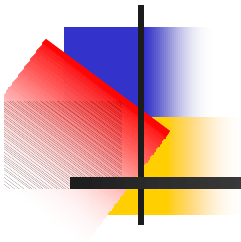
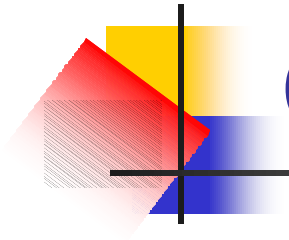


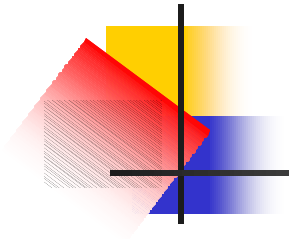
Network processors

Youngsuk jun



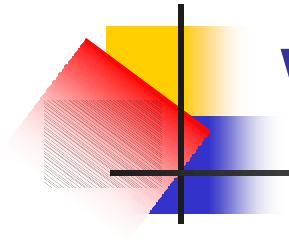
Outline

- 1.network processor-introduction
- 2.IBM PowerNP family
- 3.IBM PowerNP NP4GS3 in detail
- 4.Intel IXP family



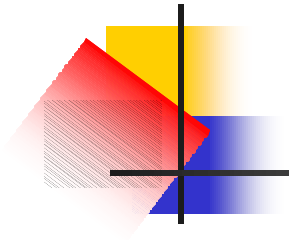
Market Segment Shift

- Delivery of raw BW to value added differentiated services.
- Charges are billed on contents rather than connection time.
- “best-effort” methods are being replaced by SLA
- “IP everywhere”
- Standard interface



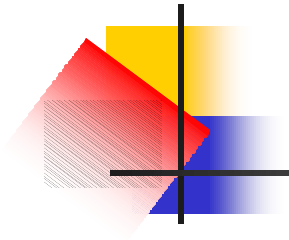
Why we need it?

- The demands on network infra continue to evolve and expand (more BW, lower latency, higher data rate, increased network intelligence): increasing BW is not always best solution!
- Network BW are outpacing processor speed: need to optimize hardware resource usage



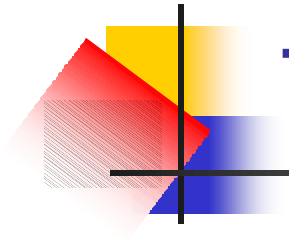
Network processor

- programmable chips that integrate the functions necessary to transport packets of data in a network
- Speed VS Flexibility tradeoff
- Real time processing, store and forward, security, switch fabric, IP packet handling
- Solve latency problem by implementing set of packet processing functions in hardware and off loading the programmable element
- Speed improvement through architecture: parallel distributed processing, pipelining



Requirement

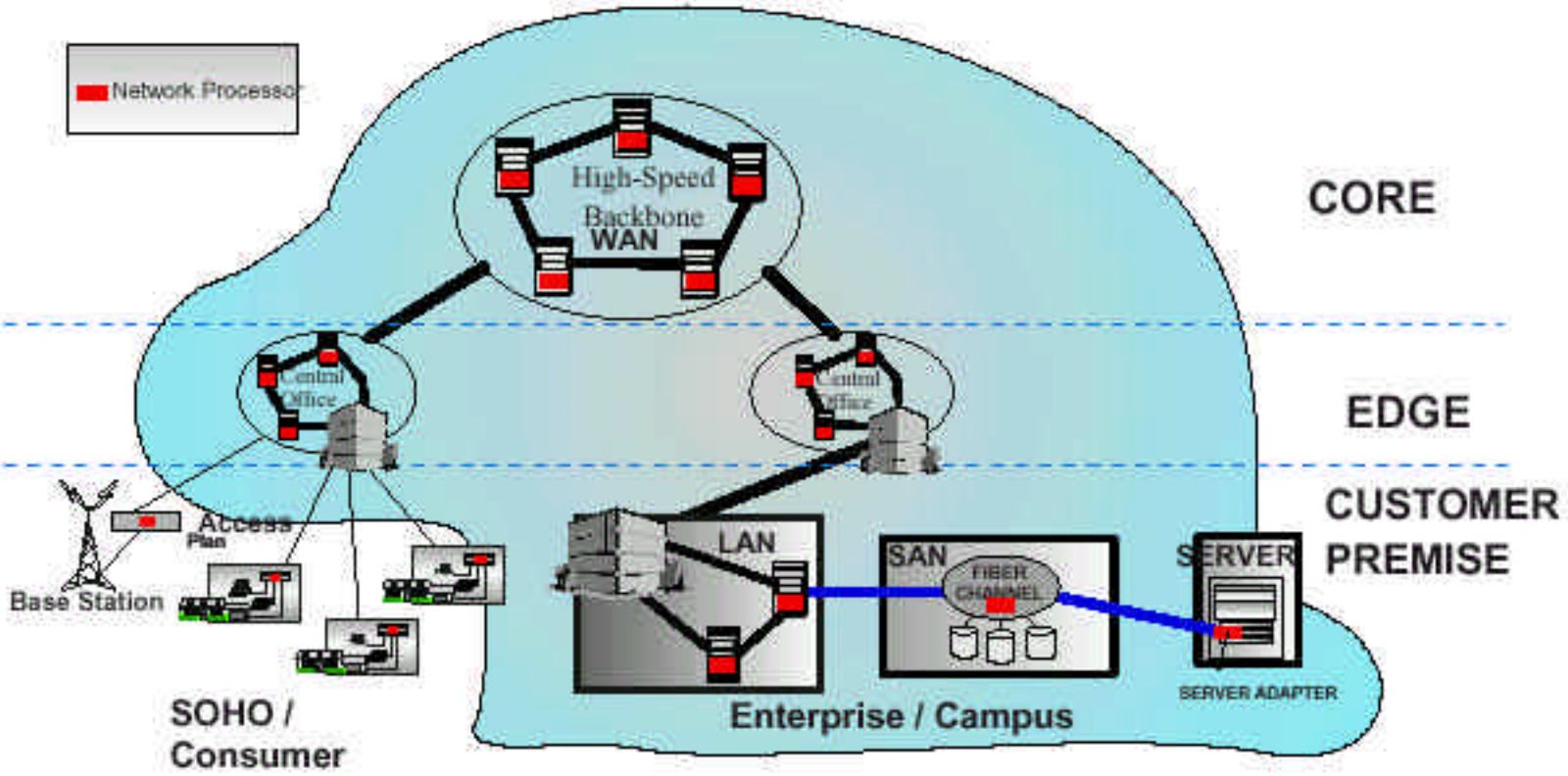
- Throughput, Flexibility, Scalability
- Support value added network service
  - Deep packet inspection at prevailing data rate
- Minimize deployment time & cost
  - Reusing code
  - Scalability

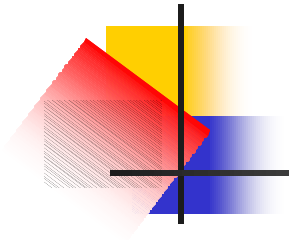


Three approaches to NP

- Pure hardware: configurable ASIC
  - Best performance, lack of flexibility
- Pure Software: programmable
  - Maximize programmability
- Hybrid: programmable ASIC
  - Current trend, maximize performance while maintaining flexibility
  - Routine network function implemented as dedicated hardware
  - Even more scalable than pure software approach

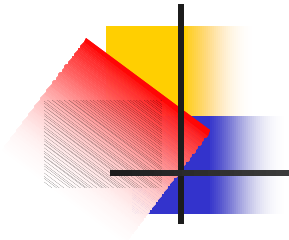
Core building block for a range of different network application





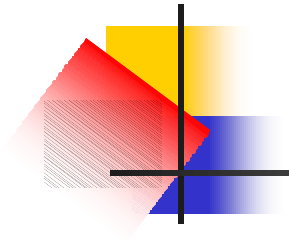
IBM PowerNP family

- PowerNP NP4GX, PowerNP NP4GS3, PowerNP NP2G, PowerNP NPe405H
- Programmable processor
- Multiple hardware accelator
- PowerPC control processor
- Allowing software portability and reuse with other PowerNP products



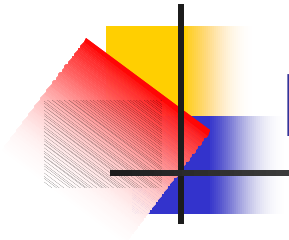
NP4GS3 provides flexibility

- Programming design which allows implementors to add function and make changes quickly and easily
- Embedded PowerPC provides additional design flexibility for network interconnect devices
- High capacity DDR DRAM memory allows for large output buffering and control store for significantly larger forwarding table, traffic flow queues at a lower cost



NP4GS3 provides scalability

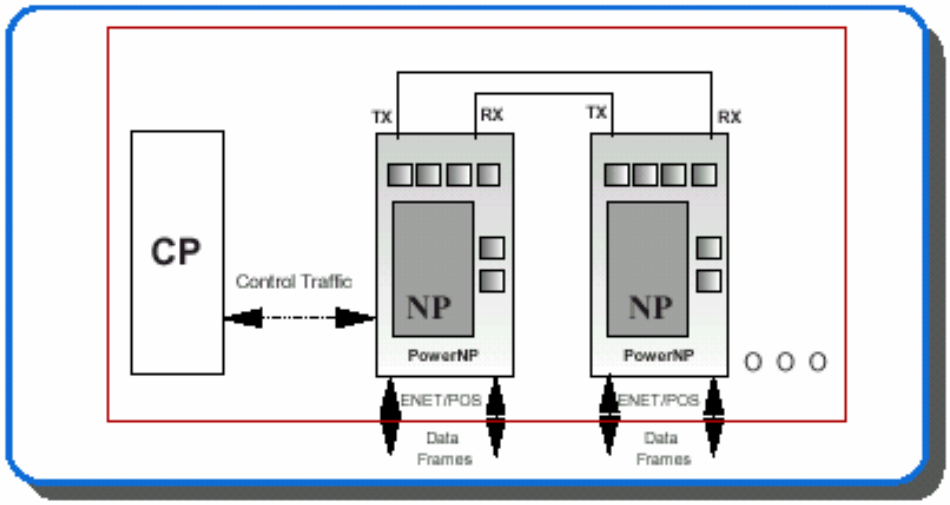
- Standalone system supporting up to 40 10/100 ethernet ports or 4GB ports
- Multiple system complex with up to 64 NPs joined together providing combination of ehernet, GB, OC3, OC12, OC48 ports



NP4GS3 achieves high performance

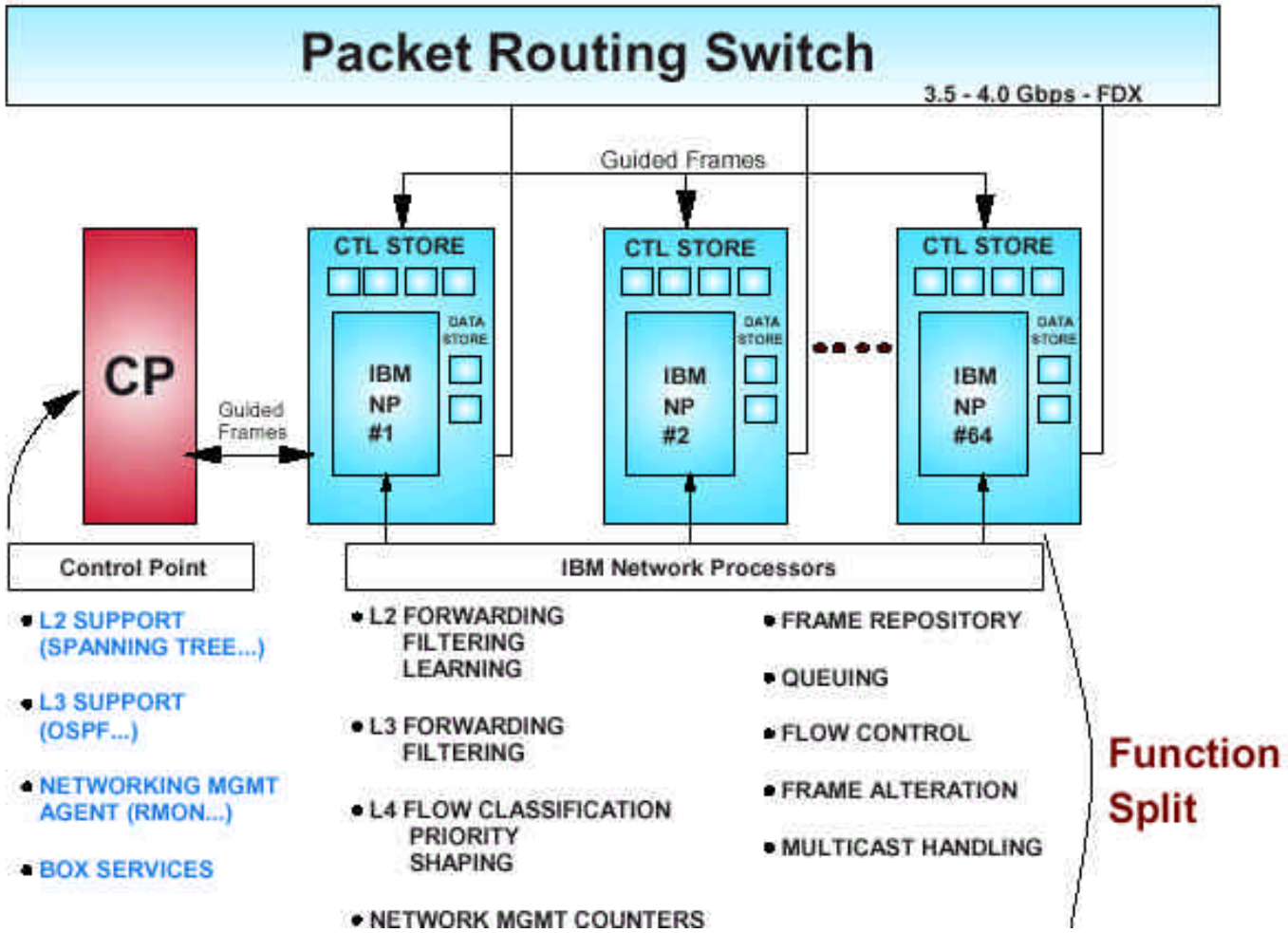
- Picoprocessor, Coprocessor, and HW assists: 8 dyadic provide 16 active threads and 16 inactive threads(up to 32 frame processing at once)
- Forwarding/filtering without data copy
- Layer2,3,4 and higher functions
- Support large lookup table

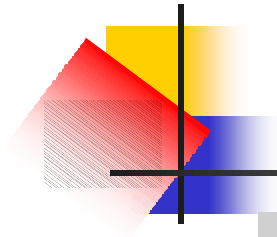
Functional distribution



- **N**etwork **P**rocessor performs *Steady State* Functions such as:
  - Filtering, Frame Forwarding, Frame Alterations
- **C**ontrol **P**oint Processor performs *Non-Steady State* Functions such as:
  - OSPF DB Updates, Box Services, Deep Frame Processing ... and executes **N**etwork **E**quipment **V**endor Applications
- One **CP** can service a system of many NPs
- The **NP** is designed to accommodate many vendor designs with alternate **CP-NP** configurations

Scalability with multiple NP

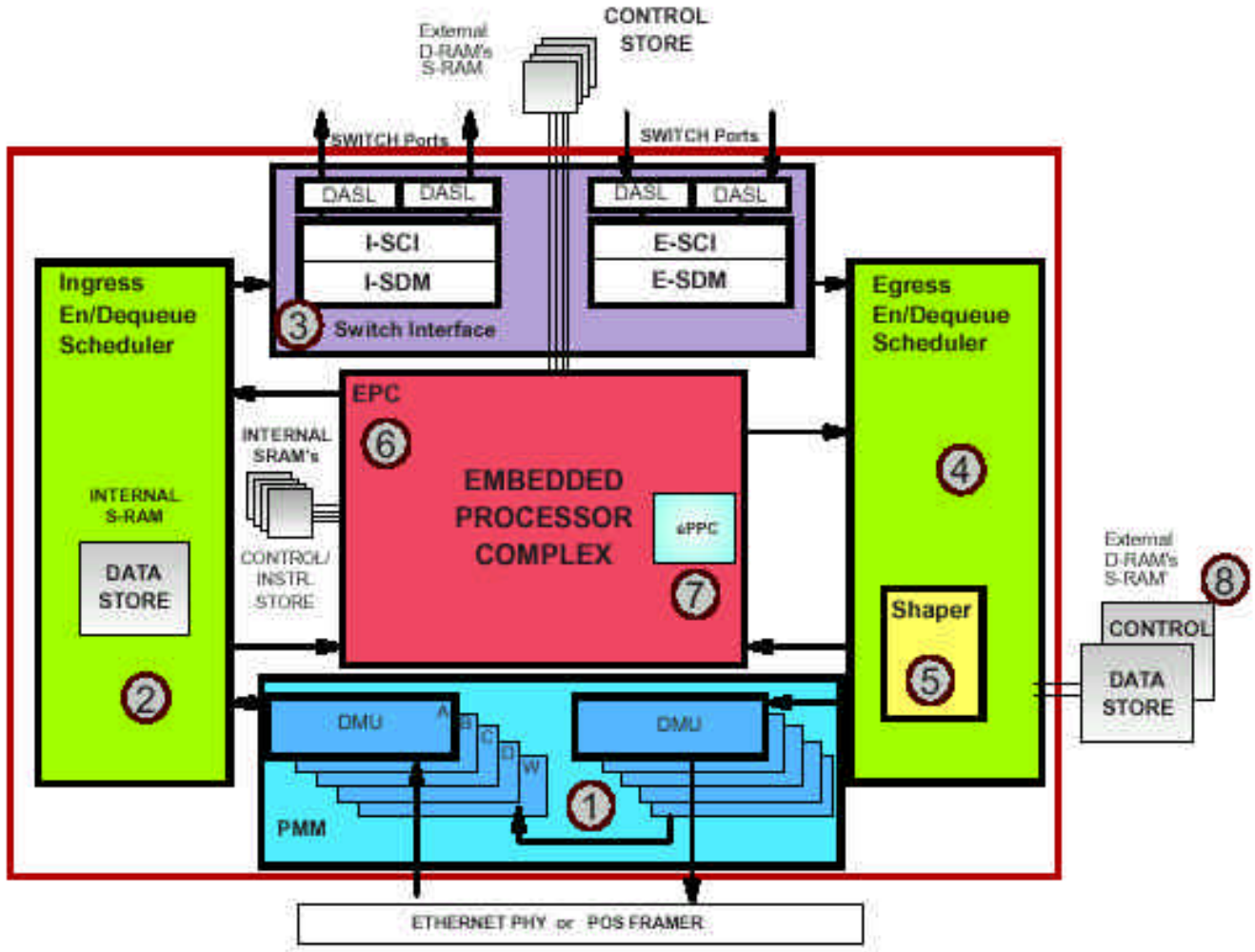




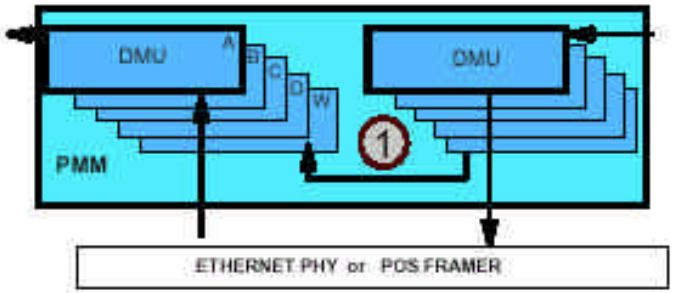
NP's major component

1. Physical MAC Multiplexor PMM
2. Ingress Enqueue / Dequeue Scheduler (I-EDS)
3. Switch Interface
   - Switch Data Mover
   - Switch Cell Interface
   - Data-Aligned Serial Links
4. Egress Enqueue / Dequeue Scheduler (E-EDS)
5. Traffic Shaper
6. Embedded Processor Complex (EPC)
7. Embedded Power PC Complex (ePPC)
8. Storage areas throughout the system

Architectural view

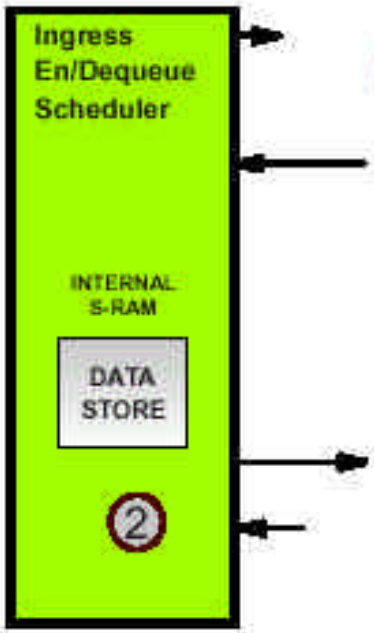


Physical MAC multiplexor



1. **Physical MAC Multiplexor (PMM): Provides interfaces with NP's external ports**
   - **Data Mover Units (DMU)**: 5 per Ingress, 5 per Egress
     - 4 ingress/egress pairs for external interfaces which provides configurable interfaces for specific media types and each can be configured to support a different media type per DMU:
       - 10 x 10/100 FDX ETHERNET ports per DMU
       - 1 x 1Gb ENET per DMU
       - 4 x OC3 POS per DMU
       - 1 x OC12 POS per DMU
       - 1 x OC48 POS per 4 DMUs
     - 1 ingress/egress pair for internal NP communication (Wrap Port)

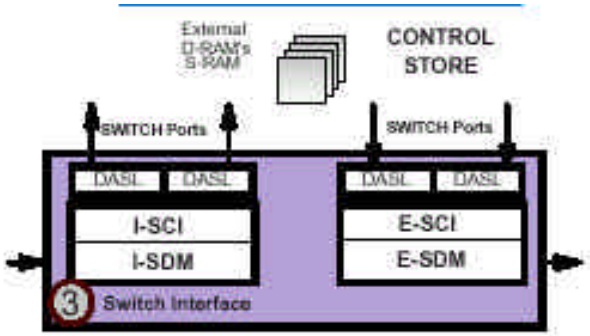
Ingress EDS



2. Ingress Enqueue / Dequeue Scheduler (I-EDS)

- Forwards Frames received via the ingress DMU
  - Stores frames from DMUs into data store
  - Performs Filtering Decisions and some standard frame alteration (VLAN Tags, etc.)
  - Dequeues frames from data store and Schedules for forwarding / discarding

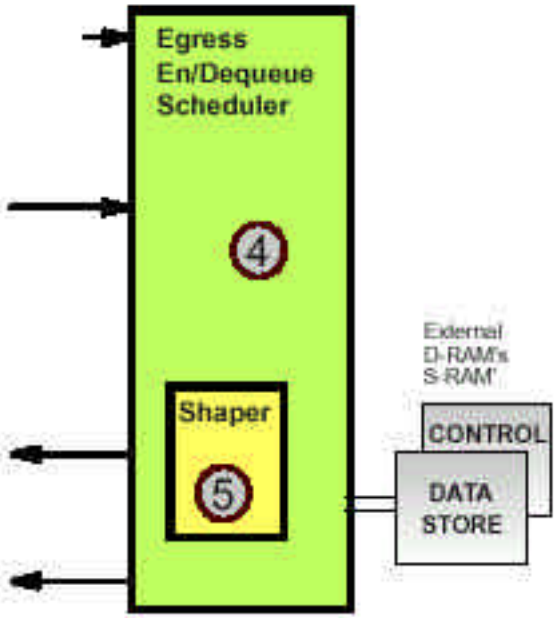
Switch interface



3. **Switch Interface (SWI)**: Provides a data cell based interface between NPs either via a switching fabric (for 3 or more NPs) or direct "wire" connections (for 1 or 2 NPs)
   - **Ingress/Egress Switch Data Mover (I/E-SDM)**: Logical interface between the I/E-EDS and the cell flow
   - **Ingress/Egress Switch Cell Interface (I/E-SCI)**: Provides/Receives cells to/from the physical interface
   - **Data-Aligned Serial Links (DASL)**: physical interface (wire) between:
     - NP and switch fabric
     - Ingress and Egress sides of one NP
     - Ingress and Egress sides of two NPs

E-EDS and Shaper

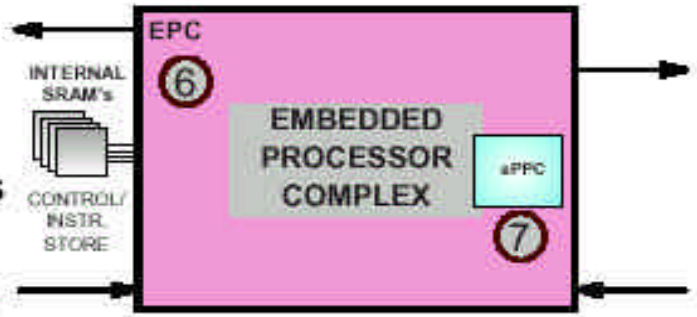
4. **Egress Enqueue/Dequeue Scheduler (E-EDS):**
   - Receives frames via the switch interface
   - Reassembles and Enqueues frames from switch interface into data store
   - Provides extensive frame processing
   - Dequeues frames from data store, and schedules for forwarding
5. **Shaper:**
   - Manages bandwidth on a per frame basis for egress-DMU ports
   - An optional (via software configuration) NP component



EPC and ePPC

6. Embedded Processor Complex (EPC):

- Contains the 8 Dyadic Picoprocessors and 9 Hardware Assist Coprocessors
- Determines what to do with frames received by the Ingress and Egress sides of the NP



- Provides the overall Steady State control and programmability of the NP - the code that makes the NP a 'programmable ASIC'

7. Embedded PowerPC (ePPC):

- PowerPC 405 engine designed for the NP
- Processor for Control Point Functions

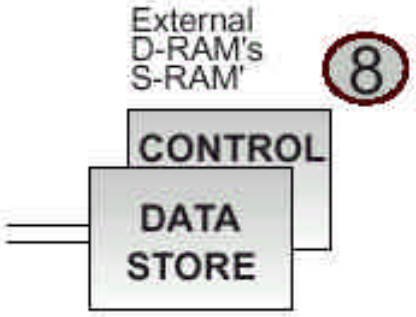
External processors may also be used in larger systems to provide CP functions and the ePPC would be available for other functions per the design

Memory

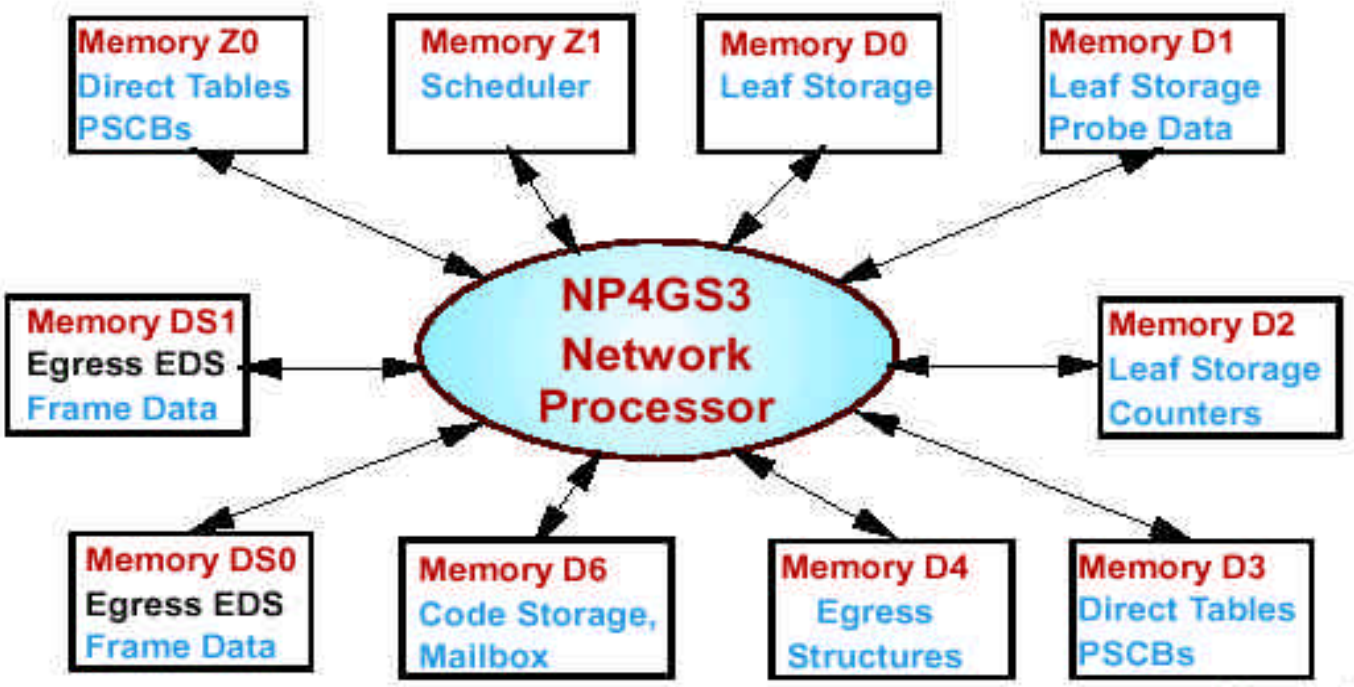
8. **System Memory: different types characterized by:**
   - ✔ **Location**
     - Internal: physically on the NP ("on-chip")
     - External: physically off but attached to the NP ("off-chip")
   - ✔ **Physical Types and Location**
     - Internal SRAM (Static)
     - External ZBT SRAM (Zero Bus Turnaround)
     - External DDR SDRAM (Double Data Rate Sync. Dynamic)
   - ✔ **Use**
     - For internal NP control information
     - For storing frame data (Access is in increments of 16 bytes - 128 bits)

RAM Locations and ID's

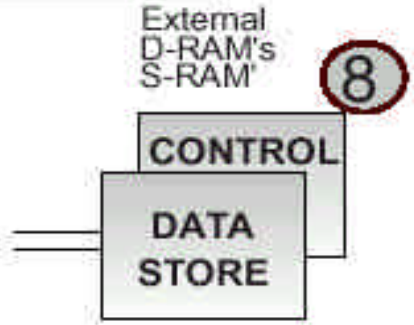
DDR SRAM - D0, D1, D2, D3, D4, D6
ZBT SRAM - Z0
DDR SRAM - DS0, DS1



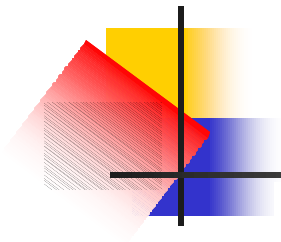
Memory Usage



Direct Tables, Leaf Storage and Pattern Search Control Blocks (PSCBs) are structures which define Trees and are used by the Tree Search Engine (TSE), when invoked, to locate / update tree data.



User programmable picoprocessors with HW assist coprocessors

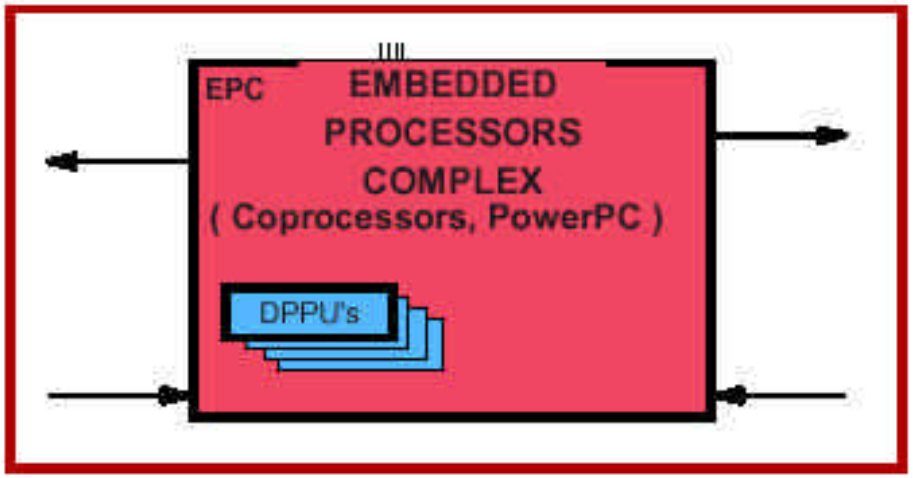


- The NP architecture allows for either an embedded, user programmable IBM 405 PowerPC ***-or-*** an external CPU to provide system control point function.
- The NP System has eight Dyadic Picoprocessors with nine Hardware Assist Coprocessors for each.
- Vendors can build unique product around the NP and use its programmability to define whatever function(s) they need
  - ✔ NEVs are now free from ASIC Development
- The NP software offerings have both high level (C APIs) and low level APIs provided by IBM's Advanced Software Offering that developers can use to interface with the NP system with their programs executing in the Control Point Processor.

This built in package is called the Embedded Processor Complex and it is the core of the NP system.

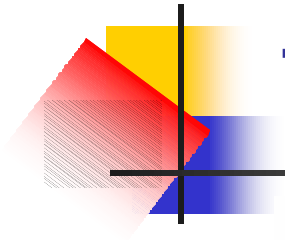
HW assist coprocessors

1. Coprocessors provide parallel function data movement
2. Maintain flow information for flow control
3. Access internal registers
4. Maintain count for flow control, etc.
5. Maintain a set of scalar registers and array per thread



The suite of coprocessors

- **Data Store Coprocessor**
  - Data transfer (R/W) between ingress/egress data stores and shared memory data pool
  - 128-bit per transfers
- **CAB Interface Coprocessor**
  - Provides all DPPUs access to internal registers, counters and memory for debug or statistics gathering
- **Enqueue Coprocessor**
  - Interfaces with the Completion Unit to enqueue frames to the switch and target port queues
- **Checksum Coprocessor**
  - Half-word data to generate half-word header checksums
  - RFC 1071 Computing Internet Checksum
  - Instr: generate checksum, verify checksum
  - Checksum in accumulation scalar register
- **String Copy Coprocessor**
  - Move multibyte data within shared memory pool
  - Command passes Saddr(14), Daddr(14), # of bytes
- **Policy Coprocessor**
  - Examines flow control information and checks for conformance with preallocated bandwidth
- **Counter Coprocessor**
  - Interface to the counter manager to a thread
  - Update count; 8 deep command queue
- **Semaphore Coprocessor**
  - Controls access to shared resources such as tables
  - Grant Modes either: Dispatch Order or Request Order

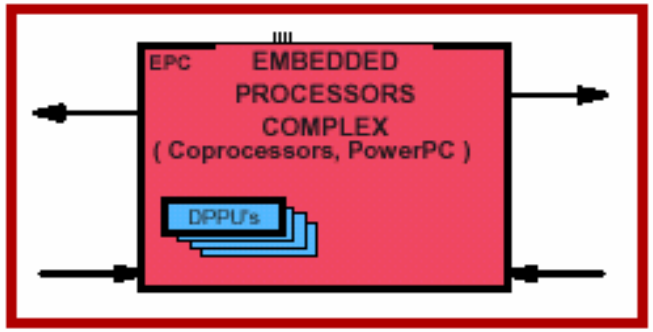


TSE-table search & update

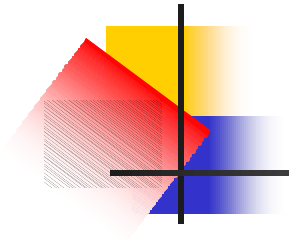
- **The Tree Search Engine Coprocessor**
  - Provides Tree Search / Modification functions for requests issued by Picocode threads. Uses two Coprocessor locations so that a thread can execute two searches simultaneously.
  - The NP relies heavily on searching of Tree Structures for functions such as:
    - L3 IP Address Routing Tables
    - L3 and higher frame filtering
    - L2 MAC address port mapping
    - Flow Control
  - There are three types of Tree Searches supported:
    1. FM - Full Match
    2. LPM - Longest Prefix Match
    3. SMT - Software Managed Trees - can have multiple leaves that can be chained in a linked list.

Other HW assist functions

- **Dispatcher**
  - Track thread usage; fetch initial frame data before thread assignment
- **Completion unit**
  - Maintains the order of frames enqueued to ingress/egress flow control and scheduler function
- **Policy Manager**
  - Performs Policy Management based on four management algorithms specified in IETF RFCs 2697 and 2698:
    - **Single Rate Three Color Marker (color blind or color aware modes)**
    - **Two Rate Three Color Marker (color blind or color aware modes)**
- **HW classifier**
  - ETHERNET type (802.3, DIX)
  - Layer 3 type (IP)
  - VLAN header detection
  - Guided Traffic



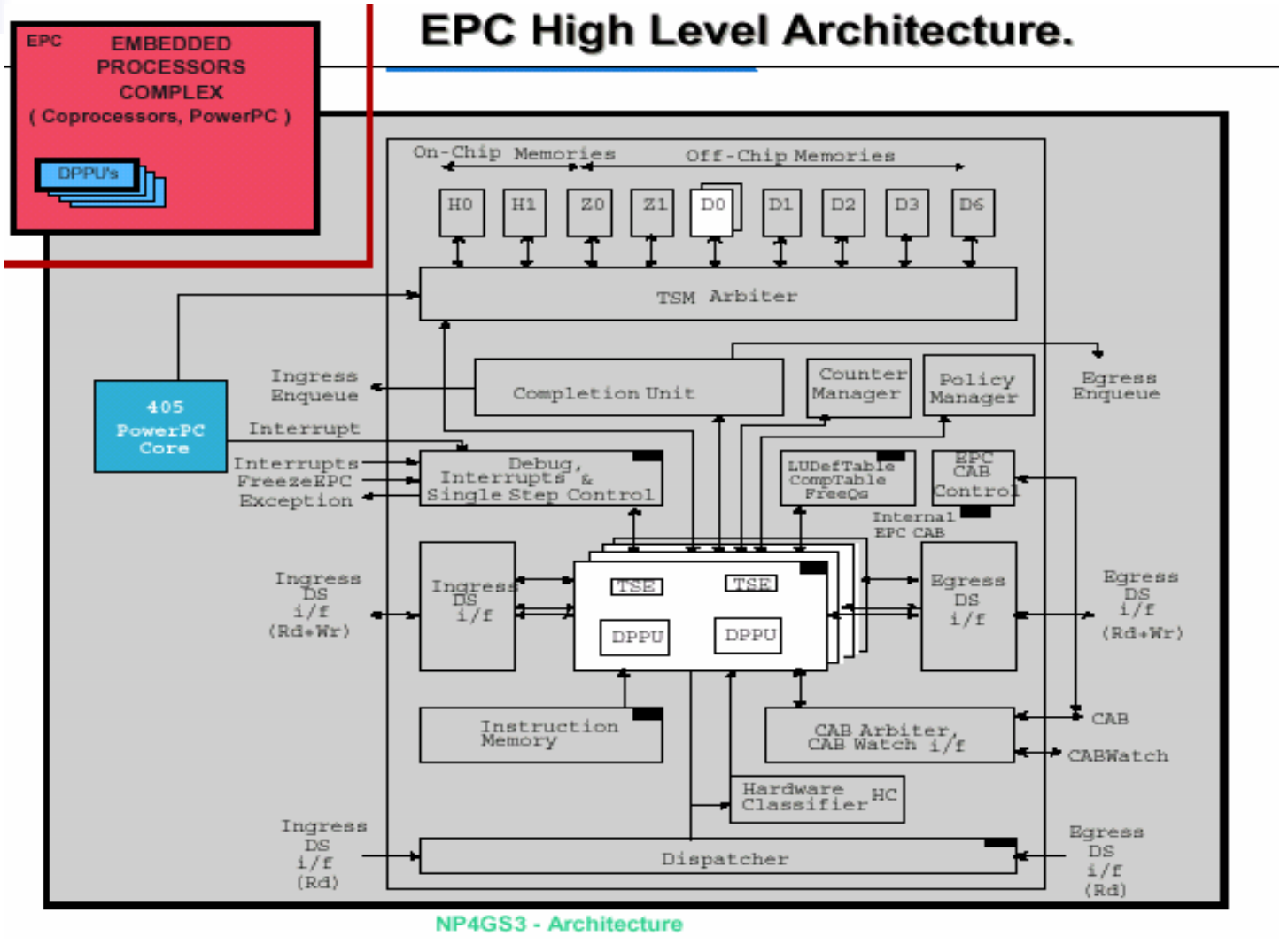
- **Counter Manager**
  - Used by EPC to control various counts used by the Picocode for:
    - **Statistics, Flow control, Policy Management**



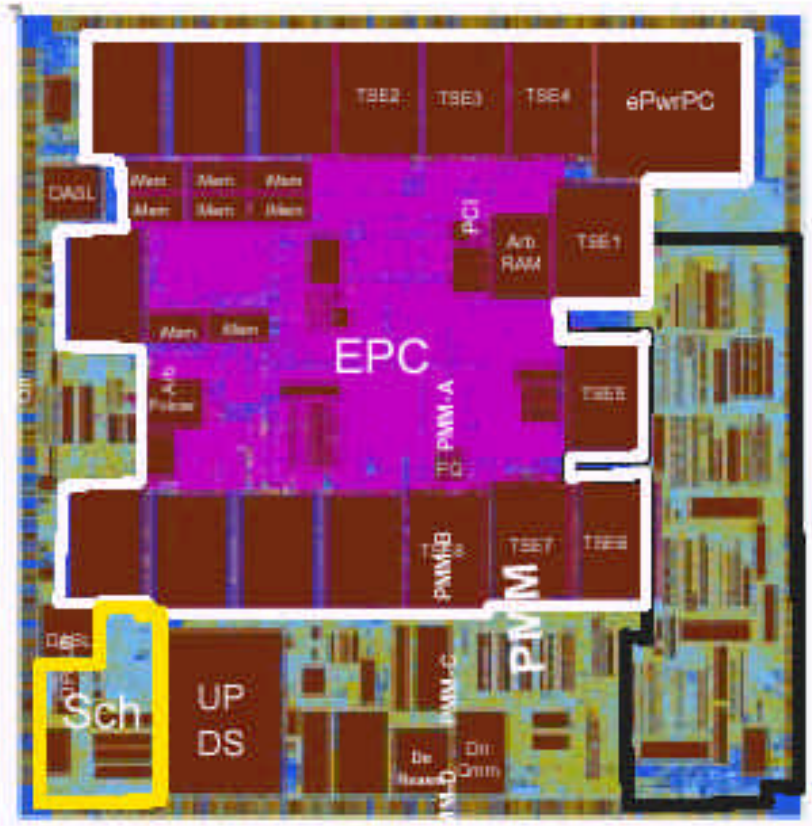
Picoprocessors

- **32-bit RISC core**
- **1-cycle ALU w/ arithmetic, logical, compare, shift/rotate, bit test/set/clear instructions**
- **Thread management w/ 0-cycle context switch**
- **16 32-bit GPRs per thread**
- **Read-only scalar register**
  - interrupt vectors, timestamp, pseudo random number, processor status, work queue status (ingress/egress data queue status, etc.)
- **16-word instruction cache (shared by threads)**
- **Coprocessor data & execution interface**
- **128 KB shared data pool**
- **Instruction execution unit**
  - executes branch instruction, instruction fetch, coprocessor access

EPC high level architecture

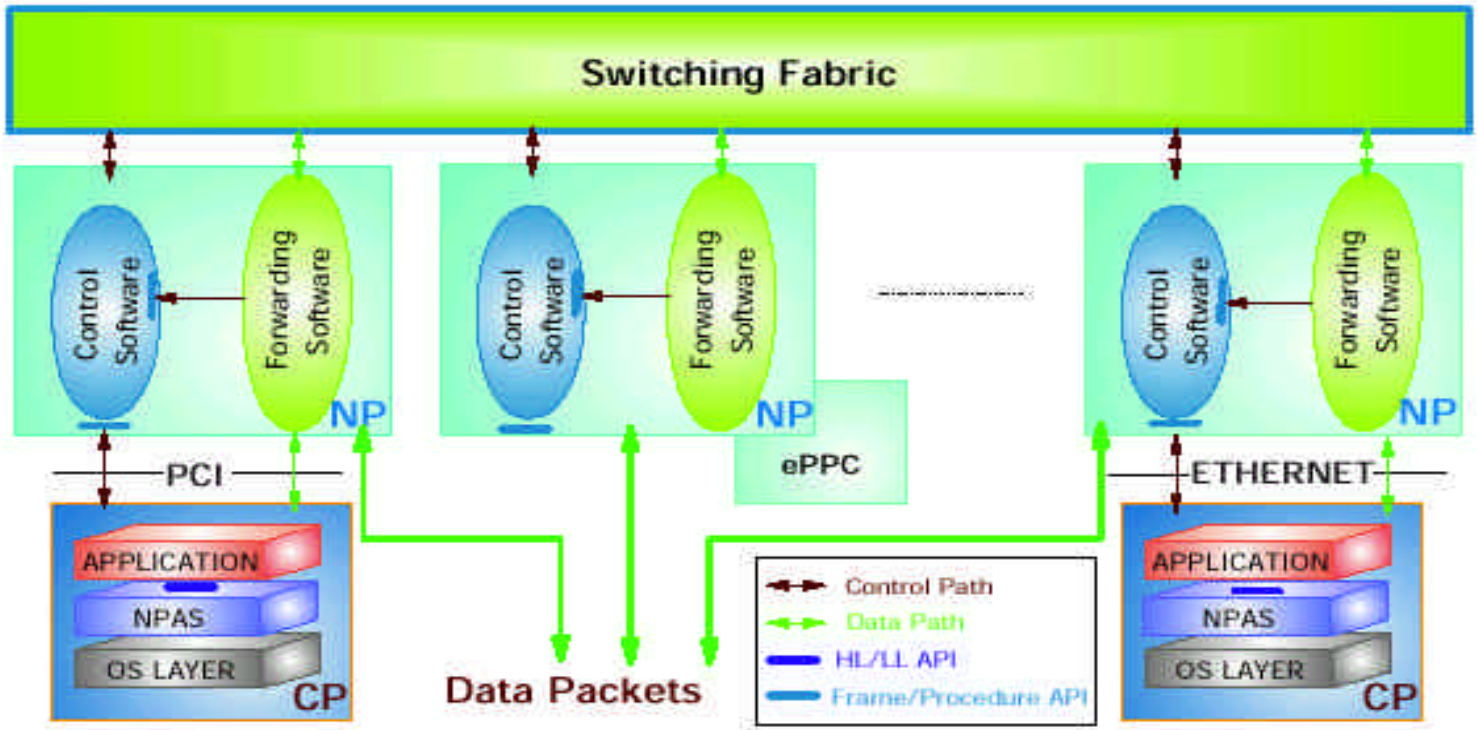


NP4GS3 physical layout

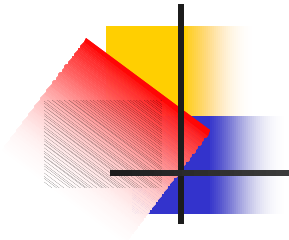


- 0.18 um Cu-process
- 133-MHz core clock
- 1088-pin package
  - 815 signal I/Os (total)
  - 777 signals used
- 24-pin debug bus
  - per island debug funct.
- 1.8V Core voltage
  - 2.5V (DDR, ZBT RAMs)
  - 3.3V (DMU, PCI)
- Power dissip. 14W (est)

Flexibility with programmable NP and CP

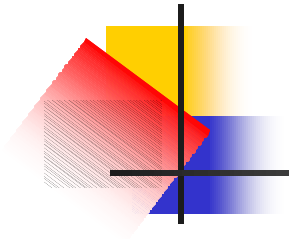


Network Equipment Vendors can create Picocode for processing in the NP or applications running in the Control Point using the NPAS APIs to communicate with the NP.



Intel IXP family

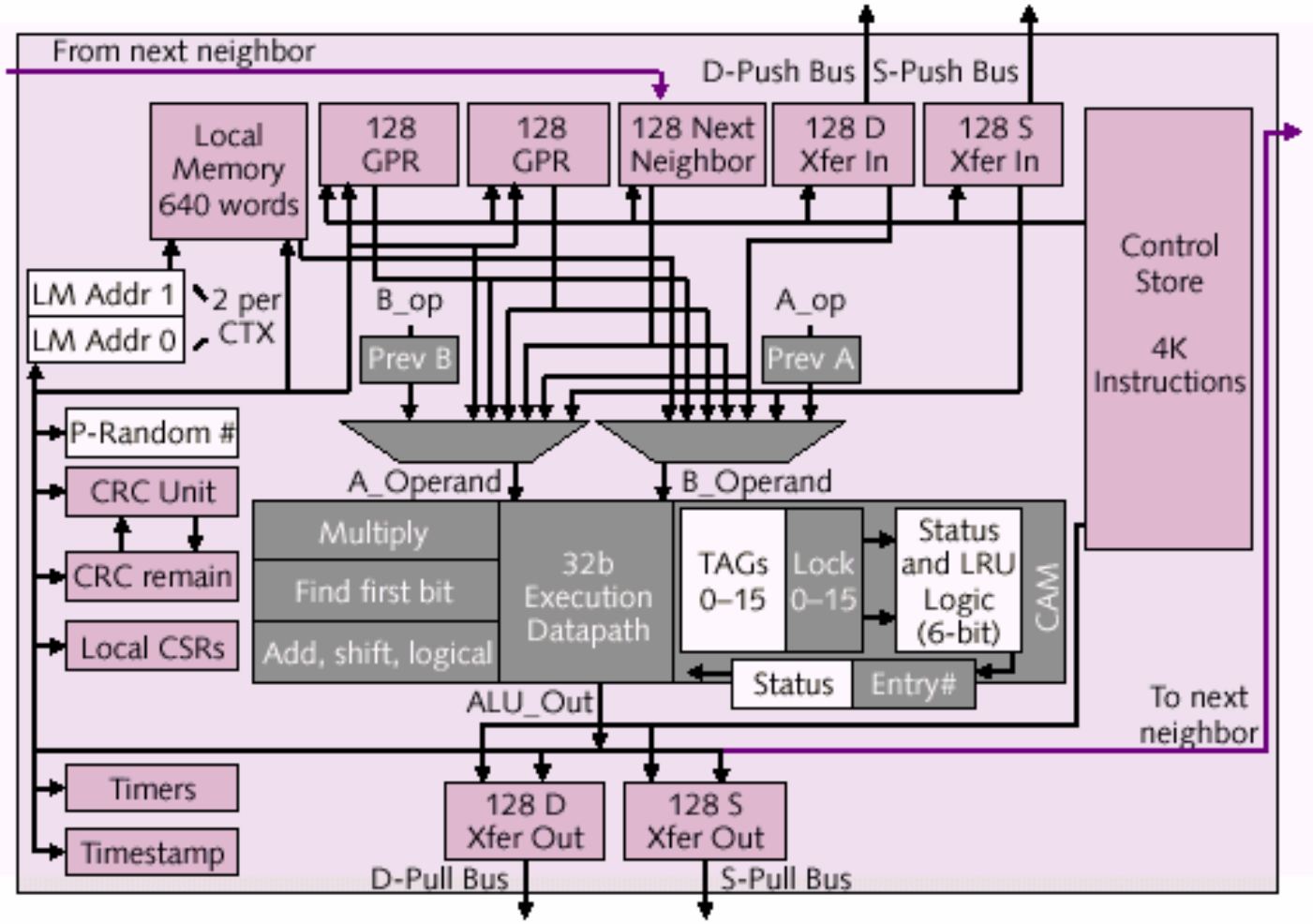
- 3 new processors designed to meet respective market segment (Core & Edge-IXP2800, Edge & Access-IXP2400, CPE-IXP425)
- Key architectural characteristic
  - Microengine technology
  - Xscale technology
  - IXA software portability framework

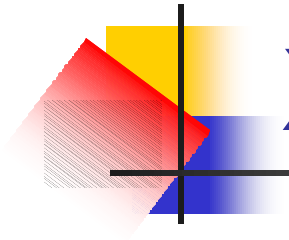


Microengine technology

- Store and forward architecture
- Highly parallel design
- High speed data plane processing
  - Hyper Task Chaining: Single stream packet/cell processing problem to be decomposed into multiple, sequential tasks.
  - Memory register: Fast inter-process communication
  - Ring buffer: Flexile software pipelining

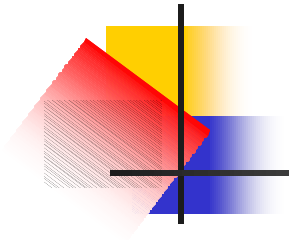
Micro Engine





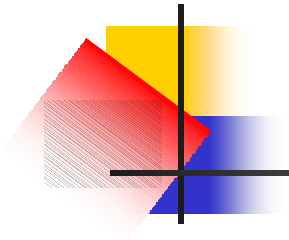
Xscale technology

- Integrated application processing in the control plane
- Managing and updating data structures
- Setting up and controlling media and switch fabric devices
- Super-pipeline technology: multi stage high efficiency processing pipeline architecture.



IXA portability famework

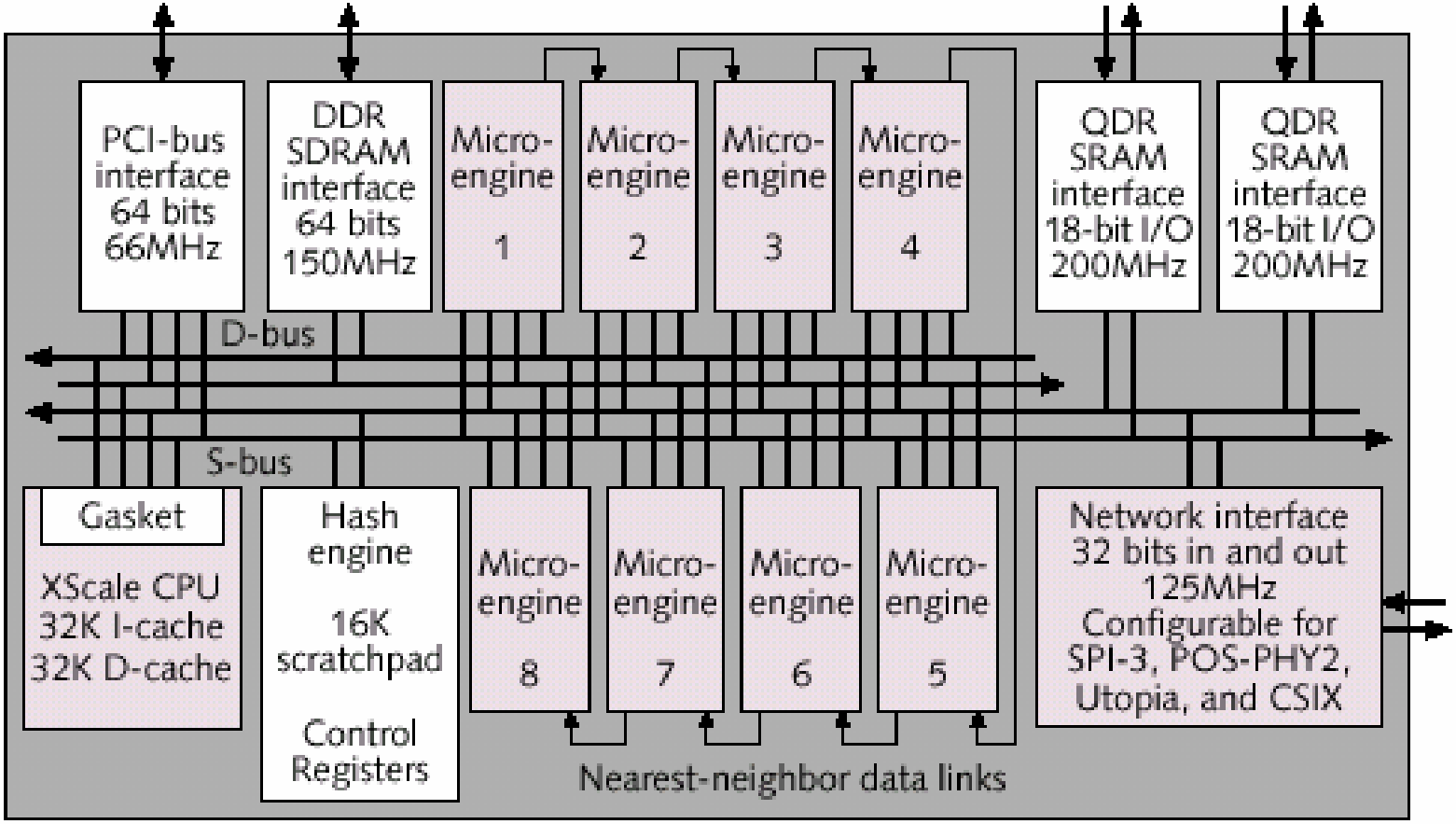
- Easy code development and reuse
- Modular programming model: optimal application partitioning across microengines and threads
- Optimized microengine libraries and tools
- Intel Xscale source code libraries: portability between multiple operation environments
- A library of standards-based NPF APIs for communication with control plane protocol stacks

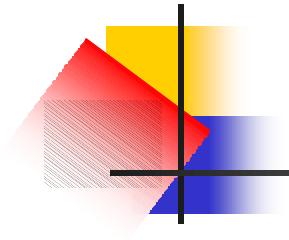


Intel IXP 2400

- From OC-12 to OC-48/2.5 Gbps Network Access and Edge Applications
- 8 fully programmable multi-threaded microengines for packet forwarding and traffic management (600MHZ)
- 5.4biilion operations per second
- Software pipelining at 2.5Gbps
- Deep packet inspection: 14million enqueue /dequeue operation per second

Intel IXP 2400





Intel IXP 2800

- For OC-192/10 Gbps Network Edge and Core Applications
- 16 programmable multi-threaded microengines for packet forwarding and traffic management(1.4GHZ)
- 23.1 giga-operations per second
- Software pipelining at 10Gbps
- Deep packet inspection: 60 million enqueue/dequeue packet operations per second

Intel IXP 2800