SWAN as interface to Hadoop clusters

Enric Tejedor for the SWAN team

https://cern.ch/swan

Hadoop User Forum
December 4th, 2023
Introduction
SWAN in a nutshell

➤ Interactive analysis with a web browser
  ▪ No local installation needed
  ▪ Based on Jupyter Notebooks
  ▪ Calculations, input data and results “in the Cloud”

➤ Easy sharing of scientific results: plots, data, code

➤ Good for data analysis, exploration and also teaching

➤ Integration with CERN resources → added value!
  ▪ Software (CVMFS)
  ▪ Storage (EOS, CERNBox)
  ▪ Computing (GPU, Hadoop / Spark, HTCondor)
SWAN as entry point to CERN computing resources
SWAN-Hadoop integration
SWAN integration with Hadoop / Spark

- SWAN is connected to Hadoop & Spark clusters at CERN
  - Physical
    - Analytix: general purpose
    - NXCALS: dedicated
  - Virtual: kubernetes cluster

- Jupyter extensions available to:
  - Connect to a certain cluster and spawn Spark executors
  - Monitor the execution of Spark jobs

- [Link](#) to Spark training on SWAN
Spark Connector

- **Spark Connector** – configure and establish a connection
  - Uses upstream configuration provided by Hadoop team
  - Provides bundled configurations tailored to custom use cases and integrations: EOS, S3
  - Allows to specify additional Spark configuration
  - Connects to cluster and allocates executors
  - Provides link to Spark WebUI
Spark Connector – configure and establish a connection

- Uses upstream configuration provided by Hadoop team
- Provides bundled configurations tailored to custom use cases
- Integrations: EOS, S3
- Allows to specify additional Spark configuration
- Connects to cluster and allocates executors
- Provides link to Spark WebUI
Spark Monitor – Spark monitoring from notebooks

- For live monitoring of Spark jobs spawned from the notebook
- Displays jobs, stages and tasks
- A graph shows number of active tasks & executor cores vs time

<table>
<thead>
<tr>
<th>Job ID</th>
<th>Job Name</th>
<th>Status</th>
<th>Stages</th>
<th>Tasks</th>
<th>Submission Time</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>toPandas</td>
<td>RUNNING</td>
<td>0/2 (1 active)</td>
<td>4/4/2011</td>
<td>a few seconds ago</td>
<td>-</td>
</tr>
</tbody>
</table>
Spark Metrics – Spark dashboard for advanced troubleshooting

- Custom CERN development, integrating Spark instrumentation with InfluxDB and Grafana services at CERN
- Displays CPU usage, active tasks, memory, shuffle, I/O, Java GC, etc
- Real-time and historical data
- Opt-in configuration (use it when troubleshooting)
Software provisioning: CVMFS

- Software is provisioned via curated stacks on CVMFS (LCG releases)
  - General-purpose: contain Spark, Java, Python
  - NXCALS: layered
    - Base layer: general-purpose, fixed
    - Top layer: NXCALS software, updated every day (enables faster development cycles)
Current usage

User session stats for SWAN kubernetes infrastructure, i.e. only half of the user load, last month

- Analytix: peak of ~10
- NXCALS: peak of ~30
Future plans
JupyterLab

> Latest interface proposed by project Jupyter
  - Notebooks, terminals, …
  - … and virtually anything via extensions

> Will be offered as an option in test mode, before end of 2023
  - Will initially coexist with old UI
  - To be made default in the future

> Spark extensions have been migrated to JupyterLab
  - Connector, Monitor
Spark Connect

- New client-server architecture introduced since Spark 3.4
  - Allows to spawn a server to which clients can connect

- Under investigation in SWAN
  - Allows multiple notebooks (clients) to share the same driver (server) and executors

Classic architecture (current in SWAN)

```
Notebook 1 → Driver 1 → Spark cluster 1
...
Notebook N → Driver N → Spark cluster N
```

Spark Connect architecture

```
Notebook 1 → Spark Connect API → Spark’s Driver
...
Notebook N → Spark Connect API → Spark’s Driver
```

Spark Connect

- Multi-tenant Application Gateway
- Analyzer
- Optimizer
- Scheduler
- Distributed Execution Engine
Thank you