Transparent expansion of a WLCG compute site using HPC resources

R F von Cube[1], M Fischer[1], M Giffels[1], A Jung[2], T Kress[2], A Nowack[2], G Quast[1], A Schmidt[2] and M Schnepf[1]

[1] Karlsruhe Institute of Technology, Wolfgang-Gaede-Str. 1, 76131 Karlsruhe, Germany
[2] RWTH Aachen University, Sommerfeldstr. 16, 52074 Aachen, Germany

E-mail: `ralf.florian.von.cube@cern.ch`

Abstract. High energy physics (HEP) experiments generate vast amounts of data, which need to be processed and analyzed. For this, experiments located at the Large Hadron Collider (LHC) rely on the Worldwide LHC Computing Grid (WLCG). This distributed computing infrastructure is composed of several computing sites around the world. Operated in close contact with the local HEP groups, these sites provide very specialized environments for computing workloads of the collaborations. The expected increase in volume of data collected by the experiments and also a shift towards the use of non-HEP computing sites, such as university and commercial computing sites and high performance computing (HPC) centers, demand for solutions to easily access and use those sites.

In this contribution we present "COBalD– the Opportunistic Balancing Daemon" and the "Transparent Adaptive Resource Dynamic Integration System" (TARDIS), a scalable solution to dynamically and transparently integrate heterogeneous resources in existing infrastructure, as e.g. the WLCG. Two setups are showcased: First a setup with several sites throughout Germany consolidated through dedicated points-of-entry for multiple experiments, and second the transparent expansion of an existing WLCG site with a local HPC site.

1. Introduction

Large-scale experiments in high energy physics (HEP) produce vast amounts of data. As of today, such data is processed on computing resources operated by scientists involved in the experiments or by computing centers, which are dedicated to the experiments. For experiments e.g. at the Large Hadron Collider (LHC), such computing resources are provided by the Worldwide LHC Computing Grid (WLCG) [1]. This infrastructure constitutes a distributed computing grid with hundreds of computing centers around the world. As those centers are operated for the experiments and their users the centers can provide very specific environments for the computational payloads.

With an expected increase in the amount of data in the future and the corresponding growing demand for computing resources across many domains responsible entities, such as funding agencies or universities, are looking for ways to equip the different communities adequately, extending the current computing infrastructure. This often results in setting up computing centers, which are not dedicated to a specific experiment or research community but are shared by multiple domains. Owing to their ephemeral usage pattern, supplementing the existing infrastructure, such centers are referred to as opportunistic resources in HEP. Such resources can be offered e.g. as university research computing centers, high-performance computing (HPC) centers, or commercial computing centers. Catering to the needs of multiple communities, such centers can not be designed for the specific needs of a single experiment, but need to be more generic.

For experiments and their users having multiple such centers at hand is beneficial in terms of the availability of computing resources however this poses a challenge when trying to access and use them as each center possibly has its dedicated authentication and resource management system. To overcome this challenge the centers should be integrated into a common, ideally already existing, infrastructure that includes a common authentication and resource management system. Such a setup would allow transparent access to and use of the centers.

Another challenge is the heterogeneity of the resources offered by the centers. Using state-of-the-art containerization and virtualization technologies well-defined environments can be provided for the computational payloads of experiments and users.

2. Dynamic resource integration using COBalD and TARDIS

To dynamically schedule resources on opportunistically integrated sites, KIT has developed "COBalD – the opportunistic Balancing Daemon" [2] and the "Transparent Adaptive Resource Dynamic Integration System" (TARDIS) [3]. Resources are made available by integrating them into a common batch system, called the overlay batch system (OBS). The OBS constitutes a single point-of-entry for users and experiments to all integrated resources. In order to provide a well-defined environment for user payloads, the jobs scheduled onto the integrated resources are started in a container by the OBS.

Software is either shipped by the user within their job, lazily loaded from within the job, or from available file systems. In the context of HEP the CERN virtual machine file system (CVMFS) is widely used. This HTTP file system can be mounted in user space and bind mount into the container. Because the file system uses HTTP, industry standard tools like squid proxies can be used for caching, furthermore, CVMFS allows for local shared caches on parallel file systems.

The flow of jobs submitted by experiments or users is shown in Figure 1. Yellow arrows show the interactions of the OBS with users and integrated resources. Green arrows indicate the feedback loop used by COBalD/TARDIS for resource scheduling, as discussed in the following.

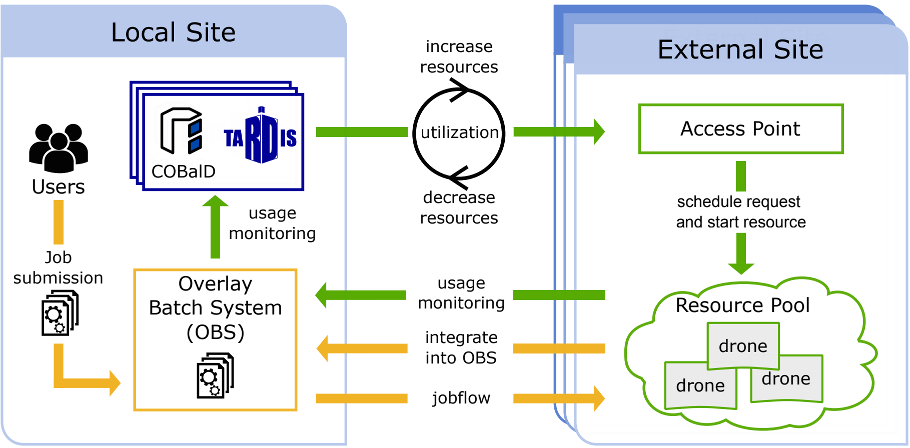

Figure 1. Flow of jobs through the system (yellow arrows) and the feedback loop (green arrows) used by COBalD/TARDIS for resources scheduling. Figure taken from reference [4].

2.1. Balancing opportunistic resources with COBalD

COBalD [2] is a resource management system for opportunistic resources. It is designed as a feedback control loop, reacting to the scheduling decisions of the OBS. As a meta-scheduler, COBalD is able to schedule resources without prior knowledge of the exact resource requirements of the jobs, and without knowledge of the resources available hardware, as the final scheduling decision is made by the OBS. By grouping indistinguishable resources into resource pools, COBalD can assess the overall utilization of those resources in the pool. COBalD then dynamically adjusts the number of resources per pool, based on the current utilization of the resources. As such, COBalD provides an abstraction layer for resource pools, which do not necessarily have to consist of compute resources, but can also be used for any other resource type, e.g. storage, network, or anything else. [5]

2.2. Integrating heterogeneous resources using TARDIS

TARDIS [3] uses COBalD and manages the lifecycle of resources and assesses their individual usage. [6] It is responsible for the integration of resources into the OBS. For this, TARDIS interacts with the local resource management system (LRMS) of the integrated resources. Using a local proxy user, TARDIS can authenticate with the LRMS and submit a job to the batch system, start a container, or spin up a virtual machine. The resulting process, which is referred to as Drone, is expected to start up the worker node component of the OBS. The Drone connects to the OBS, which is then able to schedule jobs onto the integrated resource. The OBS starts up the payload in a specified container which provides the expected environment for the job. More information on TARDIS and its modular design can be found in reference [6].

3. Integration of HPC Resources into the WLCG

COBalD/TARDIS are used at several sites in the WLCG for the opportunistic integration of heterogeneous resources like HPC and university computing centers throughout Germany. Mandatory for the integration is only a local proxy account, enabled user namespaces and an outgoing network connection on the resources, all other requirements can be operated in user space. However, close communication with the operation personnel usually allows for simplifications like e.g. on-site installation of CVMFS, or squid proxies.

3.1. GridKa: Opportunistic Resources at a Tier 1 WLCG Site

COBalD/TARDIS is used with multiple instances at the German Tier 1 WLCG site GridKa. [7] With two computing centers each from the universities of Bonn, Karlsruhe and Munich, integrated into one common infrastructure, a large number of additional, opportunistic resources can be provided to experiments and users. The jobs are submitted to HTCondor compute elements (CEs), which dispatch the jobs to an instance of HTCondor as the OBS. Based on the current demand, up to more than 7000 additional cores were provided to experiments and users in 2022. With this setup, as shown in Figure 2, more than 26 million core-hours were provided to experiments and users in 2022, as shown in Figure 3. CMS, ALICE, Belle II, ATLAS, IceCube, and LHCb benefitted from the integrated resources. As some of the resources are available only for a selected group of experiments or users, the OBS is configured appropriately. This, again, simplifies the use of multiple of such resources for experiments and users as they do not have to be aware of the underlying infrastructure constraints.

If Apptainer (formerly known as Singularity) [8] is used, activated user namespaces are an additional requirement. If CVMFS is available and configured accordingly, the instance on the host can be used directly, else CVMFS is made available for the Drones using `cvmfsexec` [9], which can mount CVMFS repositories as an unprivileged user. The Drones consisting of a lightweight container image are provided through DUCC (Daemon that Unpacks Container Images into CernVM-FS) [10] and made available on CVMFS. Within the container, the

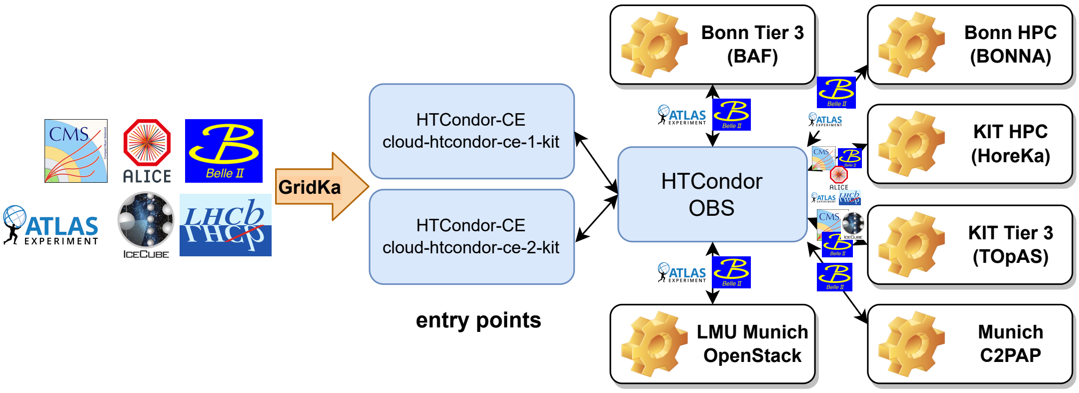

Figure 2. The setup of the opportunistically integrated resources at GridKa. Figure adapted from reference [7].

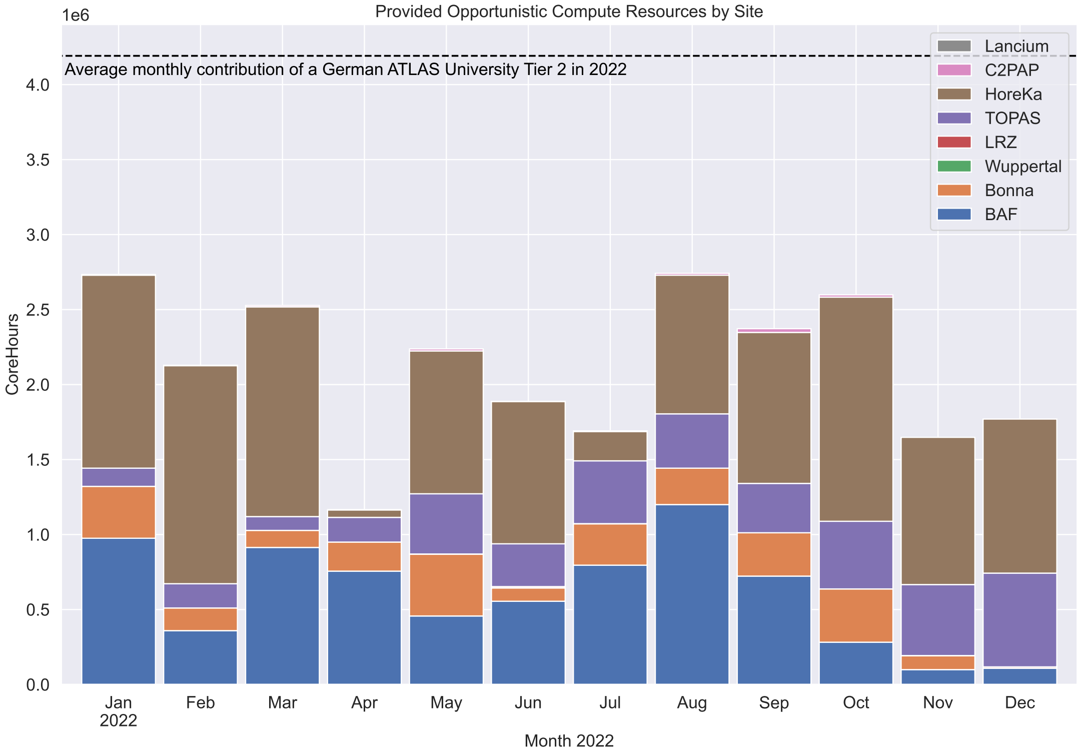

Figure 3. The number of core-hours provided by the different sites to the experiments through the setup at GridKa per month in year 2022.

appropriate HTCondor configuration is pulled from remote git-repositories and the worker node component of HTCondor, `condor_startd`, is started. The latter is configured to execute the payloads in yet another container, providing the expected WLCG site environment and is served through CVMFS.

3.2. CLAIX: Integration of an HPC Center into a Tier 2 WLCG Site

Using COBalD/TARDIS, the university high-performance computing center CLAIX could be integrated into the local Tier 2 WLCG site at RWTH Aachen University. The LRMS

HTCondor instance used to manage the resources of the Tier 2 WLCG site acts as the OBS into which the resources of CLAIX are integrated. Just like with the GridKa setup, Drones are started as lightweight containers, whose images are served through CVMFS. By starting the payloads in a container with the appropriate environment, and because of the high network bandwidth between CLAIX and the Tier 2 WLCG site, any job submitted to the site can be executed on the resources of CLAIX. Therefore, no additional CE is deployed, but the HPC resources are available completely transparent through the existing infrastructure.

With this setup, approximately 11.5 million core-hours were provided to experiments and users in 2022. As a scalability test, up to 10 000 cores were integrated into the Tier 2 WLCG site and used by the CMS experiment and users for a limited amount of time as shown in Figure 4.

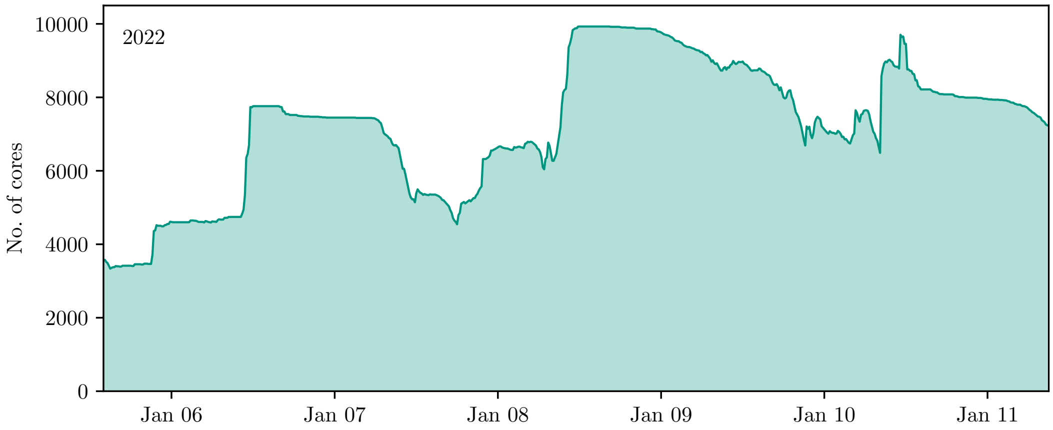

Figure 4. The number of CPU cores opportunistically integrated from CLAIX into the Tier 2 WLCG site at RWTH Aachen University during the scalability test in January 2022.

4. Conclusions

In order to make non-HEP sites transparently available, KIT has developed COBaLD and TARDIS. COBaLD is a meta-scheduler using a feedback control loop in order to estimate and adjust the number of resources to the current demand. TARDIS is a resource management system interfacing different types of sites and managing the lifecycle of temporarily integrated resources. In combination COBaLD/TARDIS can be used to dynamically and transparently integrate heterogeneous resources into existing infrastructure.

In this contribution, two example setups using COBaLD/TARDIS for the opportunistic integration of heterogeneous resources into existing WLCG infrastructure have been presented. With both setups, a significant amount of additional resources can be provided to experiments and users. They both showcase the transparent usability of such resources for experiments and users and prove the scalability of setups using COBaLD/TARDIS.

References

[1] K. Bos, N. Brook, D. Duellmann, C. Eck, I. Fisk, D. Foster, B. Gibbard, C. Grandi, F. Grey, J. Harvey, A. Heiss, F. Hemmer, S. Jarp, R. Jones, D. Kelsey, J. Knobloch, M. Lamanna, H. Marten, P. Mato Vila, F. Ould-Saada, B. Panzer-Steindel, L. Perini, L. Robertson, Y. Schutz, U. Schwickerath, J. Shiers, and T. Wenaus. *LHC computing Grid: Technical Design Report. Version 1.06 (20 Jun 2005).* Technical design report. LCG. CERN, Geneva, 2005.

[2] Max Fischer, Eileen Kuehn, Manuel Giffels, Matthias Schnepf, Stefan Kroboth, Thorsten M., and Oliver Freyermuth. MatterMiners/cobald, August 2022.

[3] Manuel Giffels, Stefan Kroboth, Matthias Schnepf, Eileen Kuehn, PSchuhmacher, Rene Caspart, Max Fischer, Florian von Cube, and Peter Wienemann. MatterMiners/tardis, August 2021.

[4] Matthias Jochen Schnepf. *Dynamic Provision of Heterogeneous Computing Resources for Computation- and Data-intensive Particle Physics Analyses.* PhD thesis, Karlsruher Institut für Technologie (KIT), 2022. 53.52.02; LK 02.

[5] Max Fischer, Eileen Kuehn, Manuel Giffels, Matthias Jochen Schnepf, Andreas Petzold, and Andreas Heiss. Lightweight dynamic integration of opportunistic resources. *EPJ Web Conf.*, 245:07040, 2020.

[6] Max Fischer, Manuel Giffels, Andreas Heiss, Eileen Kuehn, Matthias Schnepf, Ralf Florian von Cube, Andreas Petzold, and Günter Quast. Effective Dynamic Integration and Utilization of Heterogenous Compute Resources. *EPJ Web Conf.*, 245:07038, 2020.

[7] Michael Böhler, René Caspart, Max Fischer, Oliver Freyermuth, Manuel Giffels, Stefan Kroboth, Eileen Kuehn, Matthias Schnepf, Florian von Cube, and Peter Wienemann. Transparent Integration of Opportunistic Resources into the WLCG Compute Infrastructure. *EPJ Web Conf.*, 251:02039, 2021.

[8] Gregory M. Kurtzer, Vanessa Sochat, and Michael W. Bauer. Singularity: Scientific containers for mobility of compute. *PLOS ONE*, 12(5):1–20, 05 2017.

[9] Jakob Blomer, Dave Dykstra, Gerardo Ganis, Simone Mosciatti, and Jan Priessnitz. A fully unprivileged CernVM-FS. *EPJ Web Conf.*, 245:07012, 2020.

[10] Enrico Bocchi, Jakob Blomer, Simone Mosciatti, and Andrea Valenzuela. CernVM-FS powered container hub. *EPJ Web Conf.*, 251:02033, 2021.