Green Networks: Reducing the Energy Consumption of Networks

Ken Christensen
Department of Computer Science and Engineering
University of South Florida
Tampa, Florida USA 33620
christen@cse.usf.edu
http://www.csee.usf.edu/~christen

Seminar talk at UC3M
June 2, 2010

Thank you – Gracias

A big thank you to Jose Alberto Hernández and David Larrabeiti for organizing this talk. Everyone has really rolled-out the red carpet for me. Pedro Reviriego even picked me up at the airport.

I am very excited to be here in Madrid.

Acknowledging my students…

Some of the work presented here was done by past and present students including,

- Chamara Gunaratne (PhD in 2008)
  - Early Proxying and Ethernet work
- Miguel Jimeno (PhD in 2010)
  - Proxying (especially for applications)
- Mehrgan Mostowfi (MS in 2010, continuing to PhD)
  - Recent Ethernet work

Where do I come from?

University of South Florida and Tampa

47,000 students
9th largest in the US

http://wwp.greenwichmeantime.com/time-zone/usa/florida/map.htm

Yes, we have lots of alligators

Why green networks?

One of the most urgent challenges of the 21st century is to investigate new technologies that can enable a transition towards a more sustainable society with a reduced CO_2 footprint.

→ We need to reduce energy consumption

Here is one reason why…

Sea level in 2100 under "high emissions" scenario

I live here

From U.N. Intergovernmental Panel on Climate Change

The challenge to ICT

What role will ICT play in this grand challenge?

1) To directly reduce energy use of ICT

2) To enable energy savings in non-ICT

One way to be "green"…

Just have less and do less

- No houses, no cars, no travel, no PCs, no Internet, etc.

From http://strangemaps.wordpress.com/2007/12/16/218-koreas-dark-half/

North Korea at night.
A model green society?
I don't think so…

Notion of comfortable conservation

"I mean using less energy for identical performance, measured in whatever way the consumer wishes."

- Richard Muller (Physics for Future Presidents, 2008)

➡ In network speak, same QoS for less energy

Product lifecycle and green

Lifecycle of "stuff" (including ICT equipment)

Focus of this talk

Production → Use → End-of-use

Production
- Cleaner mining
- Cleaner manufacturing
- Use less toxic materials
- Use less materials overall
- Use less energy overall

Use
- Use less energy
- Extend lifetime

End-of-use
- Recycle materials
- Refurbish for reuse

1 kWh = $0.10

Energy consumed by a PC*
- Production = 2000 KWh
- Life (5 yrs) = 4200 KWh

* E. Williams, "Revisiting Energy Used to Manufacture a Desktop Computer: Hybrid Analysis Combing Process and Economic Input-Output Methods," *Proceedings of IEEE International Symposium on Electronics and the Environment*, pp. 80-85, 2004.

Roadmap of this talk

This talk has four major topics

- Briefly quantifying energy use of ICT
- Reducing induced energy consumption
- Reducing direct energy consumption
- Future challenges

Key definitions

Direct energy use

- Energy used by network links and equipment, but not end devices

Induced energy use

- Incremental additional energy used for a higher power state of end devices needed to maintain network connectivity

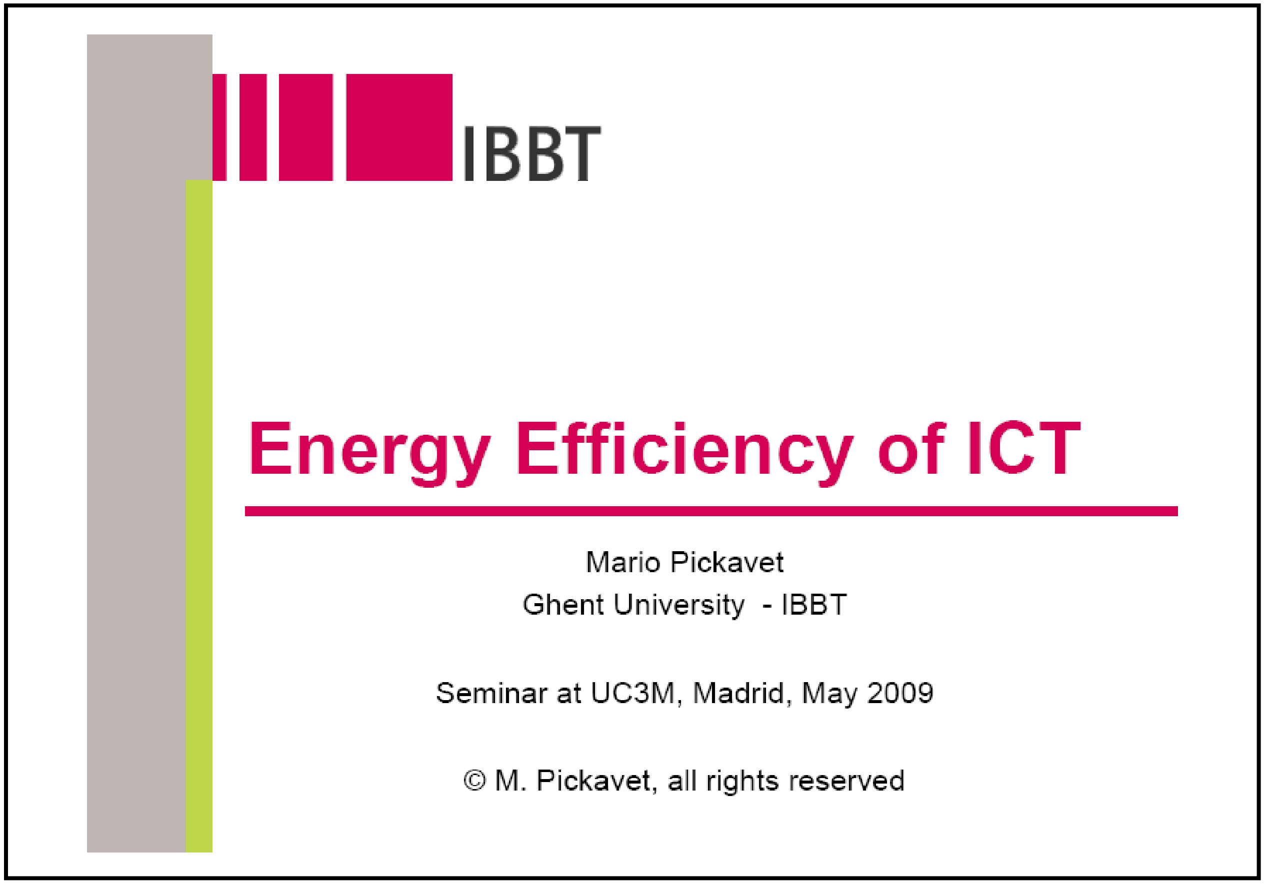
Quantifying the energy use of ICT

How much energy does ICT use?

... the Internet is part of this.

Energy consumption of ICT

Energy use presented in a previous UC3M seminar



A quick look at energy costs

In the USA

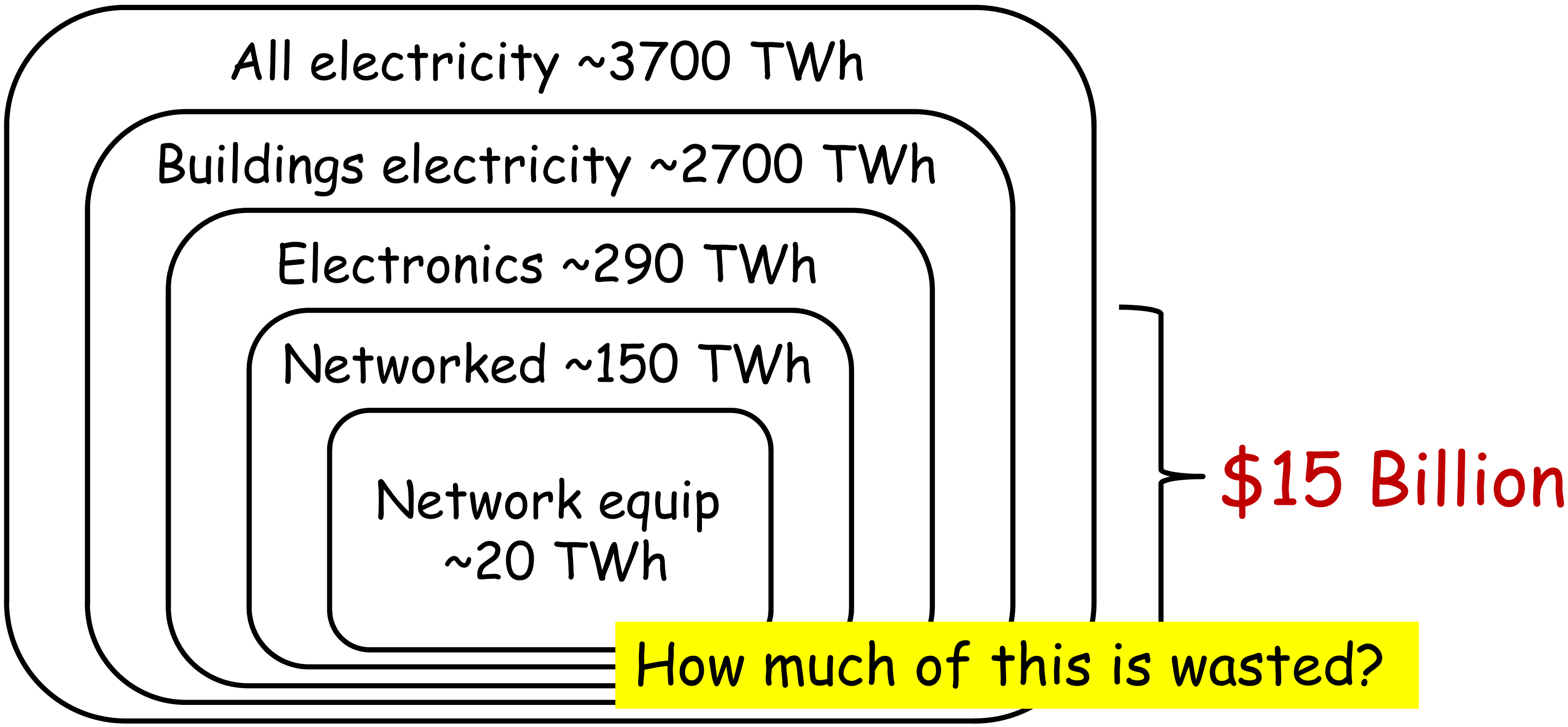
Recall that power is W and energy is Wh

- 1 kWh is about $0.10 (in the US)
- 1 TWh is about $100 million
- 1 W for 1 year is about $1 (actually, it is $0.88)

St Lucie, Florida
Twin nuclear units
About 11 TWh/year

Electricity use – big picture

Electricity use in the USA (2006, from LBNL)

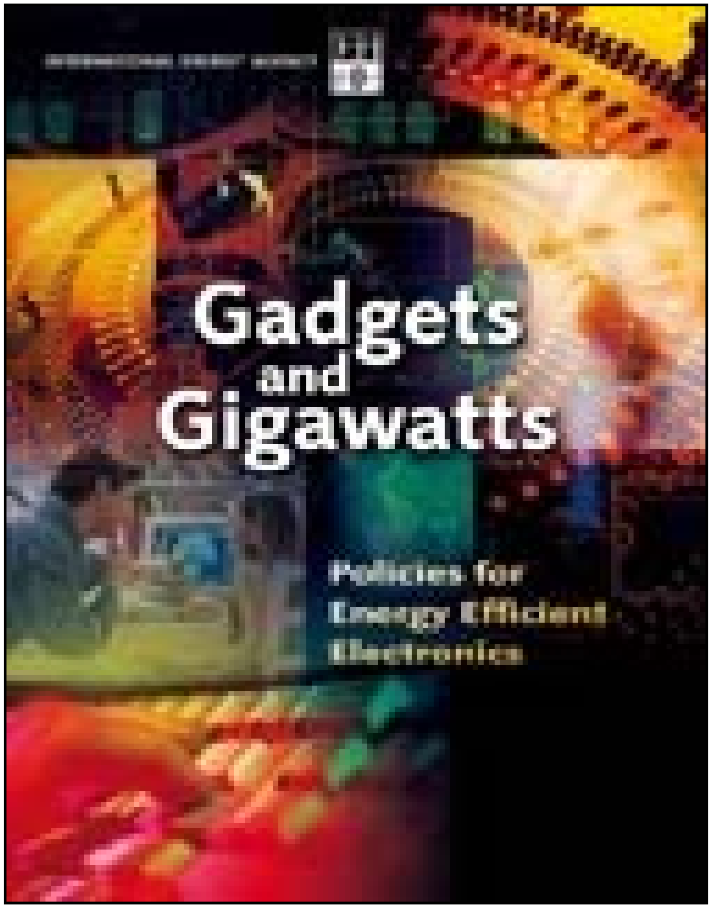


How much of this is wasted?

How much can be saved?

A view from the IEA

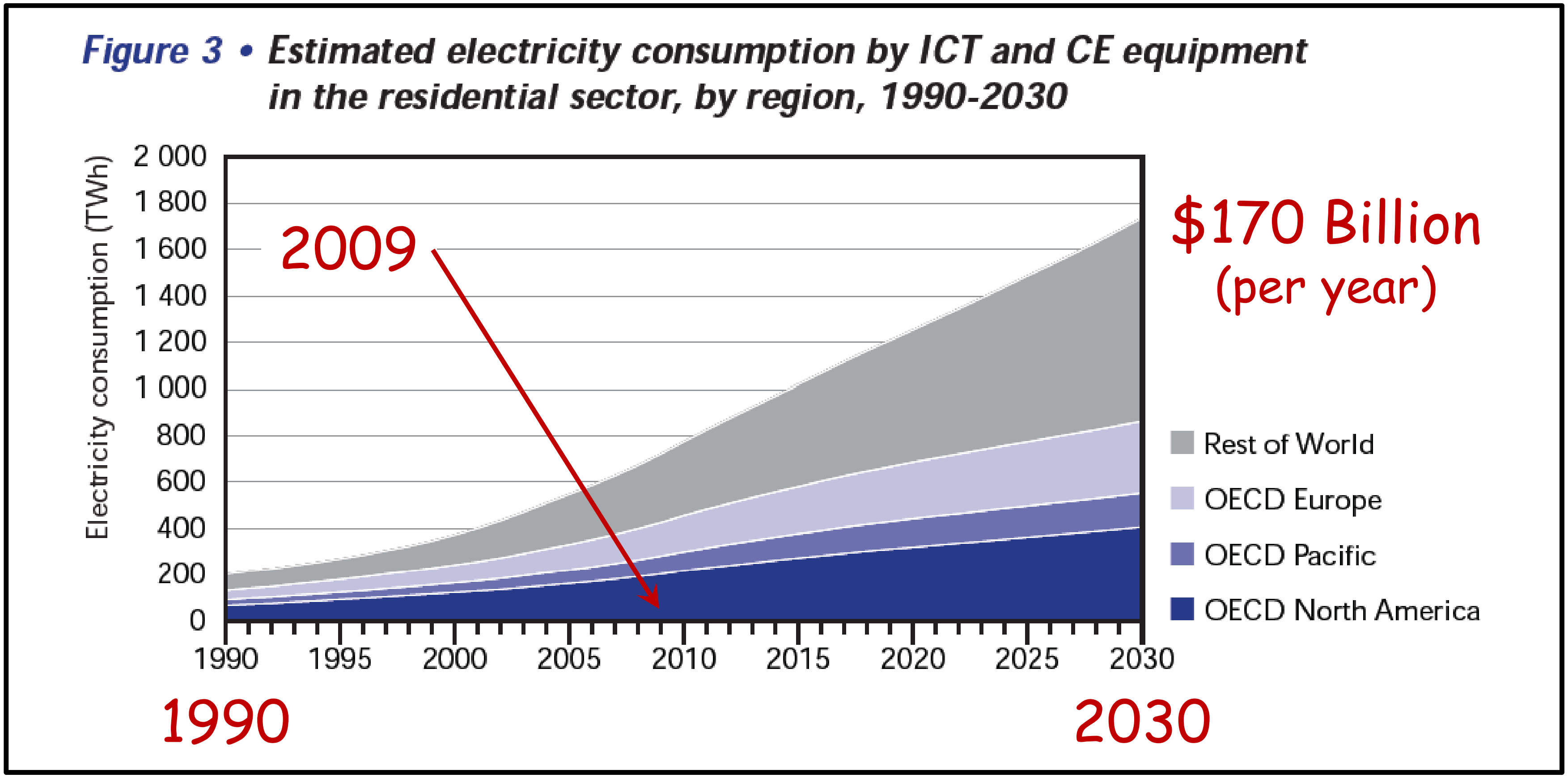
The Gadgets and Gigawatts book



- Focus is on policies for energy efficient electronics
- ICT and CE energy use is about 15% of household use
  - Growing very rapidly
- ICT and CE blur together

ICT electricity use – it is growing

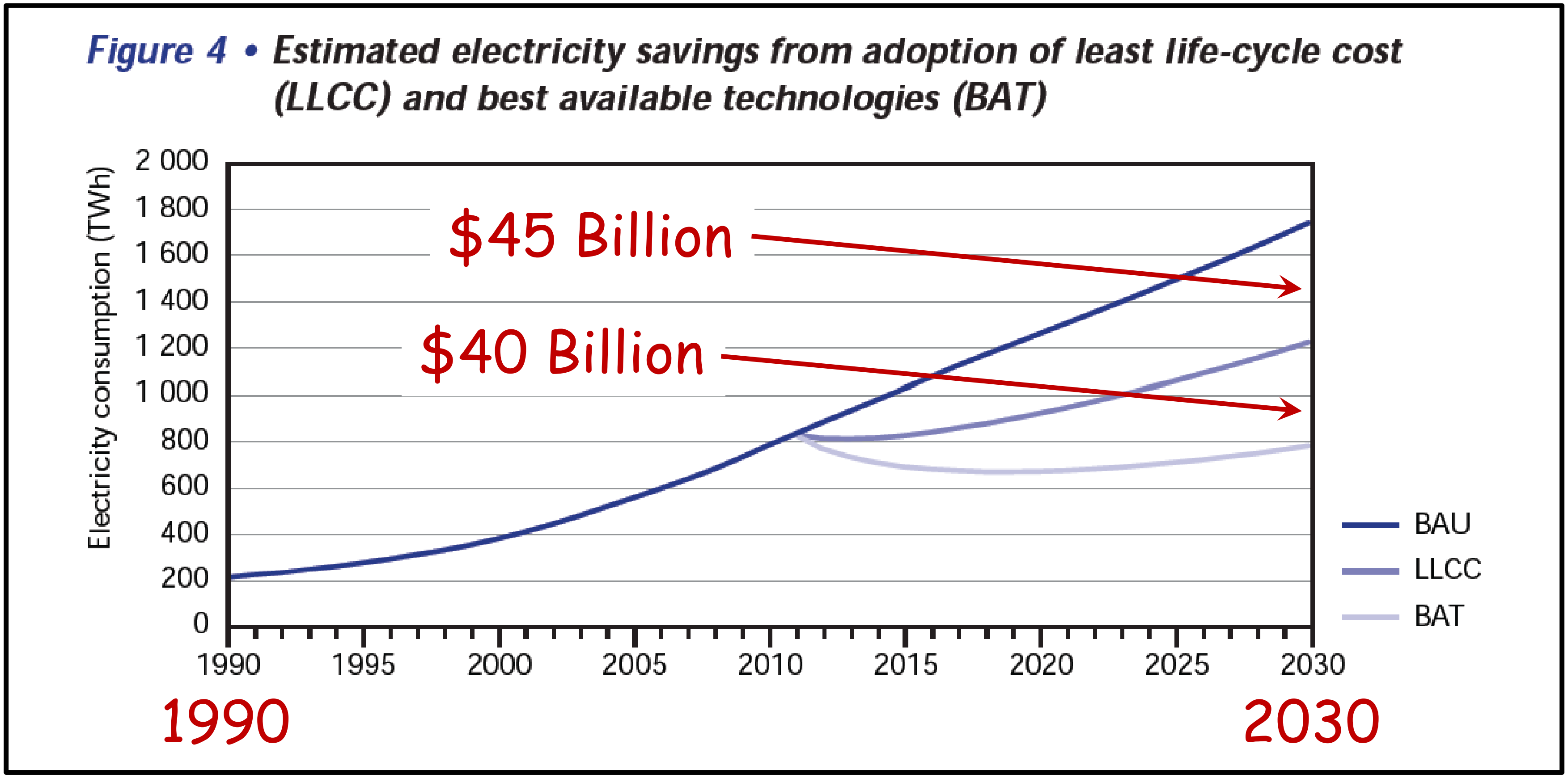
Electricity consumption estimates from IEA



From "Gadgets and Gigawatts," IEA, 2009.

ICT electricity use – possible savings

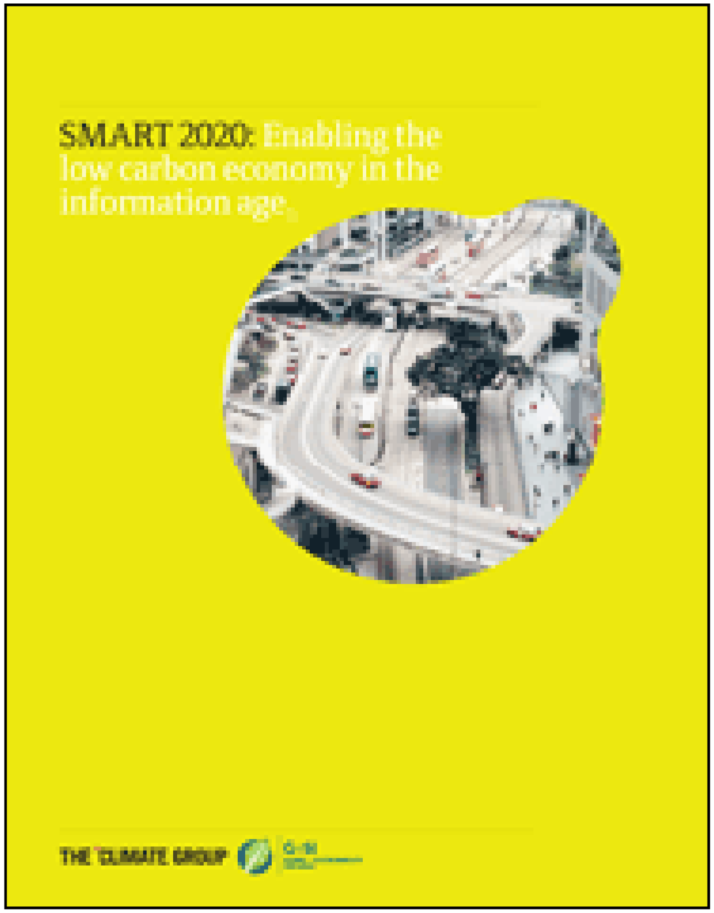
Electricity savings estimates from IEA



From "Gadgets and Gigawatts," IEA, 2009.

A view from the Climate Group

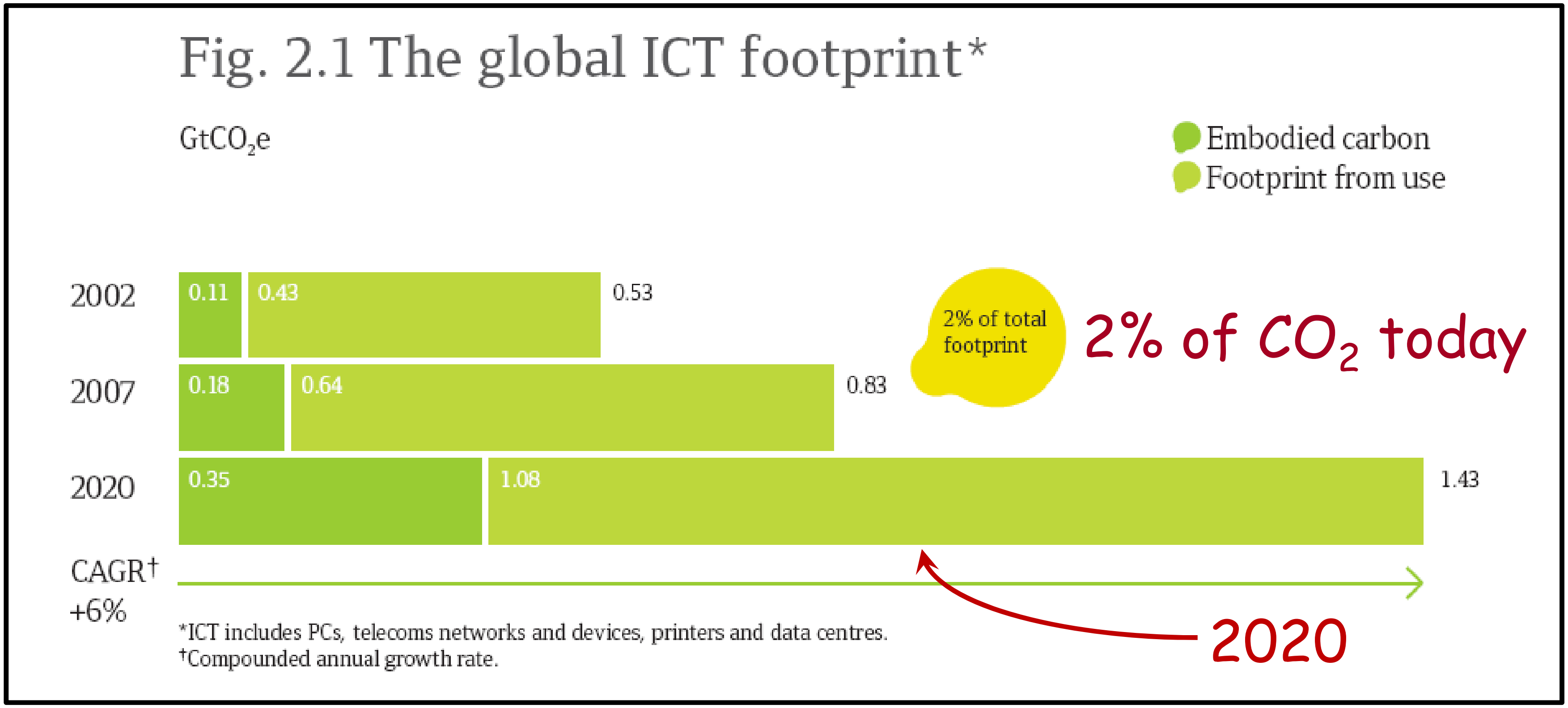
The SMART 2020 report



- Focus is on ICT's role in reducing greenhouse gases
  - Both of and by ICT
- A view of the world in 2020
  - Taking into account "likely" technology developments
- Supporting organizations
  - Include Cisco, Intel, HP, Sun, national telecoms, and telecom operators

Global ICT CO_2 footprint

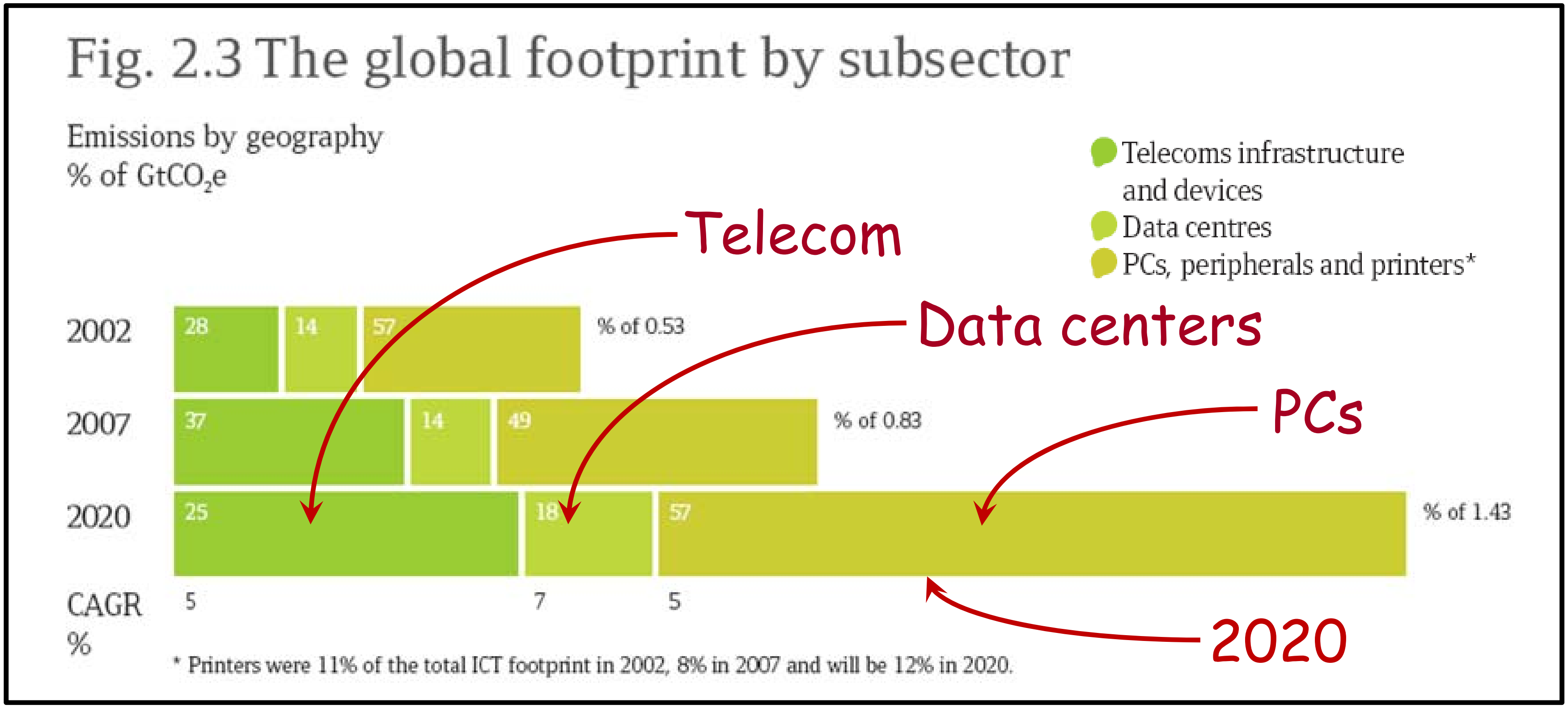
Today ICT is 2% of global CO_2



From SMART 2020 report

Global ICT CO_2 footprint <u>continued</u>

PCs (not data centers) are major CO_2 contributor



Data centers less than 15%

From SMART 2020 report

ICT CO_2 > Aviation CO_2

"The global information and communications technology (ICT) industry accounts for approximately 2 percent of global carbon dioxide (CO_2) emissions, a figure equivalent to aviation."
- Gartner Group, Inc. (2007)

➡ ICT use growing faster than airline traffic

➡ Greater impact by "fixing" ICT than airplanes

ICT energy use – the PC

The end user PC is the biggest energy consumer

"Desktop computing accounts for 45 percent of global carbon emissions from information technology."
- govtech.com

"Most PC energy use in the US occurs when no one is there, and this is greater than the total energy use of all network equipment."
- Bruce Nordman (LBNL)

Network energy use in Italy

Statistics for Italy

17.5 million broadband users,
population of Italy is 60 million

TABLE I.
(A) 2015-2020 NETWORK FORECAST: DEVICE DENSITY AND ENERGY REQUIREMENTS IN THE BUSINESS-AS-USUAL CASE (BAU). EXAMPLE BASED ON THE ITALIAN NETWORK.

	power consumption [W]	*number of devices* [#]	*overall consumption* [GWh/year]
Home	10	17,500,000	1,533
Access	1,280	27,344	307
Metro/Transport	6,000	1,750	92
Core	10,000	175	15
		Overall network consumption	1,947

From: R. Bolla, R Bruschi, K. Christensen, F. Cucchietti, F. Davoli, and S. Singh, "The Potential Impact of Green Technologies in Next Generation Wireline Networks – Is There Room for Energy Savings Optimization?", submitted to *IEEE Communications*.

Network energy use in Italy <u>continued</u>

Another statistic for Italy…

Today, the energy needed by the Telecom Italia's Network
is more than **2.000.000.000.000Wh** (>2TWh)
representing nearly 1% of the total National energy demand,
second user only to the National Railways

From: F. Cuccheietti, "Energy Efficency – An Enabler for the Next Generation Network," Presentation by Telecomm Italia, Bruxelles, January, 30 2006.

Summary for ICT energy use

ICT uses a lot of energy

- ICT contributes about 2% of human emitted CO_2
  - About equal to aviation industry
  - Rapidly growing
- Most of this energy consumption comes from the edge
  - Not from data centers
- Induced energy use is probably much greater than direct energy use
  - This comes from PCs (and PC-like devices)

Reducing induced energy consumption

Can we reduce energy used by PCs?

... this is a networking problem.

Just a few lines of code?


PASSPORT | RICKS | DREZNER | WALT | ROTHKOPF | LYNCH | THE CABLE | NET EFFECT | S

FP Foreign Policy

PASSPORT

A blog by the editors of FOREIGN POLICY

search

The New ForeignPolicy.com

Global News : Passport : Ricks : Drezner : Walt : Rothkopf : Lynch

The Cable : The AfPak Blog : Net Effect : Shadow Govt. : Madam Secretary : The Call

Microsoft could save 45 million tons of CO2 emissions with a few lines of computer code

Wed, 11/15/2006 - 2:59pm

Here's a memo to **Bill Gates** and **Steve Ballmer**:

It is estimated there are **660 million computers** in use worldwide, the majority of which run some iteration of a Microsoft operating system. Generating the electricity needed to power those computers requires hundreds of power plants that produce billions of tons of CO2 emissions. Many of those machines sit idle for 12 to 16 hours per day, burning electricity, but not doing any work, because businesses habitually leave their computers running overnight.

Microsoft has already announced that they will build **aggressive, energy-saving technology** into their next operating system, **Vista**. But that's not enough. These days, most computers are networked and can accept software upgrades

The problem is not quite that simple

The problem is network presence

- "Today, billions of dollars' worth of electricity are used to keep Ethernet (and other) connected devices fully powered on at all times only for the purpose of maintaining this connectivity." (Bruce Nordman, 2007)
- Connectivity is network presence
- The need for network presence is driving PCs to be left fully powered-on at all times

Defining "network presence" is a key challenge

Network presence for IPv4

To maintain network presence a host must…

- Maintain host-level reachability (respond to ARP requests)
- Maintain its IP address (if DHCP is used)
- Maintain its manageability (respond to ICMP such as ping)
- Support name resolution (e.g., for NetBIOS)
- Maintain application-level reachability (respond to TCP SYN)
- Preserve application state associated with network state
  - Maintain TCP connections
  - Respond to application-level requests and heartbeat message
- Wake-up only when its full resources are needed

The idea of a proxy

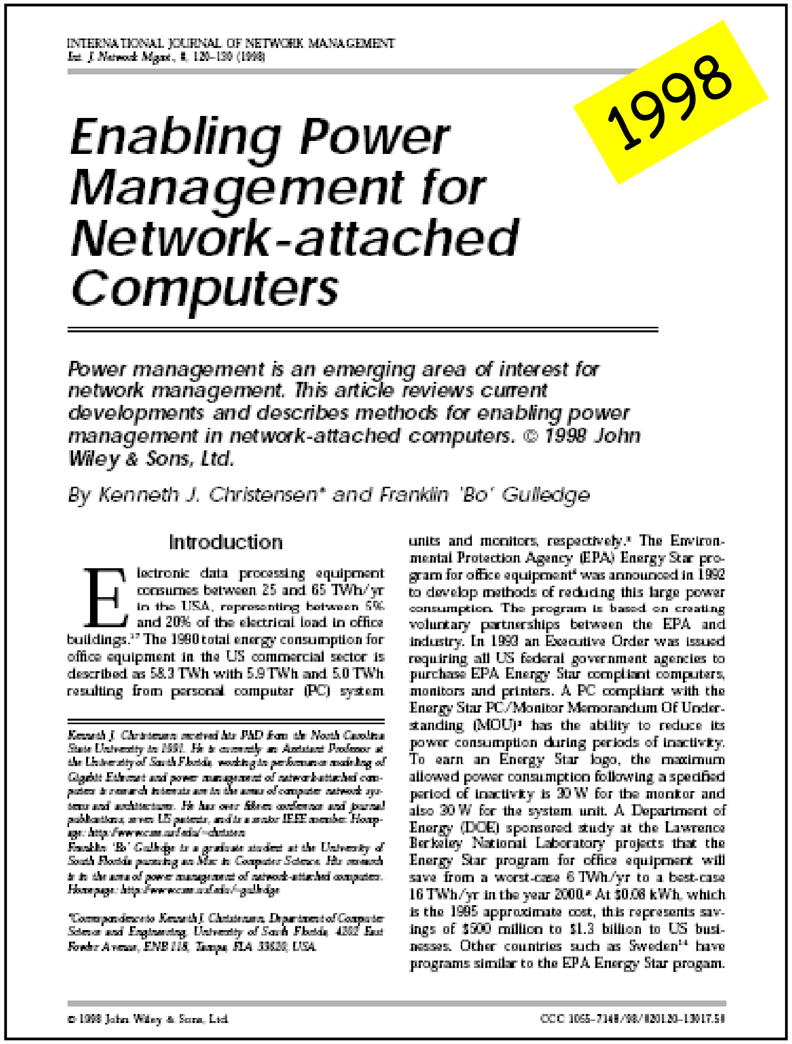
We can address network presence by…

1) Redesigning protocols and applications

2) Encapsulating intelligence for maintaining network presence in an entity other than the core of the network devices

- The second option – that of a proxy – is best for the near-term future
- A proxy is "an entity that maintains full network presence for a sleeping device"
  - Host appears to other devices as fully operational

The first work on proxying



INTERNATIONAL JOURNAL OF NETWORK MANAGEMENT
Int. J. Network Mgmt., 8, 120–130 (1998)

1998

Enabling Power Management for Network-attached Computers

Power management is an emerging area of interest for network management. This article reviews current developments and describes methods for enabling power management in network-attached computers. © 1998 John Wiley & Sons, Ltd.

By Kenneth J. Christensen and Franklin 'Bo' Gulledge*

Introduction

Electronic data processing equipment consumes between 25 and 65 TWh/yr in the USA, representing between 5% and 20% of the electrical load in office buildings.[1-7] The 1990 total energy consumption for office equipment in the US commercial sector is described as 58.3 TWh with 5.9 TWh and 5.0 TWh resulting from personal computer (PC) system units and monitors, respectively.[8] The Environmental Protection Agency (EPA) Energy Star program for office equipment[4] was announced in 1992 to develop methods of reducing this large power consumption. The program is based on creating voluntary partnerships between the EPA and industry. In 1993 an Executive Order was issued requiring all US federal government agencies to purchase EPA Energy Star compliant computers, monitors and printers. A PC compliant with the Energy Star PC/Monitor Memorandum Of Understanding (MOU)[5] has the ability to reduce its power consumption during periods of inactivity. To earn an Energy Star logo, the maximum allowed power consumption following a specified period of inactivity is 30 W for the monitor and also 30 W for the system unit. A Department of Energy (DOE) sponsored study at the Lawrence Berkeley National Laboratory projects that the Energy Star program for office equipment will save from a worst-case 6 TWh/yr to a best-case 16 TWh/yr in the year 2000.[8] At $0.08 kWh, which is the 1995 approximate cost, this represents savings of $500 million to $1.3 billion to US businesses. Other countries such as Sweden[14] have programs similar to the EPA Energy Star progam.

Kenneth J. Christensen received his PhD from the North Carolina State University in 1991. He is currently an Assistant Professor at the University of South Florida, working in performance modeling of Gigabit Ethernet and power management of network-attached computers. Research interests are in the areas of computer network systems and architectures. He has over fifteen conference and journal publications, seven US patents, and is a senior IEEE member. Homepage: http://www.csee.usf.edu/~christen

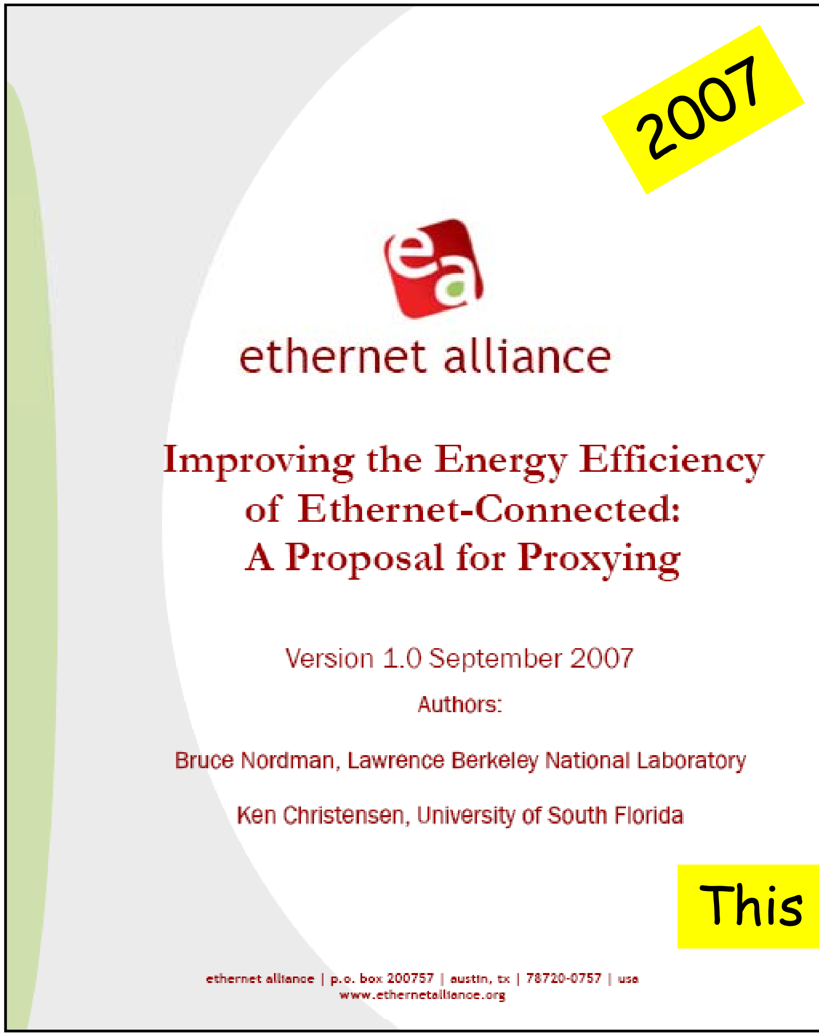
Franklin 'Bo' Gulledge is a graduate student at the University of South Florida pursuing an MSc in Computer Science. His research is in the area of power management of network-attached computers. Homepage: http://www.csee.usf.edu/~gulledge

**Correspondence to Kenneth J. Christensen, Department of Computer Science and Engineering, University of South Florida, 4202 East Fowler Avenue, ENB 118, Tampa, FLA 33620, USA.*

CCC 1055–7148/98/020120–13$17.50

- The first paper describing proxying for reducing induced energy use
- Described proxying for ARP and TCP keep-alives
- Described a centralized proxy covering for many hosts on a shared Ethernet LAN

Describing proxying to industry



2007

ethernet alliance

Improving the Energy Efficiency
of Ethernet-Connected:
A Proposal for Proxying

Version 1.0 September 2007

Authors:

Bruce Nordman, Lawrence Berkeley National Laboratory

Ken Christensen, University of South Florida

ethernet alliance | p.o. box 200757 | austin, tx | 78720-0757 | usa
www.ethernetalliance.org

- A whitepaper to bring proxying to industry folks
  - Industry folks do not read academic papers
- High-level view of proxying
  - Why we need it
  - How it might work
  - Next steps
- Ends with a proxying FAQ

This was the first step to a standard

What is the savings potential?

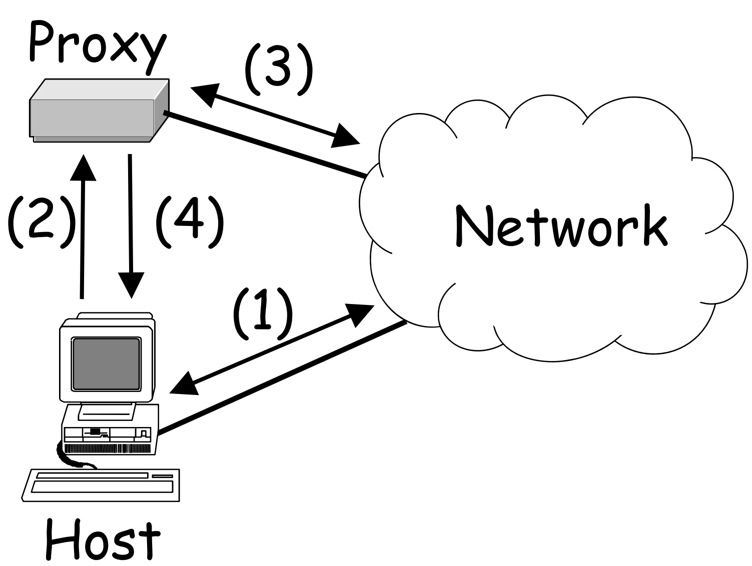
Potential for energy savings from proxying

- For desktop PCs most time is spent as on and idle
- Proxying could save *more than half* of energy used by these products (Bruce Nordman, 2007)
  - For desktop PCs most time is spent as on and idle
- Savings potential for US at $0.8 to $2.7 billion per year

High-level view of a proxy

Functional steps

1) Host awake; becomes idle
2) Host transfers state to proxy on going to sleep
3) Proxy responds to routine traffic for sleeping host
4) Proxy wakes up host as needed

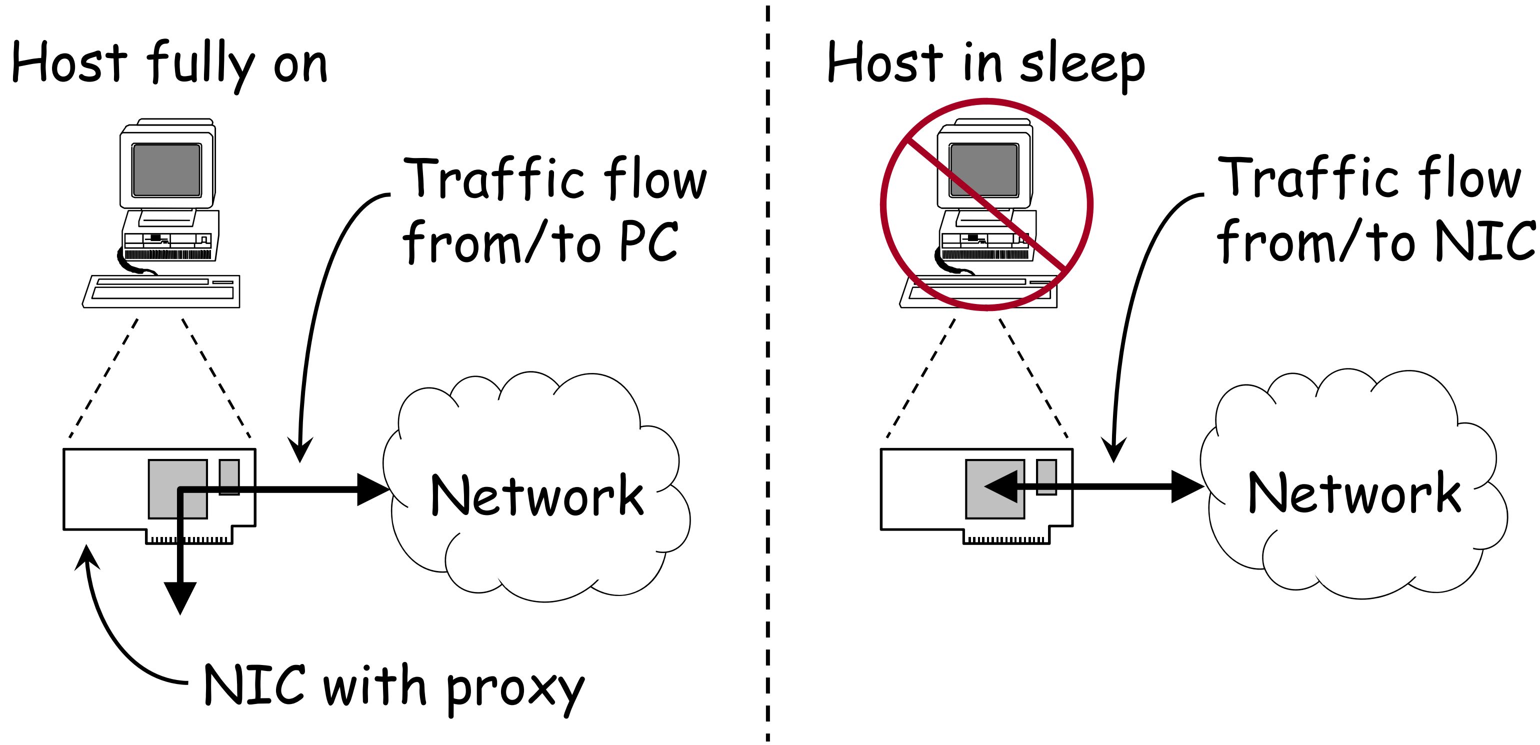


Proxy can be in separate entity, or within host NIC

The proxy in a SmartNIC

The proxy could be integrated into a NIC

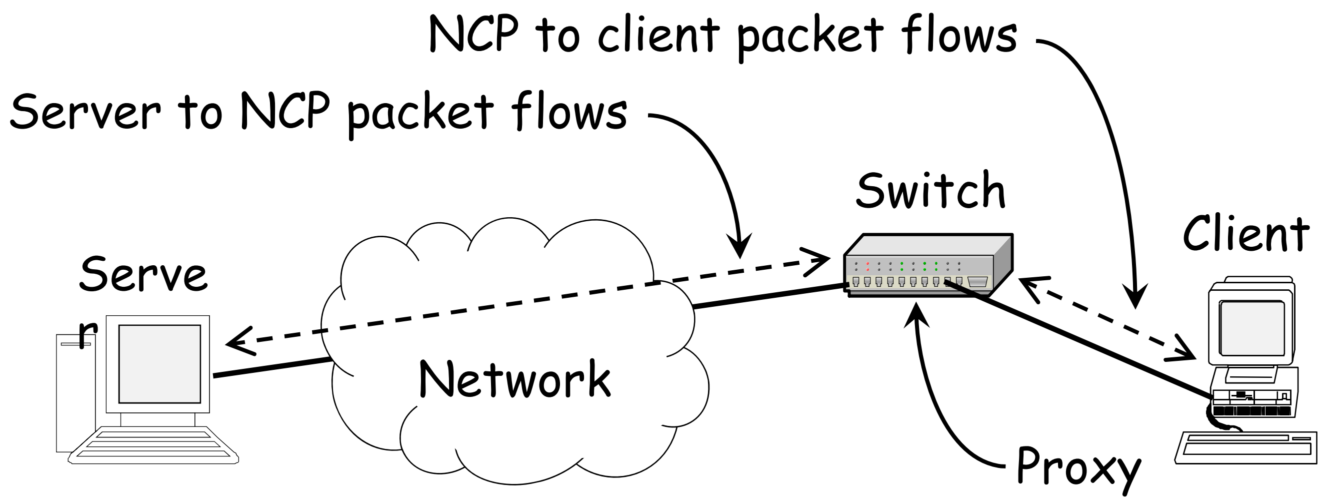
NIC is always powered-on



The proxy in a LAN switch

The proxy could be integrated into a LAN switch

LAN switch is always powered-on



Some very early work in the lab

Proxy for ARP and wake-up on valid TCP SYN

Early 2000s

From K. Christensen, P. Gunaratne, B. Nordman, and A. George, "The Next Frontier for Communications Networks: Power Management," *Computer Communications*, Vol. 27, No. 18, pp. 1758-1770, December 2004.

Goals for a proxy

This work completed in 2008

System-wide goals for a proxy

1) Host be allowed to sleep and not lose network presence
   a) Maintain IP address and remain reachable
   b) Valid TCP connection requests must be honored
   c) Existing TCP connections must not be droppd
   d) UDP packets sent to host must not be lost

2) Remote state must be maintained
   a) Application messages must be responded to

3) Changes to network applications and protocols should not be required in client or server

Requirements for a proxy

Requirements for supporting IP connectivity

1) Much smaller incremental power draw than a host
2) Know the power state of the host
3) Use IP address of the host
4) Be able to support ARP, DHCP, and ICMP
5) Be able to operate behind a NAT service

Requirements for a proxy <u>continued</u>

Requirements for supporting TCP connections

6) Be able to listen for valid TCP connection requests
   a) Be able to wake-up and enable connection to be established

7) Be able to maintain permanent TCP connections
   a) Immediately re-start data flow to host on wake-up
   b) Immediately deliver buffered TCP data to host on wake-up
   c) Close TCP connection if host has been removed

8) Be able to wake-up host when NCP buffers are nearly full

Requirements for a proxy <u>continued</u>

Supporting UDP data flows

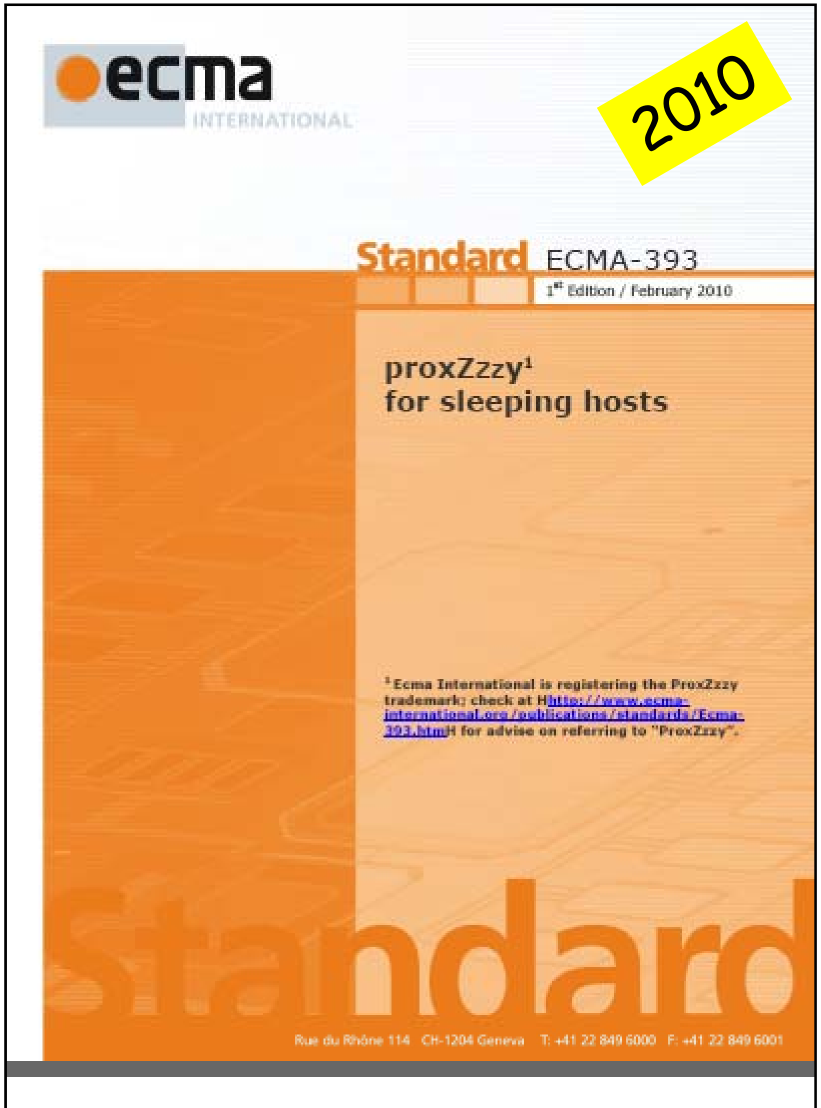
9) Be able to buffer incoming UDP packets
   a) Immediately deliver buffered UDP data to host on wake-up

Supporting applications

10) Be able to keep applications executing as if host were awake
    a) Be able to respond to routine application messages
    b) Be able to generate routine application messages

These goals and requirements fed into Ecma standards process

Ecma standard for proxying



- Standard to define proxying
- Adopted on February 2010
- Satisfies EPA Energy Star "platform-independent industry standard"
- Specifies "maintenance of network connectivity and presence by proxies to extend the sleep duration of hosts"

Does not include proxying for applications (e.g., P2P)

Ecma standard for proxying <u>continued</u>

Scope of standard

- Capabilities that a proxy exposes to a host
- Information exchanged between host and proxy
- Proxy behavior for Ethernet and WiFi
- Required and optional behavior of a proxy
  - Response to packets
  - Generating packets
  - Ignoring packets
  - Waking the host

Ecma standard for proxying <u>continued</u>

The standard was developed by

- From the press release
  - "AMD, Apple, Hitachi, HP, Intel, Lawrence Berkeley National Laboratory, Lexmark, Microsoft, Oce, Realtek, Sony, Terra Novum, and the University of South Florida collaboratively developed the ProxZzzy"

Ecma standard for proxying <u>continued</u>

Requirements Implemented	Required/Option
Media (802.3, 802.11)	Requires implementation of 6.1 or 6.2 or both
IPv4 ARP	Mandatory
IPv6 Neighbor Discovery	Mandatory
DNS	Option
DHCP	Option
IGMP	Option
MLD	Option
Remote Access using SIP and IPv4	Option
Remote Access using Teredo for IPv6	Option
SNMP	Option
Service Discovery using mDNS	Option
Name Resolution with LLMNR	Option
Wake Packets	Mandatory

Table of requirements and status

Ecma standard for proxying continued

6.3 ARP

Example of ARP

6.3.1 Configuration Data

ID	Configuration Data	Observation
C5	IP Address	IP addresses of interfaces to be proxied

6.3.2 Behavioural Requirements

ID	Requirement	M/S/O	Rationale
R18	The proxy SHALL properly respond to ARP requests received on the MAC layer broadcast address.	M	ARP operations require that the proxy be capable of receiving ARP requests (RFC 826, 1122).
R19	The proxy SHALL properly respond to ARP requests received on proxied MAC unicast addresses.	M	ARP operations require that the proxy be capable of receiving ARP requests (RFC 826 and RFC 1122).
R20	The proxy SHALL reply to ARP Requests where the target protocol address is a proxied IP address with ARP Responses.	M	RFC 5227
R21	The proxy SHALL reply to ARP Requests containing source protocol address as zero, with an ARP Response.	M	This is to support ARP probes for Address Conflict Detection. RFC 5227

Proxying in EPA Energy Star

EPA Energy Star for Computers, Version 5.0

- "Proxying refers to a computer that maintains Full Network Connectivity as defined in Section 1 of this specification. For a system to qualify under the proxying weightings above, it must meet a non-proprietary proxying standard that has been approved by the EPA and the European Union as meeting the goals of ENERGY STAR."*

The Ecma standard is key to this

* From ENERGY STAR® Program Requirements for Computers, Version 5.0, EPA, 2009.

Proxying in products

Apple Snow Leopard

- "Wake on Demand. This is Apple's name for a new networking feature that lets a Snow Leopard Mac go to sleep while a networked base station continues to broadcast Bonjour messages about the services the sleeping computer offers."

Bonjour Sleep Proxy, supports ARP, file and print serving, and SSH login initiation.

From "Wake on Demand lets Snow Leopard Sleep with One Eye Open," MacWorld, August 28, 2009

Standard does not address applications

What about application-level proxying?

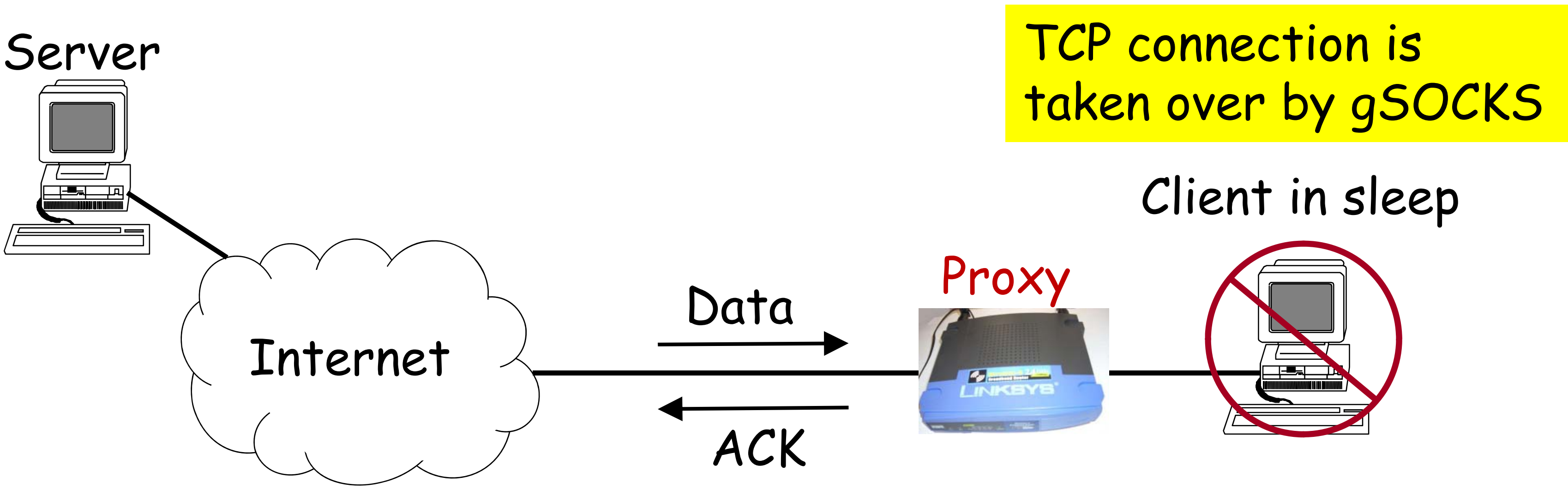
- Applications addressed in requirements
- Applications are not covered in the standard
- Applications that induce energy use include
  - Applications with TCP connections (e.g., IM and SSH)
  - P2P file sharing
  - SIP
  - UPnP
  - Others

We have addressed the above

Recent work in the lab

Proxying for TCP connecitions

- Used a modified SOCKS in a Linksys router
  - Call this "green SOCKS" (or gSOCKS)
- Works well for SSH and IM connections



Proxying for TCP connections

Hacked a small router

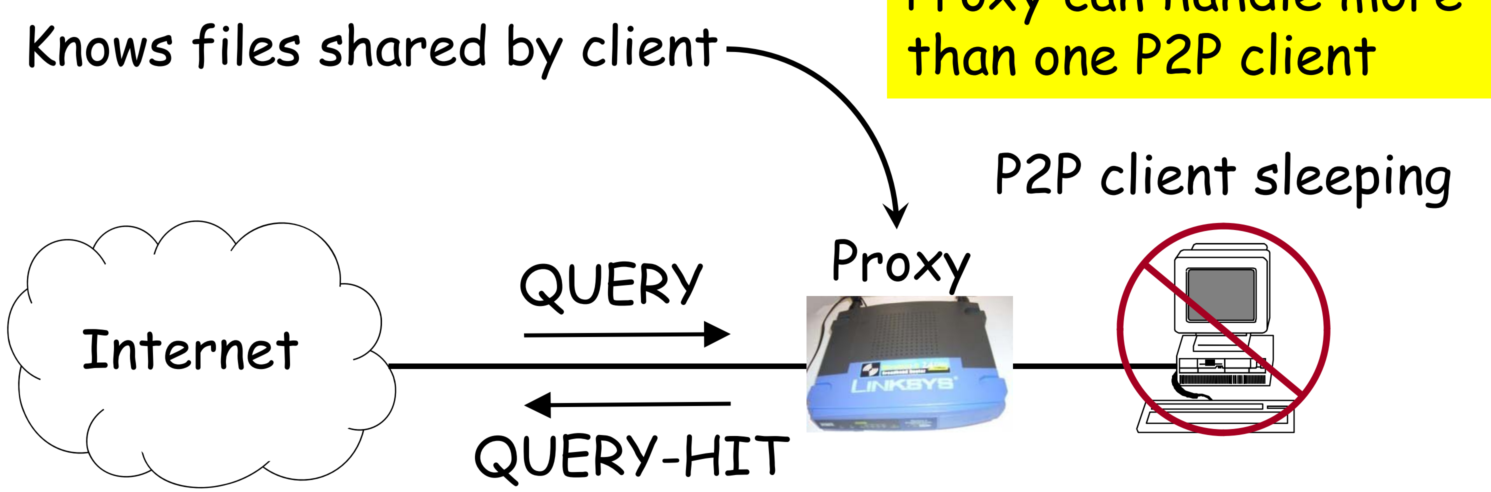
- Linksys WRT54G SOHO router with OpenWRT
- Maintains client-server TCP connections using a modified SOCKS
- Listens for messages from host
  - Two messages: "Going to sleep" and "Now awake"

Even more recent work in the lab

Proxying for Gnutella P2P connections

- Uses previous gSOCKS TCP connection proxy
- Also handles QUERY messages (sends QUERY-HIT)

Proxy can handle more than one P2P client



Other recent work – the SIP Catcher

Proxying for SIP IP Phones

- A IP phone is 10 to 20 W (so, $10 to $20 per year)
- Also have products to replace home landlines with PC and broadband
  - PC must be left-on 24/7 for incoming calls!

The SIP Catcher

Estimated energy use of IP Soft Phones

- About 20 million users (Magic Jack and other) in the US
  - Assume 70 W per PC
  - Estimate that 60% are left on 24/7 because of Magic Jack
  - This is 7.4 TWh per year

- Assume 50% can be proxied for 10 hours per day

- Savings of about 1.5 TWh per year possible
  - This is about $150 million per year

These are conservative estimates

The SIP Catcher <u>continued</u>

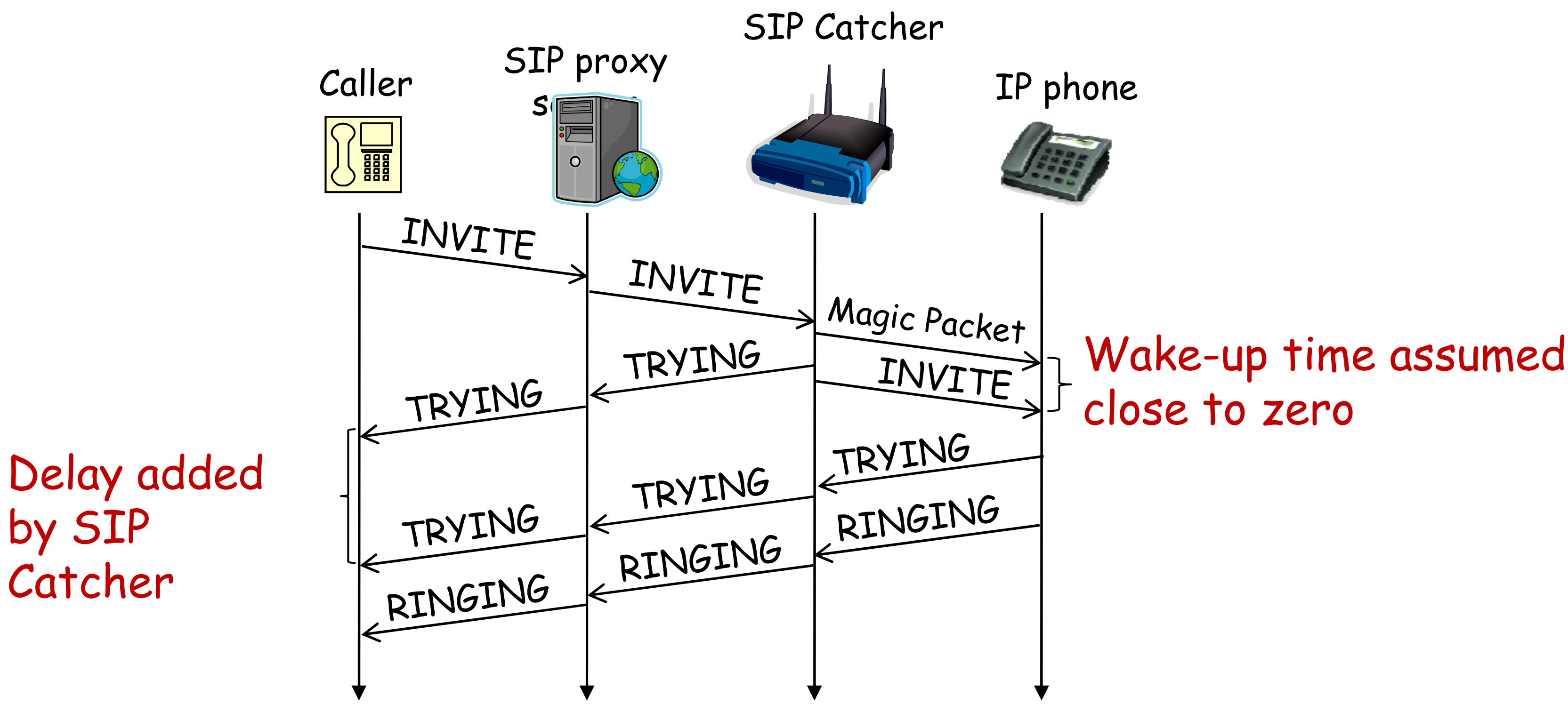
Developed a prototype SIP Catcher

- Linksys WRT54G SOHO router
  - Uses OpenWrt firmware
- IP phone represented by PC soft-phone from SJ-Labs
  - Runs SIP, connects to free SIP service provider
- NCP control point in Linksys router and PC
  - Same as used for handling TCP connections

The SIP Catcher <u>continued</u>

SIP call being proxied by SIP Catcher

1) Wakes up phone when call detected
2) Responds on behalf of phone (TRYING message)
3) Forwards INVITE to phone when it is awake



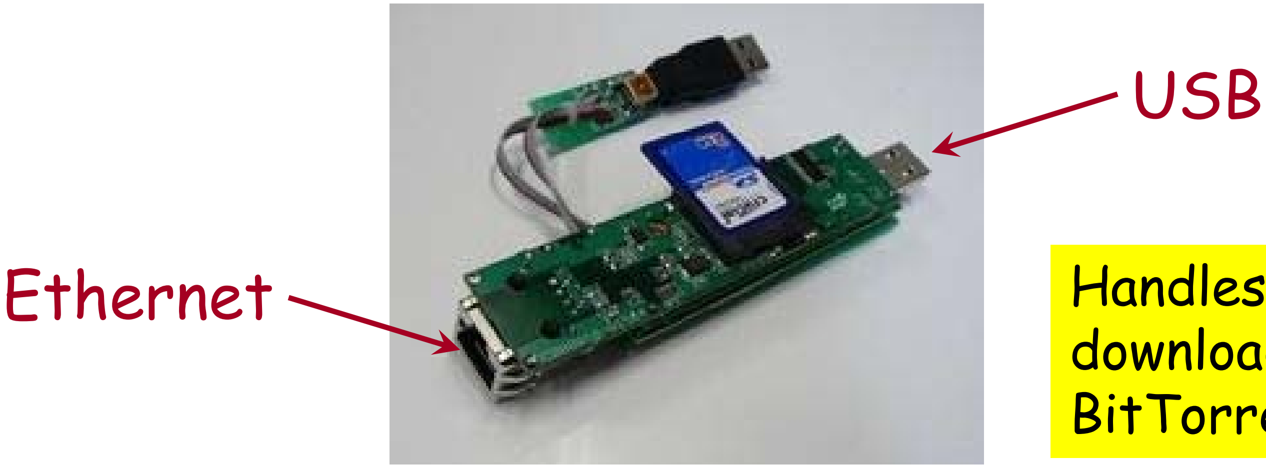
The SIP Catcher continued

On YouTube

From the lab of other folks

Somniloquy (Yuvraj Agarwal, UCSD)

- "Small USB-connected hardware and software plug-in system that allows a PC to remain in sleep mode while continuing to maintain network presence and run well-defined application functions"



Handles IM, web downloads, and BitTorrent downloads

From Y. Agarwal, S. Hodges, J. Scott, R. Chandra, P. Bahl, R. Gupta, Somniloquy: Augmenting Network Interfaces to Reduce PC Energy Usage, *Proceedings of the 6th USENIX Symposium on Networked Systems Design USENIX Association and Implementation (NSDI)*, pp. 365-380, April 2009.

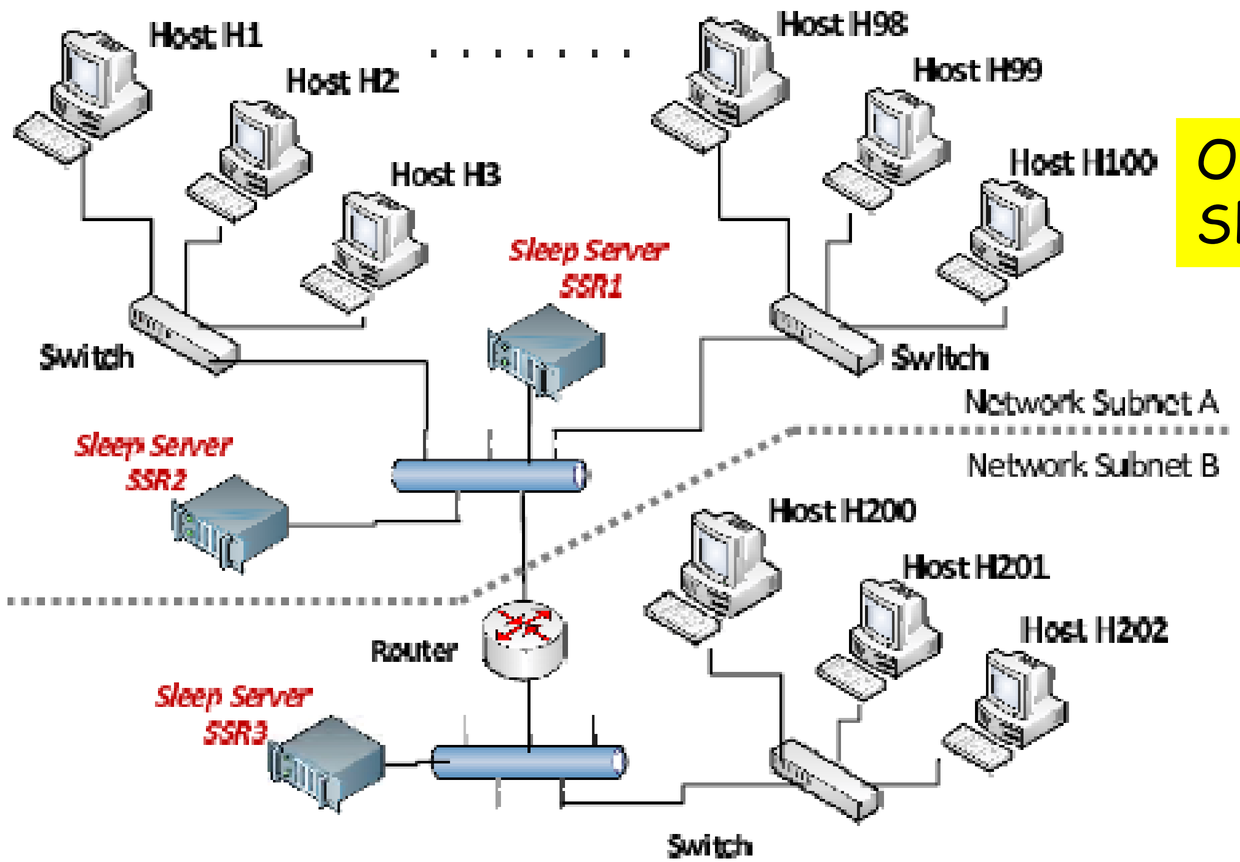
More from the lab of other folks

SleepServer (Yuvraj Agarwal, UCSD)

- A server-based solution that exploits virtualization
- Image of all PCs in a network stored on SleepServer
- PC goes to sleep, image in SleepServer takes over
- Is a counterpart to Somniloquy work
  - No new hardware
  - Centralized, not distributed

SleepServer

SleepServer architecture – network view

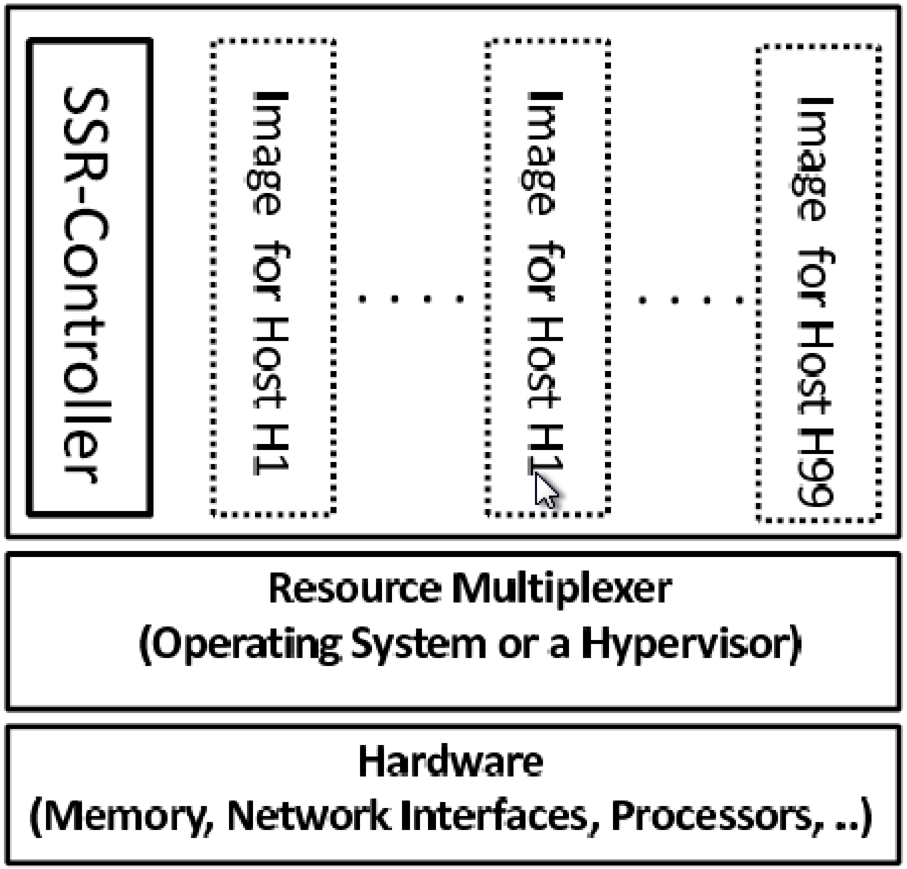


One or more SleepServers

From Y. Agarwal, S. Savage, and R. Gupta, "SleepServer: Energy Savings for Enterprise PCs by Allowing them to Sleep," *Proceedings of the USENIX Annual Technical Conference*, June 2010.

SleepServer <u>continued</u>

SleepServer architecture – SleepServer view



For each host proxied there is a VM image instantiated.

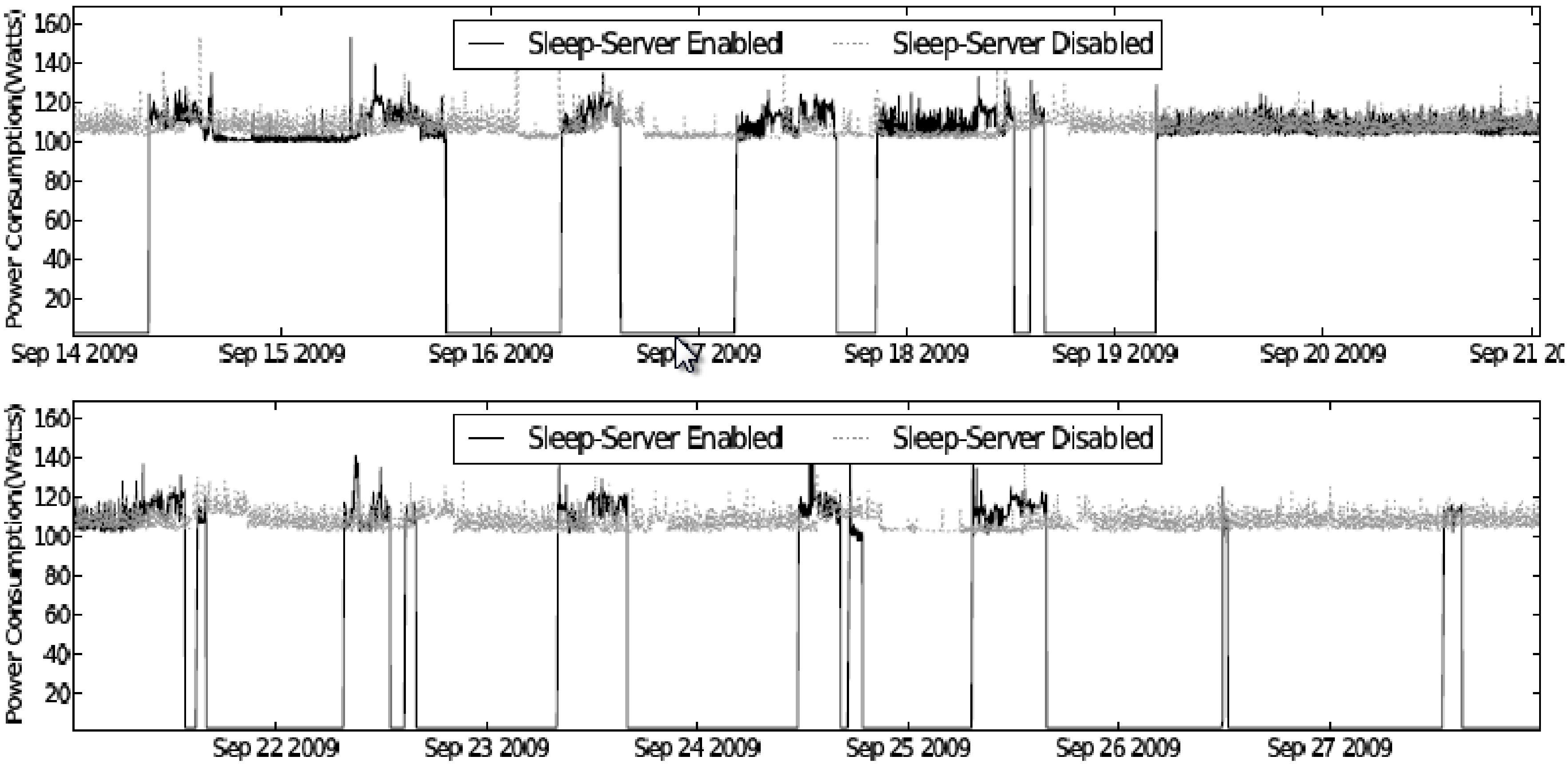
Network parameters of each image match that of the host

Each host has an agent (SSR-Client) used to communicate with the SSR

From Y. Agarwal, S. Savage, and R. Gupta, "SleepServer: Energy Savings for Enterprise PCs by Allowing them to Sleep," *Proceedings of the USENIX Annual Technical Conference*, June 2010.

SleepServer continued

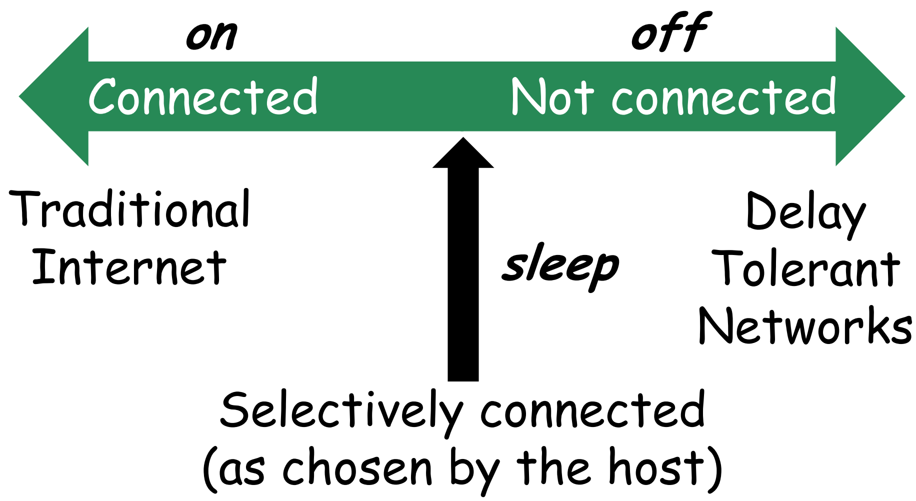
Energy savings experiments using SleepServer



From Y. Agarwal, S. Savage, and R. Gupta, "SleepServer: Energy Savings for Enterprise PCs by Allowing them to Sleep," *Proceedings of the USENIX Annual Technical Conference*, June 2010.

Next steps – architectural primitives

An entirely new view of selective connectivity



A proxy

- Assistants
- Exposing state
- Evolving state
- Host-based control
- Application primitives
- Security

From M. Allman , K. Christensen, B. Nordman, and V. Paxson, "Enabling an Energy-Efficient Future Internet through Selectively Connected End Systems,“ *Sixth Workshop on Hot Topics in Networks (HotNets-VI)*, November 2007.

Summary for proxying

Proxying can save a lot of energy waste

- Proxying is a means of reducing induced energy use
  - Enormous energy savings potential
- Moved from the lab to standards
  - Ecma standard for IPv4 and IPv6 protocols
- Now in products from Apple (Bonjour)
- More work needed "in the lab" to address applications
- Even more work needed to generalize proxying into architectural primitives for a future Internet

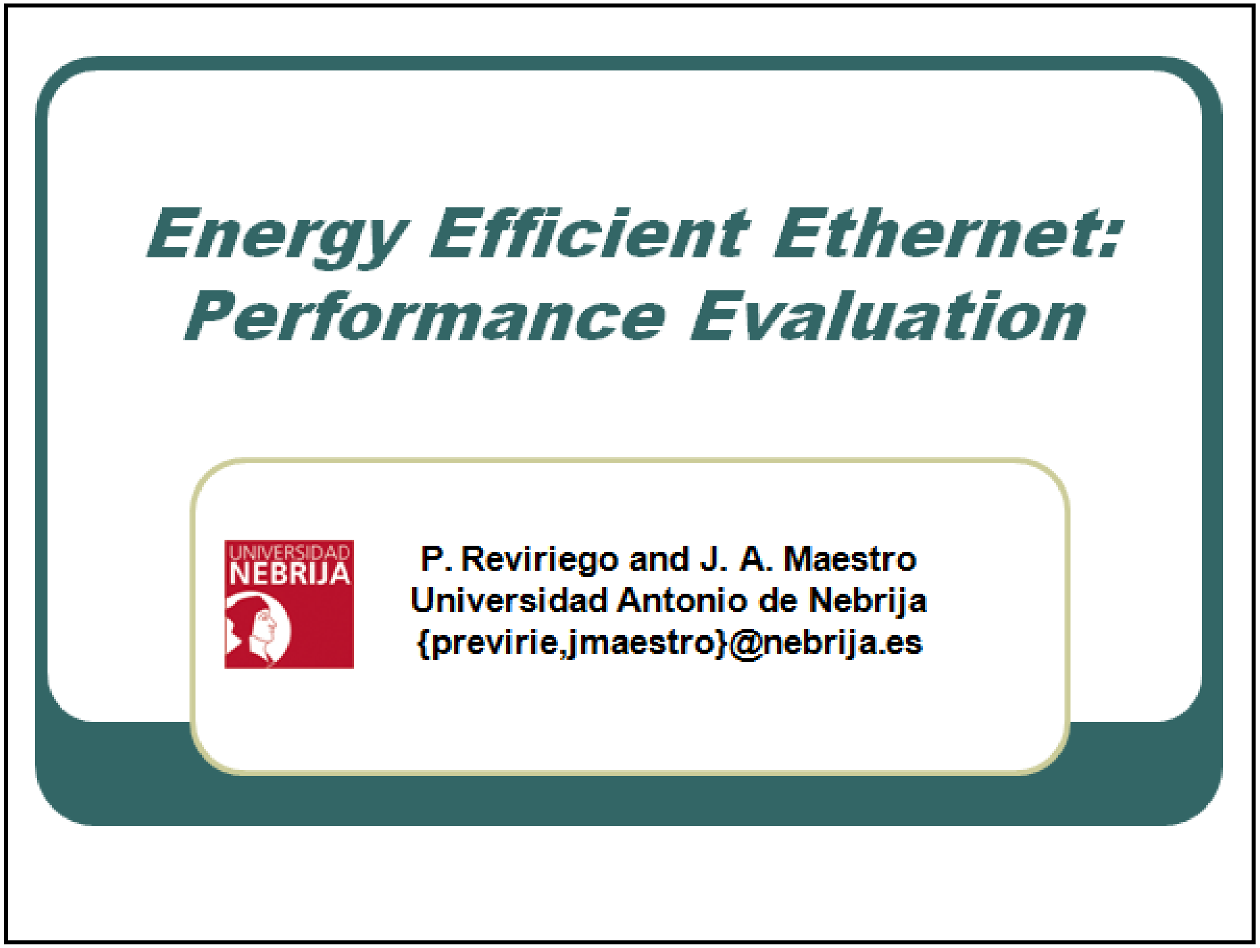
Reducing direct energy consumption

Can we reduce energy used by Ethernet?

... this is Energy Efficient Ethernet

Energy Efficient Ethernet

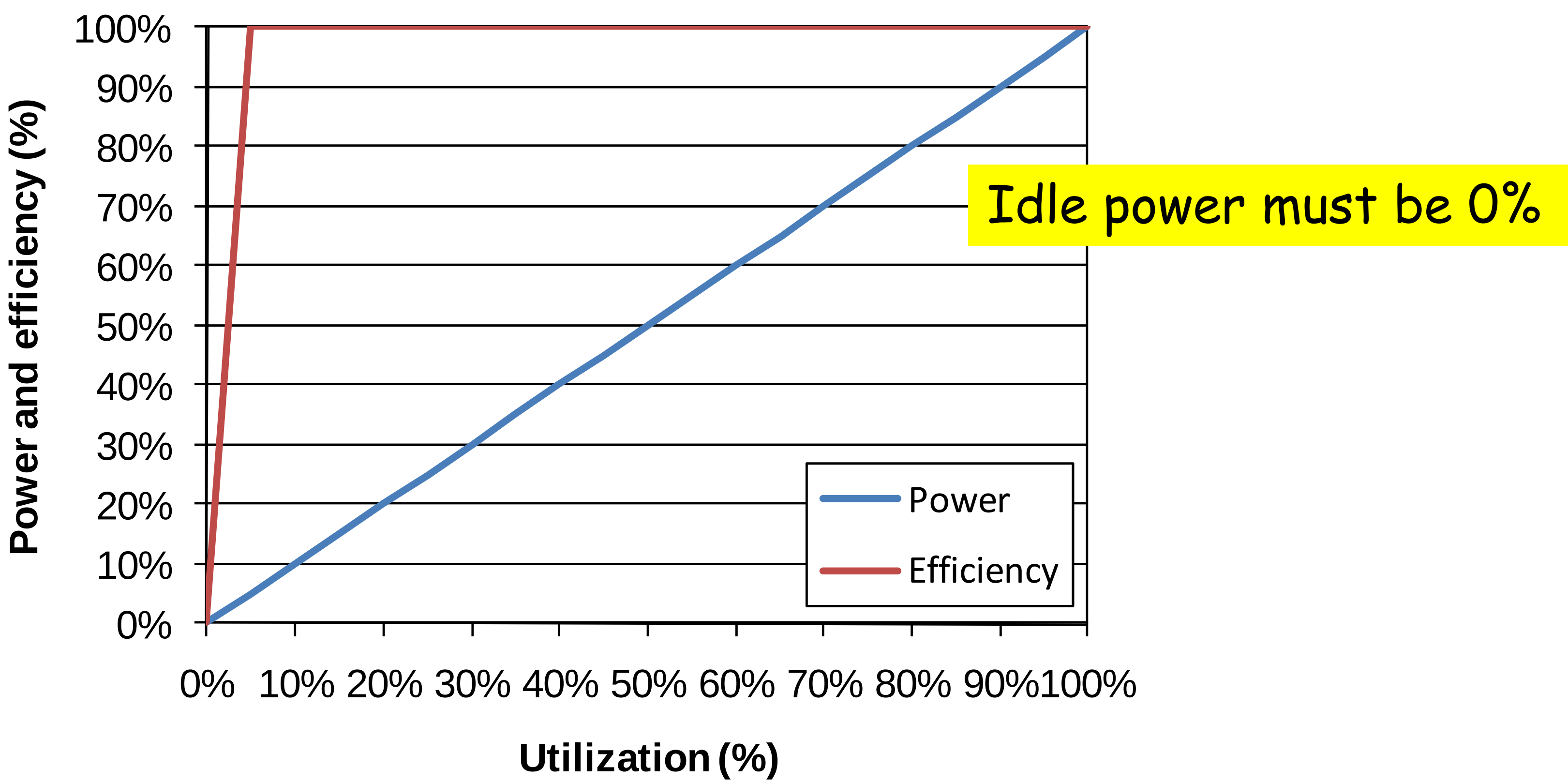
EEE presented in a UC3M previous seminar



The goal is energy-proportional

We seek energy-proportional computing

- Define efficiency as power divided by utilization



Adapting link data rate to load

Moving toward energy-proportional links

- Links are typically lightly utilized and will stay that way
  - See Odlyzko and others
- When link utilization is low, do not need "high bandwidth"
- Lower data rates consume less power
- Idea is to explore if and how links could adapt their data rate to load
  - High data rate for high load
  - Low data rate for low load (most of the time!)

Open questions in adapting to load

There are many open questions

- What is the *mechanism* for adapting to load?
  - How is the link data rate changed
- What is the *policy* for adapting to load?
  - When is the link data rate changed
- What about the delay and loss for switching between rates?
- What about oscillation – is it stable?
- Fundamentally, what is the trade-off between energy savings and performance?

Adaptive link rate (ALR) for Ethernet

Early to mid-2000s

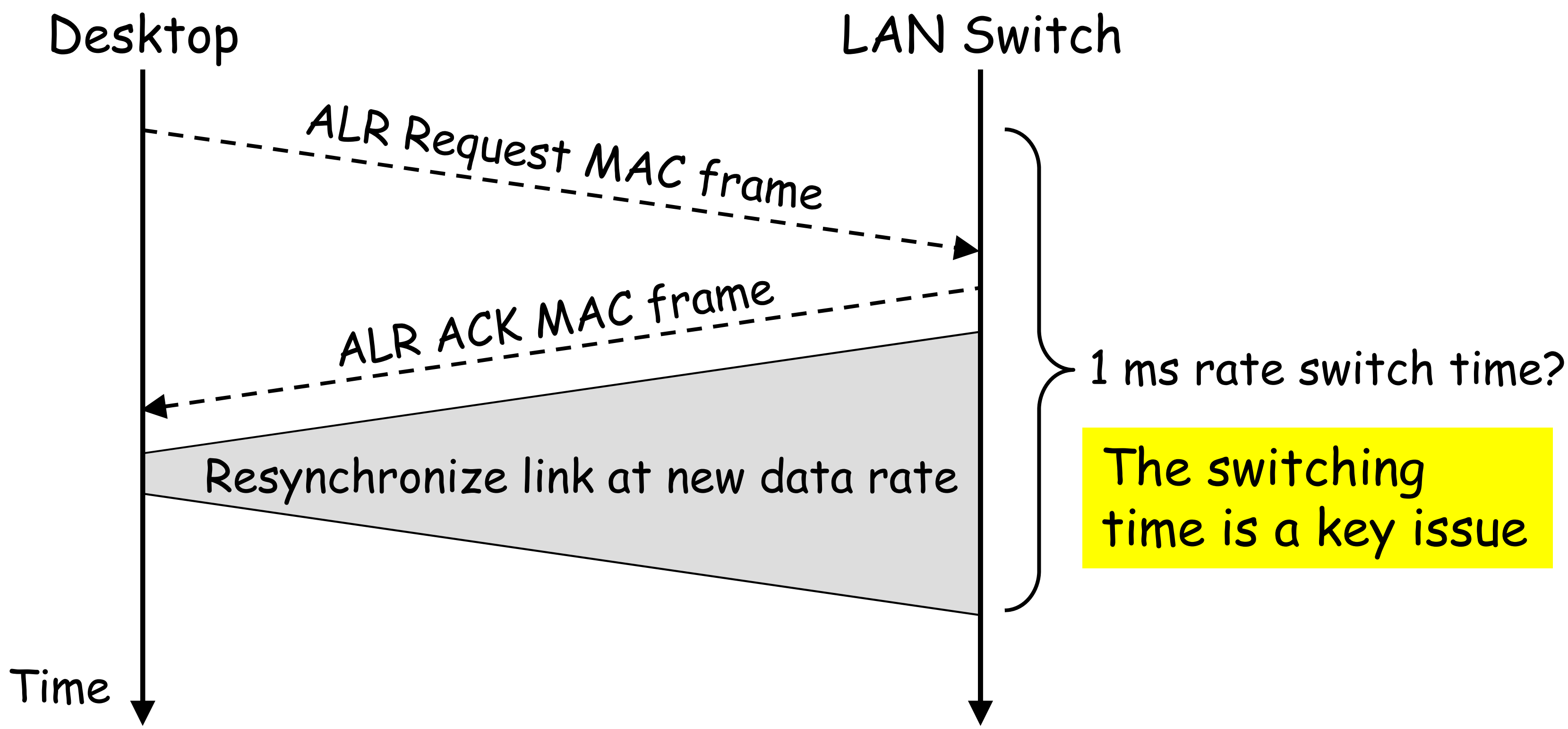
Goal: Save energy by matching link data rate to utilization

- **Change (adapt) data rate in response to utilization**
  - Use 10 or 100 Mb/s PHY during low utilization periods
  - Use 1 or 10 Gb/s PHY during high utilization periods
  - No PHY layer changes needed!
- **Need new *mechanism***
  - Current auto-negotiation is not suitable (100s of ms)
- **Need *policies* for use of mechanism**
  - *Reactive policy* possible if can switch link rates "quickly"
  - *Predictive policy* is needed otherwise

One possible ALR mechanism

Use a MAC frame handshake between ends

ALR must be supported in both ends

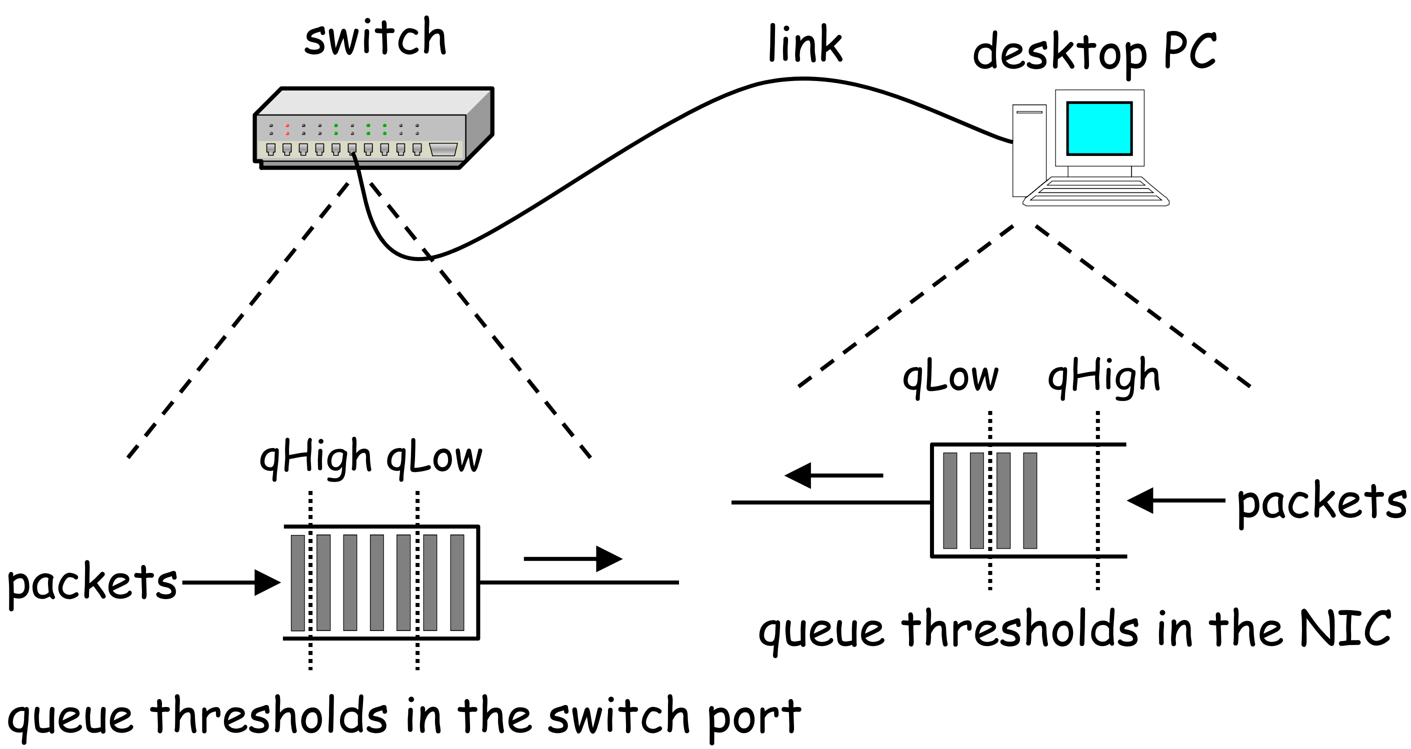


The switching time is a key issue

One possible ALR policy

Dual-threshold policy

- If queue is above qHigh then switch to high rate
- If queue is below qLow then switch to low rate



A lot of work done with ALR

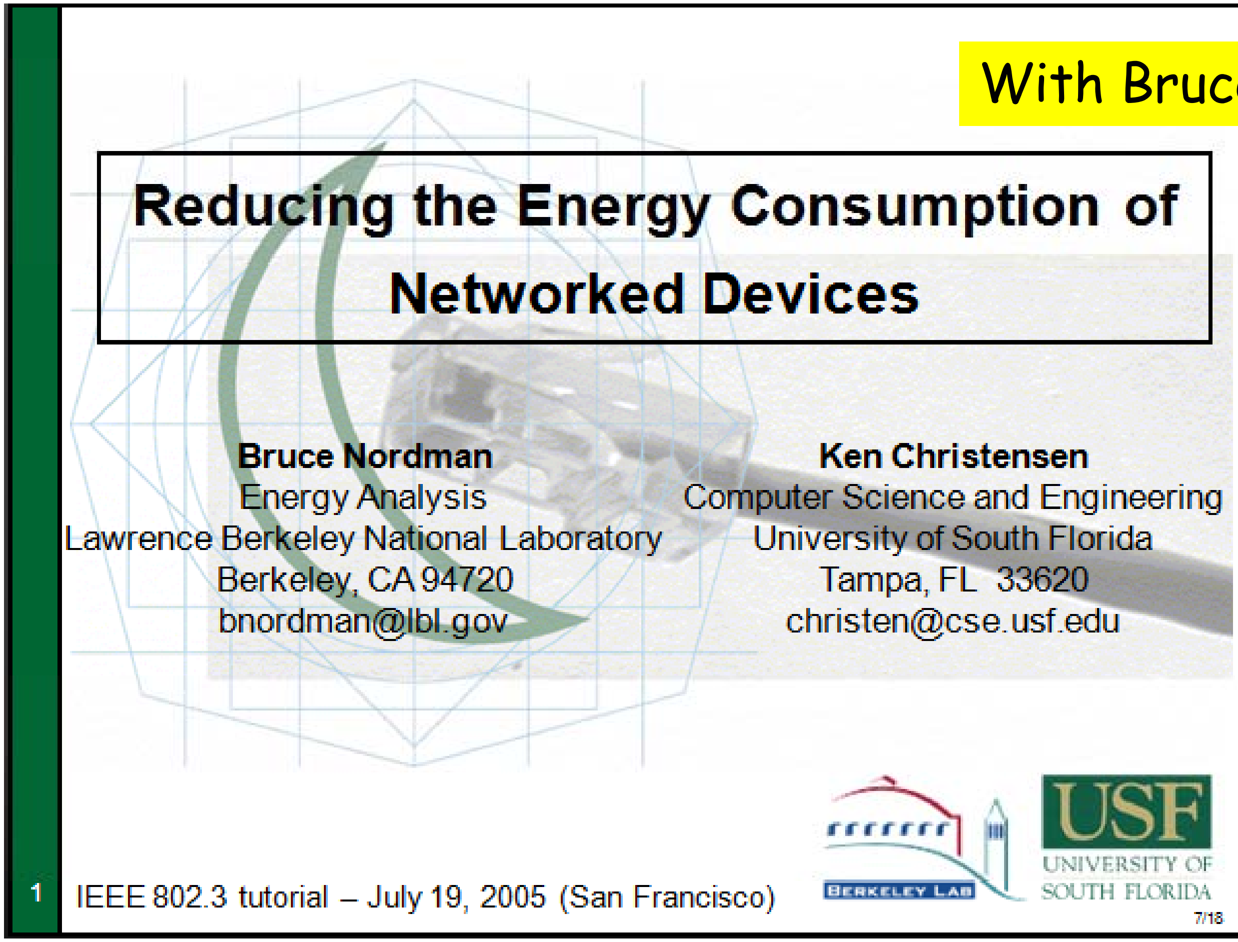
We did a lot of work with ALR

- Studied performance of ALR policies
- Effect of switching time studied
- Simulation and analytical models built
- Published findings in *IEEE Transactions on Computers*
- However, ALR was not adopted by IEEE 802.3
  - Issues with switching time
  - Issues with complexity of a mechanism

ALR and IEEE 802.3

ALR presented to IEEE 802.3 in July 2005

With Bruce Nordman



ALR and IEEE 802.3 <u>continued</u>

- **Adaptive Link Rate to IEEE 802.3 in 2005**
  - A Study Group was formed
  - Mike Bennett from LBNL is the chair
- **Became "Energy Efficient Ethernet"***
  - IEEE 802.3az task force
- **ALR became RPS, which then became LPI**
- **Standard expected to be approved in late 2010**
- **Vendors are now sampling products (based on LPI)**
  - Broadcom and Realtek

* Logo by Glen Kramer of Teknovus, Inc. (full permission for use granted via email dated January 27, 2007)

EEE in EPA Energy Star

EPA Energy Star for Computer Servers, Tier 2

- "Energy Efficient Ethernet: All physical layer Ethernet in servers covered by the Computer Server specification must meet the Energy Efficient Ethernet (IEEE 802.3az) standard upon its approval by the IEEE."*

ENERGY STAR

To be in computer (PC) spec later

* From ENERGY STAR® Version 1.0 Program Requirements for Computer Servers, Tier 2: PRELIMINARY

How did we get to LPI from ALR

ALR handshake was deemed complicated

Summary

- Lost speed change REQ, ACK or NACK
  - Very rare event
  - But need to handle with minimum disruption
- Do we really need a ACK
  - Let the link drop be the ACK
- Timeout, repeat N times or keep repeating
  - REQ packets
    - Resend, sending nothing otherwise
  - NACK packets
    - Send once then if REQs continue send again
    - Minimum loss of bandwidth for NACKer

IEEE 802.3 EEE SG – September 2007 Interim Meeting Page 12

From David Law at IEEE 802.3az meeting

From D. Law, "Packet loss in protocol based speed change," September 2007.

How did we get to LPI… continued

ALR had issues with switching time

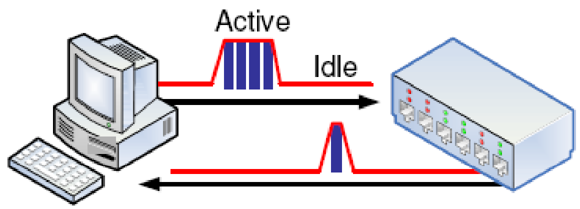
- How much switching time is acceptable?
  - During switching time link is stalled
  - Packets will be delayed and may be lost
- Switching time needed may be many seconds
  - To retrain 10 Gb/s links
  - This is clearly unaccetable
- Thus, PHY changes may be needed in any case

EEE is based on Low Power Idle (LPI)

Slide from November 2007 IEEE 802.3az meeting…

7

Active/Idle Toggling with 0BASE-x Concept



- Principle: Transmit data at fastest rate then return to idle
  - Energy savings come from power cycling between active/idle states
- Active/Idle toggling could be used *instead* of PHY rate shifting
  - Offers the best energy efficiency on links with lower utilization
  - Integrates well with existing PC power management schemes (e.g. ACPI)
  - Clock & power gating (on/off) is easier than rate shifting
- Asymmetrical operation would provide even better energy efficiency
  - Each direction could enter active & idle states independently
  - Most end-node traffic is heavily weighted toward either send or receive
  - Tx & Rx data paths already operate independently above the PHY

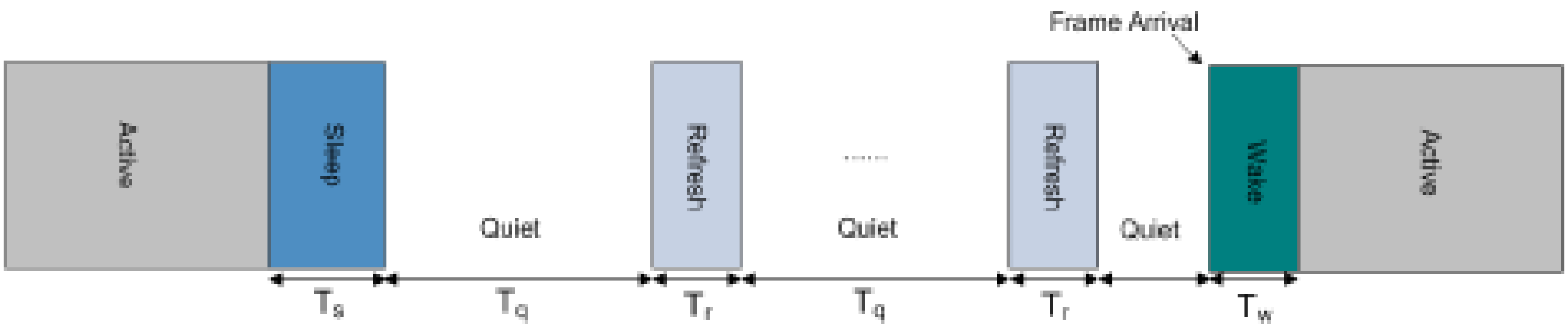
Energy Efficient Ethernet

intel

The better idea from a 2007 IEEE 802.3az meeting

How LPI works

- **Between packets the PHY "goes to sleep"**
  - Sleep is idle = about 10% of full power
  - Periodic refreshes to keep synchronized
- **LPI has wake-up and sleep transitions**
  - First packet after an idle incurs a wake-up transition
  - After last packet in a burst a go to sleep transition



From P. Reviriego, J. Hernandez, D. Larrabeiti, and J. Maestro, "Performance Evaluation of Energy Efficient Ethernet," *IEEE Communications Letters*, Vol. 13, No. 9, pp. 1-3, September 2009.

LPI overhead

LPI has overhead from T_w and T_s

- Can measure frame efficiency for single packet case

$$Efficiency = \frac{T_{Frame}}{T_{Frame} + T_w + T_s}$$

Protocol	Min T_w (μs)	Min T_s (μs)	T_{Frame} (1500B) (μs)	Frame eff.	T_{Frame} (150B) (μs)	Frame eff.
100Base-Tx	30	100	120	48%	12	8.5%
1000Base-T	16	182	12	5.7%	1.2	0.6%
10GBase-T	4.16	2.88	1.2	14.6%	0.12	1.7%

From P. Reviriego, J. Hernandez, D. Larrabeiti, and J. Maestro, "Performance Evaluation of Energy Efficient Ethernet," *IEEE Communications Letters*, Vol. 13, No. 9, pp. 1-3, September 2009.

Performance evaluation of EEE

IEEE COMMUNICATIONS LETTERS, VOL. 13, NO. 9, SEPTEMBER 2009 1

Performance Evaluation of Energy Efficient Ethernet

P. Reviriego, J. A. Hernández, D. Larrabeiti, and J. A. Maestro

- The first published work on EEE performance evaluation
- “The results show that although EEE improves the energy efficiency, there is still potential for substantial further energy savings as in many cases most of the energy is wasted in waking up and sleeping the link.”

More EEE performance evaluation

IEEE 802.3az: The Road to Energy Efficient Ethernet

Authors: K. Christensen, P. Reviriego, B. Nordman, M. Bennett, M. Mostowfi, and J.A. Maestro

- Submitted to IEEE Communications magazine in March 2010
  - For their special issue on Green Communications
- CSIM models by me (and student, Mehrgan Mostowfi) and ns-2 models by Pedro Reviriego and Juan Maestro
- Energy savings from Bruce Nordman
- History of IEEE 802.3az from Mike Bennett (chair task force)

CSIM EEE model

Developed key models in CSIM

- CSIM is a process oriented simulation engine
  - A C function library
  - From Mesquite Software

- Simple single-server queue model with EEE added
  - Customers have deterministic service time

- Adds a T_WAKE delay for first packet to leave queue

- Adds a T_SLEEP delay for last packet to leave queue
  - "last packet" means queue is now empty

EEE model experiment

Ran an experiment for 10 Gb/s

- For 10 Gb/s
  - T_WAKE = 4.16 μs
  - T_SLEEP = 2.88 μs
  - For 1500 byte packet service_time = 1.2 μs

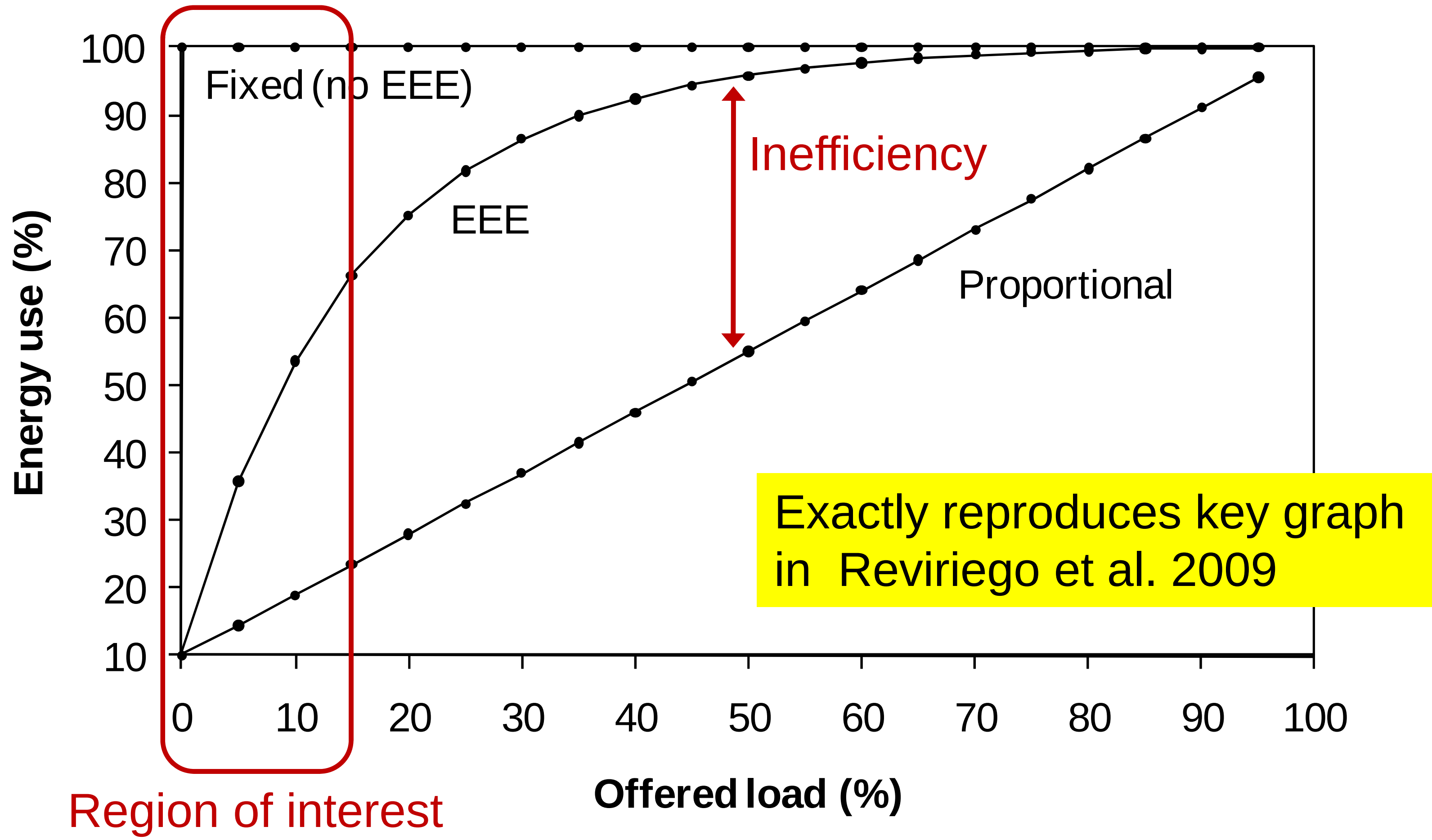
- Assume that idle power use is 10% of full power use

- Vary offered load from 0% to 95%
  - Poisson arrivals, fixed length packet

- Measure link utilization
  - Note that link utilization will be greater than offered load due to EE overhead

EEE model results

Results for 10 Gb/s



Exactly reproduces key graph in Reviriego et al. 2009

Need to “fix” inefficiency

Idea – packet coalescing to improve efficiency

- Coalescing will reduce EEE overhead
  - More packets per T_WAKE and T_SLEEP overhead
- Trade-offs are
  - Added packet delay
  - Increased burstiness of departure process

FSM for coalescing

Specify coalescing operation with an FSM

- The FSM has two states: ON and OFF
  - In OFF state generated packets are buffered, but not sent
  - In ON state packets are sent
    - » Packets in buffer at time of entry into state are sent first

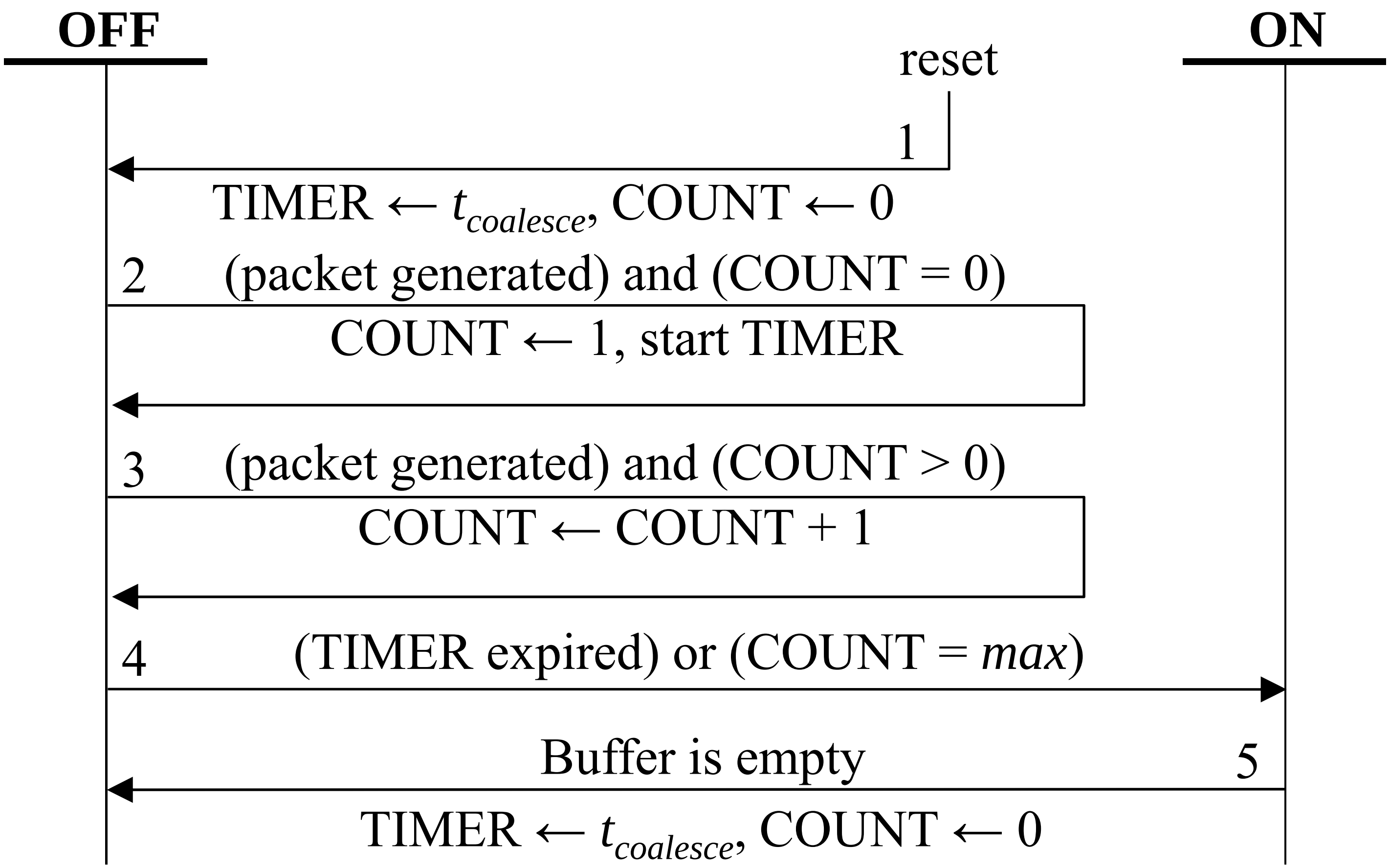
- Key variables

TIMER	Timer for coalescing
COUNT	Packet counter for coalescing
$t_{coalesce}$	Initial timer value for TIMER
max	Maximum count for generated packets

FSM for coalescing <u>continued</u>

The FSM

Only when buffer is empty does transition (5) occur. Thus, more than COUNT packets can be sent each time the ON state is entered.



CSIM model for coalescing

CSIM model

- More complicated than EEE model
- Uses a separate process for the coalescer
- Uses CSIM "wait" event – event is set by a time-out or when coalescer capacity is reached

EEE with coalescing experiment

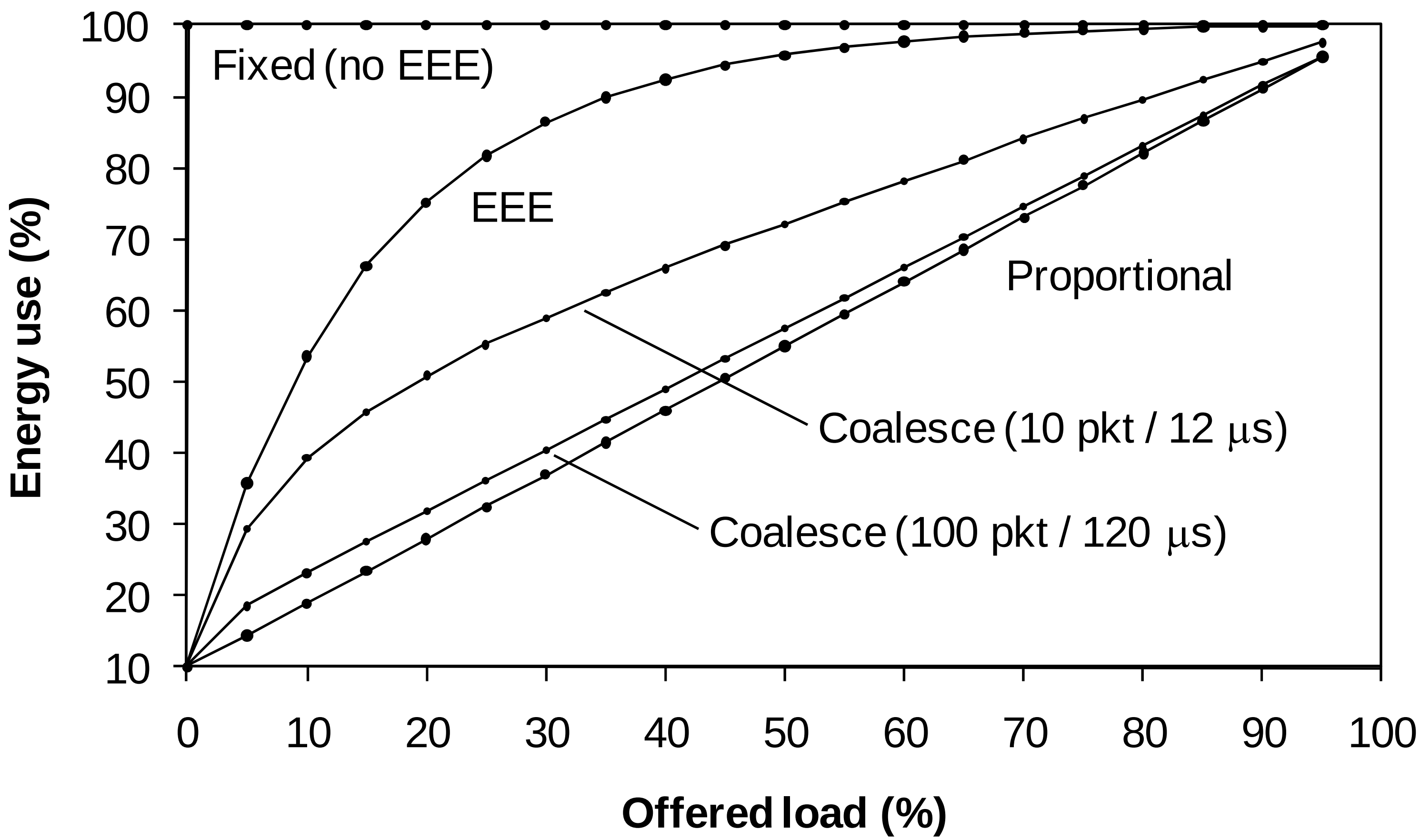
Repeat previous 10 Gb/s experiment

- For 10 Gb/s
  - T_WAKE = 4.16 μs
  - T_SLEEP = 2.88 μs
  - For 1500 byte packet service_time = 1.2 μs
- For coalescing
  - max = 10, $t_{coalese}$ = 12 μs
  - max = 100, $t_{coalesce}$ = 120 μs
- Assume that idle power use is 10% of full power use
- Vary offered load from 0% to 95%
  - Poisson arrivals, fixed length packet

EEE with coalescing results

Results for 10 Gb/s with coalescing

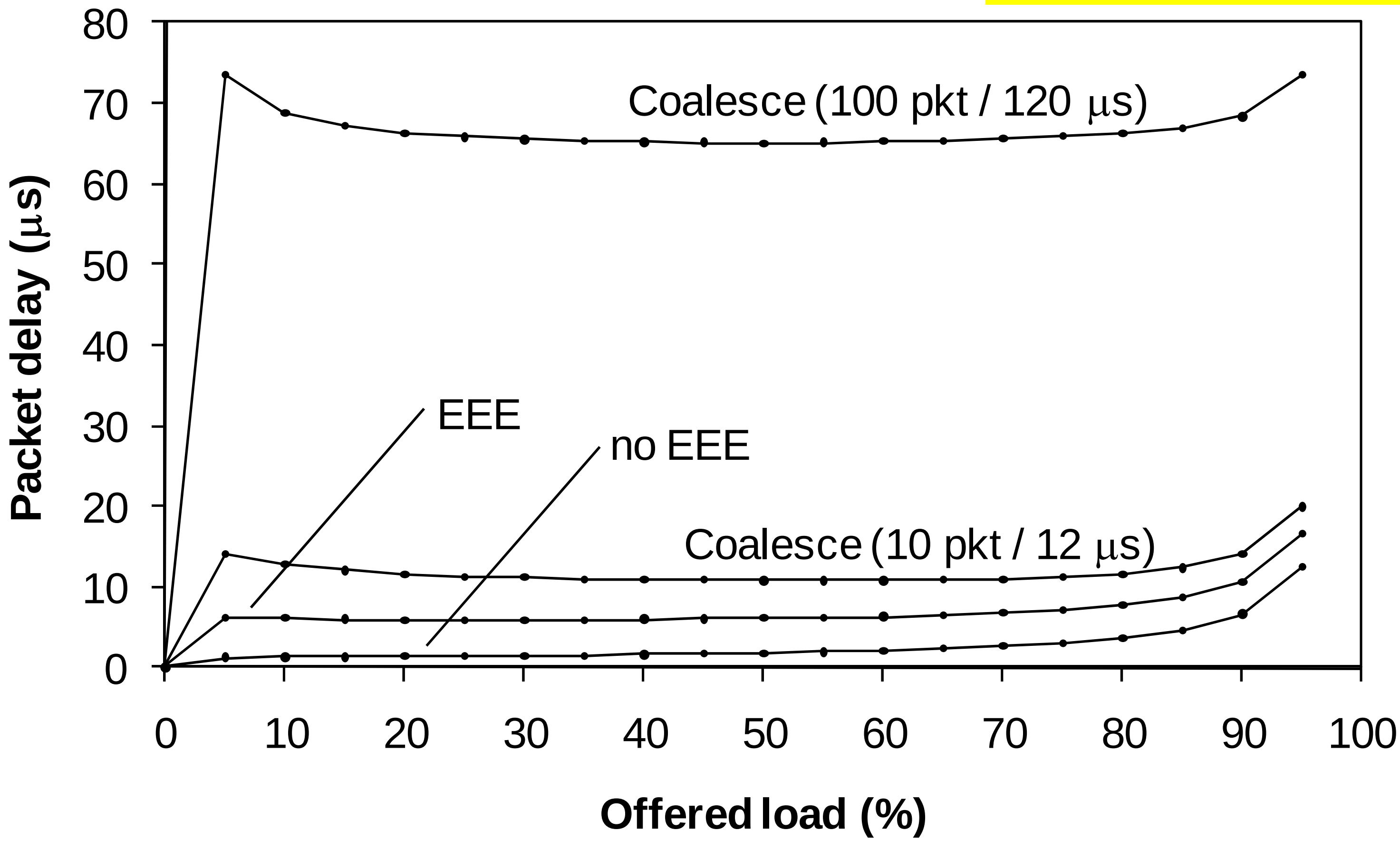
Note significant improvement.



EEE with coalescing results <u>continued</u>

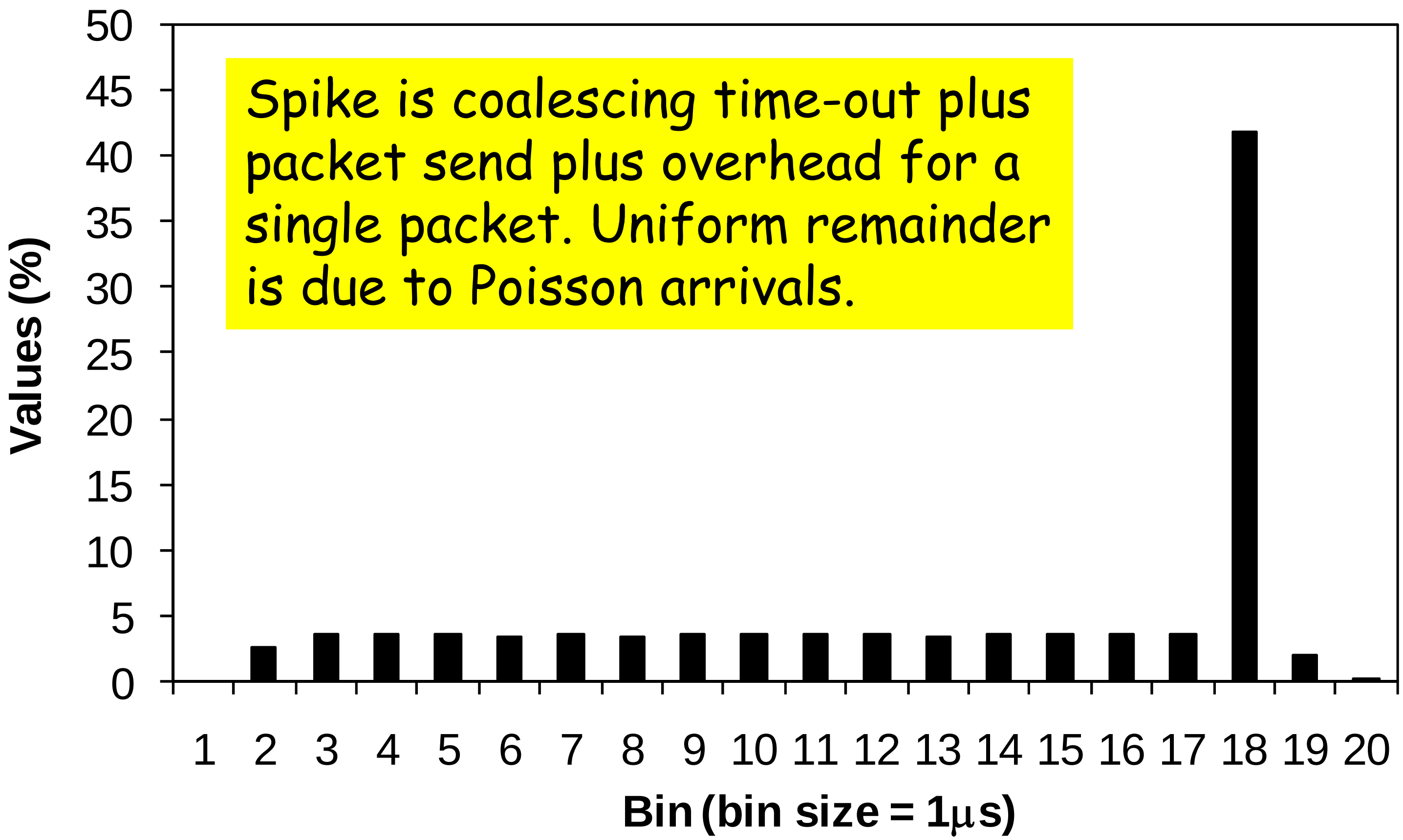
But, what about the added delay?

This is our trade-off



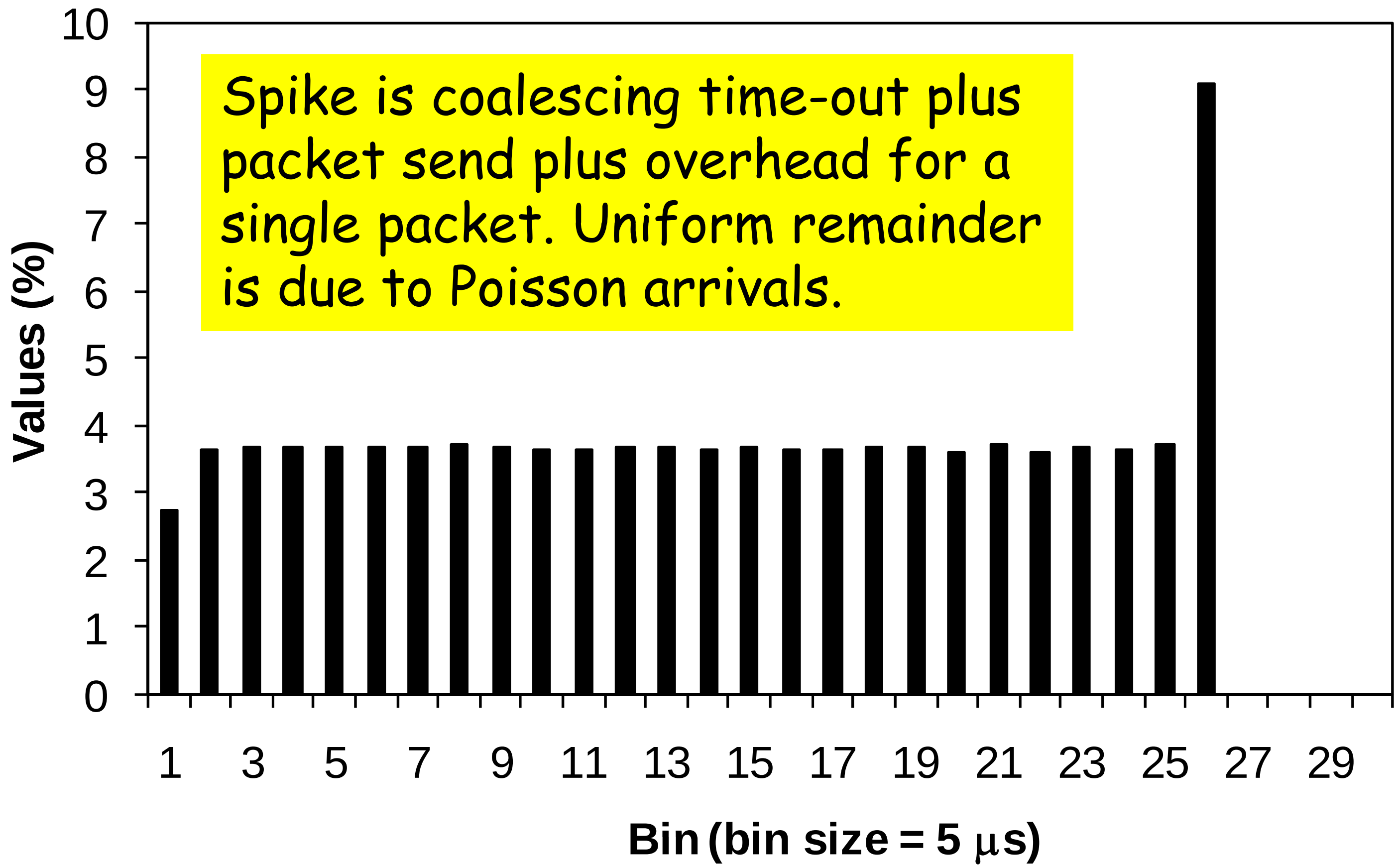
Distribution of EEE delay

For 10 packets / 12 μs and 10% offered load



Distribution of EEE delay continued

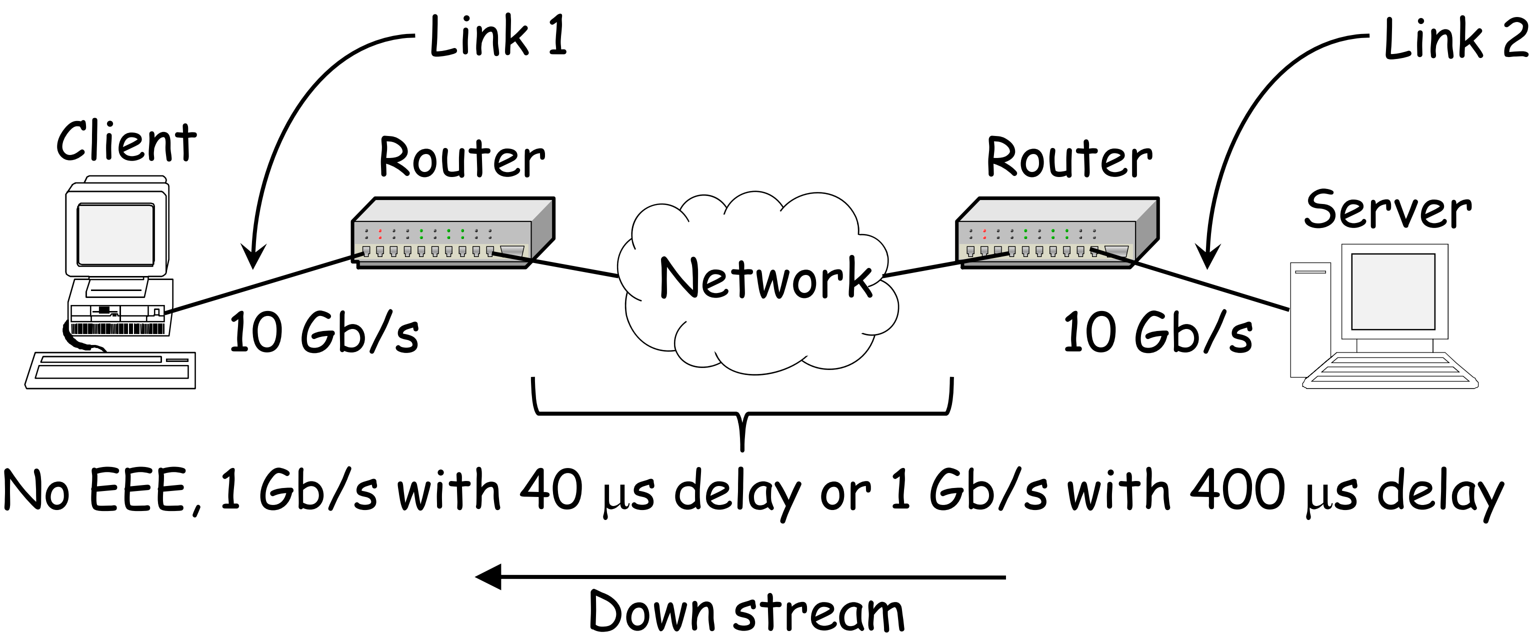
For 100 packets / 120 μs and 10% offered load



EEE file transfer

File transfer experiment with an ns-2 model

- File transfer for a 1 GB file, server to client
- Coalescing implemented in ns-2 (same parameters)



EEE file transfer <u>continued</u>

File transfer experiment parameters

- Router buffer was 100 packets
- Used ns-2 TCP Linux agent and Sack1 receiver
- TCP maximum window size of 400 packets
- The client and sever links (to the network) were 10 Gb/s
  - Without EEE (standard Ethernet)
  - With EEE
  - With EEE and coalesce-1 and coalesce-2
- Coalescing on host and router interfaces

EEE file transfer <u>continued</u>

Key measurements for the model

- File download time
- Utilization on link 1 and link 2
  - Does not include EEE overhead (only packet time)
- Energy use on link 1 and link 2
  - 100% maximum, 10% minimum

EEE file transfer model results

Link utilization

Results are "as expected" here, one ACK per data packet (ACK packets are about 1/24 the size of a data packet).

		Link utilization	
Config (10Gb/s, 40 μs delay)	**Download time** (s)	**Link 1 up**	**Link 1 down**
No EEE	0.843	4.0%	94.9%
EEE	0.843	4.0%	94.9%
EEE coalesce-1	0.843	4.0%	94.9%
EEE coalesce-2	0.847	4.0%	94.5%
Config (1Gb/s, 400 μs delay)	**Download time** (s)	**Link 1 up**	**Link 1 down**
No EEE	8.28	0.4%	9.7%
EEE	8.28	0.4%	9.7%
EEE coalesce-1	8.28	0.4%	9.7%
EEE coalesce-2	8.34	0.4%	9.7%

EEE file transfer model results <u>continued</u>

Energy use

High energy use for ACKs and for the 1 Gb/s case. Coalescing reduces energy use with little extra download time.

		Energy use (100% maximum, 10% for idle)	
Config (10Gb/s, 40 µs delay)	**Download time** (s)	**Link 1 up**	**Link 1 down**
No EEE	0.843	100%	100%
EEE	0.843	99.9%	99.9%
EEE coalesce-1	0.843	50.6%	99.9%
EEE coalesce-2	0.847	21.3%	99.5%
Config (1Gb/s, 400 µs delay)	**Download time** (s)	**Link 1 up**	**Link 1 down**
No EEE	8.28	100.0%	100.0%
EEE	8.28	65.6%	74.4%
EEE coalesce-1	8.28	38.0%	46.7%
EEE coalesce-2	8.34	17.8%	25.8%

Significance of the added delay

What is the significance of the added delay?

- Increased delay is magnitudes less than end-to-end delay on an Internet path
  - End-to-end on Internet is 10s to 100s of milliseconds
  - Coalescing delay is in the 10 to 100s of microseconds
- Increased burstiness may be an issue
  - But, coalescing is already being done for reducing packet processing load on system CPU
- Coalescing can cause TCP "ACK compression"
  - Returning ACKs come in a burst
  - Studied since early 1990s

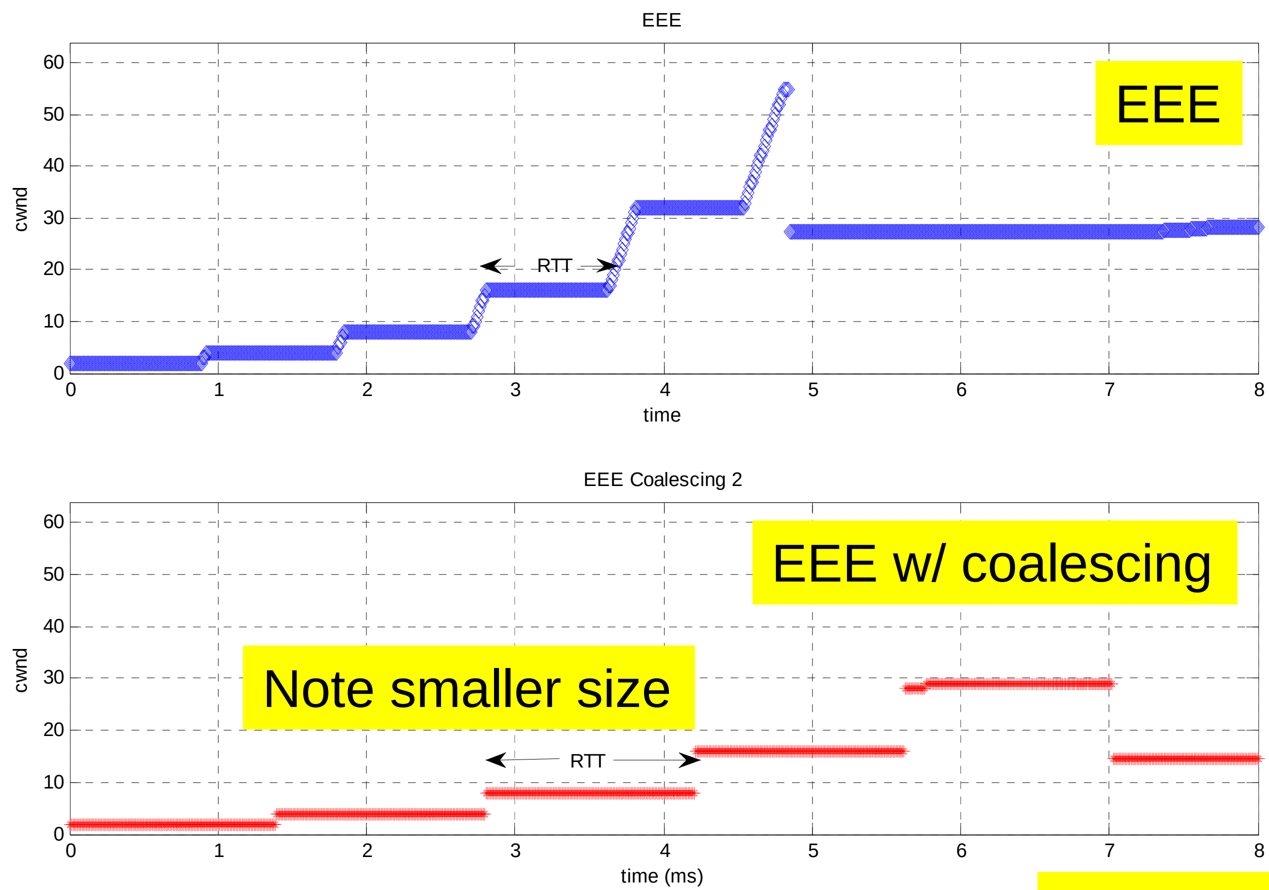
Coalescing and burstiness

A deeper understanding is needed

- Generally, coalescing will increase RTT
  - This requires a larger window size for a given throughput
- Coalescing effects are likely small if,
  - Burst size is much smaller than router and NIC buffers
  - Burst timer is much smaller than RTT
- Should explore how coalescing for reducing CPU load and coalescing to improve EEE efficiency can be combined

Coalescing and burstiness <u>continued</u>

Can explore cwnd growth in slow start with ns-2



Future work for Pedro

Economic benefits from EEE

Estimating the savings

- Savings per link is the difference between fully active and in low power mode

What are the savings from EEE?

What is the additional savings potential from coalescing?

Economic benefits from EEE <u>continued</u>

Assumptions made

- Use known 2008 stock and port count – U.S. only
  - From an estimate made for EPA in 2008
  - Thus, more 1 Gb/s than 10 Gb/s
- Increase data rates, use current power levels, and maintain assumption of low utilization
- Assume large packets, independent arrivals, and 100% PHY power consumption during transitions

Economic benefits from EEE continued

The assumptions

The savings per link may be conservative, also the mix between 1 Gb/s and 10 Gb/s

	1 Gb/s	10 Gb/s	Total
Assumptions			
Savings per link – no data (W)	1	5	—
Link utilization (%)	1 %	3 %	—
Active links (millions)	250	65	315
Electricity cost ($/kWh)	0.10	0.10	—

Economic benefits from EEE continued

EEE savings

The per link savings comes from the previous power graphs (this is the EEE overhead).

	1 Gb/s	10 Gb/s	Total
Results – EEE Savings			
Per link (%)	81 %	82%	—
Per link (W)	0.81	4.10	—
Total (MW)	200	270	470
Total (TWh/year)	1.8	2.3	4.1
Total (million $/year)	180	230	410

So, $410 million per year from EEE as is

Economic benefits from EEE <u>continued</u>

Gain from coalescing

Assumes coalescing gets us to the "ideal" line on the power graphs.

	1 Gb/s	10 Gb/s	Total
Results – Ideal Savings			
Total (TWh/year)	2.2	2.8	5
Total (million $/year)	220	280	500
Coalescing Opportunity			
Percent (last column is average)	22 %	18 %	20 %
Total (TWh/year)	0.39	0.43	0.82
Total (million $/year)	40	40	80

So, $80 million per year from coalescing.

Summary for EEE

EEE can save a lot of energy

- EEE is a means of reducing direct energy use
  - Large energy savings potential
- First proposed as ALR to IEEE 802.3 in 2005
- ALR became LPI in 2007
- LPI has overhead to wake-up and put-to-sleep a link
- LPI overhead studied and largely eliminated with coalescing

Future challenges

Where can we go from here?

... energy savings *of* and *by* ICT.

Challenges in green networks

Challenges in five areas

1) General (or overall)
2) Network equipment
3) Network hosts
4) Data centers
5) Distributed applications

Challenges in green networks <u>continued</u>

General

- Metrics
  - How do we measure energy-performance trade-offs?
- Models
  - How do we model energy-performance trade-offs?
- Exposing power and usage state
  - Need to be able to remotely determine power/use state
- Architectures for selective connectivity
  - Need mechanisms/protocols for selective connectivity
    - » Includes notions of proxying

Challenges in green networks <u>continued</u>

Network equipment

- Green routers and switches
  - Re-design routers and switches for energy efficiency
- Data caching for energy efficiency
  - Caching to reduce load network and servers
- Traffic shaping for energy efficiency
  - Shaping traffic for short-term shutdown
- Traffic engineering for energy efficiency
  - Routing to consolidate routes for long-term shutdown

Challenges in green networks <u>continued</u>

Network hosts

- Discovery of devices, capabilities, and services
  - Need to be able to discover low-power substitutes

Data center specific

- High bandwidth / low latency for dynamic virtualization
  - Useful for server shutdown

- Move computing work to where power is cheapest
  - "Follow the moon" for data center activity

Challenges in green networks <u>continued</u>

Distributed applications

- P2P, multiplayer games, and virtual worlds
  - Need to address these large and growing energy consumers
- Webcams and sensors everywhere
  - Need to address these large and growing energy consumers

Where are the "best" challenges?

My views…

- I think that the biggest challenges are at the edge
  - Most energy use there
  - Most opportunity for making changes
- Need applications and protocols that allow for and enable hosts and network equipment to sleep
- But… the biggest challenges may be in the "other 98%"
  - Many open networks problems for Smart Buildings
- Be careful to not work on problems already solved
  - Much has now been solved (the "low hanging fruit")
  - Always be able to quantify expected savings and argue that they are sufficient to be of interest

Current work in the lab

Some ideas being worked on…

- Ethernet switch power management
  - Can traffic shaping enable switches to sleep?
- Dual-channel Ethernet link for energy efficiency
  - Low-speed/low-power and high-speed/high-power
- Cooperating proxies to send requests to other machines
  - Notion of a recursive proxy
  - Protocols for discovery
- Demand response for smart appliances
  - Distributed protocols for scheduling appliances in a building

ICT can dematerialize the economy

Our economy is increasingly about…

Moving bits and not atoms

- This is how most of us now earn a living
- Made possible by networks
- Continuing trend may help us be ***comfortably green***

Conclusions

- **ICT has large and growing energy use**
- **Proxing will reduce induced energy use by hosts**
  - Potential for billions of dollars per year in the US
- **EEE will reduce direct energy use**
  - Hundreds of millions of dollars per year in US expected
- **EEE can be improved with packet coalescing**
  - Tens of millions of dollars per year in US expected
- **ICT can enable global energy savings**
  - Moving bits and not atoms = less CO_2
- **There are future challenges to be addressed**

Any questions?

Ken Christensen

http://www.csee.usf.edu/~christen/energy/main.html