Best Practices for Performance and Programmability
Our team aims to be the bridge between scientific domain researchers and computer science researchers. Promoting best practices for programmers to productively (re)structure their codes in ways that can result in high performance. To achieve this, we try to answer the following questions:

- Which common patterns can we find in scientific applications, and how can we improve their performance?
- Which common performance issues do scientific applications face, and what can we provide to avoid them?

**Philosophy or Vision**

- **Portability**: Applications must rely on parallel programming models and runtime libraries to be as agnostic as possible of underlying architectures.
- **Programmability**: A must, scientific applications are continually being developed for their field, and the performance optimization must not interfere with that development.
- **Malleability and Adaptivity**: Relying on runtime and dynamic mechanisms will allow applications to survive architecture and system software noise and variability.
- **Performance analysis**: Must be the first step in an optimization process.

**Objectives**

- Discover, promote and boost cooperation and collaboration between computer science research groups and other BSC departments.
- Promote the use of BSC tools and methodologies (i.e., performance tools, programming models, runtime libraries) by HPC applications, either from external users or from BSC.
- Provide hints on how programming models (or other system software) can be improved to address real applications issues/needs.
- Provide hints, through analysis or proof-of-concept studies, to applications on how their performance can be improved using state of the art system software features.
- Aggregate the information on scientific applications to ease the process of working with them to the different research groups and avoid duplicated efforts.