Architectures for Real-Time Caching

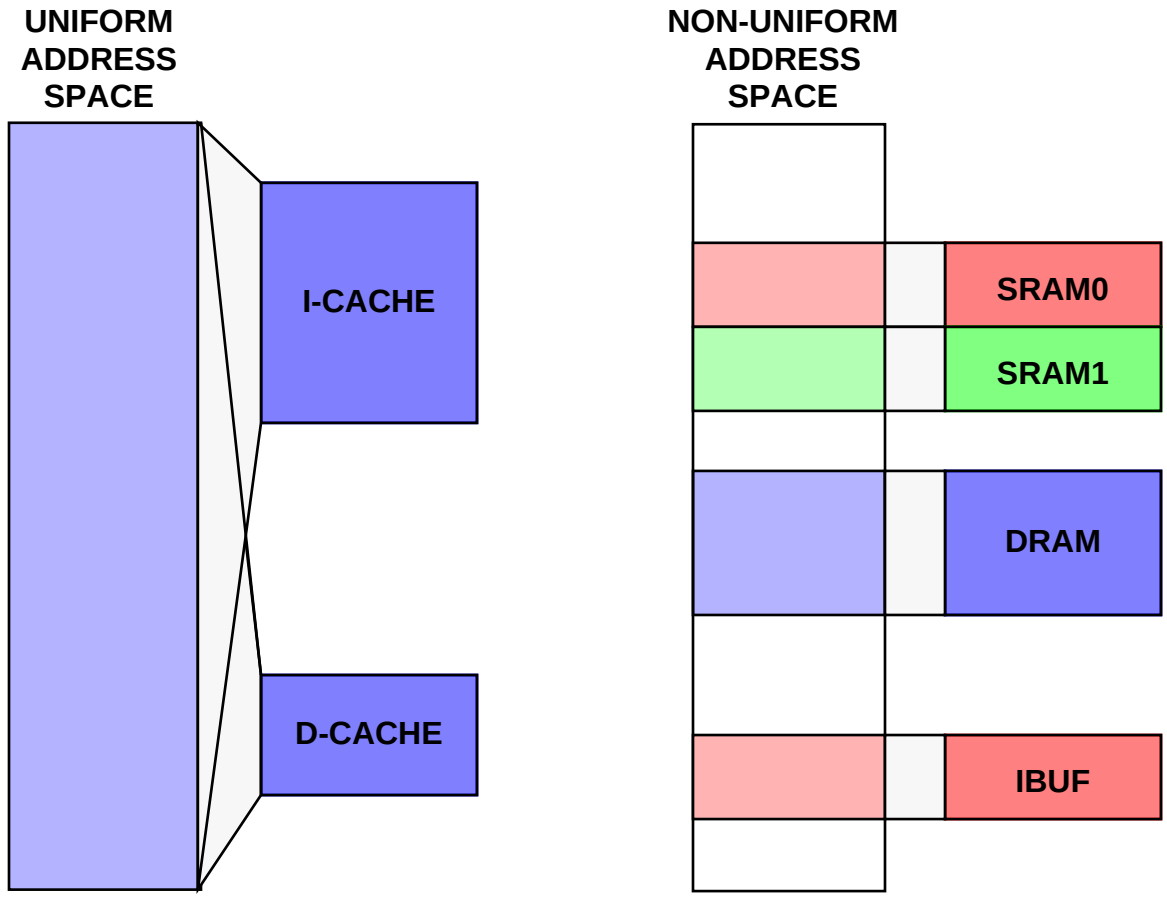
Bruce Jacob

Electrical & Computer Engineering
University of Maryland, College Park

OUTLINE:

- **Caches vs. Tagless SRAMs**
- **Data Placement and Its Complexity**
- **Solutions to the Problem**

Cache vs. Tagless SRAM



Traditional Caches **Tagless SRAMs**

MAIN DIFFERENCE: SRAM requires EXPLICIT MANAGEMENT

Tagless SRAMs

Addressing is impediment:

CONTIGUITY must be preserved

DISTANCE BETWEEN OBJECTS must be preserved

Multiply-accumulate requires TWO DISJOINT DATA SPACES

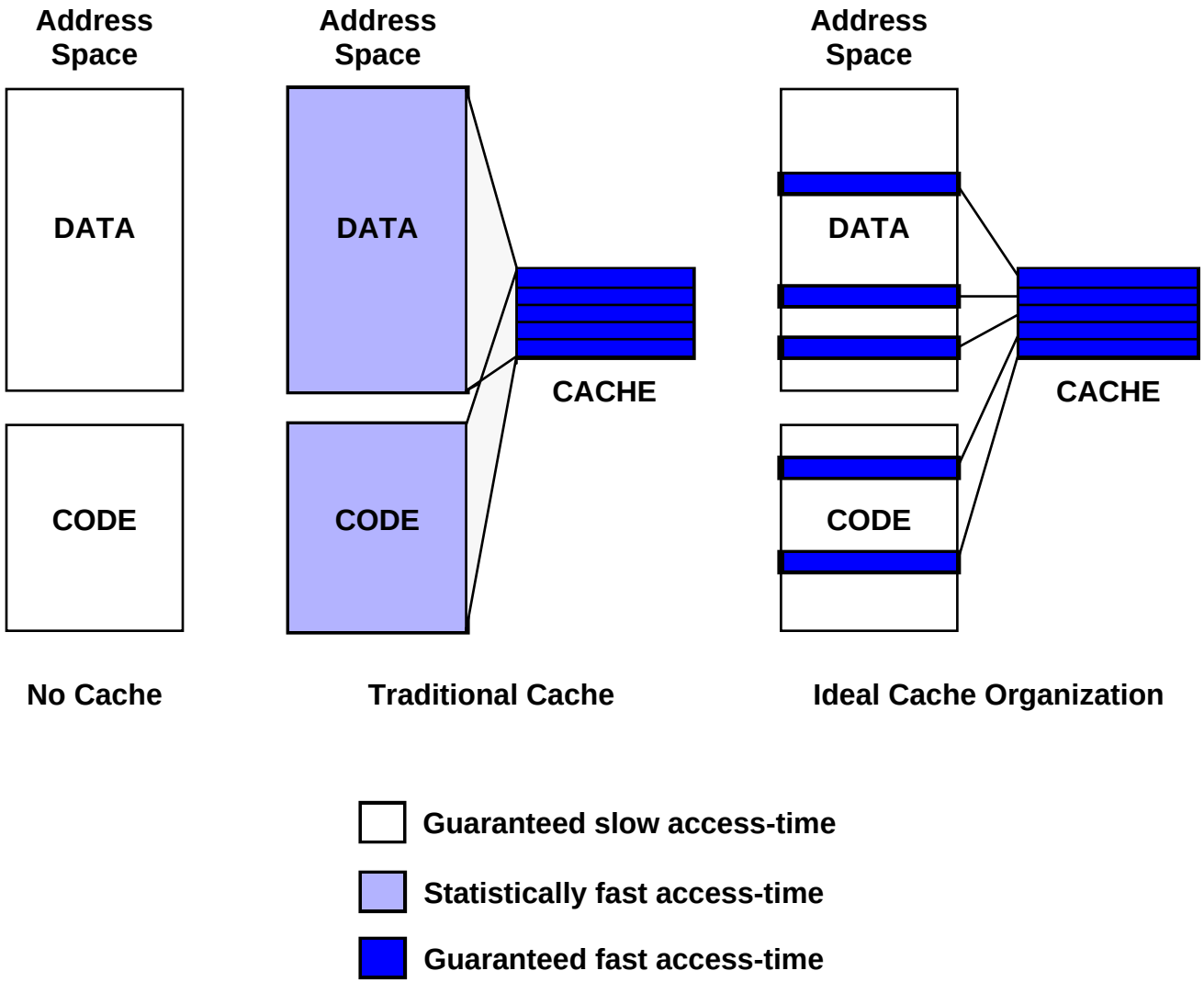
Bottom Line:

Access to memory NON-ORTHOGONAL
Separate spaces are DISJOINT

Result: COMPILATION IS HARD

Trend: UNIFORM ADDRESS SPACES

WHAT WE WANT

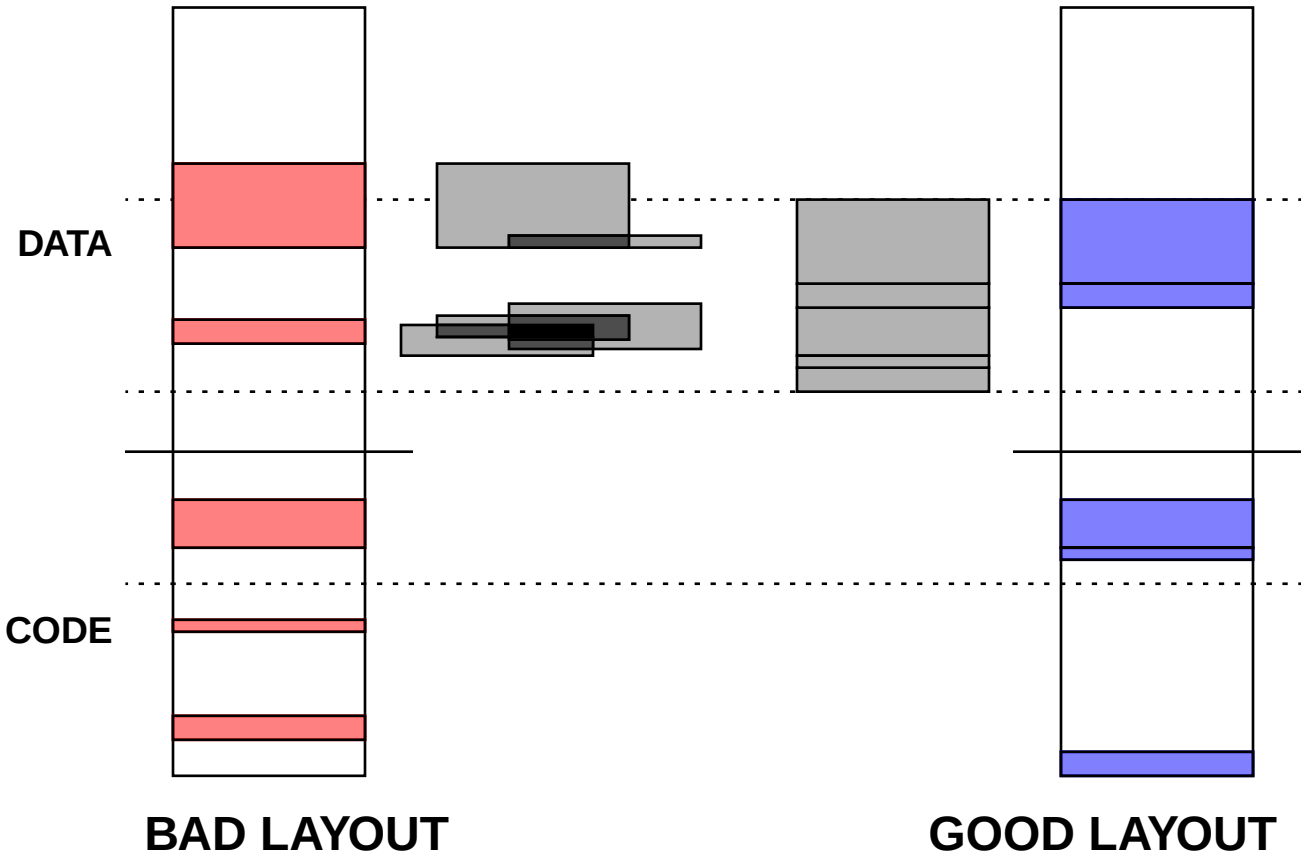

Address Space
DATA
CODE
No Cache
Address Space
DATA
CODE
CACHE
Traditional Cache
Address Space
DATA
CODE
CACHE
Ideal Cache Organization
Guaranteed slow access-time
Statistically fast access-time
Guaranteed fast access-time

WHY IT'S DIFFICULT

DATA NAME => DATA PLACEMENT

Must Group Data & Instructions

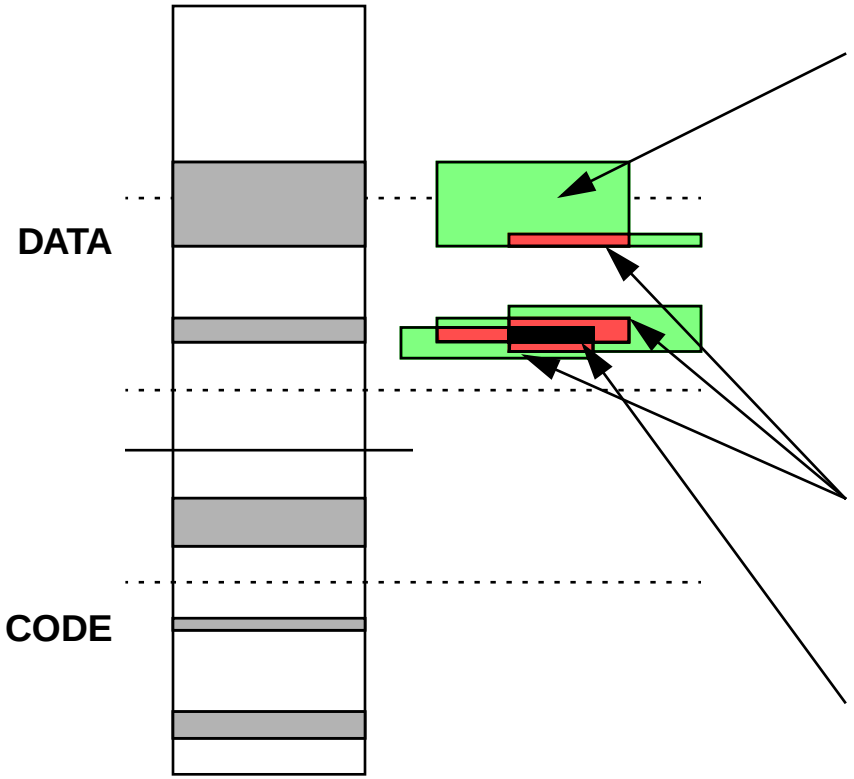
So as to Minimize Cache Conflicts



Solution #1

A BIG, HIGHLY ASSOCIATIVE CACHE

+ ability to PIN DOWN CACHE LINES



NO OVERLAP: Needs 1-way set assoc.

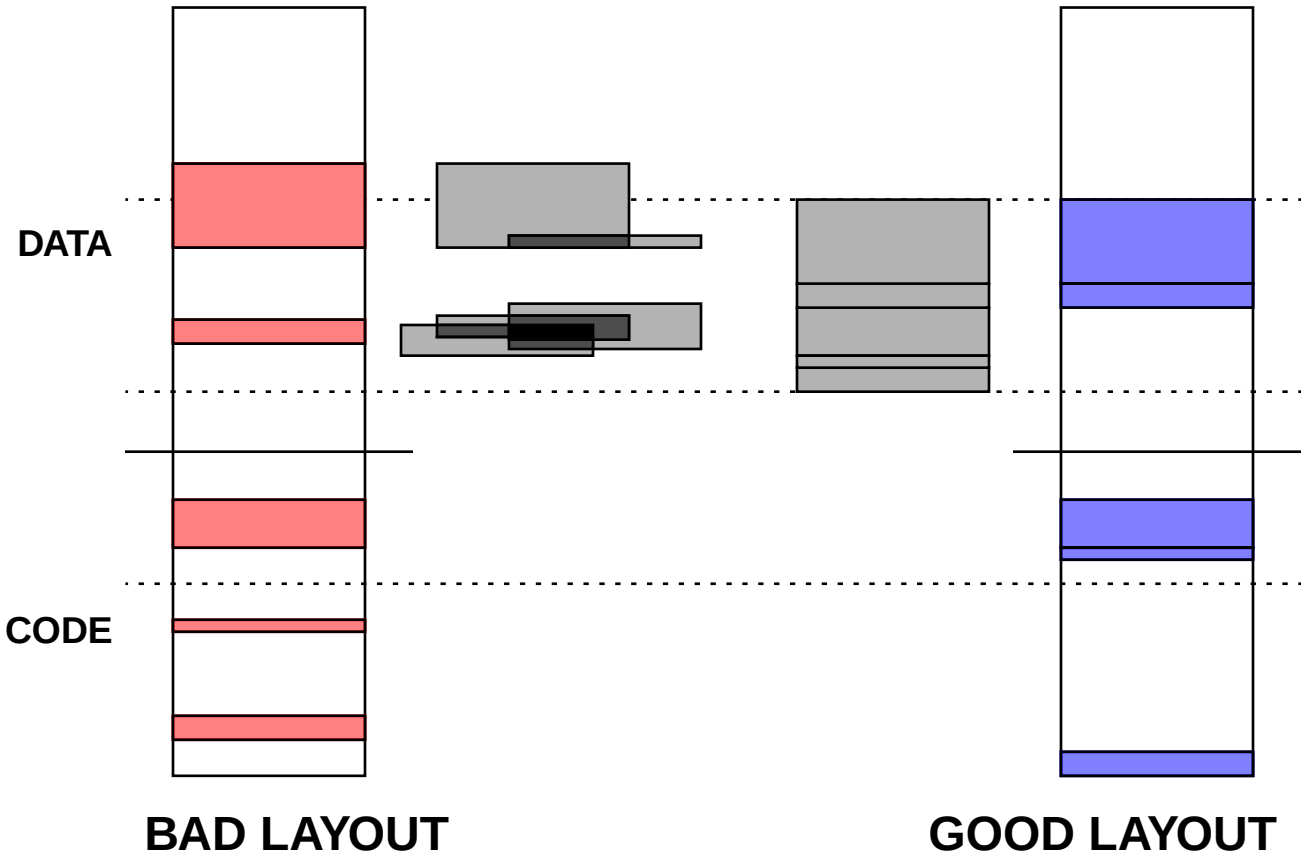
TWO-WAY OVERLAP: Needs 2-way set assoc.

THREE-WAY OVERLAP: Needs 3-way set assoc.

Enter Virtual Addressing

Disassociates NAME from PLACE

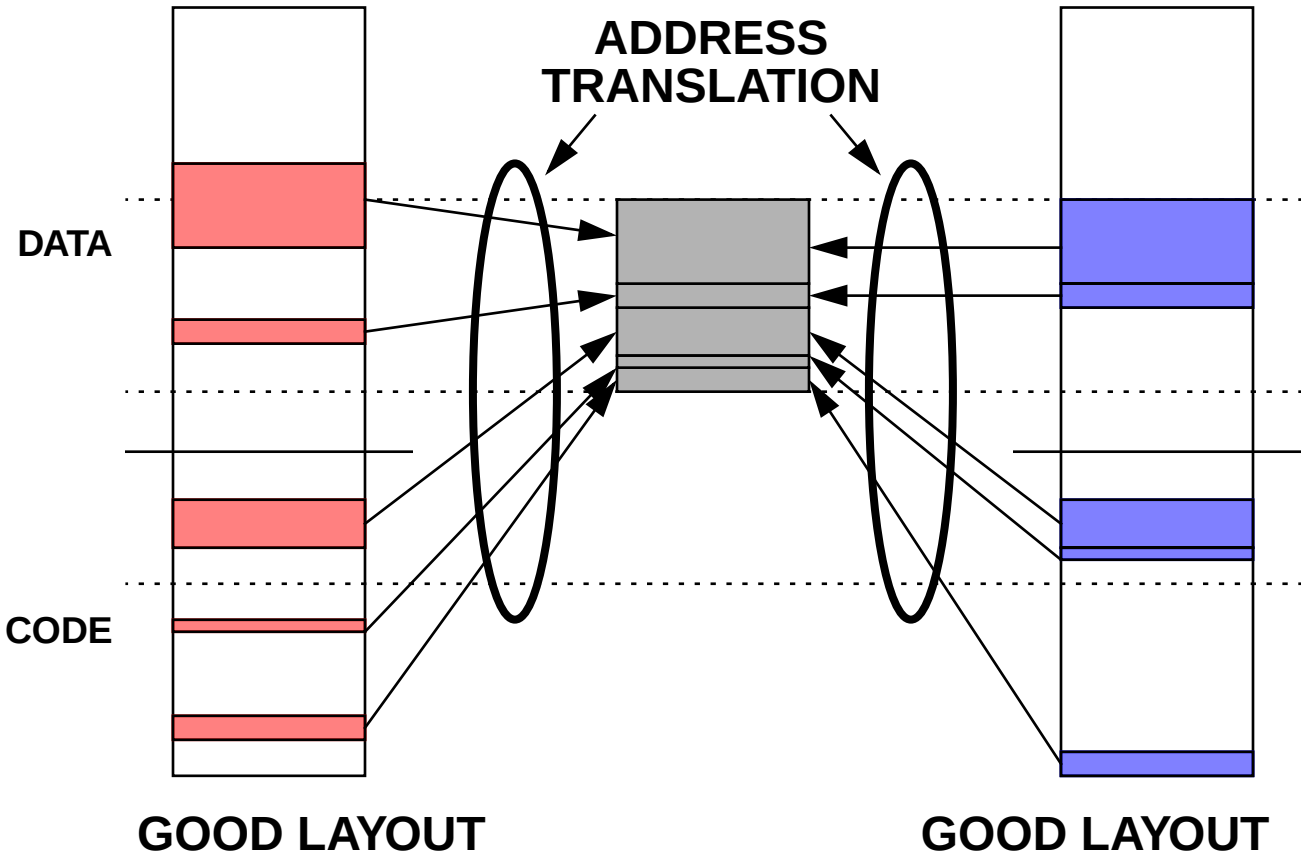
Allows you to go from THIS:



Enter Virtual Addressing

Disassociates NAME from PLACE

... to THIS:



Solution #2

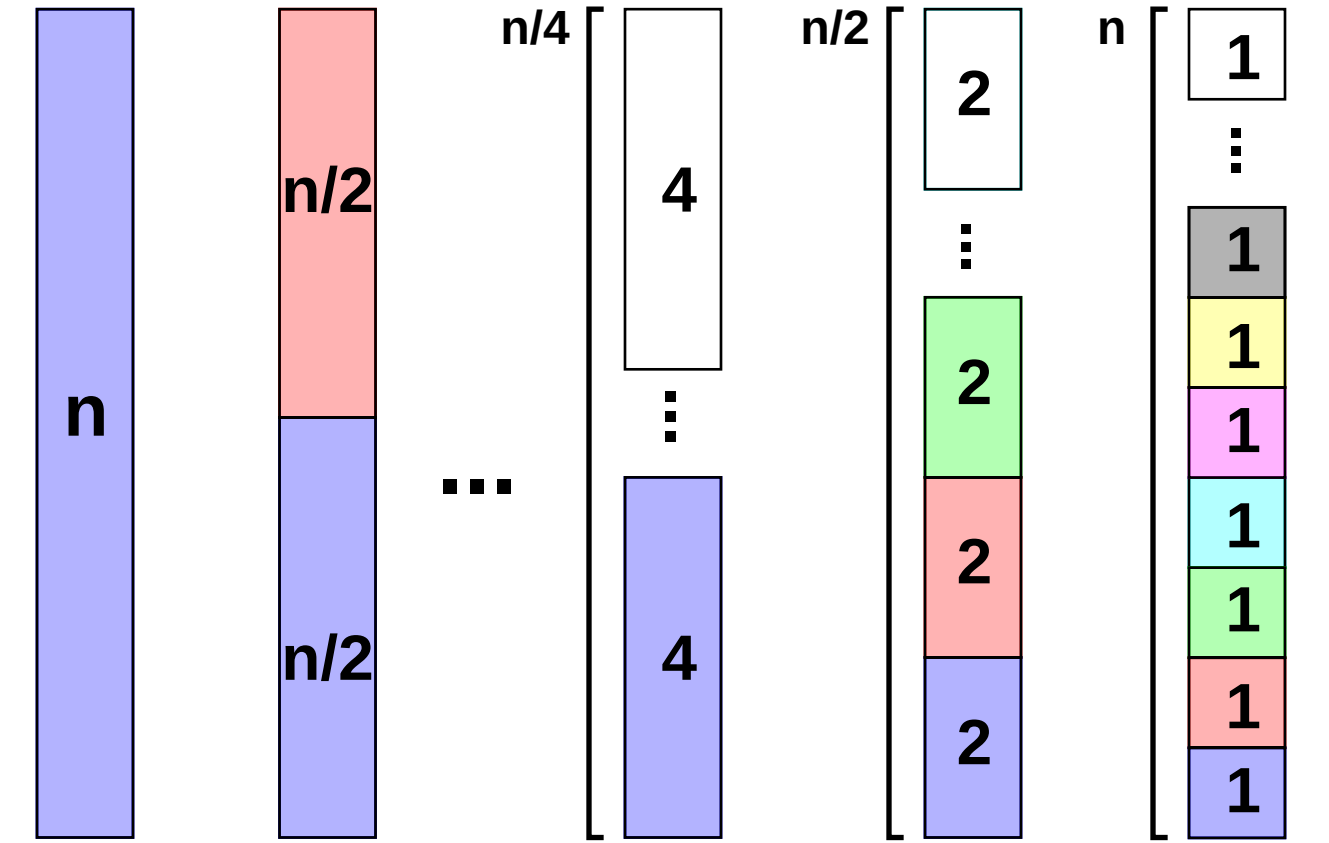
Fully-Associative Real-Time TLB
+ SW-Managed Tagless SRAM

- **TLB must fully map SRAM**
  (8KB SRAM, 256-byte page => 32 entries)
- **Can place ANY 256-byte page ANYWHERE in the SRAM**
- **Benefit: simple SRAM design**
- **Drawback: fully assoc. TLB**

Similar to software-managed FA cache with very large lines

Set-Associative RT-TLBs

Associativity vs. the Memory Space



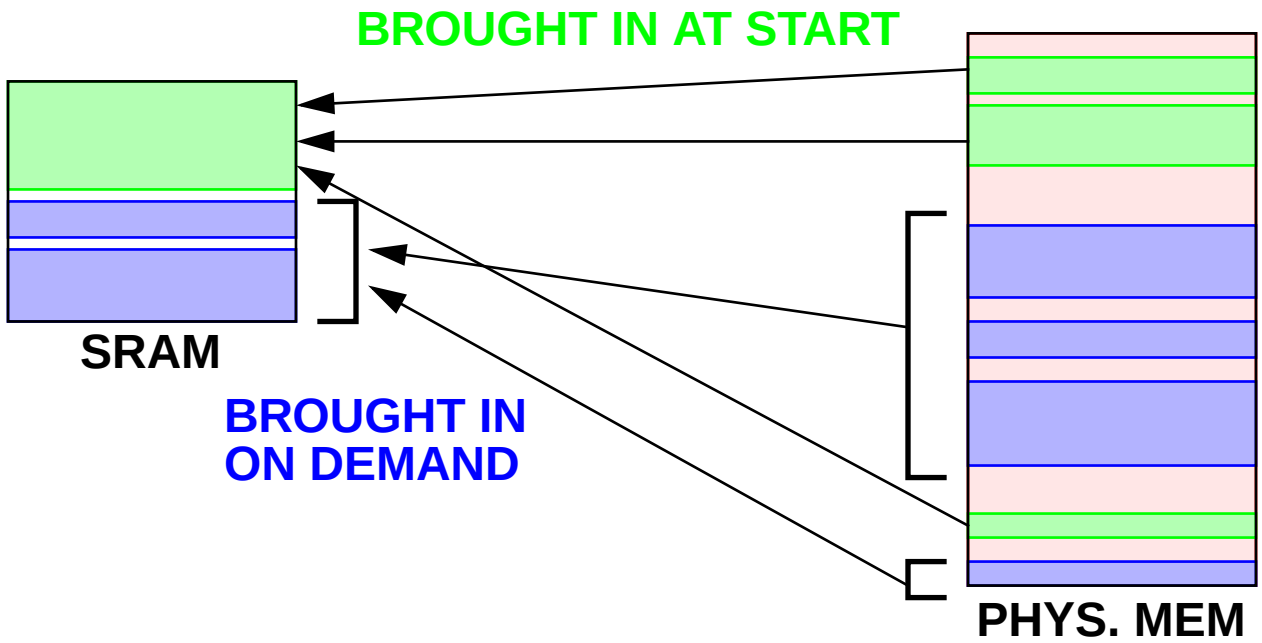
n = entries in TLB

Solution #3

What if SRAM Still Too Small?

(i.e. — previous solutions reduce CONFLICT problems, not CAPACITY problems)

Real-Time SRAM-Management



Summary

REAL-TIME CACHE ARCHITECTURES:

- **Really Big, Highly Associative Caches**
- **Virtual Addressing w/ RT-TLB**
- **Real-Time SRAM Management**

http://www.ece.umd.edu/~blj/