Multicore Association Overview

EDPS, April 2006

Sven Brehmer & Kumar Venkatramani

THE Multicore ASSOCIATION

THE Multicore ASSOCIATION

Multicore Association – Objectives & activities

- Multi-core ecosystem enablement
- Industry-wide participation
  - Including processor vendors, infrastructure developers, device manufacturers, and software and application developers.
- Current efforts
  - Communications & Resource management API's
  - Debug API
  - Transparent Inter Process Communication (TIPC)

Multicore ASSOCIATION

So why are standard API's important?

- Standard interfaces simplify development and management of multi-core software and reduces time to market
- They can broaden the availability of products
- They enable software re-use
  - From ESL modeling to real target hardware
  - From product to product
- They enable more vendor choice

Multicore ASSOCIATION

RAPI/CAPI – Objectives, goals & target domain

- To provide a standardized APIs for:
  - Management, scheduling, and synchronization of work entities among cores or within cores - RAPI
  - The basic elements of communication and synchronization between cores or among closely distributed processors – CAPI
- Provide API for source code portability
  - The primary goal of this effort is to specify an API, not an implementation
- RAPI/CAPI Target domain
  - Embedded Multi-processing
  - Task to task communication
  - Multiple cores on a chip & multiple chips on a board
  - Heterogeneous & homogeneous systems
    - Cores, interconnects, memory architectures, OSes
  - Optimized for data plane
  - Scalable: 2 – 1000's of cores
  - Allow implementations with significantly lower latency, overhead & memory footprint than MPI and TIPC

Multicore ASSOCIATION

CAPI/RAPI - Applications and Use cases

- We are looking for real use cases!!
  - Currently proposed
    - Signal processing
    - Multi-media processing
    - Wireless base station
    - Network processing
    - Distributed Object Model
    - Simulation

Multicore ASSOCIATION

Relationship to Other Standards & Technologies

Overlap with and potential usefulness (or not) for the CAPI

- OSI
- TCP/IP (or sockets)
- UDP/IP
- TIPC
- RIO
- Sun RPC
- CORBA
- TTL
- OpenMP
- MPI
- MPIRT
- Embedded MPI
- I2O
- OSEK Com
- Linx
- Others?

Similar synergistic or overlapping efforts?

- RapidIO
- MIPI
- Khronos
- CVTA
- OpenMax
- Others?

Multicore ASSOCIATION

What is CAPI?

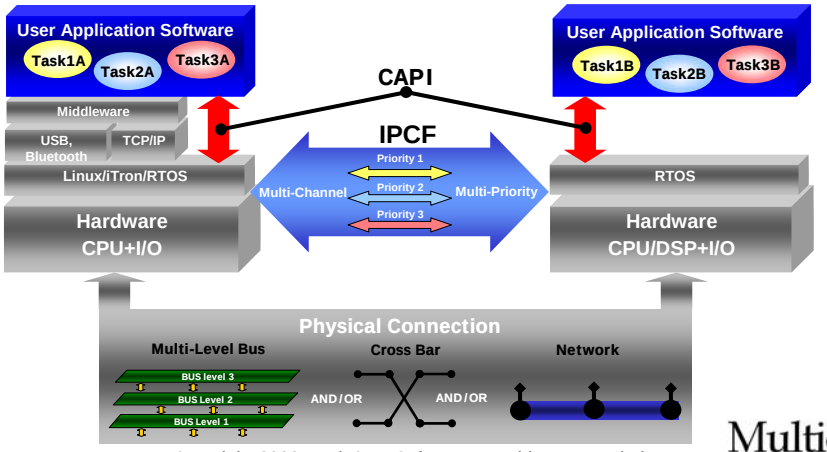
- An API to support synchronization and data movement between cores in a closely distributed embedded system
- It is agnostic to the type of:
  - Core
  - Memory architecture
  - Physical connection
  - Logical topology
  - OS

Multicore ASSOCIATION

Desired: Transparent multi-processing

- **Different / Multiple**
  - **Operating Systems**
  - **Connections - Physical & Logical**

One Messaging API



Multicore ASSOCIATION

What is resource management & RAPI?

- A RAPI entity manages...
  - *binding* of application elements to processing resources
    - Static and dynamic decisions
  - memory
    - Allocation, de-allocation
    - Static and dynamic support
- Programming model
  - Explicit task based parallelism
  - *No compiler directives*
- Control plane only
  - *No data plane*
- Memory allocation and de-allocation
  - Private and shared memory
  - NUMA (disparate memory domain) support
- Task
  - state management
    - Creation/deletion
    - Suspension/resumption
    - Basic synchronisation
  - scheduling
    - Processing resource class based, enabling...
      - Dynamic load balancing
      - Dynamic power management
    - Enabling static and dynamic logical reconfiguration
  - Context management
    - Save/restore

Multicore ASSOCIATION

RAPI Target Architectures

- Heterogeneous processing resources
  - ISA architectures
  - Non-ISA based architectures
  - I/O devices
- Heterogeneous memory architectures
  - Shared, unified, non-uniform architectures
- Heterogeneous interconnect topologies

Multicore ASSOCIATION

RAPI Goals

- "To provide a standardised API for the management, scheduling and synchronisation of resources including, but not limited to, tasks, processing and memory resources."
- Secondary goals...
  - Complimentary to CAPI
  - Support existing operating systems
  - Support existing and emerging middleware technologies
  - Support language based multicore abstractions

Multicore ASSOCIATION

CAPI and RAPI relationship

- The RAPI and CAPI combine to form a layer on top of which other abstractions or applications may be built

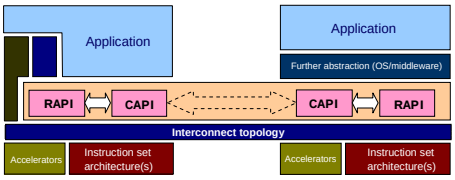


Figure : Example system embodiment showing two resource domains

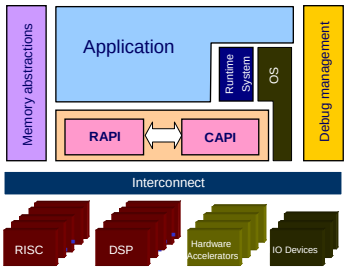
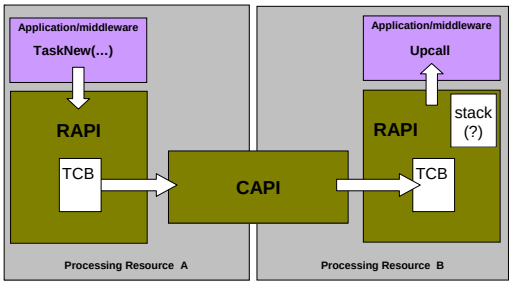


Figure : Example system embodiment showing a single resource domain

Multicore ASSOCIATION

RAPI and CAPI Interaction



- RAPI describes tasks using task control blocks (TCB)
- CAPI transports task control blocks and optional data within and between resource management domains
- RAPI schedules tasks to processing resources
- RAPI manages task execution contexts

Multicore ASSOCIATION

Scheduling with POSIX pThreads

```
int pthread_create(
  pthread_t            *tid,
  const pthread_attr_t *attr,
  void                 *(*start)(void *),
  void                 *arg);
```

```
struct pthread_attr_t {
  void *stackaddr;
  size_t stacksize;
  int detachstate;
  int schedpolicy;
  struct sched_param param;
  int inheritsched;
  int contentionscope;
};
```

- Policy oriented definition
- *Implicit* scheduler structure definition
  - Implies processing resource homogeneity

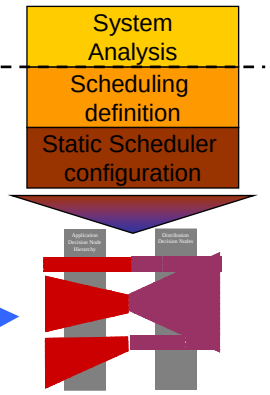
FIFO: `SCHED_FIFO`
Round robin: `SCHED_RR`
???: `SCHED_OTHER`

Multicore ASSOCIATION

Scheduling with RAPI tasks

```
RAPIReturn_t RAPITaskNew(
  RAPITCB_t             *pTask,
  BOOLEAN               IsBlocked);
```

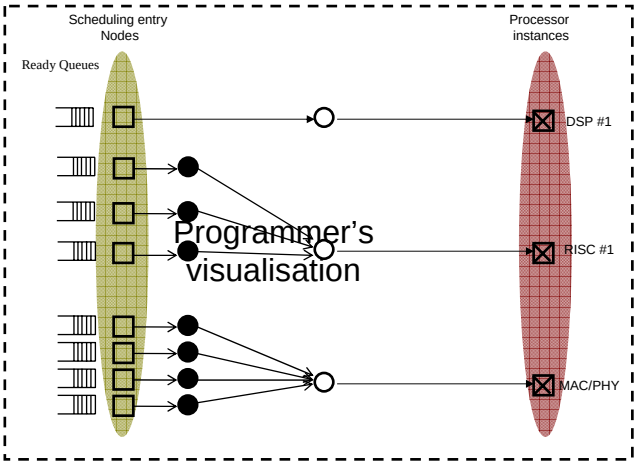
```
struct RAPITCB
{
    U32 (*fpTask)(void *);
    U32 *pParams;
    U32 *Metric;
    U16 SchedulingNode;
    // Implementation
    // specific params
    ...
};
```



- Hierarchy oriented definition.
- *Explicit* scheduler structure

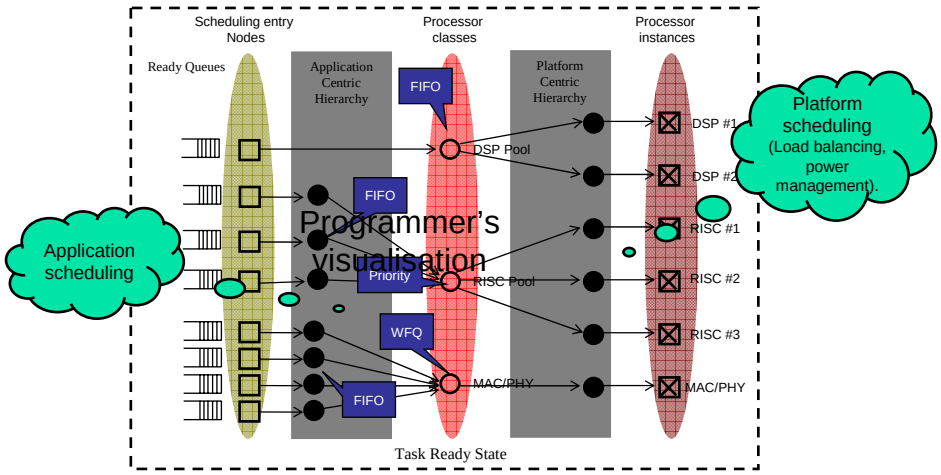
Multicore ASSOCIATION

Configuration Diagram - Simple example



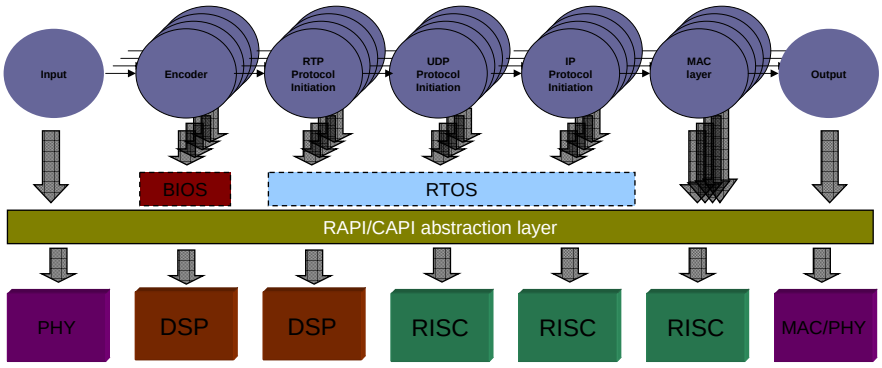
Multicore ASSOCIATION

Configuration Diagram – Complex Example



Multicore ASSOCIATION

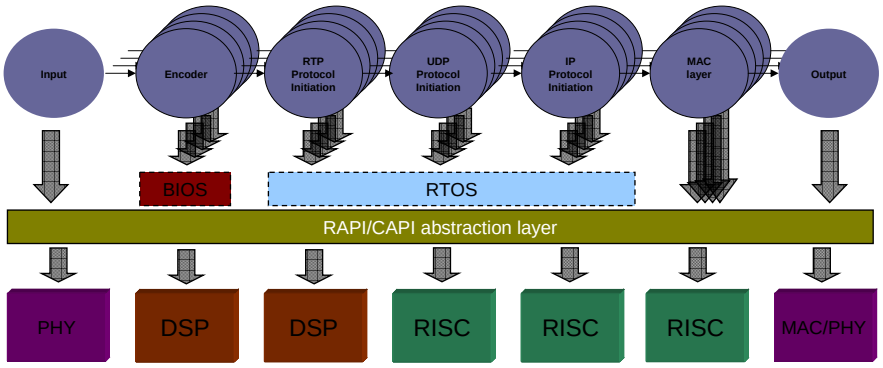
Platform / application scalability



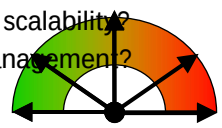
- Platform/application scalability?
- Core level power management?

Multicore ASSOCIATION

Core level power management



- Platform/application scalability?
- Core level power management?
- Efficiency?



Presented ad

Multicore ASSOCIATION

What can RAPI do for you?

- Who are you?
  - An OS vendor
    - A uniform foundation for present and future products
    - Enhanced interoperability
    - Larger addressable market
  - A chip company
    - More *application achievable* performance (MIPS/mm^2, MIPS/mW)
    - Better determinism
    - Reduced support burden
  - A software developer
    - More scalability (source code, binary?)
    - Better ease of use
    - More powerful development methodologies

Multicore ASSOCIATION

Targets for the Debug Working Group

- Embedded Multicore Systems.
  - Extend to Server, Client, HPC?
- Many CPUs on a chip, perhaps many chips on a board/system.
- Homogeneous (all of the same type of core) or heterogeneous.
- Various runtime models:
  - Single OS image; distributed processes, threads.
  - Multiple OS images, same or different.
  - Less than full OS on most or all cores, e.g. just a programming model runtime.
- Variety of programming models:
  - Multiple parallelism techniques, e.g. high-level functional and pipeline decomposition, coarse-grain & fine-grain data decomposition.
  - Range of communication models, i.e. CAPI, including streaming communication, lightweight message-passing, shared memory.
- Not limited to a specific HW/RT/programming model like SMP.
- Integrating multiple vendors & tool chains: inter-operation.

Multicore ASSOCIATION

Multicore Debug Challenges

- Parallel programming in general: multiple simultaneous threads of execution, such as synchronization bugs
- Differing user experiences for heterogeneous elements; tool chain and vendor compatibility issues
- Mapping from lower-level implementation constructs (e.g. memory, registers) to higher-level programming model constructs (e.g. messages, synchronization; see CAPI)
- Scale: what do we do beyond 2, 4, 8 cores? Need aggregate control and state inspection.

Multicore ASSOCIATION

Current Debug Initiatives

- Identifying and tying high-level requirements for multicore debugging to specific requirements on underlying infrastructures, such as OS or runtime layers.
- Extending existing debug interfaces in a standardized way to meet the needs of multicore debugging.
- Standardizing the connection between JTAG interfaces and debuggers. More specifically, enabling third-party debuggers to control systems with multiple cores with different JTAG interfaces from multiple vendors.
- We will seek to identify appropriate interfaces and drive standardization to enable a richer and more uniform debugging experience on multicore systems.

Multicore ASSOCIATION

Industry effort

- CAPI Draft Contributors

Anant Agarwal	Tilera
Todd Brian	Mentor
Sven Brehmer	PolyCore Software
Felix Burton	Wind River
Robert Craig	QNX
Simon Davidmann	Imperas
Bill Dittmann	Quadros Systems
Tomas Evensen	Wind River
Peter Flake	Imperas
Shay Gal-On	PMC-Sierra
Jim Holt	Freescale Semiconductor
Matthew Hall	Imperas
Maarten Konin	Wind River
Mark Lippett	Ignios
Radzy	Wind River
Kumar Venkatramani	Softjin

- About a dozen other companies have participated in our face to face meetings
- Leaders
  - Sven Brehmer (President, PolyCore Software) – CAPI Chair
  - Mark Lippett (CTO, Ignios Limited) – RAPI Chair
  - Jim Holt (Manager, Systems Modeling & Software Enablement Group, Freescale)

Multicore ASSOCIATION

Status CAPI / RAPI

- Working groups formed in August 2005
- Three face-to-face meetings with conference calls between
- Primary and secondary goals defined
- Use cases will be defined to drive requirements and the API specifications

Multicore ASSOCIATION

CAPI Draft time line goal

- Use cases 2 months
- Specific Requirements 2 months
- Iteration 1 specification 2 months
- Iteration 2 specification 2 months
- Draft specification review 1 month

Multicore ASSOCIATION

We want your input & Feedback

Get involved!

Multicore
ASSOCIATION