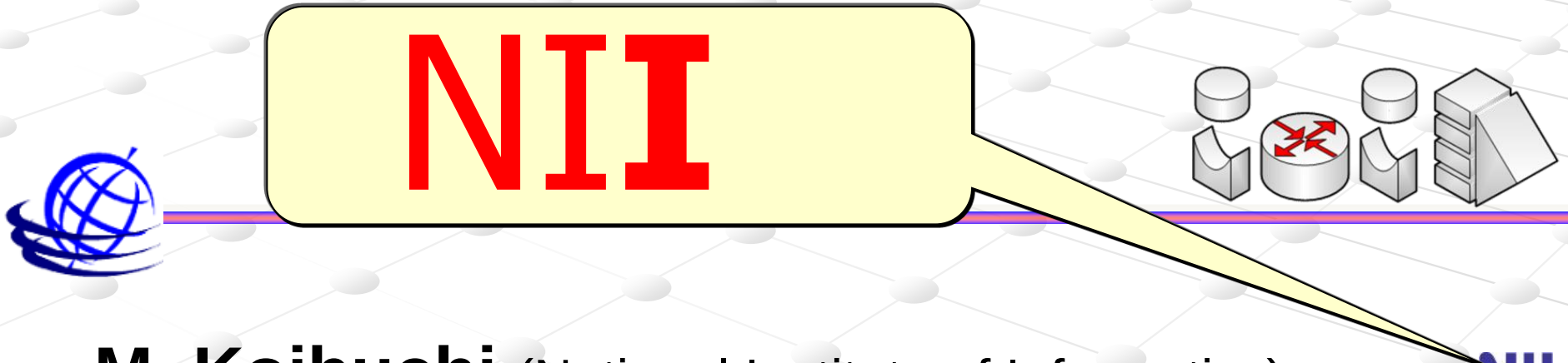
Service-oriented Open Routers



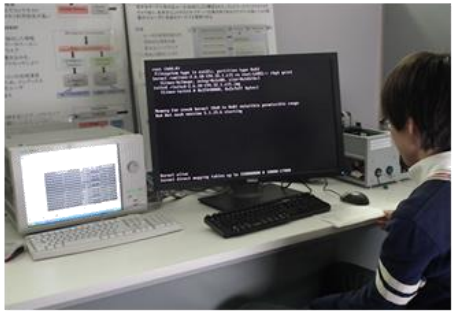
M. Koibuchi (National Institute of Informatics)

H. Nishi (Keio University)

This work is supported by NICT New Generation Network, NW Virtualization 149U02

Agenda

- Purpose:
  - Why do we need a SoR (Service-oriented router) ?
    - SoR = Router + Server
- Achievement:
  - Service-oriented switch/router software/hardware emulator
  - Hardware architecture of service-oriented router
    - REGEXP co-processor
    - DBINS co-processor
    - PPDP co-processor
- Future works:
  - Where do we have to go next?

Cats and dogs of network

Service・Application layer

- Services including search engine are provided by end-hosts, and user can access these services from end-hosts, too.
- Rich services are making continuous progress and big benefit at these end hosts.

Infrastructure layer

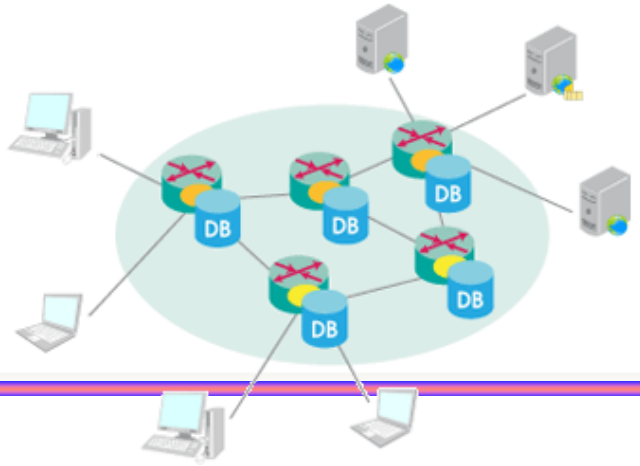
- Routers or switches always devote themselves into packet forwarding.
- They only provide a few limited services;
  protocol translation, firewall, QoS, and load balancing.
- Evaluation of them are only done by using throughput or traffic control.

Some serious problems are caused by this gap

- Service providers earn money by exchanging contents over a flat-rate Internet access services.
  Carriers seek reasonable expense of their burdens. = Network neutrality problem

Gather data at Router

- Router/switch should provide not only simple forwarding performance but also a high-level functions to all users by using its superiority in location.
- Import the degree of freedom in application layer into physical network layer.
- Service-oriented router can provide several services when installed at
  - Core area: New rating services for a search engine can be provided
  - Metro area: Access control or security service can be provided
  - Edge area: Meticulous service (user behavior analysis) can be provided



End-hosts	Switches/Routers
Placed on the edges of the Internet	Placed on branches
Active data collection	Passive data collection

Related works

- Mission Critical Field (real-time management on closed systems)
- CISCO Systems
  - AXP (Application eXtension Platform), ASR (Aggregation Service Router)
    - Provides PC extension module
    - Permits to design application by using special APIs.

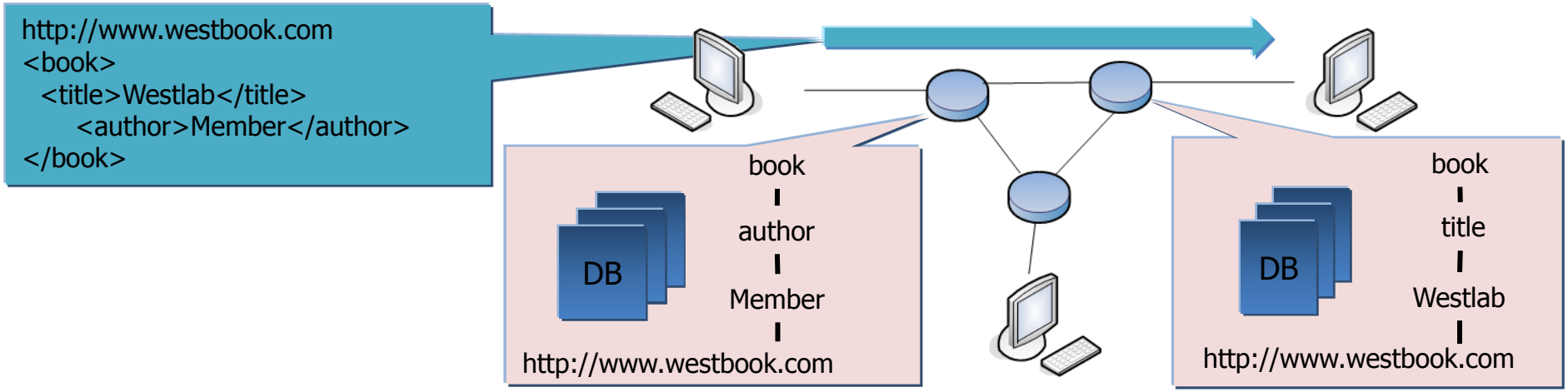
    However, acceptable events is limited; fault detection, etc.

    Now CISCO seeks what kinds of application can they provide.
- OpenFlow
  - Provides flexible routing
  - Still have restrictions in providing application layer services
- XML router proposed by Moscola, Cho, Lockwood
  - Enables Content base routing
  - Inspects and forwards XML messages in several Gbps throughput

  However, it only controls packet routing and is not a truly service-friendly router
- Why don't you use the advantage the routers and switches have?

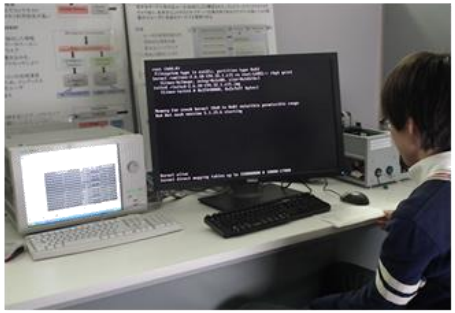
Basic operation of SoR

- Deep inspection of traffic with contents of packets
- SoR inserts REGEX-filtered contents into DB and provide API to access the DB by using SSRQL (Service-oriented Switch and Router Query Language); This open programmability brings open innovation.
- Realtime services independent of realtime communication
  - Freshness of services is the essential for users, and realtime (low latency) communication is a part of user requirement.
  - High-throughput and low-latency network is not a fundamentals to enrich services.
  - SOR enhances realtime services

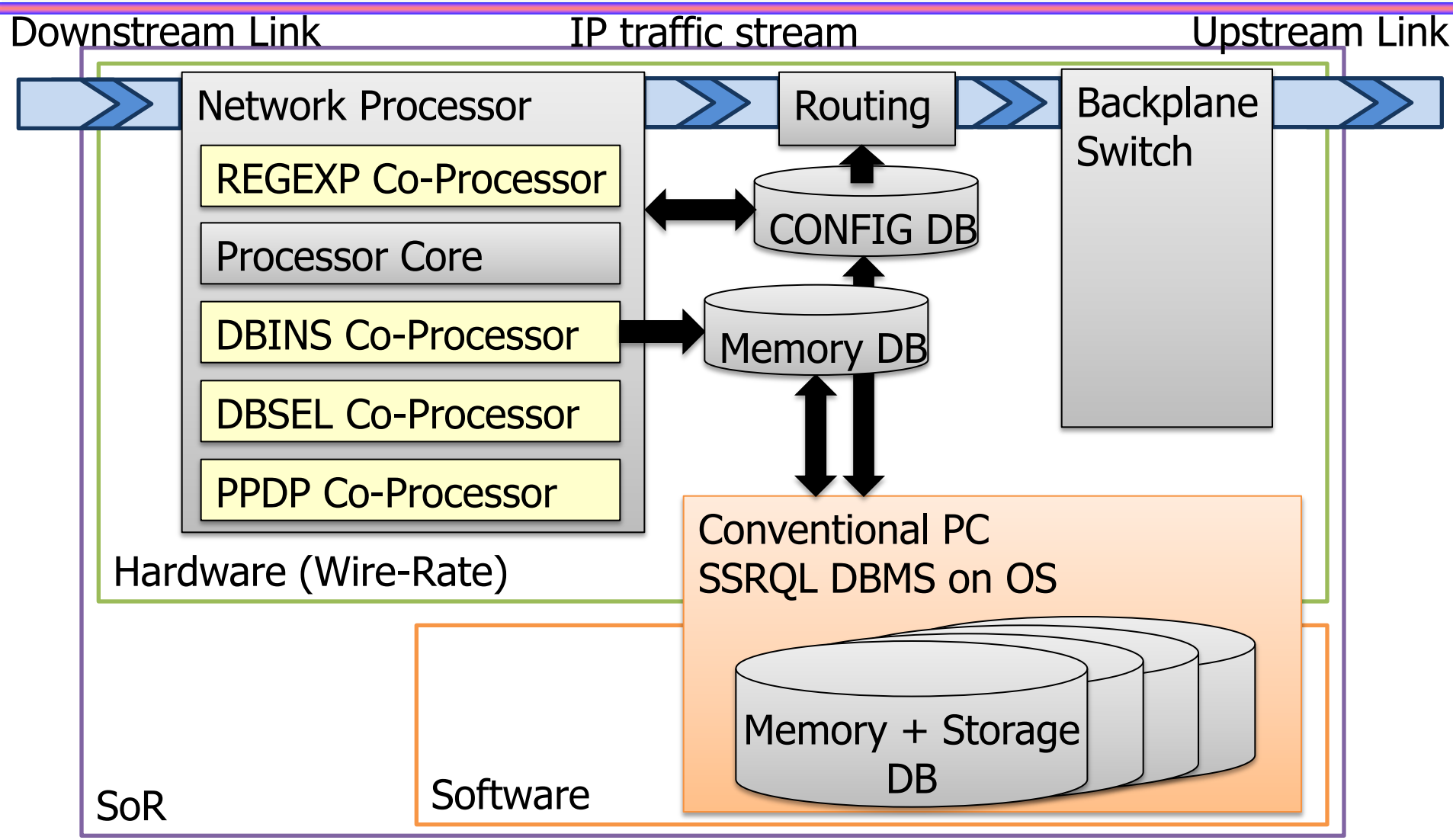


Agenda

- Purpose:
  - Why do we need a SoR (Service-oriented router) ?
- Achievement:
  - Service-oriented switch/router software/hardware emulator
  - Hardware architecture of service-oriented router
    - REGEXP co-processor
    - DBINS co-processor
    - PPDP co-processor
- Future works:
  - Where do we have to go next?

Service-Oriented Router Arhictecture



Our study components on SoRs

- Design Service-oriented Router/Switch Query Language based on SQL
  - Improvement in description and expression ability
  - Can handle packet, stream and transaction separately
- Provide applications
  - Spatial information
    - Optimization of physical path of P2P
    - To provides flexible contents delivery services (CDS): Shift to contents based CDS
  - Temporal information
    - To provide seeds of ranking for search engines according to visiting duration
  - Service-oriented database migration
    - To optimize the allocation of data resources by distinguishing attracted information
    - New recommendation services by sharing information between heterogeneous services
  - Service-oriented security
    - Contents based network intrusion detection system/service (NIDS)
    - Routing and contents based anti-phishing attack function
  - Privacy preserving data publishing

Current state of research

(1) SoR software simulator is now available

- Stream reconstruction function was proposed to reduce the size of packet buffer
- The simulator will be implemented on Juniper router and installed on SINET

(2) Hardware-based data insertion architecture was proposed

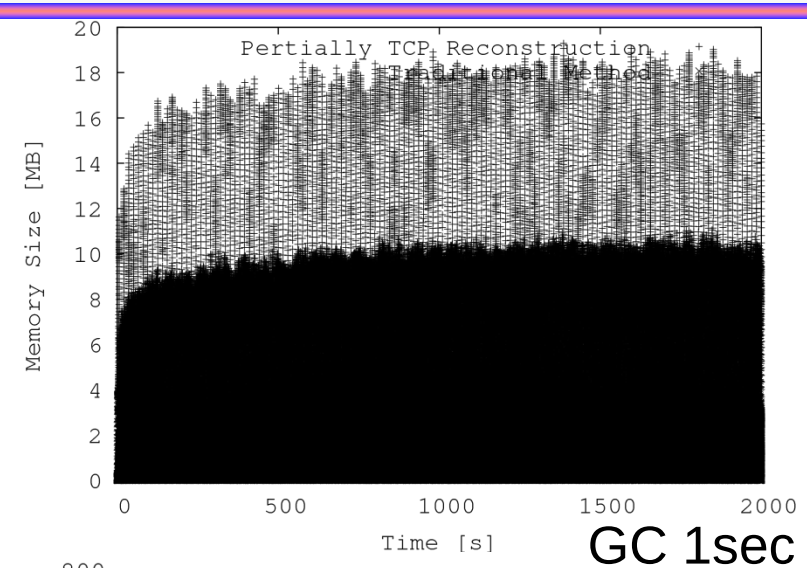
- For providing lossless data insertion and highly availability

(3) Privacy preserving hardware is now available

- FPGA based privacy preserving data publishing function is proposed, and special cache hardware is also proposed to eliminate information loss under the constraint of memory size.

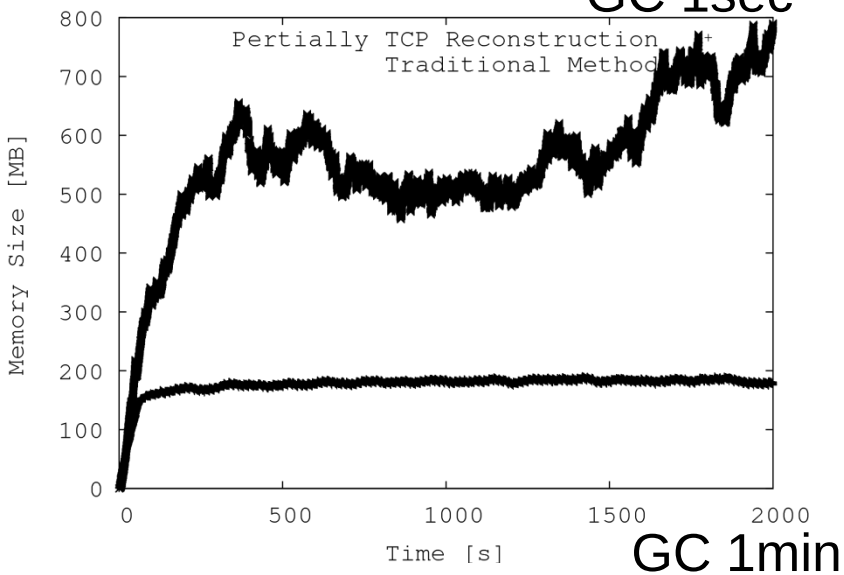
(4) Hardware based selection engine is under development

Software Version of SoR is released !!

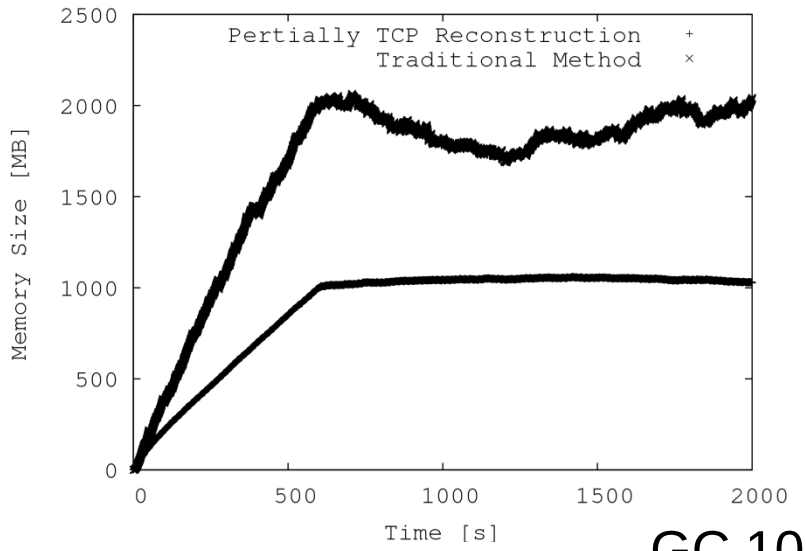


GC 1sec

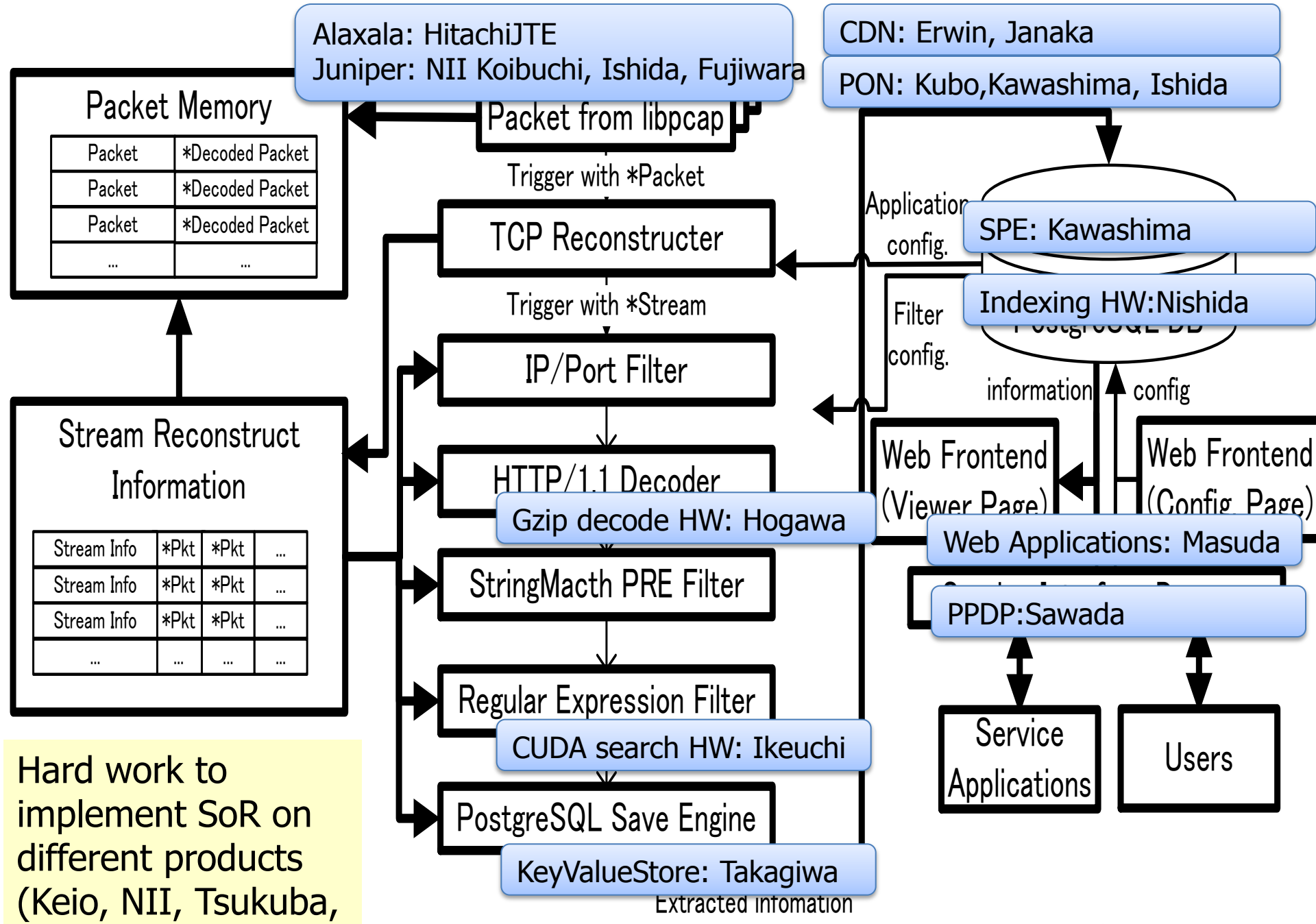
- The memory management stabilizes real ``JAM'' traffic on SLIM
- Porting to products, Junos SDK, Alaxala switch, etc



GC 1min

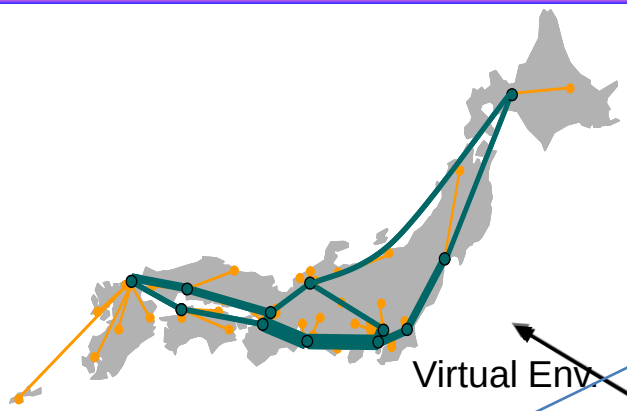


GC 10min



Hard work to implement SoR on different products (Keio, NII, Tsukuba, Hitachi JTE, NTT)

SoR on Junos SDK



Junos SDK dev.

negi on VM

Target Linux OS with remote Junos app(s)

JunosV App Engine

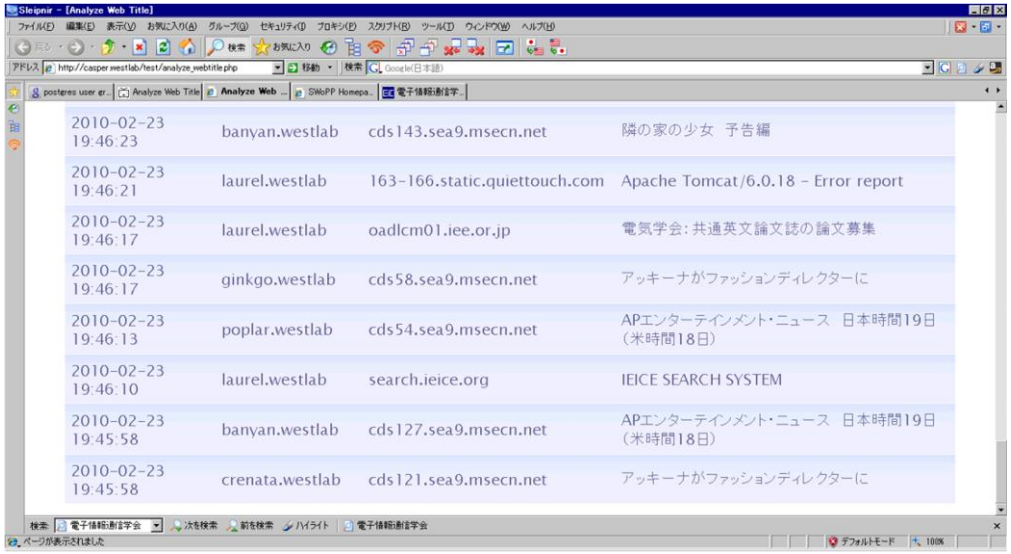
Virtual Env.

Remote API

Mirroring

Output Image

MX480(SINET4 compatible)

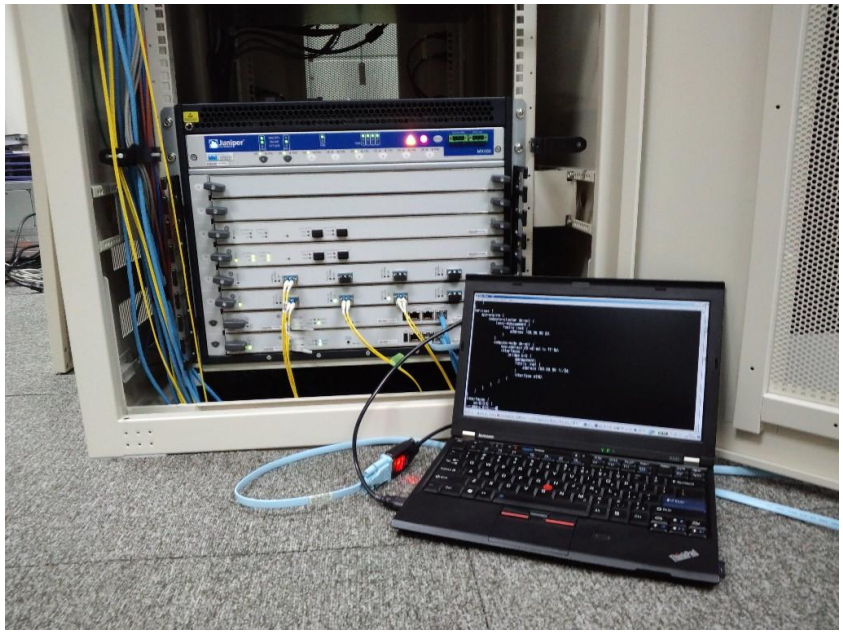


Working in collaboration with

Juniper MX480

SoR on Junos SDK (Cond)

- SoR as a VM on JunosV App Engine Env.
- In collaboration with MX480 via Junos SDK Remote API
- Software SoR C++ codes are running



Juniper MX480

Junos SDK Env. (SLIM on KVM)

Juniper Press Release

Oct, 2013: our experience with SoR

SINET network connects universities and research institutions within Japan and globally for real-time research collaboration. **In order for us to efficiently manage demand for greater bandwidth, latency-sensitive services and additional users, we needed more flexible and programmable architecture**," said Michihiro Aoki, Professor of the National Institute of Informatics (NII). "Juniper's innovative approach with the JunosV App Engine is the only solution for us to seamlessly create and simultaneously operate multiple applications on a single platform, reducing application migration time from months to just weeks."

Hardware-based data insertion

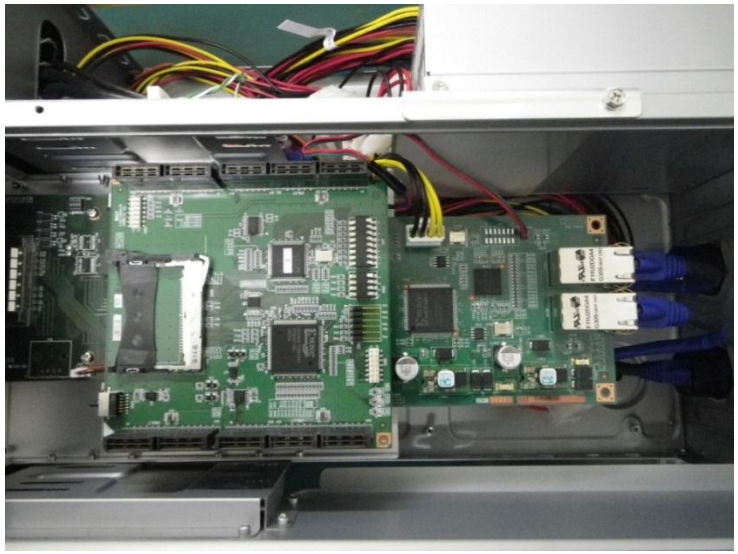
- 3 insertion methods were implemented and evaluated.
  - Simple linked list of insertion time (= network time)
  - Two-dimensional list of both insertion time and user ID for faster selection
    - Above two method can be implemented only on chip SRAM.
  - Memory effective insertion with hierarchical memory management
    - Though off-chip DRAM is required, complex selection can be handled.
  - Embedded processor with special microcodes provides flexible memory management.

MINPS = Million Insertions Per Second

	Network Throughput（Gbps）				Microcode Size (KB)
	On-chip Memory		Off-chip Memory		
Packet size	50B (min.)	1306B(HTML ave.)	50B	1306B	
Simple linked list	52.9Gbps (160MINPS)	67.4Gbps (160MINPS)	14.8Gbps (37MINPS)	67.4Gbps (37MINPS)	0.92
Two-dimensional list of both time and user ID	37.6Gbps (94MINPS)	67.4Gbps (94MINPS)	12.3Gbps (31MINPS)	67.4Gbps (31MINPS)	1.11
Hierarchical Index	0.66Gbps (1.65MINPS)	17.3Gbps (1.65MINPS)	0.62Gbps (1.5MINPS)	15.7Gbps (1.5MINPS)	1.81

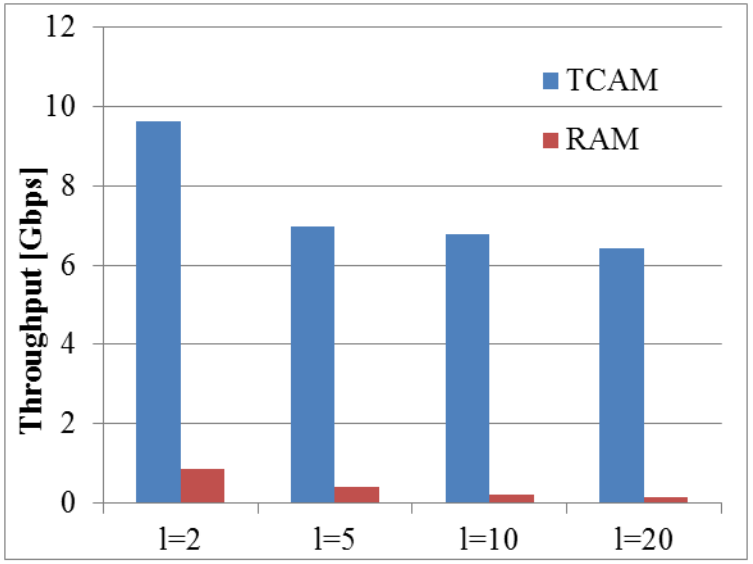
Hardware Emulator of SoR

- FPGA-base hardware emulator was designed.
  - 3 FPGAs and 2 full-duplex 1G Ethernet ports
  - 4Gbps wire-rate processing (ideal)
  - Fully hardware-based L2 to L7 packet parser
  - DBINS co-processor, µREGEX, PPDP, etc. are evaluated

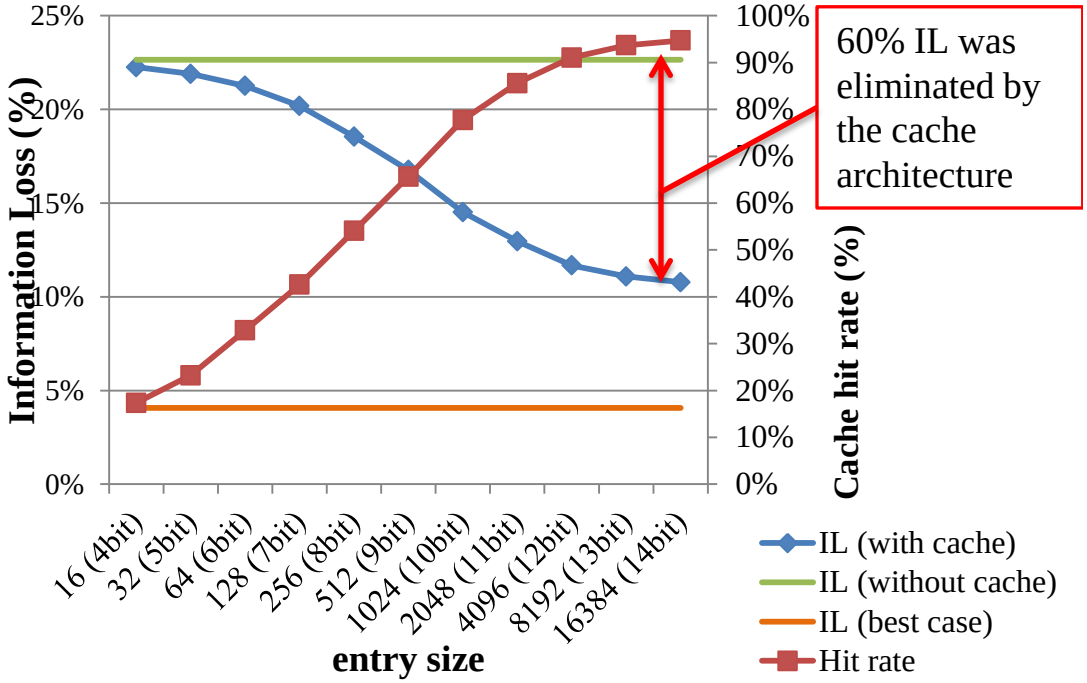


Privacy Preserving Data Publishing

- Hardware-based PPDP mechanism for wire-rate processing
- Both *k*-anonimity and *l*-diversity are supported.
- This mechanism uses TCAM for parallel data query (about 3,000 times faster than software-based PPDP) but hardware cost of TCAM is very high.
- To reduce the hardware cost, cache-based architecture are proposed.

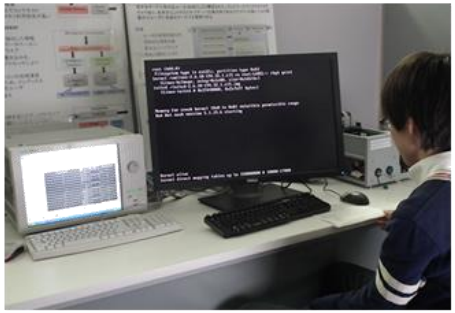


50 times faster than SRAM based architecture



Conclusions

- Purpose:
  - Why do we need a SoR (Service-oriented router) ?
- Achievement:
  - Service-oriented switch/router software/hardware emulator
  - Hardware architecture of service-oriented router
    - REGEXP co-processor
    - DBINS co-processor
    - PPDP co-processor
- Future works:
  - Where do we have to go next?

Thank you

Our web site of this project:
http://www.openinter.net