

Trends in Heterogeneous Systems Architectures (and how they'll affect parallel programming models)

Simon McIntosh-Smith simonm@cs.bris.ac.uk
Head of Microelectronics Research
University of Bristol, UK



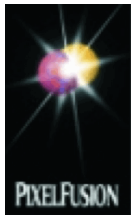
A brief biography



Graduated as Valedictorian in **Computer Science** from Cardiff University



1994 Joined **Inmos** to work for David May as a **microprocessor architect**



1999 Moved to **PixelFusion** – a high-tech start-up designing the first many-core general purpose graphics processor (GPGPU)



2002 Co-founded **ClearSpeed** as Director of Architecture and Applications



2009 Head of Microelectronics Research at the **University of Bristol**, focusing on HPC and computer architecture. FP7 EESI member, PRACE prototype panelist, Archer UK national supercomputer project group



Microelectronics Research in Bristol



Simon McIntosh-Smith
Head of Group



Prof David May



Prof Dhiraj Pradhan



Dr Jose
Nunez-Yanez



Dr Kerstin Eder



Dr Simon Hollis



Dr Dinesh
Pamunuwa

7 tenured staff, 7 research assistants, 16 PhD students

Energy Aware COmputing (EACO):

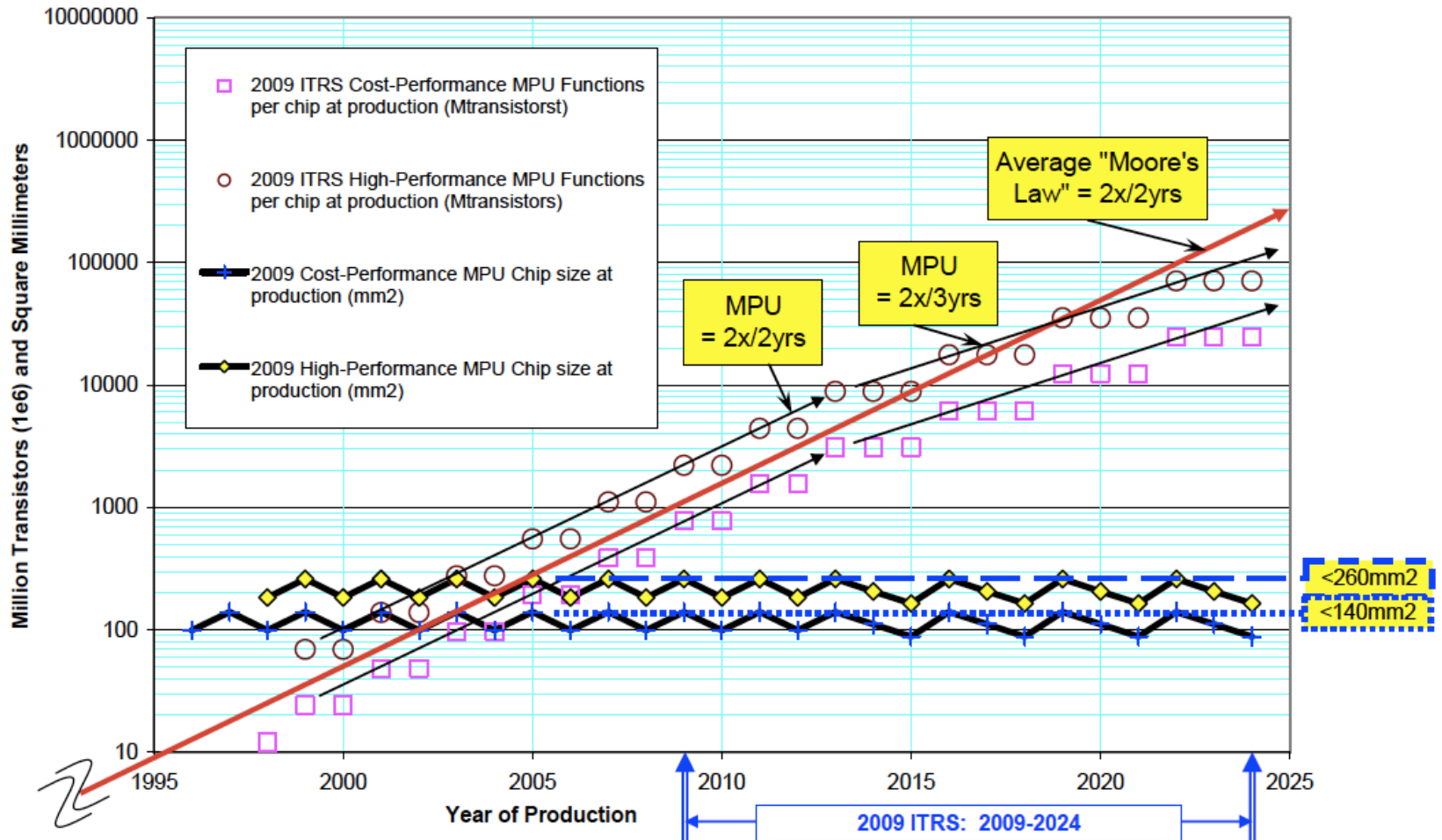
- **Multi-core and many-core computer architectures** (FP7 EESI)
 - ClearSpeed, XMOS, Inmos, Pixelfusion, ...
- **Algorithms for *heterogeneous architectures***
 - CPUs+GPUs, OpenCL
- **Electronic and Optical Network on Chip (NoC)**
- **Fault tolerant design** (hardware and software)
 - Near threshold computing for embedded medical devices (FP7 DeSyRe)
- Reconfigurable architectures (FPGA)
- Design verification (formal and simulation-based), formal specification and analysis
- Silicon process variation
- Design methodologies, modelling & simulation of MNT based structures and systems

Heterogeneous Systems Architectures



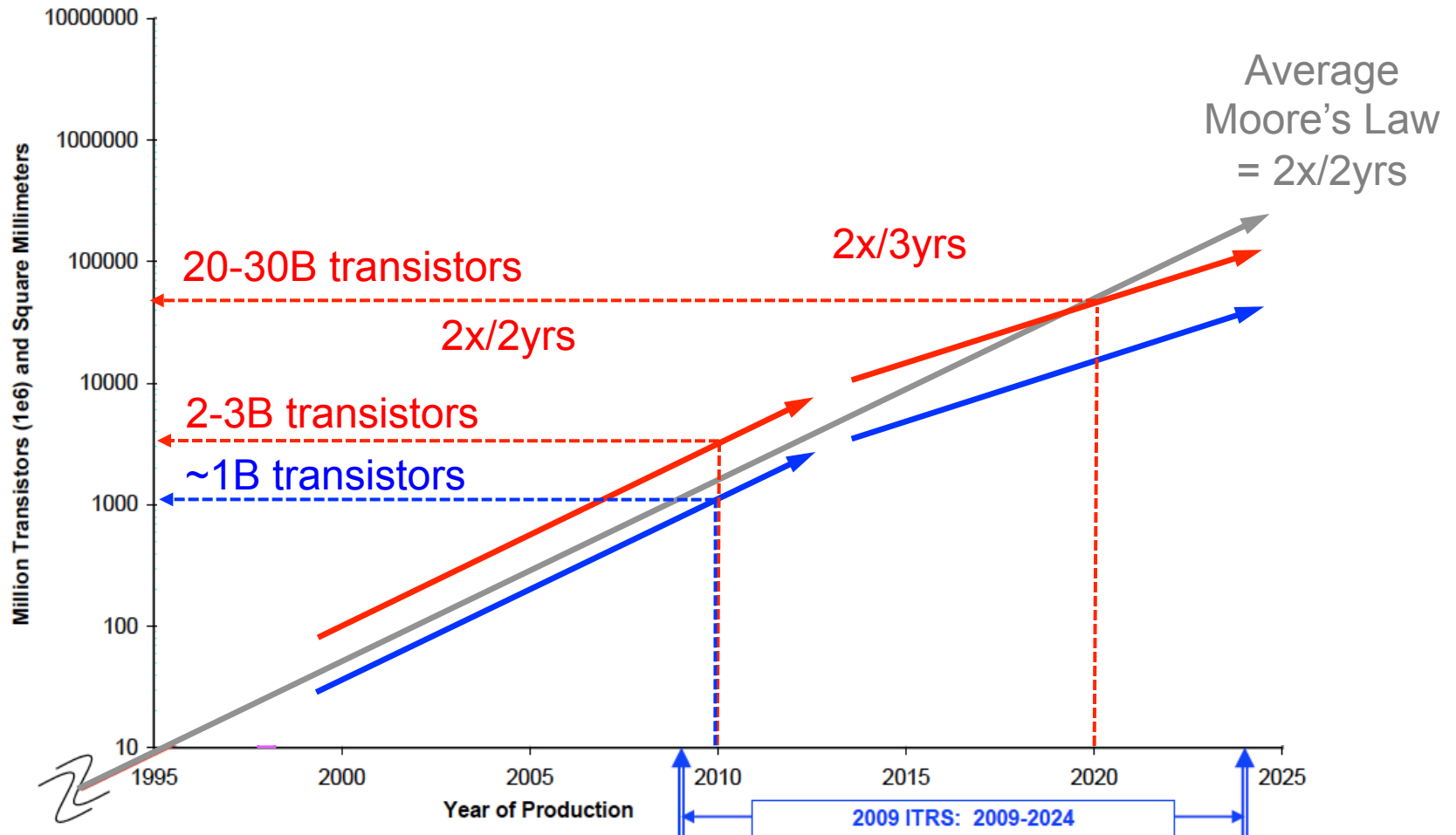
Moore's Law today

2009 ITRS - Functions/chip and Chip Size

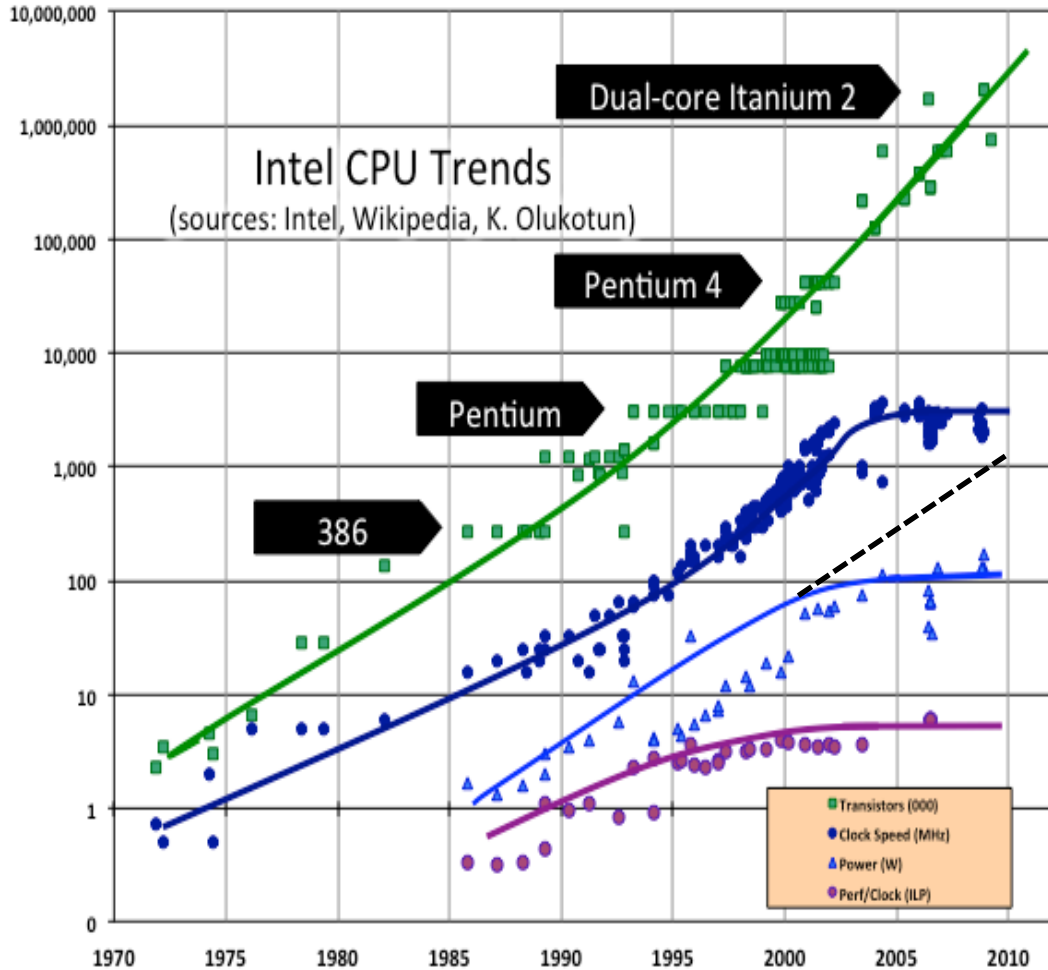


Moore's Law today

2009 ITRS - Functions/chip and Chip Size



🔥 Important technology trends



The real Moore's Law

The clock speed plateau

The power ceiling

Instruction level
parallelism limit

Herb Sutter's new outlook

<http://herbsutter.com/welcome-to-the-jungle/>

In the twilight of Moore's Law, the transitions to multicore processors, GPU computing, and HaaS cloud computing are not separate trends, but aspects of a single trend – mainstream computers from desktops to 'smartphones' are being permanently transformed into heterogeneous supercomputer clusters. **Henceforth, a single compute-intensive application will need to harness different kinds of cores, in immense numbers, to get its job done.**

The free lunch is over. Now welcome to the hardware jungle.

Major hardware trends



The five major hardware trends that will affect Exascale software

1. Heterogeneity
2. Changes to memory hierarchies
3. The impact of fault tolerance
4. Focus on energy efficiency
5. Scale

(For a discussion of 2-5 see "Major hardware trends affecting Exascale developments and their potential impact on software", PRACE- 1IP Work Package 9 Future Technologies Workshop, Daresbury, UK, Apr 11th 2012, http://www.cs.bris.ac.uk/~simonm/publications/sms_prace_exascale_2012.pdf)

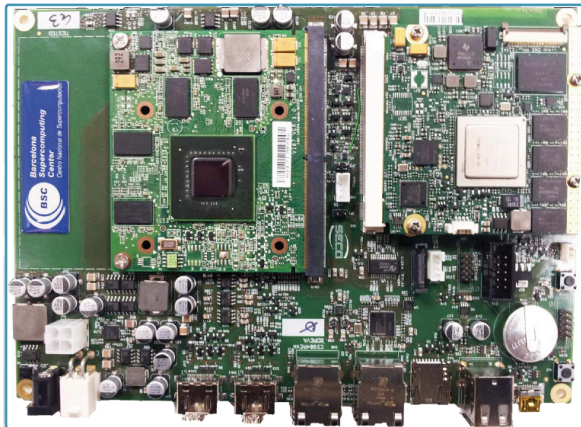
Causes of heterogeneity

- Multiple types of core
- Interconnect
- Memory type, capacity, ...
- Software (OS, middleware, tools, ...)

Heterogeneous Systems

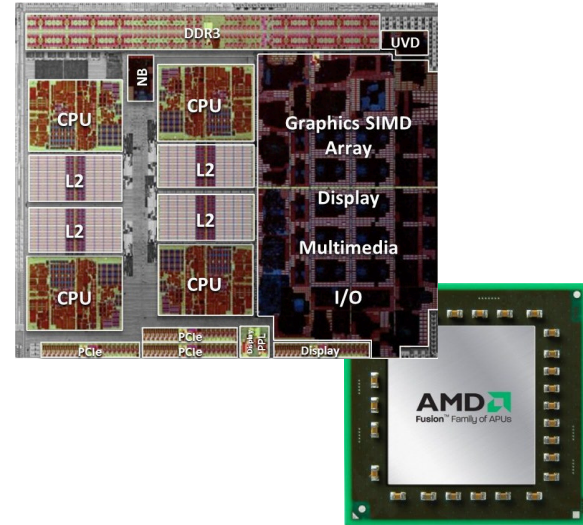


Intel MIC



FP7 Mont Blanc ARM + GPU

AMD Llano Fusion APUs



Tegra 3 The World's First Mobile Quad Core, with 5th Companion Core for Low Power

CPU	Quad Core, with 5th Companion Core — Up to 1.4GHz Single Core, 1.3GHz Quad Core
GPU	Up to 3x Higher GPU Performance — 12 Core GeForce GPU
VIDEO	Blu-Ray Quality Video — 1080p High Profile @ 40Mbps
POWER	Lower Power than Tegra 2 — Variable Symmetric Multiprocessing (vSMP)
MEMORY	Up to 3x Higher Memory Bandwidth — DDR3L-1500, LPDDR2-1066
IMAGING	Up to 2x Faster ISP (Image Signal Processor)
AUDIO	HD Audio, 7.1 channel surround
STORAGE	2-6x Faster — MMC 4.41, SD3.0, SATA-II

NVIDIA Tegra, Project Denver

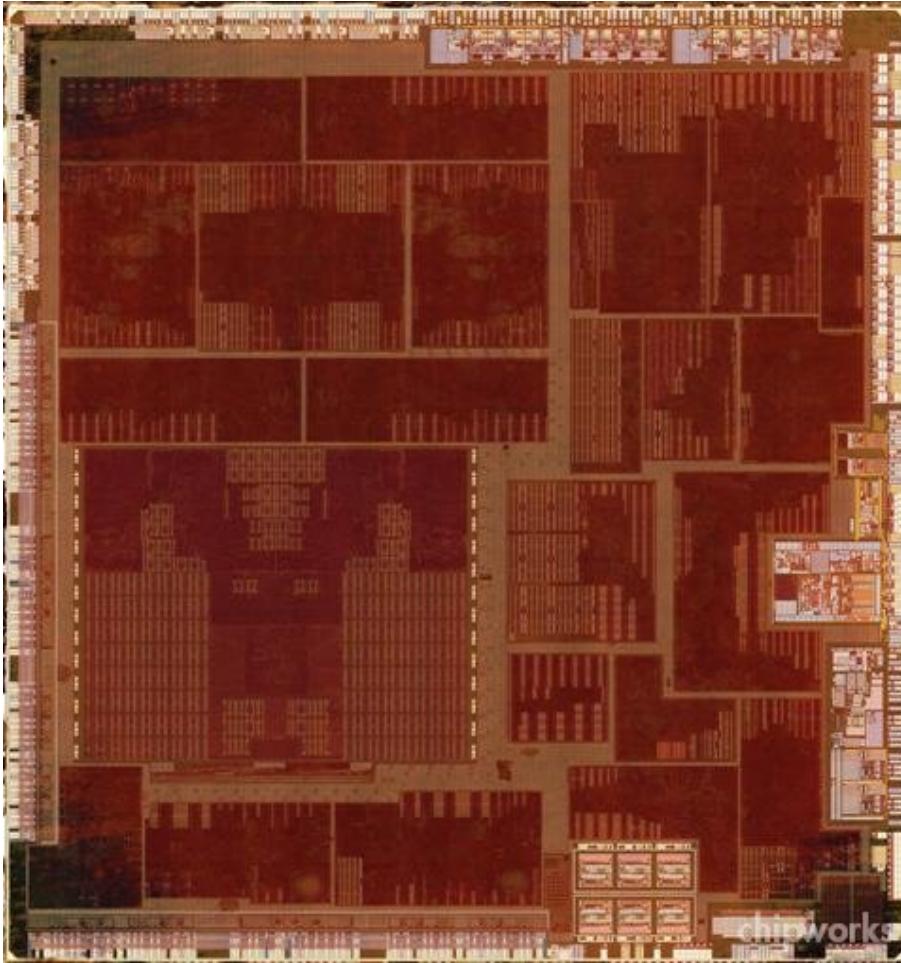
🔥 Heterogeneity is mainstream



Quad-core ARM Cortex A9 CPU

Quad-core SGX543MP4+ Imagination GPU

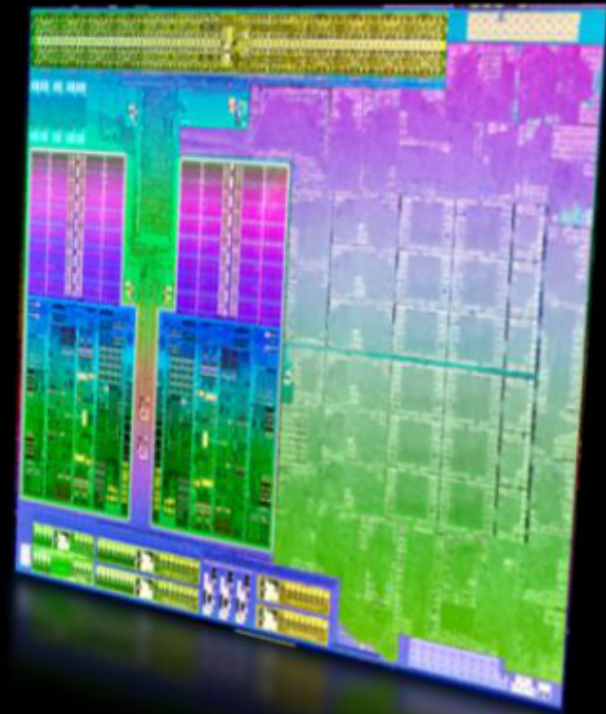
🔥 You might have one in your pocket



🔥 Hot off the press....

INTRODUCING TRINITY: SECOND GENERATION A-SERIES APU

- Premium discrete-level graphics and gaming performance
- Up to 726 GFLOPs compute
- 2X the performance/Watt of Llano
- 56% improvement in graphics performance
- Up to 12 hours battery life
- All in highly mobile, low-power form factors



Implications

- New programming languages, models, ...
- Dynamically adaptive software
 - Discover resources at run-time
- Auto-tuning
- Application frameworks and libraries



OpenCL

(Open Computing Language)

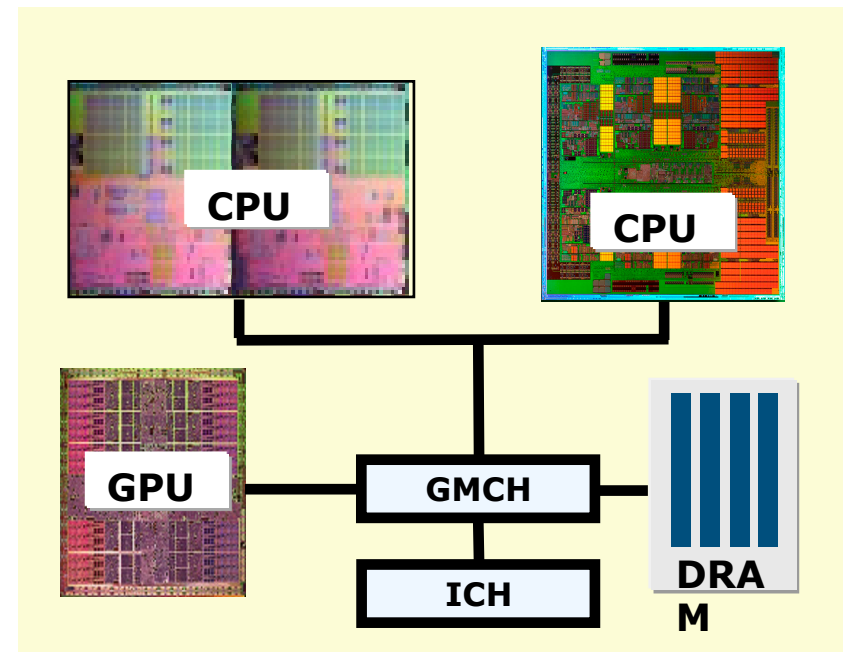


OpenCL

- Open standard for portable, parallel programming of heterogeneous systems
- Lets programmers write a single **portable** program that uses **all** resources in the heterogeneous platform

A modern system includes:

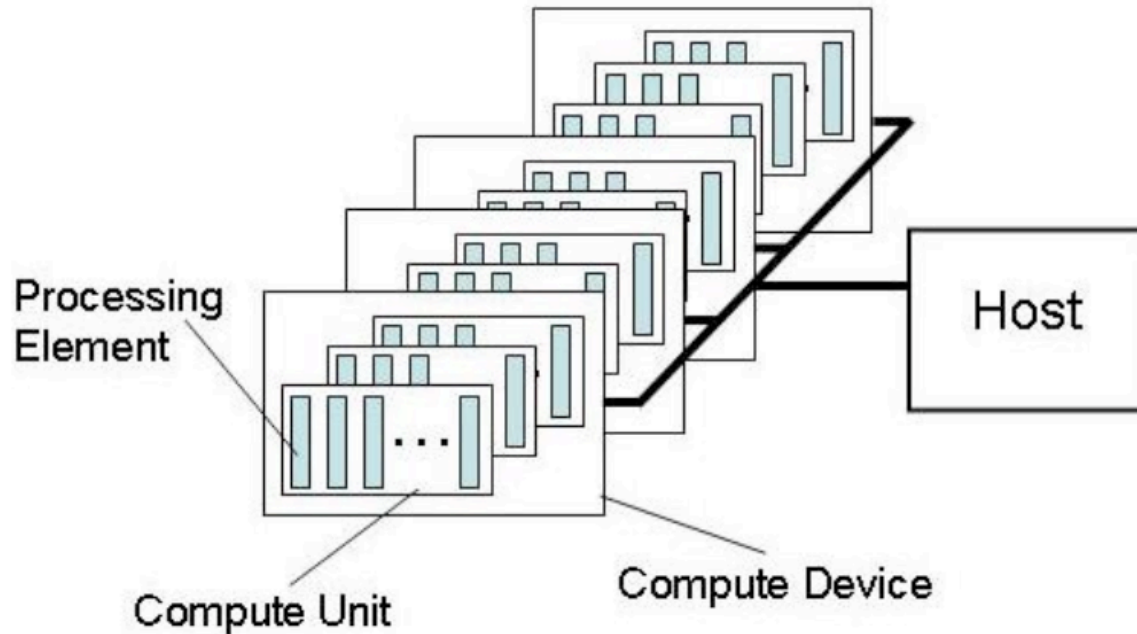
- One or more CPUs
- One or more GPUs
- DSP processors
- ...other devices?



GMCH = graphics memory control hub

ICH = Input/output control hub

🔥 OpenCL platform model



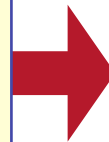
- One Host + one or more Compute Devices
 - Each Compute Device is composed of one or more Compute Units
 - Each Compute Unit is further divided into one or more Processing Elements

The BIG idea behind OpenCL

- Replace loops with functions (a kernel) executing at each point in a problem domain (index space).
- E.g., process a 1024 x 1024 image with one kernel invocation per pixel or $1024 \times 1024 = 1,048,576$ kernel executions

Traditional loops

```
void
trad_mul(const int n,
         const float *a,
         const float *b,
         float *c) {
    int i;
    for (i=0; i<n; i++)
        c[i] = a[i] * b[i];
}
```



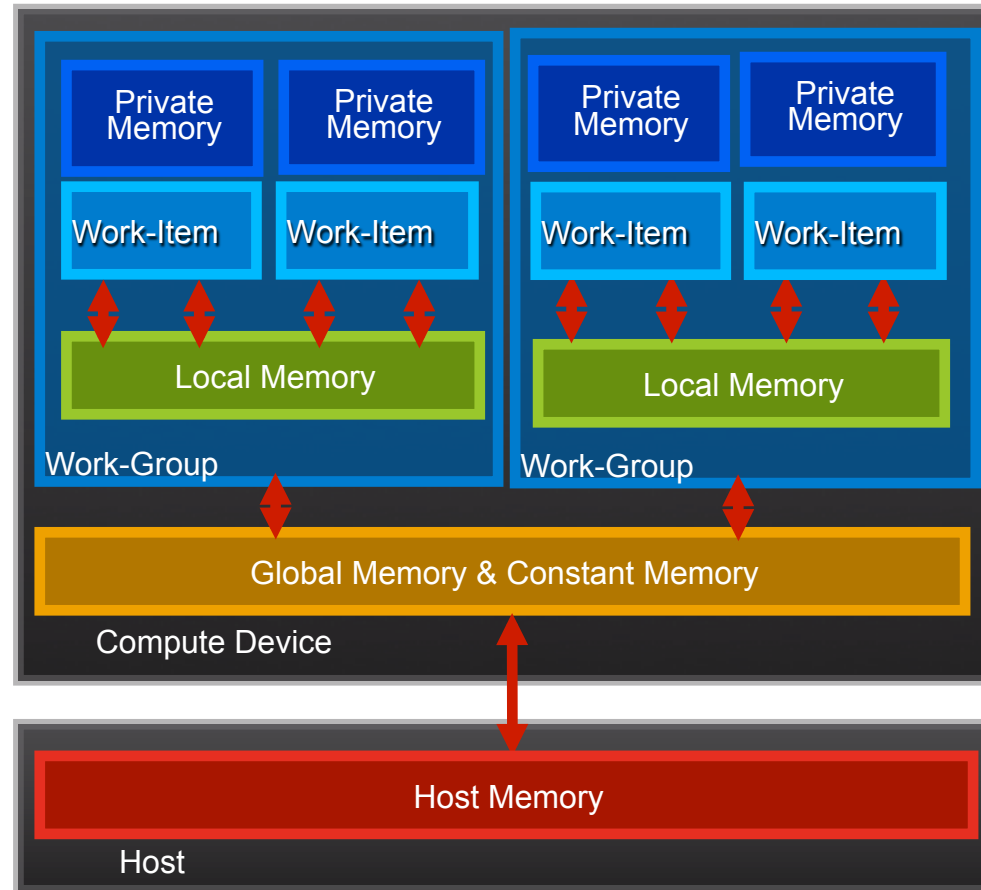
Data parallel OpenCL

```
kernel void
dp_mul(global const float *a,
      global const float *b,
      global float *c) {
    int id = get_global_id(0);

    c[id] = a[id] * b[id];
} // execute over "n" work-items
```

🔥 OpenCL memory model

- **Private Memory**
 - Per Work-Item
- **Local Memory**
 - Shared within a Work-Group
- **Global / Constant Memories**
 - Visible to all Work-Groups
- **Host Memory**
 - On the CPU



Memory management is explicit

You must move data from host → global → local *and* back

Issues with OpenCL*

- It does not **compose**
 - Disjoint memory address spaces (local/global)
 - Barriers
- It provides no resource management
 - Kernels are a statically allocated resource

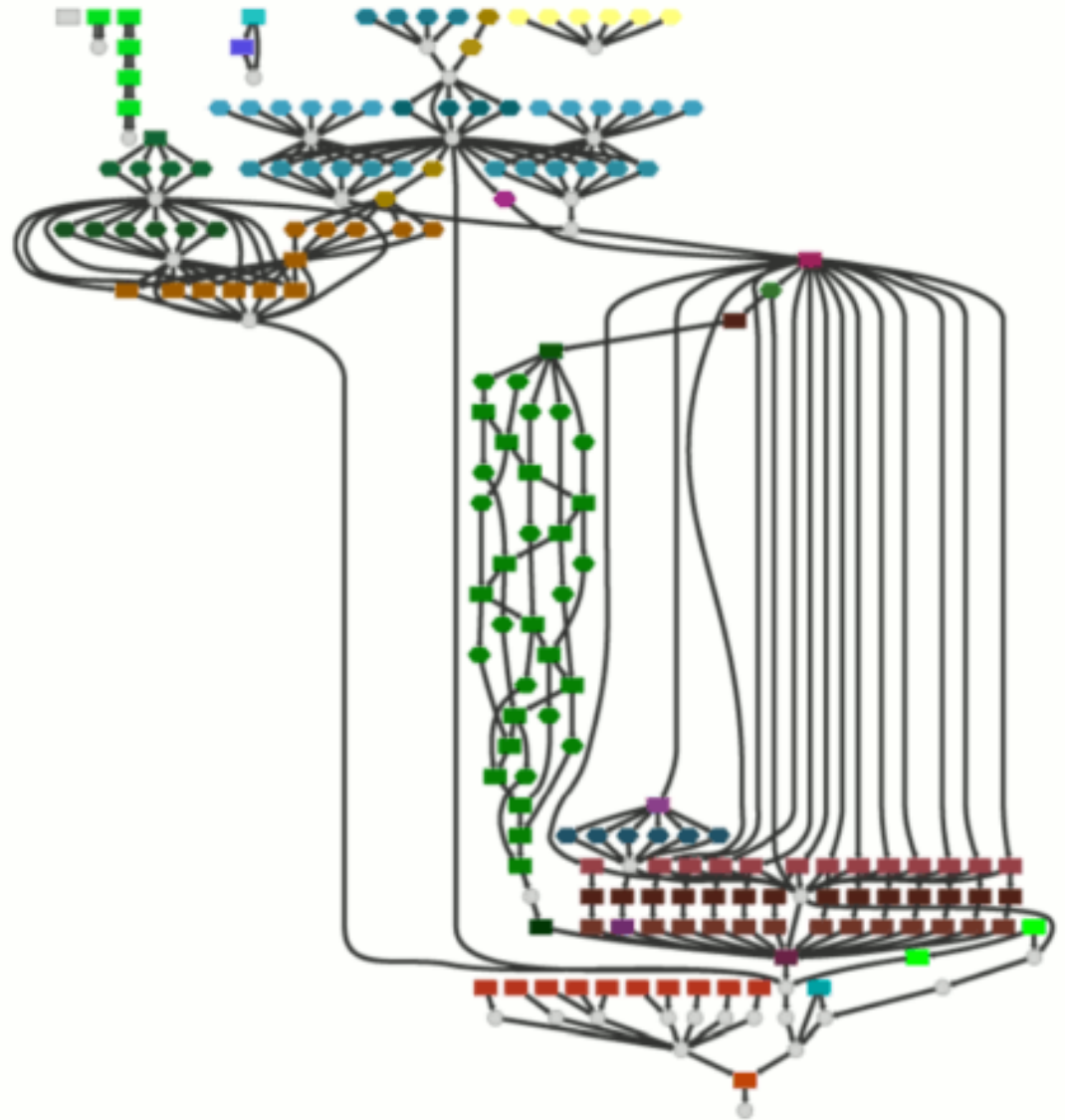
* And most other parallel programming languages

🔥 Composability

Need to support many algorithms, even within a single application

Task farms, pipeline, data parallelism, ...

See similar composability challenges with OpenMP and parallel libraries, for example



Heterogeneous System Architecture (HSA)



HSA overview

- The HSA Foundation launched this week and already includes AMD, ARM, Imagination Technology and Texas Instruments
- HSA is a new, open architecture specification
 - HSAIL virtual ISA
 - HSA memory model
 - HSA dispatch
- Provides an optimised platform architecture for heterogeneous programming models such as OpenCL, C++AMP, et al

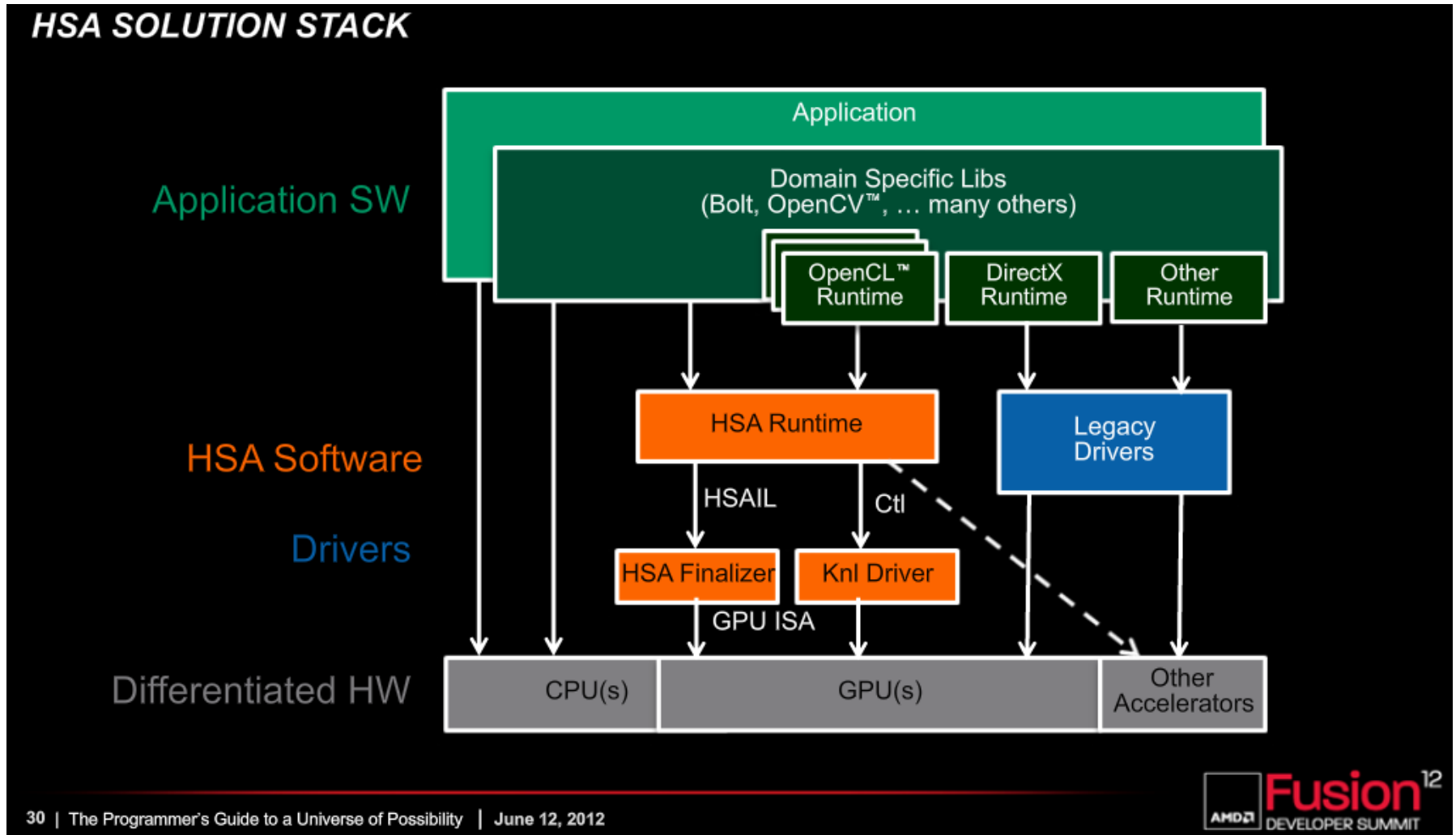
Announced 12/6/2012



THE HSA FOUNDATION: ACTIVITIES

- ◆ Nonprofit, open standardization body for HSA platforms that will own the development and evangelization of the architecture going forward
- ◆ Make heterogeneous programming easy and a first-class pervasive complement to CPU computing
- ◆ Continue to increase the power efficiency of HSA, keeping it the platform of choice from smartphones to the cloud
- ◆ Bring to market strong development solutions (tools, libraries, OS runtimes) to drive innovative advanced content and applications
- ◆ Foster growth of heterogeneous computing talent through HSA developer training and academic programs to drive both learning and innovation

HSA overview



🔥 HSA features: simplifying programming

- Much finer grained integration of CPU and GPU cores in silicon
- Unified address space for all cores
- Will support GPU context switching, preemption
- PGAS-style distributed arrays
 - Memory hierarchy abstraction to address function composition
- First class barrier objects
 - Aids composability

HSA Intermediate Layer (HSAIL)

- Virtual ISA for parallel programs
- Similar idea to LLVM IR - a good target for compilers
- *Finalised* to specific ISA by a JIT compiler
- Features:
 - Explicitly parallel
 - Support for exceptions, virtual functions and other high-level features
 - Syscall methods (I/O, printf etc.)
 - Debugging support

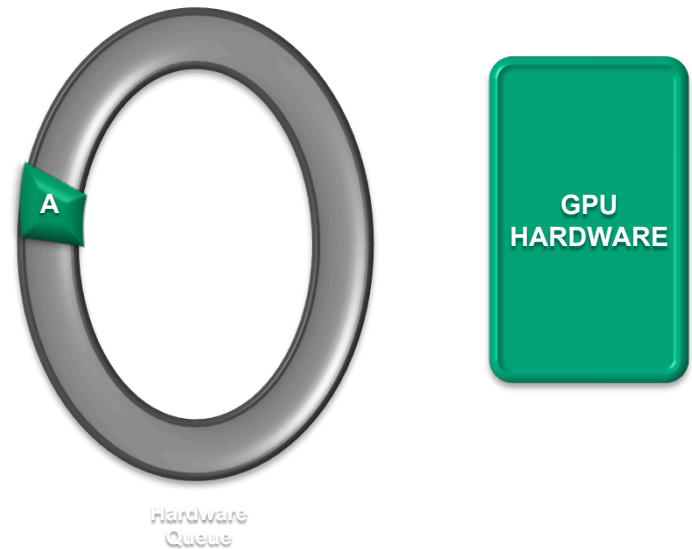
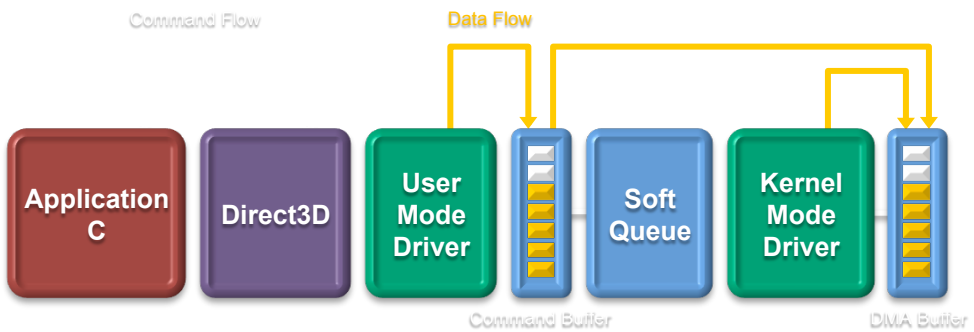
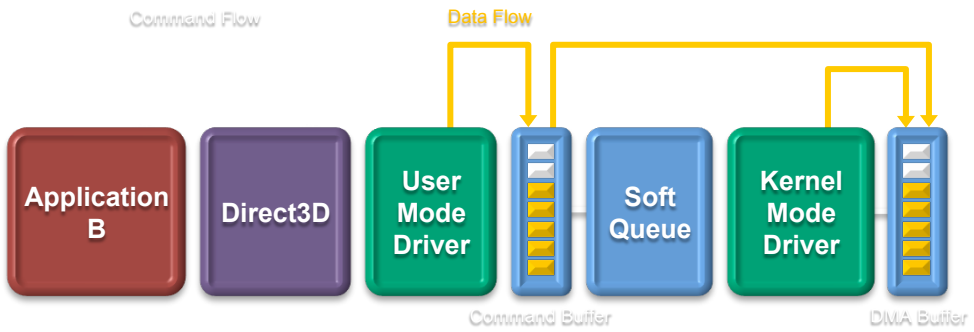
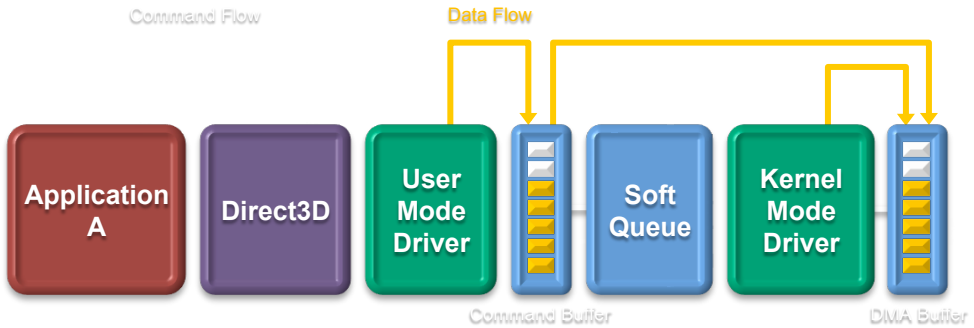
HSA memory model

- Compatible with C++11, Java and .NET memory models
- Relaxed consistency
- Designed to support both managed language (such as Java) and unmanaged languages (such as C)
- Will make it much easier to develop 3rd party compilers to target a wide range of heterogeneous products
 - Fortran, C++, C++AMP et al

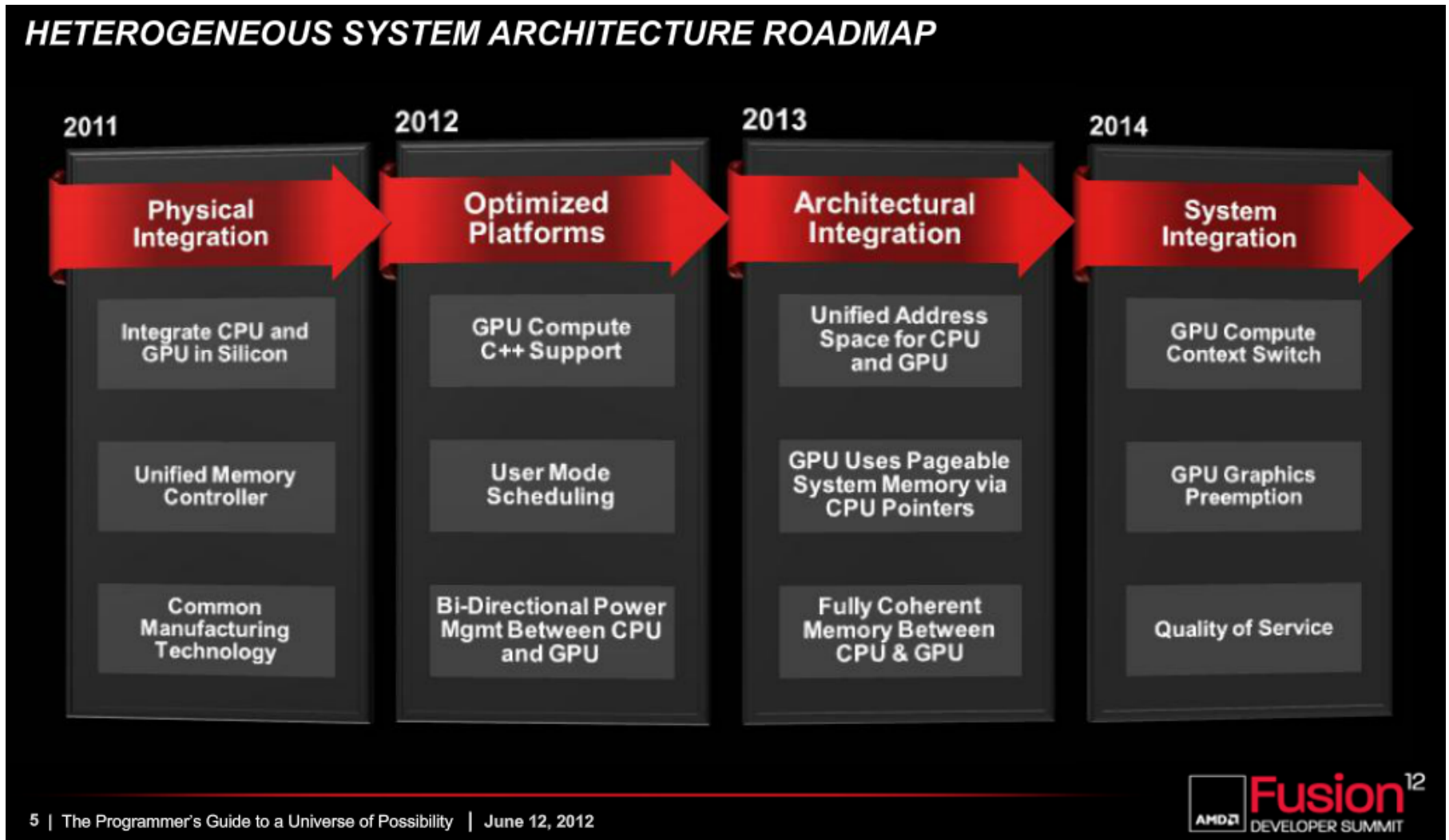
HSA dispatch

- HSA designed to enable heterogeneous task queuing
 - A work queue per core
 - Distribution of work into queues
 - Load balancing by work stealing
- Any core can schedule work for any other, including itself
- Significant reduction on overhead of scheduling work for a core

🔥 Today's command and dispatch flow



HSA roadmap from AMD



5 | The Programmer's Guide to a Universe of Possibility | June 12, 2012



HSA tools released as open source

AMD'S OPEN SOURCE COMMITMENT TO HSA

- We will open source our linux execution and compilation stack
 - Jump start the ecosystem
 - Allow a single shared implementation where appropriate
 - Enable university research in all areas

Component Name	AMD Specific	Rationale
HSA Bolt Library	No	Enable understanding and debug
OpenCL HSAIL Code Generator	No	Enable research
LLVM Contributions	No	Industry and academic collaboration
HSA Assembler	No	Enable understanding and debug
HSA Runtime	No	Standardize on a single runtime
HSA Finalizer	Yes	Enable research and debug
HSA Kernel Driver	Yes	For inclusion in linux distros

Conclusions


- Heterogeneity is an increasingly important trend
- The market is finally starting to create and adopt the necessary open standards
 - Proprietary models likely to start declining now
 - Don't get locked into any one vendor!
- Parallel programming models are likely to (re)proliferate
- Exciting times ahead!

The screenshot shows a web browser window displaying the website for the μ Research Group at the University of Bristol. The browser's address bar shows the URL <http://www.cs.bris.ac.uk/Research/Micro/>. The website header includes the University of Bristol logo and the text "Dept. of Computer Science | Dept. of Electronics & Electrical Engineering | Faculty of Engineering | University of Bristol". The main heading is "microelectronics research group". A navigation menu contains links for HOME, NEWS, DIARY, PUBLICATIONS, COLLABORATION, PROJECTS, PEOPLE, EACO, and WIKI. A large image of a microchip is featured prominently. To the right, there are sections for "Upcoming events" (listing "The Multicore Challenge II: Programming Multicore Systems" on 5 Sep 2011) and "Supporters & affiliations" (listing logos for Cadence, Mentor Graphics, Infineon, Imagination, Xmos, TVS, ARM, NVIDIA, AMD, and nag). The main content area is divided into "Recent news" and "Recent publications".

University of BRISTOL

microelectronics research group

HOME NEWS DIARY PUBLICATIONS COLLABORATION PROJECTS PEOPLE EACO WIKI



The *Microelectronics Research Group* is a cross-departmental meeting of minds, incorporating individuals from the Departments of Computer Science and Electronic Engineering at the University of Bristol. We are interested in solving tomorrow's micro-architecture challenges and work closely with industry on many exciting, innovative projects.

Recent news

Research assistant vacancy: massively parallel software libraries for high performance computing
26 Aug 2011
We are looking for another research assistant to work within the group... [read more](#).

Research assistant vacancy: Adaptive, reliable heterogeneous MPSoCs
24 Aug 2011
We are looking for a research assistant to work within the group... [read more](#).

OpenCL workshop at SC11 to be co-run by Simon McIntosh-Smith
22 Aug 2011
Simon McIntosh-Smith will be co-running an all-day workshop at the IEEE/ACM Conference on High Performance Computing, Networking, Storage and Analysis (SuperComputing) with Tim Mattson from Intel and Ben Gaster from AMD... [read more](#).

Older news...

Recent publications

Towards Safe Human-Robot Interaction
Elena Corina Grigore, [Kerstin Eder](#), Alexander Lenz, Sergey Skachek, Anthony G. Pipe and [Christopher Melhuish](#), 2011

Upcoming events

The Multicore Challenge II: Programming Multicore Systems
5 Sep 2011 at 1:00 in University of the West of England, Frenchay Campus, Bristol
Experts in multicore technology are coming together in Bristol in September to look at the challenges of developing multicore systems... [read more](#).
More events...

Supporters & affiliations

The μ Research Group works closely with the following companies and organisations.

