Energy Aware Network Computing:

Packet Processing with Multicore Processors

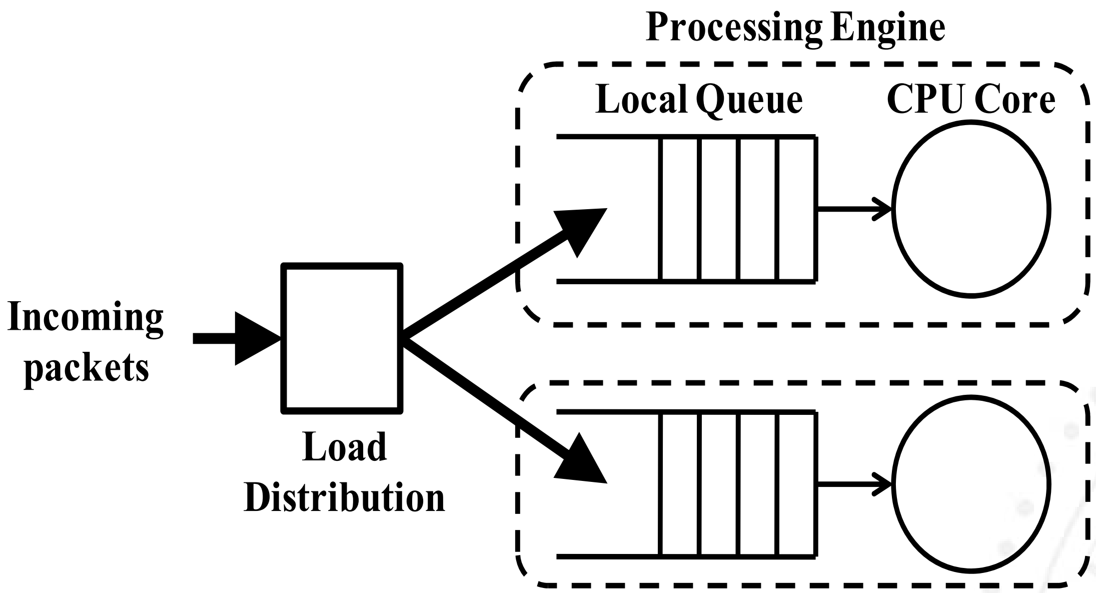
Laxmi N. Bhuyan

Computer Science and Engineering

University of California, Riverside

http://www.cs.ucr.edu/~bhuyan

What is Network Computing?

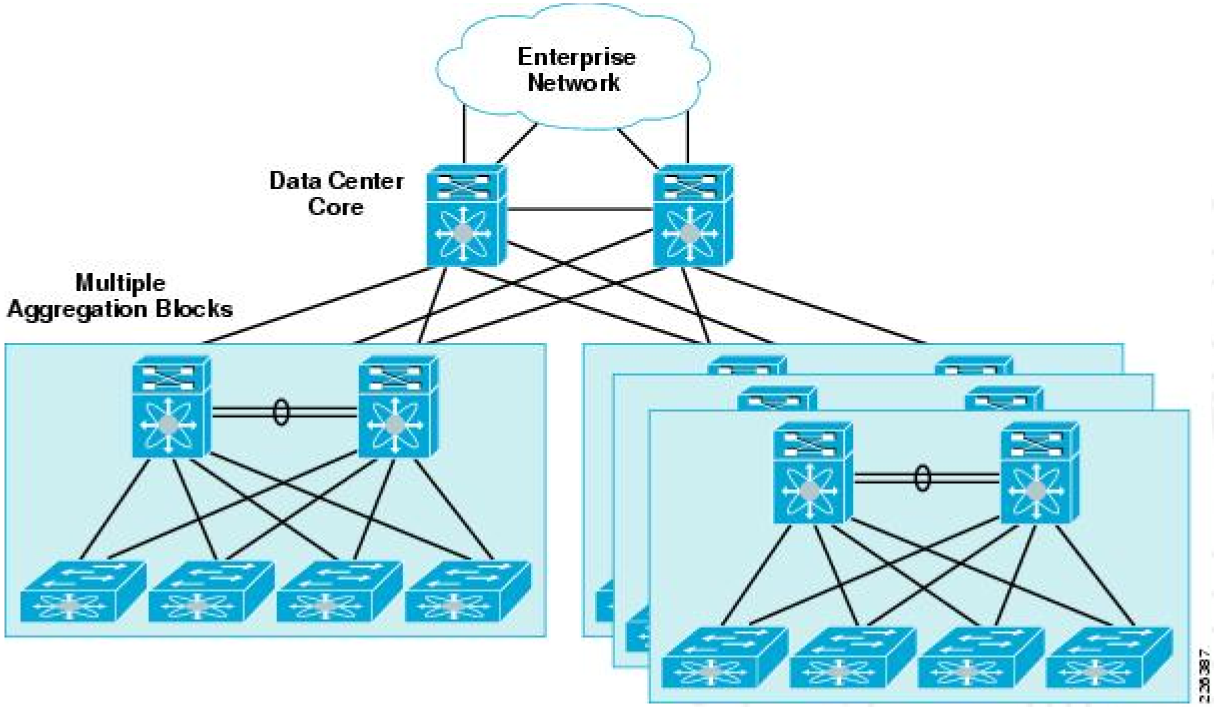


Examples:

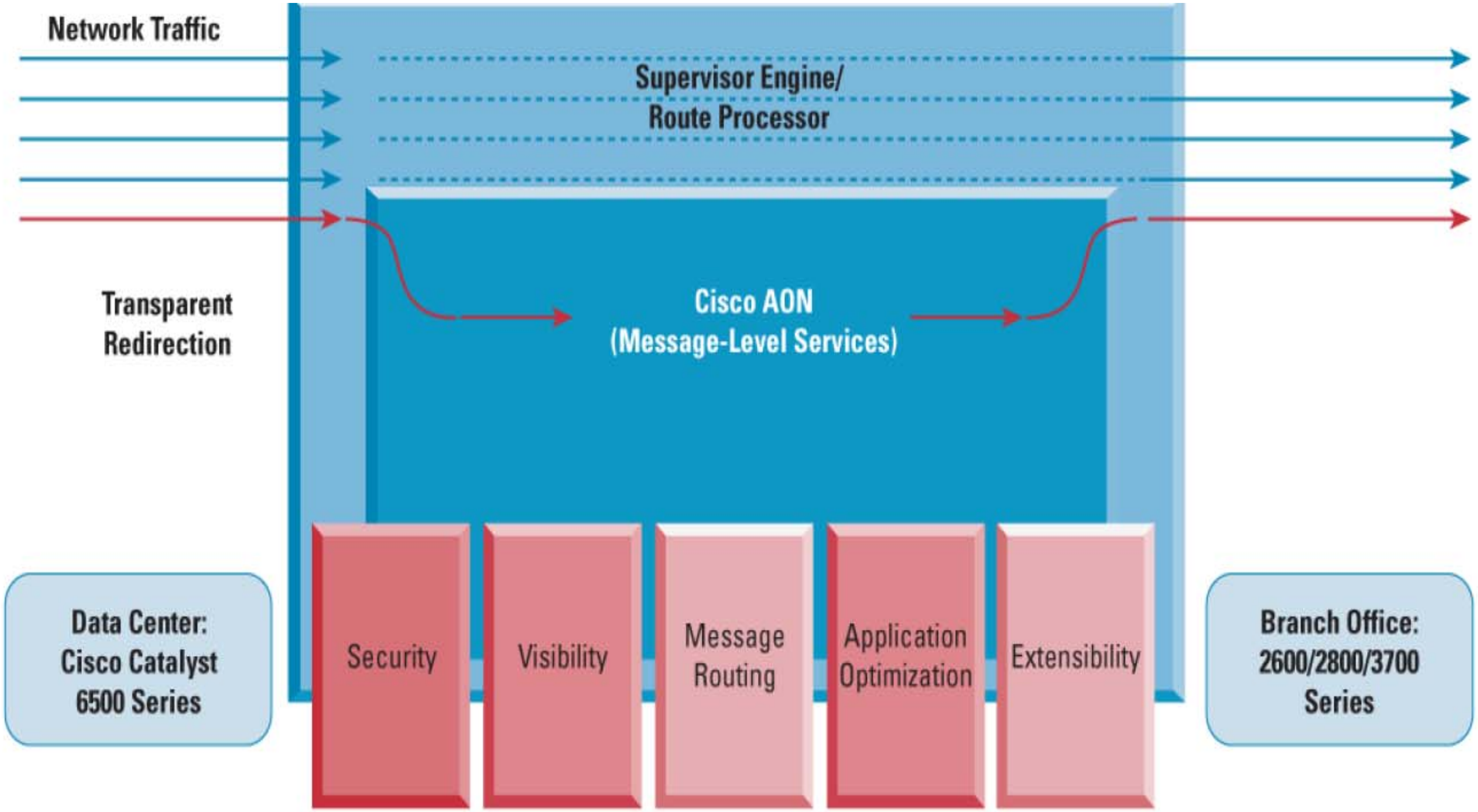
Web Servers,
Data Centers,
Routers,
Interfaces, AON,
Web Servers,
Multimedia
Servers

Computation Workload driven by request arrivals from the network, short executions and departures

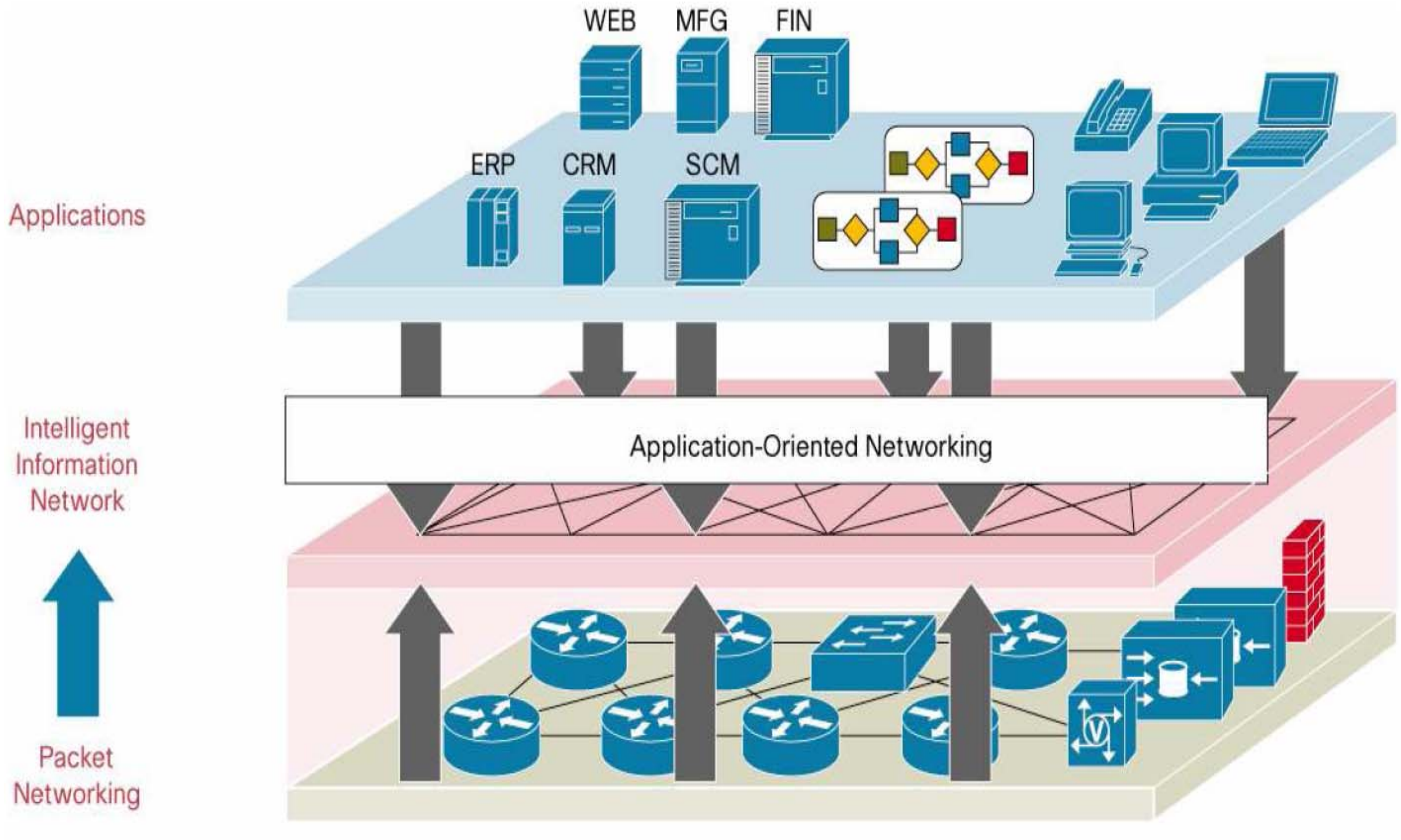
Data Center


Enterprise Network
Data Center Core
Multiple Aggregation Blocks
226387

Redirecting Traffic to Cisco AON Module

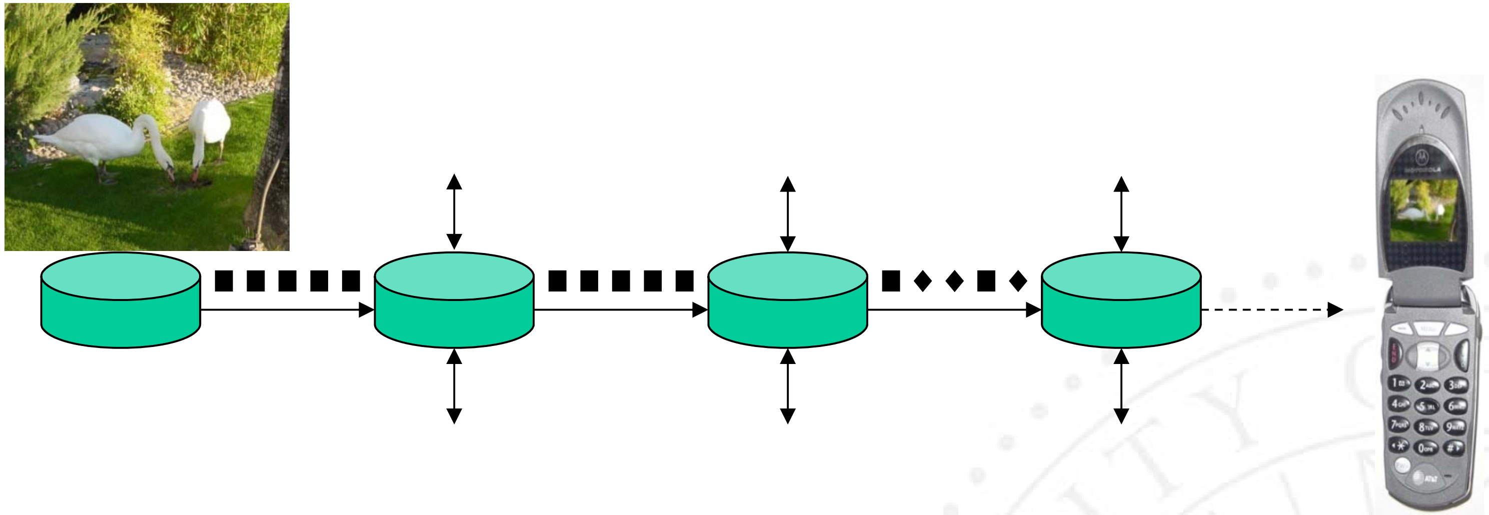


Let the network speak the language of applications! – Vertical processing – A change in networking paradigm


WEB
MFG
FIN
ERP
CRM
SCM
Applications
Intelligent Information Network
Application-Oriented Networking
Packet Networking

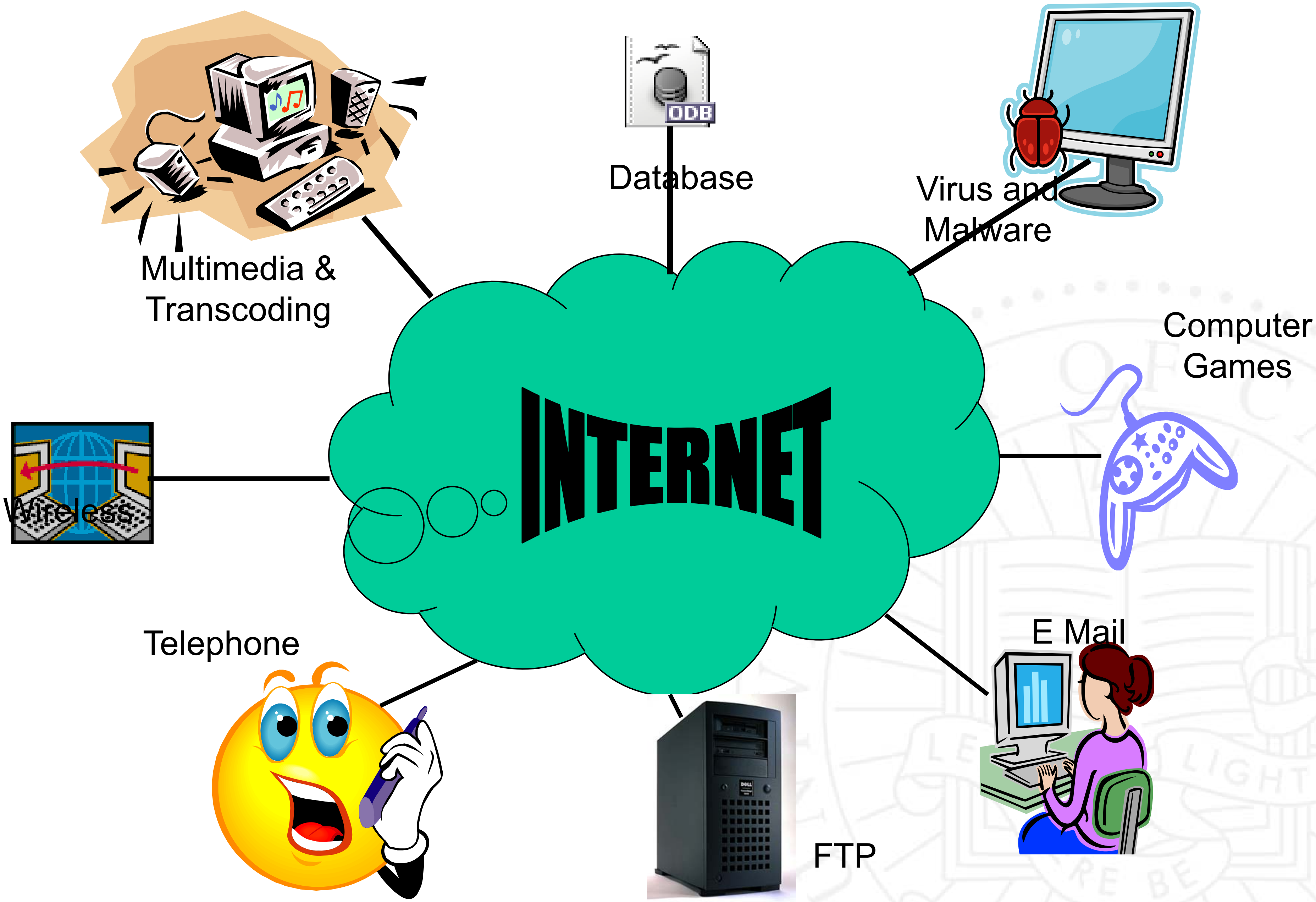
Courtesy:
http://www.cisco.com/en/US/products/ps6438/products_white_paper0900aecd8033e9a4.shtml

A Multimedia Active Router in the Network



- A large number of clients, Heterogeneity in clients' inbound network bandwidth, CPU / MEM capacity or display resolution
- A video server has to keep different versions of the video and send the appropriate one for the client
- Gives rise to overloading of the video server, reliability and bandwidth problem of the network
- Why not send the same video to all clients, but convert it in the router or base station nearest to the client? => Multimedia Transcoding

Courtesy "A Cluster-based Active Router Architecture", G. Welling, et al. IEEE Micro, January/February 2001.



INTERNET

Multimedia & Transcoding

Database

Virus and Malware

Computer Games

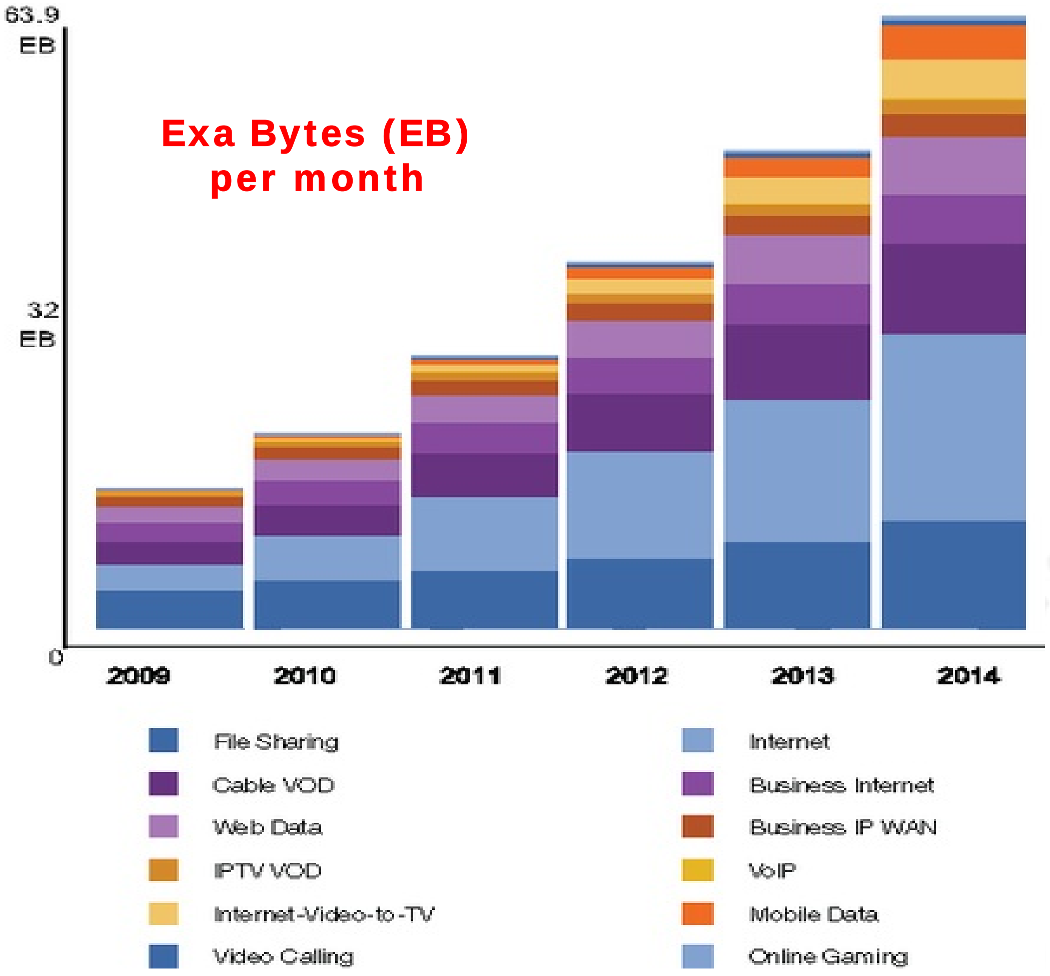
Wireless

Telephone

FTP

E Mail

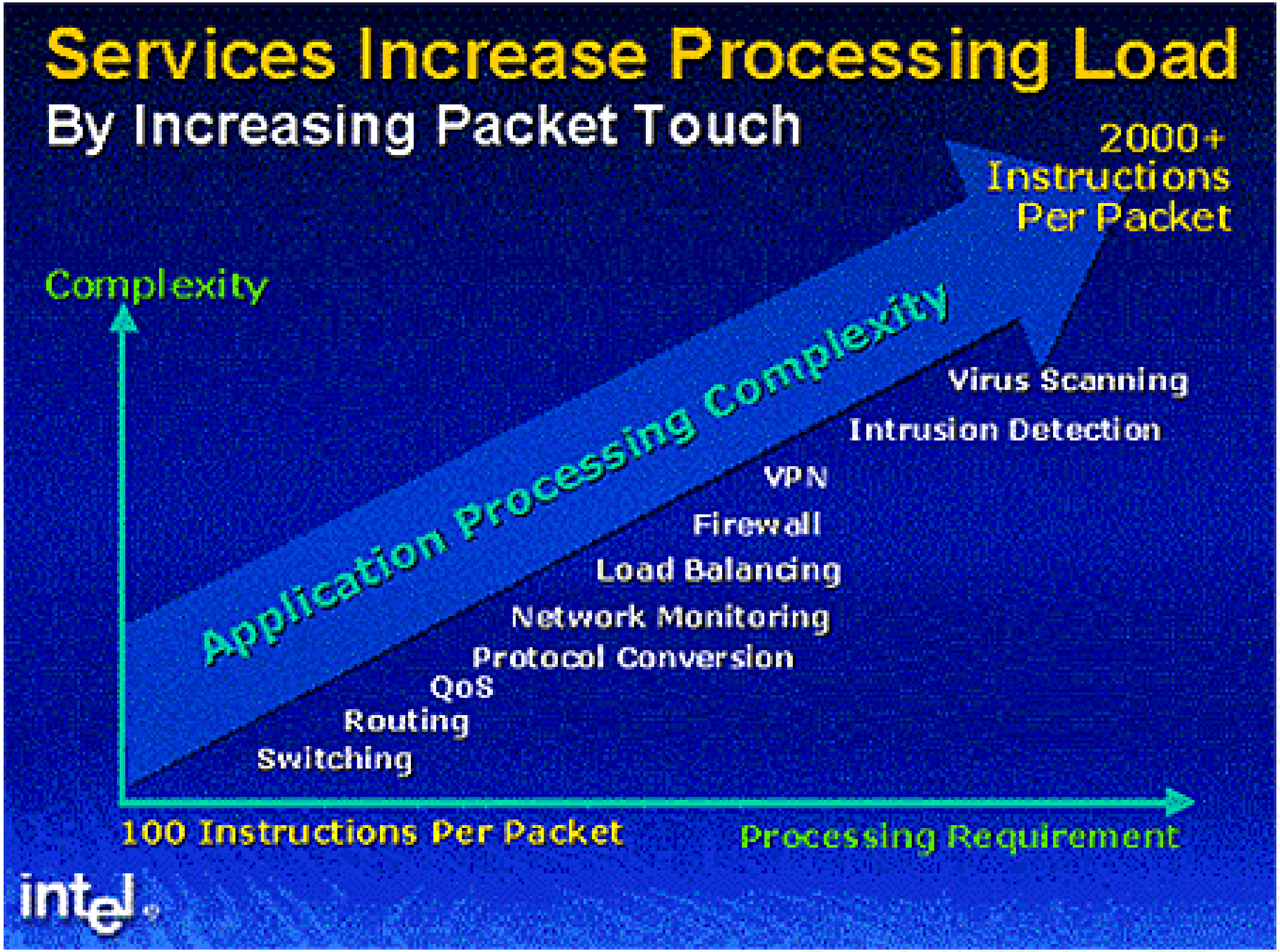
Internet traffic trends and Deep Packet Inspection at multi-gigabit data rates



Exa Bytes (EB) per month

DPI: Four major functions

- Protocol analysis and application recognition.
- Anti-malware and anti-virus.
- Intrusion Detection and Prevention (IDS/IPS).
- URL filtering



Services Increase Processing Load

By Increasing Packet Touch

Network Computing

•How to increase throughput? – Packet Level Parallelism (PLP) => Employ multicore processors

•Adaptive Scheduling and Load Balancing techniques

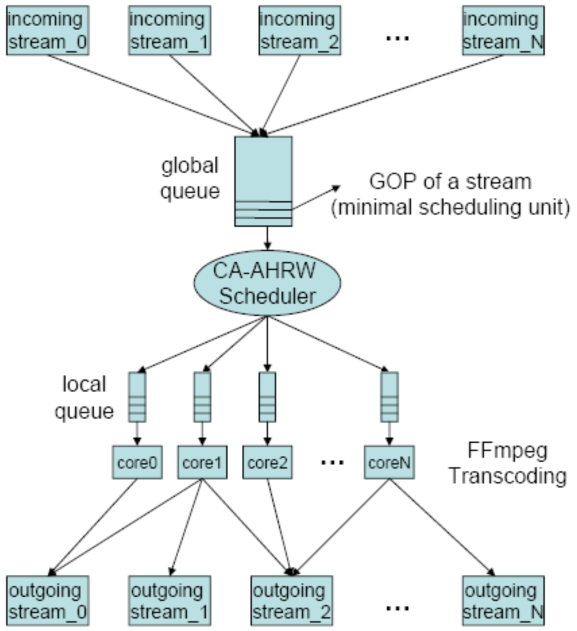
•Maintain connection locality and cache locality

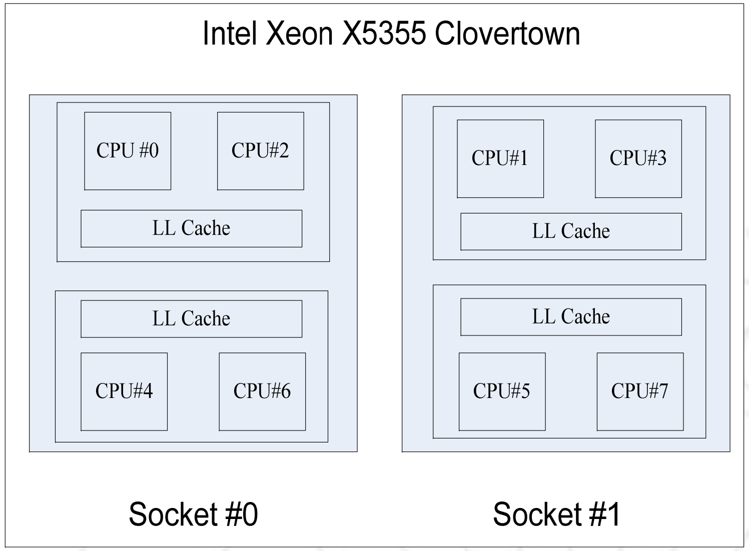
•Messages may have real-time constraints – Latency in addition to throughput

•How about QoS – Jitter and Out-of-Order departure of packets? More processors => more out-of-order packets

Similar to parallel processing but different constraints for packet processing on multicore processors

Scheduling on Multicore Machine

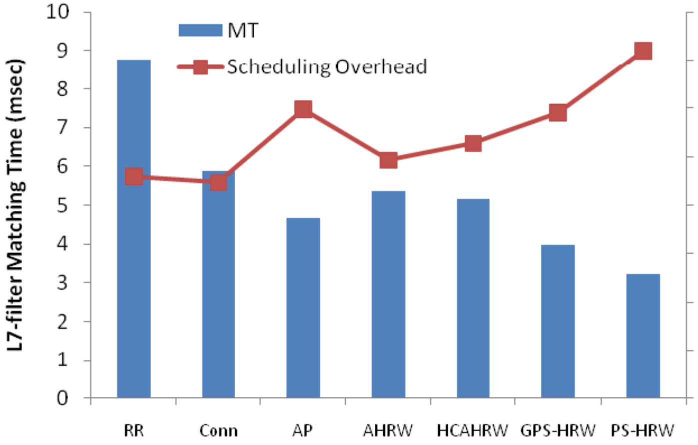




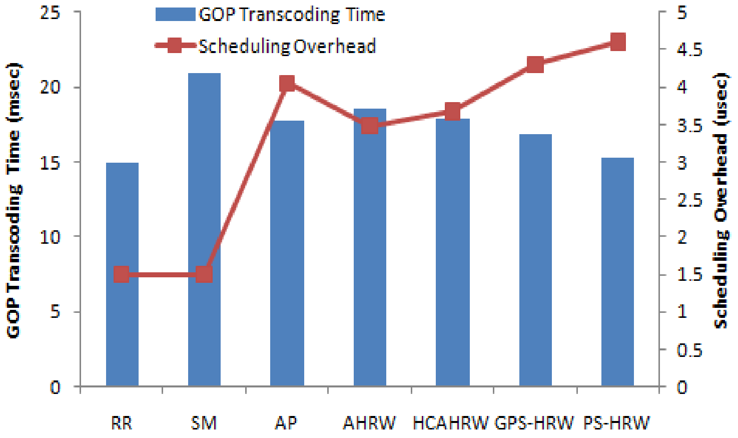
Aim: (1) Locality: Assign same stream to cores sharing the same LL cache – This will increase throughput, reduce latency, jitter and out-of-order departures.

(2) Load Balancing: Preserve load balance to increase throughput

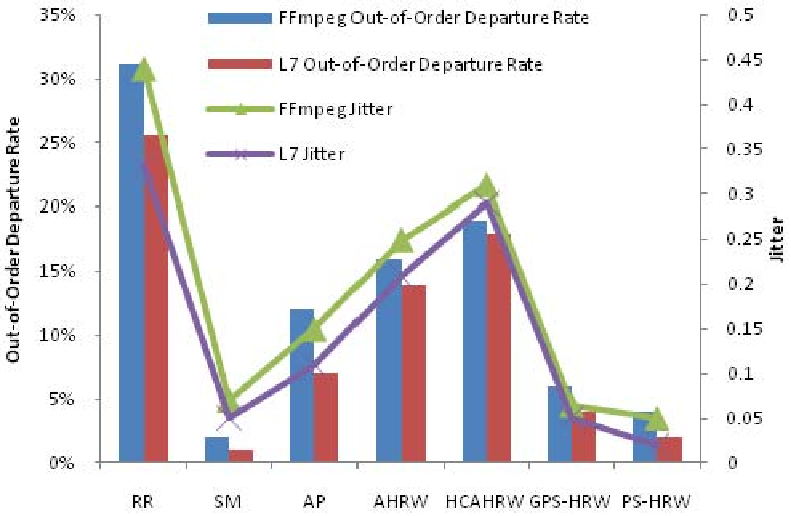
Execution Time and QoS Results



L7 Filter



Ffmpeg Transcoding



Multicore Scheduling of Network Applications

Throughput-centric (TPDS 2006, ANCS 2009, 2010, INFOCOM 2011, TON 2012)

- Connection locality [Affinity]
- Packet Level Load balance [AHRW]
- Cache topology [CA-AHRW]
- Hierarchical multicore [H-CAHRW]

QoS-centric (TPDS 2006, INFOCOM 2011)

Power/Energy-Aware (DAC 2010, INFOCOM 2010, ANCS 2011, DAC 2012, ANCS 2014)

Energy Proportionality

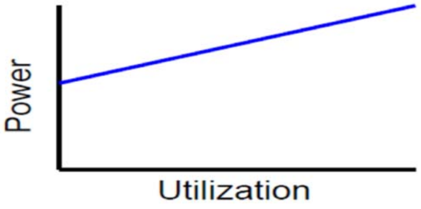
"The Case for Energy-Proportional Computing," Luiz André Barroso and Urs Hölzle, *IEEE Computer*, Dec. 2007

Server systems utilized between 10% and 50%

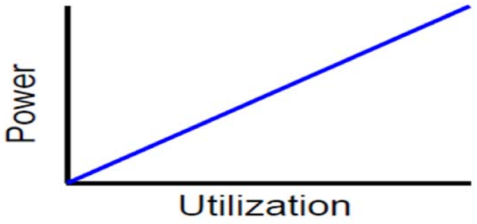
Energy use should be proportional to system activity => Ideally, Pidle = 0

This may be an unreachable ideal, in most cases

Implies a wider dynamic power range for current systems – these generally have Pidle >> 0



Proportional, P_{idle} >> 0



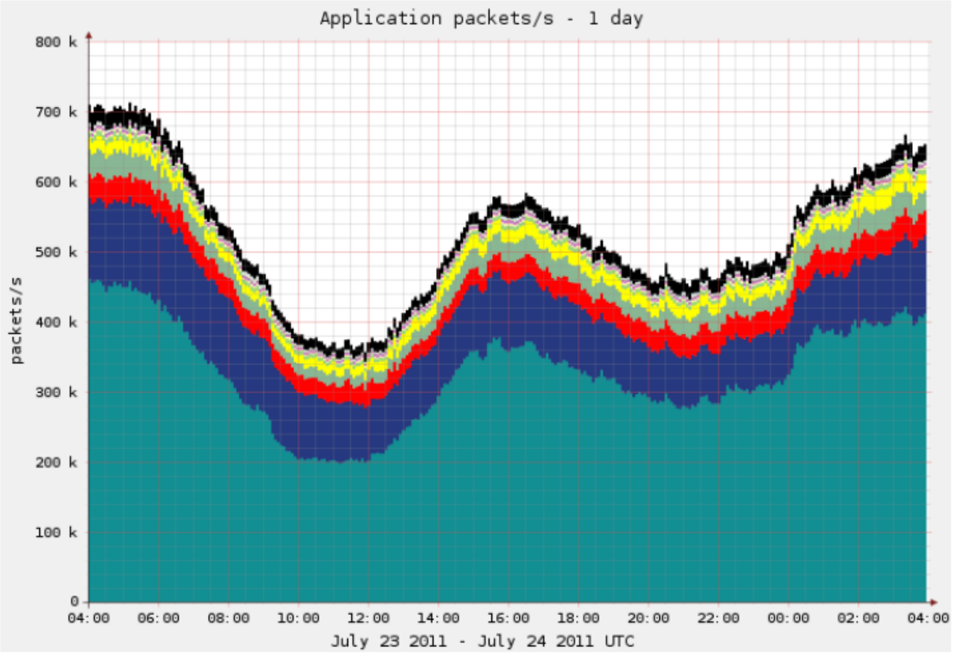
Proportional, P_{idle} = 0

Techniques for Power Saving

- Power consumption consists of Static and Dynamic Power, controlled by voltage and frequency respectively
- **DVFS**: Dynamic Voltage and Frequency Scaling => Also DVS, DFS and Rate Adaptation in networking terminology
- **Clock Gating**: Disable the clock for different processors => Core gating in multicore processors => Mostly saves dynamic power
- **Power Gating:** Disconnect power supply so that both static and dynamic power is eliminated. However, it takes longer time to wake up.

Traffic-Aware Power Optimization

Network traffic variation -> computing power fluctuation
Can we apply Rate Reduction (DVFS) to reduce power consumption at night?

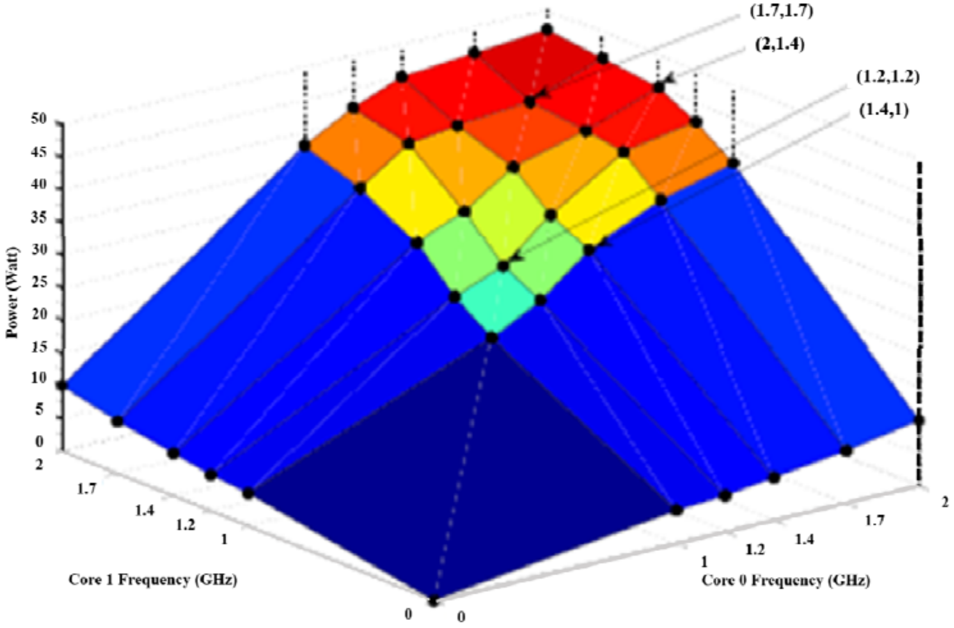


Assign different frequencies to different cores

Throughput depends on cumulative frequency (total rate) of multiple cores. Always provide the minimal cumulative core frequency that satisfies the traffic demand

For a given cumulative core frequency, the per-core frequency combination with the least standard deviation consumes the least power

Results for 2-core CPU



DAC 2012

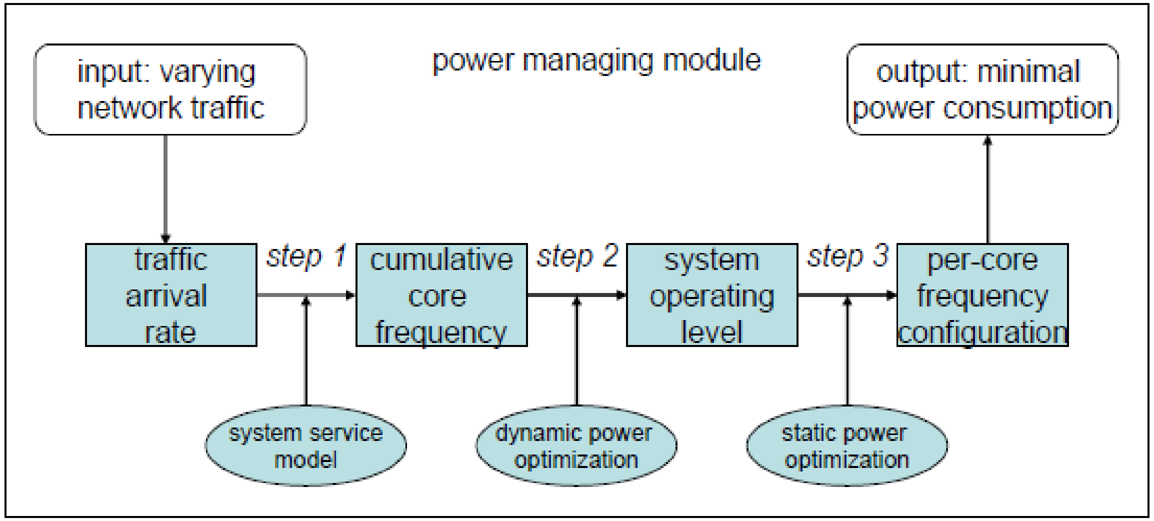
Power Management Module

Power is reduced by changing the system operating level according to varying traffic rate. Developed a runtime for automatic control.

Step 1: Building relationship between traffic, power and core frequencies

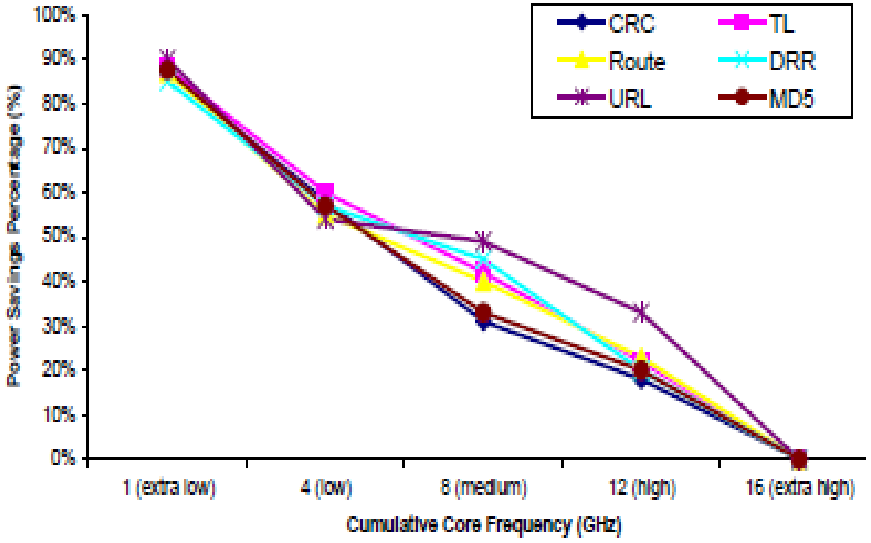
Step 2: Develop a runtime - Reduce dynamic power by varying frequencies – per core DVFS

Step 3: Reduce static power by limiting core temperature through migration

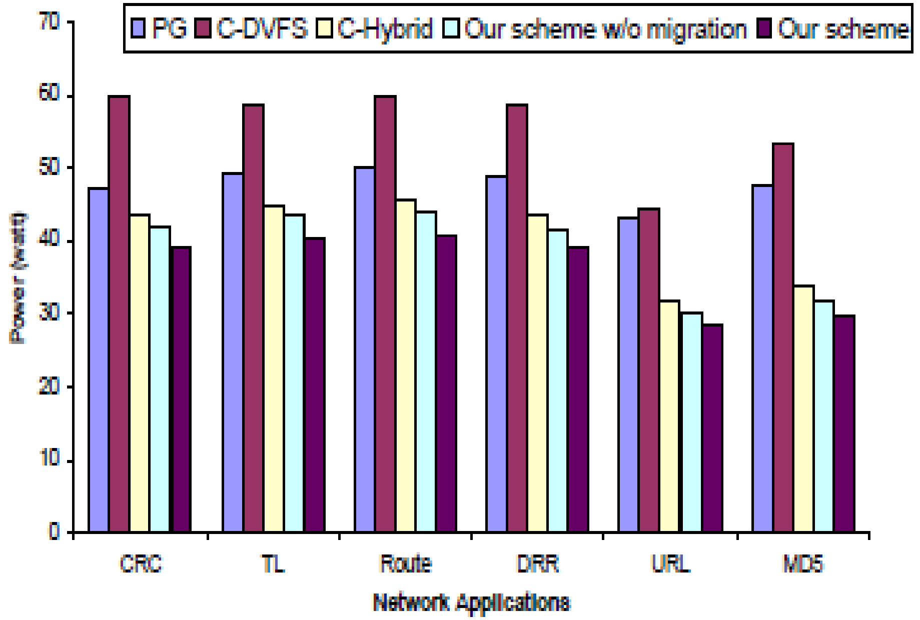

input: varying network traffic
power managing module
output: minimal power consumption
traffic arrival rate
step 1
cumulative core frequency
step 2
system operating level
step 3
per-core frequency configuration
system service model
dynamic power optimization
static power optimization

Experimental Results

Power savings percentage compared to a traffic-unaware native system



Power performance compared to three other traffic-aware schemes

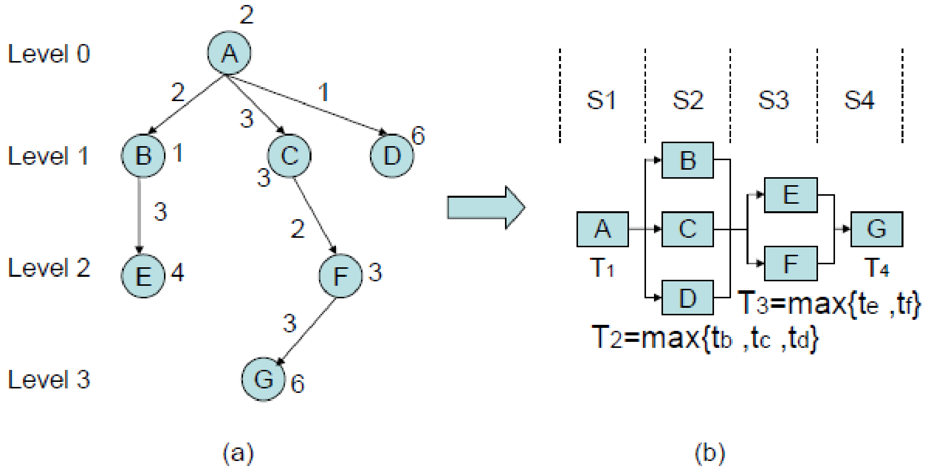


Implemented on AMD two quad-core Opteron 2350 processors.

DAC 2012

Optimizing Throughput and Latency under Given Power Budget

Network Application



(a) (b)

A parallel-pipeline scheduling from DAG.

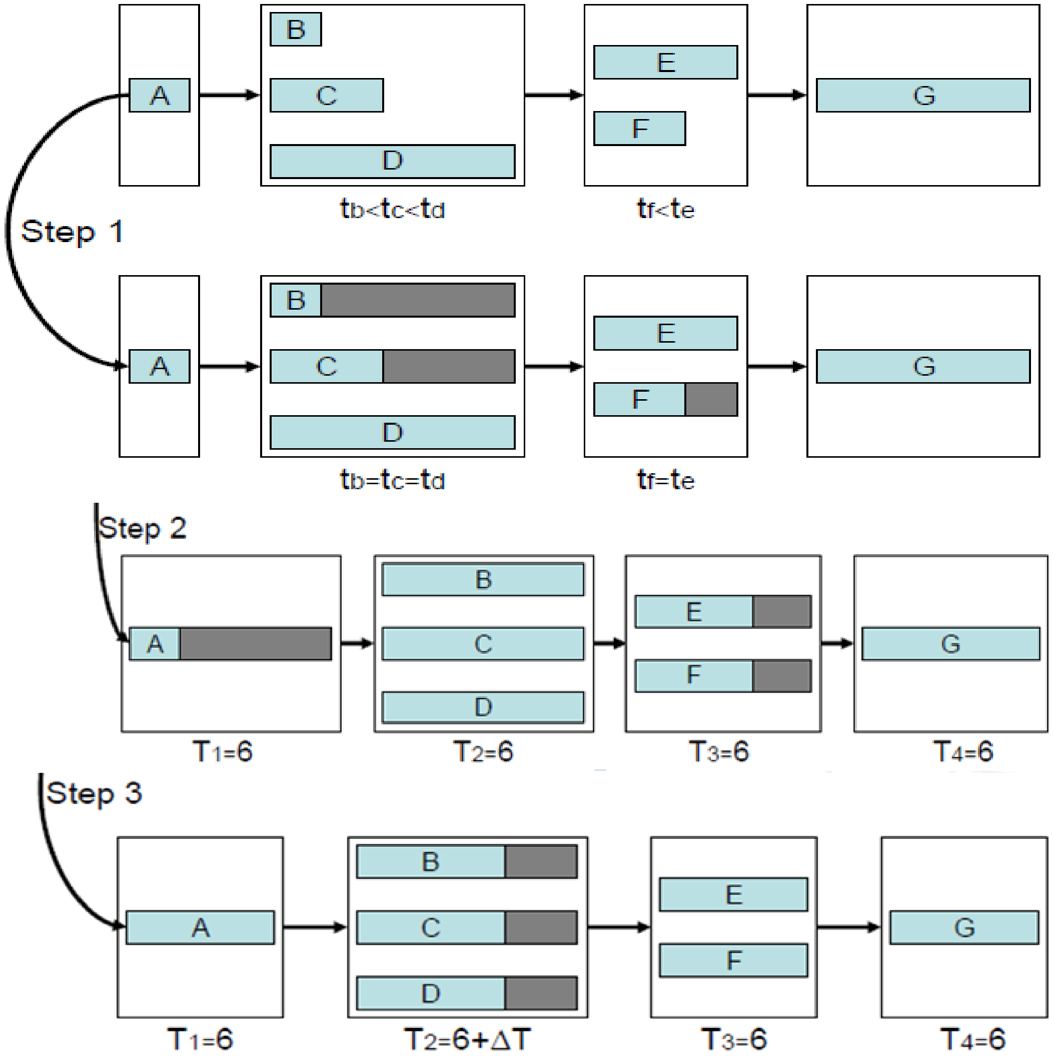
Given the parallel-pipeline scheduling and power budget, how to optimize the per-core frequency to maximize the throughput and minimize the latency.

$$Maximize \quad Th = \frac{1}{T_{max}} = \frac{1}{Max\{\frac{C_i}{f_i}\}} = Min\{\frac{f_i}{C_i}\}$$

$$Minimize \quad L = T_1 + T_2 + ... + T_S = (\frac{C_1}{f_1} + \frac{C_2}{f_2} + ... + \frac{C_S}{f_S})$$

Infocom 2010

Our Algorithm



Original DAG

(a)

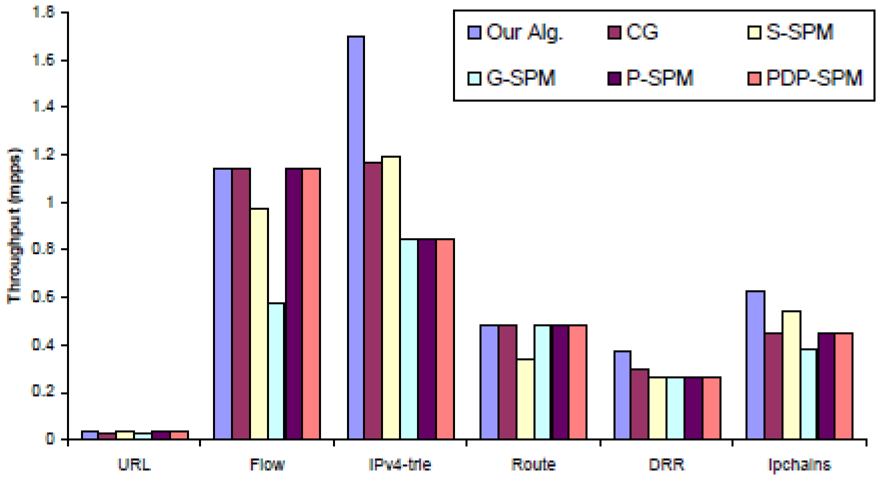
1. Reduce frequencies for B,C and F to match td

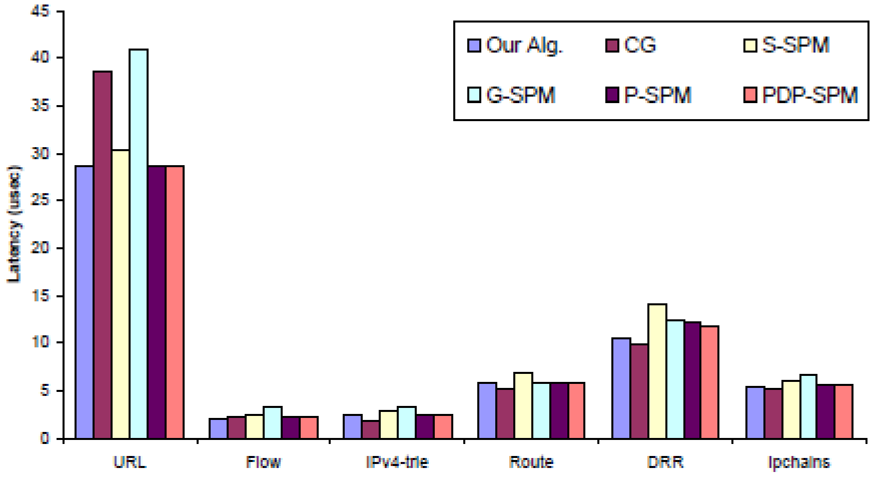
(b)

2. Reduce frequencies for A, E and F to match T2

3. Further reduce frequencies to meet power budget

Throughput and latency performance





Implemented on AMD two Quad-Core Opteron 2350 processors.
Power budget is set to be 75% of the initial power consumption

Compared to the lowest performing technique -
Throughput: Avg:64.6%, Max:100.6% (IPv4-trie)
Latency: Avg:25.2%, Max:33.2% (Flow)

Infocom 2010

Architecture support for power management-Idle power

By disabling different components, CPU can enter different sleep states (C-states) during idle.

Different C-states have different transition latency, and different EBT (energy break-even time).

To avoid inefficient usage of deep C-state, the time that the CPU will be idle should be known in advance, so that the proper C-state can be selected for the CPU.

Intel Quad-Core Ivy Bridge

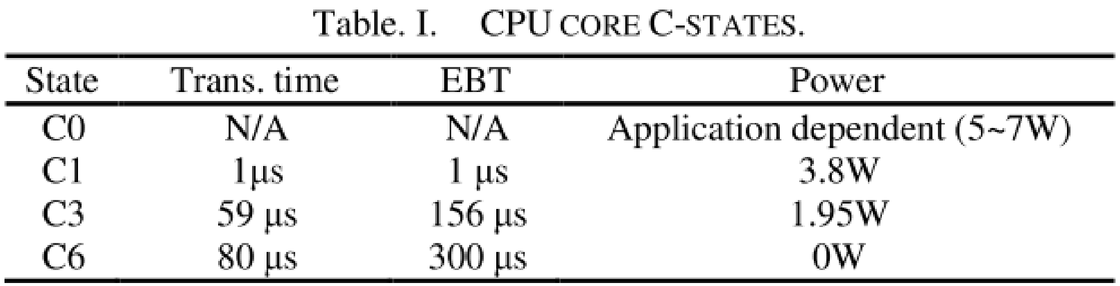
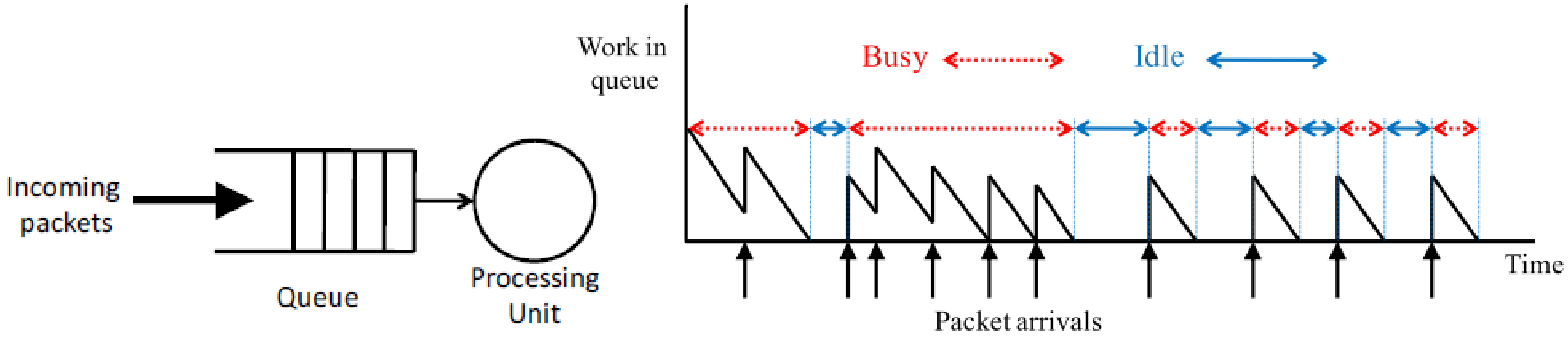


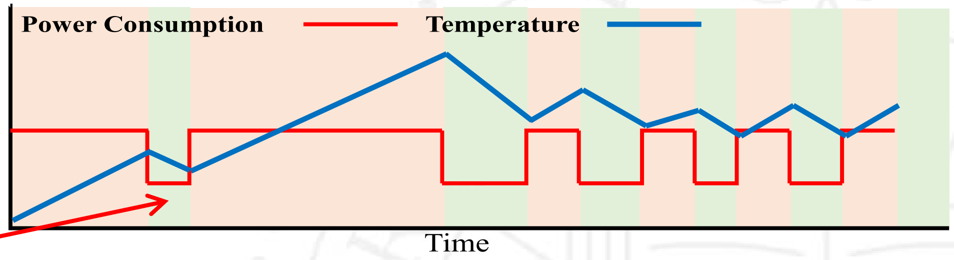
Table. I. CPU CORE C-STATES.

State	Trans. time	EBT	Power
C0	N/A	N/A	Application dependent (5~7W)
C1	1μs	1 μs	3.8W
C3	59 μs	156 μs	1.95W
C6	80 μs	300 μs	0W

Packet Processing on CPU



Random packet arrivals and execution times => random ON/OFF periods



Significant idle power consumption during OFF period

Temperature increases during ON period and decreases during OFF period

Depending on the packet processing time, the OFF periods are often short and frequent. (prevent deep C-states)

Proposed Vacation Scheme

Two states: Working and Vacation.

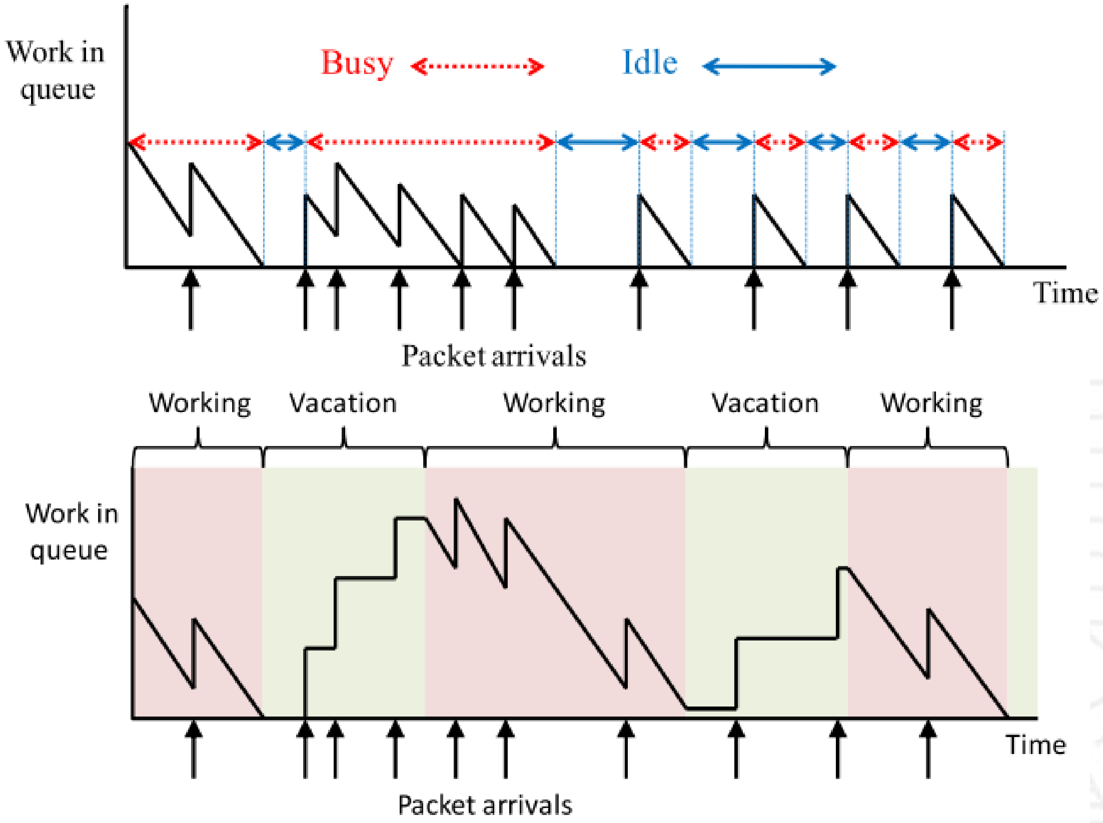
In working state, the CPU core will fetch packets from the queue and process them.

During the vacation, all incoming packets are buffered, and their processing are deferred to the next working period.

The core will transit from working to vacation state as long as there is no packet left in the queue, stay there for a period of time, and transit back to the working state.

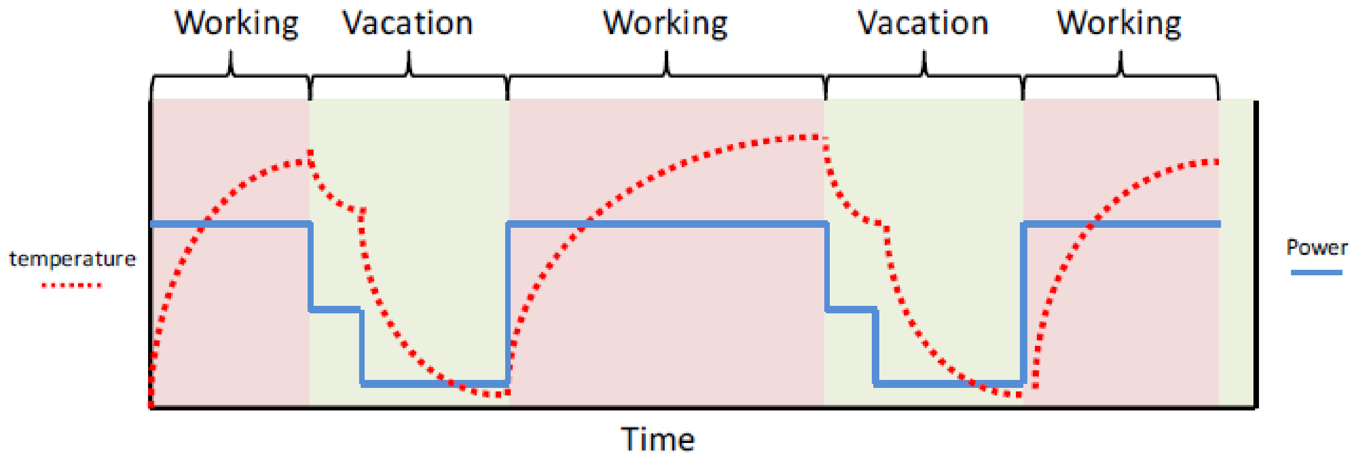
Packet processing with Vacation

The idle period is consolidated into longer vacation period, which allows the CPU core enter deep C-state depending on the vacation period. Longer vacation means more power saving but increase in packet latency

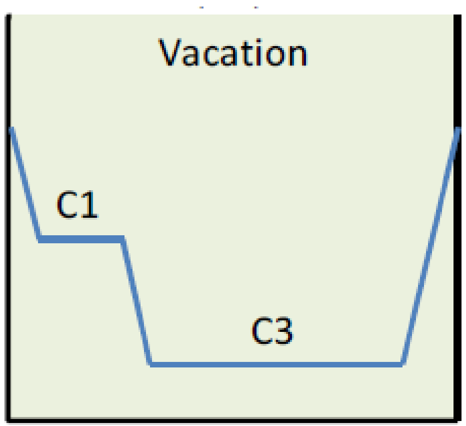


Power Consumption and Temperature Fall during Vacation

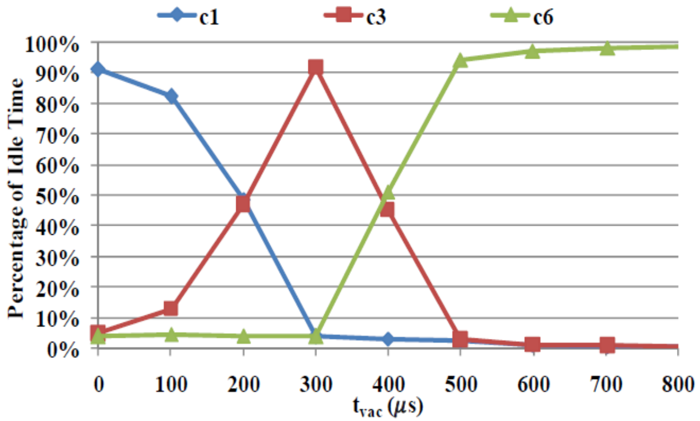
(a)



(b)



C-state transition during vacation period corresponding to figure (a)

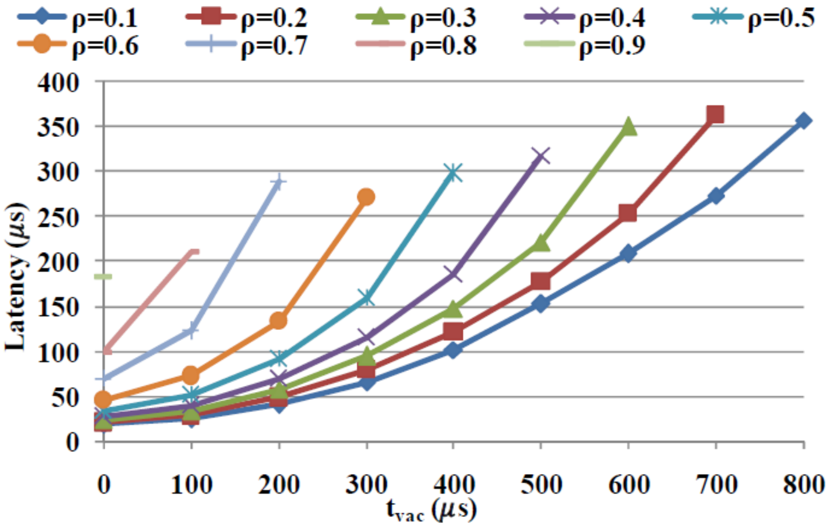


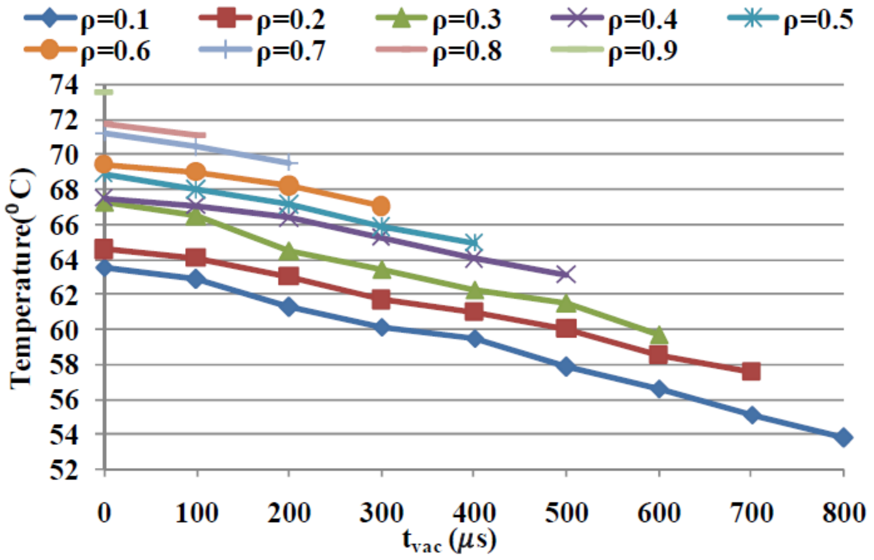
(c)

Processor states if / when it enters c6 state

Results with Synthetic Workload

Apply Gated M/G/1 Queuing theory to derive equation for Vacation Period based on tolerable per packet latency or temperature constraint



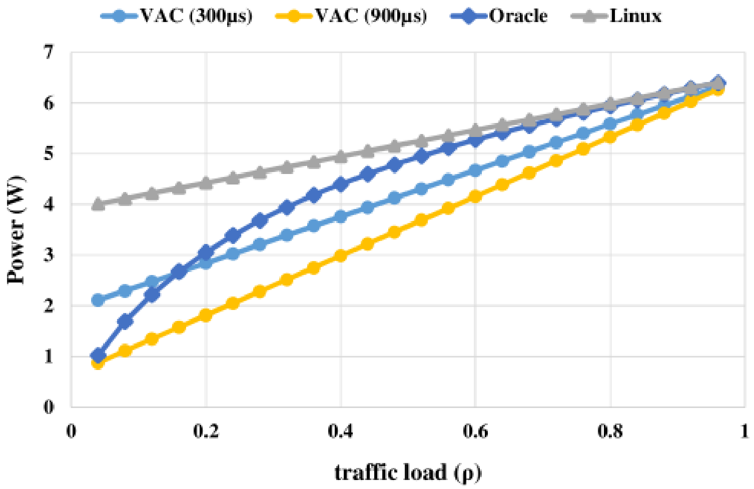


ANCS 2014

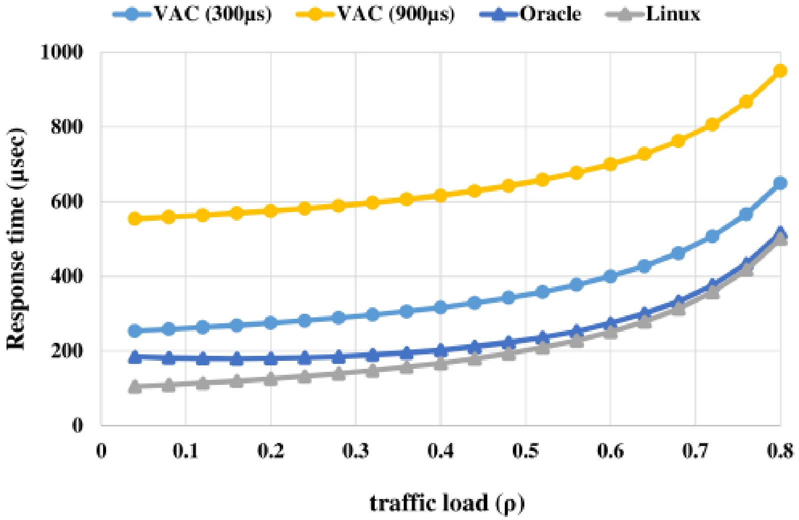
Per-core Vacation Results

Experiments using Intel i7 quad-core processor.

Power consumption and latency compared to default Linux system and theoretical Oracle system



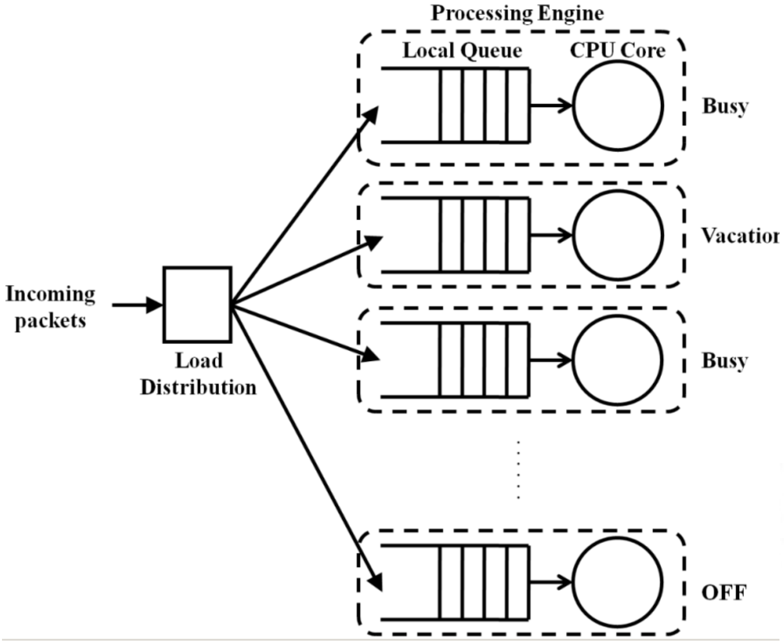
Power consumption



Response time

Multicore Vacation Scheme

With different configurations (vacation time and # of cores), we trade the power consumption saving with the additional response time.



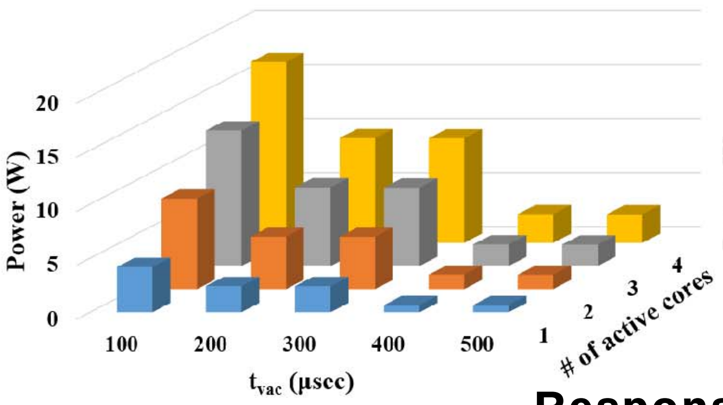
Idea is to switch on the OFF processors to vacation and then to active as the traffic increases. Develop analysis for the situation and verify experimentally

Multicore Vacation Scheme

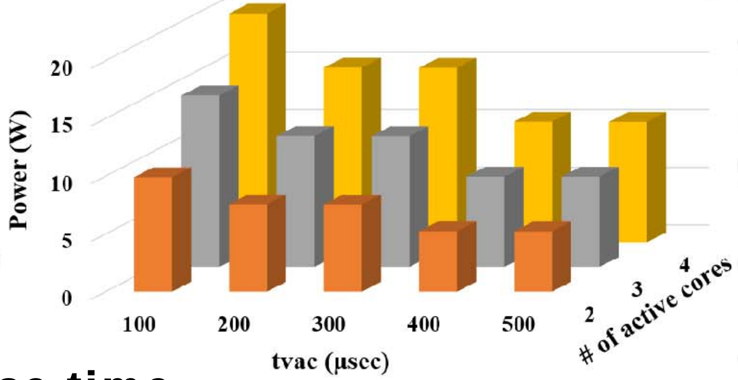
One more dimension - the number of cores

Power consumption

ρ = 0.1

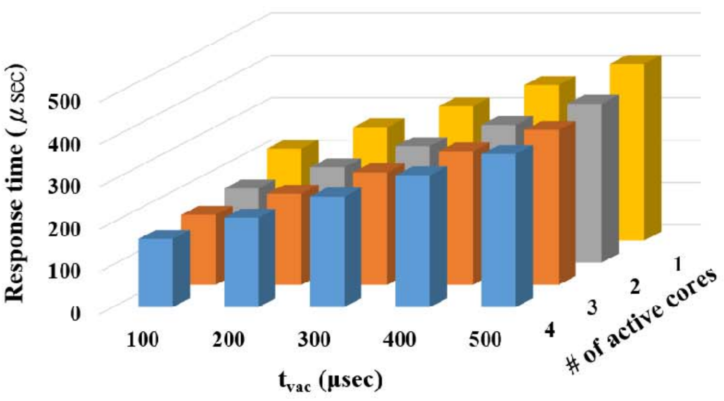


ρ = 0.4

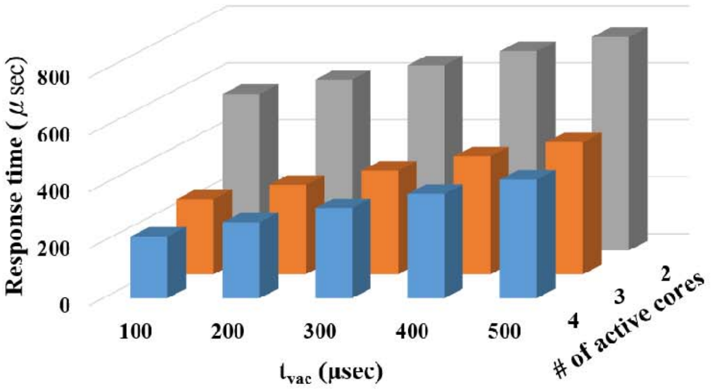


Response time

ρ = 0.1

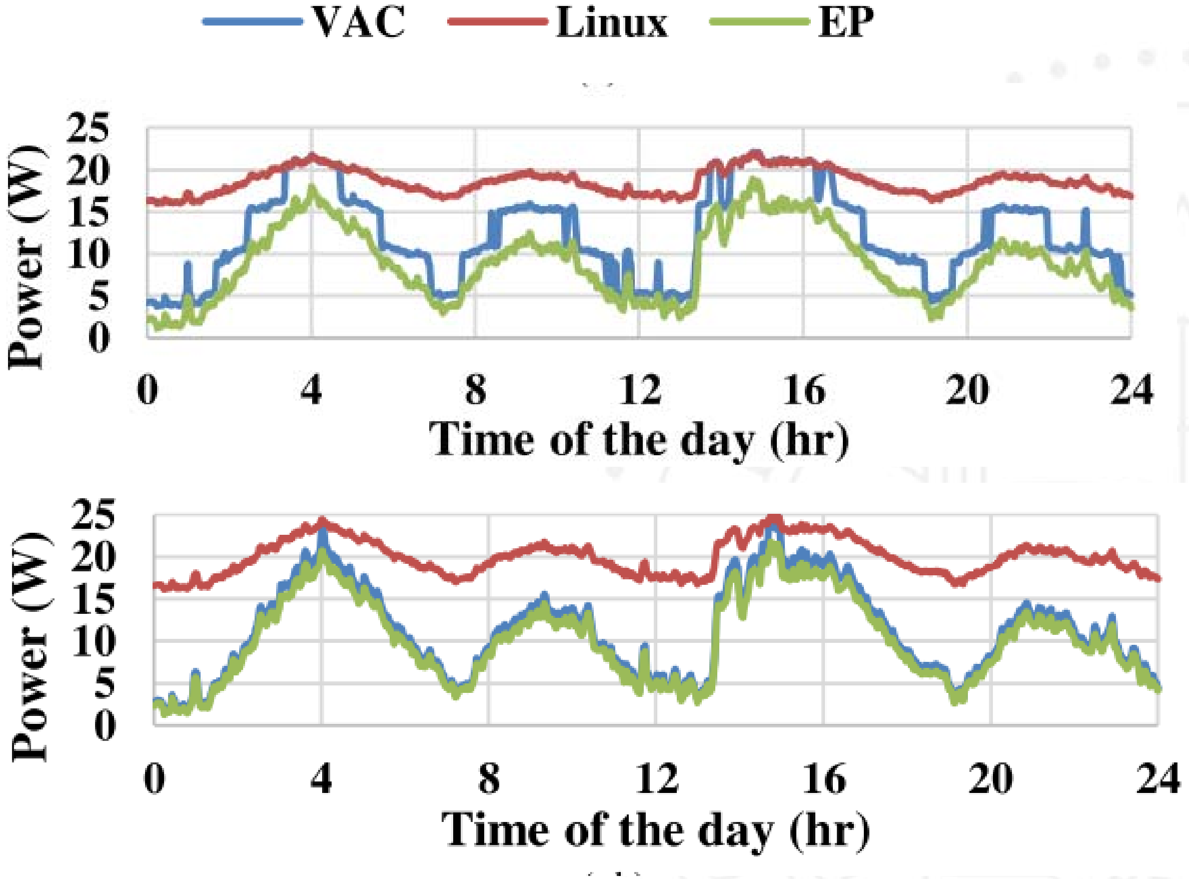


ρ = 0.4



Results under real world traces

24-hour traffic trace from CAIDA with real world packet processing applications.

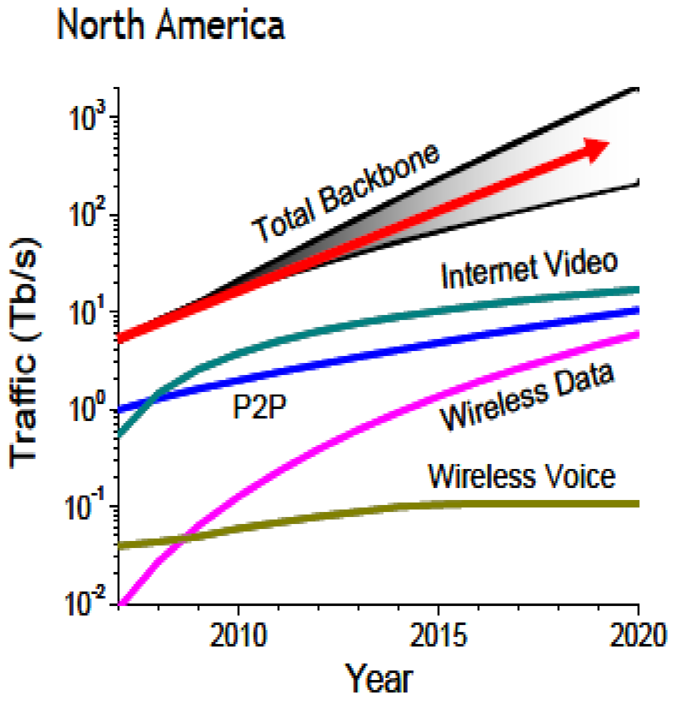


IPv4 Routing

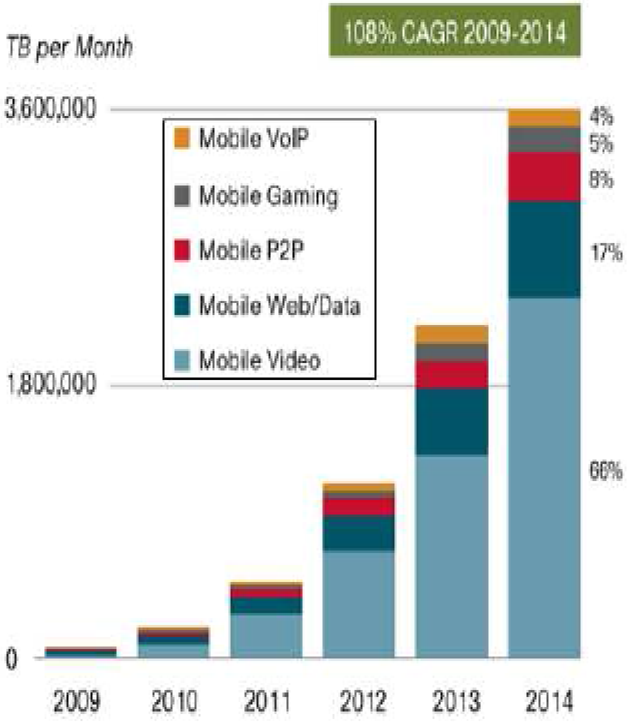
IPSEC

FUTURE RESEARCH:

MASSIVE DATA GROWTH IN MOBILE VIDEO



Data from: RHK, McKinsey-JPMorgan, AT&T, MINTS, Arbor, ALU, and Bell Labs Analysis: Linear regression on log(traffic growth rate) versus log(time) with Bayesian learning to compute uncertainty



Source: Cisco VNI Mobile, 2010

Power Saving in Mobile Phones

EXAMPLE: The big.LITTLE ARM Architecture used in Android Phones. Use A7 for Light Loads and A15 for Heavy Loads. Significant difference in power consumption between Cortex-A15 and Cortex-A7 comes due to architectural differences

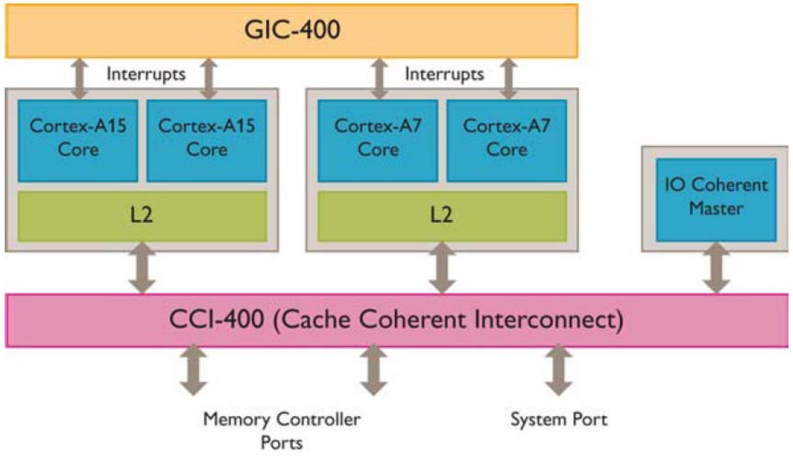
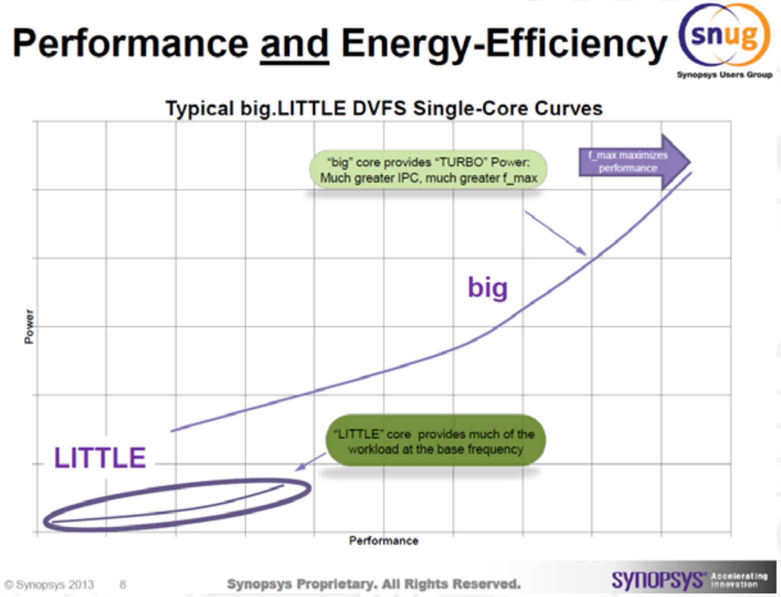


Fig 1: Typical big.LITTLE system



Performance and Energy-Efficiency

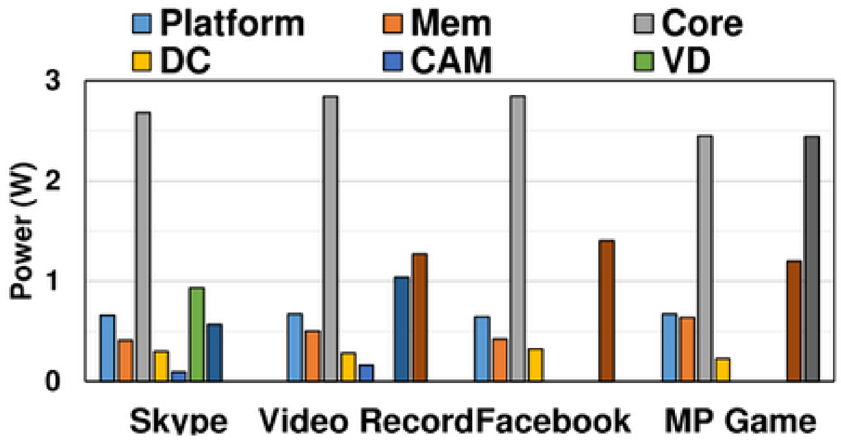
Synopsys Users Group

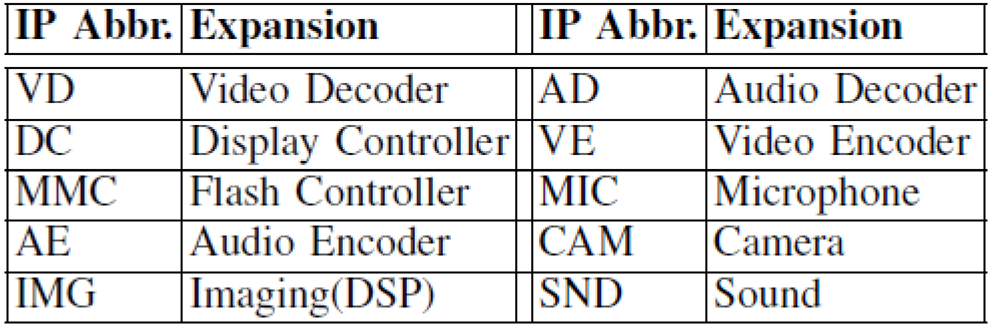
Typical big.LITTLE DVFS Single-Core Curves

© Synopsys 2013 8 Synopsys Proprietary. All Rights Reserved.

Interesting Aspects: Scheduling for low power and temperature on heterogeneous cores for mobile applications

SoC Power Consumption for Different Mobile Applications



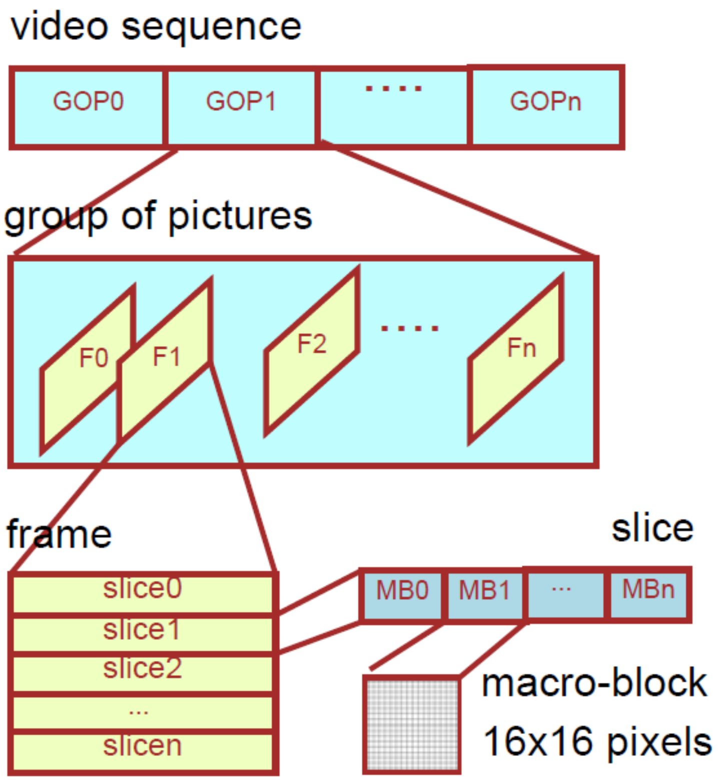


IP Abbr.	Expansion	IP Abbr.	Expansion
VD	Video Decoder	AD	Audio Decoder
DC	Display Controller	VE	Video Encoder
MMC	Flash Controller	MIC	Microphone
AE	Audio Encoder	CAM	Camera
IMG	Imaging(DSP)	SND	Sound

Nachippan etal.
HPCA 2015

Use of Approximate Computing

EX: Video Processing => H.264 video's hierarchical structure & Potential approximations by dropping frames, slices or macroblocks



CONCLUSION

- Network Computing deals with processing packets that arrive randomly over the network.
- Multiprocessing is needed to meet throughput and execution time demands
- Reducing Power Consumption is very important to network application processing – schedule packets accordingly
- Applied DVFS, clock gating, power gating and migration to deal with variation in the network traffic and save power.
- Developed a vacation scheme to put the processors to deep sleep while satisfying the latency and temperature constraints
- Future research is to develop these schemes for multicore mobile phones

Thank you!