MPPA ACCESSCORE is a software development environment providing tools to quickly develop, debug, and optimize, high-performance applications for the MPPA MANYCORE processors.

**MPPA ACCESSCORE overview**

MPPA ACCESSCORE is a complete tool suite that eases the development of applications for the Kalray manycore processor in a very short time. MPPA ACCESSCORE includes:

- **MPPA ACCESSCORE Dataflow and POSIX Programming models**
  MPPA ACCESSCORE Parallel Programming provides two programming models: dataflow and POSIX. They enable to easily expose the parallelism of applications and fully exploit the MPPA MANYCORE processors.

- **MPPA ACCESSCORE Acceleration Framework**
  MPPA ACCESSCORE Acceleration Framework enables to quickly deploy the MPPA MANYCORE processors on a card integrated into a PC or a server.

- **MPPA ACCESSCORE Integrated Environment**
  MPPA ACCESSCORE IDE integrates the compilers, debuggers, and system trace analyzers under a unified environment based on the standard Eclipse framework.

A MPPA® application is composed of program components that execute on different parts of the MPPA MANYCORE processor: the I/O subsystems, fitted with a Kalray quad-core; and the clusters, where 16 user cores and one system core share a multi-banked memory.

**MPPA ACCESSCORE Dataflow programming**

MPPA ACCESSCORE supports dataflow programming based on an extension of standard C, which provides extra keywords for exposing the parallelism of applications. The user simply has to express applications as graphs of “agents” connected through conceptually unbounded FIFOs.

The innovative Kalray dataflow compiler fully automates the mapping of applications to the MPPA® clusters, abstracting completely the parallel execution.

The compiler performs the following steps:

- scheduling the dataflow graph into run-time tasks
- sizing the buffers that implement the FIFOs
- distributing the task execution over the MPPA® clusters
- filling the dataflow run-time tables for synchronizing the task executions and communications

The dataflow compiler can be used in iterative mode, and leverage user hints across compilations.

**MPPA ACCESSCORE POSIX programming**

The basic component of MPPA® applications are POSIX processes, with a “main” process running on an I/O subsystem, and child processes running on each of the clusters.
The “main” process is hosted by a Linux kernel running on the I/O subsystem, and is able to operate the I/O devices of the MPPA MANYCORE processor. For hard real-time users, the Linux kernel is replaced by RTEMS (Real-Time Executive for Multiprocessing Systems).

The MPPA® applications components involve one process per MPPA® cluster, which are themselves connected by special communication and synchronization objects that present a classic POSIX file descriptor inter-process communication (IPC) interface. These communication and synchronization objects are directly supported by the architecture of the MPPA® NoC.

Whether generated by the dataflow compiler or manually authored, the MPPA® cluster processes are hosted by a simple operating system called ‘NodeOS’. Inside a MPPA® cluster, this system provides the POSIX thread interface for the exploitation of parallelism between cores. The OpenMP support of the GNU C/C++ compilers is also available as an alternative to explicit POSIX threads.

**MPPA ACCESSCORE Acceleration framework**

MPPA® applications can either run entirely on the MPPA® processor (autonomous system configuration), or be split between a host application and an acceleration application (PCI express acceleration card configuration). The latter use case is supported by the acceleration framework provided with MPPA ACCESSCORE.

On the host, the user relies on the acceleration framework top layer to dispatch a MPPA® application, to allocate DDR memory on the acceleration card, and to manage the work queues.

At the bottom layer of the acceleration framework, the dataflow or POSIX application executes like in the autonomous system configuration, with a main process running on an MPPA® I/O subsystem, and child processes running on each of the MPPA® clusters.

**MPPA ACCESSCORE IDE**

MPPA ACCESSCORE IDE integrates the dataflow language tools and the standard GNU software tools under a unified interface provided by the standard Eclipse® framework.

The dataflow language tools include:
- a dataflow language workstation simulator,
- a parallel execution profiler,
- an interactive dataflow graph mapping tool,
- the Kalray dataflow compiler.

The standard GNU software tools include:
- the GCC and G++ compilers,
- the GNU binary utilities,
- the Newlib standard libraries,
- the GDB debugger with multi-threading support.

These tools are complemented by Kalray performance and power monitors, and a cycle-accurate platform simulator.

This integrated environment enables to map applications onto MPPA MANYCORE processor, whether described in dataflow, in POSIX, or even in a mix of the two programming models.

For application debugging and optimization, Kalray proposes a unique observability system.

MPPA ACCESSCORE IDE includes a state-of-the-art many-core debugger and system level trace viewer.
- fine grained visibility and control of all the cores,
- insight on the overall system behavior,
- observation of live system,
- hardware trace system with low-overhead code instrumentation.