NEC

Me, Myself and High Performance Network Functions for Programmable Dataplanes

Salvator Galea

Introduction

Salvator Galea : 1st year student for CPGS (PhD)

Andrew W. Moore : Supervisor, Reader

Gianni Antichi : 2Nd Advisor,
Senior Research Associate

UNIVERSITY OF CAMBRIDGE

Department of Computer Science and Technology
NetOS group, SRG

Introduction

Salvator Galea : 1st year student for CPGS (PhD)

Andrew W. Moore : Supervisor, Reader

Gianni Antichi : 2Nd Advisor,
Senior Research Associate

UNIVERSITY OF CAMBRIDGE

Department of Computer Science and Technology
NetOS group, SRG

Introduction

Salvator Galea : 1st year student for CPGS (PhD)

Andrew W. Moore : Supervisor, Reader

Gianni Antichi : 2Nd Advisor,
Senior Research Associate

Department of Computer Science and Technology
NetOS group, SRG

Roberto Bifulco : Industrial Advisor, Senior Researcher

Agenda

- NetFPGA project
- Emu
- OSNT
- Future work

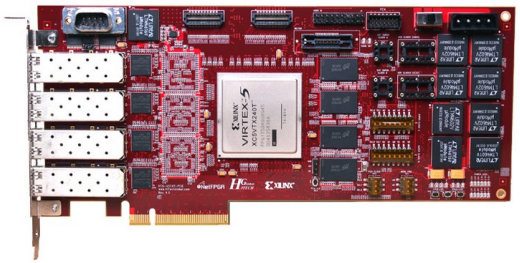
NetFPGA

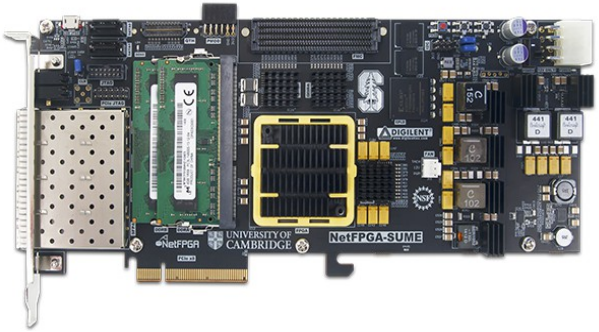
A line-rate, flexible, open-networking platform for teaching and research

NetFPGA

NetFPGA

A line-rate, flexible, open-networking platform for teaching and research

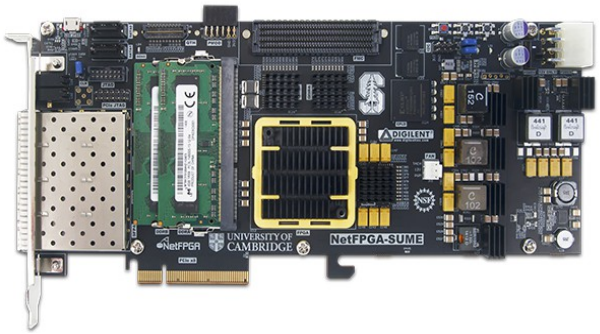




NetFPGA

NetFPGA

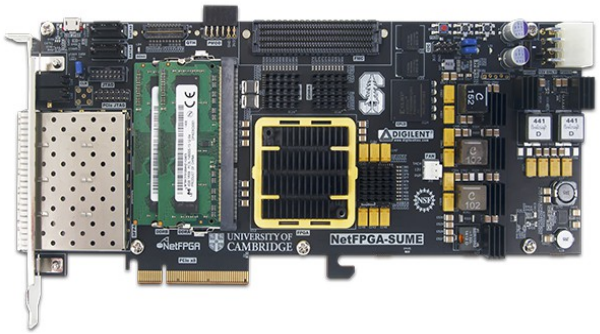
So what?



- Network Interface Card
- Hardware Accelerated Linux Router
- IPv4 Reference Router
- Traffic Generator
- Openflow Switch
- More Projects
- Add Your Project

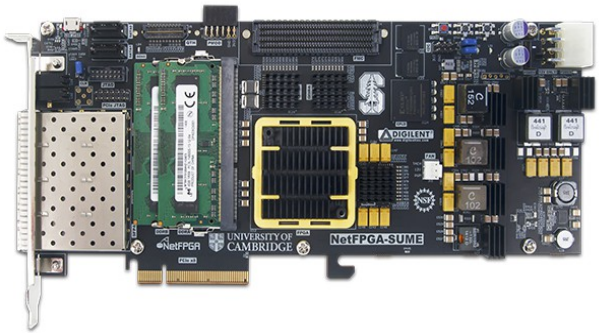
NetFPGA

So who, how, why?



- Researchers, Teachers, Students
- To build modular designs
- To prototype new network systems and measure network performance

NetFPGA



NetFPGA

Community

- Network Interface Card
- Hardware Accelerated Linux Router
- IPv4 Reference Router
- Traffic Generator
- Openflow Switch
- More Projects
- Add Your Project

NetFPGA SUME Community (since Feb 2015)
Over 600 users, using over 300 cards at
200 universities in 47 countries

EMU

Emu: Rapid Prototyping of Networking Services

Published in USENIX ATC'17

Nik Sultana, Salvator Galea, David Greaves, Marcin Wojcik, Jonny Shipton, Richard Clegg, Luo Mai, Pietro Bressana, Robert Soule, Richard Mortier, Paolo Costa, Peter Pietzuch, Jon Crowcroft, Andrew W Moore, Noa Zilberman

Using FPGAs for acceleration

Using FPGA is great because of...

- Programmability
- Performance
- Predictability
- Power efficiency

Catapult FPGA Accelerator

But why FPGAs have never became mainstream?

Programming FPGAs

– It is because of the cost doing FPGA engineering.

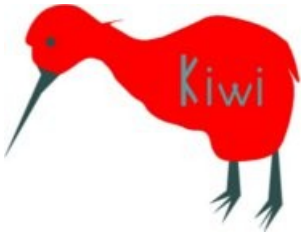
– The programming and reprogramming is done in complex, low-level hardware description languages like Verilog and VHDL.

– Lack of FPGA developers compared to number of software developers.

Led to the development of *High Level Synthesis* tools

- Use High-level programming languages
- For Scientific Applications

Kiwi



- HLS open-source compiler
- Transforms C# and F# to HDL
- Any .NET bytecode to Verilog

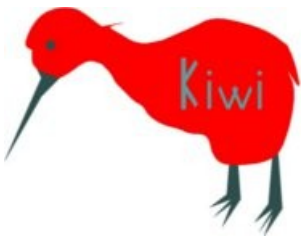
David Greaves, and Satnam Singh. "Distributing C# methods and threads over Ethernet-connected FPGAs using Kiwi." Formal Methods and Models for Codesign (MEMOCODE), 2011.

Accelerating network services

- Write the network application in C#
- Compile to Verilog using Kiwi Compiler
- Run on the FPGA
- End of story :)

Accelerating network services

- ~~Write the network application in C#~~
- ~~Compile to Verilog using Kiwi Compiler~~
- ~~Run on the FPGA~~
- ~~End of story :)~~

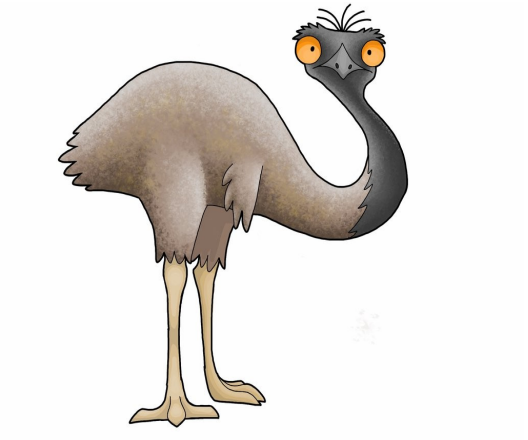


Kiwi is just a compiler, not a linker, neither provides networking libraries

Emu

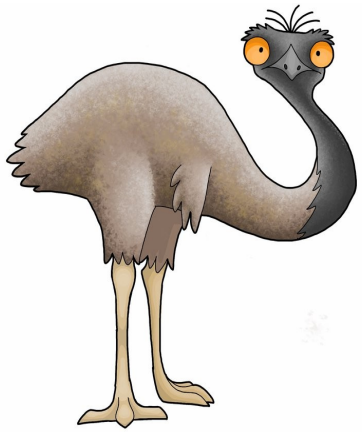
EMU

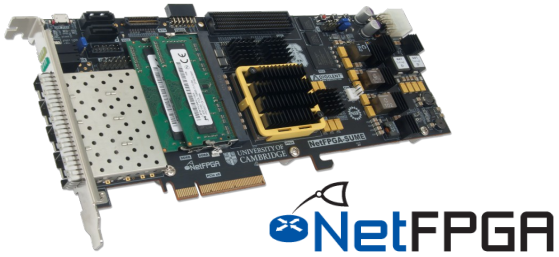
The Network Library for Kiwi



Emu: Accelerating Network Services

Goal: Rapid prototyping of network services

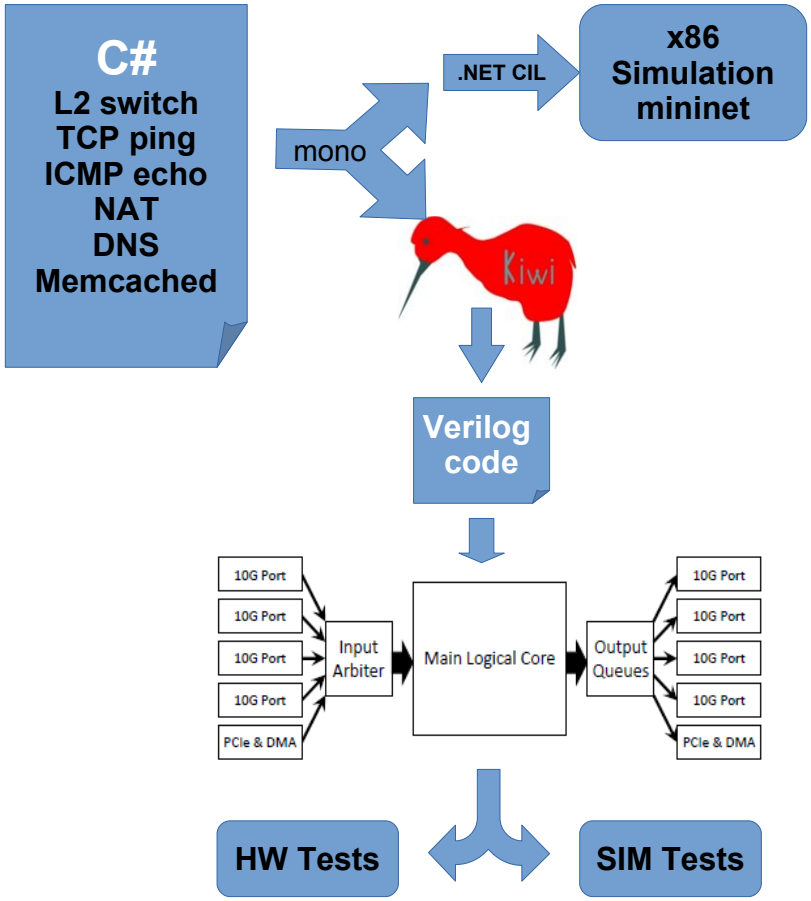




Emu provides a framework with:

- A library of functions
- Compiling to multiple targets
- A runtime environment
- Automatic implementation on FPGA
- Advanced debug capabilities

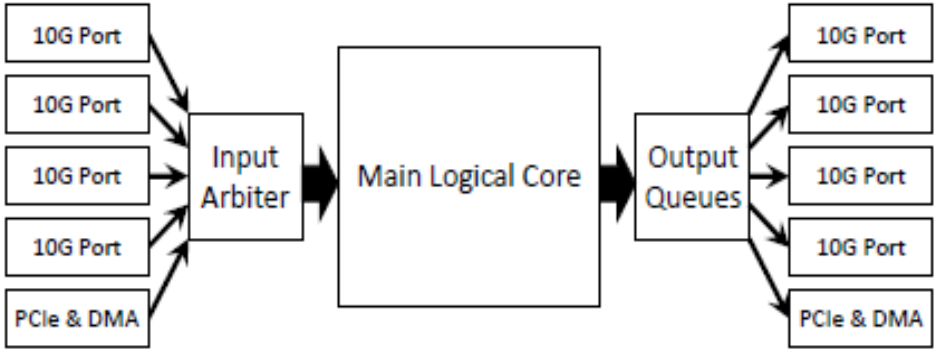
Emu Framework



- **Emu** complements a high-level synthesis compiler with a library to support network-related high level programming.
- **Emu** maps programming and networking abstractions to bus protocols, memory interfaces, and basic frame-handling functionality used on the **NetFPGA-SUME**.
- **Emu** framework provides a reference design path through which the user can compile the **C#** code and run it directly on the **NetFPGA-SUME** without further development intervention.

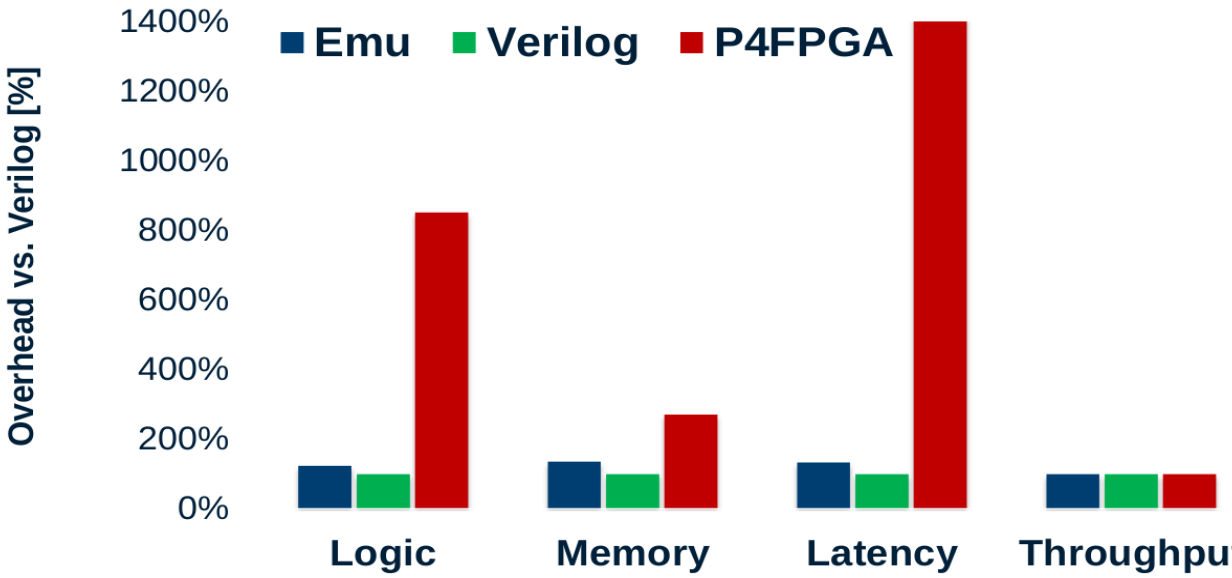
Emu: Hardware Integration

- Multiple FPGA targets using templates
- Support integration with hardware IP cores
- Support of multicore Emu cores
- Library of networking functions



NetFPGA **Datapath**

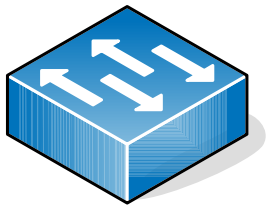
Emu: Efficiency Comparison – Layer-2 Switch



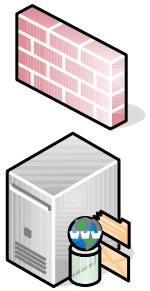
	Emu	Verilog	P4FPGA
Logic resources	3509	2836	24161
Memory resources	118	87	236
Latency [cycles]	8	6	85
Throughput [Mpps]	59.52	59.52	59.52

Emu: Use Cases

- Networking devices
  - Layer-2 Switch

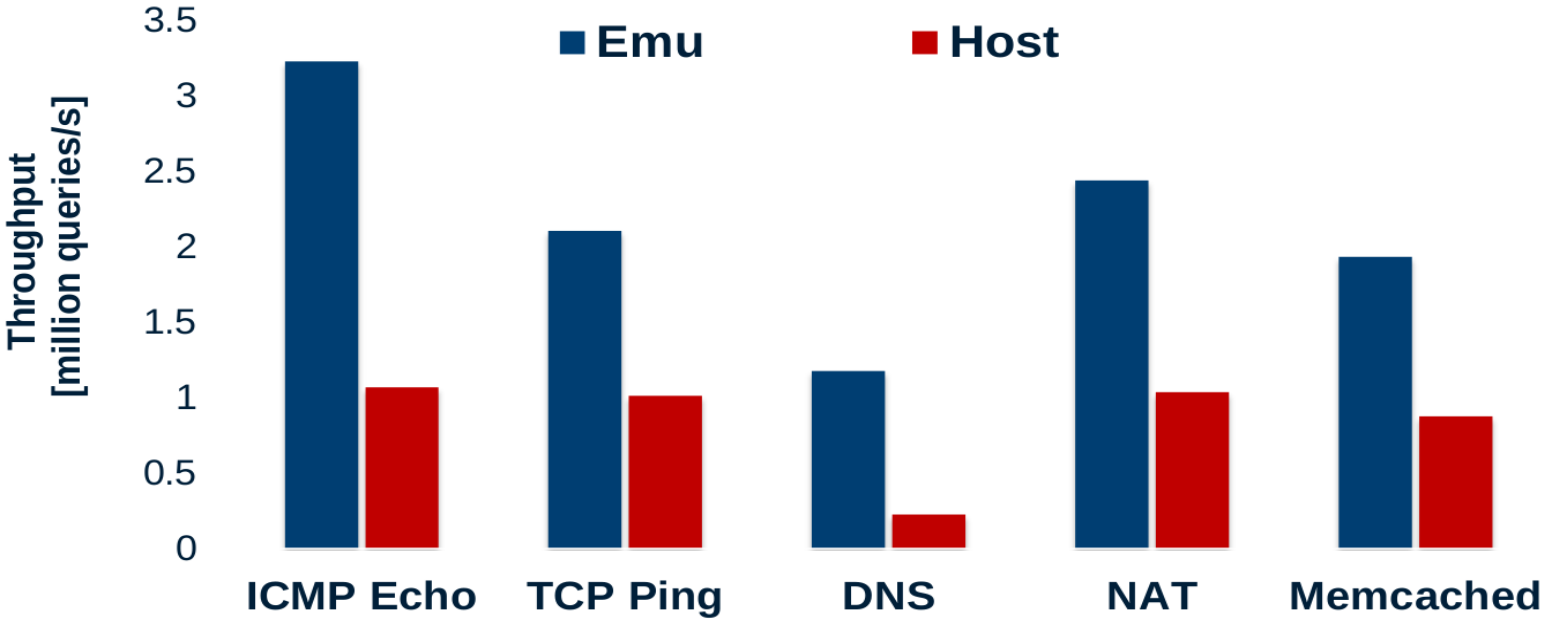


- Network Services
  - NAT
  - DNS Server
  - ICMP echo reply



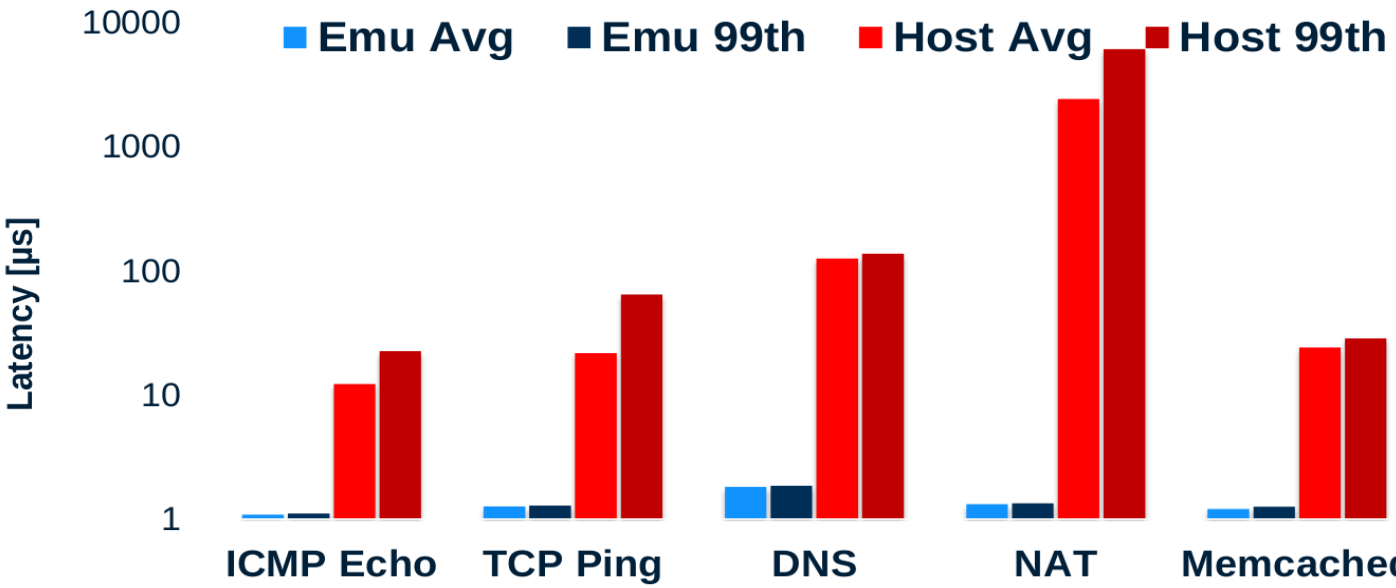
- Performance sensitive applications
  - Memcached Server

Emu: Use cases evaluation – Throughput



	Emu [mq/s]	Host [mq/s]
ICMP Echo	3.226	1.068
TCP Ping	2.105	1.012
DNS	1.176	0.226
NAT	2.439	1.037
Memcached	1.932	0.876

Emu: Use cases evaluation – Latency



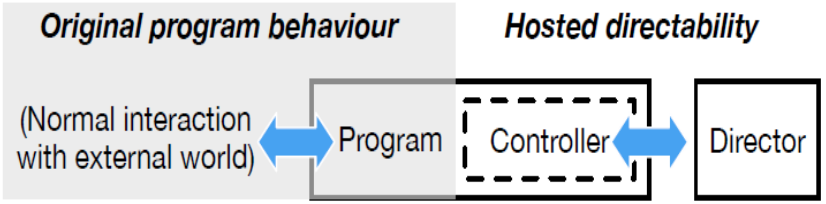
	Emu Avg [µs]	Emu 99th[µs]	Host Avg [µs]	Host 99th[µs]
ICMP Echo	1.09	1.11	12.28	22.63
TCP Ping	1.27	1.29	21.79	65.00
DNS	1.82	1.86	126.46	138.33
NAT	1.32	1.34	2444.76	6185.27
Memcached	1.21	1.26	24.29	28.65

Emu: What about Debugging

Every program comes with bugs

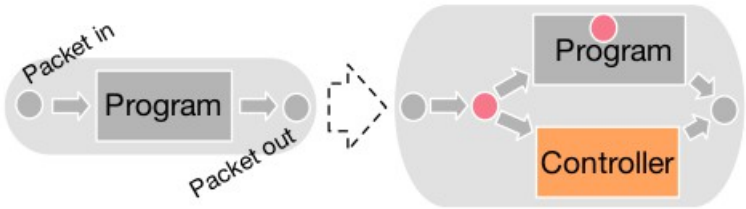
Emu: Extended Debug Capabilities

- Use *Directed Packets* to inspect the state of a device in the field
- Support *extension points* in the code
  - Observe the program from that point
  - Influence program state
- Example supported commands:
  - Print, trace, count, (un)break,(un)watch, backtrace
- Implemented using *an embedded controller* and *a program director*



Emu: Debug core overhead

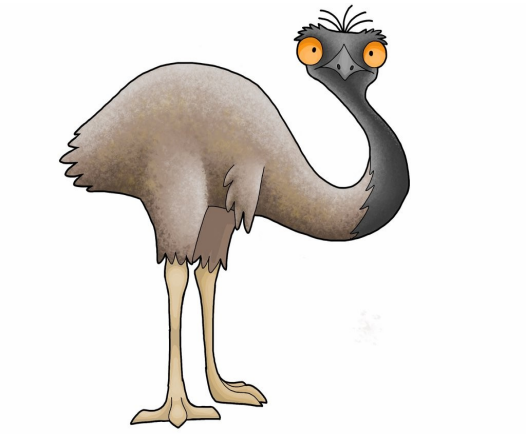
Artefact	Utilisation (%)	Performance (%)	
	Logic	Latency	Queries-per-sec
DNS	100.00	100.00	100.00
+R	103.40	100.00	100.00
+W	115.05	99.45	100.00
+I	109.79	99.45	100.00
Memcached	100.00	100.00	100.00
+R	99.17	100.00	100.00
+W	99.80	100.49	100.00
+I	100.63	100.00	100.00



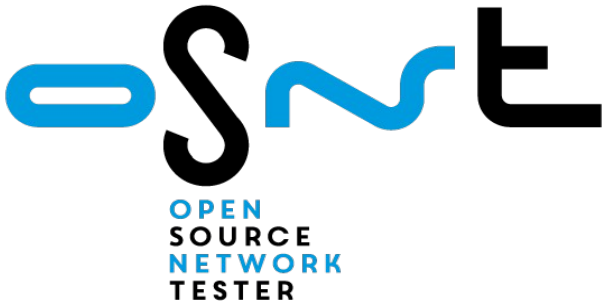
Emu: Conclusion

So what you get with the Emu framework?

- Rapid prototyping of networking services
- Code in .NET, compile to multiple targets
- Accelerates the development and debug process
- High throughput and low latency
- Open source



OSNT: Open Source Network Tester



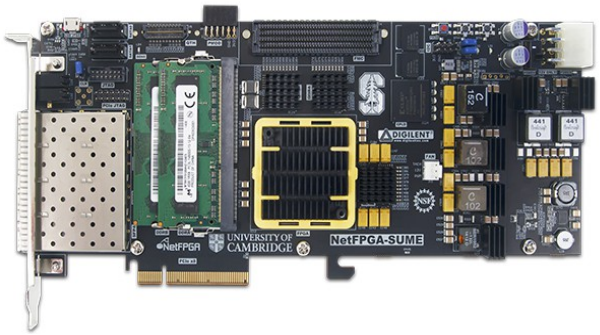
Open source hardware and software platform for network monitoring and testing

https://osnt.org

Low cost, flexible to update, scale-out, no CPU usage, nanosecond resolution measurements

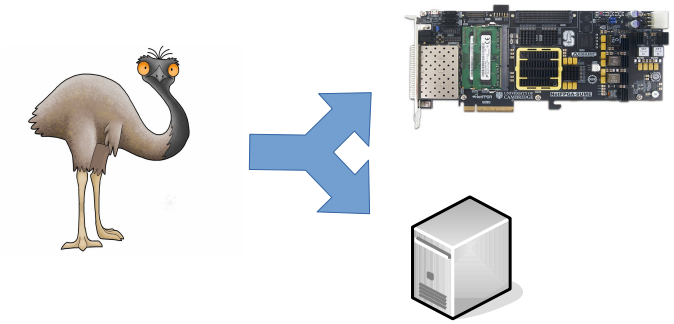
OSNT

- 4x10Gbps traffic generator.
- Capture card with high resolution timestamp (6.4nsec).
- GPS-ready synchronized measurement kit.

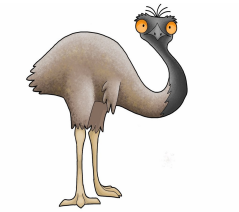


OSNT
OPEN SOURCE NETWORK TESTER

Future work



Hybrid solution for network applications
– Parts of the application run on HW
– Parts of the application run in host



Customized per network application generator and monitor tool
– more accurate latency predictor
– interactive traffic generator

Questions?