Virtualized Worker Nodes in the Grid

Owen Synge

Outline

- Scope.
- Use Case/Requirements
  - Grid.
  - Cloud.
  - Academic Cloud.
- Virtualized Workernodes.
- Limitations.
- Tricky Issues.
- Outlook.
- Lack of Progress.
- Summary

Scope

- Focussing on Glite/LHC Grid infrastructure
  - We know this Grid
    - Can also push for features/changes.
  - Large deployment globally.
- Focusing on Amazon like clouds.
  - Based upon Virtual Machine based
    - rather than Google like API and python based clouds.
  - More HEP users.
- HEP Bias
  - High Throughput Computing/Trivial parallelism.

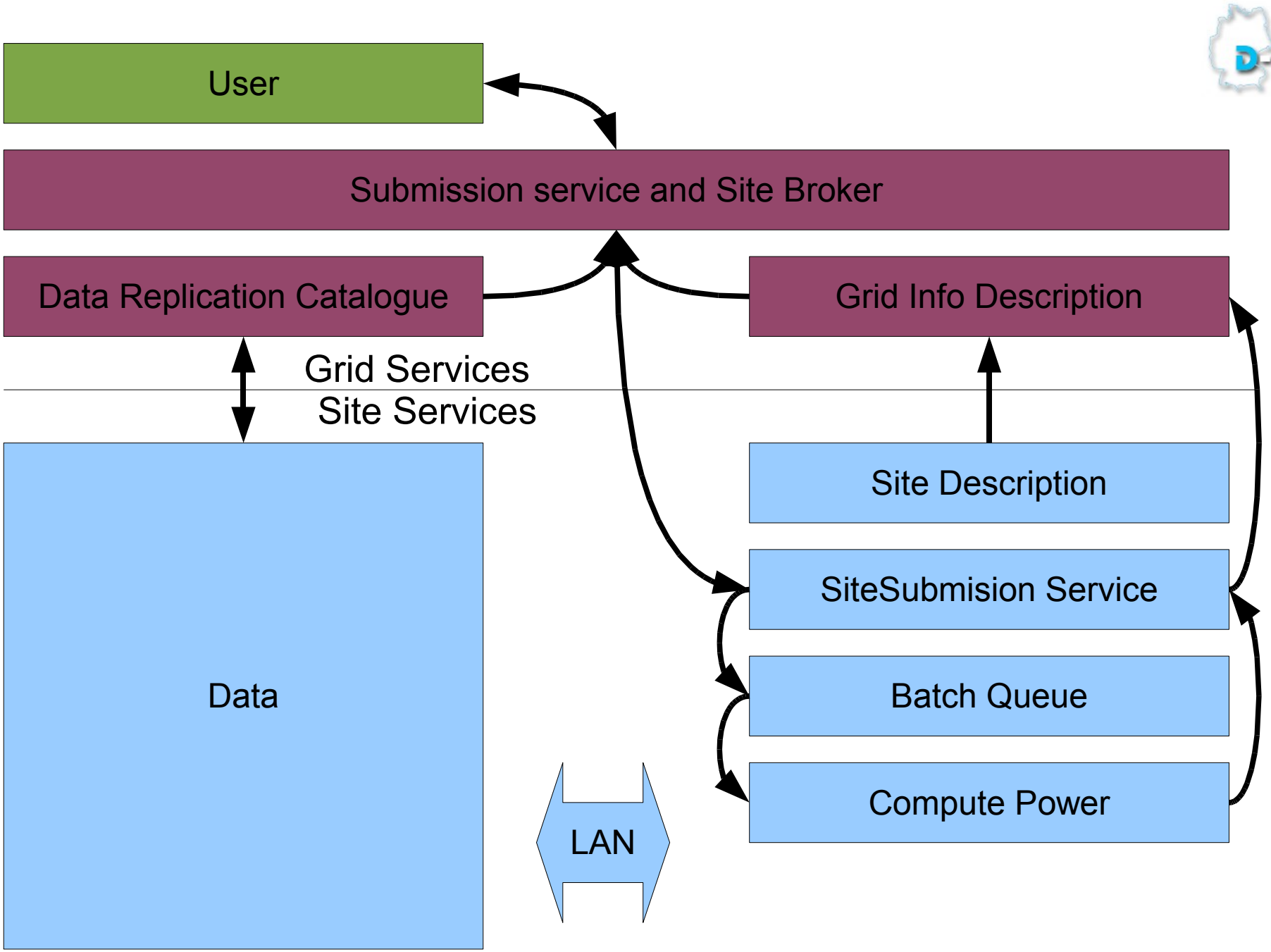
Grid Commissioning Use Case

- Scientists need to Process data with Compute
  - Come from many communities
    - HEP is relitively easy as trivial parallism
- Removing Compute cluster duplication.
  - Making a universal scientific cluster brings advantages.
    - Higher occuppancy of clusters.
    - Reduced costs to support novel work.
    - Do we rearly need a data center at our site?
- Allow scientists to work acorss accademic institiutes.
  - Remove differences between site frameworks
    - I knew how to do this at RAL, how do I do it at DESY?
  - Allow processing jobs to big for one sites budget.
    - Scaleable Compute and Data Processing

What is a Grid?

- An attempt to for fill use cases.
- Multi-Site shared distributed computing.
  - Shared National/International Grids exist.
- Service discovery system.
  - So common API's must be deployed.
- Job Submission System.
  - Submit JDL in one site and run on many.
- Cross Site data transfer coordination.
  - Scheduling transfers and replicas across sites.
  - Cross site catalogues prooved problematic.
    - Hopefull next Gen Grids fix this.

Grid Picture (Site View)



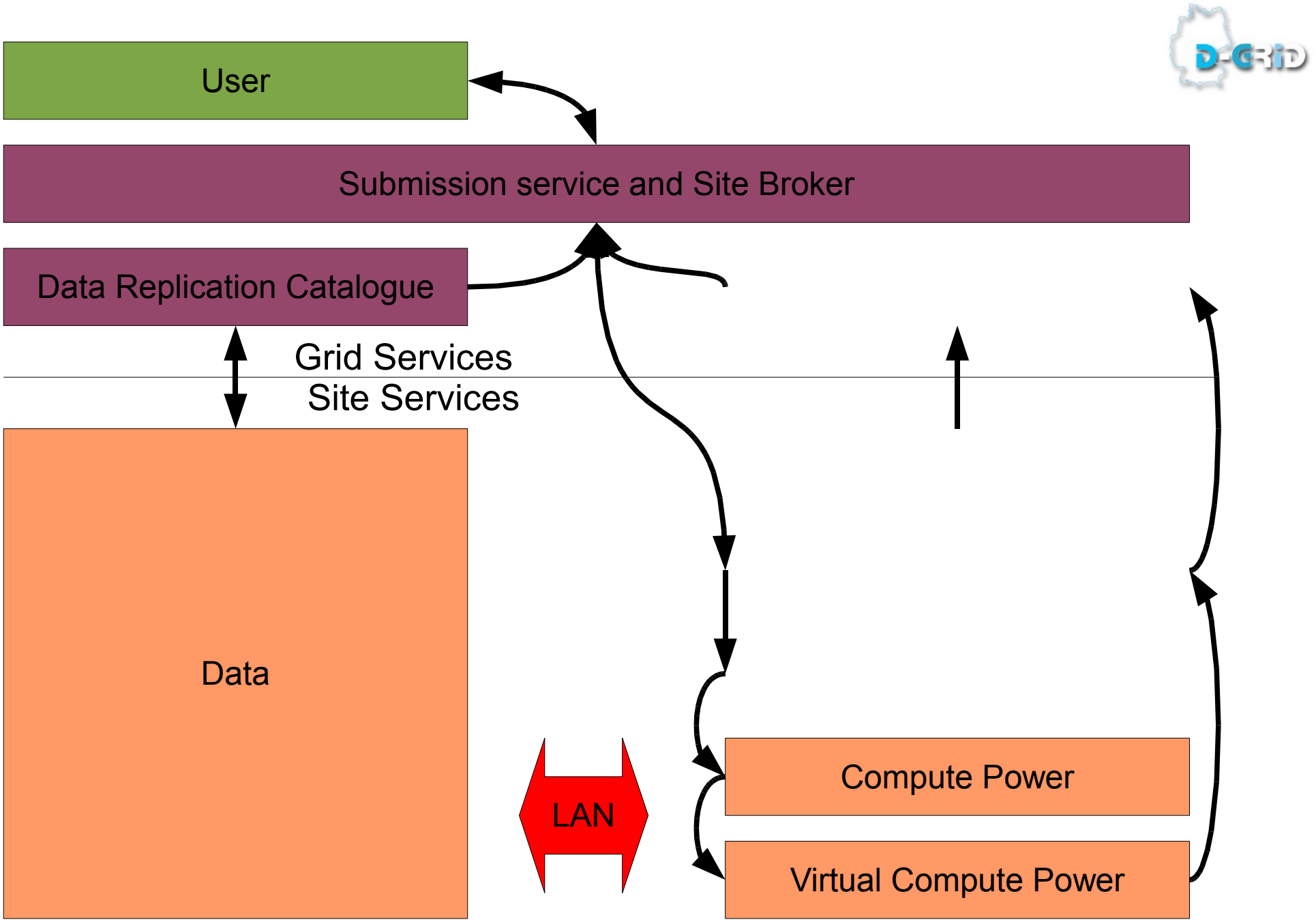
Cloud Use Cases (A best Guess)

- Provider
  - Sell data centre space.
  - Leaverage existing mangement infrastructure.
    - Management costs go up slower than farm size.
  - Support Multiple User comunities.
- Customer
  - Provide scalable resources.
    - Copeing with peak demand.
      - Eg USA arives at work 9AM morning.
  - Reduce Management Costs.
    - Do we rearly need a data center at our site?
- Note to Grid Comunity:
  - Big overlap in use cases.

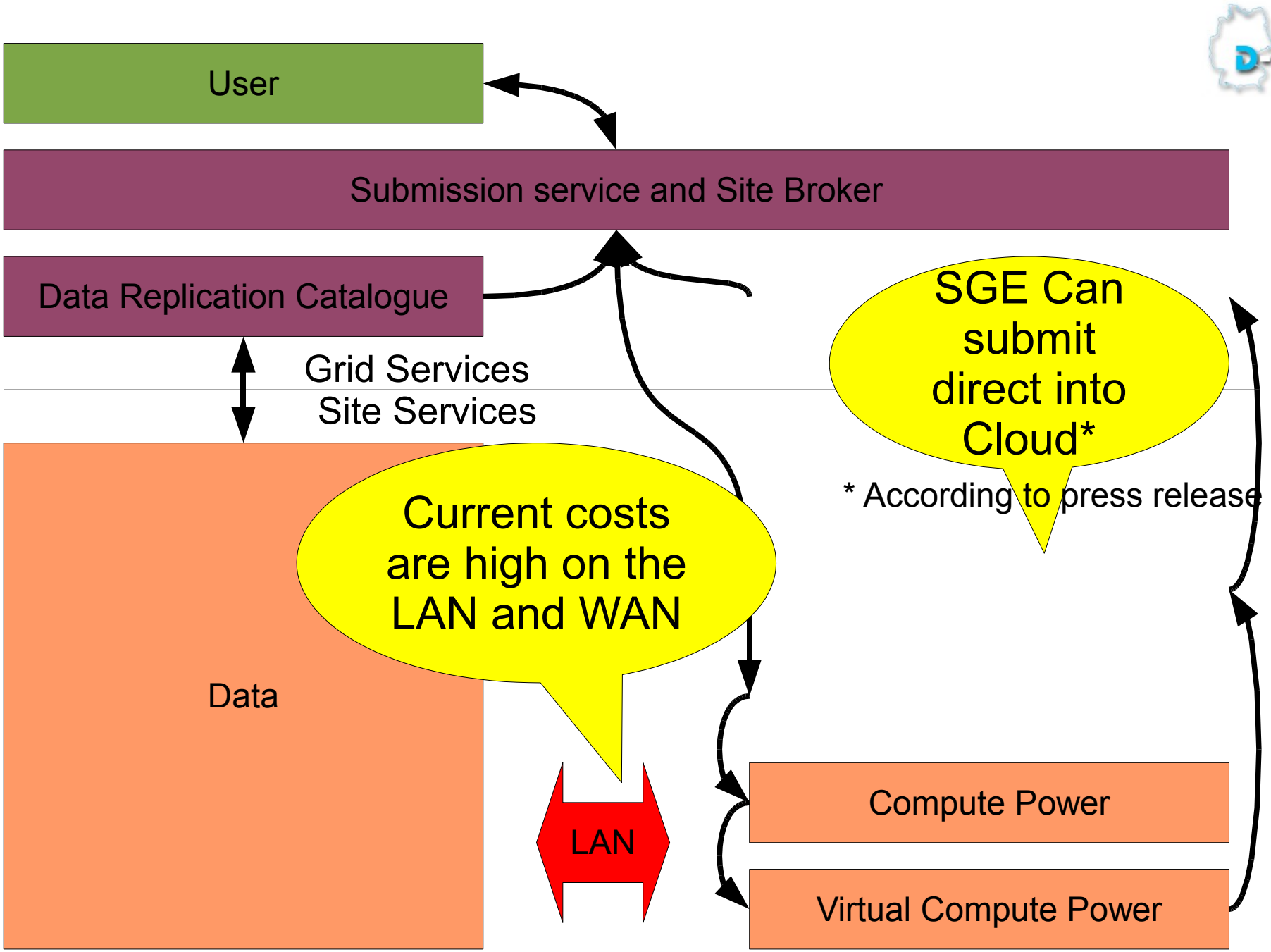
What is a Cloud?

- On demand Computer Scaling.
  - You buy resources not hardware.
  - Charged for CPU/time.
  - Charged for Network usage.
- User defined operating system.
  - You provide the image, the provider runs it.
- Implemented using Virtual Machines.
  - Your OS is just a file and a process (in KVM).
  - So can we run HEP jobs?
    - Yes we can!
      - Insert “Bob the Builder” Picture.

Cloud Picture (Note Missing Boxes)



Cloud Picture (With Comments)



Grid Consequences

- Homogeneous grids make upgrades hard.
  - All user comunitys must agree to port to new OS.
  - Enter into a Convoy situation (all at slowest speed).
  - OS driven externaly from experiments.
- Suporting mutliple user comunities.
  - We enforce the operating system on sites.
- Debugging localisation issues.
  - Why cant my jobs run in coutry XXXXXX.
    - Because they installed the localisation pack for perl!
      - Real use case from DESY experiment.

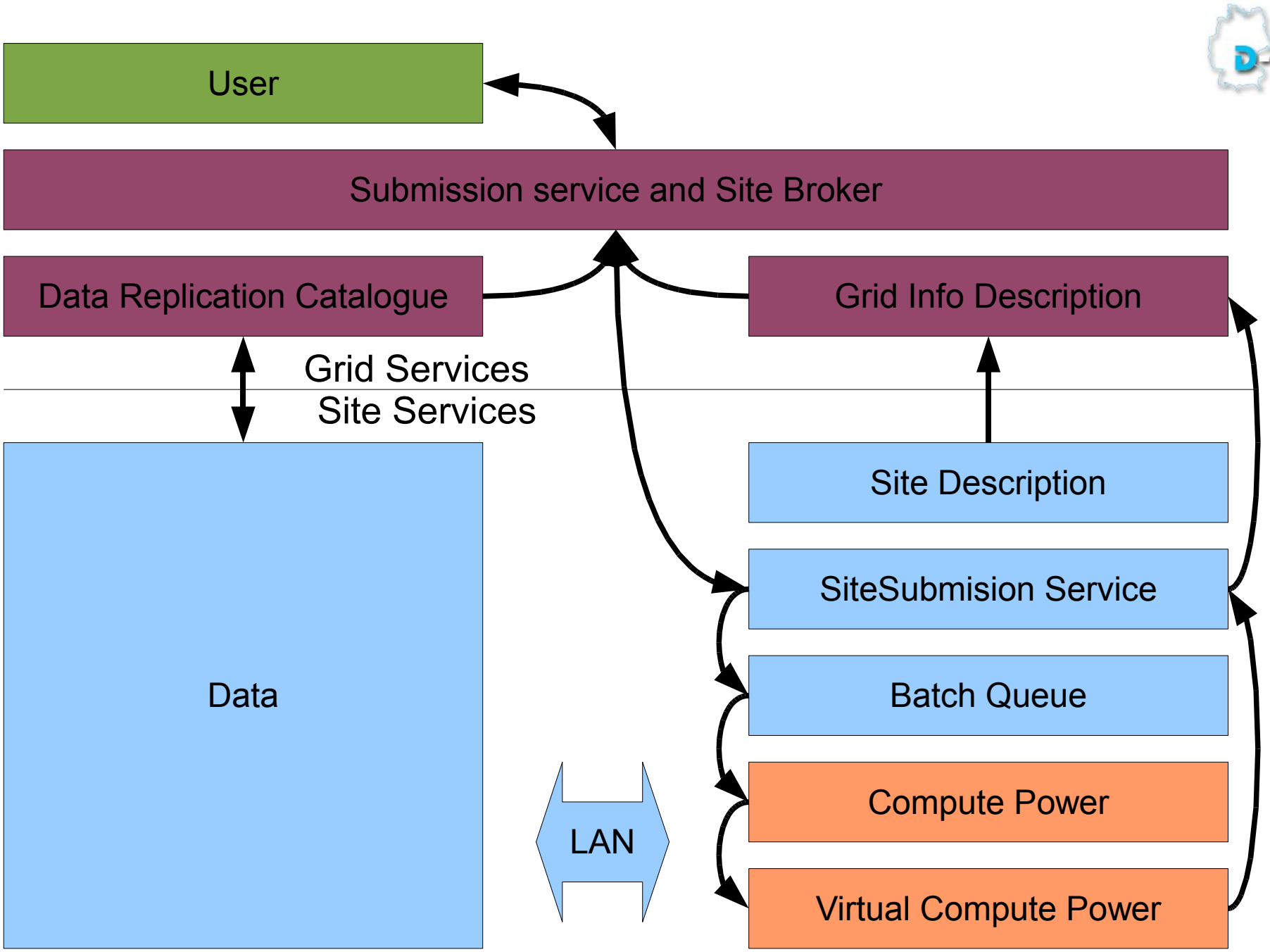
How do we do cope?

- Setting up multiple Queue's.
  - Limit to number that can be supported.
    - VO defined operating systems are impossible.
  - Typically static assignment.
    - Can lead to inefficiencies.
      - Job queues may be empty for some OS/platforms.
- Security of daemon jobs.
  - Typically cron scripts monitoring daemon list.
  - A reactive attitude to security.

What is an Academic Cloud?

- Every Job gets a fresh install of OS if wanted.
  - Some jobs may follow on with data from previous job.
- Grid Computing with Virtual Worker Nodes.
  - Fair Share using Batch Queue. (Like Grid)
- JDL defines the hardware.
  - Specifying the Memory and CPU. (Like Cloud)
- JDL defines the software.
  - Specifies the OS. (Like Cloud)
  - User Communities (VO's) could define OS variants.

Academic Cloud (Site View)



So A Virtual Worker Node?

- Multiple user communities needing different OS.
  - Simpler than running multiple queues/clusters.
- Community defined operating system.
  - CERN VM project will lead to deployment requests.
- Consistent deployment of Grid Jobs.
  - No more problems with perl localisations etc.
- Security.
  - OS can be restarted/reinstalled per job.

5 Models of worker node Virtualization

- Defined at DESY virtualization workshop.*
  - Worker node running one persistent virtual machine with a single OS image.
  - Worker node running multiple/2 persistent virtual machines with multiple/2 OS images.
  - Worker node running non persistent virtual machine images.
  - Worker node running non persistent virtual machine image from a library of OS images.
  - Worker Node running non persistent virtual machines and using user defined images.

16-17 January 2007
https://indico.desy.de/conferenceDisplay.py?confId=155

Worker node running one persistent virtual machine with a single OS images.

- Benefit
  - Job is isolated from management operating system.
  - Security of base OS image.
  - Possibly use able to suspend jobs.
  - Consistent DOM0 image.
    - No user access.
  - Easy to restore images/Maintain
  - Eases hardware abstraction.
    - Hardware needs SL4+ job needs SL3.
  - Technology is already available.

Worker node running multiple/2 persistent virtual machines with multiple/2 OS images.

- Benefit
  - Useful for parallel Jobs and back filling
    - Increasing cluster utilization when queue draining would normally be required.
      - KIT/Metacenter doing this since 2007.
  - Job is isolated from management operating system
  - Security of base OS image
  - Possibly useable to suspend jobs
  - Consistent DOM0 image (No user access)
  - Easy to restore images/Maintain
  - Eases hardware abstraction (SL3/SL4 EXAMPLE)
  - Technology is already available.

Worker node running non persistent virtual machine images.

- Benefits
  - Security is greatly enhanced
    - Worker node cleaning and job deamonization fears are eliminated.
  - Minimal modification to batch system required
    - Job submission epilogue and prologue can hide virtual machines details.
  - We believe that non persistent virtual machines will be the future of the worker node.
  - Memory available to a job is clearly split so providing better job isolation.
- Conclusion.
  - Done in University of Arizona 2007, now in DESY/KIT.

Worker node running non persistent virtual machine image from a library of OS images.

- Benefits
  - Great flexibility of run time environment
  - Multiple environments
  - Security
  - No need to balance resources.
    - Ie Queue aggregation
  - Experiments don't have to agree on SL3/4/5.
- Conclusion
  - Not possible in 2007.
  - Blah/CreamCE now makes this possible.
    - This release will be certified for mass deployment soon.

User defined images running on a non persistent virtual machines.

- Benefits.
  - Experiments don't have to agree on SL3/4
- Conclusion
  - We wont push this model.
    - Experiments should ask for it if they want it.
  - Is their much difference between user defined images?
    - If not are we just making work?
  - We see this as a a good final goal for Virtualization but don't yet see a strong use case from HEP yet.
  - Globus Virtual Workspaces has had this since 2007.
    - Why dont we see this in production in 2009?

Virtual Worker Node Requirements.

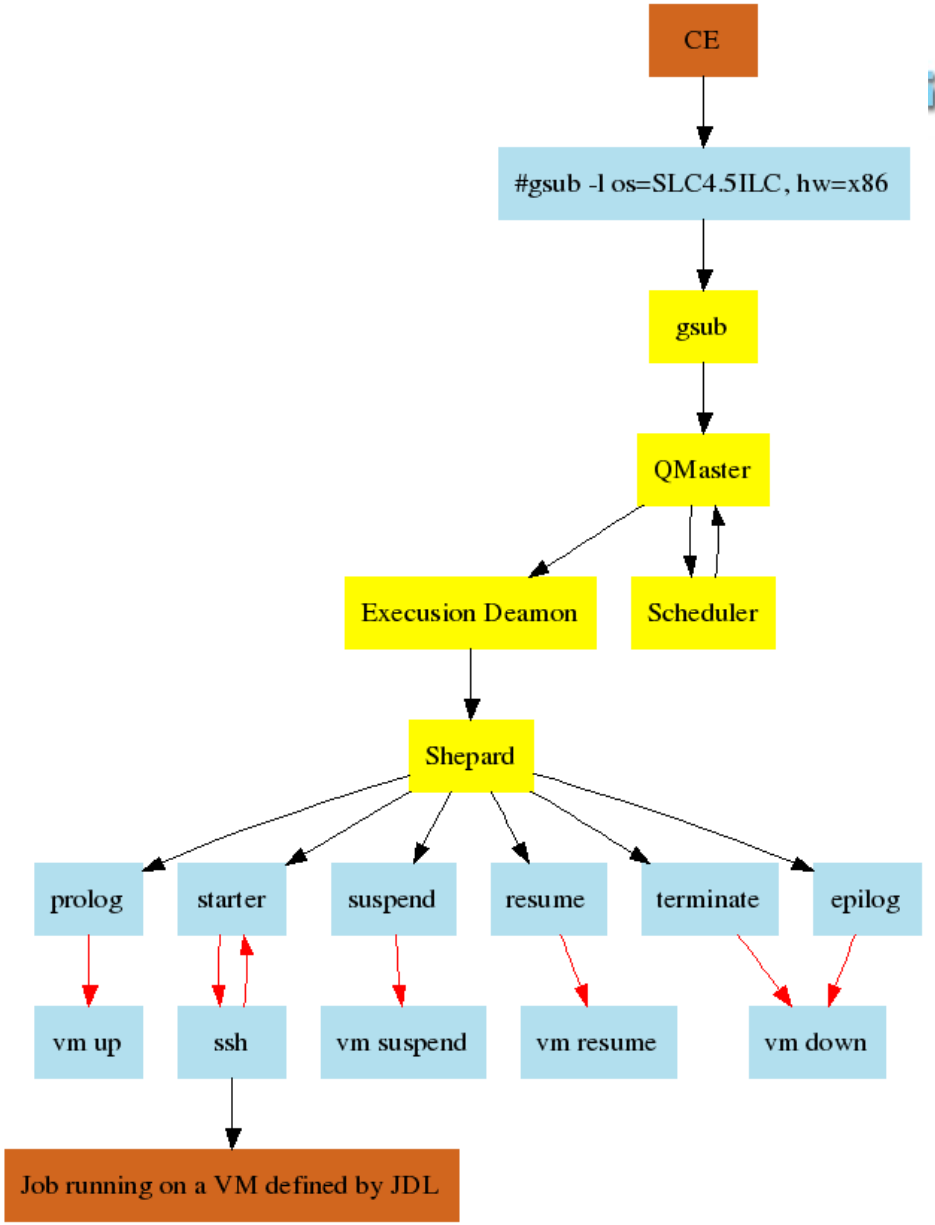
- Performance! (maybe not perfect yet).
- Must be transparent to current users.
  - Users don't need to change their environment
- Must store log files of Virtual Worker Nodes.
  - Archiving from OS/syslog-ng.
- JDL defines OS/CPU/Memory run (2nd Generation).
  - Grid integration needs better CE then LcgCE.
    - Older Grid CE's don't pass JDL through.
    - CreamCE/Blah scripts are not yet production certified.

Who has done what? The 1st Generation.

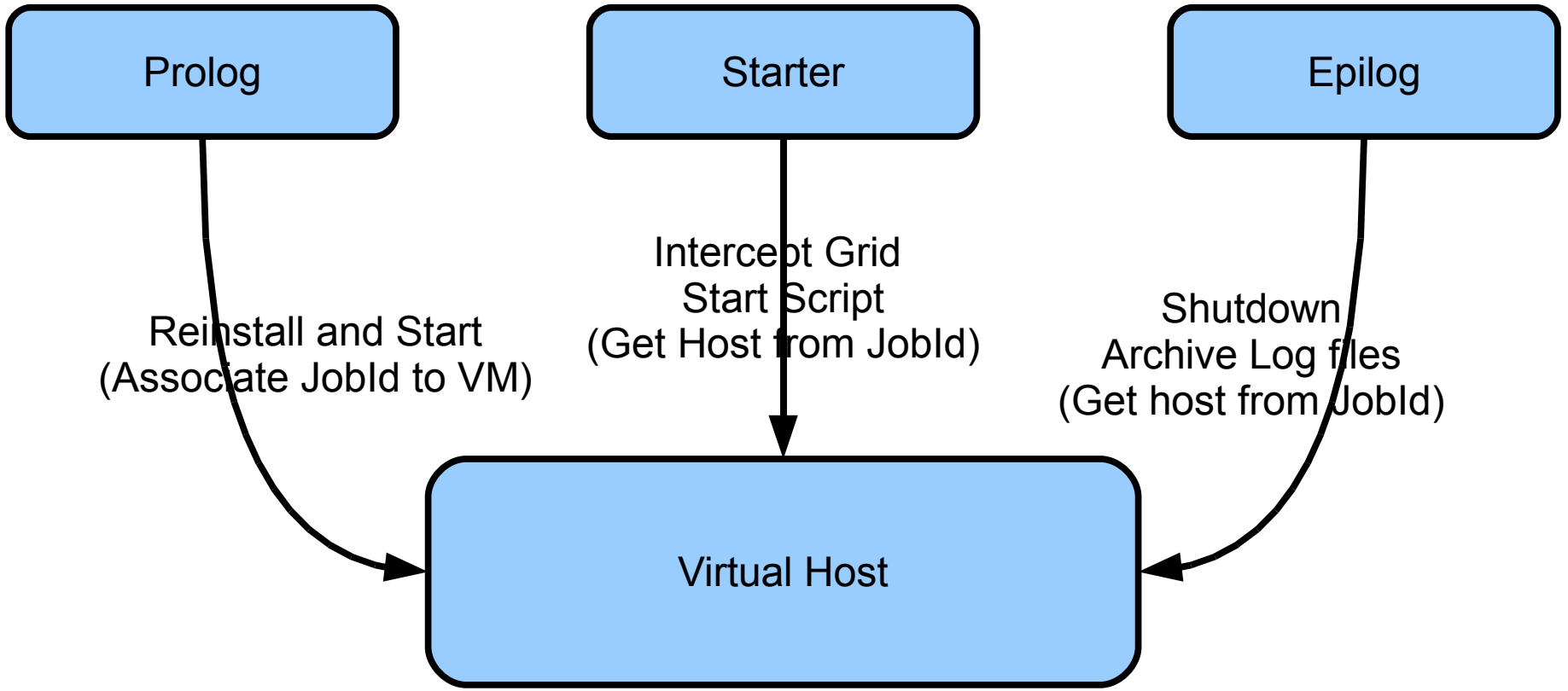
- Lots of early systems
  - Only knowledge of some/ but similar.
- DESY built a prototype solution.
  - Using PPS system/SGE testbed.
  - vmimagemanager.py + integration scripts.
  - Xen based.
- KIT built a system on SuSE to run the grid.
  - Site admins run SuSE.
  - Grid Worker Nodes should be SL4-5
  - Local jobs run on SuSE.
  - KVM based.

Current DESY SGE integration

- Similar to KIT solution
  - Prologue/Epilogue
  - Starter uses ssh
- Simple integration.
  - Bash + Python.
  - No database.
  - Lock files store state.

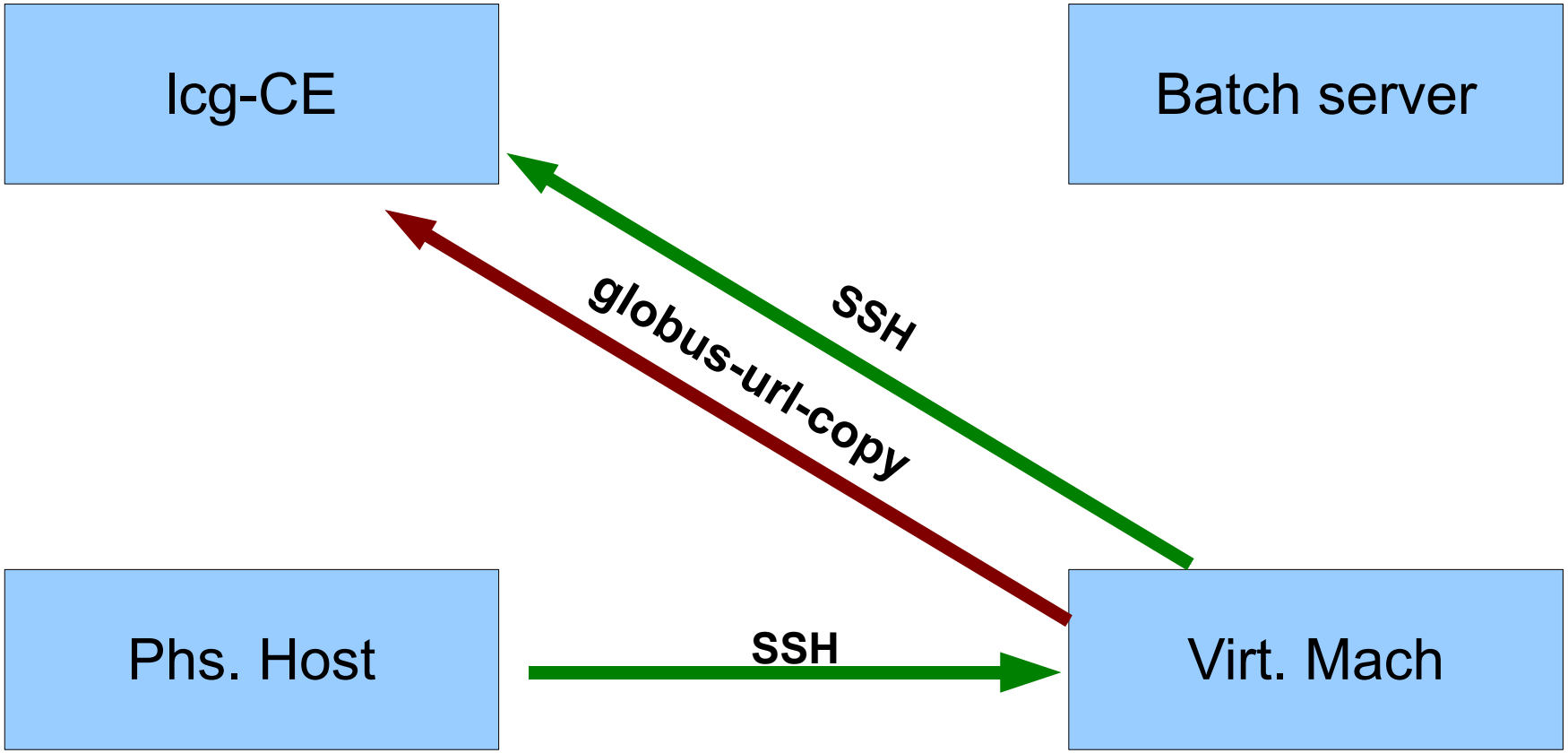


Batch Client and VM



- Only State stored is JobId to Host binding.
- VM gets all networking Via dhcp
- No DHCP cache installed on Physical node.
  Time outs getting DHCP can cause sporadic problems at DESY.

How to integrate wLCG VM.



Virtual Worker Image Distribution?

- DESY solution does not address this.
  - Thinking Bittorrent and RSS. (https for global?)
- KIT solution uses a iSCSI
  - So all images reside on SAN.
    - Not sure how this will scale.
- Propose using bittorrent and RSS over https.
  - So VO can sign images.
  - Scales with sites and requests.
  - New software required.

Current Limitations.

- Bandwidth
  - Disk and network systems are not as fast as native.
    - Xen > KVM > VirtualBox > Qemu
  - Faster solutions Chroot / BSD Jails / Solaris Containers.
- CE does not pass on JDL to Batch Queue.
  - Cream CE/Blah.
    - Uncertified latest release overcomes this.
- Can't discover if site supports your VM image.
  - Glue 2.0 has ExecutionEnviroment.
    - With VitualMachine attribute.
- User defined VM's not supported by LcgCE.
  - This changes with CreamCE.

Real Tricky Issues Unchanged.

- Security of VM's
  - Should only be an issue of network services.
    - Specifically NFS,rfio,dcap.
    - Partitioning network may be needed.
- Multiplicity
  - How many OS's do users demand on your site?
- Local customisations.
  - Setting your syslog settings on VM?
- Debugging.
  - Your environment has already ended.

Really Tricky

- Legal
  - Who's computer is it anyway?
  - Who's responsible for it being hacked?
  - Who put the XXXX content on it?
- Users want no OS updates ever?
- Getting realistic.
  - What's the difference between hacked user code and a hacked Virtual Machine?
  - Do we want unsecured network protocols?

Outlook

- Getting first Gen system was easy.
  - At least 4 other groups are doing this.
    - Russian / German / Italian
- Scheduling of Virtual machines
  - SGE consumables may help.
  - Backfilling (suspending jobs)
    - Useful concept maybe difficult.
  - Prefer to leave this to batch queues.
- Virtual site?
  - Reservoir project is proposing this project.
    - Starting with worker node and expanding to services.

Why has it not been mass deployed?

- No certification time slot.
  - NIKEF plan to do this on best effort.
  - DESY NAF is more parallel system. (No practical use)
- No process/model for user defined VM.
  - Users saw no point unless they get this.
    - Reduced Network bandwidth (like clouds).
    - Beginning to change.
  - Admins resistant. (security concerns)
  - No process for Image certification.
  - No mechanism delivered for image distribution.
    - Bittorrent and signed RSS anyone?

Academic Cloud Summary

- 1st Generation can be done now.
  - Prototyped with SGE/LCG-CE.
  - Planned for production for GridKa University. (Not FZK)
- 2nd Generation coming soon.
  - Memory/CPU/OS negotiation, Glue 2.0 support.
- JDL Defined Platform needs CreamCE
  - Uncertified CreamCE and “sge-blah” claim required functionality.
- Solutions should not (and need not) be hard.
  - Don't need any more than prefix and postfix scripts common in most batch queues.
  - SGE consumables removes most of state management. (maybe other batch queues too)