Current project status

Fabrizio Furano
DPM workshop 2019

Direction: stability and LTS

- No revolutions since quite some time
  - BDII reports 102 instances, 96PB
- The architecture is definitive and stable
- All the past and present efforts are aimed at giving long term support and longevity
  - Longevity: can adapt to the future required features (e.g. scitokens, OIDC, caches, macaroons, …)
  - Long term support through stability. We expect relatively little fixes and improvements
  - Most of them related to usability and UI

Direction: stability and LTS

- This is actually the goal seeked several years ago with the idea of DMLite and of the “DPM collaboration”
  - A healthy open source project
  - Understandable by others willing to cooperate and contribute
- IMO this has been among the best accomplishments

- The explorative project took more than expected, the results have been good in our view
- And this is the reason why we did…

LCGDM support from 01/Jun/2019

- From **1st of June, 2019** our standard LCGDM support answer will be "**there is an alternative: upgrade to DOME flavour, please**"
  - That affects: dpns, dpmdaemon, rfio, CSec, dmlite::Adapter, SRM
- LCGDM will stay in EPEL as long as it compiles untouched in Rawhide (EPEL rules will remove it the day it breaks)
- It's pure C, hence that can be even years, we don't give limits

Breaking news - security support

- [4th of June] CERN and EGI agreed on postponing the deadline for security support for the legacy codebase to end of September

- As DPM team we agree it's important, and we don't expect troubles from the core legacy components in CC7 or EL6

- Also the external older components (e.g. globus, gSOAP) are pretty stable

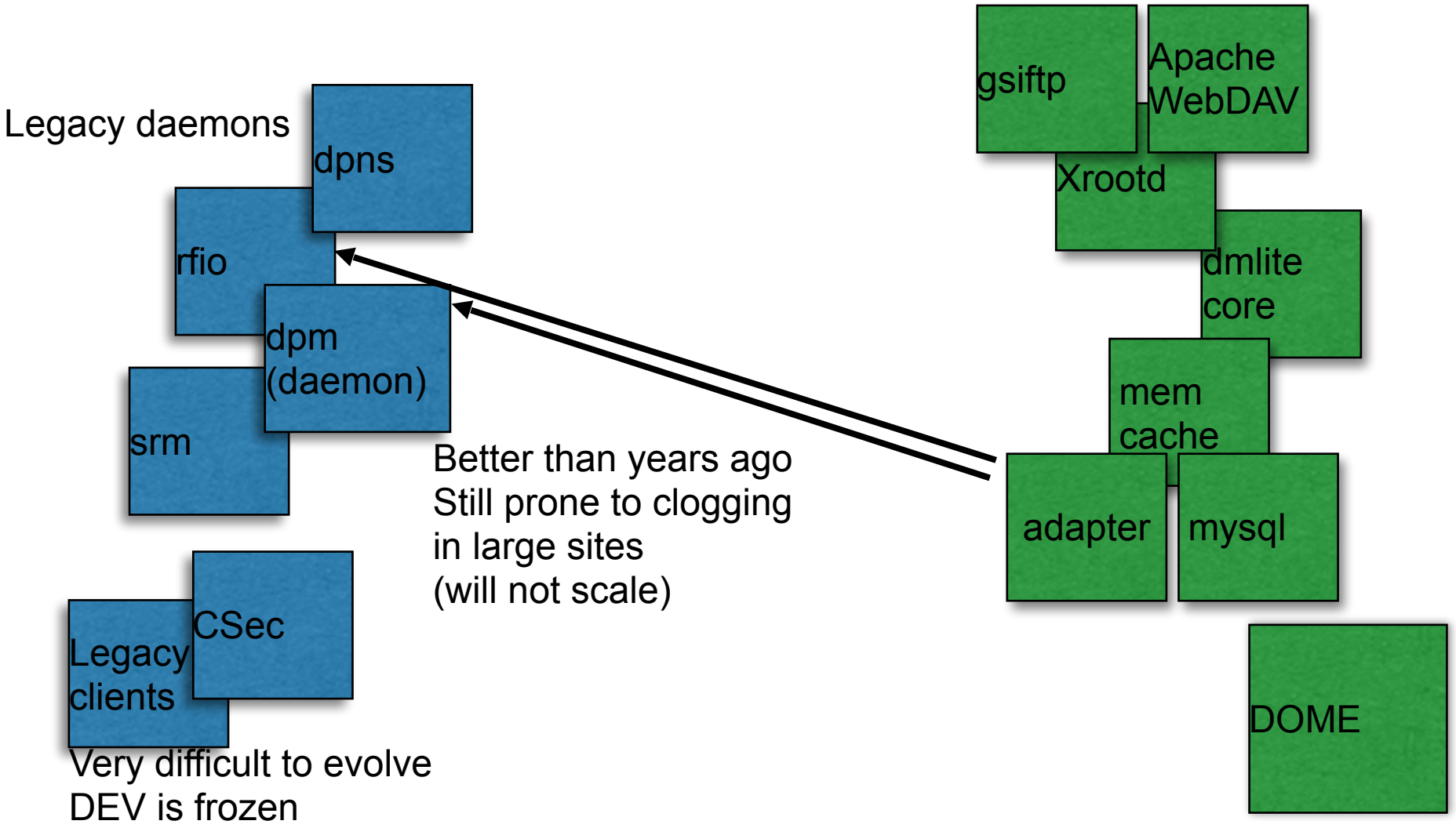
Summary: 3 DPM setup flavours

- Full legacy: dpmdaemon+dpnsdaemon+rfio+srm
  - DMLite loads the Adapter plugin to give http+gridftp+xrootd
  - The DOME daemon runs in the head node and does basically nothing (dormant)
  - gridftp can be used only through SRM
  - **This is the status of many sites…This legacy setup cannot scale up**

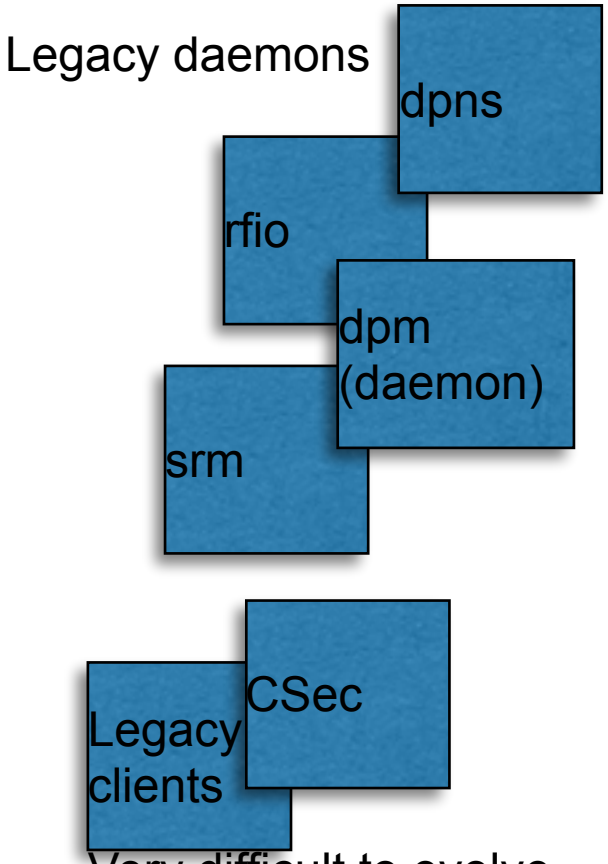
- DOME plus legacy: legacy dpmdaemon+dpnsdaemon+rfio+srm plus DOME supporting http+gridftp+xrootd
  - DMLite uses DOMEAdapter, legacy part stays legacy and does only SRM+rfio
  - gridftp can be used through SRM and directly with a gridftp client towards the headnode (gridftp redirection)
  - **This setup can scale up the number of servers, the max transaction rate (http/xrootd) is more than an order of magnitude higher than the legacy flavour**

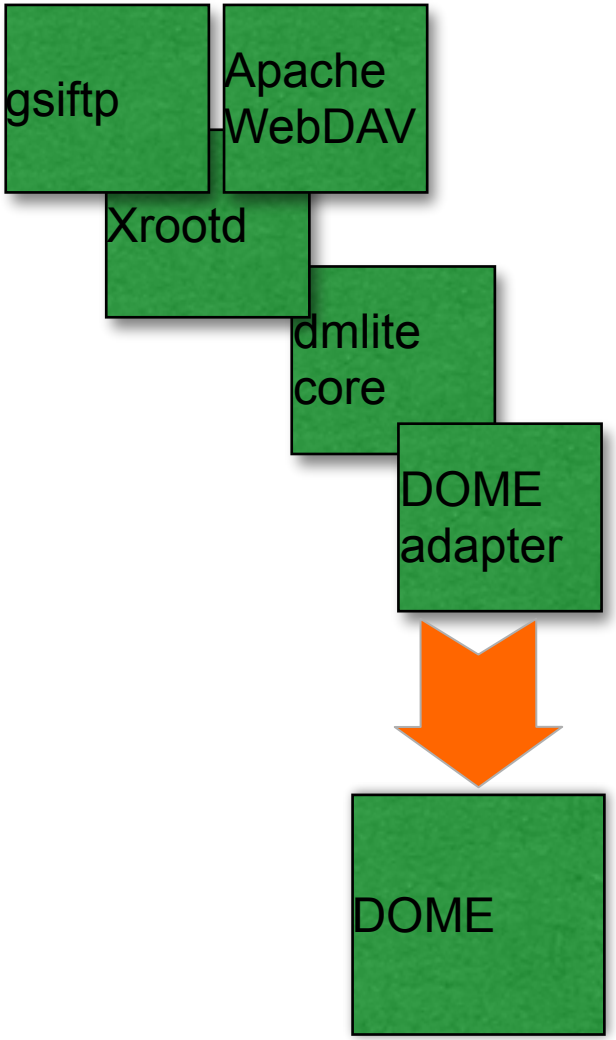
- DOME without legacy: take the previous option, stop and/or uninstall srm,dpmdaemon,dpnsdaemon and rfio.

DPM components and plugins (2017/2018)

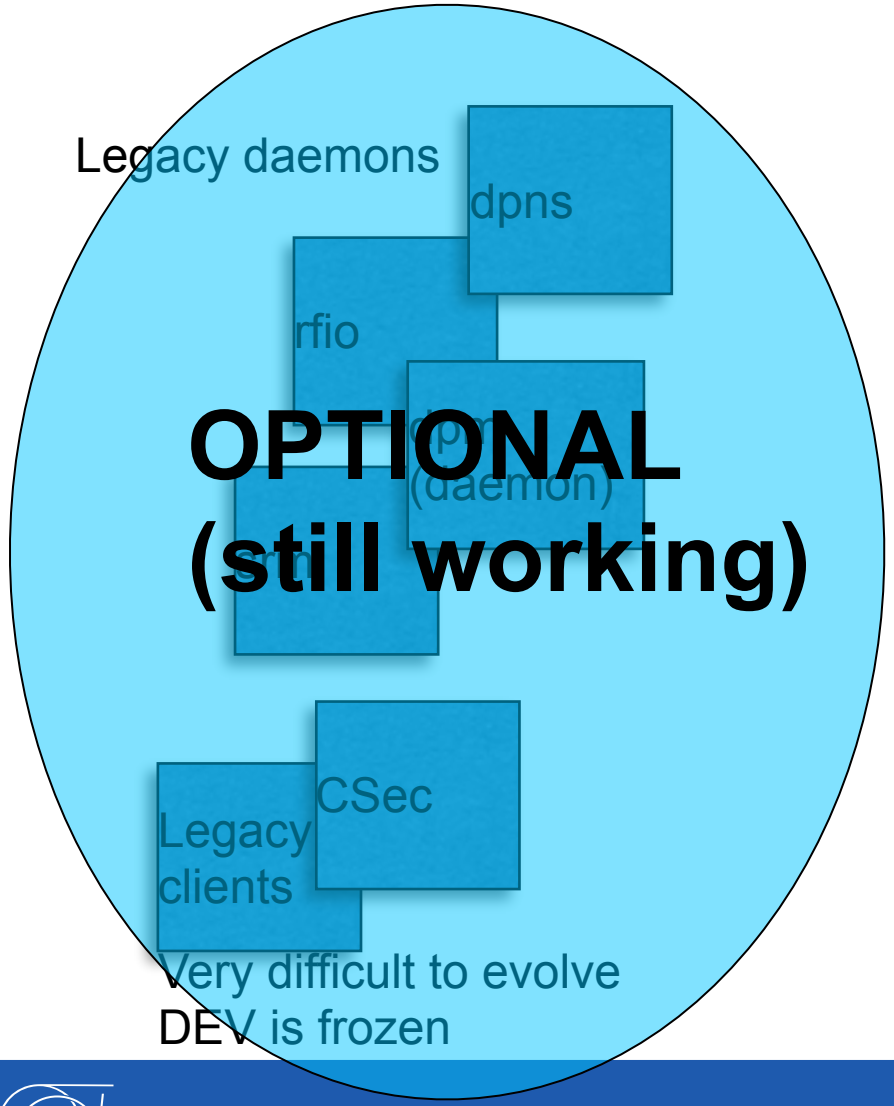


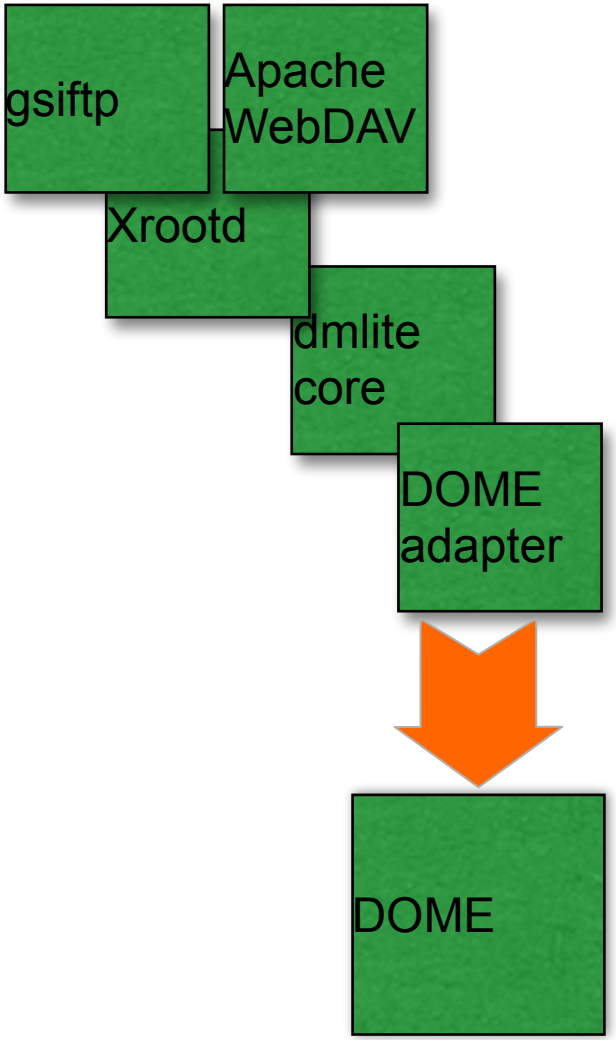
DPM components and plugins (2017/2018)





DPM components and plugins (2017/2018)





Can the 3 flavours be mixed ?

- My opinion… **better to avoid this kind of headache**. The puppet setup is quite simple and helps a lot keeping things aligned.

- Can I use headnode at 1.12 and **all** disk servers at 1.9 ?
  - In theory yes (modulo bugfixes), remind that Apache/fastcgi in 1.9 has proven to be not well supported. **So… why?**

- Can I use headnode at 1.9 and **all** disk servers at 1.12 ?
  - In theory yes, (see the previous one). **So… why?**

- Can I mix disk server versions >= 1.9 ?
  - Problem: the historical dpm_db does not allow to specify different port numbers to control different filesystems
  - Hence the older filesystems should be moved to port 1095
  - At this point upgrading them is easier. **So… why?**

- Can I mix disk server versions <= 1.9 ?
  - NO, the DOME mode needs DOME in all the disk servers

1.12.1 since end of March 2019

- Remarkably stable and fast (modulo SRM/rfio of course)
- disabled filesystems now allow deletions and drains
- checksums fixes/optimizations
  - 1.11 had the default parameters used for manual debugging, ridiculously low, so it did not work well out of the box under real load
- The xrootd checksum support is there (was linked to the availability of xrootd 4.9)

The Brunel case

- Brunel used to be configured in legacy mode (like most sites, normal), at the limit of the possibilities of the old components, a few hundred Hz
  - Likely also due to the traffic coming from AAA and from the apparently aggressive CMS workflows

- A disk server broke, and caused increased metadata load (stat requests, failed opens, repeated, …)
  - The legacy components (dpm daemon, dpns) started blowing up. Crashes and lockups like never seen, headnode load at >100

- The only possibility was to enable DOME, which was tested up to more than 10KHz (only because we don't have that many resources to go higher)

The Brunel case

- Brunel experienced quite some crashes, always the same case, solved with a fix to DOME in some days. David Smith's idea worked.
- **Then, a couple of minor things and some experience in choosing the default parameters led to the current 1.12.1 version**
- **We have to thank Raul for his proactivity and will to cooperate, now the headnode load is a fraction of what it used to be**
  - And the transaction rate is around 1.5KHz. Big success.
  - And let's not forget that the DPM transaction rate is not theoretical. It's end-to-end involving clients, disks, head, DB, cache, remote lookups, …

Contributions

- Among the best outcomes of the last year, we have seen very welcome contributions, also to the DOME core, e.g.
  - David Smith was able to understand it very quickly just by reading the code
    - And spotted the missing lock that was causing instability under production load
  - Petr Vokac contributed a wealth of fixes to many things that make the system better. Big thanks for his many contributions and will to help others in the forums
  - Others made useful contributions and comments on the setup templates and other parts

- Well done! There are things that the testbeds simply cannot see.
- Thanks to all who have taken some time to understand the components, proofread the docs and troubleshoot stuff

- This actually plays in favour of DPM being a healthy open source project

The Globus-gridftp bug

- A DPM core that is more solid than the past highlighted some suboptimal older things, e.g. the globus-gridftp race condition.
  - Globus has a ‘bug’, after a gridftp upload it says OK to the client before saying OK to DPM
  - If the client is fast, it may not find a new file while it’s being closed in the DB
  - This gives a certain (low) rate of job failures for non-SRM gridftp uploads. Happens since ever, luckily this hiccup appears not to be so frequent
  - Note: So far it has been hidden by the SRM workflow
- Years ago we had opened a ticket to Globus, never had any response AFAIK
- We don’t see hooks for improving this, apart from using xrootd or http (see DOMA-TPC) instead of gridftp

Gridftp redirection and lcg-utils

- lcg-utils is unsupported and deprecated since ~5 years, yet still used by some legacy Grid workflows

- Functionally lcg-utils still works with DPMs, performance is very poor with gridftp without SRM
  - gridftp has to tunnel the data, hence consumes 2X the LAN bandwidth
  - to avoid being a bottleneck, the headnode 'redirects' the client to a random server that must tunnel the data to the right one
  - 'random' also means that it can be disabled or broken [well, better than nothing, will be improved in 1.13]

ABI/versions status

- DPM is 1.12.1 (soon 1.13). That means a coherent set of libs popping out of the build system, at once. The lib coherency is almost complete, and will make the last small step with 1.13
- Same solid approach to the source code used in the xrootd project, or in ROOT

- Eliminates the hassle in understanding if libA versionX works fine with libB versionY. Impossible to test all the combinations, with many libraries
- Reduces the cost of EPEL pushes, simpler pushes also make longer term support easier

- Last step in that direction: dpm-dsi (the gridftp plugin) was moved into the dmlite source tree recently (1.13)

ABI/versions status

- In theory two components are missing
  - lcgdm-dav: the HTTP frontend. A bit more complex to migrate its source, because of the Apache deps
    - Moreover it has to carry around the last version of curl (infamously buggy in the regular distros), linked statically
  - dynafed (UGR). These makefiles are complex, with its own set of plugins.

- I may have a look at relocating lcgdm-dav, but not dynafed
- The situation is pretty good

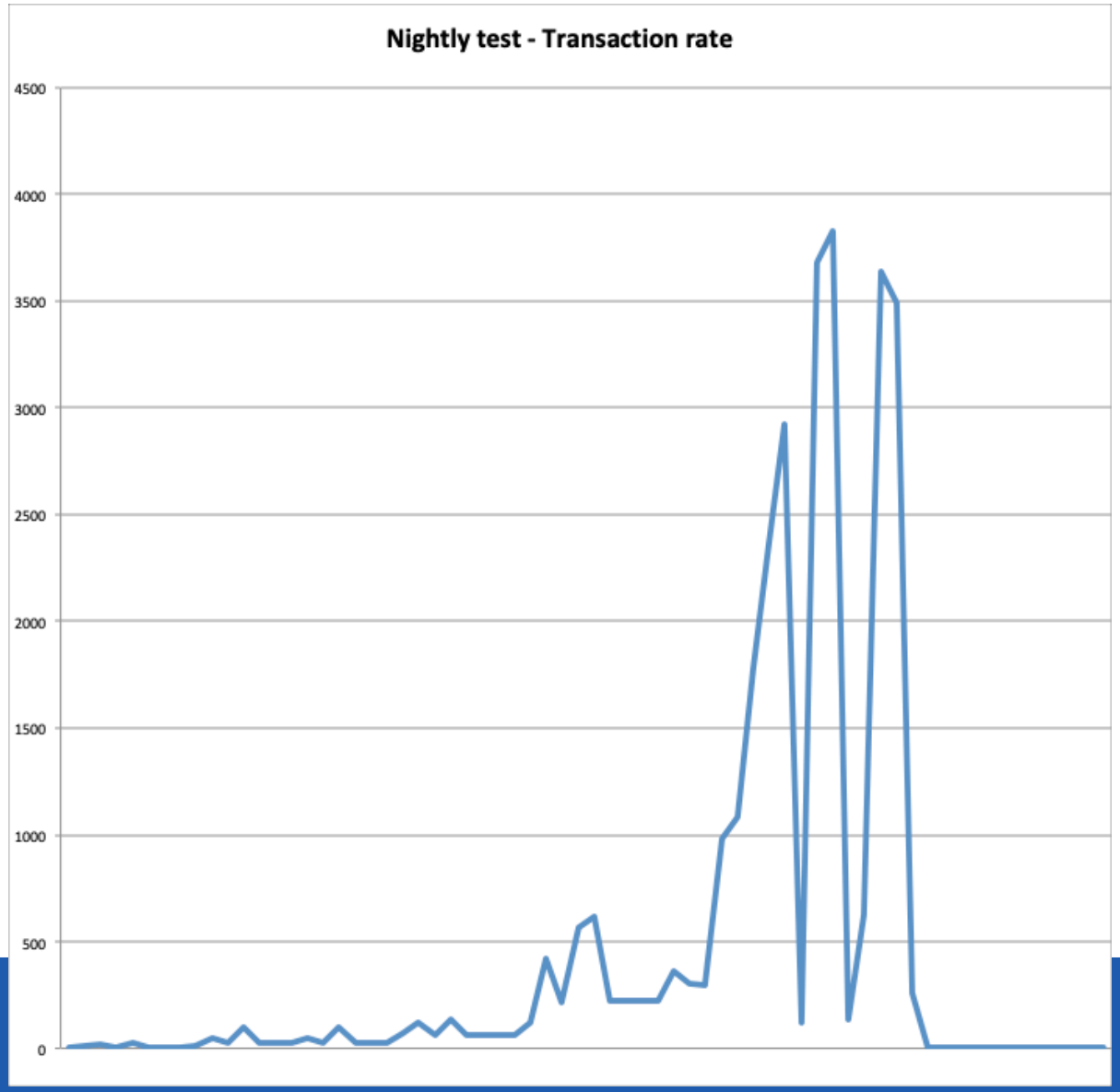
Stats on transactions

- Since DOME has a central place, calculating realtime stats becomes easy
- DOME outputs some performance stats every minute into the log
- Example:

```
Request rate: 113.938Hz (Peak: 567.2Hz) -- DB queries: 30.8154Hz -- DB
transactions: 46.1538Hz -- Intercluster messages: 15.3846Hz
```

- Loading them in a spreadsheet is straightforward, here's a quick graph from the nightly tests
  - Almost 4KHz from a single client machine, head is a poor 2-cores VM

Performance graph (Excel)



Information system - BDII

- We use it to count DPMs, since ever, and make a monthly report
- Its content is extremely useful, how else to count DPMs ?

- Will it stay? EGI says yes, what about WLCG? We don't know.

- Alternative: using the cmd dome_info on a list of hosts
  - Coming from where? Maybe GOCDB? Is it reliable/up-to-date?
  - Needs access to port 1094/xrootd, which may or may not be allowed
  - dome_info is quick, gives useful super-basic information (e.g. the version), and then closes the connection. No agents needed.

Deployment monitoring

- DOME 1.13 (the next minor release) can send existence information to external HTTP(s) hosts
- So far it's an experimental feature
- The sysadmin can add/remove them
- Setting up an HTTP server to collect these is a trivial effort
- The default string contains version, host, time, space, free
- We haven't yet discussed a default destination
- Load information can be added (by default off)

```
dome=1.13.0&host=dpmhead-trunk-new.cern.ch&t=1559638667&tot=344344621056&free=184762572800
```

Cache mode - Volatile pools

- There since Q1/2018, works interchangeably with all the protocols (modulo SRM)
- INFN-NA has an advanced testbed

- Functionally it's quite solid and well integrated in the idea of the DPM pools
- It's a full-file buffer, AFAIK more than sufficient to give the 'cache experience' (and transferring the file at the first access is way quicker than tunnelling all its chunks with some latency)
- Supports pre-populating by construction, can also be written into normally

- If/when there is any content that is worth caching we will be able to understand if its cache purging algorithm is good enough
  - I would be in favour of improving it, IF it's useful to some clear use case and IF we can document that a different algorithm would make a meaningful difference
  - Until then, the current purging algorithm is fine and the feature can be used normally

Remote pools - disk-only sites

- Theoretically it has always been possible, yet quite tricky with libshift (among the oldest components from CERN IT!) and rfio (not much younger)
  - Without forgetting firewall rules and reliability

- This workshop was triggered by Gianfranco Sciacca, who one day popped out and said “do you know I did that in production ? it works”
  - Hence, more details in Gianfranco’s talk

- In pure DOME mode the setup of a disk-only site becomes simpler, and more robust, because the intercluster communication is more solid

Security - Macaroons

- DPM has pioneered macaroons together with dCache, a few years ago.
- They work fine, being used in the DOMA-TPC exercise, and they are quite easy to configure

Security - OIDC - WLCGAuthWG

- OpenID-Connect works fine in the DPM HTTP frontend
- It loads fine and passes on its auth information. Good.
- The question is more what to do with it, how to map it and decide if a request is authorized or not, inside DOME and DomeAdapter

- Dynafed uses the same lcgdm-dav frontend as DPM, and does it right, the behaviour is totally configurable
- Although the frontend is the same, DPM has different rules and conventions for authorising a client, which must be kept backwards compatible. More difficult.

- Things will be clearer when the WLCG authorisation WG publishes its conclusions (or even better, when there's a working prototype we can use)
- I don't see big problems with this, maybe some small improvement

External packages - dpm-contrib-admintools

- The dpm-contrib-admintools package is not part of a DPM release
  - Useful scripts and queries to perform various tasks
  - Released asynchronously from time to time
  - Not versioned with the rest of DPM

- Some of these tools can only work if the legacy daemons are running
- We would not change the nature of this package, i.e. “3rd party” contributed tools
- We should agree on what to do

“The future of DPM”

- Unsurprisingly we start being posed this question more often
- The small DPM team secured the project from the technical point of view, and made it able to compete technically with the upcoming known technical challenges (e.g. higher load, scaling up, TPC, WebDAV, macaroons, xrootd, multi-site, caches, etc.). Sites can work well.
- We know that there will be the necessity of adapting “deltas” in the future, e.g. enabling OpenID-Connect in Apache. Easy things and fixes are not a big problem

- **We (DPM team) don’t know what to expect for the technical challenges that we don’t know about yet.**
  - A random invented example? Interfacing the replica scheduling of a multi-site DPM with some georeferenced information from Google Earth
  - Our very low manpower may decrease at the end of 2019. Who will support the setup in 2020?

- **At the same time, the funding and the support of any scientific open-source project depends 90% on its users**
  - **Many users, well organised —> long and prosperous project life, good support**
- Bottom line: if you are concerned by this, you should talk to your WLCG contacts and together raise the questions to the appropriate place.