

Thoughts about Operations Efforts in CMS

J. Letts (UCSD), D. Piparo (CERN) - pre-GDB on Operations Efforts - February 24, 2022



Effort for Operations in CMS

Effort for maintenance and operations in computing comes from several sources:

- Large contributions from CMS institutes and national laboratories, including interns, postdocs, and graduate students as well as permanent staff.
- Limited M&O budget (see e.g. [RRB 51](#)) available for devoperators

The operations program has been an engine of diversity and inclusion:

- Devops talent is recruited from a diverse applicant pool for highly visible positions within CMS
- Operators often move on to positions in CMS and other HEP experiments at laboratories & institutes
 - **Fill the pipeline of talent with highly-qualified engineers and scientists** from a diverse range of countries and institutions
- In this context mentoring is a key aspect of operations.

Service work in computing and software operations is a powerful way to attract and retain talent in HEP, with a very good cost/benefit ratio.

Most Demanding Aspects of Computing Operations

- Efficient Tier-0 operations during data taking
- Data and workflow management
- Interactions with sites

What is being done to reduce effort:

- **Automate procedures:** e.g. k8s deployments of the CMS computing services (e.g. data discovery), develop tools to automate consolidated practices (e.g. transforming high level dataset requests by physics groups into work for the workload management system)
- Migrate to **common computing tools:** e.g. Rucio, CRIC, HTCondor, GlideinWMS

What makes the operations of CMS easier:

- **Trusted and understood monitoring**
 - Example quantities: **network usage**, transfers (FTS and XRootD), CPU efficiency
 - Infrastructure: stable, reliable, fast
- Common strategic thinking about cyber security
 - e.g. security policies and best practices

Operations and Developments

- **Not only operations**
- Our estimate: only about $\frac{1}{3}$ of the Grid-related effort is operations and the **other $\frac{2}{3}$ goes into keeping up with/integrating new developments**
 - “Maintenance”
- Possible ways to reduce this:
 - **Far looking roadmaps** and significant overlap of old and new solution
 - **Avoid continuous replace/upgrade**
 - **Longevity** with stable support of basic components
 - **Coordination between OSG and EGI** to avoid rushing/extra effort required by intermediate solutions.

Examples:

- SE (x509 to Macaroon to WLCG tokens)
- CE (keying on token subjects before capabilities are ready/supported)
- ARC-CE (GDPR, new REST, token)
- End of GCT support by OSG vs EGI/WLCG
- End of python2 support January 2020 but HC still python2



Opportunities and Constraints

- CMS strives to efficiently use all of the resources available to us, also by **minimizing manual operations (effort)**
- It doesn't mean there is spare effort. **Effort savings typically redirected elsewhere, e.g. to R&D, not other tasks in operations**
 - **Core activities and basic evolution (e.g. tokens) are covered: new activities must be accompanied by new effort, which is hard to identify.**
 - For each (new) activity or engagement, **CMS needs to make a cost/benefit analysis**, a process also brought into focus in the preparation for the November review.
 - Example: Recent successful WebDAV integration came with additional temporary effort that did not reduce ongoing operations effort.
- Corollary: **common activities must have clear deliverables and added value to justify investments.**

Potential Risks and Synergistic Activities

- **Acquiring and retaining expertise is essential to operations.** Risks:
 - Expertise drain due to colleagues moving to new challenges in HEP and outside
 - **Working in software and computing not considered as valuable as on a detector or physics analysis: not enough early-career scientists engaging in S&C activities**
- Potential areas for synergistic activities:
 - Monitoring (e.g. network, transfers)
 - Cyber security (e.g. creation of threat models, creation of lists of concrete items to address)
- **Injections of effort directly into the experiment for work on well-scoped, time-limited activities is something we have seen be successful.**
 - Past examples: k8s deployment, WebDAV integration, HTCondor/GlideinWMS.
 - Ideas for the future: sw performance improvement, protocols evolution, service deployment mechanisms, development of threat models...