



Vac Project views

Andrew McNab

LHCb, GridPP, and
University of Manchester



Vac Project context

- Came out of GridPP clouds and VMs working group
- **Manage VMs to run jobs, using the Vacuum model**
 - Experiments' VMs instantiated by resource providers
 - Monitor outcomes to scale instance numbers up or down
- Vac - autonomous hypervisor machines managing their VMs
- Vcycle - agent to manage VMs on an IaaS like OpenStack
- **Also involved in defining VM architectures**
 - Make extensive use of CernVM and cvmfs
 - Mainly for LHCb, but also VMs for ATLAS and GridPP
 - Helping with RAL-defined CMS VMs
- Working with 6 sites, others interested / setting up



What we probably don't need ...

- Source code repository
 - Using GitHub
- Issue tracker
 - GitHub tracker, pull requests; GGUS for operational issues
- Mailing lists, web forums
 - Using CERN e-groups
- Web hosting
 - www.gridpp.ac.uk/vac and www.gridpp.ac.uk/vcycle
- Central validation team
- Access to commercial tools



Stone Soup

- Most of the uses we see are forms of communication between projects and with user communities
- These can be highly leveraged: like “Stone Soup”
 - **A good environment can catalyse an outcome which is more than the sum of its parts.**
 - (Story is: group of travellers, each with just one ingredient are prompted to share firewood, meat, vegetables, seasoning by a helpful stranger with a cooking pot and a stone, who promises each one “Stone Soup” in return for cooperating.)
- Criticism of this is: “Well, people could have done this themselves anyway.”
- But they often don’t.
- **A third-party like HSF is well-placed to encourage this to happen.**



Some concrete ideas



Technical notes

- **Hosting a repository of technical notes from projects would foster cooperation, interoperation, and common APIs**
- Much more lightweight than OGF style standards
- Purely informational so avoid a complex approval procedure
 - Just require common template, spelling, obvious typos?
 - Maybe prior-approval of scope by submitting title/abstract?
 - Key thing is to get projects and groups of projects to document their interfaces and processes
- **Getting “competing” projects to work together on APIs especially useful**
- Also likely to spawn real/virtual meetings even workshops
 - And maybe papers eventually



Meetings

- HSF can't organise lots of workshops for lots of projects
 - **However, it could easily act as an umbrella for meetings**
- People are perfectly capable of hosting meetings at their site
 - But getting other people/projects involved is more complicated
- Key is communication channels:
 - Build on existing cooperation, perhaps fostered by HSF
 - **Advertising meetings: be the central place for announcements**
 - Shared space in Indico; Vidyo
- May be possible to colocate these informal meetings in similar areas
 - Compare WLCG pre-GDB workshops
 - In addition to whatever formal HSF conferences are planned



Software validation as communication

- Approved guidelines for how to plan validation before releases
- Use ISO9001-like ideas
 - **Document what you promise to do, record what you've done this time, give users assurances about what they are getting**
- So not something like “Projects must test all aspects of HSF software with 3 or more sites/experiments” ...
- ... more like “Projects must document what features of the software will be validated, how this will be done, and provide records of the checks”
- Validators can then sign off releases (maybe only a subset of “golden releases”?)
- Gives users more idea of what quality level they can rely on
 - **Avoid “How much testing did you actually do on this?” complaints.**



Software distribution channels

- HSF has an opportunity to publish software packages from many HSF member projects in one place *very conveniently*
- Different distribution technologies:
 - eg RPMs in YUM repos, or executable code in cvmfs
- Could tie this in to the software validation process ideas
 - Raw and validated channels? Levels of validation?
- Some projects are already going through EPEL approval process
 - So some overlaps, but that's ok (especially if identical RPMs are published in both)
- **The big gain would be site admins, VM architects etc getting access to many packages by configuring one channel**
 - e.g. avoid proliferation of per-project YUM .repo files



Summary

- We see benefit from HSF primarily in fostering communication between projects and user communities, and between projects
- Well placed for highly leveraged “Stone Soup” activities
- We’d like to see HSF as an umbrella for
 - Technical Notes
 - Meetings
 - Distribution of software packages
 - Software validation process guidelines
- Doesn’t stop the other activities outlined in the White Paper