8 January 2008, Flavia Donno, Maarten Litmaath

Handling of T1D0 storage class for CCRC'08

DRAFT

Open for discussion

In this document we make a proposal on the handling of the T1D0 storage class from the Tier-1 sites and the experiments when dealing with SRM v2.2 storage services during the CCRC08 exercises in February and May 2008. We give some technical background information concerning the implementation of CASTOR and dCache in order to justify our proposal. We also list the important data management use cases that may have significant implications on the functionality desired from SE implementations and/or the configuration of a site's SE(s). In particular, we focus on system-managed disk buffers. We also elaborate on the use of space token descriptions specifically for Get and BringOnline operations.

Executive Summary

Because of the different support offered by CASTOR and dCache for the T1D0 class of storage, in the following we would like to propose a configuration to Tier-1 centers for the CCRC08 exercise in February 2008, *for production activities only*. User analysis will be handled differently. Please check section 6.3 for details

A. Proposal for Tier-1 centers:

We would like to propose the following configurations for dCache and CASTOR Tier-1 sites.
A.1 CASTOR:

- Configure storage classes for T1D0 as requested by the experiments.
- Configure a fraction of the T1D0 disk space as default space for that specific storage class in agreement with the experiments keeping into consideration their needs in terms of WAN/LAN access.
- Allow for the use of other handles such as the user identity for selecting pools when the space token description is not specified.

A.2 dCache:

- Keep T1D0 spaces relatively small and use them only as buffers for writing into the storage system.
- Keep most of the T1D0 disk space unassigned to any space tokens, so that it can be used for restoring large data sets concurrently.
- Possibly configure paths to allow for the selection of specific pools while recalling a file from tape.

A.3 Proposal for experiments:

- Use space tokens for recalls. This will allow for a controlled pool selection in case of CASTOR and to get prepared for full token usage.
- Use paths consistently as much as possible.

1. Background technical information

Both CASTOR and dCache provide SRM v2.2 implementations that are compliant to the WLCG MoU, but currently differ significantly in the behavior for tape recalls.

1.1 CASTOR behavior when recalling files from tape

CASTOR can recall files into an SRM v2.2 space, but the space in which a file was originally stored is not remembered, so the recall should either be given an explicit space token description, or the system would have to use other information (user identity, WAN/LAN flag, retention policy/access latency) to determine the desired area where the file should be restored. Please note that at the moment the current high-level Data Management tools do not allow for parameters such as the storage class and the WAN/LAN flag to be specified. Only the user identity can be derived from the VOMS proxy used by the user when issuing a request to the SRM v2.2 CASTOR server. Furthermore, no other static configuration of the server allows selecting specific pools to be used during the recall operation if the space token is not specified.

1.2 dCache behavior when recalling files from tape

dCache currently cannot recall files from tape into an SRM v2.2 space, even if the space token description is explicitly specified. Therefore, there has to be sufficient disk space that is not assigned to any space token. However, dCache allows for the selection of pools during a recall through specific static configurations of the server. For instance, pool selection can take place specifying the file paths. It should be noted that those pools must not be managed by the space manager.

2. Tier-0 data handling

Each experiment sends its raw data with rfcp to a CASTOR service class instance dedicated to Central Data Recording (CDR). Such an instance can be given a space token to allow it to be targeted by SRM v2.2 operations, if needed. The following operations should then take place in parallel:

1. The data needs to be copied to tape as soon as possible such that the copy kept on the online disk buffer can be deleted.
2. The data must be distributed to the Tier-1 centers.
3. The data must be made available for first-pass reconstruction, whose output must be written to tape and sent to the Tier-1 centers as well.

A service class instance dedicated to CDR will be optimized to handle at least step 1. For step 2 it may be needed to copy the data to another service class instance that is (better) connected to the WAN. An SRM v2.2 BringOnline (BoL) or PrepareToGet (PtG) function initiated by the FTS would then trigger a disk-to-disk (d2d) copy. Data may also be recalled from tape later. Similarly, for step 3 it may be needed or desirable to copy the data to another service class instance that is better matched to the read patterns exercised by reconstruction programs. Interference with steps 1 and 2 could then be controlled better.

3. Tier-1 data handling

A Tier-1 center receives data from Tier-0, other Tier-1 centers and Tier-2 sites. It sends data to other Tier-1 centers and Tier-2 sites. Here CERN can also take the role of a Tier-1 center and any Tier-1 center can take the role of a Tier-2 site.

1. The raw data and the first-pass reconstruction results received from Tier-0 need to be written to tape.
2. In parallel the raw data should be made available for second-pass reconstruction, whose output must also be written to tape and may need to be copied to a partner Tier-1.
3. Regularly or occasionally, data sets already present at the Tier-1 may need to be recalled from tape for reprocessing, possibly multiple times.
4. Tier-2 sites will download certain data sets for analysis.
5. Tier-2 sites will upload Monte Carlo data and certain analysis output to be written to tape.

The disk buffers that receive the data from Tier-0 may not be suited for writing the data to tape or for serving data to reconstruction programs. Also here d2d copies may be needed. For step 2 it is important that the newly arrived files remain available ("pinned") on disk until they have been processed, such that they need not be recalled from tape in this phase. For steps 2 and 3 it is important to ensure efficient staging of the necessary data, so that both the tape system and the batch farm may be used efficiently by the reconstruction jobs. A large amount of disk space would allow for tapes to be read back entirely in one go, for many jobs to reprocess data in parallel, and would reduce the need for merging small output files into files of nominal sizes. If the relevant disk space were dedicated, it would reduce the probability that some of the necessary data files would have their disk copies garbage-collected prematurely to make space for concurrent activities.

4. VO data handling

The experiment VO users fall in two categories: a small number of production managers and a large number of unprivileged users. The latter category is split into physics groups. The production managers and the physics groups each desire a guaranteed quality of service (QoS) for optimal usage of the resources available. A better QoS could imply dedicated disk spaces or a higher priority in various request queues. Both CASTOR and dCache have various handles to give certain clients a better QoS than others, including:

- Space tokens
- Name space
- User identity/role
- Client IP address

The WLCG MoU for SRM v2.2 requires any data file to be stored into one of three Storage Classes:

- Custodial-Nearline (T1D0)
- Custodial-Online (T1D1)
- Replica-Online (T0D1)

Custodial storage is of high quality, i.e. tape, whereas replica quality need not be very reliable for long-term archiving, i.e. disk. The “Tn” (“Dm”) notation indicates the guaranteed number of copies of the file on tape (disk). A nearline class instance is managed by the system, whereas an online class instance is managed by the VO. The vast majority of the data will be managed by the system most of the time, i.e. reside in a T1D0 instance.

5. Reprocessing

Large data sets residing in T1D0 instances will have to be reprocessed, possibly multiple times, with improved reconstruction programs or better calibrations. Such data sets will be recalled from tape (i.e. pre-staged) sequentially, controlled by production managers. The files will need copies on disk only for the time it takes to reprocess them, after which the disk copies can be garbage-collected. If the production managers have sufficient amounts of dedicated disk space at their disposition, they can invoke the BoL and PtG functions with the desired lifetime (“pin time”), provided they have means to ensure the files end up in the dedicated disk spaces. Such means could pertain to any of the QoS handles mentioned earlier.

Another possibility would be to change the storage class of those files temporarily to T1D1, thereby guaranteeing the presence of the data on disk as long as needed (a BoL is still needed to cause the files to be pre-staged). Originally the T1D1 class was foreseen for data needed online for a long time and not to be garbage-collected when unused for a while, but it could be used for reprocessing as well (in order to garbage-collect the files again, a new ChangeSpaceForFiles is required).

6. To use or not to use space tokens on BoL/Get operations

In what follows we analyze the use cases introduced above and make some considerations on the consequences of recalling a file in a specific space or using the default space.

6.1 Tier-0 data handling

We have already stressed that for step 2 data need to be copied to a storage buffer instance that is connected to the WAN. This needs to be done using a BoL or Get operation. Since this operation happens at CERN, and since there is the possibility to associate such a buffer to a specific storage class, CASTOR will allow production managers to specify the associated space token to perform the operation. If the space token is not specified, CASTOR may deduce the buffer dedicated to this operation e.g. from the TConnectionType value (“WAN”) that ought be supplied by the FTS, or possibly from the user identity. Files need to be appropriately pinned for the time needed to perform the transfer, if a storage class of type T1D0 is chosen for the transfer. A ChangeSpaceForFiles can also be initiated to copy the files into the new buffer, assuming this has no consequences for the tape handling.

A third possibility is to recall the files into the default space. Default spaces can be defined per VO and per storage class type. Only one default space is possible for a given class; further separation of activities in the default space would be possible based on the name space, user identity etc.

Another copy of the files will be needed for the reconstruction operations described in step 3.

Also here the BoL/Get operations can be executed using a token. If a token is not used and the default space were selected instead, then as described earlier reconstruction operations can interfere e.g. with the data export activities. The ChangeSpaceForFiles request does not seem to be an option in this case, since the concurrent activities for steps 2 and 3 may each need their own copy of a data file, whereas ChangeSpaceForFiles moves a file from one space to another single space.

6.2 Tier-1 data handling

The same considerations made for the Tier-0 data operations are valid here. For Tier-1s it is very important to ensure independence of activities in terms of storage requirements so that one data handling activity does not interfere with another one of the same importance. A T1D1 class of storage could be used for step 2. However, at Tier-1s with dCache the ChangeSpaceForFiles method cannot be used for the time being, since it is not implemented yet and it will most probably not be available in April 2008. Therefore, an explicit copy operation would have to be used to copy the files in a T1D1 space for reprocessing (the BringOnline/Get operations in dCache currently ignore the space token passed, as per WLCG MoU). In dCache, it may not be possible for data copied into T1D0 to be recalled into the same buffer where it was written. In fact, Tier-1 dCache admins typically prefer write and read requests going to different buffers, to optimize the disk I/O performance. However, pinning is honored. Therefore, data written into such buffers can be pinned for a time long enough for reprocessing to happen efficiently. Pins can also be renewed.

6.3 VO data handling

This is the most chaotic activity since physics groups will most likely compete with each other in terms of space requirements. Data will be mostly written in T1D0 space at Tier-1s and T0D1 at Tier-2s. In this case, when a token is not specified, recalled data can end-up in a “default” area or be served from a pool that is not the one dedicated to that physics group. In dCache there is no possibility to deny read access to a copy of the file to a set of users. In CASTOR this possibility exists.

During CCRC08 in February and May 2008, such activity will most probably not be exercised. However, the following recommendations are made.

For dCache, tokens and the namespace must be used by users as it is done for production activities. Sites will be configured to handle user requests in an optimal way.

For CASTOR recall requests from random users (of which there should be few) must not provide a token. Such requests will go to a small default pool which is managed by the site, thus avoiding the risk that either

a) user requests are refused if they are not allowed to write data to the given storage area (they won't be allowed write/recall access to a disk1 pool, for example), or
b) that CASTOR needlessly copies data that is already on disk and accessible to a different storage area, thus wasting disk space.