



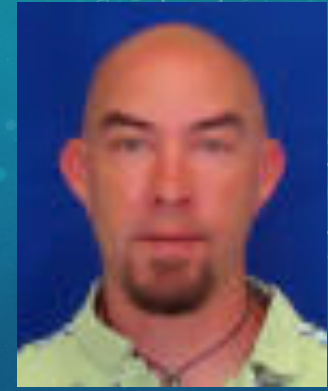
MONITORING GERRIT

STEPHEN V WILLIAMS

INTRODUCTION

The usual blurb

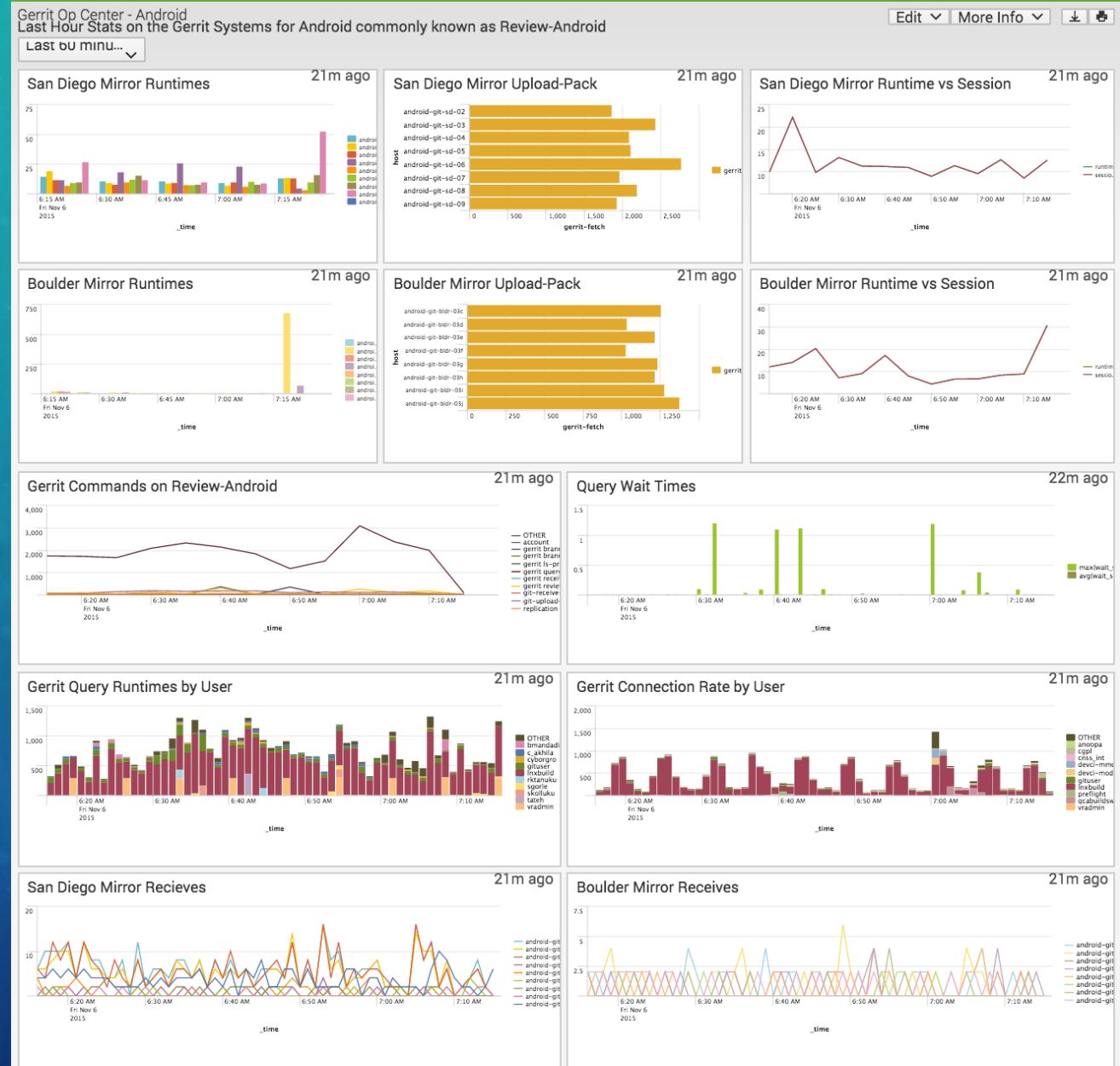
- Git & Gerrit Admin at Qualcomm
- Admin Team:
 - Earl Gresh, Aaron Crossland
- Developer Team:
 - Nasser Grainawi, Martin Fick, Kevin Degi, Craig Chappel, Raviteja Sunkara



MONITORING PHILOSOPHY

to Err is Human ... to really foul things up requires a computer

- Inventory Supplement
- Operation Center
- Fast and Loud
- 1% is better than 0%



MONITORING SOLUTIONS

a little bit of everything

- CheckMK / Nagios
- Splunk
- Elastic Search
- Cron Jobs



CRON JOBS

- Basic / Rudimentary / Step 1
- Email Warnings

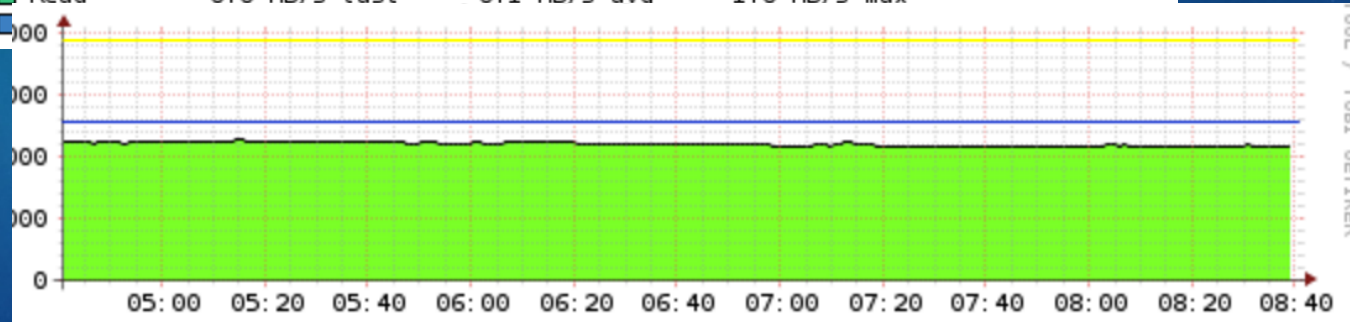
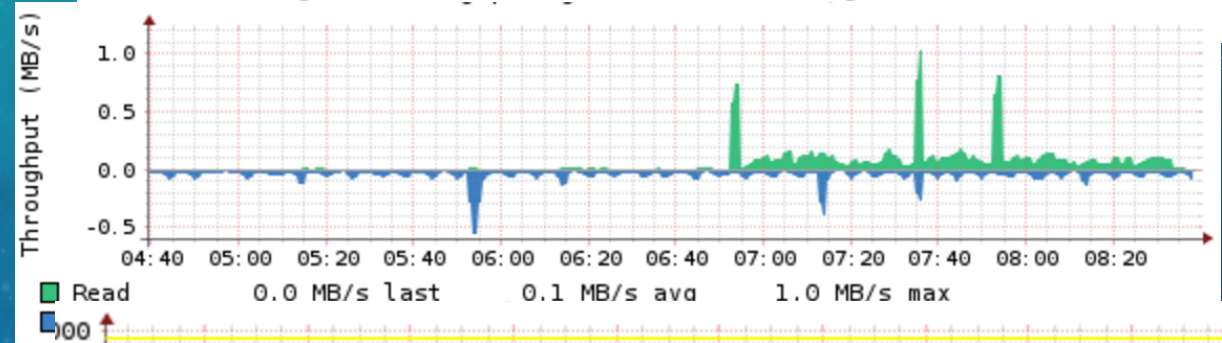
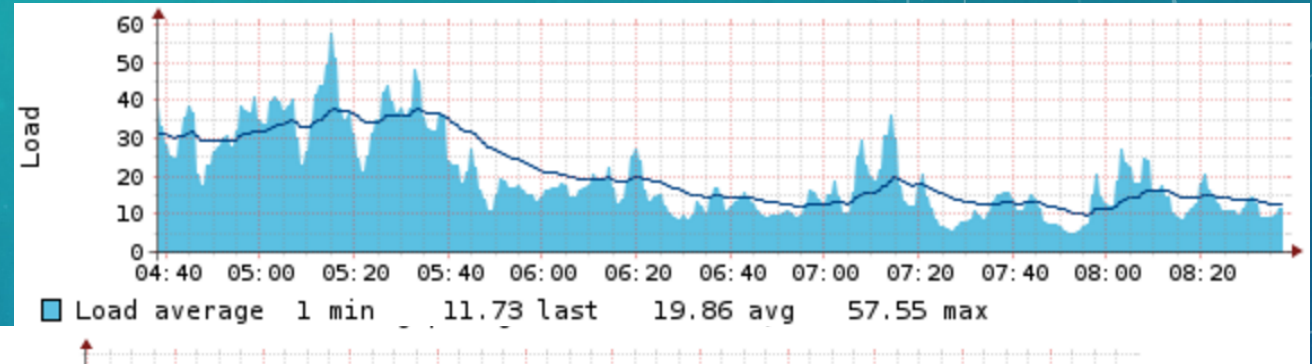
```
# gerrit user crontab  
MAILTO=lords-of-gerrit@qualcomm.com
```

```
# check for corrupt repositories daily  
15 5 * * * /srv/admin/gerrit-fsck.sh
```

CHECK_MK / NAGIOS

Std Admin Toolkit

- Standard Monitoring of Hardware
 - CPU Usage
 - Memory
 - Disk
 - I/O
 - Network
- Better form of cron job monitors



CHECK_MK / NAGIOS

Gerrit Admins Toolkit

- Greedy Upload-Packs
- Stuck Threads
- Long Running Jobs

Developer: “hmmm I need this really fast, I’m in a hurry... repo update –j 1000000000000”

Developer: “I don’t like my regional mirror... lets clone from the other side of the globe”

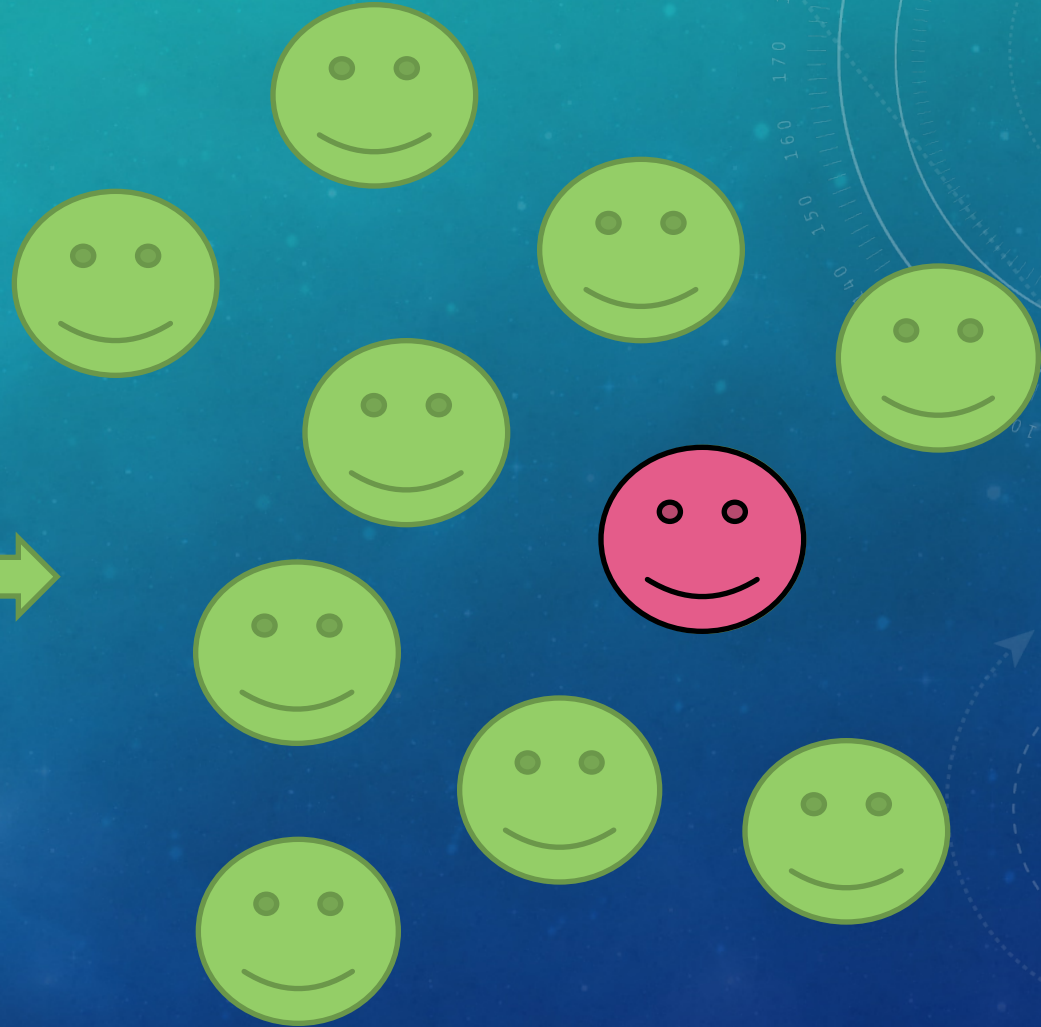
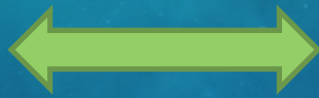
Developer: “I’m going to create a cron tab to sync code every hour... no need to make sure it completes”

SPLUNK & ELK STACK

Data Mining Logs

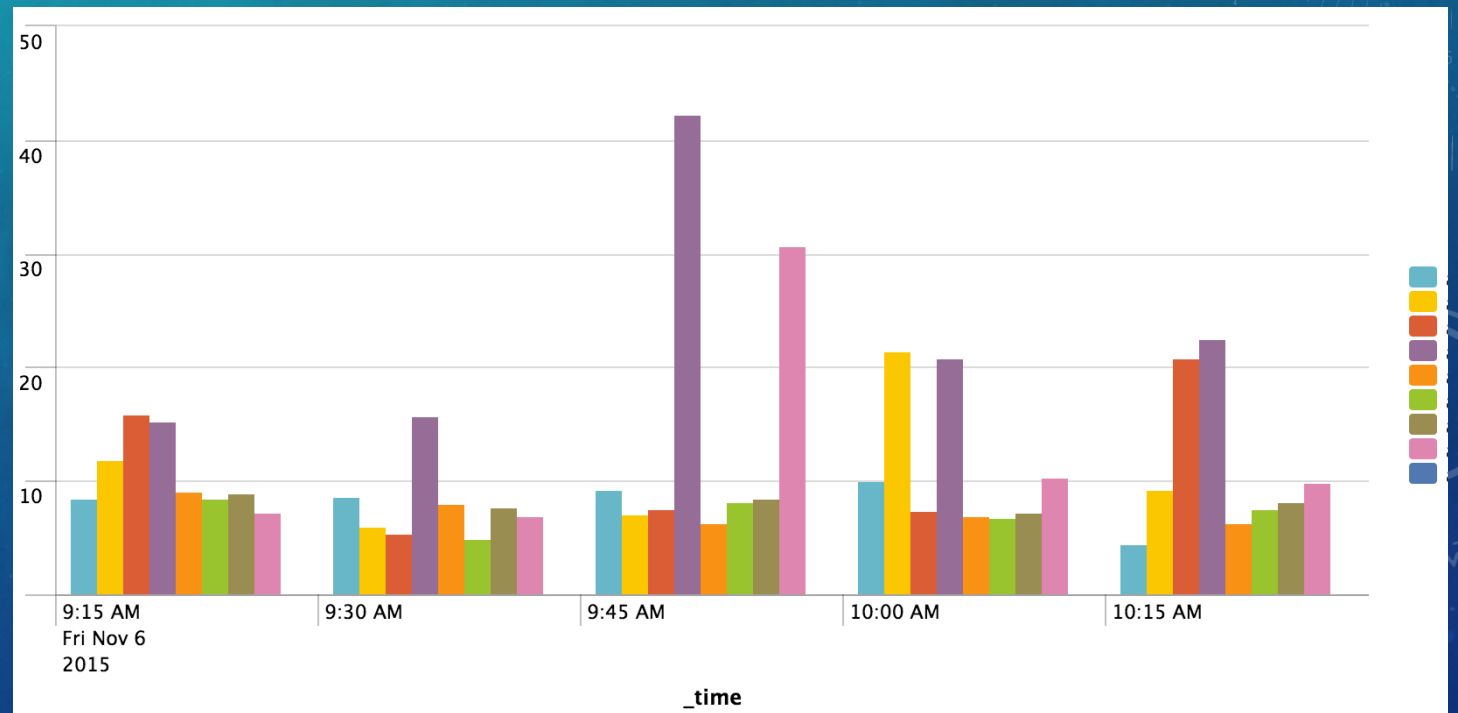
MONITORING SLAVES

- Runtime
- Upload Pack Count
- Receive Pack Count
- Runtime vs Session Time



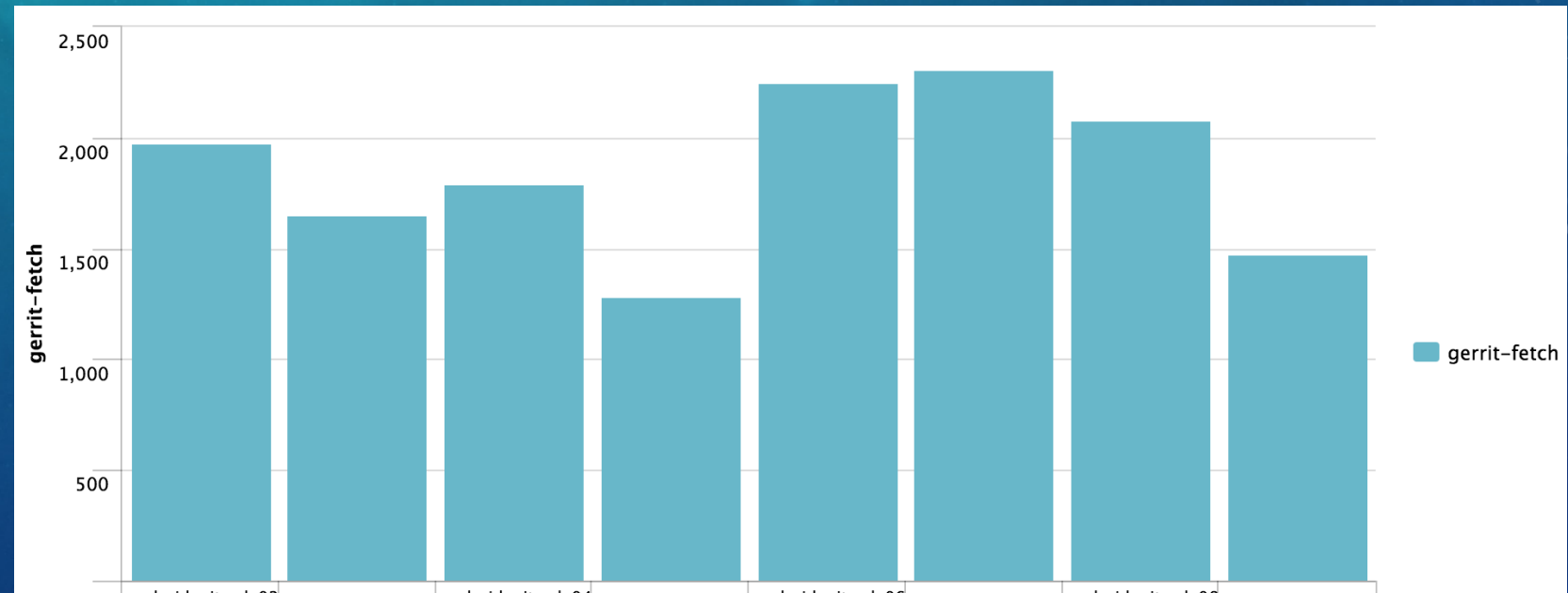
RUN TIME

```
host=git-sd*  
sourcetype=ssherror_gerrit |  
search cmd_gerrit!="gerrit ls-projects" |  
timechart  
  span=15min  
  avg(runtime_seconds) by host
```



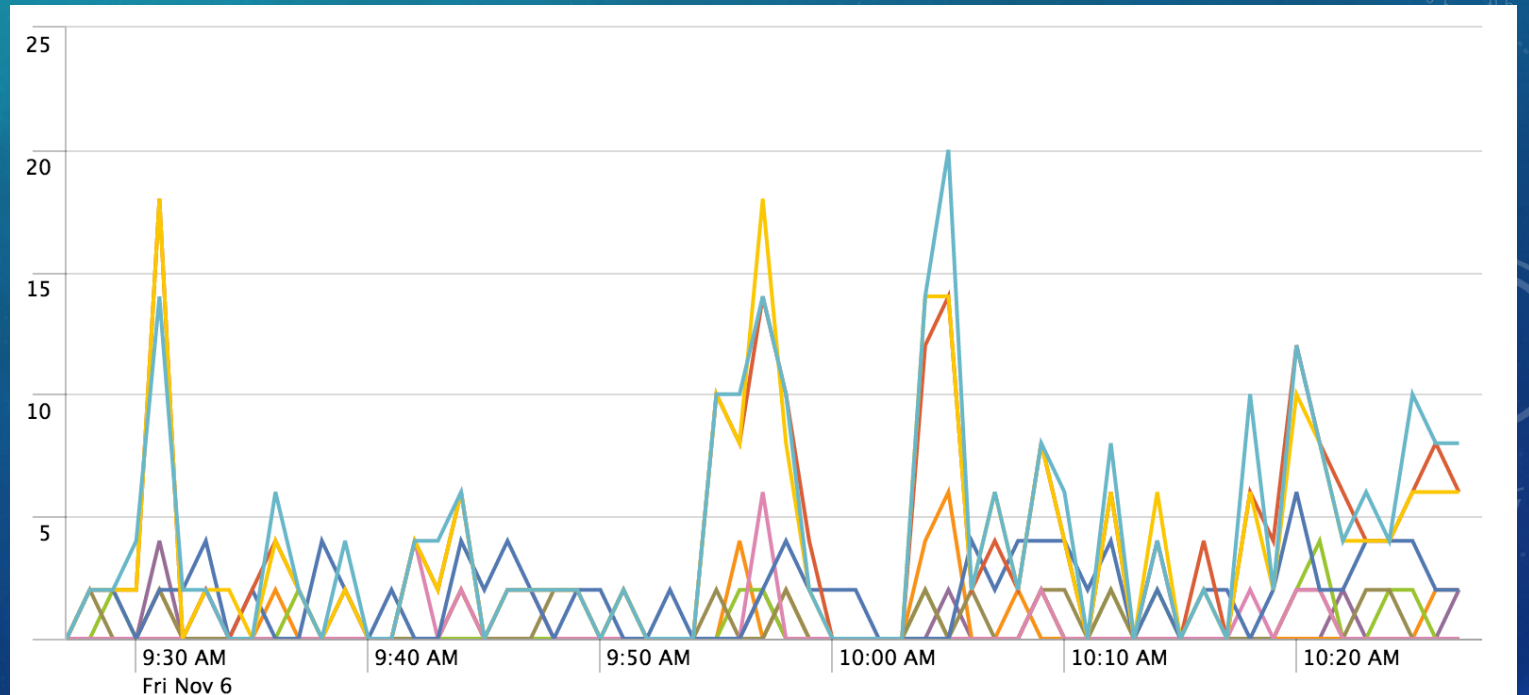
UPLOAD-PACK COUNT

```
host=git-sd*  
sourcetype=ssherror_gerrit  
cmd_gerrit="git-upload-pack" |  
stats count as gerrit-fetch by host
```



RECEIVE-PACK COUNT

```
host=git-sd-*  
sourcetype="syslog"  
"Request receive-pack for" |  
timechart count as recieve-pack by host
```



RUNTIME VS SESSION TIME

host=git-sd*

sourcetype=ssherror_gerrit

cmd_gerrit="git-upload-pack" |

timechart span=5m

avg(runtime_seconds) as runtime

avg(runtime_total) as session



MASTER MONITORING

- Commands
- Query Wait Times
- Connection Rates
- Connection Runtimes
- Java GC
- Repacking



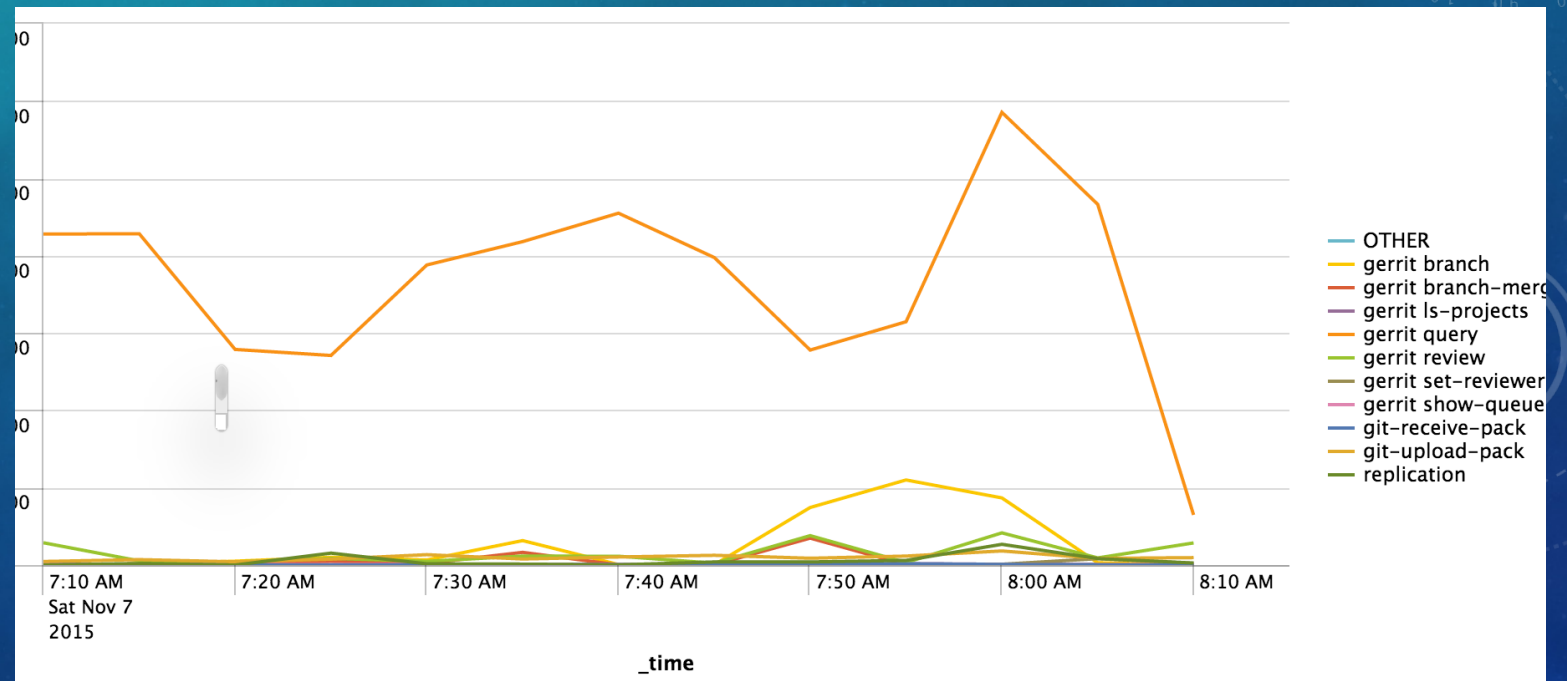
COMMANDS

Host=master

sourcetype=ssherror_gerrit

cmd_gerrit=* |

timechart span=5min count by cmd_gerrit



QUERY WAIT TIMES

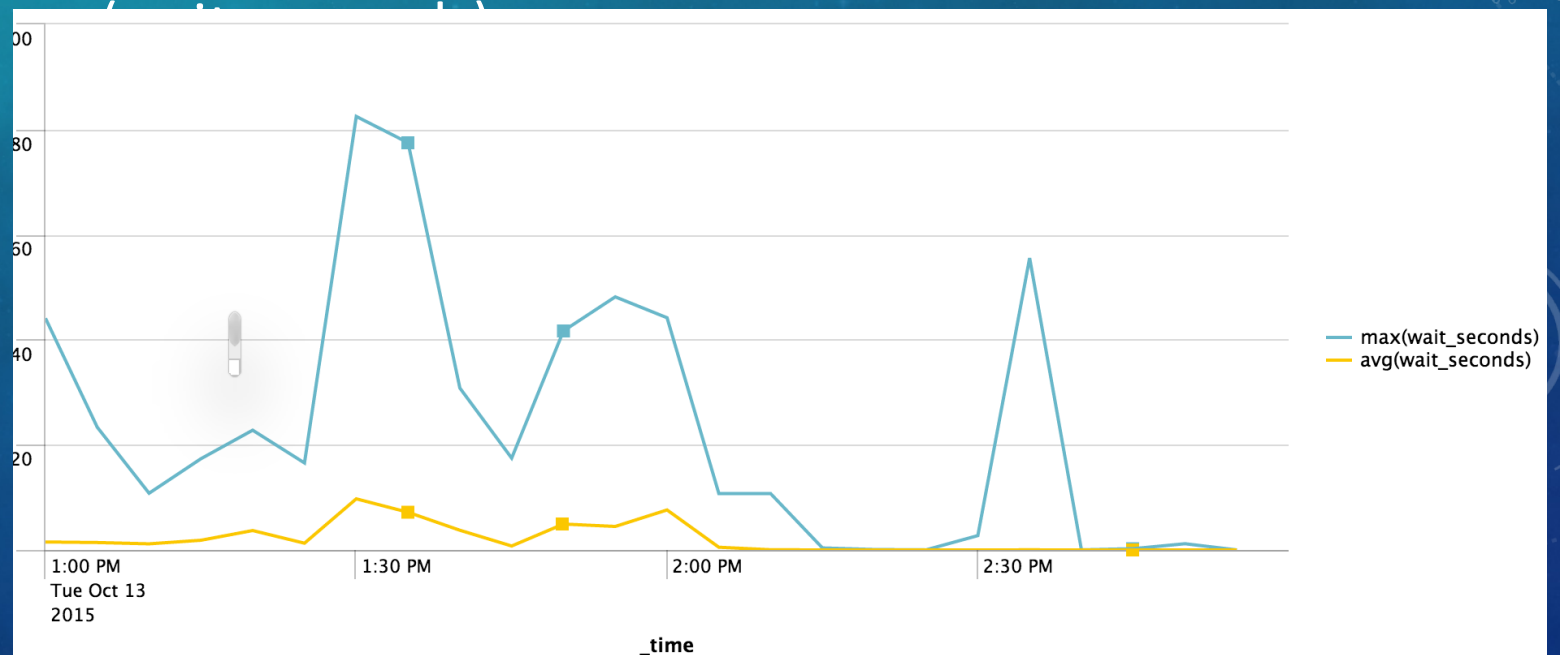
Host=master

```
sourcetype=ssherror_gerrit |
```

```
search cmd_gerrit="*" |
```

```
search cmd_gerrit!="gerrit stream-events" |
```

```
timechart max(wait_seconds)
```



CONNECTION RUNTIMES

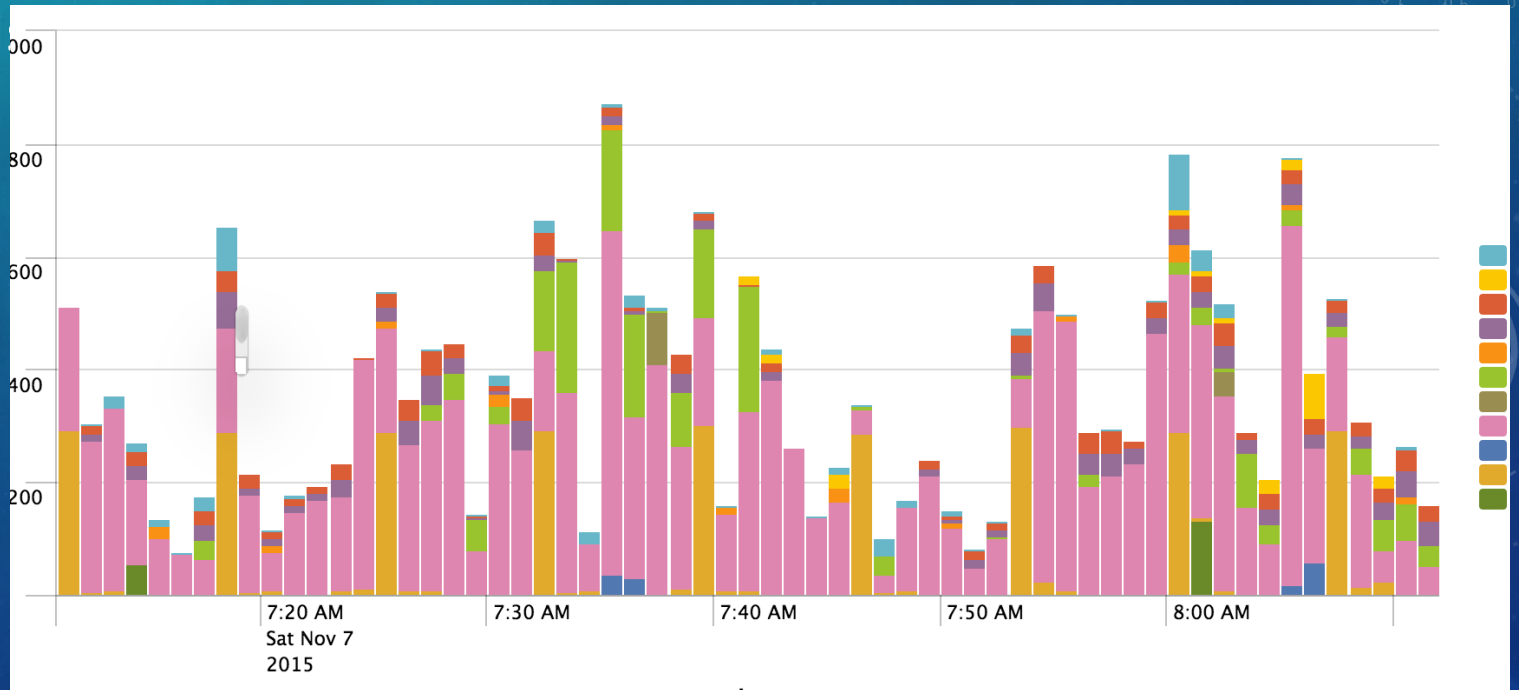
host=master

sourcetype=ssherror_gerrit |

search cmd_gerrit="*" |

search cmd_gerrit!="gerrit stream-events" |

timechart sum(runtime_second



JAVA GC

host=master

sourcetype=jvm_gerrit "Full GC" |

timechart span=1m

max(YoungGen_before)

max(YoungGen_after)

max(OldGen_before)

max(OldGen_after)

max(PermGen_before)

max(PermGen_after)

-verbose:gc -XX:+PrintGCDateStamps -XX:+PrintGCDetails



REPACKING

Host=master

sourcetype=repacking_git

repack_keeps_before=* OR repack_packs_before=* |

timechart span=1d

min(repack_keeps_after),
max(repack_keeps_before),
min(repack_packs_after),
max(repack_packs_before)

