Analysis of CERN Computing Infrastructure and Monitoring Data

Christian Nieke, CERN IT / Technische Universität Braunschweig

On behalf of the CERN IT Analytics Working Group
IT Analytics Working Group

- **Goals:**
  - Coordinate analysis and trending of application/service usage data
    - E.g. batch computing, data storage, network...
  - At different stages of maturity
    - Getting a quantitative understanding of a service (exploratory)
    - Informing strategy or planning decisions (hypothesis check)
    - Developing & validating predictive models
Data Sources - Before

No common analysis goal
No common schema
No common format
No common repository
No shared documentation
No easy way of joining
Getting the Big Picture

- Combined Activity
  - Enable integrated studies crossing single data source / service boundaries
  - Using a common base repository of prepared input data
  - Provide an exchange forum for discussion on analysis methods, tools and result validation
Common Repository

- Data Warehouse
  - Write once, read many
- Hadoop cluster
  - Raw files in any format
  - Using Hadoop jobs for cleaning and pre-processing
  - Export in CSV, Avro, Parquet, … for Analysis
## Example: EOS file system operations

### Processed Parameters

| Name | Type  | Link | Description                                           | Purpose                                                                 | Remarks                                                                                   |
|------|-------|------|-------------------------------------------------------|--------------------------------------------------------------------------|------------------------------------------------------------------------------------------|
| td   | String| -    | client trace identifier \(<<username>>.<<pid>>@<<client-host>>\) | This identifier can be used to aggregate sessions \(\text{of the same user, with the same job on the same machine}\) |                                                                                           |
| path | String| -    | Full namespace path to the file                       | The folder structure in the path could be usefulfull to match the file to a project/user/programm etc. |                                                                                           |
| ruid | Int   | -    | mapped unix user id                                   | Allows aggregation by user                                              | ruid==1 means root process, which is usually an internal process like rebalancing, draining etc. |
| rgid | Int   | -    | mapped unix group id                                  | Allows aggregation by group                                              |                                                                                           |
| host | String| **LanDB** | name of the disk server serving the file               | Allows aggregation by user host and links to LanDB                      |                                                                                           |
| fid  | Int   | -    | EOS file id                                           | Allows aggregation by file                                              | unique per EOS instance (e.g. eosatlas, eoscms,...)                                      |
| fsid | -     |      | EOS file system id, e.g. disk                         | Allows aggregation of files by file system (disk)                       |                                                                                           |
| ots  | Date  | -    | File open time as unix timestamp \(\text{in seconds since January 1st, 1970 at UTC}\) | Aggregation by time                                                      |                                                                                           |
Data Sources - Federation

Diagram showing the integration of data sources:
- **Batch Jobs**
- **Batch Nodes (Hardware and Configuration)**
- **Network**
- **Experiment Dashboards: Job Monitoring**
- **Experiment Dashboards: Data Transfers**
- **Experiments: File Popularity**
- **Data Storage Operations**

Key connections:
- Scheduler-Id
- Job-Id
- Host name

The diagram illustrates how these components interact within the Hadoop ecosystem.
Example Analysis Workflow

- Job Performance: Geneva vs. Budapest
  - Different computing centers
  - Different hardware
    - CPU, Memory, Network, ….

- Do we get the same performance?
  - Compare CPU time used per job
CPU Time and Location

- Based on batch computing logs and network configuration

We need more information to understand this distribution.
Tasks

- Based on experiment job dashboard

Different distributions for different tasks
Tasks

• Selecting a single task

Let’s randomly select this one
Tasks

• It seems like there are still more underlying effects

CMS Jobs - Subset January + February 2015, restricted to one task: 29,965 items

This is not just a simple shift
HepSpec Benchmark

- HepSpec Factor based on batch benchmarks

High benchmark result is correlated with low CPU time
Scaling by CPU Factor

- Removes “expected” deviation

CMS Jobs - Subset January + February 2015, restricted to one task: 29,965 items

Now this looks like an answer.

But what do we actually see?
- Job specific?
- AMD vs. Intel?
- Network delay?
- Data placement?
Conclusion

• Combined Effort
  • CERN IT and Experiments
  • Federated data repository for uniform access
  • Understanding the system as a whole

• Examples for Actions Taken
  • Rebalancing batch slots per machine to avoid swapping
  • User notification in case of inefficient jobs
  • Activated TTreeCache for ROOT in ATLAS
Resources

• Twiki
  - https://twiki.cern.ch/twiki/bin/view/ITAnalyticsWorkingGroup/WebHome

• Contact:
  - Dirk Duellmann, CERN IT (Working Group Chair)
  - or myself