Barcelona Supercomputing Center
Centro Nacional de Supercomputación

EXCELENCIA SEVERO OCHOA

# Programming Distributed Computing Platforms with COMPSs

Rosa M. Badia, Javier Conejero, Cristian Tatu

Workflows & Distributed Computing Group

06/02/2024

Winter School, Barcelona

# Outline

## Agenda

- Presentation of the tutorial presenters
- Introduction to COMPSs (20 min)
- PyCOMPSs: Writing Python applications (1hour)
- Break (15 min)
- Hands-on MN (1 hour)
- PyCOMPSs installation (15 min)
- SLIDES
  - http://compss.bsc.es/releases/tutorials/tutorial-WINTER_SCHOOL_2024/


# INTRODUCTION

# Motivation

- New complex architectures constantly emerging
  - With their own way of programming them
    - Fine grain: e.g. Programming models and APIs to run with GPUs, NVMs (Non-Volatile Memories)
    - Coarse grain: e.g. APIs to deploy in Clouds
  - **Difficult** for programmers
    - Higher learning curve / Time To Market (TTM)
    - What about non computer scientists???
  - **Difficult** to understand what is going on during execution
    - Was it fast? Could it be even faster? Am I paying more than I should? (**Efficiency**)
  - Tune your application for each architecture (or cluster)
    - E.g. partitioning data among nodes

# Motivation

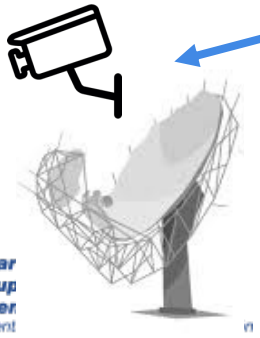
- Resources that appear and disappear
  - How to dynamically add/remove nodes to the infrastructure
- Heterogeneity
  - Different HW characteristics (performance, memory, etc)
  - Different architectures -> compilation issues
- Network
  - Different types of networks
  - Instability
- Trust and Security
- Power constraints from the devices in the edge
- Data & Storage

AI everywhere

HPC
Exascale computing
Cloud

Fog devices


Edge devices



Sensors
Instruments
Actuators

# Motivation

- Create tools that make developers' life **easier**
  - Allow developers to focus on their problem
  - Intermediate layer: let the difficult parts to those tools
    - Act on behalf of the user
    - Distribute the work through resources
    - Deal with architecture specifics
    - Automatically improve performance
  - Tools for visualization
    - Monitoring
    - Performance analysis
  - Integration of computational workloads, with machine learning and data analytics

# BSC vision on programming models

Applications

PM: High-level, clean, abstract interface

Power to the runtime

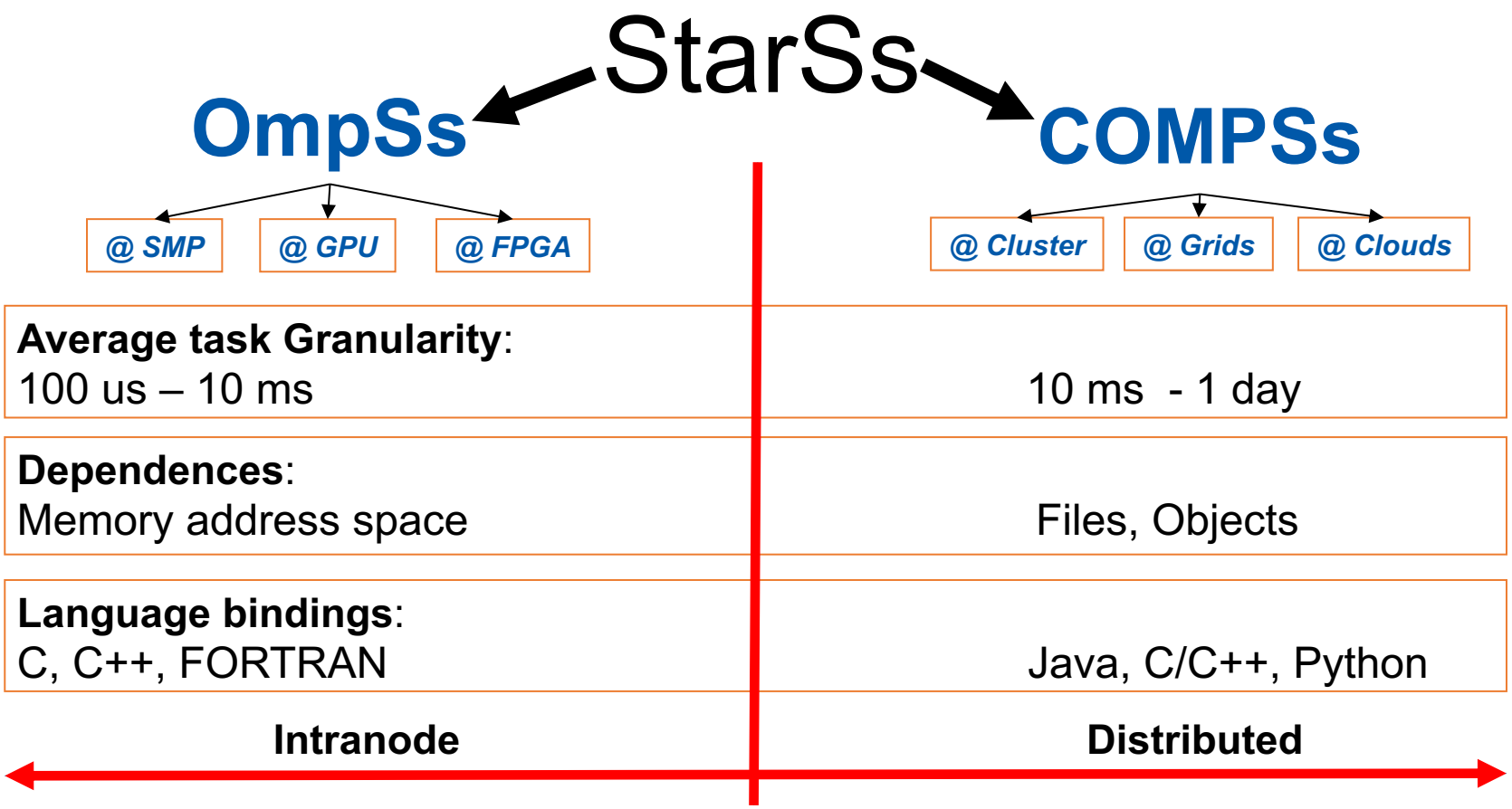
API

Program logic independent of computing platform

General purpose
Task based
Single address space

Intelligent runtime, parallelization, distribution, interoperability

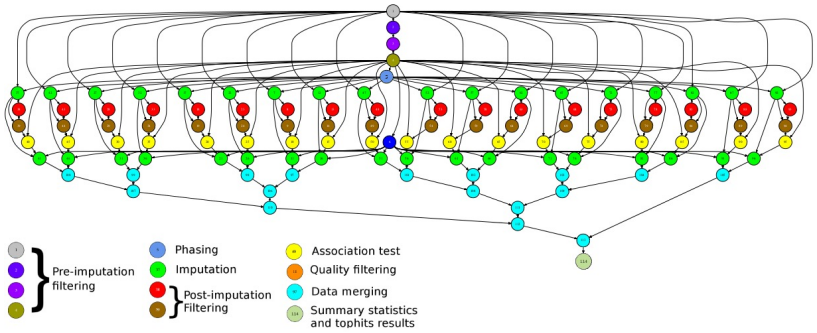
# BSC vision on programming models



# Main element: Workflows in PyCOMPSs

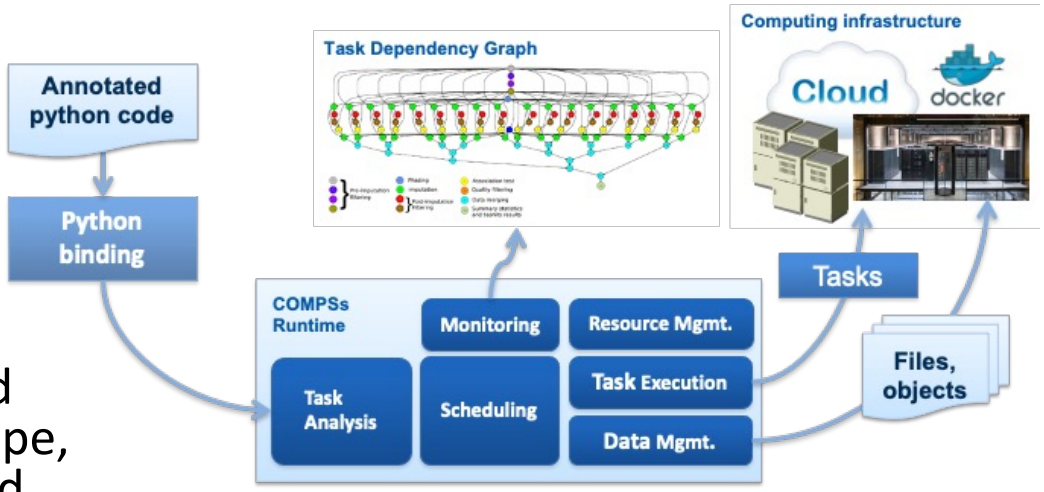
- Sequential programming, parallel execution
- General purpose programming language + annotations/hints
  - To identify tasks and directionality of data
- Builds a task graph at runtime that express potential concurrency
- Tasks can be sequential and parallel (threaded or MPI)
- Offers to applications the illusion of a shared memory in a distributed system
  - The application can address larger data than storage space: support for Big Data apps
- Agnostic of computing platform
  - Enabled by the runtime for clusters, clouds and container managed clusters

```
@task(c=INOUT)
def multiply(a, b, c):
    c += a*b
```



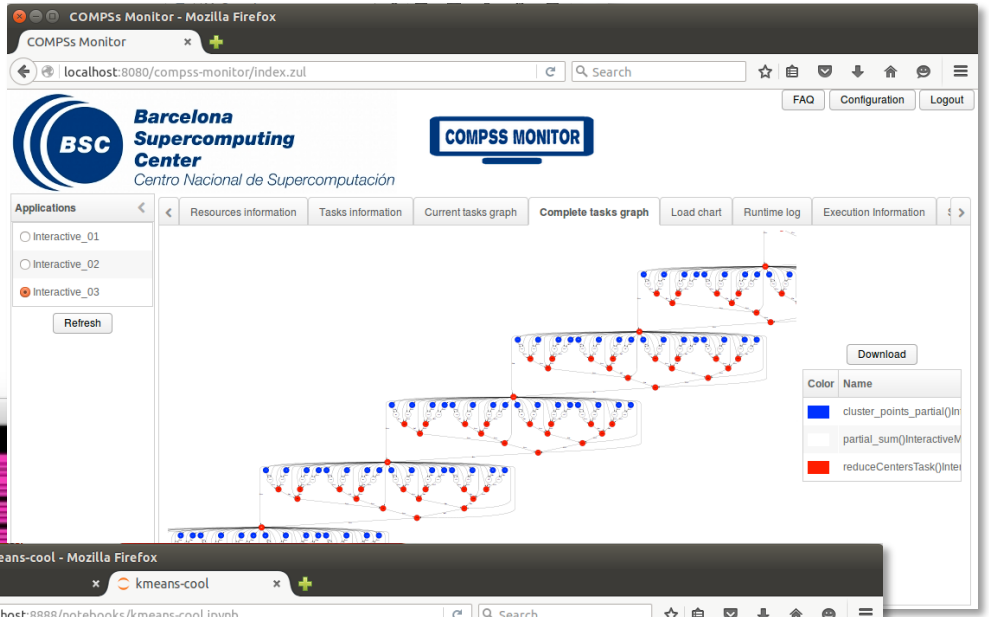
# PyCOMPSs features and runtime

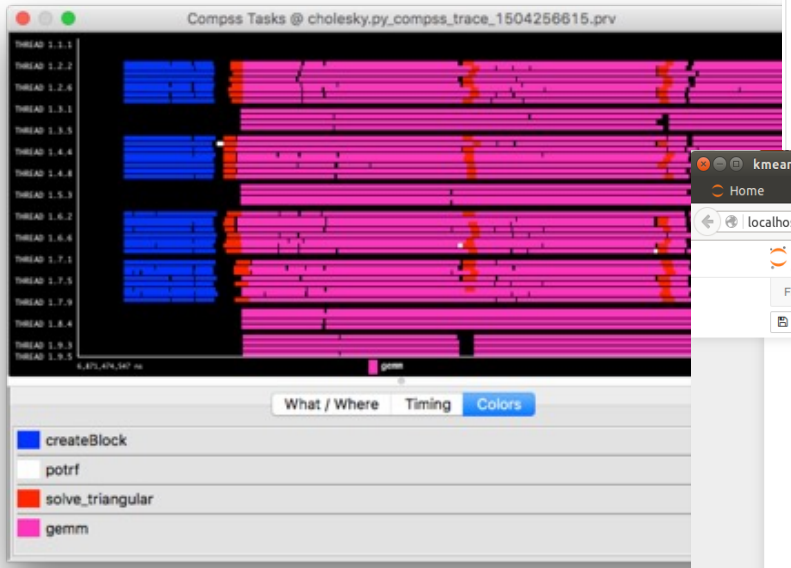
- Support for tasks' constraints – support for heterogeneous infrastructure
- Support for tasks' faults and tasks' exceptions
  - Enlarges the dynamicity of the type of workflows that we support
- Streamed data
  - ... and many others
- Runtime deployed as a distributed master-worker
- All data scheduling decisions and data transfers are performed by the runtime
- Support for elasticity
- Available in MareNostrum and other supercomputers in Europe, in the EGI Federated Cloud and in Chameleon Cloud

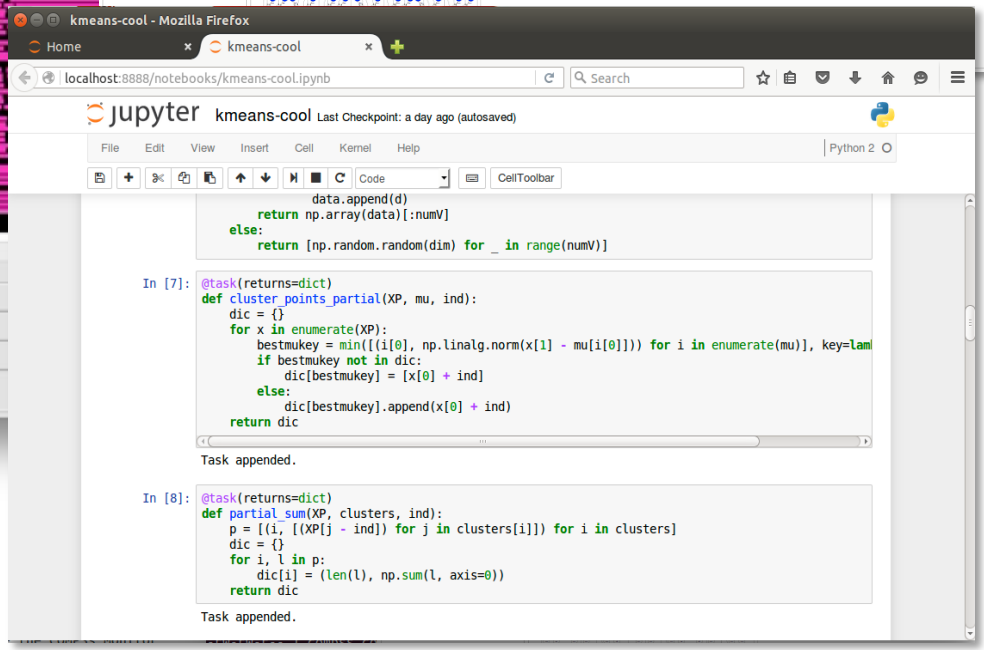


# PyCOMPSs development environment

- Runtime monitor
- Paraver traces
- Jupyter-notebooks integration

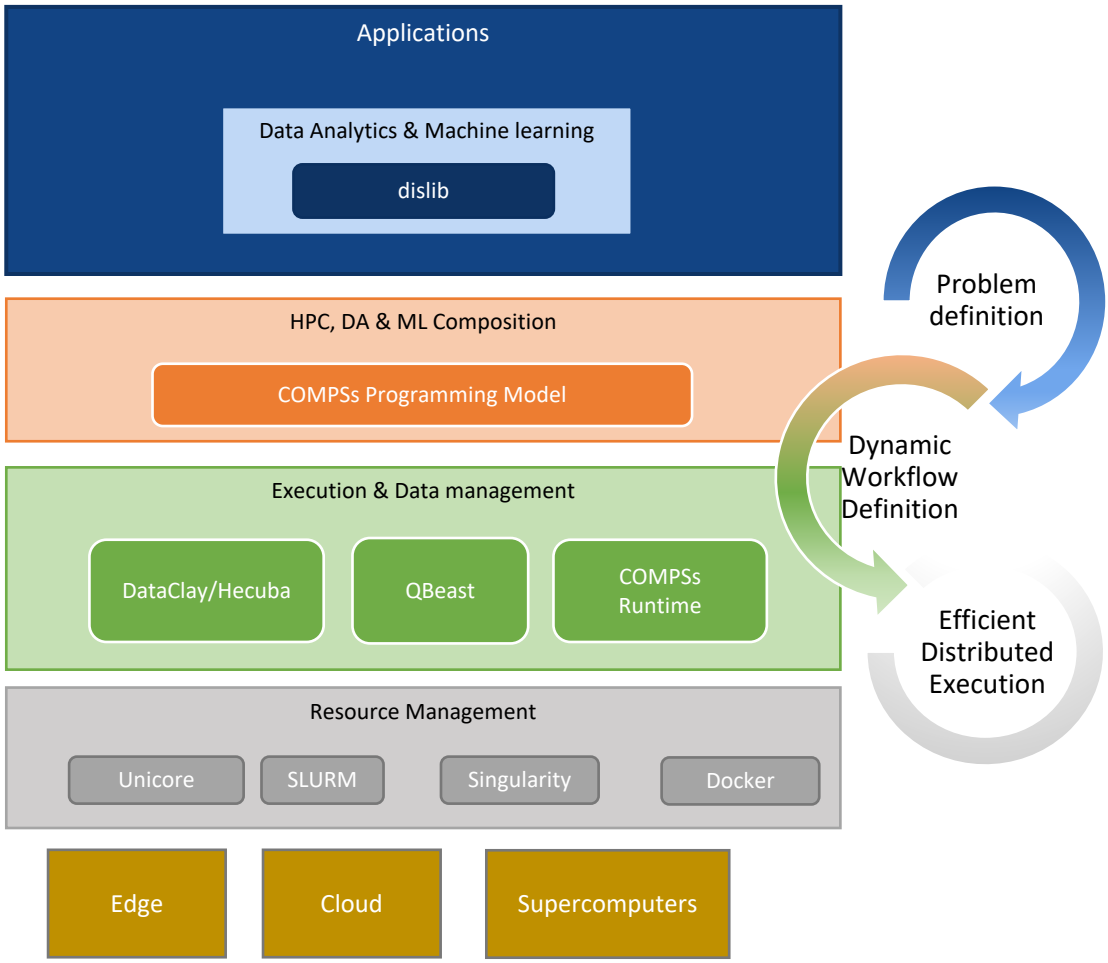


# Conclusions

- COMPSs provides a workflow environment that enables the integration of HPC simulation and modelling with big data analytics and machine learning
- Support for dynamic workflows that can change their behaviour during the execution
- Support for dynamic resource management depending on the actual workload needs
- Support for data-streaming enabling the combination of task-flow and data-flow in the same workflow
- Support for persistent storage beyond traditional file systems.



# Projects where COMPSs is used/developed

DT-GEO

CAELESTIS

ICOS

CEEC

COMPSs

CyclOps

HP2C-DT

eFlows4HPC

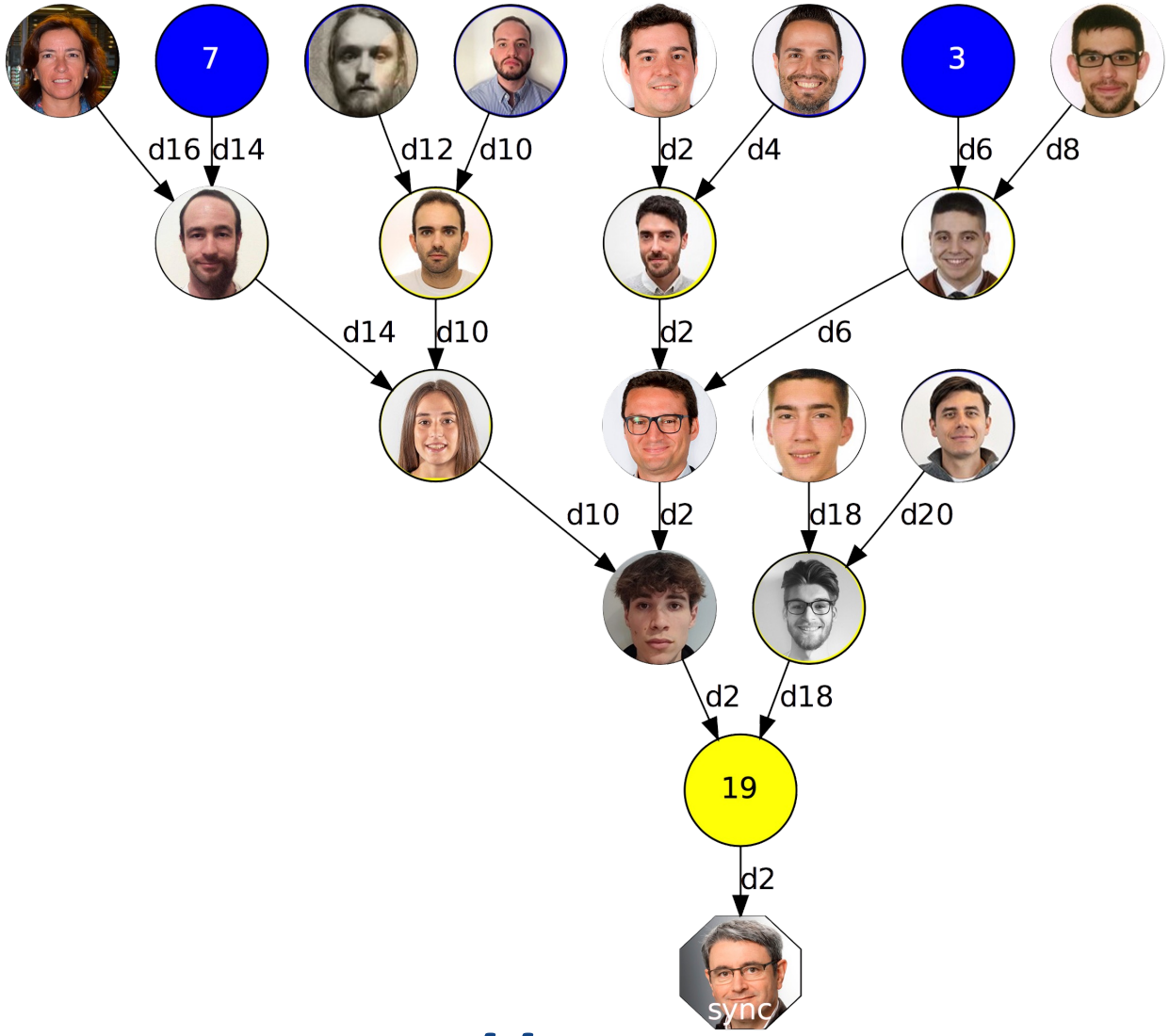
COLMENA

PERTE chip

Joint Laboratory for Extreme-Scale Computing

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

# The WDC team




http://compss.bsc.es

Barcelona Supercomputing Center
Centro Nacional de Supercomputación