The Tile Processor™ Architecture:

Embedded Multicore for Networking and Digital Multimedia

Tilera Corporation

August 20th 2007

Hotchips 2007

Markets Demanding More Performance

Networking market

- Demand for high performance
  - Services being integrated in the infrastructure
  - Faster speeds 1Gbps » 2Gbps » 4Gpbs » 10 Gbps
- Demand for more services
  - In-line L4 – L7 services, intelligence everywhere
  - Integration of video with networking


Security Appliances


Switches


Routers

Digital Multimedia market

- Demand for high performance
  - H.264 encoding for High Definition
  - Pre & post processing
- Demand for more services
  - VoD, video conferencing, transcoding, transrating

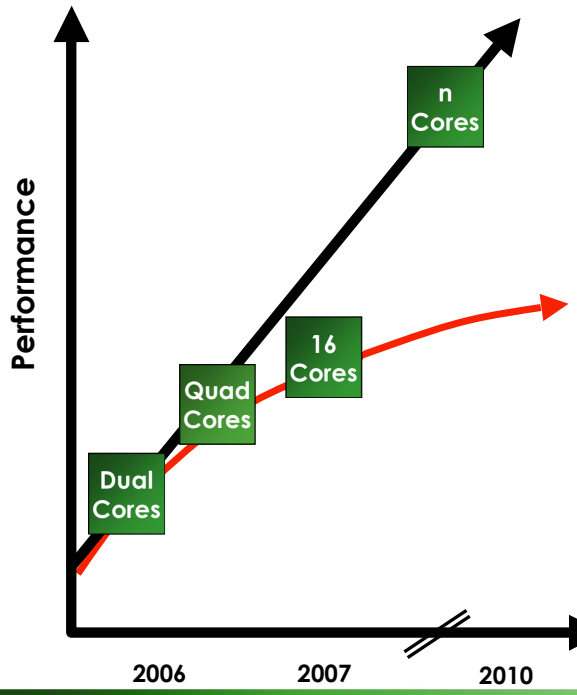

Video Conferencing


Cable & Broadcast


Surveillance DVR

… and with power efficiency and programming ease

Industry Aggressively Embracing Multicore

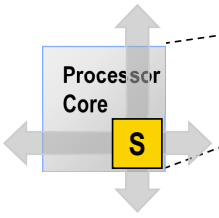


Inherent architectural bottlenecks:

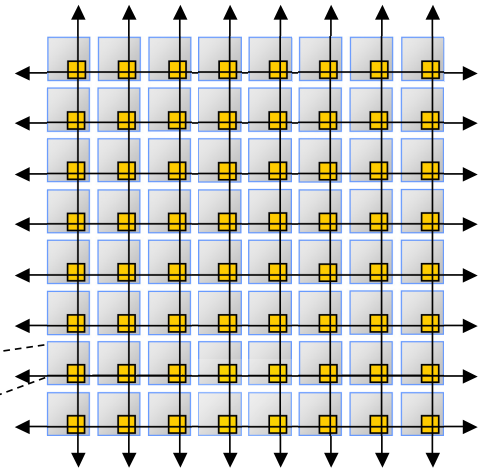
- No scalability
- Power inefficiency
- Primitive programming model

Tiled Multicore Closes the Performance Gap

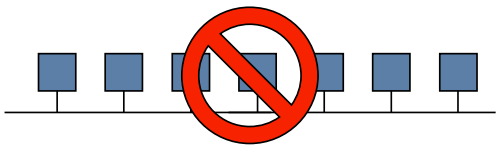
- Cores connected by mesh network
- Unlike buses, meshes scale
- Resources are distributed
  - improved power efficiency
- Modular – easy to layout and verify



Core + Switch = Tile



Current Bus Architecture



Introducing the TILE64™ Processor

Multicore Performance (90nm)

Number of tiles (general purpose cores)	64
On chip distributed cache	5 MB
Operations @ 750MHz (32, 16, 8 bit)	144-192-384 BOPS
On chip interconnect bandwidth	32 Terabits per second
Bisection bandwidth	2 Terabits per second

Power Efficiency

Power per tile	170 – 300 mW
Clock speed	600-1000 MHz

I/O and Memory Bandwidth

I/O bandwidth	40 Gbps
Main Memory bandwidth	200 Gbps

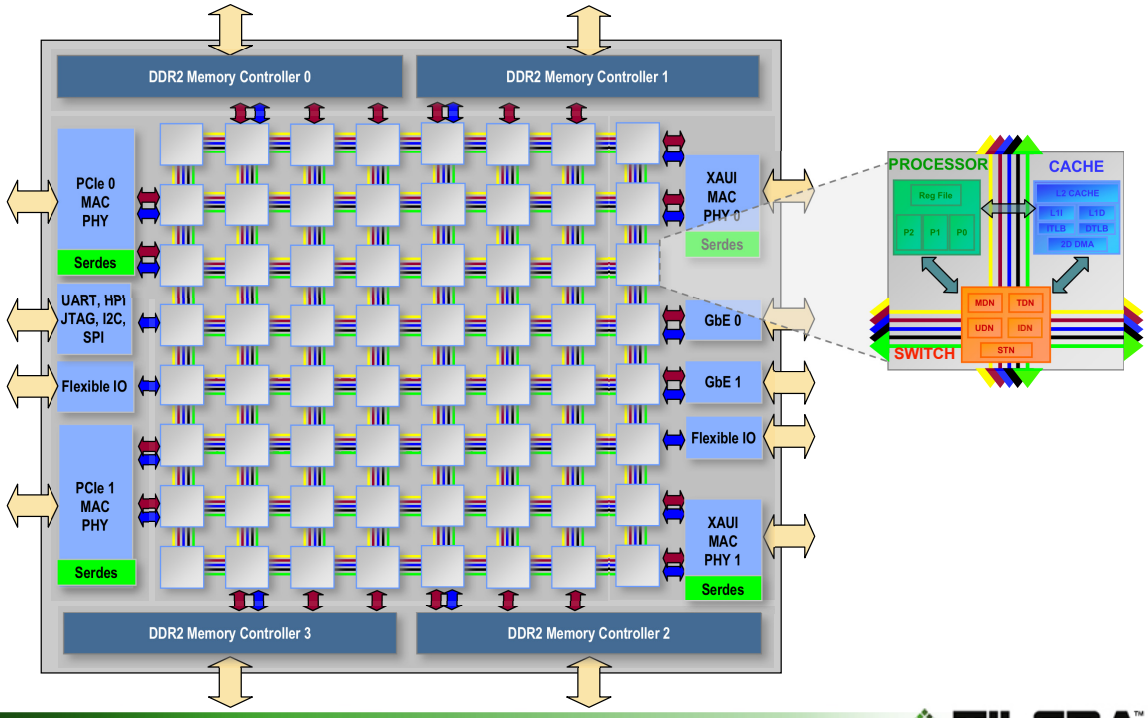
Programming

ANSI standard C
SMP Linux programming
Stream programming

The TILE64 chip is shipping today

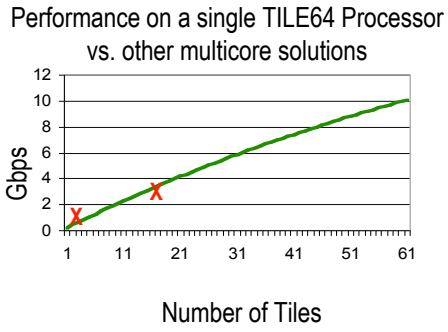
TILE64 Processor Block Diagram

A Complete System on a Chip

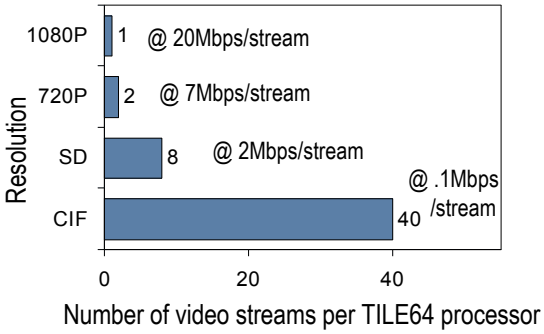


Performance in Networking and Video

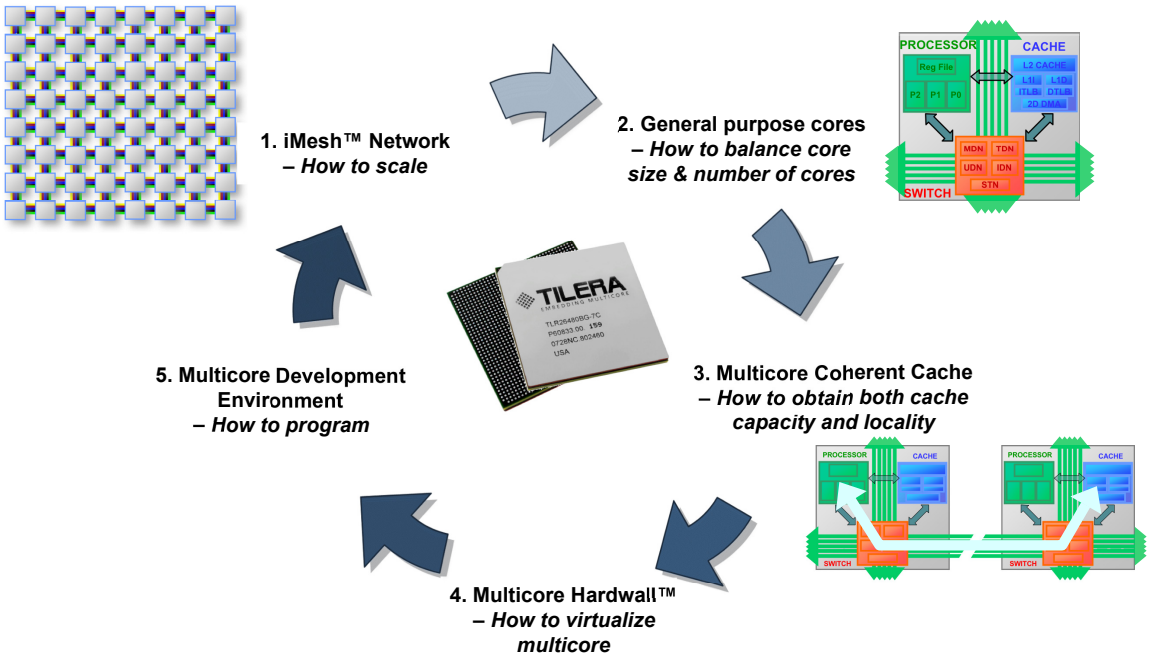
- Performance in networking
  - 10Gbps of SNORT
  - Complete SNORT database
  - All SNORT pre-processors
  - Customer's real world data
  - Open source SNORT software base



- Performance in video
  - H.264 video encode
  - Encodes 40 CIF video streams @ 30fps
  - Encodes two 720p HD streams @ 30fps
  - PSNR of 35 or more
  - Open source X264 software base

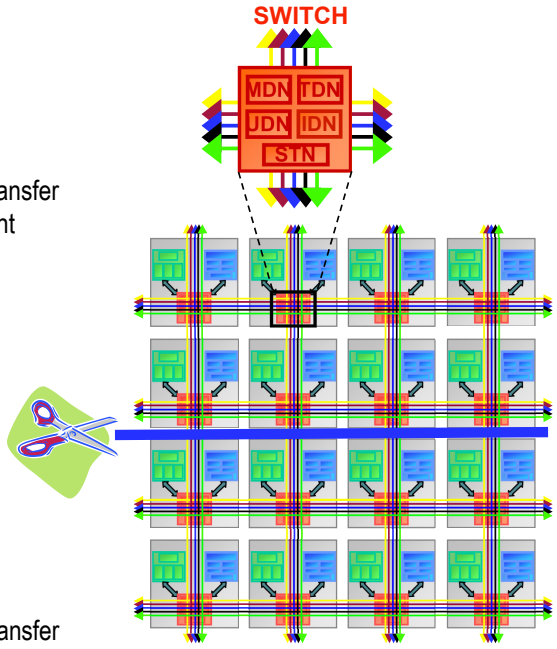


Key Innovations

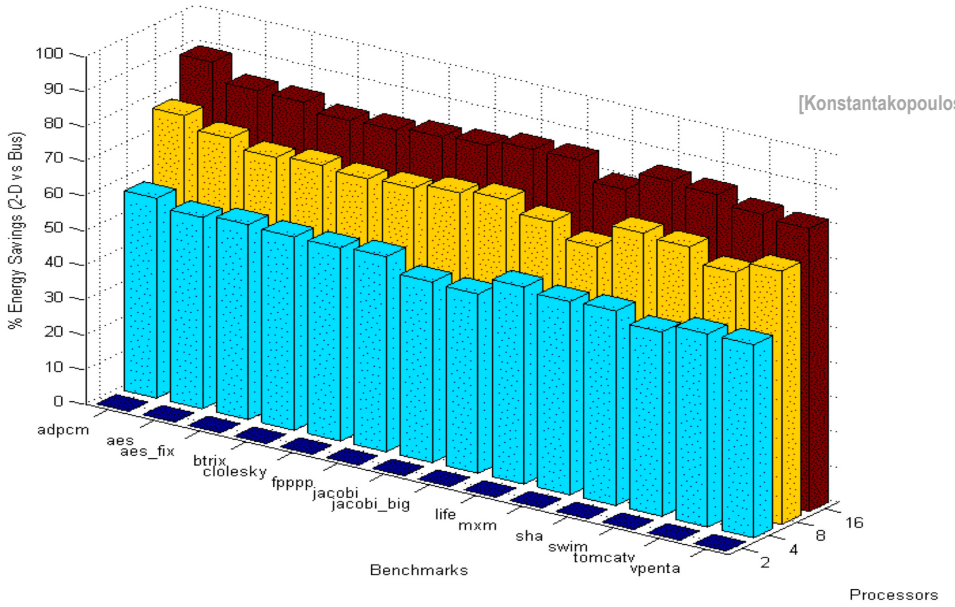


1- iMesh On-Chip Network Architecture

- Distributed resources
  - 2D Mesh Peer-to-peer tile networks
  - 5 independent networks
  - Each with 32-bit channels, full duplex
  - Tile-to-memory, tile-to-tile, and tile-to-IO data transfer
  - Packet switched, wormhole routed, point-to-point
  - Near-neighbor flow control
  - Dimension-ordered routing
- Performance
  - ASIC-like one cycle hop latency
  - 2 Tbps bisection bandwidth
  - 32 Tbps interconnect bandwidth
- 5 independent networks
  - One static, four dynamic
  - IDN – System and I/O
  - MDN – Cache misses, DMA, other memory
  - TDN – Tile to tile memory access
  - UDN, STN – User-level streaming and scalar transfer



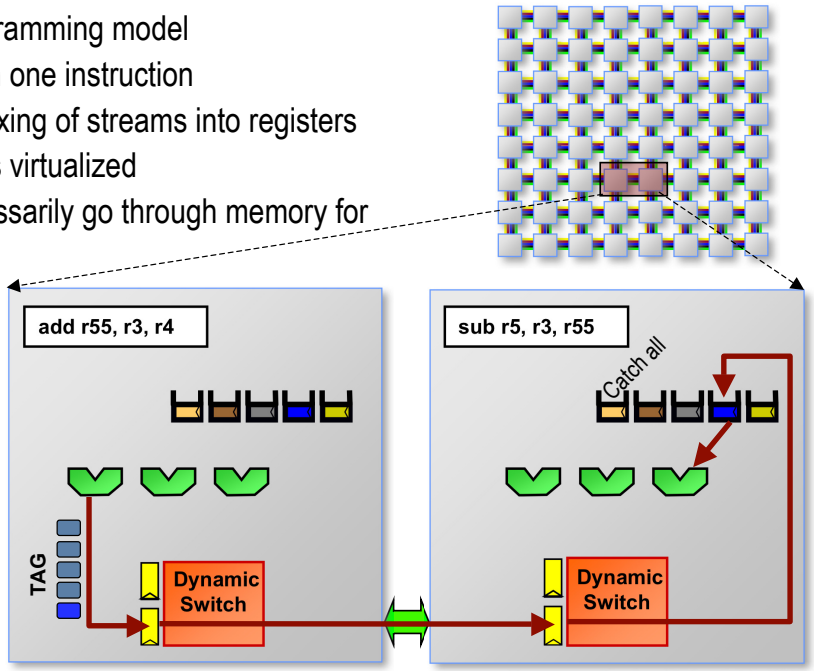
Meshes are Power Efficient



More than 80% power savings over buses

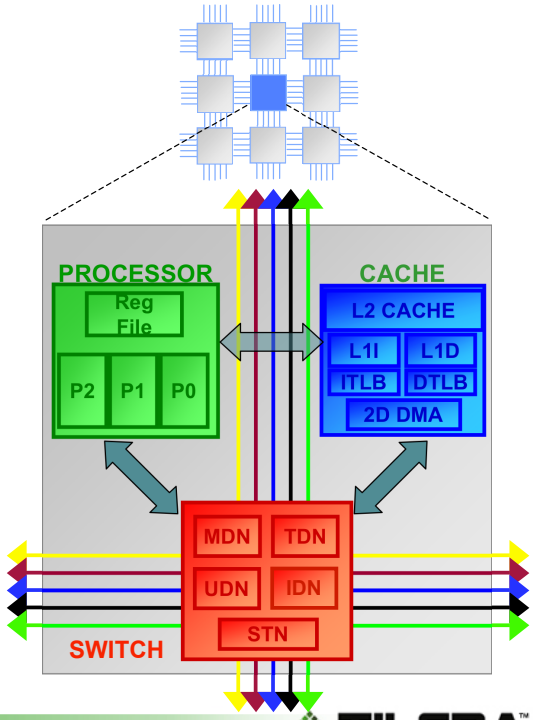
Direct User Access to Interconnect

- Enables stream programming model
- Compute and send in one instruction
- Automatic demultiplexing of streams into registers
- Number of streams is virtualized
- Streams do not necessarily go through memory for power efficiency



2- Full-Featured General Purpose Cores

- Processor
  - Homogeneous cores
  - 3-way VLIW CPU, 64-bit instruction size
  - SIMD instructions: 32, 16, and 8-bit ops
  - Instructions for video (e.g., SAD) and networking (e.g., hashing)
  - Protection and interrupts
- Memory
  - L1 cache: 8KB I, 8KB D, 1 cycle latency
  - L2 cache: 64KB unified, 7 cycle latency
  - Off-chip main memory, ~70 cycle latency
  - 32-bit virtual address space per process
  - 64-bit physical address space
  - Instruction and data TLBs
  - Cache integrated 2D DMA engine
- Switch in each tile
- Runs SMP Linux
- 7 BOPS/watt



How to Size a Core – "KILL Rule" for Multicore

Kill If Less than Linear

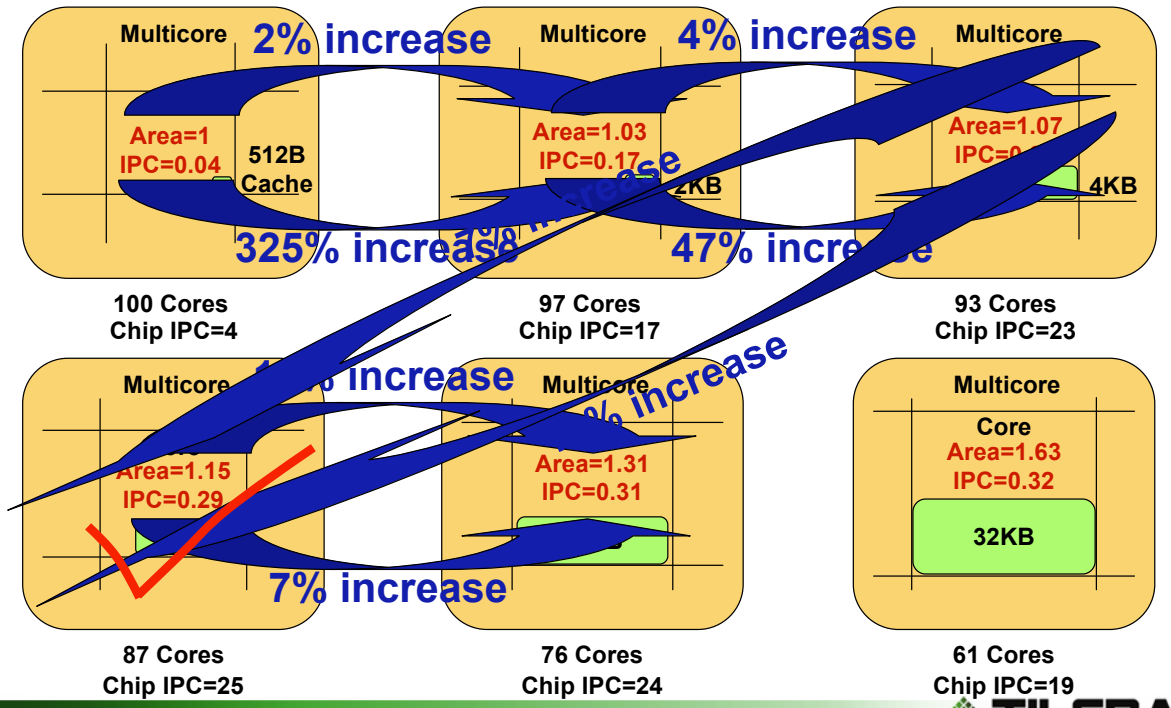
Increase resource size within a core only if
for every 1% increase in core area
there is at least a 1% increase in core performance

Insight: For parallel applications,
multicore performance can increase in proportion
to the increase in area as more cores are added

Leads to power-efficient multicore design

TILERA EMBEDDING MULTICORE

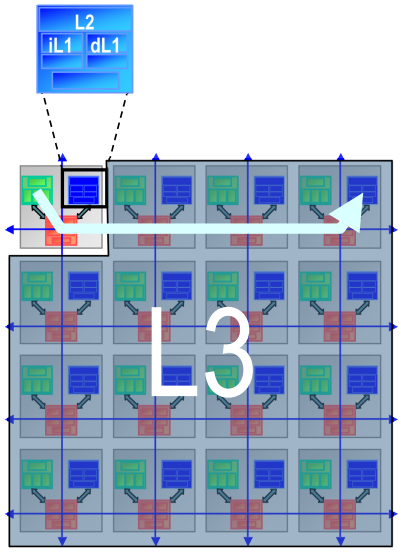
E.g., Kill Rule for Cache Size in Video Codec



TILERA EMBEDDING MULTICORE

3- Distributed Coherent Caching

- Each tile has local L1 and L2 caches
- Combined L2 caches of all tiles act as distributed 4MB L3 cache
- ***Low Latency*** of local L1 and L2 caches
- ***Capacity*** of large distributed L3 cache
- Caches are coherent, enabling running SMP Linux



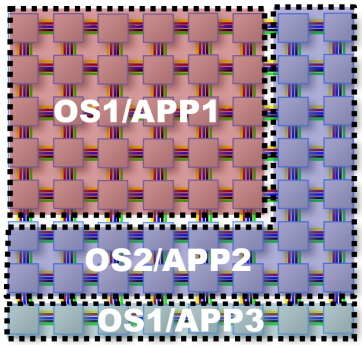
4- Multicore Hardwall Technology for Protection and Virtualization

The protection and virtualization challenge

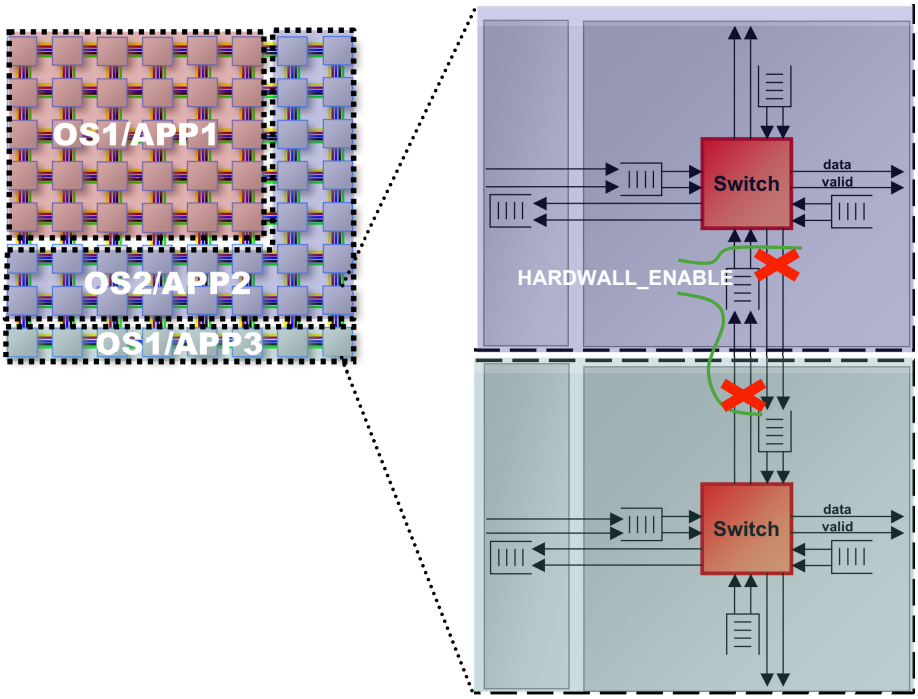
- Multicore interactions make traditional architectures hard to debug and protect
- Memory based protection will not work with direct IO interfaces and messaging
- Multiple OS's and applications exacerbate this problem

Multicore Hardwall technology

- Protects applications and OS by prohibiting unwanted interactions
- Configurable to include one or many tiles in a protected area



Multicore Hardwall Implementation



5- Multicore Software Tools and Programming

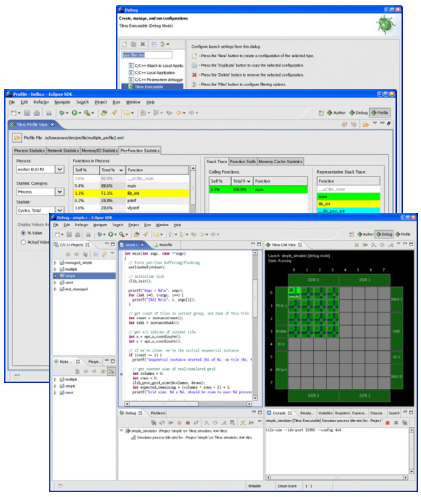
- Arguably biggest multicore challenge
- Multicore software tools challenge
  - Current tools are primitive – use single process based models
  - E.g., how do you single-step an app spread over many cores
  - Many multicore vendors do not even supply tools
- Multicore programming challenge
  - Key tension between getting up and running quickly using familiar models, while providing means to obtain full multicore performance
  - How do you program 100—1000 cores?
  - Intel Webinar likens threads to the “Assembly of parallel programming” – but familiar and still useful in the short term for small numbers of cores
  - Need a way to transition smoothly from today’s programming to tomorrow’s

Tilera's Approach to Multicore Tools: Spatial Views and Collectives

Grid view
- Provides spatial view
- For selecting single process or region
- Eclipse based

Multicore Debugger
- GDB standard based -- familiar
- Aggregate control and state display
- Whole-application model for collective control
- Low skid breakpointing of all related processes

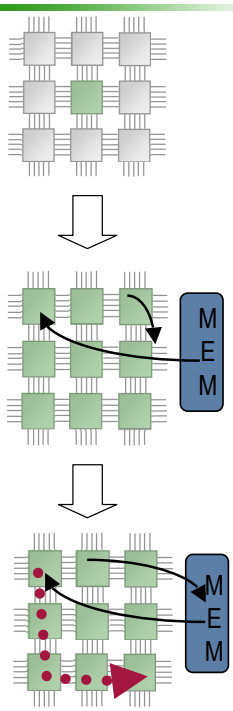
Multicore Profiler
- Collective stats
- Aggregate over selected tiles



Gentle Slope Programming™ Model

Gentle slope programming philosophy
- Facilitates immediate results using off-the-shelf code
- Incremental steps to reach performance goals

Three incremental steps
- Compile and run standard C applications on a single tile
- Run the program in parallel using standard SMP Linux models – pthreads or processes
- Use stream programming using iLib – a light-weight sockets-like API



High Performance in Small Form Factor
Example System Design

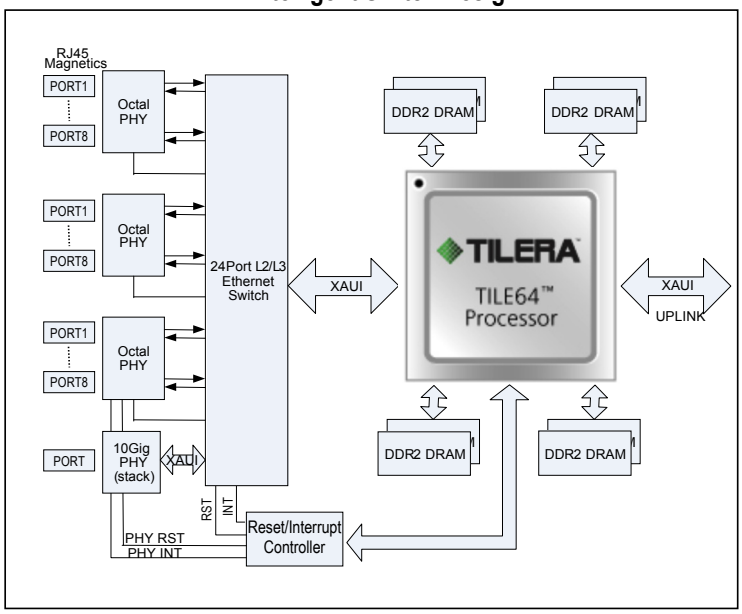
Market moving to intelligent network systems

- Growing need for in-line L4-L7 services
- Real-time protection against threats

Tile Processor enables

- Integrated in-line performance -- multiple apps at 1 to 10 Gbps of performance
- Glueless interface with leading switch vendors

Intelligent Switch Design



Tilera: World Class Company

- 64 people, proven veteran team
  - 150+ total tape-outs for revenue
  - 15 years average experience
- Proven leadership team
  - Over 150 years combined experience
  - 7 start-up companies founded
- 40+ patents pending
- Bessemer, Walden, Columbia, VTA (TSMC)
- Series B funding closed in February '07

Rich Heritage

DARPA MIT

Industry Involvement

The Multicore Association FSA

EETIMES **One of top 60 emerging startups**

RED HERRING **Named finalist for Red Herring 100**

Summary

- Current multicores face software and scalability challenges
- iMesh network based Tile Processor scales to many cores
- Gentle slope programming offers:
  - Convenience of SMP Linux/pthreads programming model
  - Performance scalability through streaming channels
- TILE64 silicon, software tools, and applications deployed in customers' systems

Additional Information

PSNR: Peak signal to noise ratio
MDN: Memory dynamic network
UDN: User dynamic network
TDN: Tile dynamic network
IDN: I/O dynamic network
STN: Static network