A Dashboard is Worth a Thousand Words

Better Monitoring for Better Ops

Luca Magnoni – Computing Engineer/CERN IT
@lucamag
About myself

- Software Engineer
- > 10 years in distributed systems & data intensive applications
- Service Manager & Project Architect
CERN’s primary mission: **SCIENCE**

Fundamental research on particle physics, pushing the boundaries of knowledge and technology
CERN

World’s largest particle physics laboratory
The Large Hadron Collider: LHC

27km
1232 dipoles
15 mt, 35t EACH

Image credit: CERN
LHC: World’s Largest Cryogenic System (1.9 K)

COLDER TEMPERATURES than outer space

(120t He)
LHC: Highest Vacuum

104 km of PIPES

$10^{-11}$ bar (~ moon)

Image credit: CERN
LHC Detectors

The Most SOPHISTICATED DETECTORS ever built
ATLAS, CMS, ALICE and LHCb

HEAVIER than the EIFFEL TOWER
~100 Mpixel
CAMERA
40 millions
pictures per second
Data Acquisition: What to Record?

Looking for Rare Events

1 billion Collisions per second
1PB/s Generated data

(10^{-13})
Data Flow to Storage and Processing

**RUN 2**

- **ALICE:** 4GB/s
- **ATLAS:** 1GB/s
- **CMS:** 600MB/s
- **LHCb:** 750MB/s
CERN Data Centre: Primary Copy of LHC Data

- 90k disks
- 15k servers
- > 300 PB on TAPES
About WLCG:
- A community of 10,000 physicists
- ~250,000 jobs running concurrently
- 600,000 processing cores
- 15% of the resources are at CERN
- 700 PB storage available worldwide
- 20-40 Gbit/s connect CERN to Tier-1s

Tier-0 (CERN)
- Initial data reconstruction
- Data distribution
- Data recording & archiving

Tier-1s (13 centres)
- Initial data reconstruction
- Permanent storage
- Re-processing
- Analysis

Tier-2s (>150 centres)
- Simulation
- End-user analysis

170 sites
WORLDWIDE
> 10000 users
Explore more than 1 petabyte of open data from particle physics!

Search examples: collision datasets, keywords:education, energy:7TeV

Explore
- datasets
- software
- environments
- documentation

Focus on
- ATLAS
- ALICE
- CMS
- LHCb
- OPERA

Get started
1958 First Electronic Computer Installed
About IT Department

- Over 300 people
- Enable the laboratory to fulfill its mission
- Data Centre and more
  - supports IT Services (Batch, Storage, Network, DB, Web Servers, etc.), Experiments Services (SW builds), Engineering (Chip design), Infrastructure (hotel, bikes), Administration
IT Infrastructure / ~ Recent Days
~ 10 years of Data Taking

2018: 88 PB
ATLAS: 24.7
CMS: 43.6
LHCb: 7.3
ALICE: 12.4
~ 2001 / Custom Fabric

- EU funded developments
- Scale and experience for LHC was special
- Custom tools had to be developed to manage infrastructures at scale
2013 / Opensource Tools

- LHC requirements kept growing (on flat budget)
- But CERN scale no longer special (e.g. Google, Facebook, Rackspace,...)
- The rise of the Clouds
~ 2013 / Opensource Tools

- Tool-Chain approach
- Embrace Opensource Communities
- Focus on Resource Provisioning, Configuration and Monitoring
CERN Data Centre: Private Openstack Cloud

More Than 300,000 cores
2019 / Even more *Tools* …

- Containers / Kubernetes
  - New deployment models
- More Clouds
  - Hybrid workflows
- *SRE* ?
Monitoring “all the things”
Monitoring Mission

- Provide **Monitoring as a Service** for CERN Data Centre (DC), IT Services and the WLCG collaboration
  - e.g. Dashboards, Alarms, Search, Archive
- Collect, transport, store and process metrics and logs for applications and infrastructure
Challenges / Rate & Volume

- from ~ 40k machines
- > 3 TB/day (compressed)
- ~ 100 kHz
Challenges / Variety

- > 150 producers
Challenges / Reliability

- spikes in rate and volume
Non-Technical Challenges

- Migrate people from legacy (custom) dashboards and tools
- Stay up to date with upstream tools & trends
- Build community, internal and external
How to provide better monitoring?

• ~ 2016 a new project started to provide a new central monitoring infrastructure to CERN IT

• Goal:
  • Effective
  • Scalable
  • Sustainable
Easy Data Integration / Telemetry

- Collectd
  - lightweight / plugin based
  - ~ 40k machines
- JSON/HTTP gateways
  - custom metrics and logs
- Prometheus
  - Kubernetes
Responsive / Multiple Backends

- **Elasticsearch**
  - search and discovery
  - 3 clusters, ~100 TB
- **InfluxDB**
  - time-series data
  - > 30 instances, 60 kHz
- **HDFS**
  - long-term archive
Data Integration is **hard**

- **Metrics** → TSDB
- **Logs** → search/index
- **All** data → archive
- **Some** Metrics → search/index
- **Some** Logs → TSDB
- **Users**: “btw, where can I tap in to get my data?”
Monitoring Pipeline

- Enrichment
- Transformation
- Metrics Extraction
On the pipeline approach

- Provides key functionalities:
  - decouples producers / consumers
  - enables stream processing
  - resilient (72 hours data retention)

- Kafka cluster:
  - on-premises (v 1.0.2)
  - Openstack VMs with Ceph volumes
  - ~15k partitions in total
Dashboards & Visualizations

- Critical for the success of the project
- Need to delegate control to users
- Multiple tools
Dashboard / Grafana

- Recommended tool for Dashboard
- Multiple Backends
- Customizable (menus, filters, etc.)
40k hosts
Network metrics enriched with DC Router Topology
WLCG transfers classified by location, country, site…

7 millions/day
Logs, Metrics combined to extract High-Level Information
Alarming

- Local (on the machine)
  - Simple Threshold / Actuators
- On Dashboards
  - Grafana Alert Engine
- External (Spark, Spectrum, etc.)
- Integrated with ticketing system
  - ServiceNow
Monitoring Technologies
Successful story

- Monitoring by Numbers:
  - ~ 900 Active Users
  - > 1000 Dashboards
  - ~ 1000000 Queries/day
  - > 30 Grafana Orgs
  - service operations, debug, troubleshooting, etc.
- Next is to profit at best from all this data
Better Ops
SRE Key Points of Interest

- Common framework for production-systems management
- Reduce operational load
- Formalize best-practices for software velocity and quality
SRE Practices / Good fit with CERN IT

• SRE Culture
  • Openness, Sharing
  • Joint-Ownerships / Accountability
    • “one person’s symptom is another person’s cause”
• Sustainability
  • Attracts skills, Career opportunities
Monitoring like SRE

- Goal: build common language & culture
- Introducing SLI & SLO
- One Dashboard at the Time
Introducing SLI / Strategy

- Build critical mass of early adopters among main IT services
- Work with service managers to extract the relevant data (some indicators already there…)
- Try to solve a real problem to help the idea spread faster
  - i.e. improve Service Availability reporting
Tackling Service Availability

- True stories from chat / mattermost snippets:
  - Is there a problem with service X?
  - Is anybody else having issues today?
  - I think service X is slow
  - I think service X is having some issue lately
  - How our service evolved? Are we doing better?
Tackling Service Availability / 2

- Availability Metric exists
- Not easy to get actual health of services today
- Would benefit from more precise and quantitative measurement
SLI Overview Dashboard

- SLI as user-facing metrics
- Focus on Golden signals, such as
  - Write/Read Latency for Storage systems
  - Rate of cloud API requests
  - Rate of batch server occupancy
  - Catalog compilation time
- Dashboard: visual feedback for users
### Overview Dashboard

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message Received</td>
<td>3.70 kpps</td>
</tr>
<tr>
<td>Message Sent</td>
<td>8.08 kpps</td>
</tr>
<tr>
<td>RTT Queue</td>
<td>18.7 ms</td>
</tr>
<tr>
<td>RTT Topic</td>
<td>13.88 ms</td>
</tr>
<tr>
<td>Grafana Queries / Average per Minute</td>
<td>899</td>
</tr>
<tr>
<td>Data Rate</td>
<td>94.3 kpps</td>
</tr>
<tr>
<td>End-to-End Log Latency</td>
<td>4 seconds</td>
</tr>
<tr>
<td>End-to-End Collected Latency</td>
<td>50 seconds ago</td>
</tr>
<tr>
<td>Service Availability in the General Purpose Region</td>
<td>98%</td>
</tr>
<tr>
<td>Average Boot Time (without DNS)</td>
<td>26 seconds</td>
</tr>
<tr>
<td>Average Boot Time (with DNS)</td>
<td>8 minutes</td>
</tr>
<tr>
<td>Catalog Compilation Time</td>
<td>29.0 s</td>
</tr>
<tr>
<td>Oldest Queued File for Migration</td>
<td>4.8 hour</td>
</tr>
<tr>
<td>Oldest Queued File for Recall</td>
<td>4.1 hour</td>
</tr>
<tr>
<td>Running Transfers</td>
<td>119</td>
</tr>
<tr>
<td>Ceph I/Ops</td>
<td>26900</td>
</tr>
<tr>
<td>Throughput</td>
<td>2.22 GBs</td>
</tr>
</tbody>
</table>

---

SLI
SLI
Drill-Down per Customer Service
Dashboard / Grafana

- **Single Stat** panels
- Drill-down on details dashboards per category
- Threshold as SLO
Tracking SLO

- Check SLI vs defined SLO
- Grafana Rules to generate alarms
- Webhook to HTTP
SLO Driven Operations

- Alert on SLO miss
  - Care about symptoms first
- Build Performance Trend
- SLO-Driven Availability
Technical Challenges

- Grafana Alert Engine
  - Rely on TSDB capabilities
  - Prometheus fairly advanced, InfluxQL has some limitation, Flux should solve

- Black-box vs White-box
  - white-box fits the usual metrics flow, black-box may benefits from common framework for probing
Non-technical challenges

• Service dependencies
  • “Not my fault”
• Big debate on user-related metrics as SLI
• Bottom-Up approach
Lessons Learned

• Dashboard-first approach works
• SLI & SLO are good starting points
• Cultural change
  • target people more than technology
Conclusion

• Successful migration to modern opensource monitoring stack and common practises
• SRE framework and culture proved to be a good direction for service operations evolution
• Just at the beginning of the SRE journey, looking forward for the next steps