Gratia v2 Development and Deployment Plan.

Introduction

The OSG Blueprint Accounting meeting was held at Fermilab on February 23rd, 2016. The main purpose of the meeting was to discuss the current status of the OSG Account Service (Gratia), evaluate the feasibility of continuation of Gratia software support and minor improvements and/or propose on how to modernize the OSG accounting and make it sustainable and expandable during the next 5 years.

This document is the outcome of this meeting.

Requirements

- Retire GRATIA database and collector by 12/31/2016!
- Keep all the existing probes without modification
- All historical data must be transferred to the new infrastructure.
    - the granularity of information kept online may be reduced to monthly aggregates for anything that is older than 1 year
    - Any plot for more than two weeks does not have to be with an hourly granularity
- The display.grid.iu.edu does not change.
- The look and feel of Gratia displays may change at both gratiaweb and myosg.
- Careful attention must be paid to guarantee that the safety of the accounting data is as assured in the future as in the past.
- A transition like we do now is easier in the future.
- New system should be designed with long term maintenance in mind:
    - no more java or alike
    - flexible to schema evolution
- “workflow data” on workflows that run on OSG should be available to the wider CS audience.
- Allow study of joint of perfSonar, HTCondor, XRootd, gridftp, network transfer statistics

Proposed Architecture

Accounting service will consists of these modules:

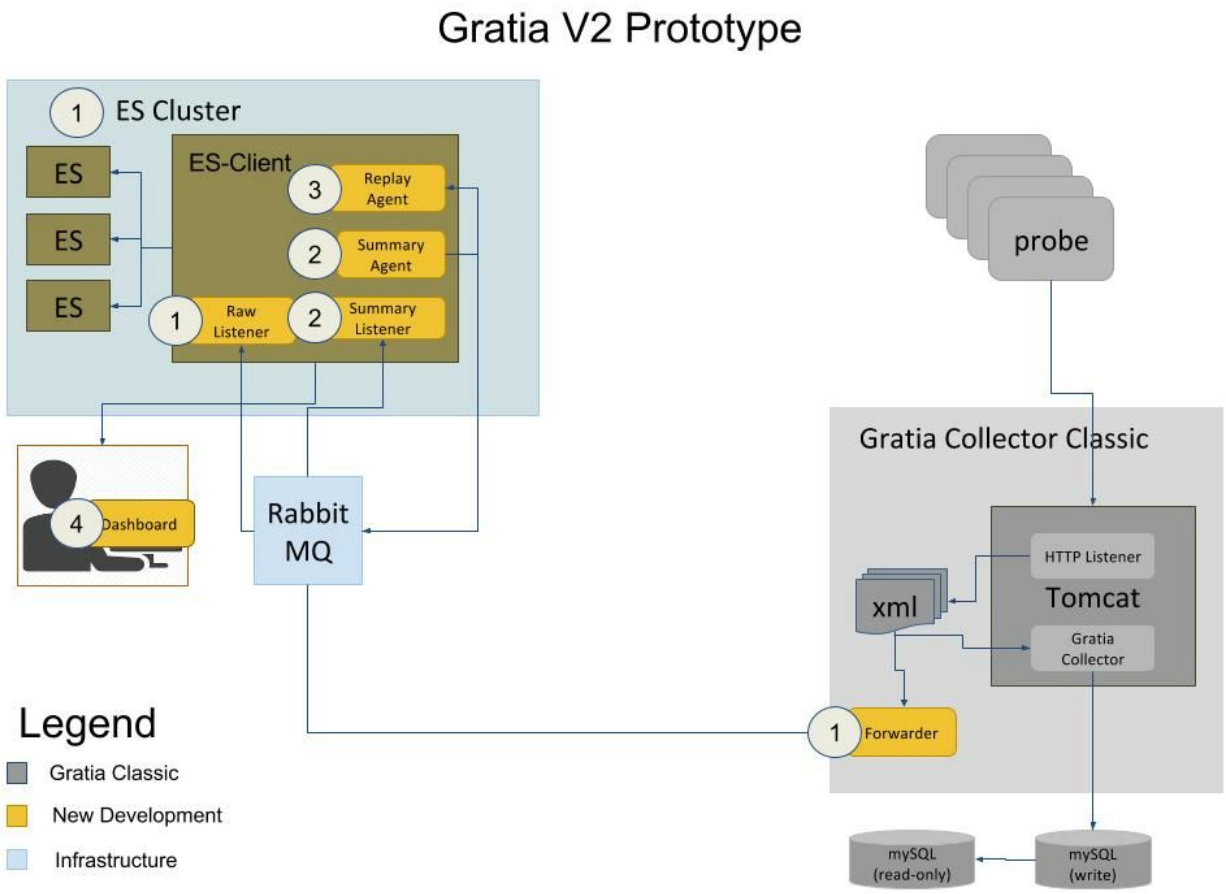
- Custom accounting probes
- Messaging Bus Service.

- Consumers (custom scripts, capable of digest, modify and replay information from/to Messaging Bus Service)
- NoSQL database for data storage
- OSG Display
- GratiaWeb
- WLCG and XSEDE reports
- Email reports

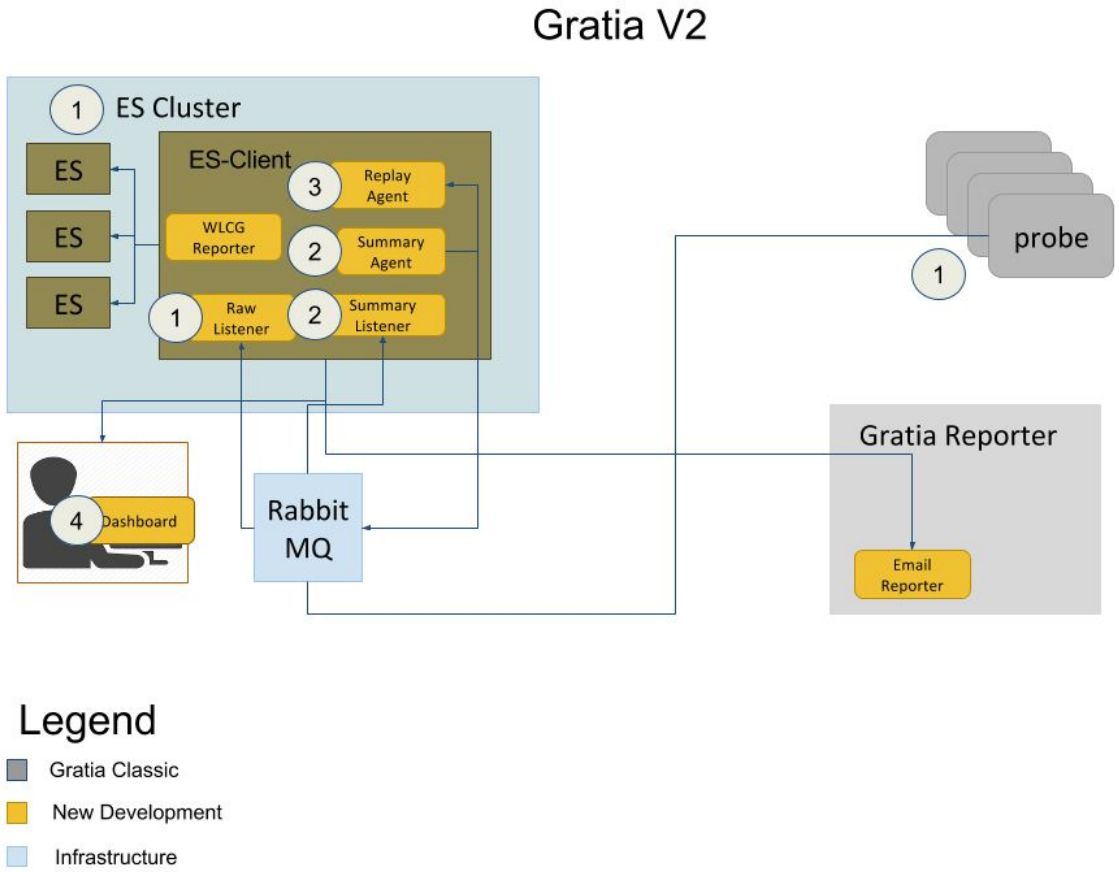
The implementation decisions and rationale:

- Custom accounting probes:
    - No changes to the current gratia probes (see requirements)
- Messaging Bus Service: Use RabbitMQ
    - RabbitMQ is an Open Source Messaging Broker. It is installed and heavily used at GOC. We can start using it right away which is crucial taking into account the deadlines.
- Consumers: custom scripts, capable to digest, modify and replay information from/to Messaging Bus Service:
    - Development needed (probably, python scripts capable of listening to RabbitMQ, aggregating, modifying and forwarding data to RabbitMQ or storage)
- NoSQL database for data storage:
    - ElasticSearch data store. Open Source product widely used by various companies as well as HEP groups (UChicago, CERN, CMS DAQ, etc)
- OSG Display:
    - WEBUI unchanged, but modified to get information from ES
- GratiaWeb :
    - capabilities mostly unchanged but using well supported Open Source software like Grafana or Kibana
    - gets information from ES
- WLCG and XSEDE reports should be modified to get information ES
- Email reports
    - need to decrease amount of emails
    - preserve only needed emails in the “modern” format
    - get information fro ES

Phase I


Gratia V2 Prototype
1
ES Cluster
ES
ES
ES
ES-Client
3
Replay Agent
2
Summary Agent
1
Raw Listener
2
Summary Listener
probe
Gratia Collector Classic
4
Dashboard
Rabbit MQ
HTTP Listener
Tomcat
xml
Gratia Collector
1
Forwarder
Legend
Gratia Classic
New Development
Infrastructure
mySQL (read-only)
mySQL (write)

Phase IV



Development Schedule

Phase	Task	Subtasks	Assignees	Due Date
I	Building Prototype			7/15/2016
	Acquire 4 nodes to run ES, ES-Client	Provide hardware spec	Kevin/Joe	3/15/2016

	and some kind of Message Bus Service	Support hardware, provide access to fermilab team	Rob Quick	4/1/2016
	Setup ES cluster		Kevin/Joe	4/15/2016
	Data Aggregation	Design and implement a replay and summary agent, replay listener and summary listener	Derek?	6/30/2016
	Emulate GratiaWeb display using data from ES		Juan	7/01/2016
	Review of the prototype		ET, Technology Team	7/15/2016
||	Updating Gratia Auxiliary Services			8/01/2016
	OSG Display		Carl/Juan	7/30/2016
	Gratia Email Reports		Tanya	8/30/2016
	OSG/XSEDE		Tanya/Mats	8/30/2016
	OSG/APEL		Scott	8/30/2016
	Design and implement scripts that allow bulk deletion/correction of raw and summary records		Derek?	8/30/2016
	Execute repopulation raw data to ES		Kevin	8/30/2016
|||	Testing and Deployment			1/1/2017
	Run in parallel gratia and ES cluster and			1/1/2017

	all auxiliary services			
	Final review and results validation			12/15/2016
IV	Final Transition			
	Point probes to send data directly to RabbitMQ			03/01/2017
	Turned off collectors and retire MySQL nodes			03/01/2017

Deployment

- The following service will be deployed at GOC, Indiana
    - ES cluster (start with 4 nodes)
    - ES client
    - RabbitMQ
    - Custom scripts (listeners and agents)
    - OSG Display
    - GratiaWeb
    - WLCG report
    - XSEDE report
- The following service will be deployed at Fermilab:
    - email reporting (Why? Could we move it to GOC as well)?

Efforts:

- Juan until July
- Kevin
- Tanya
- GOC (Rob?,Scott?)
- Software Team (Derek)?

References:

1. Blueprint preparation doc

2. Frank's presentation at blueprint
3. Tanya's presentation at blueprint