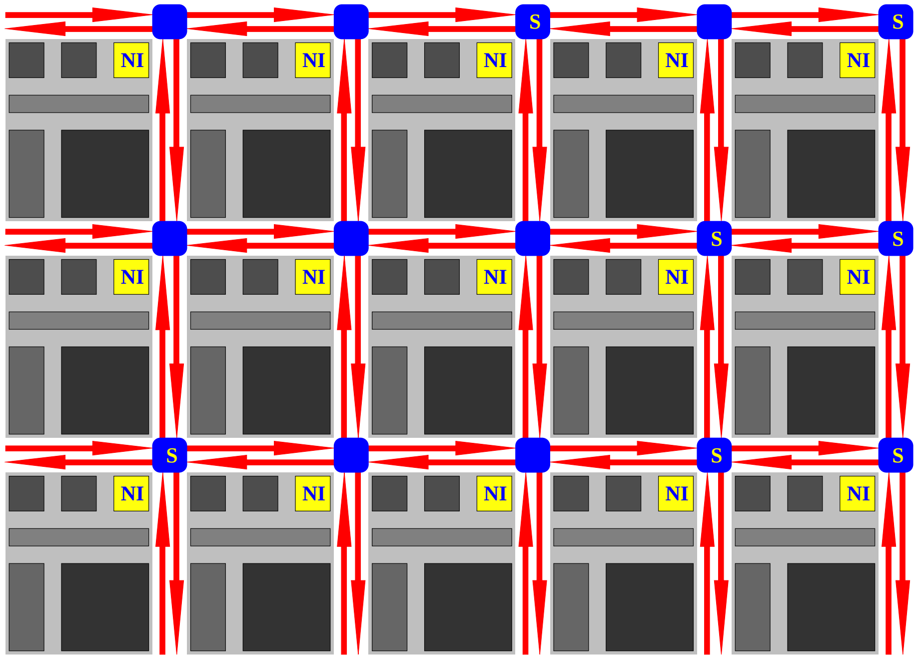
The Nostrum Network on Chip



Axel Jantsch, Zhonghai Lu, Shashi Kumar, Ahmed Hemani, Mikael Millberg, Rikard Thid, Johnny Öberg, Erland Nilsson, Xiaowen Chen, Yuang Zhang, Abdul Naeem, Sandro Penolazzi, Jean-Michel Chabloz, et al.

Royal Institute of Technology, Stockholm

June 2009

Overview

KTH Research on NoC

Topology and Structure

The Network Layer and the Switch

Quality of Service Communication

Data Management Services

NoC Research at KTH

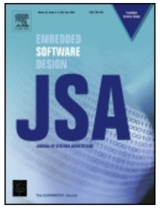
- November 2000: First papers with NoC in the title

  Ahmed Hemani, Axel Jantsch, Shashi Kumar, Adam Postula, Johnny berg, Mikael Millberg, and Dan Lindqvist. *Network on chip: An architecture for billion transistor era*. In Proceeding of the IEEE NorChip Conference, November 2000.

- September 2001: First half-day Workshop on NoC at European Solid State Circuits Conference ESSCIRC

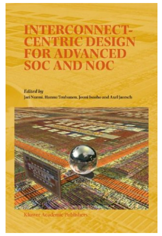
- 2003: First NoC book February 2003: *Networks on Chip*, Kluwer

- February 2004: First Special issue on NoC in the Journal of System Architecture (JSA)

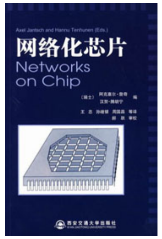


NoC Research at KTH

- April 2004: Second NoC book: *Inteconnect Centric Design for Advanced SoCs and NoCs*, ed.: Jari Nurmi, Hannu Tenhunen, Jouni Isoaho, and Axel Jantsch



- 2007: Book *Networks on Chip* translated to Chinese



- In summary:
  - ⋆ Top citation count in Google Scholar under term “Network on Chip”
  - ⋆ 40 keynotes, invited talks, tutorials on NoC
  - ⋆ 98 publications on NoC
  - ⋆ One of the pioneers and most productive groups on this topic

NoC Keynotes, Invited Talks and Tutorials

- *Networks on chip*. Presentation at the Conference RadioVetenskap och Kommunikation, June 2002.
- *Network on chip architecture*. Presentation at the EXCITE Workshop, Helsinki, November 2002.
- *Networks on chip: A paradigm change?* Presentation at the SOCWare Day, Kista, November 2002.
- *NoCs: A new contract between hardware and software*. Keynote at the Euromicro Symposium on Digital System Design, September 2003.
- *The Nostrum network on chip*. Invited presentation at ProRISC, November 2003.
- *The nostrum network on chip*. Invited seminar at Linkping University
- *The nostrum network on chip*. Invited Seminar at bo Akademi, Turku, Finland, March 2005.
- *NoC: A new contract between hardware and software?* Invited seminar at Lancaster University, October 2005.

NoC Keynotes, Invited Talks and Tutorials

- *The Nostrum network on chip*. Invited presentation at the International Symposium on System-on-Chip, Tampere, Finland, November 2005.
- *Standards for NoC: What can we gain?* Invited presentation at the Workshop on Future Interconnect and NoC, DATE, March 2006.
- Tiberius Seceleanu, Axel Jantsch, and Hannu Tenhunen. *On-chip distributed architectures*. Tutorial at the International SoC Conference, September 2006. Austin, Texas.
- *Communication performance in network-on-chips*. Short course at Tallinn Technical University, October 2006.
- *Models of computation for networks on chip*. Invited talk at the Sixth International Conference on Application of Concurrency to System Design, June 2006.
- *Network layer communication performance in networks on chip*. Tutorial at the Asian Pacitific Design Automation Conference, January 2008.

NoC Keynotes, Invited Talks and Tutorials

- *Quality of service in networks on chip*. Invited Seminar at the Research Center Telecommunciation Vienna (FTW), April 2008.
- *Resource allocation for quality of service on-chip communication*. Invited seminar at the University of Cantabria, Santander, Spain, February 2009.
- *Performance analysis and dimensioning of bandwidth and buffer capacity*. Section I of Full Day Tutorial Tutorial on Networks on Chip at the NoC Symposium 2007, May 2007.
- *NoC: State of the art, trends and challenges*. Section I of Full Day Tutorial NoC at the Age of Six: Advanced Topics, Current Challenges and Trends at DATE 2007, April 2007.

NoC Community Service

- Special issue on NoC in the Journal of System Architecture (JSA) in 2004
- OCP NOC Benchmarking Working Group, one of the initiators and main contributors, from 2006
- Steering Group of NoC Symposium since 2007
- TPC member for NoCS 2007-2009
- Co-organizer of Workshop on Diagnostic Services in Networks on Chip, 2007 (DATE), 2008 (DAC), 2009 (DATE)
- TPC co-chair for NoCS 2009
- DATE NOC Topic chair 2008, 2009
- TPC member for NoCARC 2008, 2009
- Special section in TCAD on NoC in 2010
- NOC book planned for 2010 based on EU FP7 MOSART project

NoC Projects

- NOCARC: Network-on-Chip Architecture, 2001-2004, Vinnova, Partners: Ericsson, Nokia, VTT
- NoC Design Methodology, 2001-2004, SSF
- NoC Evaluation, 2002-2005, SSF
- SPRINT, 2005-2008, EU FP6
- MOSART 2008-2010, EU FP7
- ELITE 2008-2010, EU FP7
- NoC Performance Evaluation, 2009-2011, VR

SPRINT
Open SoC Design Platform for Reuse and Integration of IPs

- EU FP6, 2005-2008
- QoS Communication, protocols and interfaces
- Partners: NXP, ARM, ST
- Main Result:
  - ⋆ Flow regulation based on Network Calculus
  - ⋆ Flow identification for QoS provision
  - ⋆ Device Level Interface (DLI) specification for QoS support
  - ⋆ ARM extends AMBA AXI protocol for QoS support based on SPRINT results

MOSART

Mapping Optimization of Multi-core Architectures

- EU FP7, 2008-2010
- Memory and Data Management for MultiCore NoCs (McNoC)
- Power management and clocking
- Partners: Thales, CoWare, IMEC, VTT, ICCS, Arteris
- Main Results so far:
  - ⋆ Date Management Engine, Patent under preparation
    - ∗ Distributed Shared Memory support
    - ∗ Cache coherence
    - ∗ Memory consistency
    - ∗ Dynamic memory allocation and ADT support
  - ⋆ Hierarchical power management archtiecture
  - ⋆ Globally Ratio-synchronous Locally Synchronous clocking scheme (GRLS)

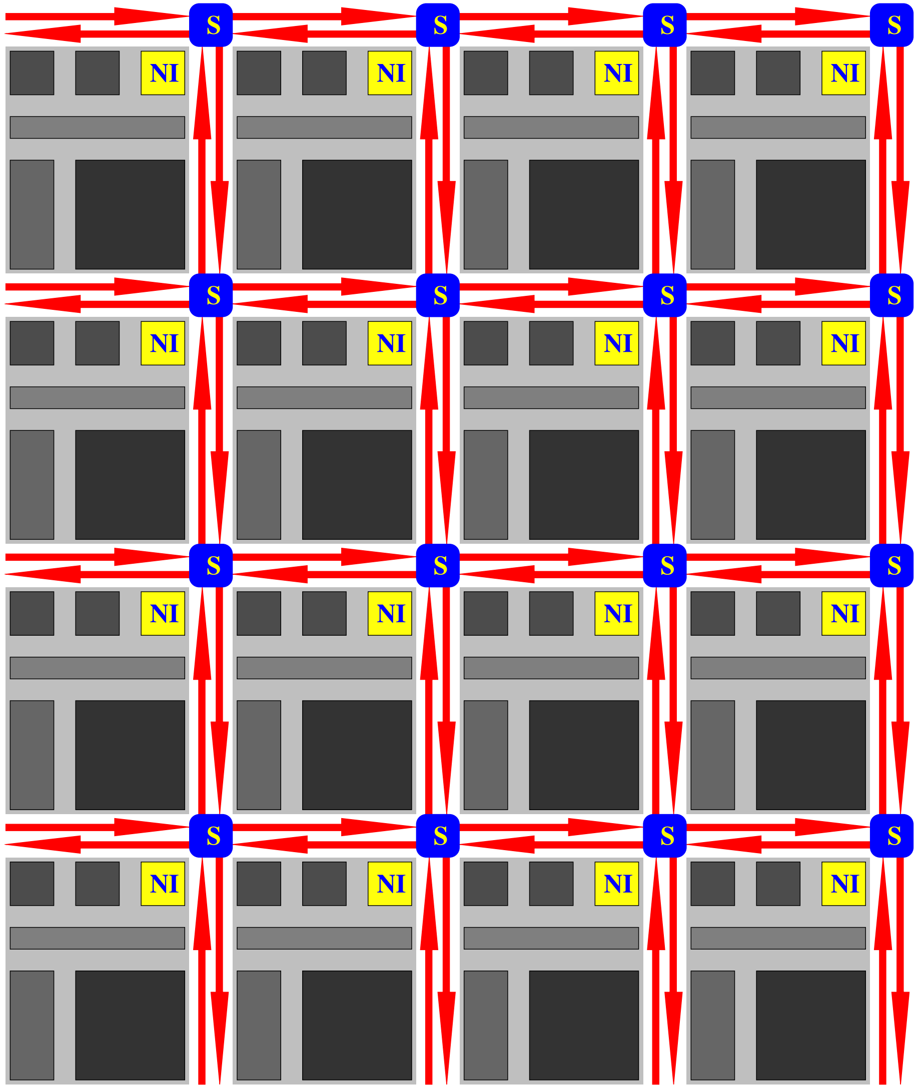
ELITE
Extended Large (3-D) Integration Technology

- EU FP7, 2007-2010
- 3D Network and Memory Architecture
- Partners: CEA LETI, Lancaster University, Hyperstone, Numonyx
- Main Results so far:
  - ⋆ 3D Router design
  - ⋆ 3D Architecture and Design space exploration

Current Group Activities

- 4 Faculty, 10 PhD students
- NoC PCB emulation platform
- Memory and Data management
- Performance Analysis
- Resource Allocation and Dimensioning for QoS
- Power Management
- Clocking and Synchronization
- 3D Architectures

Nostrum Topology: Mesh



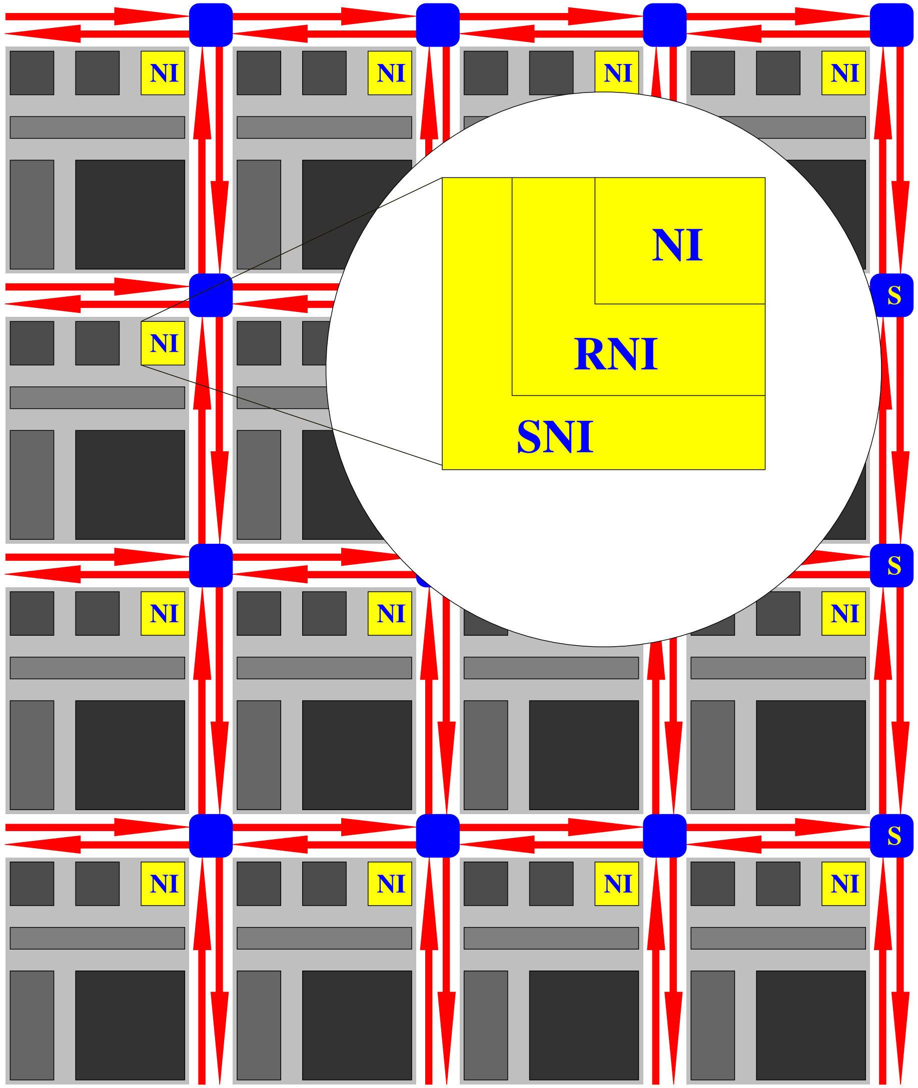
Characteristics:

- Resource-to-switch ratio: 1
- A switch is connected to 4 switches and 1 resource
- A resource is connected to 1 switch
- Average distance: $2/3n$
- Bisection bandwidth: $2n$

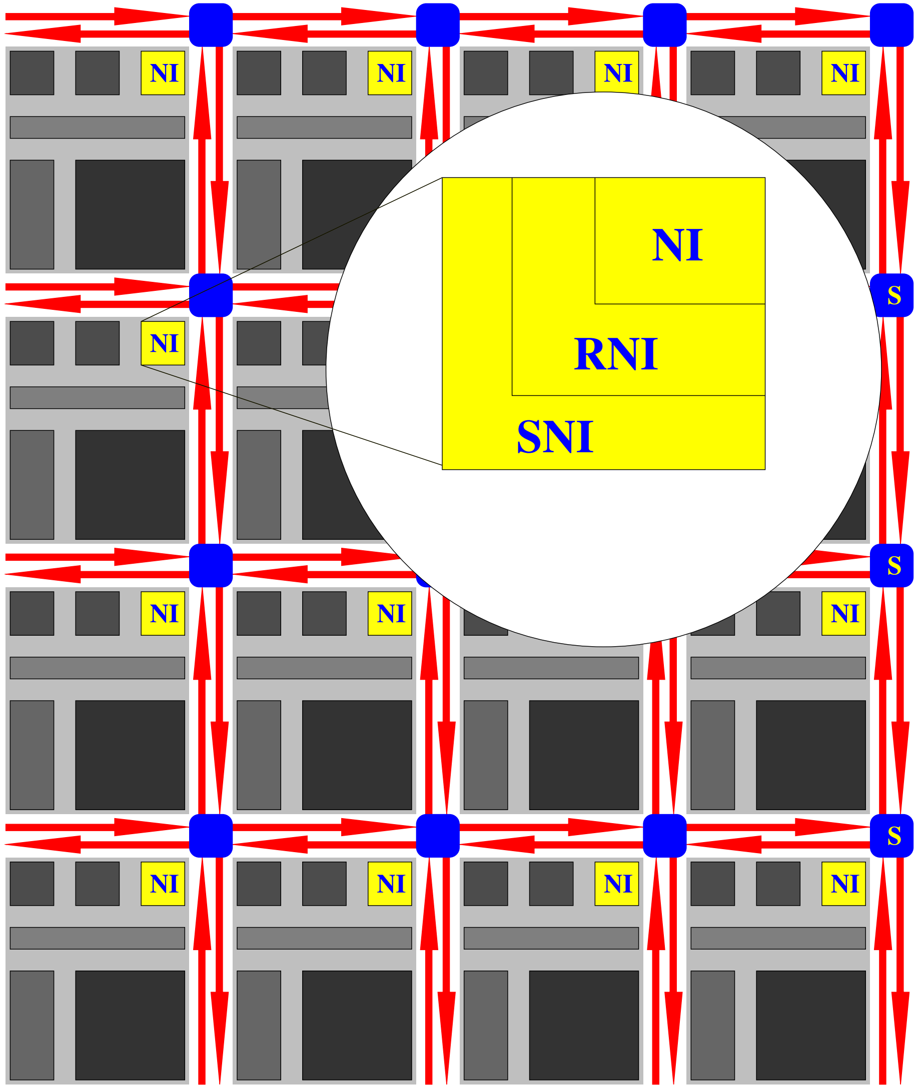
Motivation:

- Regularity of layout; predictable electrical properties
- Expected locality of traffic

The Node in a Mesh



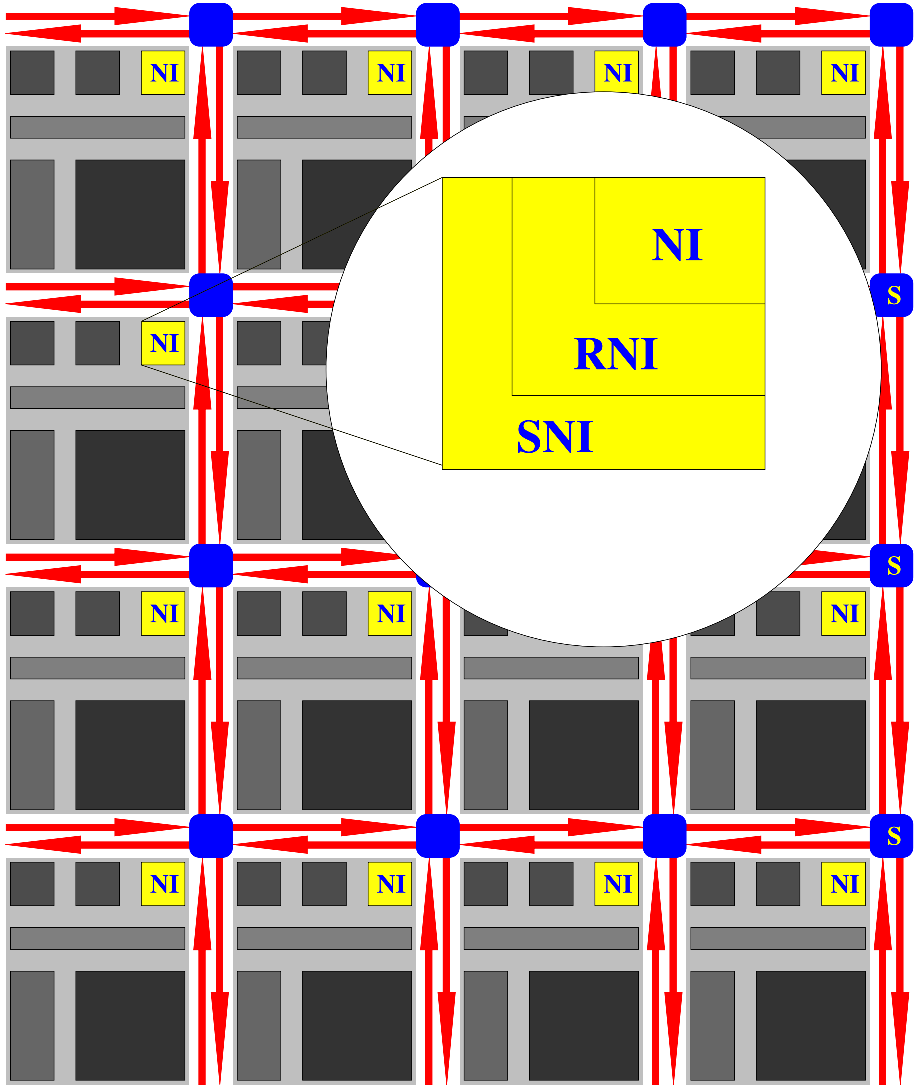
The Node in a Mesh



NI: Network Interface:

- Compulsory
- Hardware
- Implements the network layer protocol

The Node in a Mesh



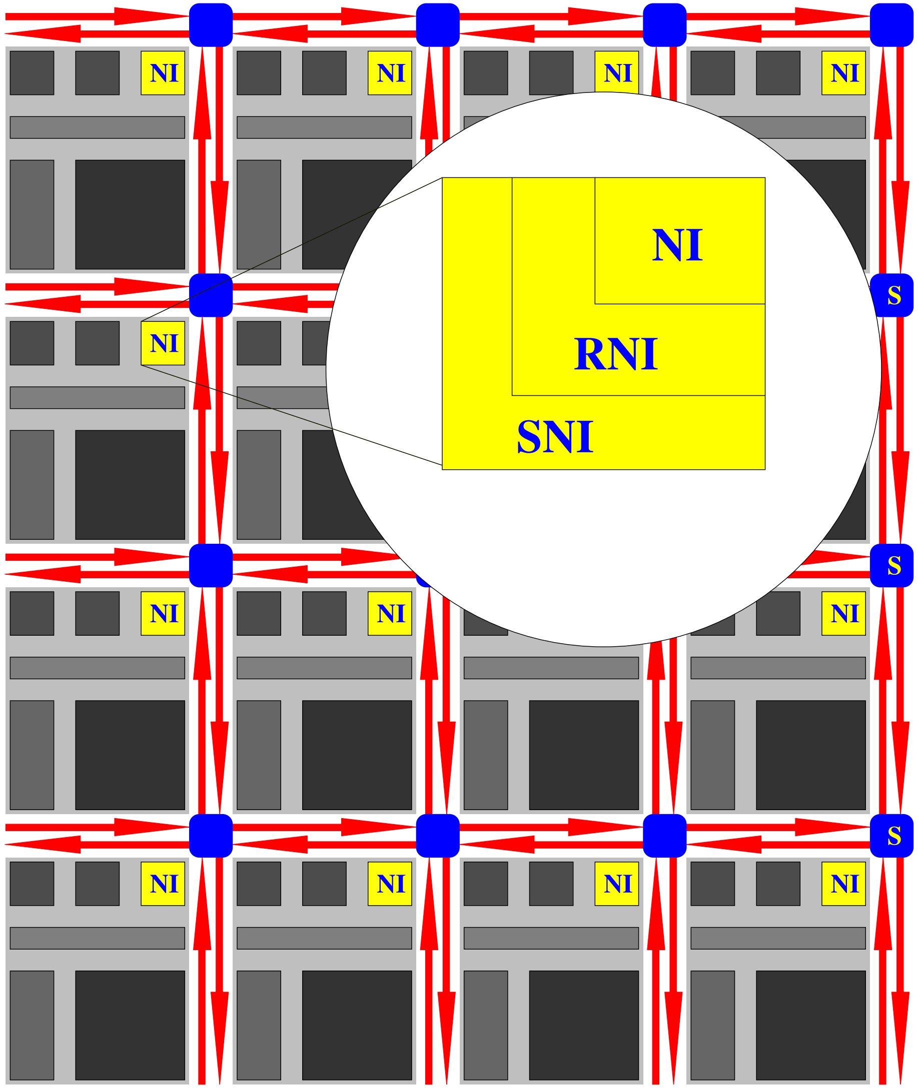
NI: Network Interface:

- Compulsory
- Hardware
- Implements the network layer protocol

RNI: Resource Network Interface:

- Optional
- Hardware and/or Software
- Implements transport layer
- Provides resource specific interfaces

The Node in a Mesh



NI: Network Interface:

- Compulsory
- Hardware
- Implements the network layer protocol

RNI: Resource Network Interface:

- Optional
- Hardware and/or Software
- Implements transport layer
- Provides resource specific interfaces

SLI: Session Layer Interface:

- Optional
- Hardware and/or software
- Implements the session layer protocol

Overview

Topology and Structure

The Network Layer and the Switch

Quality of Service Communication

Data Management Services

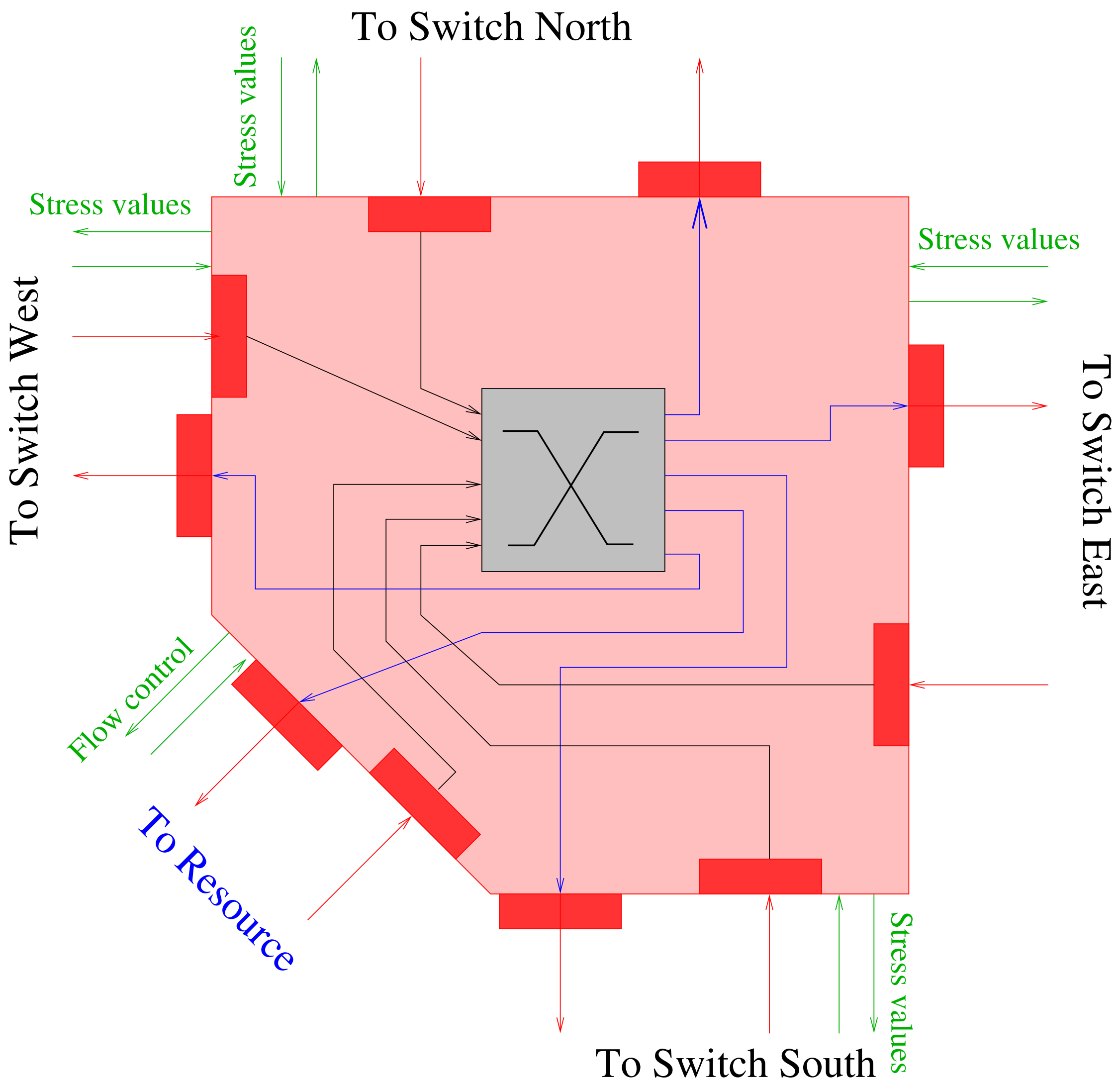
The Network Layer

- Packet switched best effort service
  - ⋆ Packets are guaranteed to arrive
  - ⋆ Packet payload may be protected (4 levels of protection)
  - ⋆ Load dependable delay in the network
  - ⋆ Load dependable delay at the network access point
  - ⋆ Admission policy for best effort traffic:
    - ∗ Network load should be below 60%
    - ∗ Load is measured locally in switch and based on neighboring stress values

The Network Layer

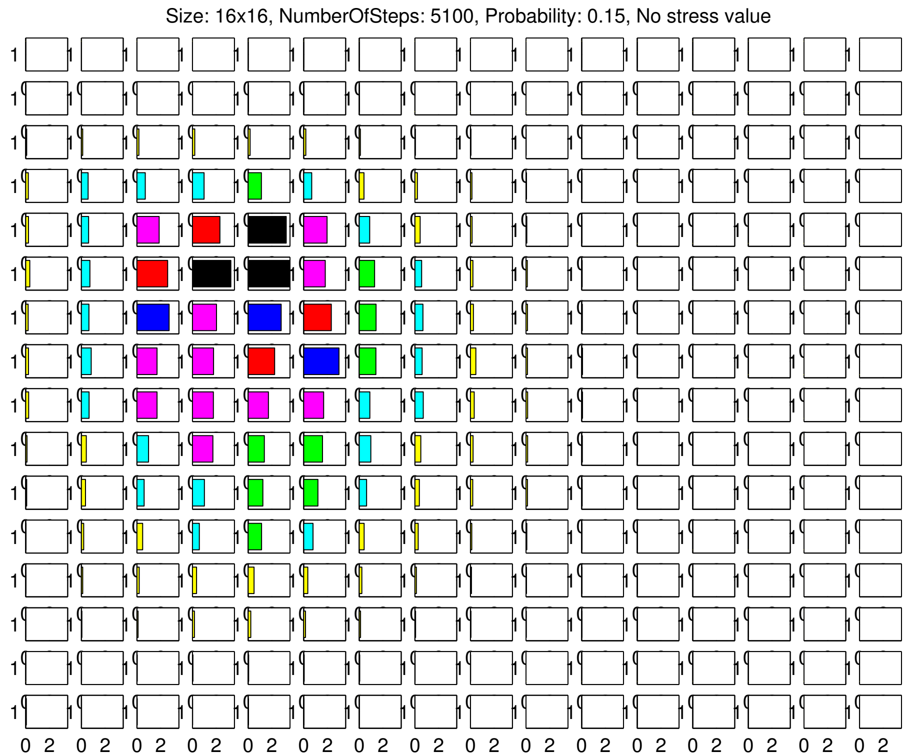
- Packet switched best effort service
  - ⋆ Packets are guaranteed to arrive
  - ⋆ Packet payload may be protected (4 levels of protection)
  - ⋆ Load dependable delay in the network
  - ⋆ Load dependable delay at the network access point
  - ⋆ Admission policy for best effort traffic:
    - ∗ Network load should be below 60%
    - ∗ Load is measured locally in switch and based on neighboring stress values
- Virtual circuit service
  - ⋆ Guaranteed bandwidth
  - ⋆ Guaranteed maximum delay
  - ⋆ Multicast circuits
  - ⋆ Static and semi-static virtual circuits
  - ⋆ Based on packet switching service

The Bufferless Switch

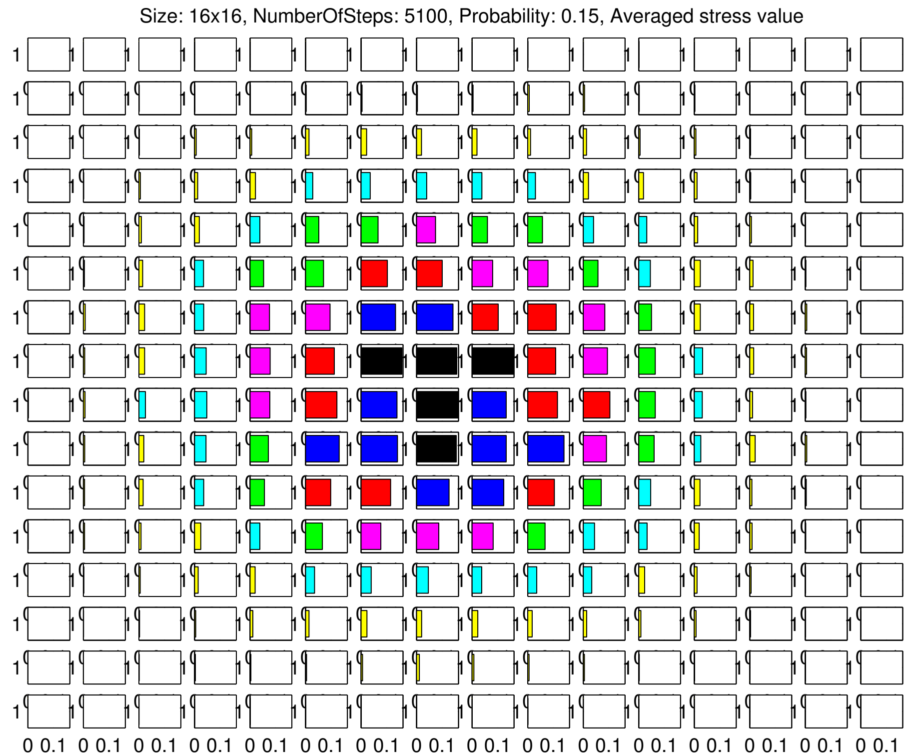


- + No buffers
- + No routing table
- + Small area
- + Short delay
- + Low power consumption
- – Non-shortest path
- – Header overhead due to destination address

Stress Value Effect on Buffer Sizes and Delays

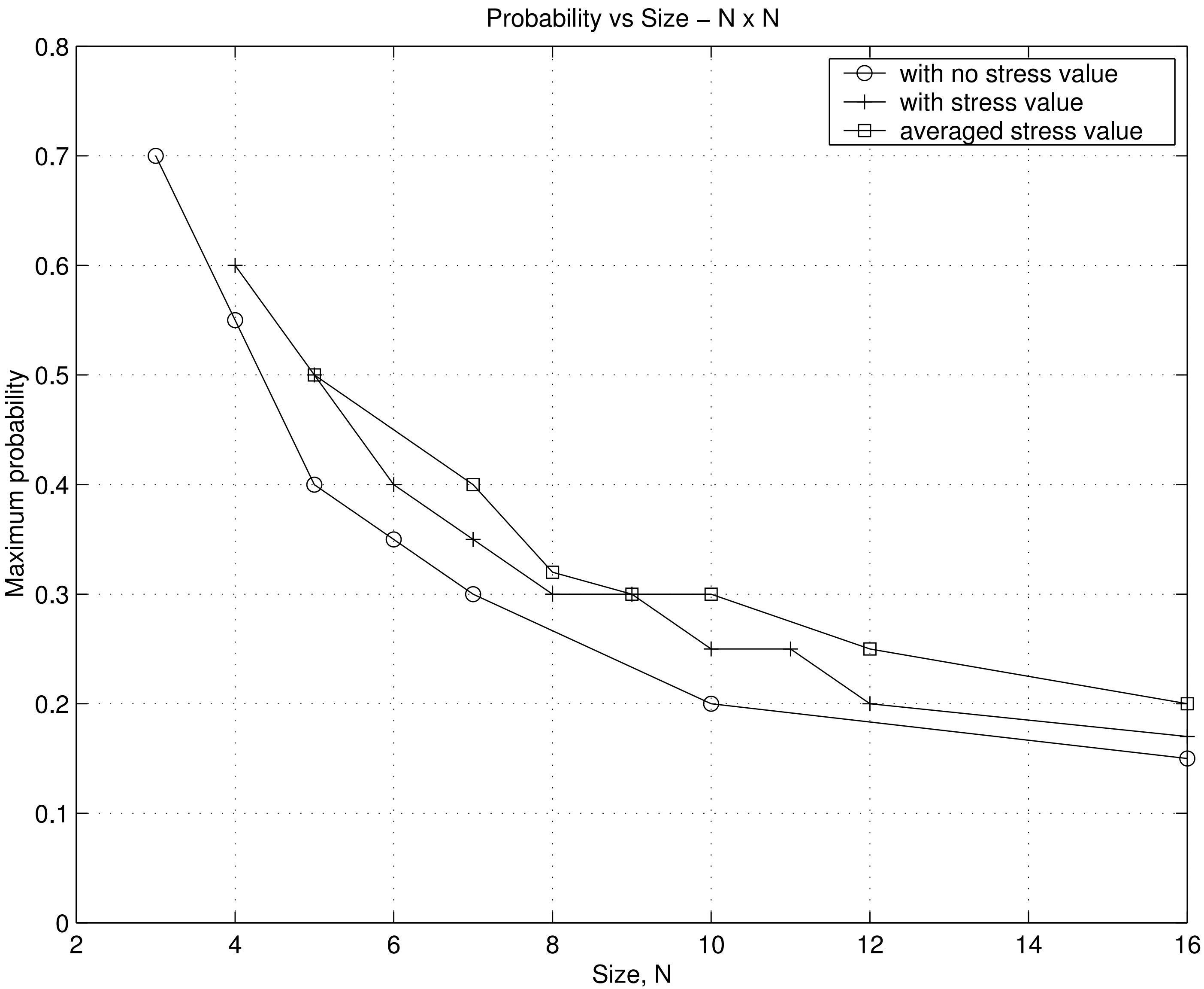


No stress value control
Largest average buffer size: 3.2 (black)



Averaged stress value control
Largest average buffer size: 0.2 (black)

Stress Value Effect on Maximum Load

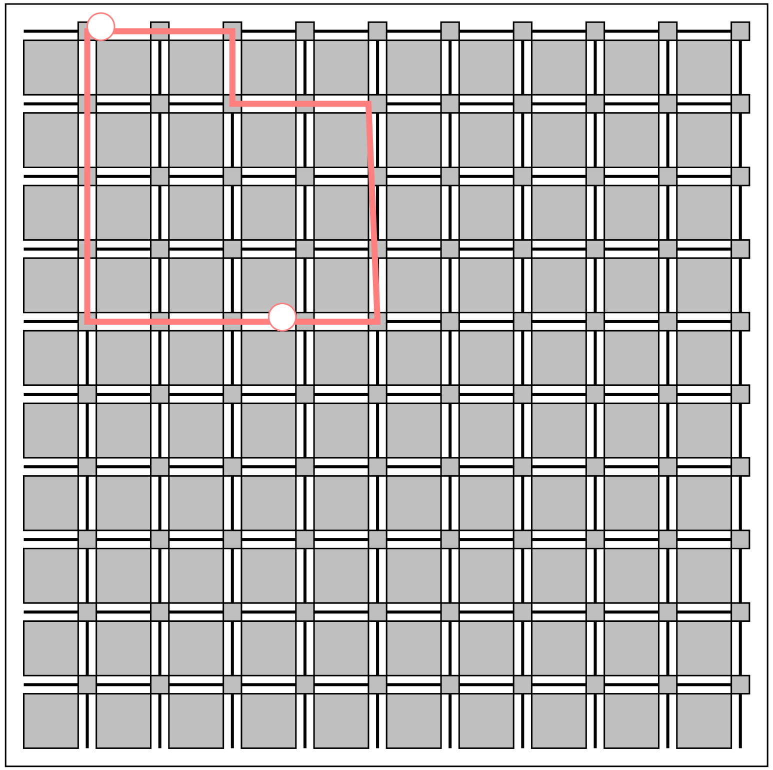


Looped Container based Virtual Circuit

- A container packet loops between two or more end points
- The looping container establishes a closed virtual circuit
- The virtual circuit allows multicast and bus protocol emulation
- Possible bandwidth allocation:

$$2^{j-d}B$$

  where B = link bandwidth, d = length of the container loop, $1 \leq j \leq d$
- Examples:
  $d = 2$: possible allocations: 100% and 50%
  $d = 4$: possible allocations: 100%, 50%, 25%, 12.5%



Implementation of Static Virtual Circuits

- Bandwidth allocation and circuit setup at design time
- Implementation alternatives:
  - ⋆ Channel containers have higher priority
  - ⋆ Look-up tables in switches
- Semi-static circuits:
  - ⋆ Active circuits: Circulating containers
  - ⋆ Inactive circuits: Containers removed
  - ⋆ Activation of circuits subject to traffic load dependent delay
  - ⋆ NI can increase stress value to activate virtual circuits

Overview

Topology and Structure

The Network Layer and the Switch

Quality of Service Communication

Data Management Services

QoS Communication

- Virtual Channel TDM based Service with guaranteed bandwidths and latency
- Virtual Channel theory and configuration method
  - ⋆ Given: Set of communication flows and requirements
  - ⋆ Result: Set of virtual channels (paths, slots) assigned to flows
- Flow regulation and resource allocation

TDM Virtual Channel Configuration Method

Given a set of VC specifications (source, destination, minimum bandwidth), determine and implement the necessary VCs.

- Path selection: For each VC determine the sequence of nodes between source and destination.
- Slot allocation: For each VC determine the allocated time slots for each buffer in the VC.

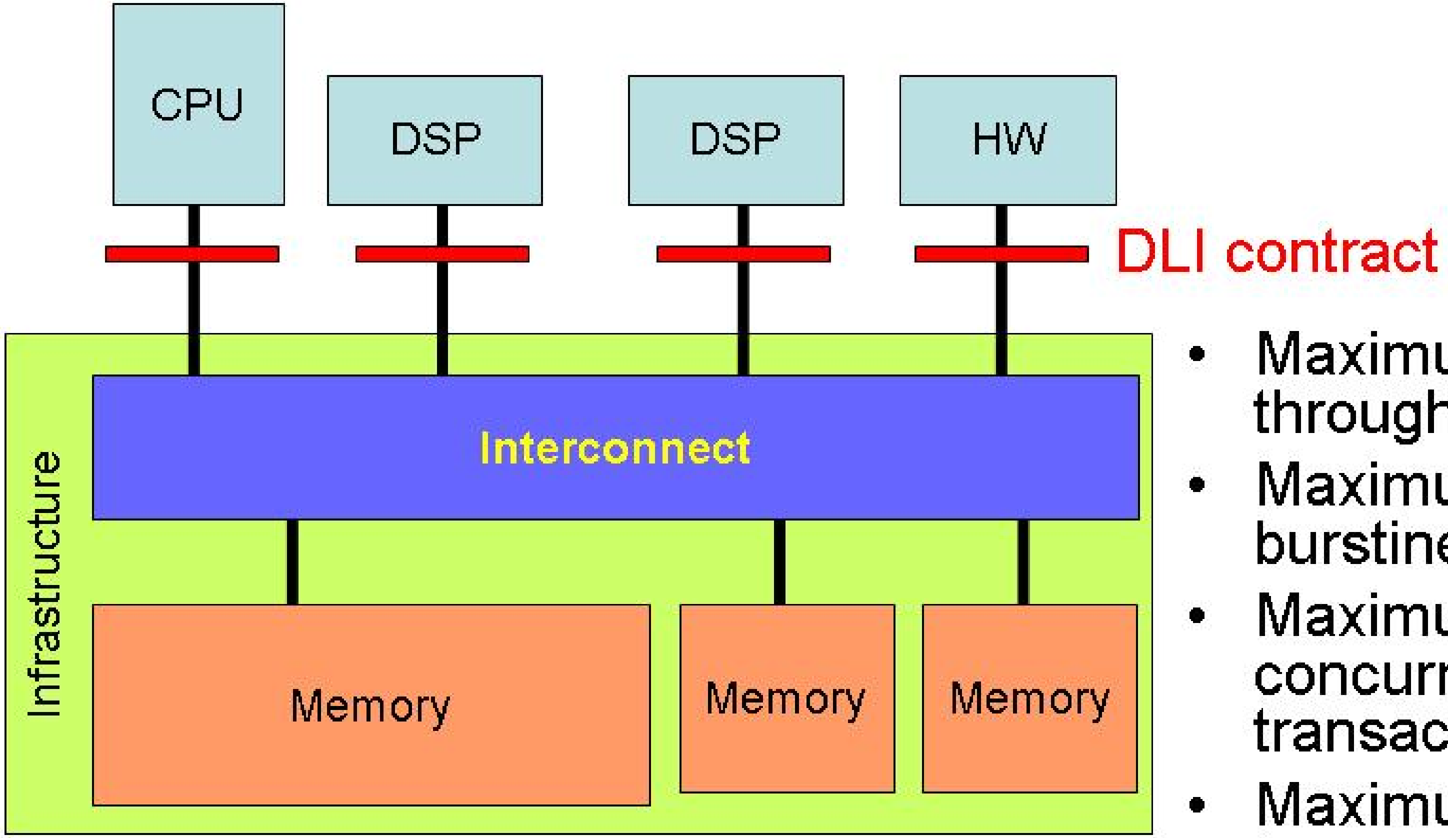
Conditions to be satisfied:

- All VCs are contention free;
- All VCs allocated sufficient number of slots to provide the require bandwidth;
- The network must be deadlock free and livelock free;
- Sufficient bandwidth must remain for best effort traffic;

Flow Regulation and Interface Contracts

- Performance contracts between
  - ⋆ IP/Application
  - ⋆ Network/infrastructure
- Allows for modular system performance analysis
- Allows for infrastructure dimensioning

DLI Contracts



- Maximum throughput
- Maximum burstiness
- Maximum concurrent transactions
- Maximum transaction delay
- Maximum transaction size

DLI Performance Contract

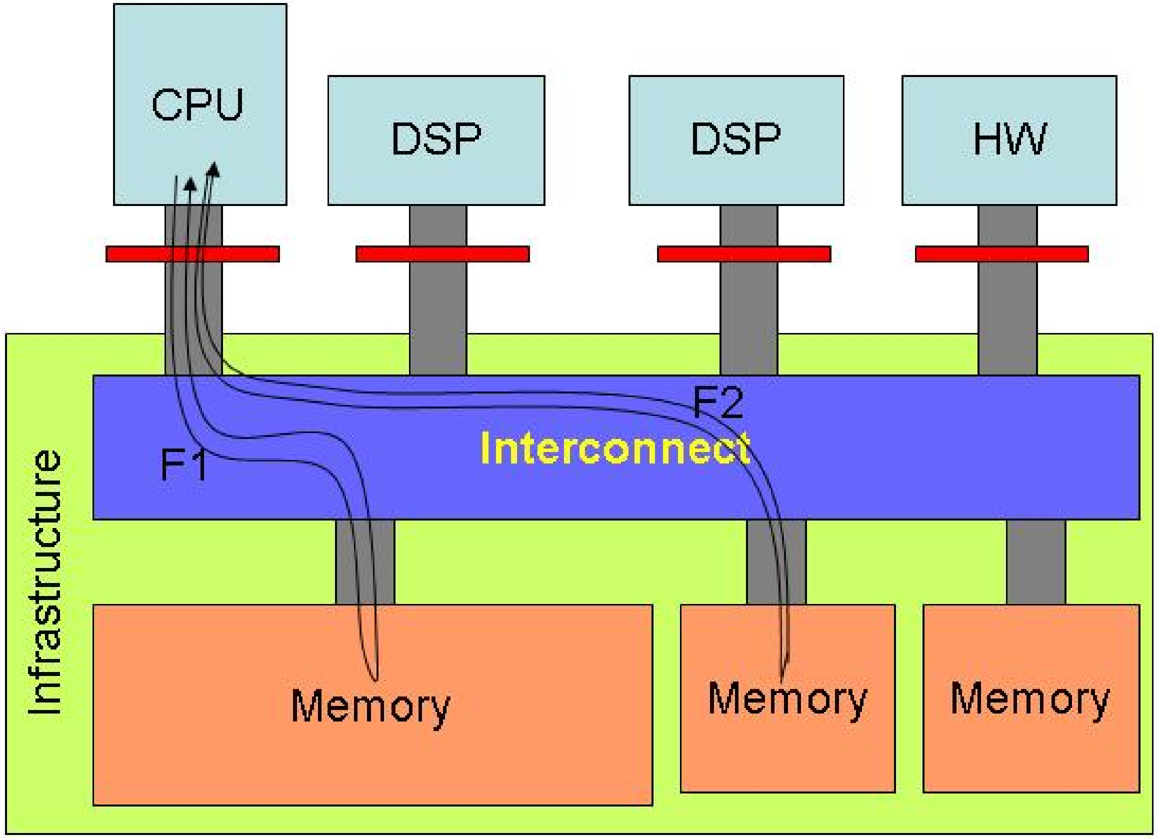
- Complements the functional DLI protocol
- Guarantees infrastructure performance to IPs and to application tasks
- Limits demands on infrastructure
- Allows for analytical infrastructure dimensioning
- Allows for analytical system performance analysis
- Allows for structured renegotiation of resource allocation

DLI Performance Contract

- IP Based Contract

or

- Flow based contract



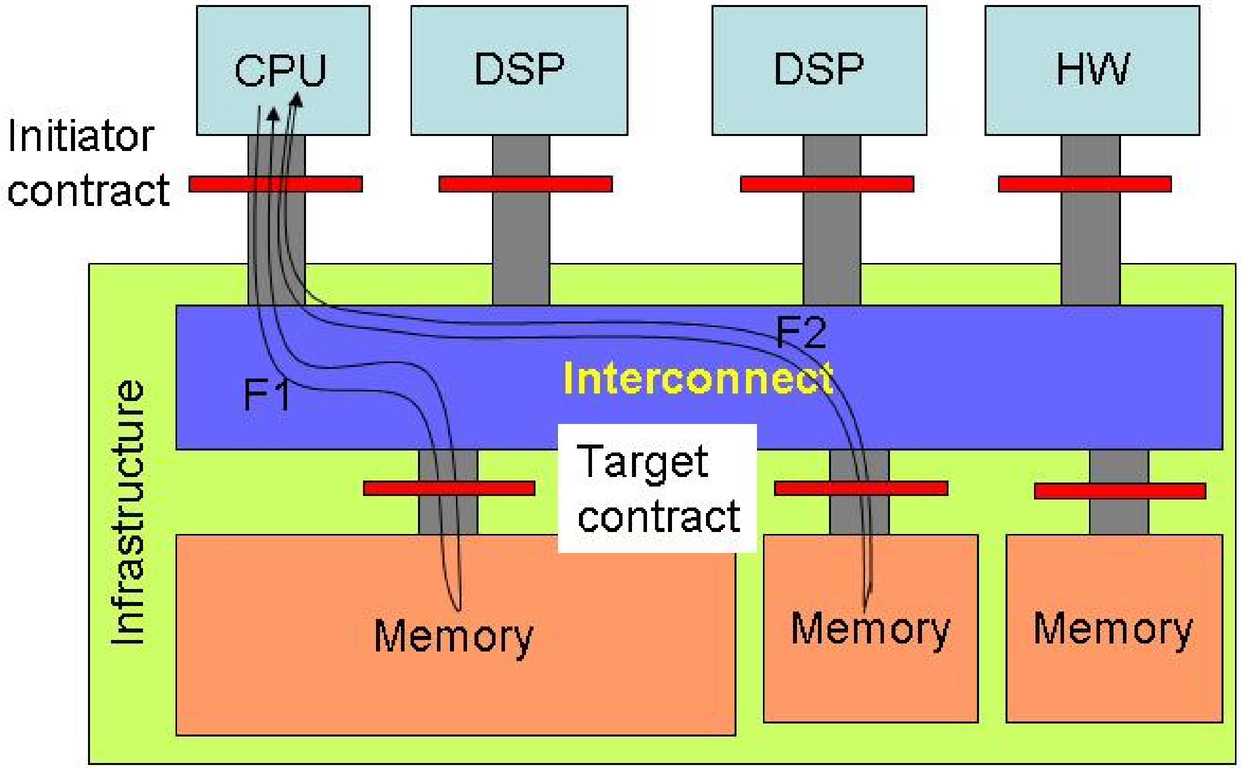
Flow based Initiator Contract

- Read contract
  - Flow ID
  - Request flow: (σ_{AR}, ρ_{AR}): max burstiness and throughput
  - Data flow: (σ_{R}, ρ_{R}): max burstiness and throughput
  - Degree: Max number of concurrent transactions
  - Delay: max delay of all transactions within a time window
  - Period: Size of sliding time window for delay constraint
  - Size: max size of transaction
- Write contract
  - Flow ID
  - Request flow: (σ_{AW}, ρ_{AW}): max burstiness and throughput
  - Data flow: (σ_{W}, ρ_{W}): max burstiness and throughput
  - Acknowledgment flow: (σ_{B}, ρ_{B}): max burstiness and throughput
  - Degree: Max number of concurrent transactions
  - Delay: max delay of all transactions within a time window
  - Period: Size of sliding time window for delay constraint
  - Size: max size of transaction

Flow Based Target Contract

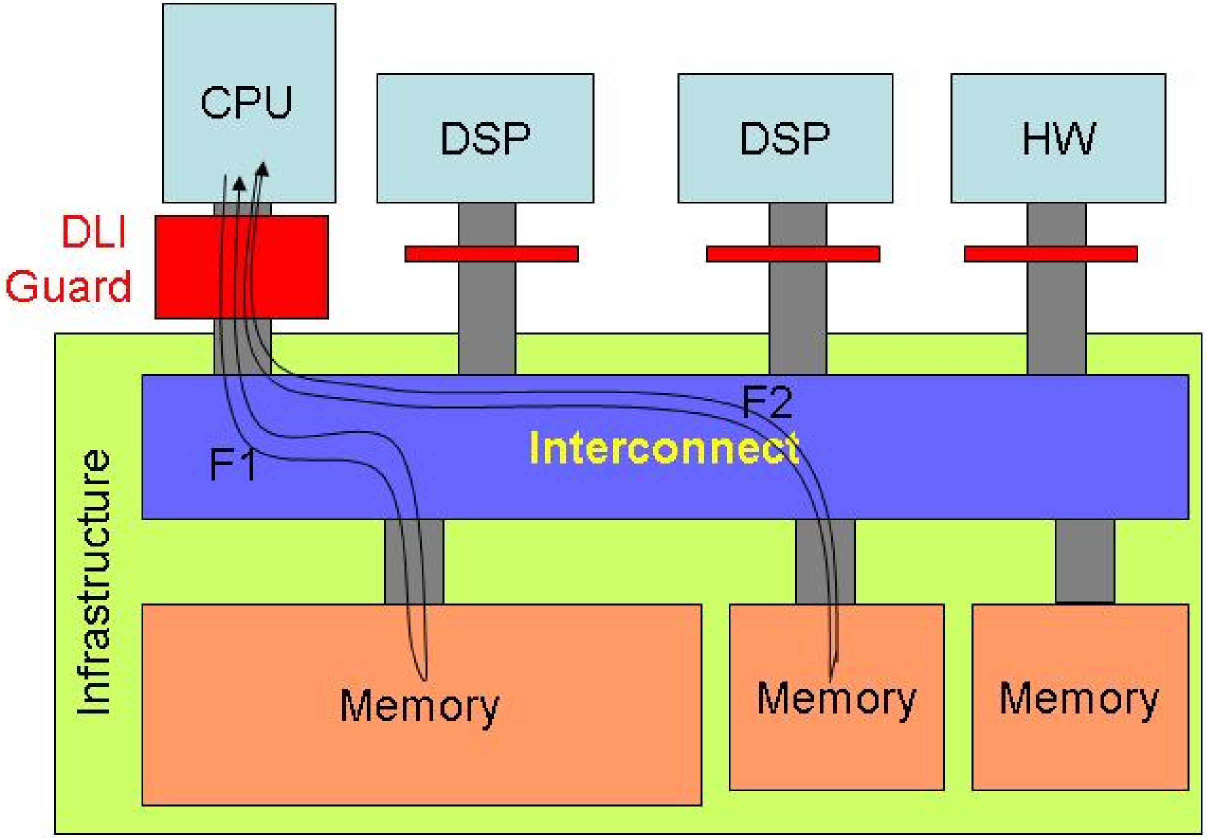
Read and write contracts constrain

- The flow to the target IP
- The response time of the target IP

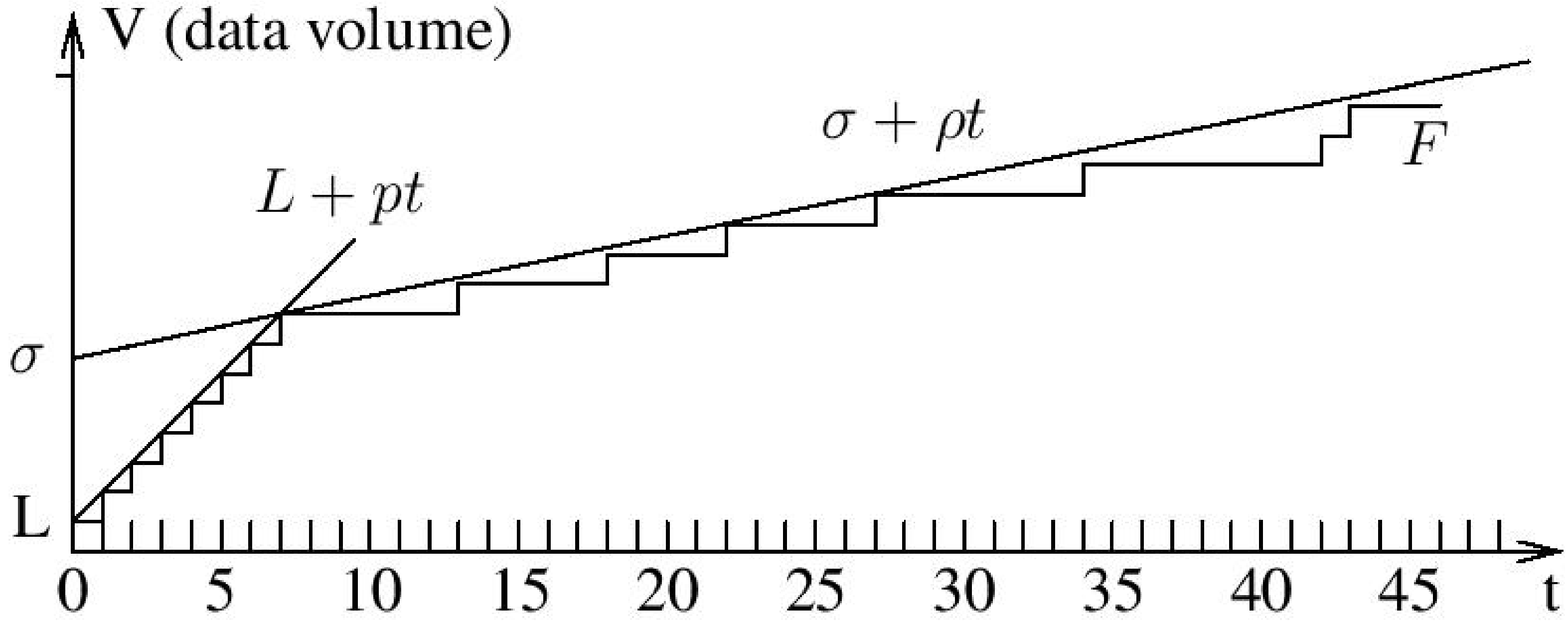


The Effect of Contracts: Traffic Shaping

- DLI Guard enforces a contract by
- Monitoring delays
- Shaping flows

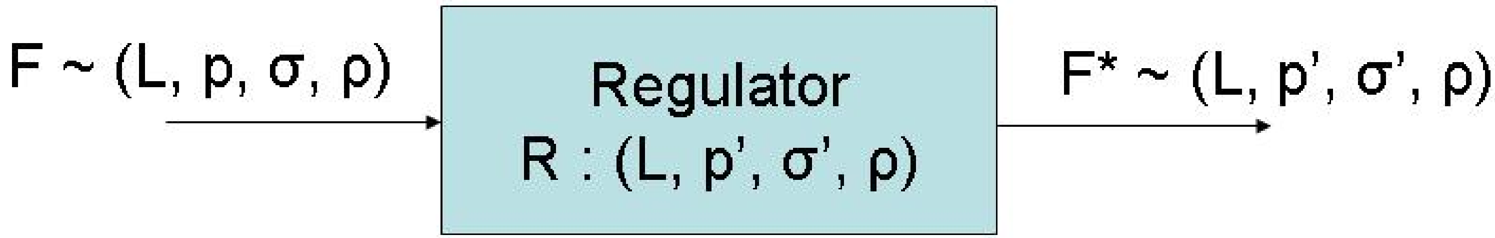


TSPEC Flow Characterization



- TSPEC(L,p,σ,ρ)
  - L: transfer size
  - p: peak rate of link
  - σ: burstiness (σ ≥ L)
  - ρ: average rate (p ≥ ρ)

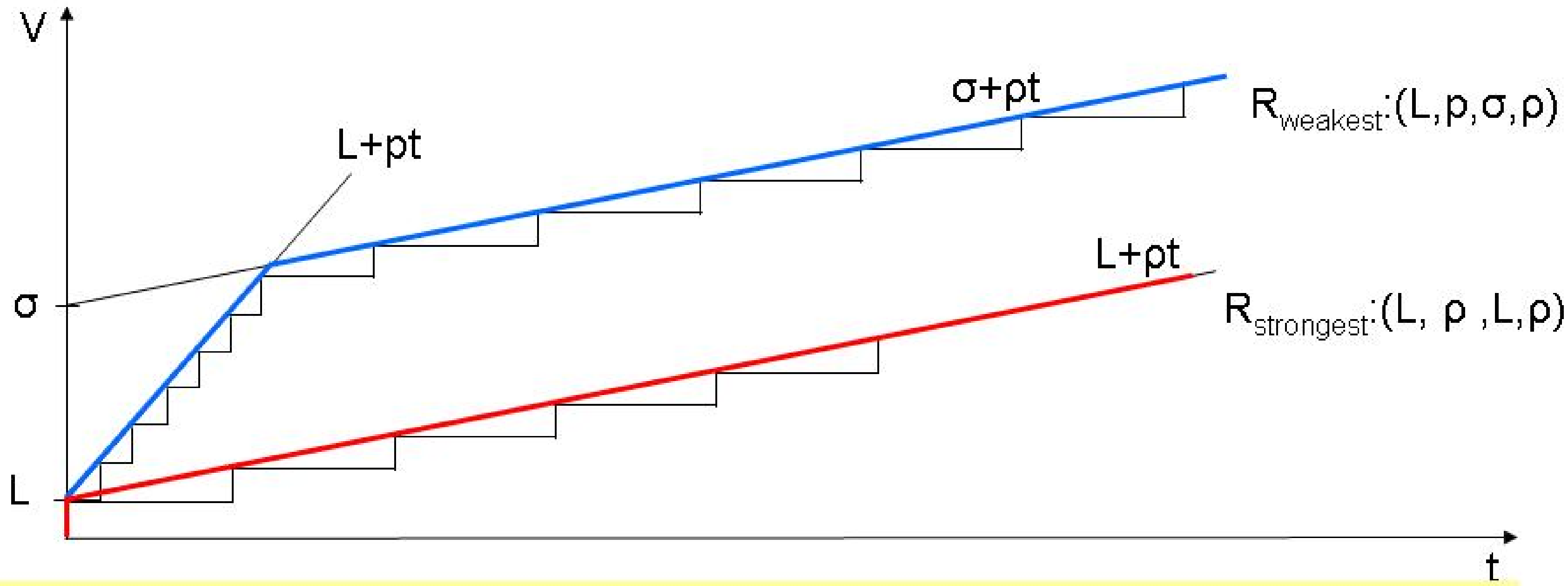
Flow Regulator



- The regulator incurs
  - Regulation Delay $B_{reg} = \Delta\sigma$
  - Regulation Buffering $D_{reg} = \Delta\sigma / \rho$

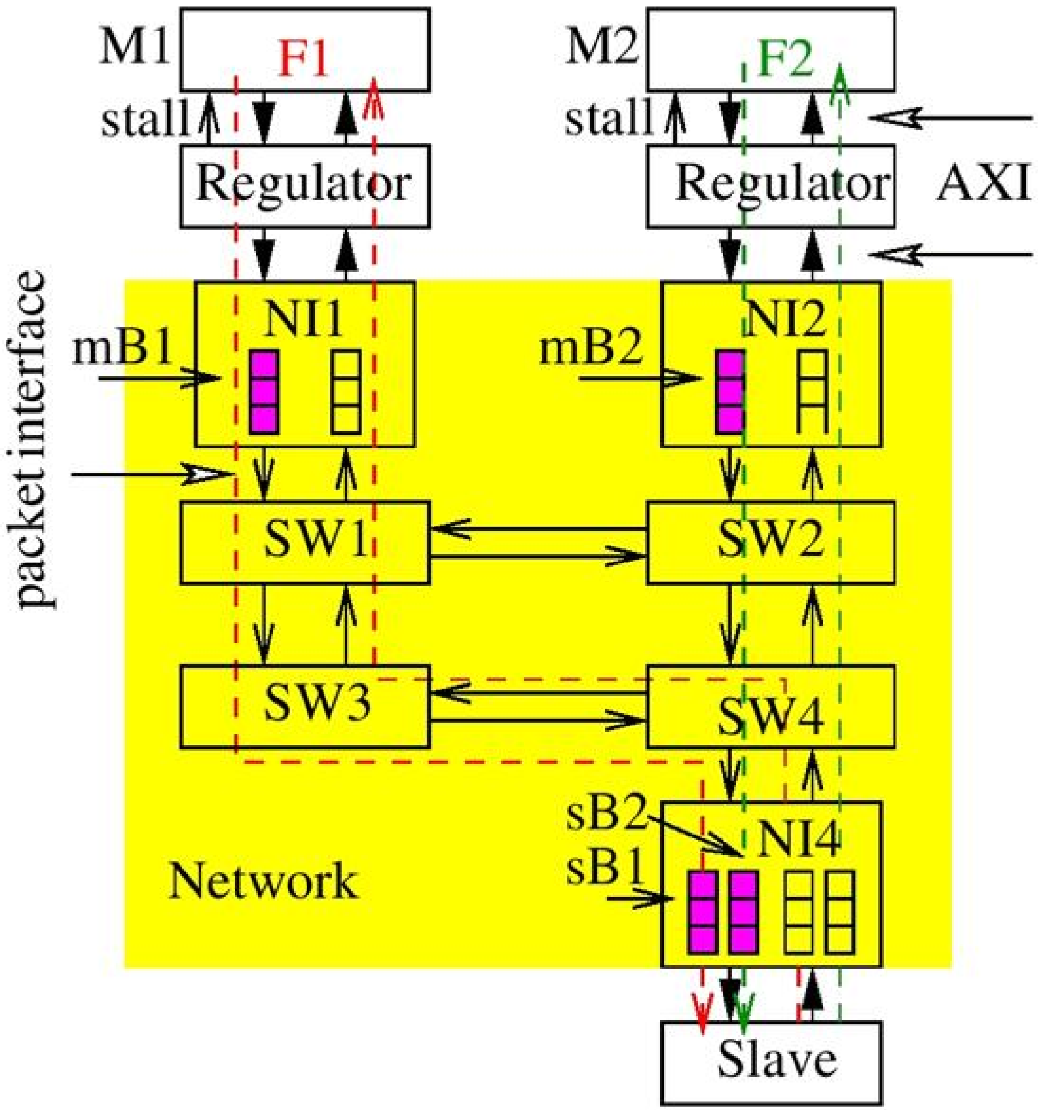
Regulation Spectrum

- Input flow: F ~ (L, p, σ, ρ)
- Regulator: R : (L, p’, σ’, ρ)
- Regulation spectrum: ρ’ ≤ p’ ≤ p; L ≤ σ’ ≤ σ



- Regulation spectrum determines design space by smoothing bursts

Regulation Effect Example



Example
Simulation - Delay and Backlog

$D_{transaction}$ Backlog	No. Reg. $(1, 1, 14.4, 0.1)$		Reg. F_1 $(1, 1, 3, 0.1)$		Reg. F_1 $(1, 1, 1, 0.1)$	
Flow	F_1	F_2	F_1	F_2	F_1	F_2
D_{data}	138	134	150	126	169	114
D_{slave}	1	1	1	1	1	1
D_{ack}	9	4	9	4	9	4
$D_{transaction}$	148	139	160	131	179	119
B_{mB}	13	13	3	13	1	13
B_{sB}	9	9	2	9	1	7
B_{total}	22	22	5	22	2	20

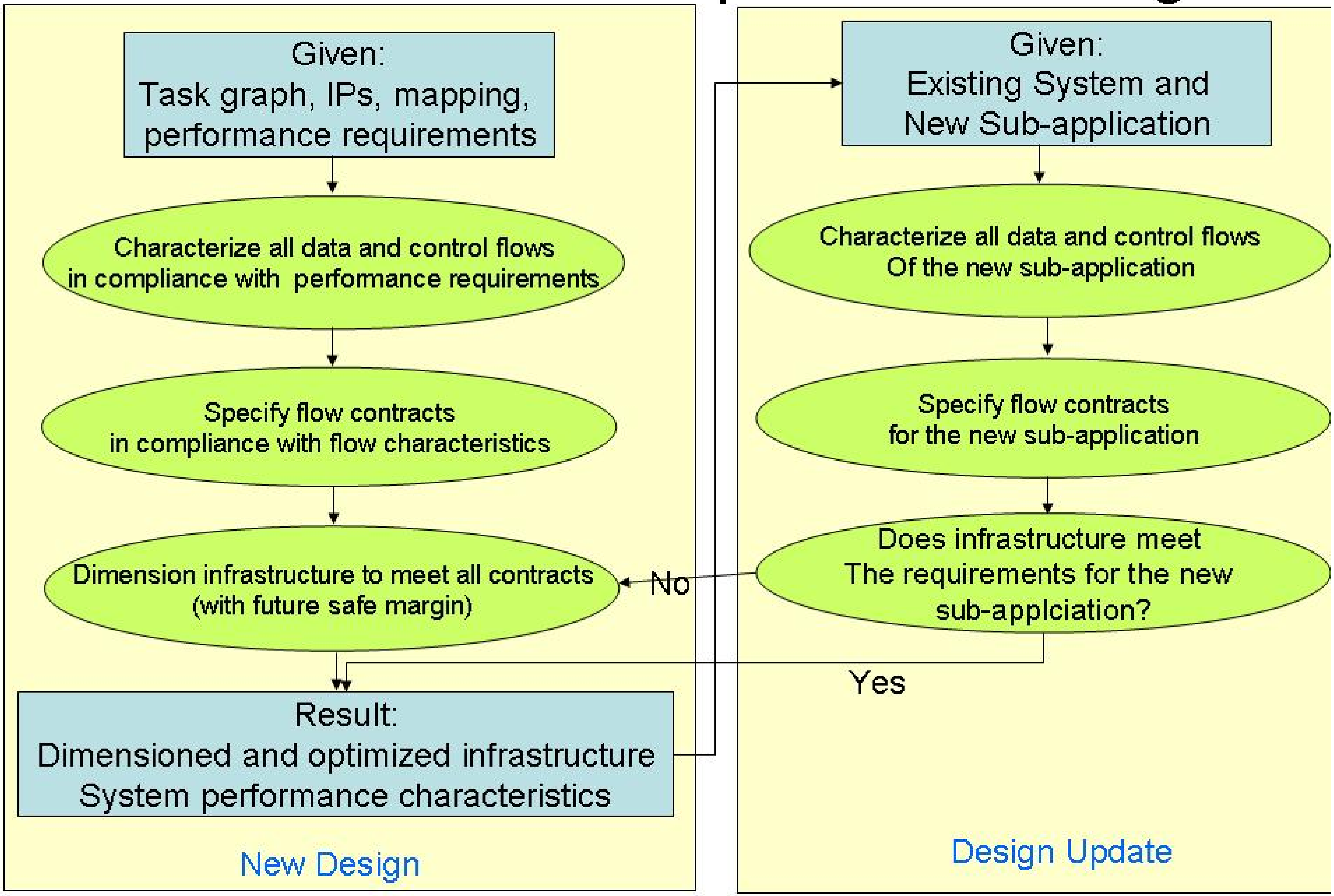
Example
Delay – Simulation vs Analysis

$D_{transfer}$	**No. Reg.** $(1, 1, 14.4, 0.1)$		**Reg.** F_1 $(1, 1, 3, 0.1)$		**Reg.** F_1 $(1, 0.1, 1, 0.1)$	
Flow	F_1	F_2	F_1	F_2	F_1	F_2
NC	124	122	35	122	20	122
SM	122	118	31	110	20	98
D_{reg}	0	0	114	0	134	0

Example
Backlog – Simulation vs Analysis

Backlog		**No. Reg.** $(1, 1, 14.4, 0.1)$		**Reg.** F_1 $(1, 1, 3, 0.1)$		**Reg.** F_1 $(1, 0.1, 1, 0.1)$	
Flow		F_1	F_2	F_1	F_2	F_1	F_2
B_{mB}	NC	12.9	12.9	3.3	12.9	1.3	12.9
	SM	13	13	3	13	1	13
B_{sB}	NC	9	9	2.5	9	1.4	9
	SM	9	9	2	9	1	7
B_{reg}		0	0	12	0	14	0
B_{total}		22	22	17	22	16	20

Contract Based Compositional Design



Summary - Contracted Flows

- Compositional design based on standard protocols and performance contracts
- Formulation of flow contracts for DLIs
- Potential of flow regulation
  - ⋆ Smoothin bursts
  - ⋆ Reducing buffers
  - ⋆ Controlling resource allocation for flows
- Contract based design flow

Overview

Topology and Structure

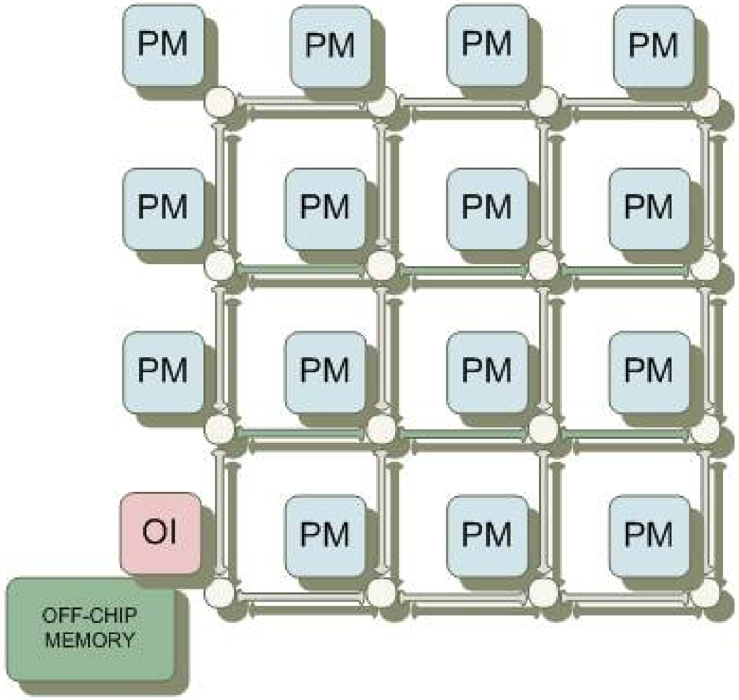
The Network Layer and the Switch

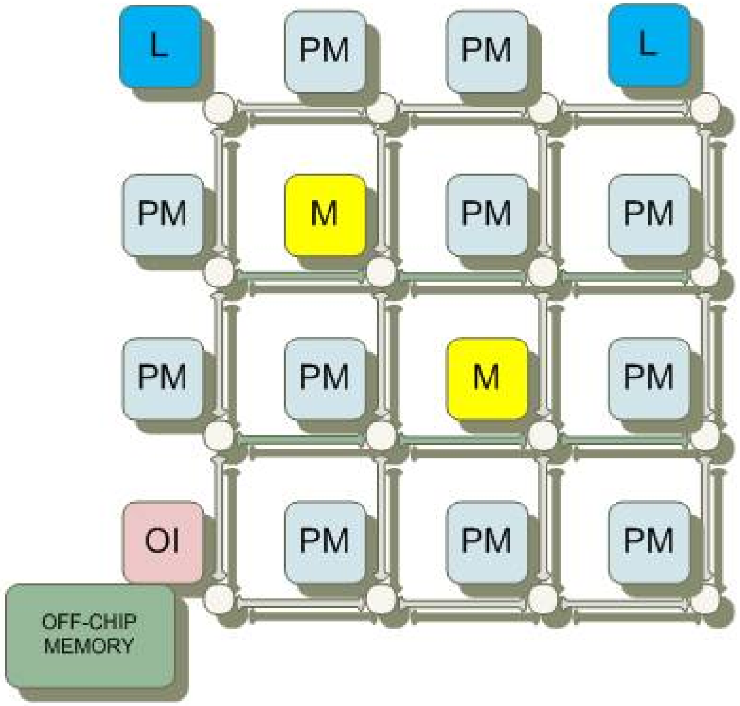
Quality of Service Communication

Data manegement Services

Homogeneous and Heterogeneous Multi-Core NoCs

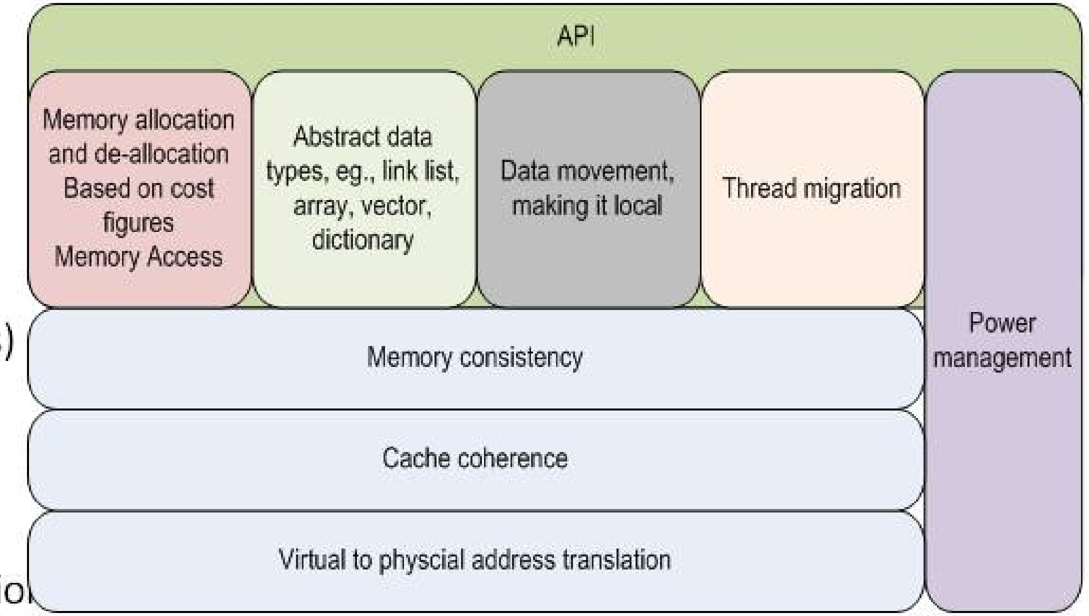
- Homogeneous and heterogenous nodes
- Hybrid physical and virtual addressing
  - Local memory divided into private and shared parts.
  - Physical addressing for local private regions
  - Logical addressing for shared regions





Multi-Core NoC Platform Services

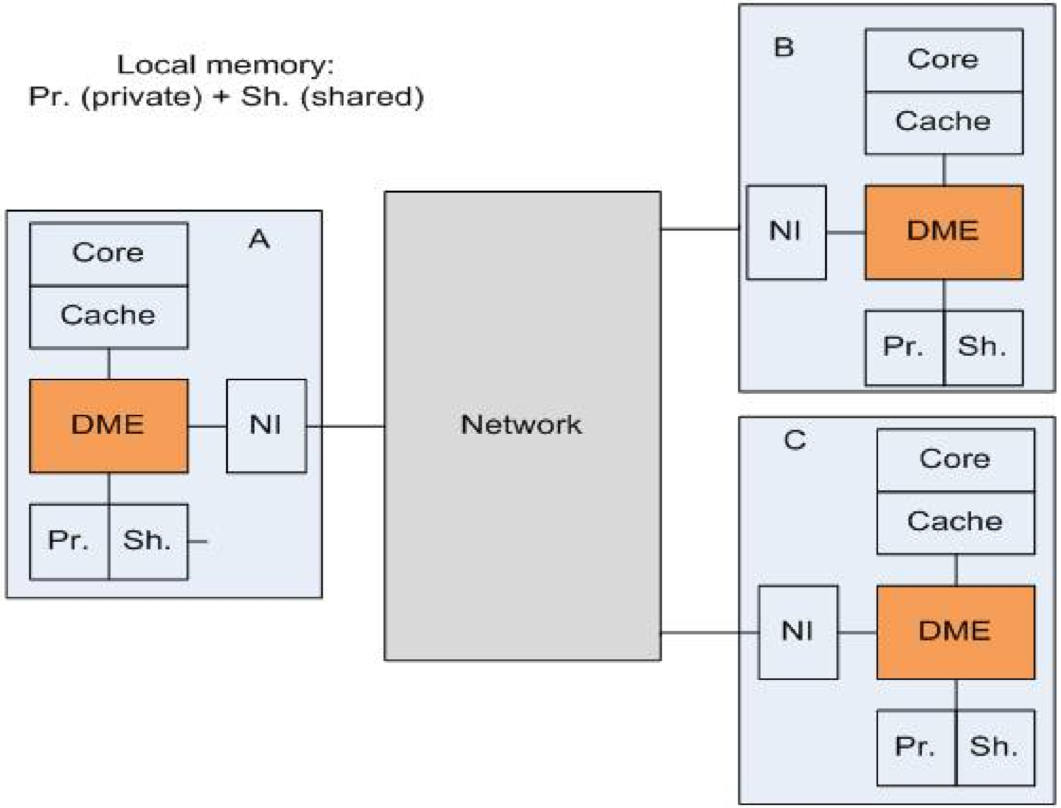
1. Architectural support
   - V2P translation
   - Cache coherency
   - Memory consistency
2. Higher level services
   - Memory allocation/de-allocation
   - Memory acesses
   - Abstract data types (ADTs)
   - Data movement
3. **APIs for** heap management, memory allocation and de-allocation, memory access and synchronization; ADT support
4. **Distributed Power management**



Data Management Engine (DME)

Interfacing core, local memory and network

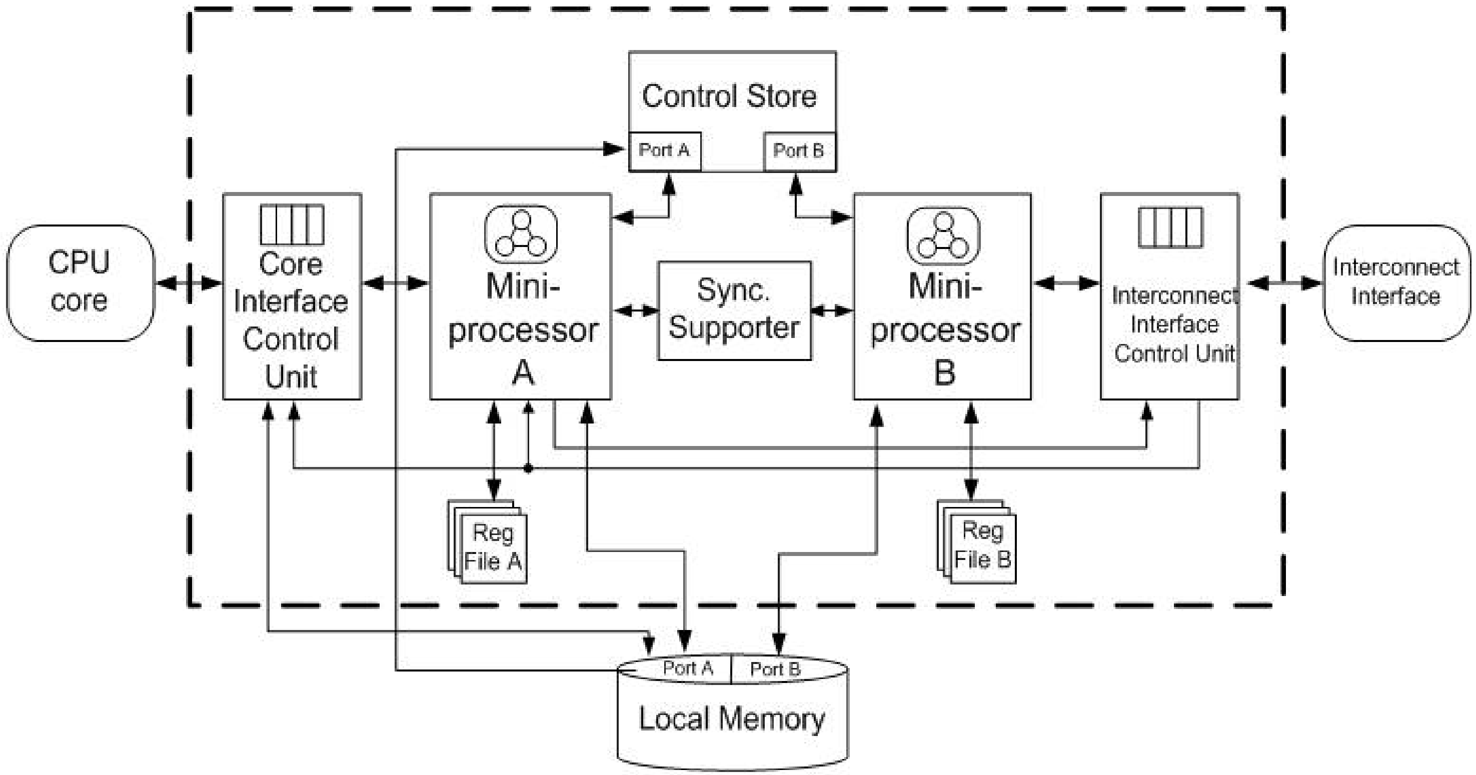
Realizing DME concepts



DME Functions:

- Virtual-Physical Address Translation
- Cache Coherence Protocols
- Memory Consistency
- ADT Support
- Support for Data Movement and Task Migration

DME Block Diagram



DME Functions and Features

- Micro-programmable
- Optimized for frequent functions
- Virtual to physical address translation
- Cache coherence protocol
- Memory consistency
- Support for dynamic memory allocation and abstract data type management

Summary of Nostrum Status

- Nostrum defines a 2 D mesh topology;
- Protocol stack for link layer, network layer and session layer;
- Packet switched and virtual circuit communication services;
- Buffer-less, loss-less switch with no routing tables;
- 2 level data protection scheme;
- QoS Features;
- Programmable Data Management Engine
- Flexible NoC Simulator;

Ongoing Work:

- Contract based QoS Provision
- Distributed Memory Architecture
- Power management architwecture
- 3D Architectures

Further information: www.ict.kth.se/nostrum