A milestone for DPM

A summary for CHEP 2018

Intro - DPM status

- DPM is one of the most popular storage systems for Grid computing
- The last release is 1.10, released in June 2018
- The infosys says: ~87PBs in total, provided to Grid computing. ~112 instances.
  - DPM lost a tail of tiny sites, while the overall storage capacity continued to grow (was ~70PB at the end of 2016)
  - Several sites larger than 2 PB, 20 sites larger than 1PB. The largest so far is 6.5PB

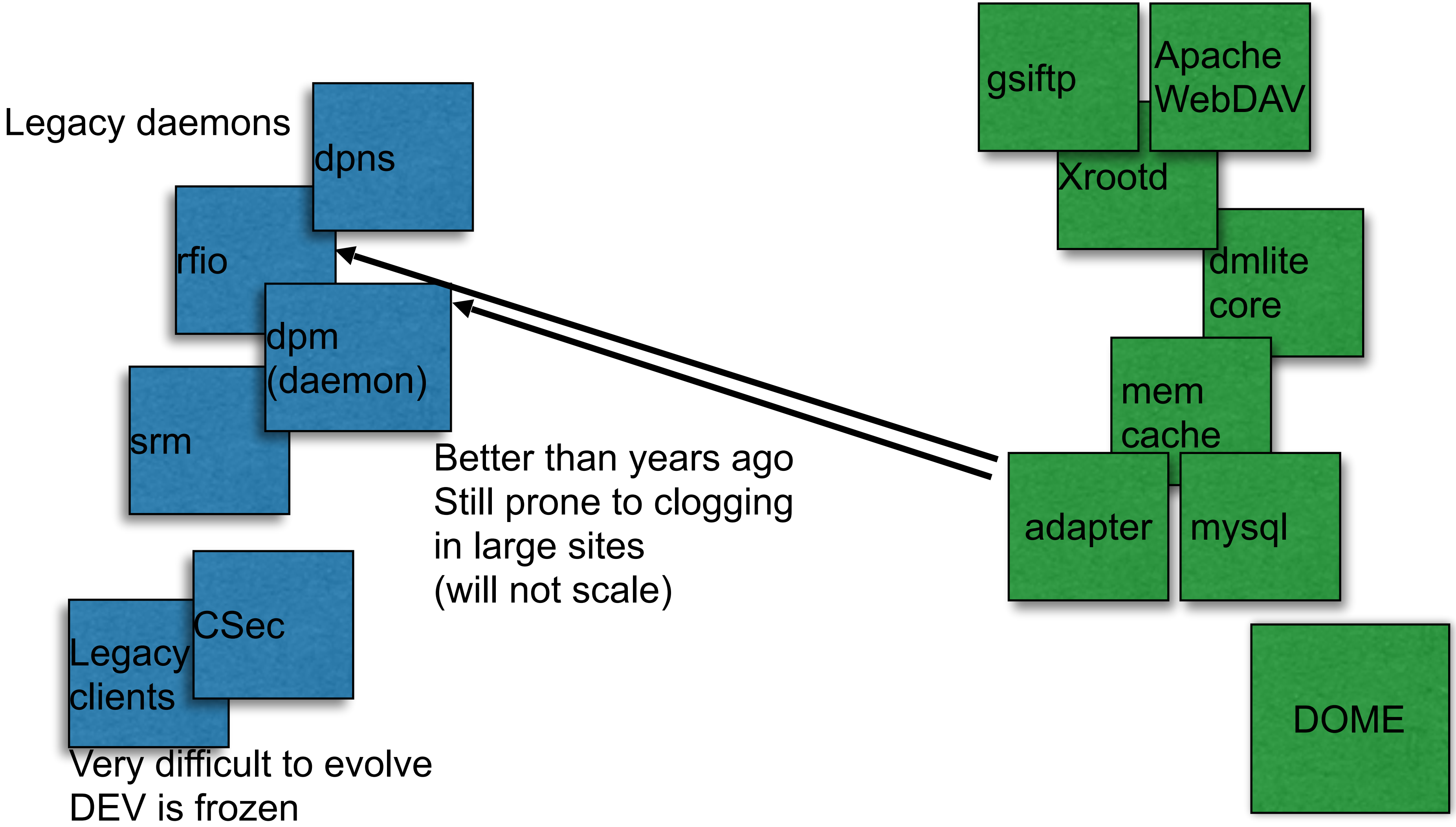
- Our focus continues to be on
  - Consolidation, keeping support and sysadmin cost at the lowest
  - Performance, scalability, current and future WLCG trends
  - High quality HTTP, WebDAV, Xrootd, GridFTP support
  - Support. In touch with sysadmins as much as we can
- What follow is an extreme synthesis of various topics. For more details please refer to the recent DPM workshop (May2018): https://indico.cern.ch/event/699602/

Main direction

- Manageability and long term support of the DPM system. Includes dev, maintenance and support
  - Benefit both the sites and the DPM team
  - This includes helping sites to adopt the “tuning hints” and the command-line puppet setup

- Clean, straightforward system based on open, well documented contemporary technologies
  - Facilitate sites to follow the next trends (e.g. bearer tokens, space reporting, …)

DPM components and plugins (2018)



Legacy daemons

dpns

rfio

dpm (daemon)

srm

Better than years ago
Still prone to clogging
in large sites
(will not scale)

CSec

Legacy clients

Very difficult to evolve
DEV is frozen

gsiftp

Apache WebDAV

Xrootd

dmlite core

mem cache

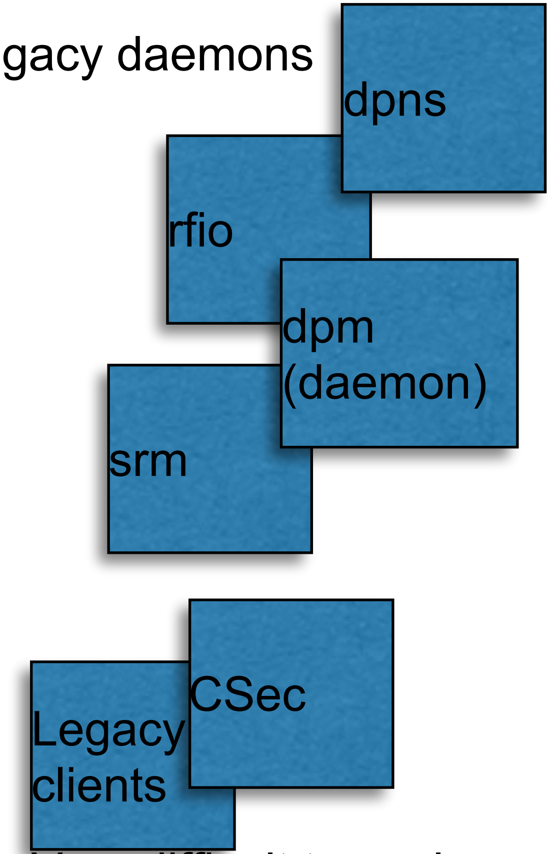
adapter

mysql

DOME

DPM components and plugins (2018)

Legacy daemons



dpns

rfio

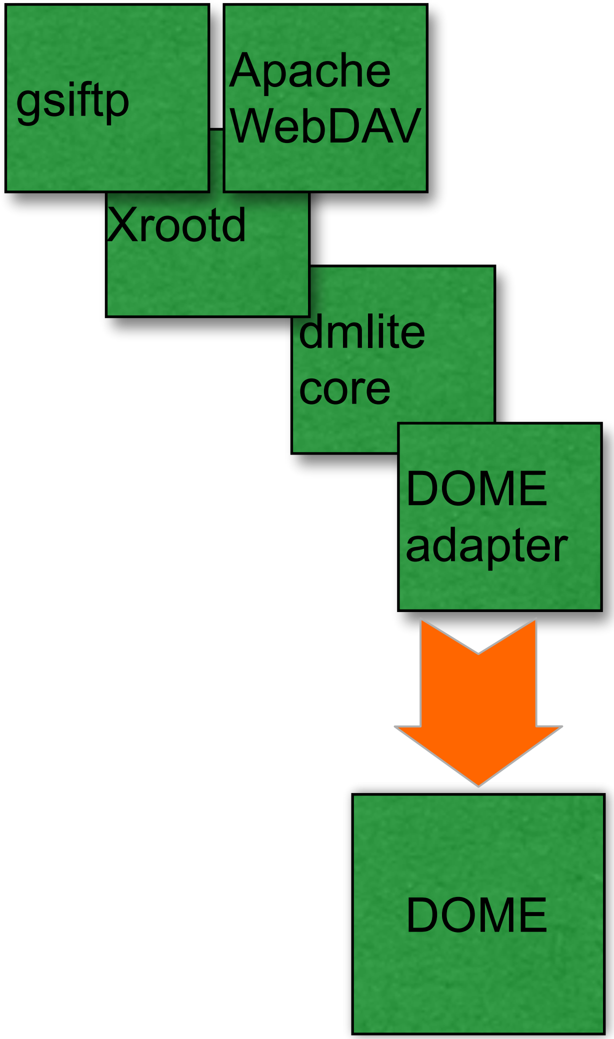
dpm (daemon)

srm

CSec

Legacy clients

Very difficult to evolve
DEV is frozen



gsiftp

Apache WebDAV

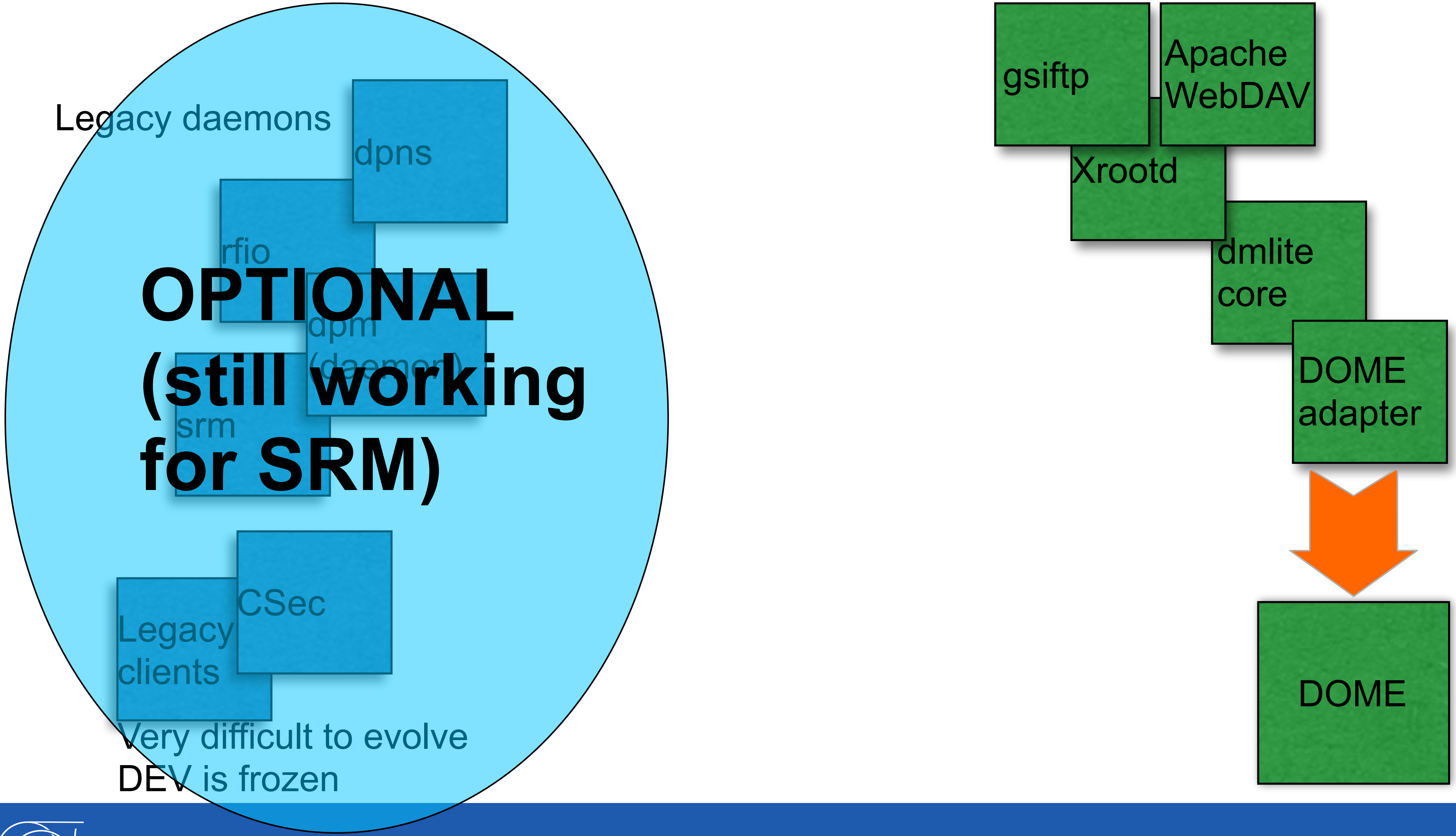
Xrootd

dmlite core

DOME adapter

DOME

DPM components and plugins (2018)



Legacy daemons

dpns

rfio

dpm (daemon)

srm

OPTIONAL (still working for SRM)

CSec

Legacy clients

Very difficult to evolve

DEV is frozen

gsiftp

Apache WebDAV

Xrootd

dmlite core

DOME adapter

DOME

Legacy and non-legacy (DOME) mode

- Legacy mode is when DMLite loads the Adapter+Memcache+MySQL plugins
  - The good old DPM daemon does the coordination work
  - Every process loading dmlite (httpd*4, gridftp, xrootd) needs a new pool of MySQL connections

- Non-legacy mode is when DMLite loads DOMEAdapter
  - DOME does the coordination work and talks to mysqld
  - DOME does disk server status detection (up/down/space)
  - The DPM daemon coordinates only itself and SRM
  - Only one internal MySQL pool is used for dmlite

One plugin to rule them all

- **In non-legacy mode** DPM now loads only DMLite::DOMEAdapter
- dmlite-adapter, dmlite-memcache, dmlite-mysql are no longer necessary
- Resource consumption (FDs, mysql, etc.) is reduced by an order of magnitude, and so complexity and cost for us all

DOME speaks REST and JSON

- DOME and its companion DOMEAdapter cover the functionalities of dpm+dpns+rfio (plus others)

```
GET /domehead/command/dome_getstatinfo
    HTTP/1.1
User-Agent: libdavix/0.6.8 neon/0.0.29
Keep-Alive:
Connection: Keep-Alive
TE: trailers
Host: dpmhead-trunk.cern.ch:1094
Content-Length: 17

> Body block (17 bytes):
{ "lfn": "/dpm" }
```

```
HTTP/1.1 200 OK
Content-Length: 250
{ "fileid": “3",
  "parentfileid": “2",
  "size": “265623786530",
  "mode": “16877",
  "atime": “1523455123",
  "mtime": “1522229608",
  “ctime": “1522229608",
  "uid": “0",
  "gid": “0",
  "nlink": “2",
  "acl": “A70,C50,F50,a70,c70,f50",
  "name": “dpm",
  "xattrs": "{\"type\": 0}”
}
```

Quotatokens

- DPM with DOME abandons the older concept of "writing into free-space" in favor of a more precise model based on **directories and spacetokens**.
- **We give disk space to directories by attaching a spacetoken to them. We called this "quotatoken"**

- The sysadmin looks at the already existing space tokens, and assigns each of them to the corresponding directory subtree
  - They will "give space to write" to that directory
  - They will be used as a quota too
  - **Needed to generate precise dir-based space reports (ATLAS)**
  - The scheme is backward-compatible

- For more information: https://indico.cern.ch/event/699602/contributions/2941791/

Volatile pools and caches

Volatile pools and caches

- In 2016 they were announced as a wish, they are available now in 1.10
- Marking a pool as “Volatile” triggers the cache-like behaviour **for that pool**. Other pools work like before.
- **A full-file site data cache that works seamlessly and interchangeably with all the data protocols: HTTP, Xrootd, GridFTP**
    - SRM can’t

- File pulls are queued and scheduled, no space for “storms”
- External stat() and file pulling are implemented by two customisable scripts. DPM can pull files from any other remote or local system

- Being deployed and tested in INFN-NA https://indico.cern.ch/event/699602/contributions/2953001/
- The mechanism has proven to work fine, extensions may be possible once the need of sites and experiments are better known

DPM multi-site (plus cache)

- A distributed DPM setup has always been technically possible
- DPM pools can be deployed in different sites, acting as satellites of a main one

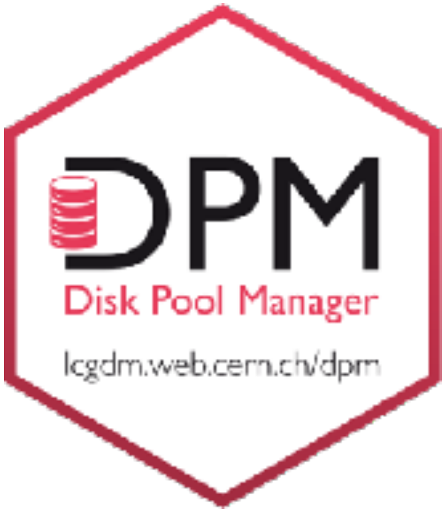
- The older components (libshift, rfio) can pose challenges, solvable on the firewalls and the configuration

- Alessandra Doria (INFN-NA) reported on the distributed setup deployed between Naples, Rome and Frascati. Also Francesco Sciacca(UNIBE) did a similar deployment in Bern
  - https://indico.cern.ch/event/699602/contributions/2941786/

- In DOME mode the setup becomes simpler, as there’s no libshift and rfio anymore. On top of that it supports volatile pools (caches), a remote pool-only site could be configured as a cache belonging to a “main” DPM head

A new logo/TWiki for DPM

- Breaking news, almost finished with the new logo



- We will also migrate from TRAC to TWiki
- Right now the new TWiki is almost complete (except for the logo…)
  - https://twiki.cern.ch/twiki/bin/view/DPM/WebHome

Looking forward

- **We got an invitation from the Bern site for the next DPM workshop, Spring 2019**
- Technically, we see no core changes at the horizon
- A few peripheral additions (e.g. xrootd checksums), and some package/build tree refactoring to further reduce the maintenance cost

- The "site consolidation" (less tiny sites, big ones become bigger) will likely continue
- When the SRM load reaches a certain level, larger sites will be more tempted to use it less, or drop it
  - Our effort has been towards making this possible, well working and well documented
  - The sysadmin now has the choice, big milestone

LCGDM support from 01/Jun/2019

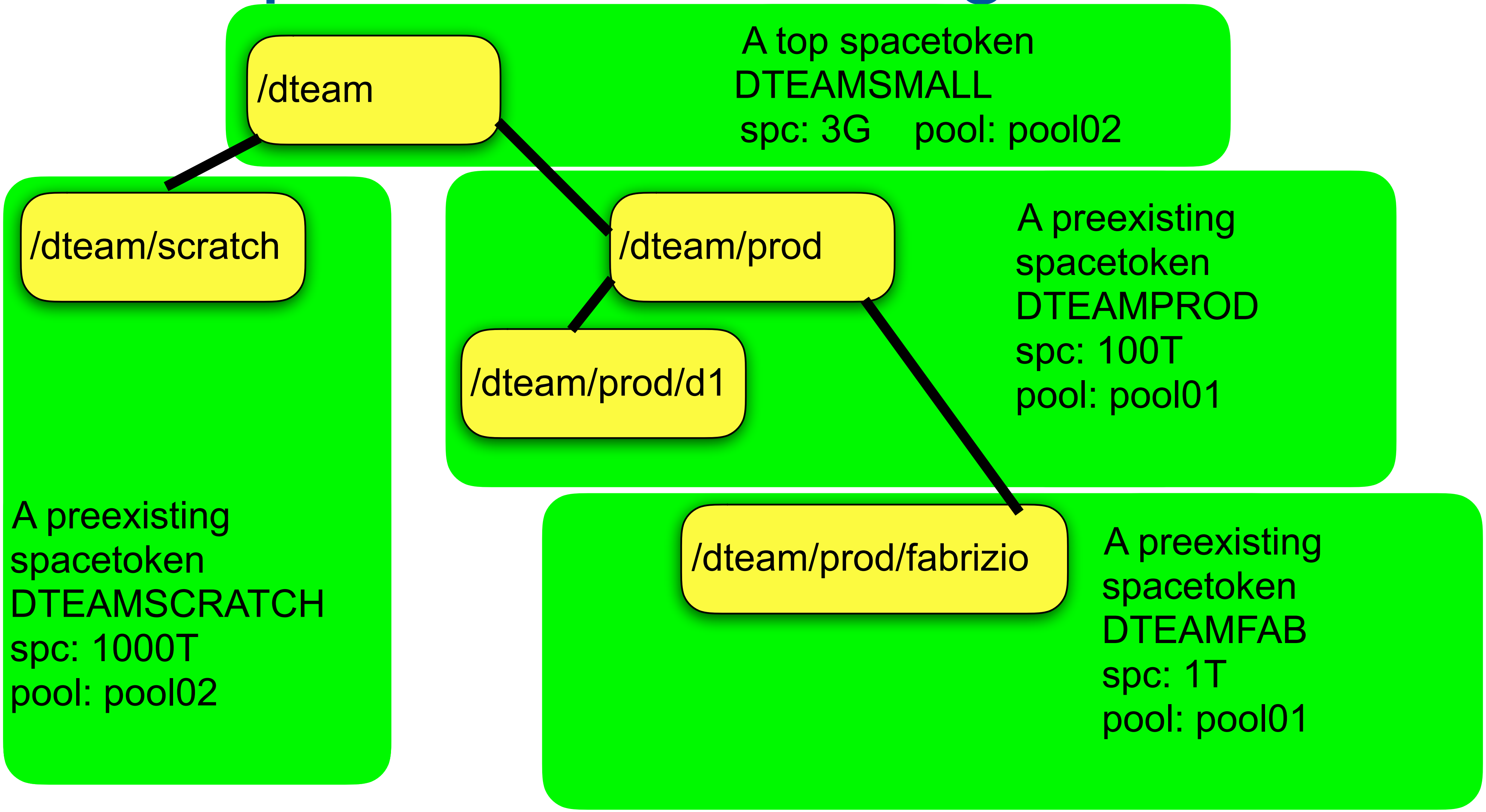
- From **1st of June, 2019** our standard LCGDM support answer will be “**there is an alternative: upgrade DPM to DOME flavour, please**”
  - A large part of DPM support requests is about dpm+rfio clogging, these problems will instantly be reduced and simply disappear as the SRM usage decreases

- LCGDM will stay in EPEL as long as it compiles untouched in Rawhide (EPEL rules will remove it if it breaks)
  - It’s pure C code, hence that can be years, we don’t give limits

- LHC Tier-2s in general can work without SRM, at least there are recipes to do it
  - e.g. for ATLAS: https://indico.cern.ch/event/699602/contributions/2944281/

Spare slides

Example dir/ST/QT diagram



/dteam

A top spacetoken
DTEAMSMALL
spc: 3G pool: pool02

/dteam/scratch

A preexisting
spacetoken
DTEAMSCRATCH
spc: 1000T
pool: pool02

/dteam/prod

/dteam/prod/d1

A preexisting
spacetoken
DTEAMPROD
spc: 100T
pool: pool01

/dteam/prod/fabrizio

A preexisting
spacetoken
DTEAMFAB
spc: 1T
pool: pool01

LCGDM - the legacy stack

- The good old LCGDM has given unprecedented service to the community
- It contains components (e.g. libshift) that are more than 30 years old
- These components have played a big role of the history of CERN data management, including various CASTOR generations
- The DPM SRM daemons are there, and have pioneered the Grid
- A few particularly unhappy choices (e.g. imake, or SEDding the code while compiling it, or an outdated approach to TCP/ threads) made life difficult
- Lots of glory, and very problematic to maintain nowadays

The fastCGI saga

- The first version of DOME (2016) used to run as a fastCGI daemon under Apache. In 2017 the performance started to suffer.
- The best hint I had from the forums for the bad fastCGI regression is that it may be related to requirements of PHP programmers
- Hence the “natural” solution for them was to disable the fastCGI connection reuse and the overlapping requests directly in the code of the Apache module
  - Connection reuse with mod_proxy_fcgi is broken and will very likely stay broken. Moreover it’s very expensive to debug

- Result: performance lower than 100 transactions per second, with very high resource consumption (hence higher instability). Almost worse than SRM.
- I (FF) have wasted one month full time on this, around Feb/March 2017, and then found a solution to wipe all this

fastCGI… other options ?

- Nginx surely fits the use case
  - Its community seems certainly more performance-aware than Apache's
  - Who wants one more daemon technology in the head node?
  - A new framework to learn and write low-ish level software for
  - More setup hassle for sysadmins and puppet gurus

- The Xrootd framework has an HTTP interface: XrdHTTP
  - Well known by our community
  - Designed to be lightweight
  - Provides a pragmatic API to provide extensions
  - Every WLCG site already has Xrootd

XrdHTTP

- XrdHTTP is the HTTP/WebDAV protocol implementation of the Xrootd framework.
- Allows extensions (e.g. new HTTP verbs) through a straightforward C++ plugin interface

- Additional bonus: Brian Bockelman selected XrdHTTP for implementing SciTokens and HTTP third party copy plugins
  - The CGI interface received several small enhancements, and it’s ready for the next generation Grid storage authorization schemes, following modern standards

XrdHTTP

- Porting DOME to XrdHTTP took 2 days, ~50 lines of code and I never looked back. This was April 2017, the test systems are under massive test since then

- Apache stays for the data access, like before
- DOME does internal metadata and coordination and resides in the Xrootd process

- Result: the metadata transaction rate is now more than 10KHz in our test machine
- A massive stat() test on DPM/DOME now rates ~9-10KHz, stable
- Incredibly, there's still space for improvement, and I don't think we need it now

- (The author of lcgdm-dav proposed to start using XrdHTTP also for the data, instead of keeping Apache and lcgdm-dav (= cost). Surely it will be cheaper, I am not yet convinced, as lcgdm-dav works reasonably well)

Be prepared

- Our goal is to reduce support for components that can't cope with the increasing requirements of WLCG computing
- The older LCGDM component (srm, dpmd, dpns, rfio, CSec, libshift) received some enhancements 2-3 years ago, and can survive some time
  - Its performance may not scale in larger sites
    - (= more client timeouts)
  - It won't support the directory-based space reports (in particular the free space)

- The future-proof solution is to upgrade, activate DOME and incrementally remove the use cases for the older LCGDM components (rfio, SRM, dpns, …)