COPA - GT Barcelona Meeting

**Date:** Monday, January 7th, 2013

**Place:** In BSC premises, Campus Nord, UPC. The exact place is to be determined.

**Meeting Agenda:** To be determined.

---

COPA - GT Barcelona Training

**Date:** Tuesday, January 8th, 2013 to Friday, 11th, 2013

**Place:** In BSC premises, Campus Nord, UPC. The exact place is to be determined.

**Specific requirements:** It is **HIGHLY RECOMMENDED** that each fellow attending the course brings his/her own laptop.

<table>
<thead>
<tr>
<th>DAY</th>
<th>MORNING</th>
<th>AFTERNOON</th>
</tr>
</thead>
<tbody>
<tr>
<td>MONDAY 7TH</td>
<td>MEETING</td>
<td>MEETING</td>
</tr>
<tr>
<td>TUESDAY 8TH</td>
<td>TRAINING PARAVIEW</td>
<td>TRAINING PARAVIEW</td>
</tr>
<tr>
<td>WEDNESDAY 9TH</td>
<td>TRAINING PARAVIEW</td>
<td>TO BE DETERMINED</td>
</tr>
<tr>
<td>THURSDAY 10TH</td>
<td>TRAINING HPC</td>
<td>TRAINING HPC</td>
</tr>
<tr>
<td>FRIDAY 11TH</td>
<td>TRAINING CARPENTRY</td>
<td>TRAINING CARPENTRY</td>
</tr>
</tbody>
</table>
COPA - GT Barcelona Training

Courses Program

ParaView: An Application and Framework for Large-Scale Data Visualization and Analysis

Tuesday morning:
- Introduction to VTK
  - VTK: data structures, algorithms, visualization pipeline

Tuesday afternoon:
- Introduction to ParaView
  - Usage of the ParaView Qt client: opening, filtering and displaying data
  - Data analysis with ParaView

Wednesday morning:
- Parallel ParaView: settings, usage and optimization
- Extending ParaView: custom filters, programmable filters, scripting, plugins

Teacher:

Philippe Pébay, Kitware

Dr. Pébay is one of the core developers of VTK and ParaView. He has over 15 years of experience in the field of applied mathematics, in particular in the context of scientific computing and visualization. Dr Pébay is currently the director of Visualization and High Performance Computing at Kitware SAS, and has taught numerous VTK and ParaView courses at various research and development organizations.

Introduction to HPC computing: a hands-on approach

This course is oriented to non-computer scientists that need to develop code in their everyday work. Today is not uncommon to find a multi-core CPU in our desktops, or even a workstation with several GPUs or Xeon Phi cards... but the question is: how can we efficiently program such platforms? This course is an introduction to this HPC equipment from an architectural point of view, with a practice section devoted to show how to program them to extract the most of their computational power.

Requirements: As this is an introductory course, no previous programming experience in HPC environments is assumed. Nevertheless, it is highly recommended a basic understanding of how a general purpose processor work, as well as some programming experience in C/Fortran.

Thursday morning:
HPC architectures

- A core, our fundamental block.
- Multi-core processors
- Graphic Processing Units (NVIDIA GPUs)
- Co-Processors (Intel Xeon Phi)

Thursday afternoon:

Programming

- Pthreads
- OpenMP / MPI / OpenAcc
- StarSs
- NVIDIA Cuda / OpenCL
- Intel LEO

Teacher:

Félix Rubio Dalmau, BSC-CNS

Felix Rubio received his MSc degree in Computer Architecture, Networks and Systems in 2010, and is now a PhD candidate at Universitat Politecnica de Catalunya. Currently he is researcher in Barcelona Supercomputing Center, the Spanish National Supercomputing facility, in which he is developing HPC tools for the Oil & Gas industry focusing in NVIDIA GPU and Intel MIC code optimization accelerators.

Software carpentry

This course aims at teaching fundamental computational skills and concepts to scientists and engineers who use computers as a research tool. In particular, the focus is on techniques and topics that are commonly used in the professional software development industry. Such techniques are designed to help productivity and improve the quality of software. Although they are part of the curricula of computer related careers, they are rarely taught to engineers, physicists, and mathematicians, even when some end up developing scientific software throughout their whole careers. The course includes a selection of general topics of software development, some hands-on experience on productivity tools, and discussion and examples aiming at developing good programming habits.

Requirements: The target audience is broad, from PhD students to advanced researchers. Some programming experience in any language is desirable.

Friday morning:

Basic computer concepts
Programming paradigms: Functional, imperative, and object oriented.

Data structures: beyond arrays (hash tables, linked lists)

Programming Tools

Shell programming, regular expressions, editors, IDEs.
Version control
Build systems

Friday afternoon:
Writing Quality Software

Program and project design
Writing clean code
Testing
Debugging
Refactoring

Teacher:

Fernando Cucchietti, BSC-CNS

Fernando Cucchietti is a researcher at the Barcelona Supercomputing Center, where he develops code for computational fluid dynamics applied to combustion and radiation, and leads the Scientific Visualization team. He received his Ph.D. in Physics in 2004 from the University of Córdoba, Argentina. He has worked at Los Alamos National Laboratory (USA) and The Institute of Photonic Sciences (Spain), researching on quantum computing and developing simulations of quantum mechanical time evolution.