ARISTA

Disruptive Innovation in ethernet switching

Lincoln Dale
Principal Engineer, Arista Networks
ltd@aristanetworks.com

AusNOG 2012

Ethernet switches have had a pretty boring existence. The odd speed increase or density jump, the odd protocol improvement in spanning tree or enhancements to link-state routing protocols, but otherwise not much has changed in 15 years. Fixed-function logic in silicon had a fixed forwarding pipeline that had limited or no flexibility and any innovation was limited to networking vendors that could afford to fund the R&D associated with a 'chipset' of forwarding silicon, ensuring all but a virtual monopoly.
Then 2012 came along.
Software Defined Networking (SDN) and OpenFlow promise to allow operators to do whatever you want with network devices.
Ever-shrinking process nodes in silicon have enabled ever higher silicon integration and flexible packet parsing, forwarding and rewrite logic are enabling 'switches' to be deployed where 'routers' used to rule.
This presentation aims to give the audience a view into the world of ethernet switch development, how modern switch silicon works, what silicon process node shrinks mean for network devices and what an open, software-defined network world may look like.
We'll cover a soup-to-nuts timeline of how network silicon used to designed and built to how it is done today and provide insights as to what the future likely holds and what that means for network operators and what it means to how network design and architecture will evolve moving forward.

22 May 1973

Robert Metcalfe sends a memo to his boss stating the possibilities of ethernet's potential

Ethernet is born.

1975/1976

Robert Metcalfe and David Boggs (Metcalfe's assistant) published a paper titled, "Ethernet: Distributed Packet-Switching For Local Computer Networks."

Ethernet: Distributed Packet Switching for Local Computer Networks

by Robert M. Metcalfe and David R. Boggs

CSL-75-7 May 1975, reprinted February 1980

Abstract: Ethernet is a branching broadcast communication system for carrying digital data packets among locally distributed computing stations. The packet transport mechanism provided by Ethernet has been used to build systems which can be viewed as either local computer networks or loosely coupled multiprocessors.

An Ethernet's shared communication facility, its Ether, is a passive broadcast medium with no central control. Coordination of access to the Ether for packet broadcasts is distributed among the contending transmitting stations using controlled statistical arbitration. Switching of packets to their destinations on the Ether is distributed among the receiving stations using packet address recognition.

Design principles and implementation are described based on experience with an operating Ethernet of 100 nodes along a kilometer of coaxial cable. A model for estimating performance under heavy loads and a packet protocol for error-controlled communication are included for completeness.

A version of this paper appeared in *Communications of the ACM*, vol. 19 no. 7, July 1976.

CR Categories: 3.81, 4.32, 6.35

Key words and phrases: computer networks, packet switching, multiprocessing, distributed control, distributed computing, broadcast communication, statistical arbitration

XEROX
PALO ALTO RESEARCH CENTER
3333 Coyote Hill Road / Palo Alto / California 94304

1983

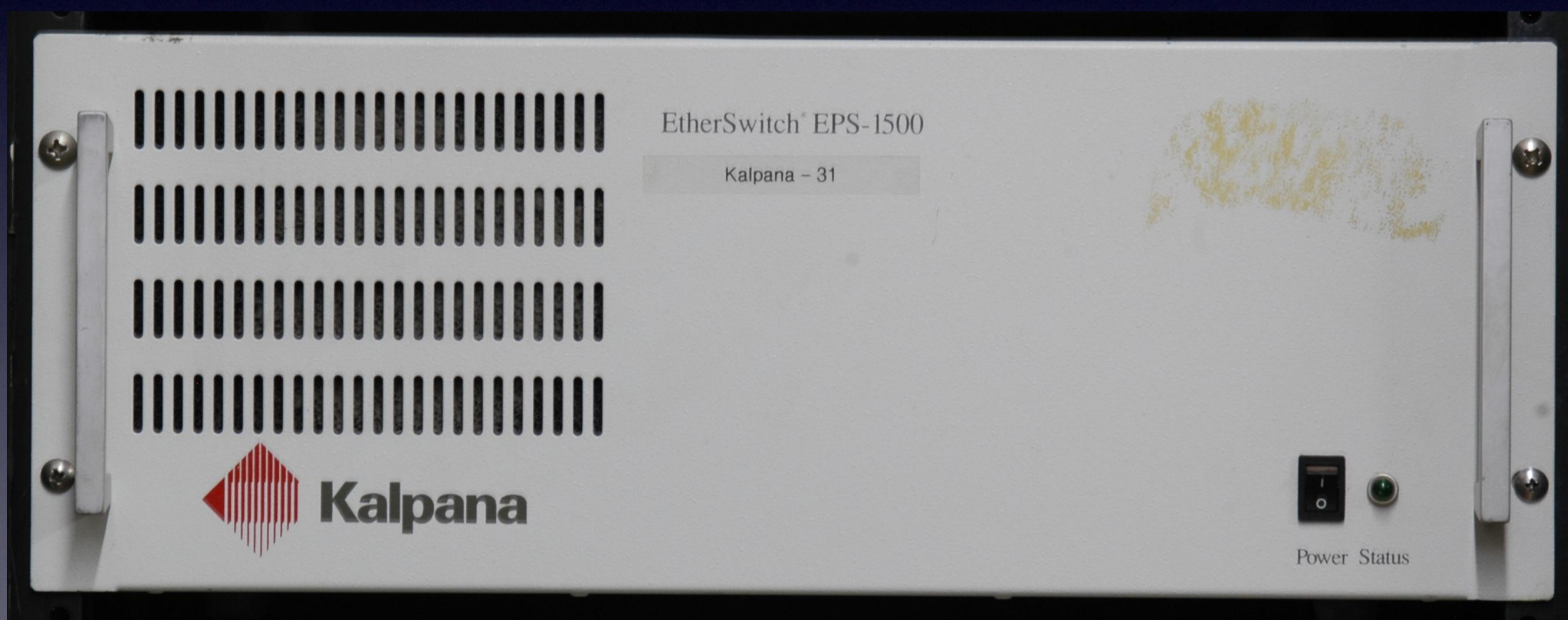
IEEE publishes 802.3 CSMA/CD standard in draft format.

Becomes a ratified standard in 1985.

(just 12 short years after Metcalfe's memo)

1989

First Ethernet switch ('multiport bridge') released!
(7 x 10BASE-T @ $1500/port)



1990

IEEE 802.1D Spanning Tree first published

(based on 1985-1989 work from Radia Perlman at DEC)

2001

IEEE 802.1w Rapid Spanning Tree introduced

reduces convergence times from 30-50s to 7s

woot

The last decade

Many protocols introduced at L2
L3 protocols pretty much the same

Spanning Tree replacement(s) TRILL, SPB and PBB introduced.

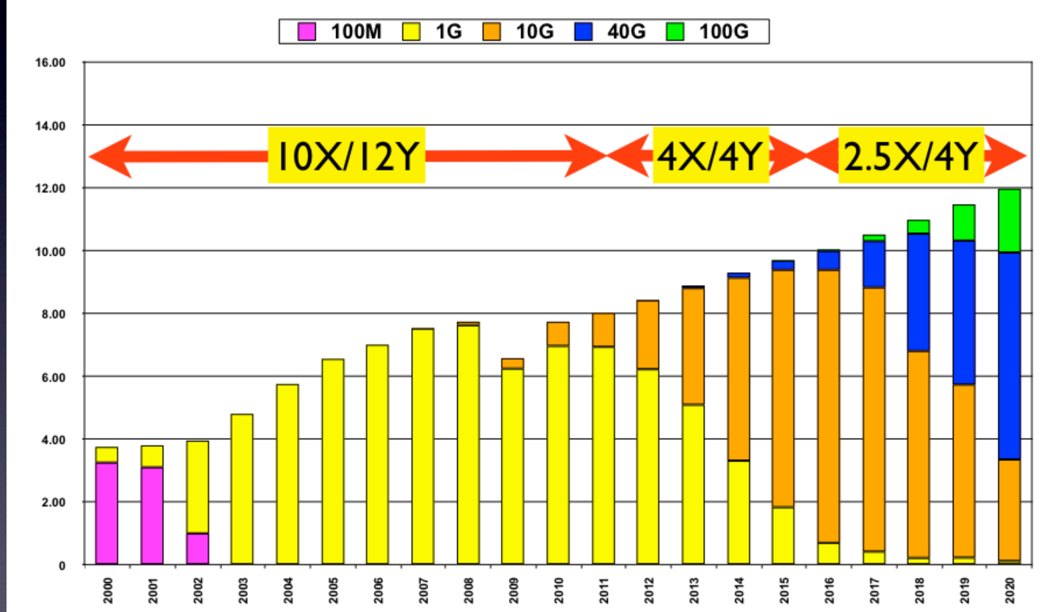
Little point in anyone implementing them.
(Friends don't let friends build large L2 networks.)

Most Ethernet switch vendors introduce ways of avoiding blocked links on Spanning Tree via MLAG/vPC offering a more evolutionary evolution.

Widespread Adoption of Ethernet (in servers)

2002:
Fast Ethernet to Gigabit Ethernet

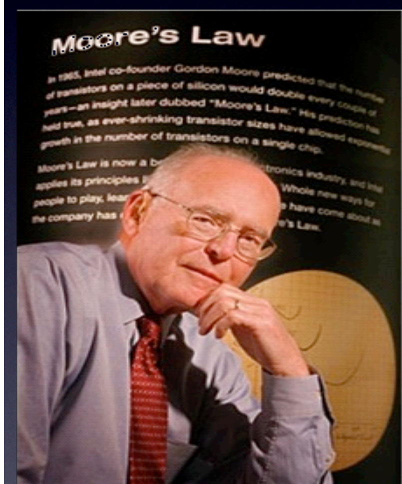
2013/14:
Crossover Gigabit to 10GbE

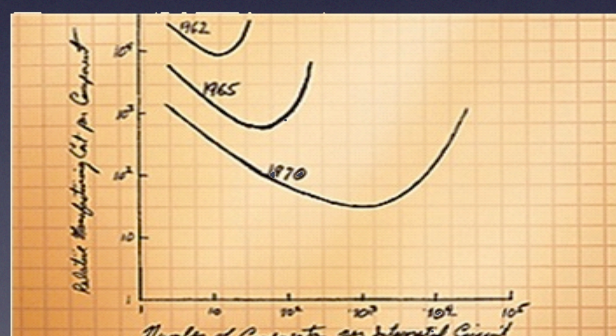


Source: Intel LAN Group

Moore's Law

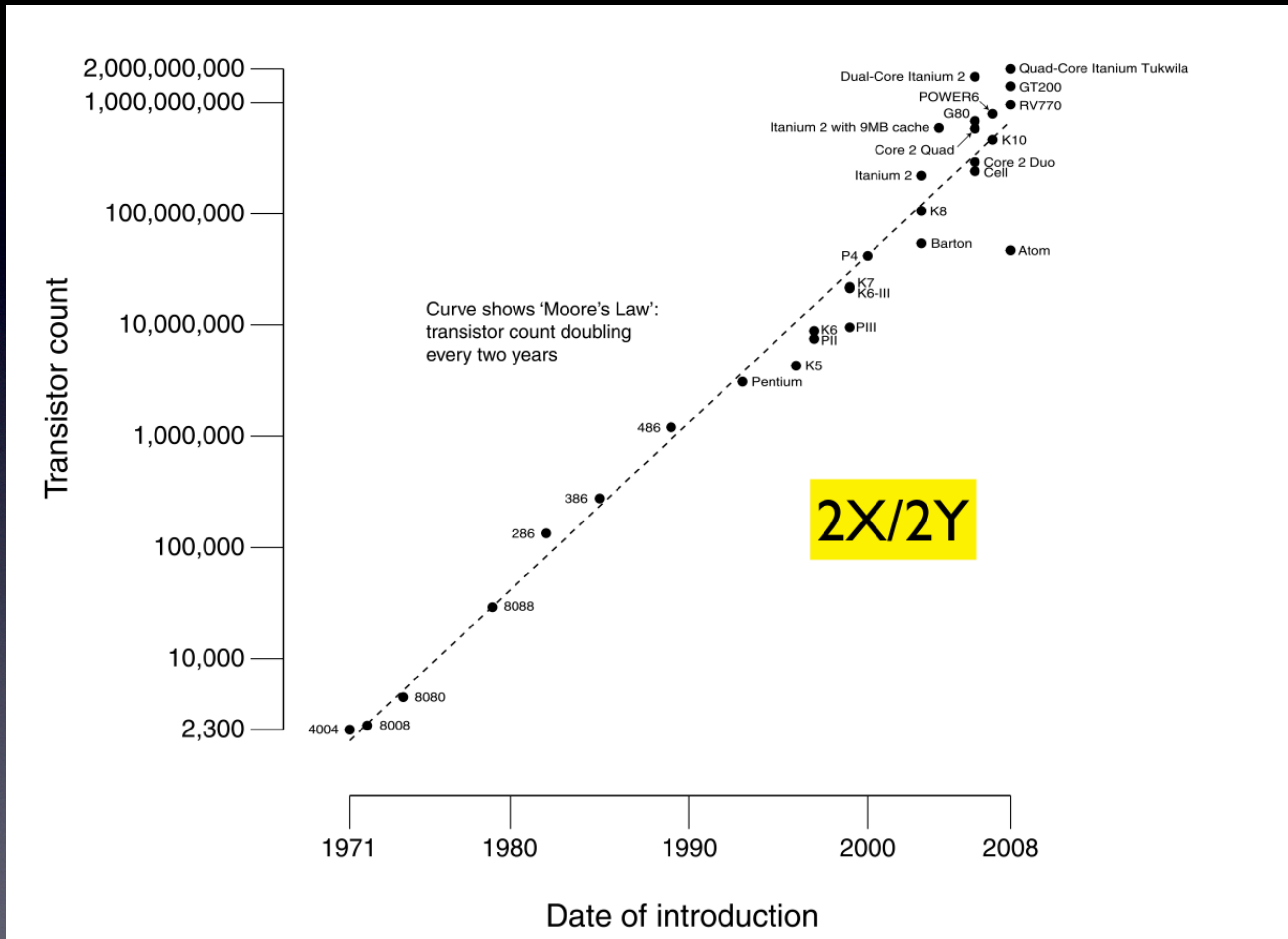
The observation made in 1965 by Gordon Moore, co-founder of Intel, that **the number of transistors per square inch on integrated circuits had doubled every year** since the integrated circuit was invented. Moore predicted that this trend would continue for the foreseeable future.



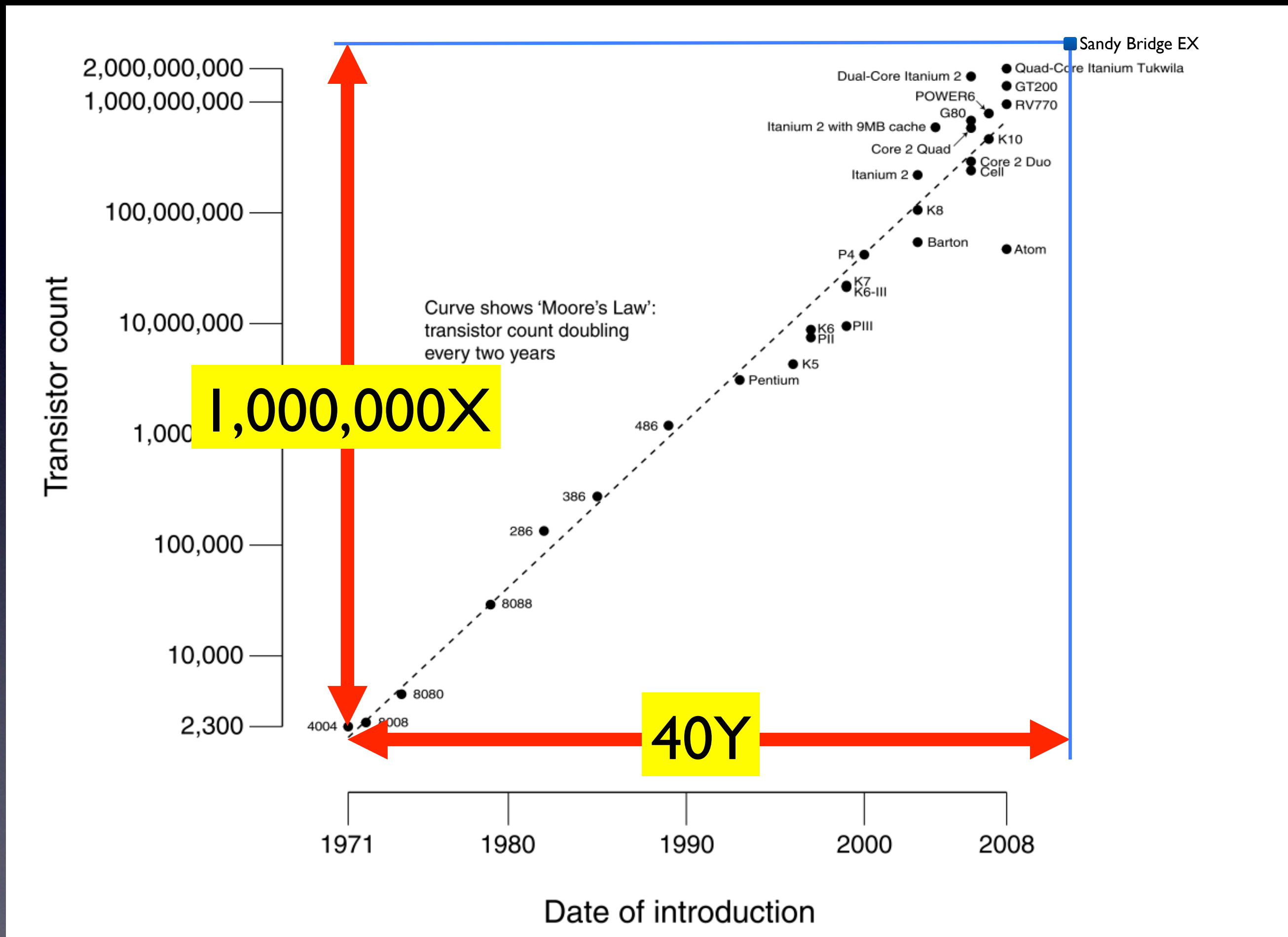


1975 Revision became known as Moore's Law: The Number of Transistors will double every 2 years

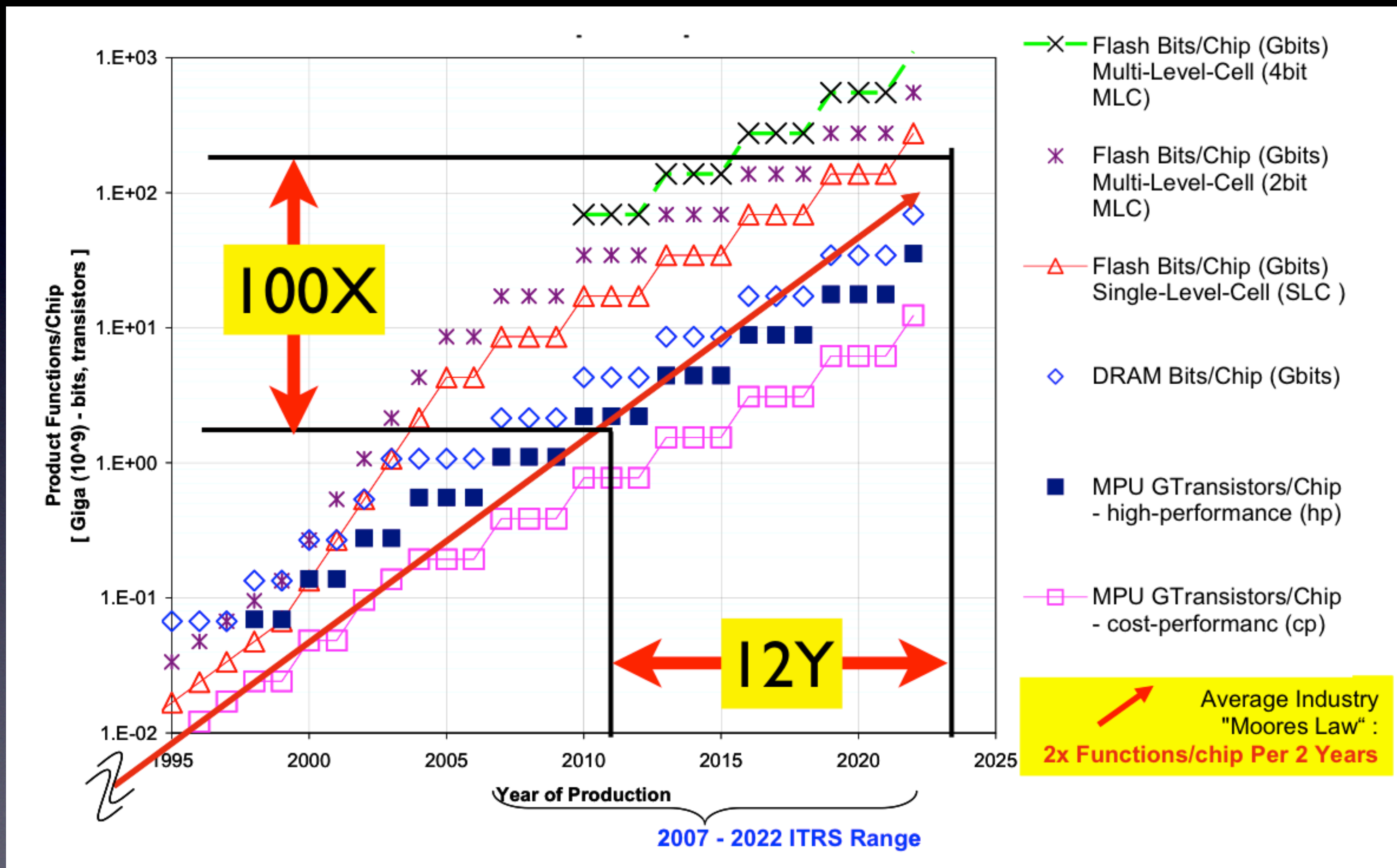
Moore's Law and CPUs



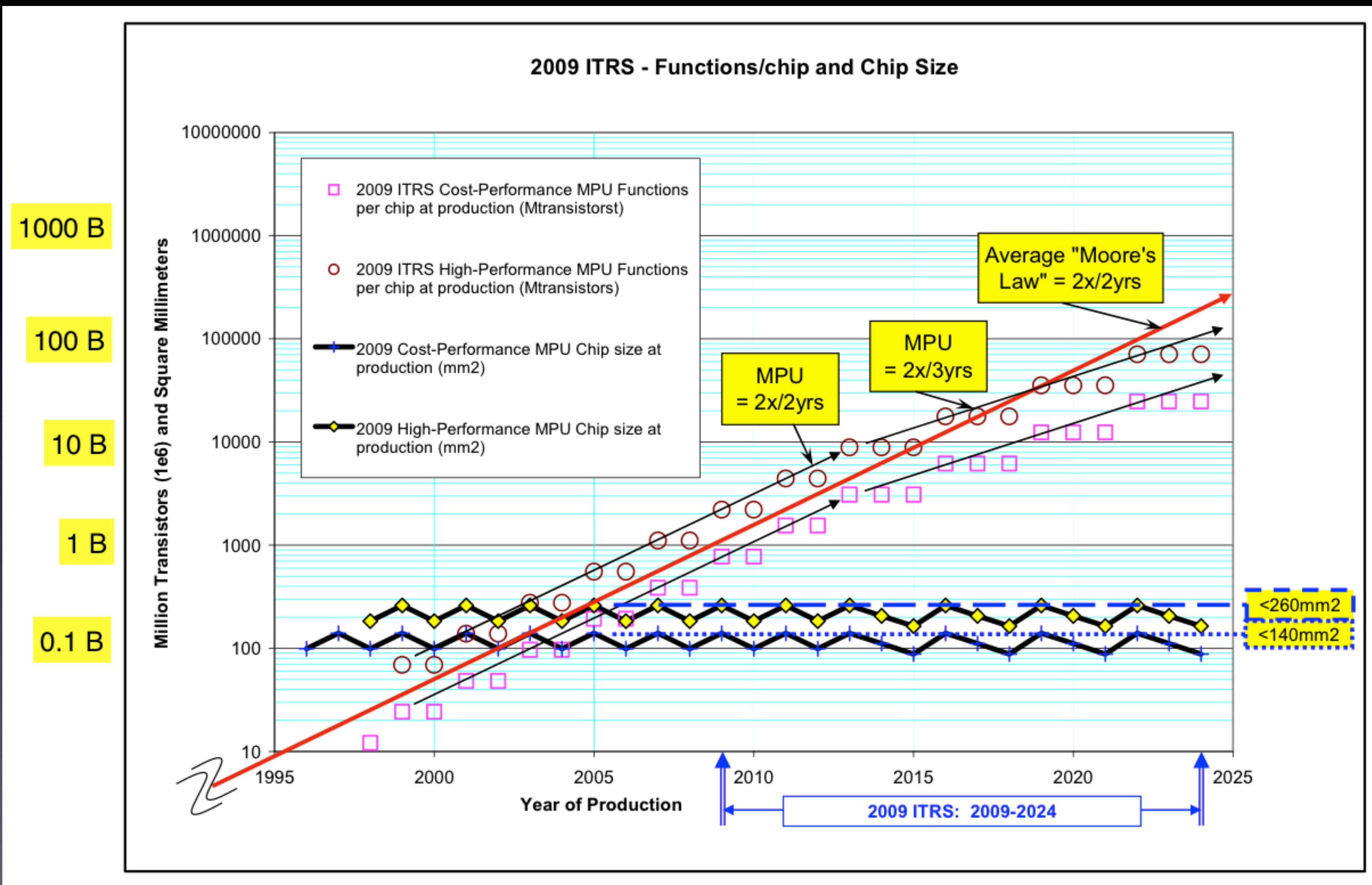
Moore's Law and CPUs



Semiconductor Technology Roadmap

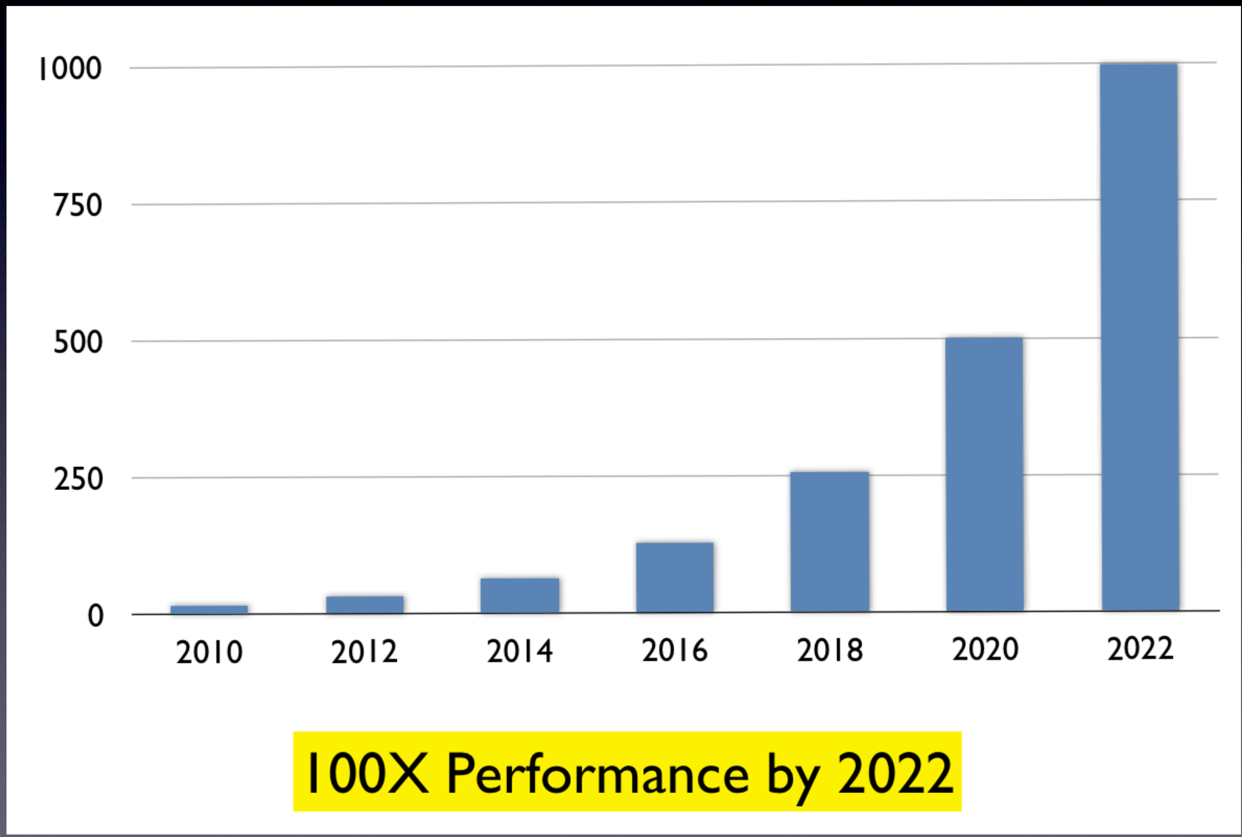


Snapshot on Logic Density



64 bit CPU Cores over Time

(if the focus was on just increasing core count)

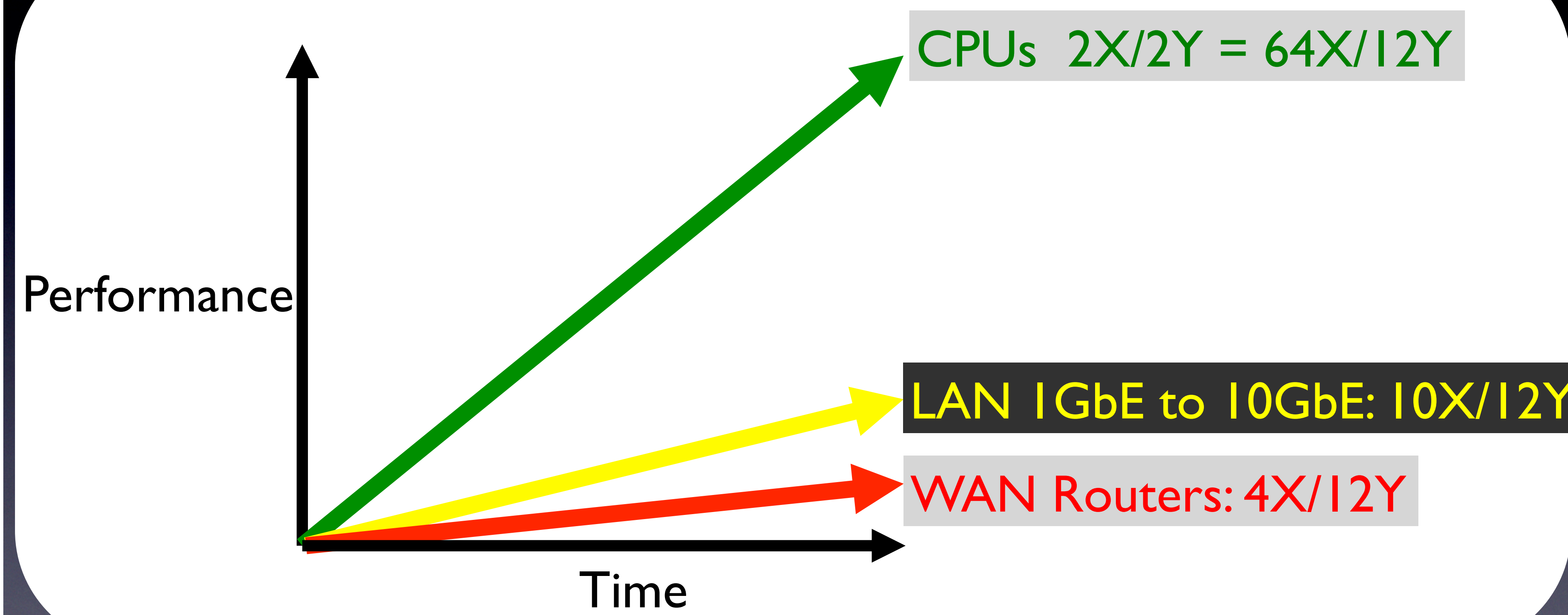


Moore’s Law Summary

- **Moore’s Law is alive and well**
  - 2X Density every 2 Years
- **Million-fold advance from 1971-2011**
  - Another factor of 100X next 12 years
- **Billion-fold advance expected 1971-2031**
  - Beyond that its hard to forecast

There has been nothing like this in the history of mankind

Moore's Law and Networking

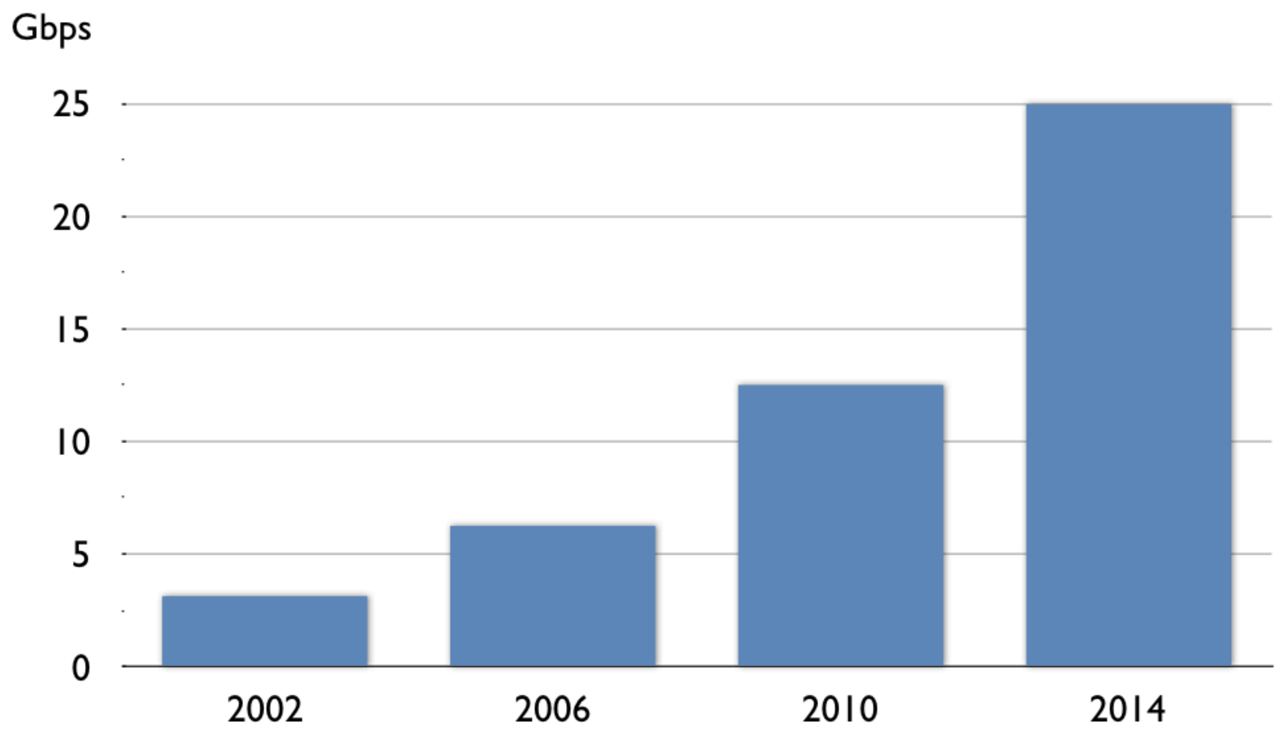


Why has Networking not kept up with Moore's Law?

Three main problems

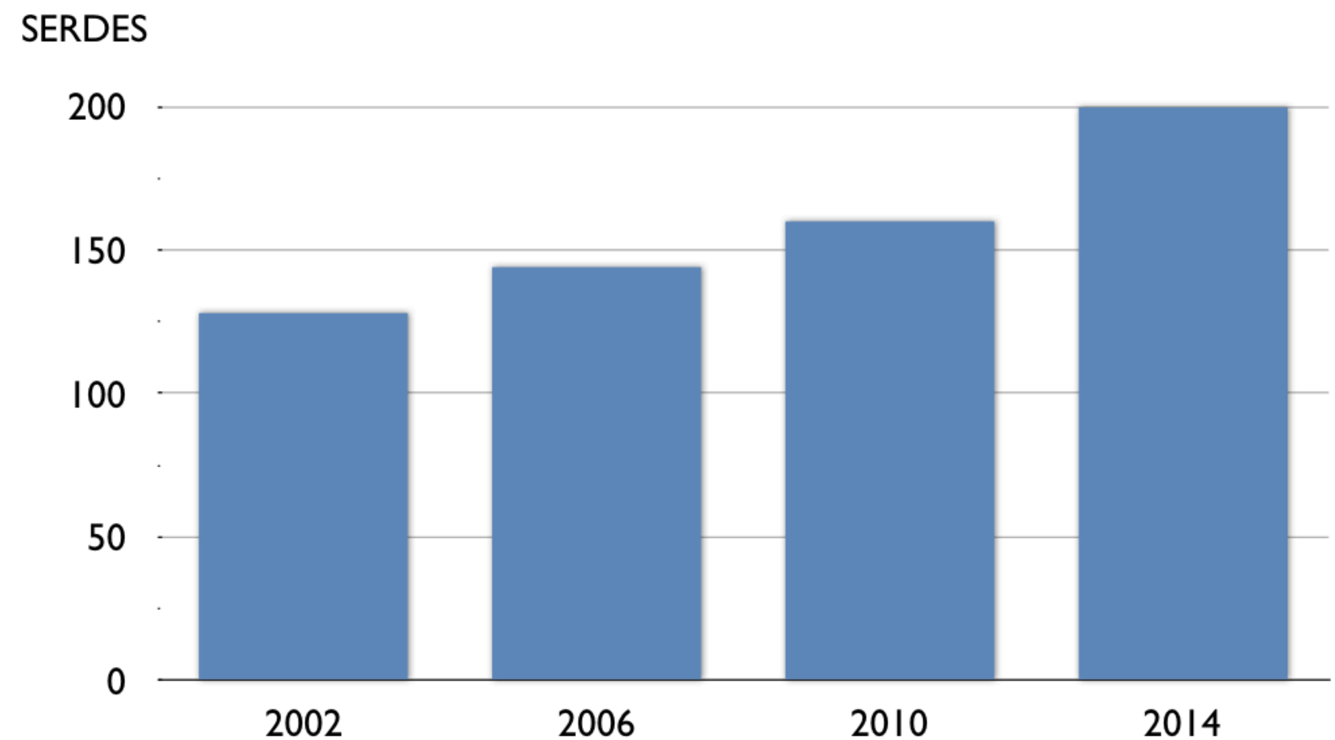
- **Moore's Law applies to Transistors, not Speed**
  - Transistor count is doubling every 2 years
  - Transistor speed is only increasing slowly
- **Number of I/O pins per package basically fixed**
  - Limited by die area and package technology
  - Only improvement is increased I/O speed
- **Bandwidth ultimately limited by I/O capacity**
  - Throughput per chip = # IO Pins x Speed/IO
  - No matter how many transistors are on-chip

SERDES Speed (high density CMOS)



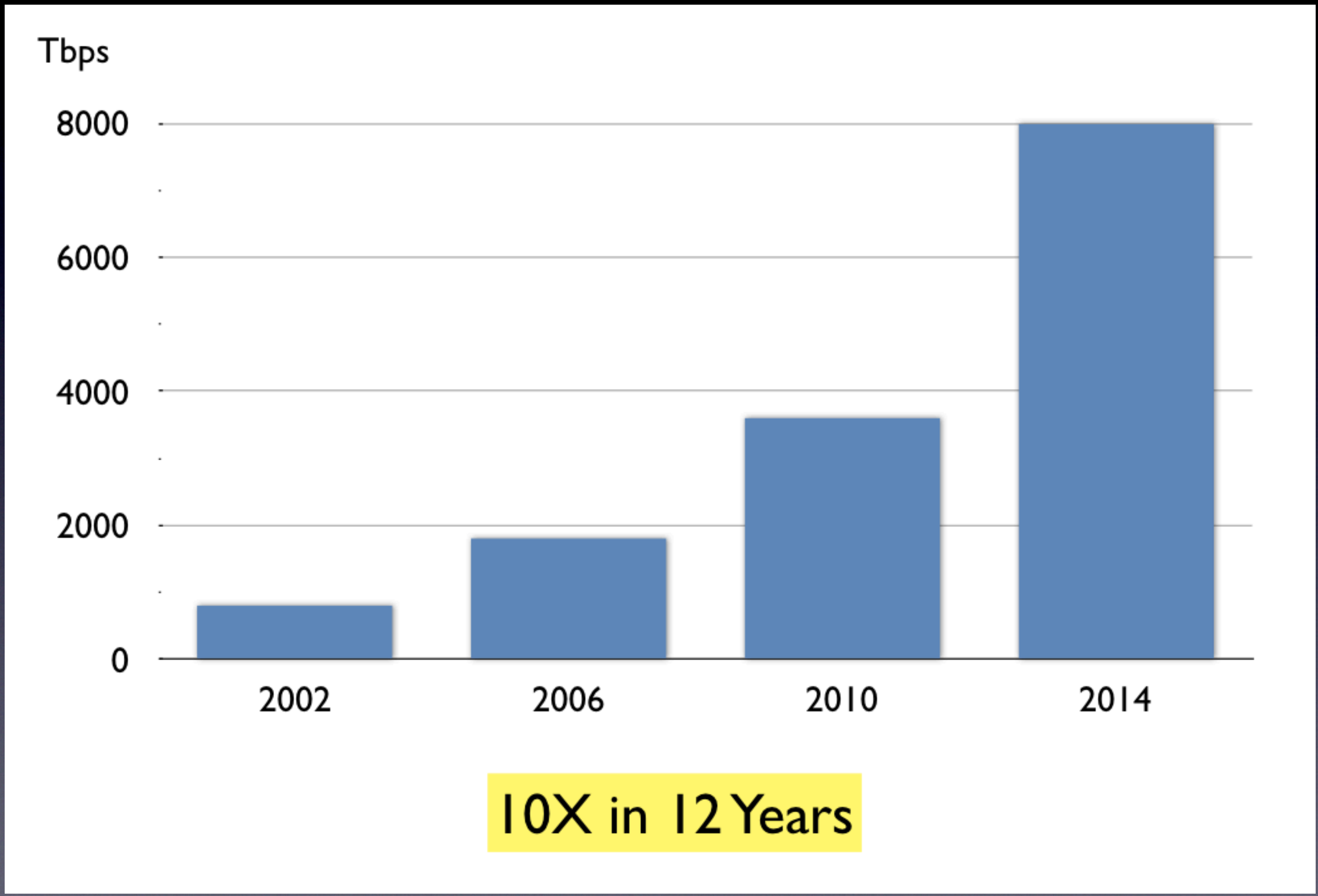
8X in 12 Years = 2X every 4 Years

Number of SERDES per Package

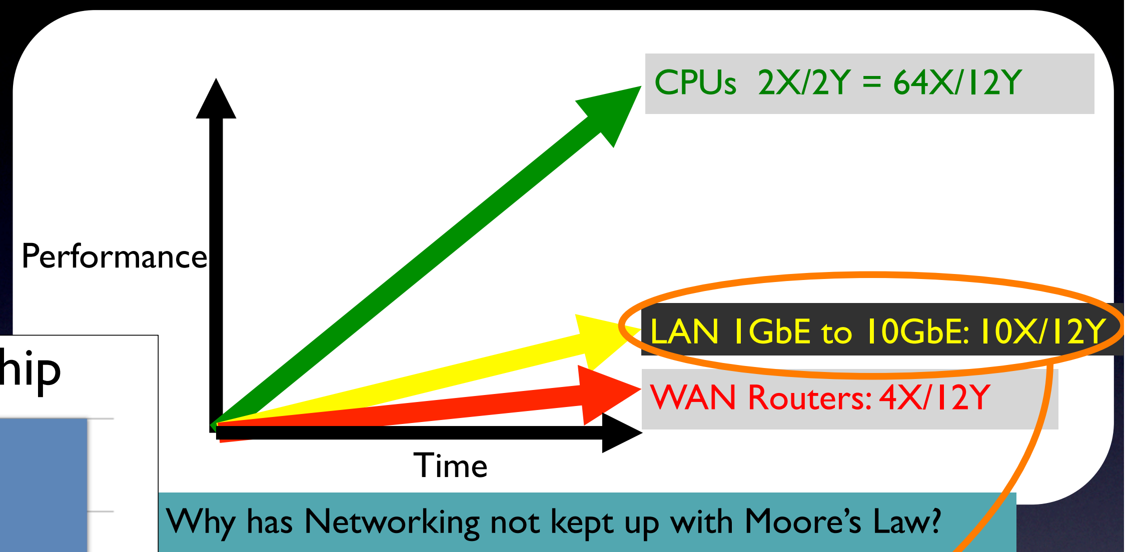


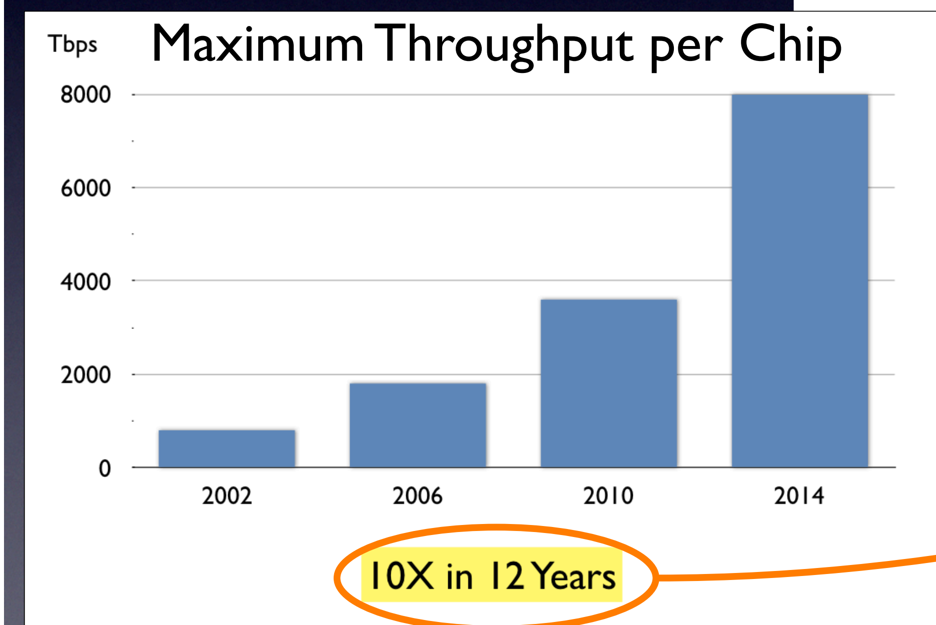
Modest Increase in 12 Years

Maximum Throughput per Chip



Moore's Law and Networking





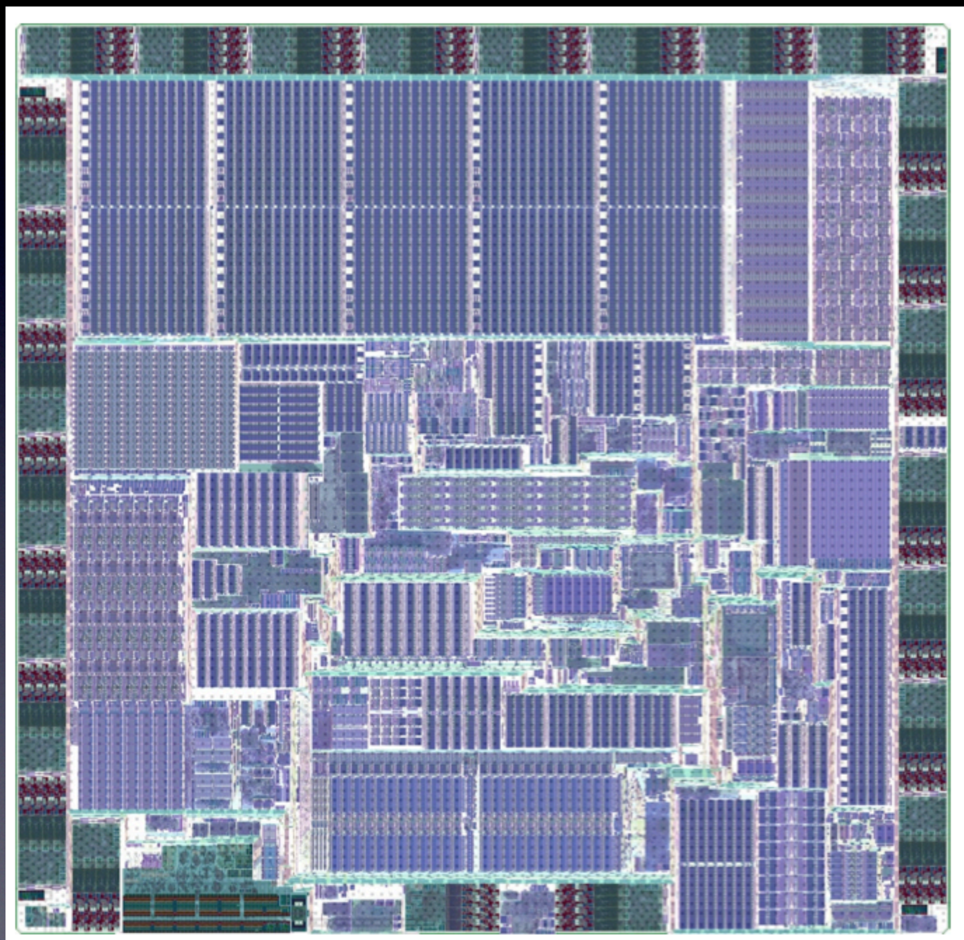
‘ASIC’ vs ‘Full Custom’ Chip Design

- **ASIC = Application Specific Integrated Circuit**
  - ‘Top-down’ design, independent of layout
  - ASIC supplier does physical implementation
  - Difficult to achieve high clock rates this way

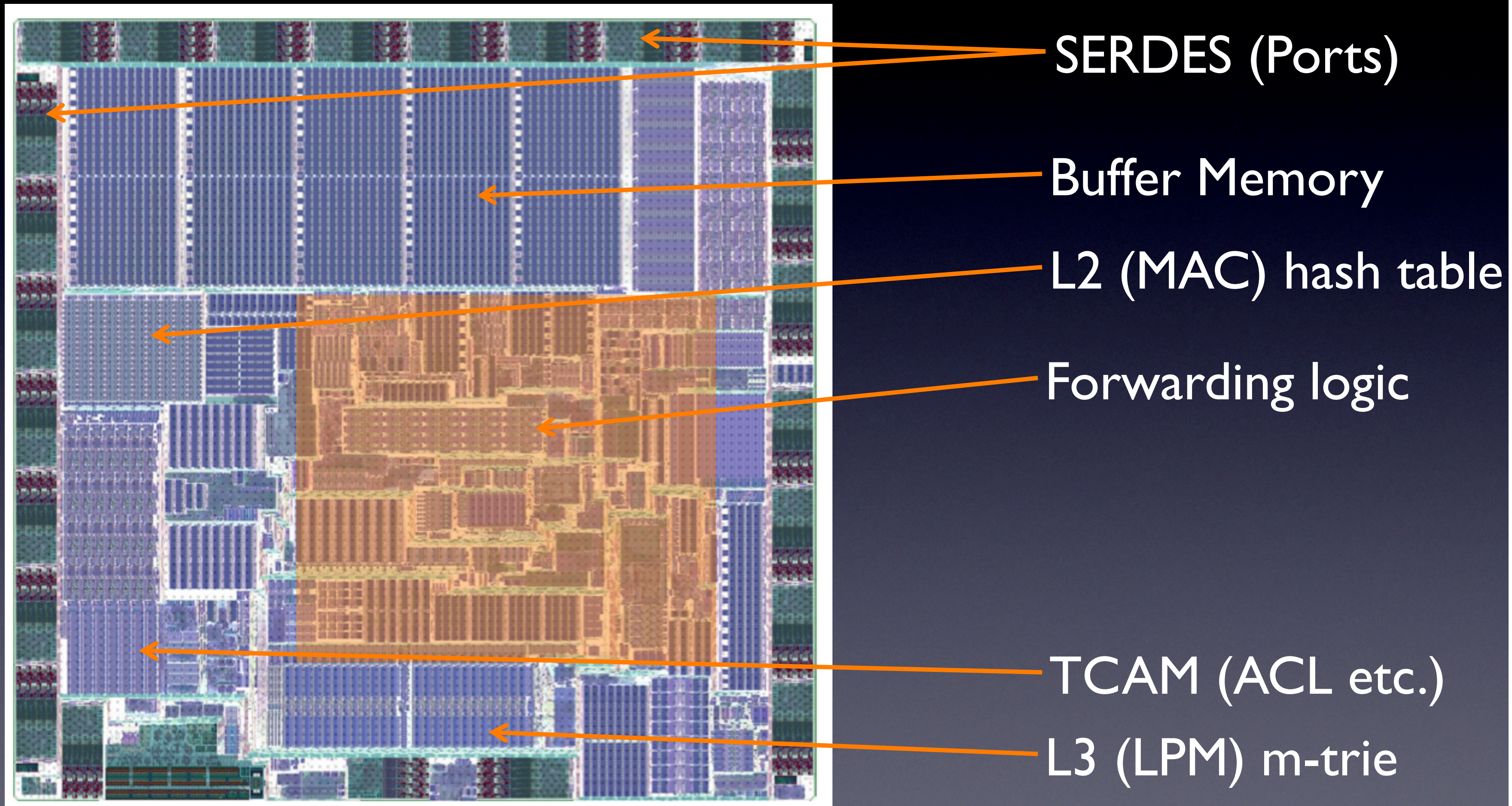
- **Full Custom Flow**
  - Chip design starts with clock rate objective
  - Data Paths designed to achieve clock rate
  - Only way to achieve high clock rates

Typical Result: 8X Higher Density in Full Custom vs ASIC

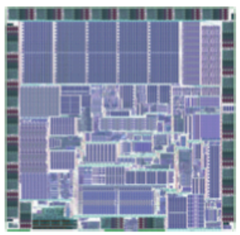
Full Custom 64 port 10G Switch Chip



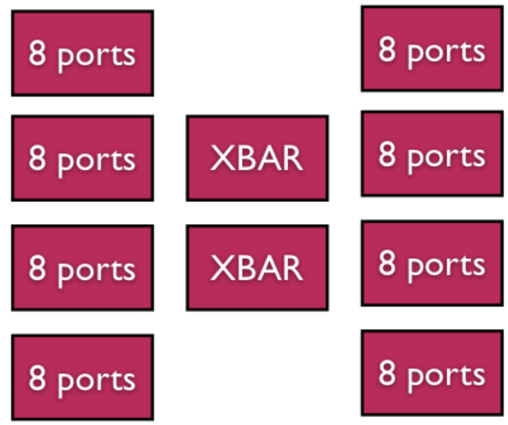
Full Custom 64 port 10G Switch Chip



64 port 10G Switch: Custom vs ASIC



Custom Design: 1 Chip
Full L2/L3



ASIC Design: 10 Chips
L2 only

Advantages of Full Custom Chips

Full Custom Switch Chips have

- more ports per chip
- much lower latency (due to fewer chip crossings)
- consume less power
- more room for additional logic/processing/functionality
- much more reliable than traditional ASIC multi-chip designs

Full Custom chips ARE on Moore's law

ASIC designs are NOT on Moore's law

Evolution of Custom Switch Silicon

Technology	130nm	65nm	40nm	28nm
10G ports	24	64	128	256
Throughput	360M PPS	960M PPS	2B PPS	4B PPS
Buffer Size	2 MB	8 MB	16 MB	32 MB
Table Size	16K	64K	128K	256K
Port Speeds	10G	10G/40G	10G/40G/100G	10G/40G/100G
Availability	2007	2011	2013	2015
Improvement	-	3X/4Y	2X/2Y	2X/2Y

Next generation custom switch silicon is on Moore's Law!

Broadcom launches Trident II switch chip

Blasting over 100 10GE ports into the clouds

By **Timothy Prickett Morgan** • **Get more from this author**

Posted in Servers, 27th August 2012 15:42 GMT

Broadcom is not revealing all the feeds and speeds of the Trident II ASIC for Ethernet switches quite yet, but the company is starting to sample the chips this week to customers and wants to give data center managers a peek at the kind of scalability and performance they can expect from switches that use the chip.

The company is touting the fact that this is the first switch ASIC that can drive more than a hundred 10GE ports from a single chip, and that there is enough bandwidth in there to make a pretty fat 40GE switch, too. The prior Trident ASICs offered 640Gbps of Ethernet switching capacity, but the new Trident II will boost that to 980Gbps for some models and up to 1.28Tbps for other models.

This is on par with the current Trident+ ASIC that Broadcom has been selling. The difference is that Trident II is designed to drive more and faster ports and has a bunch more features for the increasingly virty world.

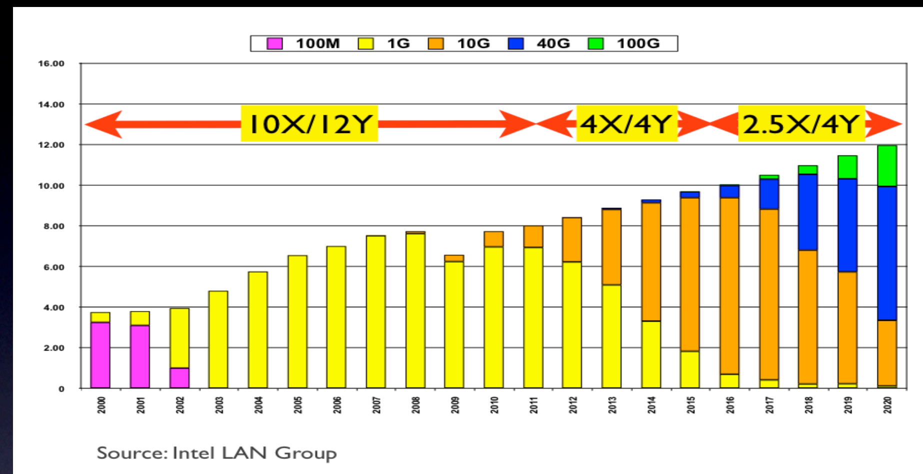
For instance, in a 2U rack form factor, customers will be able to drive 104 10GE ports from a single chip. Mui also says he expects for switch makers to offer machines with 96 10GE ports plus eight 40GE ports and 64 10GE ports plus 16 40GE ports in a 2U form factor, and even a monster 1U screamer with 32 40GE ports. The port-to-port hop latency is under 500 nanoseconds with the Trident II ASIC, and a 10GE port consumes less than 1 watt of juice as it is humming with data.

Moore’s Law and Networking

- **Next Generations scale with Moore’s Law**
  - Table sizes double every process node
  - Industry catching up on process roadmap
- **I/O Speed scales less than Moore**
  - Larger package sizes offset this constraint
  - Next step is 25G SERDES in 2013
- **Full Custom Design Flow Required**
  - ASIC design flow wastes silicon potential

CPUs driving the Network Upgrade

- **Faster CPUs need Faster Networks**
  - Intel Sandybridge driving 10GbE adoption
  - 50% attach rate 2013, 80% by 2015

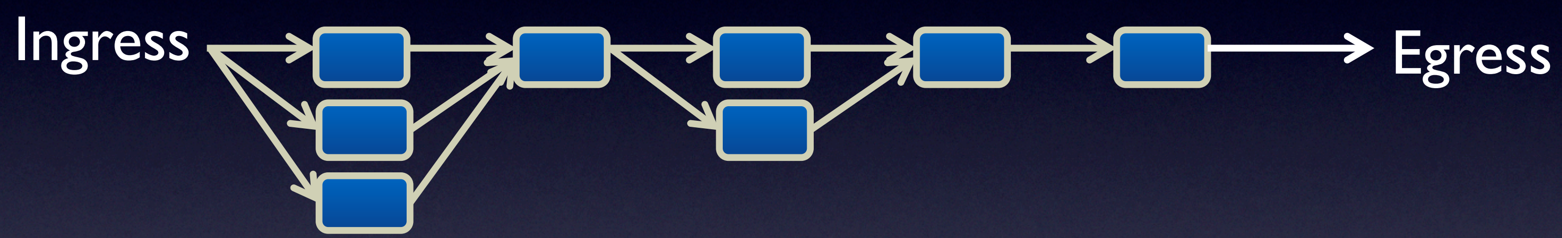


- **10/40/100G Market growing rapidly**
  - $4B in 2010 to $16B in 2016
  - From 5M ports 2010 to 67M ports 2016

- **Faster End nodes need faster Backbones**
  - Many apps drive east/west traffic not north/south
  - Cluster sizes getting larger & larger

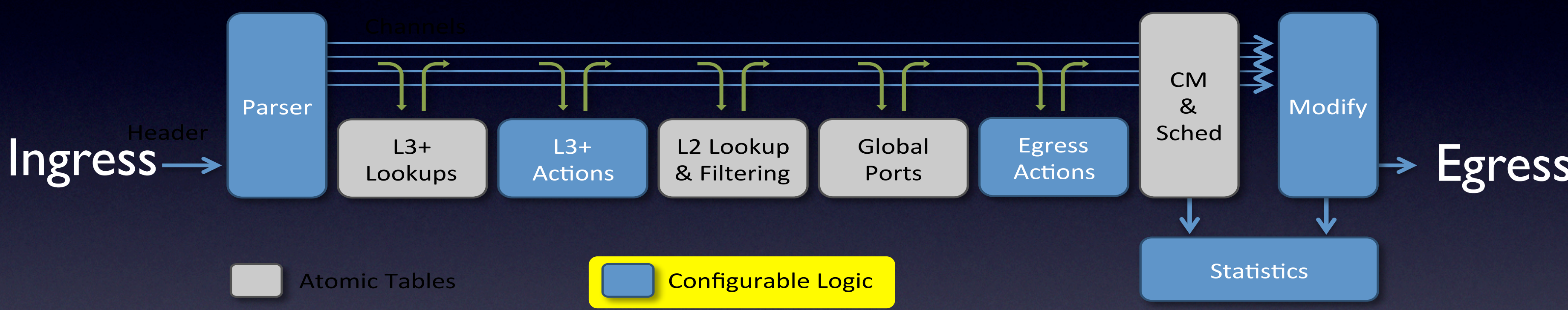
How real Clouds are Built

Besides larger tables, what else can 2X/2Y transistors be used for?



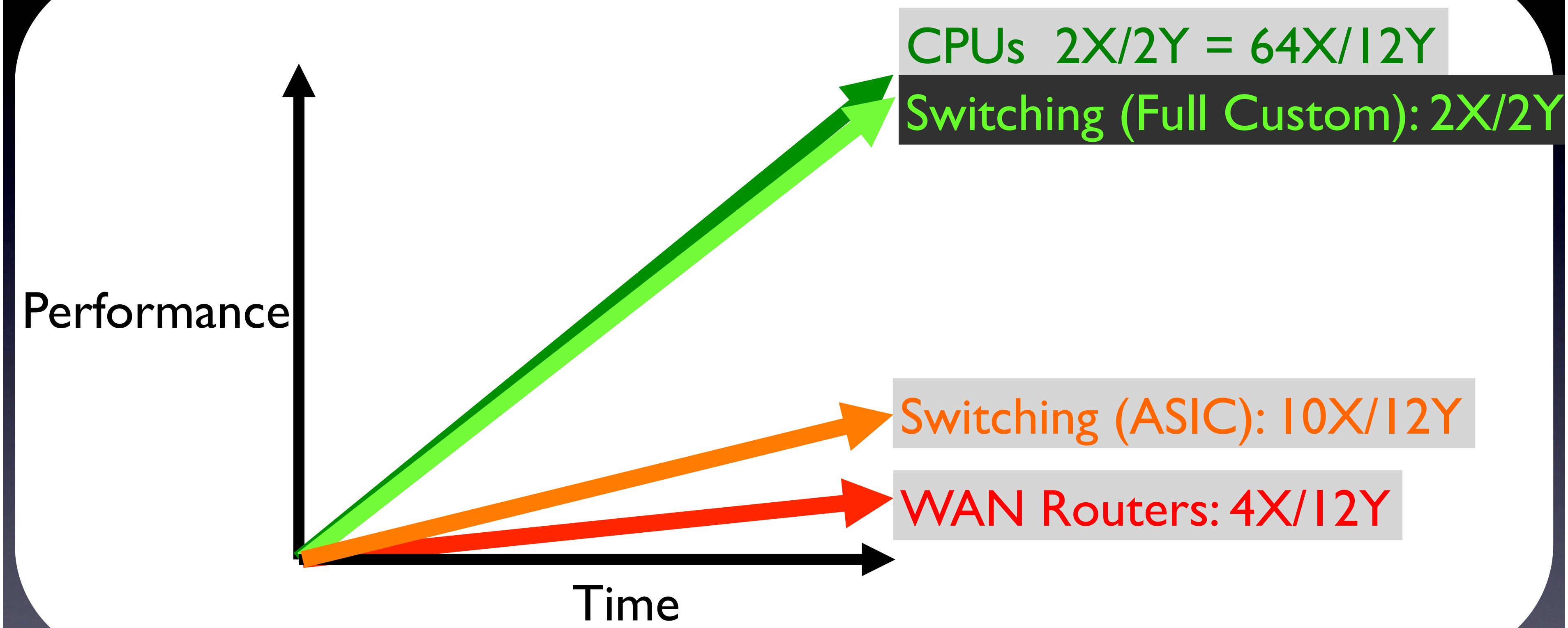
Historically the processing pipeline was fixed in switches

Besides larger tables, what else can 2X/2Y transistors be used for?



Flexible Packet Parsing and Flexible Packet Rewrite provide Router-port functionality at Switch-port pricing

Moore's Law and Networking



Besides larger tables, what else can 2X/2Y transistors be used for?

OpenFlow

SDN

Flexible forwarding requires flexible ways of exposing the underlying functionality

Besides larger tables, what else can 2X/2Y transistors be used for?

```
#!/usr/bin/env python
# Copyright (c) 2012 Arista Networks, Inc.  All rights reserved.
# Arista Networks, Inc. Confidential and Proprietary.

import Tac
flowTable = d.newEntity( "OpenFlowTable::HwConfig", "default" )
match = Tac.Value( "OpenFlowTable::Match" )
matched = Tac.Value( "OpenFlowTable::MatchFieldSet" )
actions = Tac.Value( "OpenFlowTable::Actions" )
enabled = Tac.Value( "OpenFlowTable::ActionSet" )

match.inIntf = "Ethernet1"  # match traffic arriving on ethernet1
matched.inIntf = True

match.vlanId = 100          # match traffic ingress vlan 100
match.vlanIdMask = 0x0fff
matched.vlanId = True

match.ipSrc = "10.0.0.1"    # match src ip of 10.0.0.1 only
match.ipSrcMask = "255.255.255.255"
matched.ipSrc = True

match.ipDst = "10.0.0.2"    # match dst ip of 10.0.0.2 only
match.ipDstMask = "255.255.255.255"
matched.ipDst = True

match.l4Dst = 80            # match http traffic only
matched.l4Dst = True
match.matched = matched
```

```
actions.outputIntf["Ethernet23"] = True  # send out eth23
actions.outputIntf["Ethernet44"] = True  # and et44
enabled.outputIntf = True
actions.enabled = enabled

print "Adding to flow table"
flow = flowTable.newFlowEntry( "flow100", match, actions )

def printFlowTable():
   for flowName, flow in flowTable.flowEntry.items():
      print "%s:" % flowName
      print "  match: %s" % flow.match
      print "  actions: %s" % flow.actions
      print "  priority: %s" % flow.priority

print "Printing flow table"
printFlowTable()

print "Deleting flow"
del flowTable.flowEntry["flow100"]
```


ARISTA