UNIVERSITY of York

Priority-based Wormhole Networks-on-Chip: challenges and opportunities

Leandro Soares Indrusiak

Real-Time Systems Group
Department of Computer Science
University of York
United Kingdom

ECRTS 2017
27-30 June
Dubrovnik, Croatia

RTN 2017

RTS York

Outline

- Wormhole Networks
- Networks-on-Chip
- Real-Time Analysis
- Resource Management

Motivation

- Multiprocessor and multicore systems forced a shift towards communication-centric design
  - abundant computation resources
  - shared communication media

- Inter-processor communication
  - point-to-point
  - bus
  - networks

source: IBM, Intel

Networks – key characteristics

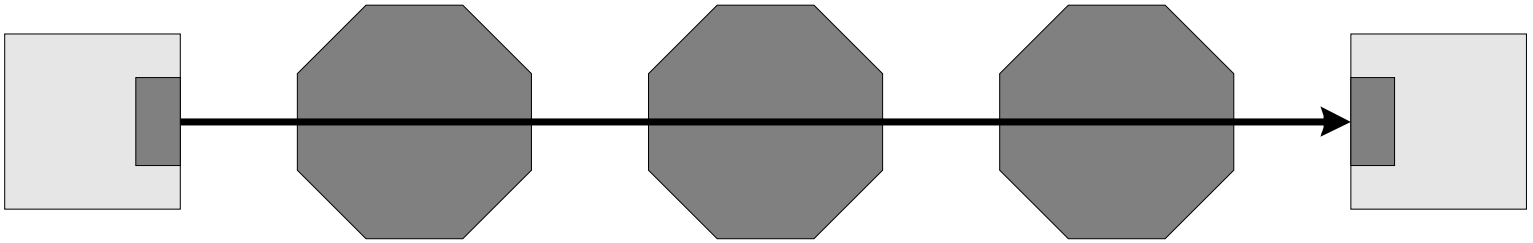
- topology
  - mesh, star, torus…
- routing protocol
  - deterministic, adaptive…
- arbitration
  - round-robin, priority preemptive, priority non-preemptive, TDM…
- buffering
  - FIFO, SAFC, SAMQ, DAMQ, hot potato…
- flow control protocol
  - handshake, credit-based…
- switching protocol
  - circuit, store-and-forward, wormhole

Networks – key characteristics

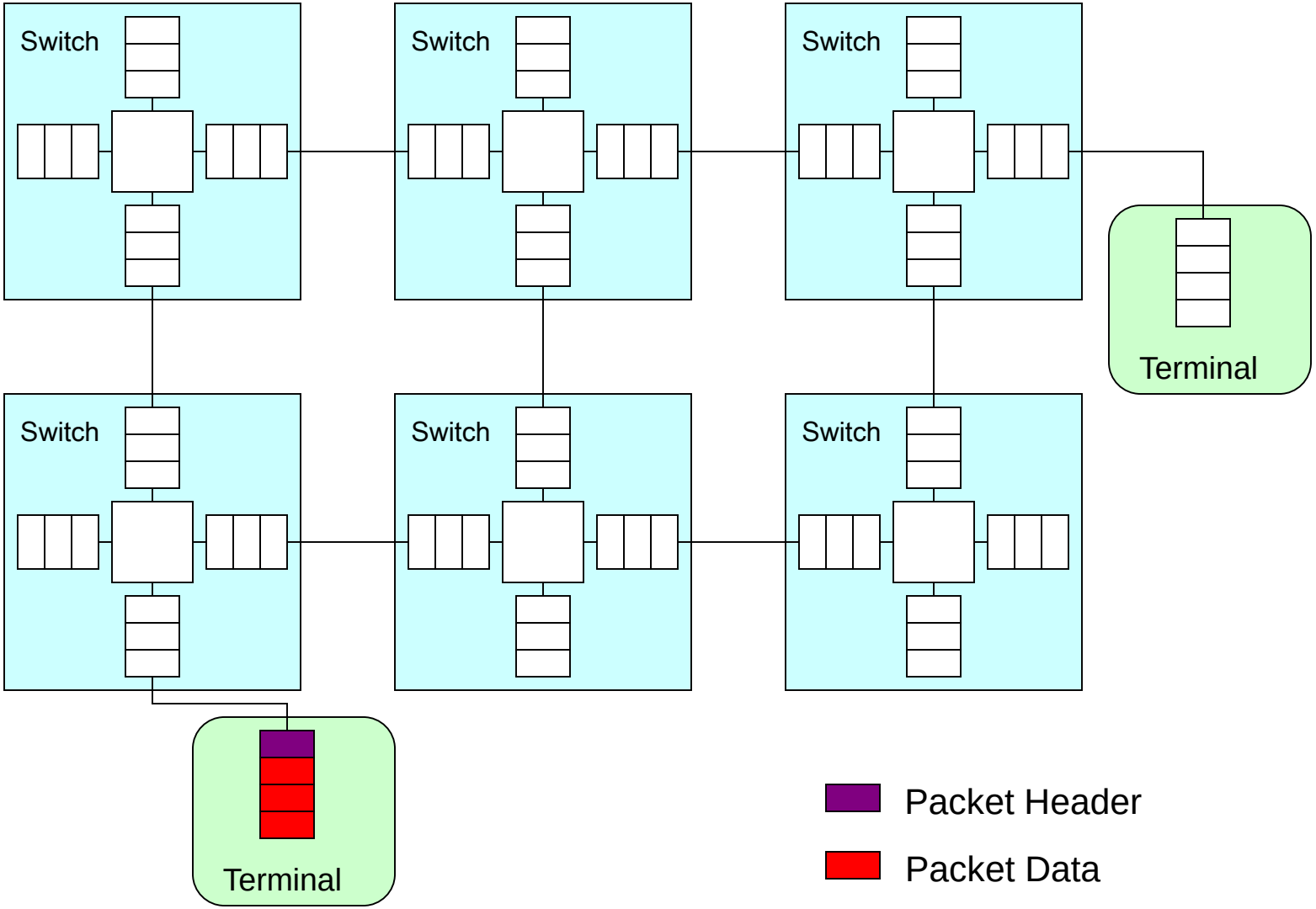
- topology
  - mesh, star, torus…
- routing protocol
  - deterministic, adaptive…
- arbitration
  - round-robin, priority preemptive, priority non-preemptive, TDM…
- buffering
  - FIFO, SAFC, SAMQ, DAMQ, hot potato…
- flow control protocol
  - handshake, credit-based…
- switching protocol
  - circuit, store-and-forward, wormhole

Circuit switching

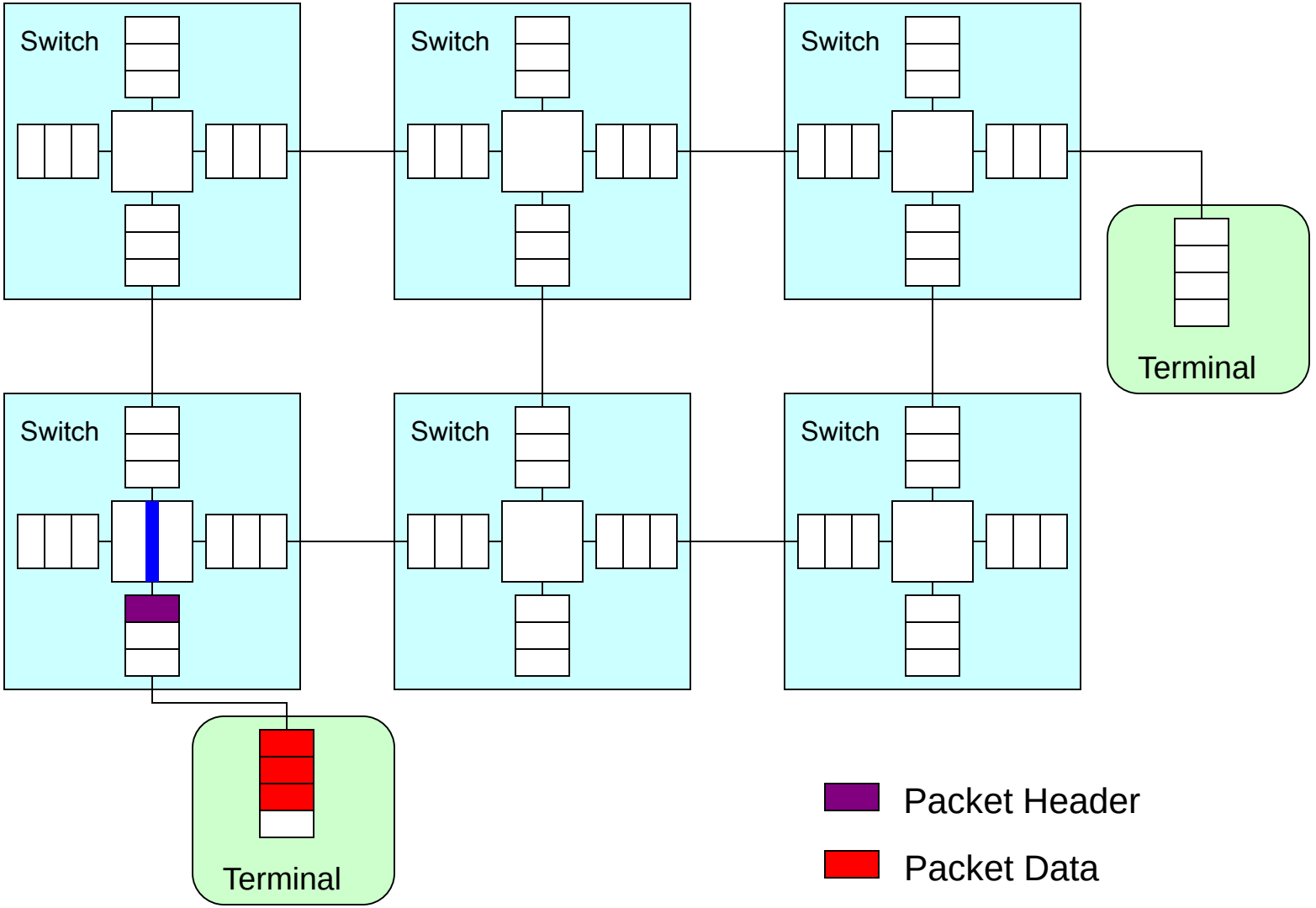
- Packets are forwarded through dedicated paths that are kept until the transmission is finished
- Contention can arise when establishing a path
  - no further contention once path is established
- Temporary packet buffering in routers is not required
- Suitable to long and infrequent messages
  - time to establish a path can be high



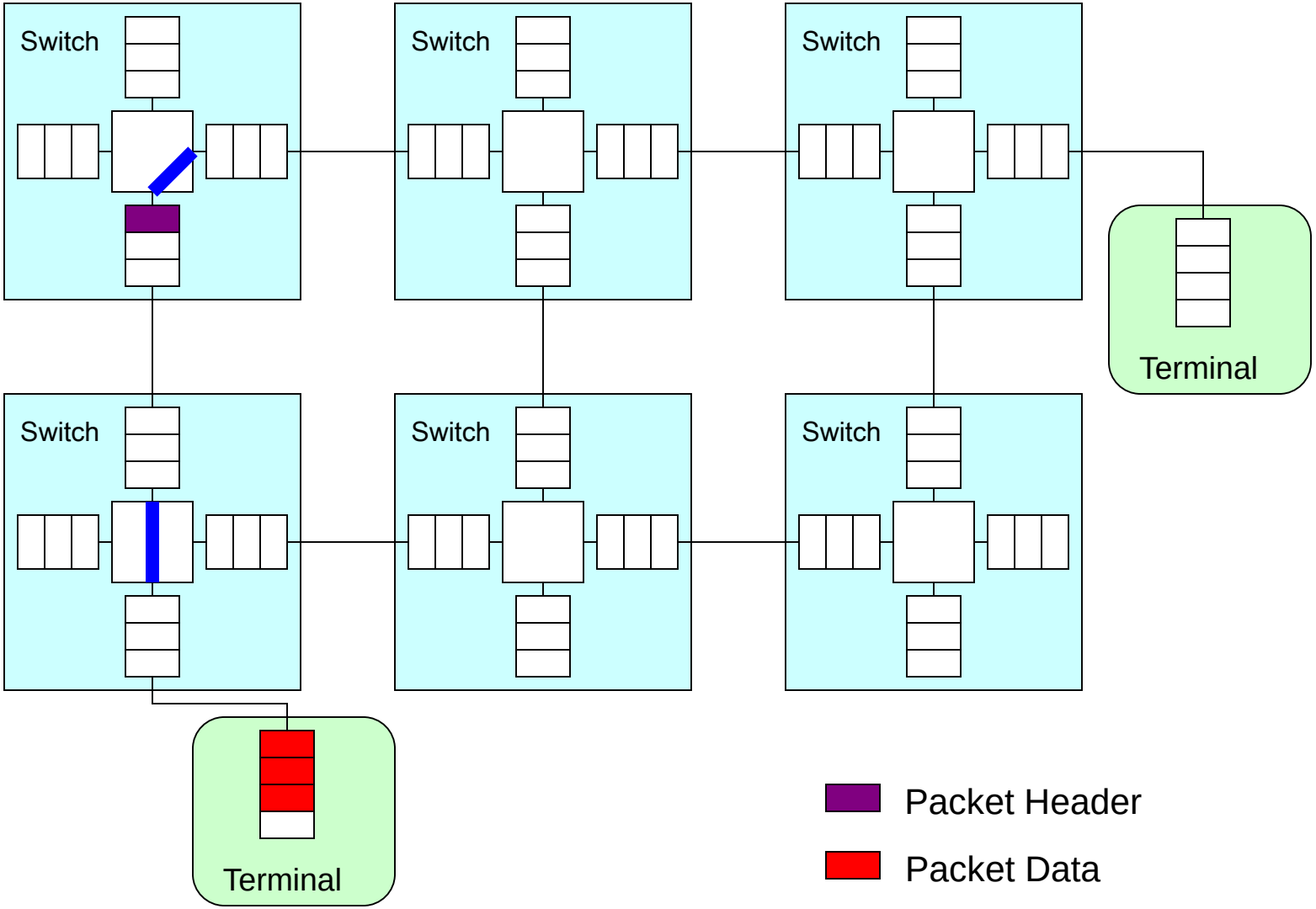
Circuit Switching



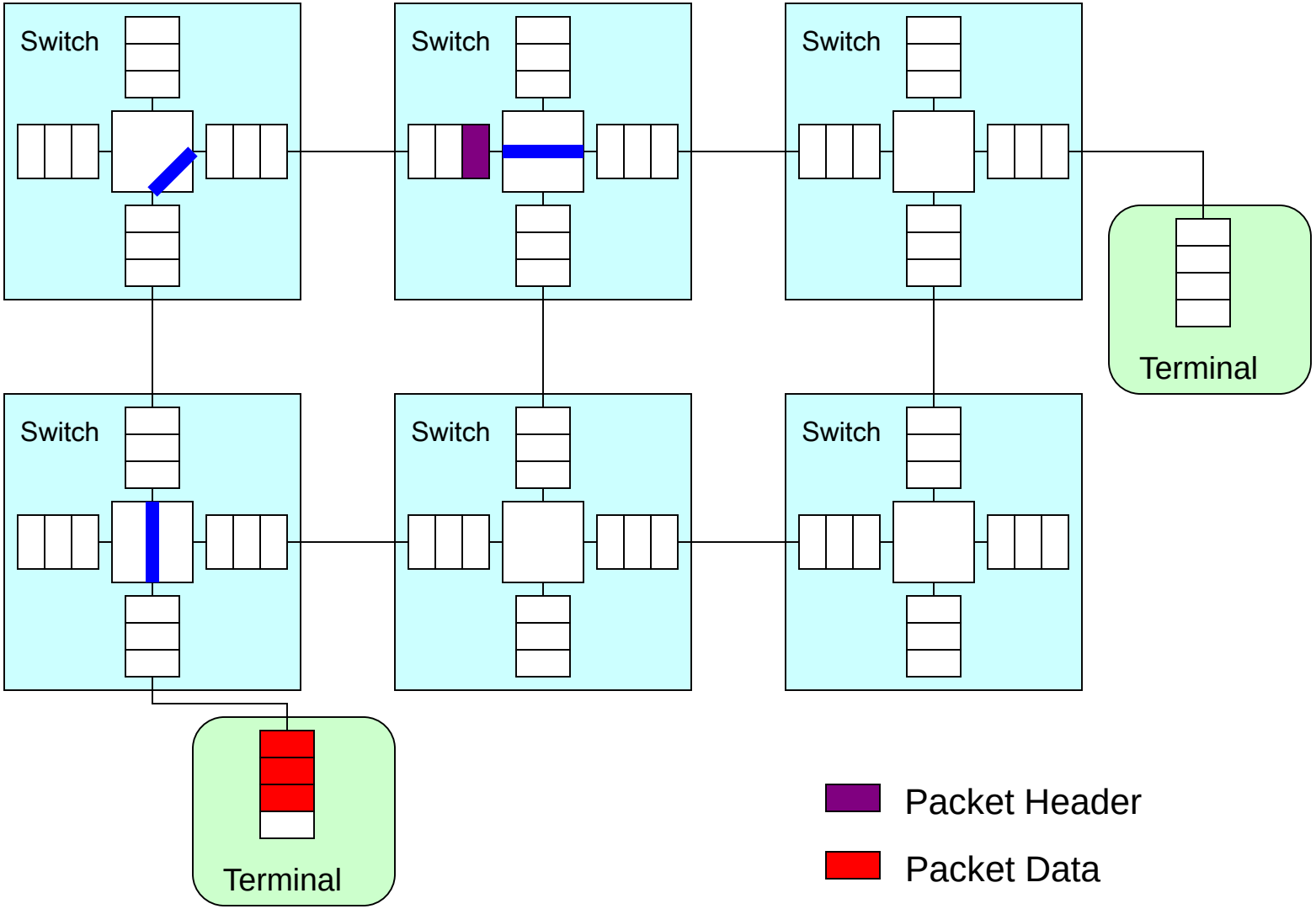
Circuit Switching



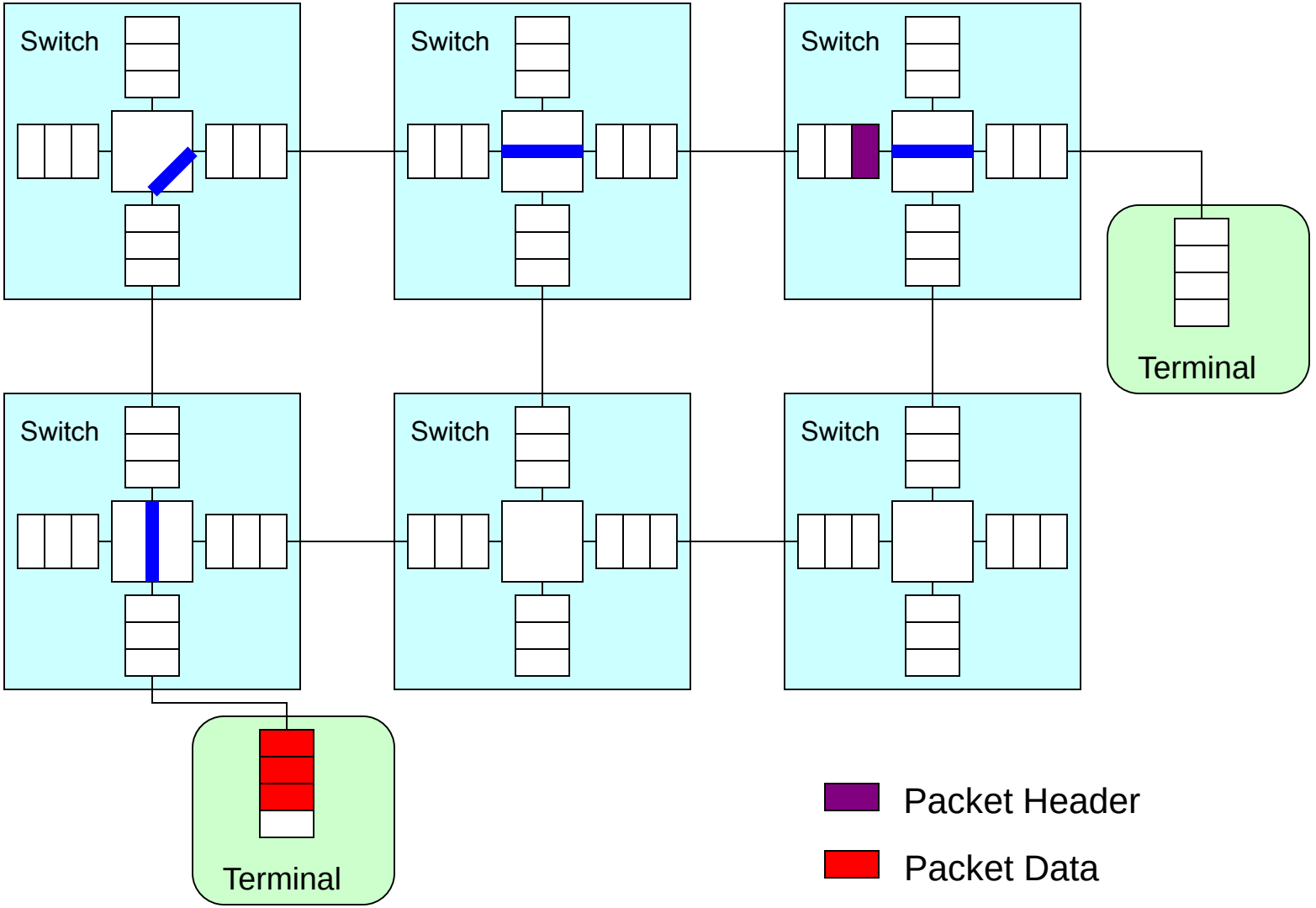
Circuit Switching



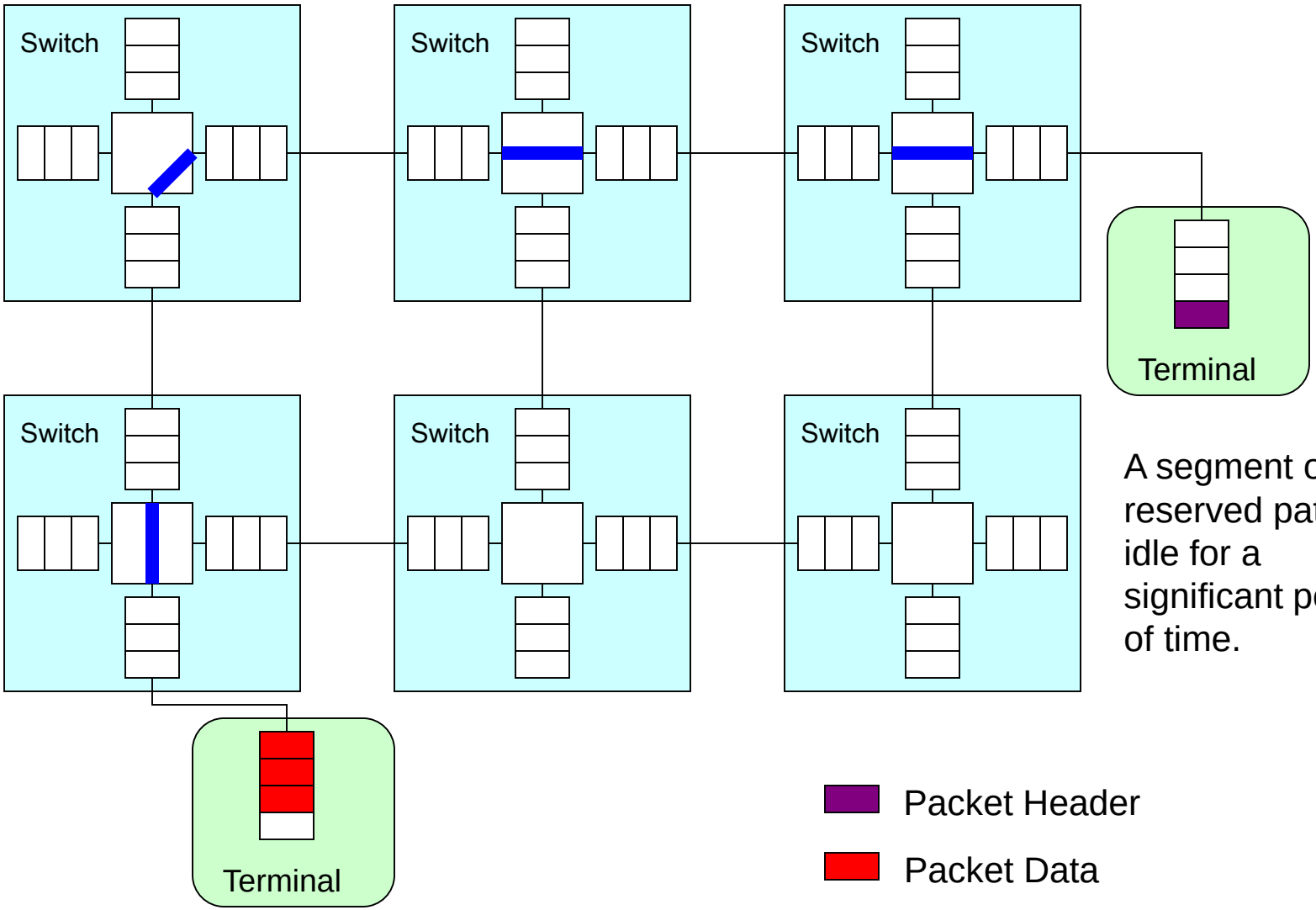
Circuit Switching



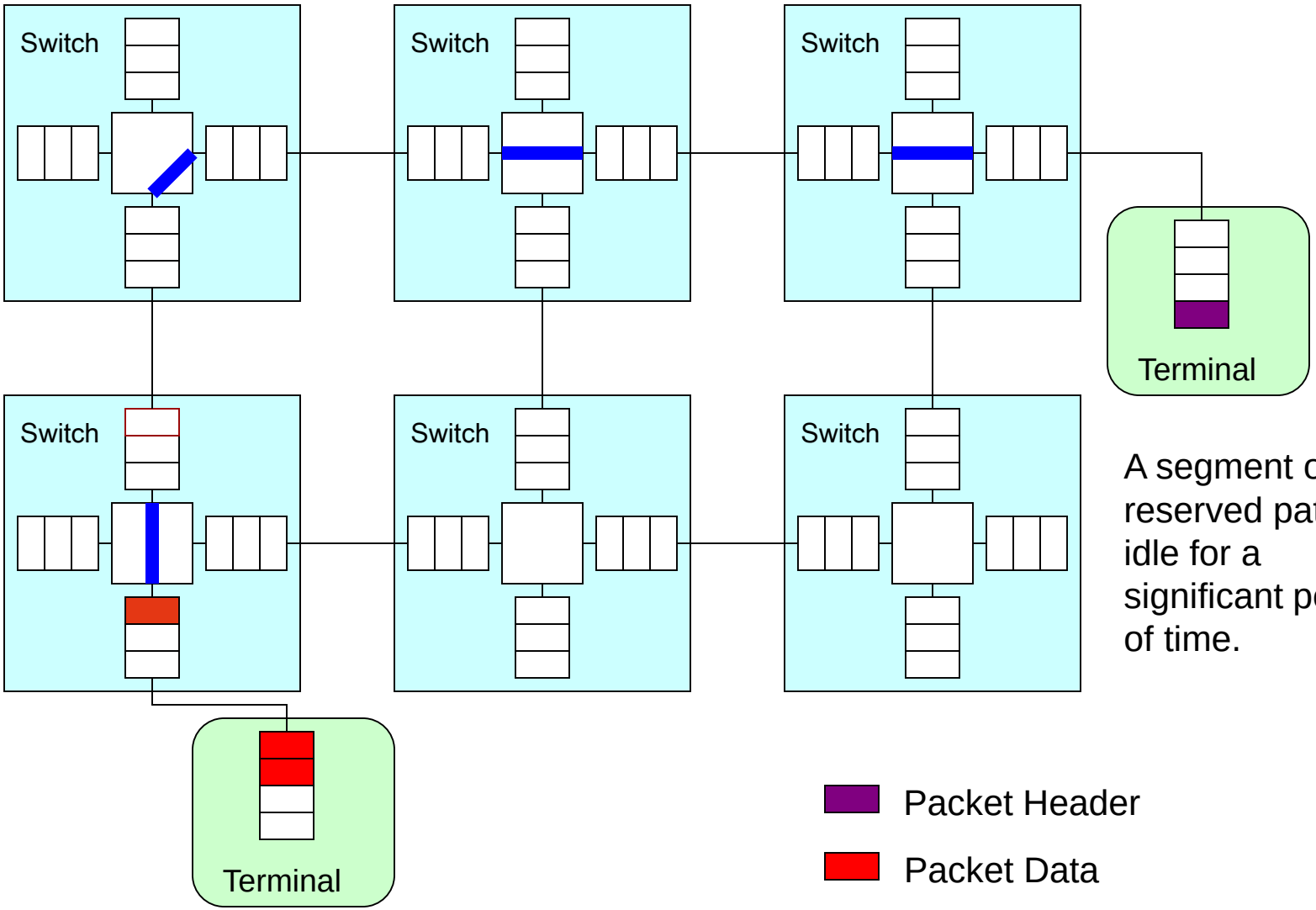
Circuit Switching



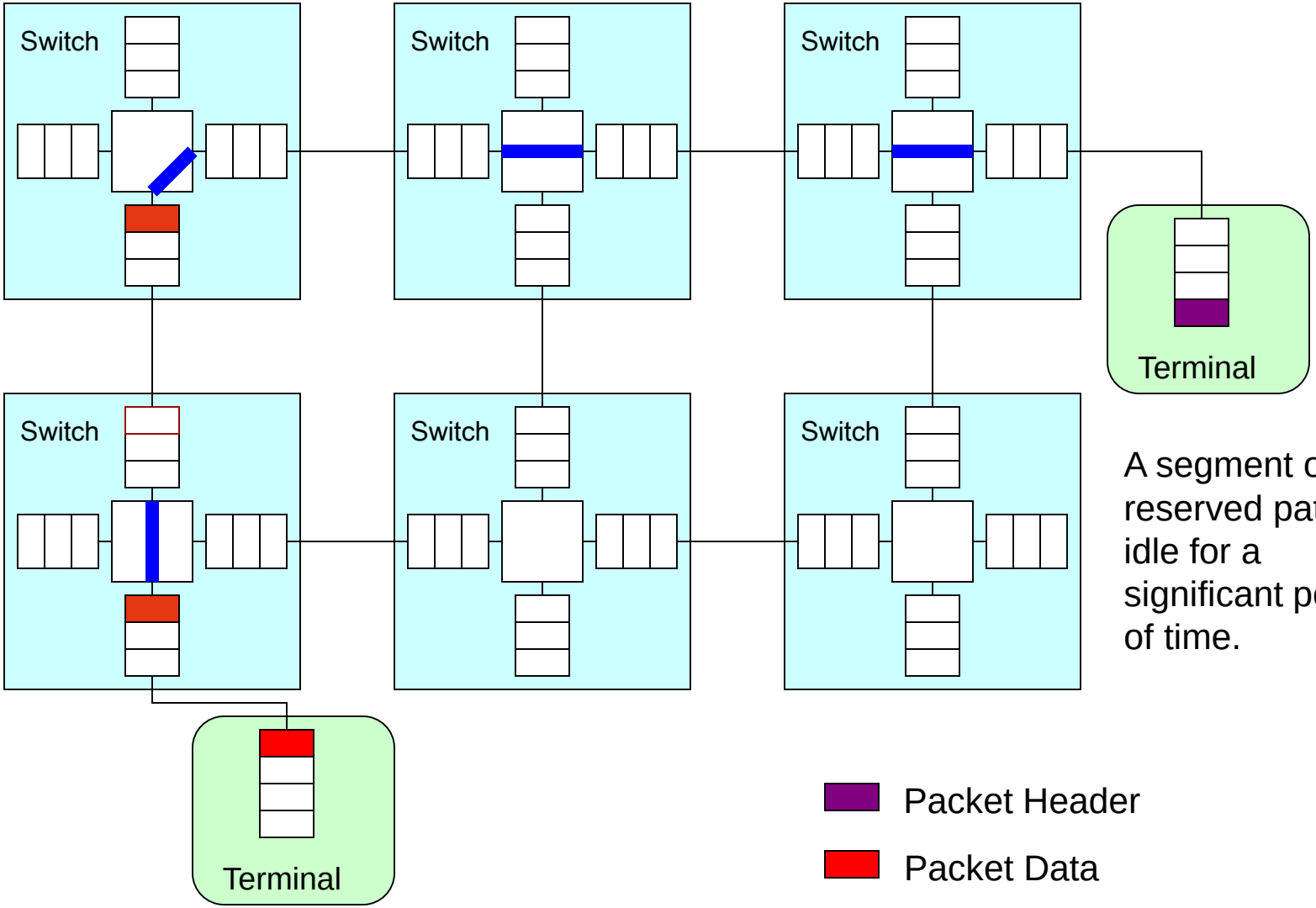
Circuit Switching



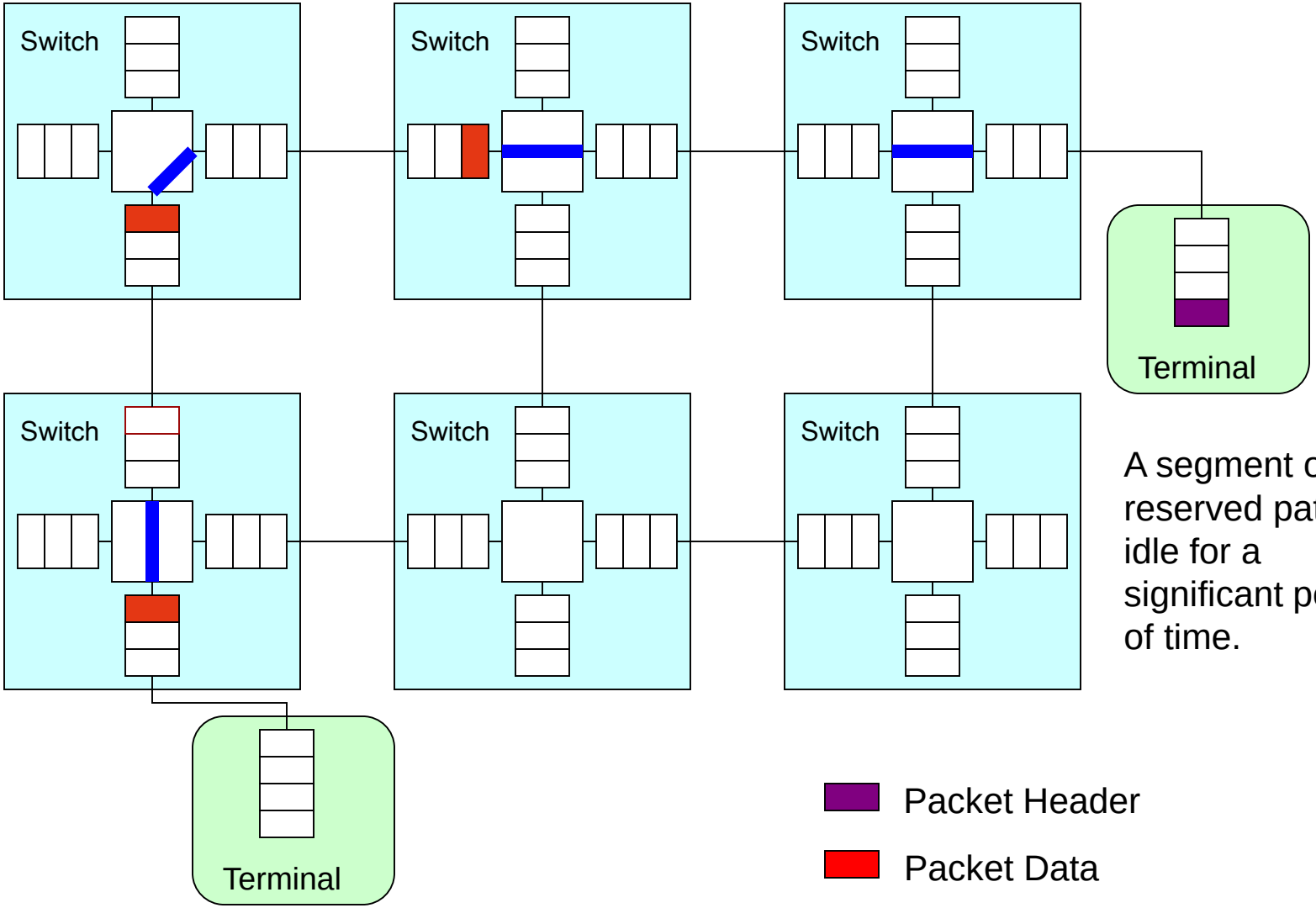
Circuit Switching



Circuit Switching

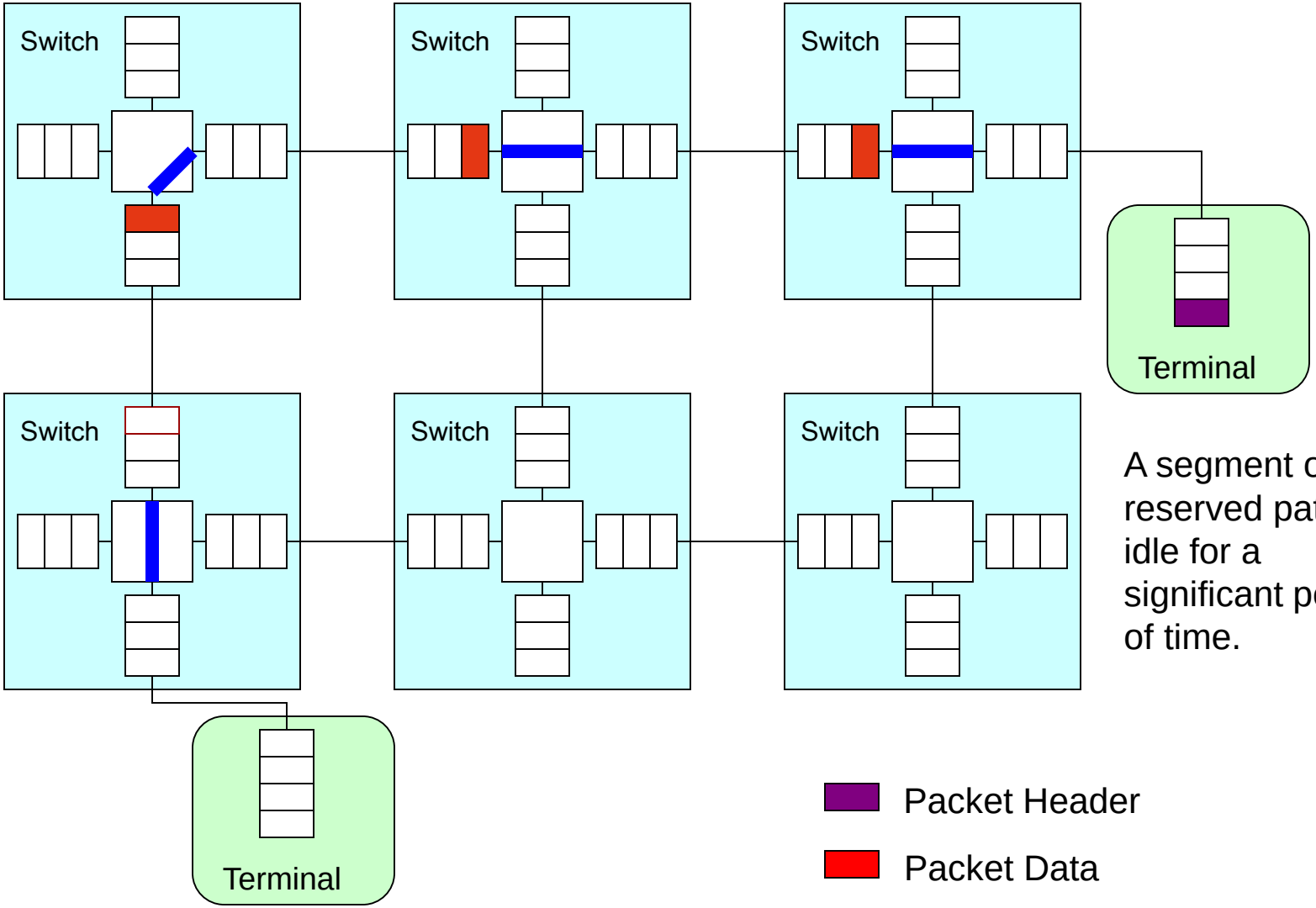


Circuit Switching

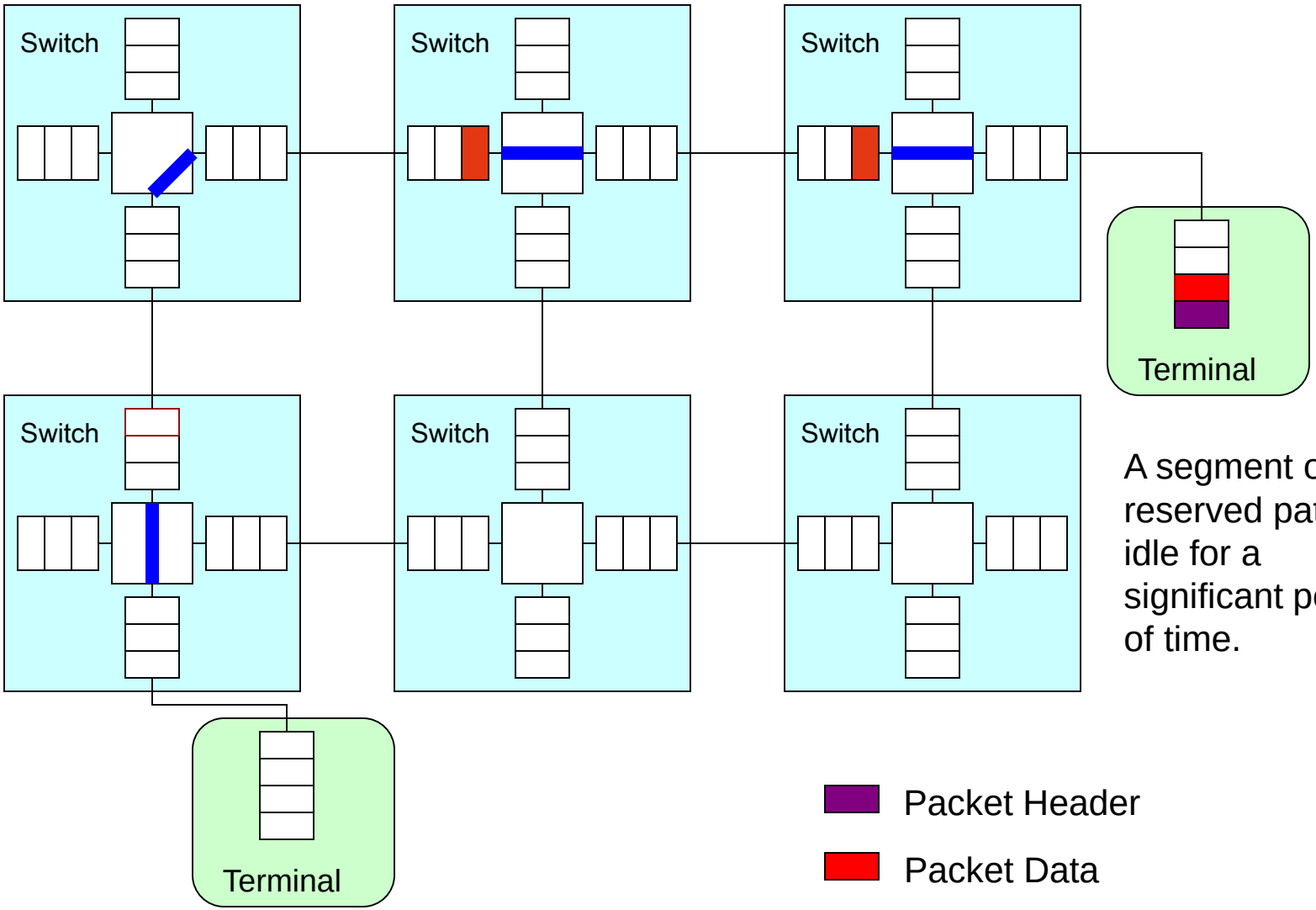


A segment of reserved path is idle for a significant period of time.

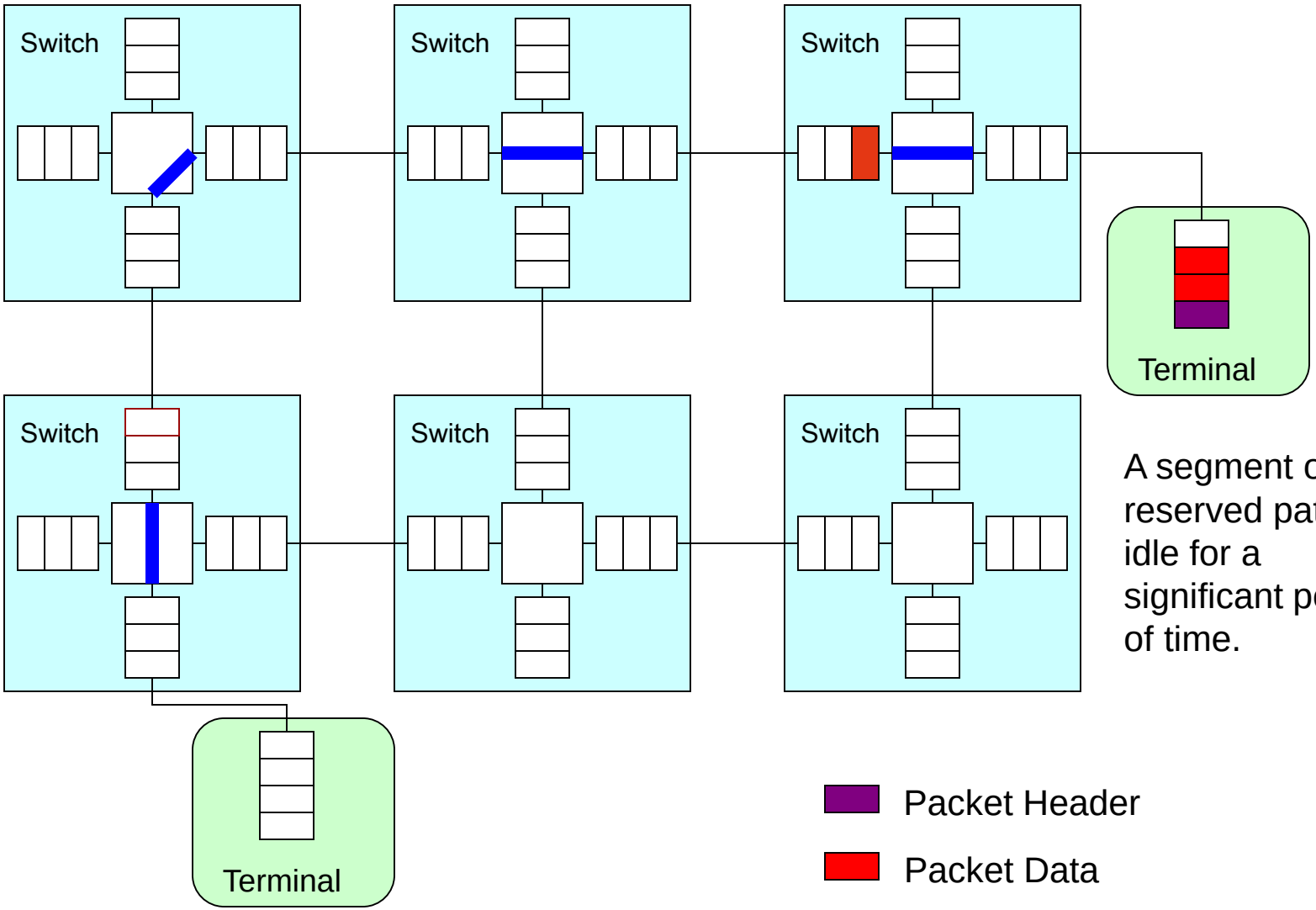
Circuit Switching



Circuit Switching

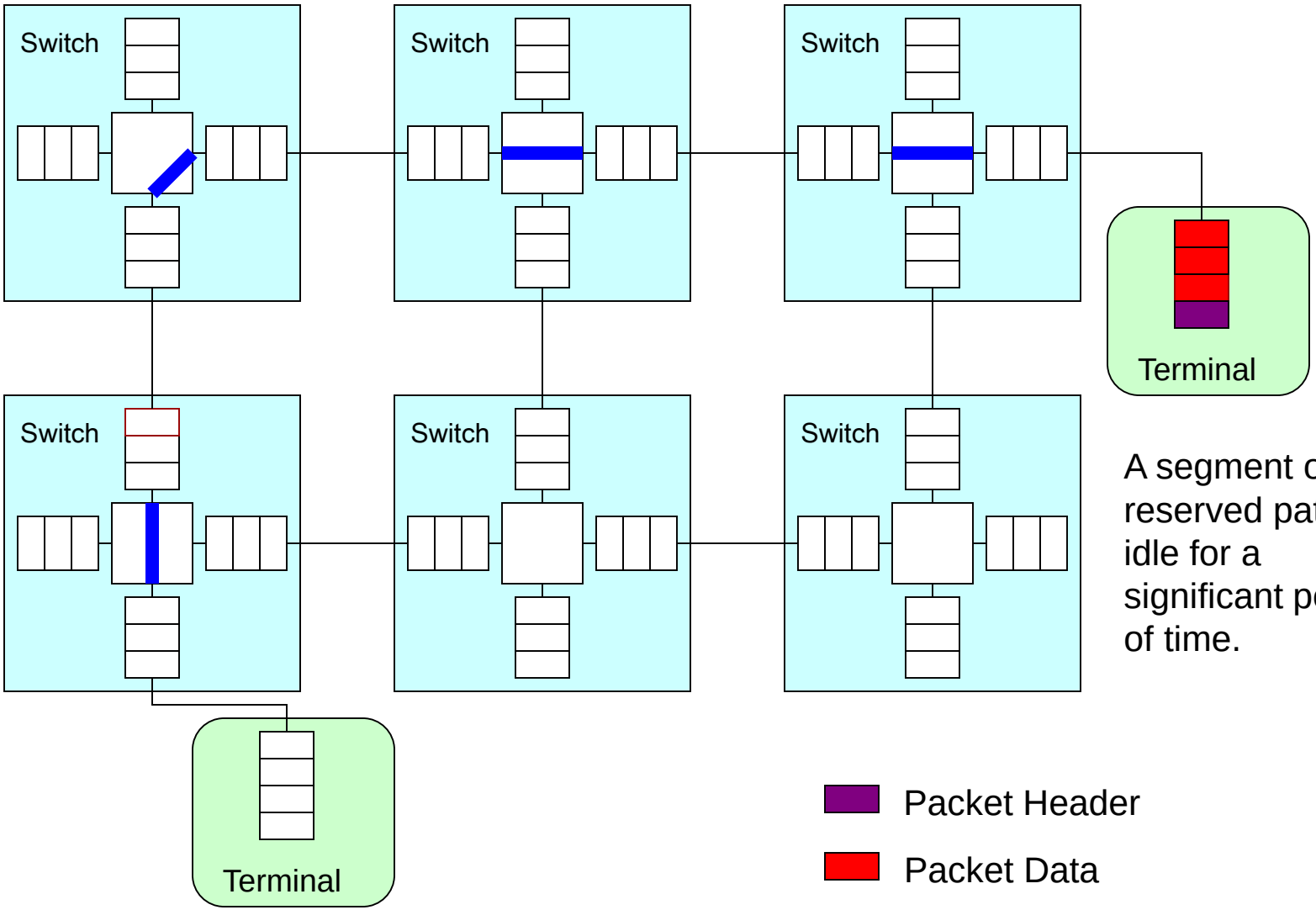


Circuit Switching

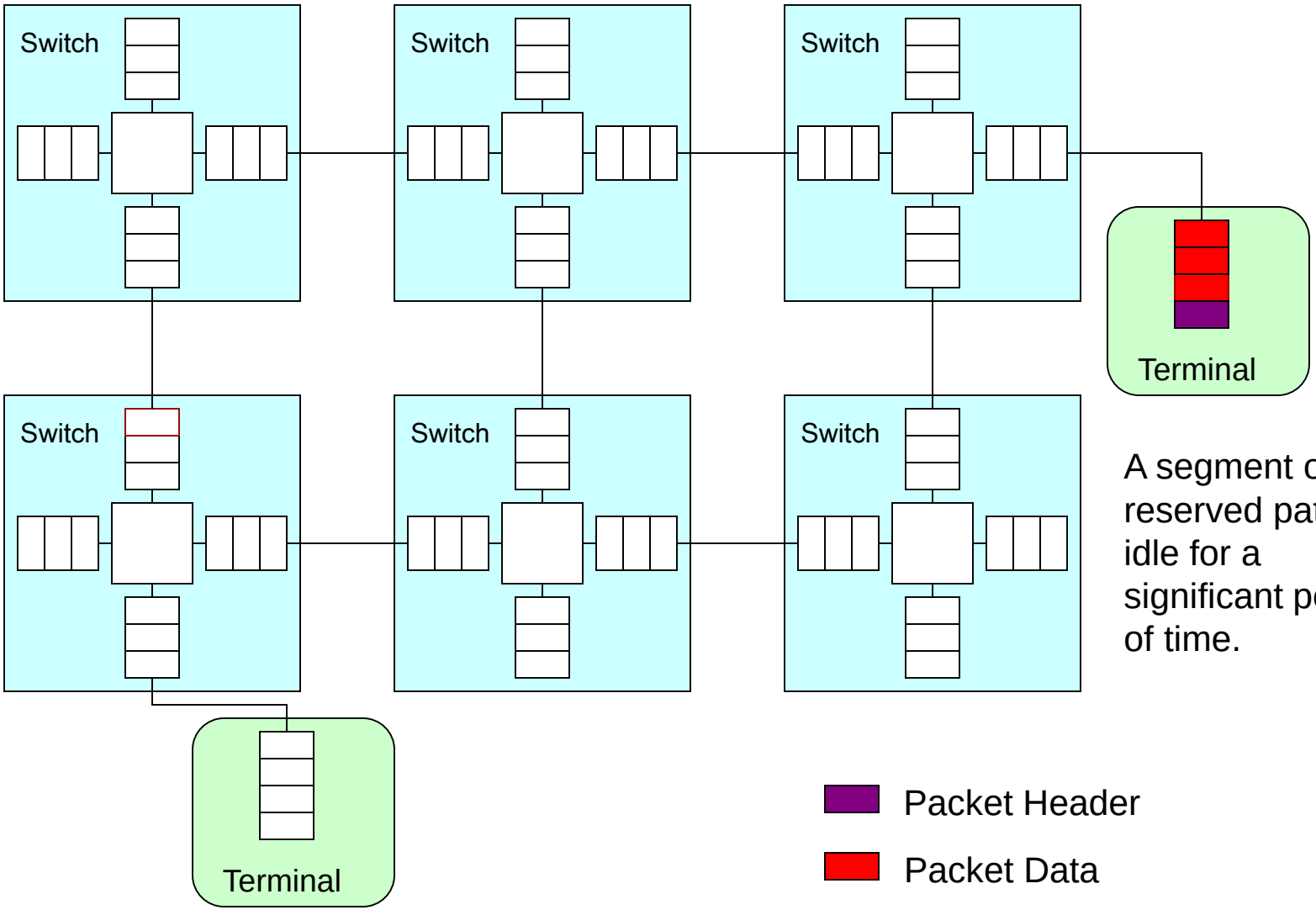


A segment of reserved path is idle for a significant period of time.

Circuit Switching

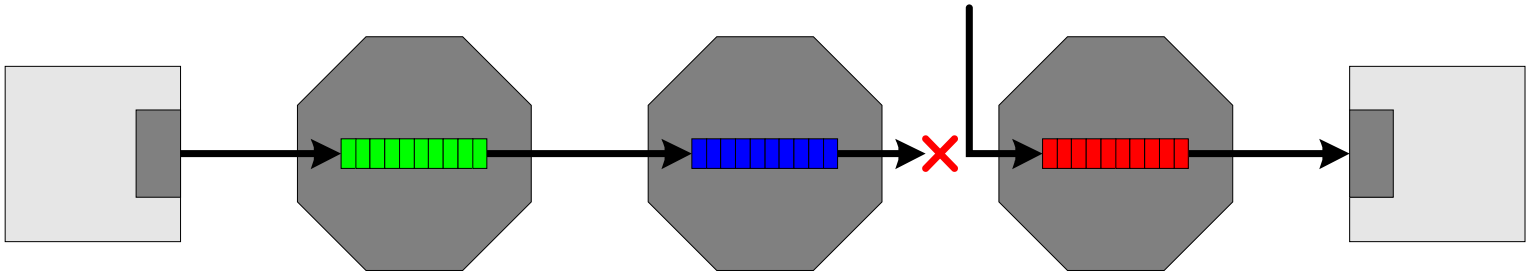


Circuit Switching

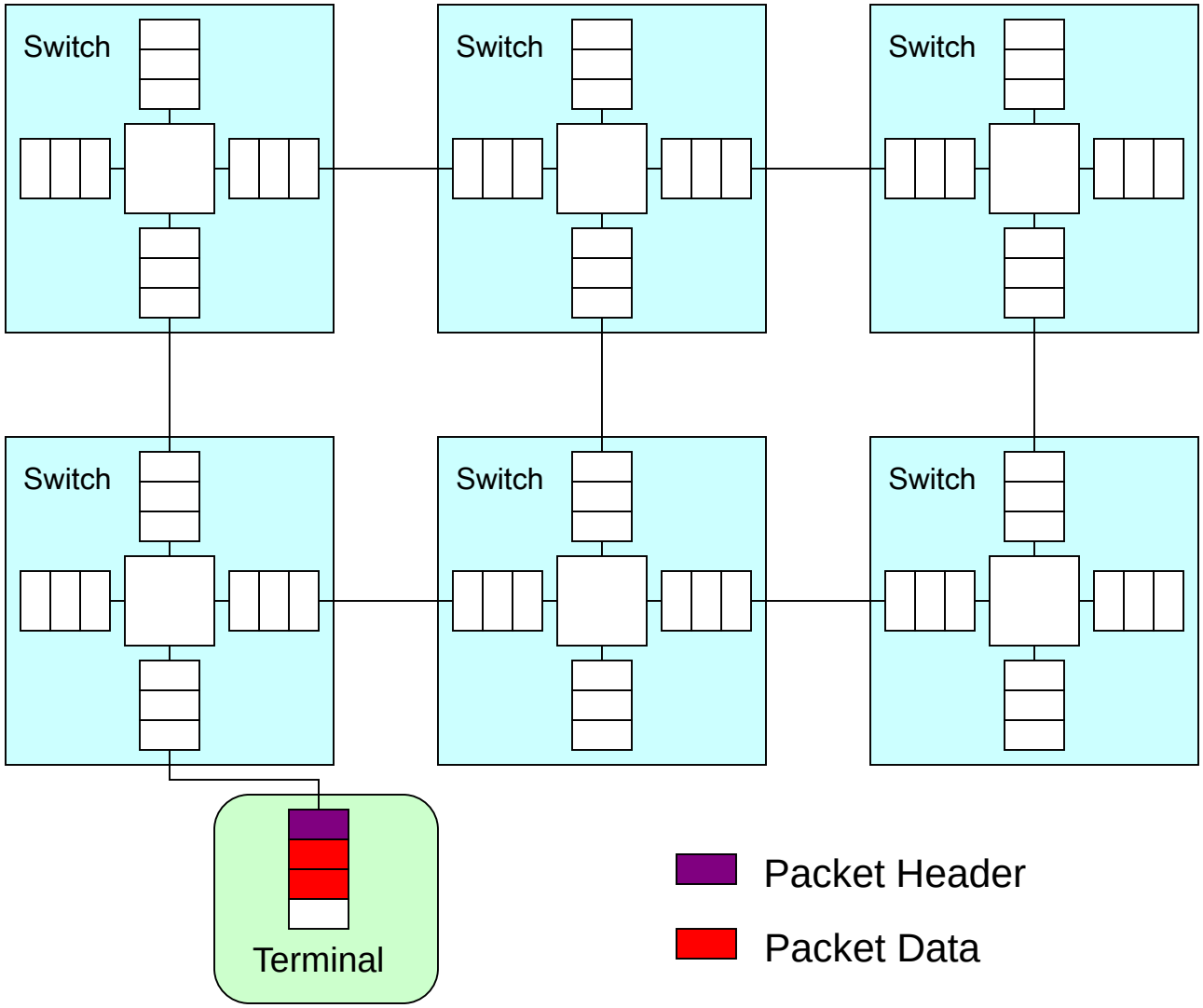


SAF Switching

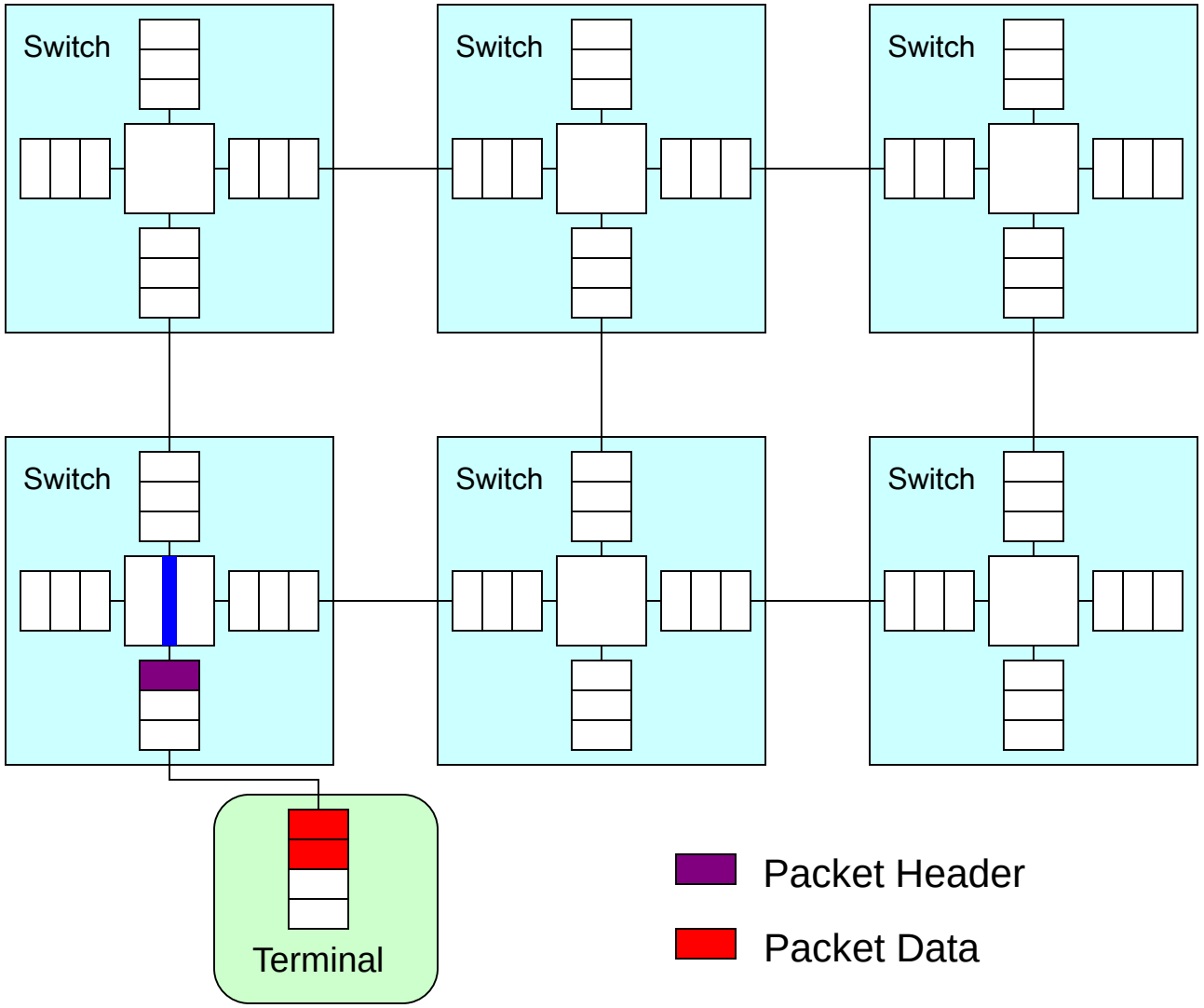
- SAF - Store And Forward
- Routers can only forward a packet once it is completely received and stored
  - packet acquires one link at a time
- Router input ports must have enough buffering space to temporarily store a complete packet



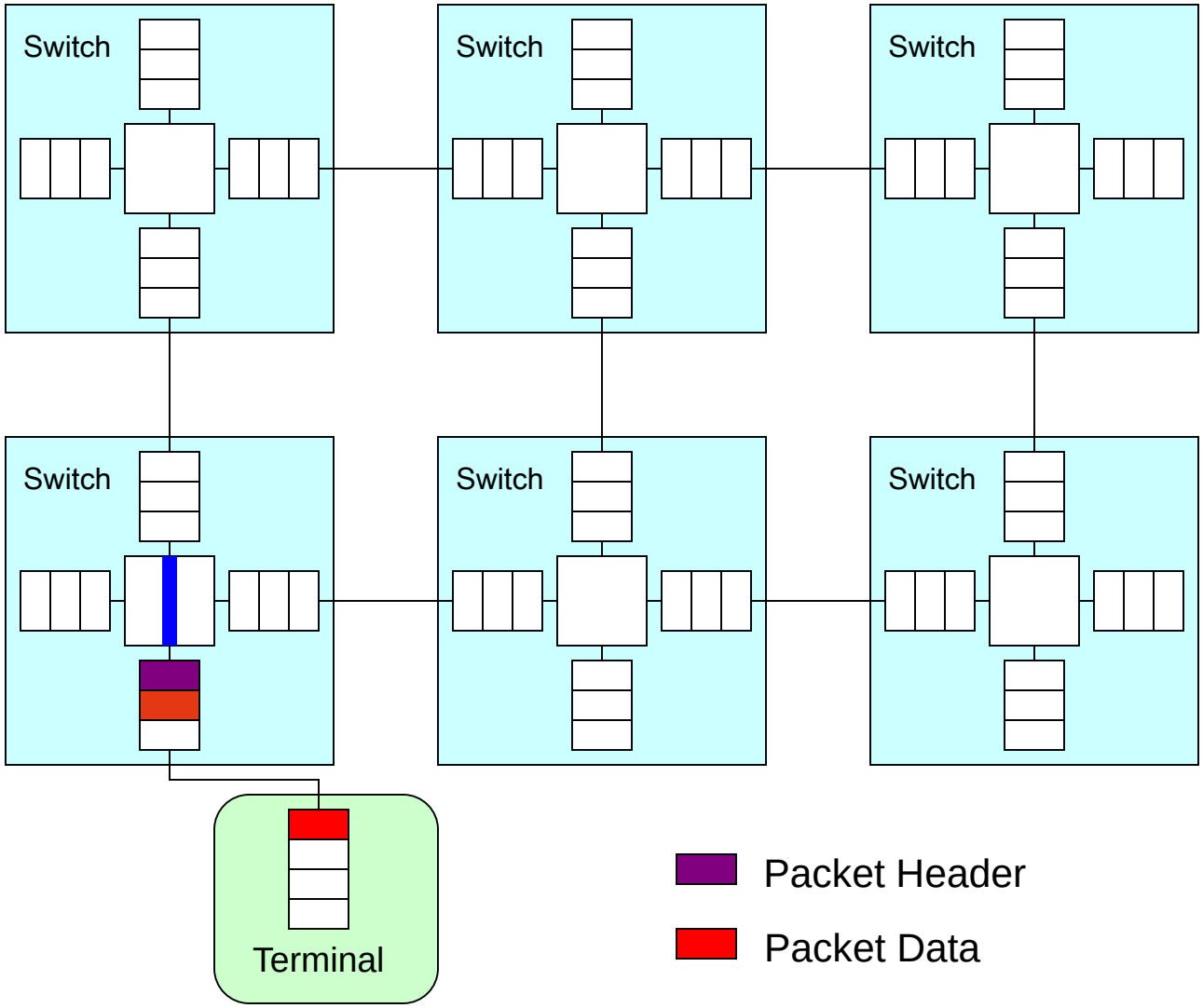
SAF Switching



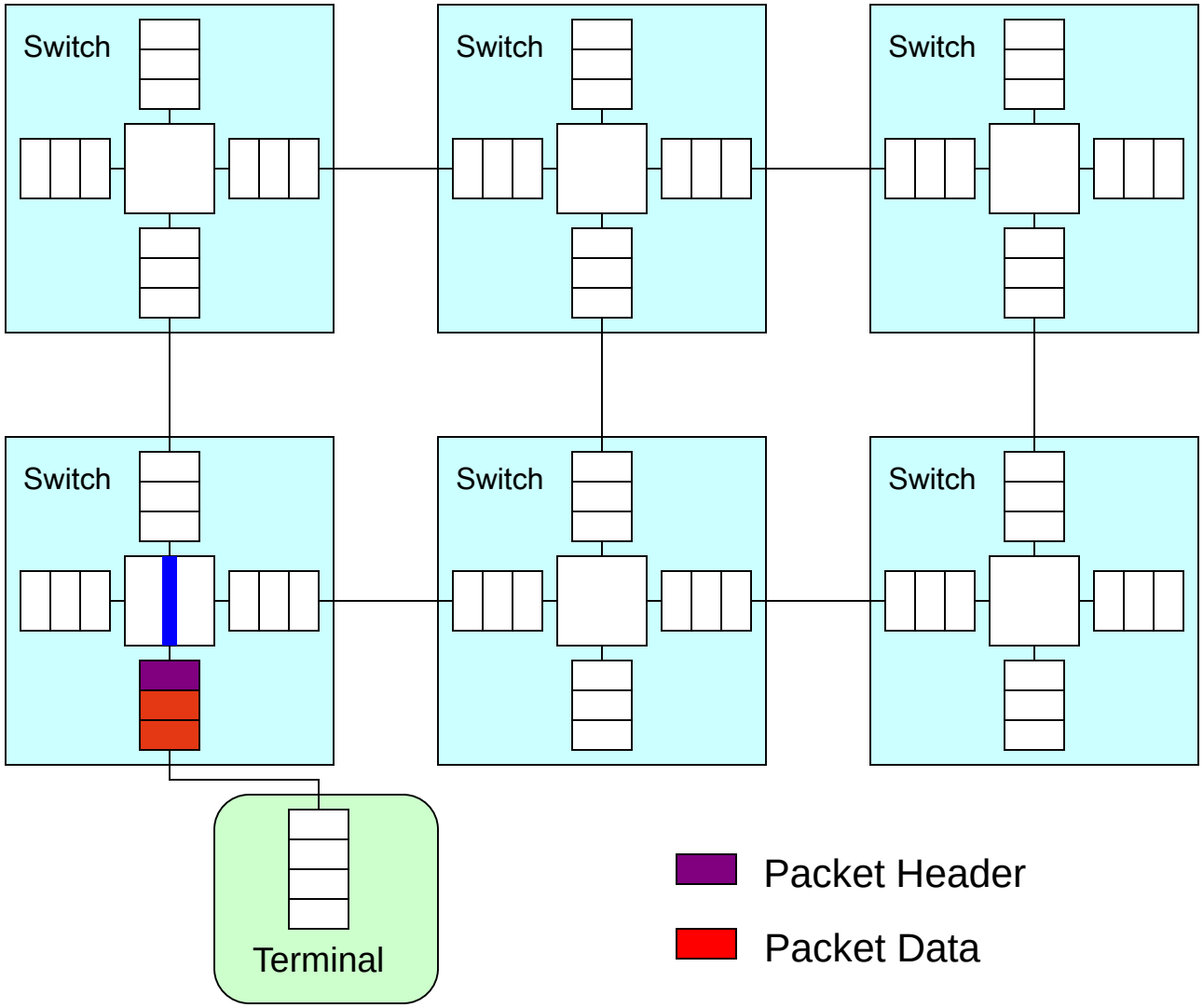
SAF Switching



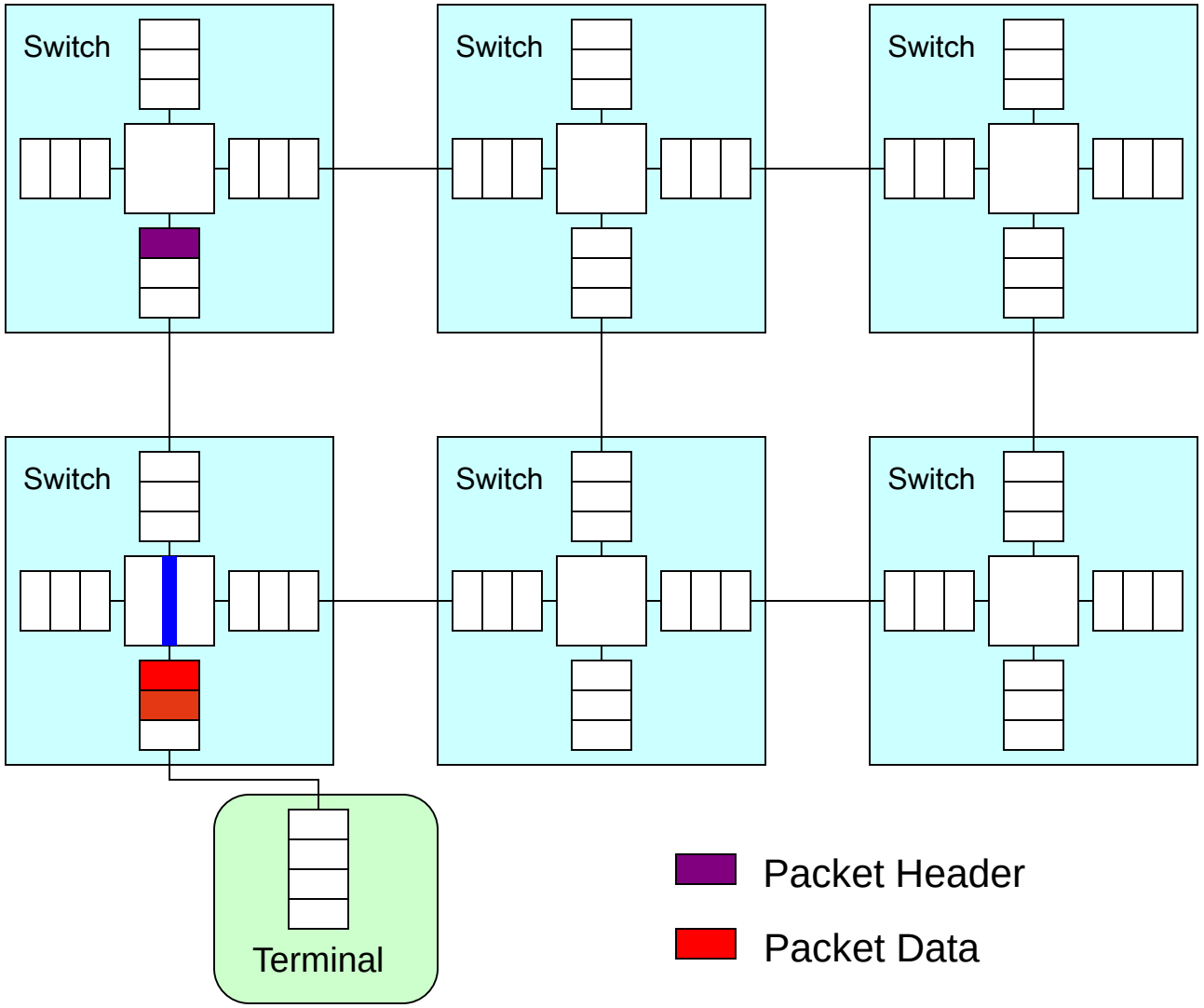
SAF Switching



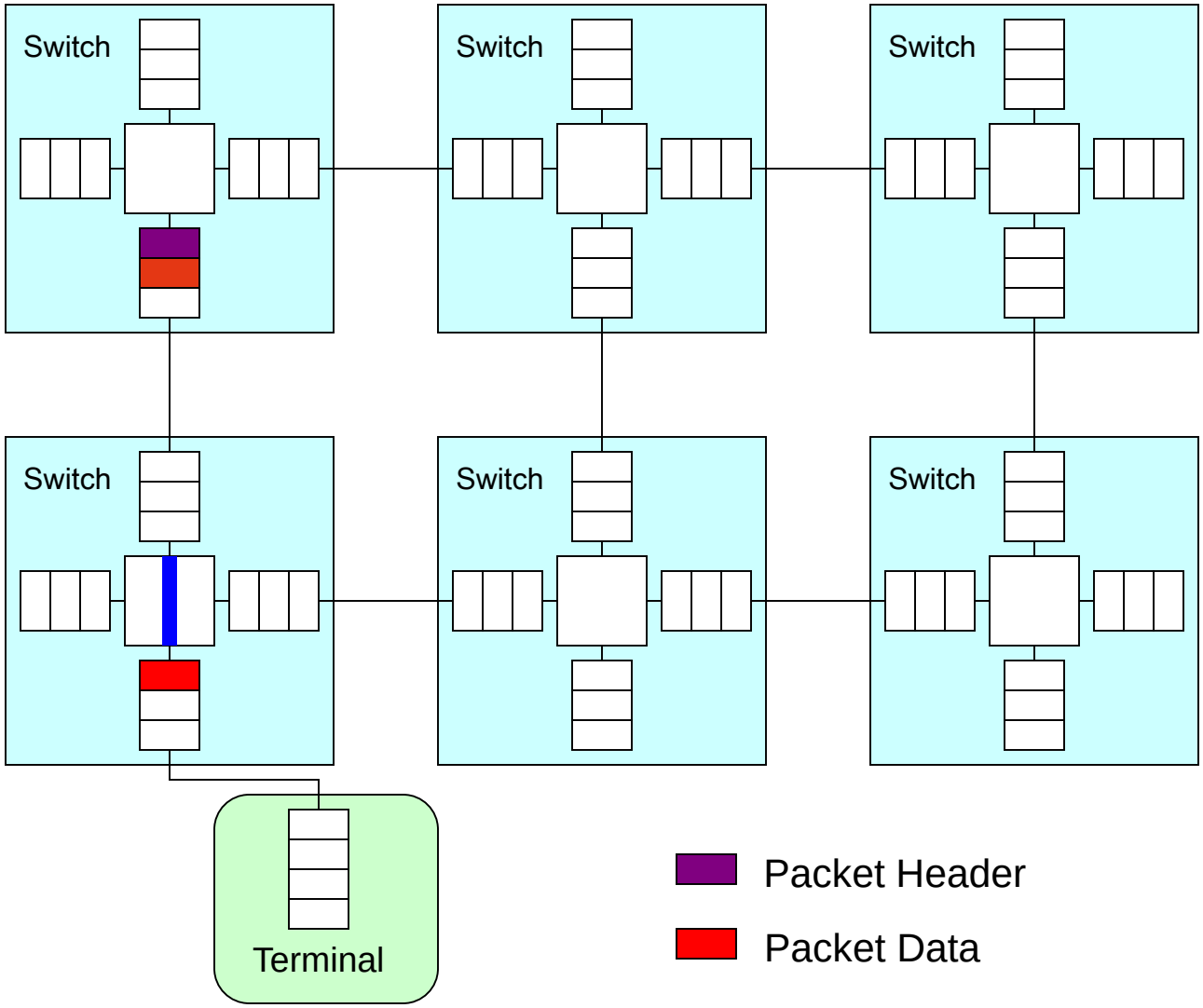
SAF Switching



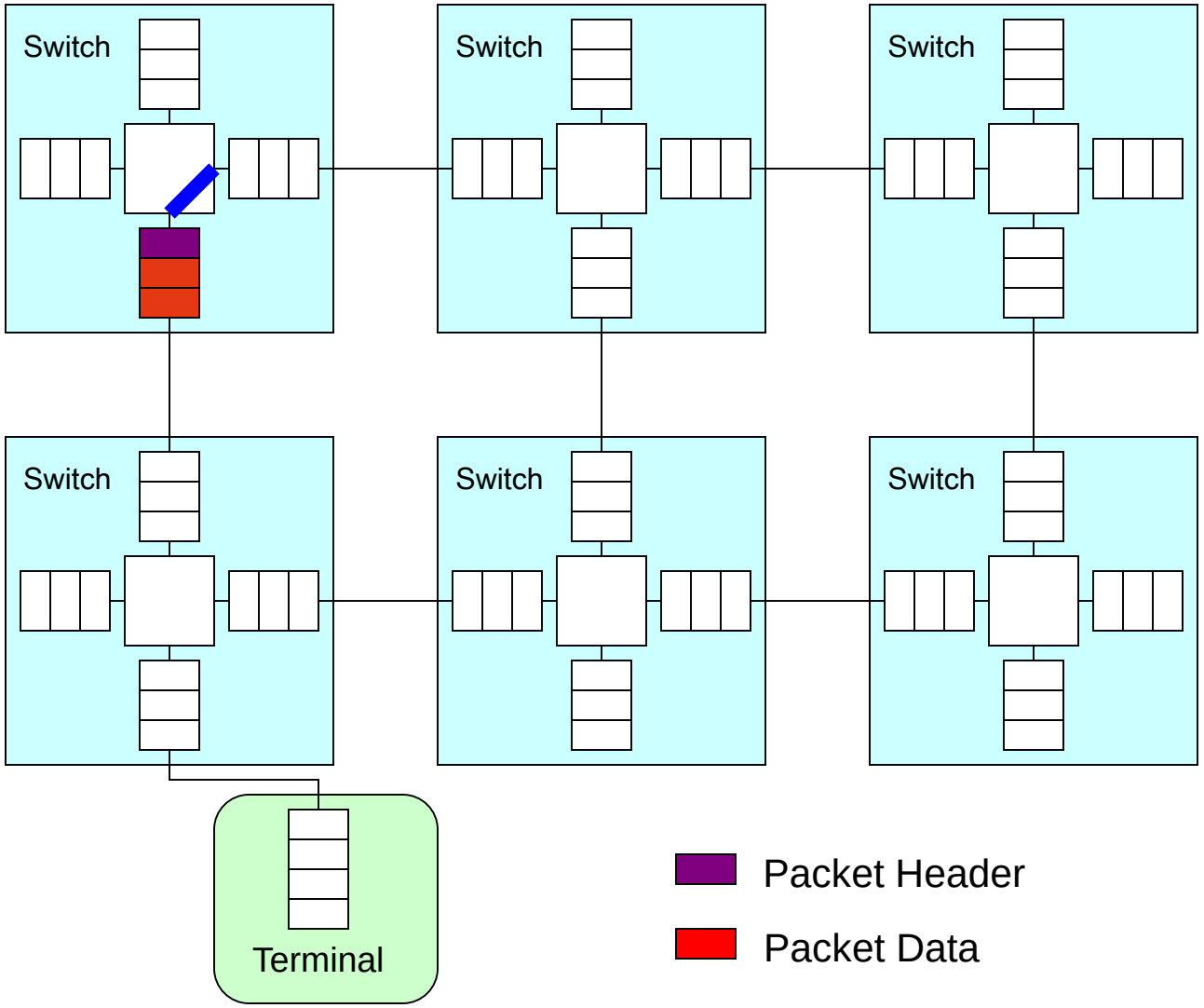
SAF Switching



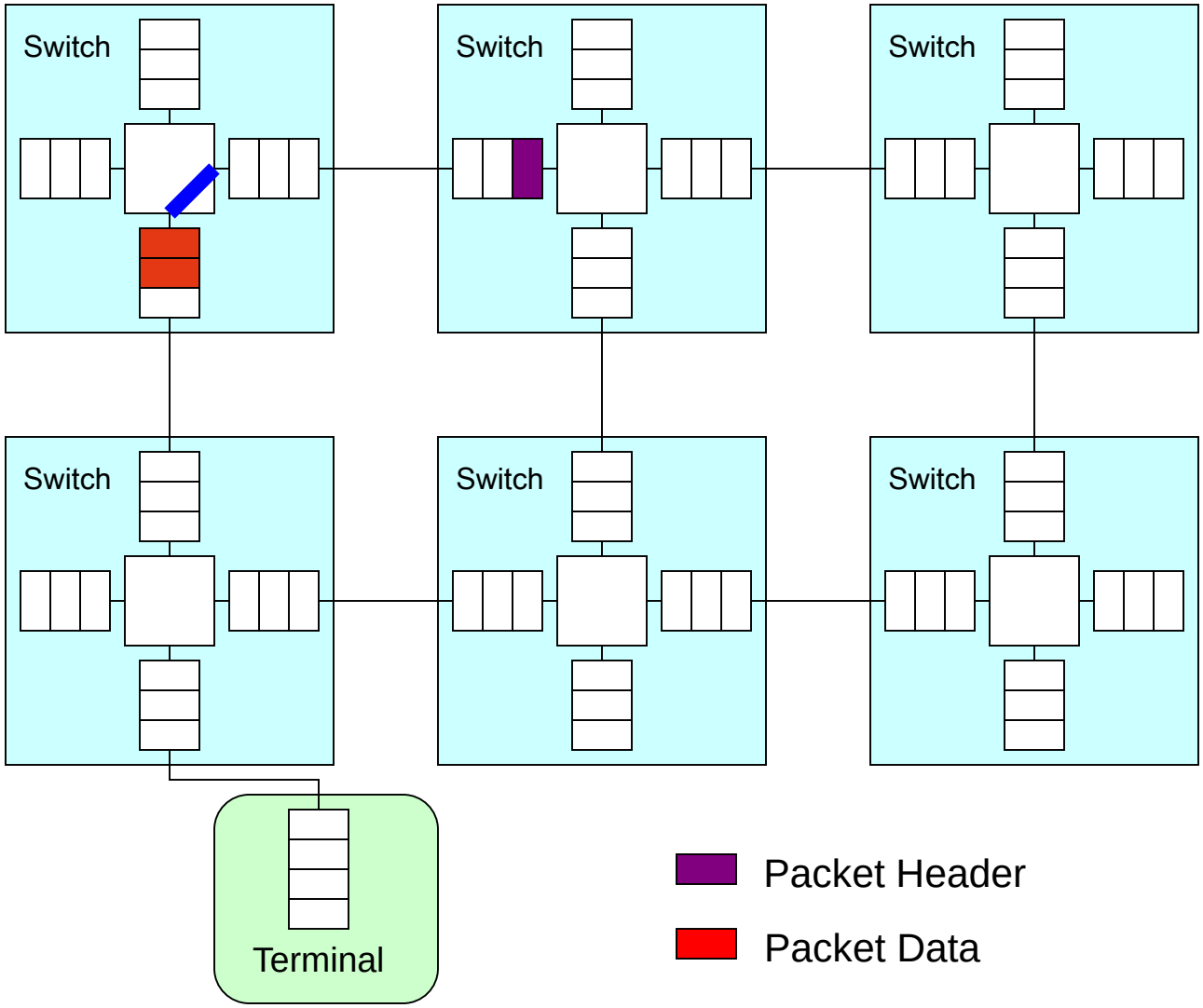
SAF Switching



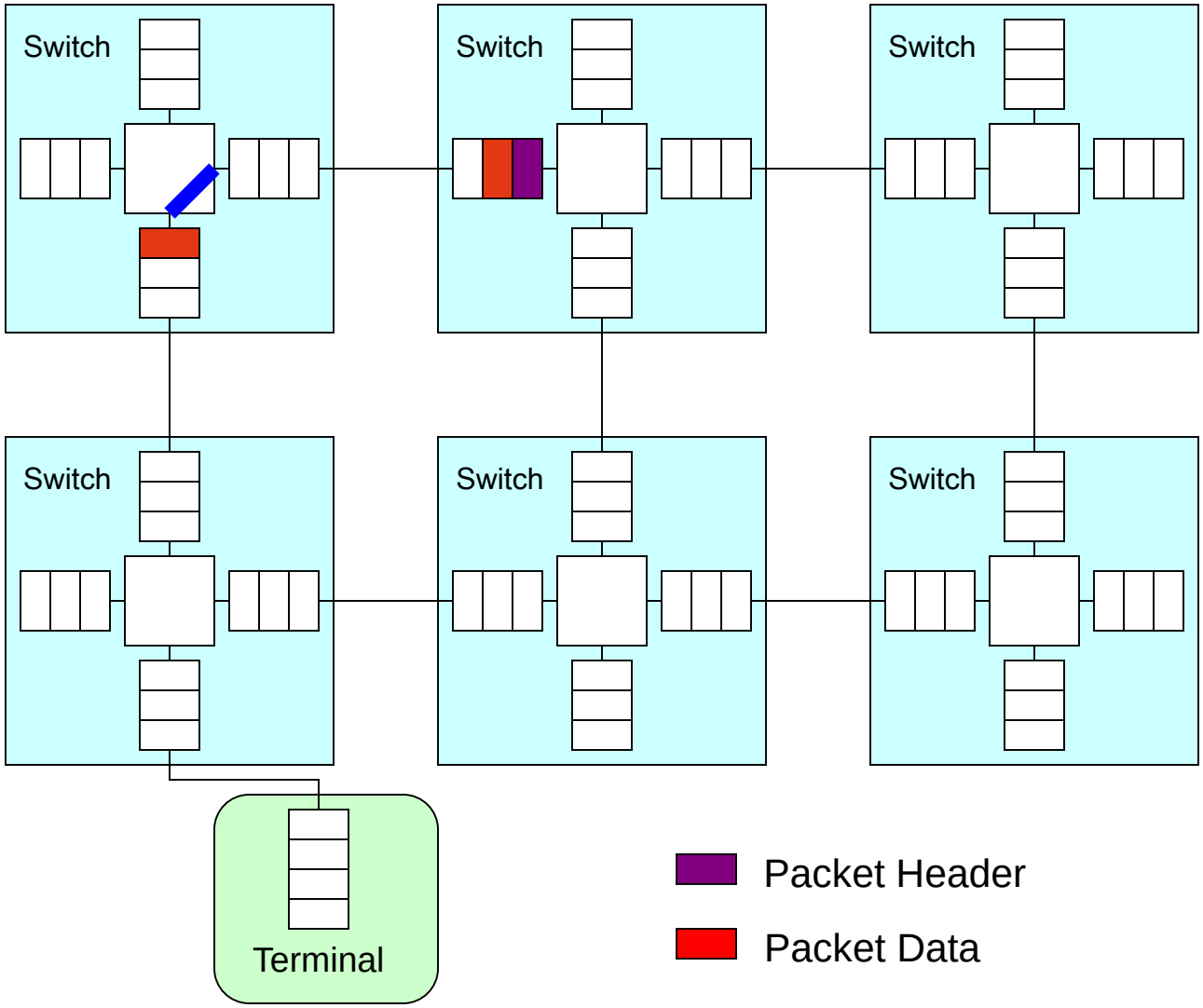
SAF Switching



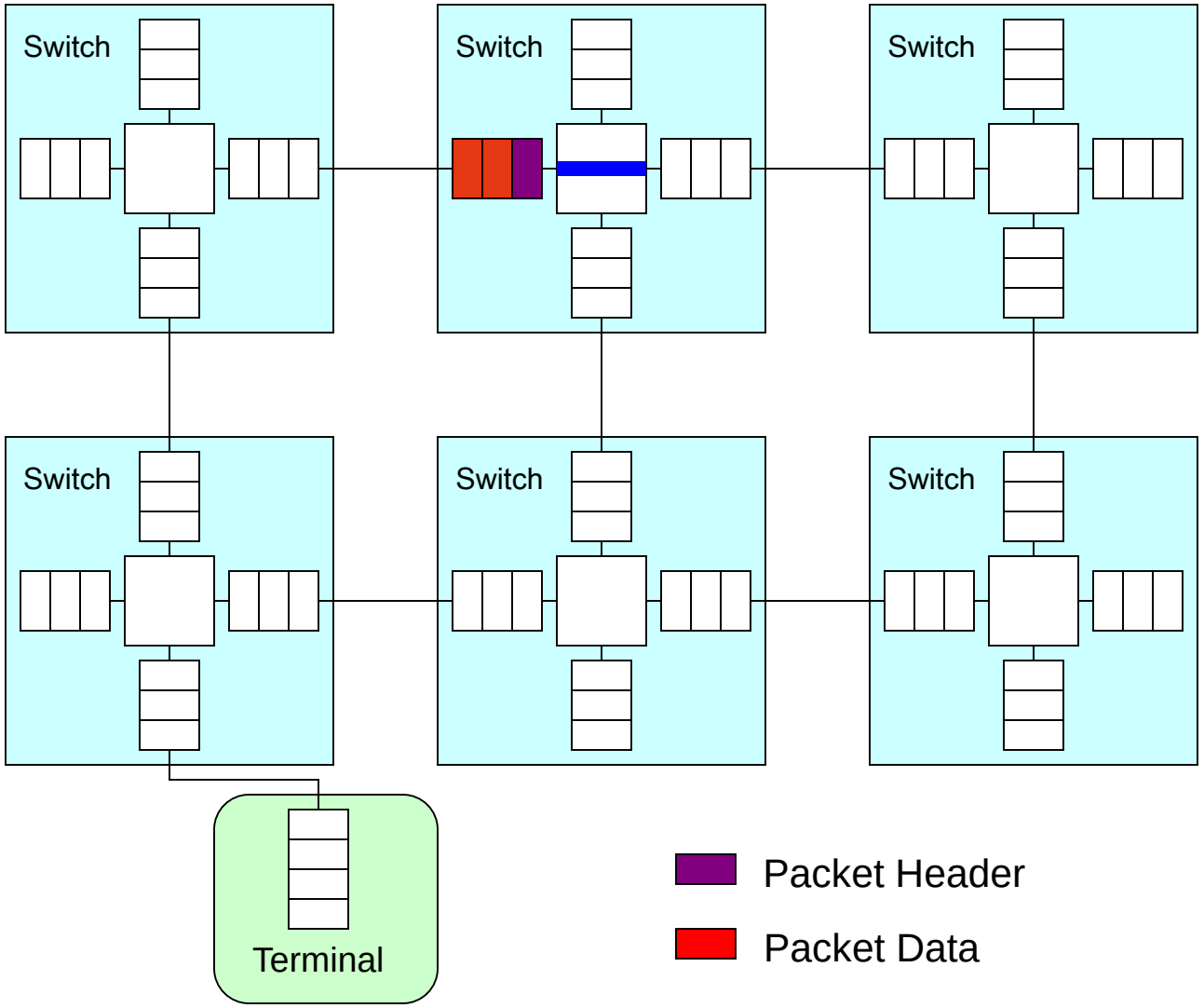
SAF Switching



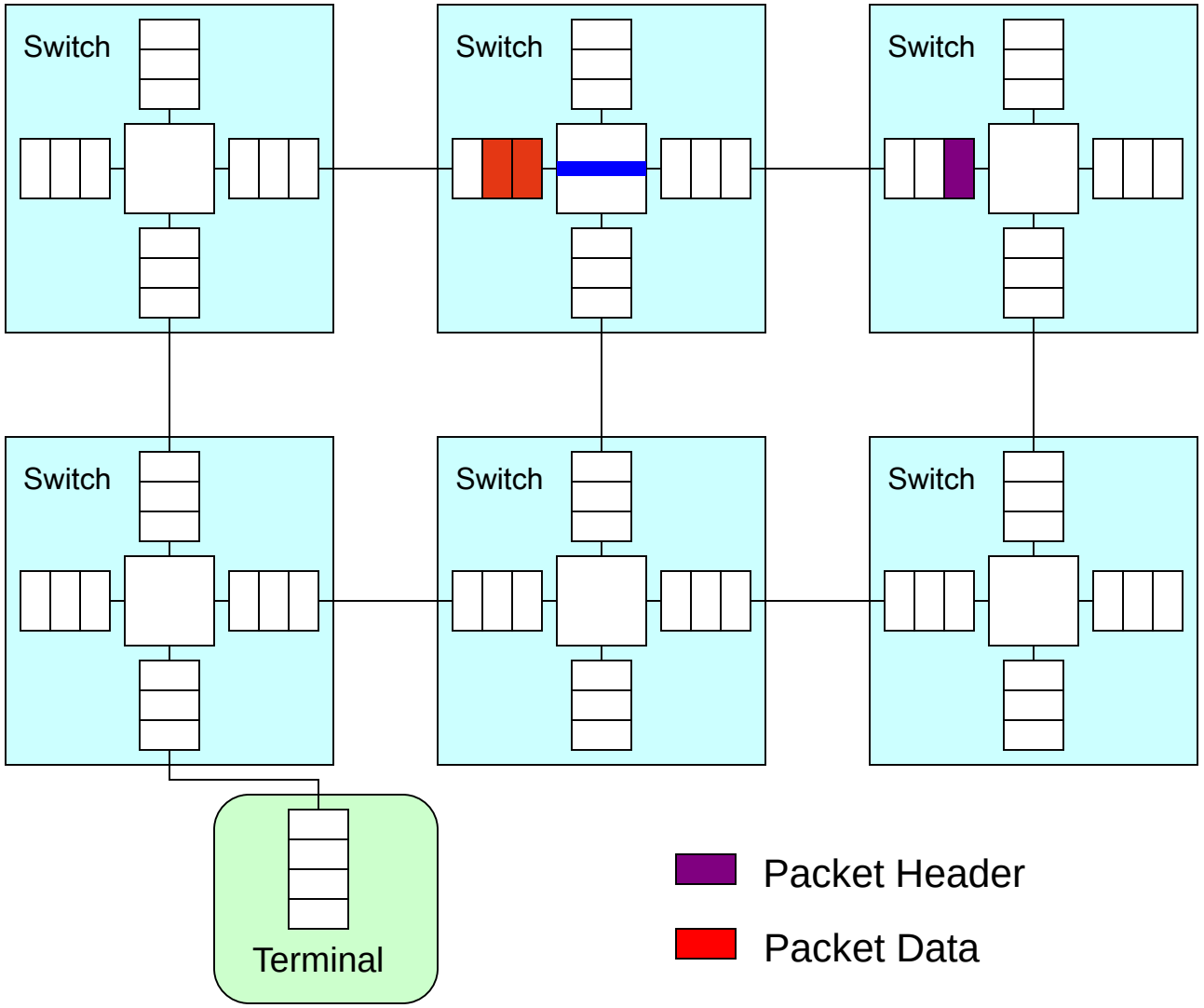
SAF Switching



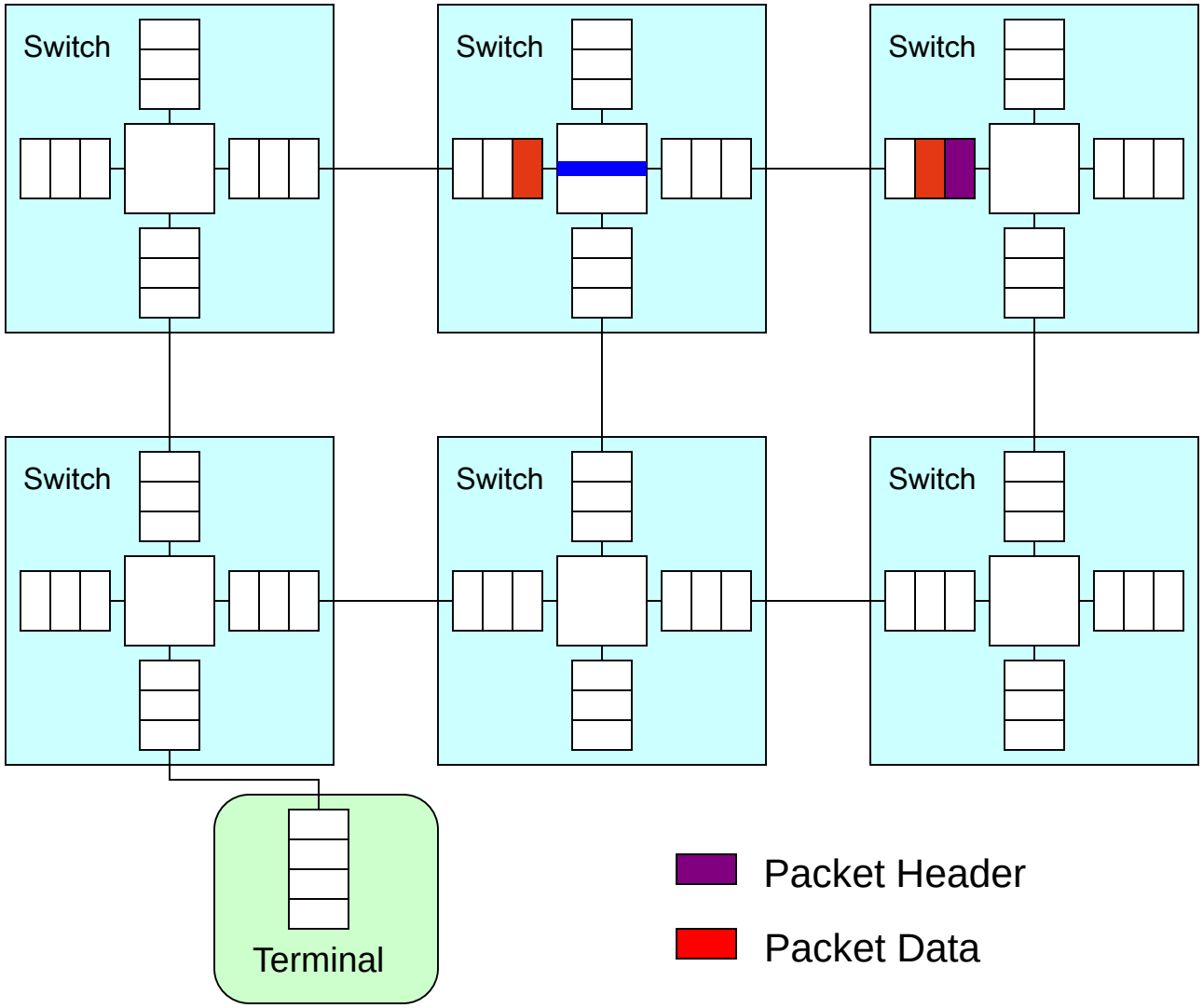
SAF Switching



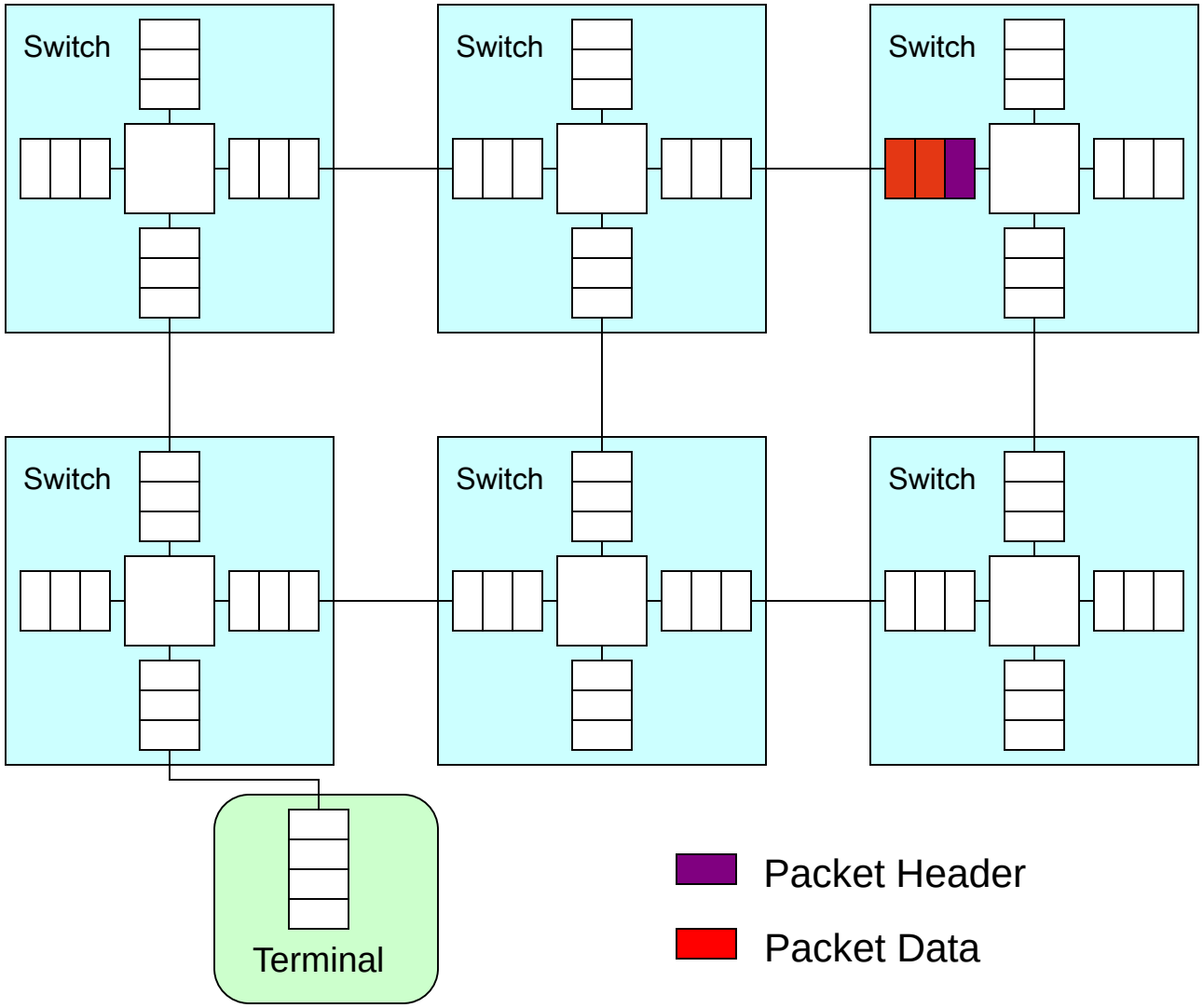
SAF Switching



SAF Switching

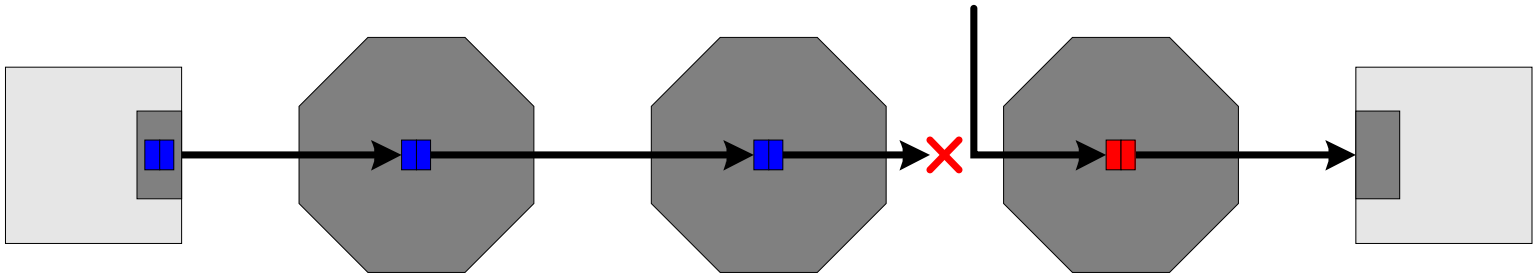


SAF Switching

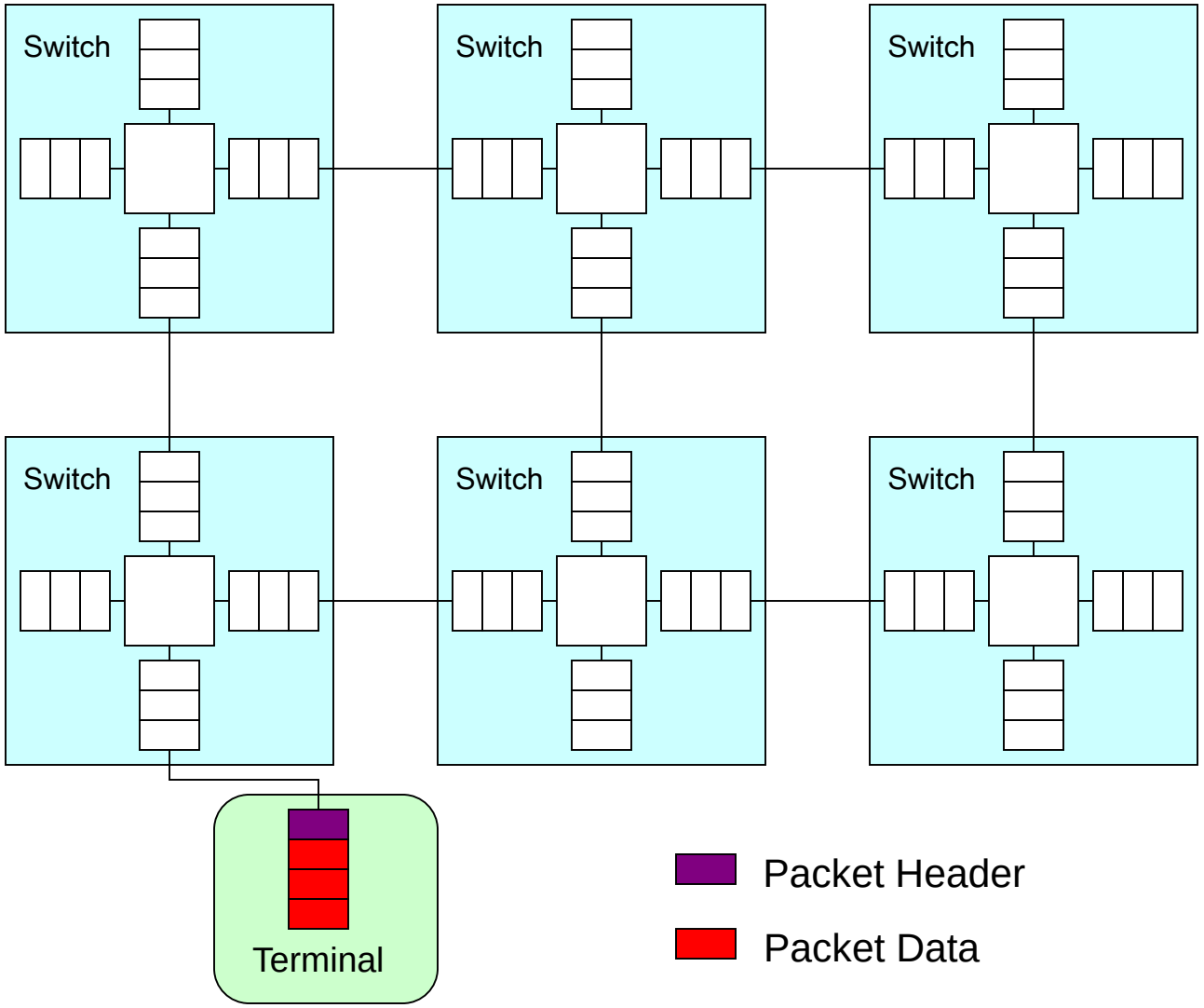


Wormhole switching

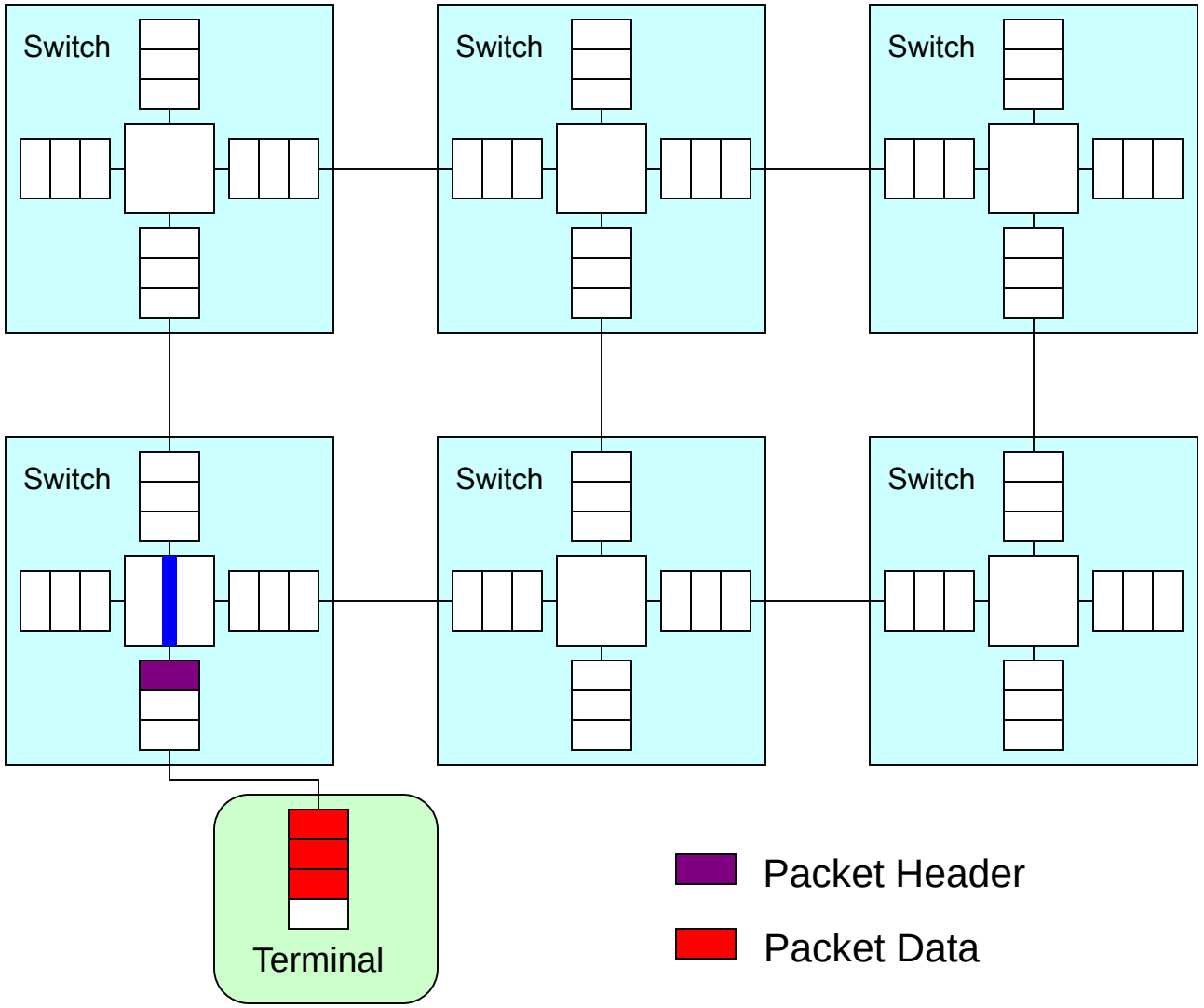
- Packet is routed and forwarded as soon the header flit has arrived
  - payload flits follow header
- Input ports does not need to buffer a complete packet
  - flits of a packet can be stored across multiple routers
- Trade-off between buffer overheads and network contention



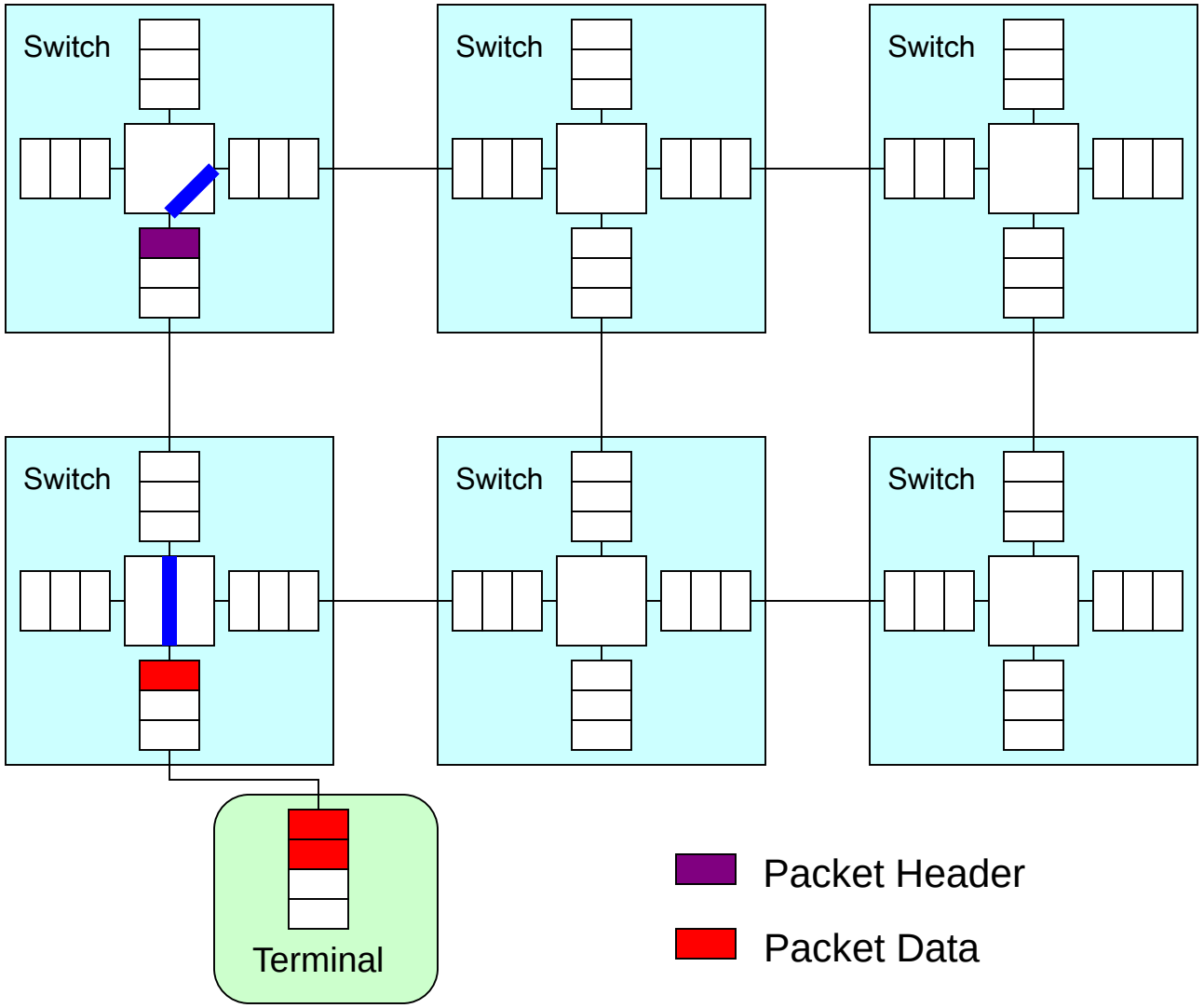
Wormhole Switching



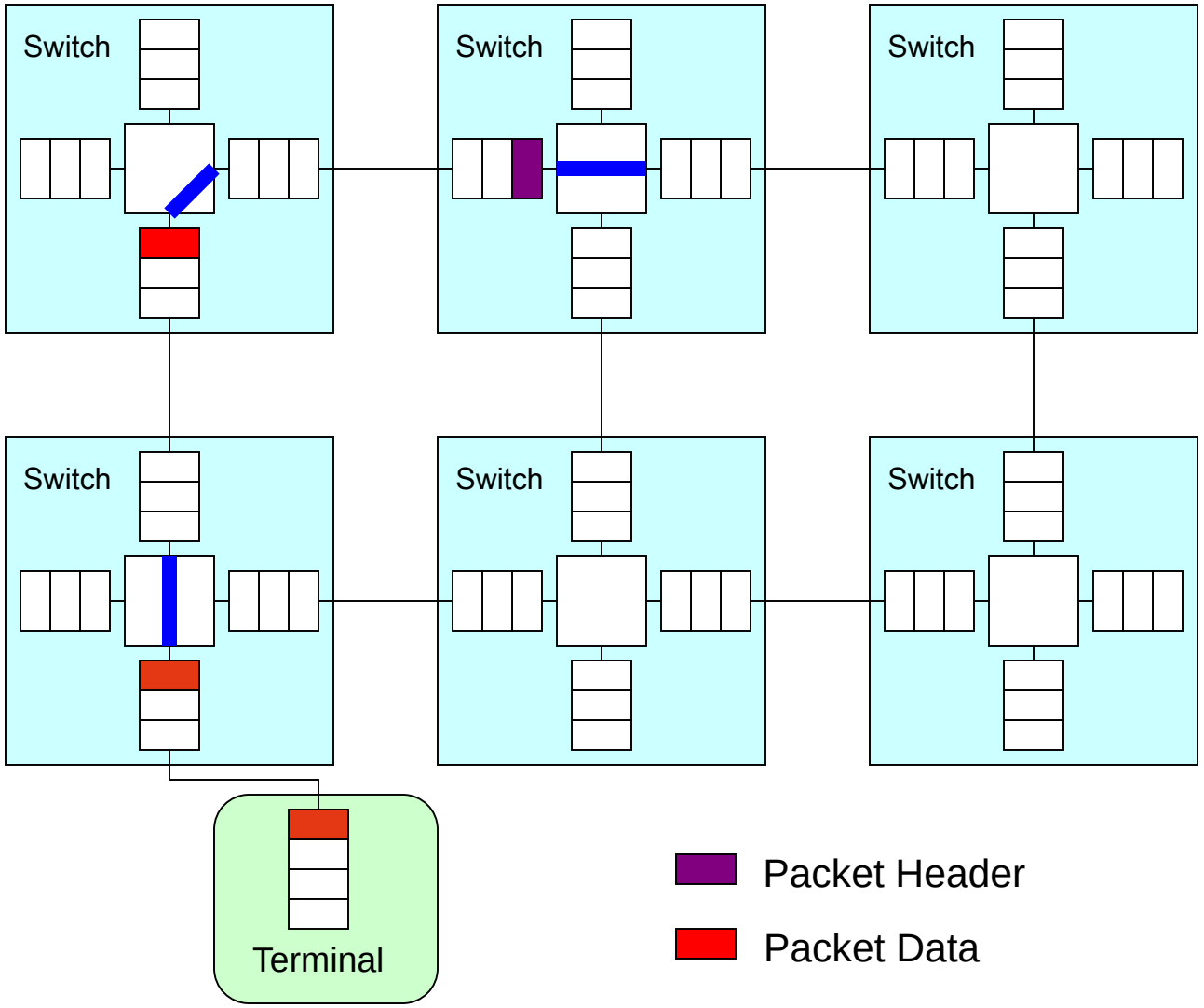
Wormhole Switching



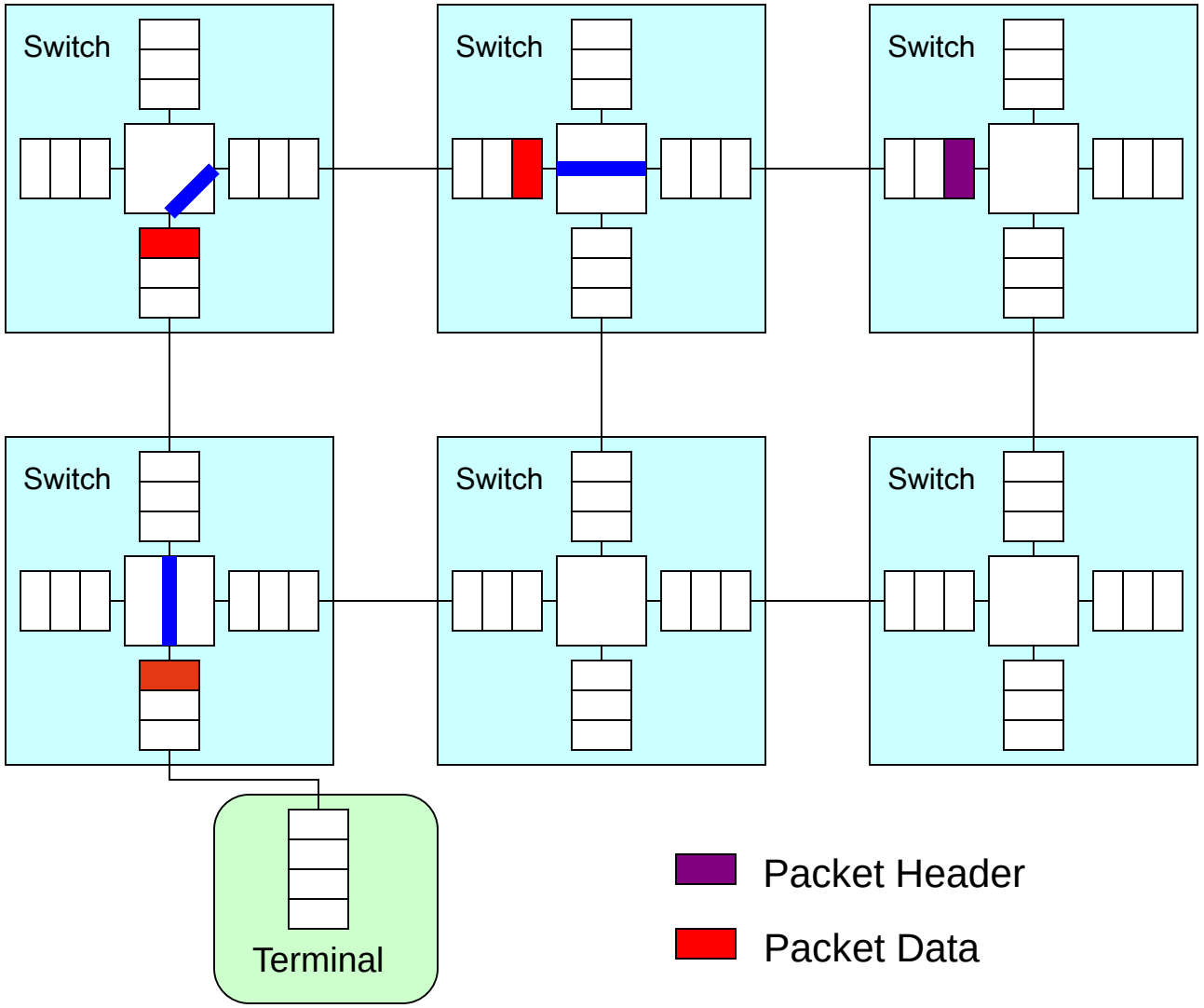
Wormhole Switching



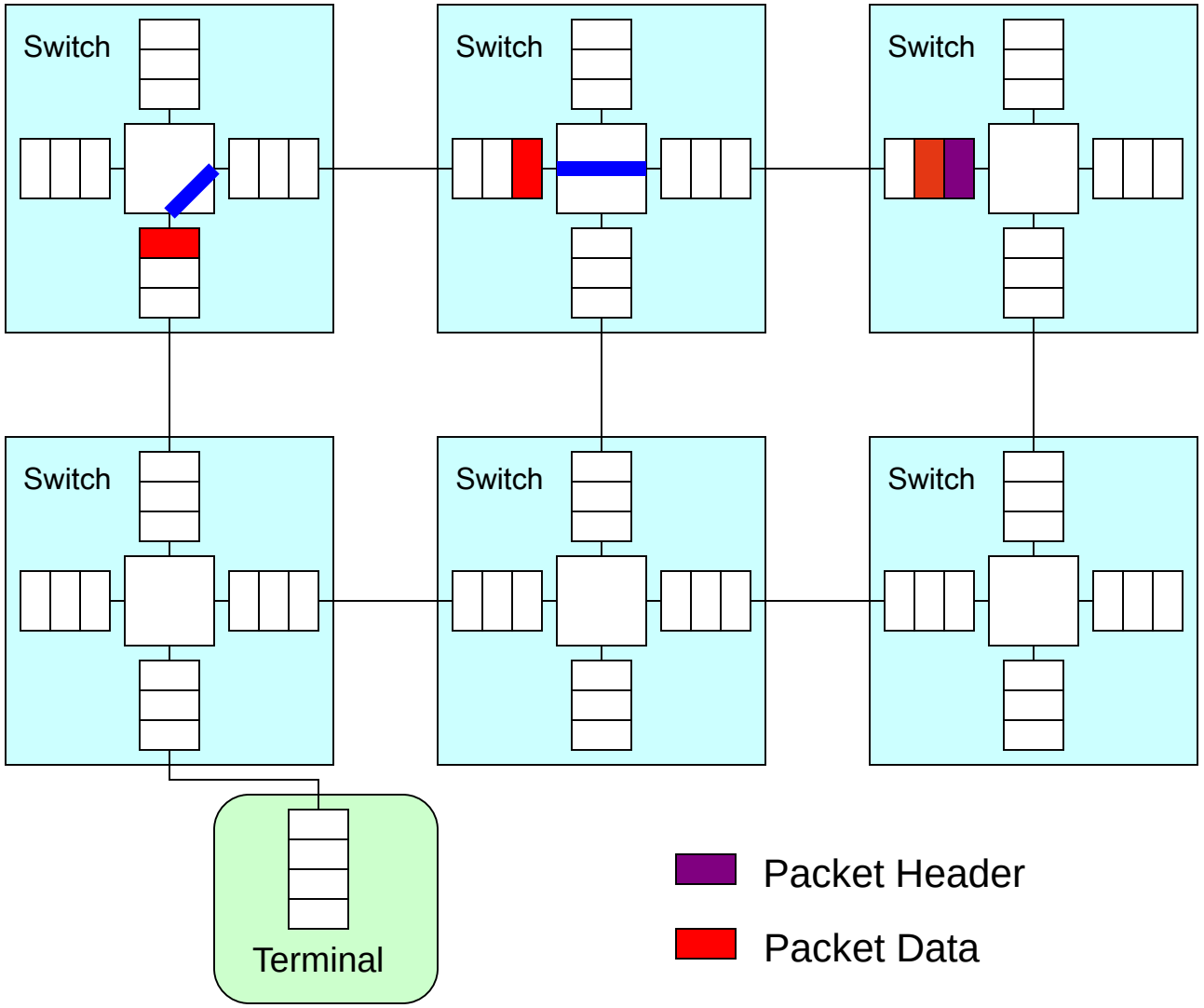
Wormhole Switching



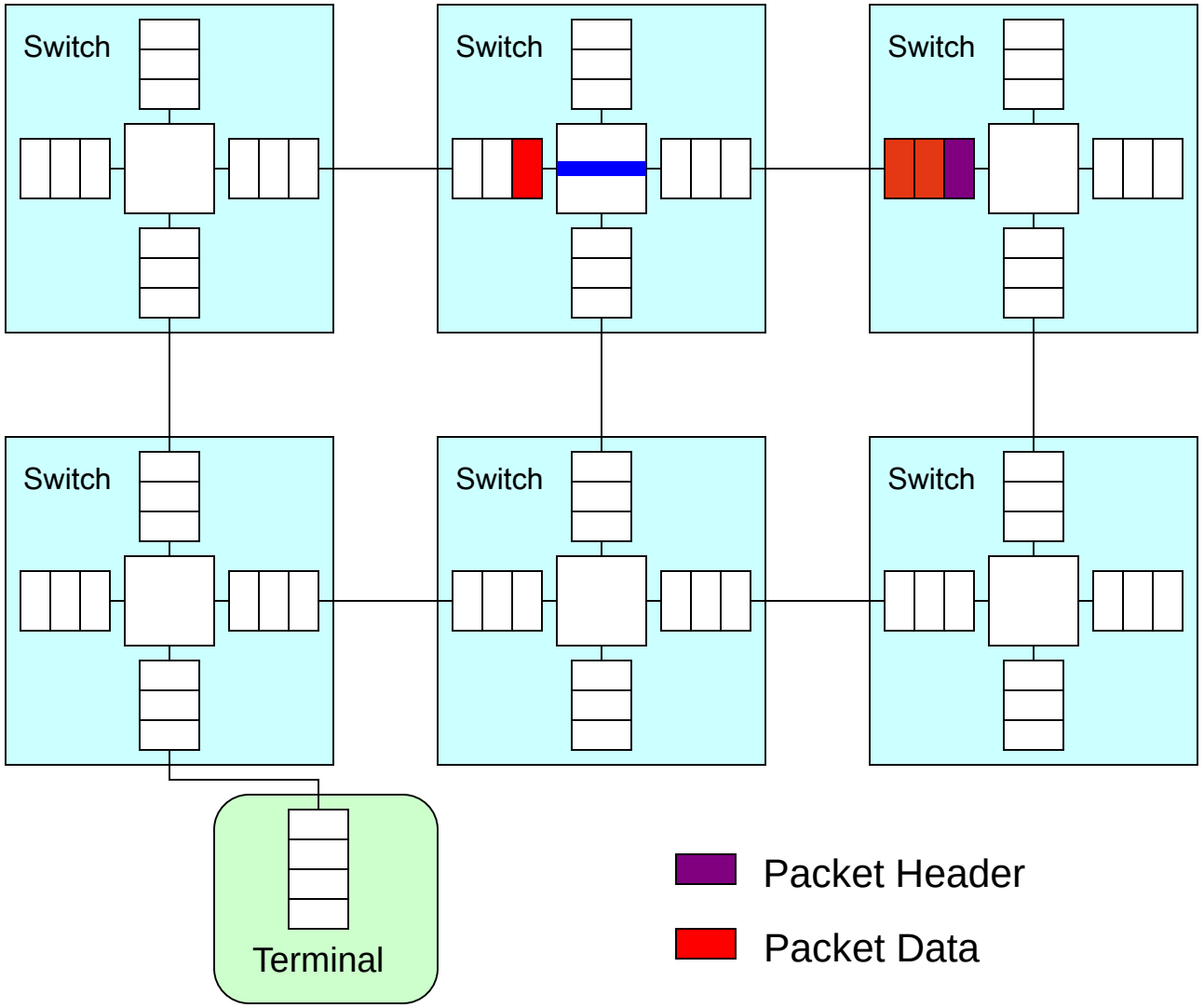
Wormhole Switching



Wormhole Switching

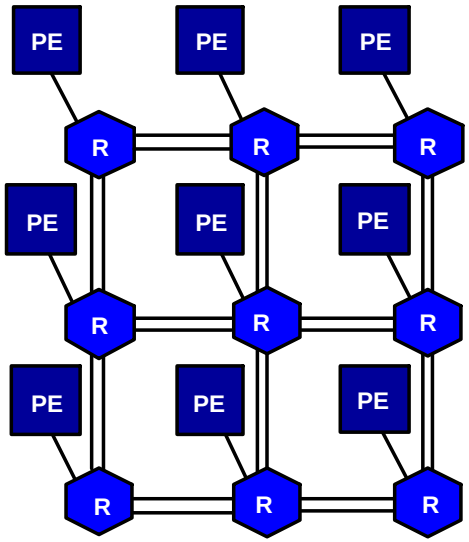


Wormhole Switching

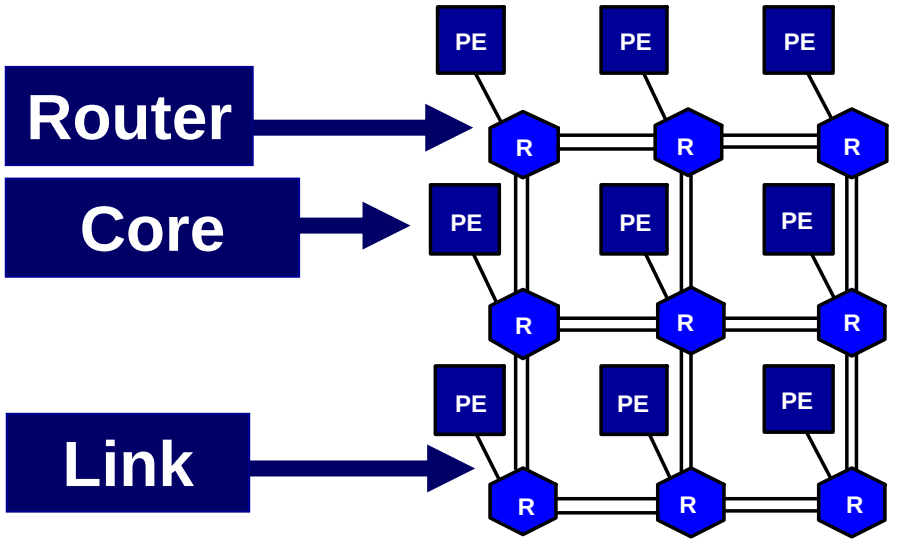


Wormhole Networks-on-Chip

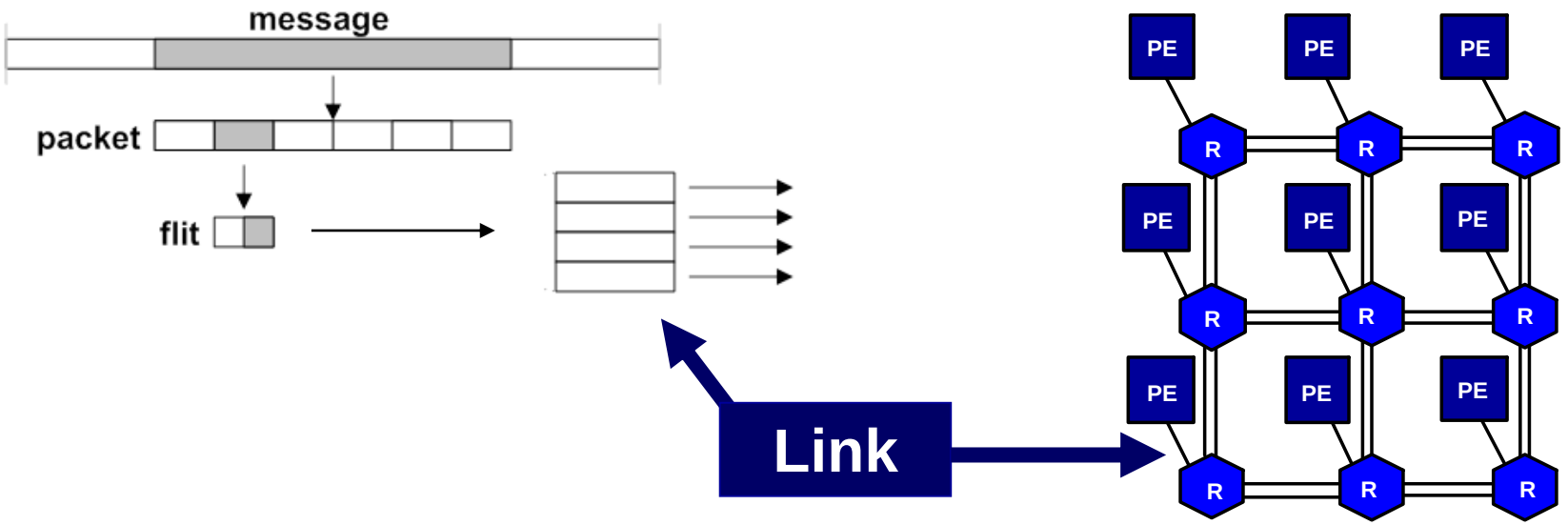
- Small buffering overheads of wormhole networks is particularly attractive to a special class of resource-constrained networks: Networks-on-Chip (NoCs)
  - small buffers mean smaller area and lower energy dissipation



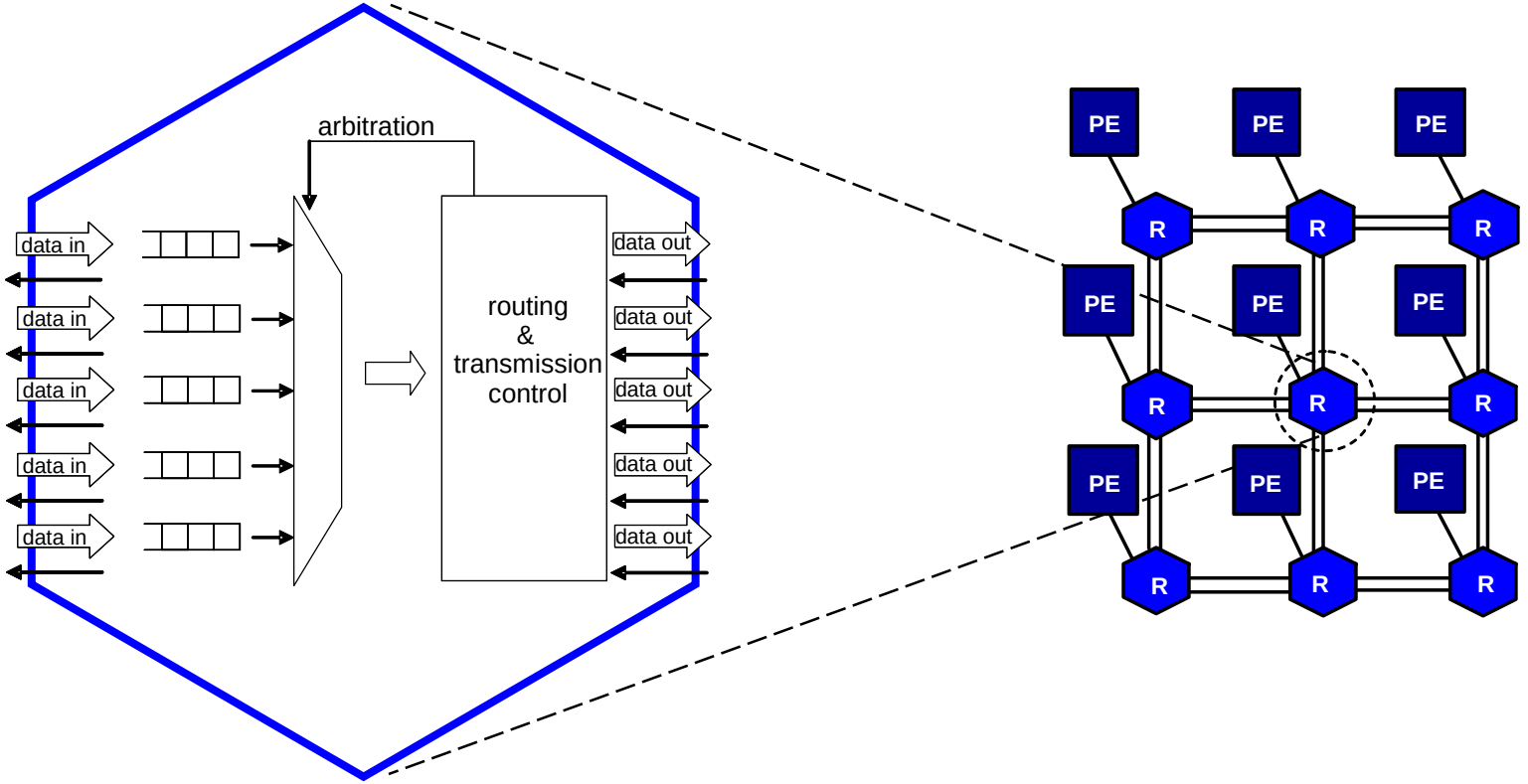
Wormhole Networks-on-Chip



Wormhole Networks-on-Chip

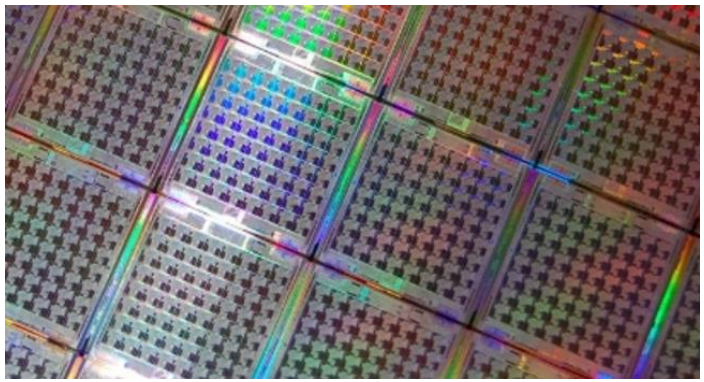


Wormhole Networks-on-Chip



Wormhole Networks-on-Chip

TILERA



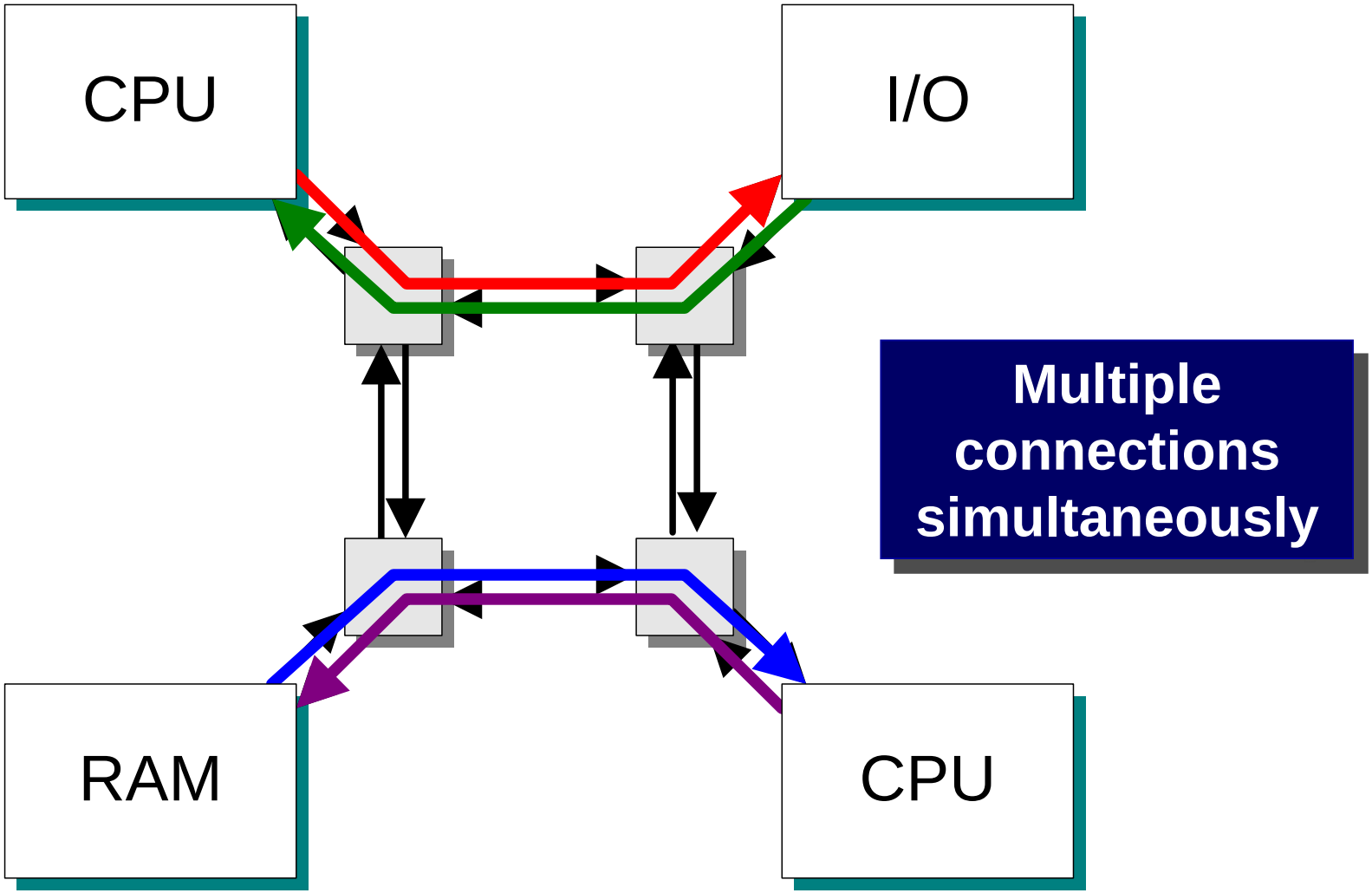
KALRAY

intel

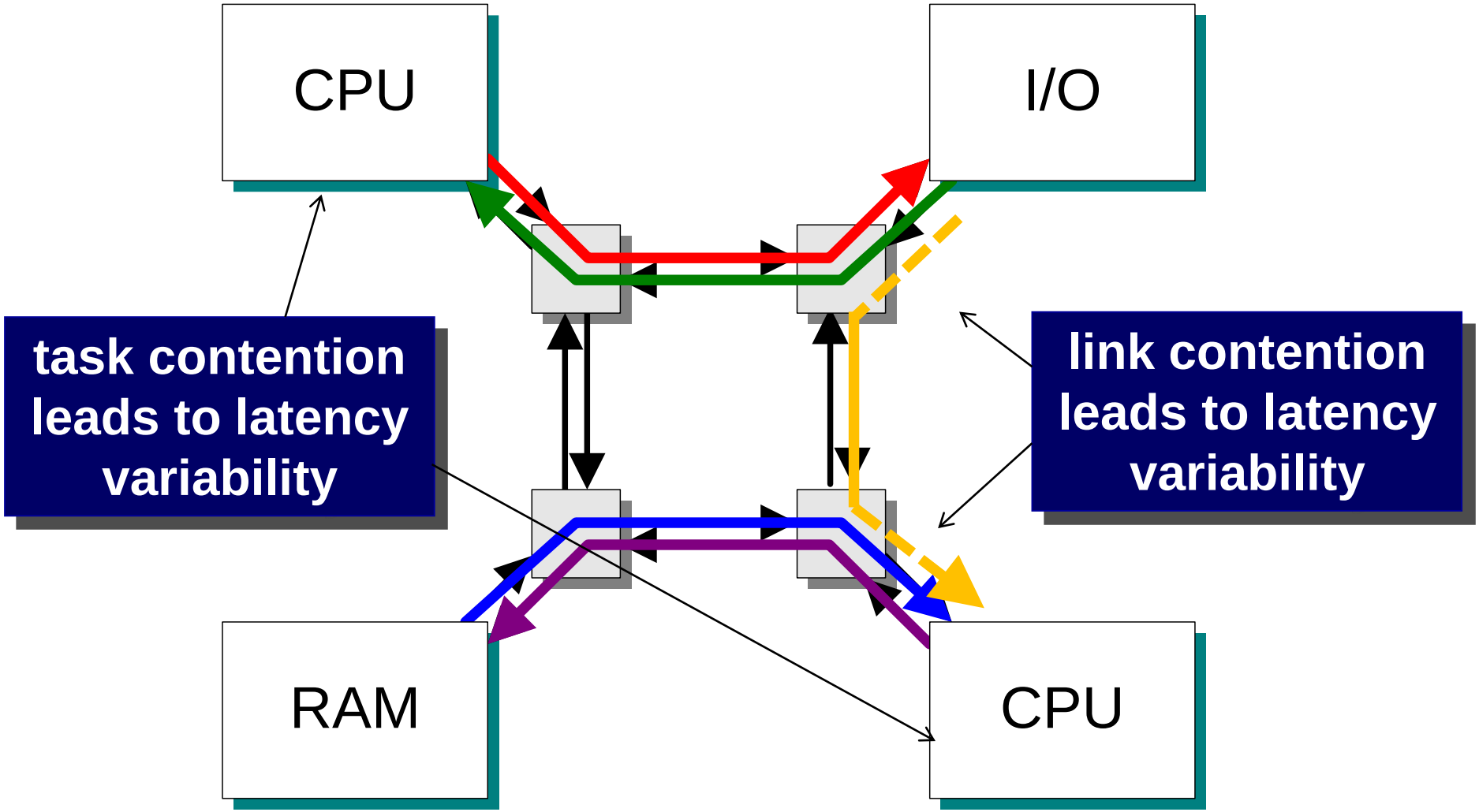
Arteris™
THE NETWORK-ON-CHIP COMPANY

Dream CHIP

NoC parallelism and scalability

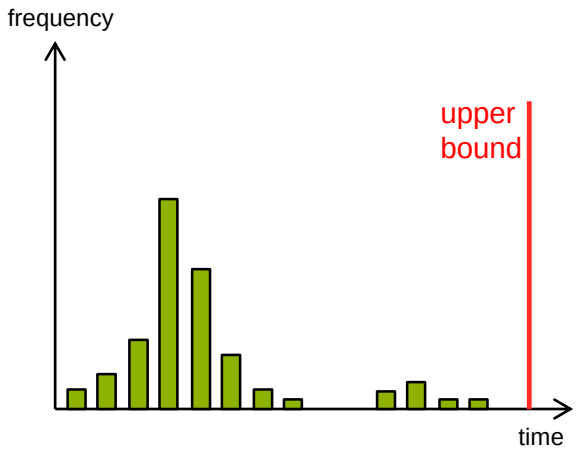


NoC performance

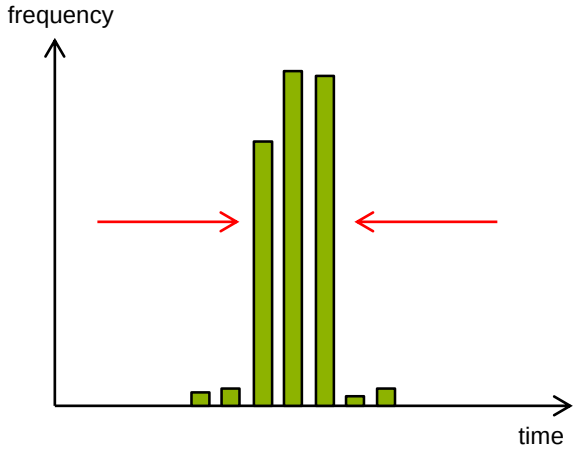


Time predictability in embedded NoCs

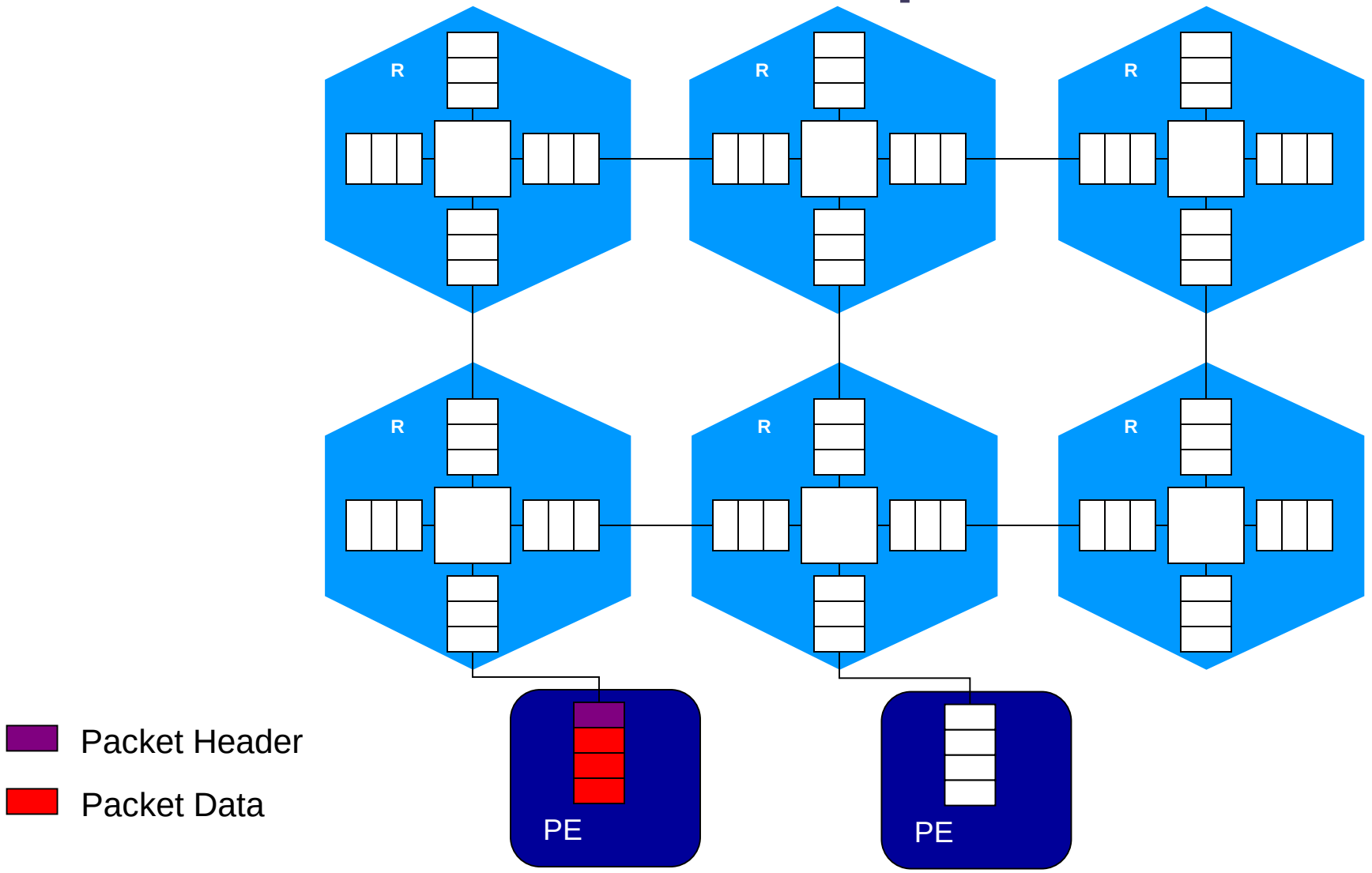
- Ability to guarantee an upper bound on the system's temporal behaviour
  - worst-case response time of each task
  - worst-case latency of each NoC packet
  - worst-case end-to-end latencies of communicating task chains



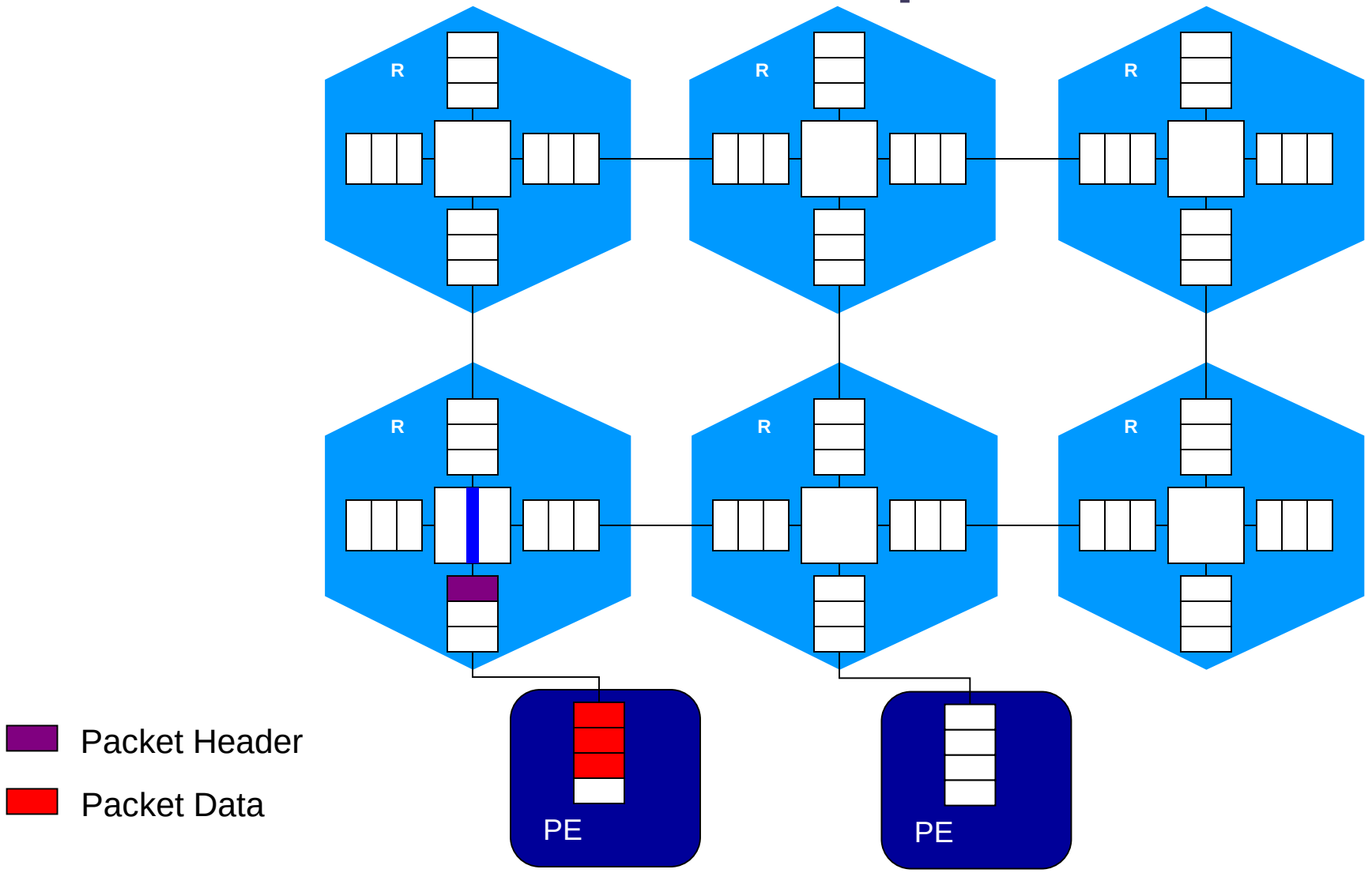
- Ability to constrain the variability of the system's temporal behaviour
  - limited best/worst case difference



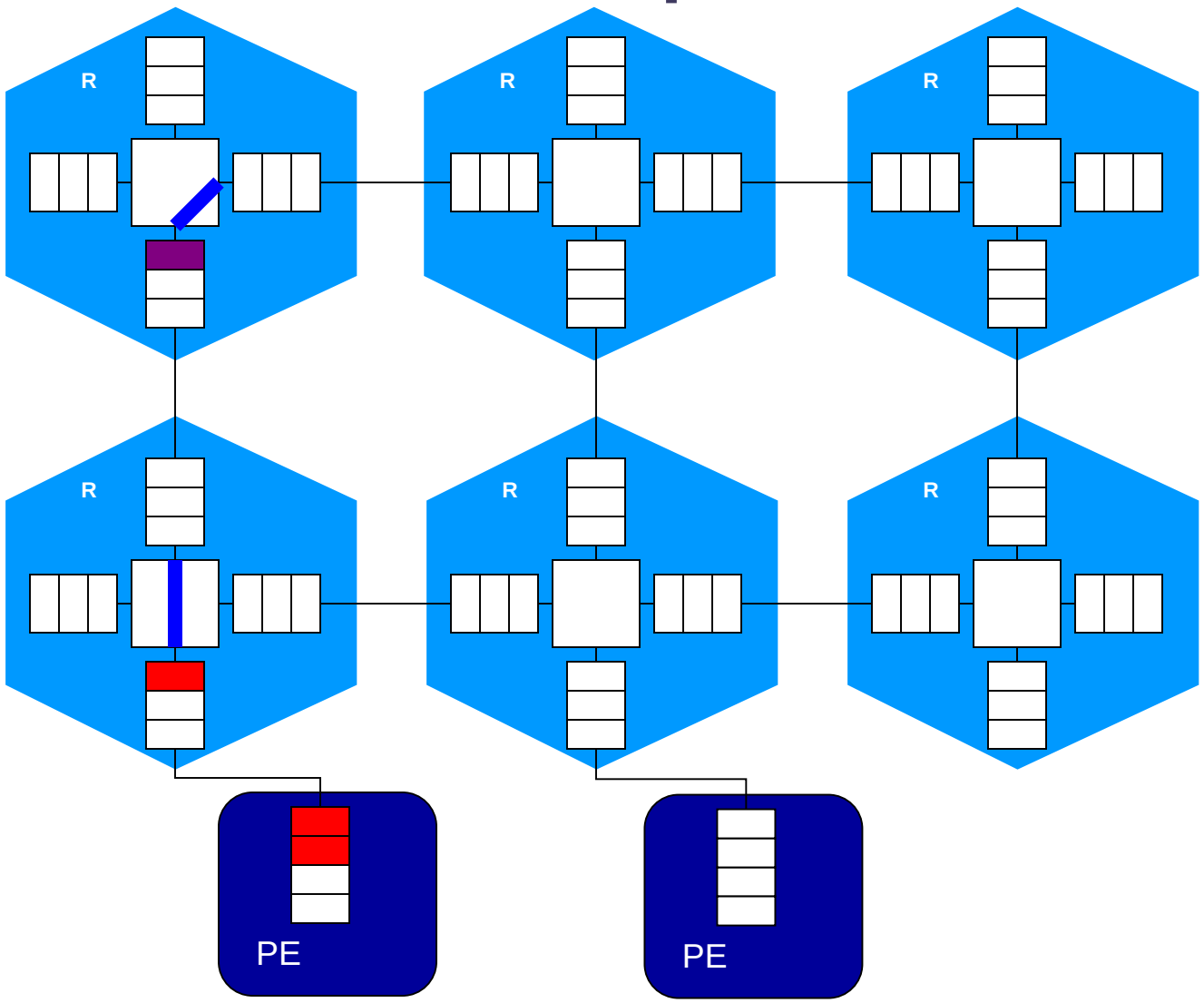
Wormhole Networks-on-Chip



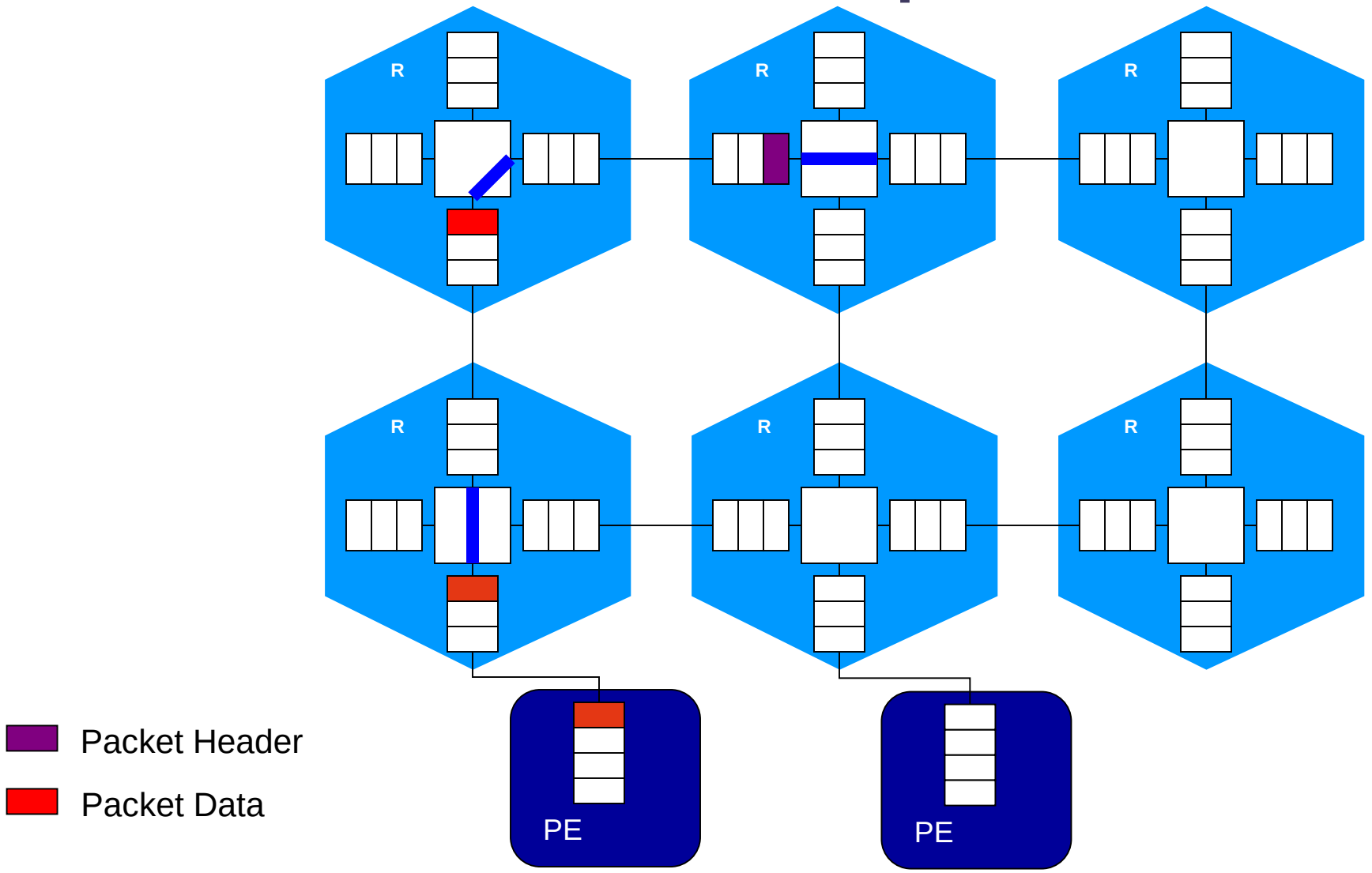
Wormhole Networks-on-Chip



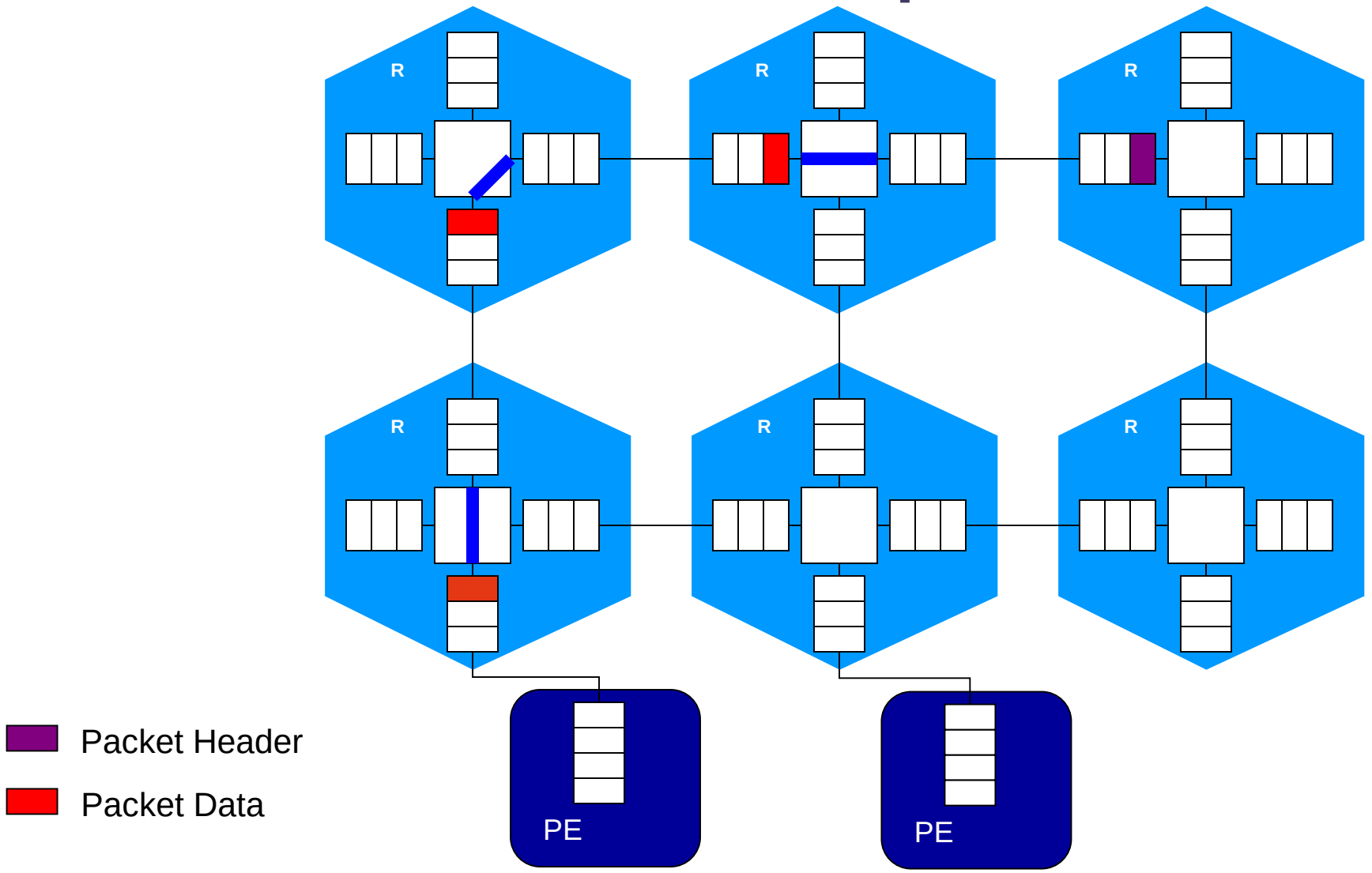
Wormhole Networks-on-Chip



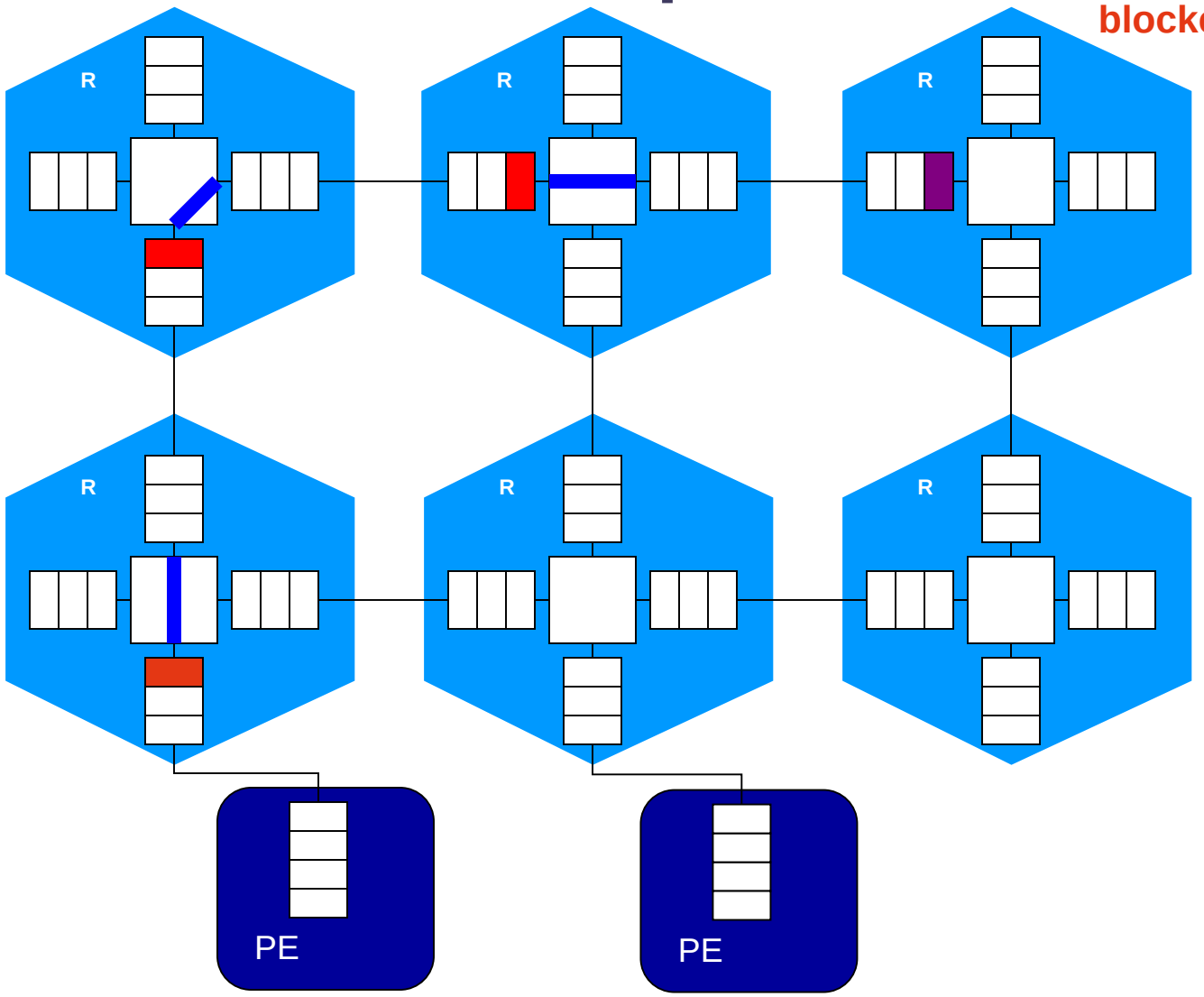
Wormhole Networks-on-Chip



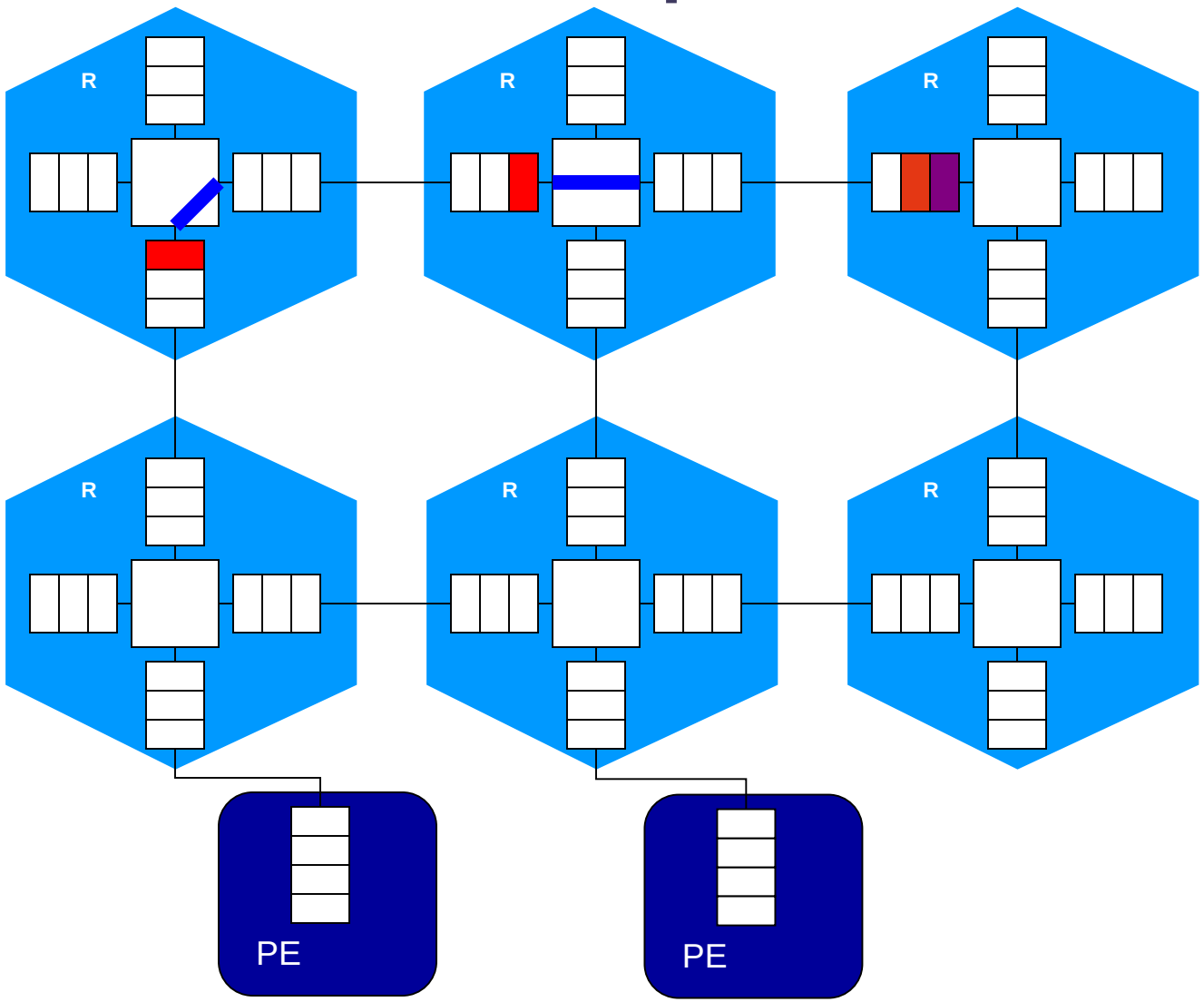
Wormhole Networks-on-Chip



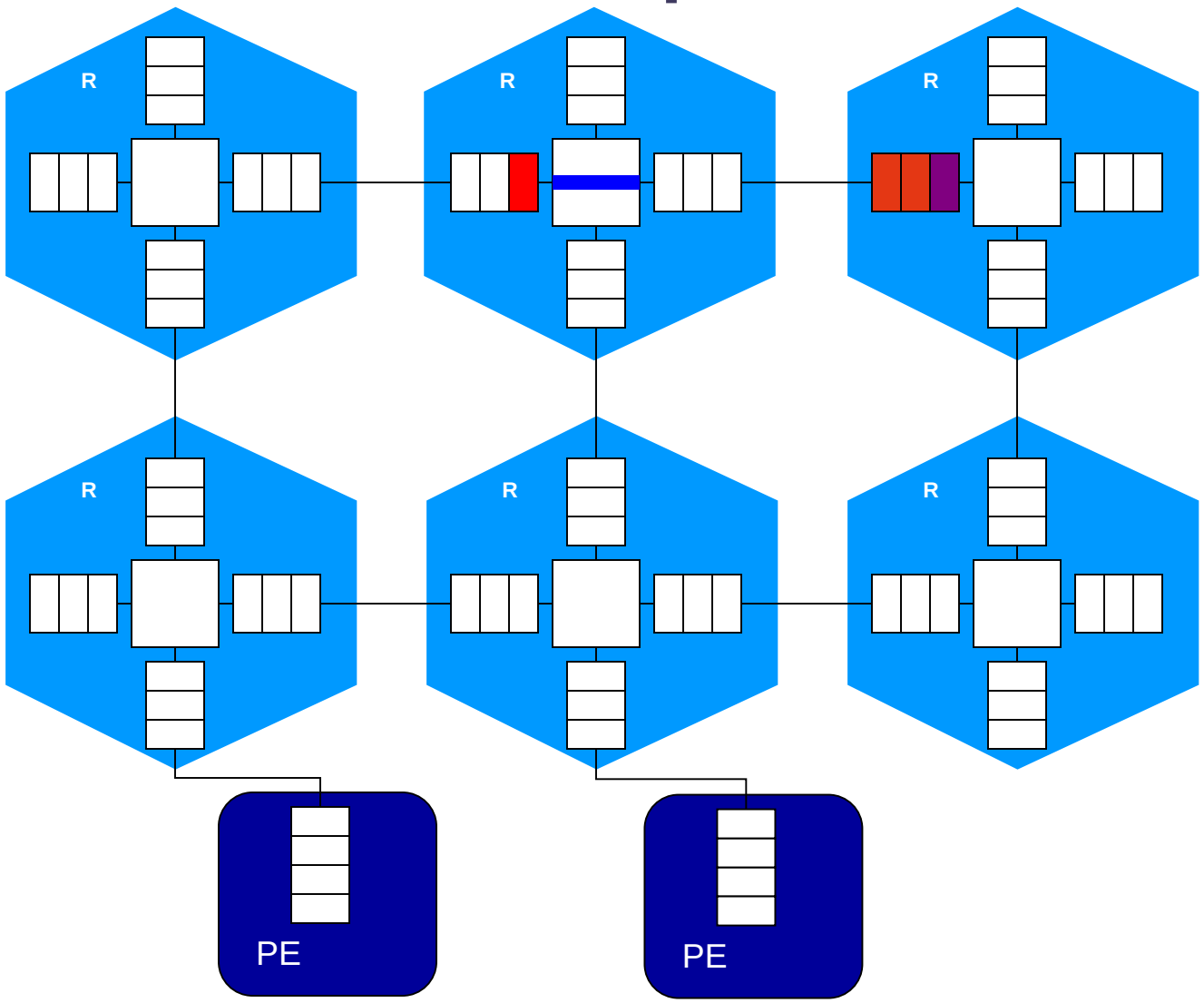
Wormhole Networks-on-Chip



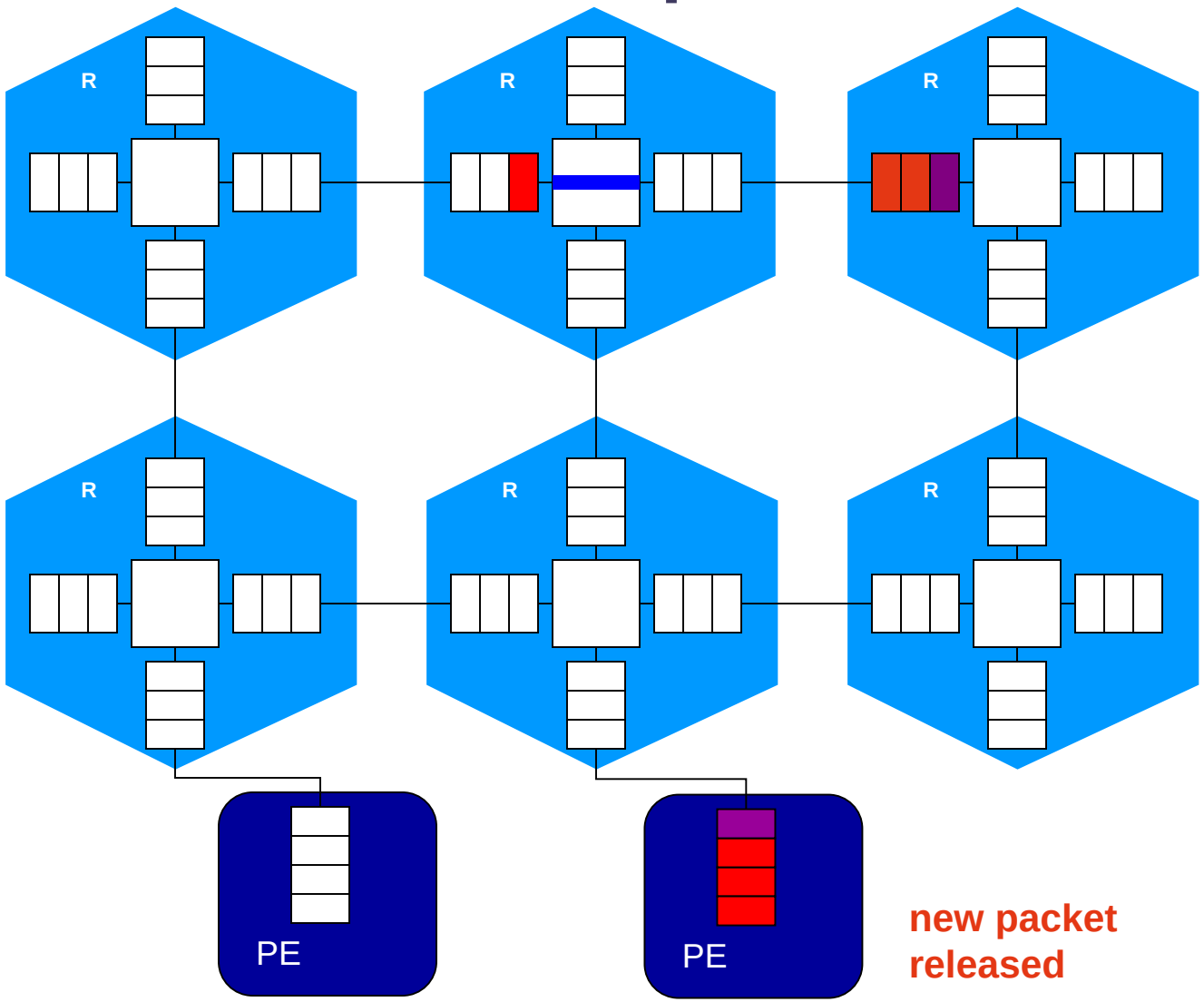
Wormhole Networks-on-Chip



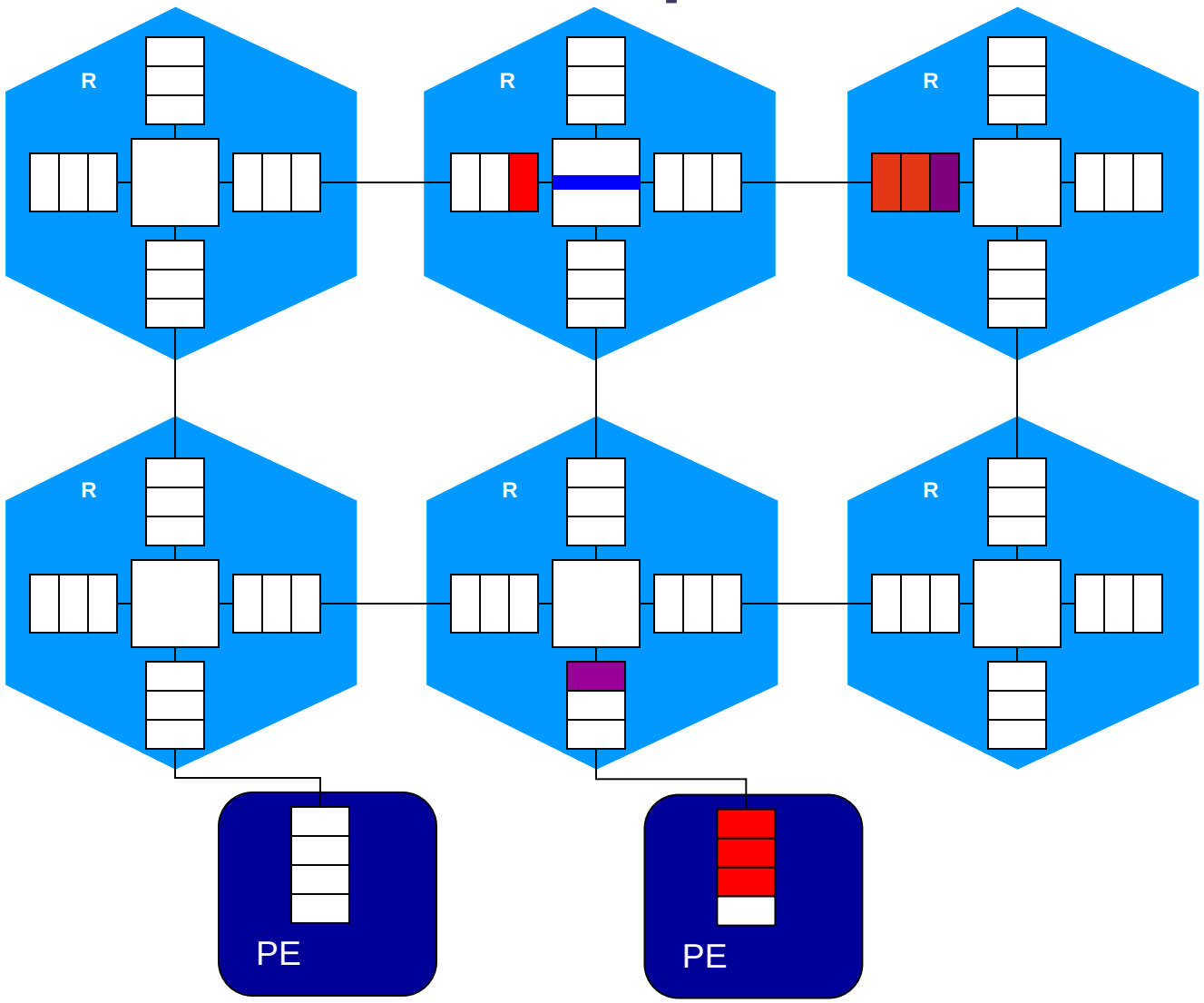
Wormhole Networks-on-Chip



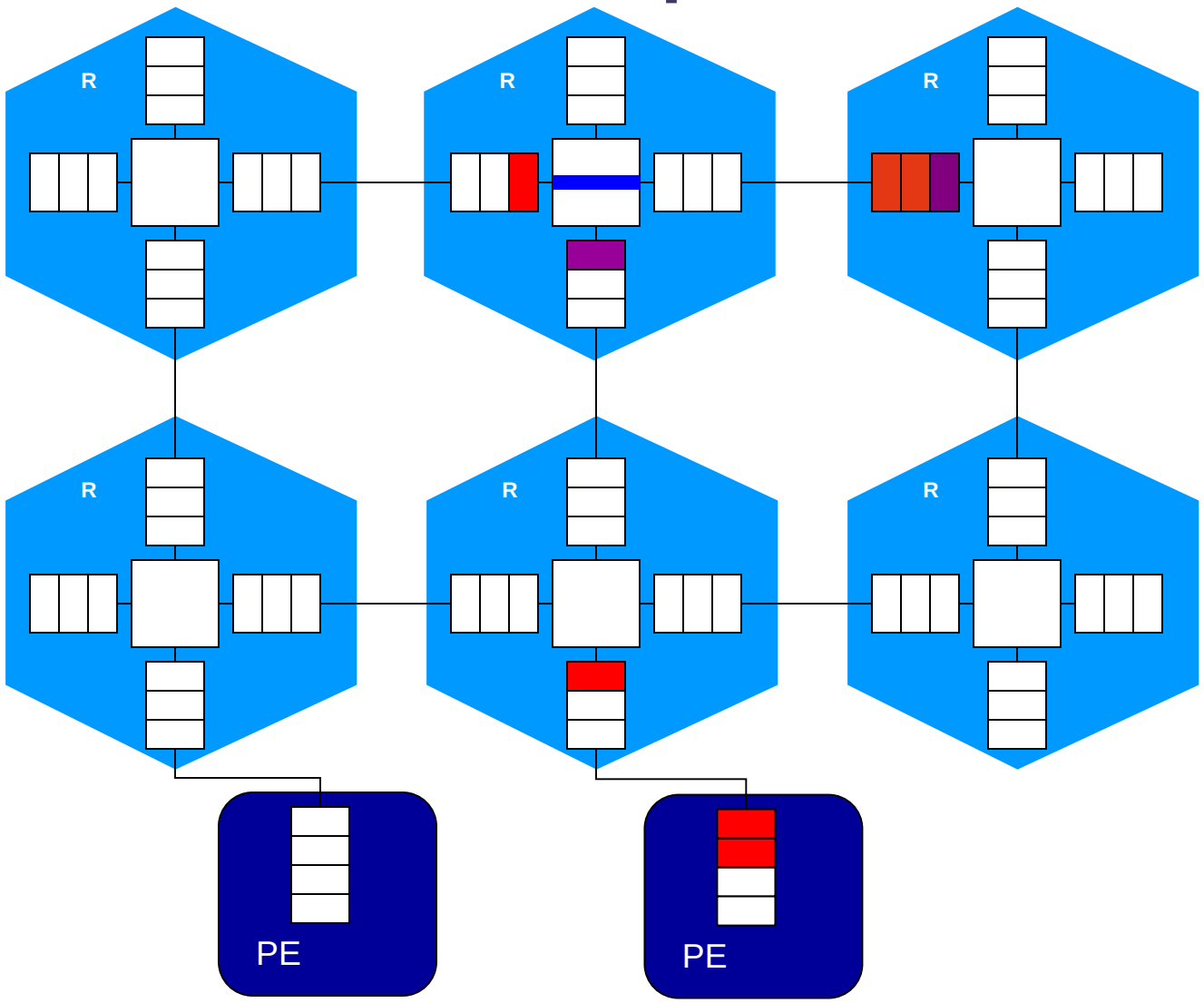
Wormhole Networks-on-Chip



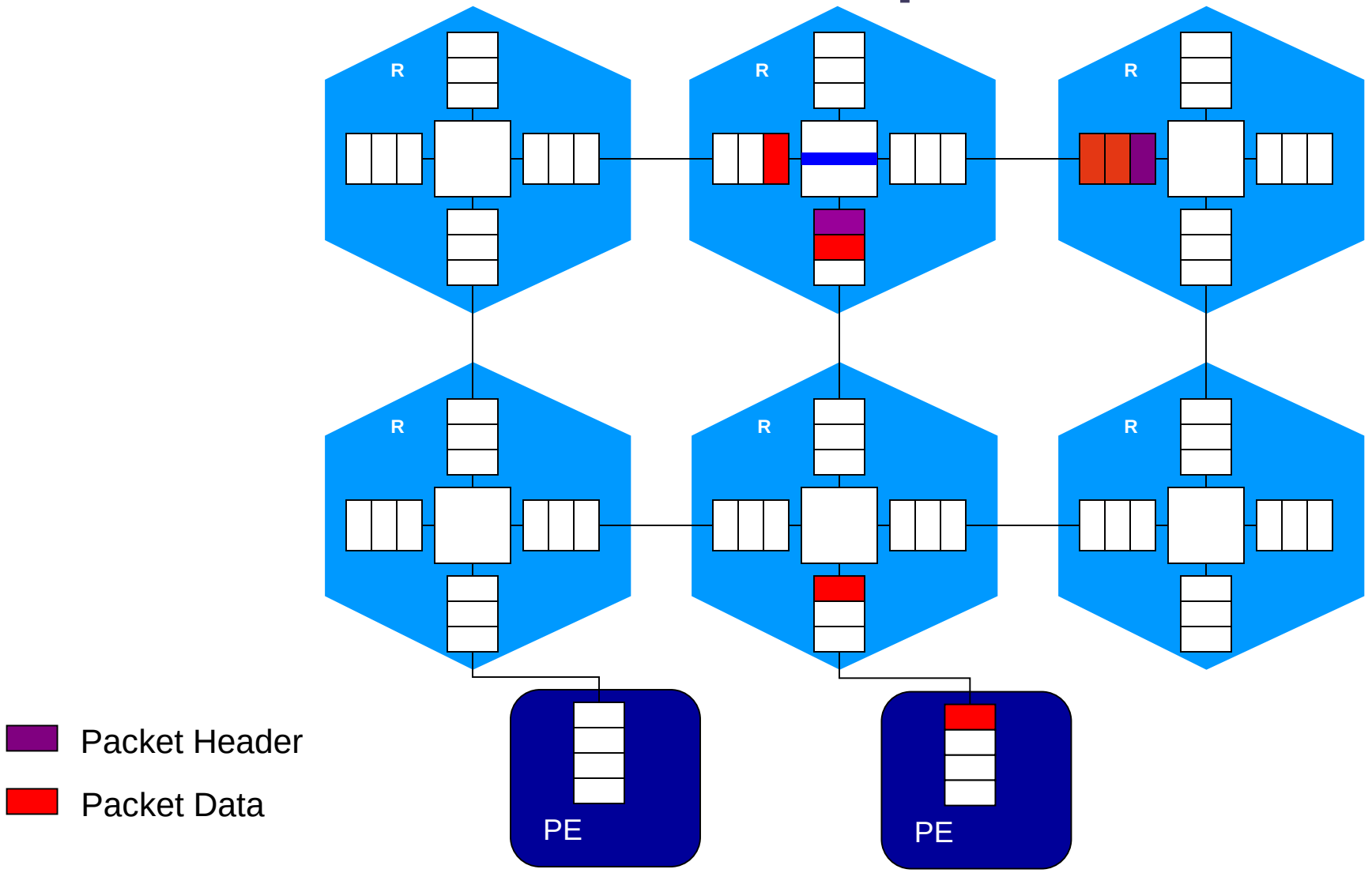
Wormhole Networks-on-Chip



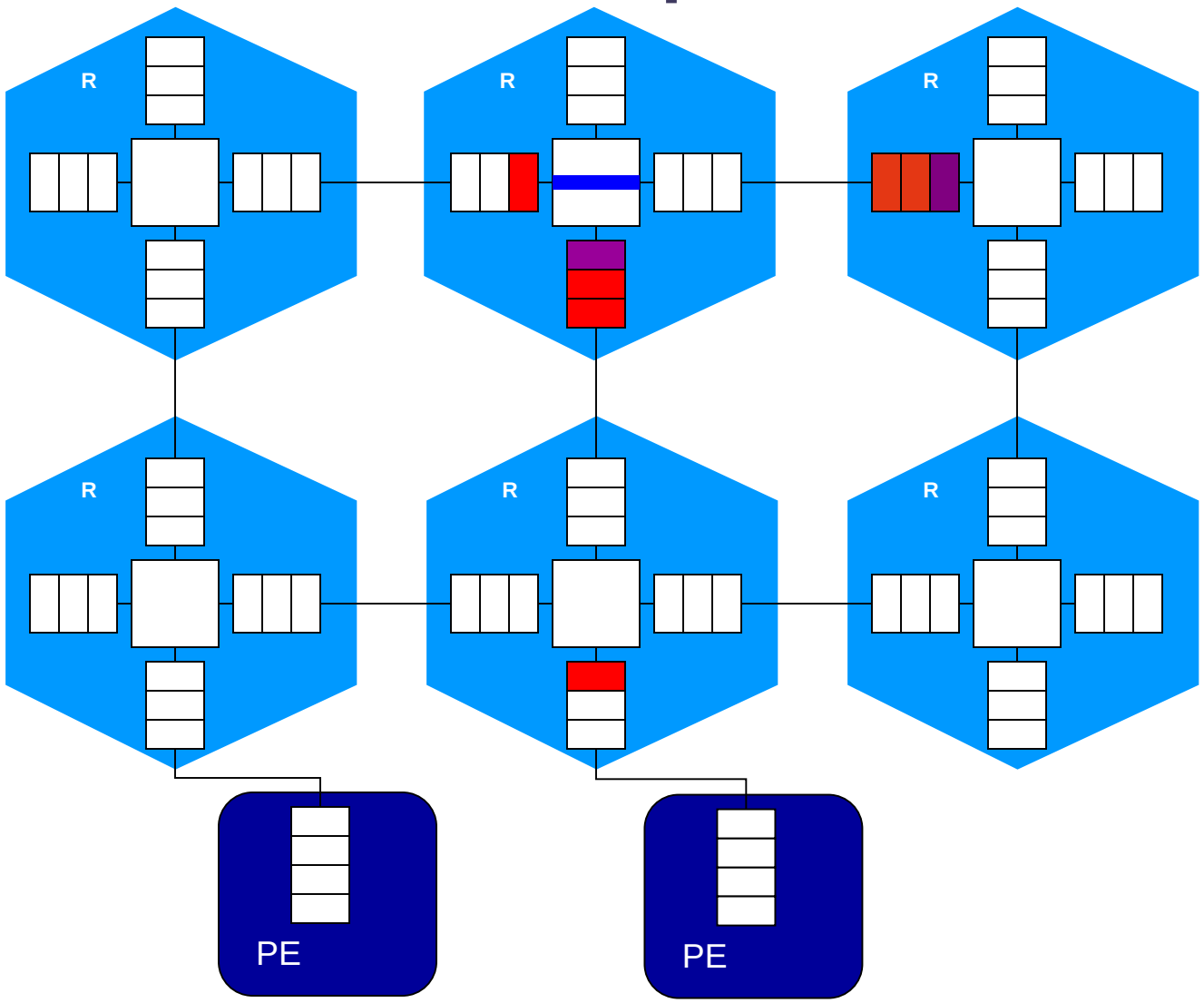
Wormhole Networks-on-Chip



Wormhole Networks-on-Chip



Wormhole Networks-on-Chip



Performance guarantees in embedded NoCs

- As the core counts increase, NoC link contention tends to me the dominant source of latency variability

- Current solutions
  - Full traffic separation (i.e. no link contention)
    - deterministic routing, fully disjoint routes (e.g. Hermes)
    - multiple overlay networks (e.g. Tilera)
      - contention over NIs and memory still possible
    - circuit switching (e.g. PNoC)
      - unpredictable circuit setup time
    - very low utilisation
    - state of the art: mixed criticality, virtual traffic separation

Performance guarantees in embedded NoCs

- As the core counts increase, NoC link contention tends to me the dominant source of latency variability

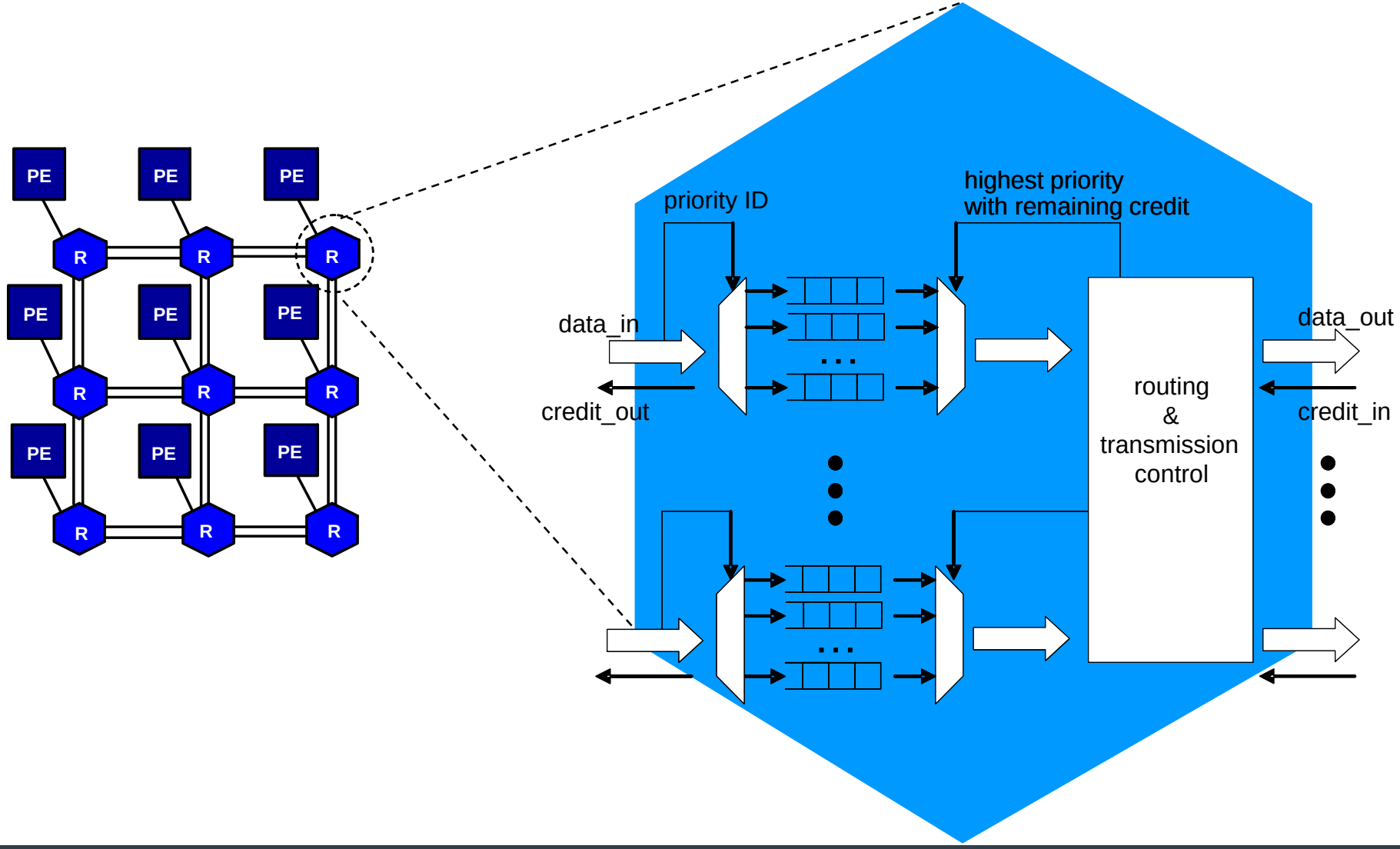
- Current solutions
  - Virtual traffic separation
    - time-division multiplexing (TDM)
      - fixed traffic slotting (e.g. Aethereal, AElite)
    - round-robin (RR)
      - rate controlling (e.g. Kalray, Nostrum, IDAMC)
    - fixed-priority (FP)
      - priority-arbitrated virtual channels (e.g. QNoC)

Priority preemptive virtual channels

- Wormhole NoCs using virtual channels with priority preemptive arbitration can discriminate packets of different levels of urgency

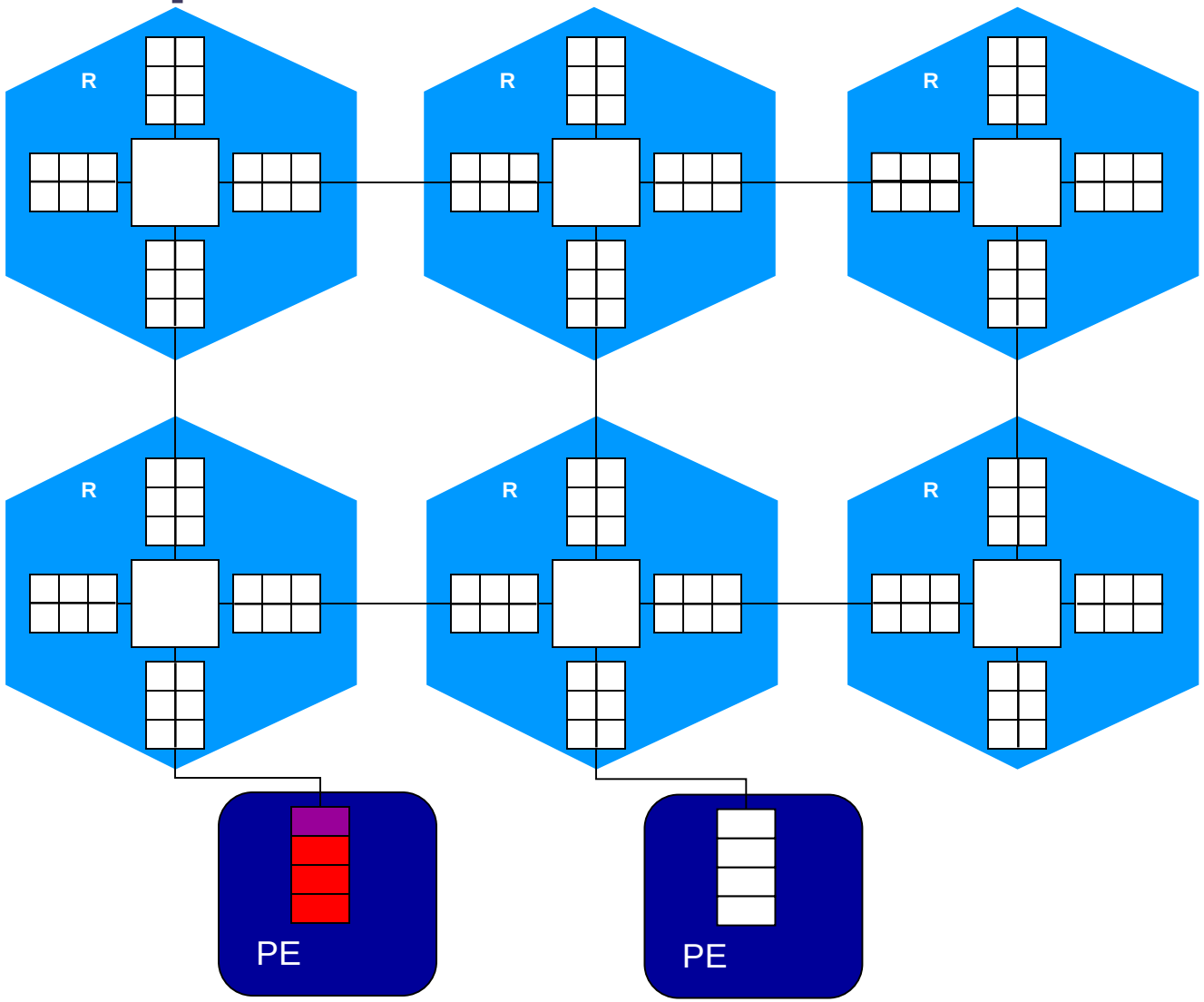
- Matches previous work on schedulability analysis in priority-preemptive wormhole networks

Priority preemptive virtual channels



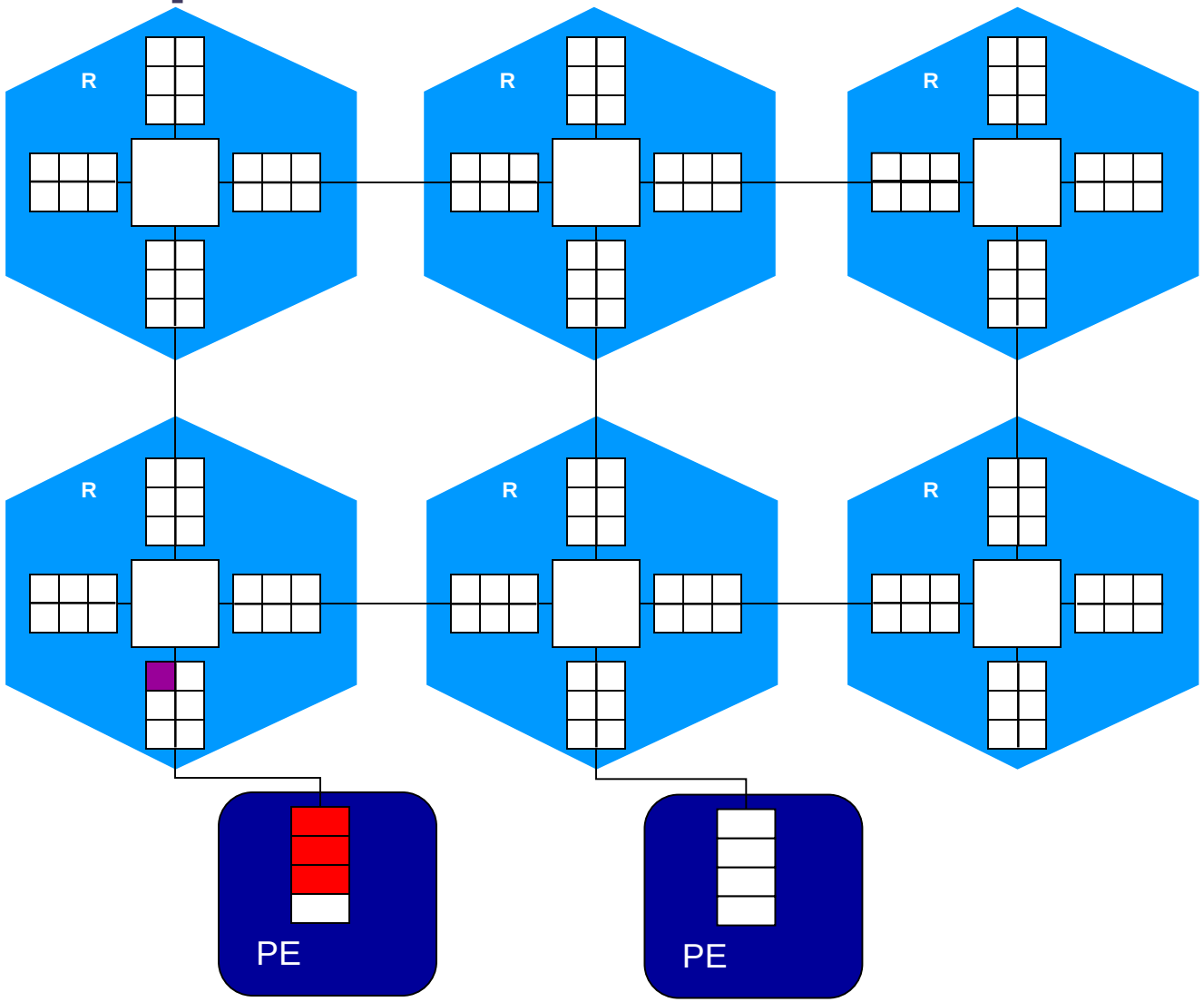
Priority preemptive virtual channels

wormhole NoC with priority preemptive virtual channels



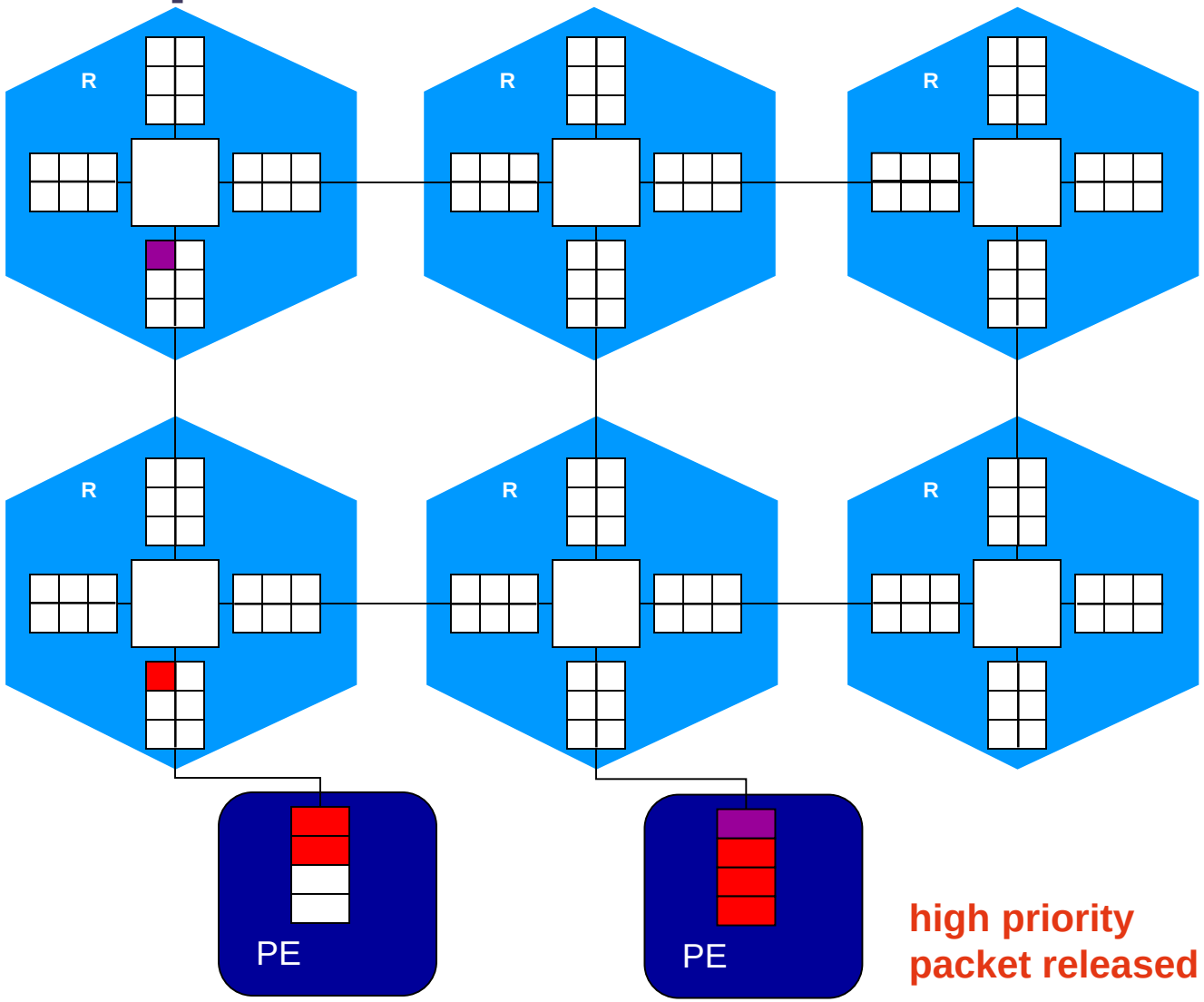
Priority preemptive virtual channels

wormhole NoC with priority preemptive virtual channels



Priority preemptive virtual channels

wormhole NoC with priority preemptive virtual channels



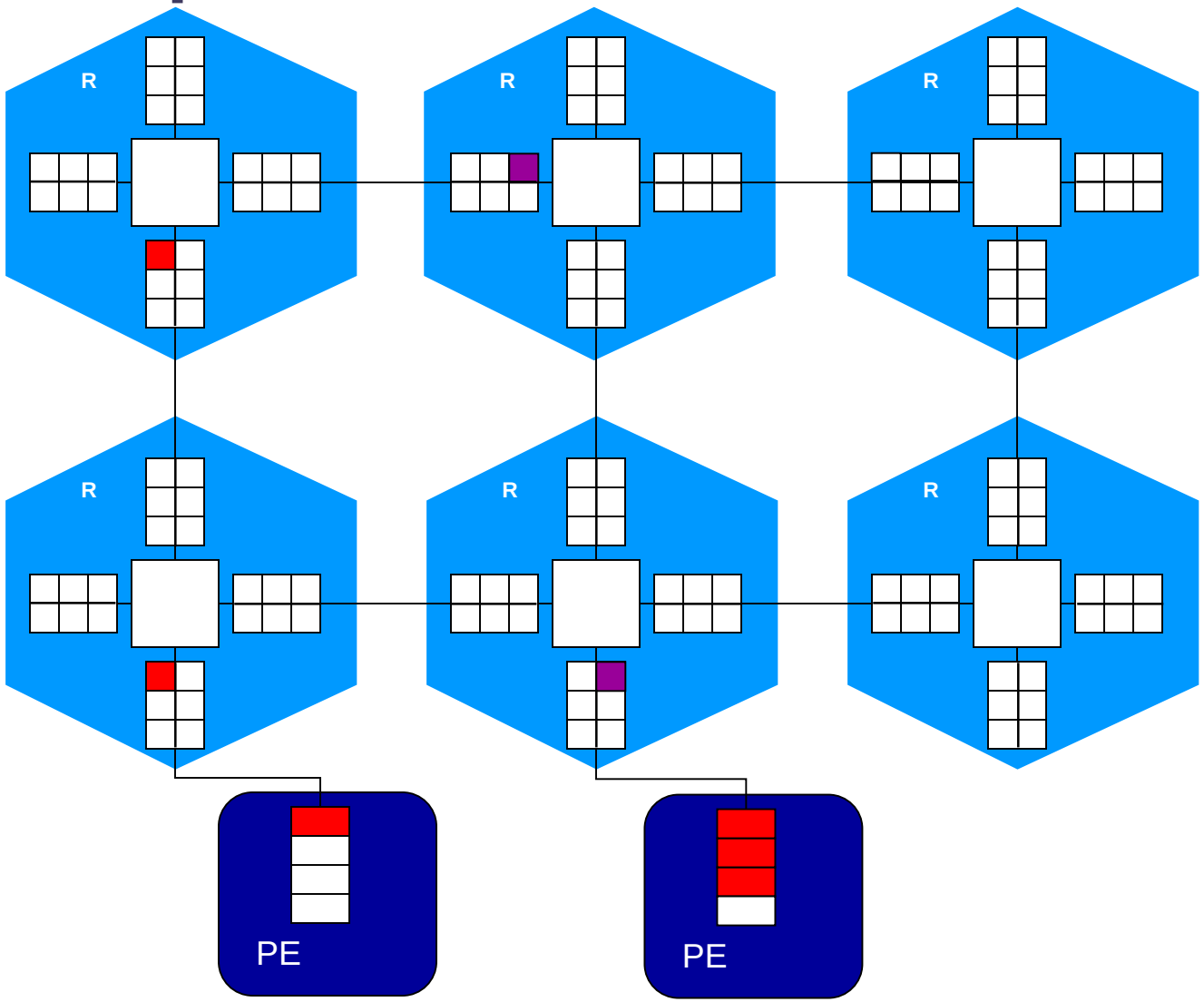
Packet Header

Packet Data

high priority packet released

Priority preemptive virtual channels

wormhole NoC with priority preemptive virtual channels

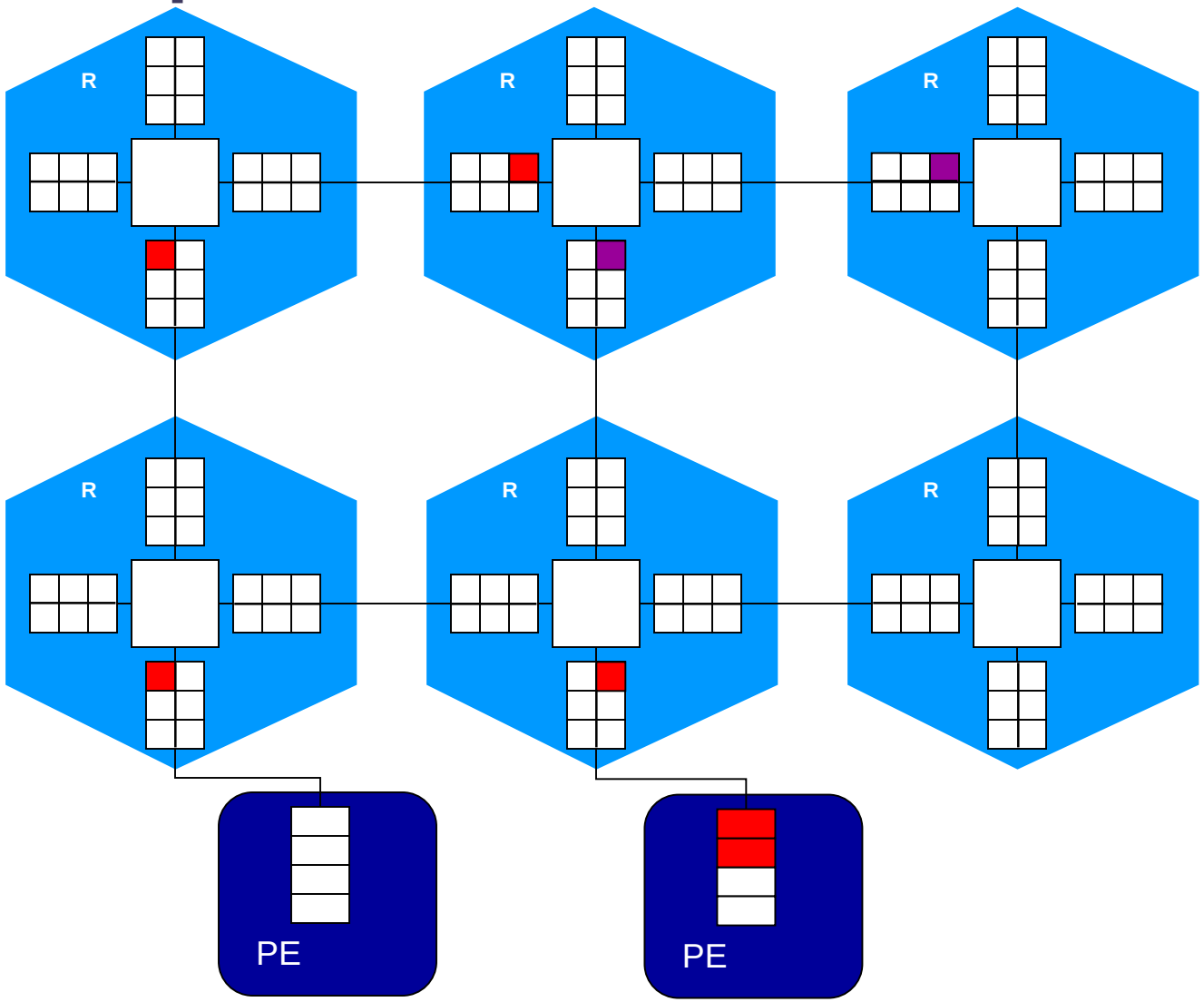


Packet Header

Packet Data

Priority preemptive virtual channels

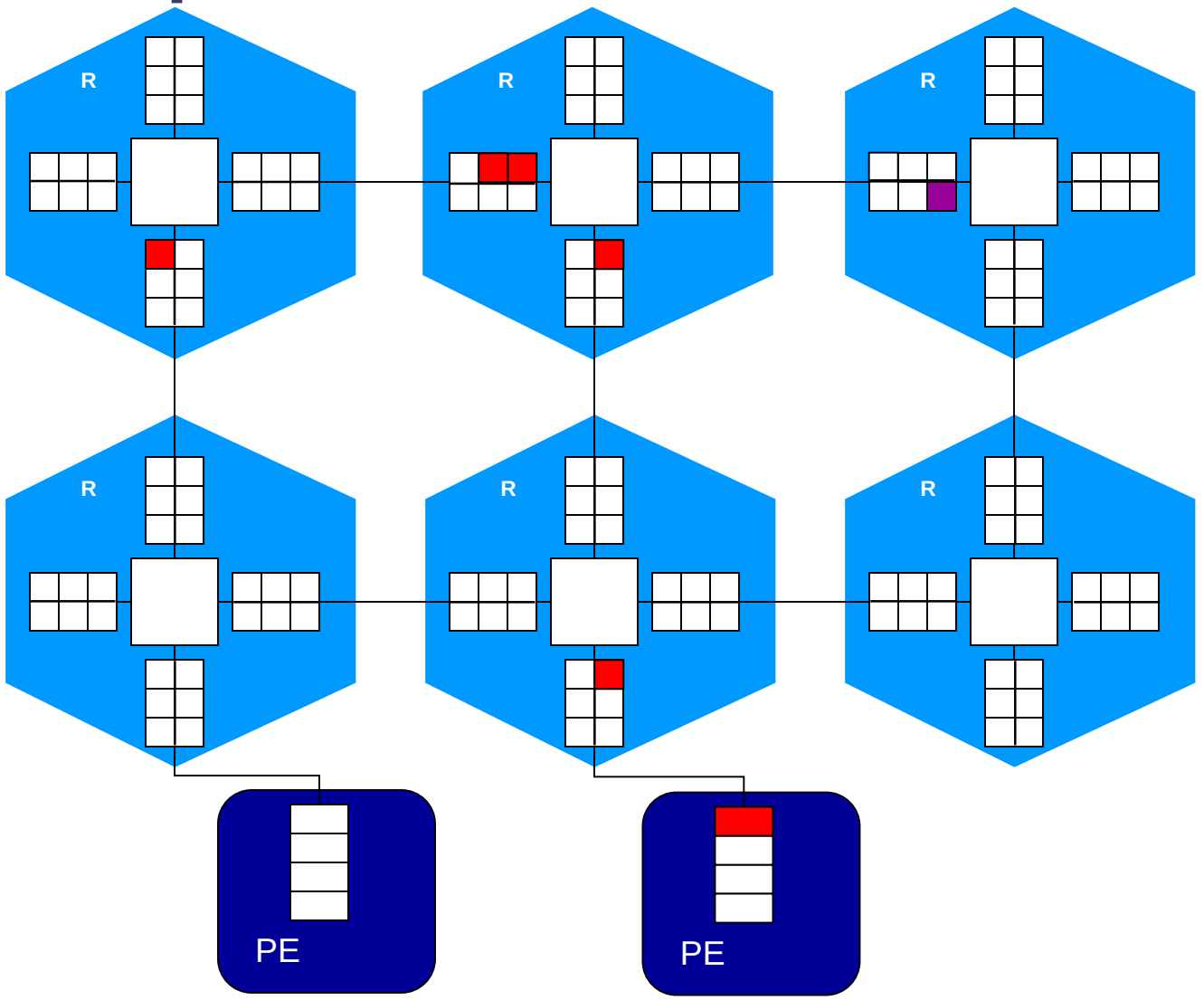
wormhole NoC with priority preemptive virtual channels



Priority preemptive virtual channels

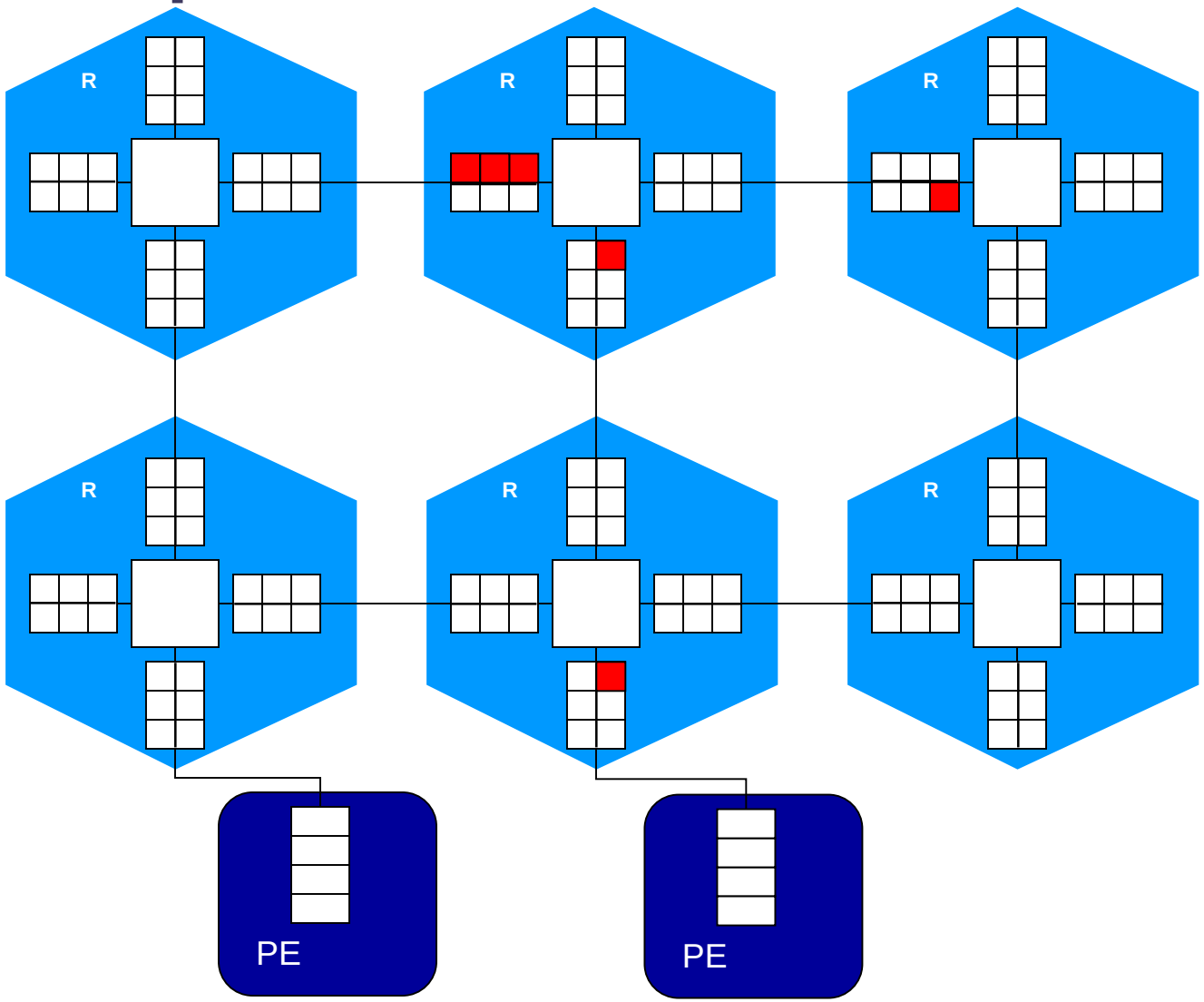
first packet is preempted

wormhole NoC with priority preemptive virtual channels



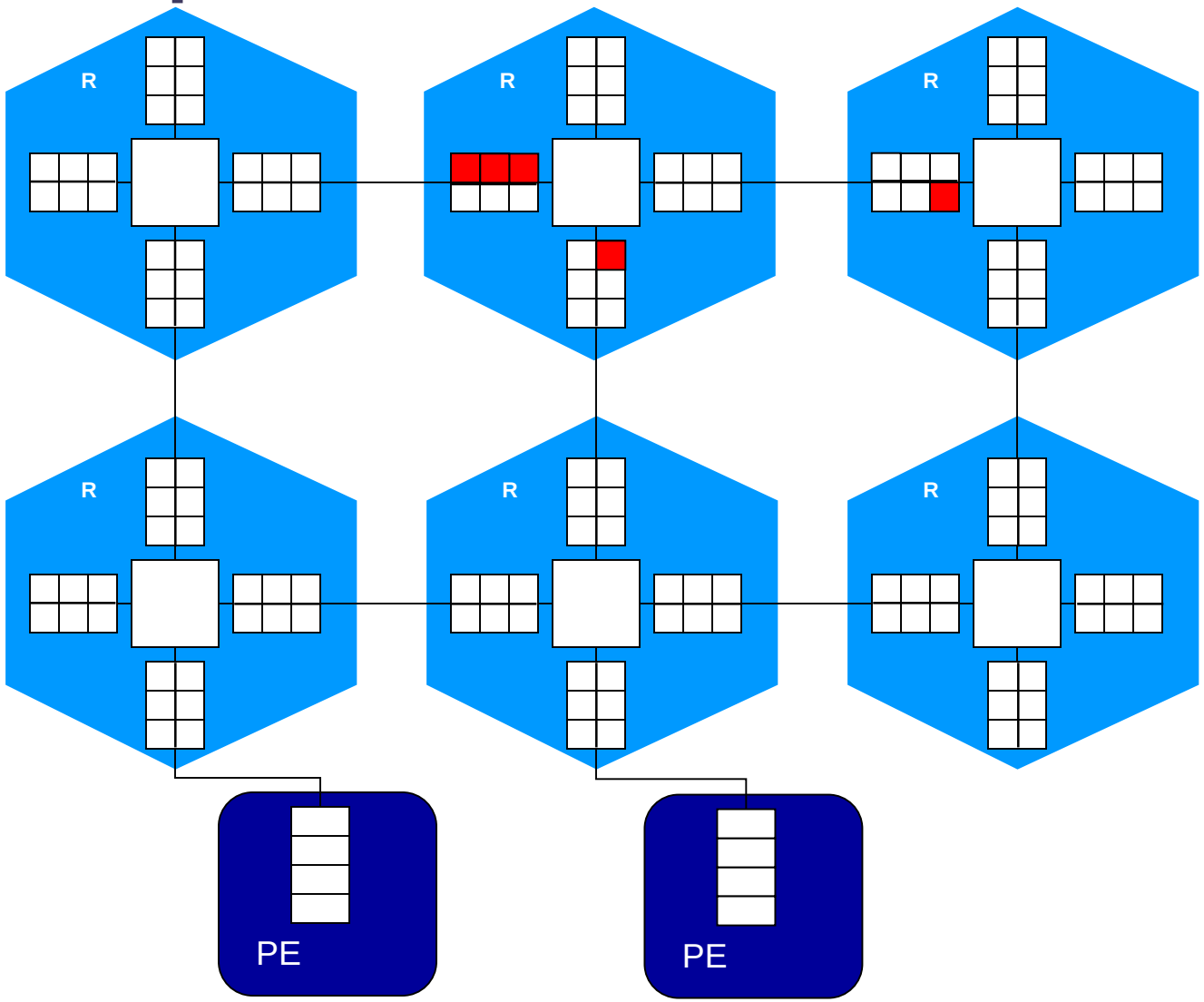
Priority preemptive virtual channels

wormhole NoC with priority preemptive virtual channels



Priority preemptive virtual channels

wormhole NoC with priority preemptive virtual channels

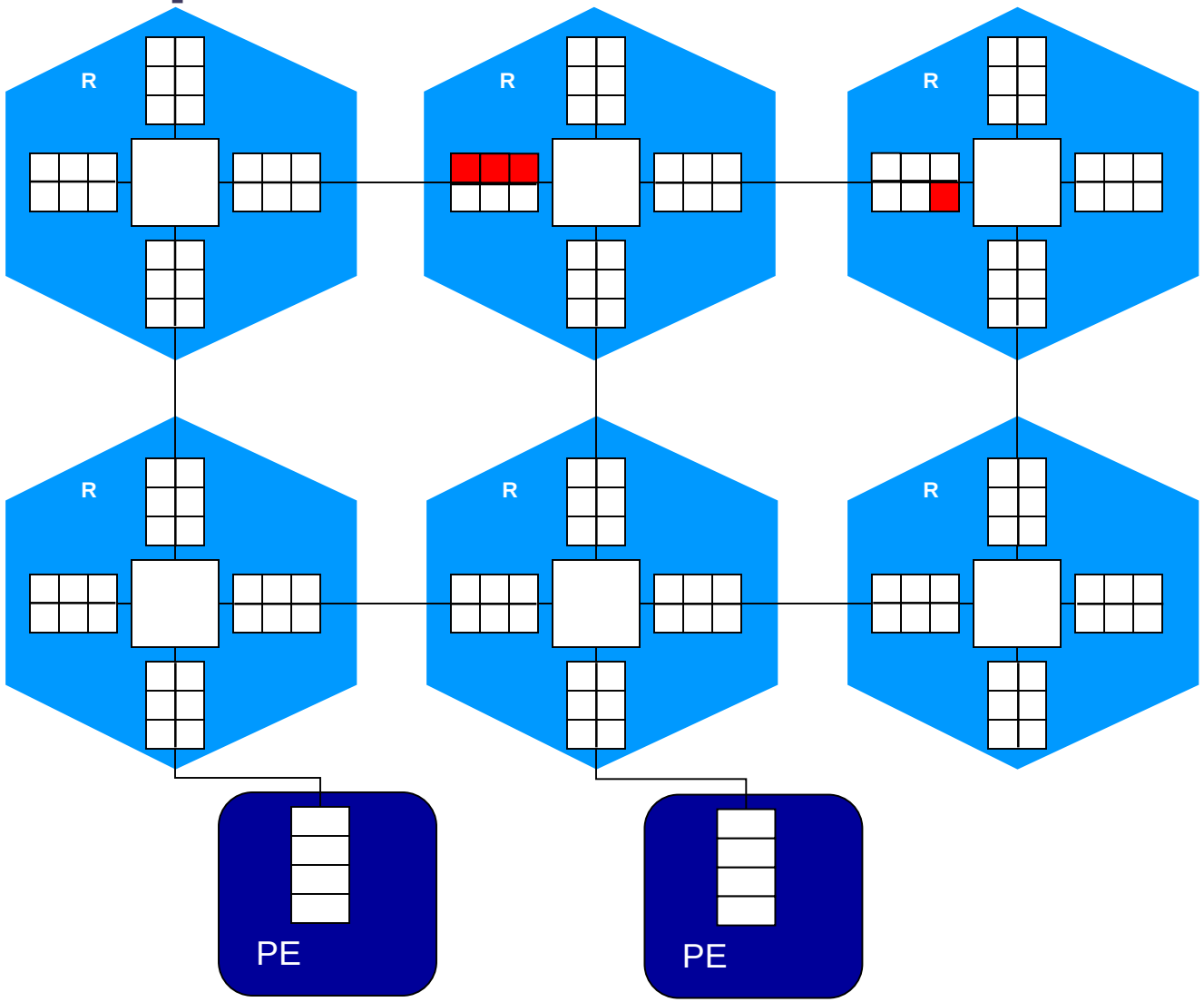


Packet Header

Packet Data

Priority preemptive virtual channels

wormhole NoC with priority preemptive virtual channels

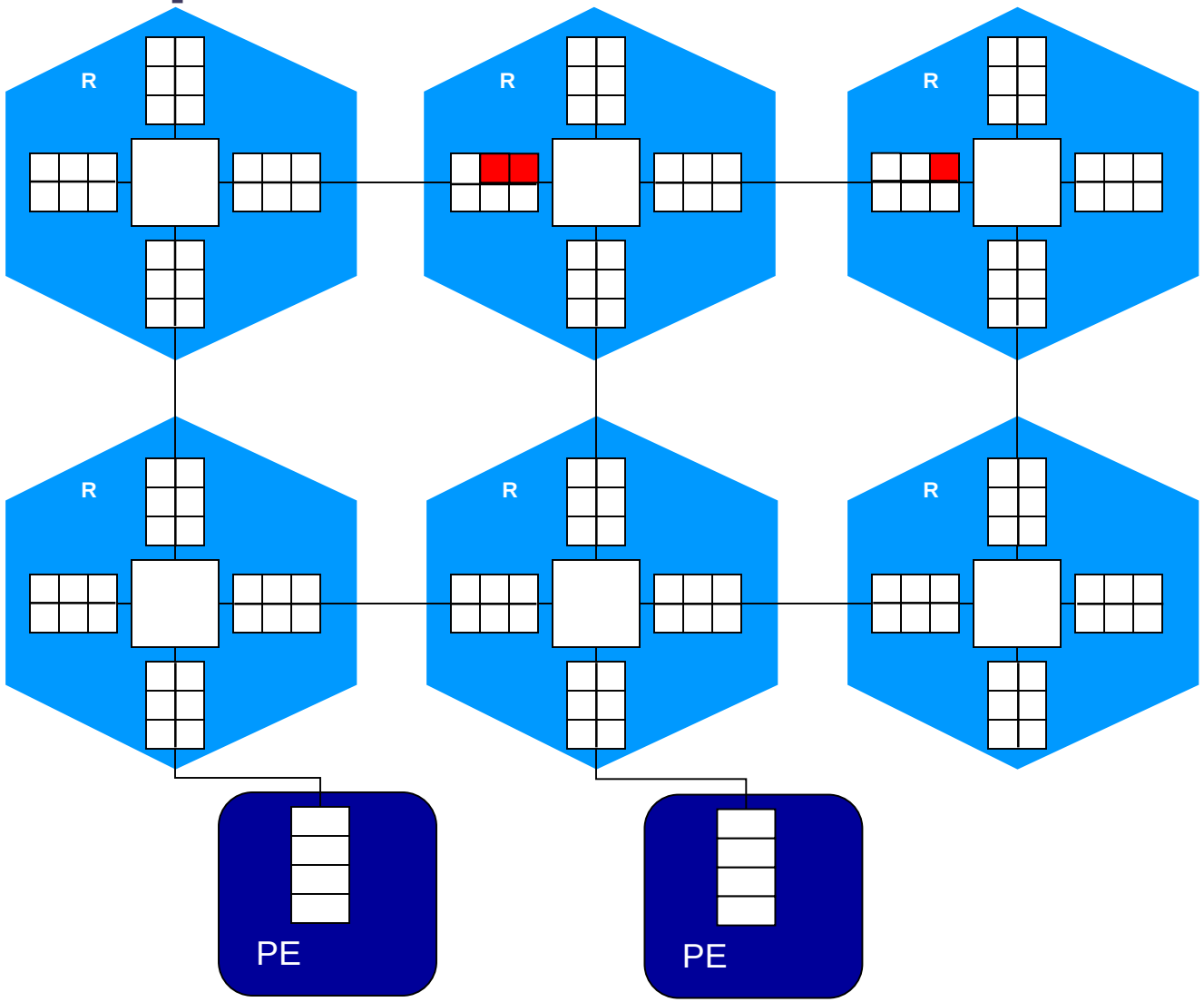


Packet Header

Packet Data

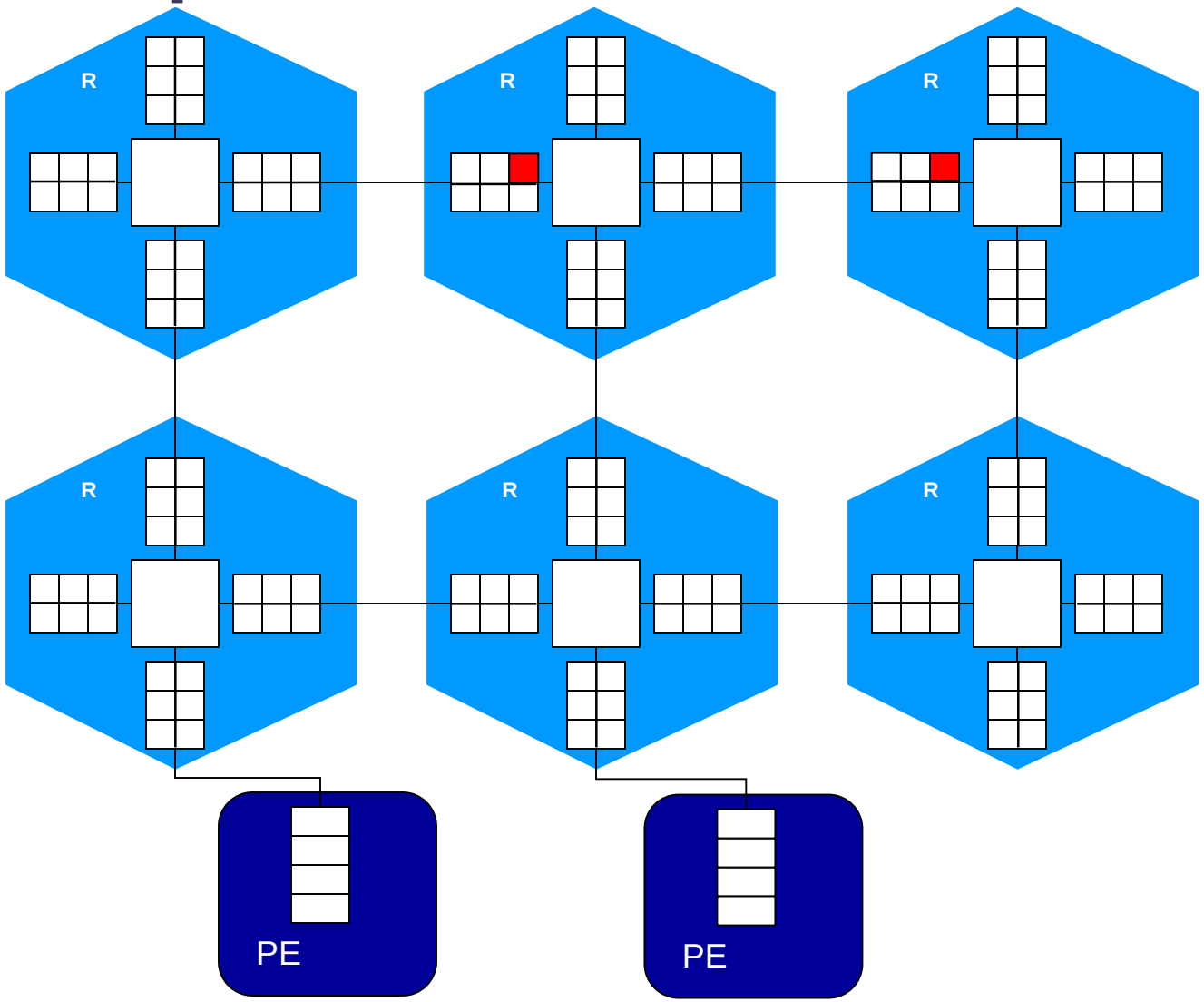
Priority preemptive virtual channels

wormhole NoC with priority preemptive virtual channels



Priority preemptive virtual channels

wormhole NoC with priority preemptive virtual channels

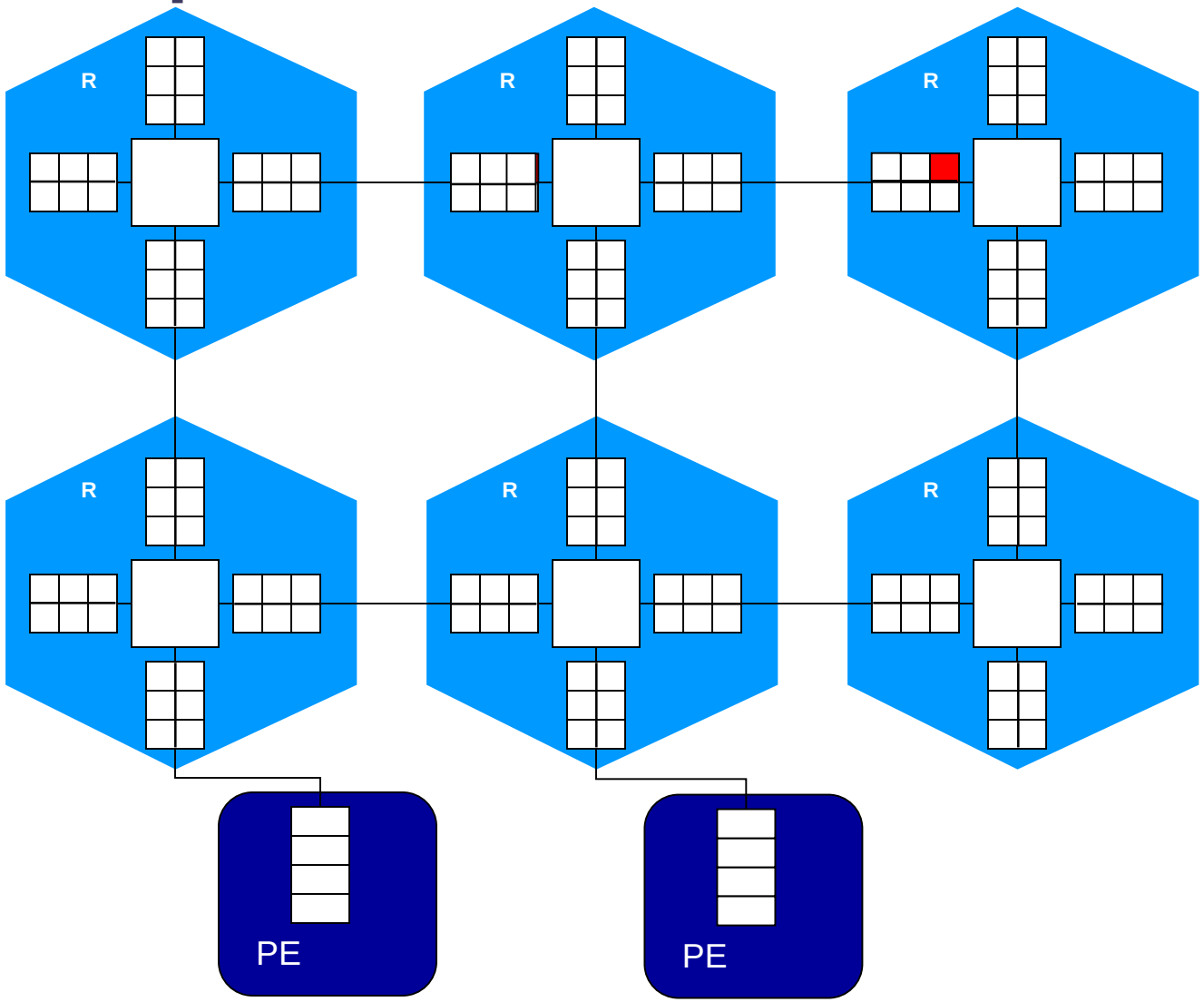


Packet Header

Packet Data

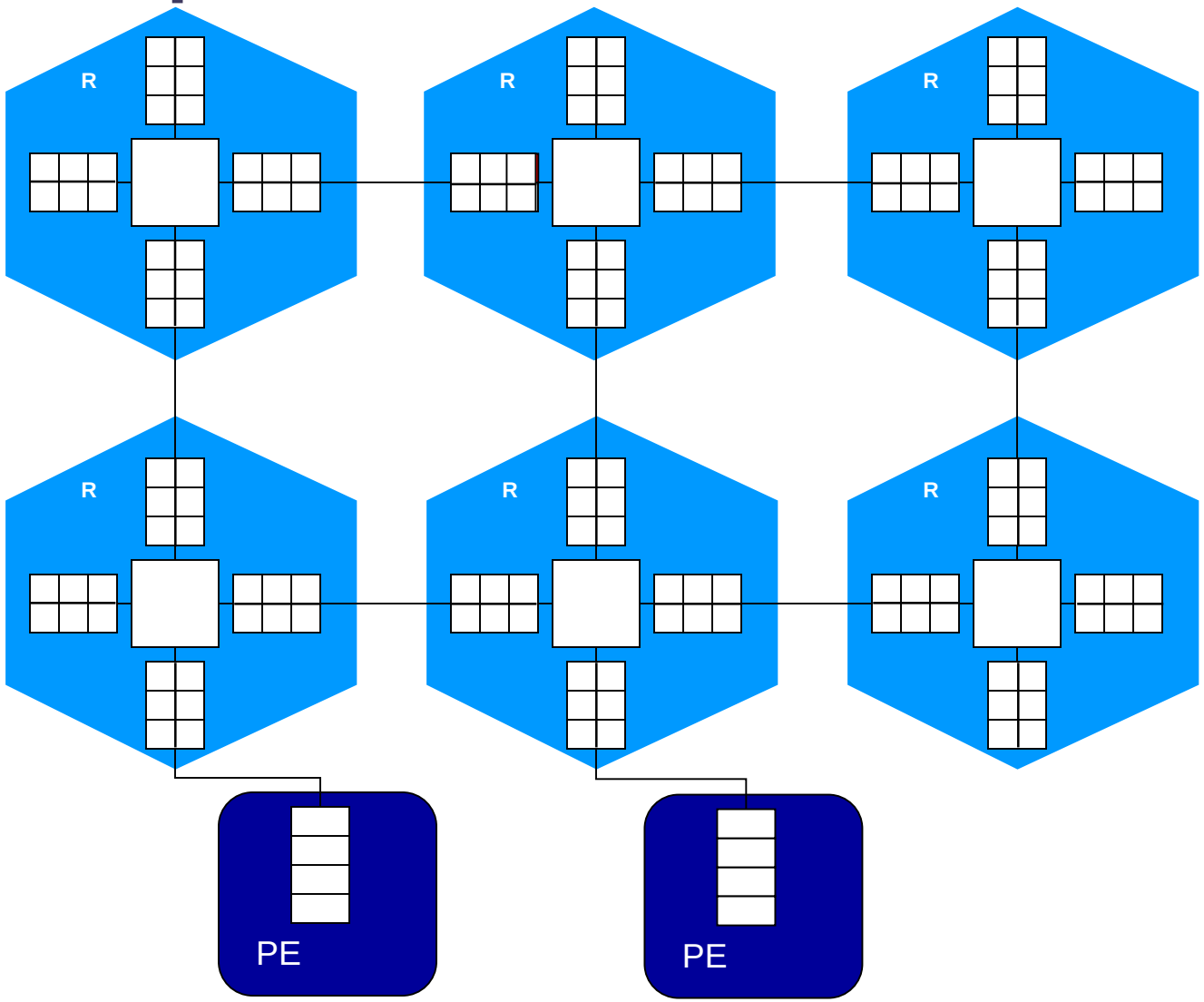
Priority preemptive virtual channels

wormhole NoC with priority preemptive virtual channels



Priority preemptive virtual channels

wormhole NoC with priority preemptive virtual channels



Priority-preemptive wormhole NoCs

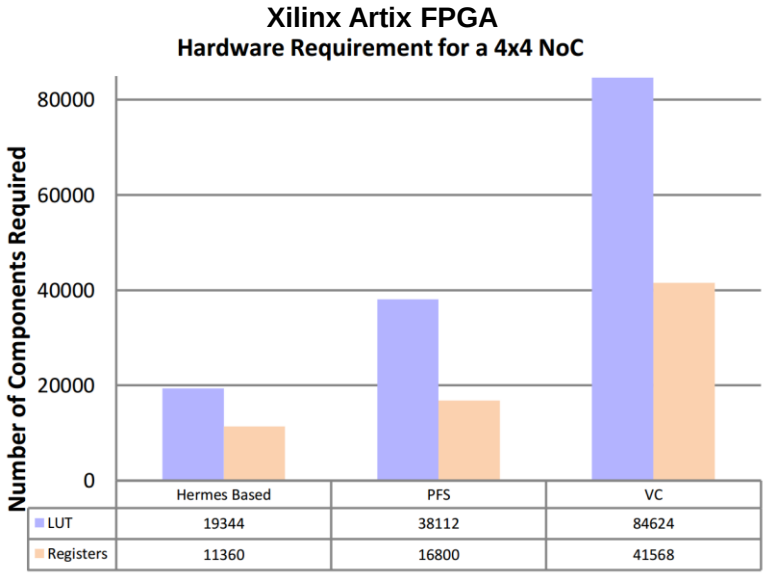
Pros vs Cons

Priority-preemptive wormhole NoCs

- Cons
  - not available as COTS

Priority-preemptive wormhole NoCs

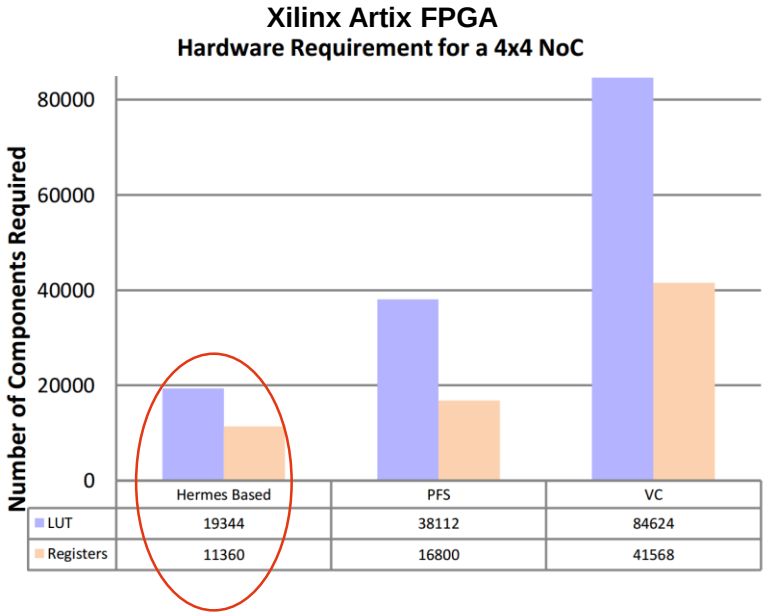
- Cons
  - hardware overhead related to virtual channel buffering and arbitration



B. Sudev, L. S. Indrusiak: Low overhead predictability enhancement in non-preemptive network-on-chip routers using Priority Forwarded Packet Splitting. ReCoSoC 2014.

Priority-preemptive wormhole NoCs

- Cons
  - hardware overhead related to virtual channel buffering and arbitration

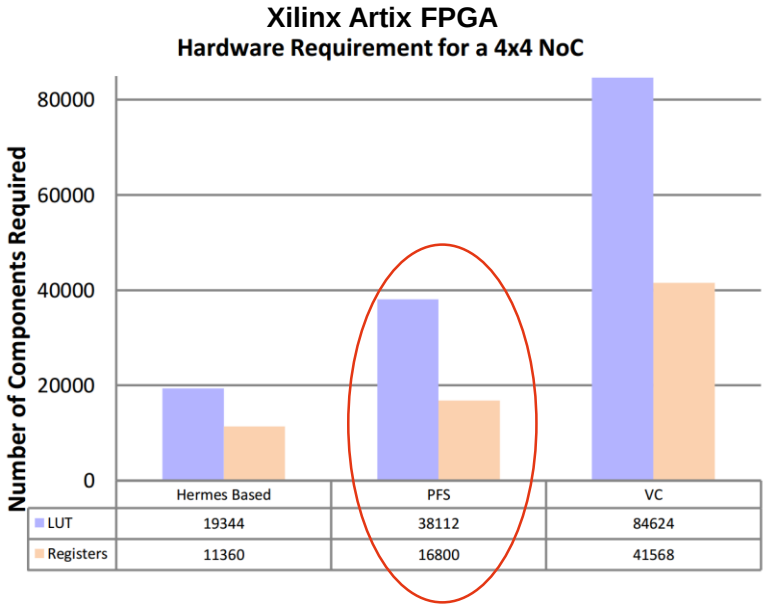


simple round-robin,
no traffic shaping

B. Sudev, L. S. Indrusiak: Low overhead predictability enhancement in non-preemptive network-on-chip routers using Priority Forwarded Packet Splitting. ReCoSoC 2014.

Priority-preemptive wormhole NoCs

- Cons
  - hardware overhead related to virtual channel buffering and arbitration



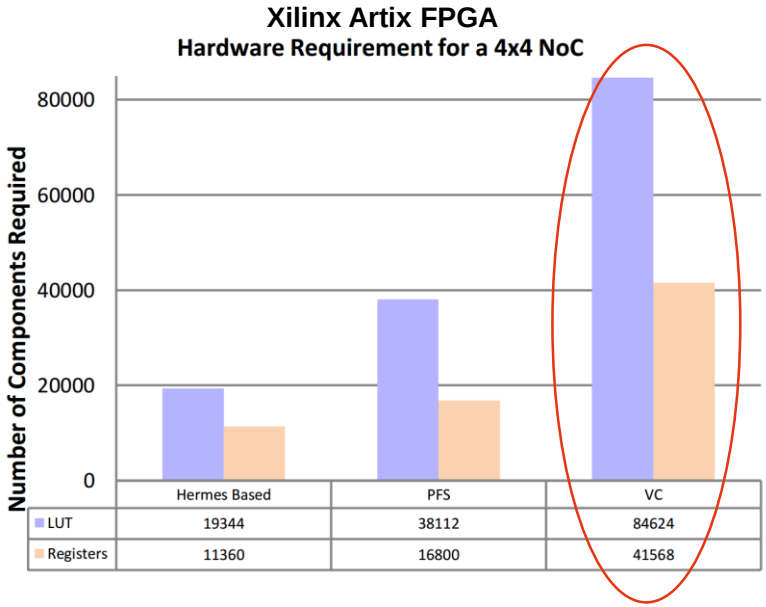
OPEN PROBLEM ALERT

priority non-preemptive arbitration [Sudev & Indrusiak, ReCoSoC 2014]

B. Sudev, L. S. Indrusiak: Low overhead predictability enhancement in non-preemptive network-on-chip routers using Priority Forwarded Packet Splitting. ReCoSoC 2014.

Priority-preemptive wormhole NoCs

- Cons
  - hardware overhead related to virtual channel buffering and arbitration



priority preemptive arbitration, 4 VCs with 2 position buffers each

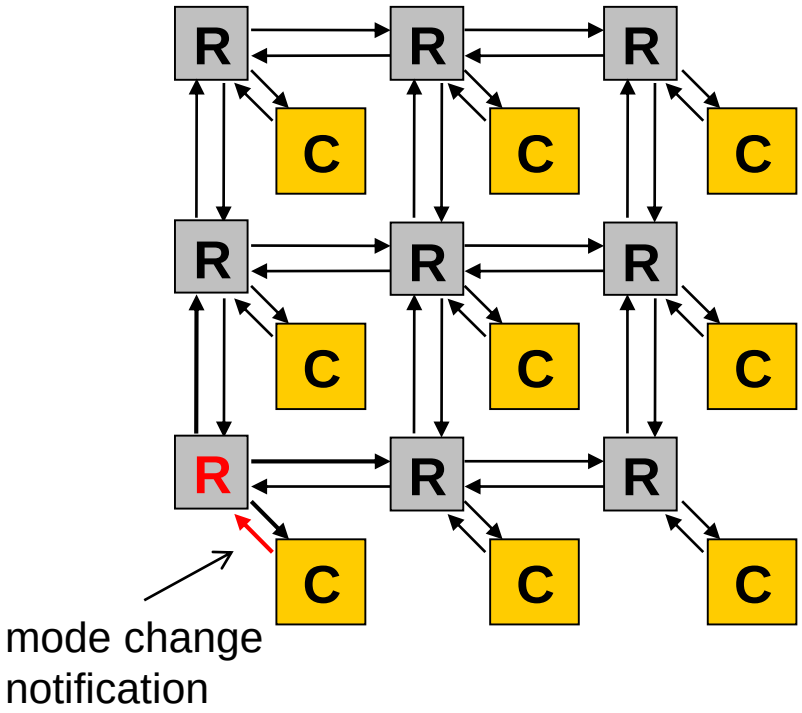
B. Sudev, L. S. Indrusiak: Low overhead predictability enhancement in non-preemptive network-on-chip routers using Priority Forwarded Packet Splitting. ReCoSoC 2014.

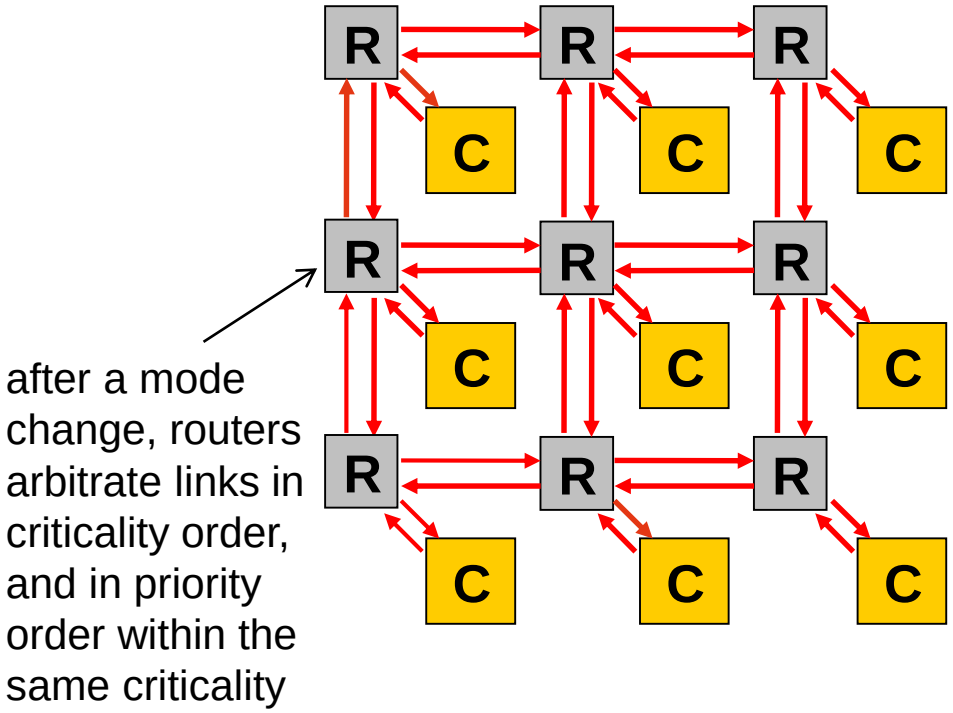
Priority-preemptive wormhole NoCs

- Pros
  - notion of priorities is very intuitive and natural
  - no waste of bandwidth through reservation mechanisms
  - amenable to tight analysis methods (more on this later)
  - virtual separation of traffic
  - accommodates change in traffic properties (periods, packet sizes, jitter)

Priority-preemptive wormhole NoCs

- Pros
  - simple protocols to handle mixed-criticality traffic





L. S. Indrusiak, J. Harbin, A. Burns: Average and Worst-Case Latency Improvements in Mixed-Criticality Wormhole Networks-on-Chip. ECRTS 2015.

Priority-preemptive wormhole NoCs

Outline

- Wormhole Networks
- Networks-on-Chip
- **Real-Time Analysis**
- Resource Management

Performance evaluation

- How to estimate performance figures for a particular application mapped to a Network-on-Chip?
  - full system prototyping
    - cores + NoC in FPGA, running OS + application
    - extremely costly setup time, can only explore few design alternatives
  - accurate system simulation
    - cycle-accurate model of cores + NoC, running OS + application
    - extremely long simulation time, can only explore few design alternatives
  - approximately-timed system simulation
    - approximately-timed model of cores + NoC, executing an abstract model of the OS + application
  - analytical system performance models
    - average or worst-case latency estimation for restricted application styles (periodic independent tasks, synchronous dataflow, etc.)

Performance evaluation

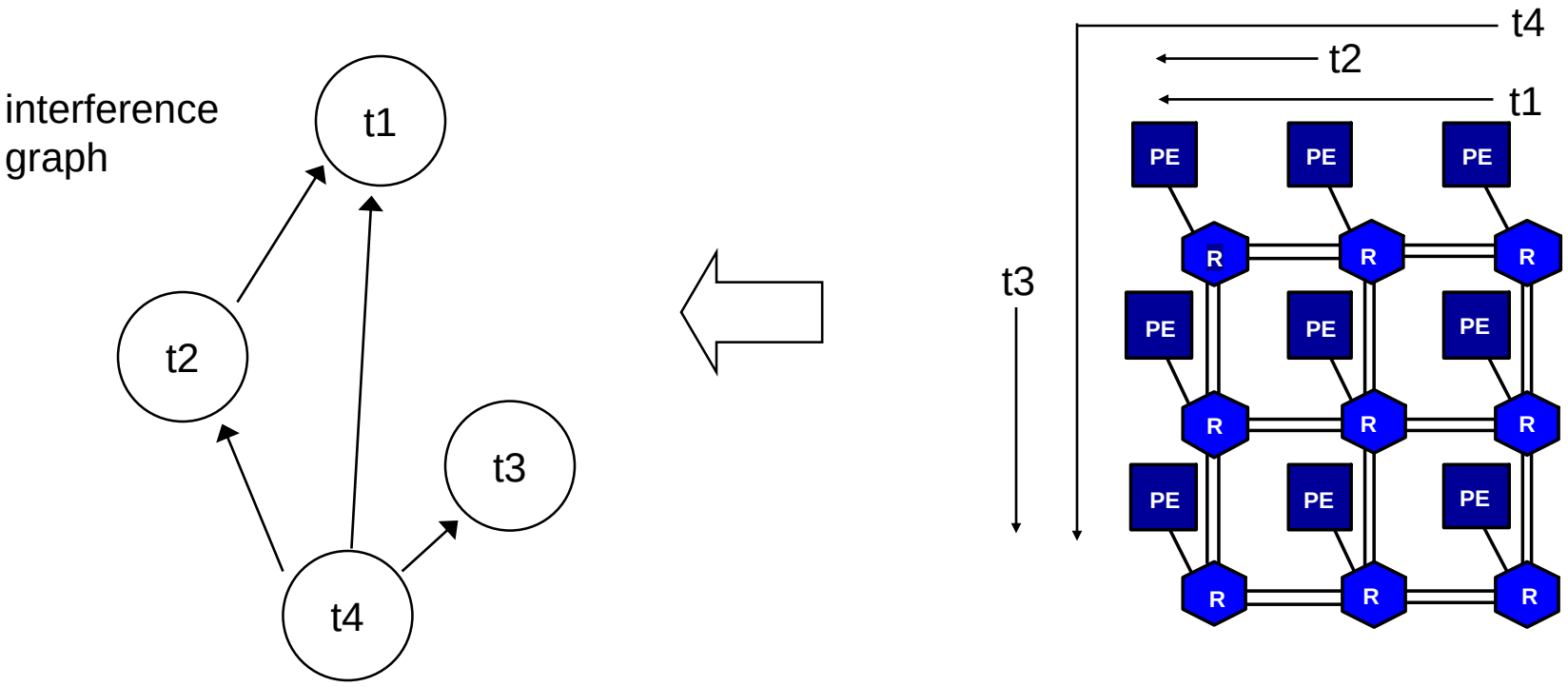
- How to estimate performance figures for a particular application mapped to a Network-on-Chip?
  - full system prototyping
    - cores + NoC in FPGA, running OS + application
    - extremely costly setup time, can only explore few design alternatives
  - accurate system simulation
    - cycle-accurate model of cores + NoC, running OS + application
    - extremely long simulation time, can only explore few design alternatives
  - approximately-timed system simulation
    - approximately-timed model of cores + NoC, executing an abstract model of the OS + application
  - analytical system performance models
    - average or worst-case latency estimation for restricted application styles (periodic independent tasks, synchronous dataflow, etc.)

Real-Time Analysis

- First approaches to analyse priority-preemptive wormhole networks came during the 90s
  - Mutka (1994)
  - Hary and Ozguner (1997)

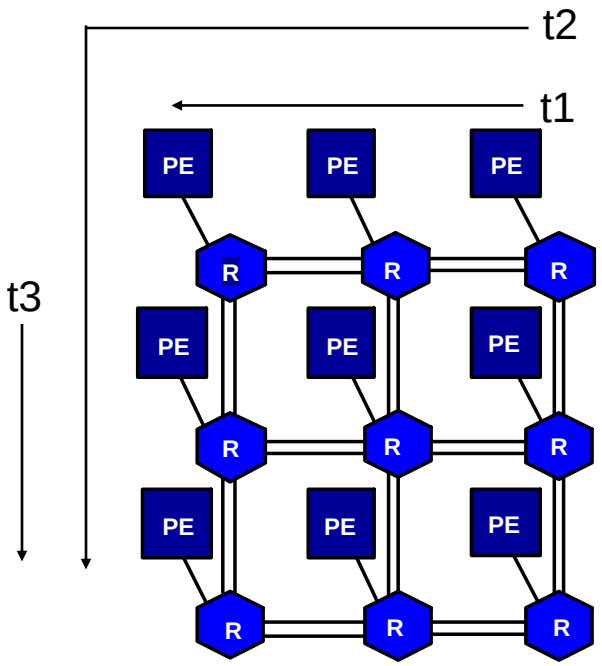
- Key idea is to consider the entire path of a packet as a single shared resource
  - worst-case latency bound of a packet flow can be found by analysing the higher priority packet flows that share at least one link of its route

Real-Time Analysis



Real-Time Analysis

- Kim et al (1998) recognised that direct interferences are not enough to produce correct upper bounds
- Indirect interference must be considered, in order to take into account back-to-back hits caused by upstream indirect interference



pri(t1)>pri(t2)>pri(t3)

Real-Time Analysis

- With the introduction of Networks-on-Chip in the 2000s, the approach of Kim et al was revisited by Lu et al (ASP DAC 2005)
  - aiming to provide upper bounds to sporadic packets over NoCs with priority preemptive virtual channels
  - flawed assumption of a critical instant where all packets start flowing simultaneously

Real-Time Analysis

- Shi and Burns (NOCS 2008) corrected the flaw on Lu et al and produced a response time formulation that uses a conservative approach to upstream indirect interference

$$R_i^{n+1} = L_i + \sum_{\forall Flow_j \in S_i^D} \left\lceil \frac{R_i^n + J_j^R + J_j^I}{T_j} \right\rceil L_j$$

OPEN PROBLEM ALERT

interference jitter
$J_j^I = R_j - L_j$

Real-Time Analysis

- Several lines of work were derived from Shi and Burns 2008
  - highly cited: 145 (Google Scholar)
  - many works on priority assignment and task mapping
  - a few on analysis improvement, aiming to make it tighter
    - Nikolic et al (arxiv 2016) considered that the interference should not be calculated based on the full path, but the contention domain
    - Kashif et al (Trans Comp 2015) attempted to analyse packet paths on a link-by-link manner, but assumed infinite buffering (i.e. did not consider backpressure)
    - Kashif and Patel (RTAS 2016) attempted to consider buffering and backpressure effects
    - all of them upper-bounded by Shi and Burns 2008

Real-Time Analysis

- Xiong et al (GLSVLSI 2016) has made two key contributions
  - new formulation to the upstream indirect interference problem, aiming to be tighter than Shi and Burns 2008
  - new formulation to the downstream indirect interference problem, aiming to capture a previously unseen issue, and showing that Shi and Burns 2008 is optimistic and unsafe (and so are all the analyses upper-bounded by it)

Real-Time Analysis

- Indrusiak et al (arxiv 2016) has shown that
  - Xiong et al's formulation to the upstream indirect interference problem was flawed
  - Xiong et al's formulation to the downstream indirect interference problem was correct, but unnecessarily pessimistic (i.e. it assumed all indirect interference as if it is direct interference)
  - a tighter upper bound that considers the downstream indirect interference problem is possible

- Xiong et al published a corrected analysis on IEEE Trans Comp in 2017

OPEN PROBLEM ALERT

Outline

- Wormhole Networks
- Networks-on-Chip
- Real-Time Analysis
- **Resource Management**

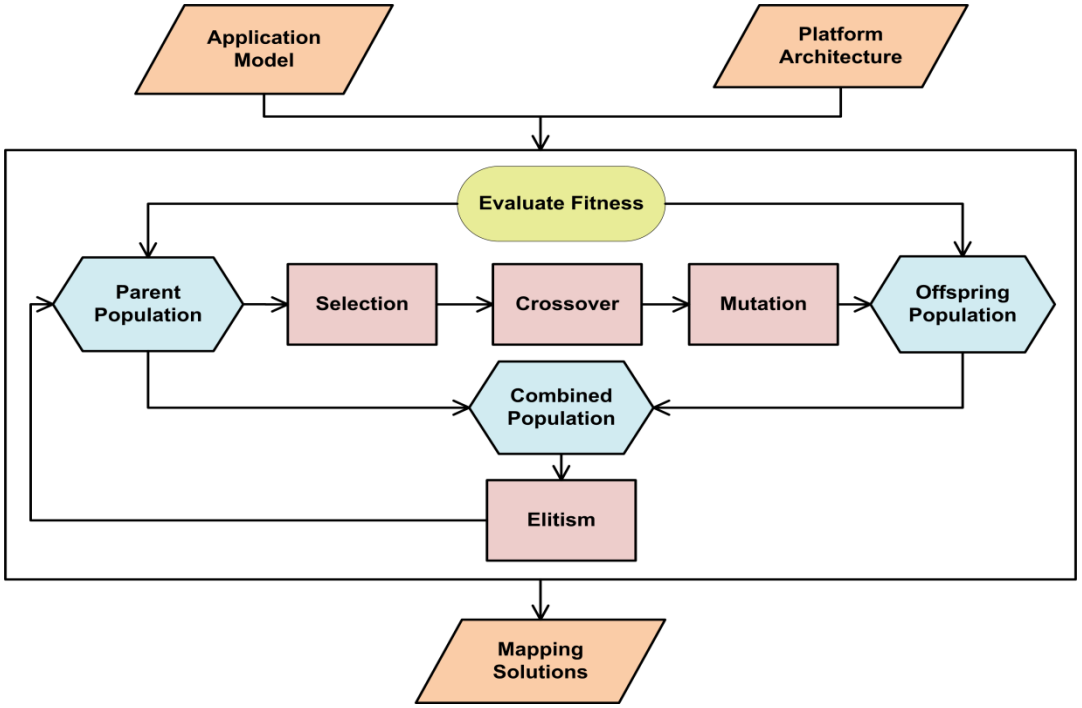
Resource Management

- Analytical models can be used to test whether a particular NoC configuration meets its hard real-time constraints

- It can be used as a fitness function in a search-based optimisation
  - guides the search towards full schedulability
  - much faster than simulation, therefore can cover a wider search space

Optimisation Algorithm

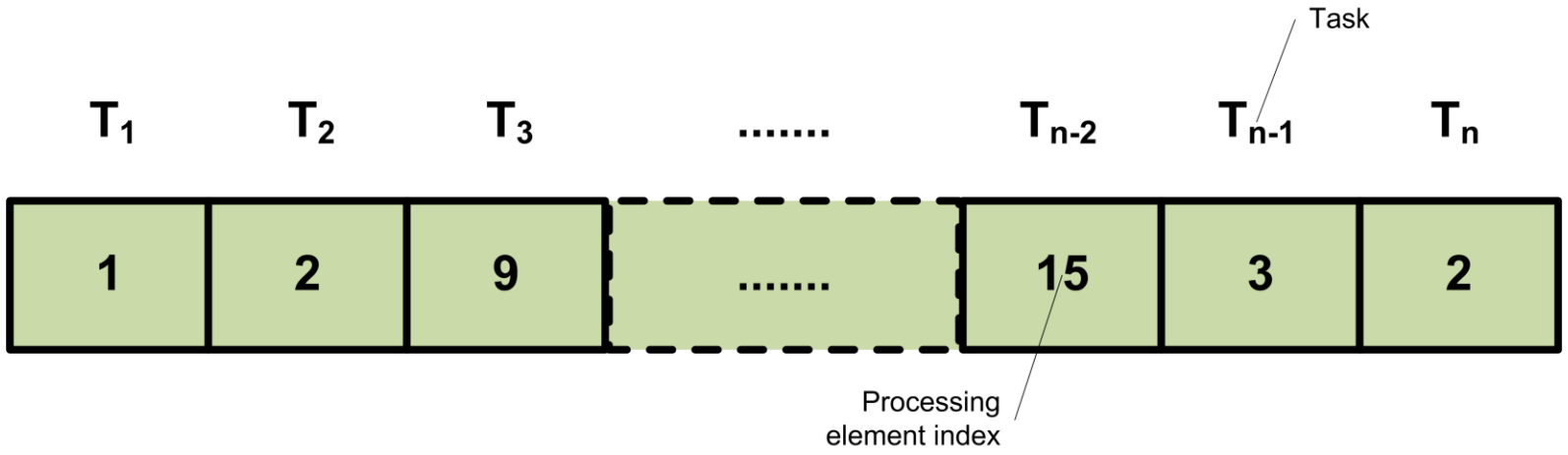
- Optimisation performed by a population-based evolutionary algorithm



P. Mesidis and L. S. Indrusiak, “Genetic mapping of hard real-time applications onto NoC-based MPSoCs — A first approach,” in Int Workshop on Reconfigurable Communication-centric Systems-on-Chip (ReCoSoC), 2011.
M. N. S. M. Sayuti and L. S. Indrusiak, “Real-time low-power task mapping in Networks-on-Chip,” in IEEE Computer Society Annual Symposium on VLSI (ISVLSI), 2013.

Optimisation Algorithm

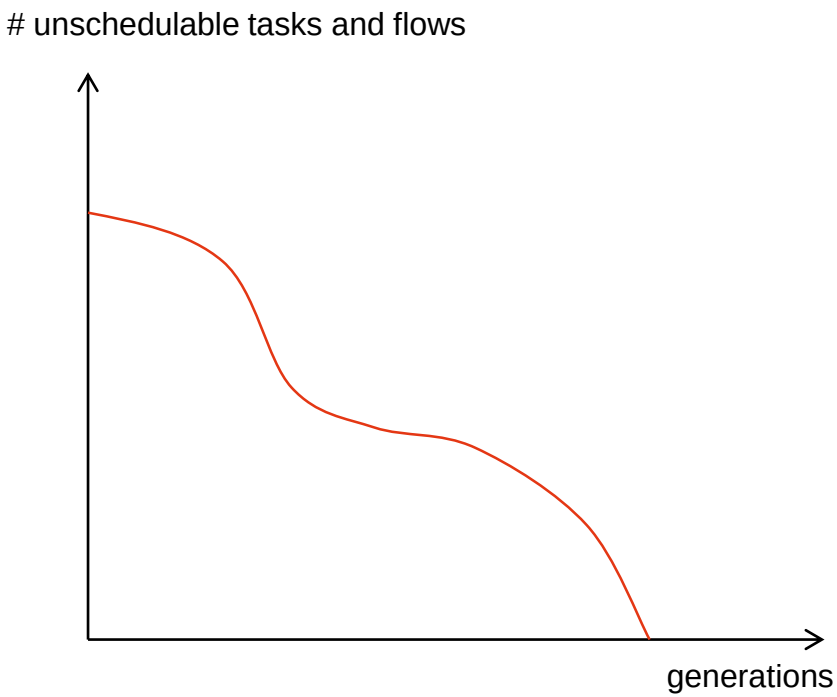
- A population contains a group of individuals represented by a chromosome structure



- Creation of new individuals is facilitated by operators:
  - Selection
  - Crossover
  - Mutation

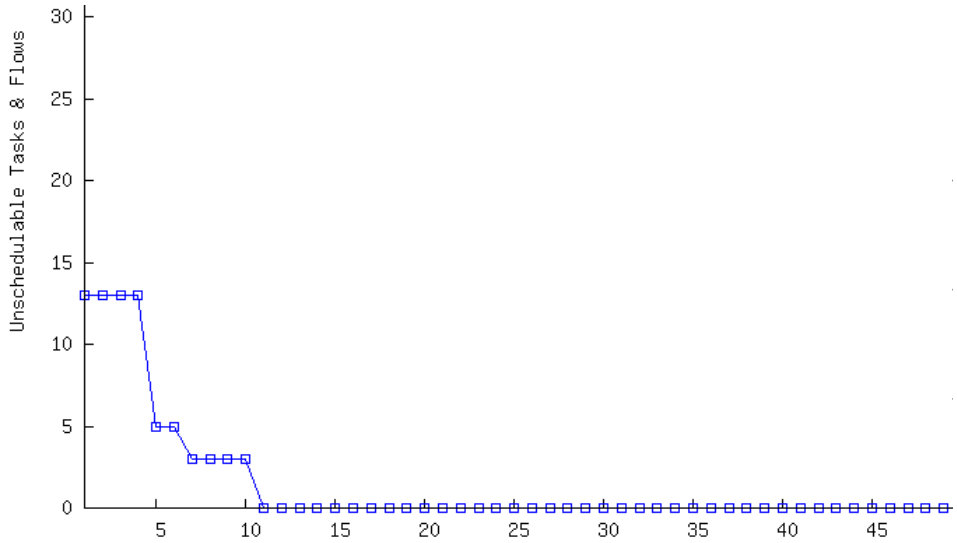
Optimisation algorithm

- Goal: evolve a fully schedulable mapping over generations



Experiment results

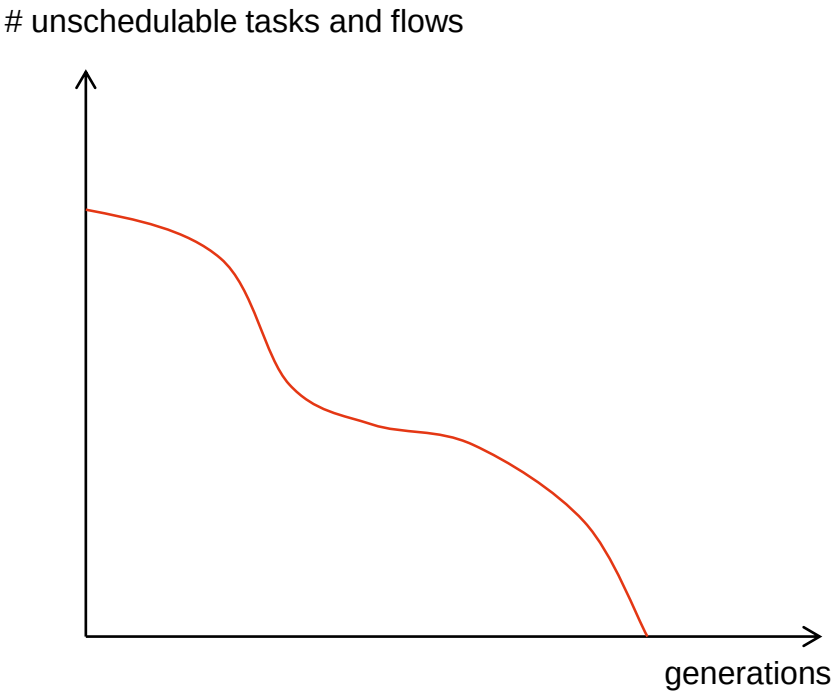
- Autonomous vehicle application (AVA) benchmark (38 communicating tasks), 4x4 Mesh NoC

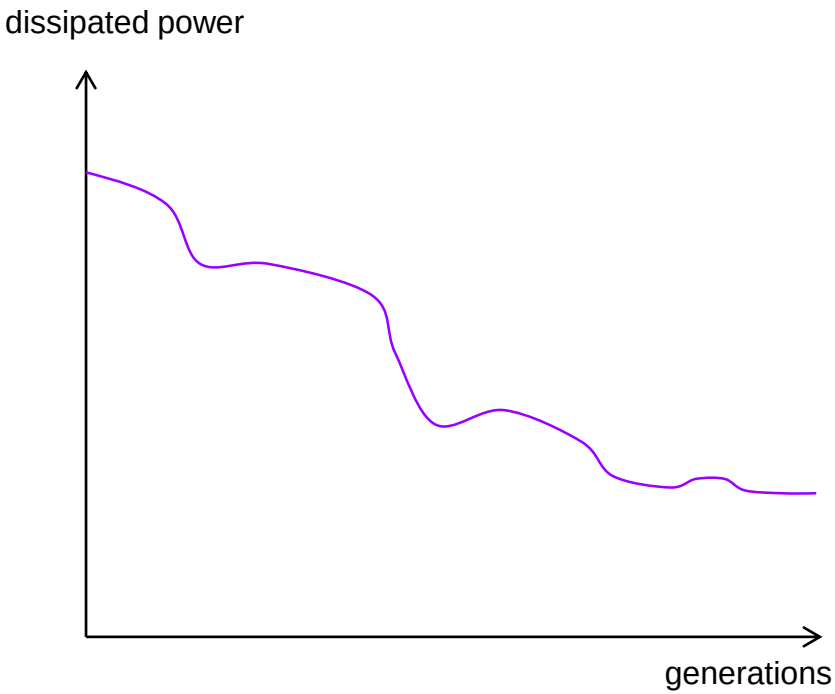


M. N. S. M. Sayuti and L. S. Indrusiak, “Real-time low-power task mapping in Networks-on-Chip,” in IEEE Computer Society Annual Symposium on VLSI (ISVLSI), 2013.

Multi-objective optimisation algorithm

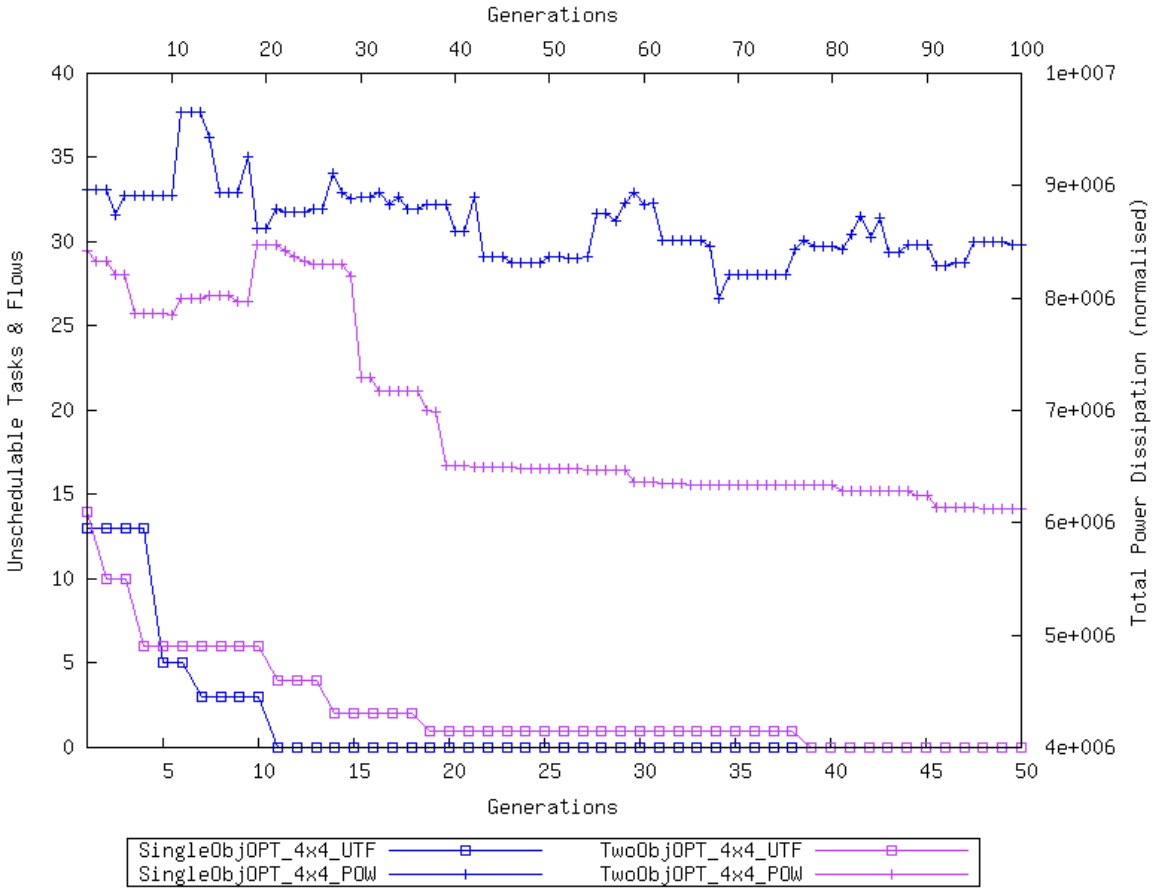
- Additional fitness functions can be added to the evolutionary algorithm
  - example: evolve mappings that are fully schedulable and low power





Experiment results

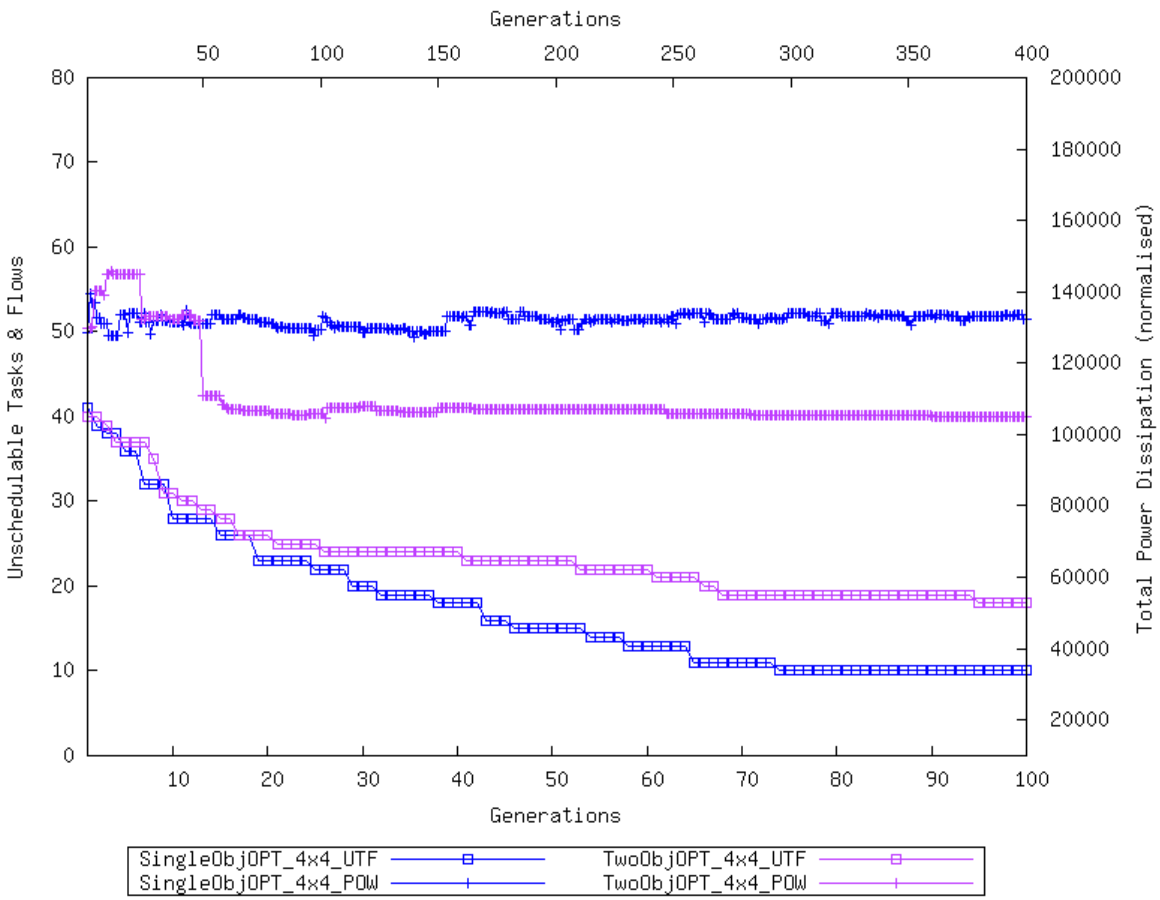
- Autonomous vehicle application (AVA) benchmark, 4x4 Mesh
- Comparison of best solution convergence between single and multiple objectives



M. N. S. M. Sayuti and L. S. Indrusiak, “Real-time low-power task mapping in Networks-on-Chip,” in IEEE Computer Society Annual Symposium on VLSI (ISVLSI), 2013.

Experiment results

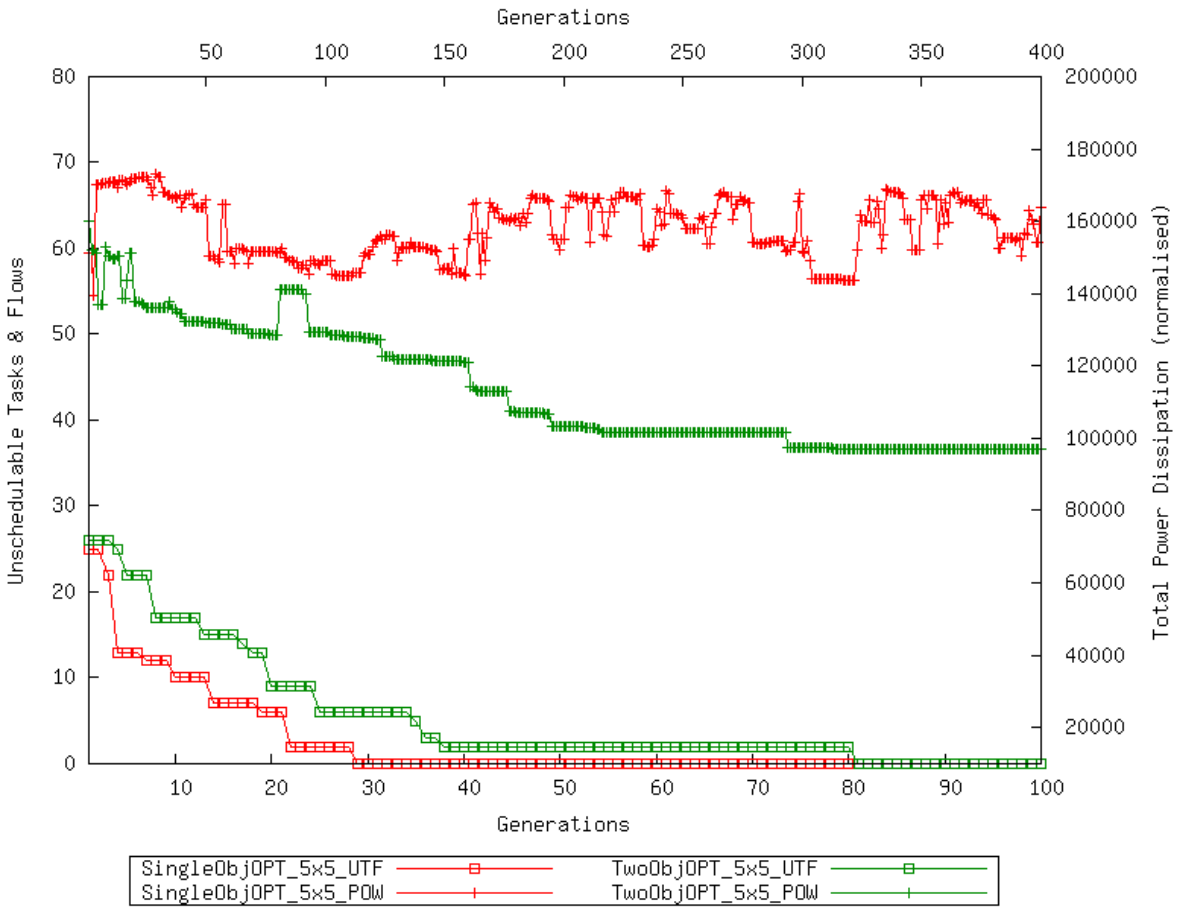
- Synthetic application (SA) benchmark, 4x4 Mesh
- Comparison of best solution convergence between single and multiple objectives



M. N. S. M. Sayuti and L. S. Indrusiak, “Real-time low-power task mapping in Networks-on-Chip,” in IEEE Computer Society Annual Symposium on VLSI (ISVLSI), 2013.

Experiment results

- Synthetic application (SA) benchmark, 5x5 Mesh
- Comparison of best solution convergence between single and multiple objectives



M. N. S. M. Sayuti and L. S. Indrusiak, “Real-time low-power task mapping in Networks-on-Chip,” in IEEE Computer Society Annual Symposium on VLSI (ISVLSI), 2013.

Conclusions

- Priority-preemptive wormhole networks have interesting properties, making them amenable to real-time analysis
- Analysis is less trivial than originally thought
  - advances needed on accounting for indirect interference
- Analytical models are useful guides to search heuristics attempting to configure multiple system aspects
  - task mapping, packet routing, priority assignment, security features, voltage and frequency scaling

OPEN PROBLEM ALERT

OPEN PROBLEM ALERT

UNIVERSITY of York

Priority-based Wormhole Networks-on-Chip: challenges and opportunities

Leandro Soares Indrusiak

with contributions by: Alan Burns, Rob Davis, Iain Bate, Neil Audsley, James Harbin, Piotr Dziurzanski, Amit Singh, Paris Mesidis, Adrian Racu, Norazizi Sayuti, Zheng Shi, Yunfeng Ma, Rosh Mendis, Bharath Sudev

with funding from: EU (DreamCloud, T-CREST), EPSRC (MCC, MCCps)

RTN 2017