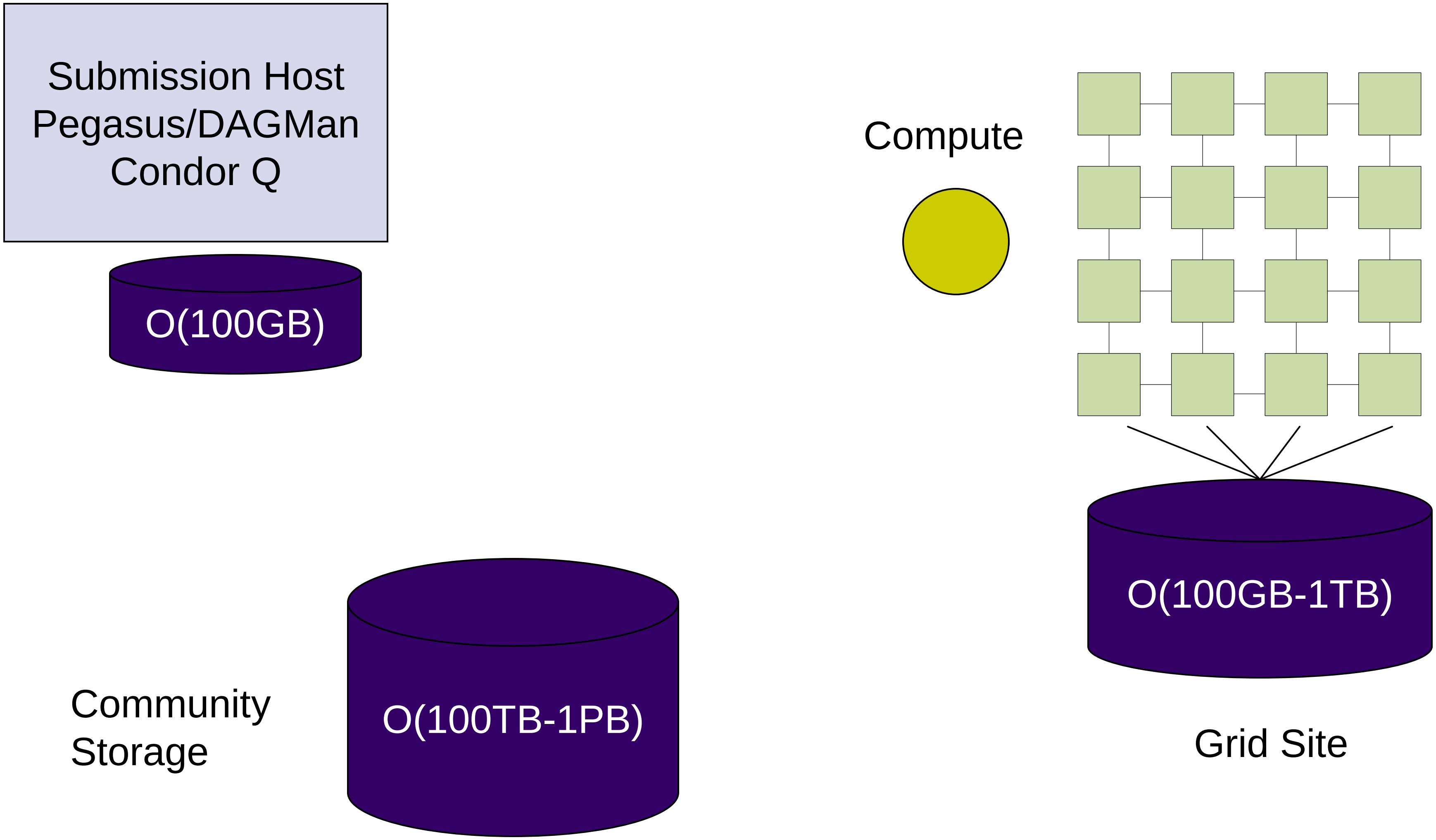
O(100GB-1TB)

Grid Site

Community
Storage

O(100TB-1PB)

Storage on the Grid



Submission Host
Pegasus/DAGMan
Condor Q

O(100GB)

Compute

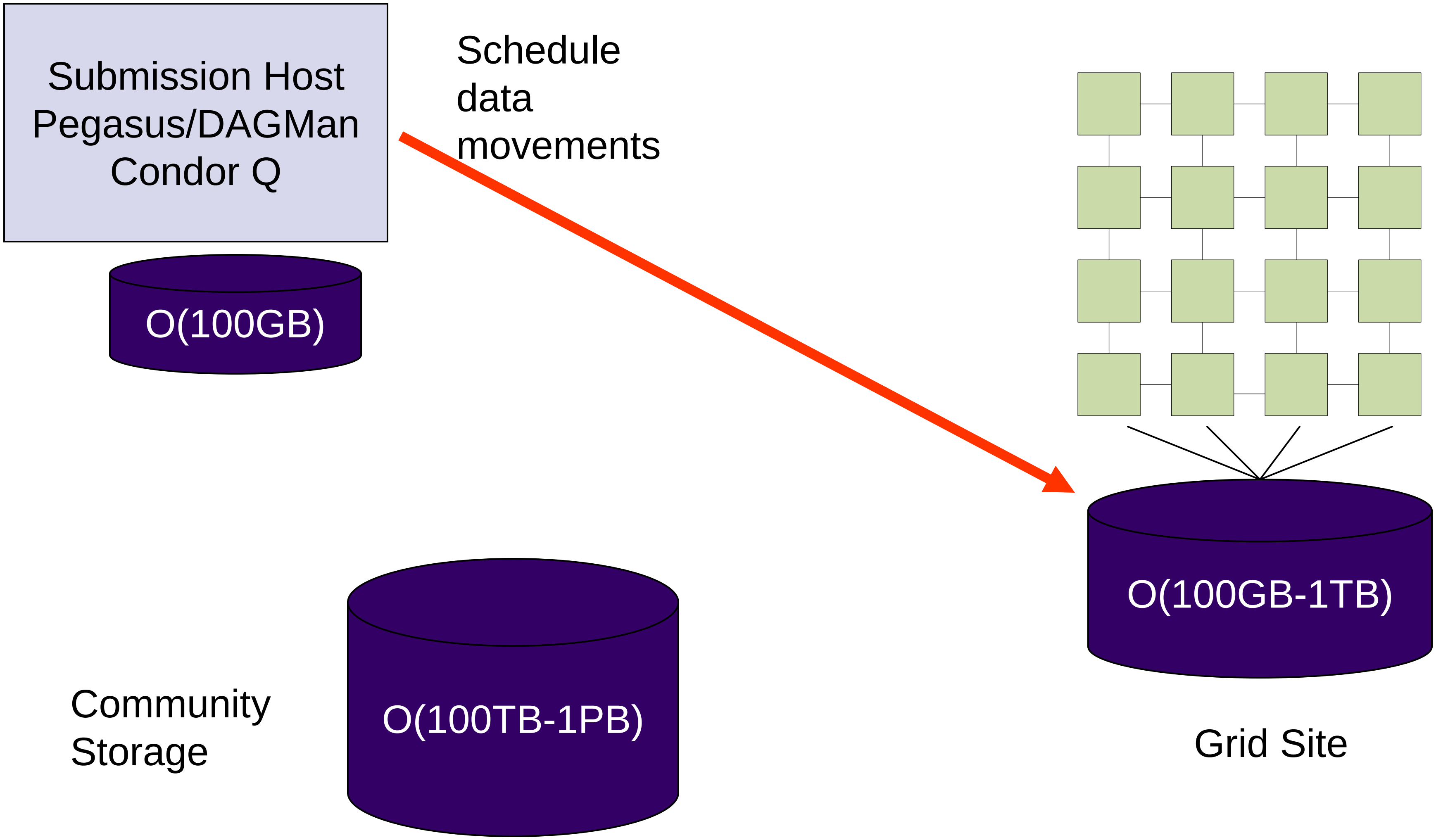
O(100GB-1TB)

Grid Site

Community
Storage

O(100TB-1PB)

Storage on the Grid



Submission Host
Pegasus/DAGMan
Condor Q

O(100GB)

Schedule
data
movements

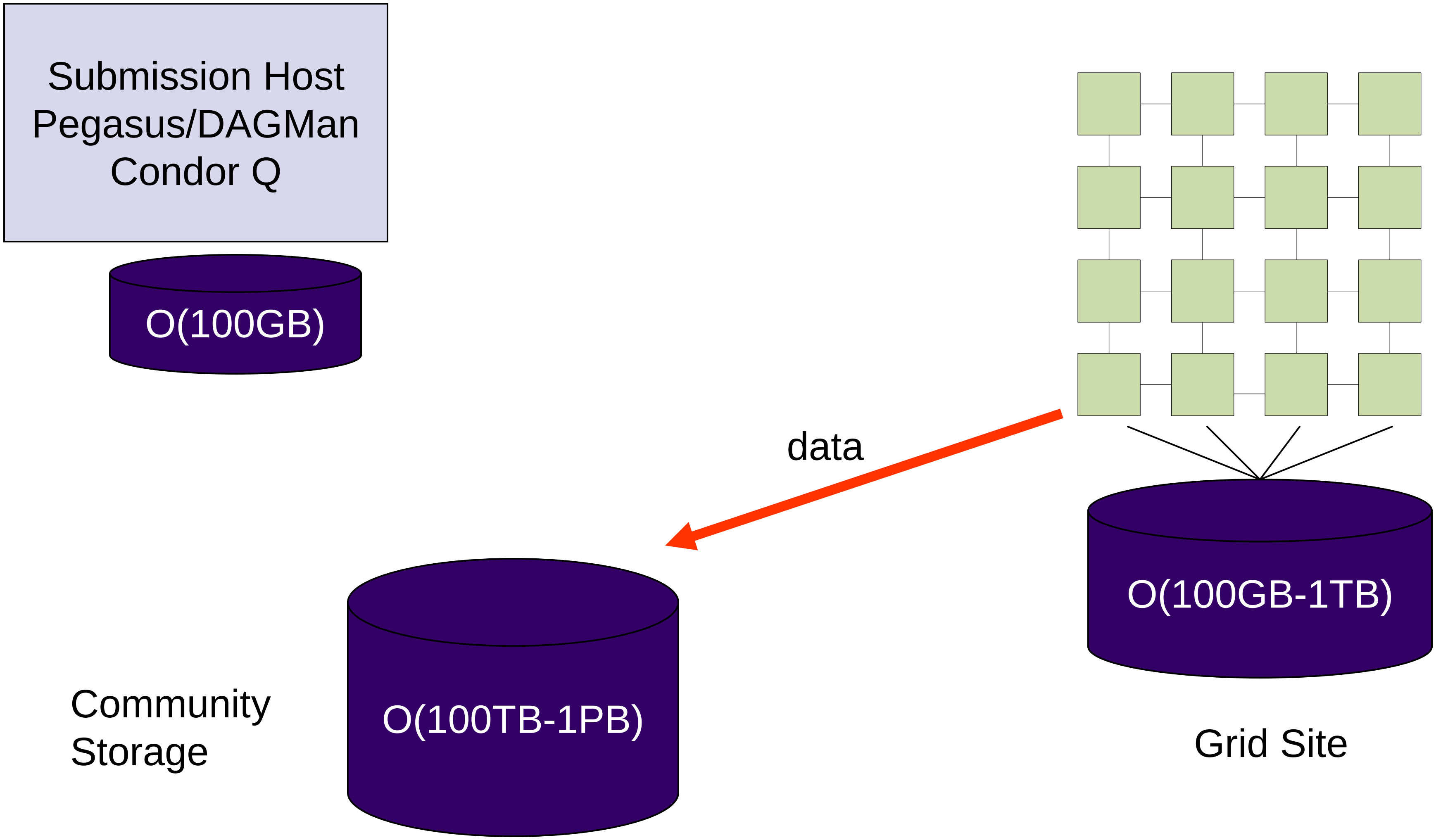
O(100GB-1TB)

Grid Site

Community
Storage

O(100TB-1PB)

Storage on the Grid



Submission Host
Pegasus/DAGMan
Condor Q

O(100GB)

data

O(100GB-1TB)

Grid Site

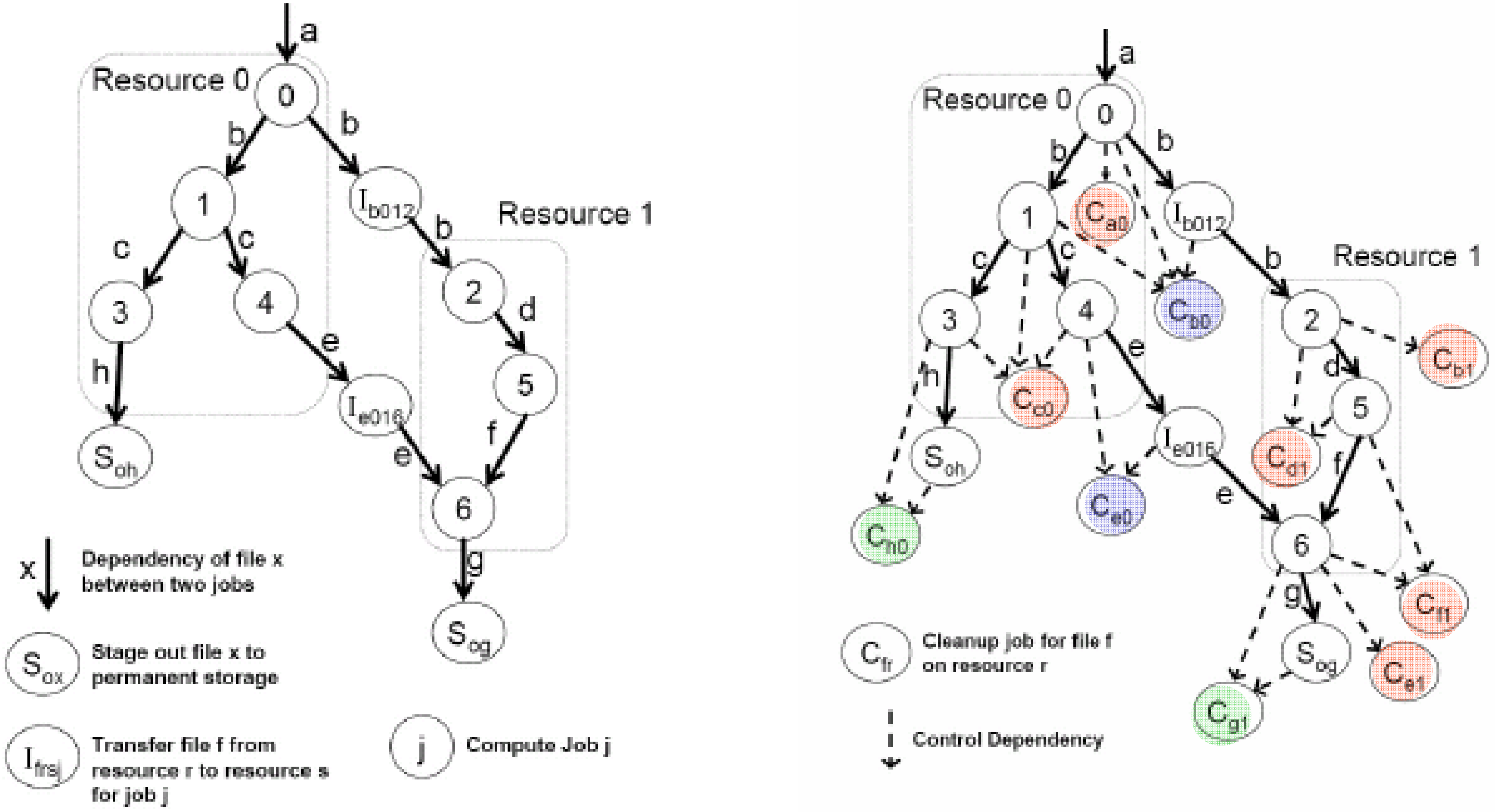
Community
Storage

O(100TB-1PB)

Workflow Footprint

- In order to improve the workflow footprint, we need to determine when data are no longer needed:
  - Because data was consumed by the next component and no other component needs it
  - Because data was staged-out to permanent storage
  - Because data are no longer needed on a resource and have been stage-out to the resource that needs it

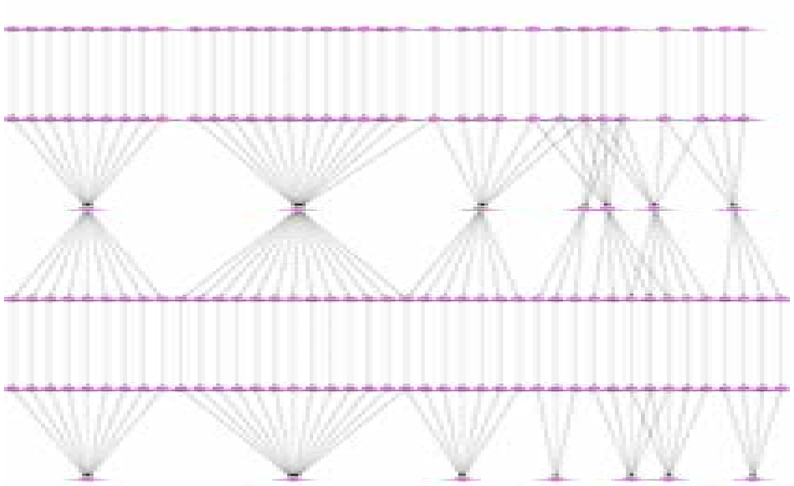
Cleanup Disk Space as Workflow Progresses



- For each node add dependencies to cleanup all the files used and produced by the node
- If a file is being staged-in from r1 to r2, add a dependency between the stage-in and the cleanup node
- If a file is being staged-out, add a dependency between the stage-out and the cleanup node

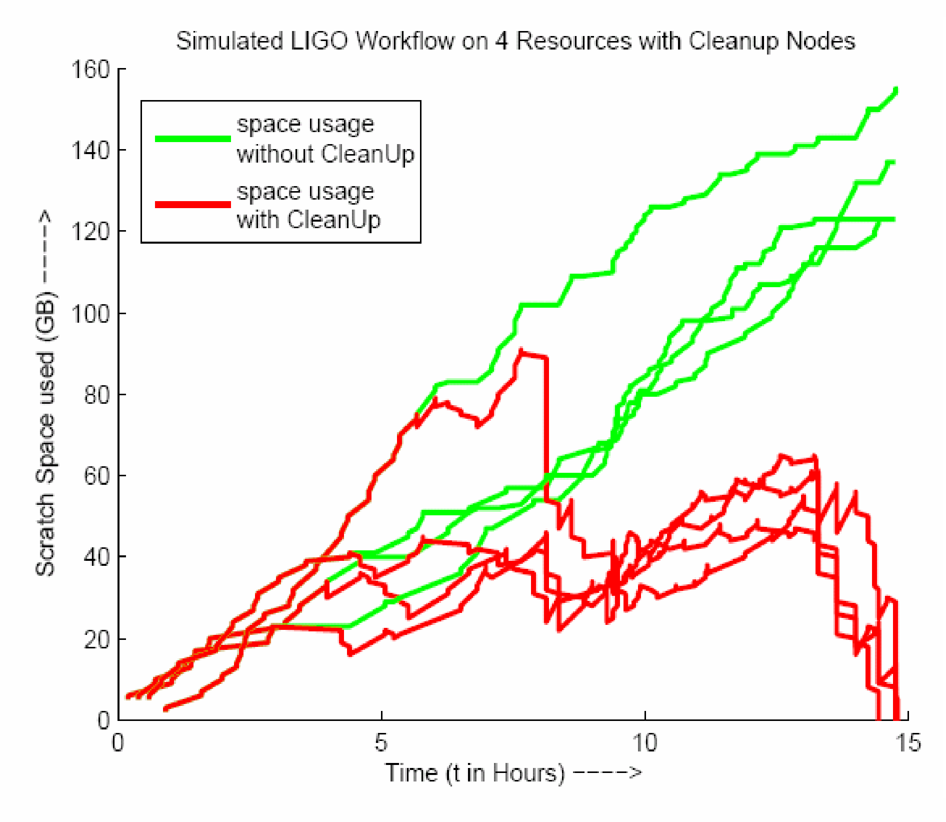
Evaluation

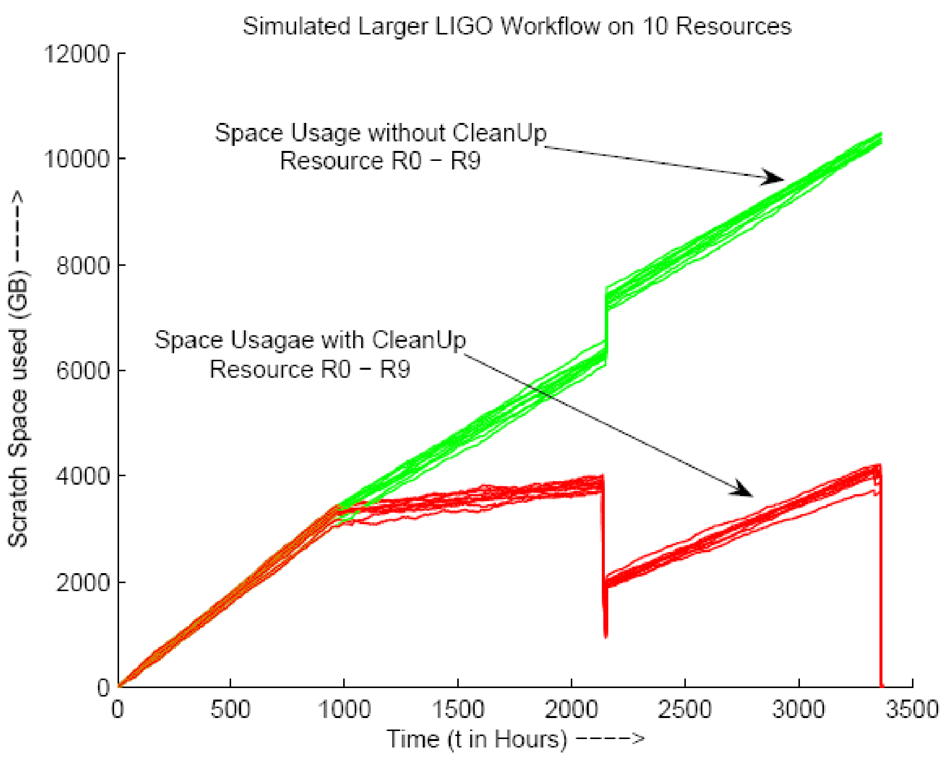
- Simulations
  - Extended Gridsim simulator
  - 4 and 10 resources
  - Random task scheduling
  - Assume sufficient storage



- Simulated LIGO workflows
  - Small test workflow, 166 tasks, 600 GB max total storage (includes intermediate data products)
  - Large-scale analysis, 38,954 tasks, ~100 TB total (includes intermediate data products)

Small and Large LIGO Workflow


Simulated LIGO Workflow on 4 Resources with Cleanup Nodes


Simulated Larger LIGO Workflow on 10 Resources

- Approximately 50% improvement in workflow data footprint

Storage-aware scheduling

- For all ready tasks
- Identify all resources that can accommodate the data
  - Expected disk usage EDU(i) = input (i) + output(i)
- Allocate tasks to the resource which can achieve the earliest finish time
- Cleanup unnecessary files as before
- If no resources satisfy the space requirements of any ready task, the algorithm halts with failure

Details in A. Ramakrishnan, et al., "Scheduling Data -Intensive Workflows onto Storage-Constrained Distributed Resources," in *Seventh IEEE International Symposium on Cluster Computing and the Grid — CCGrid 2007, Rio de Janeiro, Brazil*

Results

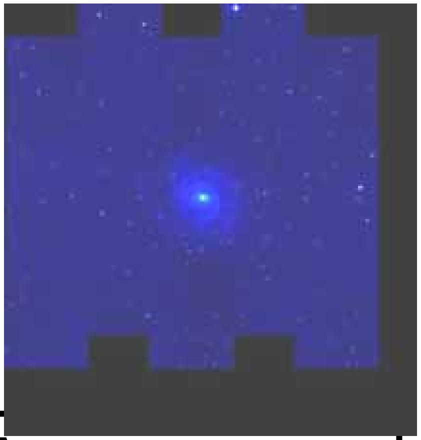
- Algorithms Simulated:
  - Storage-aware scheduling with cleanup *Storage/Cleanup*
  - Random scheduling with cleanup *(Random/Cleanup)*
  - Storage-aware scheduling without cleanup *Storage/No Cleanup*
- Application: Small LIGO workflow
- Environment:
  - Number of resources: 3, 6, 9
  - Network speed 1, 10, 100 MB/sec
  - Disk storage per resource: 10, 15, 20, 30

6 resources, time in seconds

Network Speed (MB/sec)	Storage GB/resource	Storage/Cleanup	Random/Cleanup	Storage/No cleanup
100	20-30	2,154	2,548	2,154
100	15	2,154	2,548	Fail
100	10	2,154	Fail	Fail
10	20-30	3,584	6,308	3,854
10	15	3,584	6,308	Fail
10	10	3,584	Fail	Fail
1	20-30	17,889	43,910	17,889
1	15	17,889	43,910	Fail
1	10	17,889	Fail	Fail

Experiments on the Grid and Astronomy Application

Montage: Generating mosaics of the sky: Composing a large image based on many individual images



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

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

Some issues with initial cleanup algorithm

- We have as many cleanup nodes as files
- Have some redundant dependencies
- May result in inefficiencies in workflow execution in real deployments

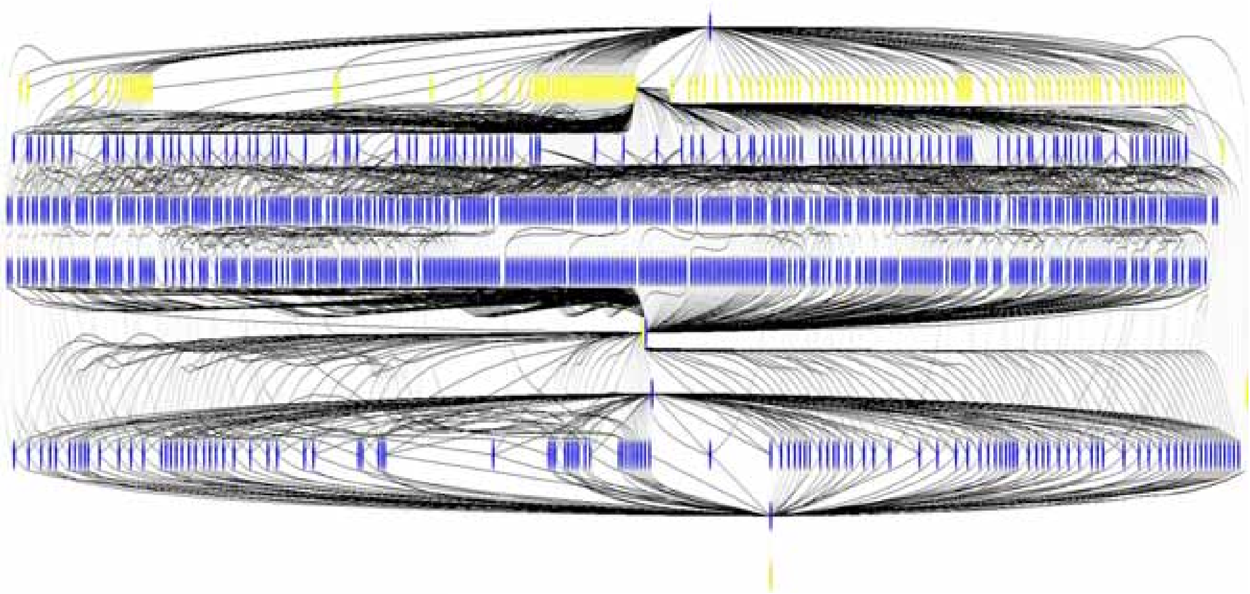
- New solution
  - have at most one cleanup task per computation task

Comparison via Simulation of the Data Cleanup Algorithms, Showing the Reduction in the Number of Cleanup Tasks and the Number of Dependencies.

LSC workflow	Max Space Used (MBs)	No of CleanUp Jobs	No of dependencies
Algorithm I	1027.13	237	840
Algorithm II	1028.23	96	238
2-degree MONTAGE	**Max Space Used (MBs)**	**No of CleanUp Jobs**	**No of dependencies**
Algorithm I	2405.71	2029	4211
Algorithm II	2409.71	731	1296

Algorithm I– One cleanup node per file
Algorithm II- At most on node per task

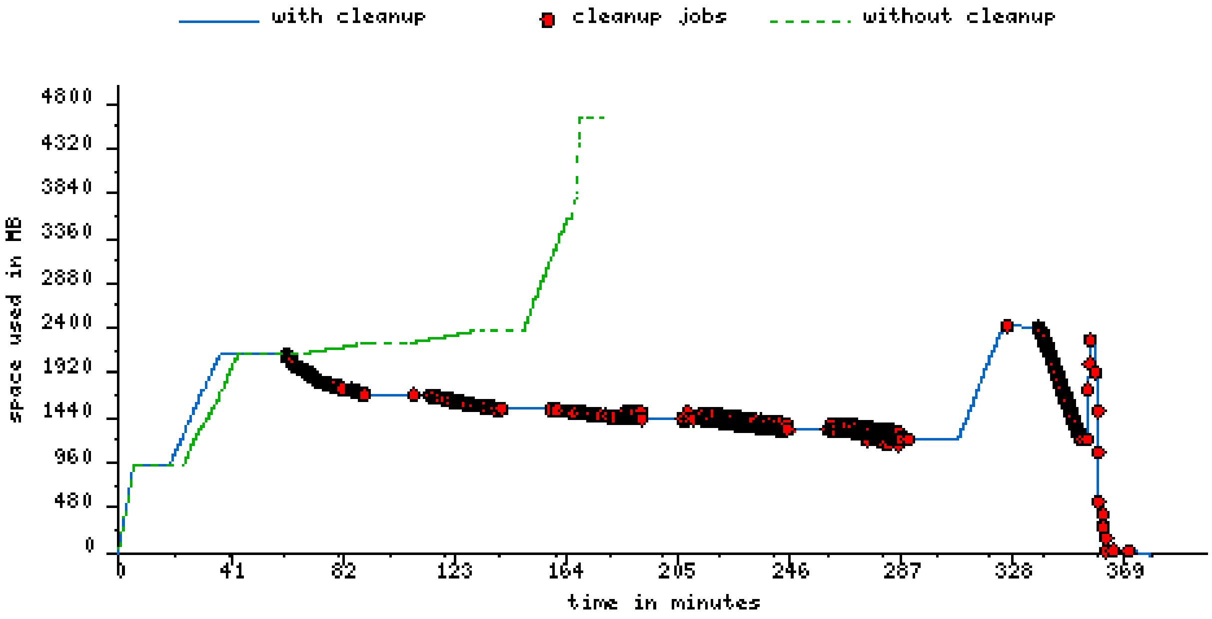
Cleanup on the Grid, Montage application

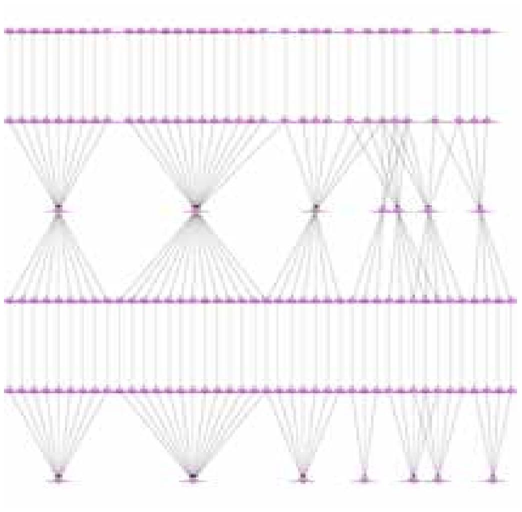


~ 1,200 nodes

1.25GB versus 4.5 GB

Open Science Grid

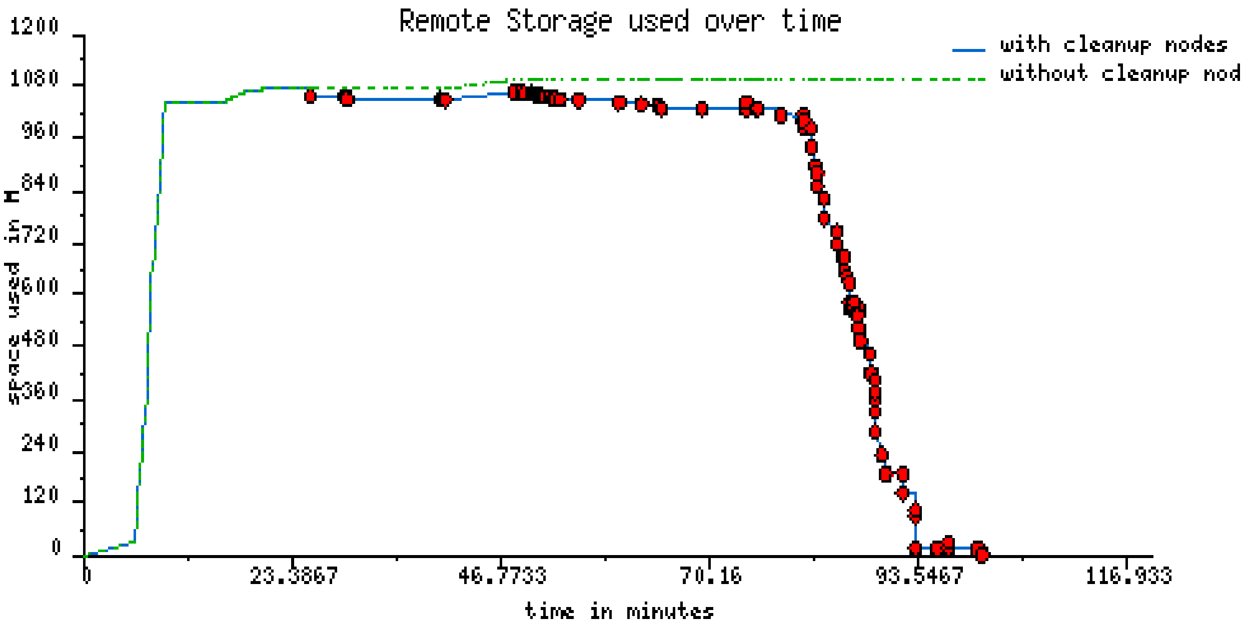


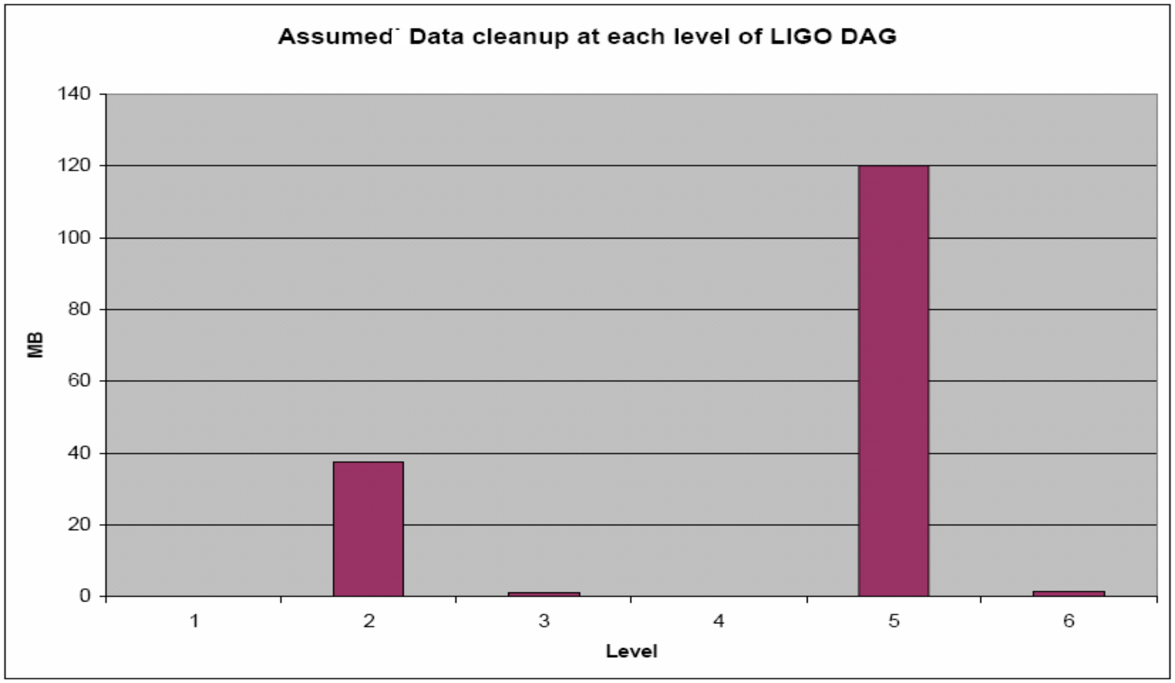


LIGO Inspiral Analysis Workflow

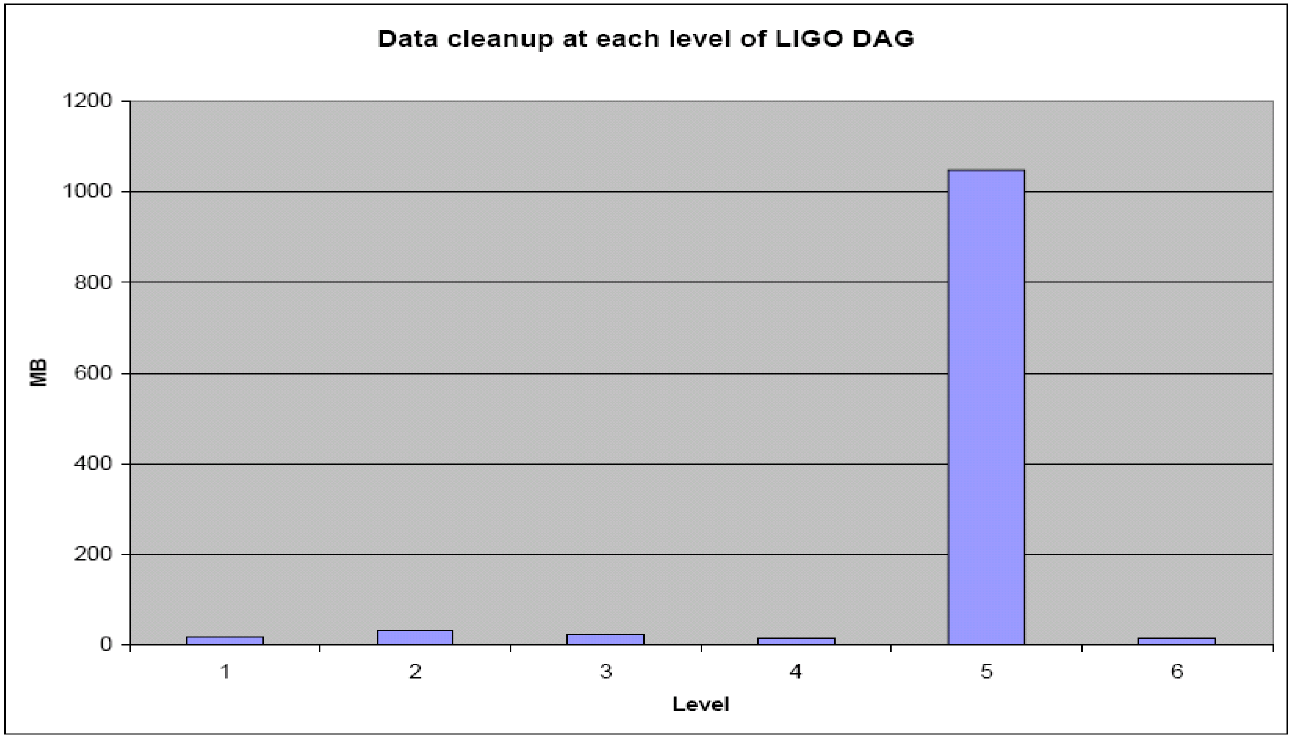
Small Workflow: 164 nodes

LIGO workflow running on OSG

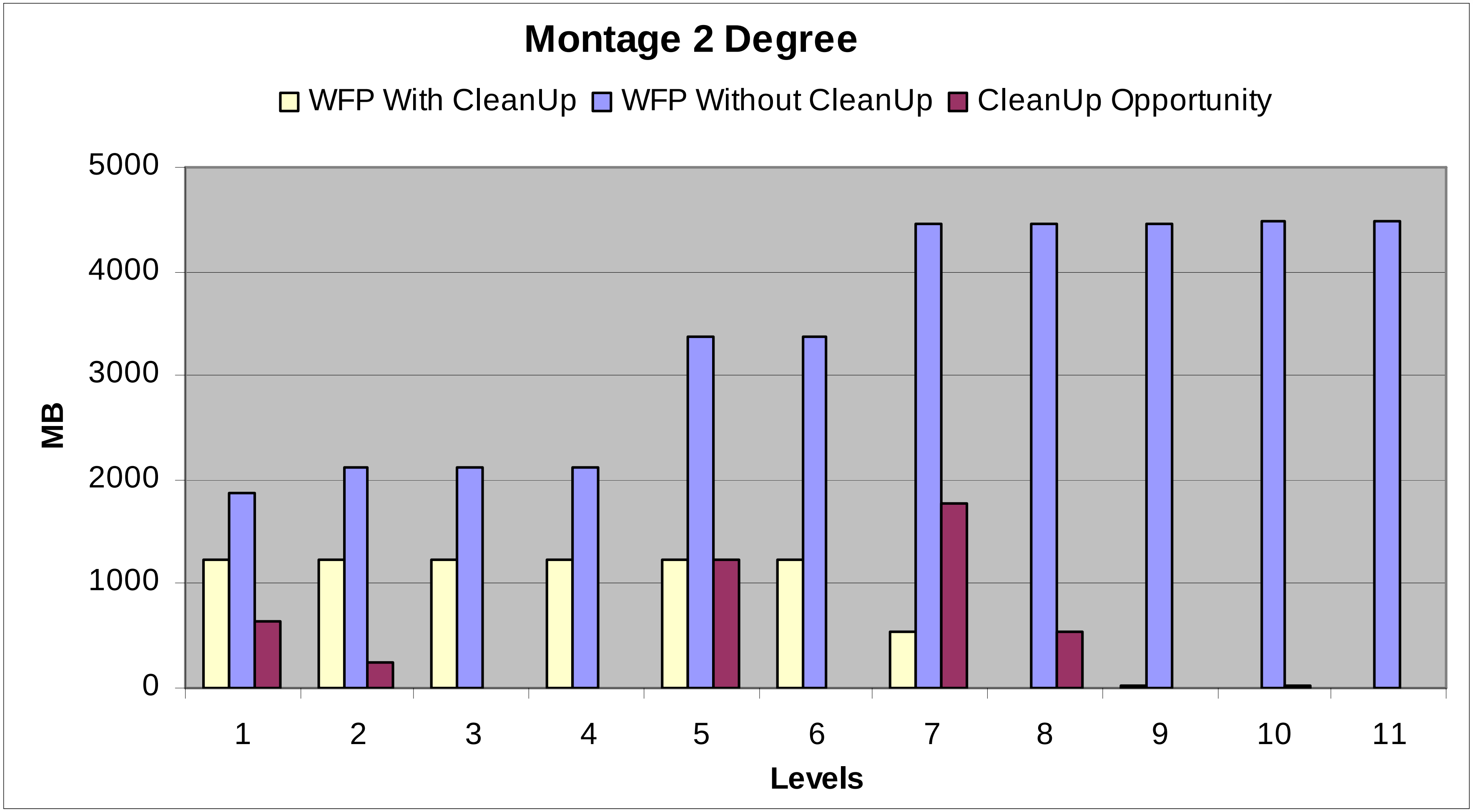


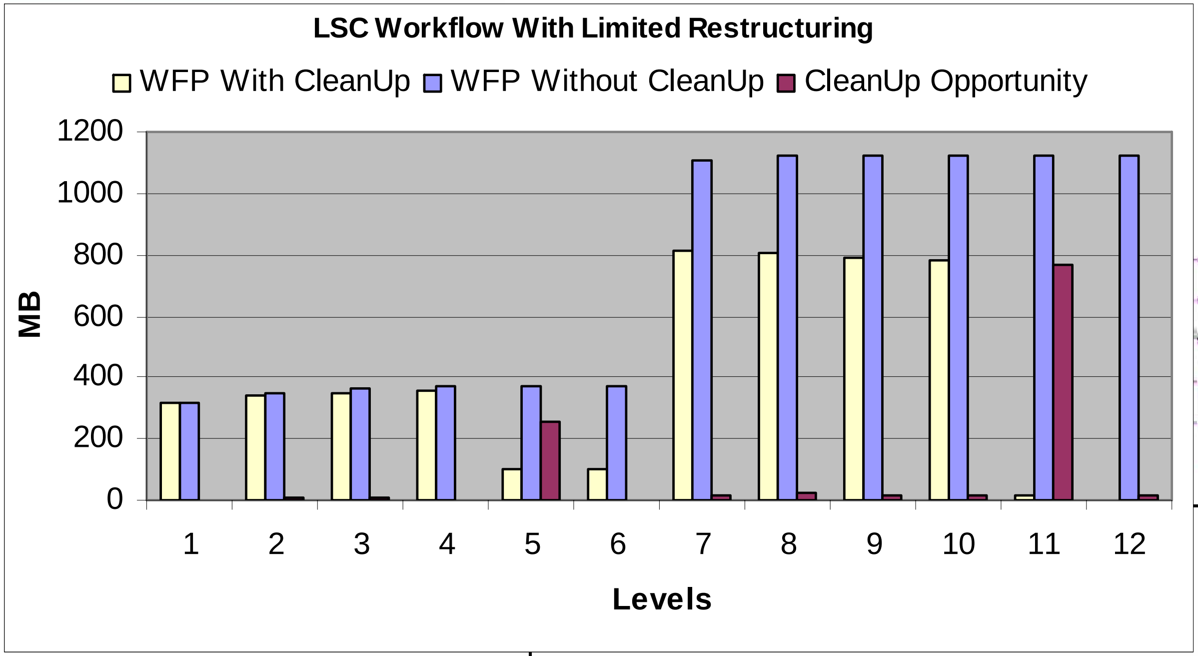


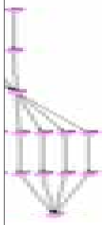
Assumes level-based scheduling, all nodes at a level need to complete before the next level starts



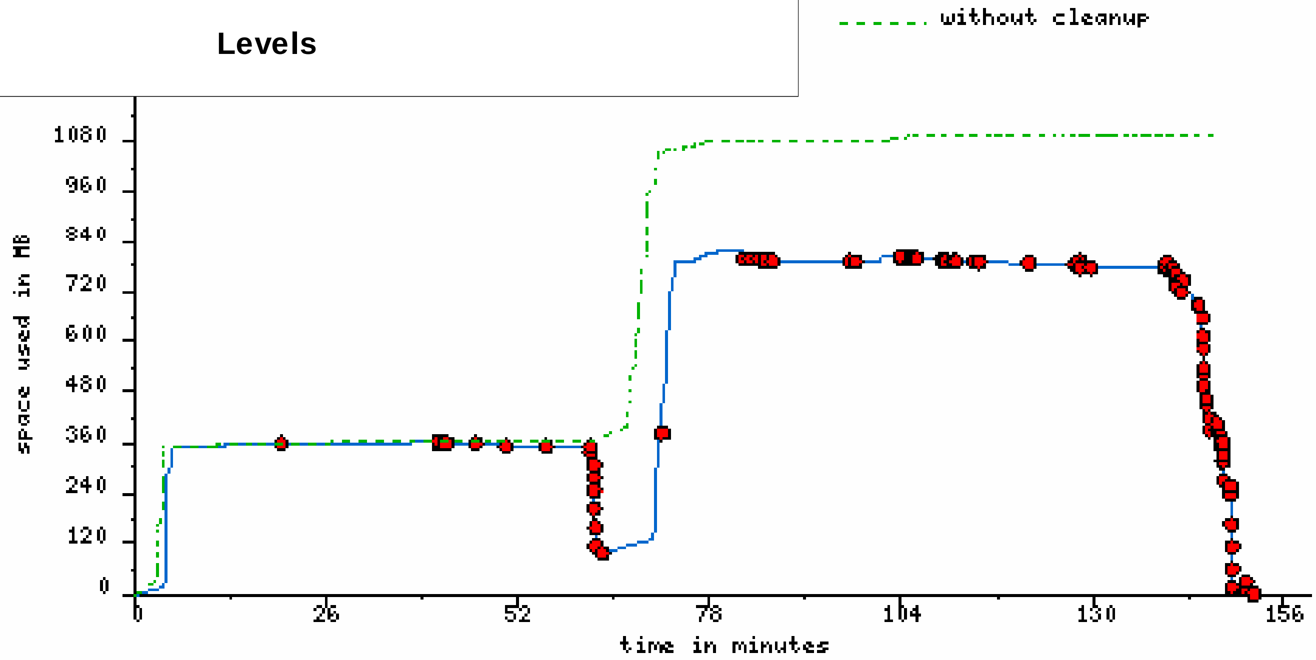
Montage Workflow







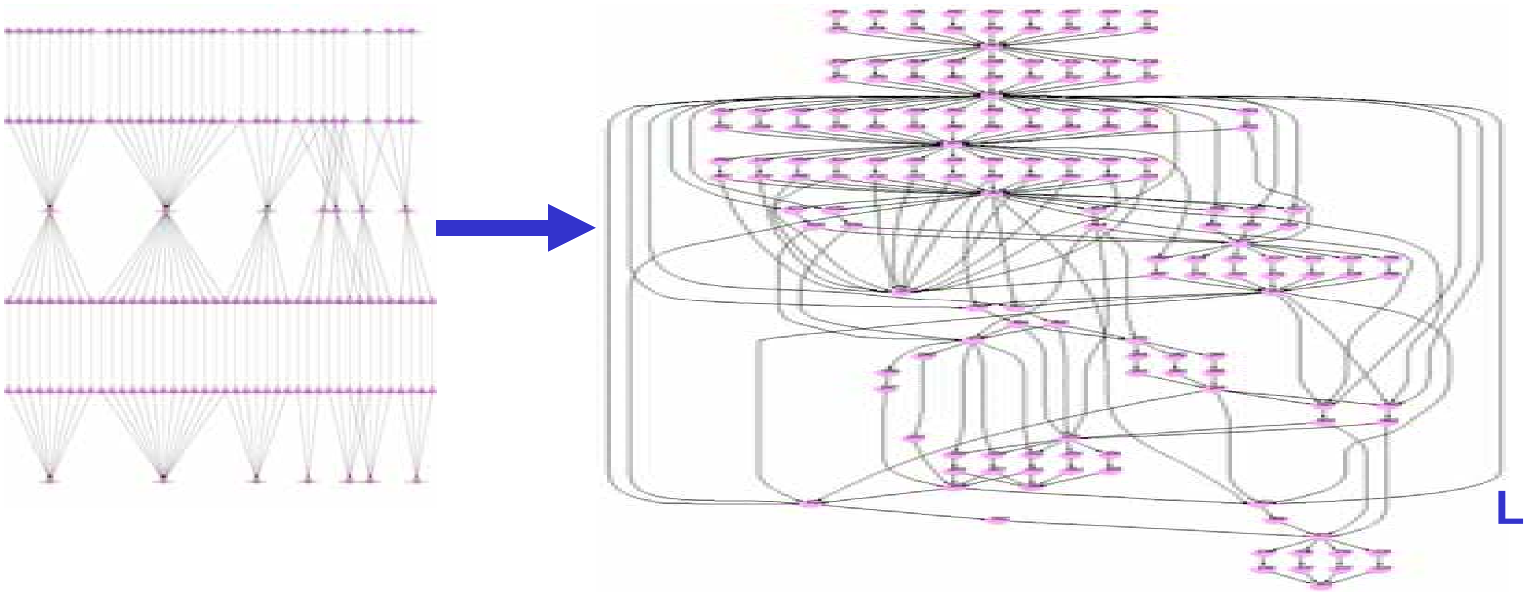
LIGO Workflows

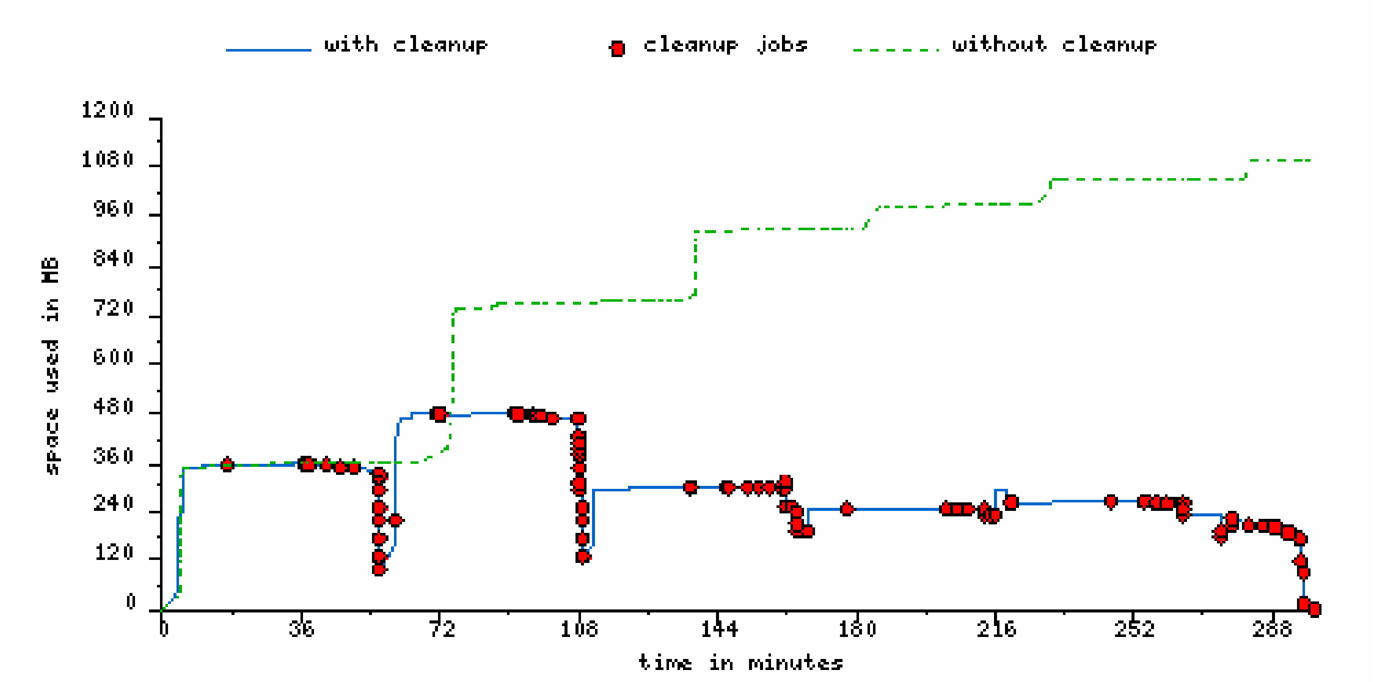


26% Improvement In disk space Usage

50% slower runtime

LIGO Workflows



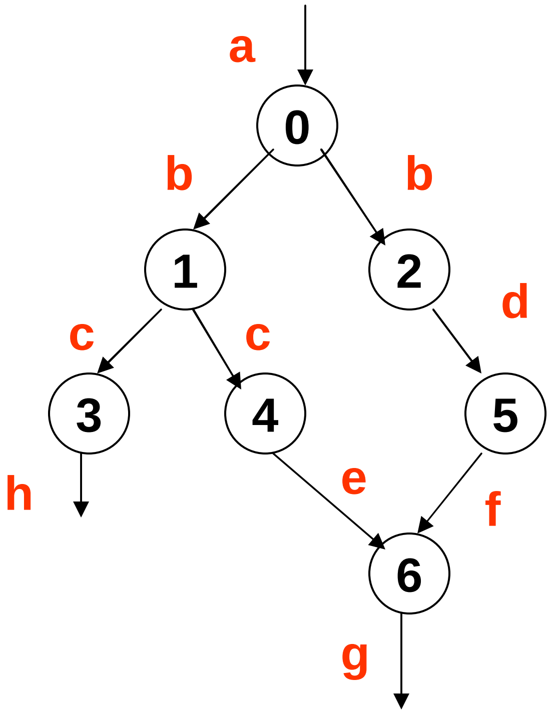


56% improvement in space usage

3 times slower in runtime

Challenges in implementing data space-aware scheduling

- Difficult to get accurate performance estimates for tasks
- Difficult to get good estimates of the sizes of the output data
  - Errors compound in the workflow
- Difficult to get accurate estimates of data storage space
  - Space is shared among many users
  - Hard to get allocation estimates
  - Even if you have space when you schedule, may not be there to receive all the data



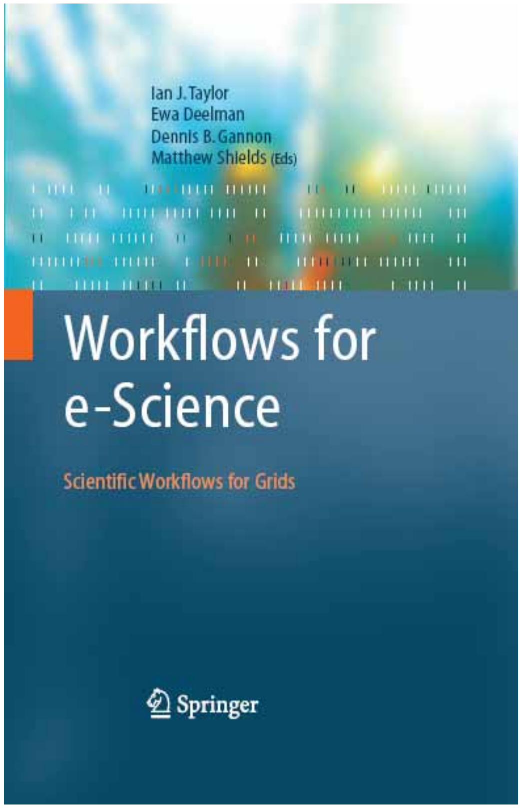
Conclusions

- Data are an important part of today’s applications and need to be managed
- Optimizing workflow disk space usage
  - Data workflow footprint concept applicable within one resource
  - Data-aware scheduling across resources
- Proposed an algorithm which can cleanup the data as a workflow progresses
  - The effectiveness of the algorithm depends on the structure of the workflow and its data characteristics
- Proposed an algorithm for data-aware scheduling with cleanup and evaluated it through simulations
- Showed that simulation and practice can differ
- Workflow restructuring may be needed to decrease footprint

Relevant Links

- **Pegasus:** pegasus.isi.edu
- **LIGO:** www.ligo.caltech.edu/
- **Montage:** montage.ipac.caltech.edu/
- **Open Science Grid:** www.opensciencegrid.org

- **Workflows for e-Science**
  I.J. Taylor, E. Deelman, D. B. Gannon
  M. Shields (Eds.), Springer, Dec. 2006
- **NSF Workshop on Challenges of Scientific Workflows :** www.isi.edu/nsf-workflows06,
  E. Deelman and Y. Gil (chairs)
- **OGF Workflow research group**
  www.isi.edu/~deelman/wfm-rg



Ian J. Taylor
Ewa Deelman
Dennis B. Gannon
Matthew Shields (Eds)

Workflows for
e-Science

Scientific Workflows for Grids

Springer