#### Introducing the dCache info service



Paul Millar <paul.millar@desy.de>

### Who are you, listening to this?

- You are one of:
  - a member of the dCache collaboration,
  - a Tier-A integration collaborator (hello Tanja!),
  - a Tier-1 site-admin,
  - a valued college from CERN (hello Flavia!).
- What you will understand after listening:
  - An understanding of how the info service works.
  - Some ideas about how the data may be accessed and used.



#### Overview of talk

- The goals and non-goals of the info service.
- A medium-to-high level view of how it works.
- An overview of the provided information.
- How to obtain "live" information.
- Summary.



#### What is the info service?

- A robust, best-effort, "one-stop shop" overview of the current status of a dCache instance for external consumption.
  - Robust: the info service will continue to work independently of the rest of dCache.
  - Best-effort: there may be delays in information being updated (1 minute order-of-magnitude).
  - One-stop shop: you should be able to get all the information you require.
- It decouples updates from queries:
  - Querying is fast and robust



#### What the info service is not...

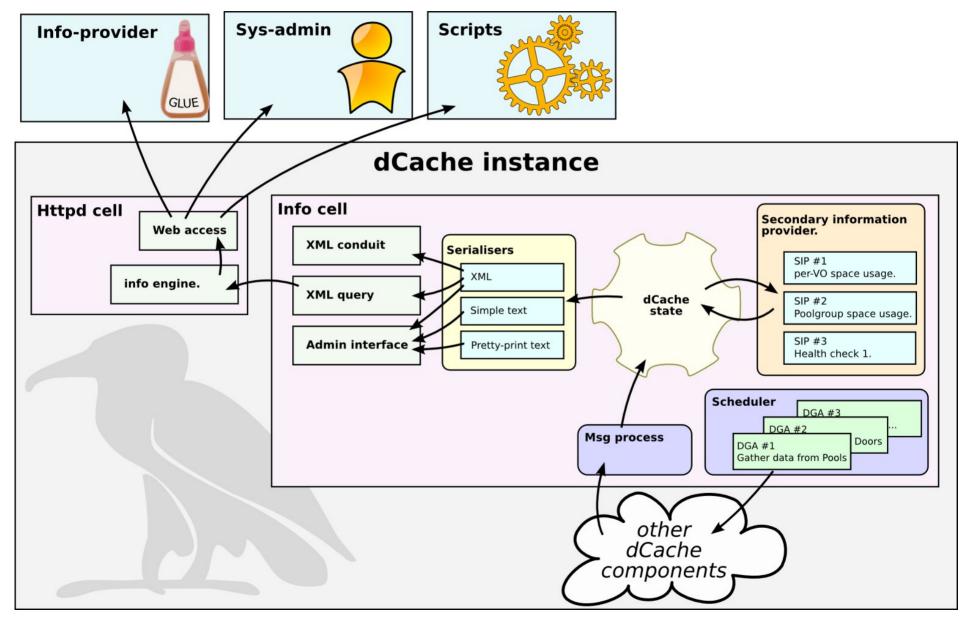
- Not used by other dCache components [\*]
  - As this would:

[\*] this is a small white lie: more details in the next slide

- break dCache's distributed architecture and introduce a (new) single point-of-failure,
- increase latency in information propagation,
- Doesn't provide historic data:
  - Duplicating what sites (almost certainly) already have:
    - Ganglia, Nagios, LEMON, Mondin, Zabbix, ZenOSS, ...
- It is not an info-provider:
  - although the new WLCG GLUE info-provider is a customer of the info service.



#### How it works.



### How updates are scheduled

- Data-Gathering Activities (DGAs) schedule fresh data requests.
  - DGAs give strong control over how often messages are sent.
  - We're careful not to over-burden components.
- Reply messages are processed independently.
  - Info cell is robust against messages being lost or the replies delivered out-of-order.
  - Allows for asynchronous updates: much faster updates, low bandwidth overhead (W.I.P.)





- Derived data is the set of metrics calculated from other data within the state tree.
  - For example, space.total metric of a poolgroup is the sum of all space.total metrics of member pools.
- A Secondary Information Provider (SIP) calculates derived data.
  - Only triggered when *important* metrics change value.
- Uses:
  - Calculate aggregated space statistics (current)
  - High-level internal health checks (under investigation)



#### Storage in a tree

- Most metrics are held for finite time.
  - When time expires, they are automatically flushed.
  - Robust against stale information.
- A serialiser generates representations of state.
  - Adding new output formats is easy.
- Tree storage is abstracted.
  - Currently only uses memory, so serialisation is fast.
  - Structure is not hard-coded in the storage.
    - Storing new branches and/or metrics is easy.



# What information is provided?

- Shallow structure with cross-links.
  - Normalised data: metrics appear only once.
- The following top-level branches exist:
  - summary: various aggregated information,
  - All other top-level branches are lists of items (as branches) with item-specific information below.
  - These lists include: pools, poolgroups, links, linkgroups, domains, doors, reservations.
- There's too much information to detail here.



#### Tiny fragments of state tree

reservation created metric a -"2008-06-11T12:42Z" metric name"simple" type"stri "Jun 11, 14:42:30 CEST metric name's metric name - authorisation metric name's metric nam matric "NEARLINE space metric ner metric ner "30000" metric na metric a • metric a "80001" + metric and CUSTODIAL reservation recreated + metric -"2008-05-17T08:34Z" metric name unte -1213691681 metric none'smple" type'strie "Jun 17, 10:34:41 CEST" metric name of metric name's · metric ..... -70000 · metric · - "ONLINE" + space metric metric a metric nor metric no + metric name in + metric na "REPLICA" reservation · created "2008-06-11T12:42Z" metric name units" + metric on "Jun 11, 14:42:09 CEST · metric composite to and state -RESERVED + metric name Trie authorisation metric nam metric nam metricos "NEARLINE" , space metric na metric name" metric and metric

- dCache



#### troducing the dCache info service

- domain name-stracketomain \*System@infoDomain:System@dCacheD... cells created . metric . metric name unix type intege "1218537122" "Aug 12, 12:32:02 CEST" metric name type" type"string matric a"dmg.cells.network.LocationMgrTun. - metric name "event-spece size" to L-"0" • metric a ·2\* versior metric m metric name ellowestat created - metric cane 150-86 "2008-08-12T10:31Z" · metric name unit type \*1218537103\* + metric name aim Aug 12, 12:31:43 CEST "System" dmg.cells.network.LocationMgrTun. . metric -"0" • metric .2" • version metric a metric name ella created + metric -· metric care boot tone boot -"1218537100" + metric name sin "Aug 12, 12:31:40 CEST" • metric name "type" type" string "Generic" metric dmn.cells.services.LocationManan • metric -0" metric version metric national "cells" metric name ellas + created - metric - "2008-08-12T10:33Z" -+ metric name 'say -Aug 12, 12:33:41 CEST metric name type "System" • metric as dmg.cells.network.LocationMgrTun. -0" metric version metric m metrica -"CA-1.28" cell name 1-20 created metric name 150-8601" type "2008-08-12T10:312"



# How do I start the info service?

- Configure dCache so it's started on a node.
- ... or start it manually:
  - /opt/d-cache/bin/dcache info start
  - (optionally) have a cup of coffee (2—3 minutes) whilst initial set of data is populated.
- Start querying the information.
- No configuration needed for the info service [\*].
  - [\*] The GLUE info-provider *does* require careful configuration; but the info-provide is a separate, distinct component from the info service.



#### Accessing the information

- Via the admin interface
  - Commands for navigating state, like: cd, ls, pwd.
  - Choice of output format
- Via XML Conduit
  - A TCP connection get complete state as XML.
- Web front-end
  - Preferred method, but requires the httpd cell.
  - For example:
    - http://dcache.example.org:2288/info
    - http://dcache.example.org:2288/info/pools





- Info service provides a best-effort overview of a dCache instance.
- Maintaining state and divulging information are decoupled:
  - fast, robust.
- Supports some advanced features:
  - Derived data (re-)calculated as state changes.
  - Multiple output formats and transports.
- If additional metrics, data formats or transports are needed, they can be added.