

# \* as a Service

## Introduction

The traditional way an experiment setups their software infrastructure is to follow the ‘grid’ model

- pre install the experiment software on a network accessible disk
- pre startup any persistent services (e.g. database servers)
- submit batch jobs to do analysis, simulation and data processing. These jobs run only as long as needed to complete their task and then go away.

However, industry has, by and large, changed to a cloud model

- infrastructure components are encapsulated in virtual machines
- a group of virtual machines containing all the needed infrastructure are brought up at a site at one time
- the components in the virtual machines are left running for an extended period of time and to process all requests.

HEP could use a similar model to provided ‘Experiment as a Service’

- Analysis as a Service: physicists interact with client programs which send queries to a processing farm filled with ‘Analysis Servers’ (similar to ROOT’s PROOF)
- Simulation as a Service: Simulation requests are send to a group of Servers which run the requests and access needed information (e.g. DB queries) from related Servers.
- Data Processing as a Service: The processing configuration and list of events are processed by a series of ‘Data Processing Servers’ that are somehow coordinated and which are getting needed information from other Servers on the site.

## Possible Discussion Points

- What value would HEP derive from using a cloud based model?
  - Which of the experiment tasks might be best suited for the cloud based model?
- What would be the drawbacks?
- Does it make sense to move parts of the current workflow to the cloud, or even move it in its entirety (Analysis, Simulation and Data Processing as a Service)?
- What would the architecture of such a system be like?
  - What kind of job submission/control is needed?
  - How would software versions be handled?
  - What would the processing ‘servers’ be like?
  - How would the different components communicate?
- How could machines at a site be shared across different services or experiments?
  - How could resource accounting be done?
- How could we transition to such a model?