# Programmable Switch Hardware

## ECE/CS598HPN

*Radhika Mittal*

# Conventional SDN

- Programmable *control plane*.
- Data plane can support high bandwidth.
  - But has limited flexibility.
- Restricted to conventional packet protocols.

# Software Dataplane

- Very extensible and flexible.
- Extensive parallelization to meet performance requirements.
  - Might still be difficult to achieve 100's of Gbps.
- Significant cost and power overhead.

# Programmable Hardware

- More flexible than conventional switch hardware.
- Less flexible than software switches.

- Slightly higher power and cost requirements than conventional switch hardware.
- Significantly lower than software switches.

# Other alternatives?

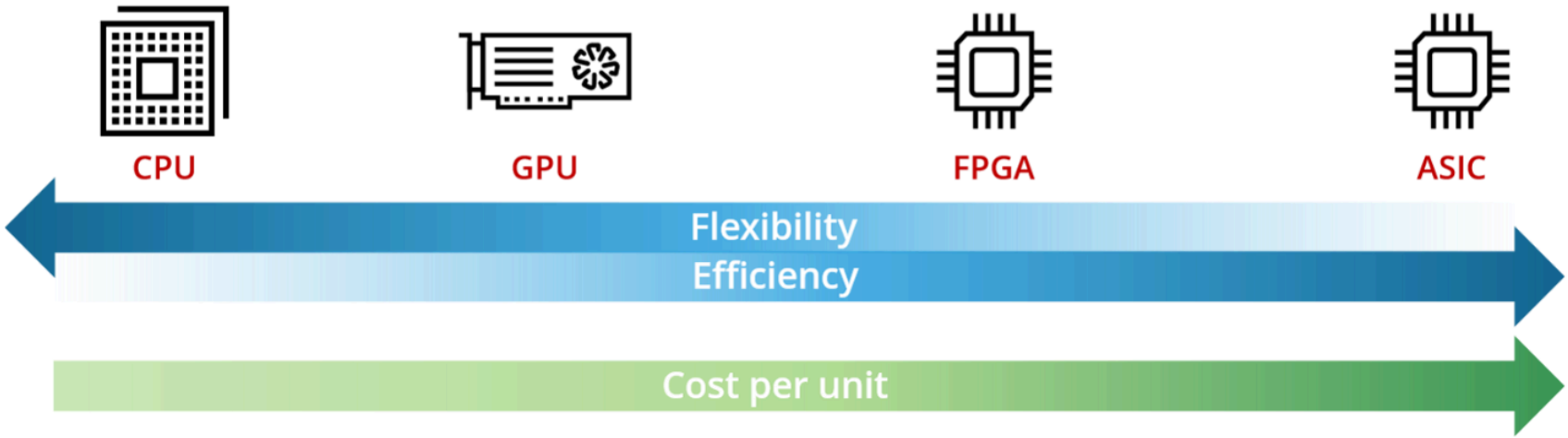



*Image copied from somewhere on the web.*

# Forwarding Metamorphosis: Fast Programmable Match-Action Processing in Hardware for SDN

Pat Bosshart, Glen Gibb, Hun-Seok Kim,
George Varghese, Nick McKeown, Martin Izzard,
Fernando Mujica, Mark Horowitz

*Acknowledgements: Slides from Pat Bosshart's SIGCOMM'13 talk*

What are the limitations of a fixed function switch?

# Need for flexibility….

- Flexibility to:
  - Trade one memory size for another
  - Add a new table
  - Add a new header field
  - Add a different action
- SDN accentuates the need for flexibility
  - Gives programmatic control to control plane, expects to be able to use flexibility
  - OpenFlow designed to exploit flexibility.

# What the Authors Set Out To Learn

- How to design a flexible switch chip?
- What does the flexibility cost?

# RMT Switch Model

Enables flexibility through….

- Programmable parsing: support arbitrary header fields
- Ability to configure number, topology, width, and depths of match-tables.
- Programmable actions: allow a flexible set of actions (including arbitrary packet modifications).

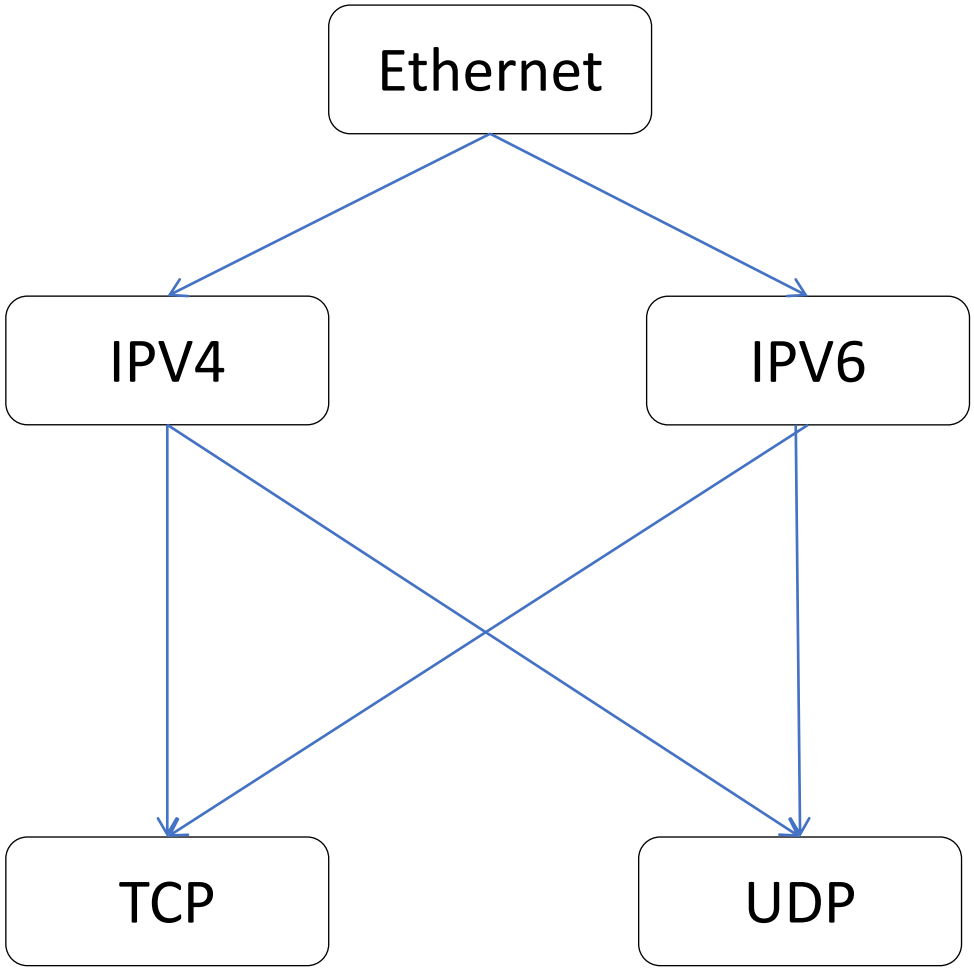
# Design Considerations

- Chip size
- High frequency
- Wiring and crossbars
- Amount of memory

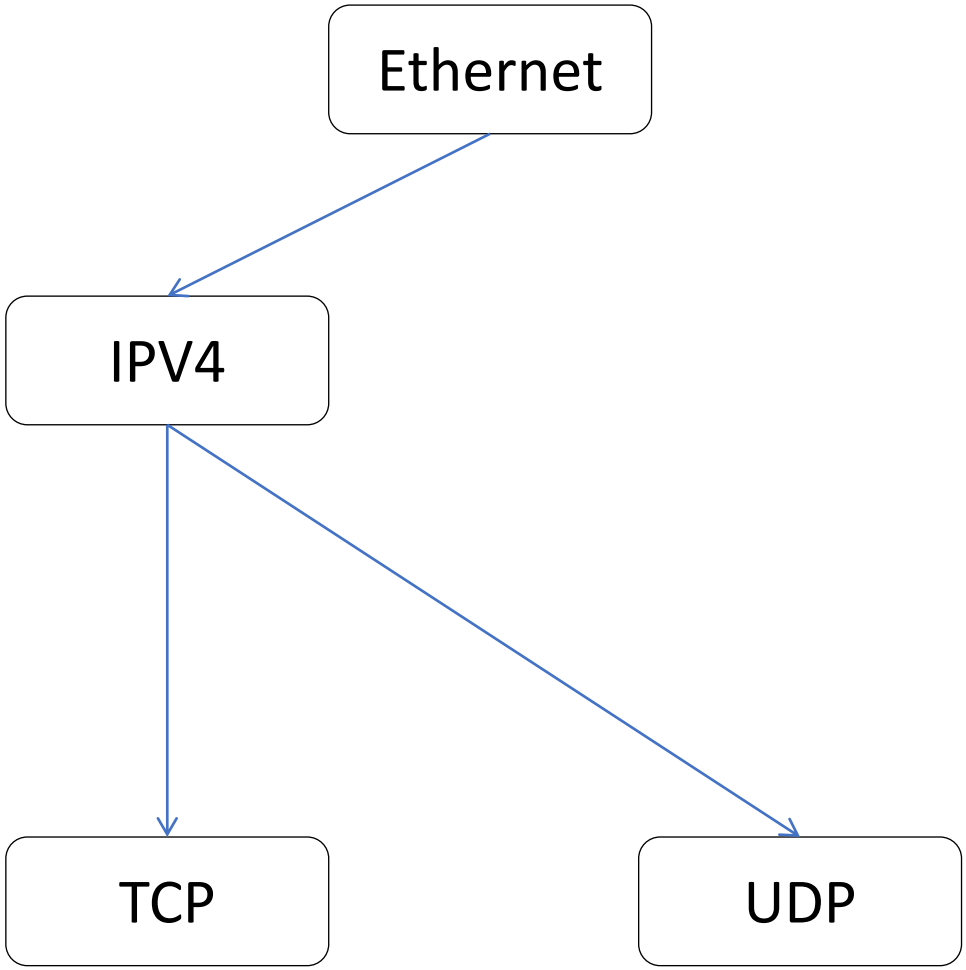
# The RMT Abstract Model

- Parse graph
- Table graph

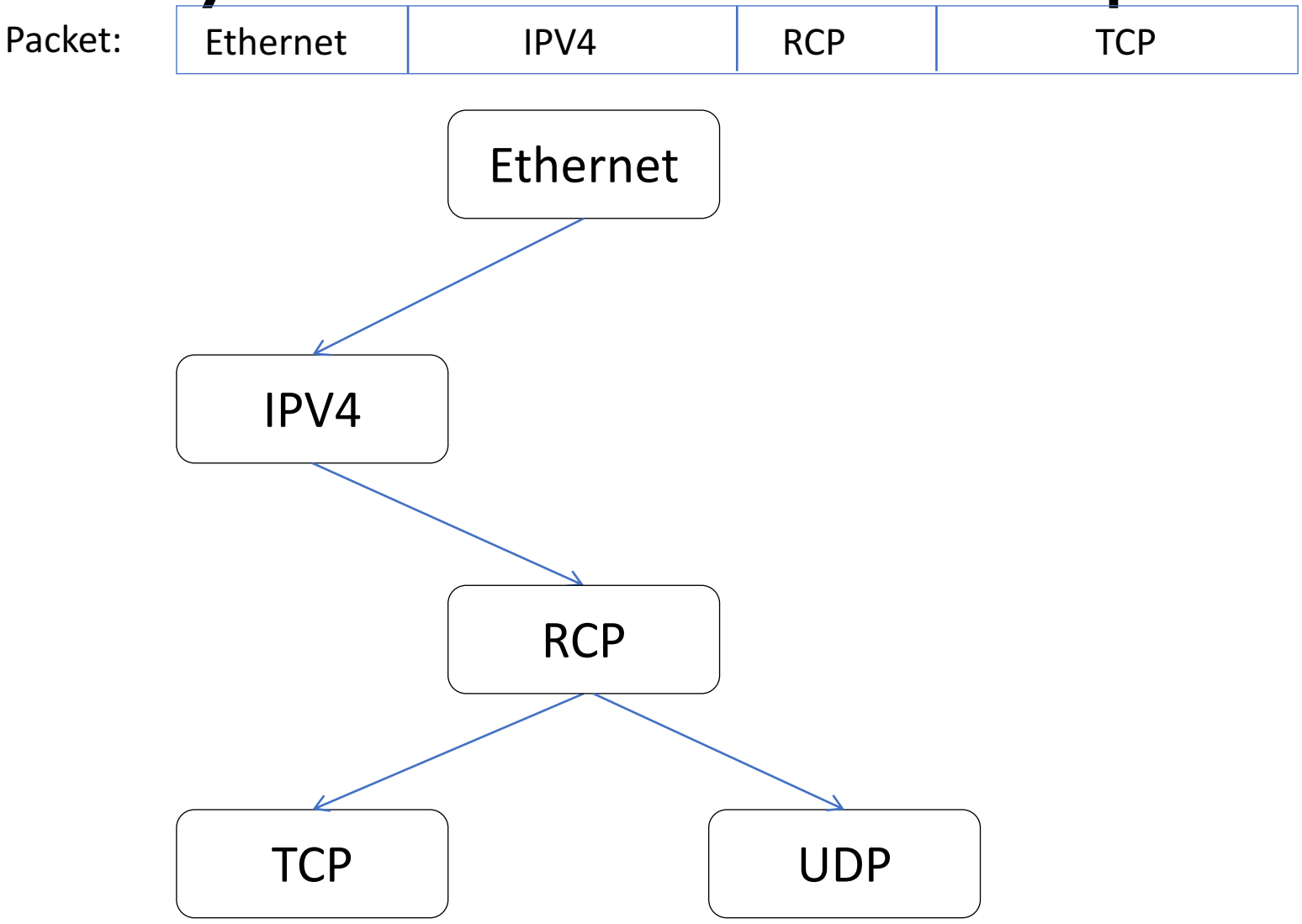
# Arbitrary Fields: The Parse Graph

# Arbitrary Fields: The Parse Graph

# Arbitrary Fields: The Parse Graph



# Arbitrary Fields: Programmable Parser

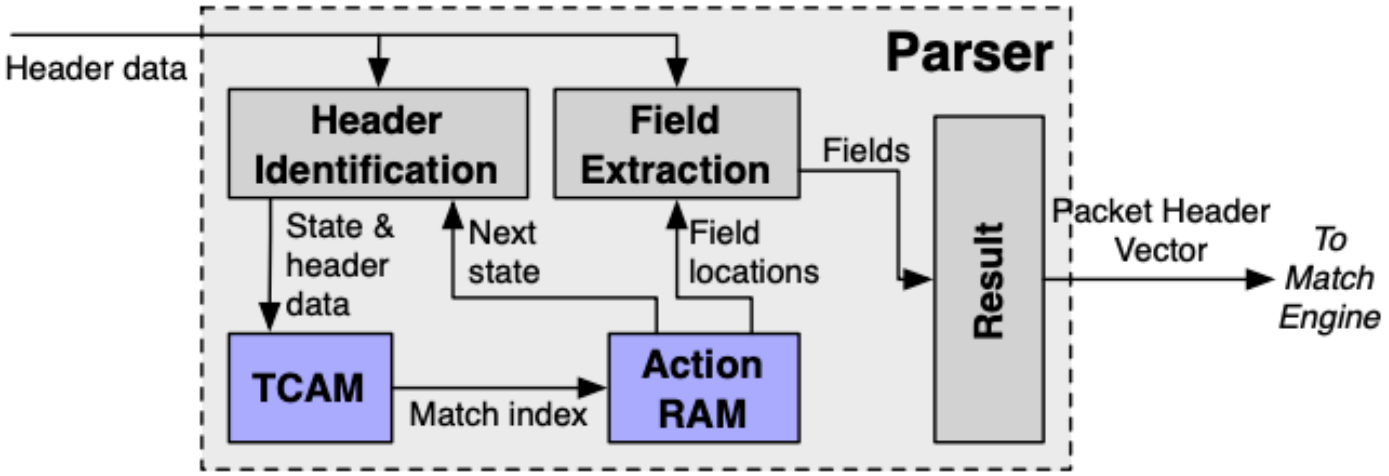
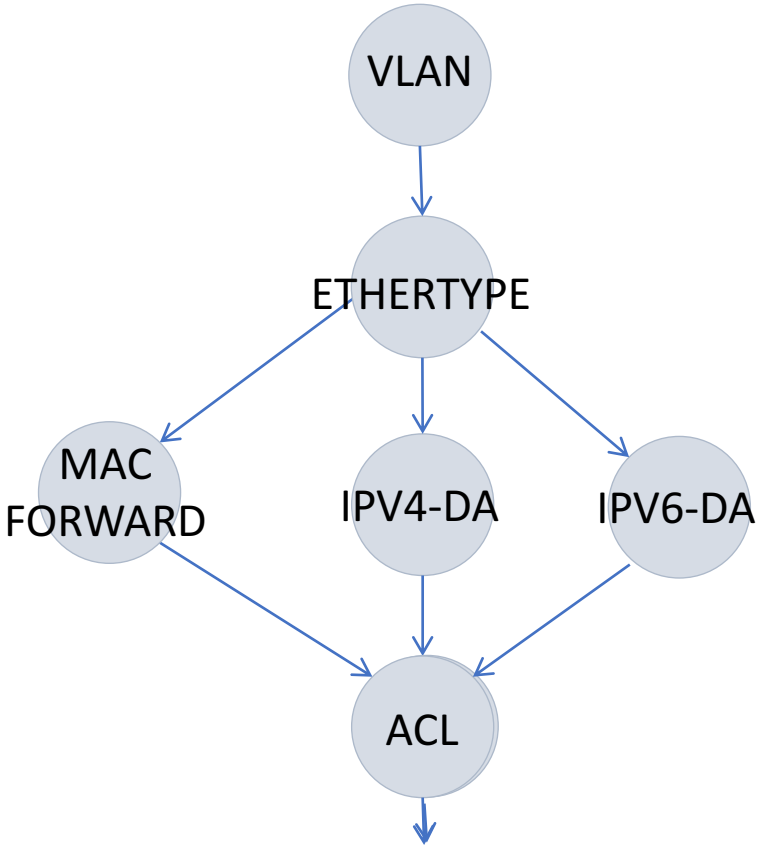
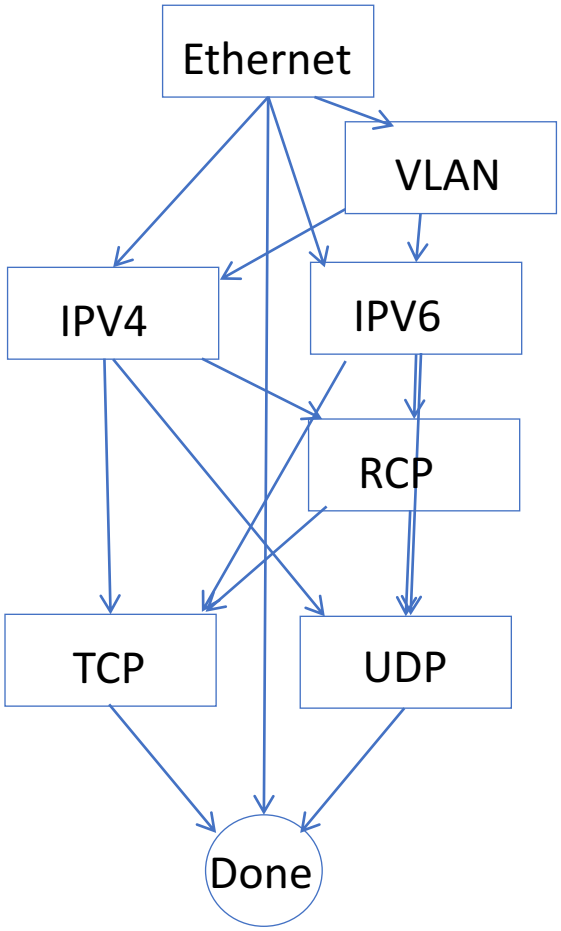



Figure 4: Programmable parser model.

# Reconfigurable Match Tables: The Table Graph



# Changes to Parse Graph and Table Graph




Parse Graph

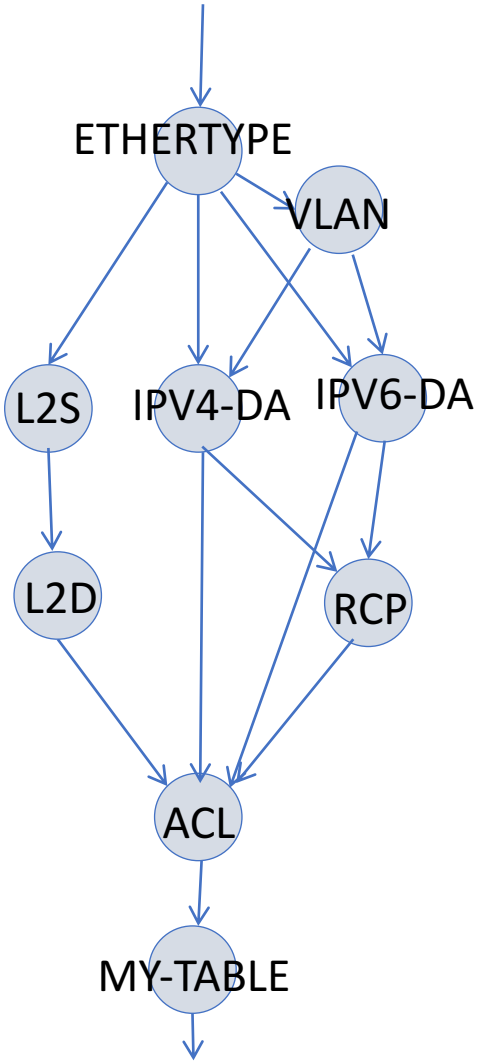
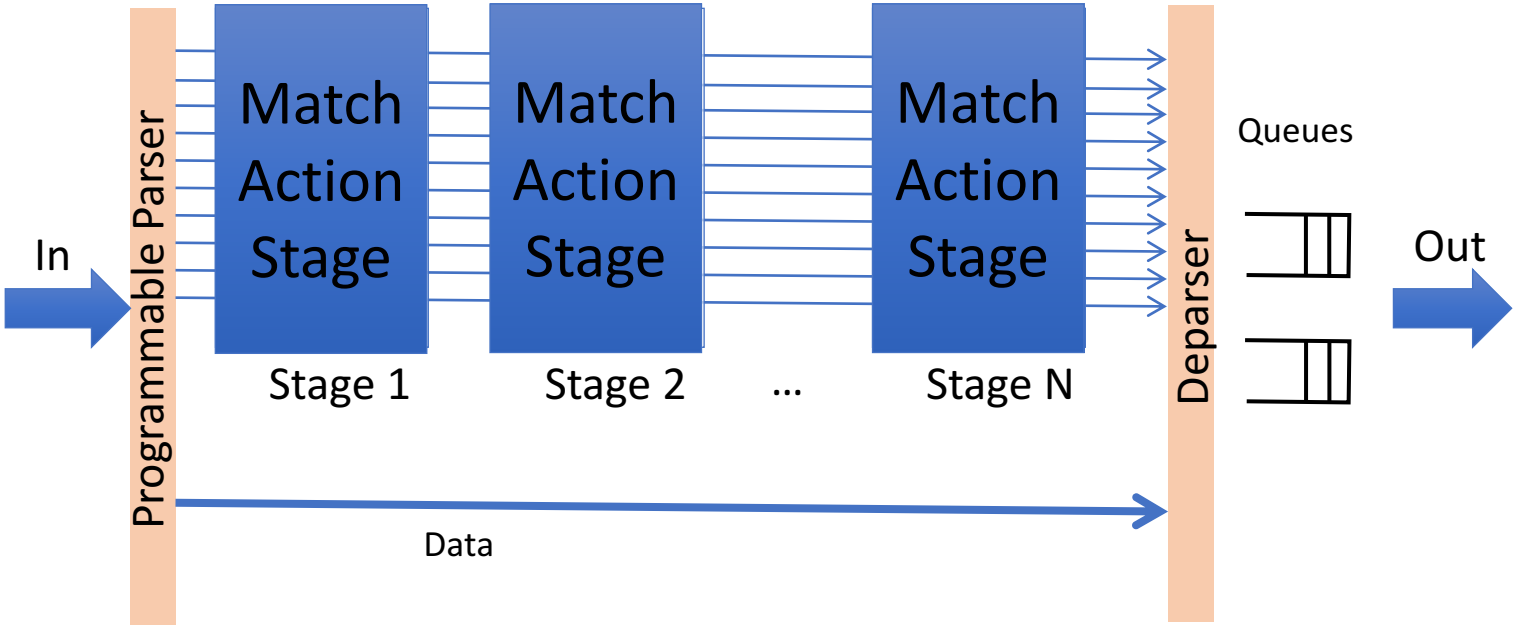


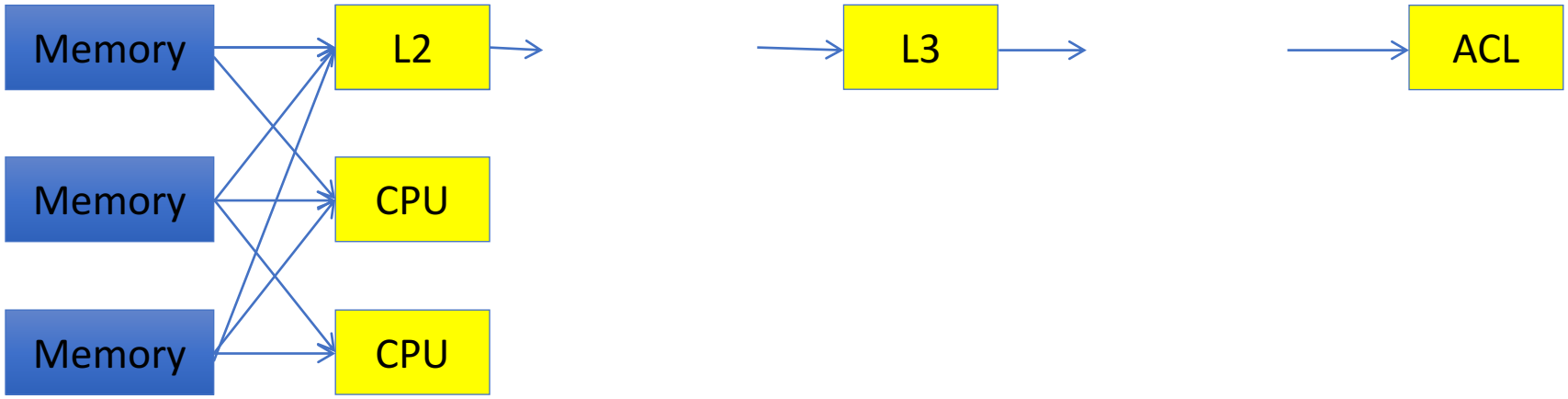

Table Graph

# Match/Action Forwarding Model

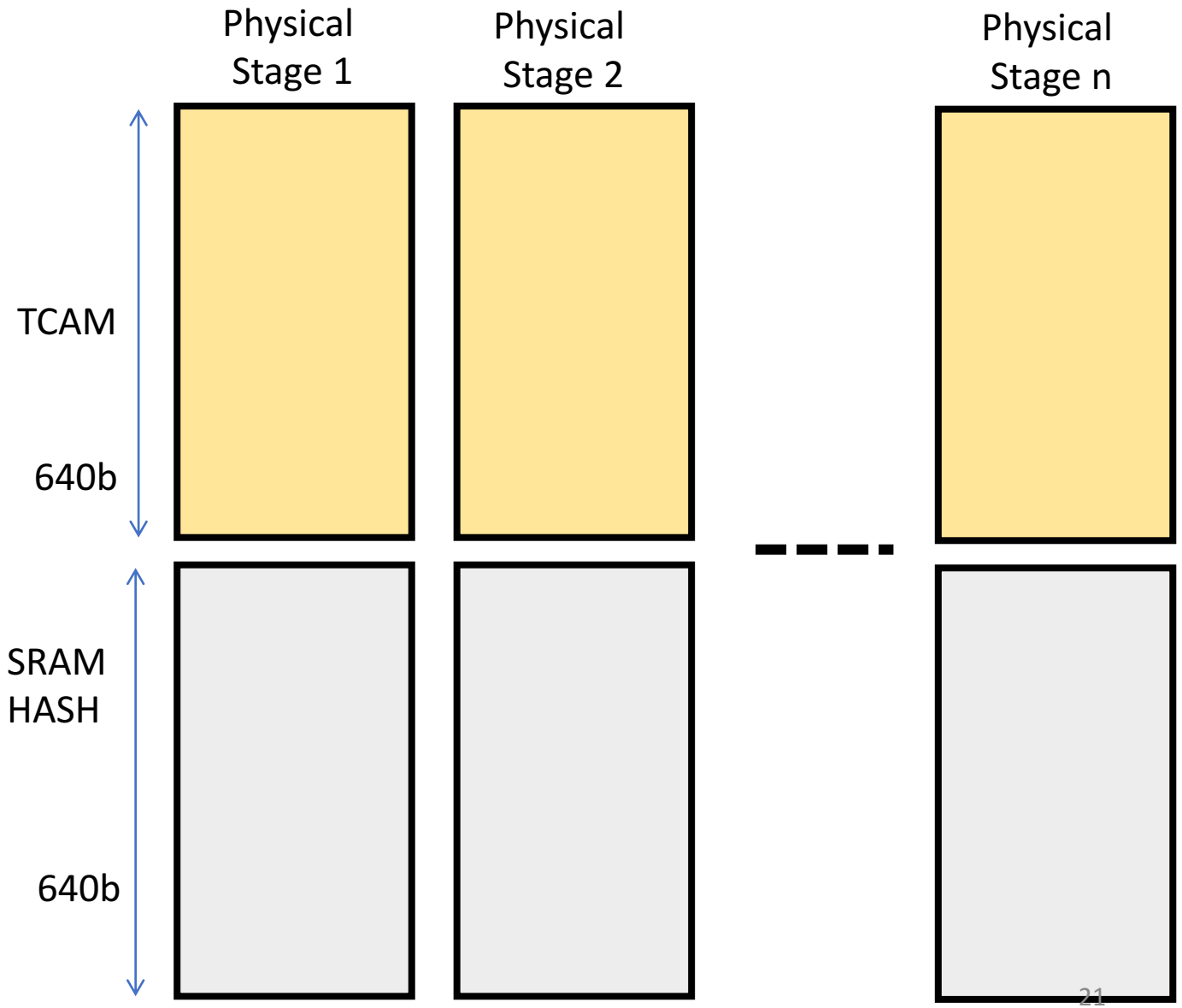


# Feature 1: flexible compute close to memory

- Multiprocessor: memory bottleneck
- Change to pipeline
- Fixed function chips specialize processors
- Flexible switch needs general purpose CPUs



# Feature 2: logical to physical mapping



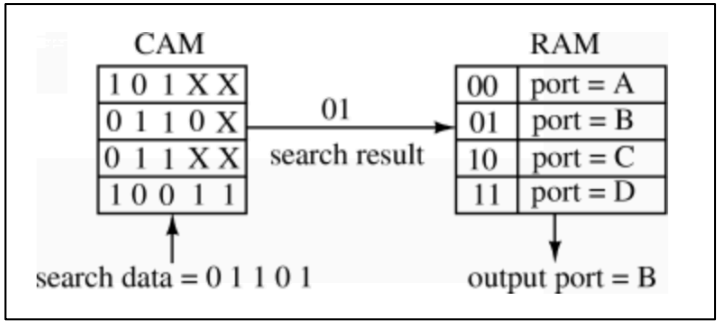
# Tiny Detour: CAMs and RAMs

- RAM:
  - Looks up the value associated with a memory address.

- CAM
  - Looks up memory address of a given value.
  - Two types:
    - Binary CAM: Exact match (matches on 0 or 1)
      - Can be implemented using SRAM.
    - Ternary CAM (TCAM): Allows wildcard (matches on 0, 1, or X).

# Tiny Detour: CAMs

| Line No. | Address (Binary) | Output Port |
|---|---|---|
| 1 | 101XX | A |
| 2 | 0110X | B |
| 3 | 011XX | C |
| 4 | 10011 | D |




Source: https://www.pagiamtzis.com/cam/camintro/

# Feature 2: logical to physical mapping

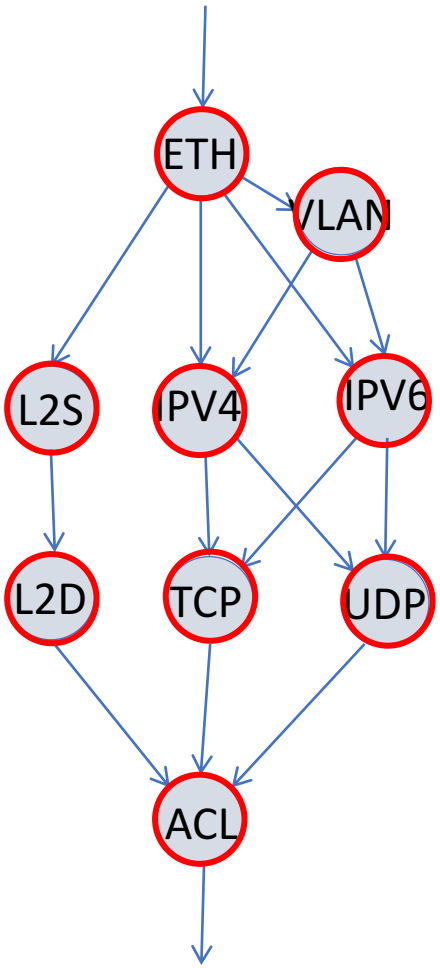
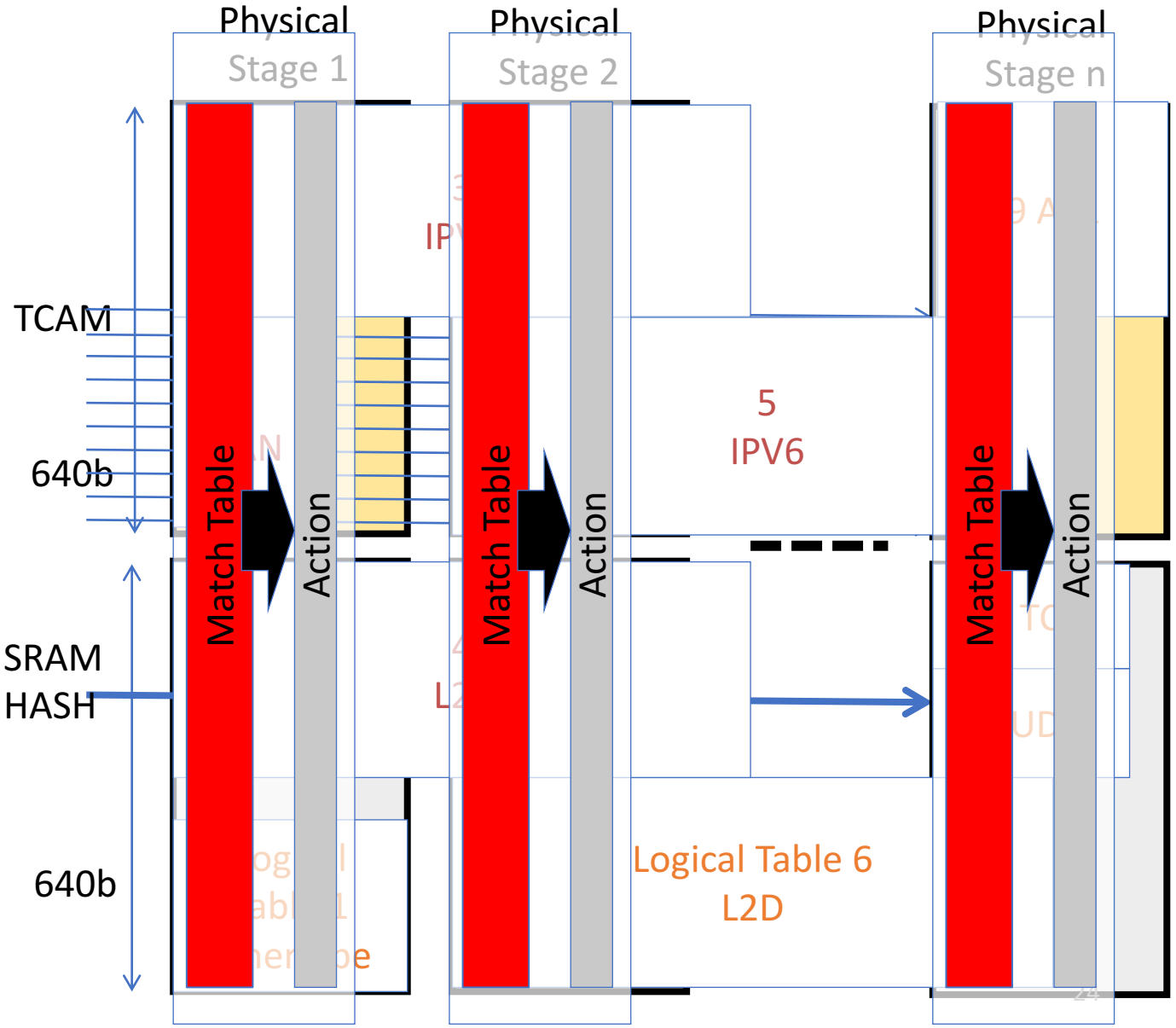
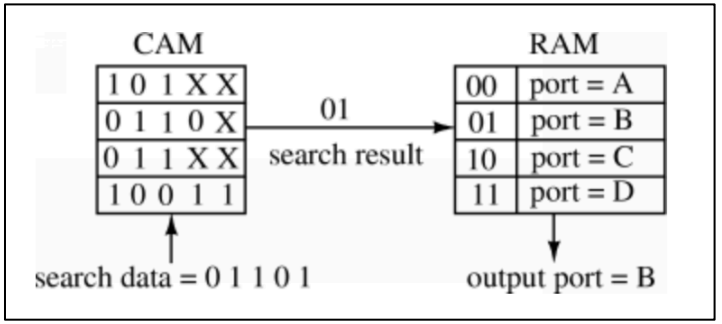


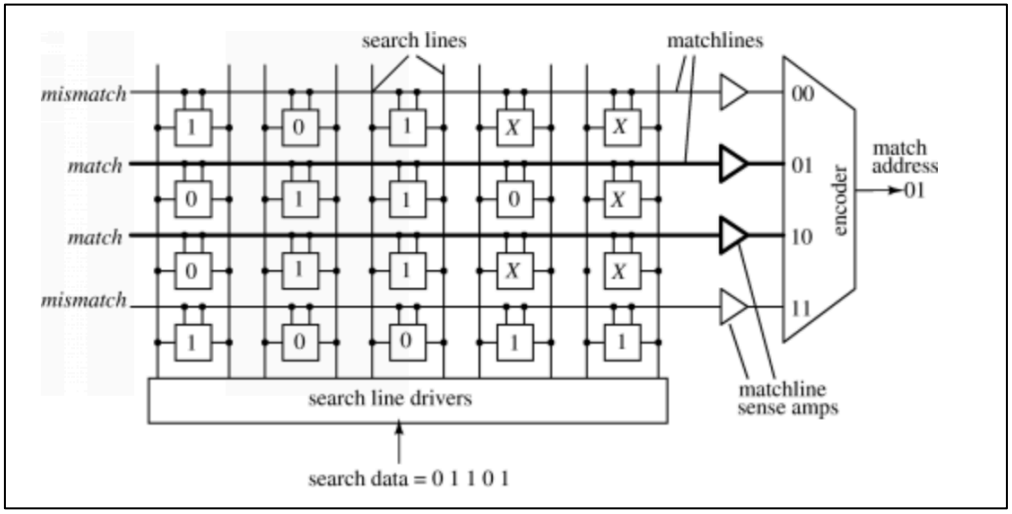

Table Graph



# Detour: CAMs

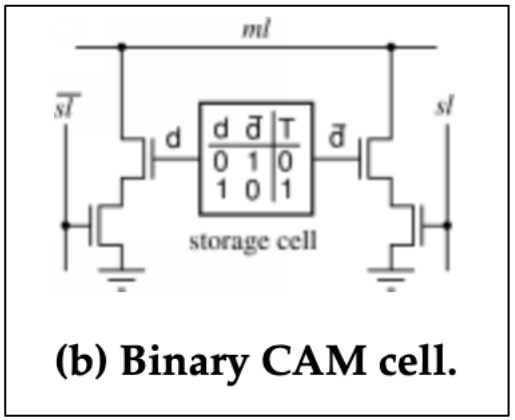
| Line No. | Address (Binary) | Output Port |
|---|---|---|
| 1 | 101XX | A |
| 2 | 0110X | B |
| 3 | 011XX | C |
| 4 | 10011 | D |

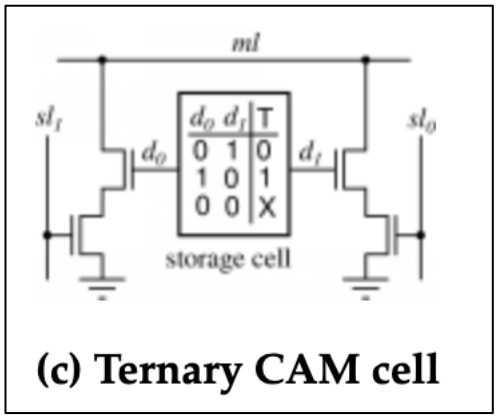


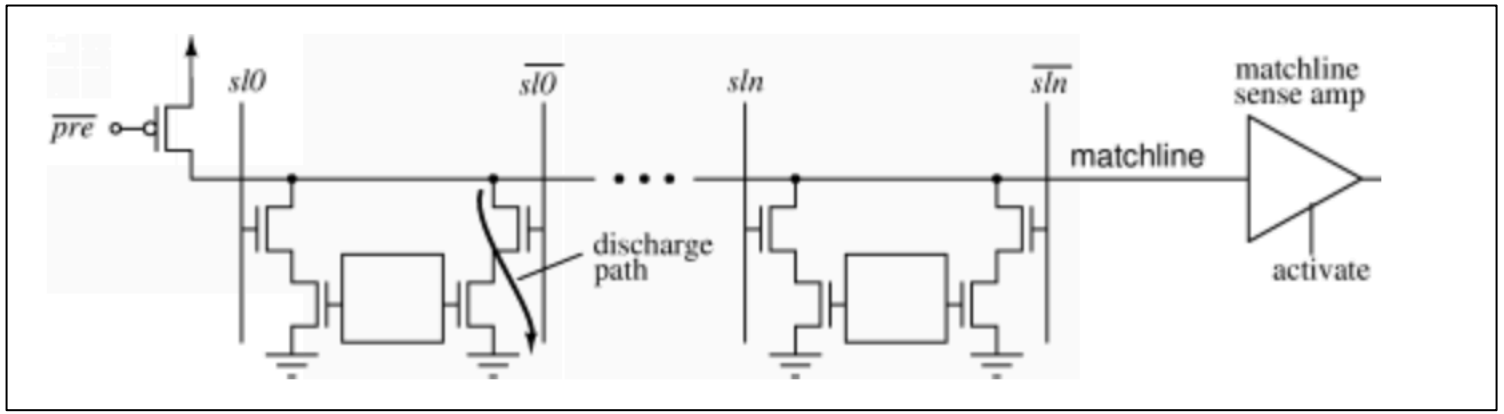




Source: https://www.pagiamtzis.com/cam/camintro/

# Detour: CAMs



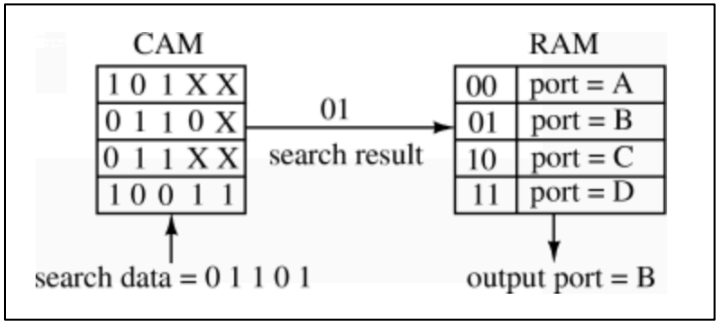



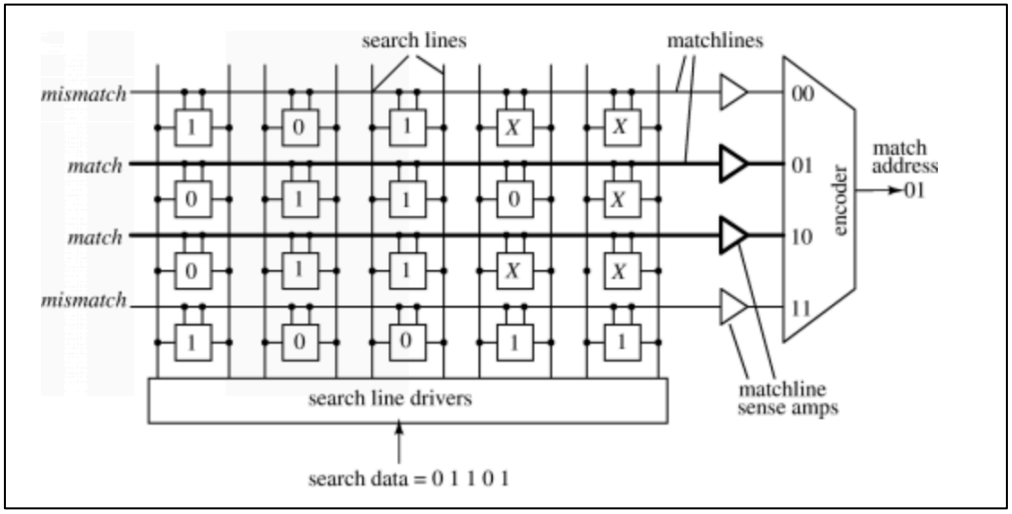




Source: https://www.pagiamtzis.com/cam/camintro/

# Detour: CAMs

| Line No. | Address (Binary) | Output Port |
|---|---|---|
| 1 | 101XX | A |
| 2 | 0110X | B |
| 3 | 011XX | C |
| 4 | 10011 | D |




Source: https://www.pagiamtzis.com/cam/camintro/

# RMT Logical to Physical Table Mapping

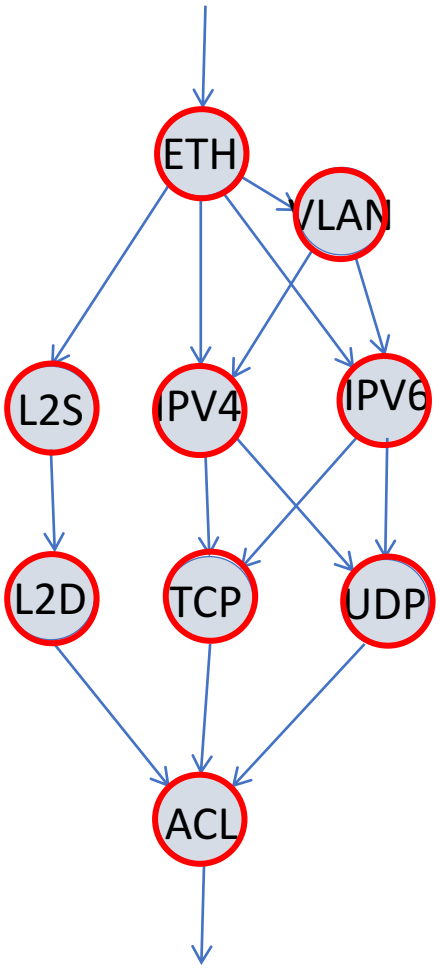
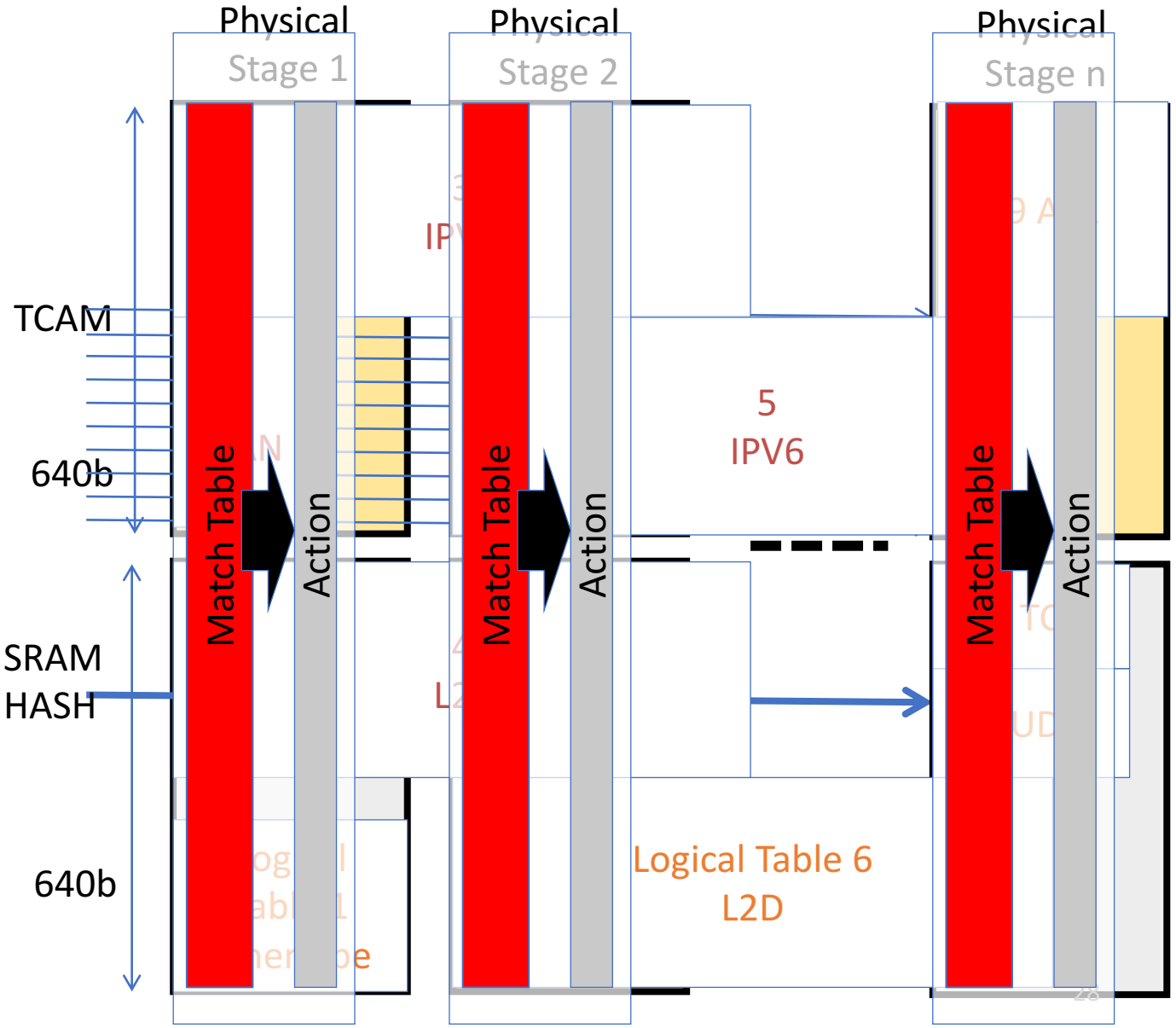
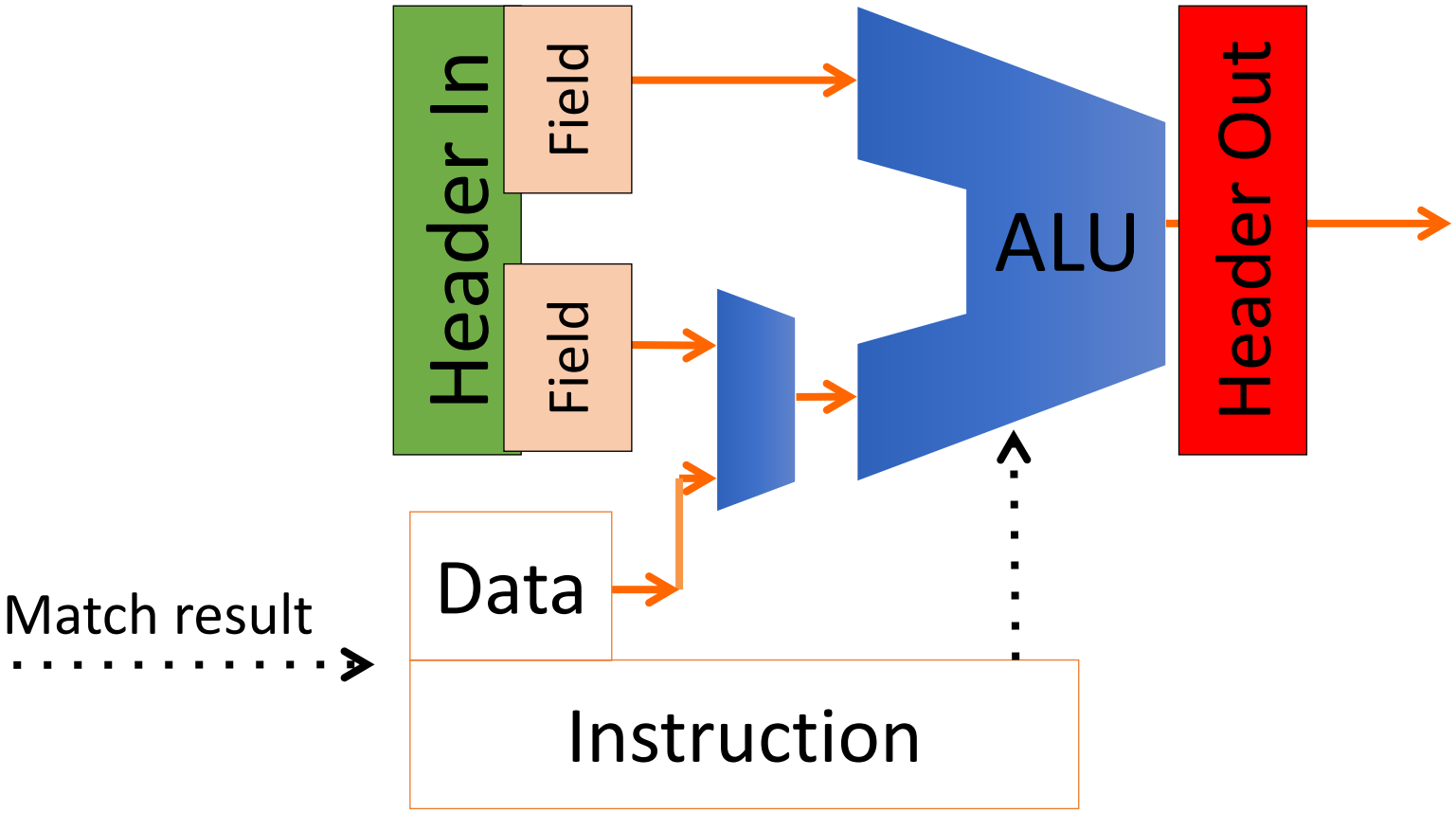


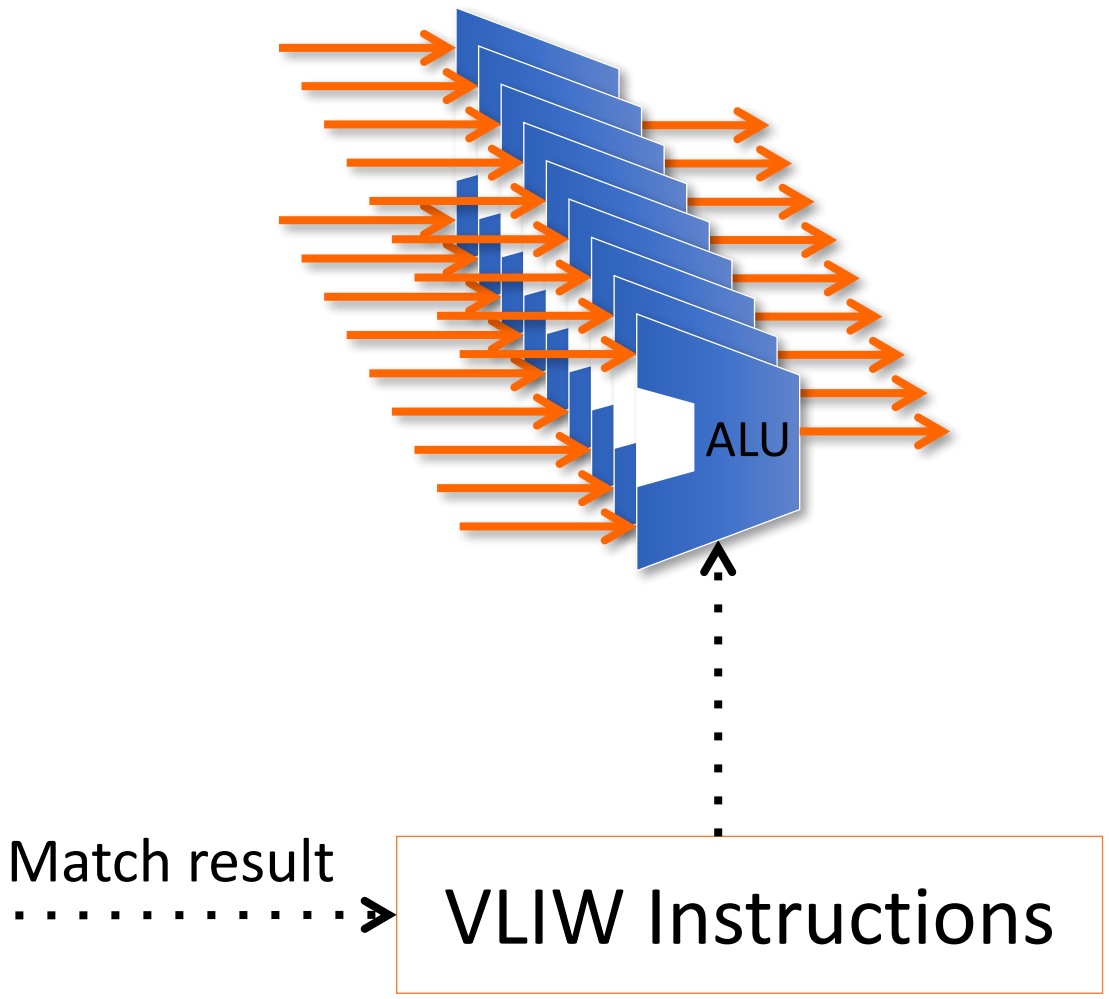

Table Graph



# Action Processing Model



# Modeled as Multiple VLIW CPUs per Stage



# RMT Switch Design

- 64 x 10Gb ports
  - 960M packets/second
  - 1GHz pipeline
- Programmable parser
- 32 Match/action stages

- Huge TCAM: 10x current chips
  - 64K TCAM words x 640b
- SRAM hash tables for exact matches
  - 128K words x 640b
- 224 action processors per stage
- All OpenFlow statistics counters

# Cost of Configurability: Comparison with Conventional Switch

- Many functions identical: I/O, data buffer, queueing…
- Make extra functions optional: statistics
- Memory dominates area
  - Compare memory area/bit and bit count
- RMT must use memory bits efficiently to compete on cost
- Techniques for flexibility
  - Match stage unit RAM configurability
  - Ingress/egress resource sharing
  - Allows multiple tables per stage
  - Match memory overhead reduction and multi-word packing

# Summary

- Conventional switch chip are inflexible
- SDN demands flexibility…sounds expensive…
- How do they do it: The RMT switch model
- *Flexibility costs less than 15%*

# Chip Comparison with Fixed Function Switches

Area

| Section | Area % of chip | Extra Cost |
|---|---|---|
| IO, buffer, queue, CPU, etc | 37% | 0.0% |
| Match memory & logic | 54.3% | 8.0% |
| VLIW action engine | 7.4% | 5.5% |
| Parser + deparser | 1.3% | 0.7% |
| **Total extra area cost** | | **14.2%** |

Power

| Section | Power % of chip | Extra Cost |
|---|---|---|
| I/O | 26.0% | 0.0% |
| Memory leakage | 43.7% | 4.0% |
| Logic leakage | 7.3% | 2.5% |
| RAM active | 2.7% | 0.4% |
| TCAM active | 3.5% | 0.0% |
| Logic active | 16.8% | 5.5% |
| **Total extra power cost** | | **12.4%** |

# Conclusion

- How to design a flexible chip?
  - The RMT switch model
  - Bring processing close to the memories:
    - pipeline of many stages
  - Bring the processing to the wires:
    - 224 action CPUs per stage
- How much does it cost?
  - 15%

How is this paradigm different from active networking?

# What are the limitations on flexibility?

# Since 2013….

- RMT switch has been commercialized
  - Barefoot Tofino
  - 6.5Tb/s
- Adoption of these switches?

# On research….

- Disaggregated RMT
  - SIGCOMM'17
- Runtime programmability
  - HotNets'21
- Enabling stateful processing
  - HotNets'20
- And many others….