Software-Hardware Interface for Multi-many-core

SHIM

Introduction

Masaki Gondo
Chair of SHIM Working Group
Software CTO, eSOL
The Multicore Association

• Established in 2005

• Mission: Improve time to market through the use of industry standards

• Membership: board, working group, university

• Committee-based standards development
Multicore Communications API (MCAPI) 2.0
- Semantic for communication and synchronization between processing cores in embedded systems.
- Growing number of implementations: www.multicore-association.org/products/index.php
- Low-level layer for higher-level programming models such as OpenMP

Multicore Resource Management API (MRAPI)
- Capabilities required by multicore applications to allow coordinated concurrent access to system resources (i.e. memory, mutexes)

Multicore Task Management API (MTAPI)
- Leveraging task parallelism on embedded devices (homogeneous or heterogeneous multicore processors).
- Dynamic scheduling and task mapping to processor cores

Multicore Programming Practices Guide (MPP)
- 120+ pages dedicated to various multicore programming techniques
Strategic Roadmap of the Multicore Association
What is a SHIM?

Shim (spacer)
From Wikipedia, the free encyclopedia

A shim is a thin and often tapered or wedged piece of material, used to fill small gaps or spaces between objects. Shims are typically used in order to support, adjust for better fit, or provide a level surface. Shims may also be used as spacers to fill gaps between parts subject to wear.

- Multicore tools, as well as OS/middleware, help applications run optimally on multicore chips while hiding the hardware specifics.
- However, the tools must understand the hardware specifics.
- SHIM provides tools with the specific multicore hardware description in a standardized, open XML model.
Problem-Solution-Benefit

• Problem
  – Multicore and manycore architectures are diverse
  – Architectural information is critical for tools and runtime systems to be able to parallelize, optimize, analyze, and manage the multi-manycore software system
  – Currently architectural information is available in proprietary format, primarily in natural language (i.e. documentation)
  – Costly for tool support, reduced tool availability, and subject to misinterpretation

• Solution
  – Define a standard for multicore architecture description
  – Hardware vendors provide description (i.e. in XML format) -> tool vendors consume

• Benefit
  – Chip vendors, OEMs, and programmers have more tools available
  – Customers get quicker time-to-market
  – Provide a more consistent interface and documentation between hardware and software vendors
SHIM is

• An interface defined as an XML schema
  – XML hardware description is written or generated according to the schema
• An extraction of hardware properties that matter to multicore tools
  – Processor core, number of cores, synchronization mechanism, inter-core communication channels, memory system, NoC/interconnect, virtualization
• A HW model described from a SW point of view

• NOT a functional model of hardware – it is descriptive
• NOT a 100% description of hardware – only the properties that matter to software
• NOT a tool itself – tools are implemented by various vendors that use SHIM
SHIM can

- Help tools roughly estimate SW performance
- Help tools configure themselves and/or auto-generate the HW-specific configuration UI
- Help configure device drivers or hardware abstraction layer (HAL)

- NOT estimate SW performance with 100% accuracy
- NOT auto-generate HAL
Use Cases

• Performance estimation
  – Performance information is critical for most design-aid tools
  – Examples are auto-parallelizing compilers, other parallelizing tools, performance analysis tools, etc.

• System configuration
  – OS, middleware, and other runtime libraries need basic architectural information to configure itself
  – Other tools previously mentioned also need this

• Hardware modeling
  – May serve to configure a HW model (i.e. simulator)
  – May be useful for architecture exploration
Thank You!

Join the SHIM Working Group and contribute to this international standard

www.multicore-association.org