

Table of Contents

SIMPLE Grid Project.....	1
Introduction.....	1
Principles.....	2
The Community.....	2

SIMPLE Grid Project

The WLCG unites resources from over 169 sites spread across the world and the number is expected to grow in the coming years. However, setting up and configuring new sites to support WLCG workloads is still no straightforward task and often requires significant assistance from WLCG experts. A survey presented in CHEP 2016 revealed a strong wish among site admins for reduction of overheads through the use of prefab Docker containers or OpenStack VM images, along with the adoption of popular tools like Puppet for configuration. In 2017, the Lightweight Sites project a.k.a. the SIMPLE Grid Project was initiated to construct shared community repositories providing such building blocks. The SIMPLE Grid project is an acronym for Solution for Installation, Management and Provisioning of Lightweight Elements on the Grid (Worldwide LHC Computing Grid). It comprises a modular and extensible core system that abstracts low-level details through a YAML based site-wide configuration file which is used to configure all distributed components through a single command. To accommodate the diverse scenarios at different sites, the project will enable site admins to cherry pick their background technologies and methodologies for orchestration (Puppet, Ansible, ...), clustering (Docker Swarm, Kubernetes, ...) and networking (dedicated networks, custom overlay networks or a combination of both).

Introduction

The WLCG is a very diverse ecosystem. There are various flavors of Compute Elements (CreamCE, ARC, CondorCE), Batch systems (Torque/PBS, Slurm, Condor), middleware packages for various grid services etc. that are available. Moreover, site admins have their own preferences for tools they wish to use for provisioning/orchestrating, configuring and maintaining their infrastructure and services (for instance, Puppet, Ansible, YAIM, Docker, Kubernetes, OpenStack etc.). A WLCG site can be set up using any valid permutation of a logical combination of grid services and infrastructure tools mentioned before. For instance, let's say Site Admin A configures the CreamCE nodes using INFN's CreamCE Puppet module and the Torque/PBS based batch and worker nodes using YAIM in order to support workloads submitted to his site from the Alice VO. Another site admin, let's say Site Admin B could set up a similarly equipped grid site (Cream, Torque) using just YAIM for configuration management. Yet another site admin, let's say Site Admin C, might be required to support workloads from the CMS VO and hence he set up a HTCondorCE, Batch and Worker nodes using some preferred infrastructure orchestration and configuration technology. Similarly, Site Admin D,E,F.....X,Y,Z..and so on, might end up using completely different recipes to set up their grid sites.

Each of these recipes might be accompanied with its own unique set of orchestration/configuration challenges. Most sites, especially the newer or smaller Tier-2 and Tier-3's, might not have the required experience with grid services and/or enough manpower to handle and resolve such challenges without external help/intervention (from the grid experts at CERN, which takes more time compared to the on-site resolution of issues). During a survey presented in CHEP 2016 [\[7\]](#), several site admins expressed their support towards shared repositories for prefab Docker containers/ Puppet modules, OpenStack VM's etc. (technologies that site admins are more familiar with when compared to the low-level WLCG tools and services) that could be used for setting up classic grid sites in a simpler and more uniform manner.

It should also be noted that such simplification of site setup process could potentially enable more new sites (smaller T2/T3's) to join the WLCG. Moreover, these smaller sites have a potential to grow in the future and getting started as a lightweight site might just be their entry point to the WLCG. While smaller sites might not seem interesting in terms of pledged resources when looked at individually, when pooled together, the combined potential of smaller sites cannot be ignored. A brilliant example of this is BOINC, an easy to set up volunteer computing tool, through which ~400k cores were made available to support WLCG workloads during the 8th BOINC Pentathlon in 2017.

The SIMPLE Grid project aims to address the above-mentioned challenges and take the first step in the direction where

- We have sites that can run with minimal oversight and operational effort from the people at the site.
- They run almost "by themselves"
- Provide grid services with preferred technology and less effort for configuration/management through a uniform interface
- Keep the functionality at sites similar to classic grid sites, but make the whole process of setting up and managing sites easier for admins

Principles

To ensure we deliver SIMPLE Grid as a framework that can effectively handle the challenges mentioned in the previous section and be scalable at the same time, the project is modeled around the following design principles:

Abstraction

Abstract the low-level details of several popular CE/WN/Batch technologies from grid components from the site admins, as much as possible. This enables us to provide a consistent interface to set up a Grid Site despite the choice of grid components and background technologies preferred by site admins.

Modularity

Allow site admins to plug and play grid components they wish to use at their site.

Ease of Deployability

Use easier deployment strategies based on containers/VM's.

DRY - Don't Repeat Yourself

There are several configuration parameters that are reused by several grid components. Use DRY strategies to reduce configuration requirements and also make it easy for any changes to propagate changes across the framework

Single Node for Configuration

One node to rule them all! Basically, the site admin must be able to describe all the grid services and components at one place and then the framework should configure all the nodes and services based on the admin's preferences. Modern configuration management systems like Puppet, Ansible, SaltStack etc. support this strategy out of the box.

Extensibility

As stated earlier, there can be several permutations of combining the grid services and infrastructure technologies to set up a WLCG site. To make as many of these permutations 'lightweight' i.e. supported by SIMPLE Grid project, the framework should be extensible and community driven

The Community

The SIMPLE Grid is an open source and community driven effort!

For Developers

For Site Admins and Users

-- MayankSharma - 2018-07-02

This topic: LCG > SIMPLEGridProject

Topic revision: r5 - 2020-03-04 - MaartenLitmaath



Copyright &© 2008-2021 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

or Ideas, requests, problems regarding TWiki? use [Discourse](#) or [Send feedback](#)