

STATUS OF THE SRS INTEGRATION INTO THE TOTEM DAQ SYSTEM

ON BEHALF OF TOTEM DATA ACQUISITION GROUP

F.S.CAFAGNA¹, A.FIERGOLSKI^{1,3}, M.QUINTO^{1,2} E.RADICIONI¹
(INFN-BARI¹, UNIVERSITY OF BARI², WARSAW UNIVERSITY OF TECHNOLOGY³)

RD51 Mini week – CERN, June, 16-19 2014



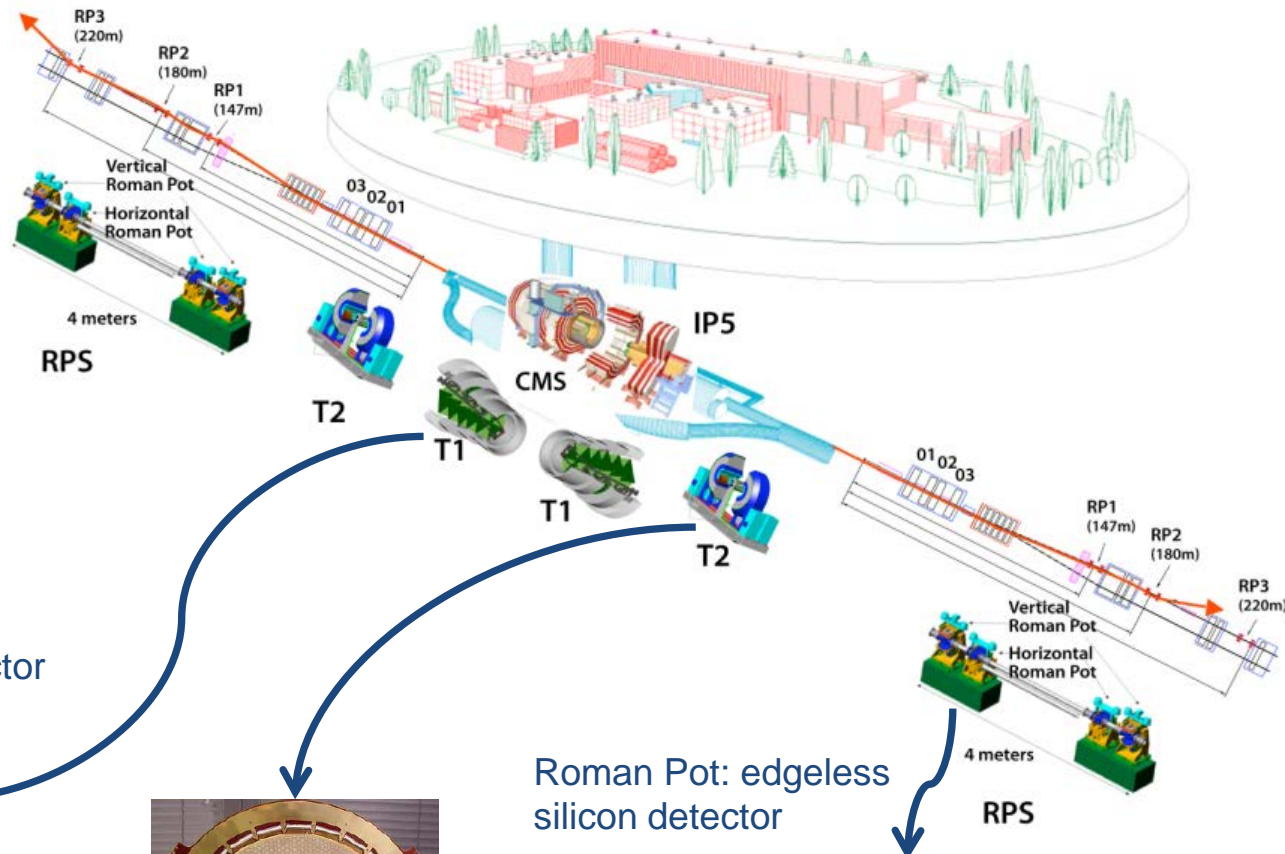
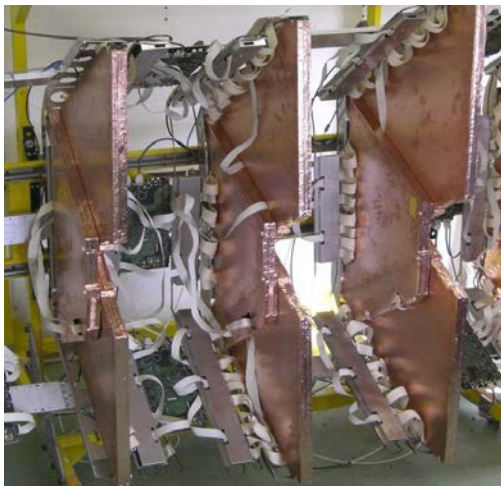
Outline

- Consolidation of the TOTEM DAQ System with SRS;
- Status of firmware development;
- SRS readout stress test results;
- Hardware procurement;
- Firmware design & development strategies;
- Outlook

TOTEM Experiment at LHC

- TOTEM measures total p-p cross section at the LHC energies and studies diffractive processes.
- TOTEM adopts three detectors symmetrically placed at the Interaction Point 5 (T1, T2, Roman Pot).
- All TOTEM detector adopts common readout and trigger electronics. The VFAT chip provides readout and trigger capabilities.

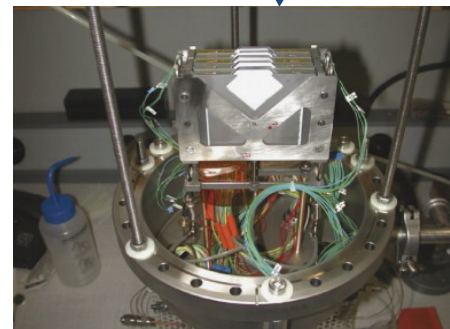
T1: Cathode Strip Chamber detector



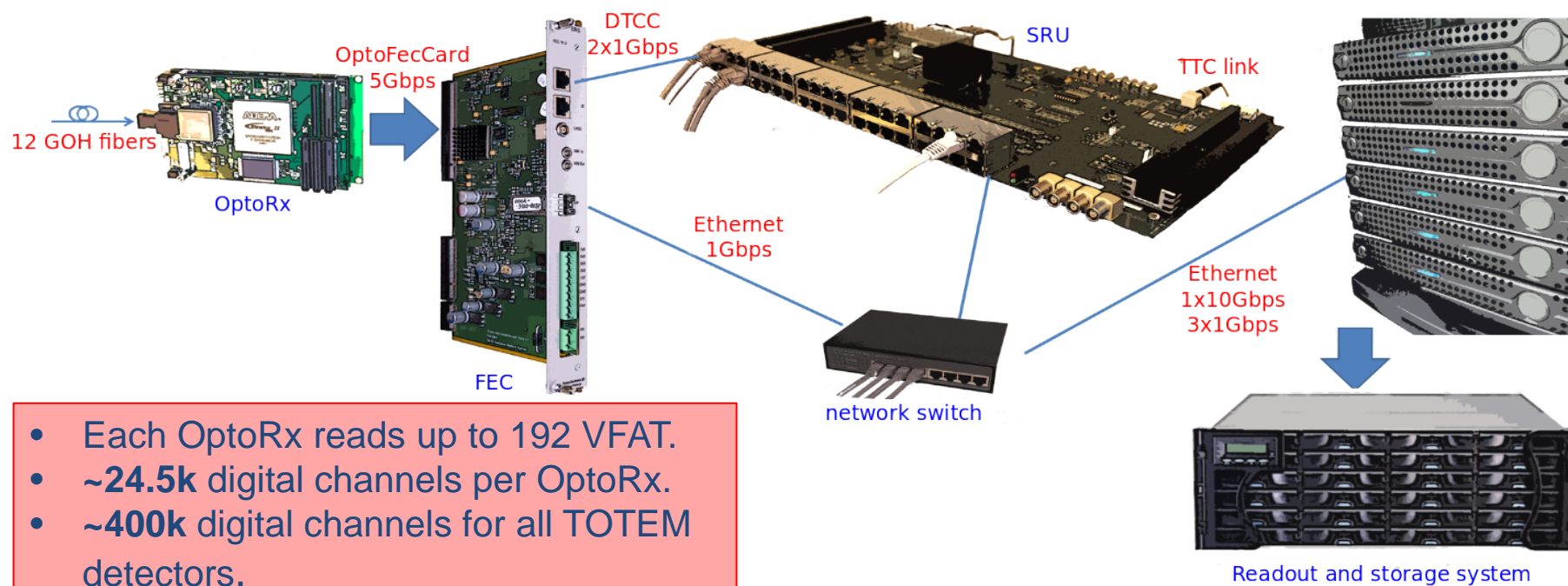
T2: triple GEM detector

Michele Quinto

Roman Pot: edgeless silicon detector



Consolidation of TOTEM DAQ with SRS System



- SRS provides a **cost effective** alternative to the present VME based solution.
- The Ethernet based readout offers many off-the-shelf hardware solutions (NICs, switches) that naturally improve their performances driven by the market.
- SRS offers many possibilities of implementing **data reduction and filtration** by leveraging the SRS resources at different stages in the DAQ chain.
- SRS flexibility allow system extension for future TOTEM upgrades.

SRS Firmware development and test for TOTEM

- Implement a robust Ethernet data transmission using UDP:
 - Complete review of the UDP stack;
 - Development of readout software for both DATE and stand-alone application using C++ and Boost libs.
- Extensive stress test in different configurations:
 - 1 FEC to 1PC;
 - 4 FEC to 1PC via switch;
 - 2x2 FEC to 2 PC via switch.
- Implement the LHC TTC full set of command and timing signals through DTCC Links SRU -> FECs -> OptoRx;
- Full TTCrx test of timing and fast commands propagation to the font-end.

Implementation of a new UDP Stack block

- UDP Stack is freely distributed by opencore.org in which we are collaborating for testing and developing.
- AXI4-Stream standard bus;
- Support for 255 entries ARP table (essential for multi-host connection);
- IP TX and UDP RX, TX interfaces;
- **Lower latency.** One FIFO memory level in the TEMAC, **no memory to memory copy**;
- **Lower memory and logic resource usage**;
- Good performance observed during readout test

	Slices	Slice Reg	LUTs	LUTRAM	BRAM/FIFO
UDP Stack and TEMAC	843	1375	1343	20	9

50% less LUTs

- Trigger rate at flat top ~25kHz
- Readout bandwidth close to the link limit 118MB/s
- System stability over more than 140M events

LDC status display	
LDC name	aloneldc
host	ttf43
Current Trigger rate	24744.000
Average Trigger rate	24494.273
Number of sub-events	146622726
Sub-event rate	24744
Sub-events recorded	146622721
Sub-event recorded rate	24747
Bytes injected	703789084944
Byte injected rate	118.774 MB/s
Bytes recorded	703789051344
Byte recorded rate	118.788 MB/s
Nb. evts w/o HLT decision	0
Nb. of Readout FIFO full	0
mem allocation failed	0
average time bmAllocate	

UDP based readout pitfalls

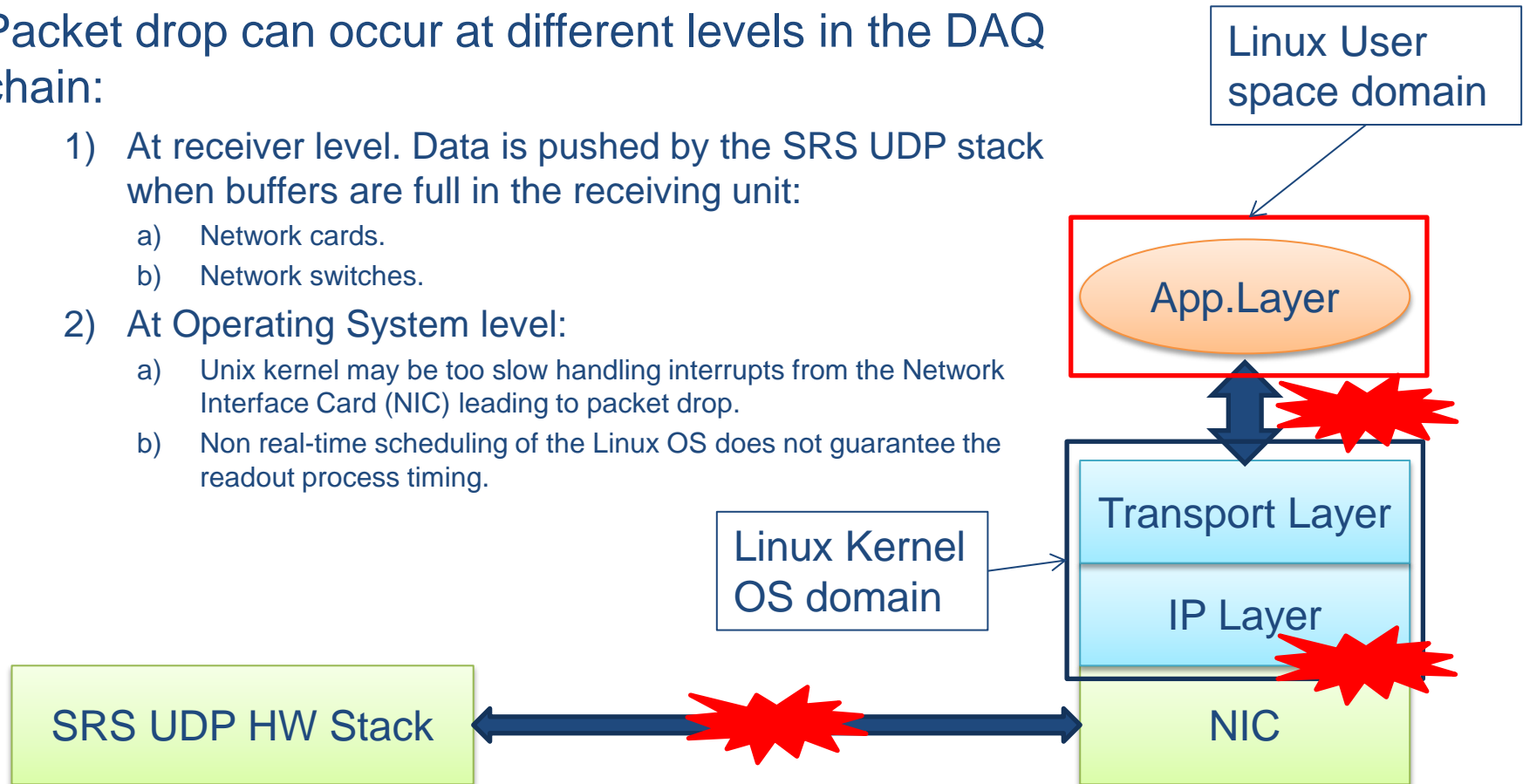
- Ethernet protocols such as TCP/UDP are lossy by design. Packet loss can occur:
 - TCP-IP solves the issue of packet loss implementing retransmission -> very costly to implement on FPGA a TCP/IP offload engine.
 - Dropped packet are simply lost using UDP.
- UDP based readout CONS:
 - ✓ Packet loss rates are not acceptable for long-running experiments with event building such as TOTEM. Event synchronization must be guaranteed.
- UDP based readout PROS:
 - ✓ UDP stack is as lightweight as simple to implement on FPGA devices.

Is it possible to have a reliable UDP readout?

UDP based readout pitfalls

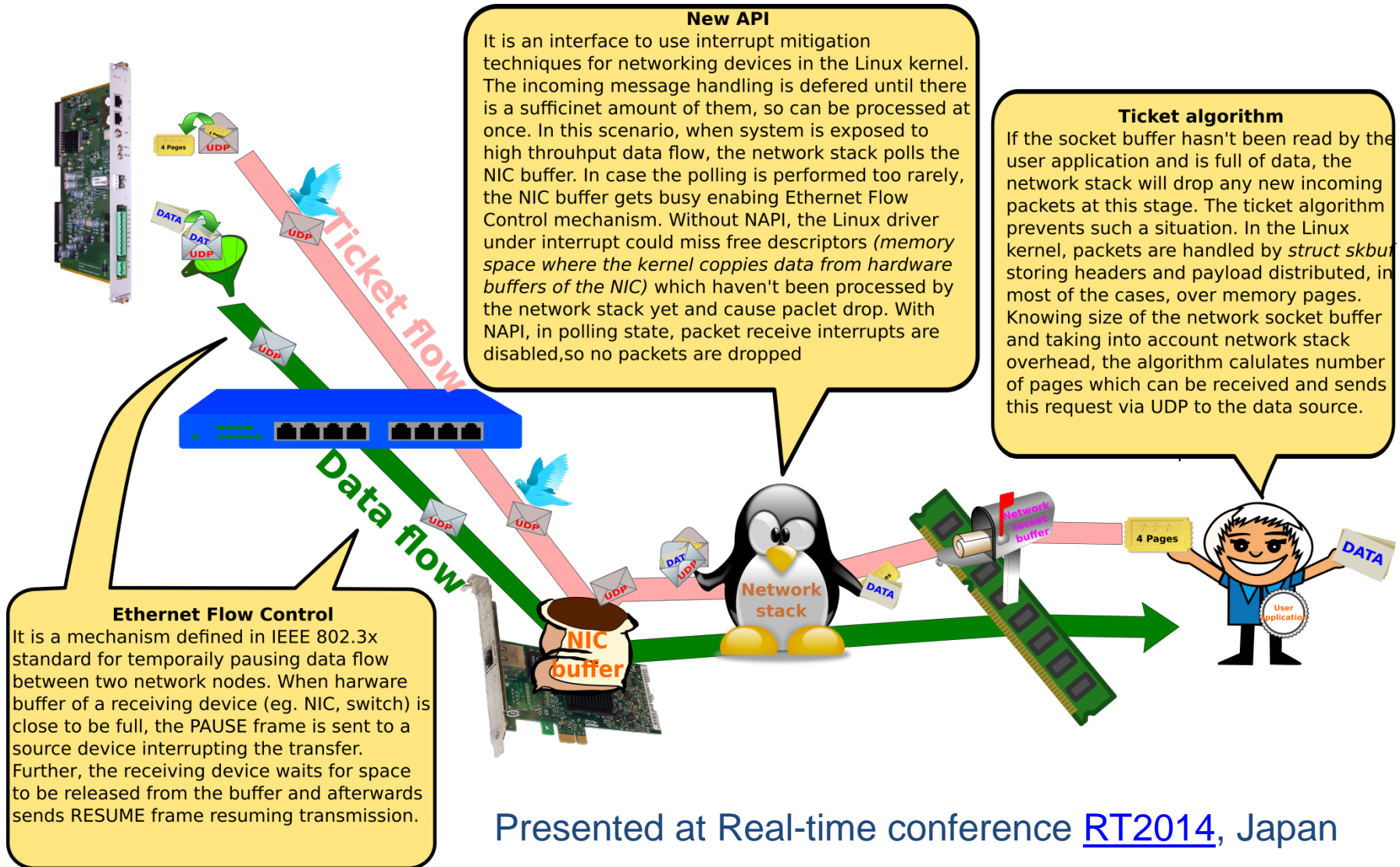
Packet drop can occur at different levels in the DAQ chain:

- 1) At receiver level. Data is pushed by the SRS UDP stack when buffers are full in the receiving unit:
 - a) Network cards.
 - b) Network switches.
- 2) At Operating System level:
 - a) Unix kernel may be too slow handling interrupts from the Network Interface Card (NIC) leading to packet drop.
 - b) Non real-time scheduling of the Linux OS does not guarantee the readout process timing.



Is it possible to have a reliable UDP readout?

Answer is: YES! ... And here is the recipe:



UDP Readout stress test results

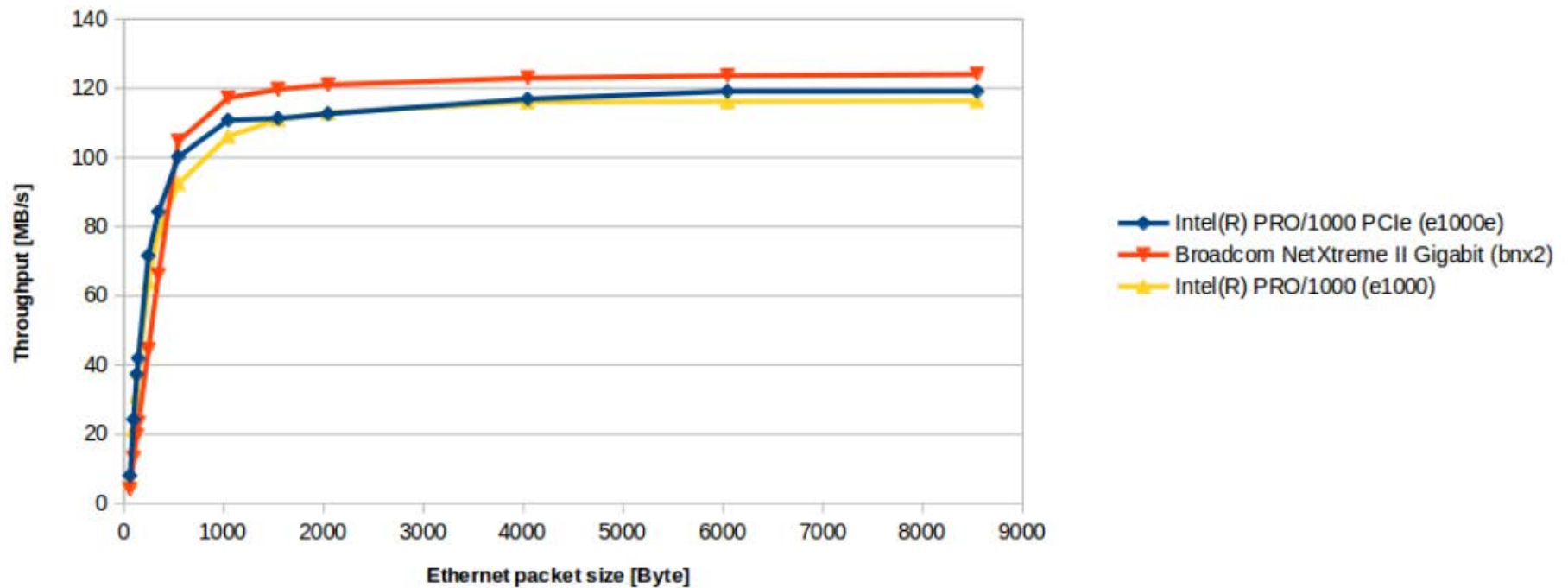
First test scenario:

- 1 FEC streaming data to 1 PC.
- Processor Intel i3.
- Linux Kernel 2.6 Intel.
- Several NICs tested: PCI NIC e1000, PCIe NIC e1000e, Broadcom NetXtreme.
- Writing to a standard HD drive ~65MByte/s.
- Generating others high demand IO processes active in parallel on the same CPU and filesystem.

Packet Error Rate (PER) < $9 \cdot 10^{-11}$ BERR < $2.5 \cdot 10^{-15}$

UDP Readout stress test results

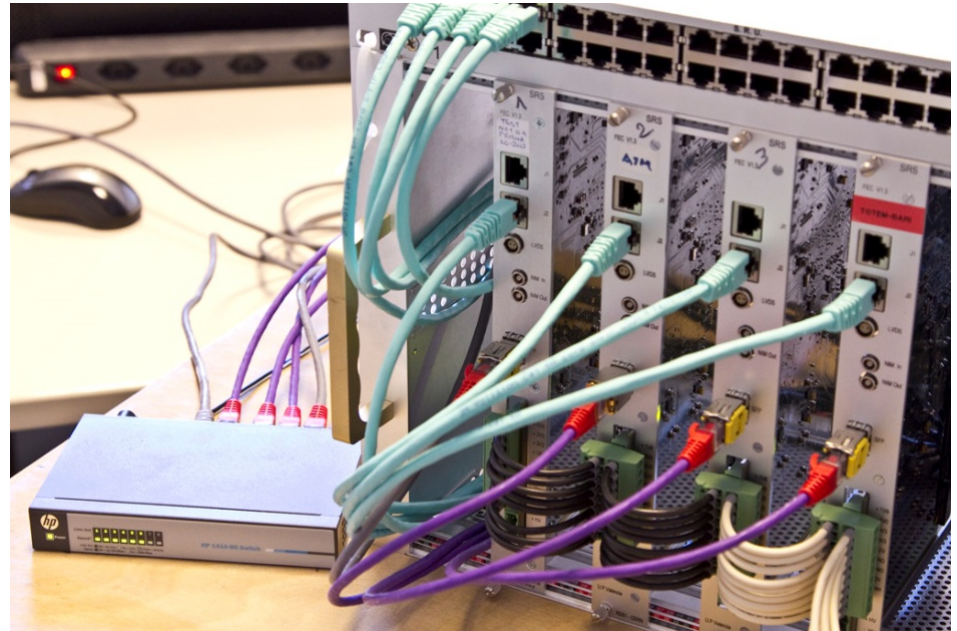
Lossless UDP Transmission performance



UDP Readout stress test results

Second test scenario:

- 4 FEC streaming data @1Gbit/s to 1 PC receiving @1Gbit/s
- Processor Intel Xenon
- Linux Kernel 2.6 Intel
- Broadcom NetXtreme
- **Commercial switch** form HP and data server managed 3Com 4200G switch

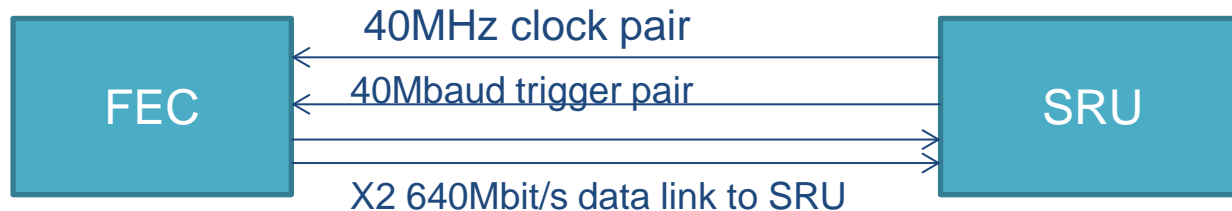


Packet Error Rate (PER) < 9.9*10⁻¹⁰

- Single packet is drop observe once in a while.
- Performance worst than point-to-point connection due to the switch, but it is still acceptable.
- Simply relying on the Ethernet Flow Control + the ticketing mechanism the system rate adjust to 1GBit and cope with system fluctuations.
- **No trigger rate tuning needed!!!**

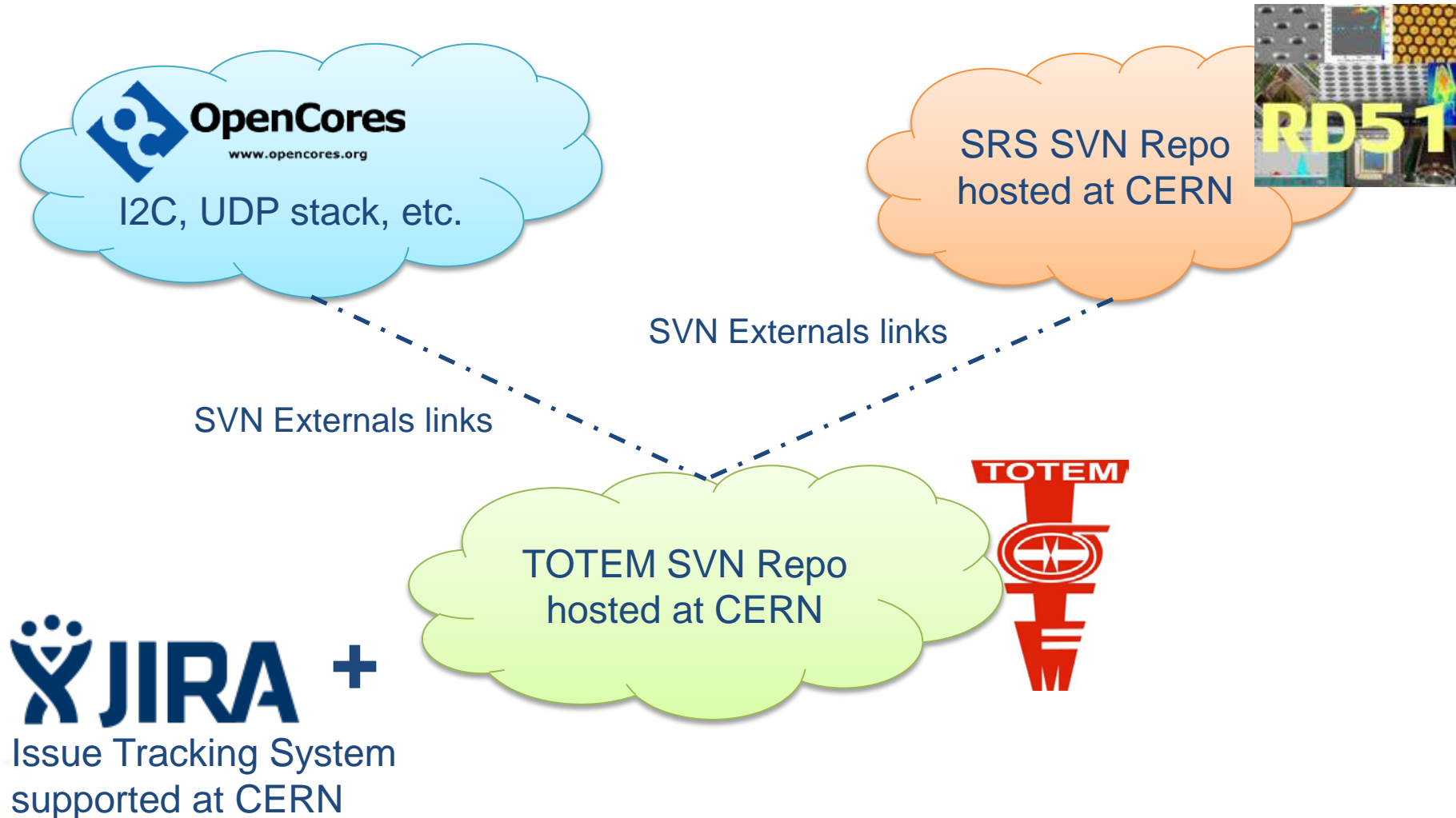
DTCC Link Implementation for TOTEM

- The DTCC Link trigger frame for TOTEM is 8 bit long + 8 bit control word.
- The Link Baud rate is 40Mbaud, compliant with TTCci system requirements.

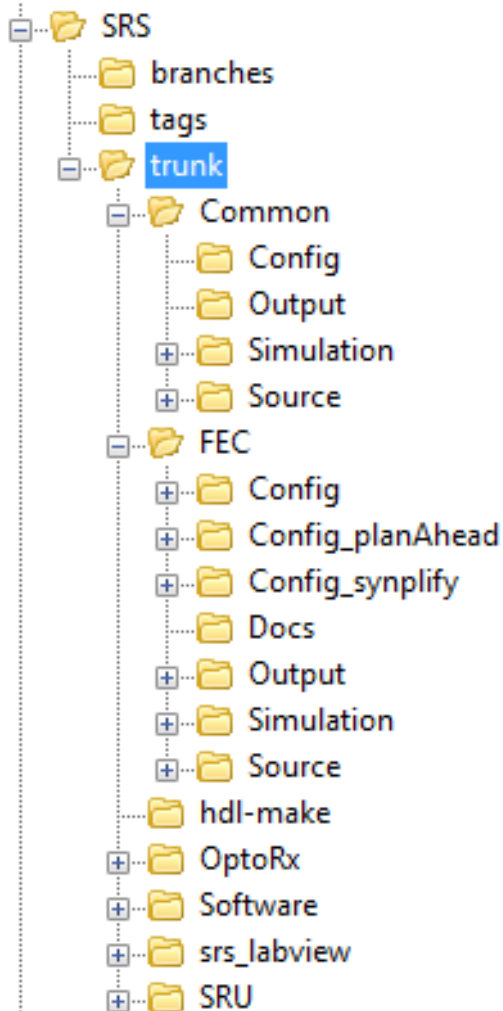


- TTCrx Commands are broadcasted to all DDTCC Links:
 - Level 1 Accept;
 - BC0, EC0, OC0;
 - Resync;
 - Back-pressure
- This DTCC implementation is TOTEM specific; however it uses the DTCC firmware core blocks developed and offered to us by A.Tarazona.
- It can be integrated in the main release.
- Unfortunately the main SRS SVN Repo has, so far, not up-to-date DTCCL code.

Firmware design and development strategies



TOTEM SVN Structure and release policy



- All subsystem projects (SRU, FEC, OptoRx) sit in the same repository trunk.
- Software package (Slow Control, readout, drivers, APIs) is common to all subsystems.
- No SW binaries files on Repo.
- FPGA programming binary files go to a dedicated directory: Output.
- Tags include the full project: all subsystems + software -> This implies compatibility and interoperability of software and firmware versions.
- The TOTEM SVN Librarian ask developers to pass through a test procedure before making a tag.
- Commit to tagged version in /tag is not allowed of course!

JIRA and SVN

Search

project = TOTEMDAQ AND resolution = Unresolved ORDER BY component ASC, priority DESC

Activity

Subversion

- TOTEMDAQ-26
UDP algorithm tests
- TOTEMDAQ-35
OptoRx processing
- TOTEMDAQ-17
RT2014
- TOTEMDAQ-36
Prepare poster and paper about UDP algor...
- TOTEMDAQ-27
align SRS Readout software (stand-alone ...
- TOTEMDAQ-29
DTCC review

✓ Helps in tracking issues in development and production phase for both users and developers.

✓ Facilitates sharing information between users, avoids long loops of e-mails!

Outlook and future work

- SRS System has been fully qualified for the TOTEM DAQ Consolidation program:
 - Stable readout under stress conditions for several hours.
 - Successful of field test at IP5.
 - Full back-compatibility with legacy system (TTC System, Software).
- Target data reduction and 2nd level trigger algorithm leveraging the FPGA processing power on SRS.
- Port the design to FEC_V6 beta cards, we have received 2 FEC_V6.
- Proceed with purchasing of full set of hardware ~20FEC.
- Extend DTCCL Links SRU <-> ~20 FECs.

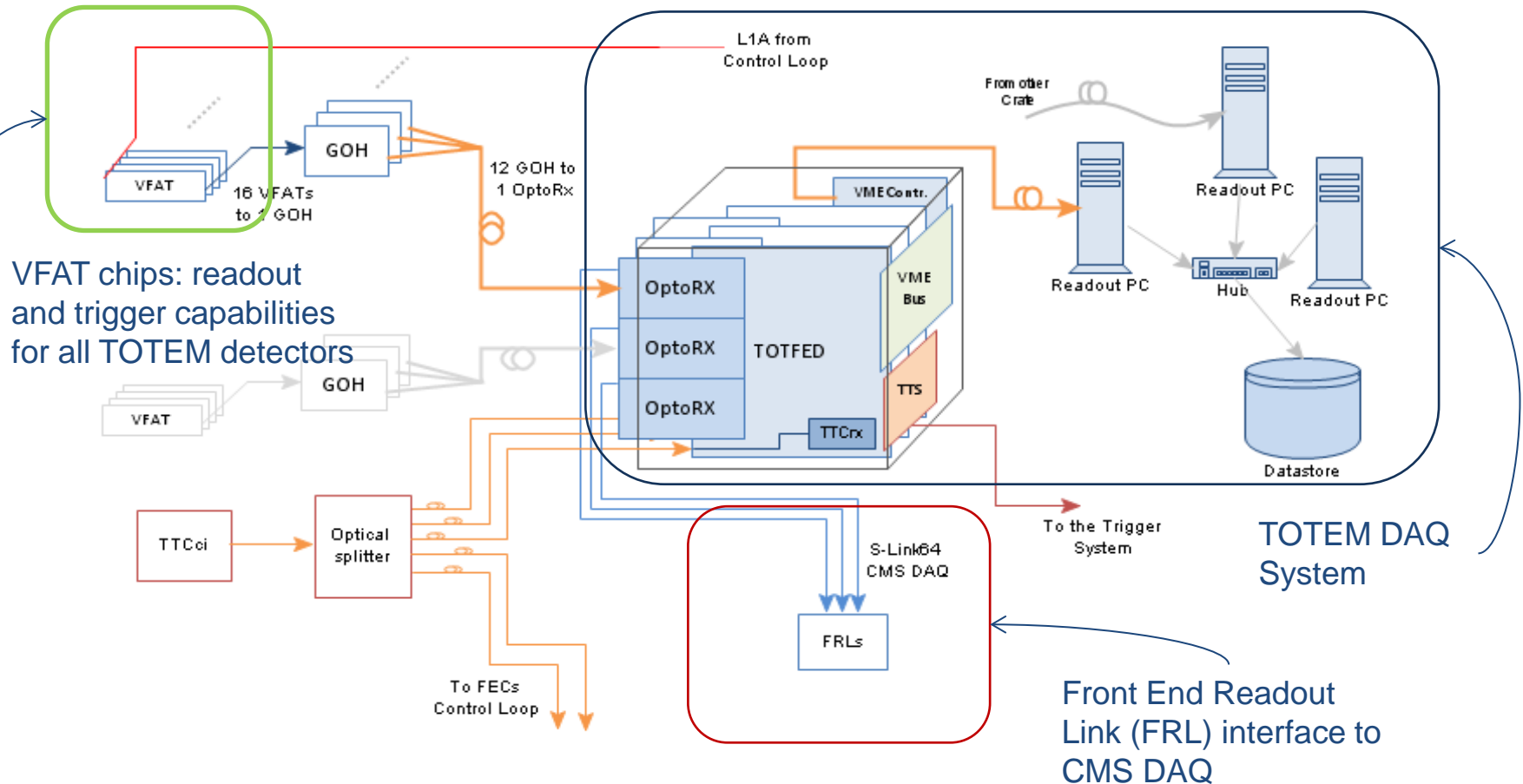


Thank you for your attention!

A thick, horizontal yellow brushstroke with a textured, painterly appearance spans the top of the slide.

Back-up

The Current TOTEM DAQ System Architecture



- In the TOTEM standalone configuration, the VME bus bandwidth limits the trigger rate to 1kHz.