MULTICORE DESIGN SIMPLIFIED

imperas

System-Level Automation Tools for MPSoC Designs

Peter Flake

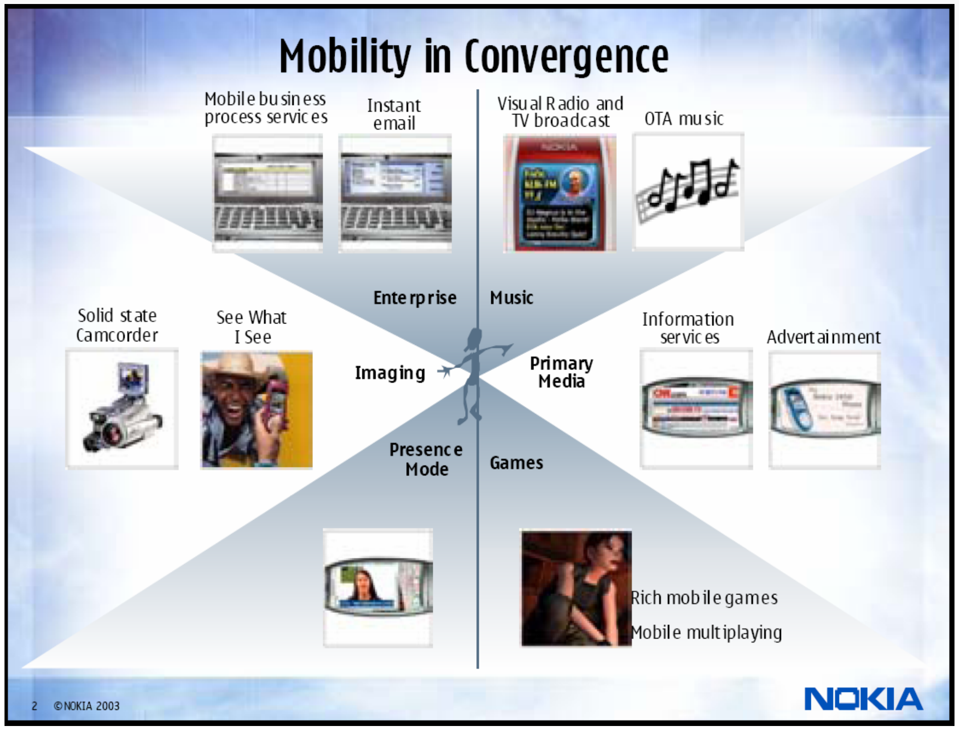
August 14th 2006

Agenda

- **Introduction**
- Trends and Challenges
- Exploration Tools
- Requirements for MPSoC Design
- Solution Providers
- Conclusion

It’s all your fault …

- End consumer is very demanding …
- Convergence
  - More performance
  - Less power
  - Lower cost
  - Shortest time to market
- Skyrocketing chip development cost
  - Flexibility
  - Longest time in market



Source: Anssi Vanjoki
Executive Vice President and General Manager
Nokia
Nokia Capital Market Days

… today's methods are running out of steam!

MULTICORE DESIGN SIMPLIFIED

- The complex devices of the future will consist of heterogeneous parallel processor frameworks executing huge software applications.
- New key technologies and methodologies are required to automate and streamline Multi-Core IC design & programming

MULTICORE DESIGN SIMPLIFIED imperas

Agenda

- Introduction
- **Trends and Challenges**
- Exploration Tools
- Requirements for MPSoC Design
- Solution Providers
- Conclusion

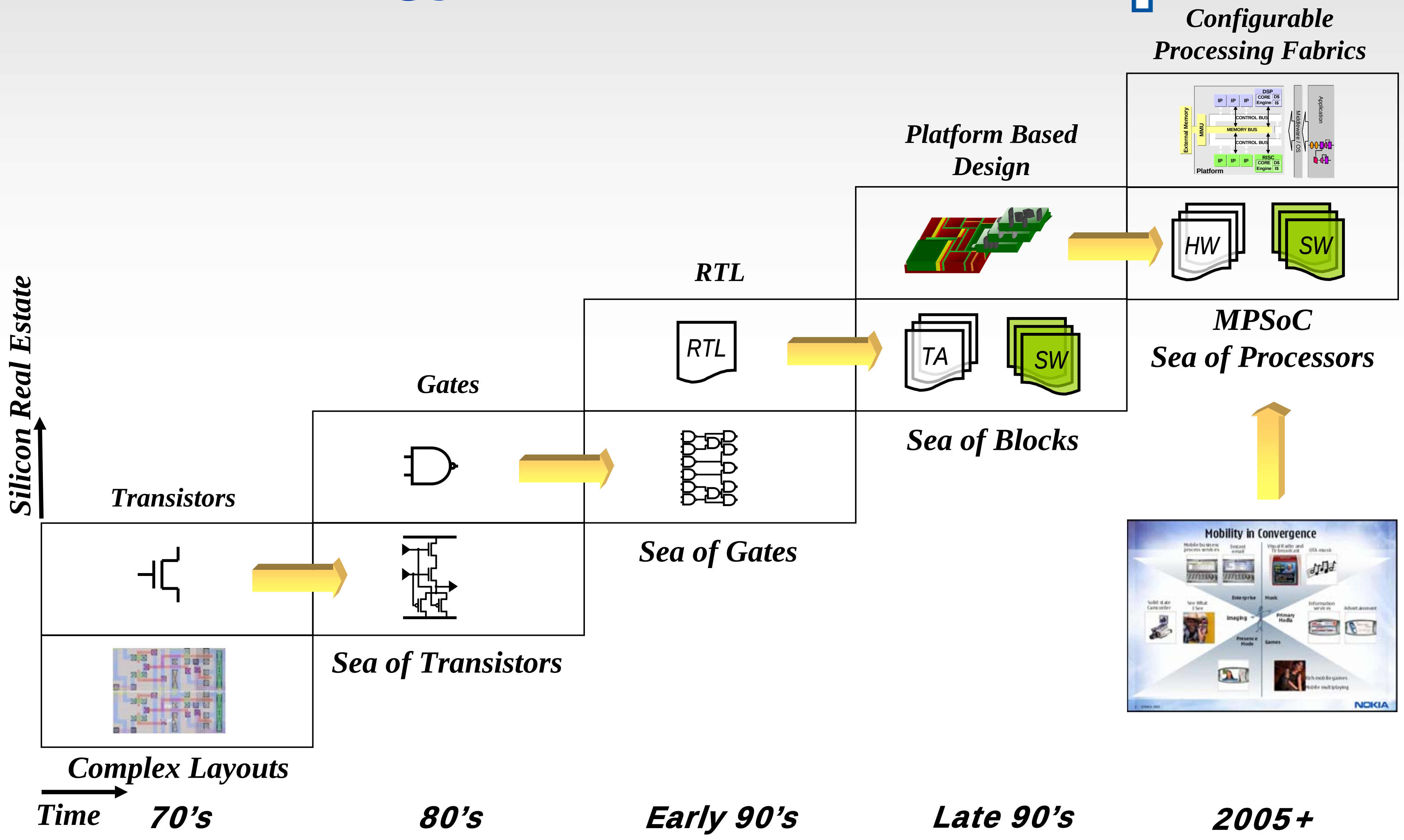
Observations….

- Prof. Kurt Keutzer, Berkeley

  - The **ad hoc approach** to SoC design simply **cannot scale with Moore's Law** because it does not sufficiently reduce the complexity of SoC design

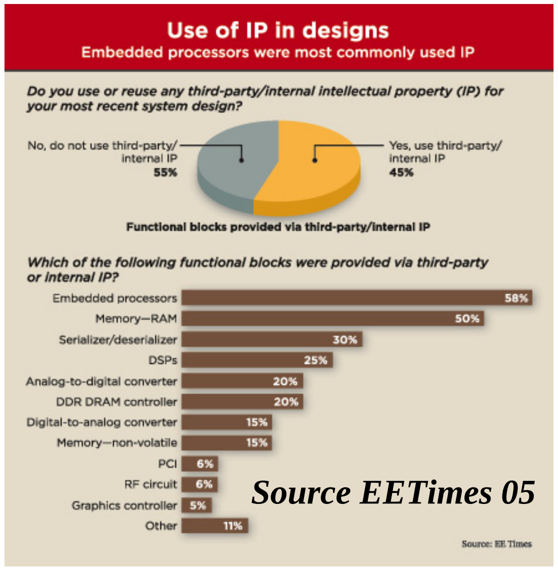
  - The "**software-development environment as afterthought**" era of IC design is rapidly **drawing to a close**

Methodology Evolution



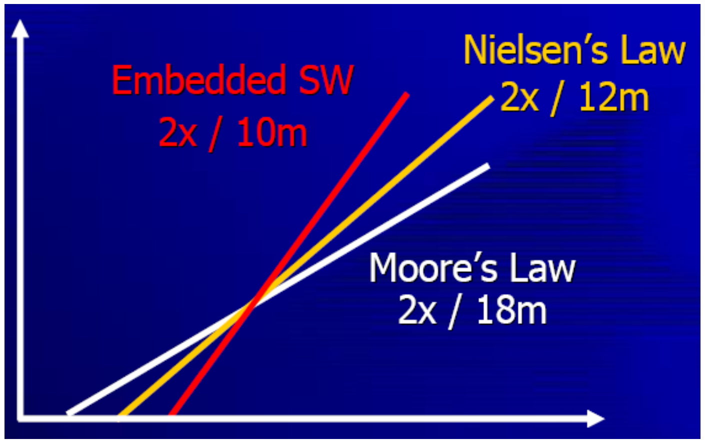
The Multi-core IC Trend to MPSoCs

Multi-Core IC usage is rapidly increasing, and will take over as the dominant method of executing large design projects efficiently



Embedded SW increasing - Doubling annually

Source: ITRS 05

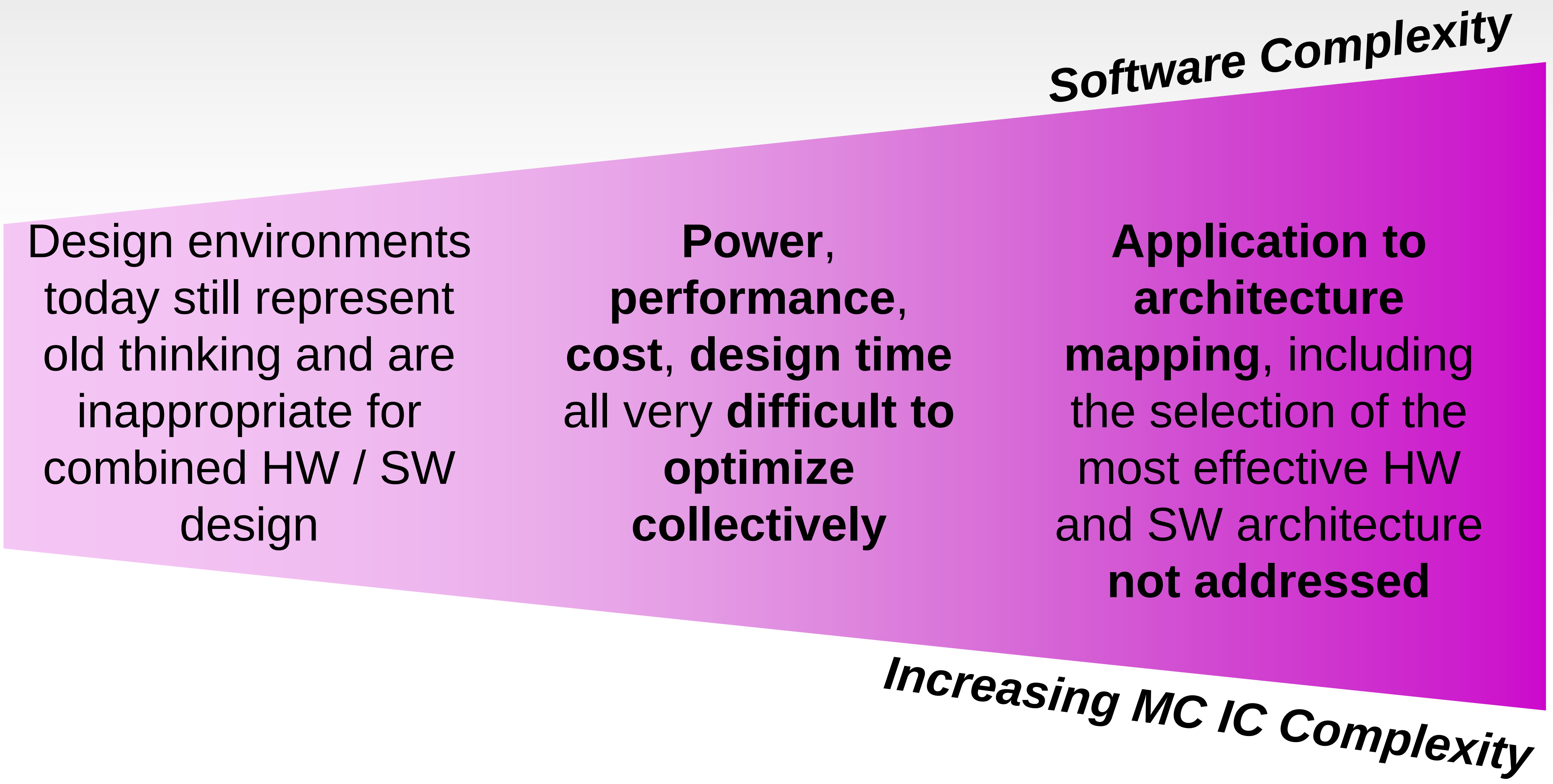


Dataquest: Use of processor based platforms growing 8-10% CAGR

Collett: ">60% of designs now contain more than one processor"

The MPSoC Development Challenges

MULTICORE DESIGN SIMPLIFIED imperas



Design environments today still represent old thinking and are inappropriate for combined HW / SW design

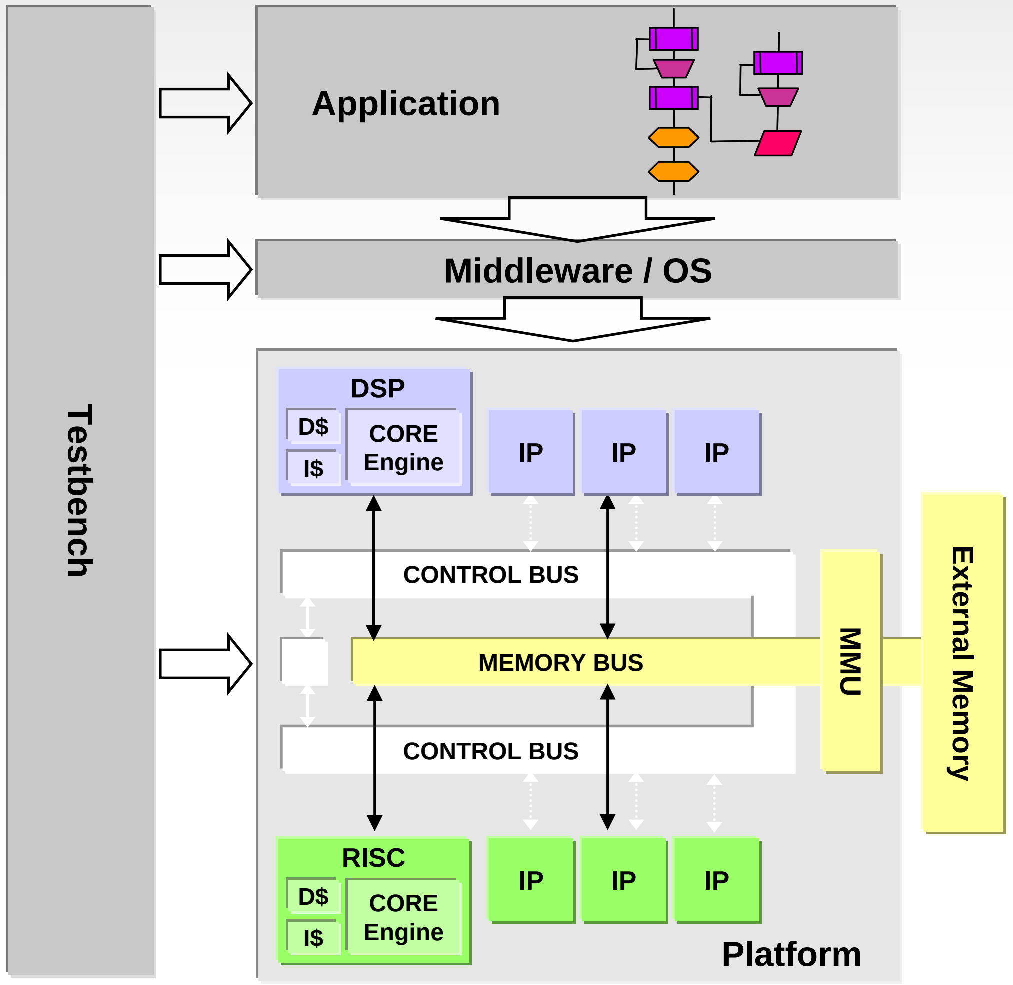
Power, **performance**, **cost**, **design time** all very **difficult to optimize collectively**

Application to architecture mapping, including the selection of the most effective HW and SW architecture **not addressed**

Agenda

- Introduction
- Trends and Challenges
- **Exploration Tools**
- Requirements for MPSoC Design
- Solution Providers
- Conclusion

Requirements for MPSoC Design



- How do I program it and express parallelism?
- How do I simulate this at reasonable speeds?
- How do I debug this?
- How do I optimize the software?
- How do I optimize across hardware and software?
- How do I deliver it to my software users?

Requirements: Programming

MULTICORE DESIGN SIMPLIFIED
imperas

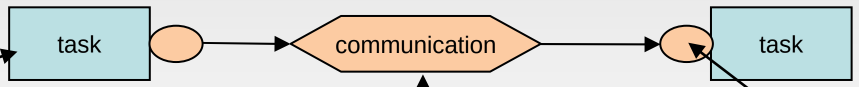
Today

- Various languages
  - Do not express parallelism
  - Often limited to specific application domains
- Several incompatible programming models used for special applications
  - OpenMP
  - YAPI
  - DSOC
  - SMP
  - xUML
  - …

MPSoC Requirements

- Appropriate Programming Models
  - Task level parallelism
  - Flexibility
  - Efficiency

One Approach ...



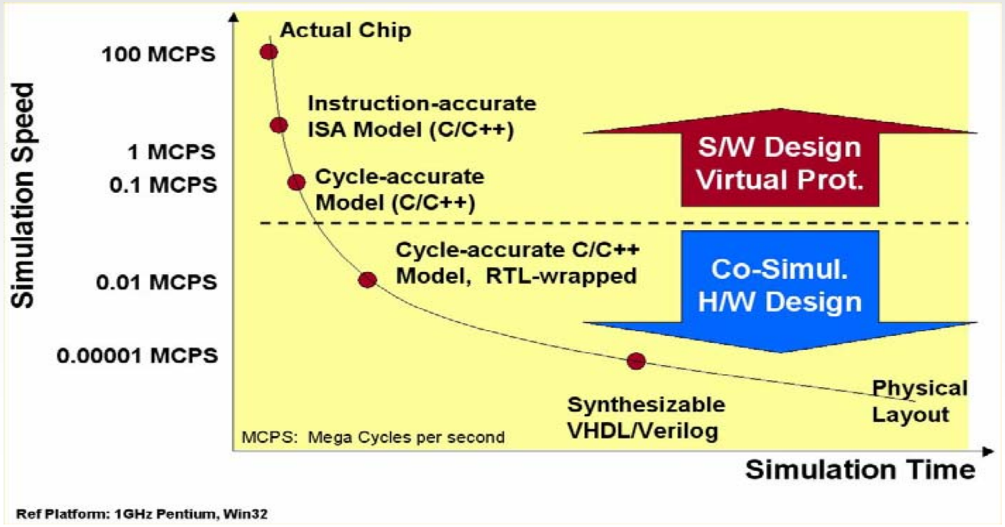
Tasks are units of work in the application

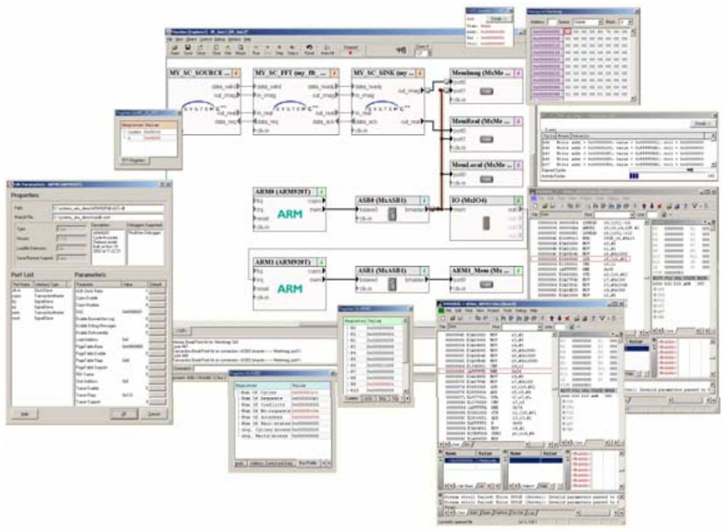
Tasks communicate through different mechanisms over different channel types

Tasks talk connect to channels via ports

- Communication structure is separated from tasks
  - Coordination language
- Various modes of communication can be supported
  - blocking, non-blocking
- Communication can be implemented in various ways
  - Depending on the platform

Requirements: Debug & Simulation





Sources: ARM IQ Magazine

- Today, simulation speed is limiting
- Need faster simulation
  - Enabling trade offs
  - Flexibility: appropriate accuracy at appropriate speed

- Today, single core debugging approaches don't scale to MPSoC
- Need true multi processor debug
  - Focused on threads
  - Scaling to 10+ processors

Today’s Approaches with SystemC

MULTICORE DESIGN SIMPLIFIED
imperas

SW development

- Run application code compiled for host
  - Fast
  - Not instruction accurate
  - May give different results
- Model peripherals and communication in SystemC
  - Special OS code
  - Not timing accurate
  - Performance bottleneck

SW verification

- Run application code on ISS
  - Slow
  - Instruction accurate or cycle accurate
  - May use vendor debugger
- Wrap ISS in SystemC
  - Memory inside or outside
  - Speed or accuracy
- Model peripherals and communication in SystemC
  - OS can run on ISS

Other Approaches …

MULTICORE DESIGN SIMPLIFIED imperas

Code Morphing

- Run application code compiled for ISS but translated into host instructions
  - Fast
  - Instruction accurate
  - May use vendor debugger

Hardware

- FPGA Development systems
  - Fast
  - Late in the flow … a fair amount of implementation has to be done
- Emulation
  - Pretty fast …
  - Sometimes painful to set up (order of weeks)
  - Also late in the flow

Requirements: Software & Automation

MULTICORE DESIGN SIMPLIFIED imperas

Today

- Limited SW support
  - SystemC models slow for SW developers.
  - Models not well verified
  - Debugger integration poor
- Focus on Analysis
  - “here you go … now fix it yourself manually and re-simulate”

MPSoC Requirements

- True HW/SW Interaction
  - Higher levels of speed/accuracy trade-off
  - Easily verifiable models
- True HW/SW Automation
  - SW Mapping & Optimization
  - HW/SW Optimization
  - “here is the solution for your power/performance objectives”

MULTICORE DESIGN SIMPLIFIED imperas

Agenda

- Introduction
- Trends and Challenges
- Exploration Tools
- **Requirements for MPSoC Design**
- Solution Providers
- Conclusion

Different users have different needs …

MULTICORE DESIGN SIMPLIFIED
imperas

Platform Designer

- Ensure that selected applications can be run at required performance and efficiency
- Optimize platform architecture

- Programming of compute intensive portions of application
- Efficient modeling of platform options

Platform User

- Try new platform
- Add new applications to existing platform
- Check performance and power constraints
- Find optimal SW to HW mapping
- Optimize hardware parameters

- Platform independent IDE
- Platform models
- Exploration of various SW partitioning options

… which are not met today!

Today

- Lots of individual single and fixed core offerings
- Parallelism, configurability and multiplicity of processing not appropriately addressed

- *“[…] the design of complex embedded systems with multiple configurable, extensible processors demands new ESL tool capabilities that go well beyond current offerings!”*

Grant Martin
Chief Scientist
Tensilica

MPSoC Requirements

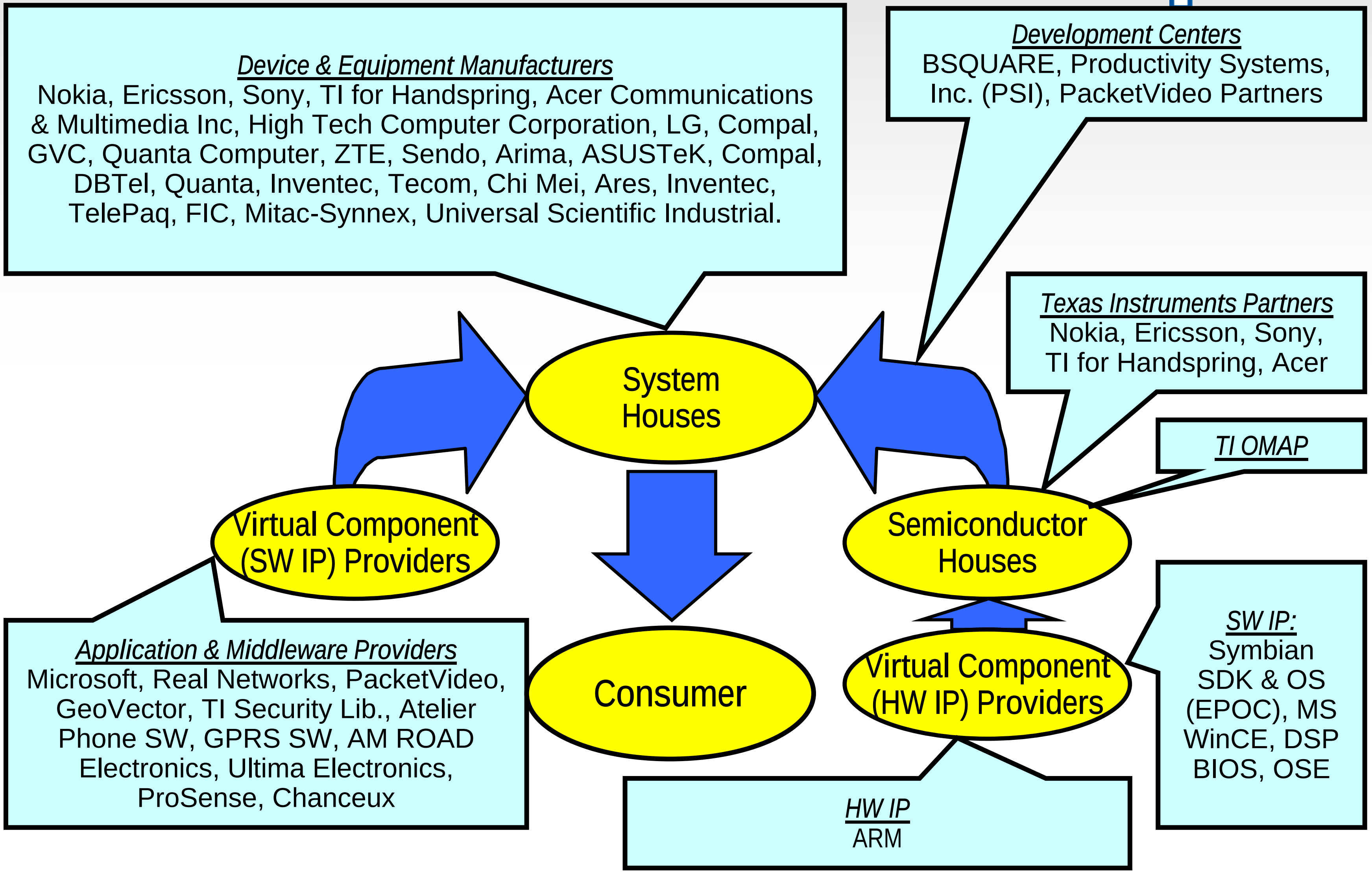
- Solutions with parallelism and multiplicity of processing in mind
  - Compilation
  - Simulation
  - Debug
  - Programming
  - SW/SW Optimization
  - SW/HW Optimization
- True System Design Automation across hardware and software

Agenda

- Introduction
- Trends and Challenges
- Exploration Tools
- Requirements for MPSoC Design
- **Solution Providers**
- Conclusion

Platform Eco-System: e.g. TI OMAP

MULTICORE DESIGN SIMPLIFIED
imperas



Device & Equipment Manufacturers
Nokia, Ericsson, Sony, TI for Handspring, Acer Communications & Multimedia Inc, High Tech Computer Corporation, LG, Compal, GVC, Quanta Computer, ZTE, Sendo, Arima, ASUSTeK, Compal, DBTel, Quanta, Inventec, Tecom, Chi Mei, Ares, Inventec, TelePaq, FIC, Mitac-Synnex, Universal Scientific Industrial.
Development Centers
BSQUARE, Productivity Systems, Inc. (PSI), PacketVideo Partners
Texas Instruments Partners
Nokia, Ericsson, Sony, TI for Handspring, Acer
TI OMAP
System Houses
Virtual Component (SW IP) Providers
Semiconductor Houses
Consumer
Virtual Component (HW IP) Providers
Application & Middleware Providers
Microsoft, Real Networks, PacketVideo, GeoVector, TI Security Lib., Atelier Phone SW, GPRS SW, AM ROAD Electronics, Ultima Electronics, ProSense, Chanceux
SW IP:
Symbian SDK & OS (EPOC), MS WinCE, DSP BIOS, OSE
HW IP
ARM

Source: IEEE Computer

Who can provide Solutions?

MULTICORE DESIGN SIMPLIFIED
imperas

- Current tools provided by different parts of the Eco-System
  - Not well integrated
- Next generation System Design Automation tools
  - Will be provided by specialist suppliers
  - Close cooperation with hardware and software designers required
  - Will probably have to be funded by the hardware world
    - Software developers expect a state of the art software development environment supporting the platform
    - Otherwise they will simply switch platforms or remap the application

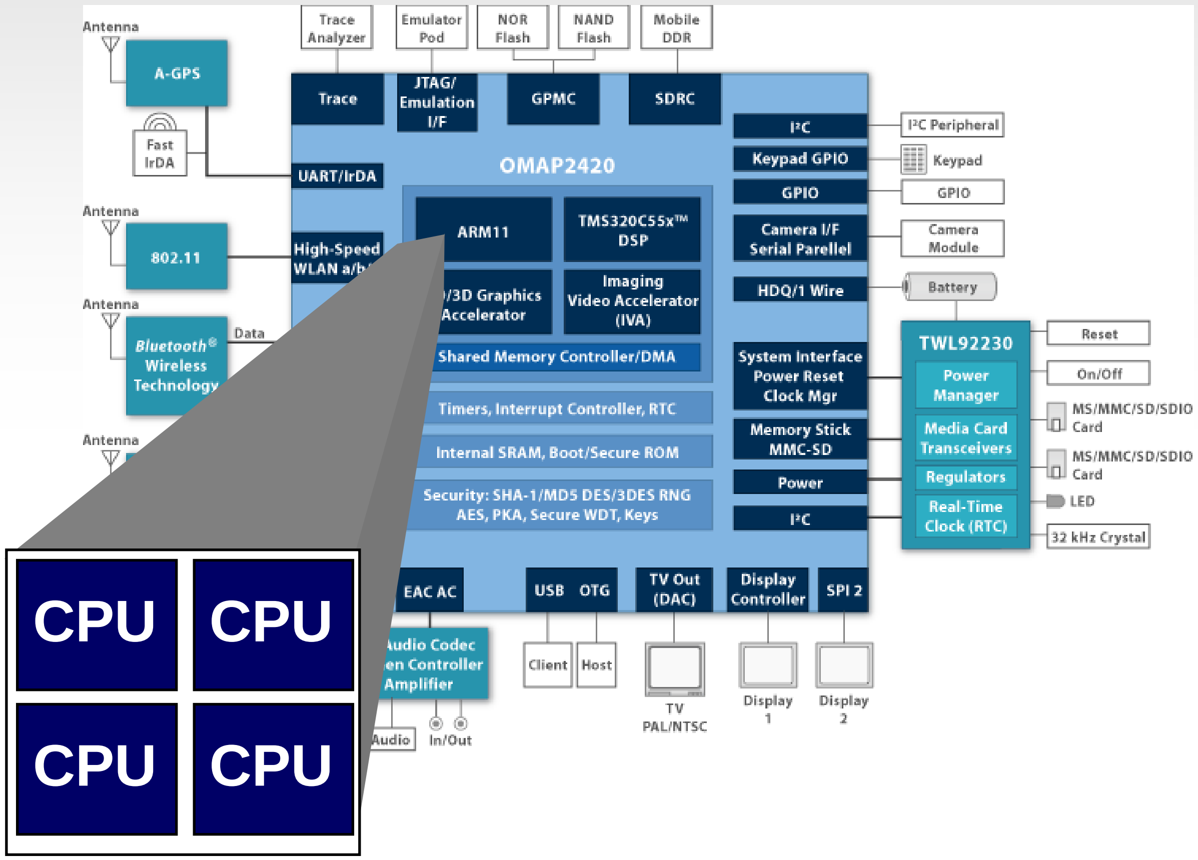
➢ New MPSoC Methodology will be supported

Agenda

- Introduction
- Trends and Challenges
- Exploration Tools
- Requirements for MPSoC Design
- Solution Providers
- **Conclusion**

The Future of IC design: MPSoC

Processor performance growth through improved technology is becoming **exhausted**, so the next phase is **<u>multi-core</u> to provide additional processing capability**



<u>Discontinuity:</u>

How will these devices be programmed?

Can all the device developers provide good programming tools?