Increase Efficiency with Hardware Specialization
Datacenter Environment

- Software services change monthly
- Machines last 3 years, purchased on a rolling basis
- Machines repurposed ~½ way into lifecycle
- Little/no HW maintenance, no accessibility

- Homogeneity is highly desirable

The paradox: Specialization and homogeneity
Efficiency via Specialization

Source: Bob Broderson, Berkeley Wireless group
One Application’s Accelerator

Flexibility

Xeon CPU

Efficiency

NIC

Accelerator Opportunities
One Application’s Accelerator

Flexibility

Xeon CPU

Application Changes

Search Acc. (FPGA)

Search Acc. (ASIC)

Efficiency

NIC

Wasted Power, Holds back SW

One more thing that can break
Our Design Requirements

Don’t Cost Too Much
<30% Cost of Current Servers

1. Specialize HW with an FPGA Fabric
2. Keep Servers Homogeneous

Don’t Burn Too Much Power
<10% Power Draw (25W max, all from PCIe)

Don’t Break Anything
Work in existing servers
No Network Modifications
Do not increase hardware failure rate
Datacenter Servers

- Microsoft Open Compute Server
- 1U, ½ wide servers
- Enough space & power for ½ height, ½ length PCIe card
- Squeeze in a single FPGA
- Won’t fit (or power) GPU
• Two 8-core Xeon 2.1 GHz CPUs
• 64 GB DRAM
• 4 HDDs, 2 SSDs
• No cable attachments to server
Catapult FPGA Accelerator Card

- Altera Stratix V D5
  - 172,600 ALMs
  - 2,014 M20Ks
  - 1,590 DSPs
- PCIe Gen 3 x8
- 8GB DDR3-1333
- Powered by PCIe slot
- Torus Network
Scalable Reconfigurable Fabric

- 1 FPGA board per Server
- 48 Servers per ½ Rack
- 6x8 Torus Network among FPGAs
  - 20 Gb/s over SAS SFF-8088 cables

Data Center Server (1U, ½ width)
+ An Elastic Reconfigurable Fabric

Web Search Pipeline

CPU
CPU
CPU
CPU

Top-of-Rack Switch (TOR)

PCIe (8.0 GB/s)
An Elastic Reconfigurable Fabric

Math Acceleration Service

Physics Engine

Comp. Vision Service

Web Search Pipeline

CPU

CPU

CPU

CPU

PCIe (8.0 GB/s)

SLIII (2.0 GB/s)

400 ns latency/hop
Shell & Role

- **Shell** handles all I/O & management tasks
- **Role** is only application logic
- FIFO access to Shell
- Role is Partial Reconfig boundary

![Diagram](image-url)
Bing Document Ranking Flow

Selection as a Service (SaaS)
- Find all docs that contain query terms,
  - Filter and select candidate documents for ranking

Ranking as a Service (RaaS)
- Compute scores for how relevant each selected document is for the search query
  - Sort the scores and return the results

Query → Selected Documents → 10 blue links
FE: Feature Extraction

- Document
- 

~4K Dynamic Features

FFE: Free-Form Expressions

~2K Synthetic Features

MLS: Machine Learning Scoring

Score

Query: “FPGA Configuration”

Number of Occurrences

- `NumberOfOccurrences_0 = 7`
- `NumberOfOccurrences_1 = 4`
- `NumberOfTuples_0_1 = 1`

---

A field-programmable gate array (FPGA) is an integrated circuit designed to be configured by the customer or designer after manufacturing—hence field-programmable. The FPGA configuration is generally specified using a hardware description language (HDL), similar to that used for an application-specific integrated circuit (ASIC) circuit diagrams were previously used to specify the configuration, as they were for ASICs, but this is increasingly rare.

FPGAs can be used to implement any logical function that an ASIC could perform. The ability to update the functionality after shipping, partial reconfiguration, of a portion of the design and the low non-recurring engineering costs relative to an ASIC design not to mention the generally higher unit cost, offers advantages for many applications.

FPGAs contain programmable logic components called logic blocks, and a hierarchy of reconfigurable interconnects that allow the blocks to be wired together—somewhat like many (changeable) logic gates that can be inter-wired in (many) different configurations. Logic blocks can be configured to perform complex combinational functions, or merely simple logic gates like AND and XOR. In most FPGAs the logic blocks also include memory elements, which may be simple flip-flops or more complex blocks of memory.

In addition to digital functions, some FPGAs have analog features. The most common analog feature is programmable biasing and drive strength on each output pin, allowing the engineer to set slow rates on lightly loaded pins that would otherwise cause unacceptable, and to set stronger, faster rates on heavily loaded pins on high-
FFE: Free Form Expressions

FFE #1 = \(\frac{2 \times \text{NumberOfOccurrences}_0 + \text{NumberOfOccurrences}_1}{2 \times \text{NumberOfTuples}_0_1}\)

\[
\begin{align*}
\text{NumberOfOccurrences}_0 &= 7 \\
\text{NumberOfOccurrences}_1 &= 4 \\
\text{NumberOfTuples}_0_1 &= 1
\end{align*}
\]

FFE #1 = 9
Feature Extraction Accelerator

- 196 feature families
- 54 state machines
- 2.6K dynamic features extracted in less than 4us (~600us in SW)
FFE Engines

- Softcore for multi-threaded throughput
- 4 HW threads per core
- 6 cores share a complex ALU
- log, divide, exp, float/int conversions
- 10 clusters (240 HW threads) per FPGA
FPGA Accelerator for RaaS

Document

FE: Feature Extraction

FFE: Free-Form Expressions

MLS: Machine Learning Scoring

Score

8-Stage Pipeline

FPGA 0

FPGA 1

FPGA 2

FPGA 3

FPGA 4

FPGA 5

FPGA 6

FPGA 7

Route to Head

Route to Head

Document Scoring Request

Return Score

Document Scoring Request

Return Score

RaaS Servers

Server

Server

Server

Server

Server

Server

Server

Server

15TH ANNUAL
Microsoft Research Faculty Summit 2014
Scalable Deployment Challenges

• **Issues with Spanning Multiple FPGAs**
  - Health monitor to detect stalled pipelines
  - Reconfiguration protocol to remove lockups
  - Re-mapper shifts images on machine failure

• **General Issues with an FPGA Fabric**
  - PCIe driver tuning for FPGA configuration
  - SEU scrubbing of the FPGA
  - Wiring and board check at integration
Accelerating Large-Scale Services – Bing Search

1,632 Servers with FPGAs Running Bing Page Ranking Service (~30,000 lines of C++)
Conclusions

• Hardware specialization is a (the?) way to gain efficiency and performance
• An FPGA fabric offers a flexible, elastic pool of resources to accelerate services
• Results for one service: ½ the number of ranking servers, lower latency, lower variance
• Proven scalability, proven resilience, and huge potential for future apps
But when will an FPGA handle my Bing Search?
“This Isn’t A Toy”

- Bing is going into production with FPGAs

- Early 2015 – Bing will begin serving searches based on computed by the FPGA fabric
Huge thanks to our partners at

Microsoft

**Top Row:** Eric Peterson, Scott Hauck, Aaron Smith, Jan Gray, Adrian M. Caulfield, Phillip Yi Xiao, Michael Haselman, Doug Burger

**Bottom Row:** Joo-Young Kim, Stephen Heil, Derek Chiou, Sitaram Lanka, Andrew Putnam, Eric S. Chung,

**Not Pictured:** Kypros Constantinides, John Demme, Hadi Esmaeilzadeh, Jeremy Fowers, Gopi Prashanth Gopal, Amir Hormati, James Larus, Simon Pope, Jason Thong
Save the planet and return your name badge before you leave (on Tuesday)