egee

Enabling Grids for E-science in Europe

www.eu-egee.org

(Web) Service
Common management

Gavin McCance
EGEE JRA1 Data Management Cluster

Issues

- Not going to talk about web services
  - I'm looking from a web service point of view but hopefully the comments are general

- Talk about common service components
  - Mostly from a management and administration point of view
  - Highlight a few issues for discussion

Common management functionality

- Possible common functionality of services
  - That all our services should provide
- Standardised administration and management functions
  - Agree a common state management mechanism
    - start, stop, restart, shutdown, draisuspend
    - drain (shutdown): service all existing clients (before stopping)
    - Agree on how the service administrator actually does this
  - Agree live configuration management mechanism:
    - Live changes to a service's running parameters
    - Examples:
      - swapping database backends without service interruption
      - gracefully throttling the number of connections if the service looks like it's becoming overloaded
    - Agree service status monitoring: how many connections, what load, what memory usage, status of external resources used by the service: databases, bandwidth.
    - Alarm plugins .. "AutoSMS at 3am when the log partition fills up …"

Possible common management mechanism

- Propose: a standard look-n-feel per-service web portal for per service management
  - Many of the specifics will be different, and some services have more complex management issues than others, but the general "first point of call" for the service administrator should be the same
  - Might be too expensive for non-container managed services

- Propose: On-service common port-type for service management

- Site wide portal for administering all services on one site
  - Using the service management port-type exposed by the services

- Global (VO) portal for administering multiple services
  - Again using the service management port type exposd by the services

Services: common components

- Other useful things we might agree

- Standardized logging mechanism and formats
  - Common interface? Common format? Transaction ID?
- Policy management
  - Common format for policy. Common way of deploying them.
    - e.g. transaction rollback policy. How to handle failures
    - Again, service specific, but the way of doing it should be common
- Per-service security management
  - Common way of deploying your authorization policy on the service
  - Common log files format for auditing

How to proceed?

- What functionality should be common?

- Who provides it and supports it?
    - Each service provider or common code?

- How / when do we define it?
    - Need to define it soon