Monitoring Evolution at CERN

pedro.andrade@cern.ch
it would be nice to have a dashboard with my hosts and services… which I could share…

the 10-years report of my service write operations per week shows that…

for sure I can do something useful with this temperature and humidity data…
Monitoring Reality
Old tools need replacement

... not scaling to current needs
... old code difficult to maintain
... only providing static displays
Identify
what you want to do with the data

Simplify
the data processing workflow

Rely
on existing technologies & standards
Identify
what you want to do with the data
Alerts
to get notified of problems affecting nodes/services

Archives
to create a data lake of monitoring events for offline analysis

Displays
to plot relevant monitoring data in realtime

Streaming
to programmatically process monitoring data
Simplify
the data processing workflow
Based on the **lambda architecture** data processing principles: stream & batch

Capable of sustaining **large quantities** of timestamped events: metrics and logs

Common to all monitoring producers, easy to replace individual components
Rely
on existing technologies & standards
- technology watch
- follow upstream releases

- osquery
- sensu
- kafka
- grafana
- hue
- ...

graphite
ActiveMQ
hadoop
Flume
kibana
Spark
Distributed service for collecting large amounts of data, robust and fault tolerant, horizontally scalable, multi-tier deployment, many ready to use input and output plugins.

Running Flume 1.5.0
- scalable 3 tiers deployment
- one flume agent in each DC node (~15k)
- 50 VMs aggregating data, writing to HDFS
Distributed framework for large data sets processing, distributed fault-tolerant filesystem for low cost hardware, suitable for applications with large data sets

Running HDFS 2.5.0 (CDH 5.3.1)
- dedicated 16 nodes cluster
- data being archived since 2013
- currently storing ~32 TB data
Distributed search and analytics engine, real time acquisition, data indexed in real time, automatically balanced shards and replicas, schema free, document oriented

Running **ElasticSearch 1.4.4**
- 3 masters, 4 search, 20 data nodes
- 1 week raw data, 1 year history
- 10 shards/index, 2 replicas/shards
**what**

Distributed large-scale data processing engine, improvement of MapReduce, support for streaming, SQL, machine learning, and graph processing

Running **Spark 1.2.0 (CDH 5.3.1)**

- prototyping on different use cases
- data centre temperatures and power
- detection of nodes “no contact”
Modular lambda architecture
Appropriate technologies
=
Useful monitoring solutions for CERN
IT Service Managers and Experiments
In [1]: from pyspark import SparkContext, SparkConf

In [2]: conf = SparkConf()
   .setAppName("TempPlotting")
   sc = SparkContext(conf=conf)

...

In [9]: from matplotlib.colors import LogNorm
from pylab import *

def plot_data(rdd):
    x = rdd.map(lambda sample: sample[0]).collect()
    y = rdd.map(lambda sample: sample[1]).collect()

    hist2d(x, y, bins=40, norm=LogNorm())
    colorbar()
    show()

In [10]: plot_data(bound_wf_power_temperature)
Clear architecture & modular technologies

Different solutions for alerting, streaming, displays, and archive

Successfully deployed and running in production for +1 year
Thank you !!!

http://cern.ch/itmon
itmon-team@cern.ch