Efficient Cache Structures of IP Routers to Provide Policy-Based Services

IEEE International Conference on Communications   **June 2001**

Shingo Ata

Graduate School of Engineering
Osaka City University
ata@info.eng.osaka-cu.ac.jp
http://www.tama.info.eng.osaka-cu.ac.jp/

Research Backgrounds

- ✓ Bottleneck will be Shifting from Backbone to Edge Router, End Terminal, ...
- ✓ Policy-Based Service is Important to Satisfy Various Requirements of Applications

> ✓ Fairness among Users/Applications
> ✓ SLA (Service Level Agreement)
> ✓ Adaptive Rate/Route Control
> ✓ Security

**Policy Service
Flow-based Operation**

Shingo Ata

Policy Service Router Requirements

- ✓ Flow Classification
  - ➢ Search multiple fields in IP Header
    (IP Addresses, Port Numbers, Protocols...)
- ✓ Flow Based Scheduling
- ✓ Storing Flow State (Flow Value)
  - ➢ Store millions of flow values
  - ➢ Update the flow value for each packet arrival

Related Works

- ✓ Fast Address Lookup Algorithms by Using Cache Memories or CAMs
  - ➢ Not suitable because it is optimized for destination address lookups
- ✓ Flow Classification
  - ➢ Customized hardware is required
  - ➢ Increase the cost of routers
- ✓ Ternary CAM (Content Addressable Memory) Based Classification
  - ➢ Number of flow values is quite limited

Shingo Ata

Research Objectives

- ✓ A New Flow Classification Algorithm
  - ➢ based on traffic characteristics
  - ➢ supports storing flow values (flow state)
  - ➢ uses a general-purpose CPU and DRAMs
  - ➢ decreases the cost of implementation
  - ➢ is easy to speed up by replacing a new CPU
- ✓ Effect of CPU Cache Memory to Performance
  - ➢ Degradation of number of memory accesses
  - ➢ Effect of parameters about cache memories

Definition of Flow Classifier

✓ Sequence of Packets belonging to the Same Header as a Single "Flow"
{Src IP, Dest IP, Src Port, Dest Port, Protocol}

➢ Field length : 104 bits (IPv4)

➢ Few applications use multiple *Protocol* values

➢ Multiple connections by the same application can be treated as an aggregated single flow

✓ We use following three fields
{Src IP, Dest IP, Port Number}

➢ *Port Number* is the lower value between *SrcPort* and *DestPort*

# *Requirements for Storing Flow Values*

✓ Flow-based RED (Random Early Detection)

- ➢ Threshold of random packet discursion
- ➢ Queue length

✓ Application to the Core-stateless Router

- ➢ Rate can be considered with the unit of capacity (Link Capacity / (Max. # of Flows))

✓ DRR (Deficit Round Robin) Scheduling

- ➢ Supporting the deficit counter up to the maximum congestion window size in TCP

2Bytes flow value is sufficient

Structures of Cache Memories

- ✓ 2 Level Cache – Widely Used in Many CPUs
  - ➢ Clock speed Level 1 (L1) Cache
    - ☞ Implemented on CPU die
    - ☞ Cache size is quite limited (e.g., 16KB)
  - ➢ Level 2 (L2) Cache
    - ☞ Slower than L1 Cache
    - ☞ Larger than L1 Cache (512, 1024 or 2048 KB)
  - ➢ Cache Hit or Cache Miss

CPU	L1 Cache	L2 Cache	Main Memory

Shingo Ata
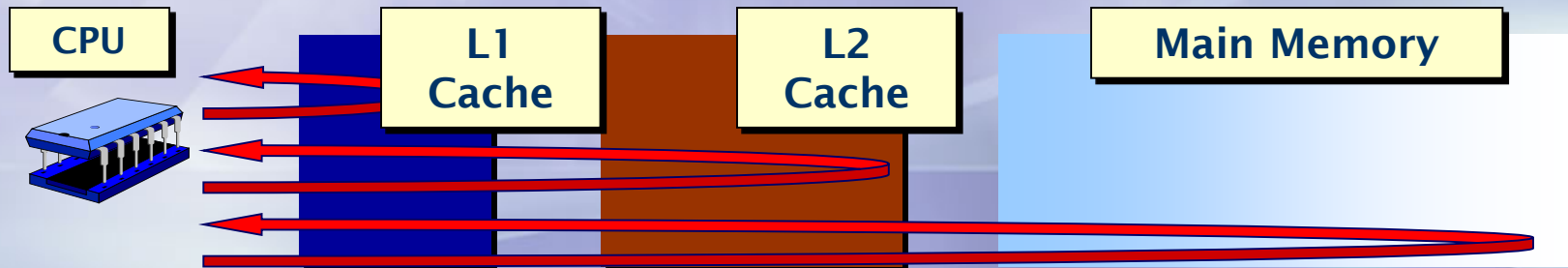
Effect of Cache Parameters

- ✓ Cache Size
  - ➤ Trade-off between costs and cache hit ratio
- ✓ Set Associative Mapping
  - ➤ Associability also increases cache hit ratio
  - ➤ Improves the utilization of cache memory
- ✓ Cache Block Size
  - ➤ Unit of data for reading / writing cache
  - ➤ Effective for accessing continuous addresses
- ✓ Cache Replacement Policy
  - ➤ In case the cache is not empty

Partitioning the Flow Classifier

- ✓ Flow Classifier Length : 80 bits
  - ➢ requires 2×2^{80} Bytes for direct access
- ✓ Sparse Utilization
- ✓ Divide into 3 Parts
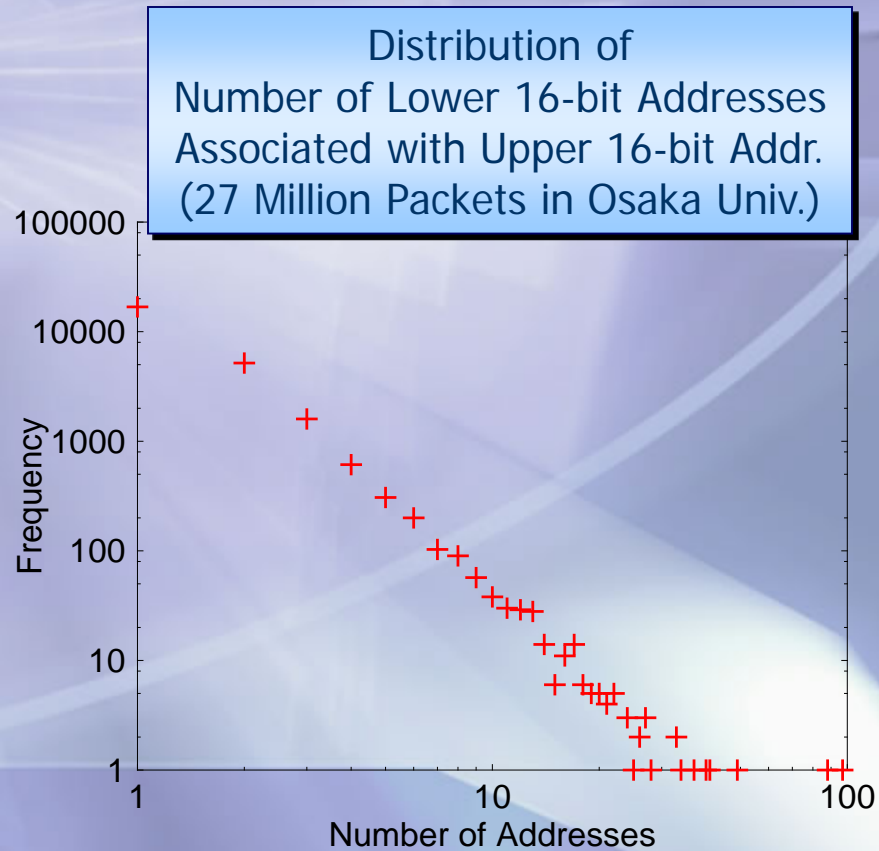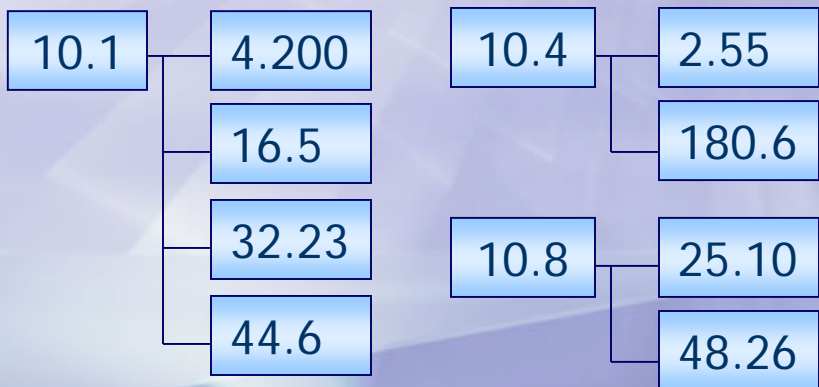 - ➢ Upper 16-bit Dest.
  - ➢ Lower 16-bit Dest.
  - ➢ Src. and Port

10.1	4.200
	16.5
	32.23
	44.6

10.4	2.55
	180.6
10.8	25.10
	48.26

Distribution of
Number of Lower 16-bit Addresses
Associated with Upper 16-bit Addr.
(27 Million Packets in Osaka Univ.)

Shingo Ata

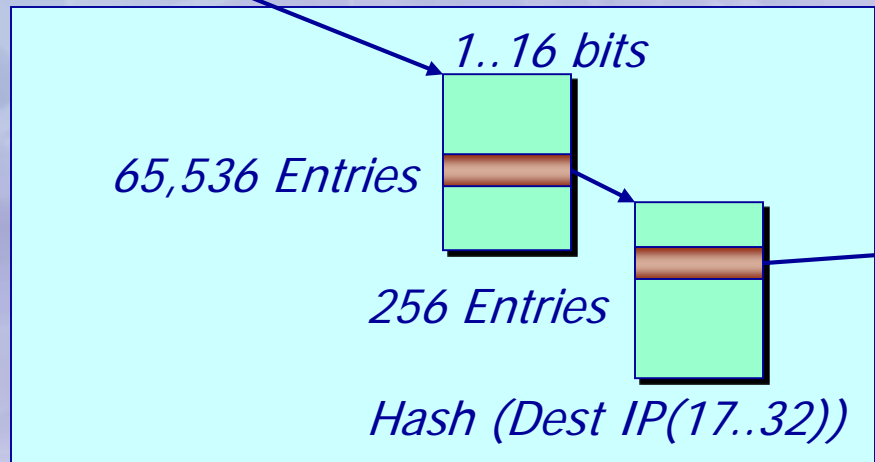# *Table Compression with Hash Function*

- ✓ Max. Number of Lower 16-bit Addresses : 97
  - ➢ Less than 1% of 2^{16} = 65536 entries
  - ➢ Use 256-entry hash table
- ✓ Max. Number of (Src., Port) Pairs : 766
  - ➢ Use 1,120-entry hash table

Distribution of
Number of (Src. IP, PortNumber)
Pairs Associated with Dest. Addr.
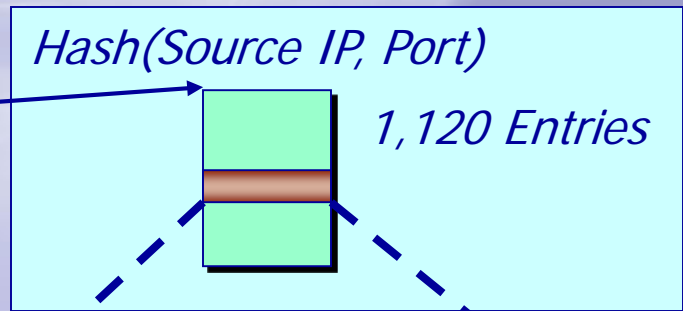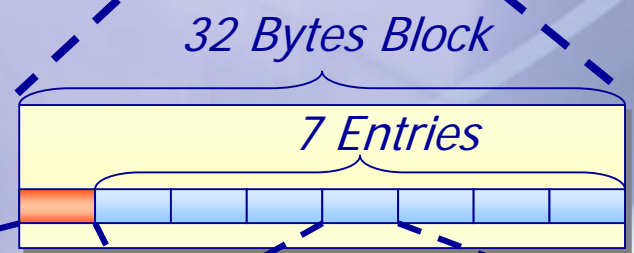(27 Million Packets in Osaka Univ.)

Frequency vs Number of Addresses

Shingo Ata
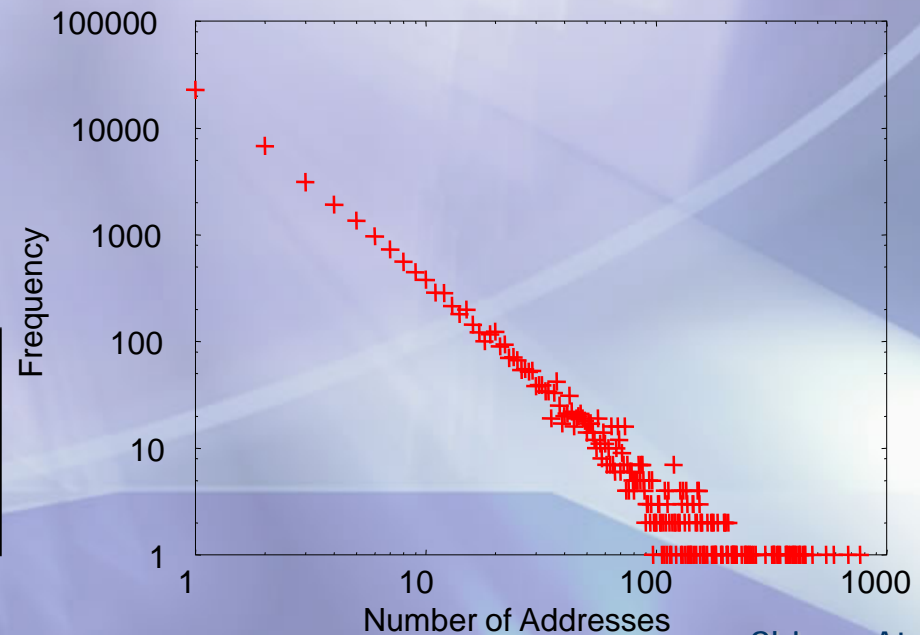
Algorithm of Flow Classification

Dest. IP Address

1..16 bits

65,536 Entries

256 Entries

Hash (Dest IP(17..32))

Destination Address
Lookup Table

Source Address and
Port Lookup Table

Hash(Source IP, Port)

1,120 Entries

32 Bytes Block

7 Entries

Destination IP
Address

Hash ID   Flow Value

Shingo Ata

Performance Evaluation

- ✓ Implemented on
  - ➢ Intel Pentium II 450 MHz
    - ☞ 16 KB 4 way set associative L1 cache
    - ☞ 512 KB 4 way set associative L2 cache
    - ☞ 32 Bytes block size
  - ➢ 256 MBytes Main Memory
  - ➢ Linux 2.0.36 / gcc version 2.95.2
- ✓ Packet Traced Data from OC3MON (Osaka Univ.)
  - ➢ 27 million packets, 9 million flow values
- ✓ Trace Driven Cache Simulator
- ✓ Calculate CPU Cycles for Flow Classification

Effect of Cache Memory Size

✓ Increase Cache Size ⟹ Small Access Delay

➢ Effect of L1 cache is significant

✓ No Improvement in Large Cache Size

➢ Over 64 KBytes (L1), 256 KBytes (L2)

L1 Cache
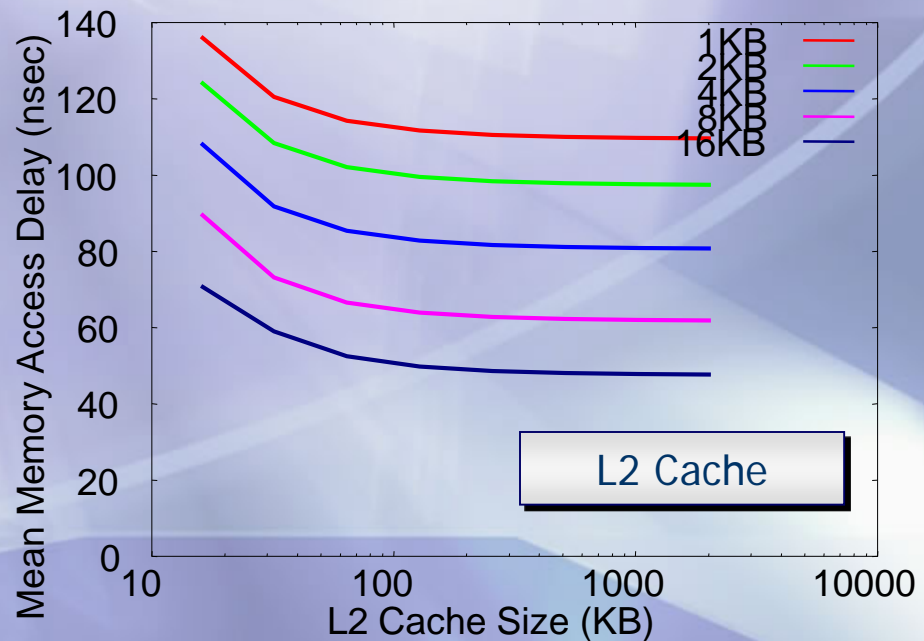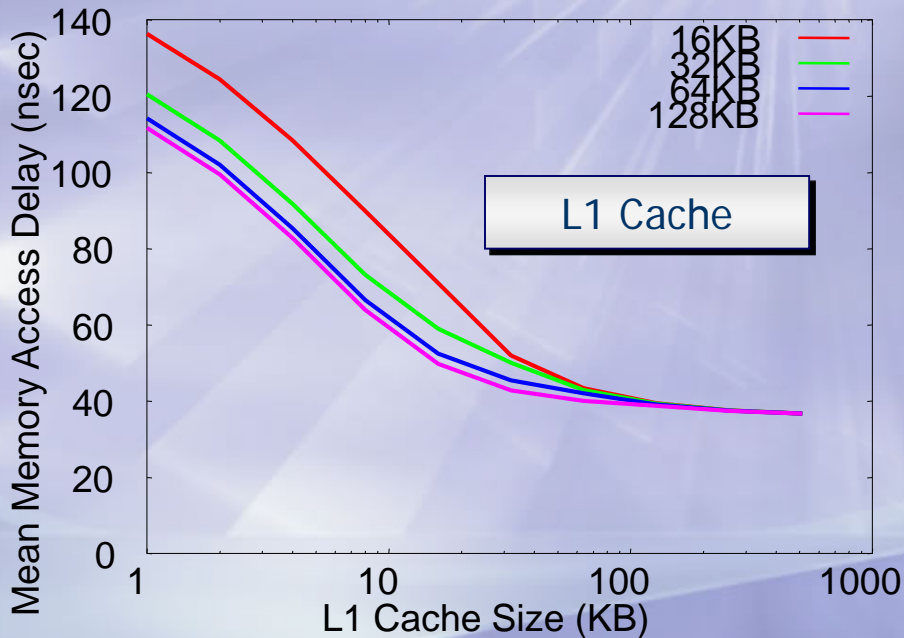
L2 Cache

Effect of Set Associative Mapping

✓ Increasing Associability is Effective

➢ L2 cache is small (64KB)

✓ 16 KB 4way = 32KB 2way

16 KB L1, 64 KB L2

Relation between Associability and Cache Size （L1）
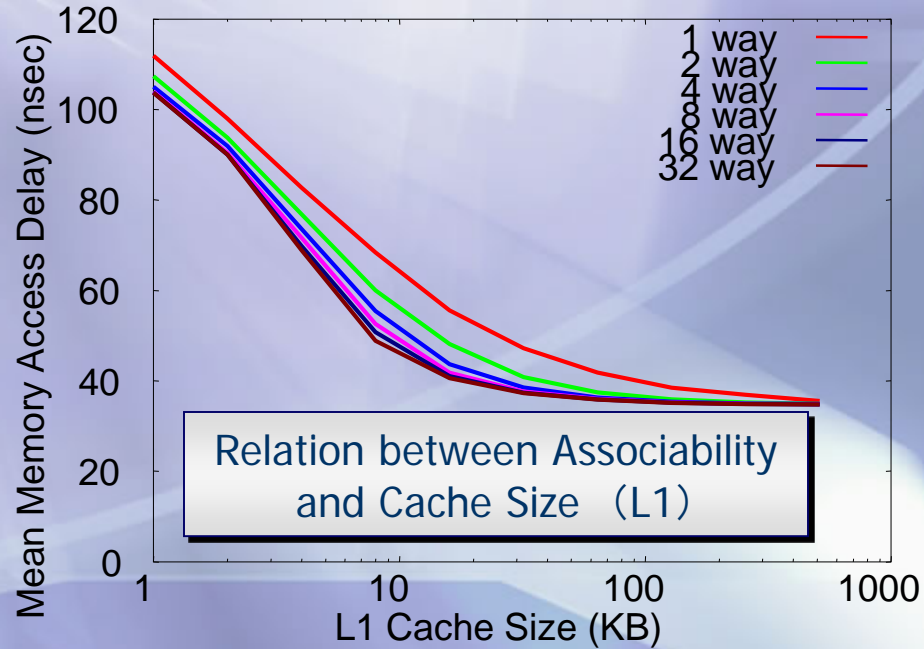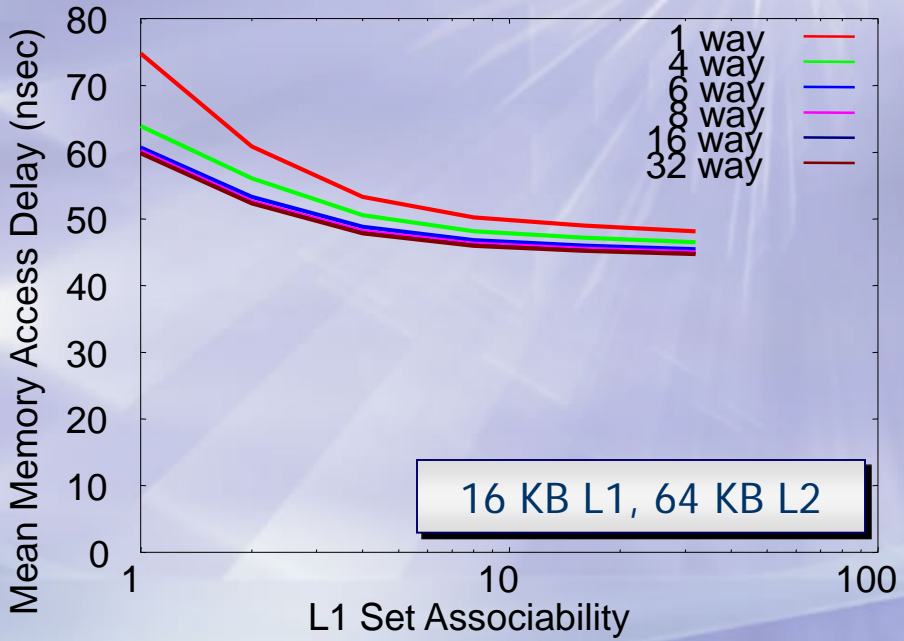
CPU Cycles and Required Memory Size

- ✓ Average 176 Cycles

- ✓ 391 nsec in Pentium II 450MHz

- ✓ Average Packet Processing Rate
    = 2.551 Million Packets/Second

- ✓ Required Memory Size : 49.1 MBytes
    - ➢ Reduced by Dynamic Table Allocation

Shingo Ata

Conclusion

- ✓ New Packet Classification Algorithm
  - ➢ Capable to store flow values
  - ➢ with commercially available CPU and RAMs
- ✓ Effect of Cache Parameters
  - ➢ L1 cache size gives a great impact
  - ➢ Set associative mapping is effective in small cache (approx. doubling the size)
  - ➢ Cache block size does not improve the performance
  - ➢ LRU is best, but FIFO is reasonable

Effect of Cache Block Size

✓ Increased Access Delay by Large Block Size

  ➢ Significant in small cache size (8KB)

  ➢ Increase the penalty of cache miss

  ➢ Effective in large cache size

Shingo Ata