Storage Classes in CASTOR

Shaun de Witt
CCLRC/RAL

CCLRC
Rutherford Appleton Laboratory

Introduction

- Storage Classes via SRM
  - focus mainly on SRMv2
    - …or this is what I say!
- Storage Classes at RAL Tier 1
  - current and future
    - … or this is what I mean!

Shaun de Witt
CCLRC/RAL

CCLRC Rutherford Appleton Laboratory

Storage Classes via SRMv1

- No concept of spaces/storage areas/ service classes
- SRM v1 managed service classes using stagemap file
  - mapped user to service class via DN

Shaun de Witt
CCLRC/RAL

CCLRC
Rutherford Appleton Laboratory

Storage Classes via SRM-2

- Agreed ‘Storage Classes’
  - disk1tape1; disk1tape0; disk0tape1
  - ‘permanent’, ‘durable’, ‘volatile’ terms deprecated – at least by me!
- Represents some level of quality of service and accessibility of data sets held within them

Shaun de Witt
CCLRC/RAL

CCLRC
Rutherford Appleton Laboratory

Mapping to SRM terms

- SRM defines these as:
  - Retention Policy:
    - <CUSTODIAL|REPLICA|OUTPUT>
  - Access Latency
    - <ONLINE | NEARLINE>
- **disk1Tape1** -> ONLINE & CUSTODIAL
- **disk1Tape0** -> ONLINE & REPLICA
- **disk0Tape1** -> NEARLINE & CUSTODIAL

Shaun de Witt
CCLRC/RAL

CCLRC Rutherford Appleton Laboratory

Other SRM Terms

- Access pattern
  - <TRANSFER_MODE | PROCESSING MODE>
- connection type
  - <WAN | LAN>
- client networks
  - not used by CASTOR
- protocols
  - supported/required protocols

Shaun de Witt
CCLRC/RAL

CCLRC Rutherford Appleton Laboratory

Support within CASTOR

- access pattern, connection type and client network arrays silently ignored.
  - additional service classes can be set up to cover these
  - accessed via spaceTokenDescription
- SRM holds service class names
  - ‘spaceTokenDescription’ == ‘spaceToken’ == svcClass

Shaun de Witt
CCLRC/RAL

CCLRC
Rutherford Appleton Laboratory

Limitations

- No dynamic space allocation
  - support for static spaces only
    - asynchronous API only.
  - ‘infinite’ lifetimes
- access pattern, connection type and client array silently ignored
- New service classes need to be added to SRM
- spaceTokenDescription must be unique within a VO

Shaun de Witt
CCLRC/RAL

CCLRC Rutherford Appleton Laboratory

Service Classes within RAL Tier-1

- Currently running SRMv1
  - 3 service classes via three SURL endpoints (per VO)
  - SURL of form …/<vo>/diskNtapeM
    - i.e. storage info part of namespace
  - Each one has distinct SRM serving it
    - or will have…

Shaun de Witt
CCLRC/RAL

CCLRC Rutherford Appleton Laboratory

RAL SRM mapping

End Point	SA	Service Class
ral-srma	…/cms/disk1tape1 …/atlas/disk1tape1 …/lhcb/disk1tape1 …	cmsdisk1tape1 atlasdisk1tape1 lhcbdisk1tape1 …
ral-srmb	…/cms/disk0tape1 …/atlas/disk0tape1 …/lhcb/disk0tape1 …	cmsdisk0tape1 atlasdisk0tape1 lhcbdisk0tape1 …
ral-srmc	…/cms/disk1tape0 …/atlas/disk1tape0 …/lhcb/disk1tape0 …	cmsdisk1tape0 atlasdisk1tape0 lhcbdisk1tape0 …

Shaun de Witt
CCLRC/RAL

CCLRC
Rutherford Appleton Laboratory

Moving to SRM v2

- Use mapping previously supplied
- Other service classes accessible via name
  - spaceToken and spaceTokenDescription

Shaun de Witt
CCLRC/RAL

CCLRC Rutherford Appleton Laboratory

Issues and Limitations

- Single CASTOR instance may mean having to share ‘default’ space
  - should be mitigated in SRM 2.2
  - work still in-progress
- SRM1->SRM2 migration
  - V1 SURLs will have SA metadata embedded at RAL, but not necessary for V2 (different endpoints for each SA)
  - May need to migrate SURLs
- diskservers can only support a single service class per VO
  - all filesystems must be in same class
  - caused by castor-gridFTP limitation since service class not passed in.
  - implies possible addition resource requirements, or fewer resources/service class/diskservers cannot be shared between VOs
    - possibly solved by castor gftp-2

Shaun de Witt
CCLRC/RAL

CCLRC Rutherford Appleton Laboratory

Issues and limitations

- Can not guarantee directly disk1…
  - disk1 => user managed space, but not always well managed!
  - can cause CASTOR meltdown when disks overfill
  - If single CASTOR instance => not available for any user.
  - possible solutions to be discussed
    - Garbage collect older/least accessed files
    - As above but with tape back-end
    - careful monitoring and updating of resources
    - Stop accepting write requests to diskserver
      - available in later release of Castor
    - Use d-cache/dpm for disk only
      - only if desperate – extra support effort required
- CASTOR solution actively being worked on

Shaun de Witt
CCLRC/RAL