REX Monitoring

Marc Mengel, Joe Boyd

# Introduction

Design and implement a REX–wide monitoring system for grid jobs, data handling, and some REX specific hardware tracking.

Goal is to have one unified monitoring system with visualization. The visualization is not coupled with the data collection.

Goal is to have “slurper” scripts that collect monitoring data.

Goal is to have a central RRD file archive (reliable, familiar, already in use for some monitoring)

Goal is to have web-based visualization

Goal is to have alarming for out of tolerance conditions

# Tasks

## Design and implement a downtime database

Be able to file planned and unplanned downtimes to correlate with affected monitoring data. Make downtimes show up on graphs and to quiet alarms during planned downtimes. People: Marc & Jeremy. Duration: two weeks. Starting: Mid-June.

## Javascript Graph Utility

Refactoring the Google Graphs stuff. Integrate with the downtime info. Investigate other technologies. People: Marc, Joe, Jeremy. Duration: two weeks. Starting: Mid-June

## Threshold info in browser page

Display an alert icon on the web page that shows which values at-a-glance are over threshold. People: Jeremy. Duration: 2 days. Starting: Mid June

## Updates to Joe's batch pages

Add information to the batch job information pages (e.g. experiment information). People: Joe. Duration. 2 weeks. Starting: ??

## Design and implement monitoring integration slurpers

How to pull in monitoring from other systems, like enstore, etc. Have a design meeting.

People: Marc. Duration 3 weeks. Starting: end of June

## Job state monitoring

Incorporate monitoring from IFDATA and other tools.

People: Marc & Jeremy. Duration: 3 weeks. Starting: mid-July

# Milestones

Release batch monitoring – End of May

Complete the visualization overhaul – End of June

Complete the downtime database and integrate into the main monitoring – End of June

Integrate with IFDATA information – End of July

Release Monitoring v1.0, tested and accepted by stakeholders – End of August

# Risks

## Sensitive information made public by the monitoring

A mitigation may be to use standard http authentication tools.

## Scalability risk

Could we have so many slurpers that the server cannot keep up? Last summer Jeremy did a stress test and found that the system could not handle > 1000 slurpers. Use syslog and other scalable tools to help with the slurping. Perhaps split the logging and analysis tasks onto multiple machines.

# Conclusions

When this is complete, we have comprehensive REX specific monitoring so we can notice issues before they become problematic. The system will be flexible such that we can add new monitoring items easily.