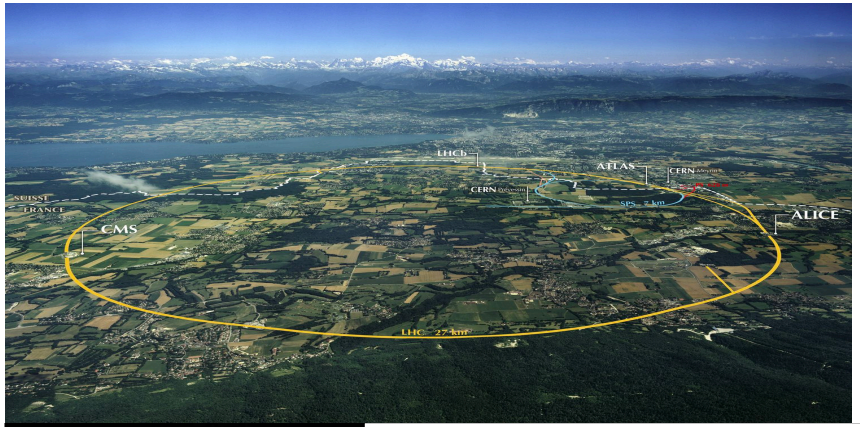
SIMPLE Grid Framework

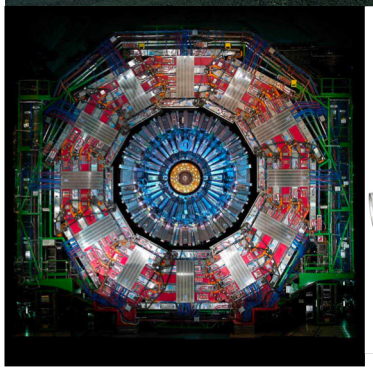
Mayank Sharma (CERN, speaker)
Maarten Litmaath (CERN)
Eraldo Silva Junior (CBPF, Brazil)

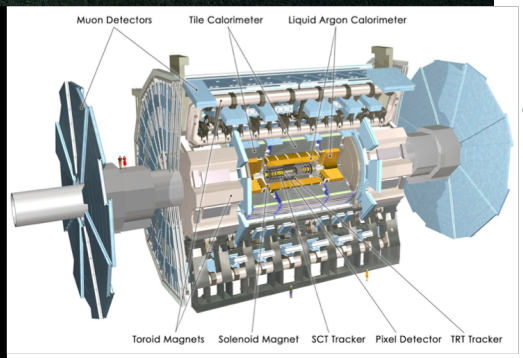
$>whoami

- Software Engineer, CERN.
- Developer of SIMPLE Grid Framework.
- Google Summer of Code, Google Code-In
- Release Manager, OpenMRS Platform 2.0
- Hackathon/Startups/ IoT

CERN: Quick overview



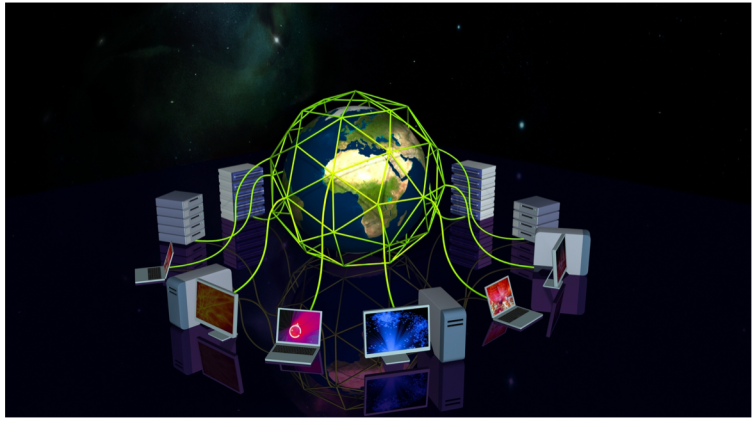




- Largest Particle Accelerator located on the Franco-Swiss border.
- LHC: A 27km long tunnel through which high energy particle beams are accelerated.
- Particle beams, travelling in opposite directions collide at 4 main experiments (Atlas, CMS, Alice and LHCb).
- Popular contributions: Higgs Boson (2012), World Wide Web, Hardon Therapy (Medical Applications) etc.

The LHC challenge

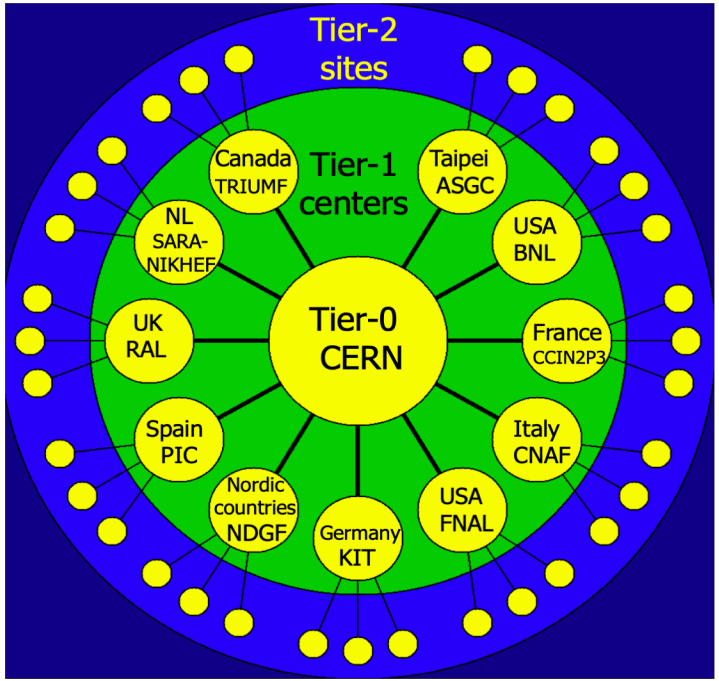
- 50+ PetaByte/year (Raw data), 80+ PetaByte/year (Simulated/Derived data).
- Data Analysis requires ~500k typical CPU processor cores.
- Scientists spread around the world.
- CERN can provide 20-30% of CPU and storage.
- 70-80% are provided by Worldwide LHC Computing Grid (WLCG) providers.



The WLCG Answer

- 170+ Computing Centers, 35+ countries.
- 15 Large centres for long term data management
  - CERN = Tier-0
  - 14 Tier-1 Center
    - New: Korea, Russia
    - Fast Network Links
- 70+ federations of 140+ smaller Tier-2 centers.
- Tens of Tier-3 sites.
  - University resources dedicated to smaller physics groups

Read More here!



Diversity in WLCG

Types of **WLCG services and middleware packages.**

Technologies preferred by site admins for managing their infrastructure

Site Admin's Perspective

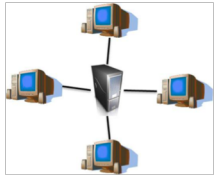
- Lightweight Sites Survey: http://cern.ch/go/rhV9
- 51 Sites responded to the questionnaire that shows potential benefits of **shared repositories**
- **Conclusion:**
  - Most sites still require **classic grid services** which can be complicated to configure/deploy
  - **Simpler mechanisms** for orchestration of sites utilizing **modern infrastructure tools** will be beneficial
  - Strong support for **Docker, Puppet, OpenStack images**

SIMPLE

- **S**olution for **I**nstallation, **M**anagement and **P**rovisioning of **L**ightweight **E**lements
- Support diversity in WLCG sites with **minimal oversight and operation efforts**
- Keep **functionality the same**, but easier for site admins to setup and maintain

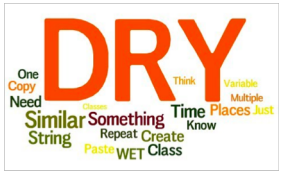
Principles

Abstraction



One node to configure the site

DRY (Don't Repeat Yourself)



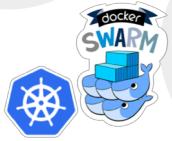
SIMPLE

Extensibility

Community Effort

Modularity

Simple Deployment



kubernetes

What SIMPLE Grid does

- Set up a **grid site with O(100) lines** of YAML
- **Modular** and easy to extend to **support other grid services**
- **Community Driven**: Open source and open discussion channels.

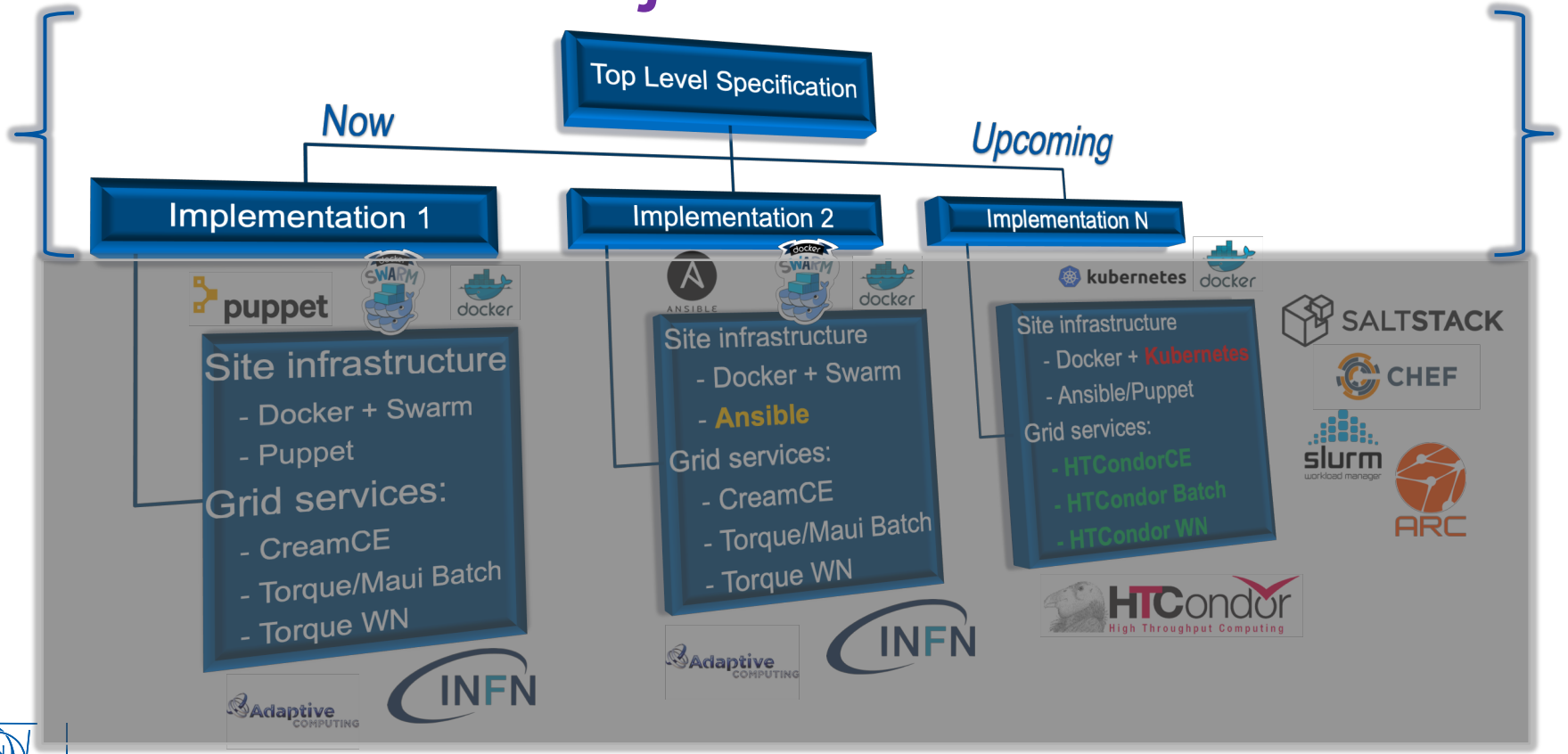
Wait, but what am I doing here?

- We took our **abstraction**, **modularity** and **extensibility** principles too seriously!
- With a **few lines of YAML**, you can create a **complex computing cluster** that runs your **desired software packages** and services.
- Application **Beyond CERN**: **Economics/ Finance**, AI/Machine Learning, Medicine/Microbiology IoT

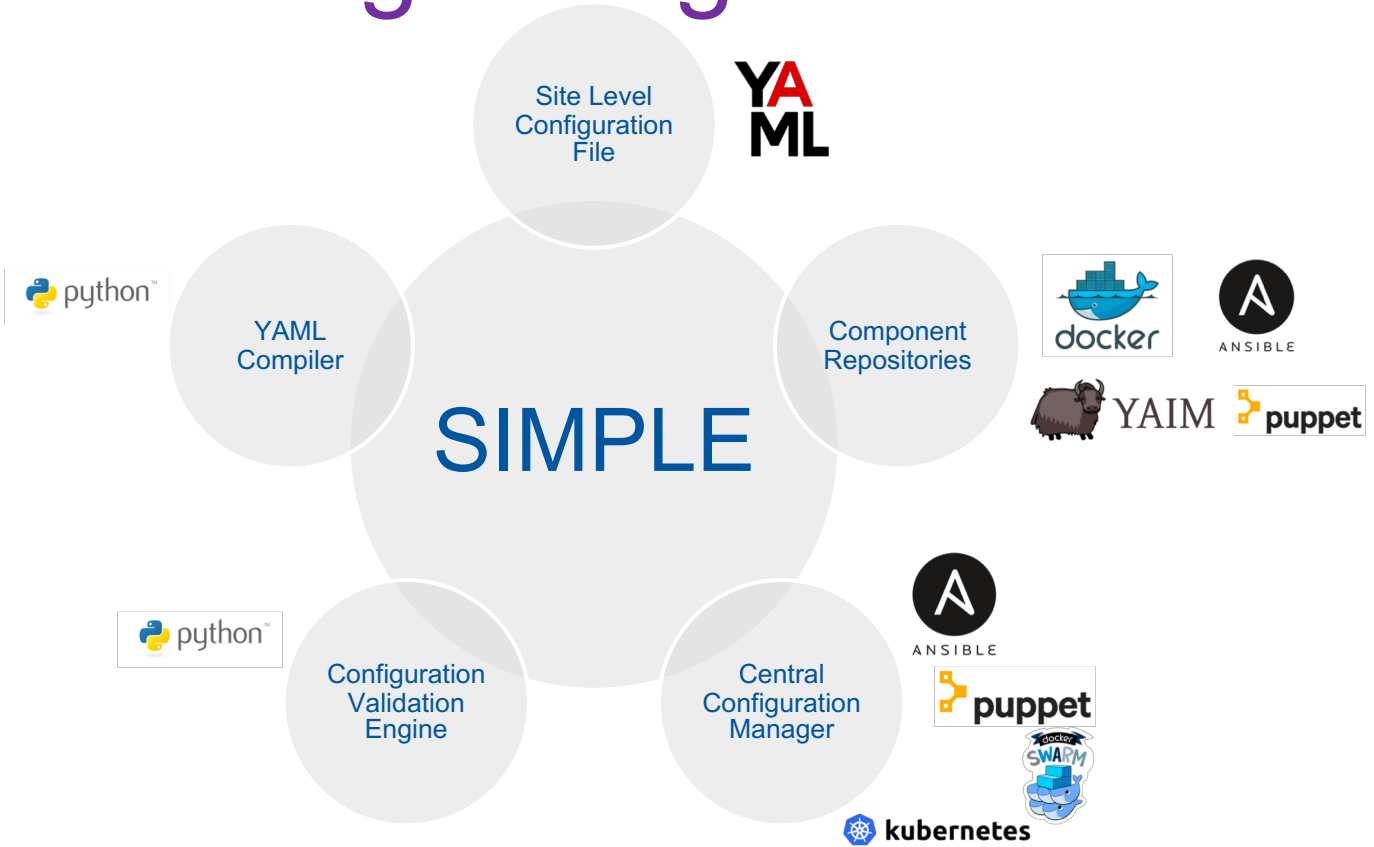
Wait, but what am I doing here?

- **2 of 3 SIMPLE Core Components** are **python packages**.
- **Open Source** and **Community Driven**.
- Develop a **Robust core with SIMPLE Grid**, Parallelly enable the community to lead other applications.

SIMPLE – Project Structure

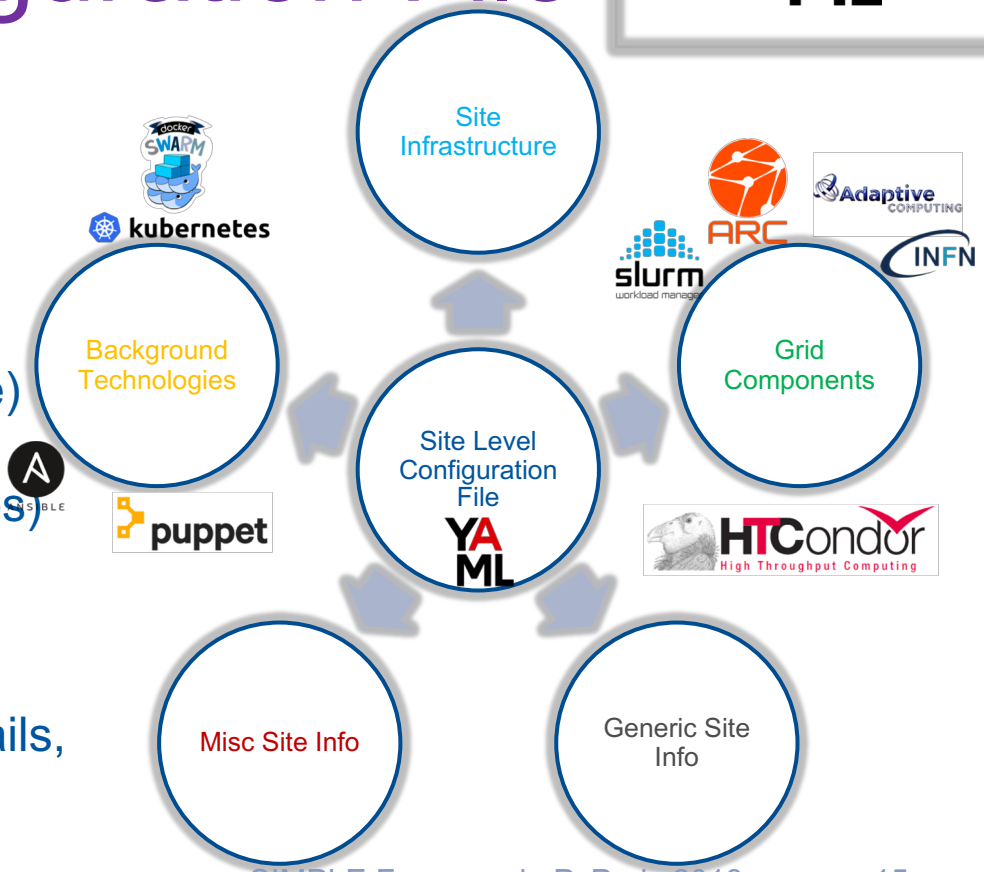


SIMPLE – Lightweight Elements



Site Level Configuration File

A single YAML file to describe:
Site-Infrastructure (Hostnames, IP addresses, OS/Kernel, Disk/Memory)
Service Components (What components to install and configure)
Background Technologies (Puppet/Ansible, Docker/Kubernetes)
Specific to Grid Use-Case:
- **Generic Site Info** (Users, Groups, Supported VOs)
- **Misc. Site Info** (Security emails, location etc.)



Component Repositories

- Publicly hosted repositories on GitHub that provide
  - **Dockerized** services that are executed on the Cluster. For instance, CE/WN/Batch/Squid etc.
  - **Meta information** for configuration of containers using different configuration management tools
- 1 repository for every cluster service (for the Grid use case, CreamCE, CondorCE, Torque, Slurm reside in separate repositories)
- Grid Examples: CreamCE, TorqueWN

YAML Compiler

- **Minimize configuration** requirements via
  - **Variables**
  - Sensible **default values** for site-level configurations
  - Ability to **override values**
  - **support additional parameters** not defined in the system
  - Builds on top of **PyYAML** and **Ruamel**
  - Split configuration into **multiple logically related YAML files** that can be shared

Configuration Validation

- Built on top of ***Yamale***.
- Configuration validation engine to ensure information supplied in site configuration file:
  - **meets the configuration requirements** of desired site component
  - is **realizable on the available infrastructure** using available background technologies
- http://cern.ch/go/CvS8
- Possibility to inject custom validation rules

Compiler + Config Validation

- New keywords:
  - **__from__** : (Resolve complex anchor/variable hierarchies)
  - **__include__** : (Similar to *import* in python)
- Support for **Runtime Variables**
- **Custom data types**, **schema files** and **default values**.

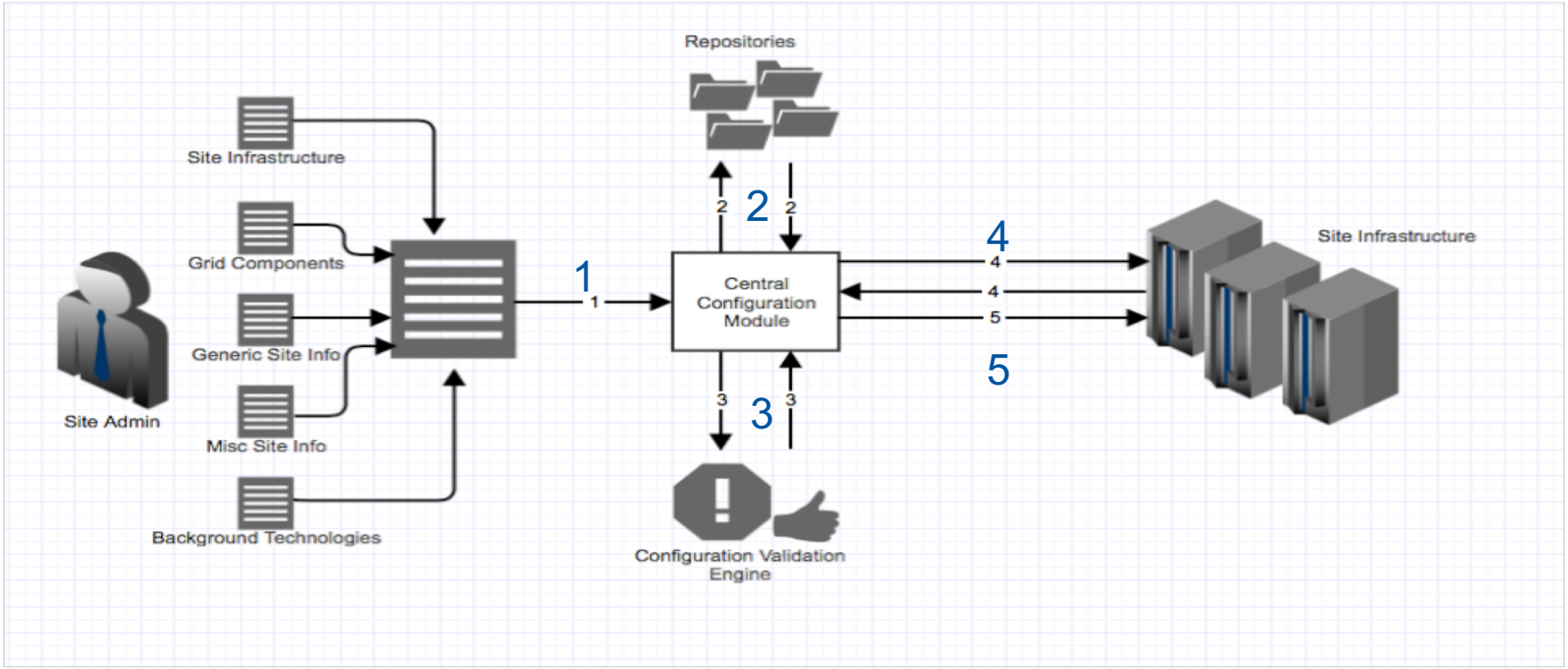
Central Configuration Manager

- The **main module** for centrally configuring everything at the site
- **Uses Validation Engine** to check site-configuration file
- Checks **status of available Site Infrastructure** that needs to be orchestrated
- Installs and **configures component repositories** from the GitHub repositories

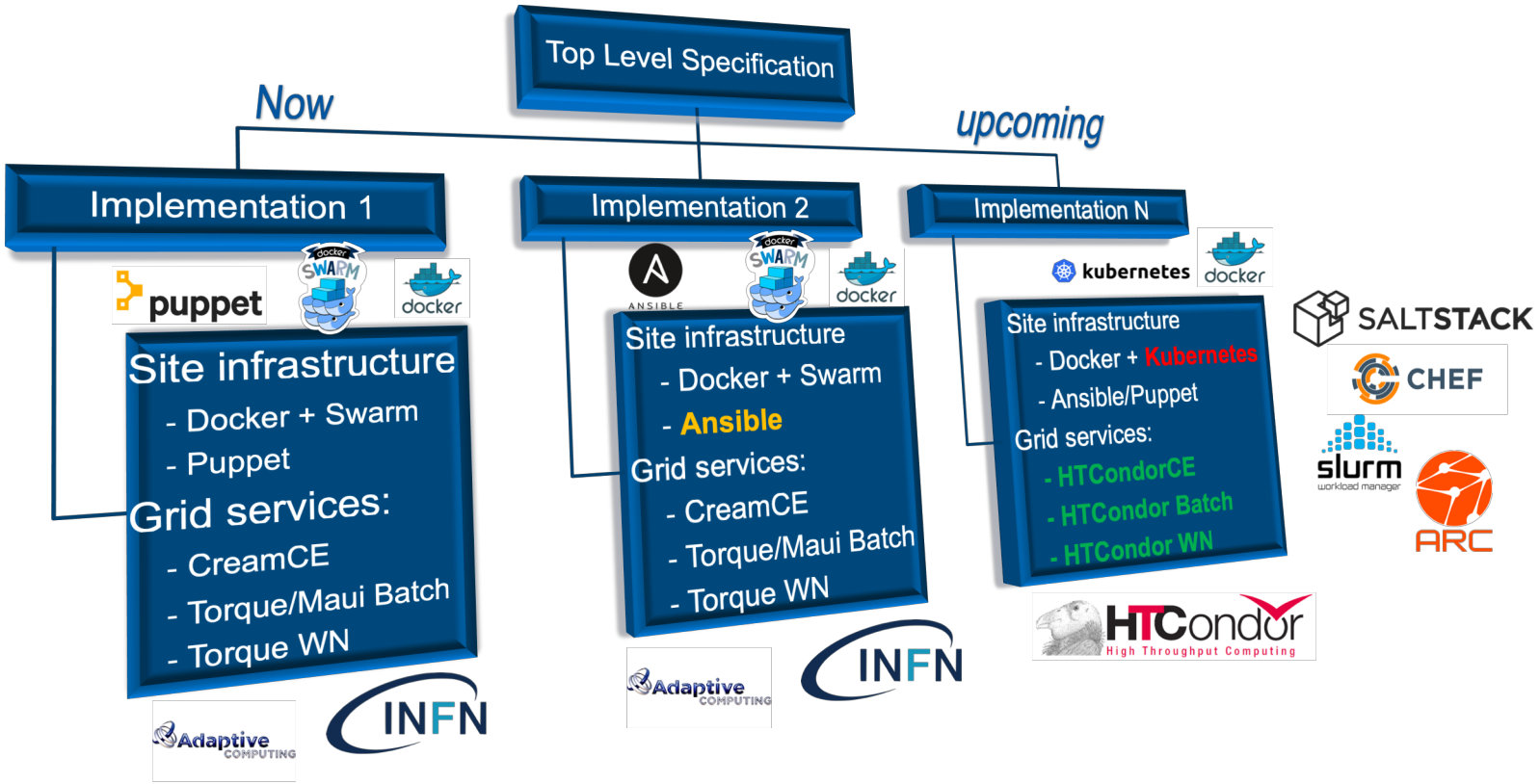
Central Configuration Manager

- Implements a **Networking strategy** (overlay/dedicated)
- Executes **lifecycle callbacks** on the Hosts and Containers of component repositories.
- Runs **tests** to check for success or failure of site configuration

Specification: Putting it Together



WLCG Example



Implementations

- **Site Level Configuration File YAML Compiler**
  - Python command line utility
- **Configuration Validation Engine**
  - Python command line utility
- **Central Configuration Management System**
  - Puppet
  - Ansible
  - …  


Google Summer of Code 2018 Project
Alpha candidate developed by **Tarang Mahapatra**, University of British Columbia, Vancouver

Implementations

- **Repositories for Components**
  - Cream Compute Element + Torque Batch System
  - Torque Worker Node
  - …    
- **Repositories for Other Applications**  
  - **Economics: Julia Gavrilenko (REU), Sergei Belov (JINR)**
  - … 
    - **But, How to support my use case?**

      Create a new GitHub repository with your containerized services.

      The framework takes care of the rest!

The Open Source Community

Project Homepage

http://cern.ch/go/9lHd

GitHub Repositories

http://cern.ch/go/kr7p

Simple Grid Specification

http://cern.ch/go/8JLH

Technical Discussion List (E-Groups)

Name: WLCG-Lightweight-Sites-Dev
Link: http://cern.ch/go/l9wZ

Google Forum

Name: WLCG Lightweight Sites
Link: http://cern.ch/go/Hz7S

Mattermost (IM):

Team: WLCG
Name: WLCG-Lightweight-Sites
Link: http://cern.ch/go/8HWP

Conclusions

- Setup a **robust and complex computing infrastructure** with a **few hundred lines of YAML** description.
- **Only standard SysAdmin** know-how required.
- **Focus on your code** and not your infrastructure.
- **Open Source** and **Community** Driven!

Questions?

Sounds Interesting?

Let's talk:

Mayank Sharma

mayanksharma94
maany_shr
mayank.sharma@cern.ch
maany
devmaany.co

Eraldo Silva Junior

eraldojunior
ejunior@cbpf.br
ejr004

Important Links:

Website: https://wlcg-lightweight-sites.github.io
GitHub Org:WLCG-Lightweight-Sites
Mailing List: Google Groups
Wiki: CERN Twiki
Technical Roadmap (WLCG): CERN TWiki
Issue Tracking: v1