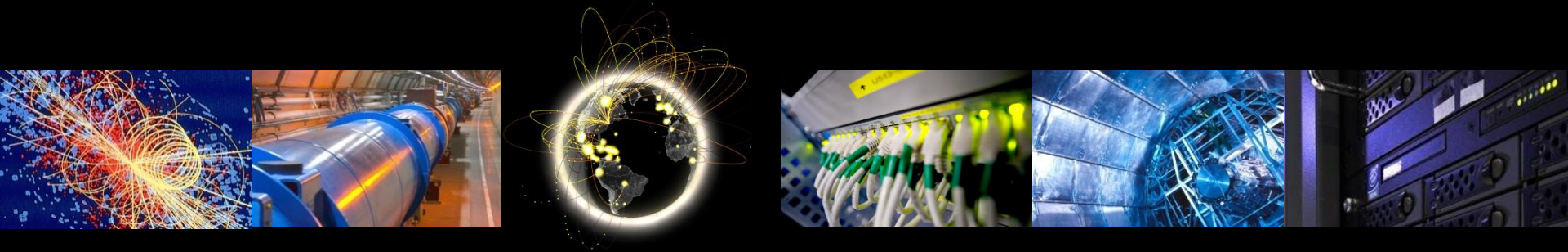


Priorities

Markus Schulz

January 2012 Amsterdam



Concentrate on one issues

- Faster turnaround of new software, including the rollout of client and services
 - Why?
 - this is the vehicle for change (see Caching-BDII)
 - not working well in the moment

Clients

- Are we installing clients in the right way?
 - local installation of grid clients (local control) + experiment specific software
 - local clients + some by experiments as part of their software
 - grid clients and experiment software by experiments
- Why not: All grid clients as part of the experiment software (via CERNVMfs)
 - default also in CernVMfs for non HEP sites

Clients

- Advantages
 - concurrent use of different versions
 - Pilot and staged rollout of clients by experiments
- Sites manage fabric and standard OS
 - decoupling of updates on sites and middleware
- Couldn't we use a similar approach for (small) updates to the services?
 - like OS updates

Services

- Changes and new features need to be handled quicker
 - however when done the take up is still slow
- Why?
- Everyone agrees that Pilots and Staged Rollout are the way to go!
 - improved pre-release testing will not improve the quality massively (diminishing return)
- Why isn't it easy to follow this paradigm?

What do Pilots cost?

- For a site setting up a Pilot service and participating in Staged Rollout is costly
 - often a singleton
 - tool geared towards many instances
 - significant risk that there will be problems
 - operational overhead, doing all over again.....
 - Staged Rollout
 - who goes first stands a higher risk to get into harms way
 - reliability/availability figures ...
 - figuring out config. details...
 - certain probability to rollback and redo everything

What can a Site win?

- In the past: Significant easier to run service
- Now: A bit better, small gain for experiments
 - sites and users have workarounds for many known issues
- Recognition from the community!
 - if all goes well.....
- Does this sound attractive?
 - Hm, not to me...

What to do?

- Come up with reasonable incentives
 - better than “Site of the Month” concept
 - at least sharing the risks
 - accepting some volatility

Longer Term

- Simpler, less layered services
 - Complexity moved into the experiments frameworks