BIG DATA & MACHINE LEARNING
SUPPORT SERVICES
OPERATIONS DEPARTMENT

Carlos Tripiana Montes
Resources

MareNostrum 4 (main cluster).
- 11.15 PetaFLOPs.
- 3,456 nodes.
- 48 cores, 96 or 384GB RAM, OmniPath.
MareNostrum 4 (CTE- Power9 cluster).
- 52 nodes.
- 40 cores (160 threads), 512GB RAM, 2 Tesla V100, 1 EDR Infiniband.
MinoTauro cluster.

- 39 new nodes.
- 16 cores, 128GB RAM, **2 Tesla K80**, 1 FDR Infiniband.
Nord 3 (HW from MareNostrum 3).

- 28 GigaFLOPs.
- 84 nodes.
- 16 cores, 128GB RAM, 1 FDR Infiniband.
Services

**Big Data & ML 4 HPC:**
- Installation & maintenance of **Big Data & ML tools/stacks**.
- Develop necessary **tools** to adapt **Big Data clusters** in HPC envs.

**Advising (and best practices):**
- **Code** development.
- **Data** management and formatting.

**Collaboration with researches:**
- Applied Learning Methods.
- Big Data Frameworks.
- Data-Center Optimization.
- Data-Centric Architectures.
- Internet of Things and Stream Processing.
Applications

- Hadoop.
- Spark.
- Cassandra.
- Hive.
- TensorFlow.
- Caffe.
- Theano.
- … (Sonnet, Lasagne, Scikit-Learn, Keras, PyTorch).
- Virtually anything you need (and request).
MN4, CTE-Power, MT, Nord3.

```
~> module av python
```

/gpfs/apps/NVIDIA/modules/modulefiles/tools

python/2.7.10  python/2.7.12_ML  python/2.7.3  python/3.2.3
python/3.5.2-INTEL  python/3.6.0_ML  python/3.6.0_spark
python/2.7.12-INTEL  python/2.7.2(default)  python/2.7.8  python/3.5.0
python/3.5.2_ML  python/3.6.0+_ML  python/3.6.3_ML

We try to keep all the basic tools on every stack, and keep them updated.
Python ML Stacks

MN4, CTE-Power, MT, Nord3.

```
$ module show python/3.6.3_ML
/gpfs/apps/NVIDIA/modules/modulefiles/tools/python/3.6.3_ML:
[...]
prereq K80/default
prereq impi/2018.1
prereq mkl/2018.1
prereq cuda/8.0
prereq CUDNN/7.0.3
[...]
```

```
$ module purge && module load K80/default impi/2018.1
        mkl/2018.1 cuda/8.0 CUDNN/7.0.3 python/3.6.3_ML
```
Spark4HPC: Overview

A fast, and general engine to deploy Spark.
Spark4HPC: Overview

A fast, and general engine to deploy Spark.
- Cluster portability (MN4, MT, Nord3).
Spark4HPC: Overview

A fast, and general engine to deploy Spark.
- Cluster portability (MN4, MT, Nord3).
- Batch/Queue manager portability (LSF, Slurm).
Spark4HPC: Overview

A fast, and general engine to deploy Spark.
- Cluster portability (MN4, MT, Nord3).
- Batch/Queue manager portability (LSF, Slurm).

And…
Spark4HPC: Overview

- A fast, and general engine to deploy Spark.
  - Cluster portability (MN4, MT, Nord3).
  - Batch/Queue manager portability (LSF, Slurm).

- Engine for systematically test Spark applications.
Spark4HPC: Overview

A fast, and general engine to deploy Spark.
- Cluster portability (MN4, MT, Nord3).
- Batch/Queue manager portability (LSF, Slurm).

Engine for systematically test Spark applications.

Tool to test application, and hardware scalability.
Spark4HPC: Overview

- A fast, and general engine to deploy Spark.
  - Cluster portability (MN4, MT, Nord3).
  - Batch/Queue manager portability (LSF, Slurm).

- Engine for systematically test Spark applications.

- Tool to test application, and hardware scalability.

- Tool to find the best hardware config for a production environment.
Spark4HPC: Components

```
bsc99776@login3:--> spark4mn
Usage: spark4mn [-h | --help] | [[-s | --silent] [CONFIG_FILE] ]

This command launches a Spark cluster, then executes as many prologues as needed, and executes as many domains as wanted, and finally as many epilogues as requested. Everything as an all-in-one job.

Parameters:
- `-h`, `--help` (optional): Prints this message.
- `-s`, `--silent` (optional): Do not log.

CONFIG_FILE: Path to a valid configuration file. See /apps/SPARK4MN/1.4.1/doc/spark_job.rc.template to create your own file.

Configuration file is mandatory, unless the help parameter is present.
```
spark4hpc_benchmark

```
bsc99776@login3:~> spark4mn_benchmark
Usage: spark4mn_benchmark [-h | --help] | [[-s | --silent] [-r | --restart] [CONFIG_FILE] ]

This command launches a set of spark4mn jobs.

Parameters:
-h, --help (optional): Prints this message.

-s, --silent (optional): Do not log.

-r, --restart (optional): Executes one job for each geometry that has less
*.metrics files than geometry that has the maximum number of these
files. Often used to resend jobs that went wrong.

CONFIG_FILE: Path to a valid configuration file. See
/apps/SPARK4MN/1.4.1/doc/spark_benchmark.rc.template to create
your own file.

Configuration file is mandatory, unless the help parameter is present.
```
Spark4HPC: Components

```bash
spark4hpc_plot
```

Usage: `spark4hpc_plot [-h | --help] | [-c | --check] [-C | --check-only] [-g | --gaps] [-H | --homogeneous] [CONFIG_FILE]`

This command parses metrics of finished Spark jobs, if they were created with `spark4mn_benchmark`.

Parameters:
- `-h`, `--help` (optional): Prints this message.
- `-c`, `--check` (optional): Check values to see if they biases too much.
- `-C`, `--check-only` (optional): Only check values, do not output plot files.
- `-g`, `--gaps` (optional): Allow plotting skipping those geometries without any metrics files. This will generate gaps in the plot. This option is incompatible with `-H` flag.
- `-H`, `--homogeneous` (optional): Do not plot if there is not the same amount of metrics files for each geometry. This option is incompatible with `-g` flag.

CONFIG_FILE: Path to a valid `spark4mn_benchmark` configuration file. See `/apps/SPARK4MN/1.4.1/doc/spark_benchmark.rc.template` to create your own file.

Configuration file is mandatory, unless the help parameter is present.
Spark4HPC: How it works?

spark4hpc sets up the cluster, and launches applications, everything as one job.
Spark4HPC: How it works?

spark4hpc sets up the cluster, and launches applications, everything as one job.

spark4hpc_benchmark launches N jobs.
Spark4HPC: How it works?

- spark4hpc sets up the cluster, and launches applications, everything as one job.

- spark4hpc_benchmark launches N jobs.

- Each job, if ended successfully, has a metrics file.
Spark4HPC: How it works?

- **spark4hpc** sets up the cluster, and launches applications, everything as one job.

- **spark4hpc_benchmark** launches N jobs.

- Each job, if ended successfully, has a metrics file.

- So…
Spark4HPC: How it works?

- spark4hpc sets up the cluster, and launches applications, everything as one job.

- spark4hpc_benchmark launches N jobs.

- Each job, if ended successfully, has a metrics file.

- spark4hpc_plot gets all benchmark metrics, and extracts plots.
Spark4HPC: How it works?

spark4hpc config_file
Spark4HPC: How it works?

- `spark4hpc config_file`
- `spark4hpc_benchmark bench_config_file`
Spark4HPC: How it works?

- spark4hpc config_file
- spark4hpc_benchmark bench_config_file
- spark4hpc_plot bench_config_file
Thank you!
For further information, please, contact support@bsc.es