open.michigan

Author(s): August E. Evrard, PhD. 2010

Citation Key

for more information see: http://open.umich.edu/wiki/CitationPolicy

Use + Share + Adapt

{ Content the copyright holder, author, or law permits you to use, share and adapt. }

PD-GOV **Public Domain – Government**: Works that are produced by the U.S. Government. (17 USC § 105)

PD-EXP **Public Domain – Expired**: Works that are no longer protected due to an expired copyright term.

PD-SELF **Public Domain – Self Dedicated**: Works that a copyright holder has dedicated to the public domain.

ZERO **Creative Commons – Zero Waiver**

BY **Creative Commons – Attribution License**

BY-SA **Creative Commons – Attribution Share Alike License**

BY-NC **Creative Commons – Attribution Noncommercial License**

BY-NC-SA **Creative Commons – Attribution Noncommercial Share Alike License**

GNU-FDL **GNU – Free Documentation License**

Make Your Own Assessment

{ Content Open.Michigan believes can be used, shared, and adapted because it is ineligible for copyright. }

PD-INEL **Public Domain – Ineligible**: Works that are ineligible for copyright protection in the U.S. (17 USC § 102(b)) *laws in your jurisdiction may differ

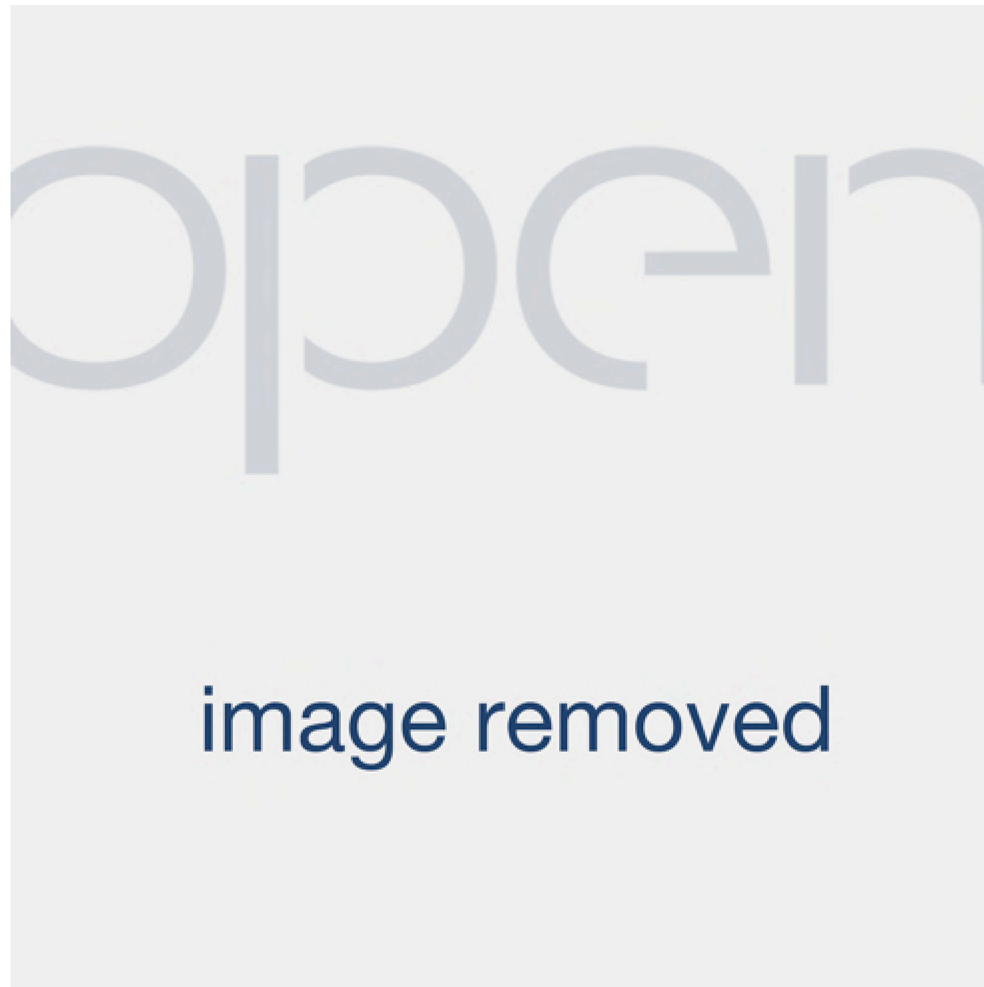
{ Content Open.Michigan has used under a Fair Use determination. }

FAIR USE **Fair Use**: Use of works that is determined to be Fair consistent with the U.S. Copyright Act. (17 USC § 107) *laws in your jurisdiction may differ

Our determination **DOES NOT** mean that all uses of this 3rd-party content are Fair Uses and we **DO NOT** guarantee that your use of the content is Fair.

To use this content you should **do your own independent analysis** to determine whether or not your use will be Fair.

Cyberscience: Computational Science and the Rise of the Fourth Paradigm



Please see original article on the "New Era in IT" and cloud computing at http://www.vmware.com/files/pdf/cloud/VMware-and-Cloud-Computing-BR-EN.pdf.

Honors 352, Class #0.15
August E. (Gus) Evrard, PhD

Fall 2010

University of Michigan

today

* group project updates

    Note: Final presentations are 40 minutes each

    7, 9 December (last two meetings)

* lecture: Grid and Cloud Overview

    accessible introduction to Grid @ http://www.gridcafe.org/

* **Thursday**: guest lecture by

    Prof. Dan Atkins, UM School of Information (founding Dean),

    and Vice-president for Cyberinfrastructure, OVPR

history of grid computing concepts

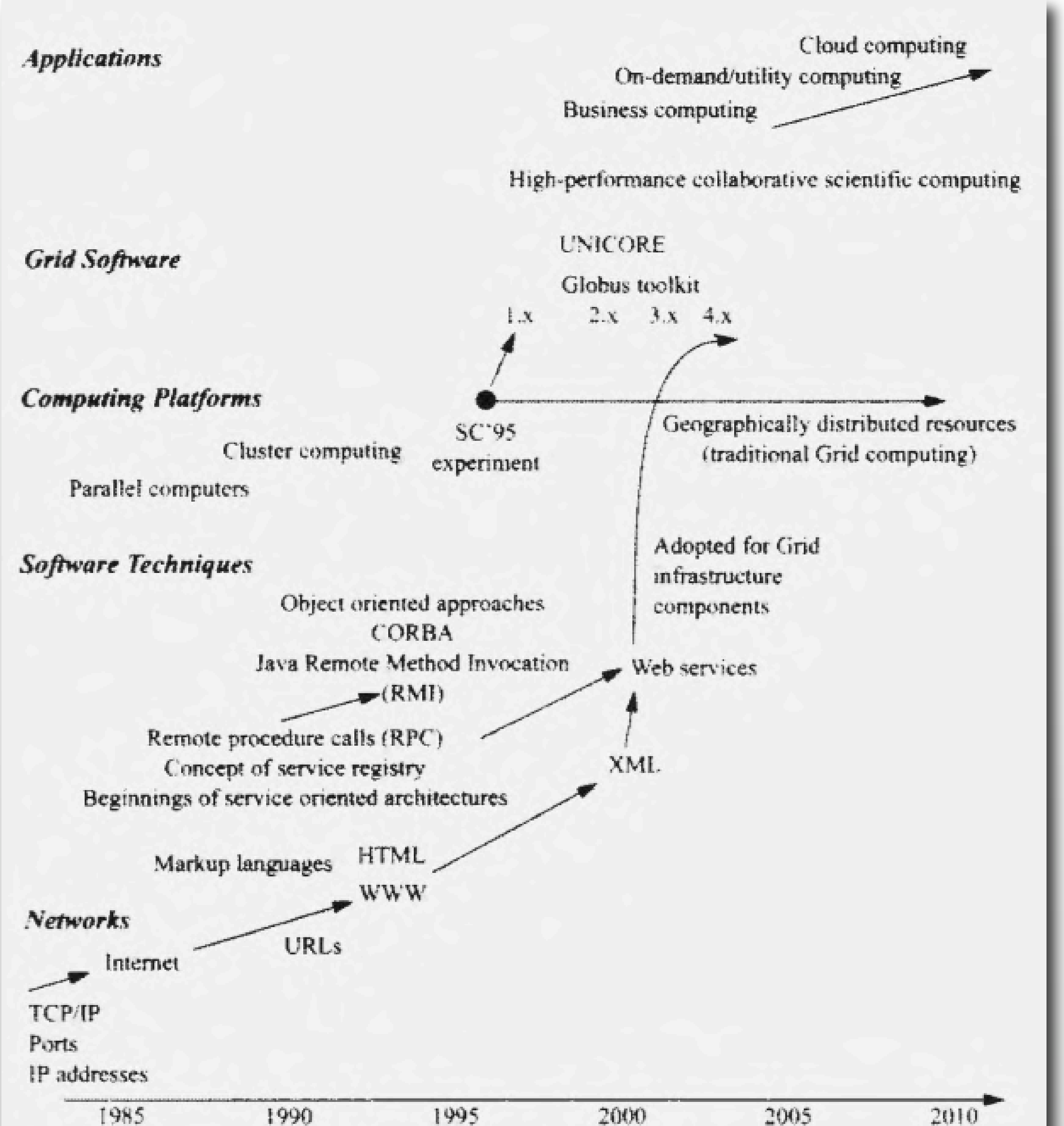


Figure 1.2 Key concepts in the history of Grid computing.

 Wilkinson, *Grid Computing: Techniques and Applications*, pg. 10.

Globus toolkit

<u>Goal:</u> coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations. Grid2, p. 37

* offers mechanisms to enable a distributed computing environment, with tools to support

– communication

– resource location

– resource scheduling

– authentication

– data access

* philosophy

– no centralized control

– standard, open protocols

– non-trivial Quality of Service (QoS)

"Together, the various Globus toolkit modules can be thought of as defining a metacomputing virtual machine. The definition of this virtual machine simplifies application development and enhances portability by allowing programmers to think of geographically distributed, heterogeneous collections of resources as unified entities."

Ian Foster and Carl Kesselman,
"Globus: A Metacomputing Infrastructure Toolkit,"
The International Journal of Supercomputer Applications
and High Performance Computing, 1997.

GLOBUS: A METACOMPUTING INFRASTRUCTURE TOOLKIT

Ian Foster

MATHEMATICS AND COMPUTER SCIENCE DIVISION
ARGONNE NATIONAL LABORATORY
ARGONNE, IL 60439

Carl Kesselman

INFORMATION SCIENCES INSTITUTE
UNIVERSITY OF SOUTHERN CALIFORNIA
MARINA DEL REY, CA 90292

Summary

The Globus system is intended to achieve a vertically integrated treatment of application, middleware, and network. A low-level toolkit provides basic mechanisms such as communication, authentication, network information, and data access. These mechanisms are used to construct various higher level metacomputing services, such as parallel programming tools and schedulers. The long-term goal is to build an adaptive wide area resource environment (AWARE), an integrated set of higher level services that enable applications to adapt to heterogeneous and dynamically changing metacomputing environments. Preliminary versions of Globus components were deployed successfully as part of the I-WAY networking experiment.

The International Journal of Supercomputer Applications and High Performance Computing,
Volume 11, No. 2, Summer 1997, pp. 115-128

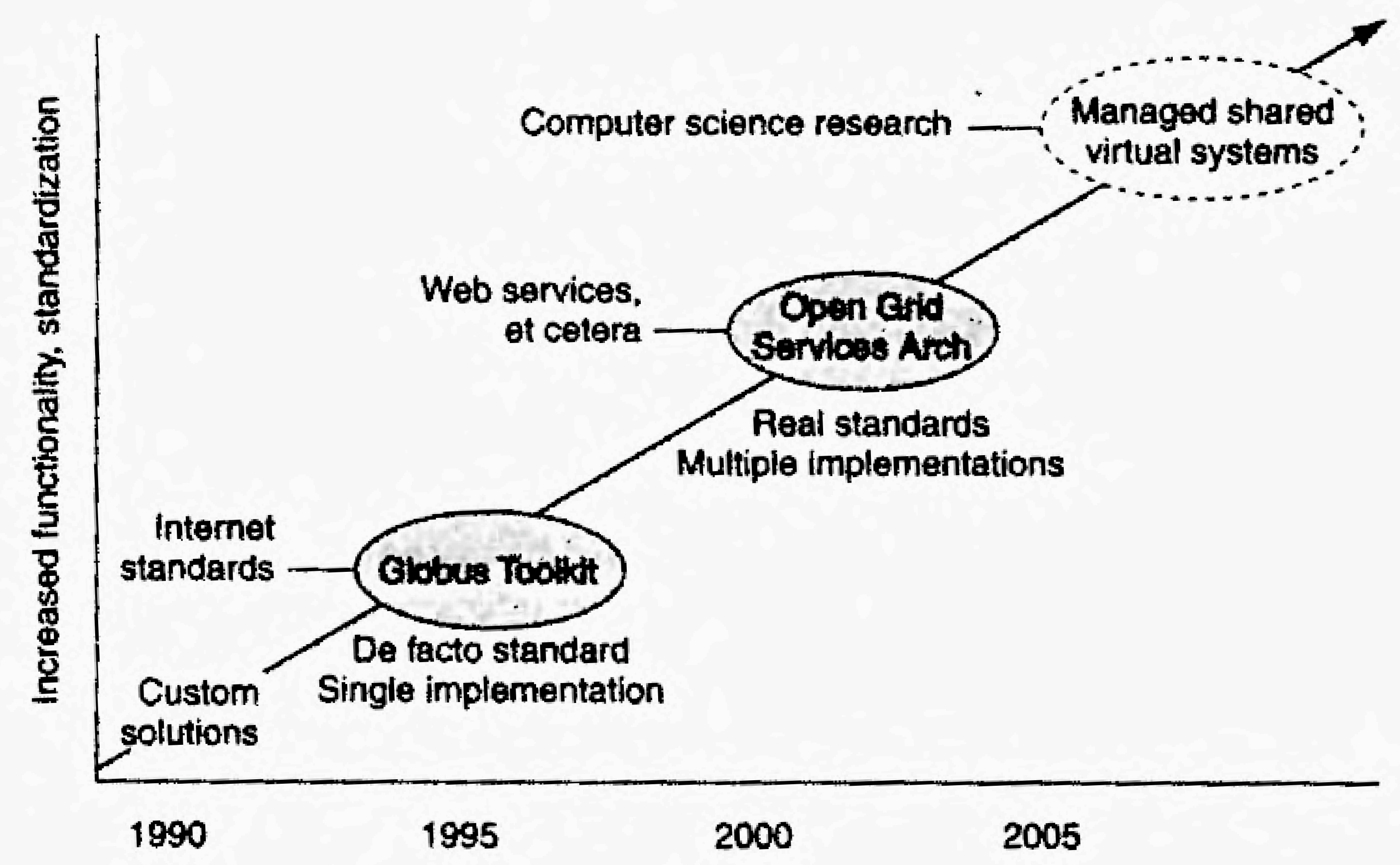
Globus evolution

* early `custom' period (I-WAY demo)

* Globus Toolkit 2 (GT2, 1997) thousands deployed

* GT3, Open Grid Services Architecture (OGSA, 2002) alignment with industry needs, web services

* today - expanding scope (cell phones) and scale, active management



The evolution of Grid technologies.

 Foster and Kesselman, *The Grid 2*, pg. 44.

Globus and the Grid

We define a Grid as a system that coordinates distributed resources using standard, open general-purpose protocols and interfaces to deliver nontrivial qualities of service. Key elements of this definition- Grid2, p.46

- *Coordinates distributed resources.* A Grid integrates and coordinates resources and users that live within different control domains—for example, the user's desktop versus central computing, different administrative units of the same company, and/or different companies—and addresses the issues of security, policy, payment, membership, and so forth that arise in these settings. Otherwise, we are dealing with a local management system.
- *Using standard, open, general-purpose protocols and interfaces.* A Grid is built from multipurpose protocols and interfaces that address such fundamental issues as authentication, authorization, resource discovery, and resource access. As we discuss in material to follow, it is important that these protocols and interfaces be *standard* and *open*. Otherwise, we are dealing with an application-specific system.
- *To deliver nontrivial qualities of service.* A Grid allows its constituent resources to be used in a coordinated fashion to deliver various qualities of service—relating, for example, to response time, throughput, availability, and security—and/or coallocation of multiple resource types to meet complex user demands, so that the utility of the combined system is significantly greater than that of the sum of its parts.

 Foster and Kesselman, The Grid 2, pg. 46.

Grid slides from UCSD CSE225, Spring 2004

Andrew A. Chien

Andrew is the Vice President, Director of Intel Research

He is also an Adjunct Professor in the Dept of Computer Science and Engineering at the University of California, San Diego and an Adjunct Professor in the Dept of Computer Science at the University of Illinois.

Andrew's Office: 2226 EBU3b (CSE), Email: achien at ucsd dot edu

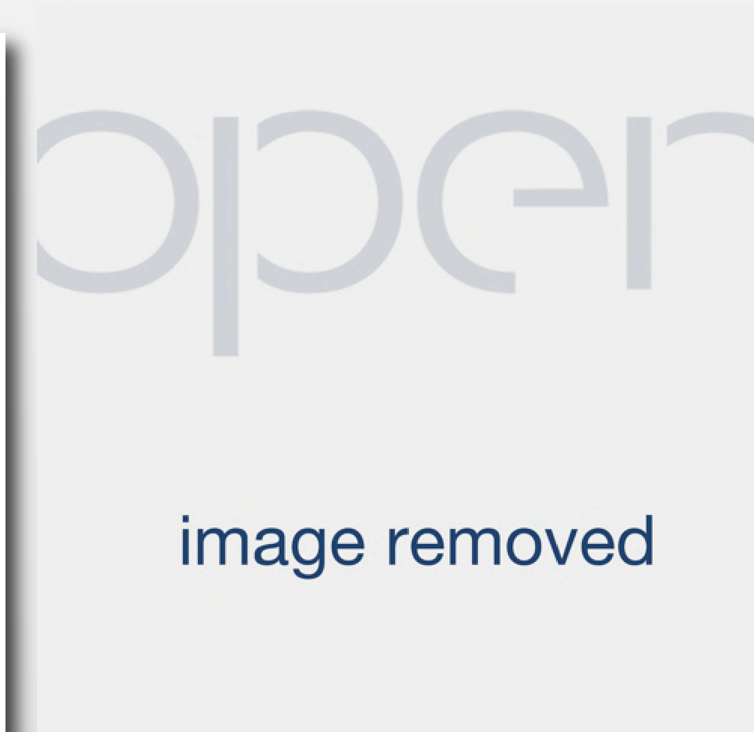
Andrew is Fellow of the Association for Computing Machinery (ACM), the Institute for Electrical and Electronics Engineers (IEEE), and in 2008 was elected a Fellow of the American Association for the Advancement of Science (AAAS).

Grids and High Performance Distributed Computing

Andrew Chien

March 31, 2004

CSE225, Spring 2004



Please go to the original slide show on this talk at http://www-csag.ucsd.edu/teaching/cse225s04/Lectures/Lec2-Globus-Grid-Architecture.pdf and reference slides 6-20, and 22 which have been removed from this presentation.

Global Resource Allocation Management

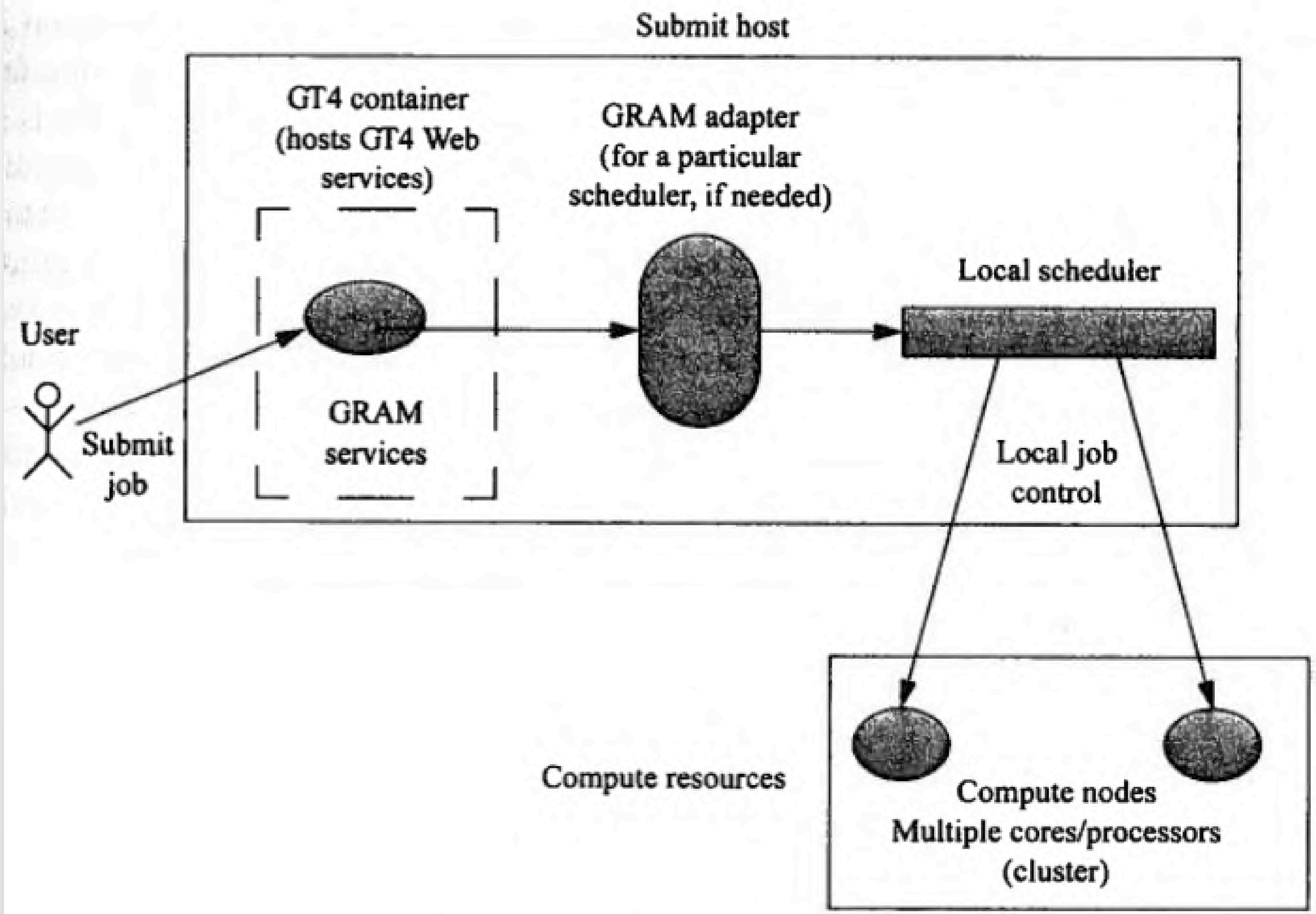


Figure 2.3 Basic job submission components.

 Wilkinson, *Grid Computing: Techniques and Applications*, pg. 39.

Job Description Elements in RSL-2/JDD

TABLE 2.1 SOME RSL-2/JDD ELEMENTS (PARTIAL LIST)

Element	Meaning
`argument`	A command line argument for the executable.
`count`	The number of executions of the executable. Default: 1.
`directory`	Path of default directory used for the requested job. Default: `${GLOBUS_USER_HOME}`
`environment`	Definition of environment variables in addition to default variables.
`executable`	Name of the executable file.
`factoryEndpoint`	Managed Job Factory service endpoint for submission of job.
`fileCleanUp`	Files local to the job to be removed.
`fileStageIn`	Files to be staged to nodes which will run job. Each specified as pair ("remote URL" "local file").
`fileStageOut`	Files to be staged from job. Each specified as pair ("local file" "remote URL").
`job`	Job description element. Contains elements describing job.
`jobType`	Specifies how the jobmanager should start the job: `single`, `multiple`, `mpi`, or `condor`.
`maxCpuTime`	Maximum CPU time for a single execution of executable, in minutes.
`maxMemory`	Maximum amount of memory for a single execution of executable, in Megabytes.
`minMemory`	Minimum amount of memory for a single execution of executable, in Megabytes.
`multiJob`	Multiple job element. Contains job elements.
`stderr`	Name of remote file to store standard error from job.
`stdin`	Name of file to be used as standard input for executable on remote machine.
`stdout`	Name of remote file to store standard output from job.

 Wilkinson, *Grid Computing: Techniques and Applications*, pg. 44.

File Staging

Quote on JDD staging process removed. See Wilkinson, *Grid Computing: Techniques and Applications*, pg. 57.

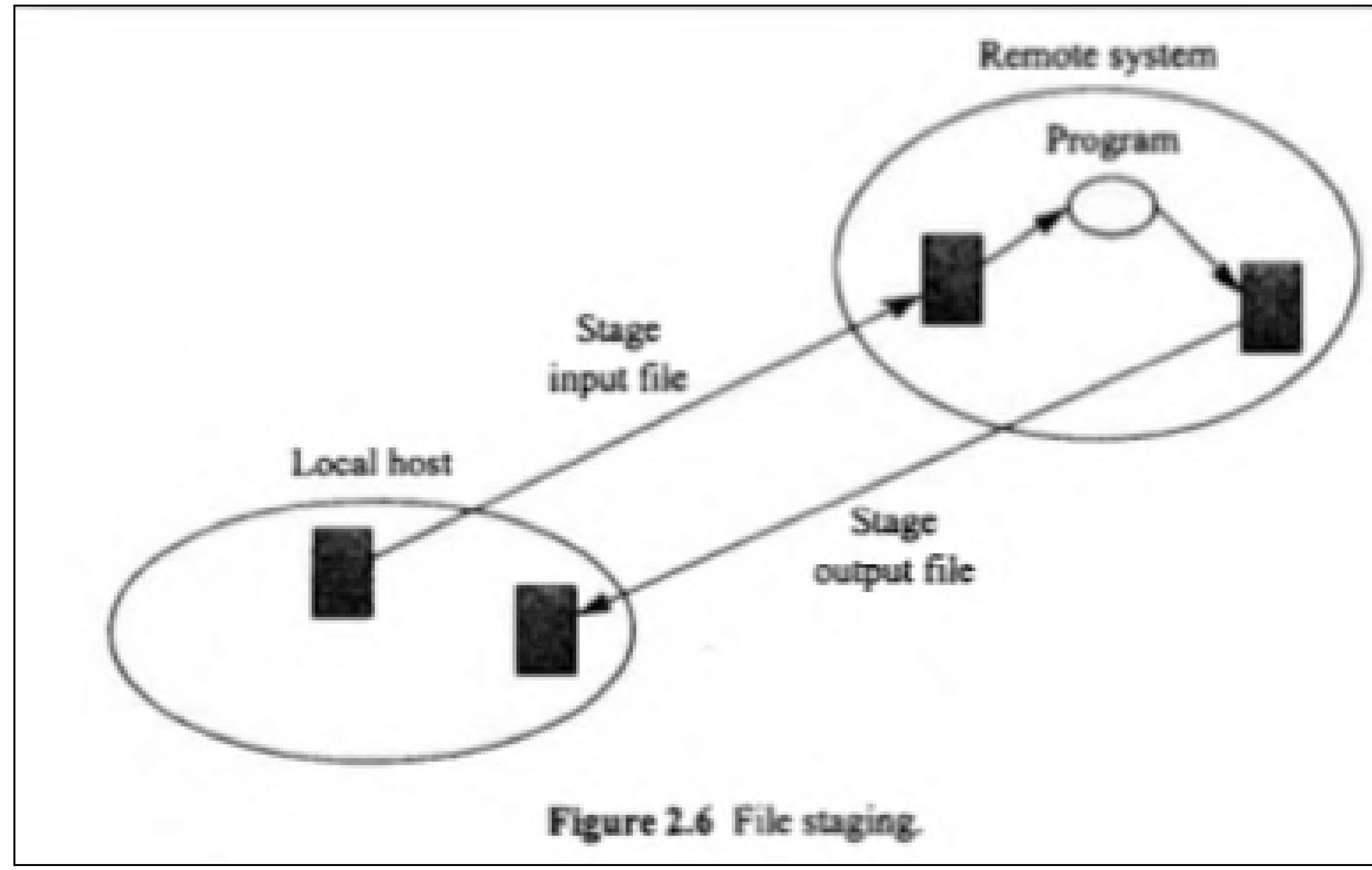


Figure 2.6 File staging.

Wilkinson, *Grid Computing: Techniques and Applications*, pg. 57.

Grid authentication and authorization

authentication- establishing the identity of an entity (user or process)
built atop Public Key Infrastructure (PKI) encryption

authorization- establishing access to resources for authenticated entities
built atop Access Control Lists (ACL's)

single sign-on- use certificates granted by a Certificate Authority to authenticate to multiple resources

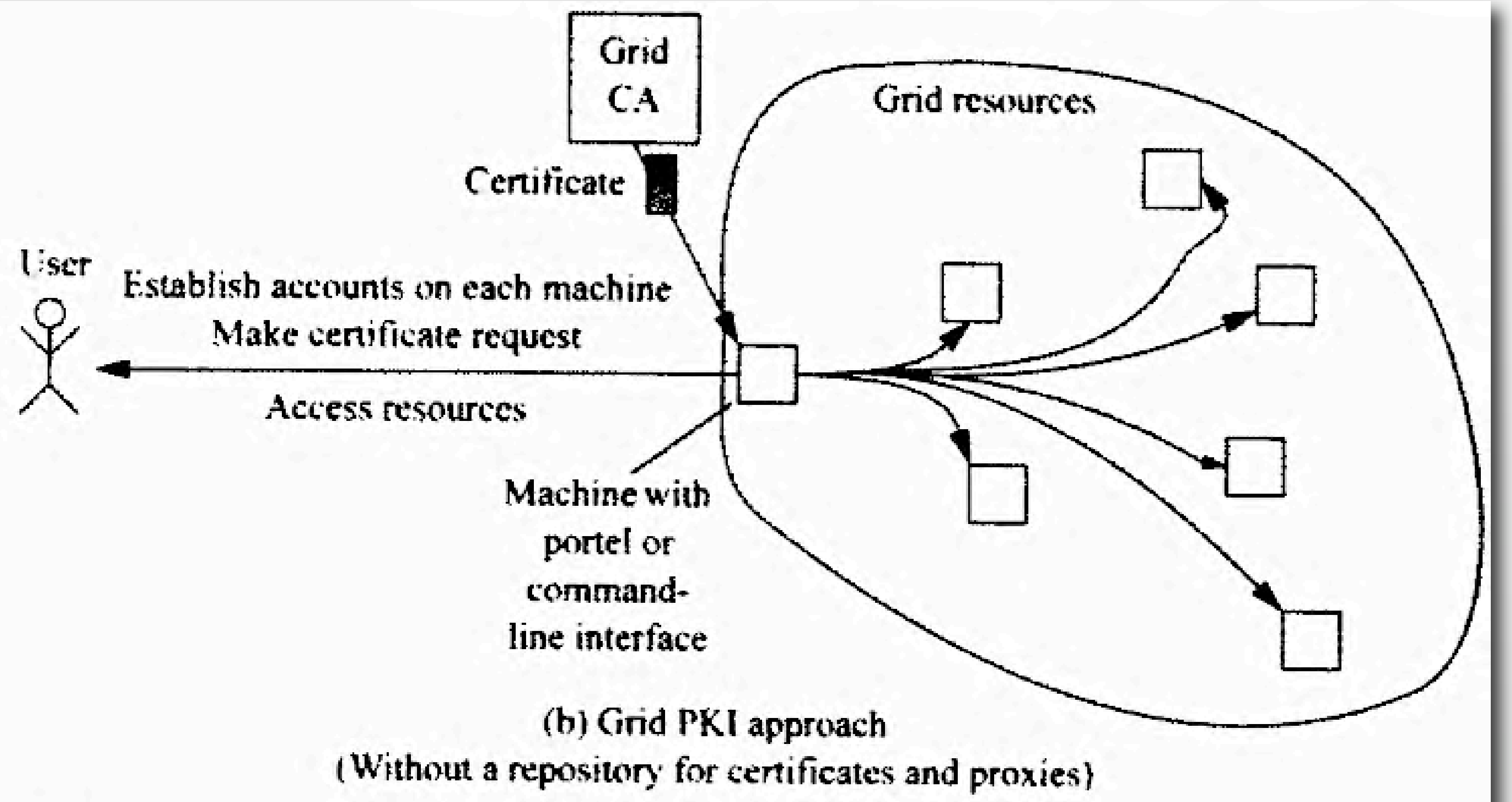


Figure 5.1 ssh login to remote machines compared to a Grid computing environment.

 Wilkinson, *Grid Computing: Techniques and Applications*, pg. 152.

Grid certificate process

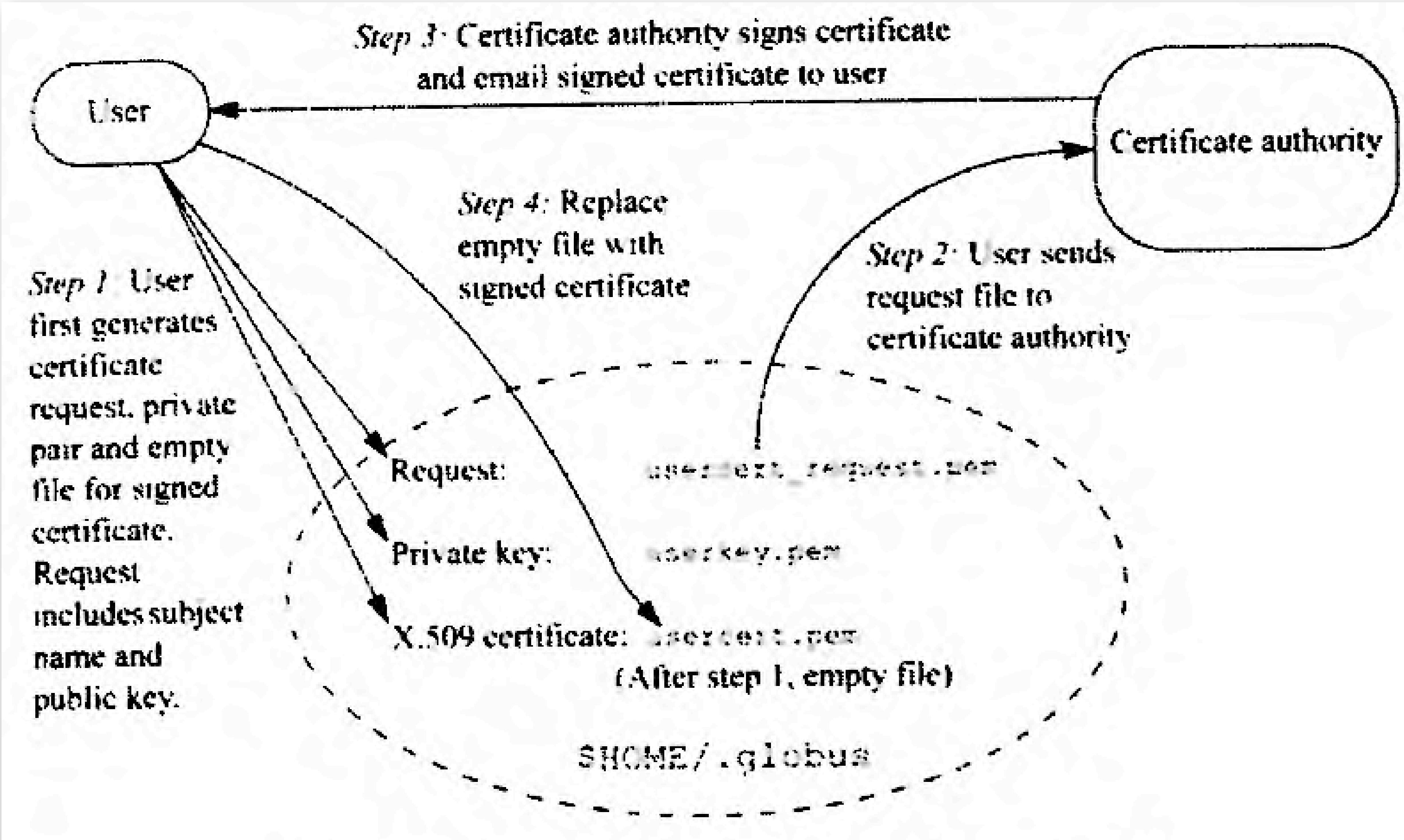


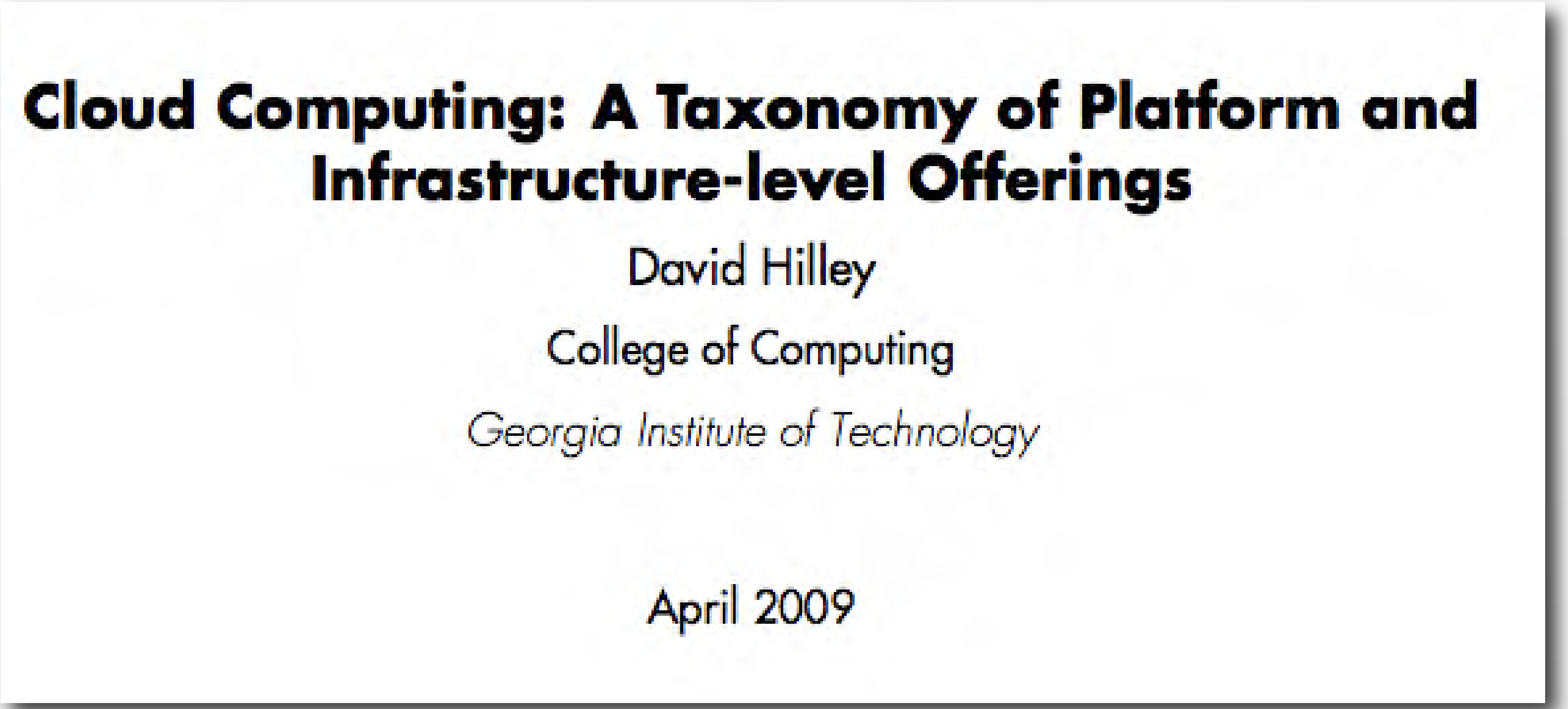
Figure 5.4 Getting a signed certificate using Globus commands.

 Wilkinson, *Grid Computing: Techniques and Applications*, pg. 160.

certificate process exercise

Programming instructions on grid computing removed. For more information, go to Wilkinson, *Grid Computing: Techniques and Applications,* pg. 62.

virtualization and the cloud



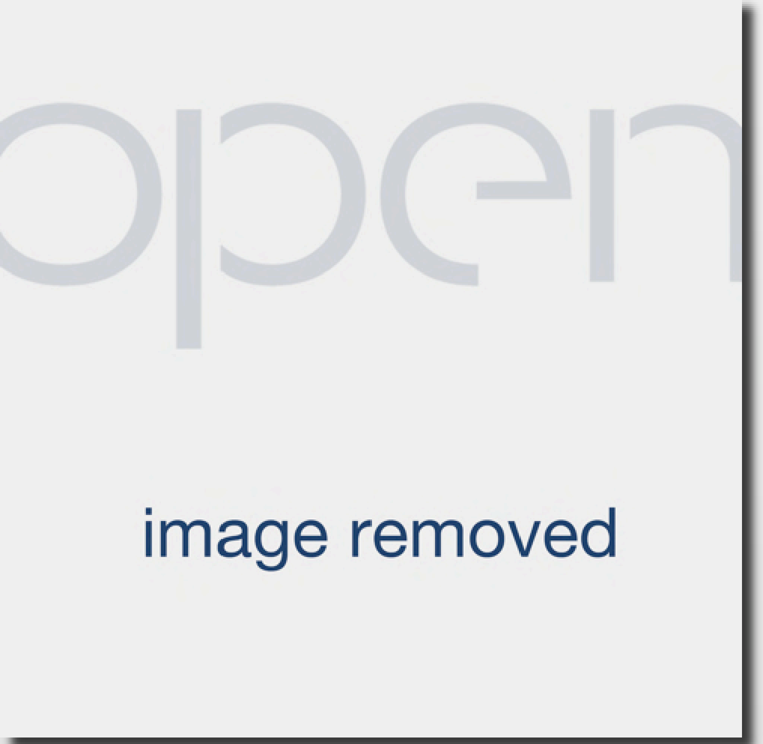
Cloud Computing: A Taxonomy of Platform and Infrastructure-level Offerings

David Hilley

College of Computing

Georgia Institute of Technology

April 2009



Background section from paper noted above removed.

efficiencies to be gained

* hardware and operations costs decline as data center size increases

Technology	Cost in Medium-sized DC	Cost in Very Large DC	Ratio
Network	$95 per Mbit/sec/month	$13 per Mbit/sec/month	7.1
Storage	$2.20 per GByte / month	$0.40 per GByte / month	5.7
Administration	140 Servers / Administrator	>1000 Servers / Administrator	7.1

Table 1: Cloud economies of scale

 Hilley, *Cloud Computing: A Taxonomy of Platform and Infrastructure Level Offerings*, pg. 4.

* minimizing waste by scaling capacity with demand (or by scheduling tasks to maintain near-peak performance)

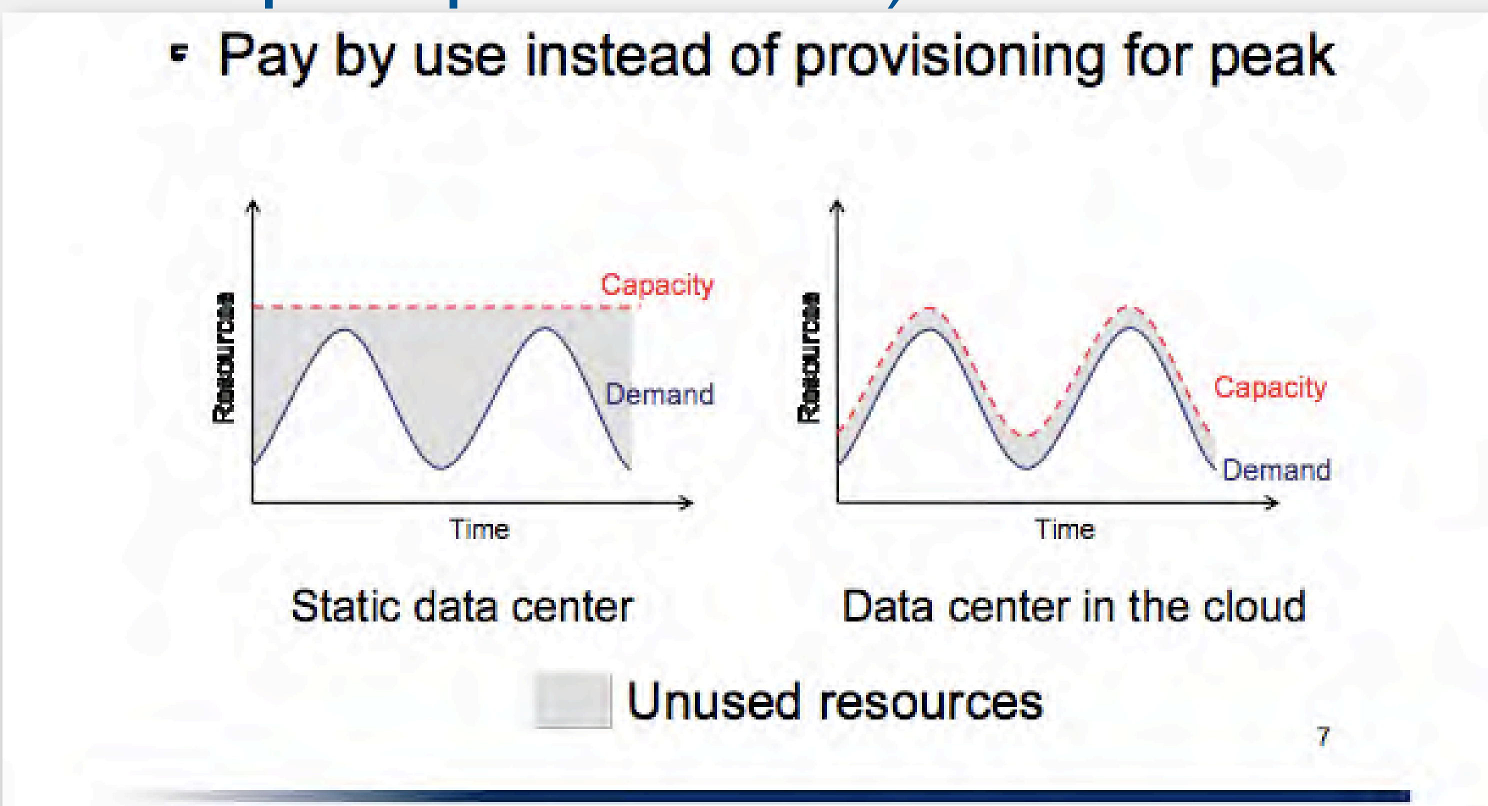


Figure 1: RADLab Presentation: Cloud Value Proposition
From RADLab Presentation "Above the Clouds: A Berkeley View of Cloud Computing" [10, 21]

 Hilley, *Cloud Computing: A Taxonomy of Platform and Infrastructure Level Offerings*, pg. 4.

(contentious) cloud service categories

Figure 2: A Hierarchy of Cloud Offerings
From "Cloud Computing with Linux" [34].

 Hilley, *Cloud Computing: A Taxonomy of Platform and Infrastructure Level Offerings*, pg. 7.

different levels of abstraction/virtualization

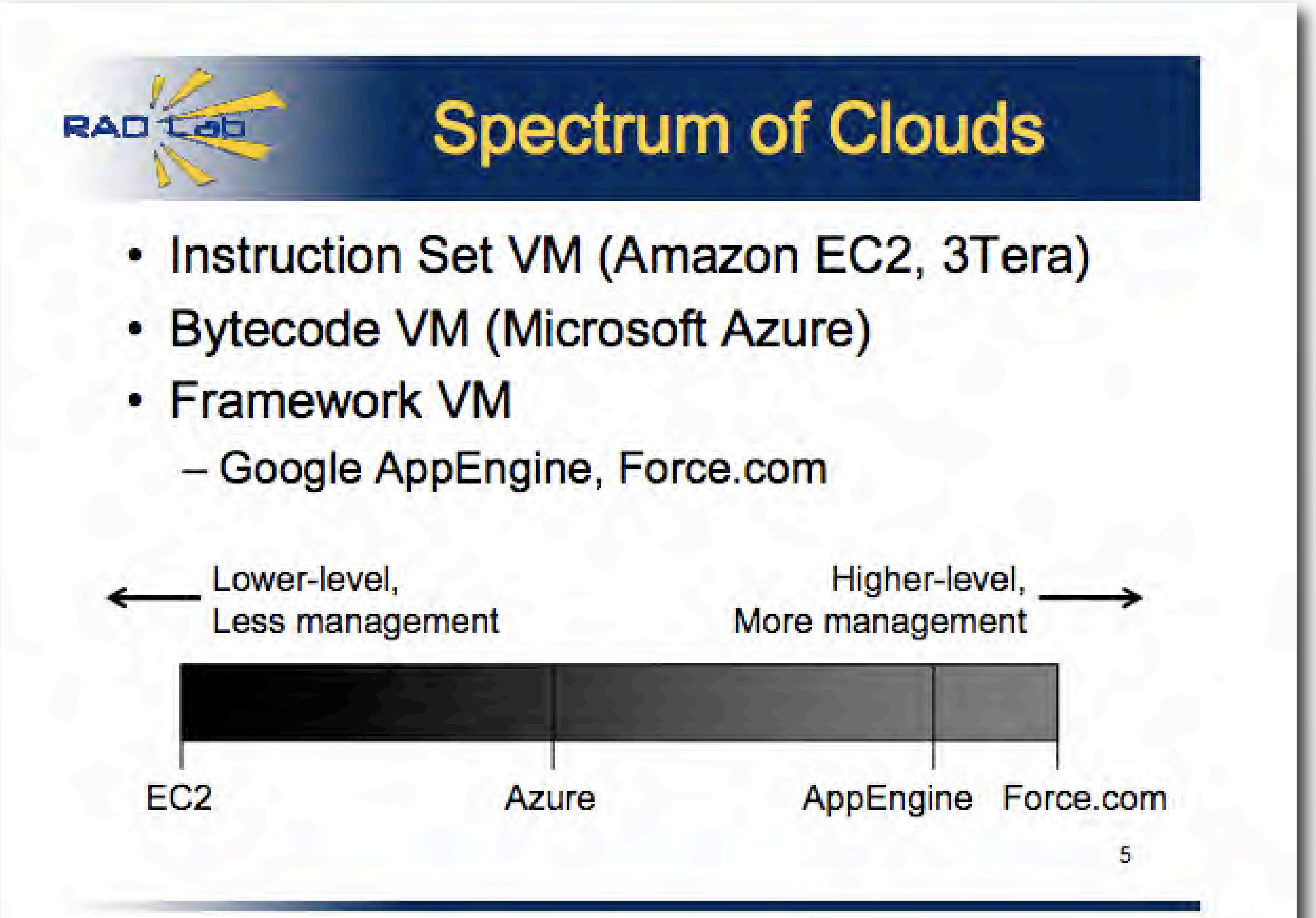


Figure 6: RADLab Presentation: Levels of Cloud Abstraction or Virtualization
rom RADLab Presentation "Above the Clouds: A Berkeley View of Cloud Computing" [10, 21]

PD-INEL Hilley, *Cloud Computing: A Taxonomy of Platform and Infrastructure Level Offerings*, pg. 19.

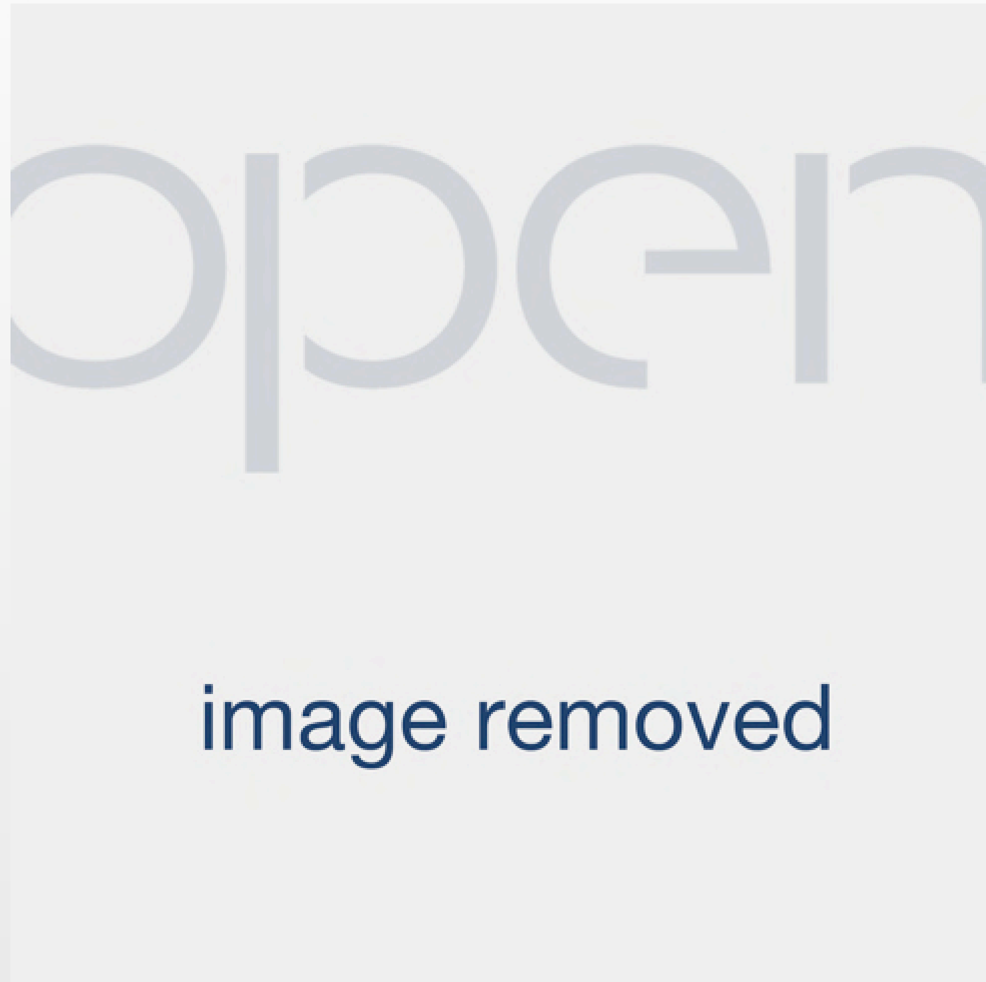
$$ must change hands in the cloud

Storage	
$0.150 per GB	first 50 TB / month of storage used
$0.140 per GB	next 50 TB / month of storage used
$0.130 per GB	next 400 TB /month of storage used
$0.120 per GB	storage used / month over 500 TB
Data Transfer In	
$0.100 per GB	all data transfer in
Data Transfer Out	
$0.170 per GB	first 10 TB / month data transfer out
$0.130 per GB	next 40 TB / month data transfer out
$0.110 per GB	next 100 TB / month data transfer out
$0.100 per GB	data transfer out / month over 150 TB
Requests	
$0.01 per 1,000	PUT, COPY, POST, or LIST requests
$0.01 per 10,000	GET and all other requests (except DELETE)

 Hilley, *Cloud Computing: A Taxonomy of Platform and Infrastructure Level Offerings,* pg. 27.

Hilley 2009, p. 27

the outlook on clouds? cloudy...



Conclusion section on cloud computing removed. For more information, please go to Hilley, *Cloud Computing: A Taxonomy of Platform and Infrastructure Level Offerings,* pg. 30.

Additional Source Information

for more information see: http://open.umich.edu/wiki/CitationPolicy

Slide 3: Please see original article on the "New Era in IT" and cloud computing at http://www.vmware.com/files/pdf/cloud/VMware-and-Cloud-Computing-BR-EN.pdf.

Slide 5: Barry Wilkinson, *Grid Computing: Techniques and Applications*, CRC Press, 2009, pg. 10.

Slide 6: Foster, I. and Kesselman, C. Globus: A metacomputing infrastructure toolkit. The International Journal of Supercomputer Applications and High Performance Computing, 11:2, 1997, 115-128.

Slide 7: Ian Foster and Carl Kesselman, The Grid 2, Second Edition: Blueprint for a New Computing Infrastructure, Morgan Kaufmann Publishing 2003, p.44.

Slide 8: Ian Foster and Carl Kesselman, The Grid 2, Second Edition: Blueprint for a New Computing Infrastructure, Morgan Kaufmann Publishing 2003, p.46.

Slide 9, Image 1 (top): http://www-csag.ucsd.edu/individual/achien/achien.html
Slide 9, Image 2 (bottom): Andrew Chien, *Grids and High Performance Distributed Computing,* http://www-csag.ucsd.edu/teaching/cse225s04/Lectures/Lec2-Globus-Grid-Architecture.pdf

Slide 10: Barry Wilkinson, *Grid Computing: Techniques and Applications*, CRC Press, 2009, pg. 39.

Slide 11: Barry Wilkinson, *Grid Computing: Techniques and Applications*, CRC Press, 2009, pg. 44.

Slide 12: Barry Wilkinson, *Grid Computing: Techniques and Applications*, CRC Press, 2009, pg. 57.

Slide 13: Barry Wilkinson, *Grid Computing: Techniques and Applications*, CRC Press, 2009, pg. 152.

Slide 14: Barry Wilkinson, *Grid Computing: Techniques and Applications*, CRC Press, 2009, pg. 160.

Slide 15: Programming instructions on grid computing removed. For more information, go to Wilkinson, *Grid Computing: Techniques and Applications, CRC Press, 2009,* pg. 62.

Slide 16: Hilley, David, Cloud Computing: A Taxonomy of Platform and Infrastructure-Level Offerings, 2009.

Slide 17,(both images): Hilley, David, Cloud Computing: A Taxonomy of Platform and Infrastructure-Level Offerings, 2009, pg. 4.

Slide 18: Hilley, David, Cloud Computing: A Taxonomy of Platform and Infrastructure-Level Offerings, 2009, pg. 7.

Slide 19: Hilley, David, Cloud Computing: A Taxonomy of Platform and Infrastructure-Level Offerings, 2009, pg. 19.

Slide 20: Hilley, David, Cloud Computing: A Taxonomy of Platform and Infrastructure-Level Offerings, 2009, pg. 27.

Slide 21: Hilley, David, Cloud Computing: A Taxonomy of Platform and Infrastructure-Level Offerings, 2009, pg. 30.