



Unit VII - More realistic SRM scenarios

prepared by

Martin Radicke
&
Owen Synge

GEFÖRDERT VOM

- to start this unit from a clean state, either
go back to the snapshot from yesterday

OR

get a fresh VirtualBox image from

<http://trac.dcache.org/trac.cgi/wiki/dCacheToGo>

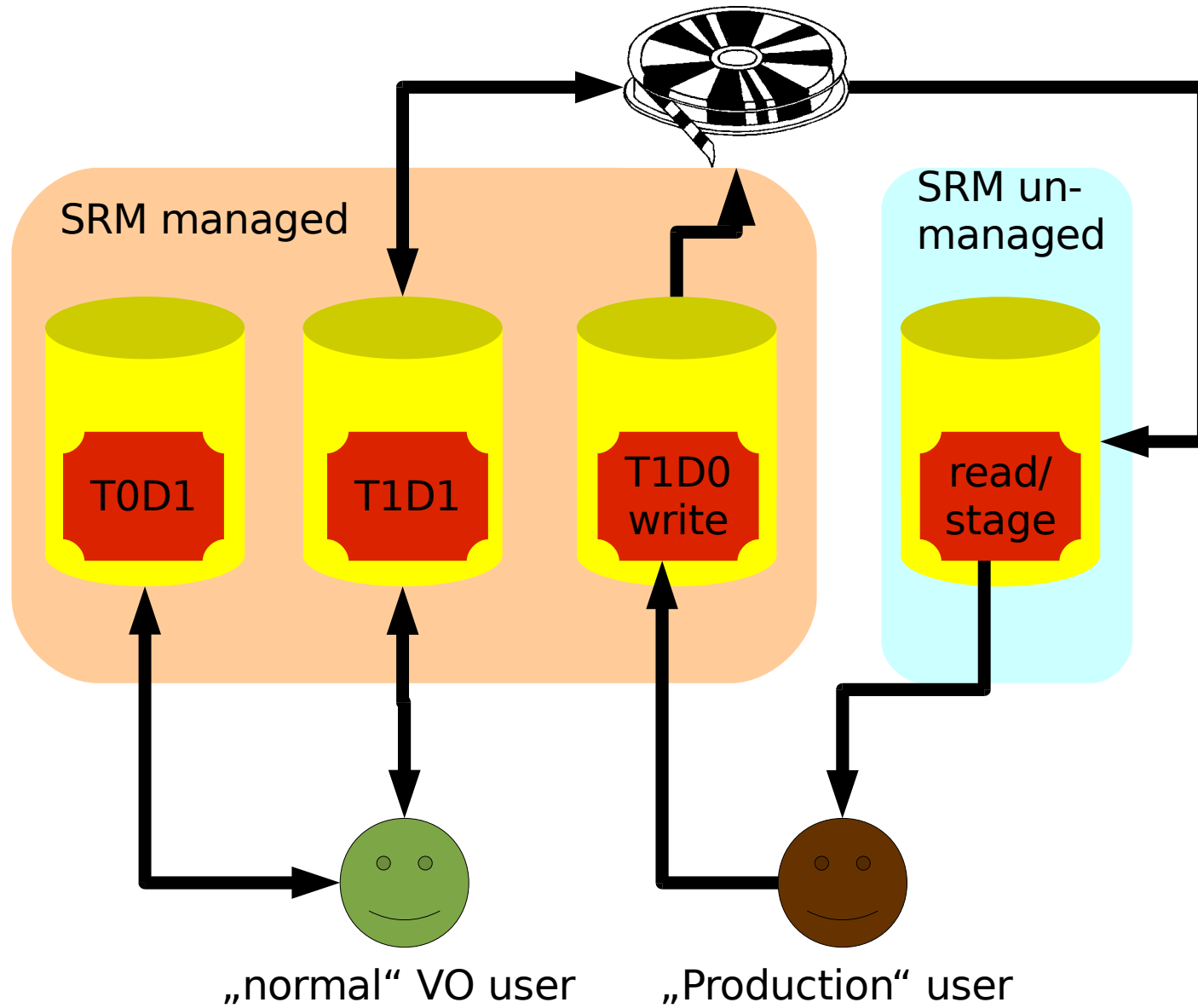
- more complex scenarios as required from experiments in WLCG
 - all 3 different types of spaces simultaneously
 - role-based authorization
 - dCache connected to tertiary storage (aka tape), allowing files to be custodial

A tape backend inside a VM ??



- you will connect some of the pools to a **fake HSM**
 - a script, which acts as an HSM, but only pretends to flush files to/stage from Tertiary Storage
- **WARNING**
 - the files won't be copied anywhere due to the limited hard disk space inside the VM
 - on staging: files will be recreated with the correct size, but consist only of Zeros!!!

The 3 space types



- „Kermit“ + „/desy/Role=NULL“ → desy01
 - T0D1, T1D1 spaces authorized for „normal“ VO member mapped to local user 'desy01'
- „Kermit“ + „/desy/Role=production“ → desy02
 - T1D0 space authorized for and writeable only by 'desy02'

- create a linkGroup with attributes **REPLICA/ONLINE**, attach one pool
- write into it using implicit *space reservation*
- make sure the file went to the correct pool

Task 2: Attach the fake HSM



- edit the pools to link them with the HSM script
- write a file to one of those pools (still unmanaged space), flush it manually to „tape“
- remove the file from the pool's disk, stage it back in
- automate the full life cycle:

write file → file get's flushed → remove file from pool → read file (triggers stage on demand)

- create a **CUSTODIAL/ONLINE** linkgroup, one pool
- reserve a space token, srmcp into it
- watch the file on the pool being marked as
 - „Cached“ - safely flushed to tape
 - AND „Pinned“ - file will never be removed from pool disk

- create a **CUSTODIAL/NEARLINE** linkgroup
- one pool for write/flush only, a 2nd one for stage/ read only
- this space writeable only by production user
- watch the movement of files between pools
 - read the file back after writing it into this space
 - triggers pool2pool transfer write-pool → read-pool
 - delete the cached copy on the write-only-pool, then read
 - stages the file from „HSM“ to the read-pool