System Tools and Advanced Runtimes (STAR)
The System Tools and Advanced Runtimes (STAR) group focus on research crossing multiple software layers, from OS, runtimes and low-level APIs to programming models, ... Analytics on massively parallel HPC and Cloud platforms. We are involved in several European and industrial projects.

Objectives

Data-flow programming models
We develop and maintain the OmpSs-2 programming model, a versatile data-flow programming model to build HPC applications. OmpSs-2 main targets are multi-core, heterogeneous and distributed systems. The OmpSs-2 programming model is implemented by the LLVM compiler and the Nanos6 runtime system.

Compilers and Domain-Specific Languages (DSLs)
The OmpSs-2 compiler for C and C++ languages leverages the LLVM infrastructure to generate high-performance code. We also demonstrate the feasibility of DSLs on HPC systems and it leverages many of the OmpSs-2 features to run on HPC systems.

OS and Runtimes Systems
Runtime systems are a crucial component to exploit current multi-core and heterogeneous systems. Our research on runtime systems includes the development of the Nanos6 runtime system and the LLVM OpenMP runtime system. We also work with the Linux kernel to extend it to better support advanced runtime systems. Finally, we also have augmented the LLVM OpenMP runtime with low-level APIs required to support the Task-Aware (TA) libraries described below.

Interoperability with other programming models and APIs
Our Nanos6 and LLVM OpenMP runtime systems provide several features to integrate other programming models and low-level APIs. We have developed several Task-Aware (TA) libraries that ease the orchestration of complex hybrid and heterogeneous applications. The TAMPI and TAGASPI libraries support MPI and GASPI message-passing APIs, respectively, while TA-CUDA, TA-OpenCL and TACL leverage CUDA, OpenCL and other frameworks to manage all data transfers between the host and accelerators. This research line is conducted closely with the Programming Models and AccelCom groups.

Benchmarks, libraries and tools
The Garlic application suite comprises several mini-apps and benchmarks that represent common computational patterns. These mini-apps include various computational kernels and benchmarks that can be used to obtain detailed execution traces, facilitating the analysis and optimization of applications and programming models.