Many-core systems and their challenges for OS

Klaus Danne

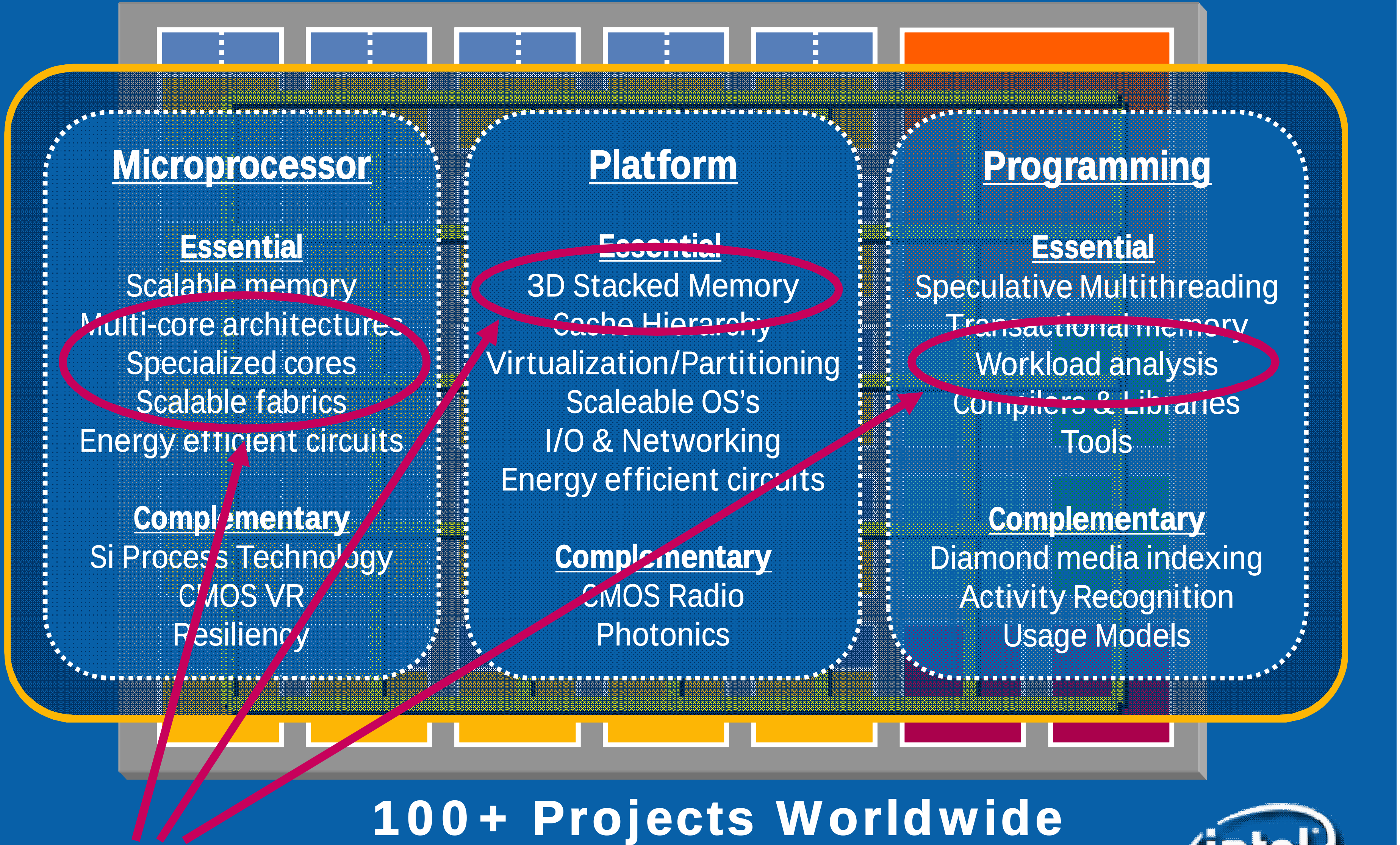
12. November 2009

Herbsttreffen
Gesellschaft für Informatik
Fachgruppe Betriebssysteme

Intel Braunschweig

- Since 2000 (Intel acquired Giga, HC 30)
- Today, HC ~100
  - ~50 in Intel Labs
  - ~50 in product groups
- Intel Germany Research Center
  - German part of Intel Labs
  - Responsible for Research and Technology Development
  - Engage & collaborate with Academia
  - Est. 2005
  - Intel's largest European Research site

Tera-Scale Computing Research



Microprocessor

Essential
Scalable memory
Multi-core architectures
Specialized cores
Scalable fabrics
Energy efficient circuits

Complementary
Si Process Technology
CMOS VR
Resiliency

Platform

Essential
3D Stacked Memory
Cache Hierarchy
Virtualization/Partitioning
Scaleable OS's
I/O & Networking
Energy efficient circuits

Complementary
CMOS Radio
Photonics

Programming

Essential
Speculative Multithreading
Transactional memory
Workload analysis
Compilers & Libraries
Tools

Complementary
Diamond media indexing
Activity Recognition
Usage Models

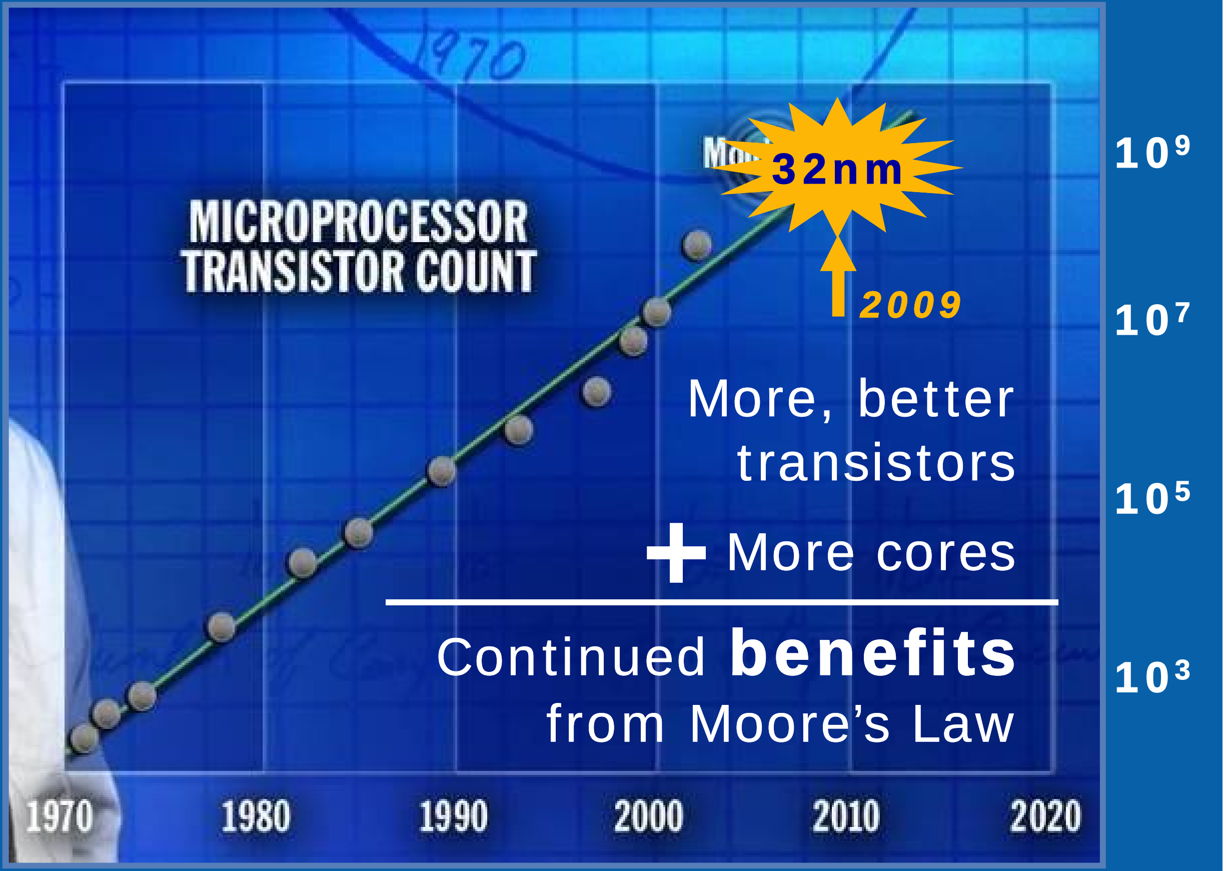
100+ Projects Worldwide

IGRC Focus

Outline

- Microprocessor power challenge
  - Many-cores power advantage
  - Specialized cores
  - Heterogeneous systems
- OS challenges
  - Scheduling to heterogeneous system
  - Power management

Moore’s Law Motivates Multi-Core



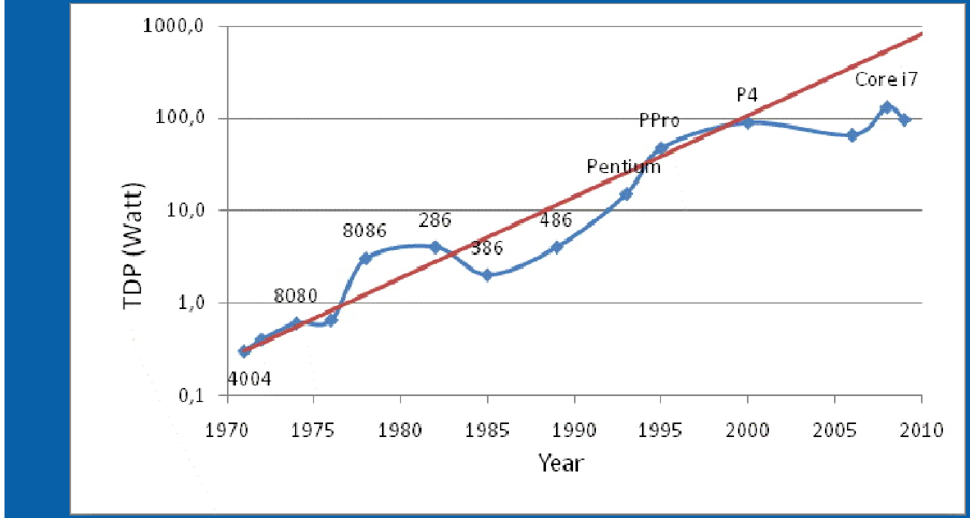
More, better transistors

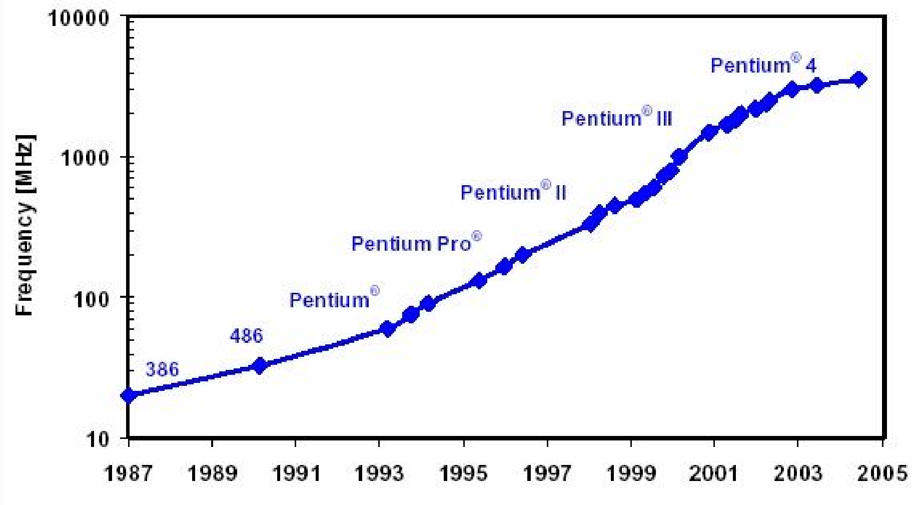
+ More cores

Continued **benefits** from Moore’s Law

Power Limitation

- Max power envelope is limited (by cost)
- End of frequency paradigm
  - Power is linearly related to frequency with no voltage scaling
  - Power is cubically related to frequency and voltage scaling
  - Performance is not linearly related to frequency



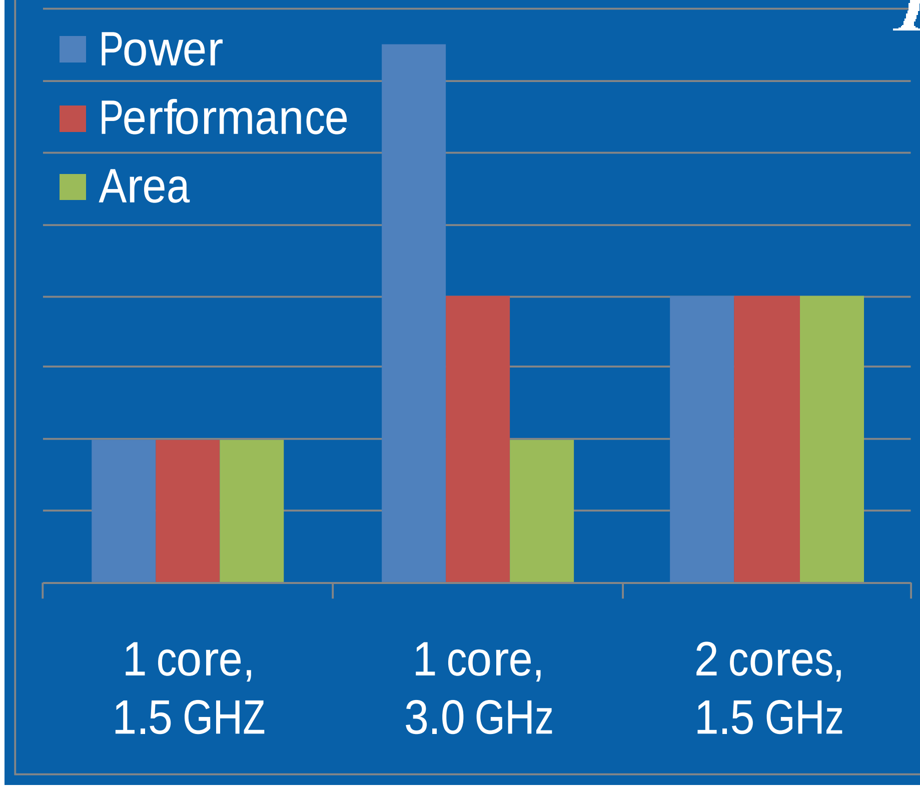


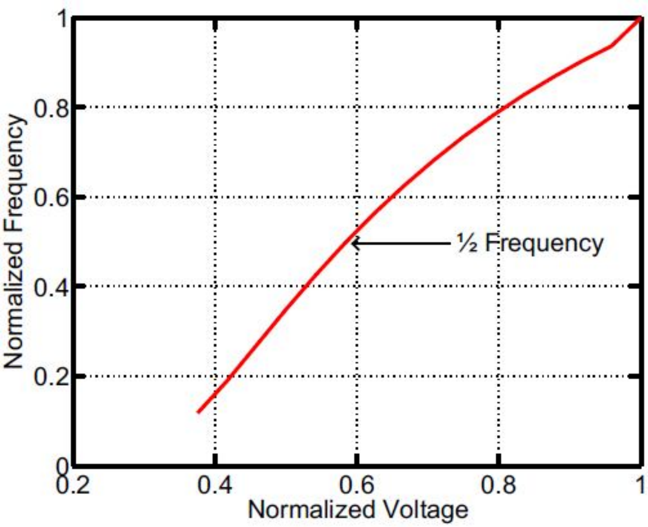
[src: F. Pollak MICRO99 and www.intel.com/procducts/processors/]

Multi core power advantage

- Lower frequency reduces power over proportionally
- 2 slow cores can deliver same performance as 1 fast core at less power, same architecture & technology

$$P = k \cdot C \cdot V^2 \cdot f + a \cdot V \cdot I_q$$

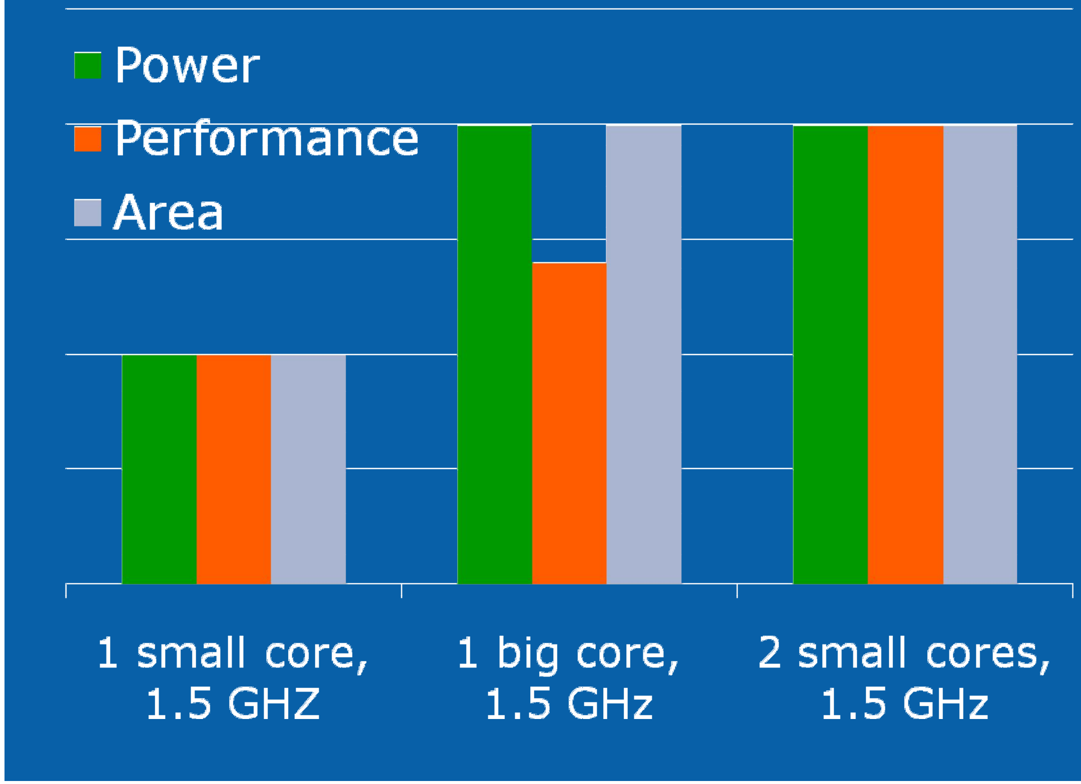




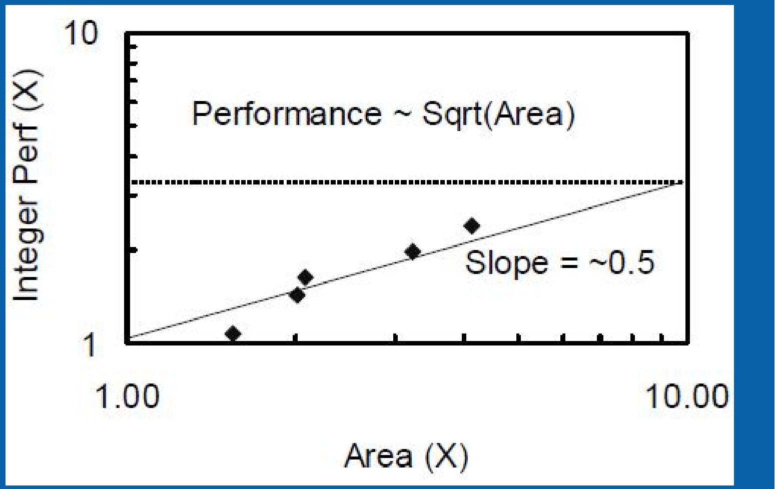
[src: Rangan, ISCA07]

Multi core power advantage

- Smaller cores reduce power over proportionally
- 2 small cores can deliver more performance as 1 complex core at same power, area, frequency, technology



Pollack’s Rule



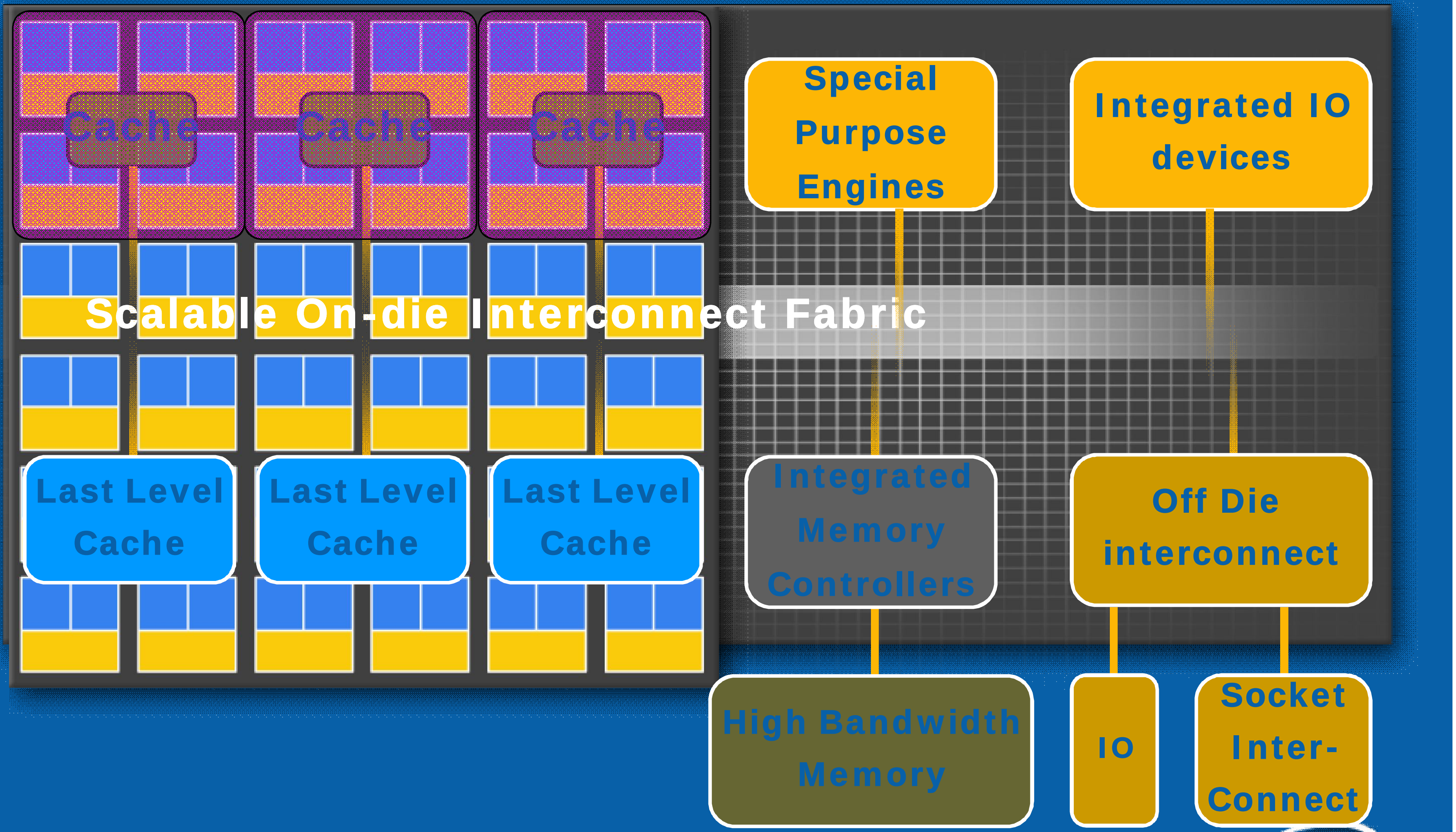
[src: S. Borkar DAC07, F. Pollak MICRO99]

Multi-core challenges

- Interconnect and communication
  - power overhead
  - communication overhead
  - -> efficient network to connect many cores

- Parallel software
  - Legacy software is often single threaded
  - Amdahl's law limits speed up by parallelization
    - ...but many future workloads have parallel nature
  - Parallel programming challenges

A Tera-scale Platform Vision



Platform power management

Power constraint => can’t run all cores at full speed

- Some power management done at device level
  - DVFS (e.g. Turbo Boost Technology)
  - Fast control loops; avoid permanent damage
- Today: processors offers ACPI interface to OS
  - P-state (voltage/frequency pair) and C-state
  - P-state may be not per core but per cluster
- Tomorrow: more advanced interfaces may be required to measure TDP and control performance

Fine Grain Power Management

example of 80 tile 65nm research chip

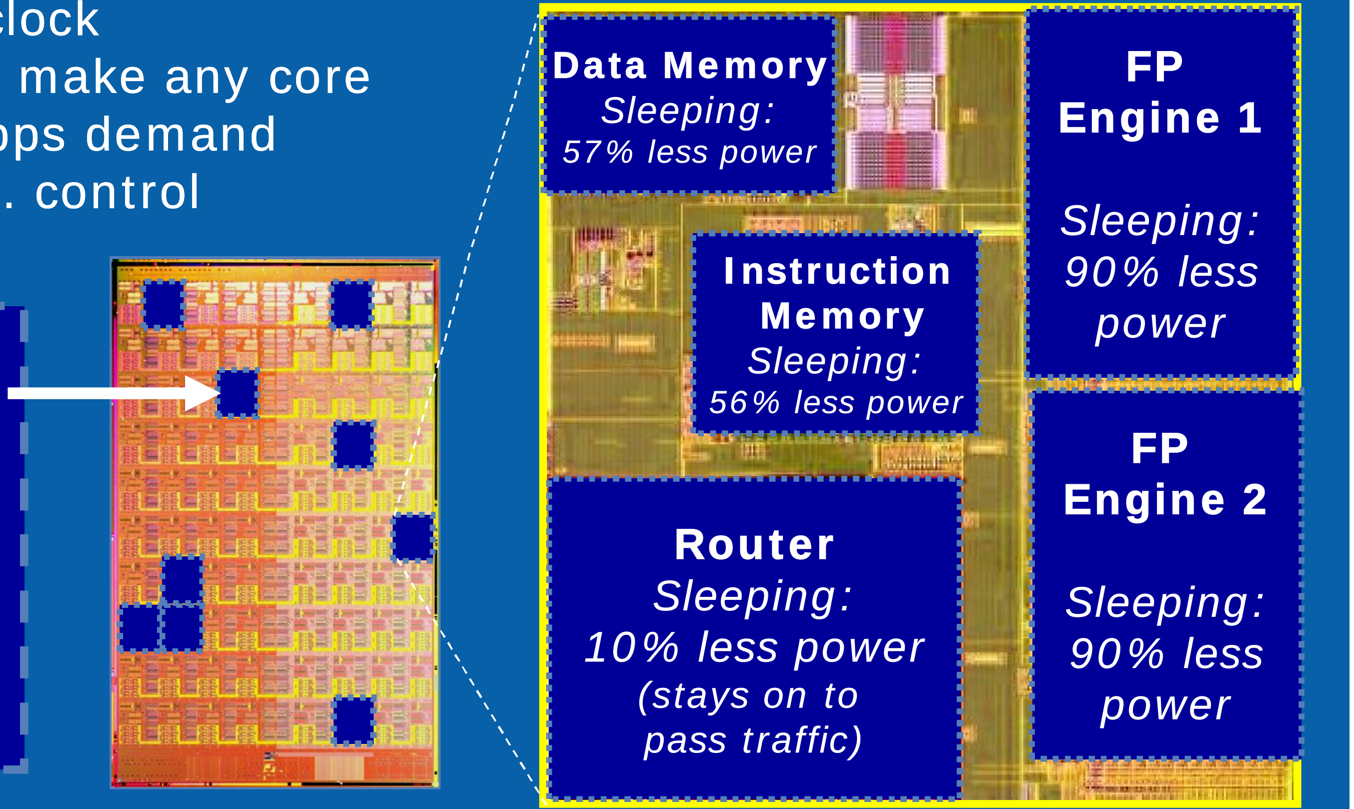
- Novel, modular clocking scheme saves power over global clock
- New instructions to make any core sleep or wake as apps demand
- Chip Voltage & freq. control
  (0.7-1.3V, 0-5.8GHz)

Dynamic sleep

STANDBY:
- Memory retains data
- **50%** less power/tile

FULL SLEEP:
- Memories fully off
- **80%** less power/tile

21 sleep regions per tile *(not all shown)*



Data Memory
Sleeping:
57% less power

Instruction Memory
Sleeping:
56% less power

Router
Sleeping:
10% less power
(stays on to pass traffic)

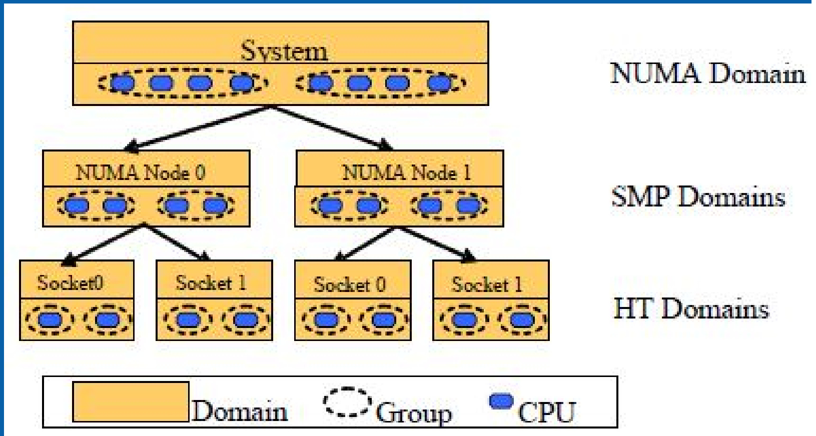
FP Engine 1
Sleeping:
90% less power

FP Engine 2
Sleeping:
90% less power

Industry leading energy-efficiency of 16 Gigaflops/Watt

OS challenges – topology

- Scheduling in heterogeneous systems
  - NUMA, SMP, SMT
- Asymmetric multi core
  - Performance
  - ISA extensions



[src: Siddha et. el., ITJ07]

[Li07] Efficient OS Scheduling for Performance Asymmetric Multi-Cores

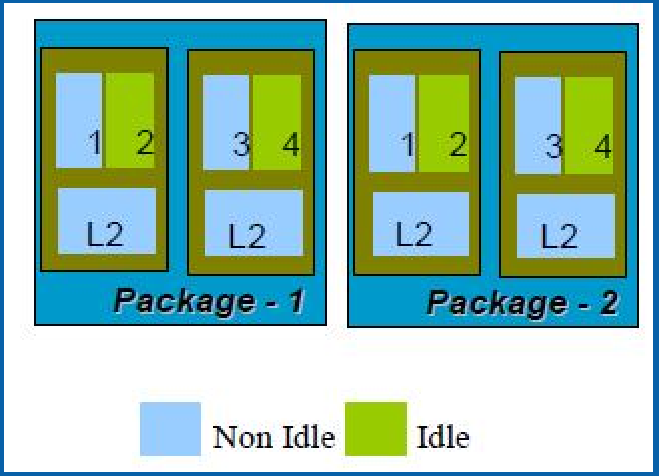
[Li08] OS Support for Shared-ISA Asymmetric Multi-Cores

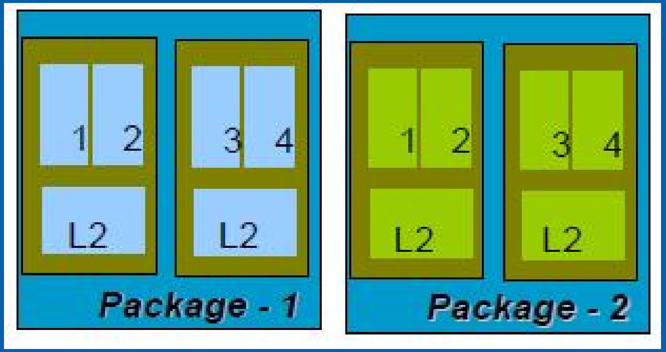
OS challenges – power management

- Power constraint => can’t run all cores at full speed
- OS should optimize for performance and power
- Platform characteristics:
  - DFVS performance states, processor power states
    - Penalty of transition
  - Asymmetric power efficiency of cores
  - Dynamic-,leakage-power, temperature
- Application characteristics:
  - Application threads’ demands, dependencies
  - CPU-, MEM-, I/O-bound
  - Observe at runtime

OS power aware example

- 2 package SMP platform with Intel Core2 quad processors
- 4 tasks
- Assignment strategies:
  1. Different L2 caches
     - high performance
  2. Same package
     - low power
- Workload dependent
- Similar problem for multi-core on die

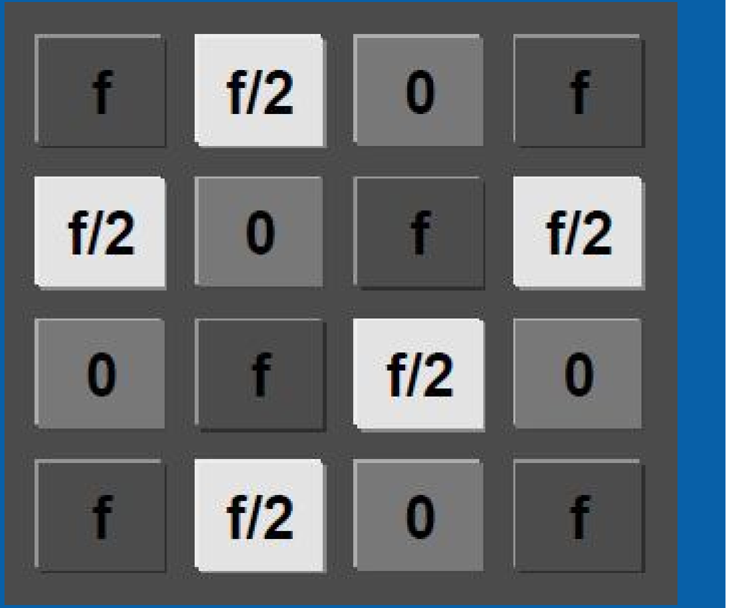




[src: Siddha et. el., ITJ07]

Two-frequency approach

- Many core platform supporting per core:
  - f = full speed
  - f/2, at ~25% of power
  - off, saving leakage
- Just 2 voltage levels
- Simple synchronous interfaces

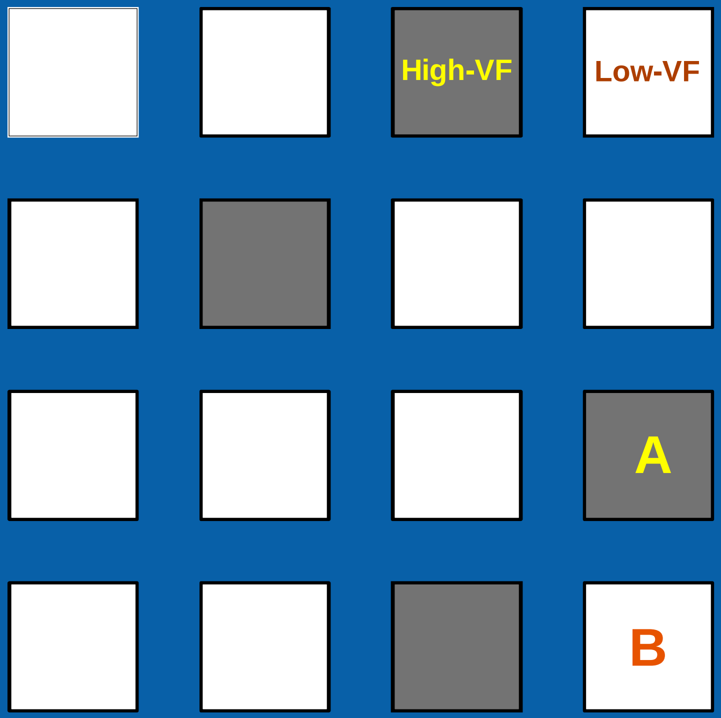


f	f/2	0	f
f/2	0	f	f/2
0	f	f/2	0
f	f/2	0	f

[src: S. Borkar DAC07]

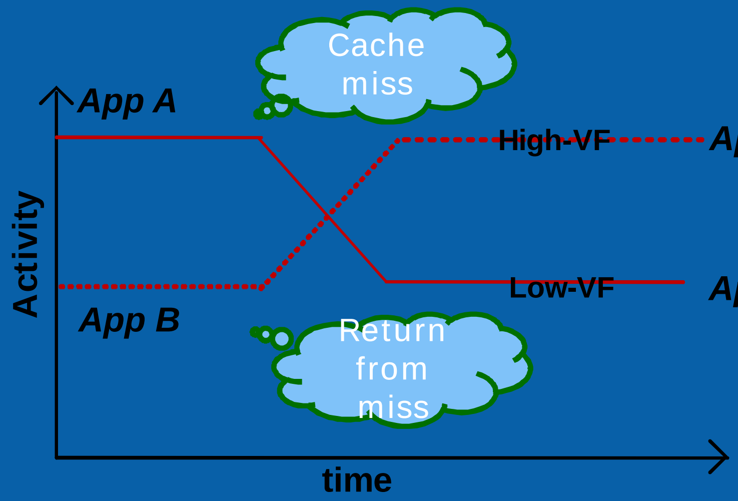
- How will OS optimal select settings?
- Further HW support desired?

Thread migration approach



[src: Rangan et. al., ISCA07]

Exploiting fine-grained application variability



Frequent thread movement with 2-VF:

- High-IPC applications spend more time on high-VF core
- Low-IPC applications spend less time on high-VF core

[Rangan et. al., ISCA07] *Thread Motion: Fine-Grained Power Management for Multi Core Systems*

[Chaparro et. al., TPDS07] *Understanding the Thermal Implications of Multicore Architectures*

Summary

- Future devices will likely be
  - power constraint (not all cores at full speed)
  - asymmetric (big, small, special cores)
- OS should consider
  - Platform topology/asymmetry
  - Power-budget, -characteristics, -efficiency
  - Application behavior, needs, dependencies

intel®

Leap ahead™