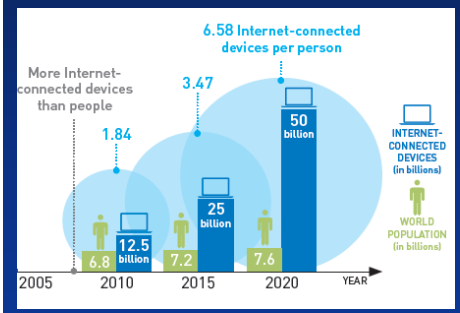
Accelerate Cloud Computing with MapReduce Accelerator

Christoforos Kachris, Ph.D.
Senior Research Engineer
Democritus University of Thrace, Greece

European Union
European Social Fund
Co- financed by Greece and the European Union
OPERATIONAL PROGRAMME EDUCATION AND LIFELONG LEARNING
investing in knowledge society
MINISTRY OF EDUCATION, LIFELONG LEARNING AND RELIGIOUS AFFAIRS
MANAGING AUTHORITY
NSRF 2007-2013
programme for development
EUROPEAN SOCIAL FUND
GSRT
GENERAL SECRETARIAT RESEARCH AND TECHNOLOGY

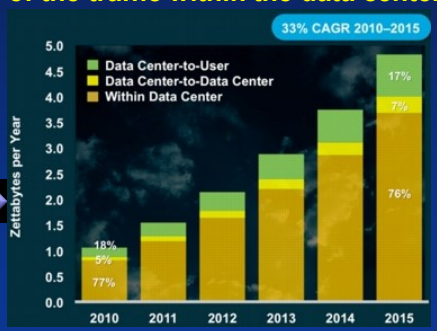
Network traffic growth in data centers

- New sources of network traffic leads to significant increase in the data center network traffic



[Source: D. Kilper et al., "Optical Networks come of age", Optics and Photonics, News, September 2014]

¾ of the traffic within the data center

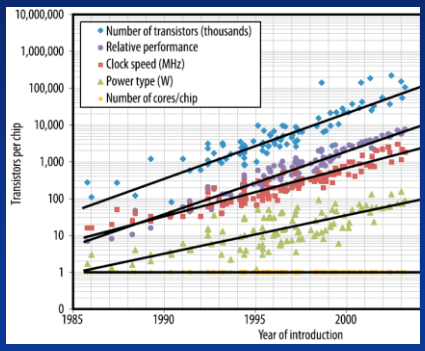


[Source: Cisco Global Cloud Index: Forecast and Methodology, 2012–2017]

Santa Clara, CA USA
November 2014

The era of scale-up

- 1970 – 2005 [Increase clock frequency]
  - Moore's law (2x transistors every 18m)
  - Higher frequency => higher performance

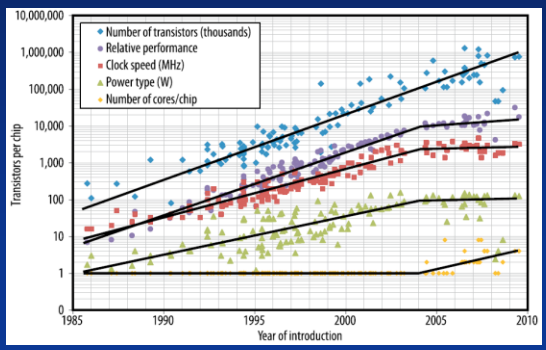


[Ref: S.H. Fuller, L.I. Millett, "Computing Performance: Game over or Next Level, Computer 2011]

3

The era of scale-out (multi-cores)

- 2005 – 2014 [Increase number of cores]
  - Chip energy consumption is limited by thermal constraints
  - More cores => **slightly** higher performance (limited)

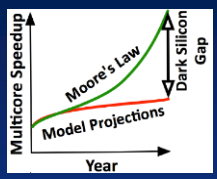


[Ref: S.H. Fuller, L.I. Millett, "Computing Performance: Game over or Next Level, Computer 2011]

4

The era of Dark Silicon

- Keep increasing core counts can't directly translate into performance improvement (interconnects overhead, off-chip bandwidth, power constraints)
- Dark Silicon: We can put more cores in the same die but due to power constraints we cannot afford to power the whole chip
- Solution: Use abundant die area by utilizing **application-specific** cores/accelerators
- Dynamically power up only the specific modules designed for the given workload



[Ref: H. Esmaeilzadeh et al., Dark Silicon and the End of Multicore Scaling, ISCA 2011,
N. Hardavellas et al., Toward Dark Silicon in Servers, Micro 2011,
M. Ferdman et al., "A Case for Specialized Processors for Scale-out Workloads, Micro, May/June 2014,
B. Falsafi, Big Data and Dark Silicon, HiPEAC CSW Athens, 2014]

MapReduce applications

- MapReduce is widely deployed to data center applications
- MapReduce can also be used for multi-core programming (e.g. Phoenix MapReduce from Stanford)

hadoop

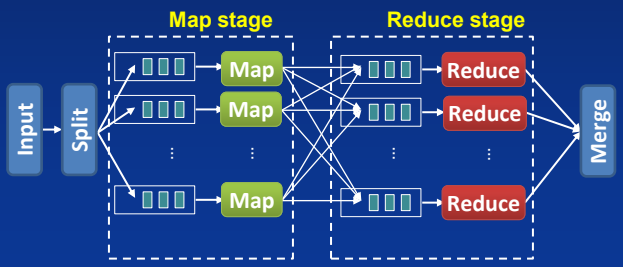
Infinispan

mongoDB

CouchDB
relax

MapReduce Framework

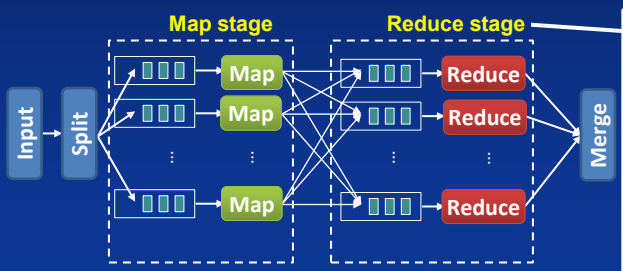
- MapReduce is a programming model for processing large data sets using high number of nodes.
- The user specifies the **Map** and the **Reduce** functions and the MapReduce scheduler performs the distribution of the tasks to the processors



Santa Clara, CA USA
November 2014
7

MapReduce Applications

- **Map** function processes a portion of the data and generate a set of intermediate key/value pairs.
- **Reduce** function merges all intermediate values associated with the same intermediate key.



1. Search key
2. Load (key,value) from the memory
3. Process the value
4. Store back the (key, value) to the memory

Santa Clara, CA USA
November 2014
8

MapReduce Programming model

Original Code:

```
Map{
  Process data
  Emit_Intermediate (key, value);
}
Reduce{
  process(key, value);
}
```

Most of the Reduce function are common:
- **Accumulate**
- **Average**

Using a hardware accelerator for the Reduce functions

Code using the accelerator:

```
Map{
  Process data
  Emit_Intermediate_accel(key, value);
}
```

- **Map** functions are executed in software => flexibility
- **Reduce** functions are executed in hardware => performance

9

MapReduce accelerator

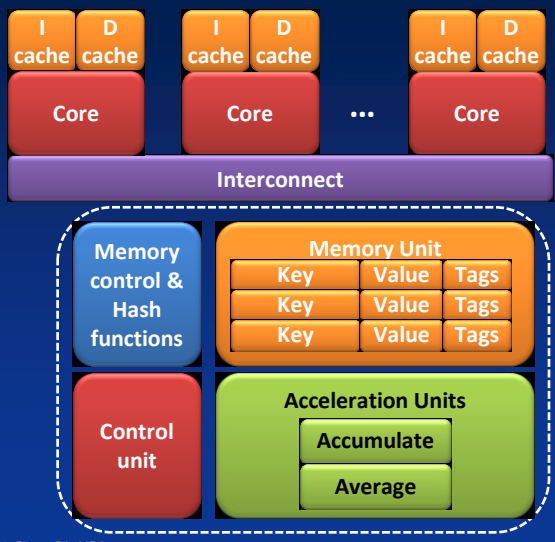
- A hardware accelerator has been developed and incorporated to multi-core server processors that can be used to:

  - Allow **faster indexing**, **searching** of the (key, value) pairs
  - Allow **faster processing** of the values (accumulation, average, etc.)
  - **Offload** the processor from the *Reduce* functions (more computing power for *Map* functions)
  - Free space in cache => **less cache misses**

Santa Clara, CA USA
November 2014

10

MapReduce Accelerator in SoC



Map
Functions in SW for higher flexibility

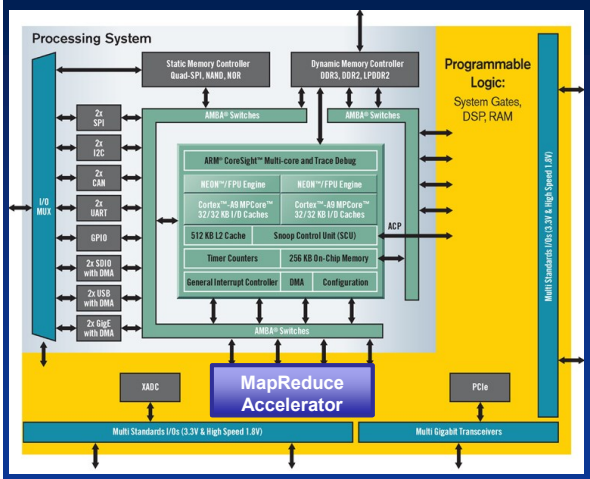
Reduce accel.
Functions in HW for accelerating processing

Santa Clara, CA USA
November 2014

[Under provisional patent, USPTO]

11

ARM meets FPGAs: Xilinx Zynq SoC



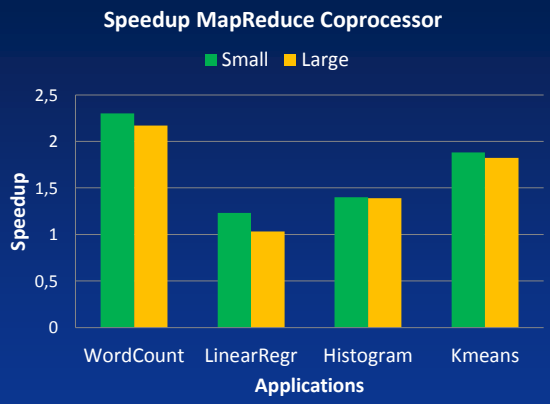
- Performance evaluation in hardware using an Zynq SoC (FPGA with 2 ARM Cortex A9 cores)
- Under Linux
- Phoenix MapReduce benchmark

Santa Clara, CA USA
November 2014

12

Performance evaluation

- Improved performance (up to 2.3x speedup)
- Lower energy consumption
- Alleviate the processor for more Map tasks

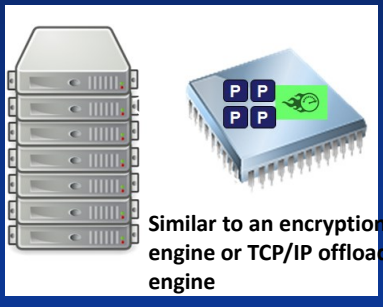


Performance evaluation of the Phoenix MapReduce in the Zedboard with Zynq7000

Ref: C. Kachris et al. "A Configurable MapReduce Accelerator for Multi-core FPGAs", FPGA 2014
C. Kachris et al., "A Reconfigurable MapReduce Accelerator for multi-core all-programmable SoCs", SOC 2014

Utilization of MapReduce accelerator

- **Fine grain (in the SoC)**
  As co-processor in multi-core Server Processors
- **Coarse grain (in the rack)**
  As a stand-alone blade accelerator in the Rack



Similar to an encryption engine or TCP/IP offload engine

Benefits and Opportunities

- Applications specific accelerators can be incorporated to future multi-core servers/microservers Chips or Blades to sustain future server's workload
- Need for Open Interface for accelerators

Thank you!

The research project is implemented within the framework of the Action «Supporting Postdoctoral Researchers» of the Operational Program "Education and Lifelong Learning" (Action's Beneficiary: General Secretariat for Research and Technology), and is co-financed by the European Social Fund (ESF) and the Greek State.