Data Center Traffic and Measurements: SoNIC

Hakim Weatherspoon

Assistant Professor, Dept of Computer Science

CS 5413: High Performance Systems and Networking

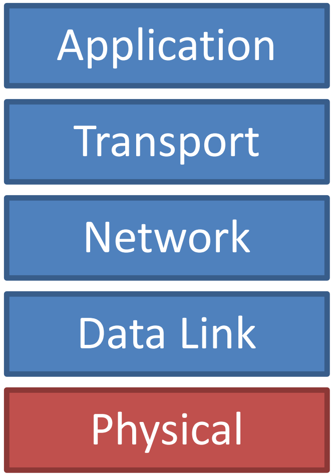
November 12, 2014

Slides from USENIX symposium on Networked Systems Design and Implementation (NSDI) 2013 presentation of “SoNIC: Precise Realtime Software Access and Control of Wired Networks,”

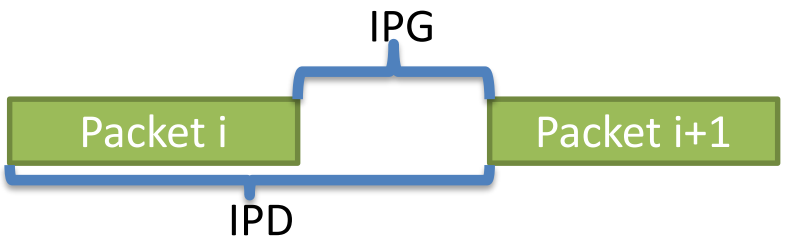
Goals for Today

- Analysis and Network Traffic Characteristics of Data Centers in the wild
  - T. Benson, A. Akella, and D. A. Maltz. In Proceedings of the 10th ACM SIGCOMM conference on Internet measurement (IMC), pp. 267-280. ACM, 2010.

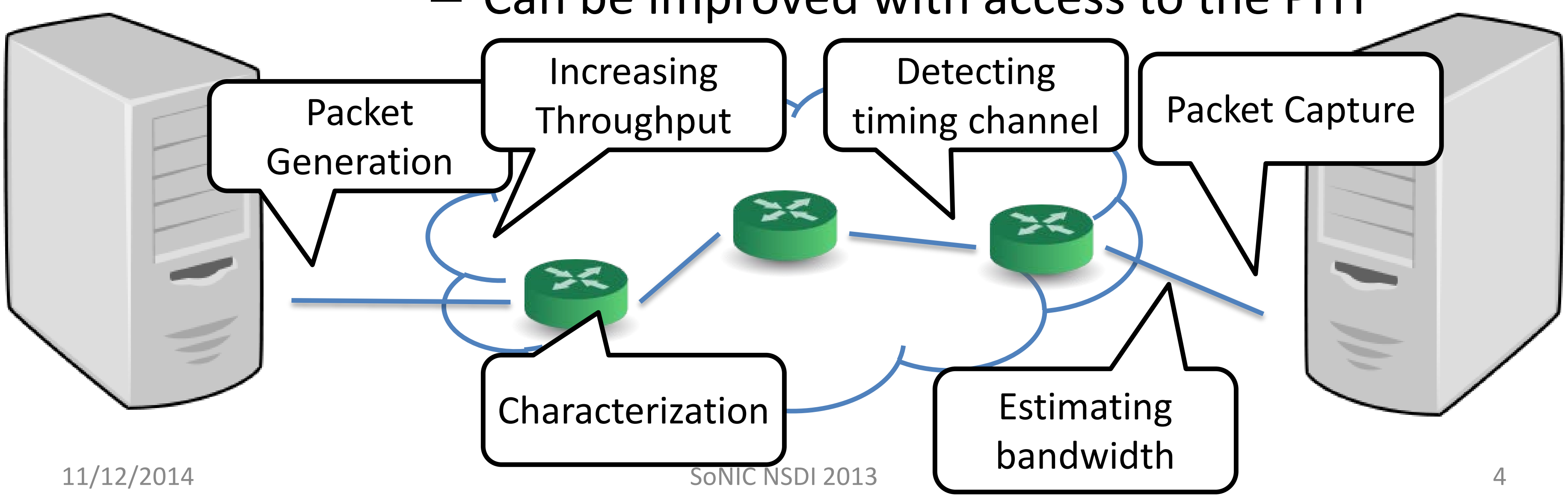
Interpacket Delay and Network Research



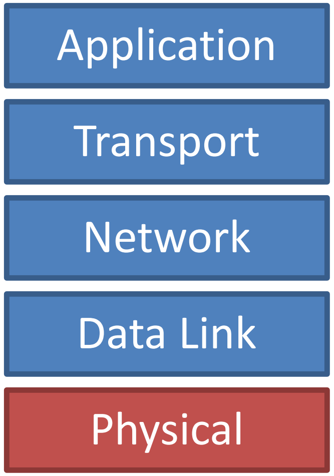
- Interpacket gap, spacing, arrival time, …



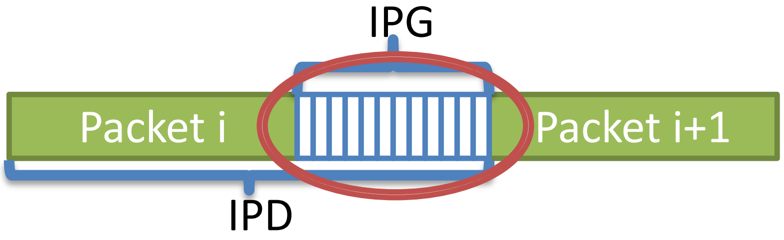
- Important metric for network research
  - Can be improved with access to the PHY



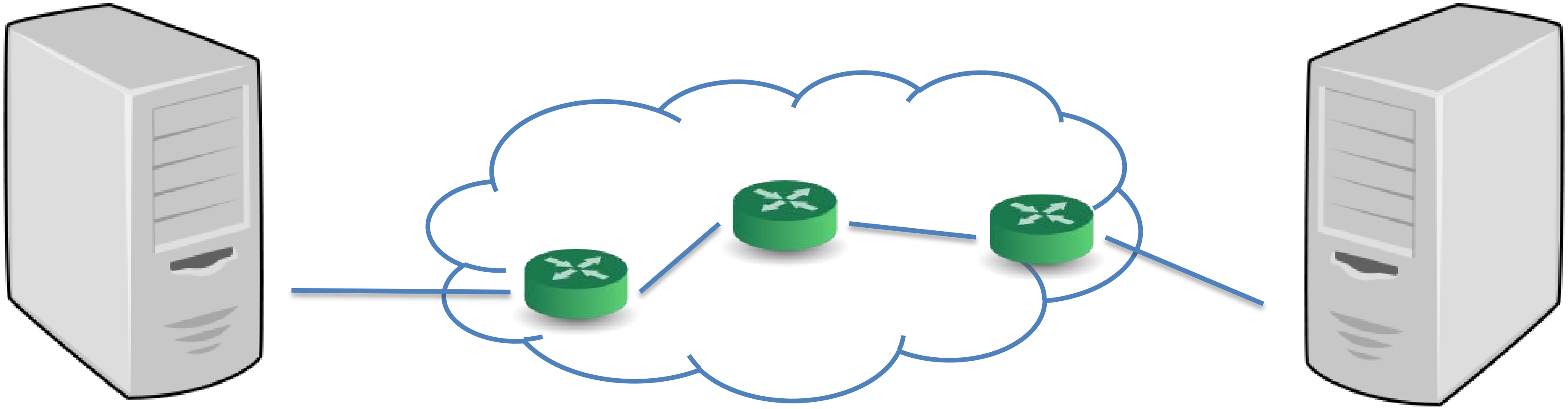
Network Research enlightened via the PHY



- Valuable information: Idle characters



  – Can provide precise timing base for control
    - Each bit is ~97 *ps* wide



Network Research enlightened via the PHY

- Application
- Transport
- Network
- Data Link
- Physical

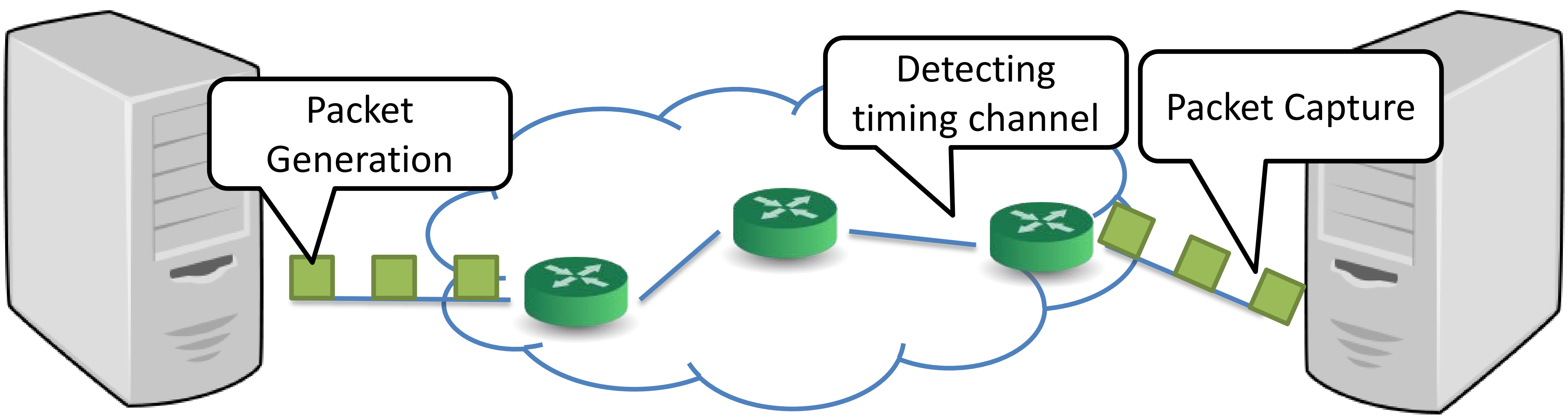
- Valuable informa… rs

12 /I/s = 100bits = 9.7ns

IPG

One Idle character (/I/) = 7~8 bits

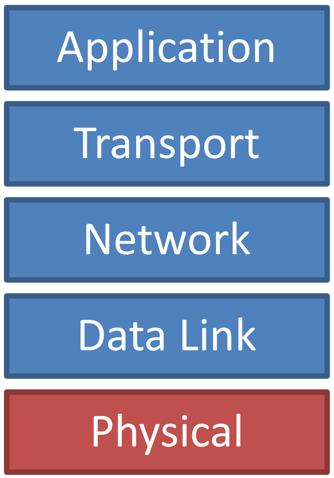
  – Ca… ming base for control
    - Each bit is ~97 *ps* wide



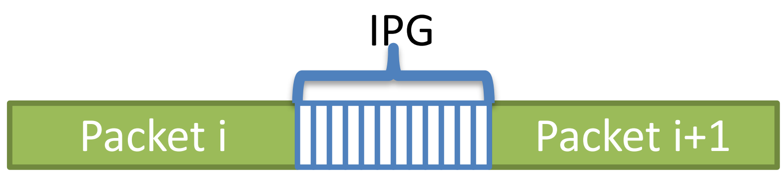
Principle #1: Precision

Precise network measurements is enabled via access to the physical layer (and the idle characters and bits within interpacket gap)

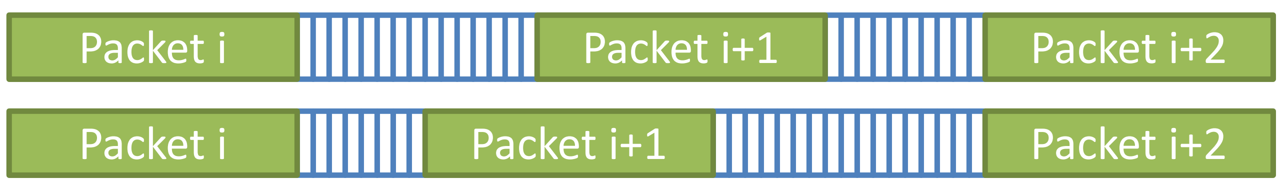
How to control the idle characters (bits)?



- Access to the entire stream is required



- Issue1: The PHY is simply a black box
  - No interface from NIC or OS
  - Valuable information is invisible (discarded)

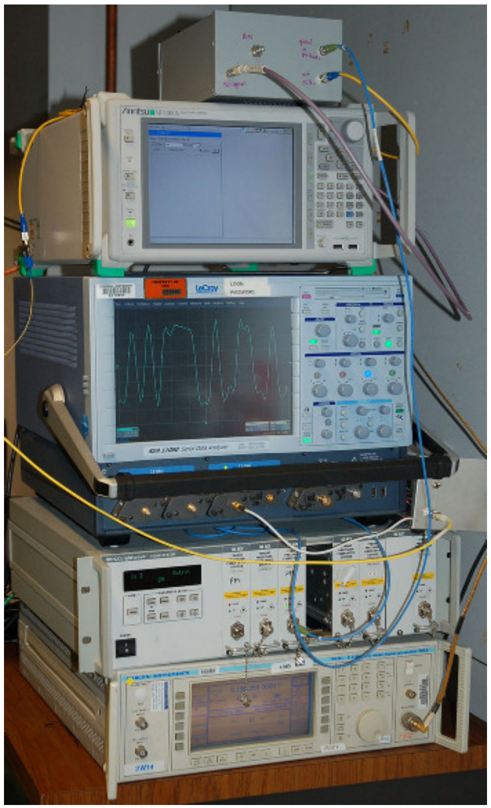


- Issue2: Limited access to hardware
  - We are network systems researchers
    a.k.a. we like software

Principle #2: Software

Network Systems researchers need software access to the physical layer

Precision + Software = Physics equipment???



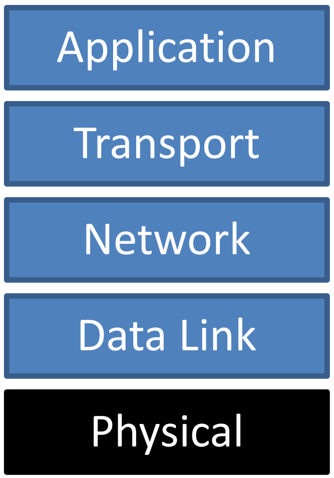
- BiFocals [IMC'10 Freedman, Marian, Lee, Birman, Weatherspoon, Xu]
  - Enabled novel network research
  - Precision + Software =

    Laser + Oscilloscope + Offline analysis
  - Allowed precise control in software
- Limitations
  - Offline (not *realtime*)
  - Limited Buffering
  - Expensive

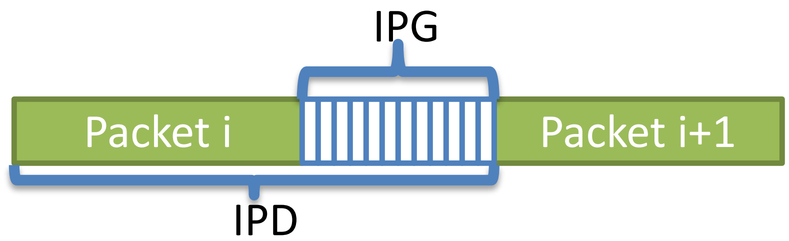
Principle #3: Realtime

Network systems researchers need access and control of the physical layer (interpacket gap) continuously in realtime

Challenge



- Goal: Control *every* bit in *software* in *realtime*



  – Enable novel network research

- Challenge
  – Requires unprecedented software access to the PHY

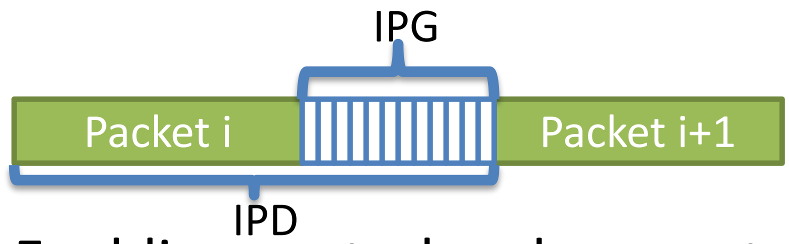
Outline

- Introduction
- SoNIC: Software-defined Network Interface Card
  - Background: 10GbE Network Stack
  - Design
- Network Research Applications
- Conclusion

SoNIC: Software-defined Network Interface Card

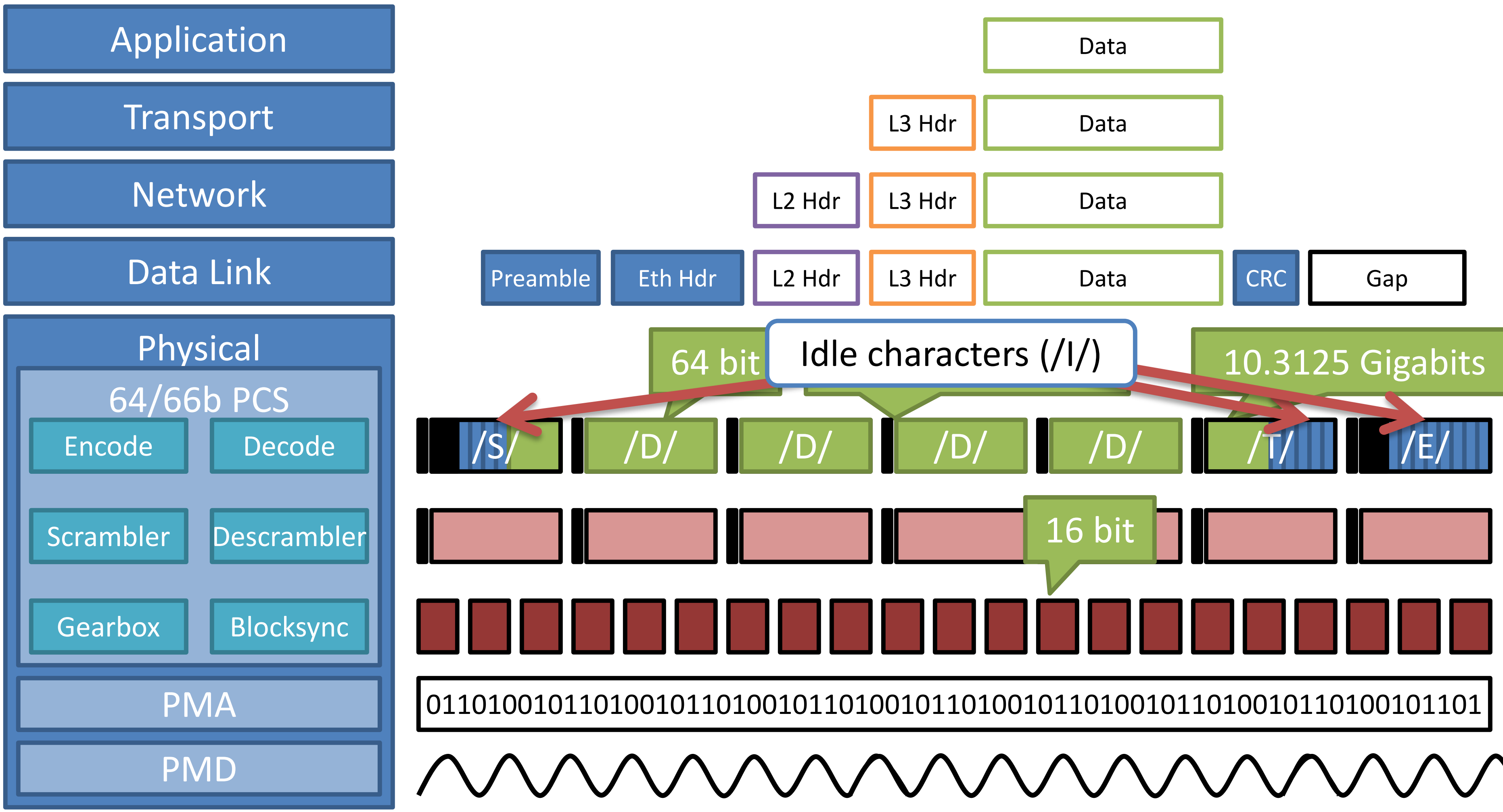
- Application
- Transport
- Network
- Data Link
- **Physical**

- Implements the PHY in software

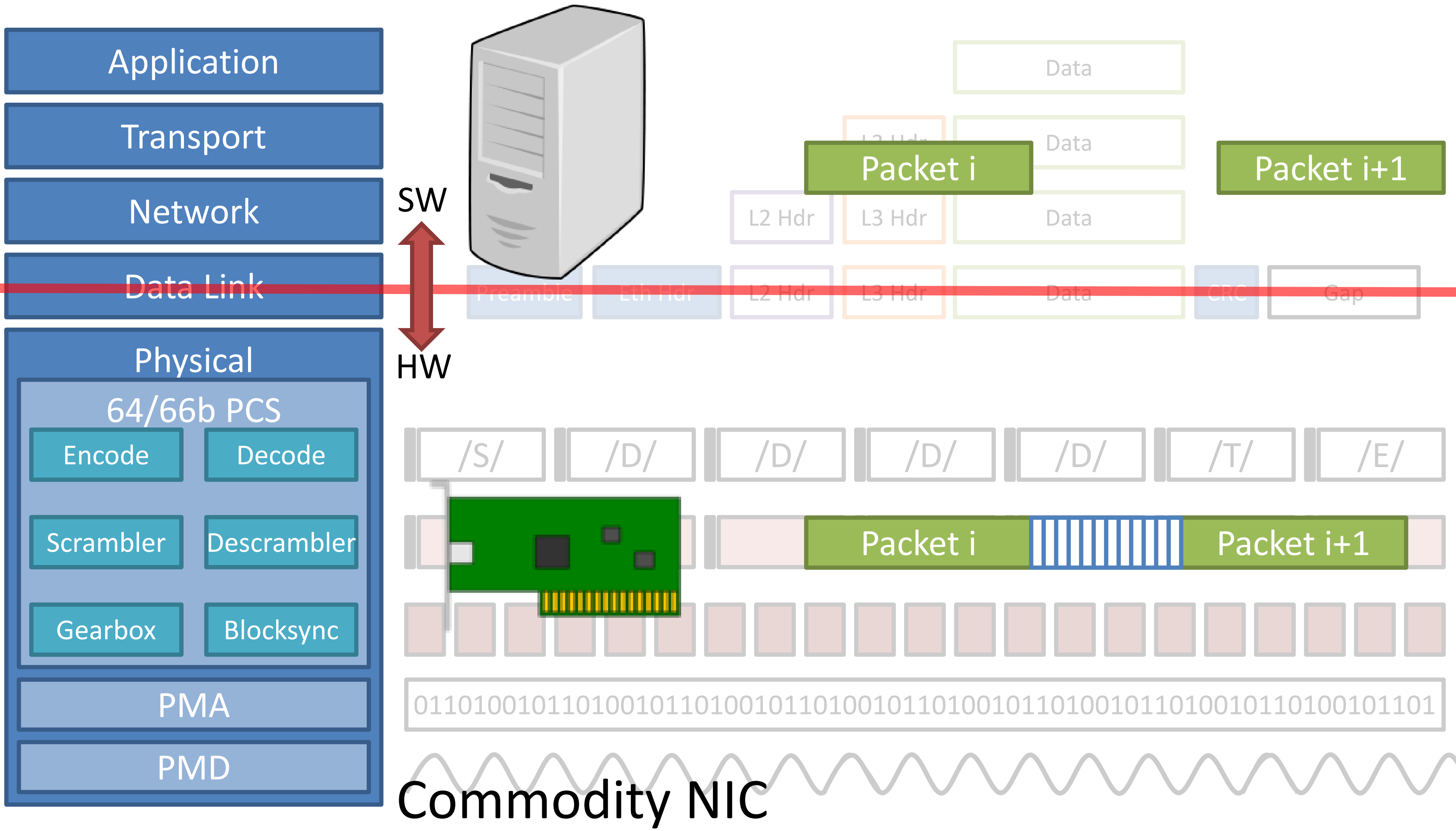


  - Enabling control and access to every bit in realtime
  - With commodity components
  - Thus, enabling novel network research
- How?
  - Backgrounds: 10 GbE Network stack
  - Design and implementation
    - Hardware & Software
    - Optimizations

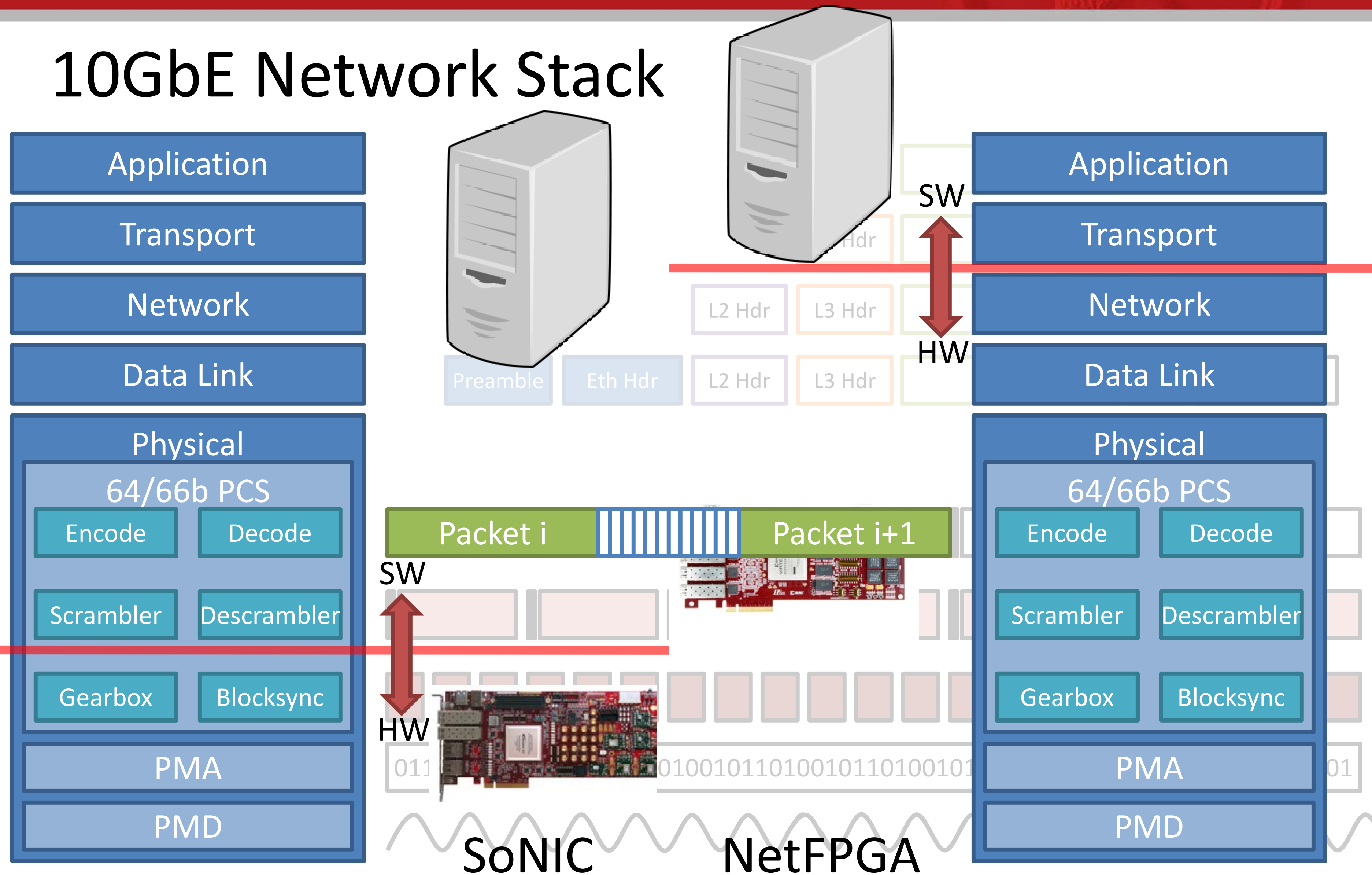
10GbE Network Stack



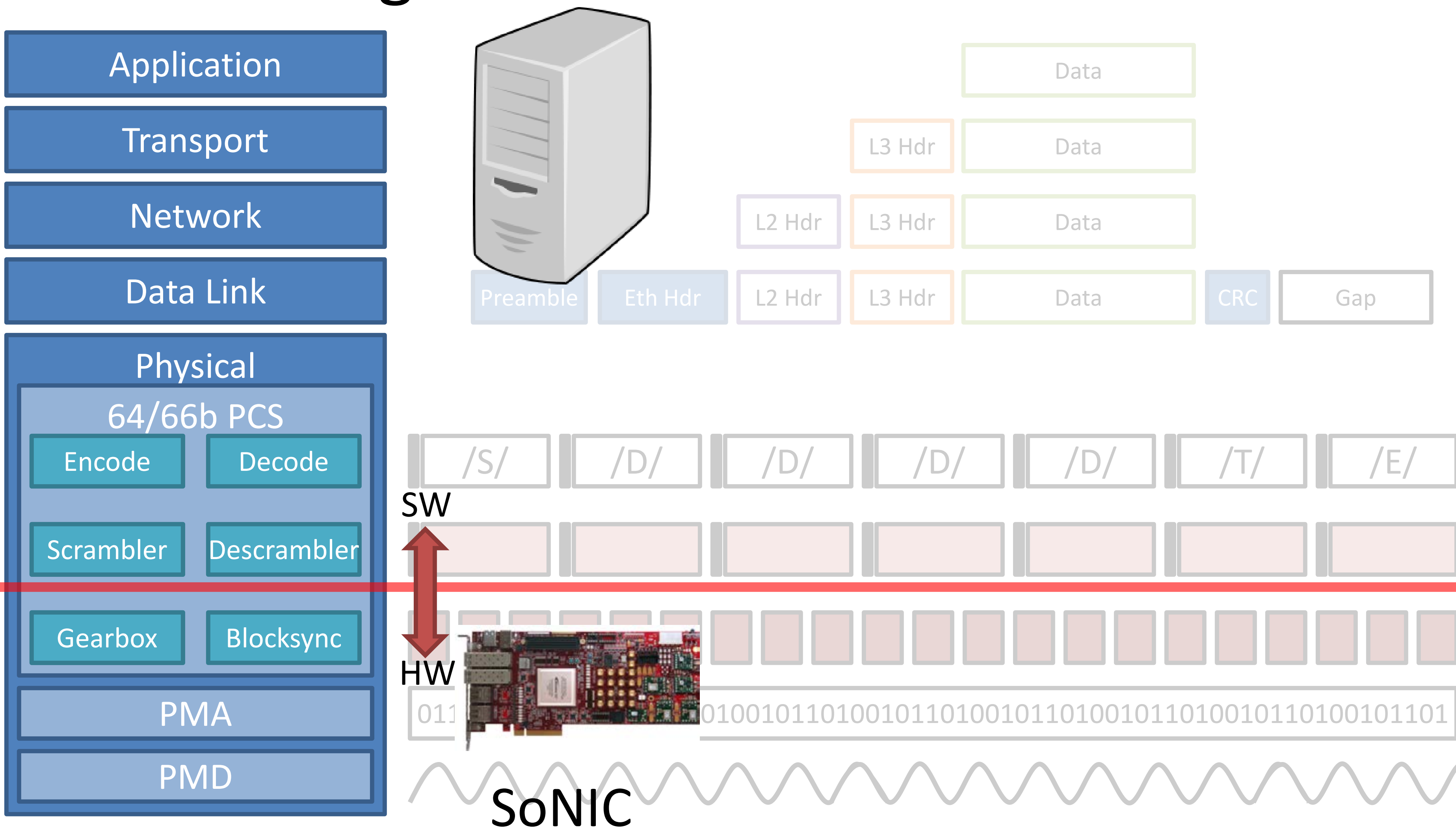
10GbE Network Stack



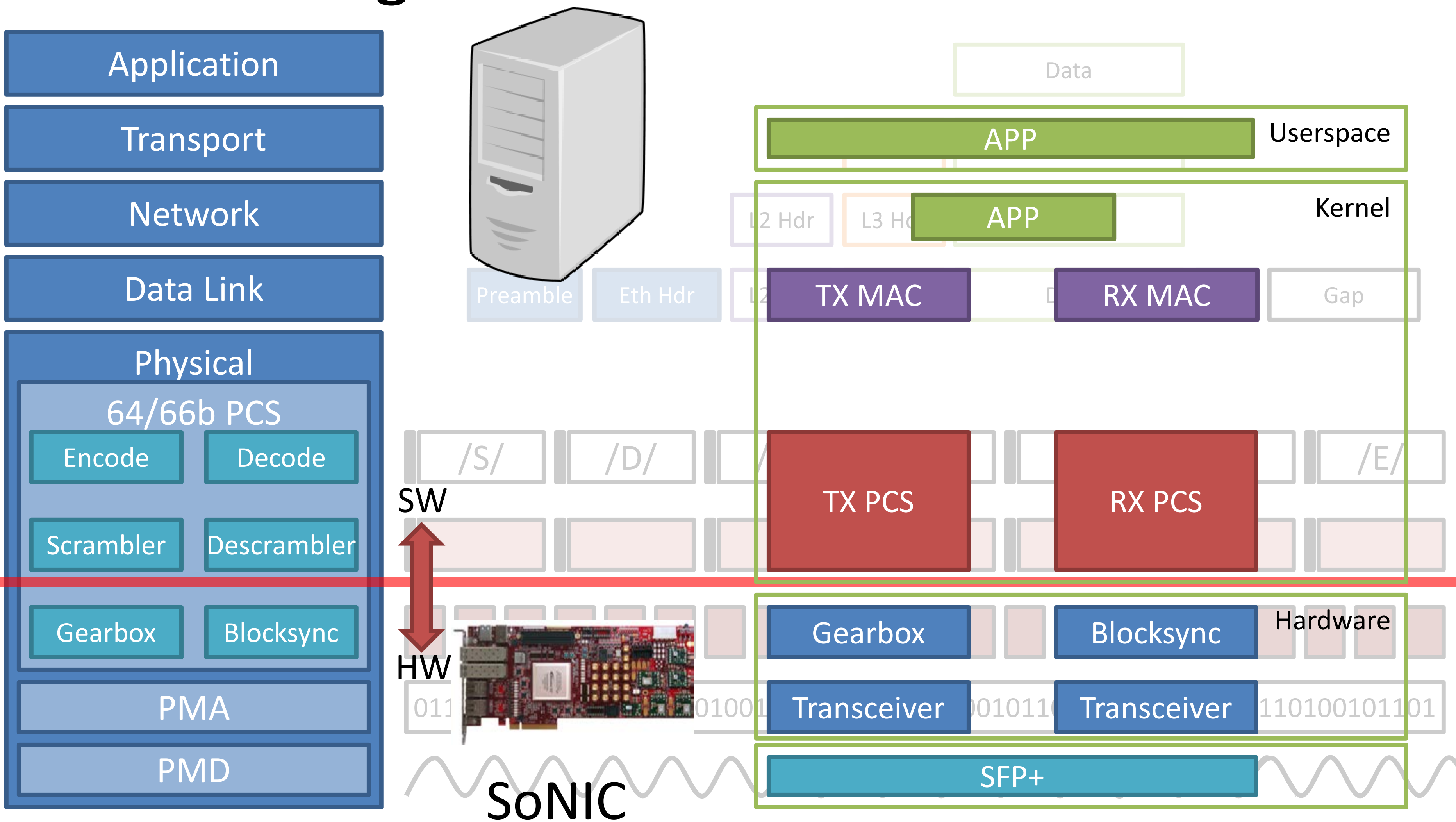
10GbE Network Stack



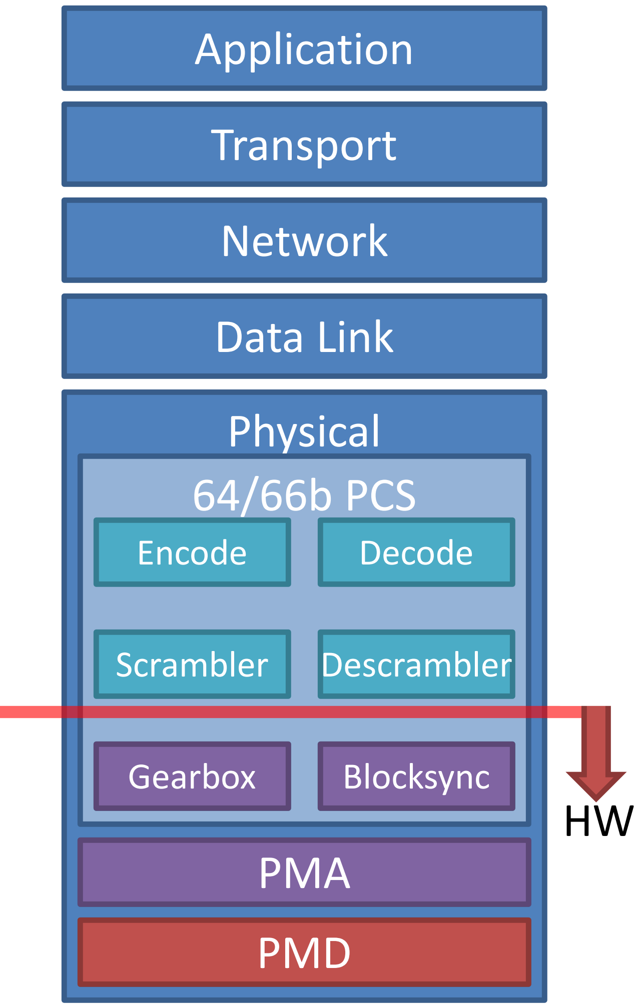
SoNIC Design



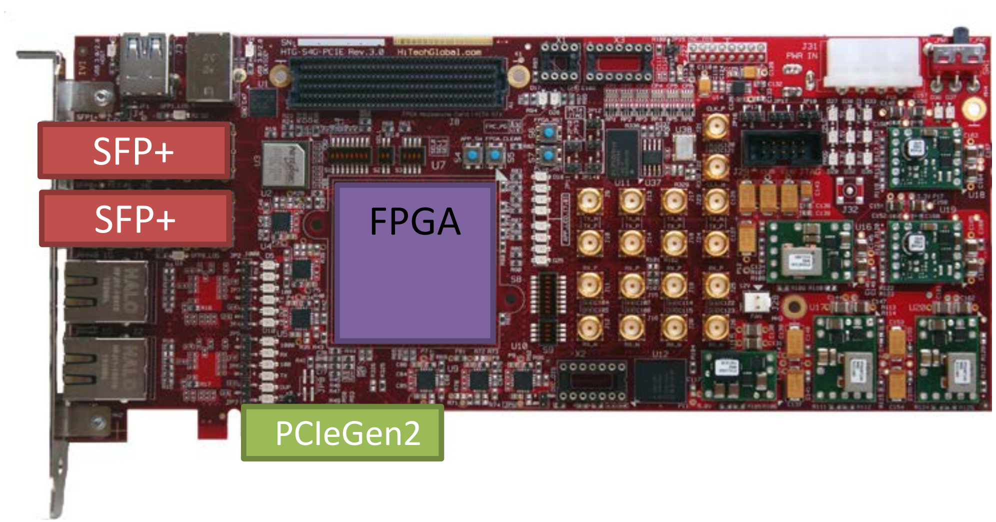
SoNIC Design and Architecture



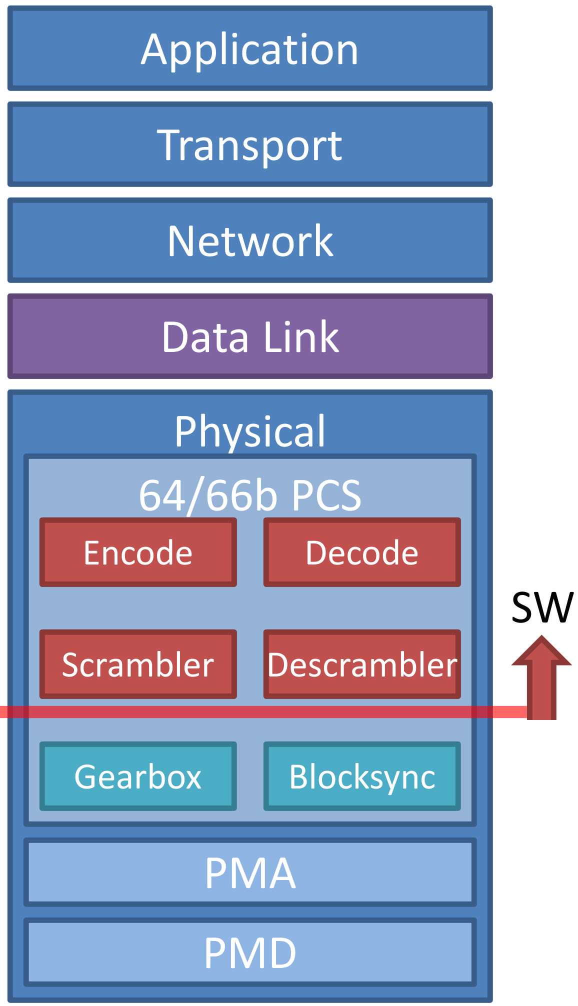
SoNIC Design: Hardware

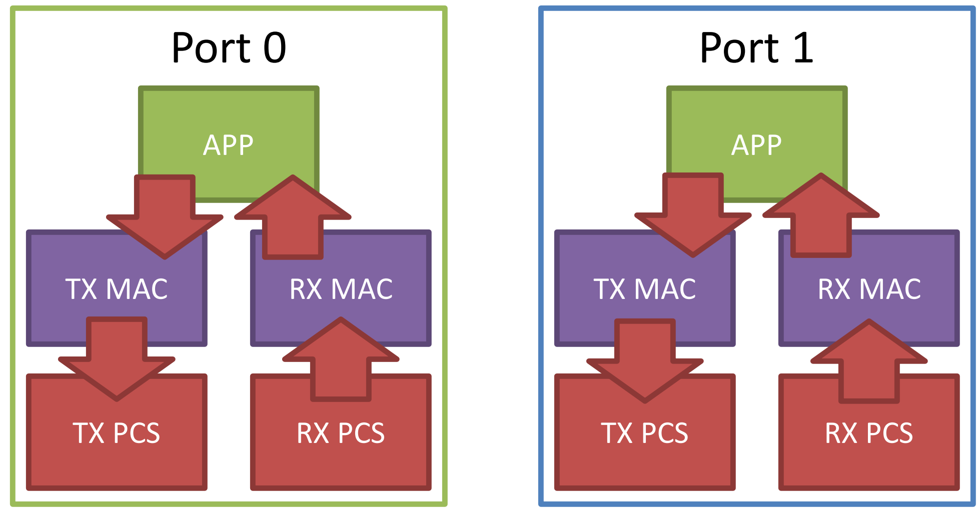


- To deliver every bit from/to software
  - High-speed transceivers
  - PCIe Gen2 (=32Gbps)
- Optimized DMA engine

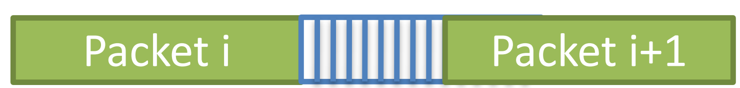


SoNIC Design: Software

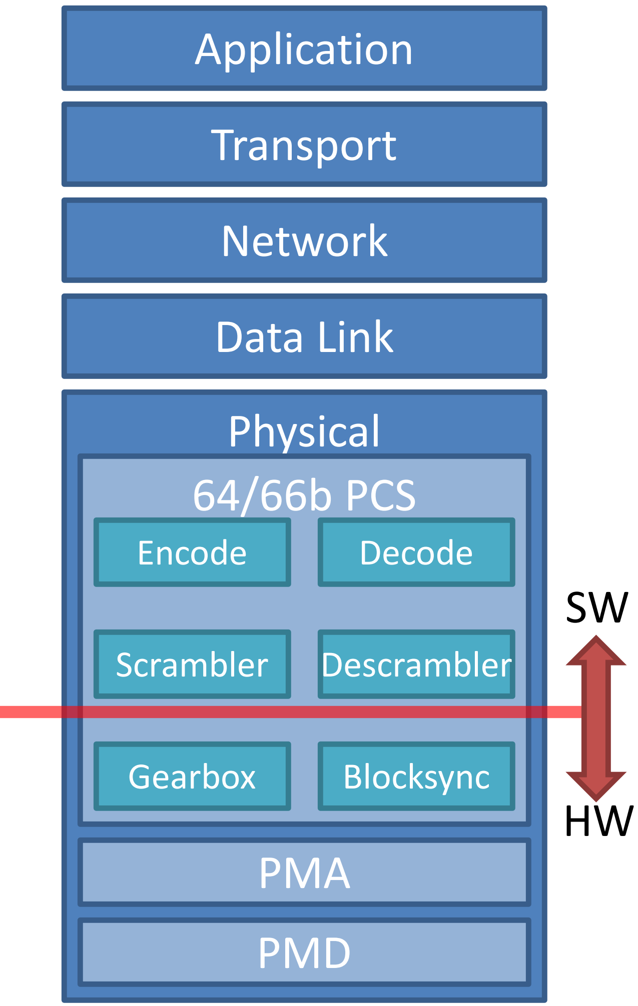


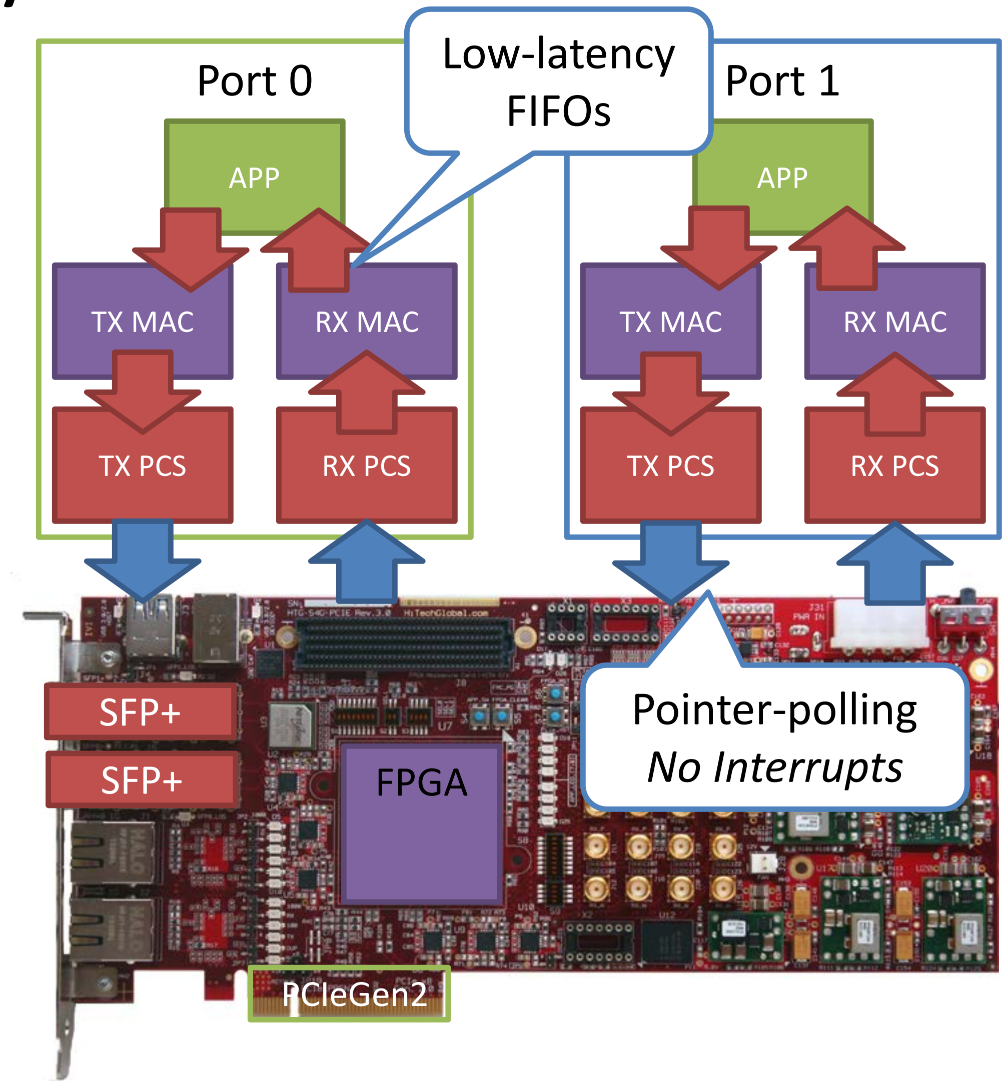


- Dedicated Kernel Threads
  - TX / RX PCS, TX / RX MAC threads
  - APP thread: Interface to userspace

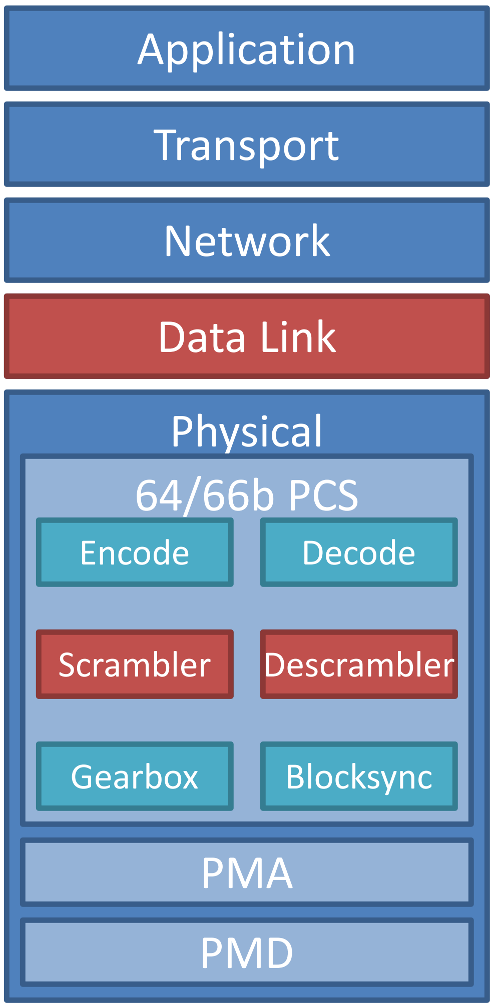


SoNIC Design: Synchronization





SoNIC Design: Optimizations



- Scrambler $G(x) = x^{58} + x^{39} + 1$

Naïve Implementation

```
s ← state
d ← data
for i = 0 → 63 do
    in ← (d >> i) & 1
    out ← (in ⊕ (s >> 38) ⊕ (s >> 57)) & 1
    s ← (s << 1) | out
    r ← r | (out << i)
    state ← s
end for
```

- CRC computation
- DMA engine

SoNIC Design: Interface and Control

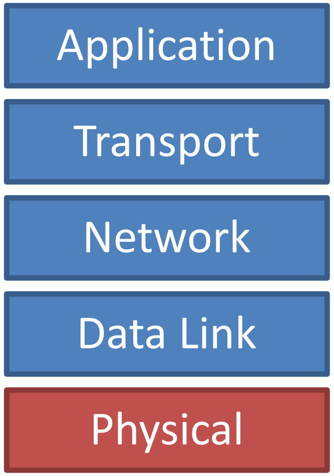
- Hardware control: *ioctl* syscall
- I/O : character device interface
- Sample C code for packet generation and capture

```
#include "sonic.h"

struct sonic_pkt_gen_info info = {
.mode = 0,
.pkt_num = 1000000000UL,
.pkt_len = 1518,
.mac_src = "00:11:22:33:44:55",
.mac_dst = "aa:bb:cc:dd:ee:ff",
.ip_src = "192.168.0.1",
.ip_dst = "192.168.0.2",
.port_src = 5000,
.port_dst = 5000,
.idle = 12,
};

/* OPEN DEVICE*/
fd1 = open(SONIC_CONTROL_PATH, O_RDWR);
fd2 = open(SONIC_PORT1_PATH, O_RDONLY);
/* CONFIG SONIC CARD FOR PACKET GEN*/
ioctl(fd1, SONIC_IOC_RESET)
ioctl(fd1, SONIC_IOC_SET_MODE, PKT_GEN_CAP)
ioctl(fd1, SONIC_IOC_PORT0_INFO_SET, &info)

/* START EXPERIMENT*/
ioctl(fd1, SONIC_IOC_START)
// wait till experiment finishes
ioctl(fd1, SONIC_IOC_STOP)

/* CAPTURE PACKET */
while ((ret = read(fd2, buf, 65536)) > 0) {
// process data
}

close(fd1);
close(fd2);
```

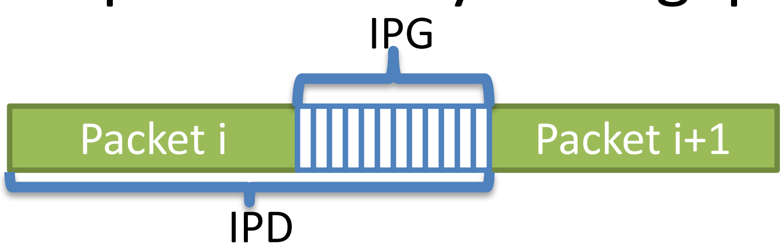
Outline

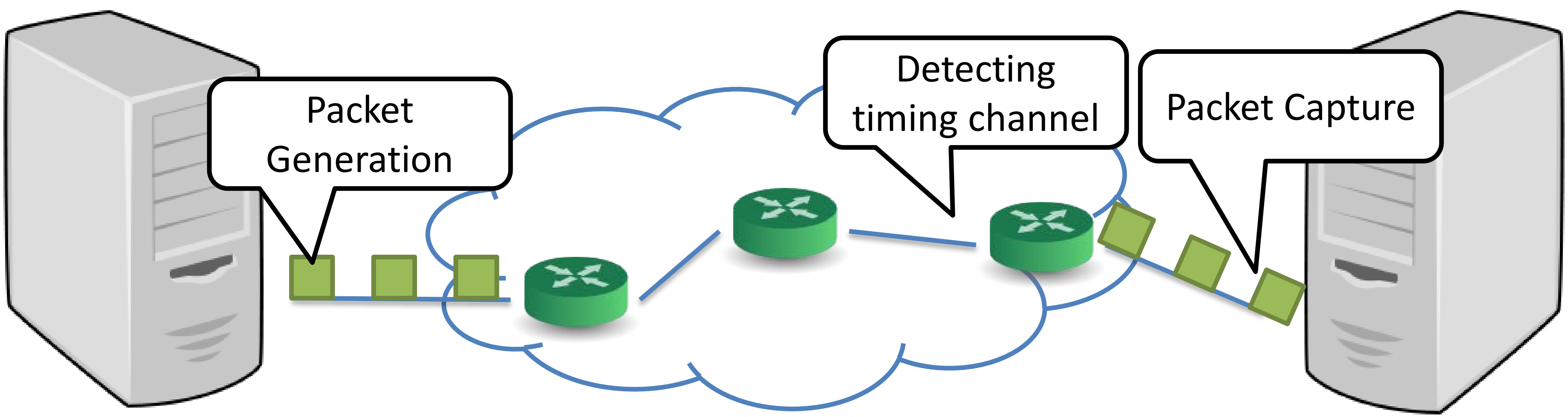
- Introduction
- SoNIC: Software-defined Network Interface Card
- Network Research Applications
  - Packet Generation
  - Packet Capture
  - Covert timing channel
- Conclusion

Network Research Applications



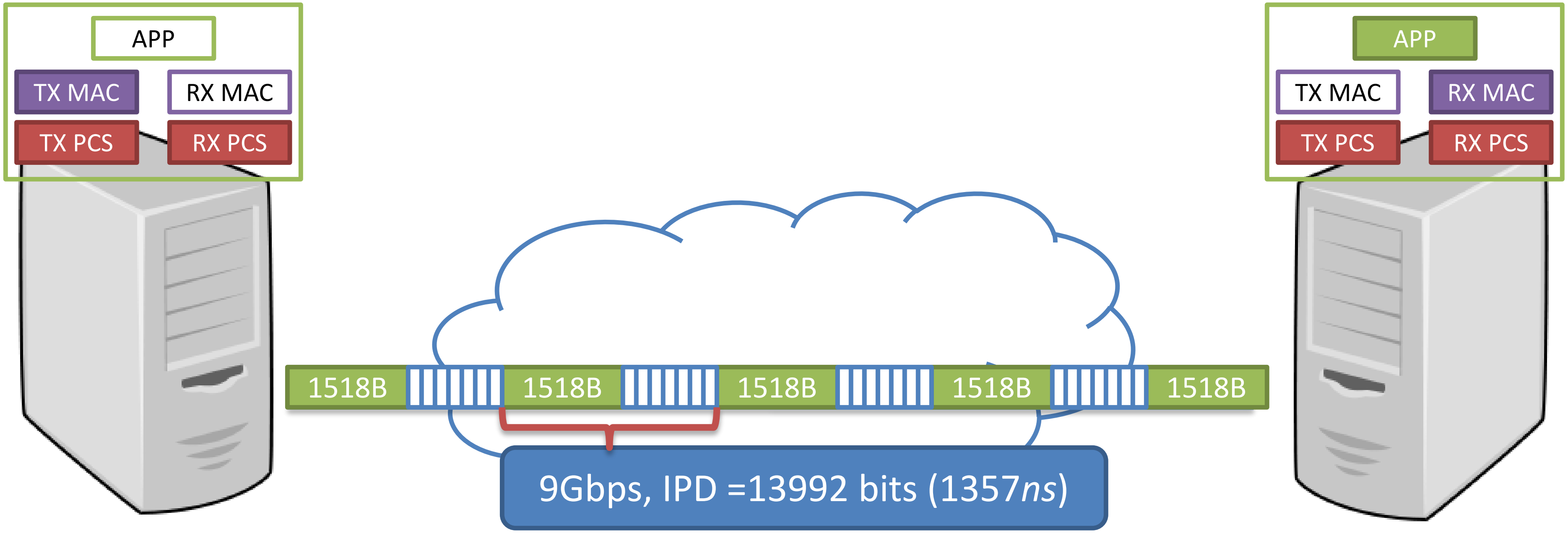
- Interpacket delays and gaps





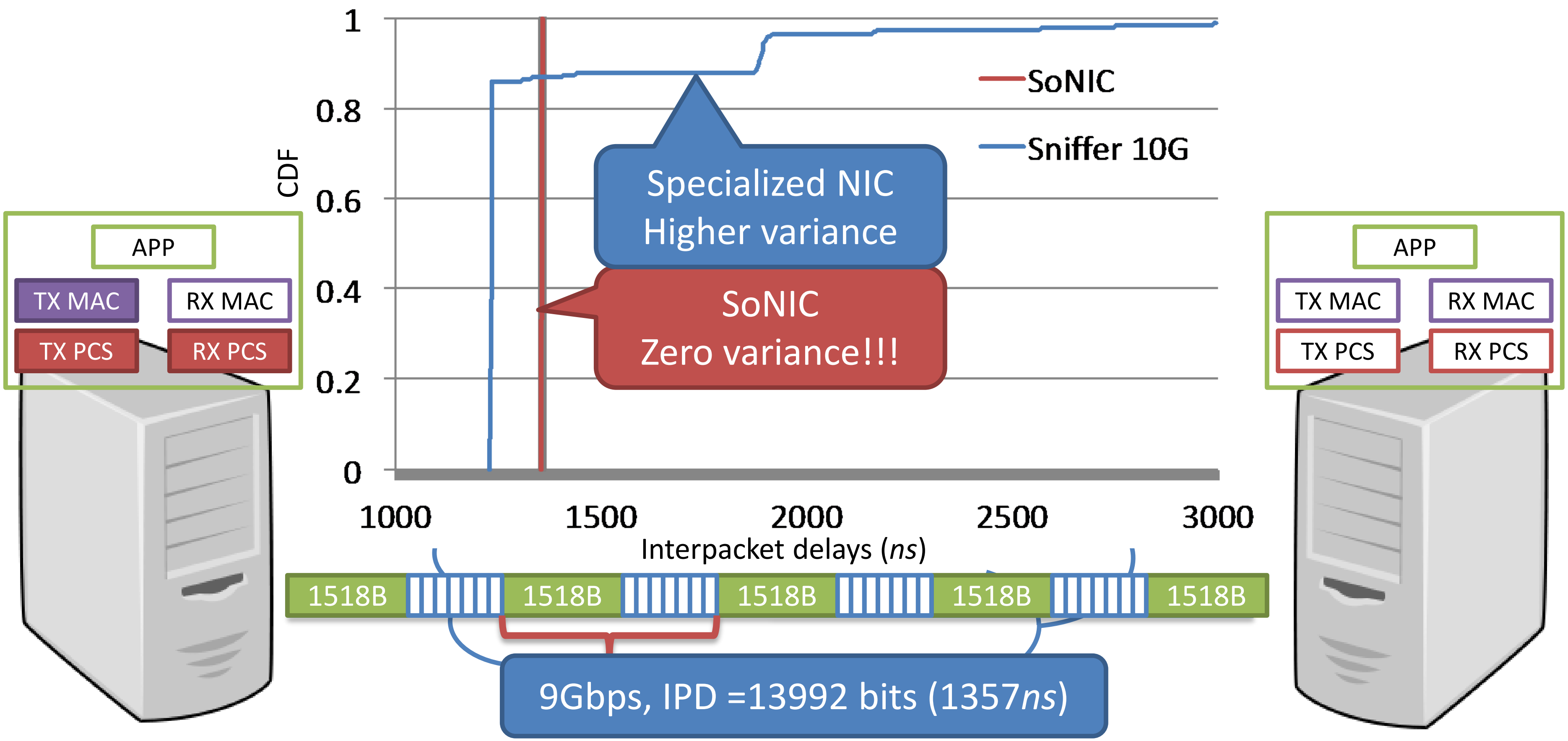
Packet Generation and Capture

- Basic functions for network research
  - Generation: SoNIC allows control of IPGs in # of /I/s
  - Capture: SoNIC captures what was sent with IPGs in bits



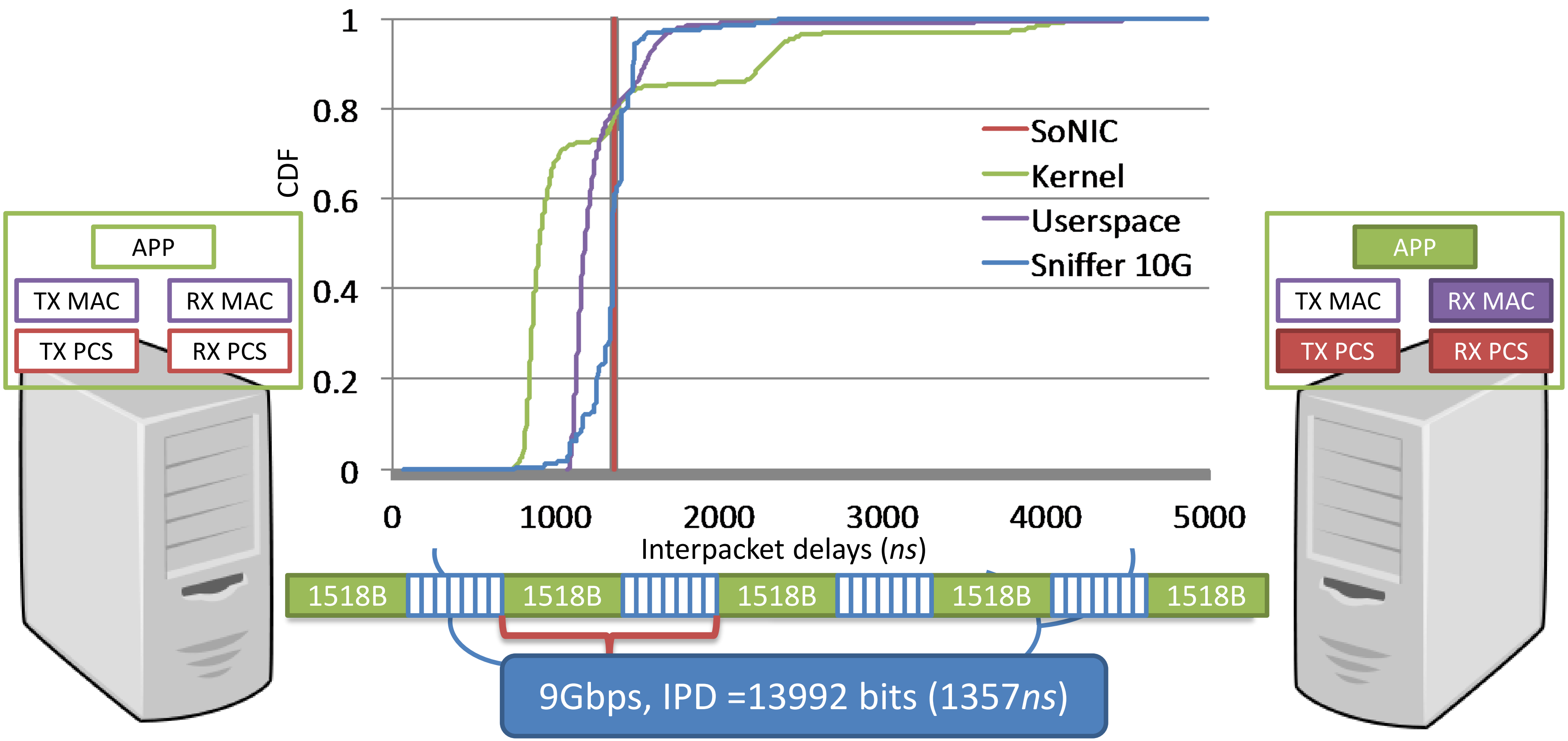
Packet Generation

- *SoNIC allows precise control of IPGs*



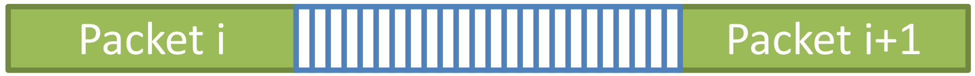
Packet Capture

- *SoNIC captures what is sent*

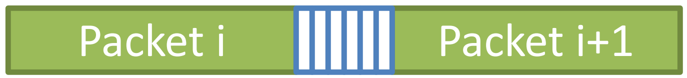


Covert Timing Channel

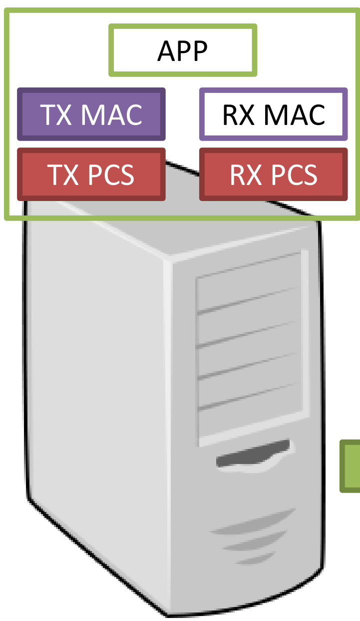
- Embedding signals into interpacket gaps.
  - Large gap: ‘1’



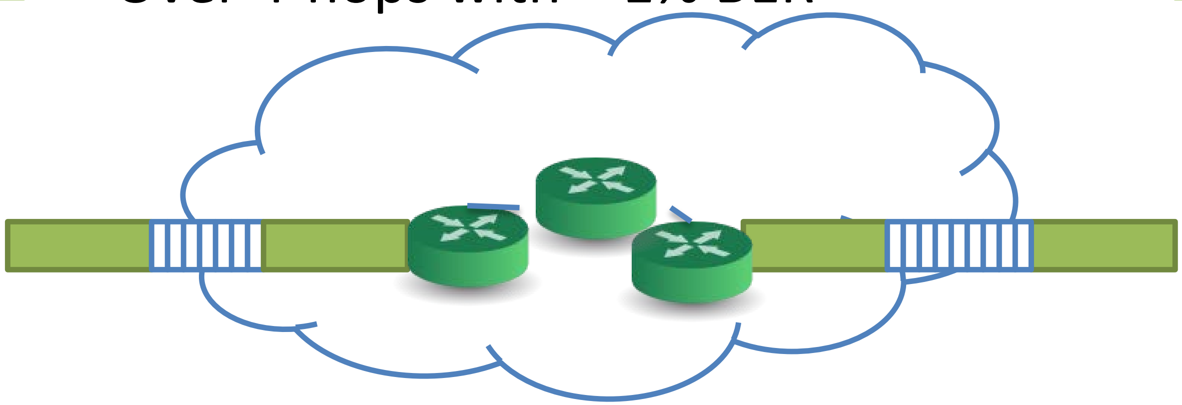
  - Small gap: ‘0’

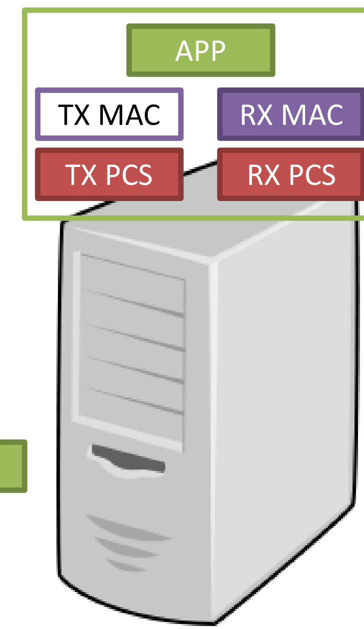


- Covert timing channel by modulating IPGs at 100ns



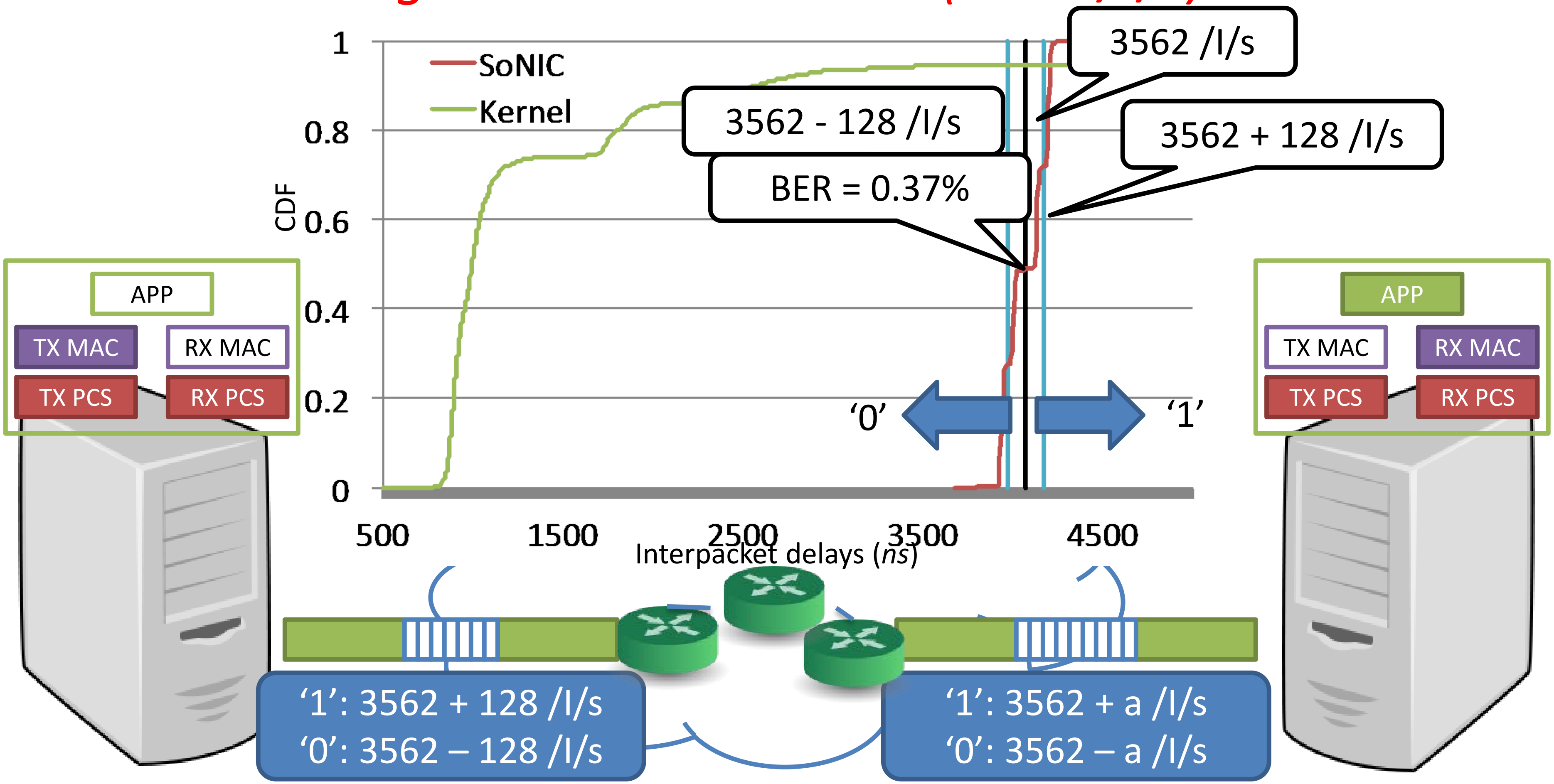
- Overt channel at 3 Gbps
- Covert channel at 250 kbps
- Over 4-hops with < 1% BER





Covert Timing Channel

- *Modulating IPGS at 100ns scale (=128 /I/s)*



Contributions

- Network Research
  - Unprecedented access to the PHY with commodity hardware
  - A platform for cross-network-layer research
  - Can improve network research applications
- Engineering
  - Precise control of interpacket gaps (delays)
  - Design and implementation of the PHY in software
  - Novel scalable hardware design
  - Optimizations / Parallelism
- Status
  - Measurements in large scale: DCN, GENI, 40 GbE

Conclusion

- Precise Realtime Software Access to the PHY
- Commodity components
  - An FPGA development board, Intel architecture
- Network applications
  - Network measurements
  - Network characterization
  - Network steganography
- Webpage: http://sonic.cs.cornell.edu
  - SoNIC is available Open Source.

Before Next time

- Project Interim report
  - **Due Monday, November 24.**
  - And meet with groups, TA, and professor
- Fractus Upgrade: Should be back online

- ***Required review and reading for Friday, November 14***
  - Timing is Everything: Accurate, Minimum Overhead, Available Bandwidth Estimation in High-speed Wired Networks, H. Wang, K. Lee, E. Li, C. L. Lim, A. Tang, and H. Weatherspoon. ACM SIGCOMM Internet Measurement Conference (IMC), November 2014.
  - http://conferences2.sigcomm.org/imc/2014/papers/p407.pdf

- Check piazza: http://piazza.com/cornell/fall2014/cs5413
- Check website for updated schedule