Overcoming XenoPhobia: Virtualization, Workspaces, and Everything

Kate Keahey

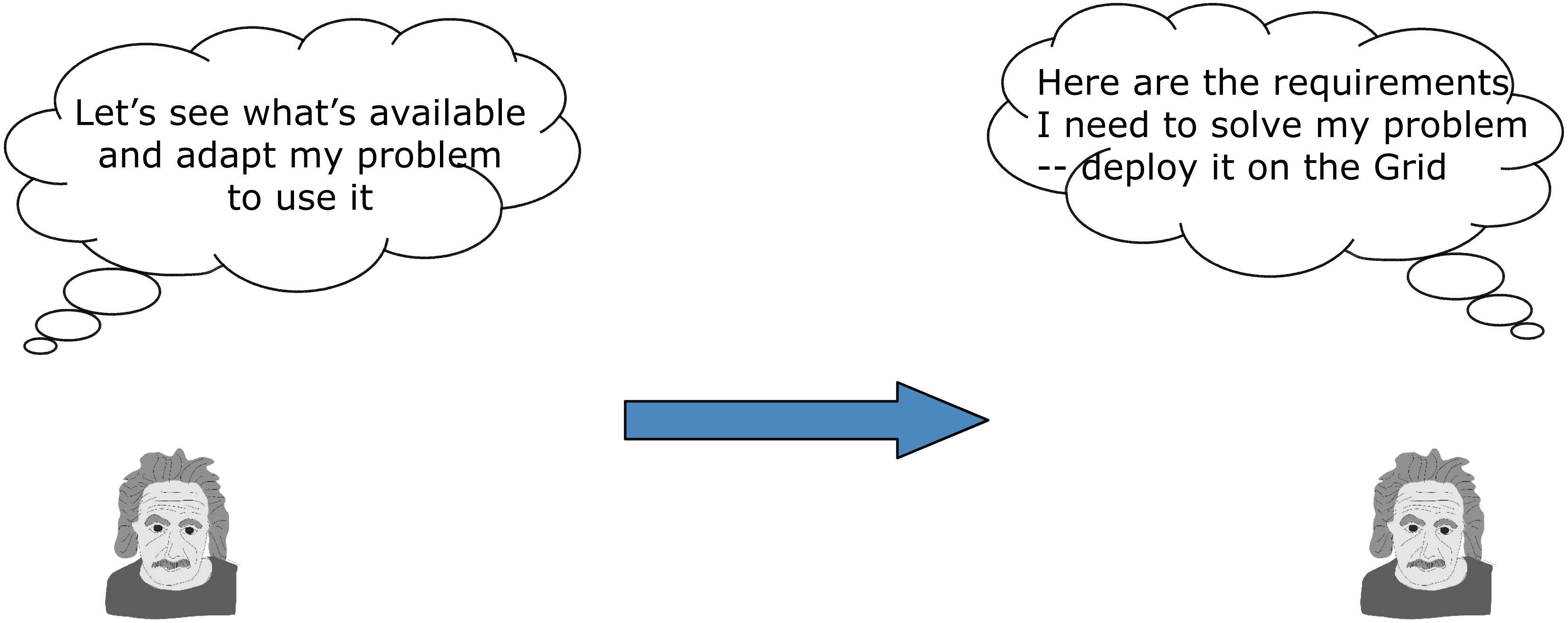
keahey@mcs.anl.gov

Argonne National Laboratory

Overview

- Virtualization: changing the question
- Scenarios
- Middleware: the Workspace Service
- Workspace Applications
- Deployment Issues
- Overcoming XenoPhobia

Changing the Question: Users

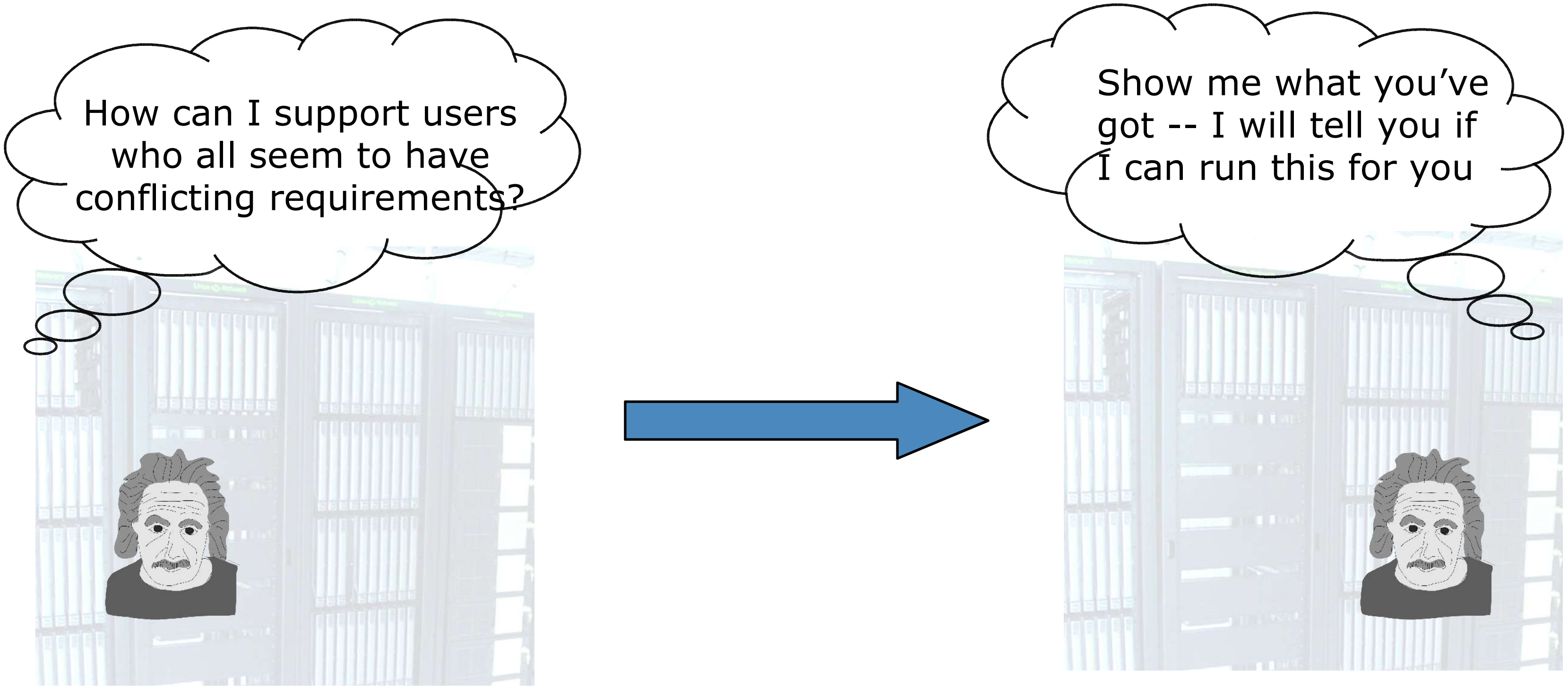


Let's see what's available and adapt my problem to use it

Here are the requirements I need to solve my problem -- deploy it on the Grid

Requirements can be defined in terms of environment or resource allocation

Changing the Question: Providers



How can I support users who all seem to have conflicting requirements?

Show me what you've got -- I will tell you if I can run this for you

Based on policies and image properties only very few images may actually be run

New Trade-offs New Middleware

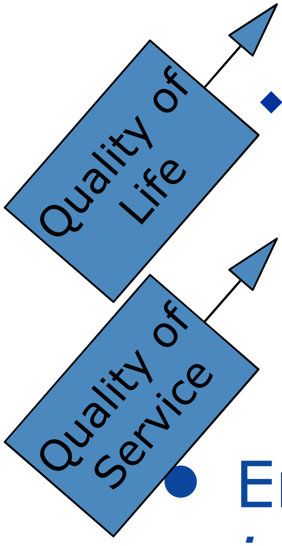
- What have VMs changed?
  - The idea of a virtual machine goes way back
  - Are VMs like jobs?
    - Significant security and resource management differences
  - Cost-effectiveness
    - We can now do new things, not because a need was suddenly discovered, but because they became cost-effective
- Newly (more) cost-effective
  - Grid and opportunistic computing
  - Short-term and dynamic leases
  - Strict sharing models

Example Scenarios

- Leasing (advance reservations)
  - Short-term leasing
  - Applications: a class, an experiment, a longer-term resource loan
- Opportunistic computing
  - A VM pops up and registers itself and sends notification of its state to a resource management module
- Demand-based service management
  - Acquire new resources based on need

Virtual Workspaces

- Focus on execution environments
- Two aspects of workspaces:
  - *Environment definition*: We get exactly the (software) environment me need on demand.
  - *Resource allocation*: Provision and guarantee all the resources the workspace needs to function correctly (CPU, memory, disk, bandwidth, availability), allowing for dynamic renegotiation to reflect changing requirements and conditions.



- Environment and resource allocation are now *independent*

VW Implementations

- Configuring physical machines
  - Slow and invasive
  - Environments are hard to describe
  - Limited/none enforcement options
  - Using environment management tools
- Virtual Machines
  - Fast to deploy, much less invasive
  - Environments are easy to describe
  - Bonus: isolation, serialize, redeploy, migrate

GT4 Workspace Service

- The GT4 Virtual Workspace Service (VWS) allows an authorized client to deploy and manage workspaces on-demand.
  - Started out in 2003 with an investigation of different VMs including Vmware, Vserver, later Xen
  - GT4 WSRF front-end
  - Leverages GT core and services
    - Notifications, security, etc.
    - Very solid, well-tested implementation
  - Implements multiple deployment modes
    - Best-effort, leasing, etc.
  - Currently implements workspaces as VMs
    - Uses the Xen VMM but others could also be used
  - Current release 1.2.1 (January '07)
  - http://workspace.globus.org

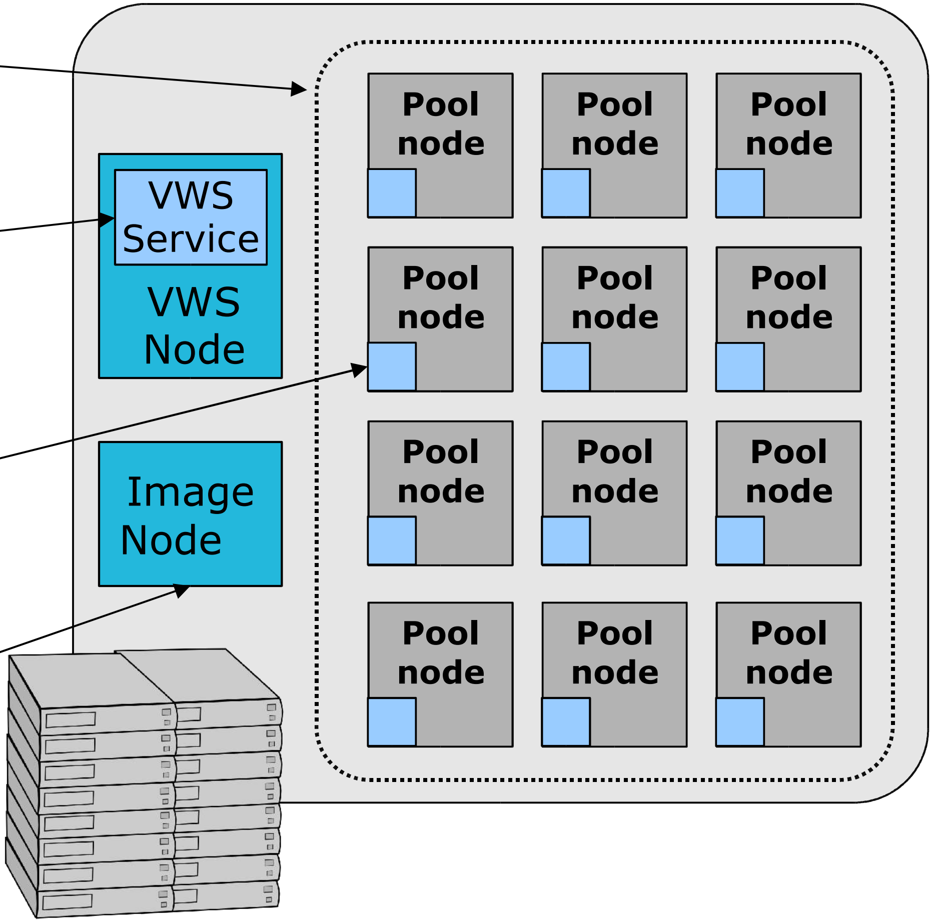
Workspace Service Backstage

The VWS manages a set of nodes inside the TCB (typically a cluster). This is called the *node pool*.

The workspace service has a WSRF frontend that allows users to deploy and manage virtual workspaces

Each node must have a VMM (Xen) installed, along with the *workspace backend* (software that manages individual nodes)

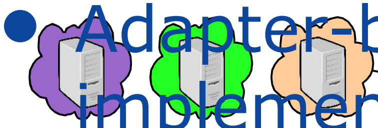
VM images are staged to a designated image node inside the TCB



Trusted Computing Base (TCB)

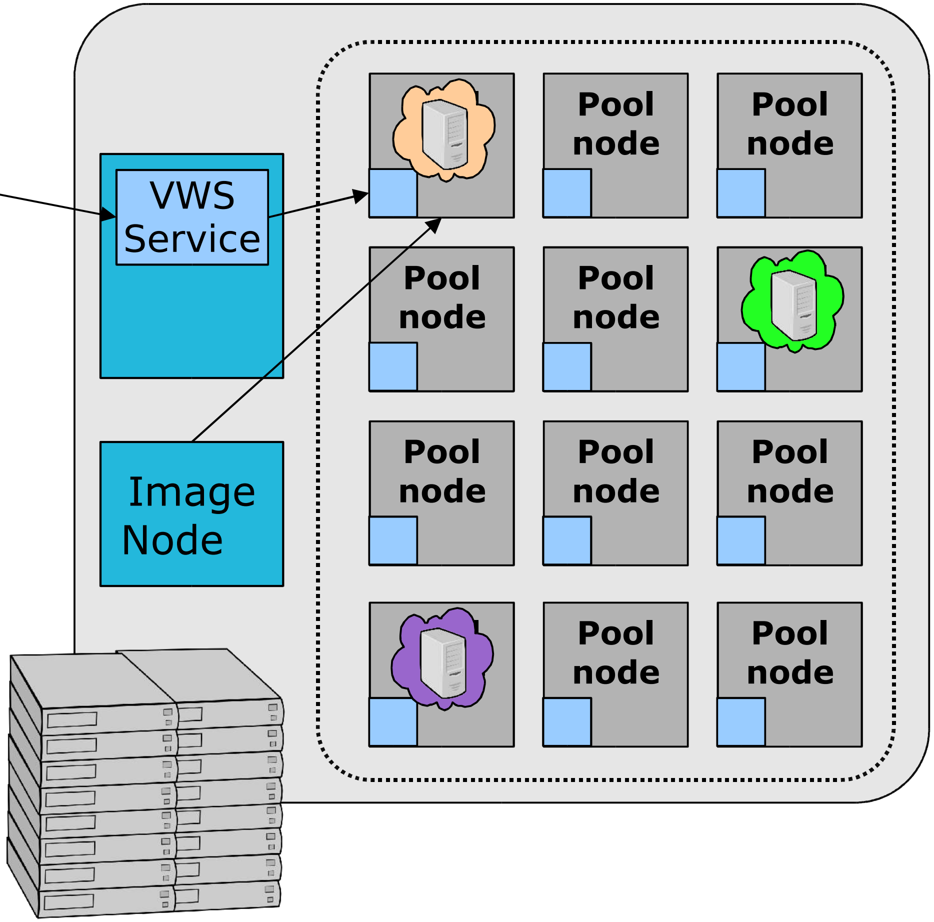
Deploying Workspaces

- Adapter-based implementation model
  - Transport adapters
    - Default scp, then gridftp
  - Control adapters
    - Default: ssh
    - Deprecated: PBS, SLURM
  - VW deployment adapter
    - Xen
    - Previous versions: VMware



Workspace

- Workspace metadata (with image location)
- Deployment Request



Workspace Request Arguments

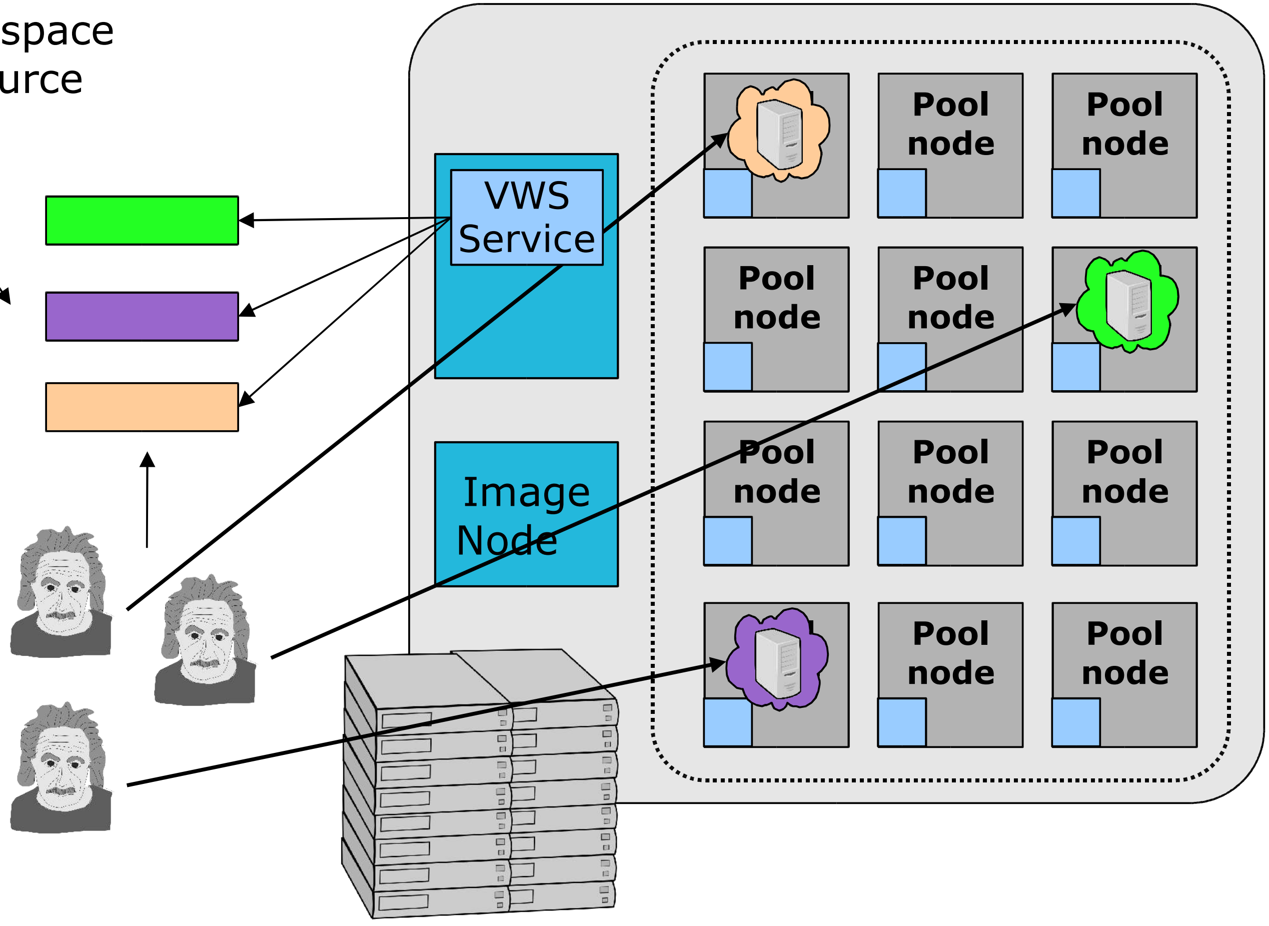
- A workspace, composed of:
  - VM image
  - Workspace metadata
    - XML document
    - Includes deployment-independent information:
      - VMM, kernel, and any other requirements
      - NICs + IP configuratoin
      - VM image location
  - Need not change between deployments
- Resource allocation
  - Specifies availability, memory, CPU%, disk
  - Changes during or between deployments

Interacting With Workspaces

The workspace service publishes information on each workspace as standard WSRF Resource Properties.

Users can query those properties to find out information about their workspace (e.g. what IP the workspace was bound to) or subscribe to notifications on changes

Users can interact directly with their workspaces the same way the would with a physical machine.



Trusted Computing Base (TCB)

A Word about VM Schedulers…

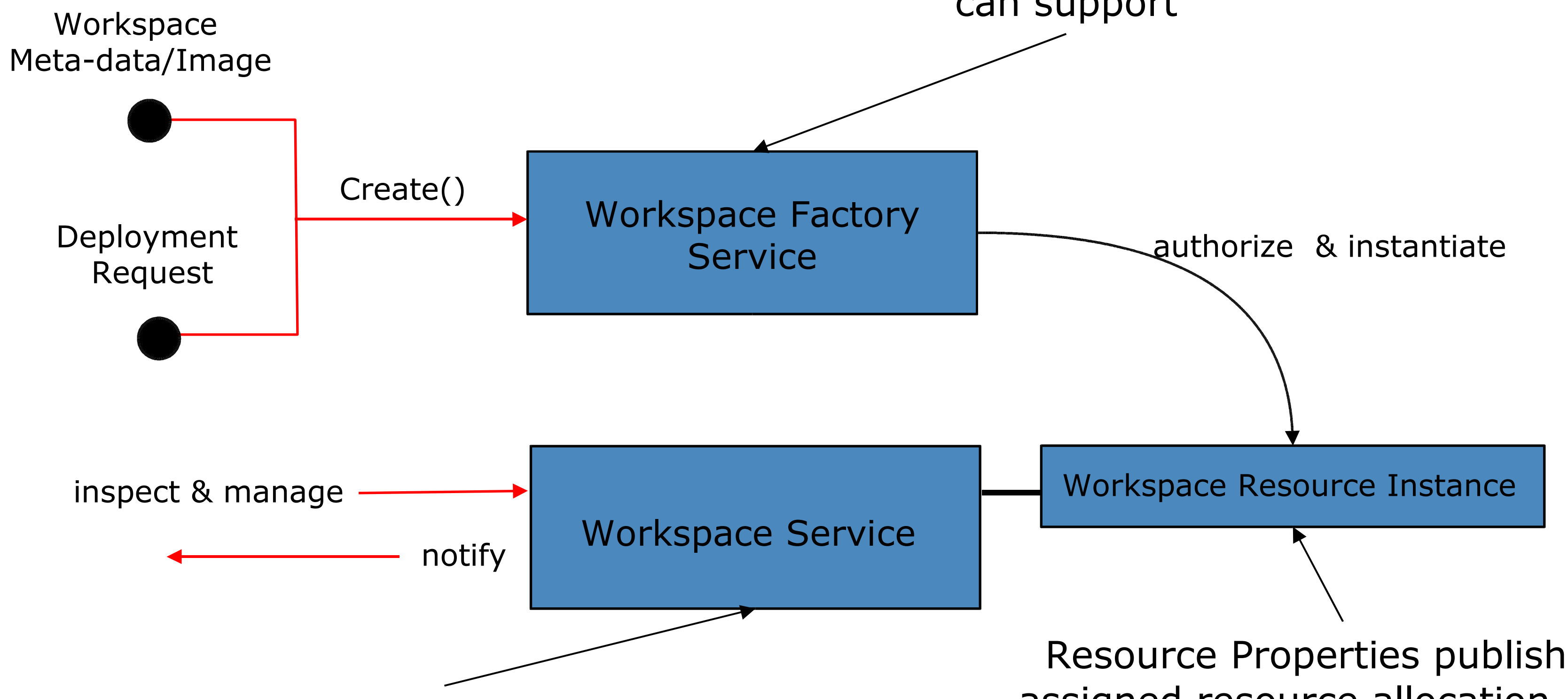
- University of Marburg work
  - A new backfilling technique
  - Paper in the VTDC06 workshop
- Workspace scheduler
  - Workspaces are scheduler-independent
  - PBS, SLURM, Margathea all possible options
  - The released Workspace scheduler version is very basic
  - The research version
    - Integrating leasing and best-effort/batch VM deployment
    - User-oriented resource model
      - Managing deployment and run-time overhead
    - Fine-grained management
    - Paper in VTDC '06 workshop: http://workspace.globus.org/vtdc06
    - Extensions to SGE and Torque

A Word about Virtual Networks

- "Logging into the Grid" metaphor
  - grid-proxy-init
  - Also logs you into a private network
- Multiple efforts in this area
  - ViNE
    - University of Florida, J. Fortes & M. Tsugawa
  - VNET, vnet, VIOLIN
- Combine with network performance overlays

Workspace Service Interfaces

Handles creation of workspaces. Also publishes information on what types of workspaces it can support



Handles management of each created workspace (start, stop, pause, migrate, inspecting VW state, ...)

Resource Properties publish the assigned resource allocation, how VW was bound to metadata (e.g. IP address), duration, and state

Status

- Latest Release: 1.2.1
  - Better IP handling
  - Emphasis on reliability: test harness and documentation
- To be included in the next VDT release
- VW is an incubator project in dev.globus
  - New governance model for Globus Toolkit
  - http://dev.globus.org
  - All software released under Apache license 2.0
  - Support via mailing lists

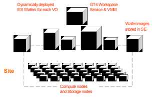
Team

- Workspace team
  - Kate Keahey
  - Tim Freeman
  - Borja Sotomayor
- With guest appearances by:
  - Ian Foster, Frank Siebenlist, Elizeu Santos-Neto
  - Others: Karl Doering, Xuehai Zhang

Support

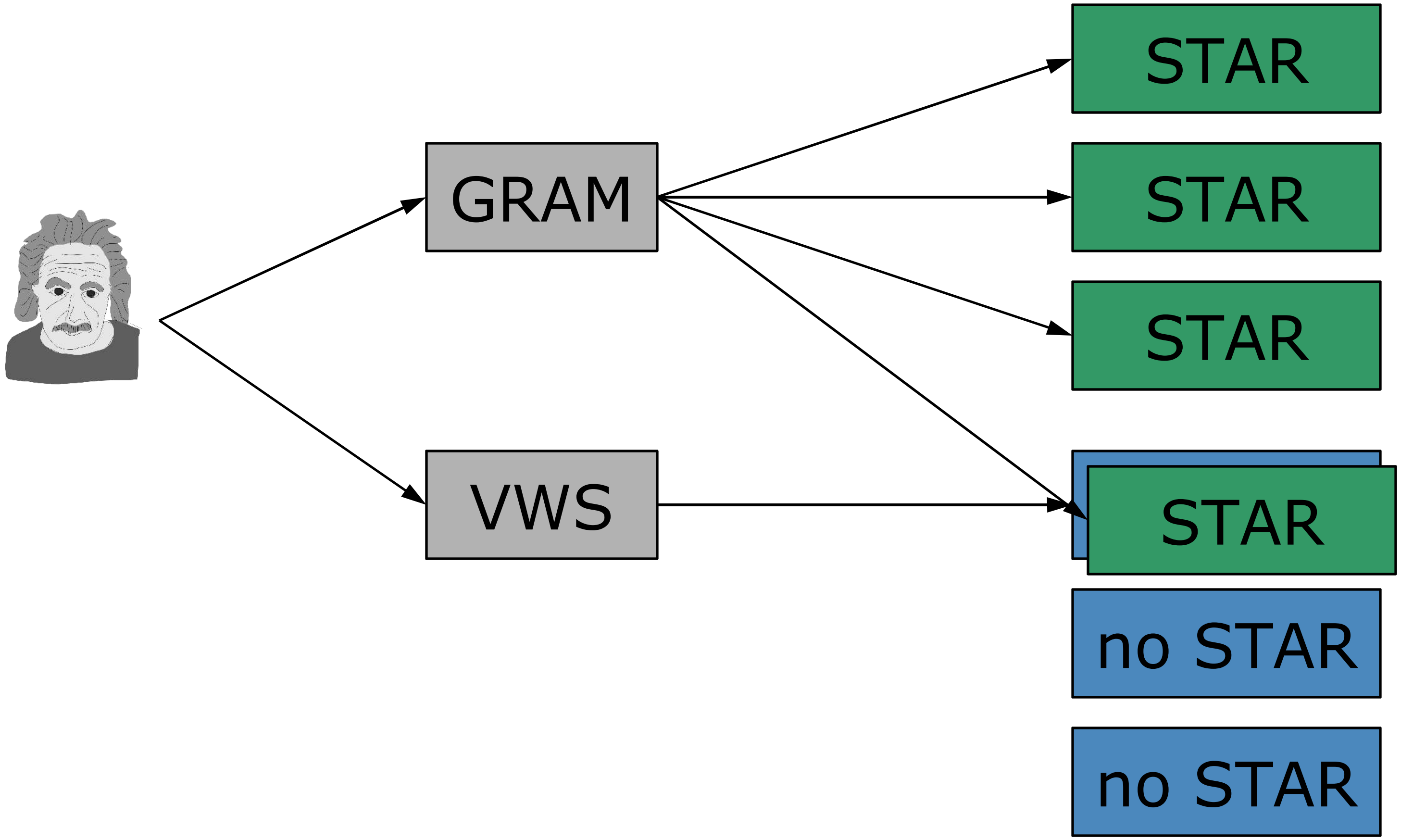
And that's what we do to bugs!

Applications: ESF



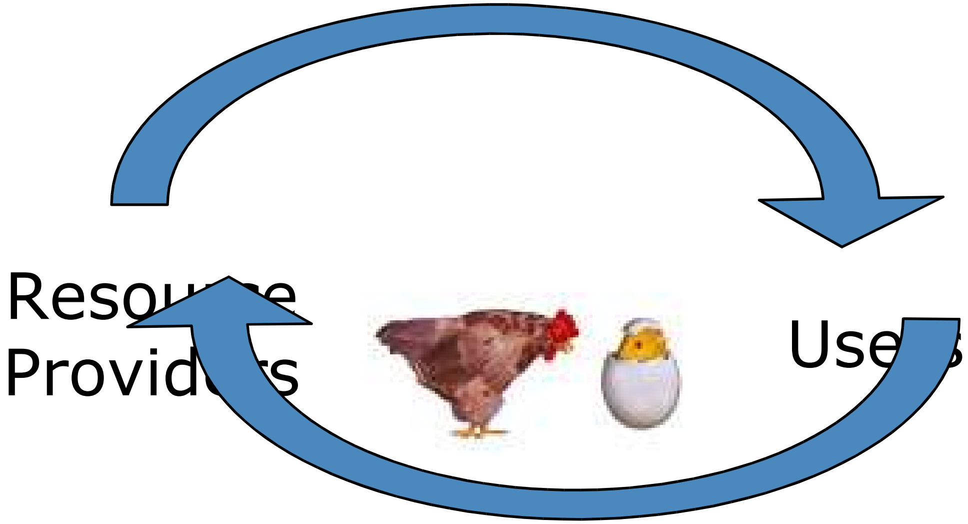
Deployment: OSG SDSC sites
Services: D-cache, Frontier
www.opensciencegrid.org/esf

applications: STAR



Provisioning STAR nodes on TeraPort (UC): demonstrated at SC06 show floor

Current Issues: a chicken and egg problem



The Chicken

- Overcoming Xenophobia
  - VMMs are “invasive”
  - Security: the cure or the disease?
    - On the whole the cure, but it is a new tool
  - Will it scale?
    - This is not a question that a simulation could answer!
  - More work is needed in this area
- Commercial deployments are moving faster
  - Hosting services, Amazon’s EC2, others…
  - There are more incentives
- Pioneering is hard!

The Egg

- Suppose you have this infrastructure deployed, now what?
  - Where would be iTunes without music?
- A library of VM images…
  - Labor intensive
  - Images "age"
  - Attestation information
- "Assembly line" approach
  - rPath: scientific appliances and rBuilder
    - Appliance = application + its environment
  - BCFG2: configuration management tool
    - Producing and managing images
  - Deployment-time configuration
    - Configuring a virtual cluster -- integrating late information
- How do we describe, indentify, and query to find the right image?

Overcoming XenoPhobia

- Let’s share experiences
  - Share images, experiences, problems, ideas, technology
- A Virtualization Forum
  - Case studies/Success stories
  - Virtual Grid resources
  - Share images
  - Forum Q&A
  - Technology forum: share technologies
- If you are interested let me know:
  - keahey@mcs.anl.gov

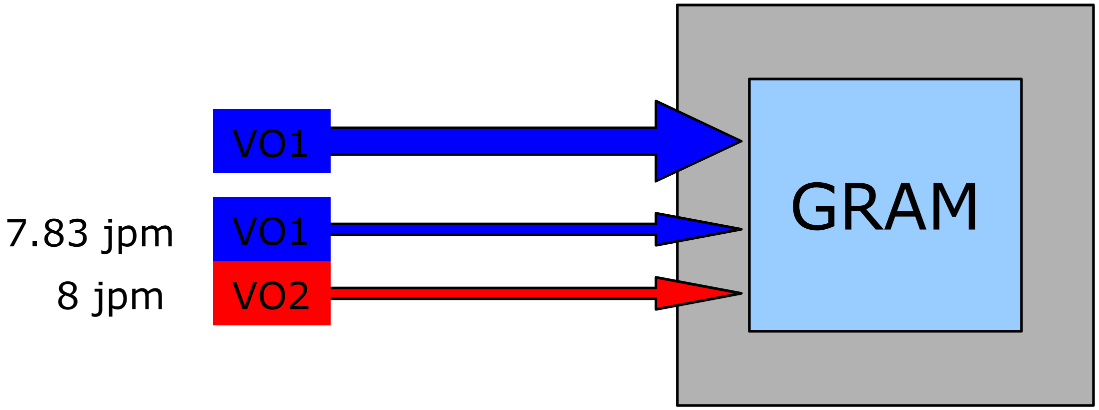
Conclusions

- Virtualization adoption constitutes a significant paradigm shift
  - Much potential
  - BUT ALSO MANY CHALLENGES
    - (some of which we still don’t know about)
- There is a “critical mass” to virtualization adoption
  - Starting simple is good
  - Stopping simple is bad
- Many technologies need to come together to make a difference
  - Requires participation of resource provider, users, technology developers

Fine-grained Enforcement

Edge Services Today

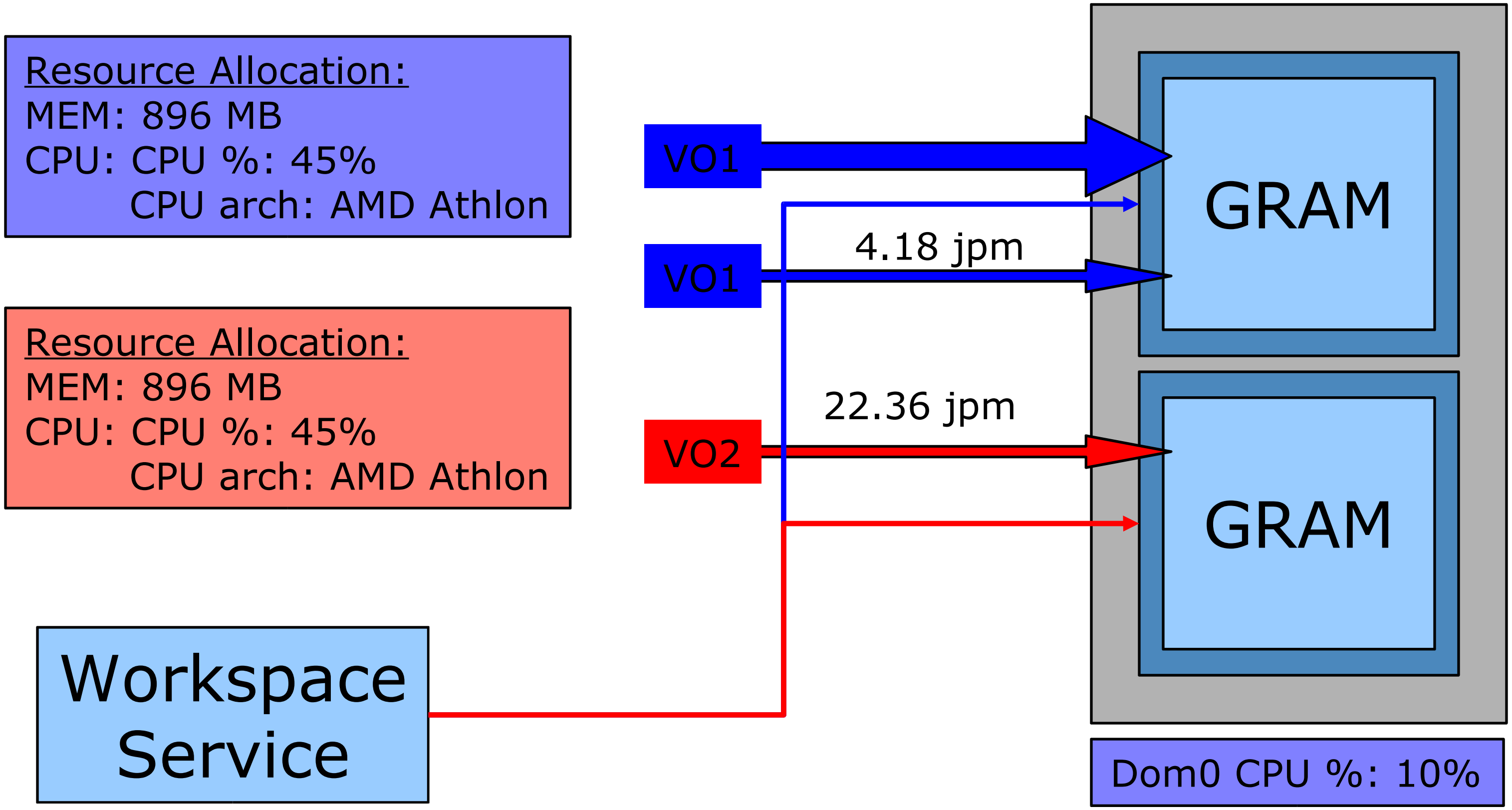
Compute Element (CE) implemented as GT GRAM



Both VOs share the same resource

Job throughput is low as both VOs are equally impacted by the high VO1 traffic

Allocating Resources for Edge Services

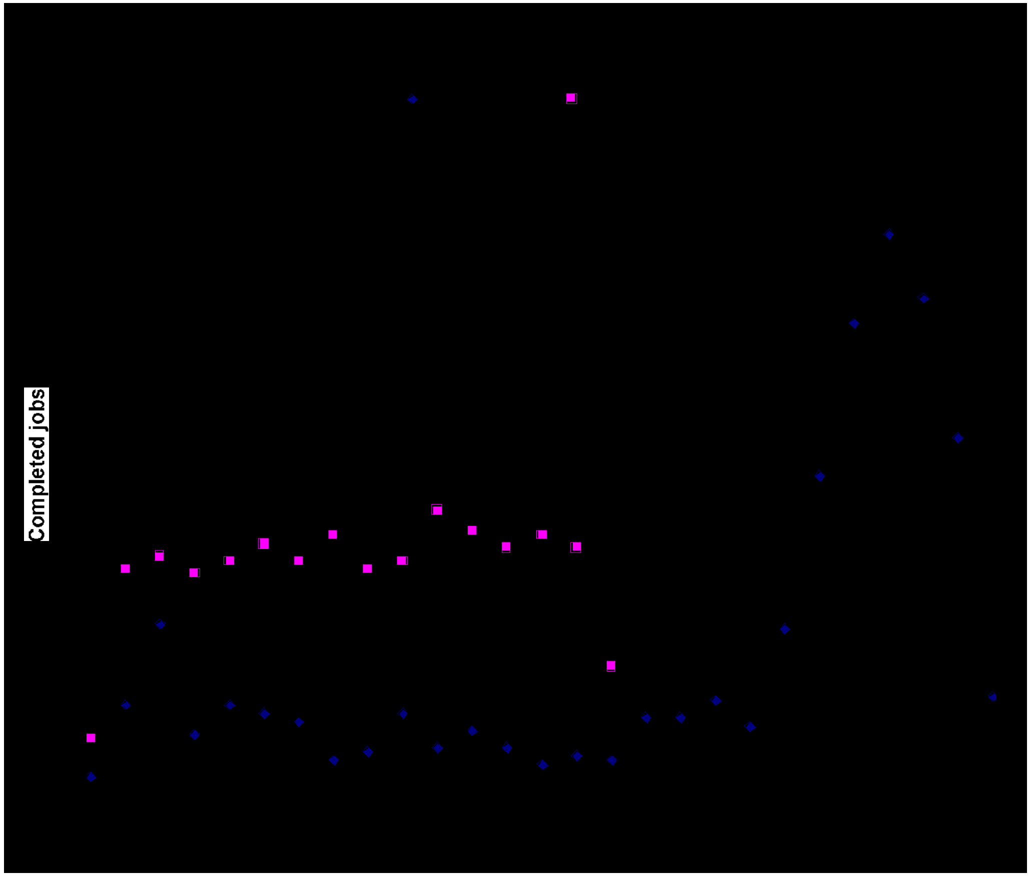


Resource Allocation:
MEM: 896 MB
CPU: CPU %: 45%
CPU arch: AMD Athlon

Resource Allocation:
MEM: 896 MB
CPU: CPU %: 45%
CPU arch: AMD Athlon

Job throughput for VO2 is high as it is unimpacted by the high VO1 traffic

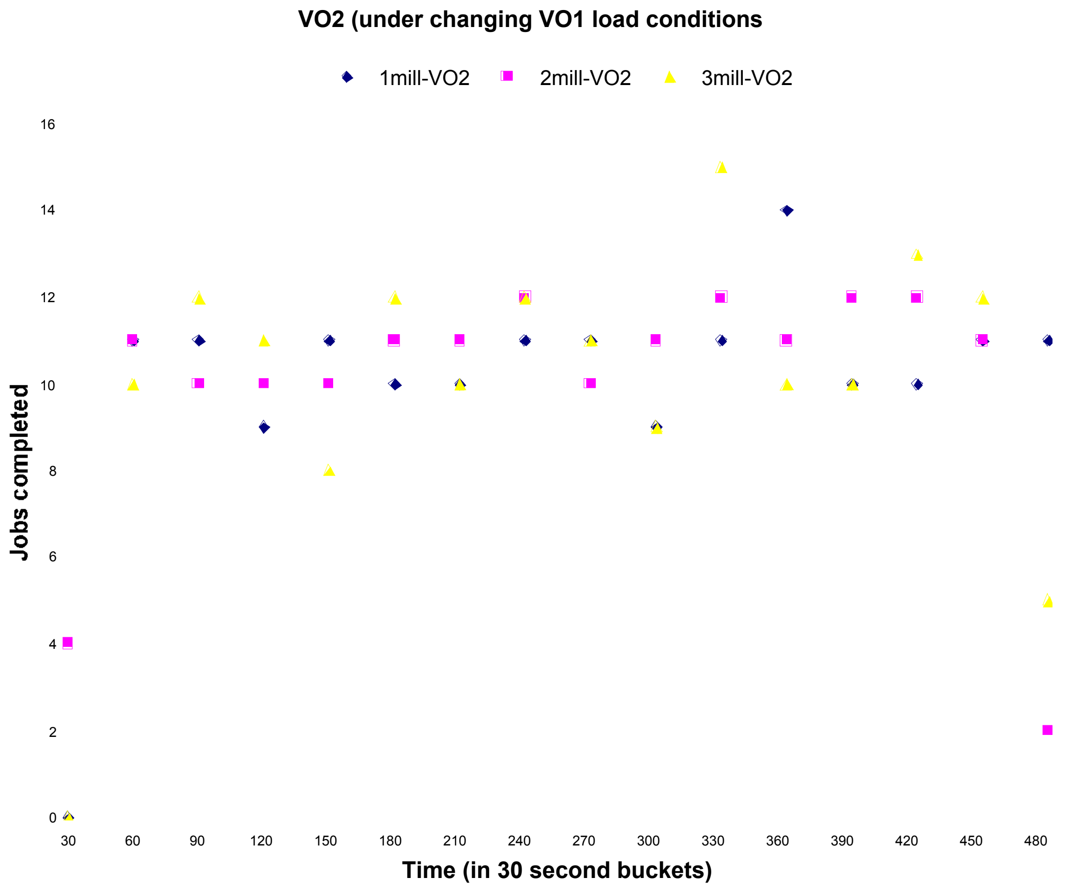
Tracking Requests Overtime



- *Histogram of request throughput*

- *Resource usage is enforced on an "as needed" basis*

Increasing Load on VO1

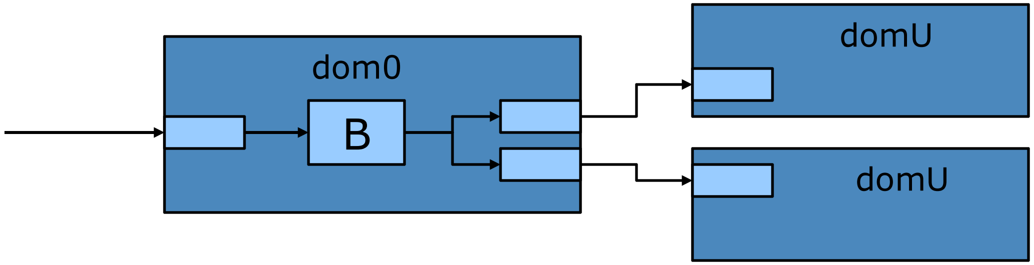


- *Histogram of request throughput*

- *The load on VO1 increases 2x and 3x*

- *Request throughput for VO2 is unimpacted*

Network Resource Allocation

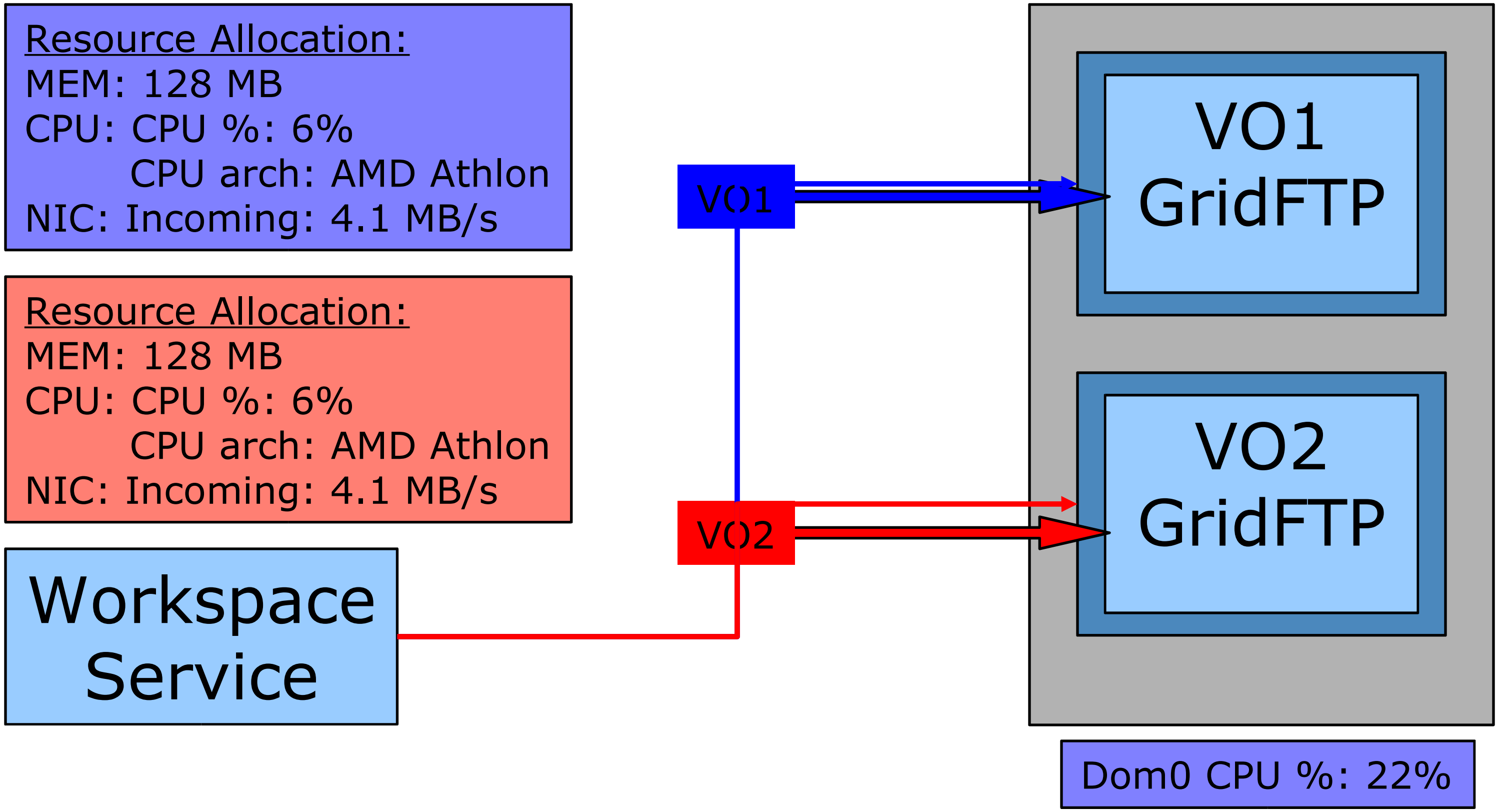


- Processing network traffic requires CPU
- In Xen: for both dom0 and guest domains
  - CPU allocation tradeoffs
  - Scheduling frequency
- The mechanism is general
  - Save for direct drivers

Network Resource Allocation

- Network Allocation Implementation
  - CPU allocations based on a parameter sweep
    - Close to maximum bandwidth
  - Linux network shaping tools
- Negotiating network resource allocations
  - Policy: accepting only CPU allocations that match the bandwidth

Storage Element (SE) Edge Service

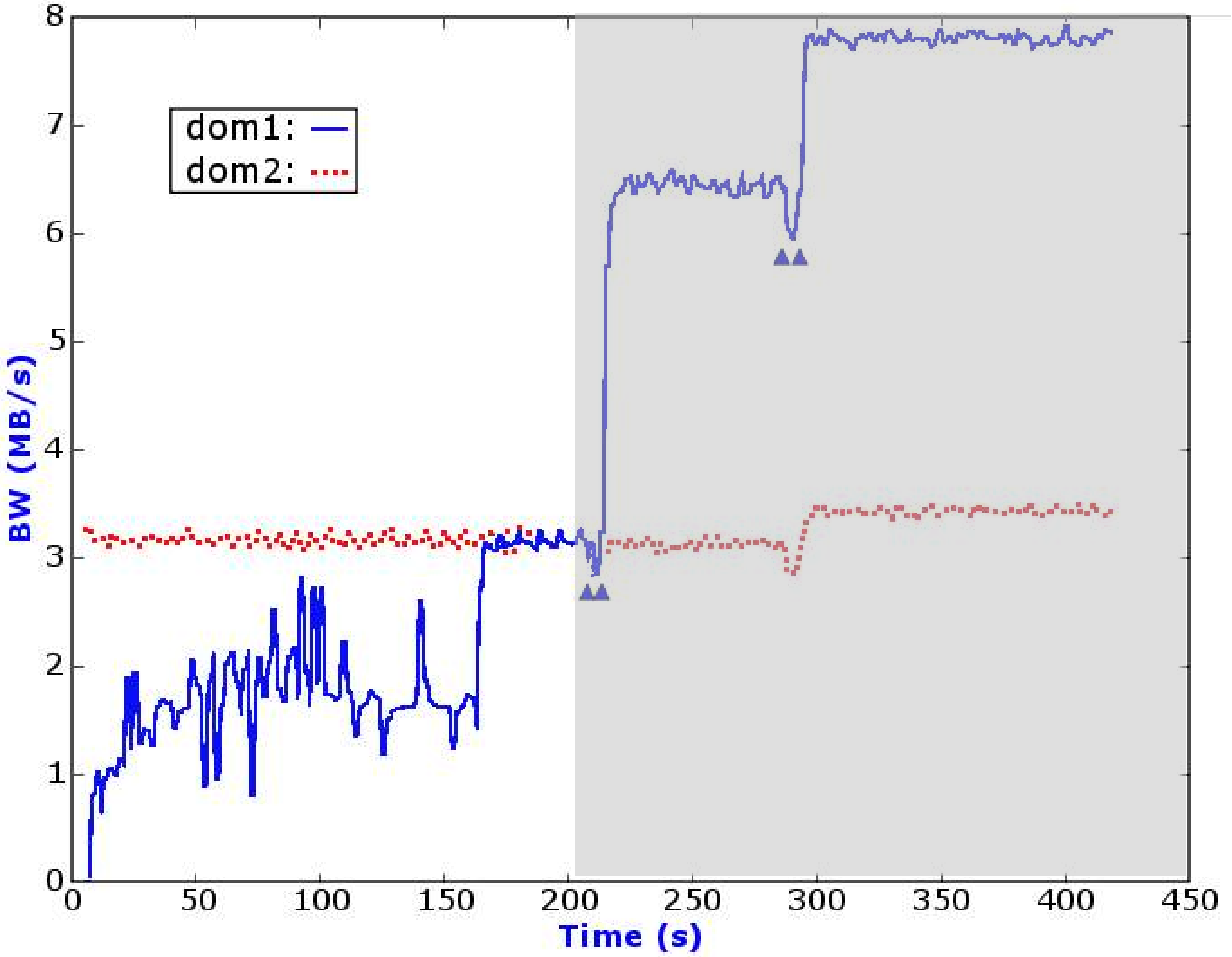


Resource Allocation:
MEM: 128 MB
CPU: CPU %: 6%
CPU arch: AMD Athlon
NIC: Incoming: 4.1 MB/s

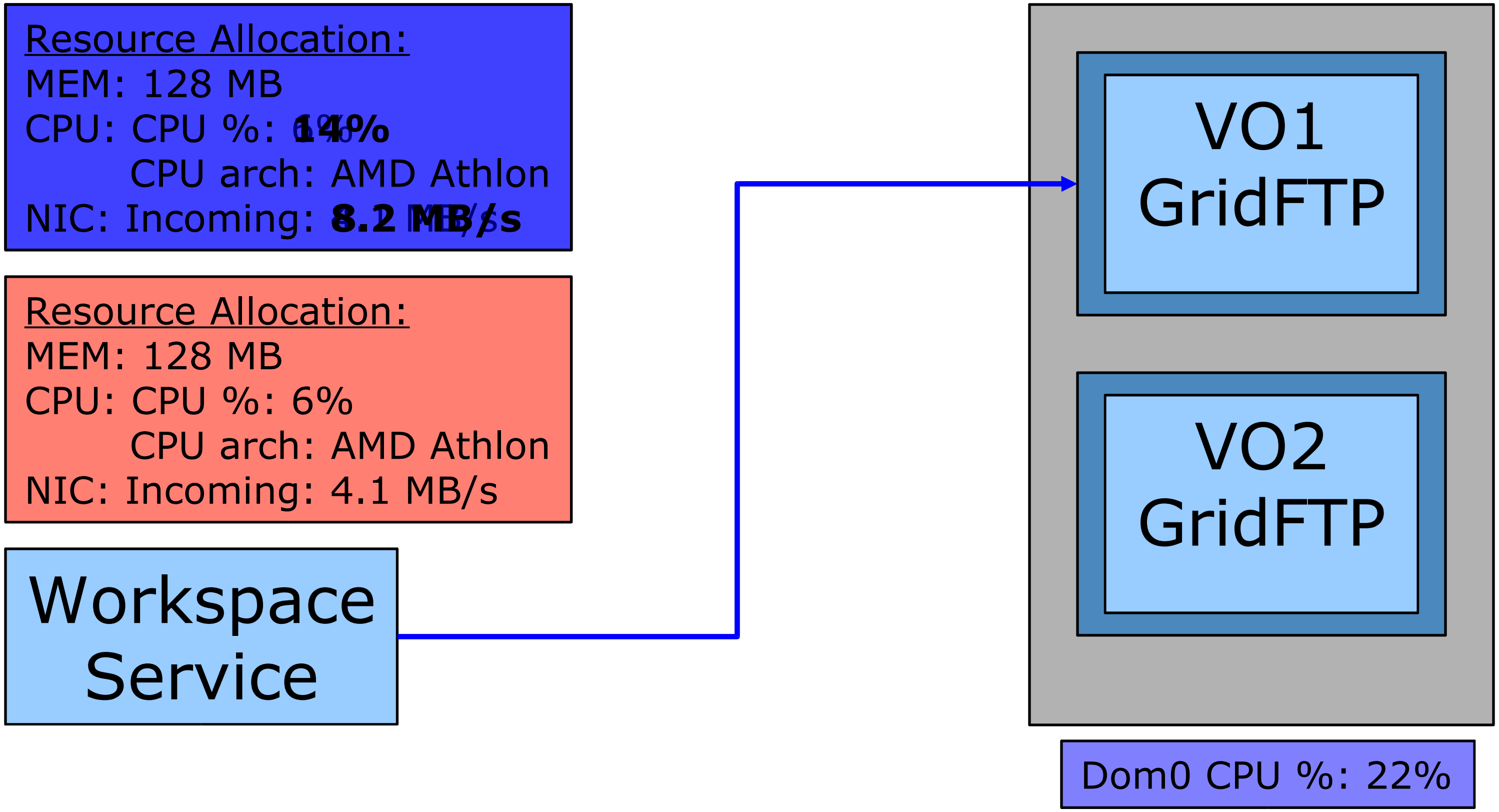
Resource Allocation:
MEM: 128 MB
CPU: CPU %: 6%
CPU arch: AMD Athlon
NIC: Incoming: 4.1 MB/s

Dom0 CPU %: 22%

Negotiating Bandwidth



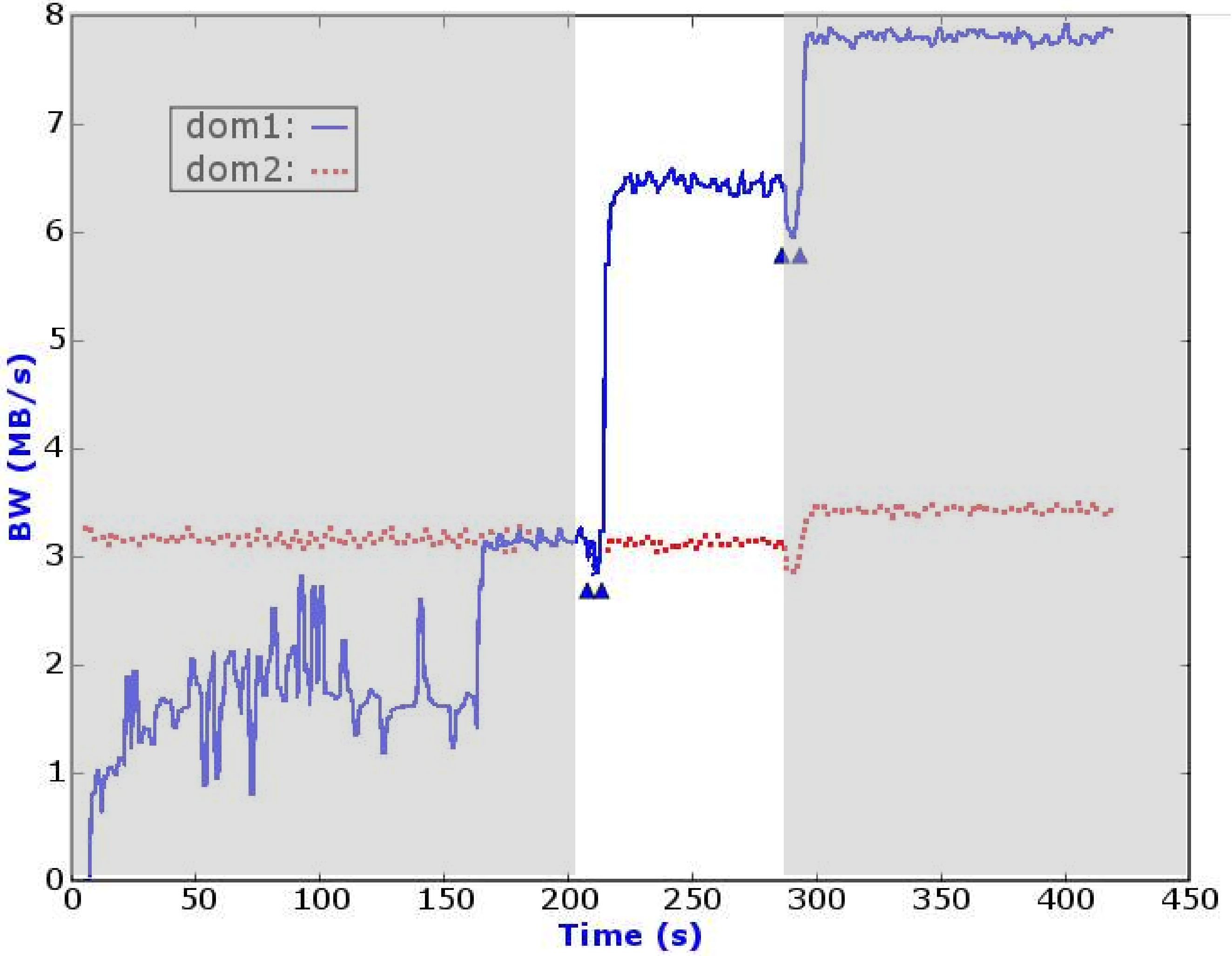
Renegotiating CPU and Bandwidth



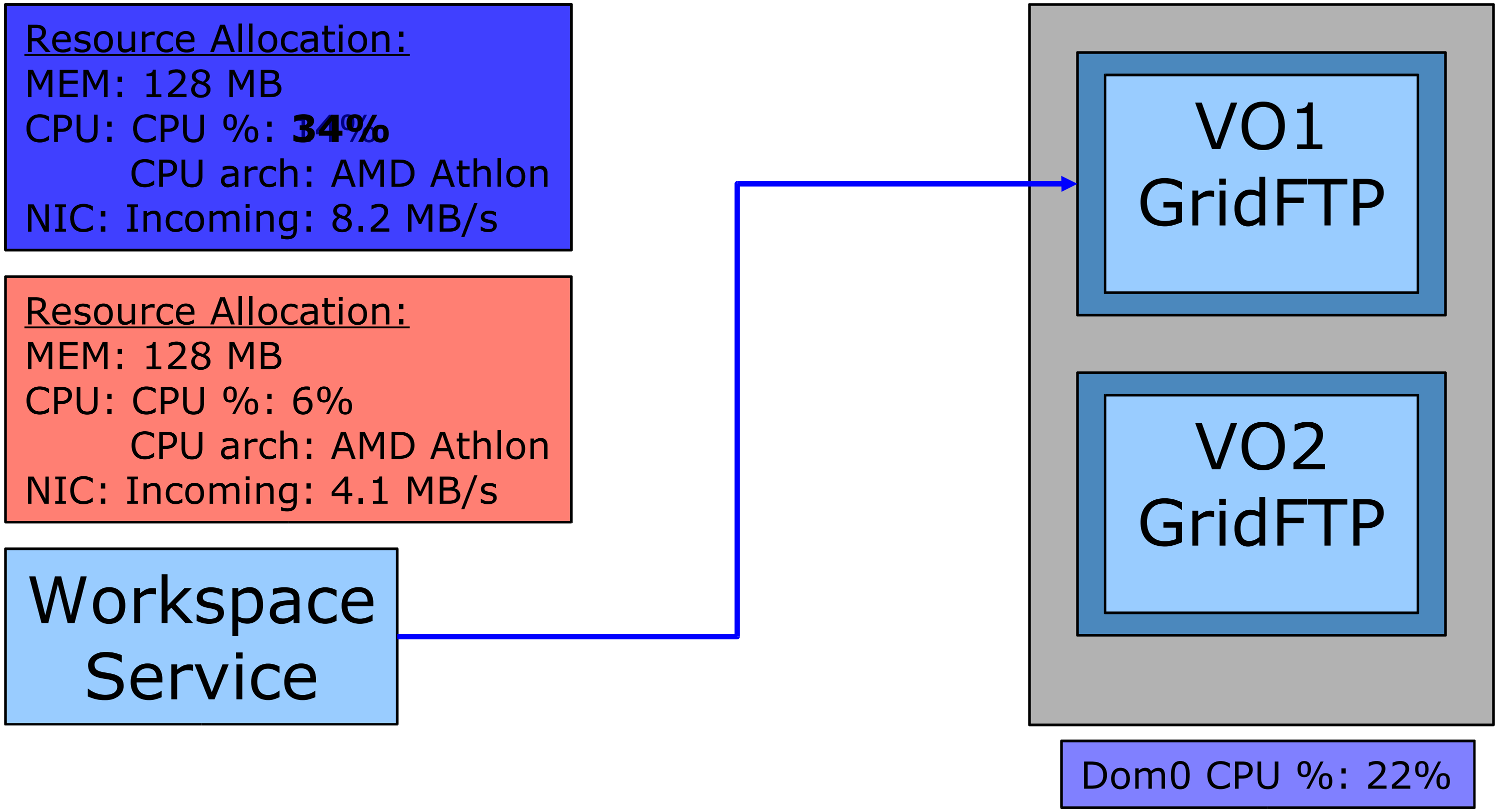
Resource Allocation:
MEM: 128 MB
CPU: CPU %: **14%**
CPU arch: AMD Athlon
NIC: Incoming: **8.2 MB/s**

Resource Allocation:
MEM: 128 MB
CPU: CPU %: 6%
CPU arch: AMD Athlon
NIC: Incoming: 4.1 MB/s

Renegotiating CPU and Bandwidth



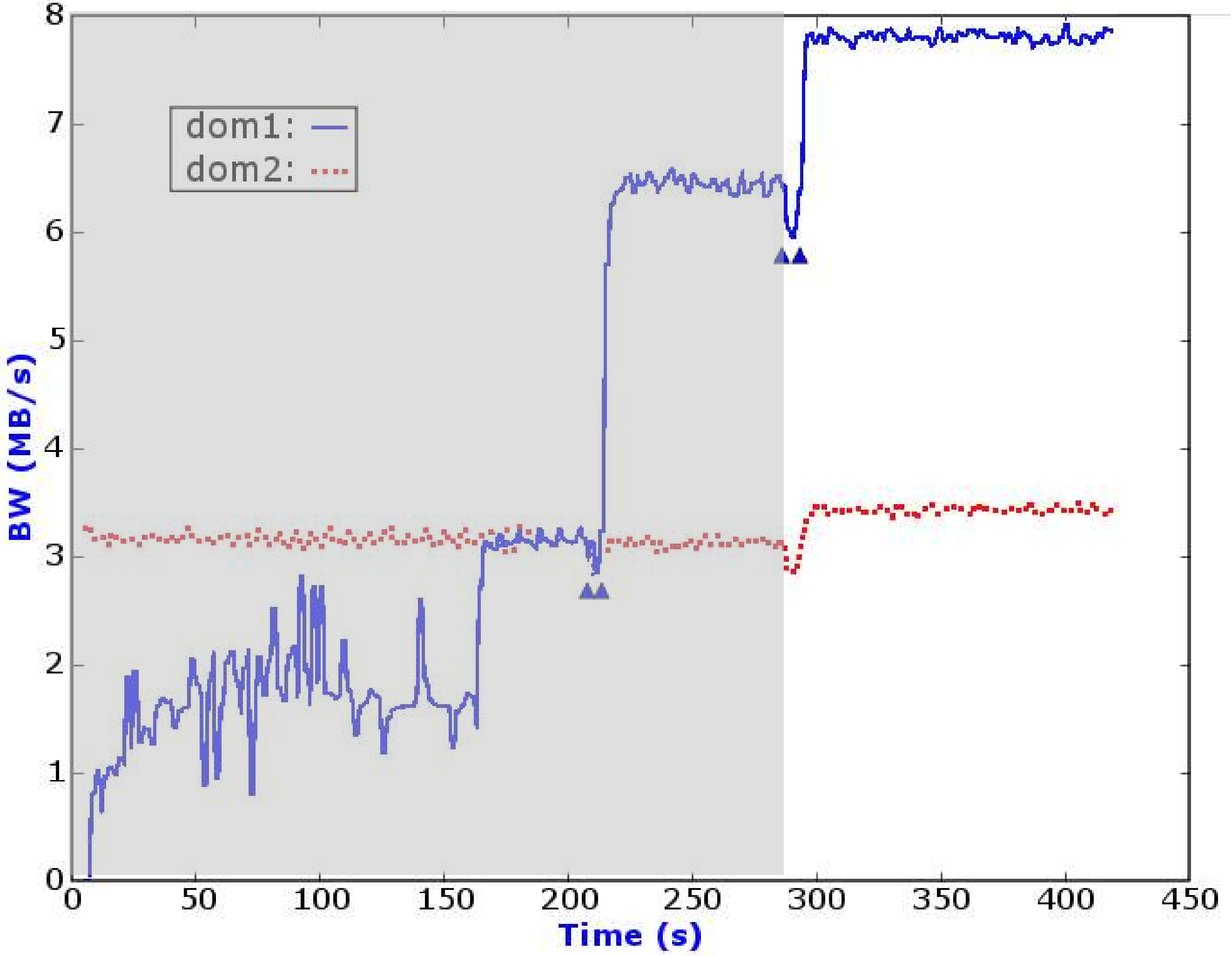
Renegotiating CPU



<u>Resource Allocation:</u>
MEM: 128 MB
CPU: CPU %: **34%**
CPU arch: AMD Athlon
NIC: Incoming: 8.2 MB/s

<u>Resource Allocation:</u>
MEM: 128 MB
CPU: CPU %: 6%
CPU arch: AMD Athlon
NIC: Incoming: 4.1 MB/s

Renegotiating CPU


BW (MB/s)
Time (s)
dom1:
dom2: