RENATO MANCUSO

THE SINGLE-CORE EQUIVALENCE (SCE) TECHNOLOGY PACKAGE

This research is supported by NSF under grant number #CNS-1302563

CMAAS @ CPSWeek 2017

MULTI-CORE PLATFORMS

2006+

MAINSTREAM

The major manufacturing companies produce multi-core systems for general-purpose computing.

2010

EMBEDDED

Multi-core become mainstream for embedded applications.

2016

FAA POSITION

FAA publishes CAST-32A position paper to address multi-core systems. **Robust Partitioning** + **SafetyNet.**

2017

NO STANDARD

No certification standard, no consolidated technology.

MULTI-CORE PLATFORMS

2006+

MAINSTREAM

The major manufacturing companies produce multi-core systems for general-purpose computing.

2010

EMBEDDED

Multi-core become mainstream for embedded applications.

2016

FAA POSITION

FAA publishes CAST-32A position paper to address multi-core systems. **Robust Partitioning** + **SafetyNet.**

2017

~ 1.7 million lines of code in a F-22 Fighter Jet

~ 6.5 million lines of code in a Boeing 787

~ 20 million lines of code in S Class Mercedes-Benz

INTRODUCTION

MULTI-CORE CHALLENGES

PORTING / INTEGRATION

How can existing code-bases be reused when adopting multi-cores?

PREDICTABILITY

How to achieve a level of predictability that is equivalent to single-cores without excessive pessimism?

UNDERSTANDING

Multi-cores are significantly more complex machines. Can sufficient understanding be achieved for a safe use?

CERTIFICATION

How to certify multi-core platforms? And how much will that cost?

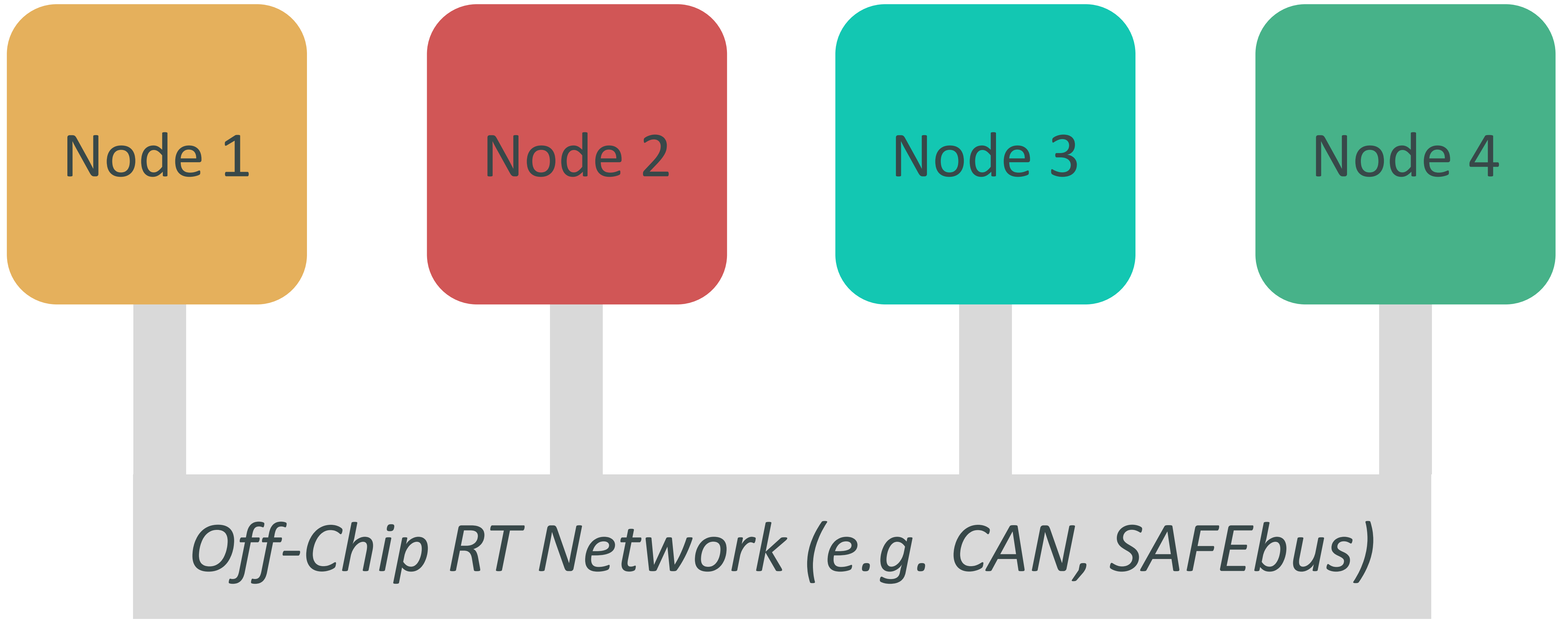
THE TIMING ISSUE

PREDICTABILITY

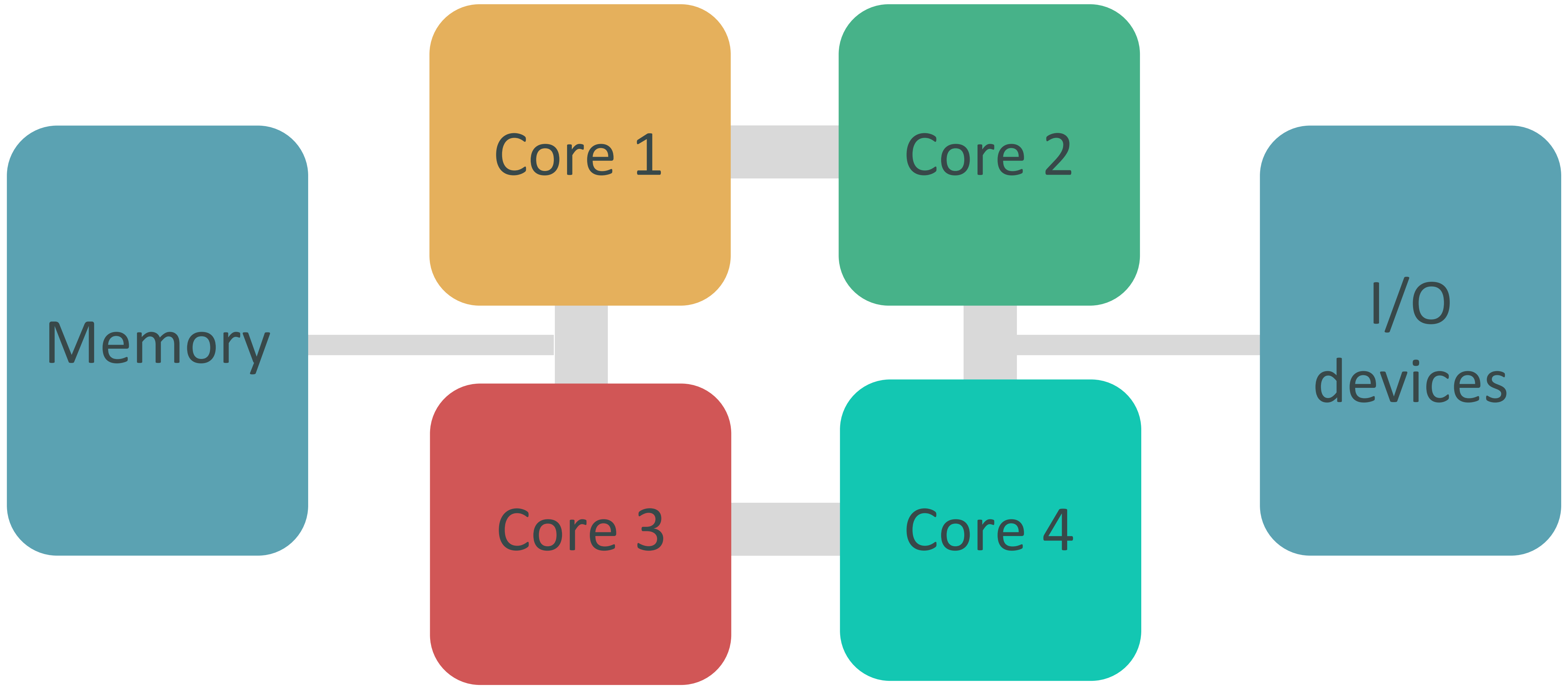
computation is performed in parallel, but

shared **I/O** and **memory**

From **multiple** single-core systems



To a **single** multi-core system

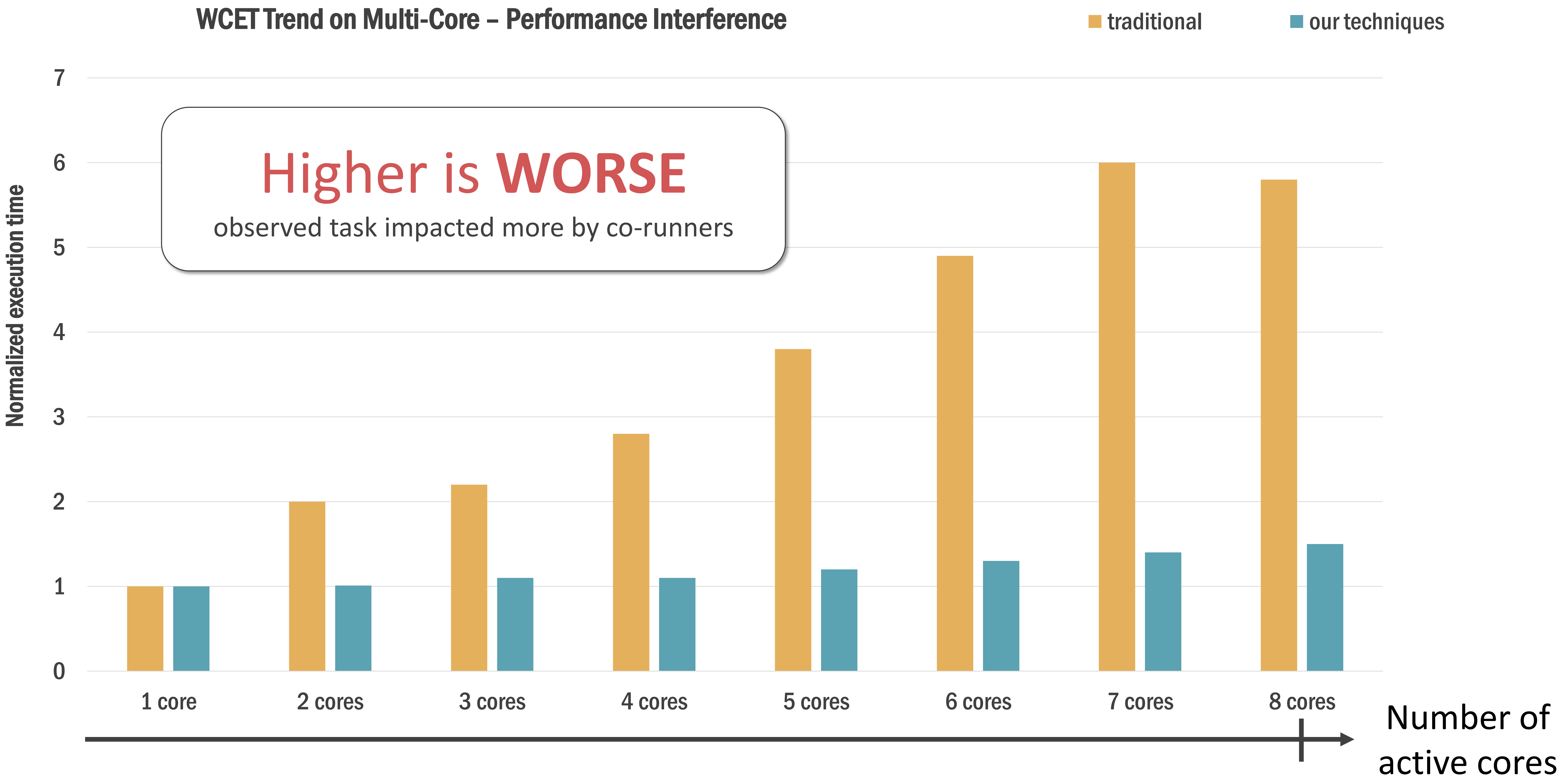


THE TIMING ISSUE

by LOCKHEED MARTIN

Setup

- Observe execution time of **single** task
- Run **independent** tasks on other cores (co-runners)
- Observed task is **memory intensive**
- Co-runners are **memory intensive**



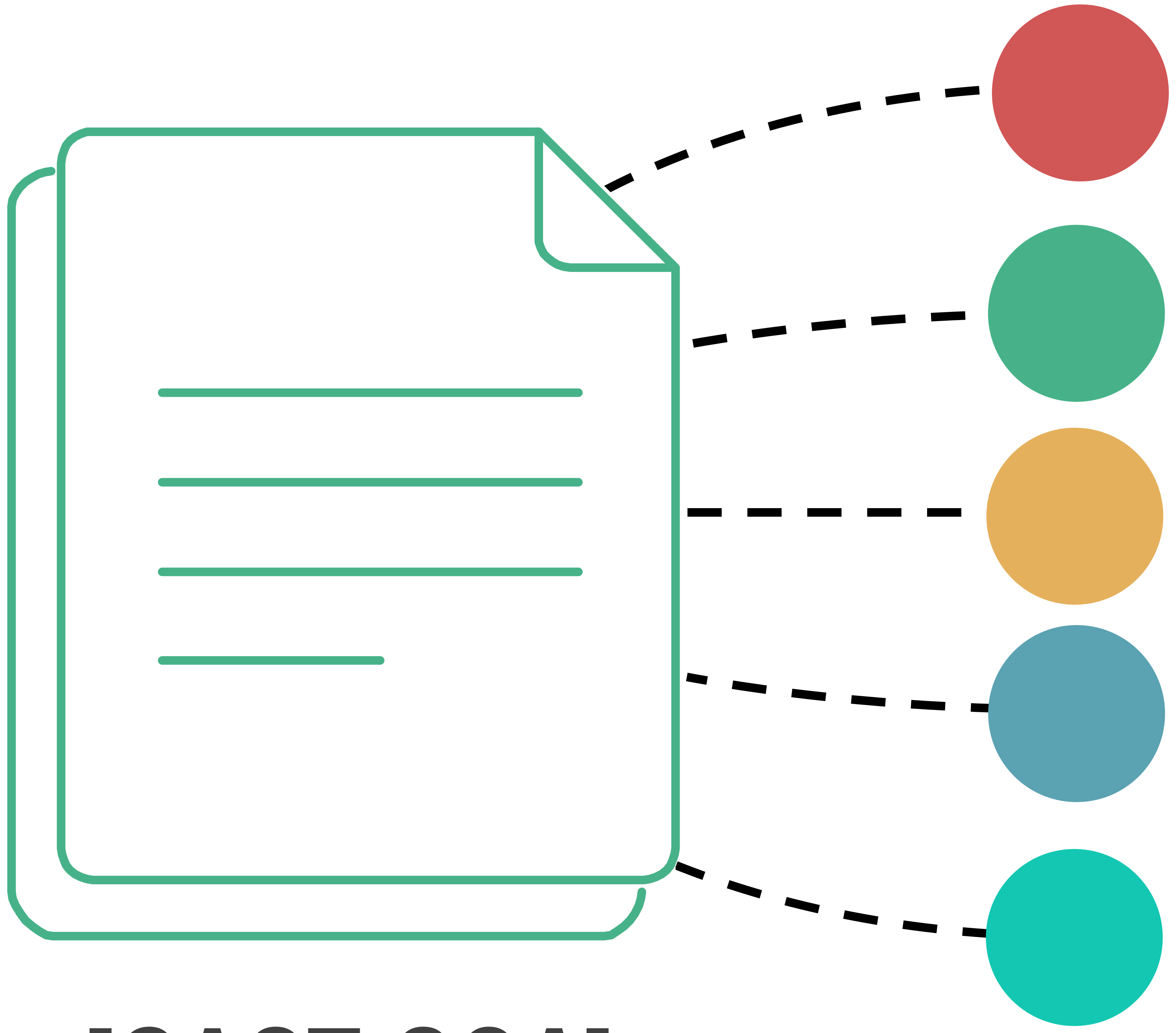
WCET Trend on Multi-Core – Performance Interference

Higher is **WORSE**

observed task impacted more by co-runners

SCE & CAST-32A

How SCE fits in the context of state-of-the-art multicore avionics certification guidance



MCP Platforms with **Robust Partitioning**

MCP_Planning_2: *how shared resources are used so as to **avoid** or **mitigate** the effect of **contention***

MCP_Resource_Usage_3: ***identification** of interference channels (shared memory, cache, interconnect, I/O) and **means of mitigation***

MCP_Resource_Usage_4: ***identification** of resources, their **allocation**, and **verification** that usage does not exceed limits*

MCP_Software_1: *with **robust partitioning** it is possible to "verify applications on the MCP and determine their WCETs **separately**"*

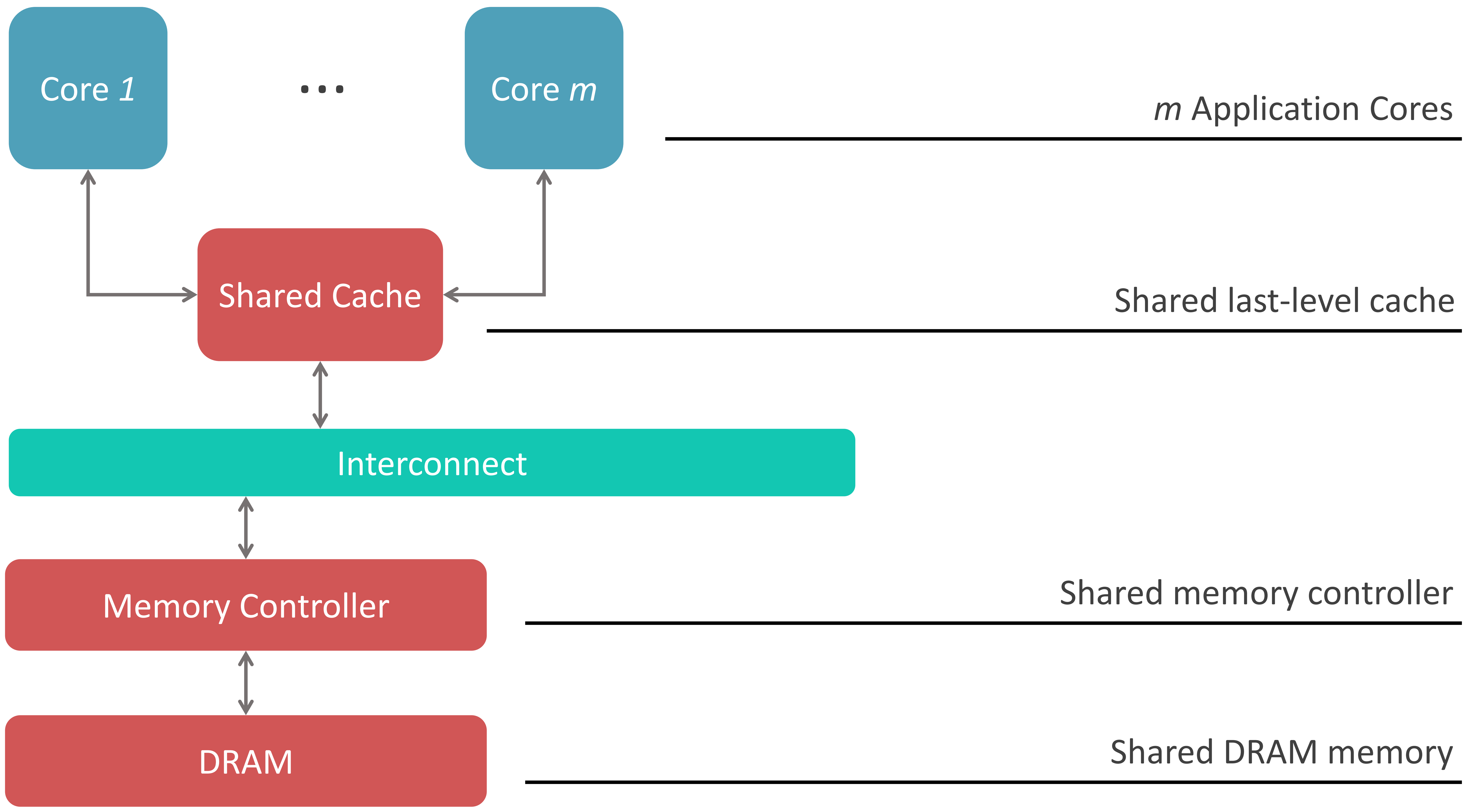
[CAST-32A]

November 2016

[CAST-32A] CAST Position Paper on Multi-core Processors, Certification Authorities Software Team (CAST), November 2016 (Rev 0). Available at: https://www.faa.gov/aircraft/air_cert/design_approvals/air_software/cast/cast_papers/media/cast-32A.pdf

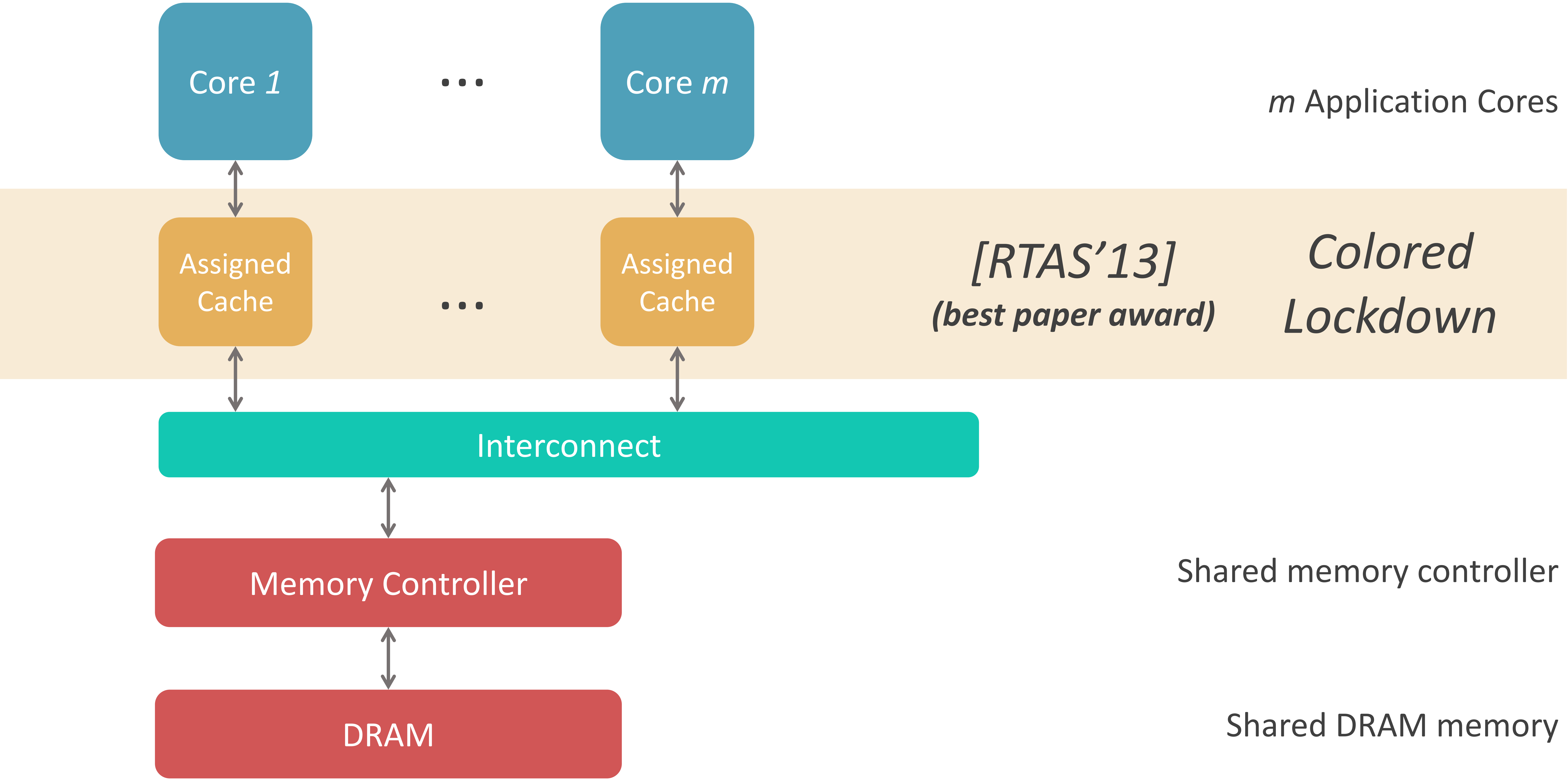
CACHE-BASED PLATFORMS

Instance of software reference architecture for commercial cache-based multi-core systems.



CACHE-BASED PLATFORMS

Instance of software reference architecture for commercial cache-based multi-core systems.



m Application Cores

[RTAS’13]
(best paper award)

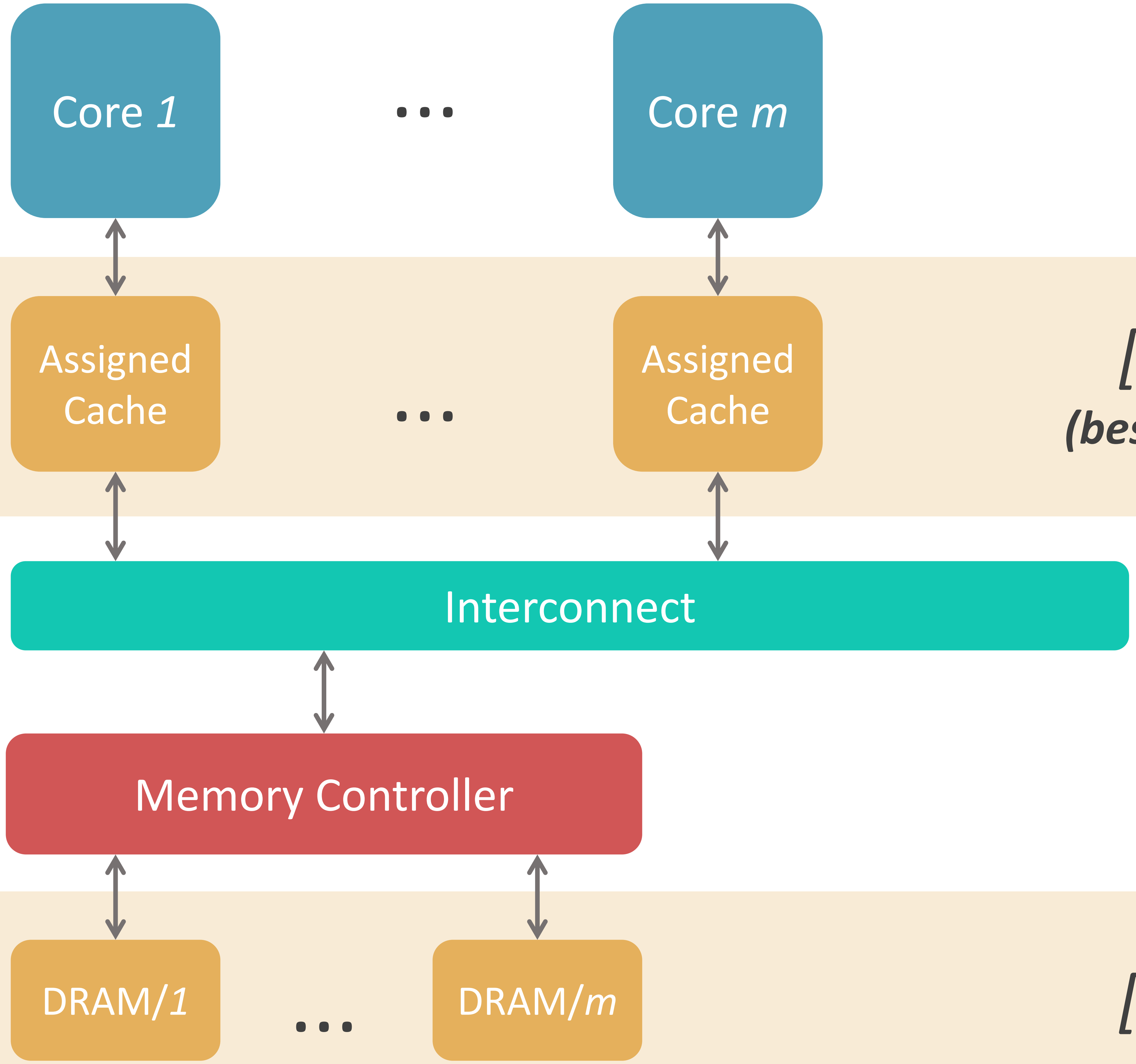
Colored Lockdown

Shared memory controller

Shared DRAM memory

CACHE-BASED PLATFORMS

Instance of software reference architecture for commercial cache-based multi-core systems.



m Application Cores

[RTAS'13]
(best paper award)

Colored Lockdown

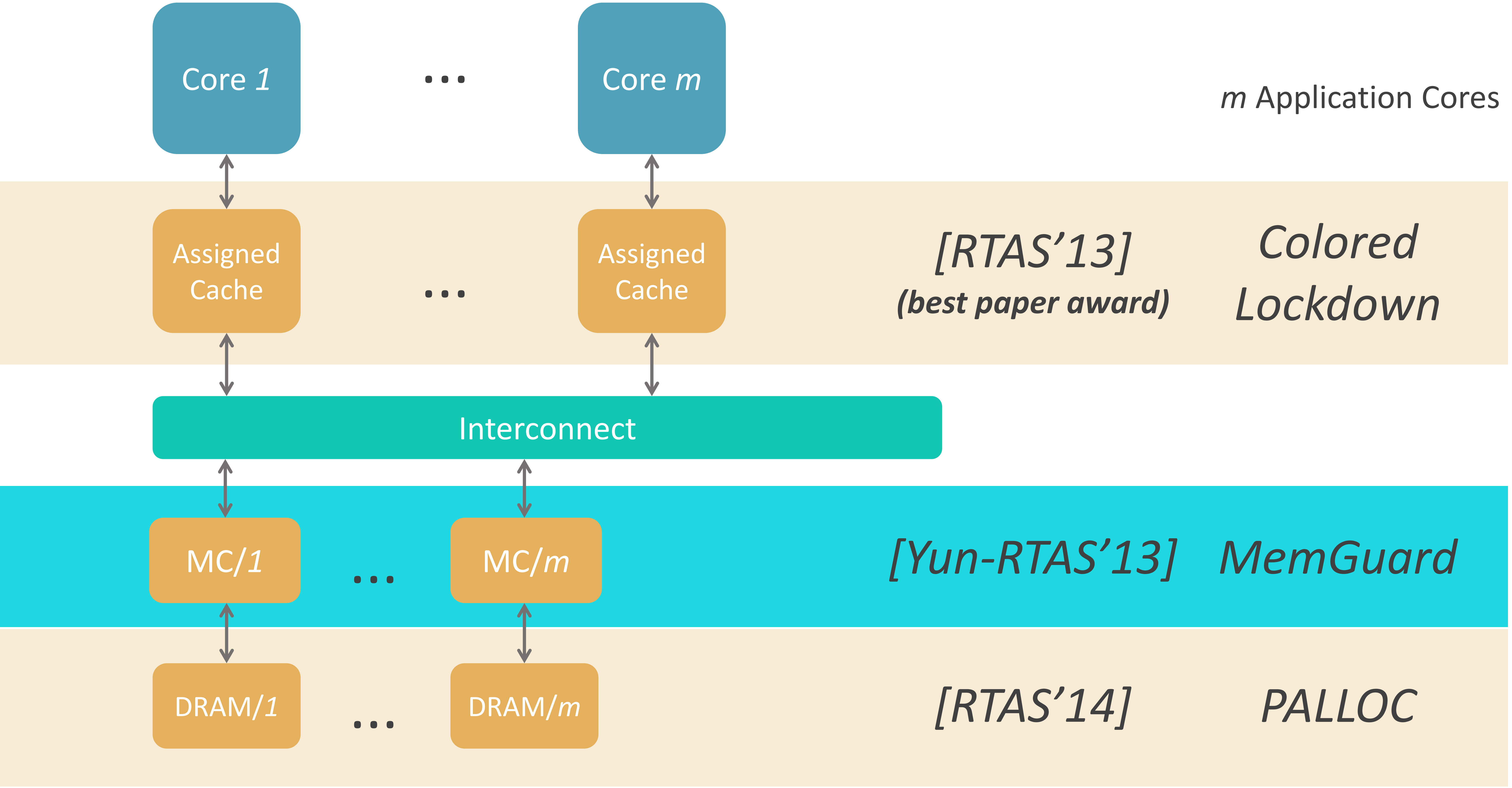
Shared memory controller

[RTAS'14]

PALLOC

CACHE-BASED PLATFORMS

Instance of software reference architecture for commercial cache-based multi-core systems.



Core *1* … Core *m*

m Application Cores

Assigned Cache … Assigned Cache

[RTAS'13]
(best paper award)

Colored Lockdown

Interconnect

MC/*1* … MC/*m*

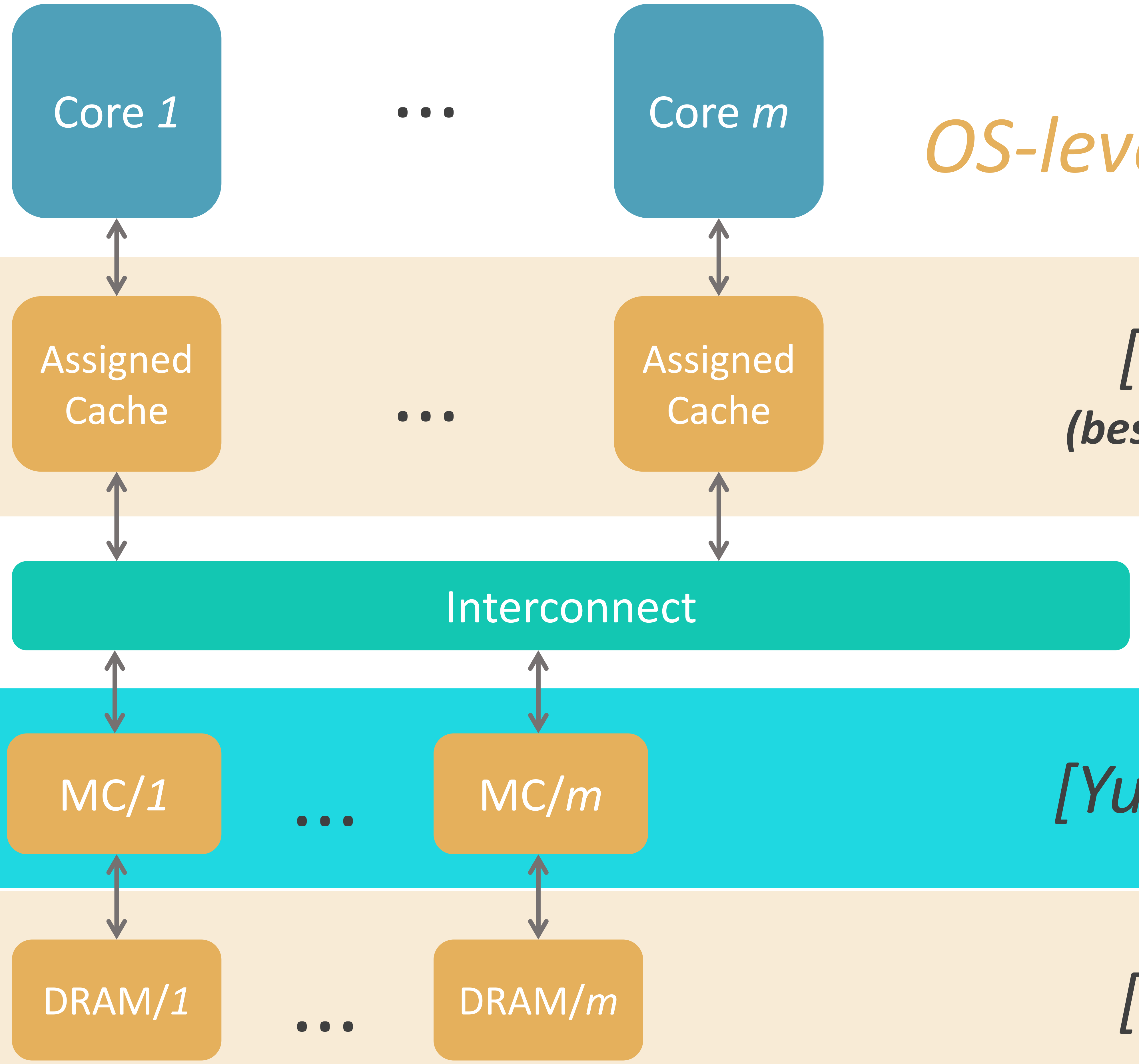
[Yun-RTAS'13] *MemGuard*

DRAM/*1* … DRAM/*m*

[RTAS'14] *PALLOC*

CACHE-BASED PLATFORMS

Instance of software reference architecture for commercial cache-based multi-core systems.



framework of
OS-level Software Solutions

[RTAS'13] ***(best paper award)*** *Colored Lockdown*

[Yun-RTAS'13] *MemGuard*

[RTAS'14] *PALLOC*

SCE

single-core equivalence

[IEEE Comp'16] [ECRTS'15]

MANAGEMENT

SCE Implementation

using COTS hardware *

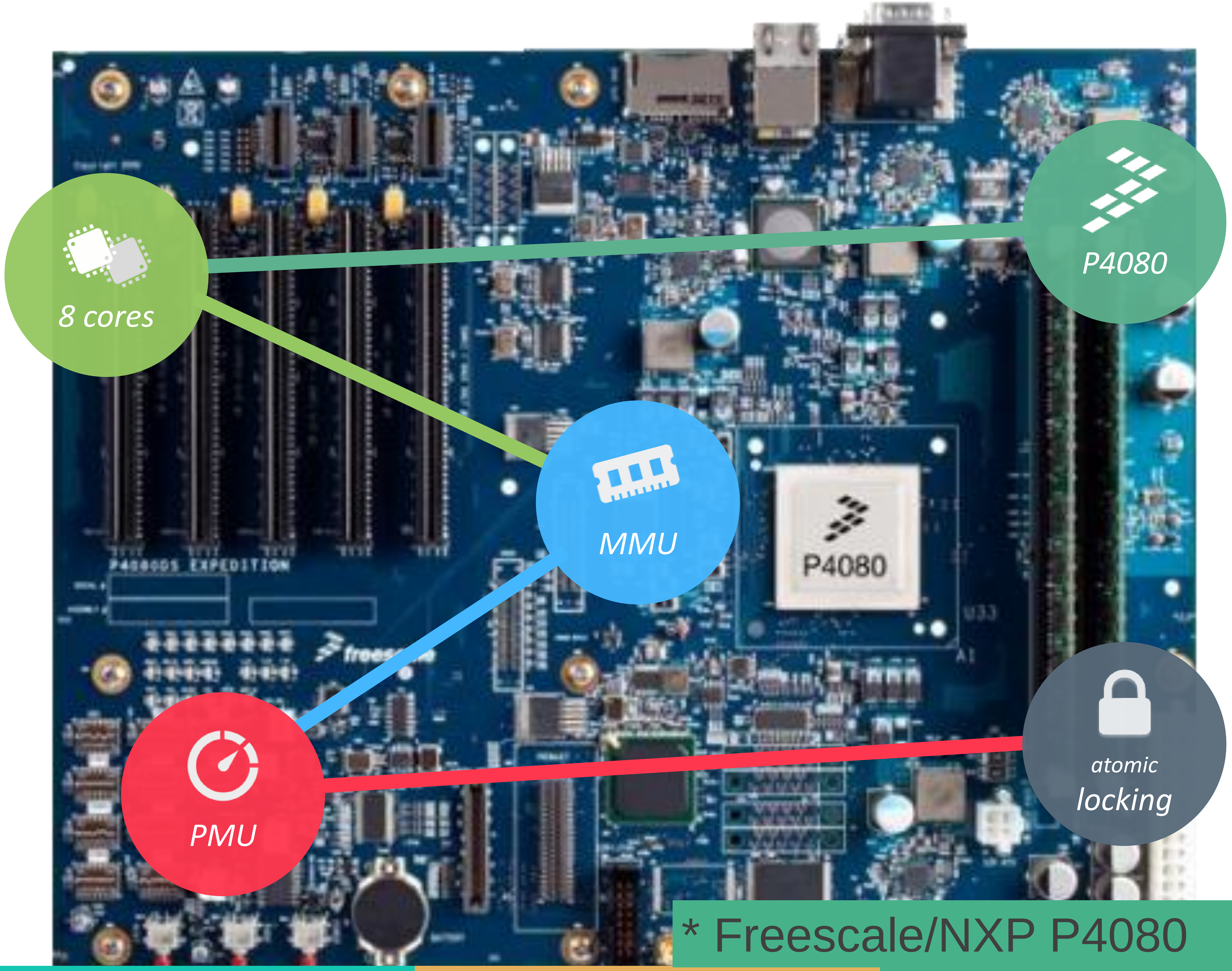
Colored Lockdown

MemGuard

PALLOC

SCE

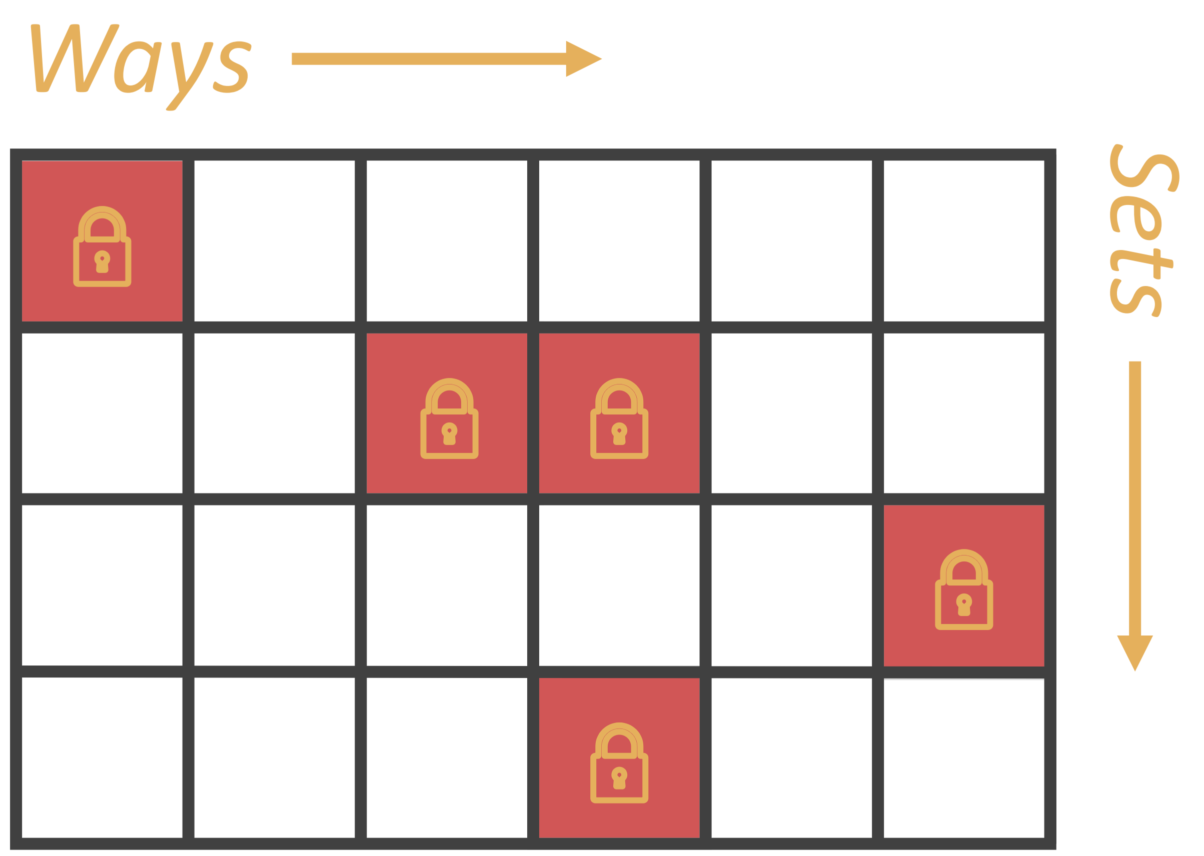
single-core equivalence



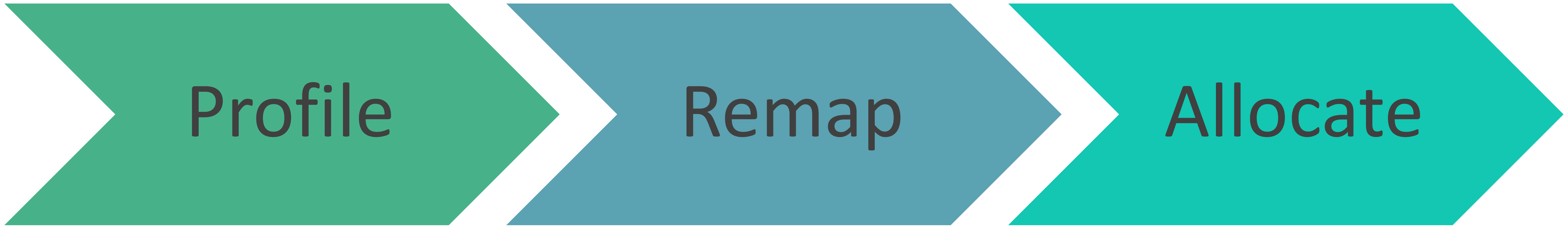
* Freescale/NXP P4080

COLORED LOCKDOWN *[RTAS'13]* *(best paper award)*

LAST-LEVEL CACHE MANAGEMENT MODEL



- ✓ Addresses all the sources of interference
- ✓ Converts the LLC cache in a deterministic object at the granularity of a single memory page
- ✓ Allows the use of legacy code
- ✓ Provides flexibility in cache assignment

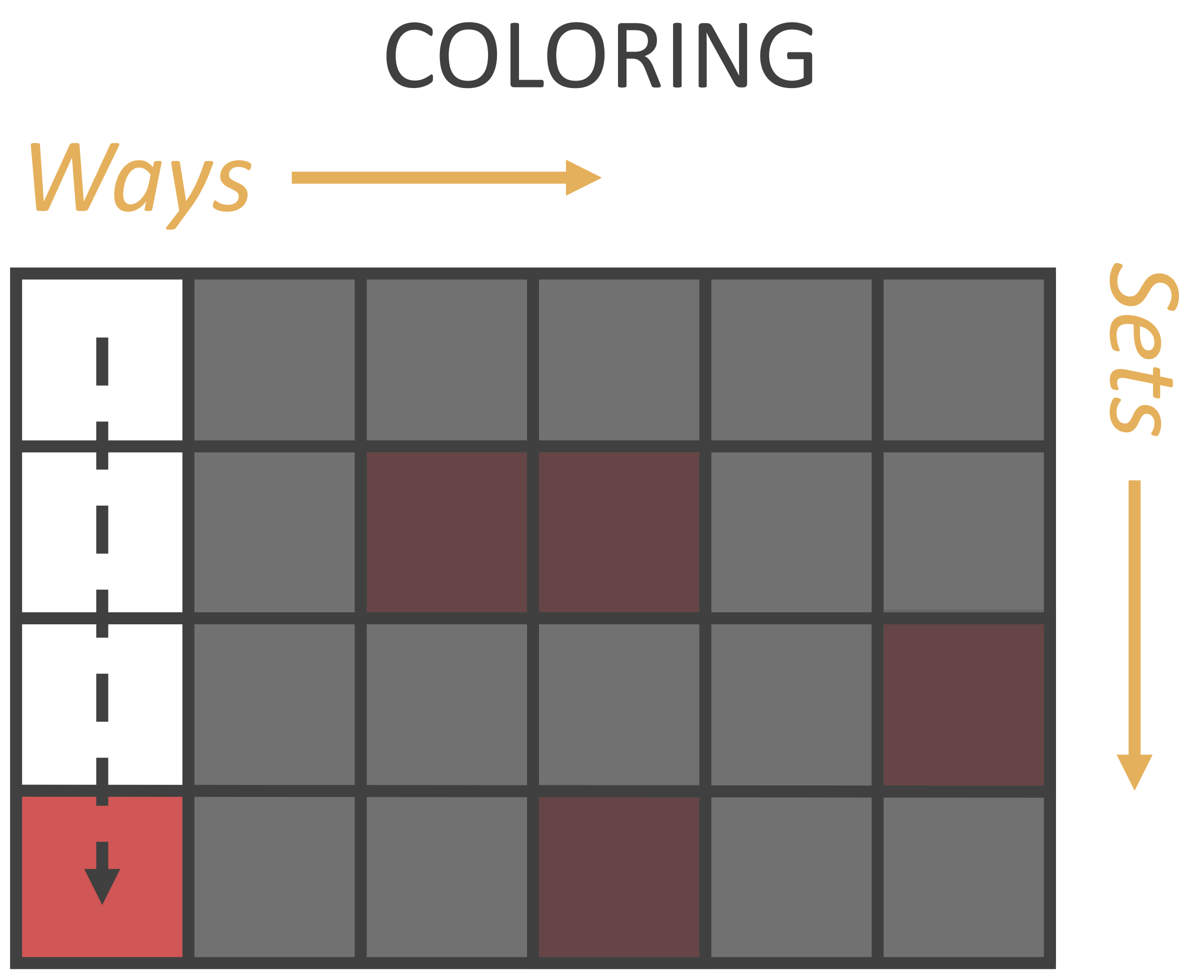

Profile → Remap → Allocate

100% hits on **allocated** pages

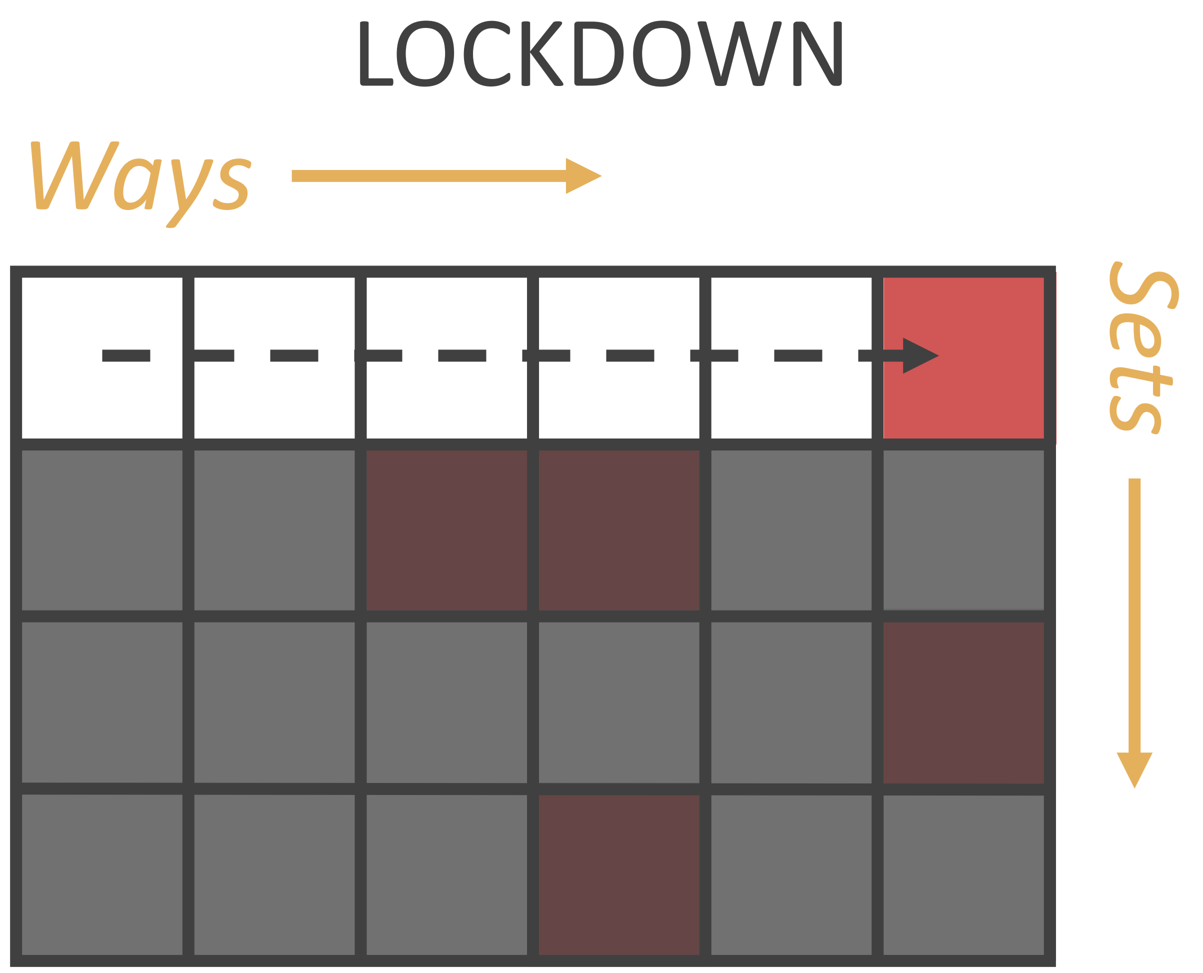
100% misses on **non-allocated** pages

[RTAS'13] **Renato Mancuso**, Roman Dudko, Emiliano Betti, Marco Cesati, Marco Caccamo, Rodolfo Pellizzoni, Real-Time Cache Management Framework for Multi-Core Architectures. *In Proceedings of the 19th IEEE International Conference on Real-Time and Embedded Technology and Applications Symposium* (RTAS 2013), Philadelphia, PA, USA, 2013

COLORED LOCKDOWN *[RTAS'13]*



- Used to **move page mapping** across sets (up/down)
- Leverages on the **virtual** → **physical** translation layer
- Transparent to the **application**



- Used to **allocate pages** on selected ways (left/right)
- Relies on **architecture-specific** lockdown features
- Once allocated, pages trigger **cache hits** until deallocation

PROFILING FOR CACHES *[RTAS'13]*

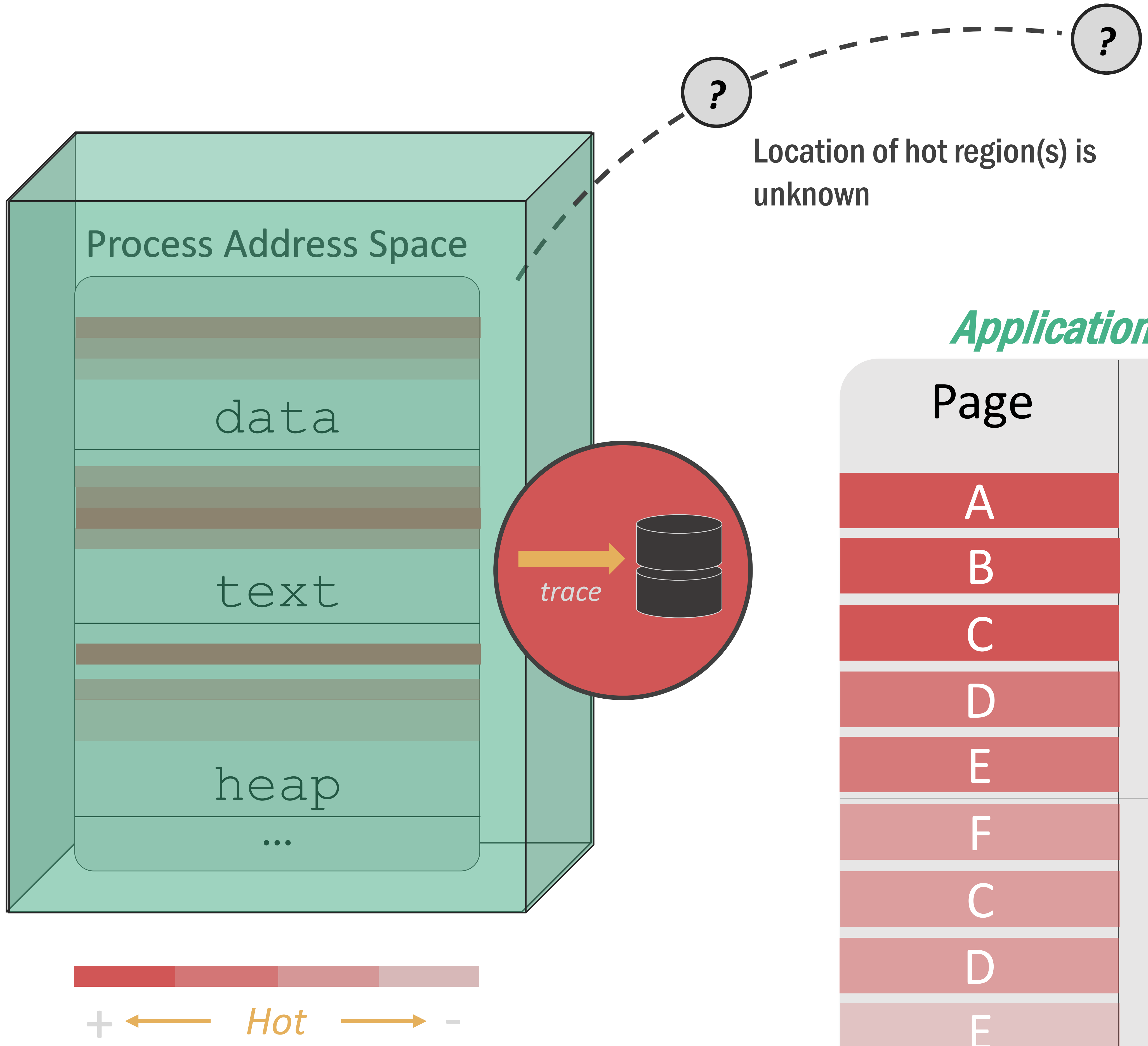
PROFILE-DRIVEN CACHE ALLOCATION

PROBLEM

Caches are critical, constrained resources. Optimal allocation ?

PROFILE MEMORY

Extract memory traces and produce memory usage profile.



Location of hot region(s) is unknown

Absolute virtual memory addresses may change

Application Profile

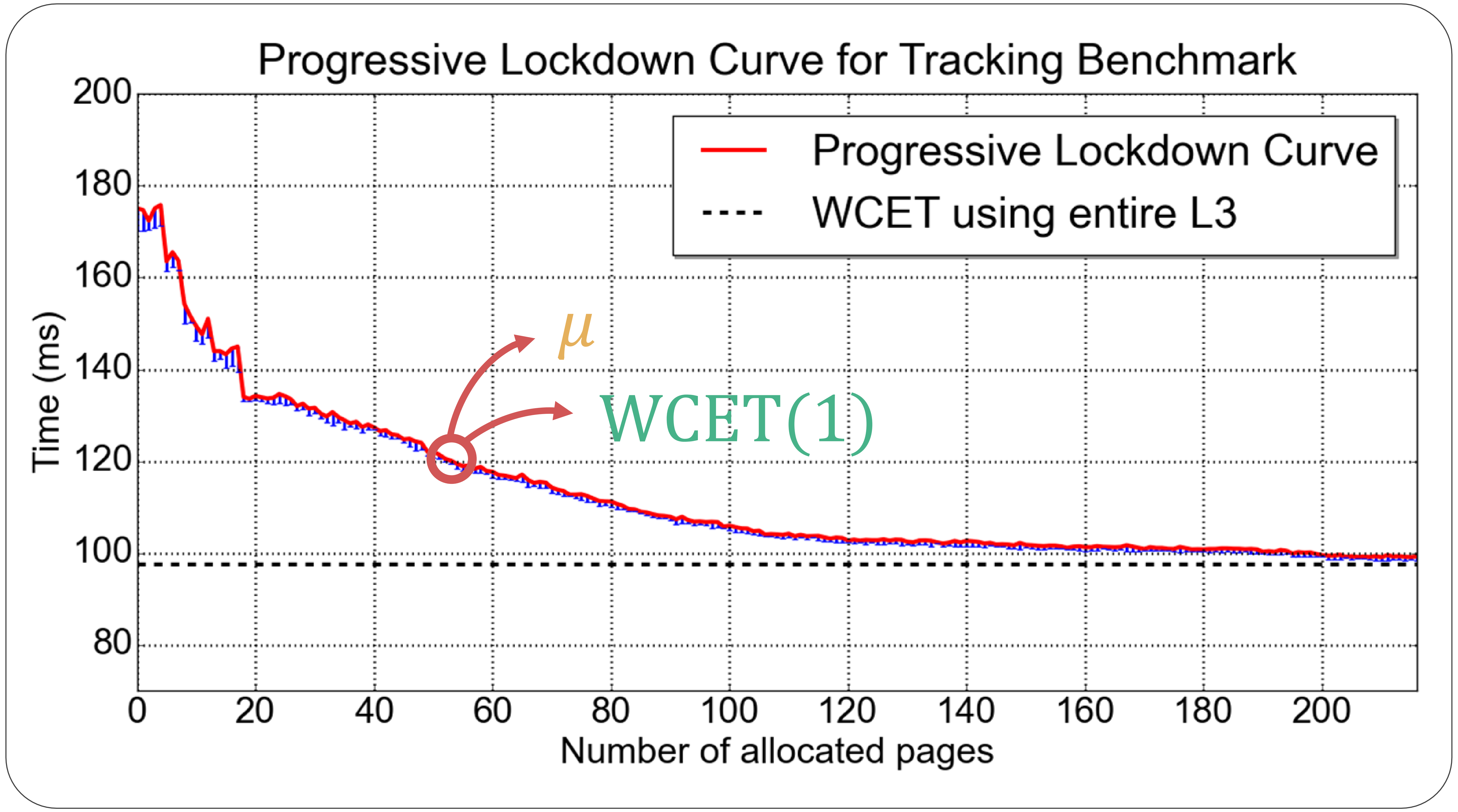
Page	Accesses
A	100 K
B	10 K
C	1 K
D	700
E	500
F	90
C	50
D	10
E	5

threshold

μ

residual cache misses

PROGRESSIVE LOCKDOWN *[ECRTS'15]*



Application Profile

Page	Accesses
A	100 K
B	10 K
C	1 K
D	700
E	500
F	90
C	50
D	10
E	5

threshold

μ

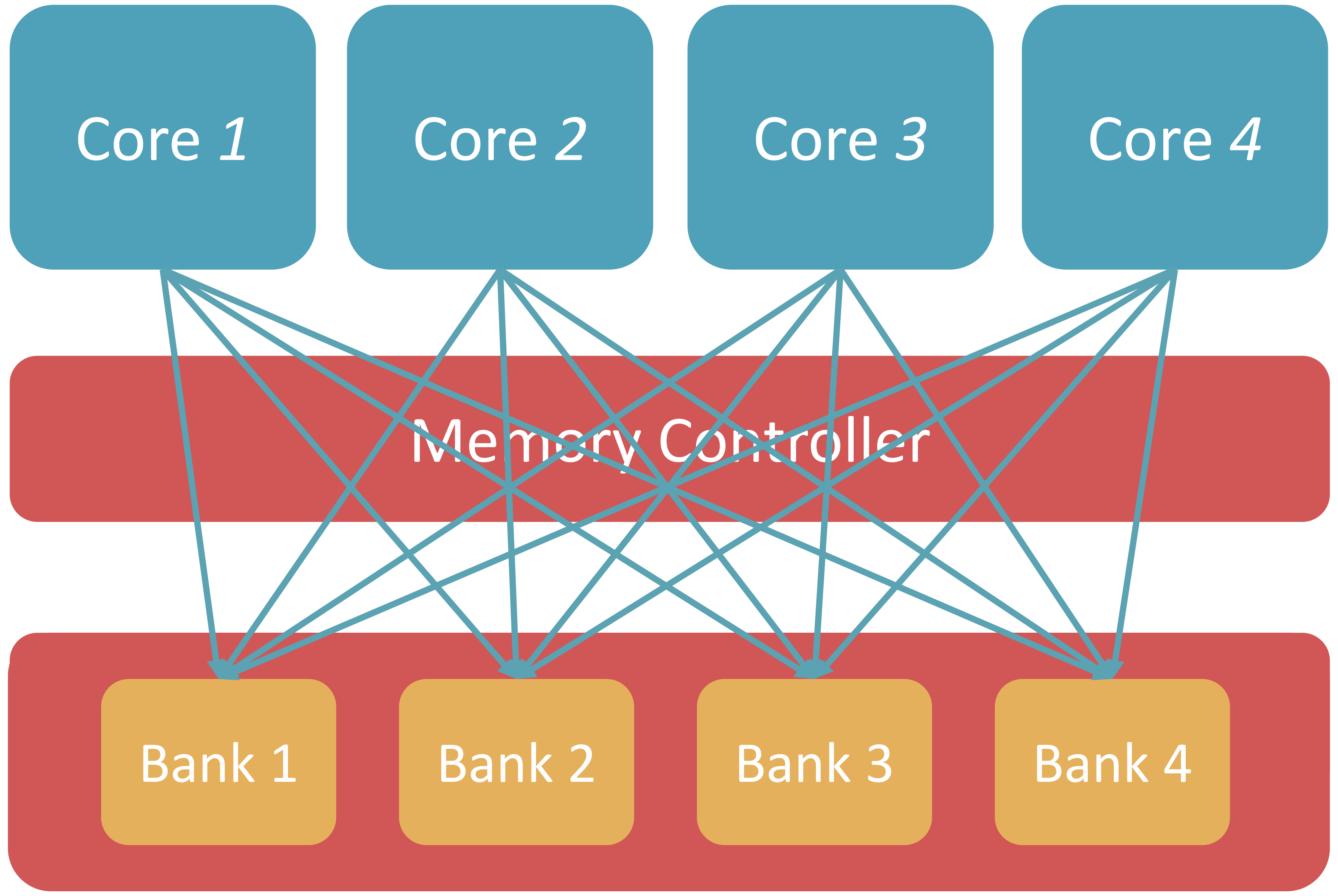
residual cache misses

WCET(1) := task **w**orst-**c**ase **e**xcution **t**ime with **1** active core

PALLOC *[RTAS'14]*

DRAM PRIVATE BANK ENFORCEMENT

- Each DIMM contains multiple (8~16) **banks**
- Different banks can be accessed in parallel



AVERAGE

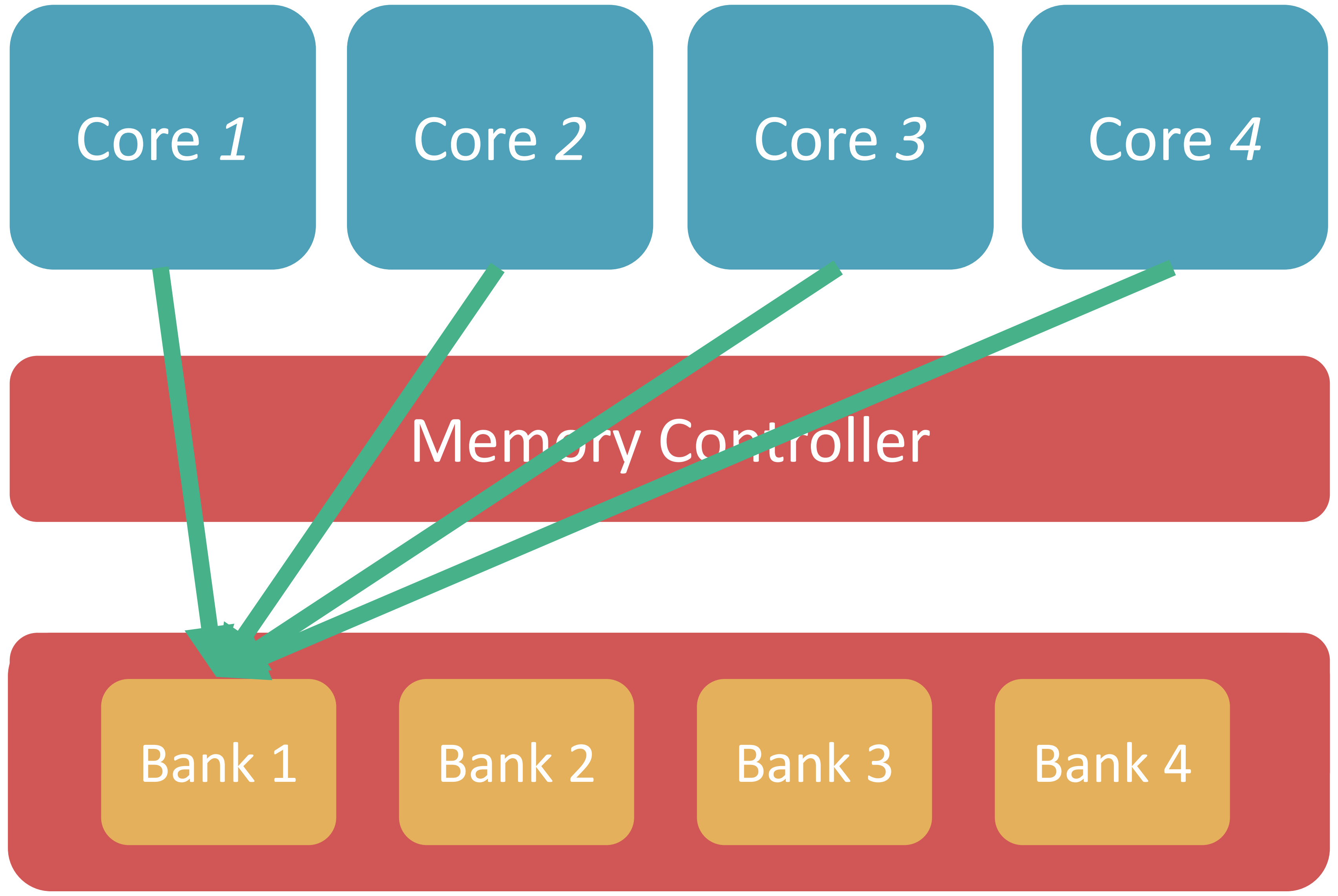
Tasks in each core access all the available DRAM banks.

MANAGEMENT

[RTAS'14] Heechul Yun, **Renato Mancuso**, Zheng-Pei Wu, Rodolfo Pellizzoni. PALLOC: DRAM Bank-Aware Memory Allocator for Performance Isolation on Multicore Platforms. *IEEE Intl. Conference on Real-Time and Embedded Technology and Applications Symposium* (RTAS 2014), Berlin, Germany, 2014

DRAM PRIVATE BANK ENFORCEMENT

- Each DIMM contains multiple (8~16) **banks**
- Different banks can be accessed in parallel



AVERAGE

Tasks in each core access all the available DRAM banks.

WORST-CASE

Tasks in all the cores access a single DRAM bank.

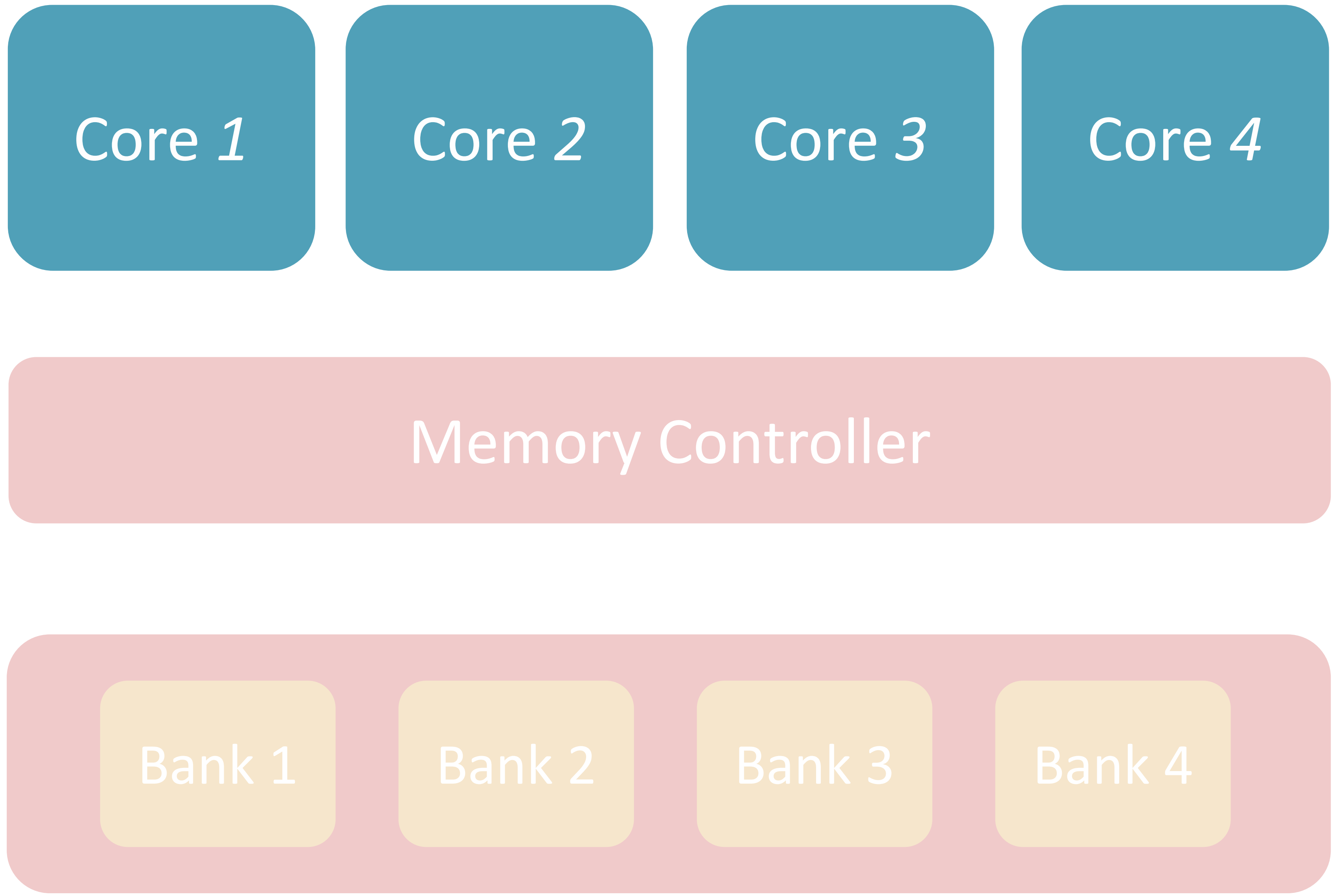
MANAGEMENT

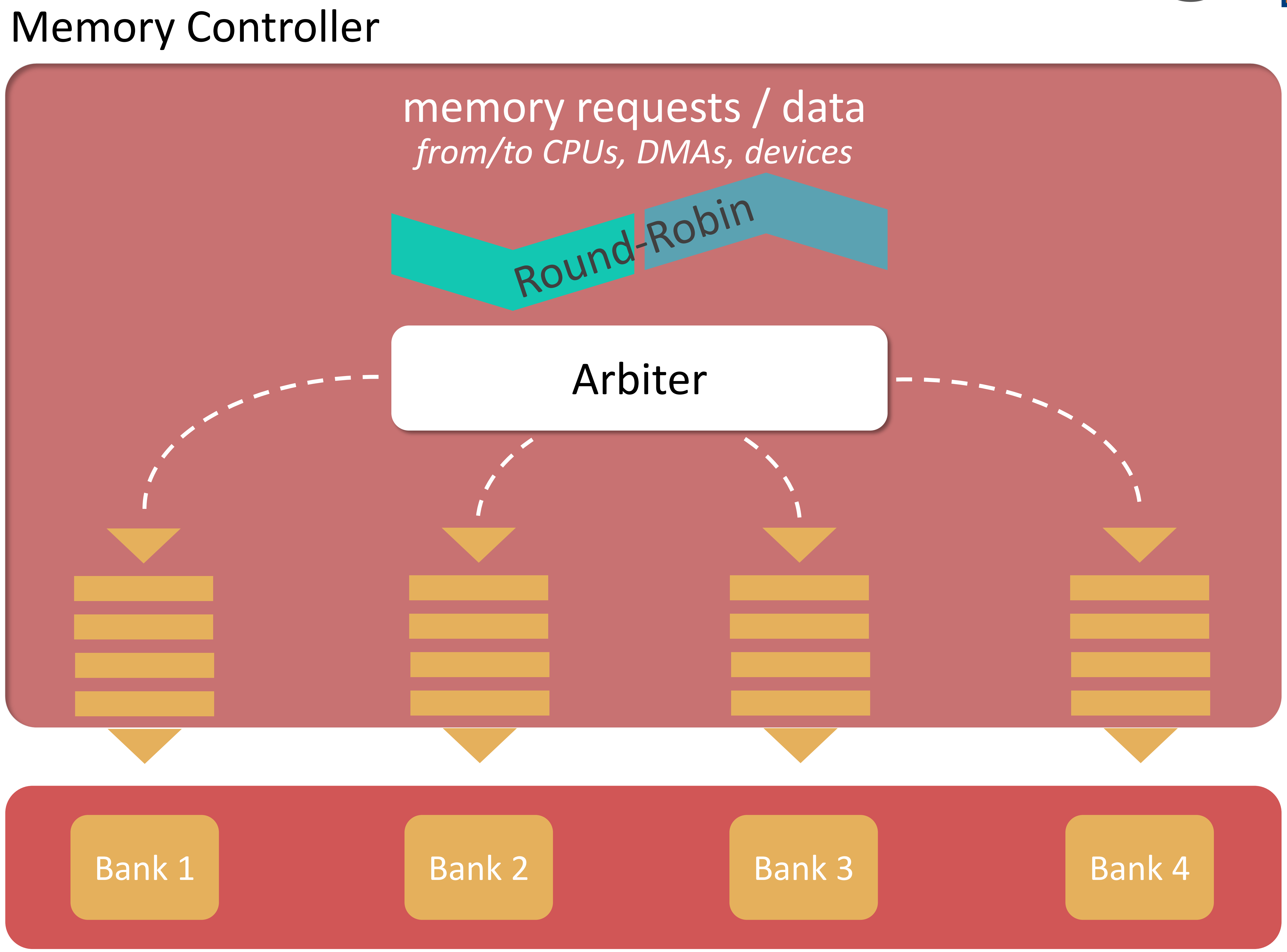
[RTAS'14] Heechul Yun, **Renato Mancuso**, Zheng-Pei Wu, Rodolfo Pellizzoni. PALLOC: DRAM Bank-Aware Memory Allocator for Performance Isolation on Multicore Platforms. *IEEE Intl. Conference on Real-Time and Embedded Technology and Applications Symposium* (RTAS 2014), Berlin, Germany, 2014

PALLOC *[RTAS'14]*

DRAM PRIVATE BANK ENFORCEMENT [2]

- Each DIMM contains multiple (8~16) **banks**
- Different banks can be accessed in parallel

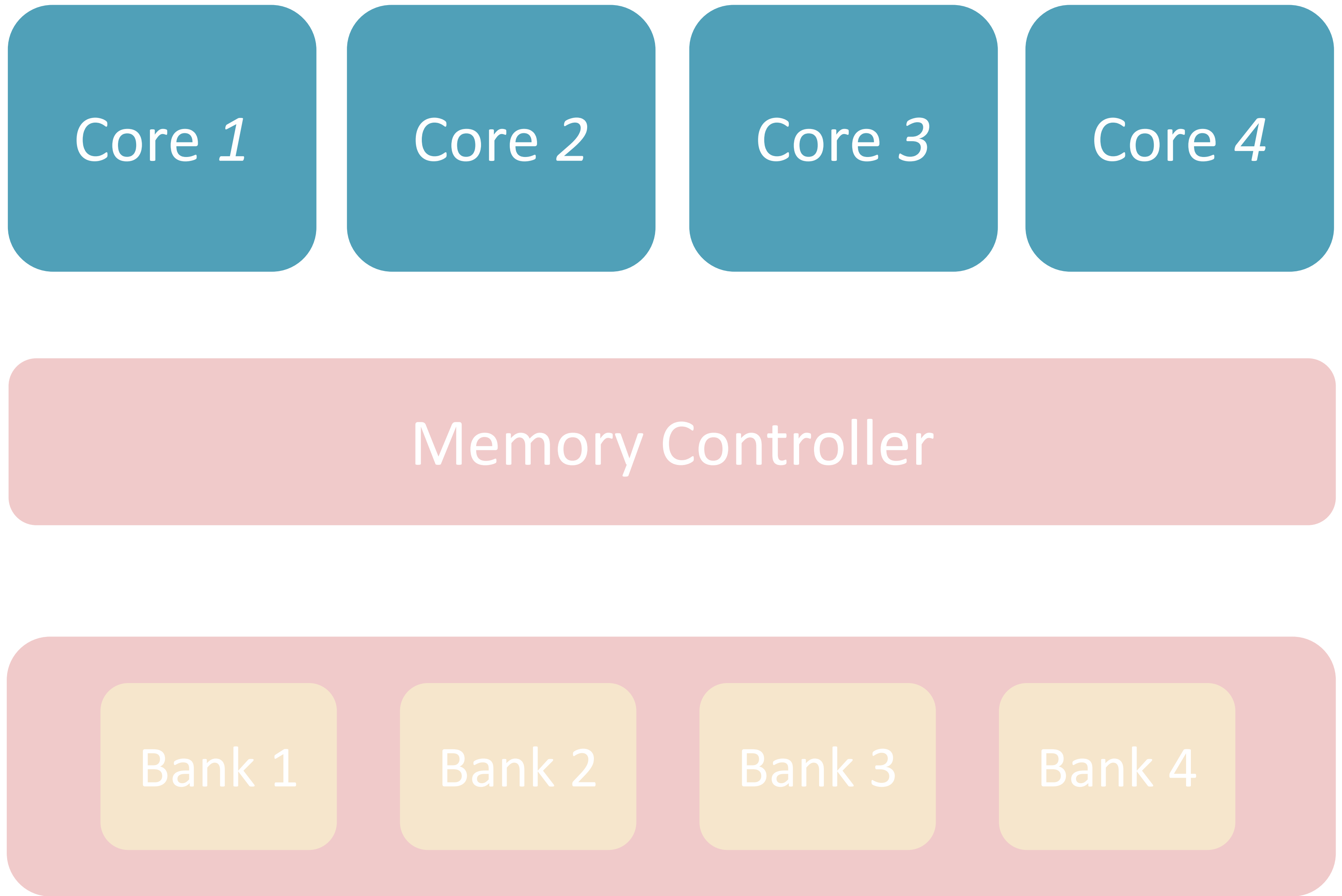


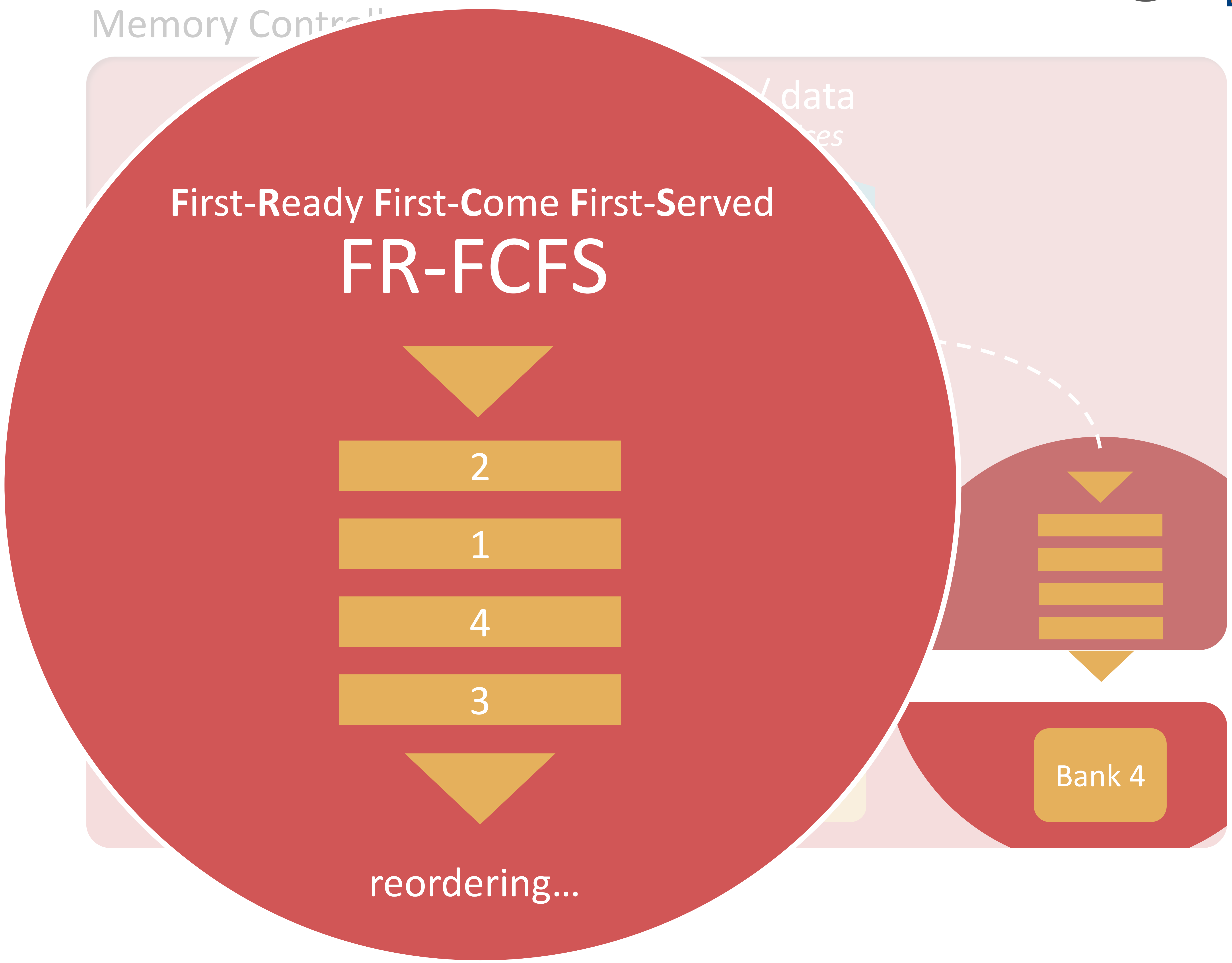


PALLOC *[RTAS'14]*

DRAM PRIVATE BANK ENFORCEMENT [2]

- Each DIMM contains multiple (8~16) **banks**
- Different banks can be accessed in parallel





First-**R**eady **F**irst-**C**ome **F**irst-**S**erved

FR-FCFS

reordering…

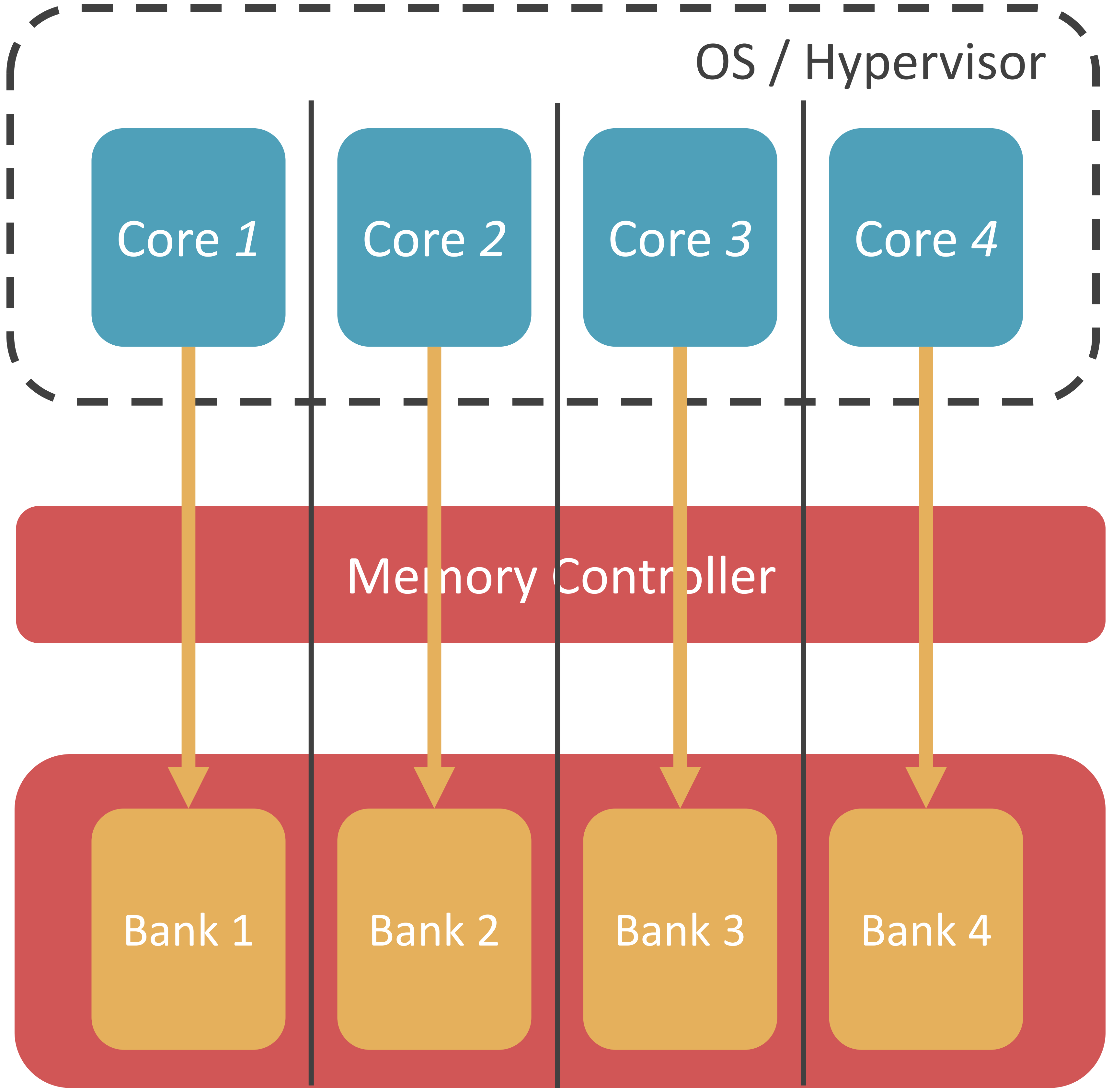
PALLOC *[RTAS'14]*

PROBLEM

In general, the OS / Hypervisor ignores page-to-bank mapping.

PALLOC

Modified allocator to export mapping between physical memory and DRAM banks.



PALLOC *[RTAS'14]*

PROBLEM

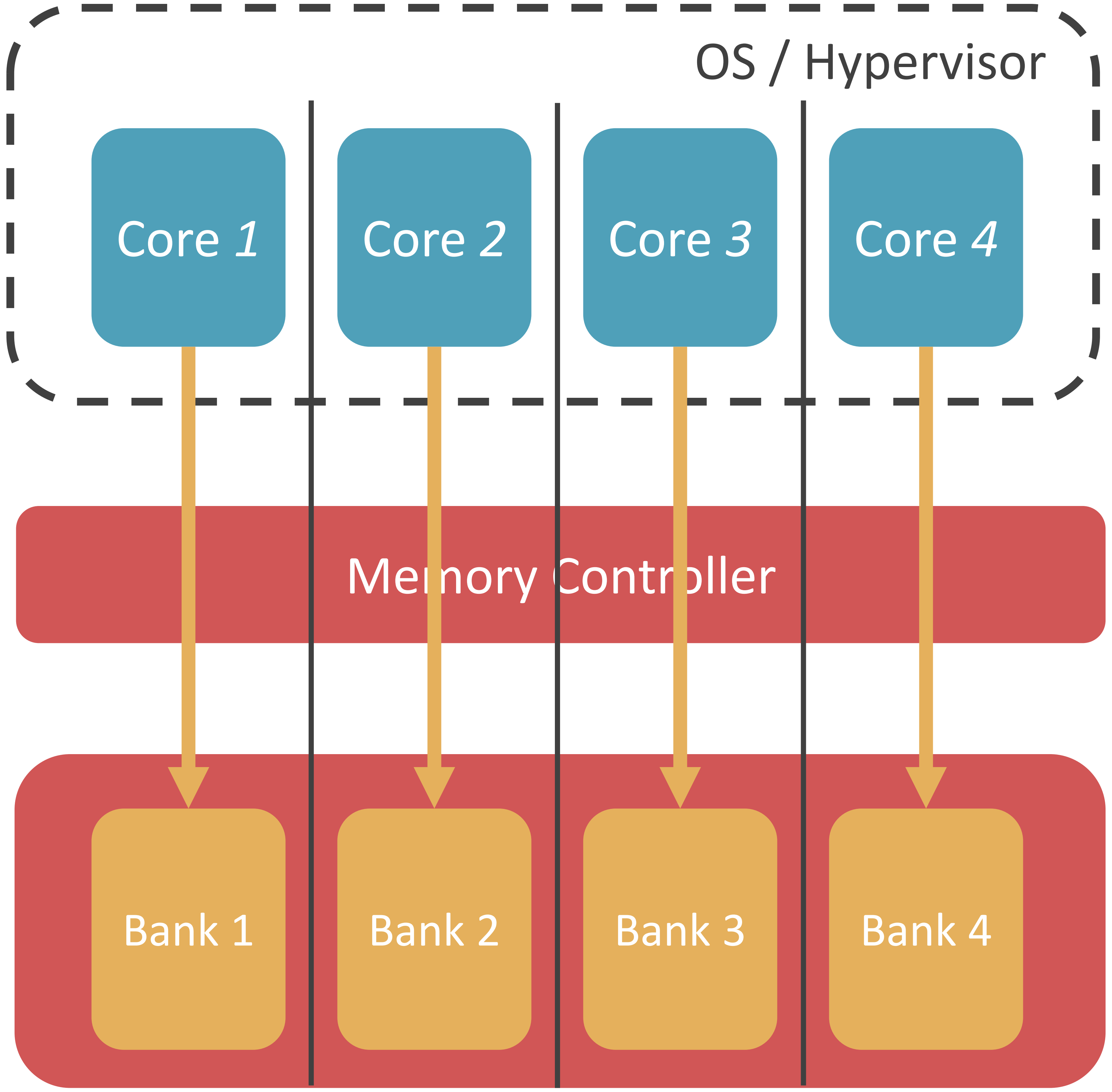
In general, the OS / Hypervisor ignores page-to-bank mapping.

PALLOC

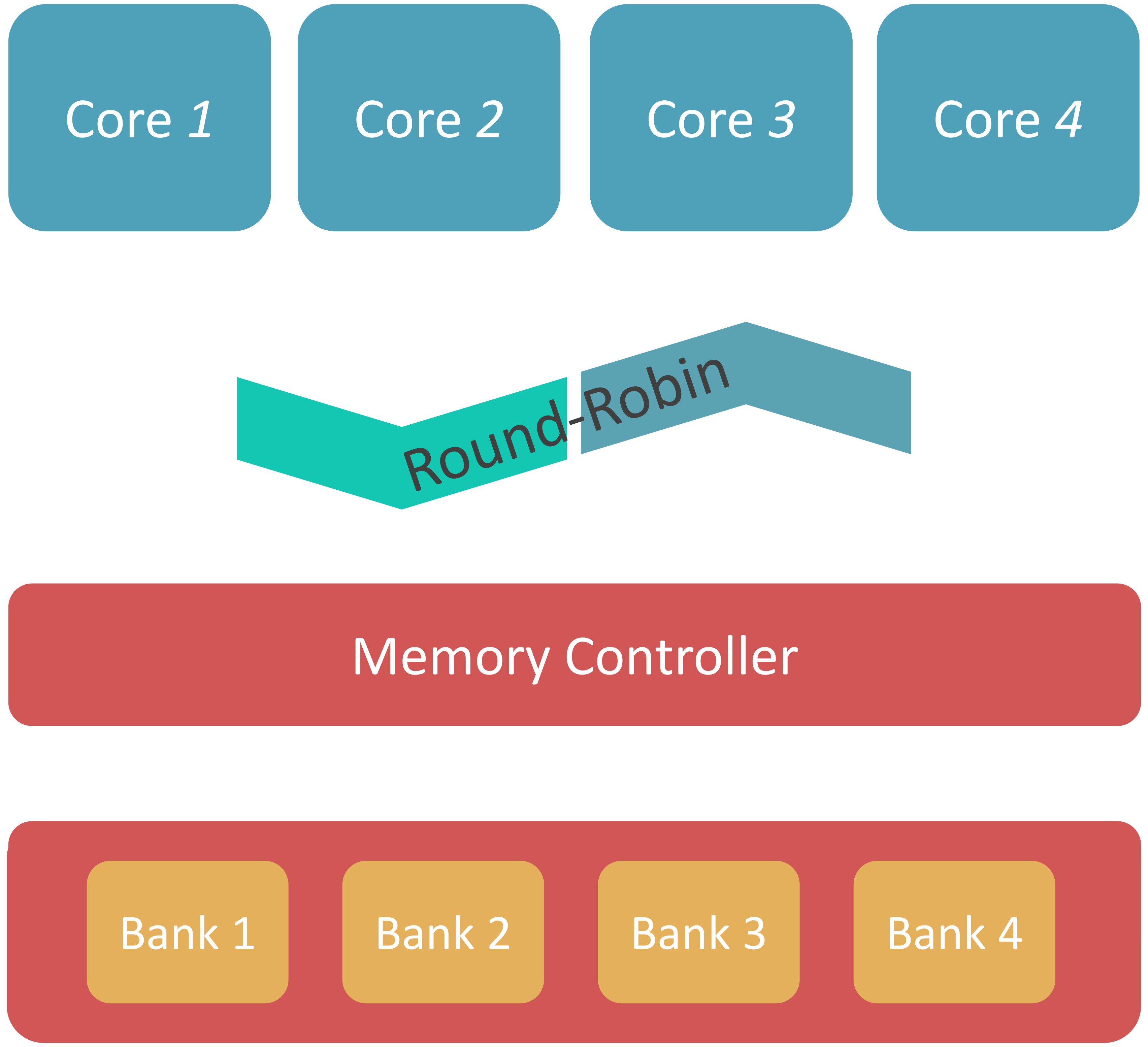
Modified allocator to export mapping between physical memory and DRAM banks.

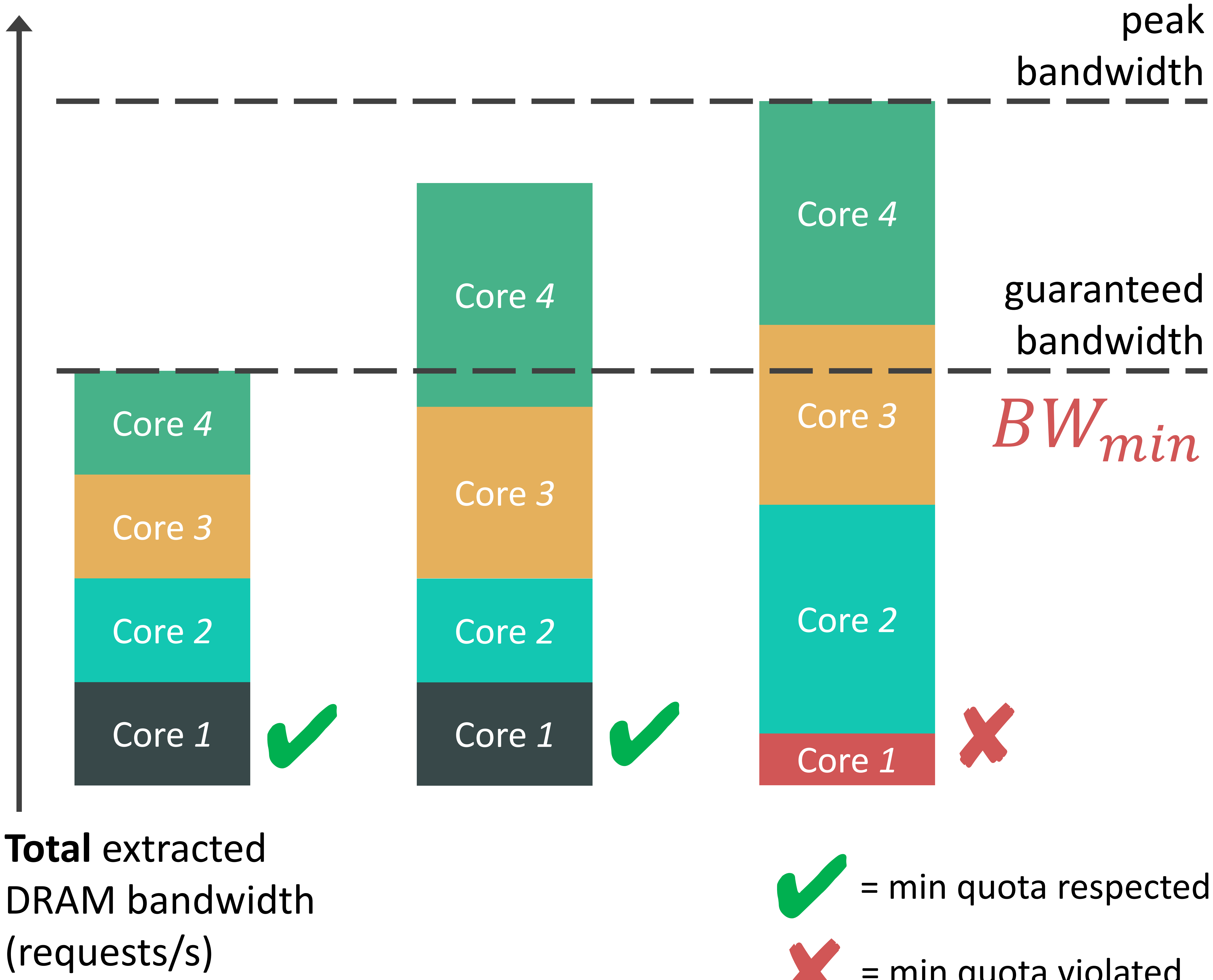
✓ Prevents request re-ordering at bank queue
* requests from same core may still be re-ordered

✓ Prevents inter-core induced row misses



ARBITER CONGESTION

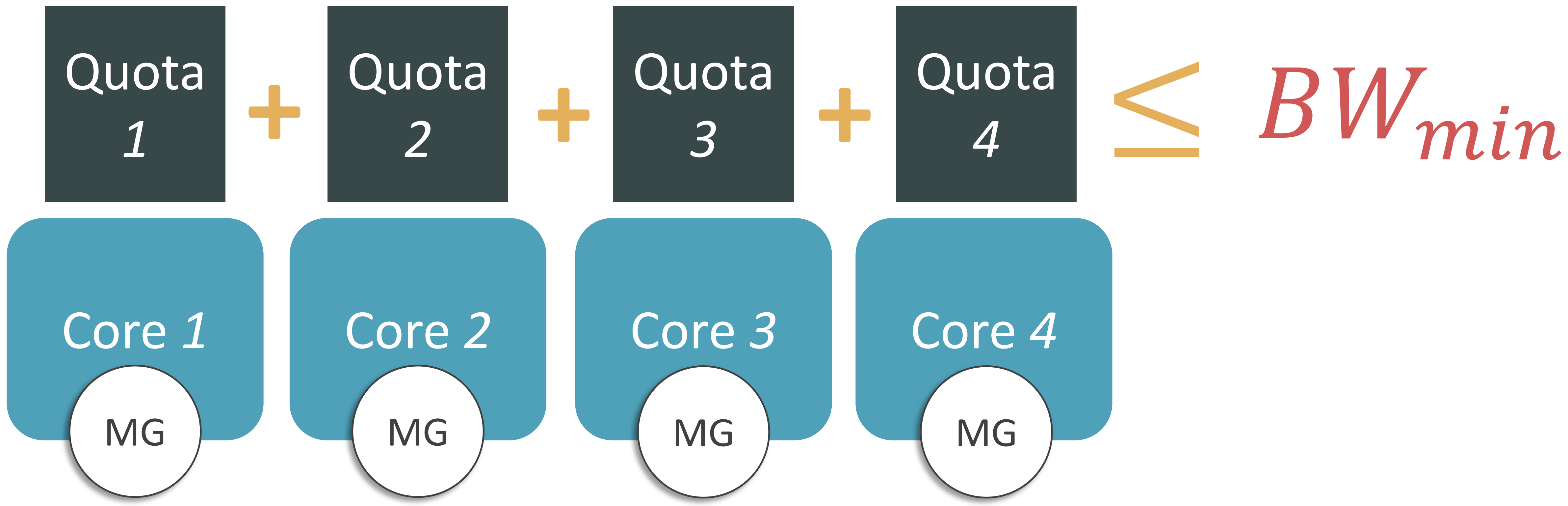




✓ = min quota respected

✗ = min quota violated

SCE WCET(m) *[ECRTS'15]*



- ✔ Use **Colored Lockdown** to derive **WCET(1)** and μ
- ✔ Use **PALLOC** to avoid **FR-FRCS re-ordering**
- ✔ Use **MemGuard** to prevent **arbiter saturation**

With **even** budget assignment, upper-bound WCET **regardless** of the activity of other cores:

WCET(m)

$$\left[\mathrm{WCET}(1) + \underbrace{\mu \cdot L_{size}}_{\text{Total DRAM traffic}} \left(\underbrace{\frac{m}{BW_{min}}}_{m\text{-th fraction of guaranteed bandwidth}} - \frac{1}{BW_{max}}\right)\right]$$

[ECRTS'15] **Renato Mancuso**, Rodolfo Pellizzoni, Marco Caccamo, Lui Sha, Heechul Yun, WCET(m) Estimation in Multi-Core Systems using Single Core Equivalence. *In Proceedings of the 27th Euromicro Conference on Real-Time Systems* (ECRTS 2015), Lund, Sweden, 2015

SINGLE-CORE EQUIVALENCE

SCE

single-core equivalence

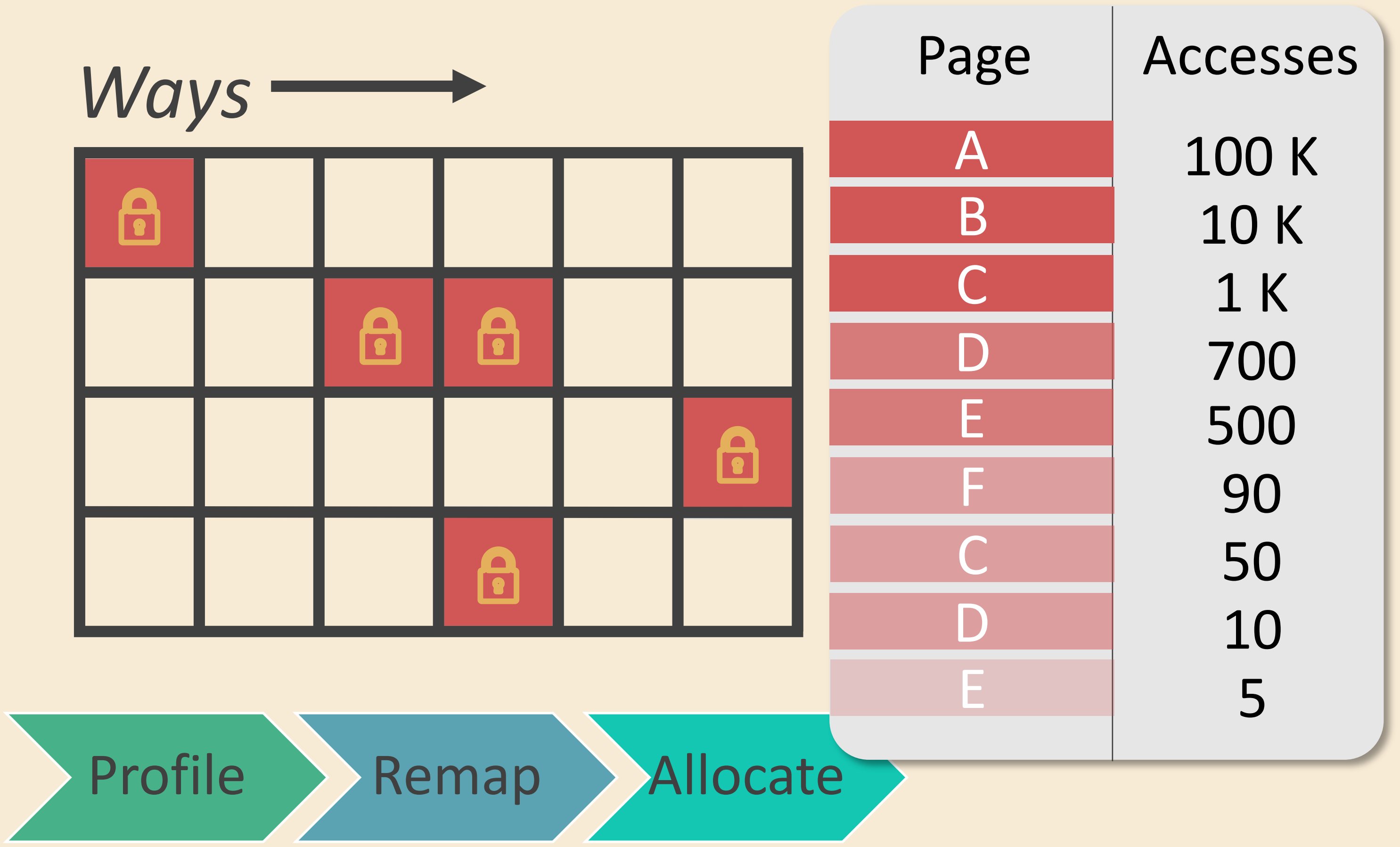
[IEEE Comp'16]

Colored Lockdown

last level cache management

[RTAS'13]

(best paper award)

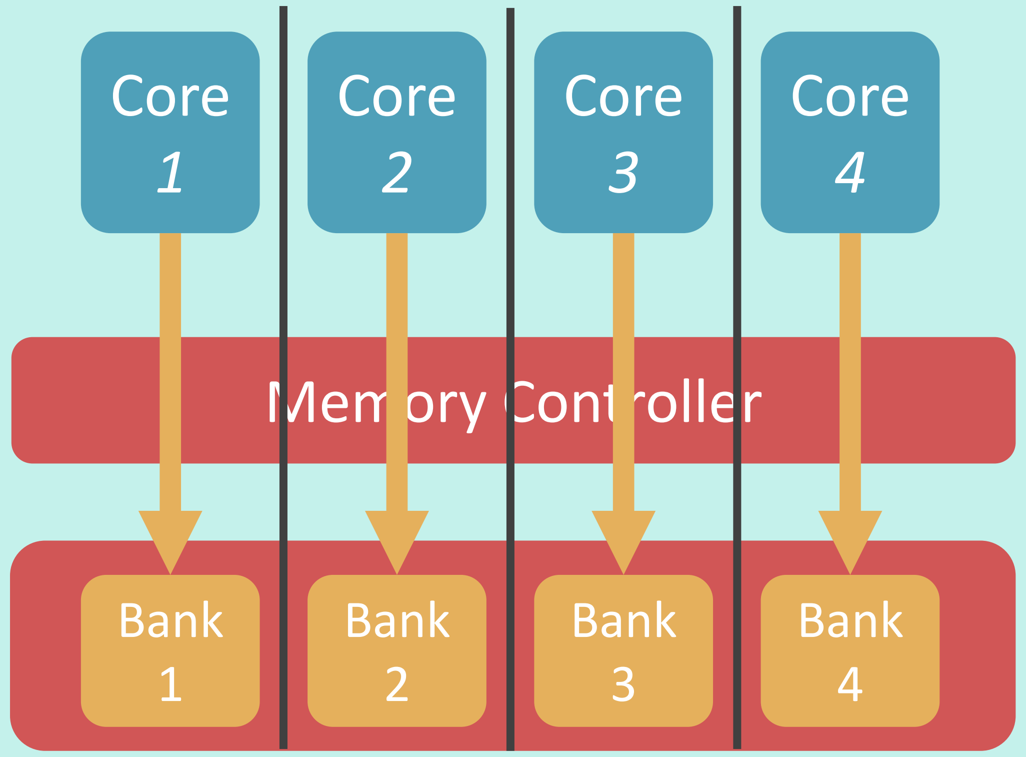


Page	Accesses
A	100 K
B	10 K
C	1 K
D	700
E	500
F	90
C	50
D	10
E	5

PALLOC

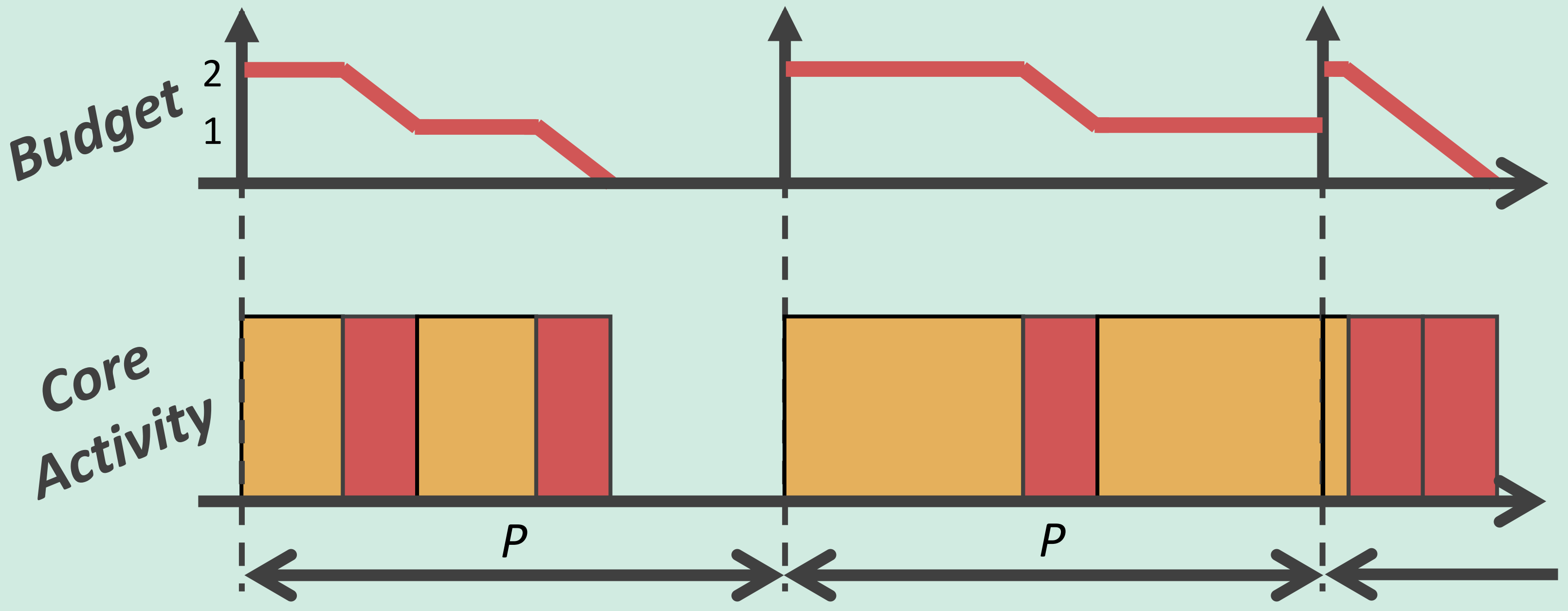
private per-core DRAM banks

[RTAS'14]



MemGuard

DRAM BW management



SINGLE-CORE EQUIVALENCE

SCE

single-core equivalence

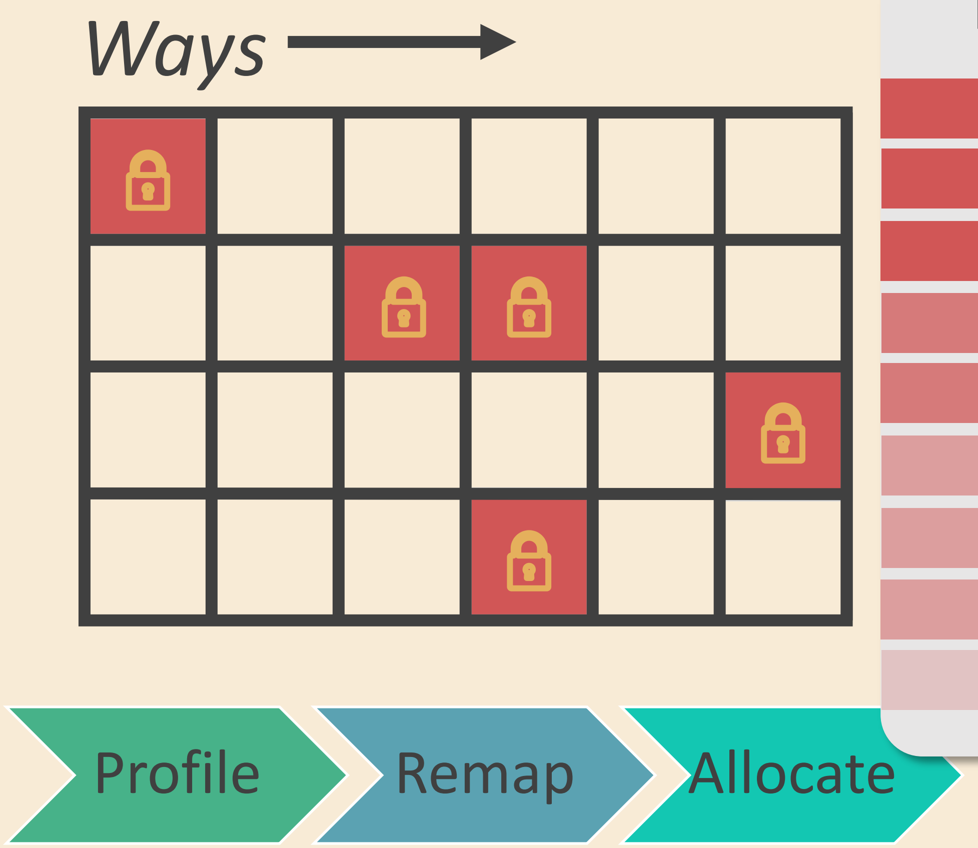
[IEEE Comp'16]

C

last l

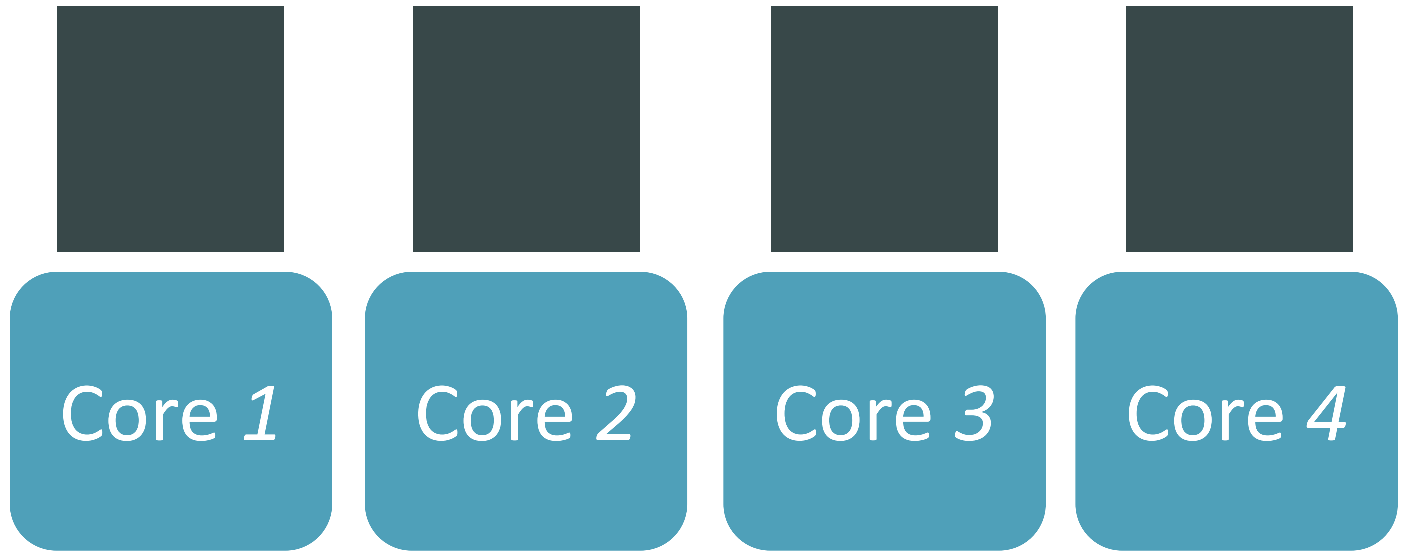
]

rd)



Page	Accesses
A	100 K
B	10 K
C	1 K
D	700
E	500
F	90
C	50
D	10
E	5

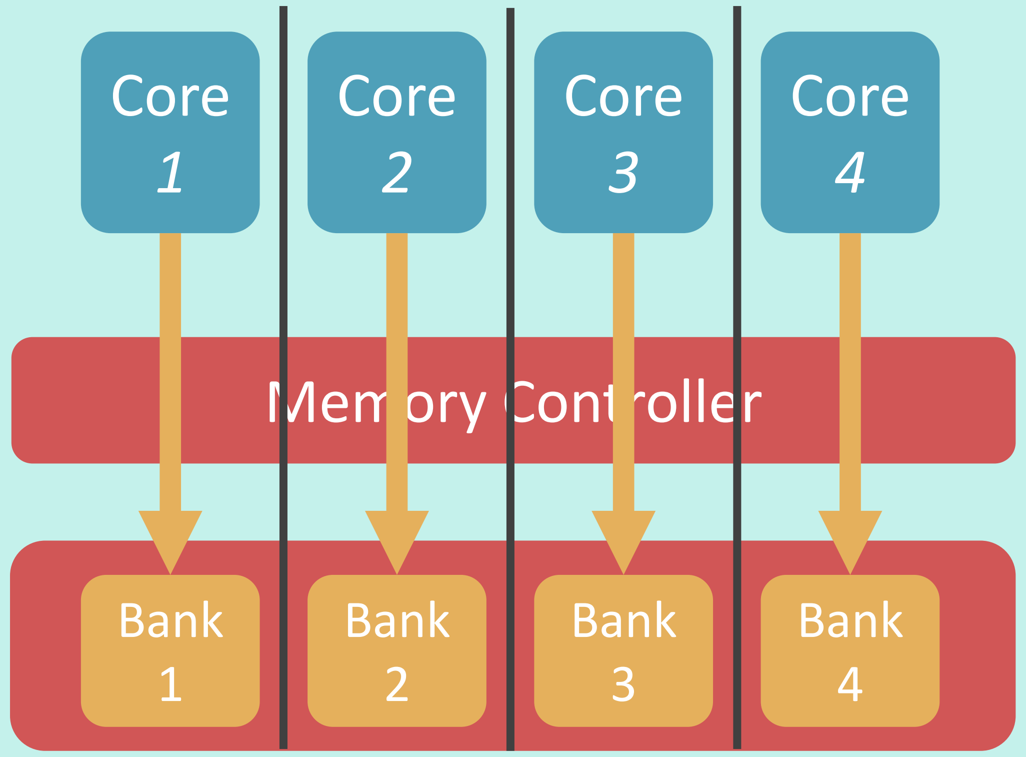
[ECRTS'15]



WCET(m)

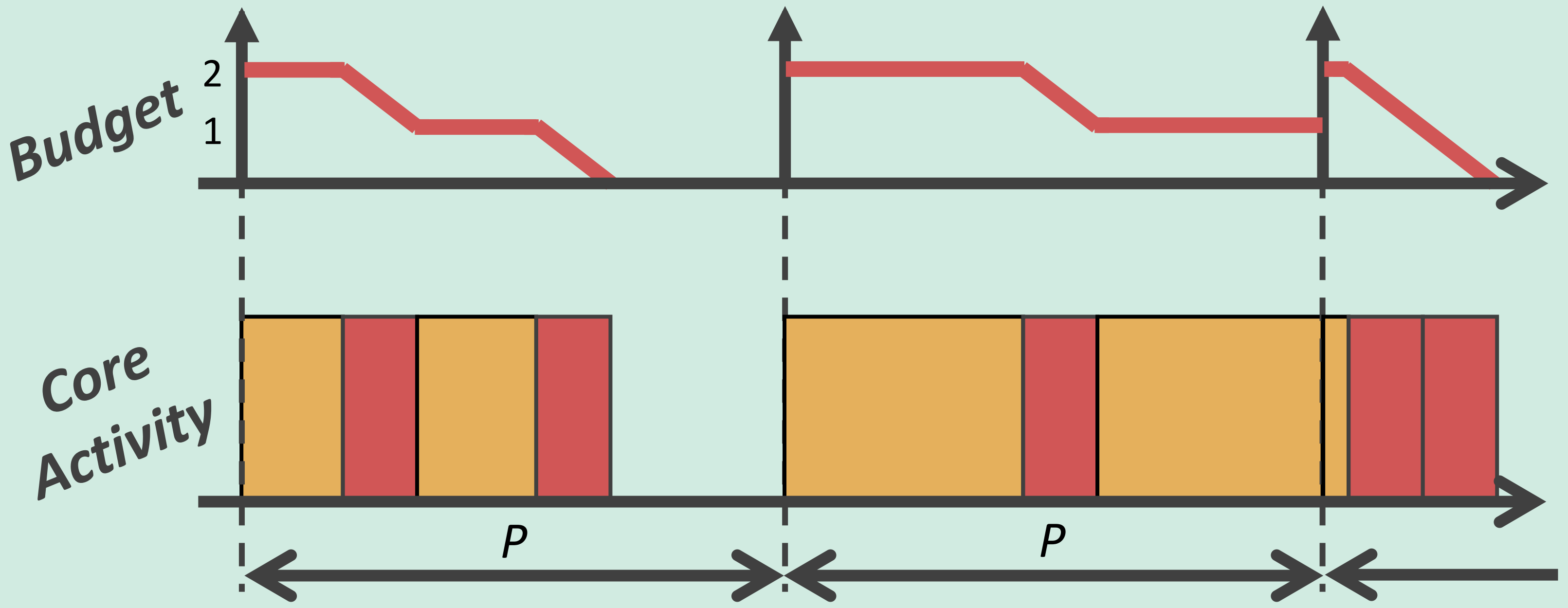
4]

private per-core DRAM banks



MemGuard

DRAM BW management

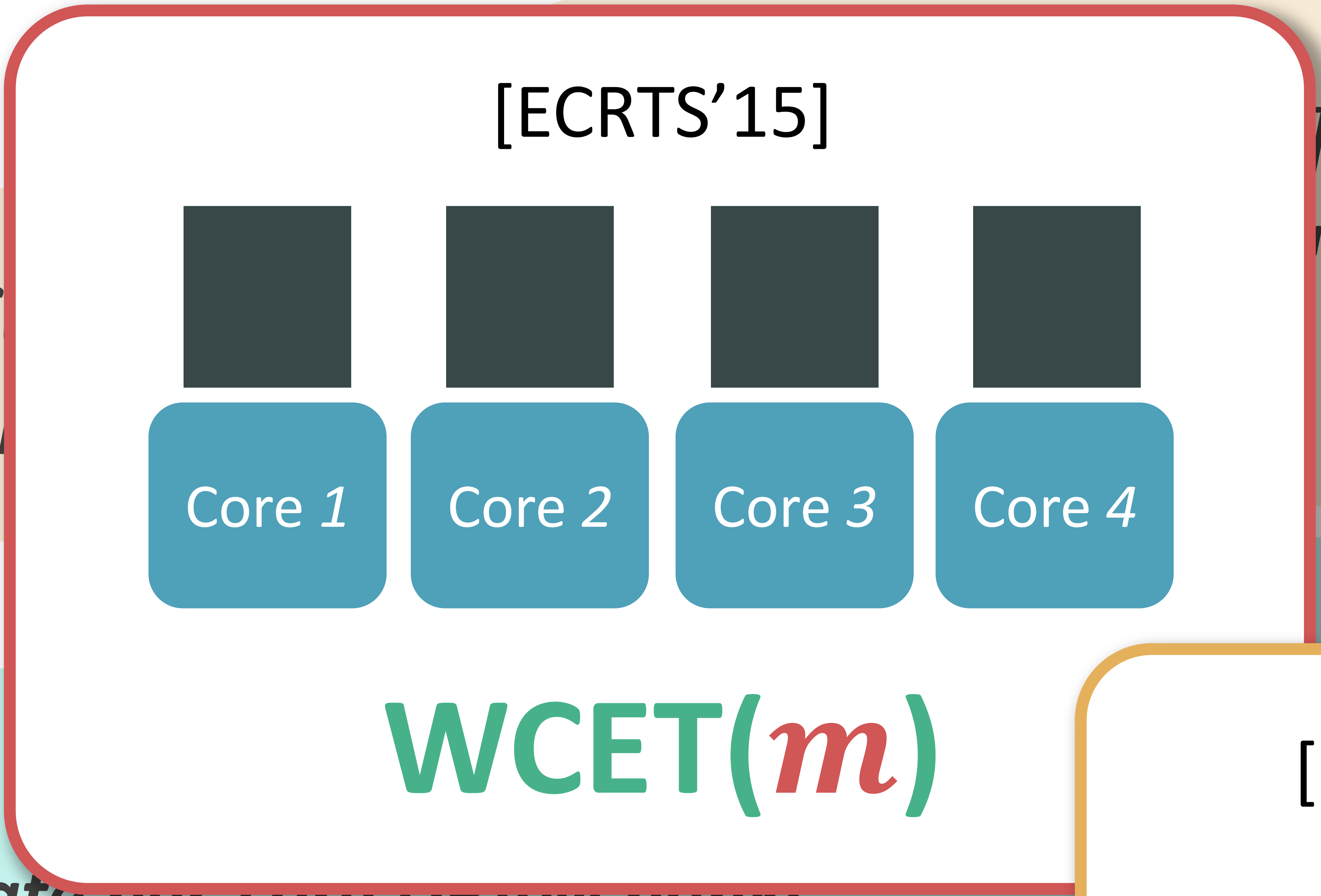


SINGLE-CORE EQUIVALENCE

SCE

single-core equivalence

[IEEE Comp'16]



[ECRTS'15]

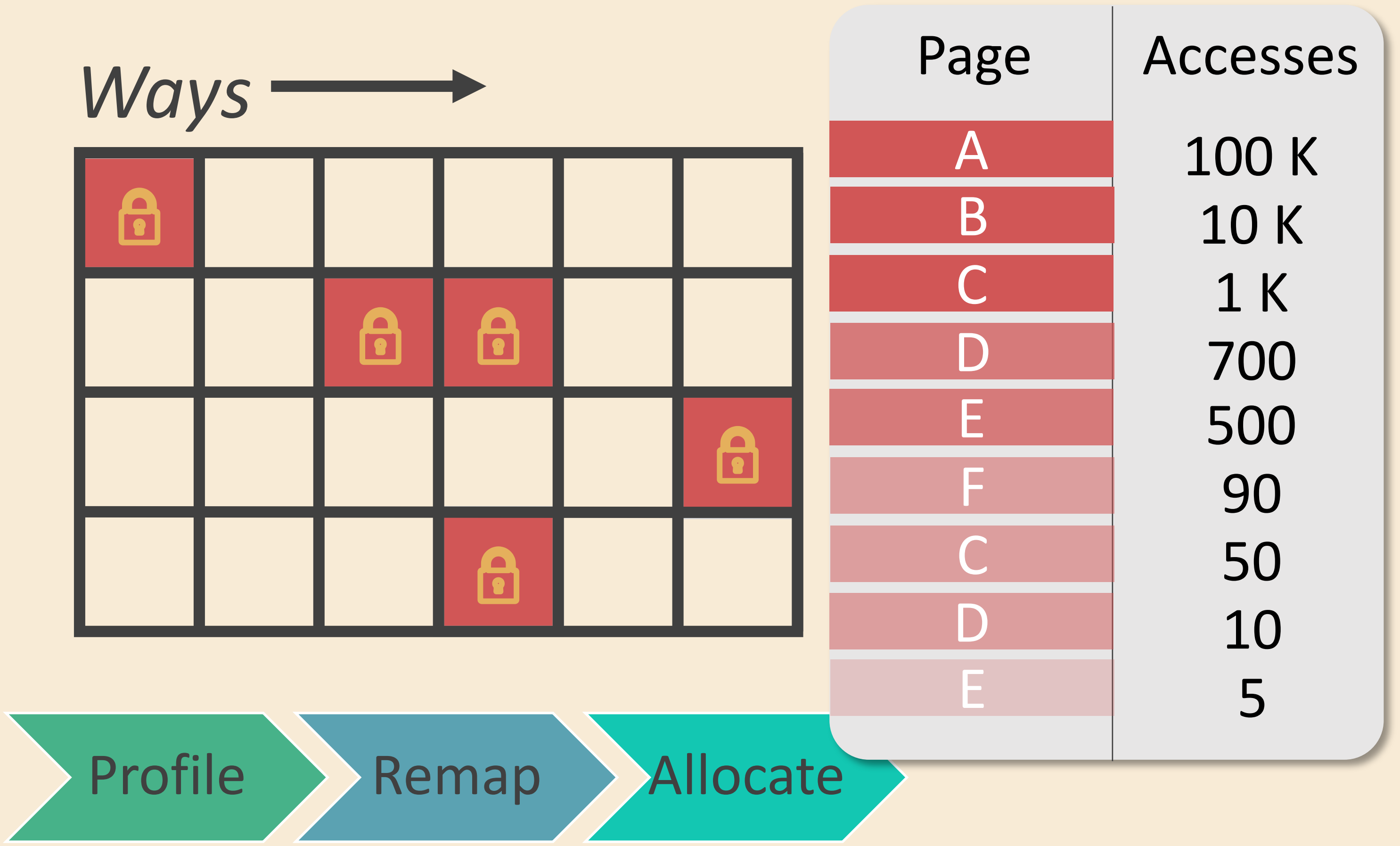
WCET(m)

C

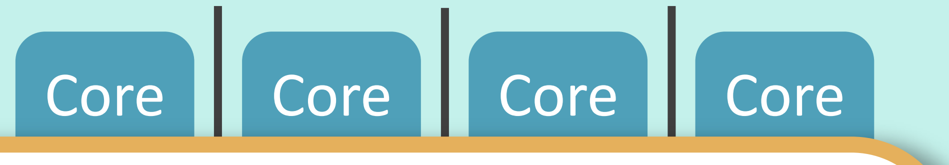
last l

]

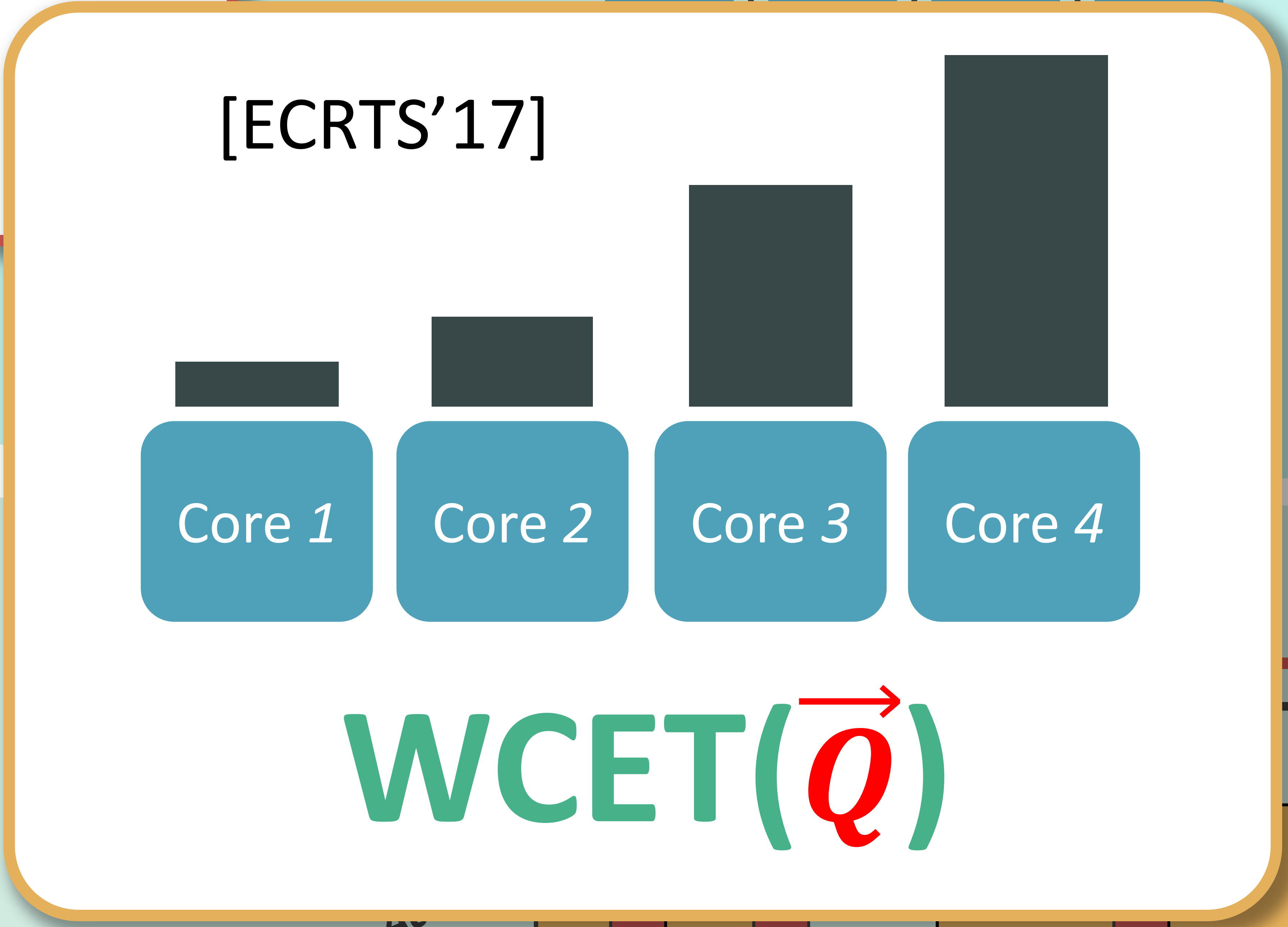
rd)



Page	Accesses
A	100 K
B	10 K
C	1 K
D	700
E	500
F	90
C	50
D	10
E	5



private per-core DRAM banks

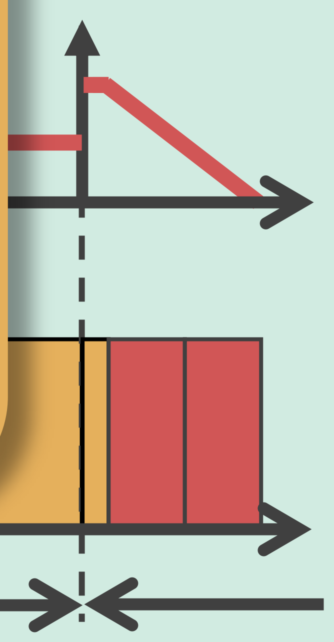


[ECRTS'17]

WCET($\overrightarrow{Q}$)

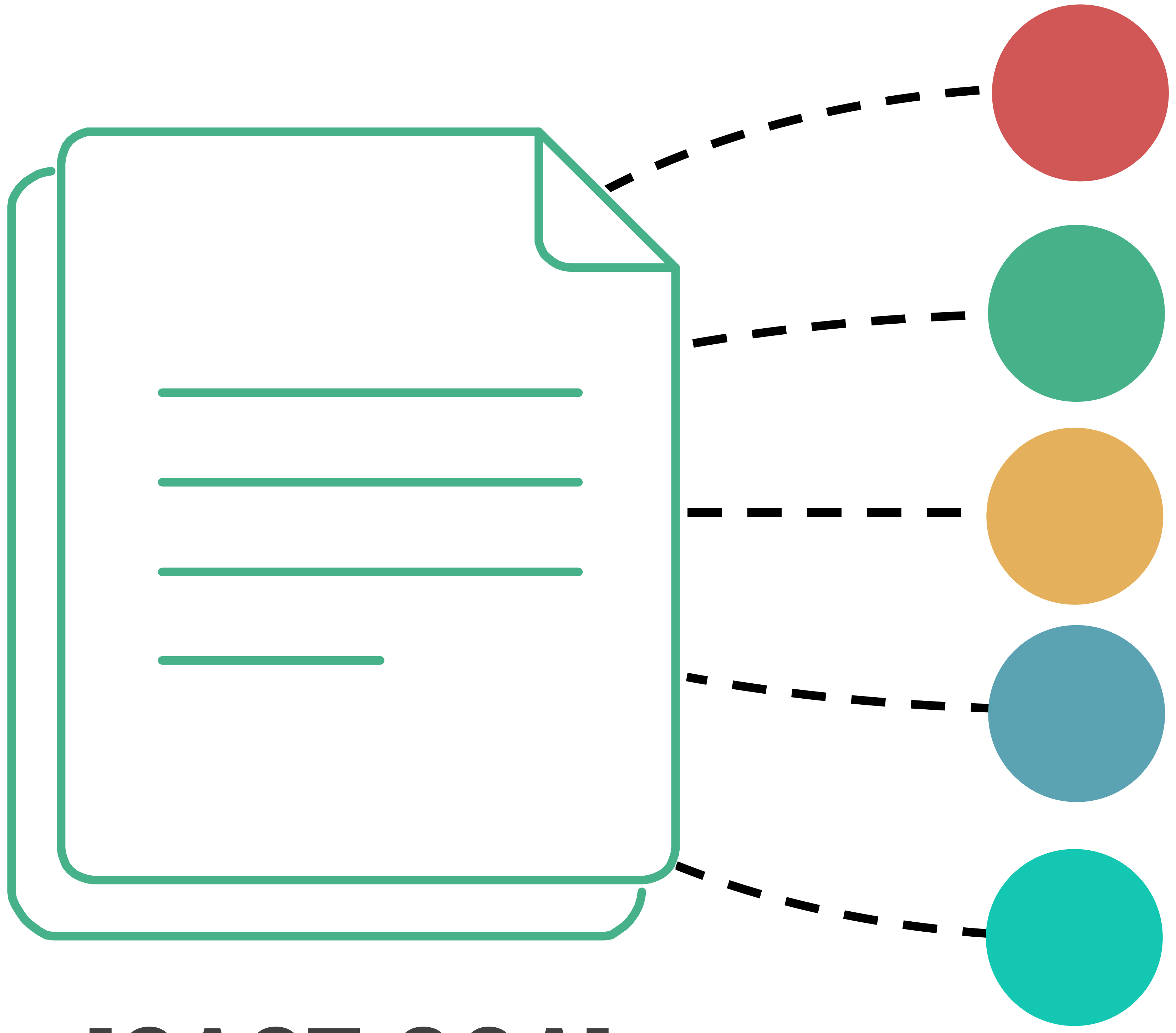
MemGuard

DRAM BW management



SCE & CAST-32A

How SCE fits in the context of state-of-the-art multicore avionics certification guidance



MCP Platforms with **Robust Partitioning**

MCP_Planning_2: *how shared resources are used so as to* ***avoid*** *or* ***mitigate*** *the effect of* ***contention***

MCP_Resource_Usage_3: ***identification*** *of interference channels (shared memory, cache, interconnect, I/O) and* ***means of mitigation***

MCP_Resource_Usage_4: ***identification*** *of resources, their* ***allocation****, and* ***verification*** *that usage does not exceed limits*

MCP_Software_1: *with* ***robust partitioning*** *it is possible to "verify applications on the MCP and determine their WCETs* ***separately****"*

[CAST-32A]

November 2016

MANAGEMENT

[CAST-32A] CAST Position Paper on Multi-core Processors, Certification Authorities Software Team (CAST), November 2016 (Rev 0).
Available at: https://www.faa.gov/aircraft/air_cert/design_approvals/air_software/cast/cast_papers/media/cast-32A.pdf

SCE & CAST-32A

How SCE fits in the context of state-of-the-art multicore avionics certification guidance

Goal of SCE: achieve robust **time** partitioning

Premise of SCE: a plan to **avoid** resource contention

SCE Workflow: first step is interference analysis

SCE Design: operate resources below saturation

SCE Outcome: analyze & verify applications **in isolation**

MCP Platforms with **Robust Partitioning**

MCP_Planning_2: *how shared resources are used so as to* ***avoid*** *or* ***mitigate*** *the effect of* ***contention***

MCP_Resource_Usage_3: ***identification*** *of interference channels (shared memory, cache, interconnect, I/O) and* ***means of mitigation***

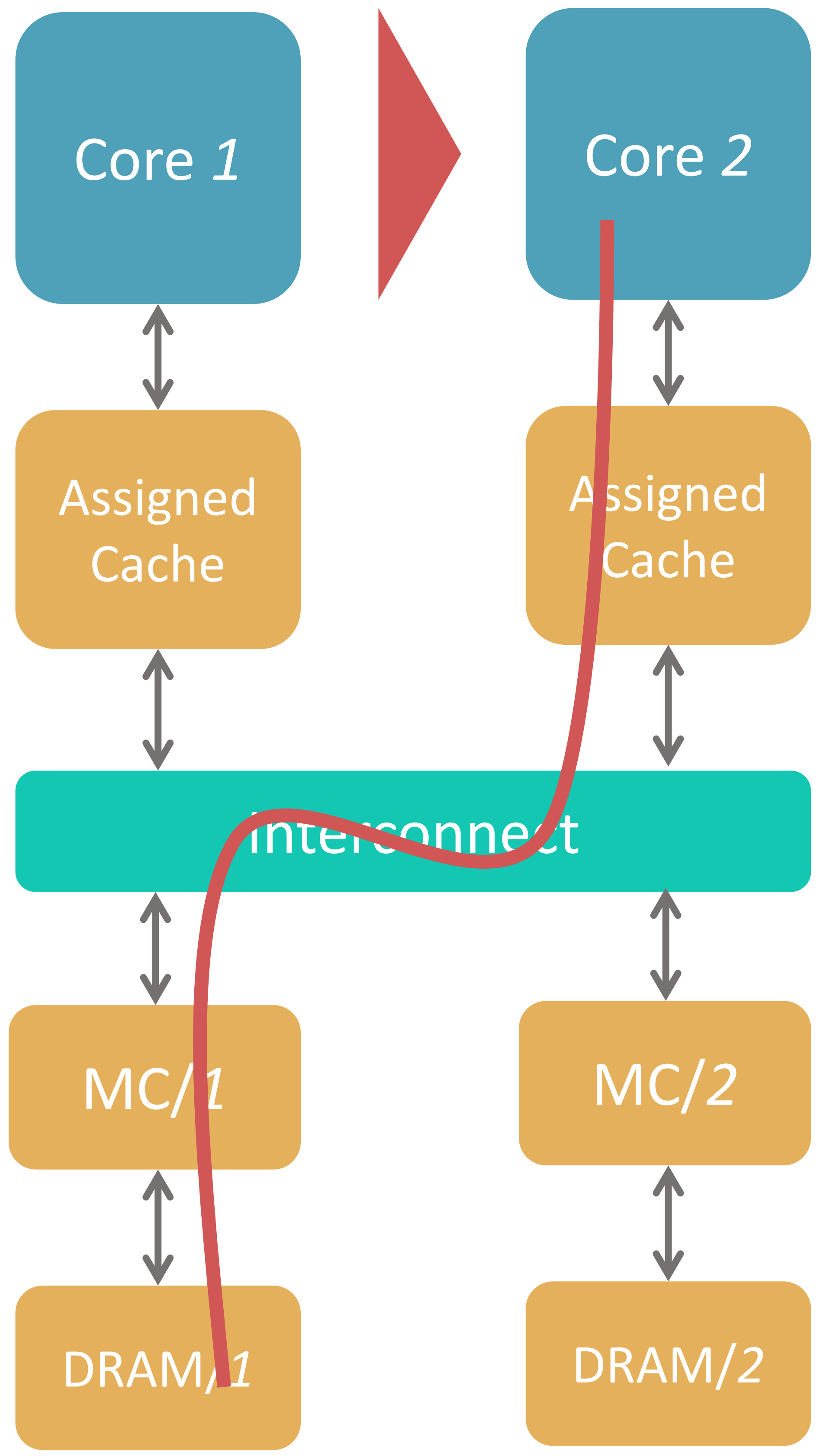
MCP_Resource_Usage_4: ***identification*** *of resources, their* ***allocation****, and* ***verification*** *that usage does not exceed limits*

MCP_Software_1: *with* ***robust partitioning*** *it is possible to "verify applications on the MCP and determine their WCETs* ***separately****"*

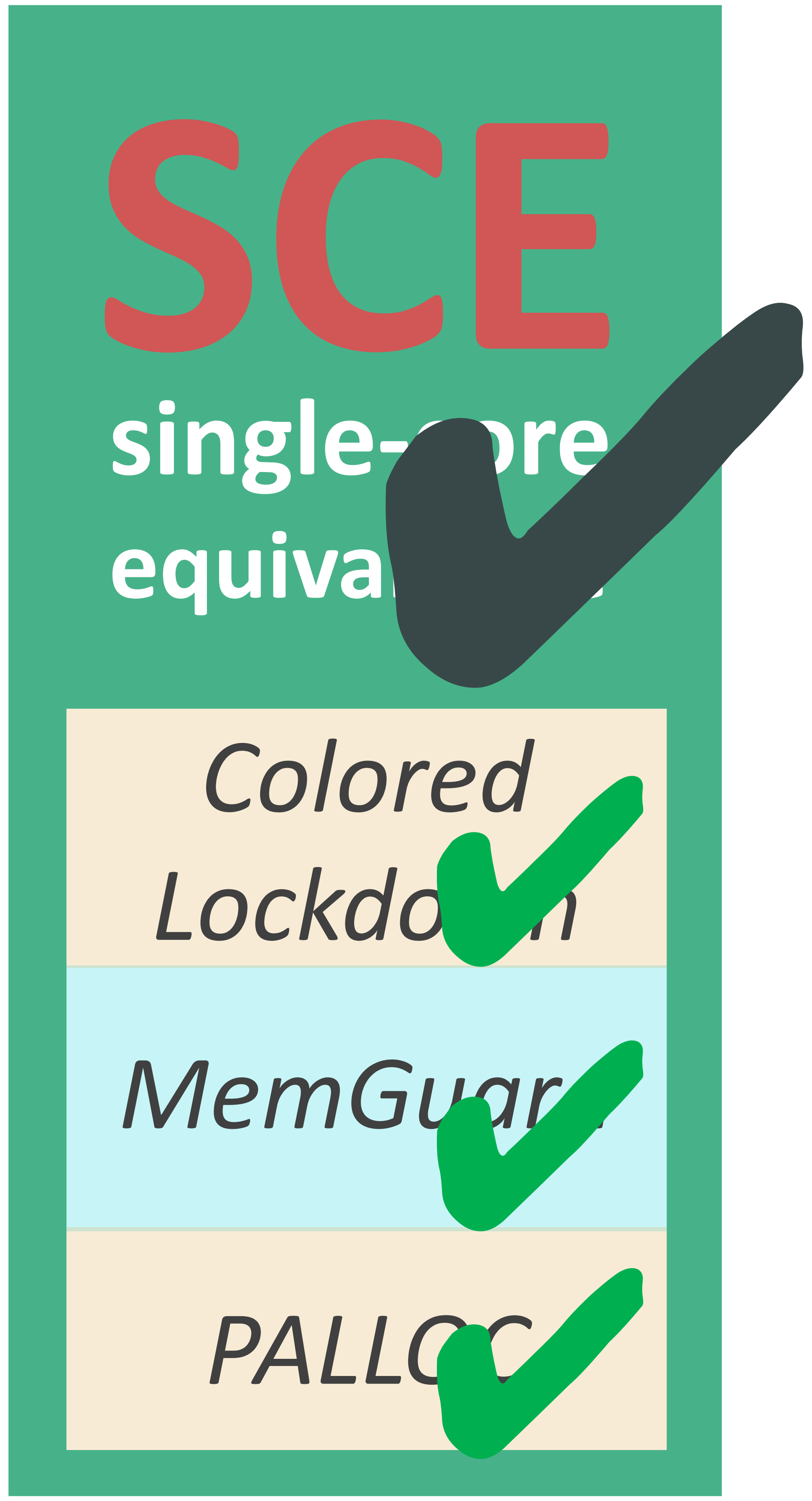
[CAST-32A] CAST Position Paper on Multi-core Processors, Certification Authorities Software Team (CAST), November 2016 (Rev 0).
Available at: https://www.faa.gov/aircraft/air_cert/design_approvals/air_software/cast/cast_papers/media/cast-32A.pdf

OPEN CHALLENGES

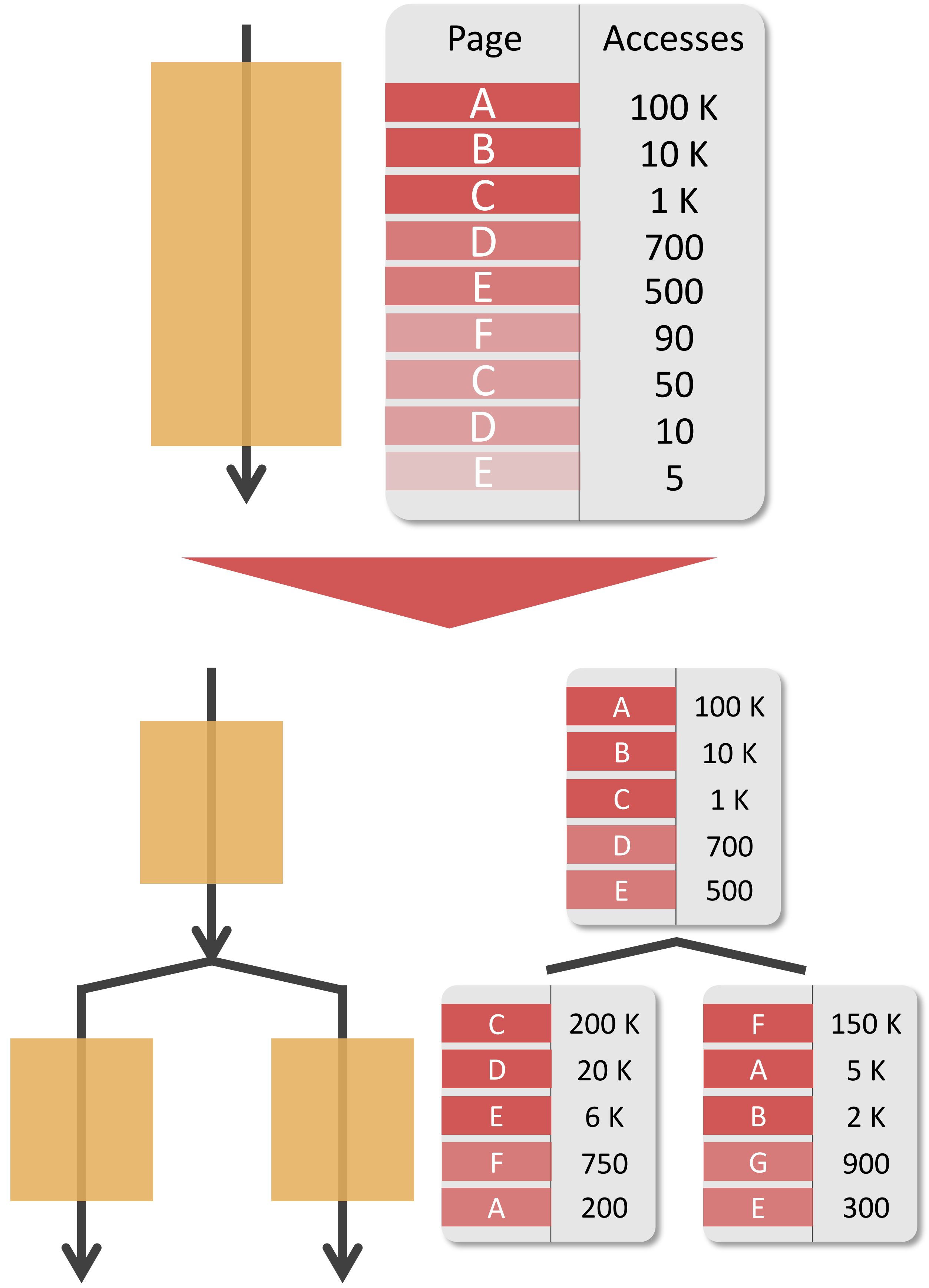
Communication



Verification



Optimization



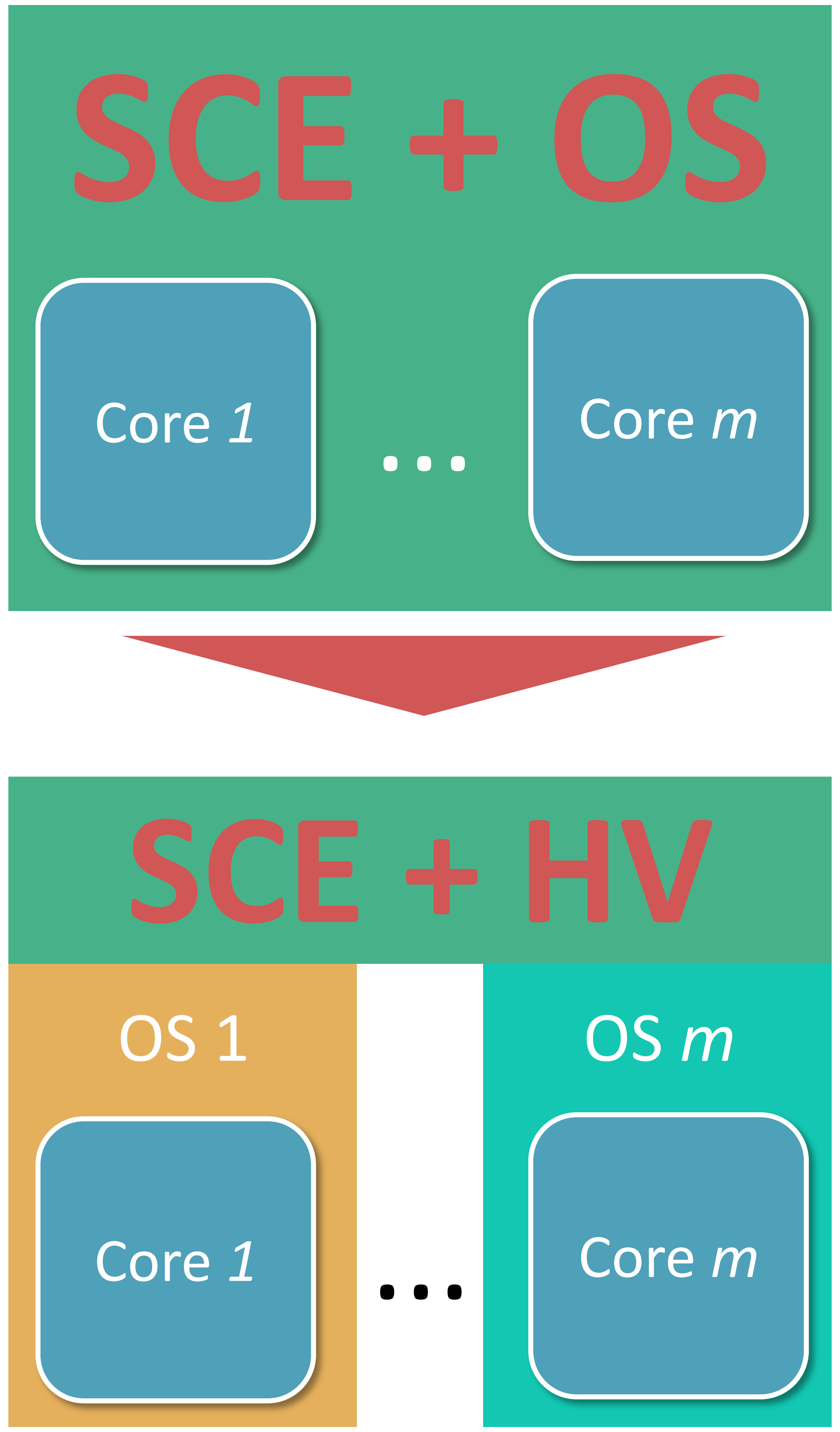
Page	Accesses
A	100 K
B	10 K
C	1 K
D	700
E	500
F	90
C	50
D	10
E	5

A	100 K
B	10 K
C	1 K
D	700
E	500

C	200 K
D	20 K
E	6 K
F	750
A	200

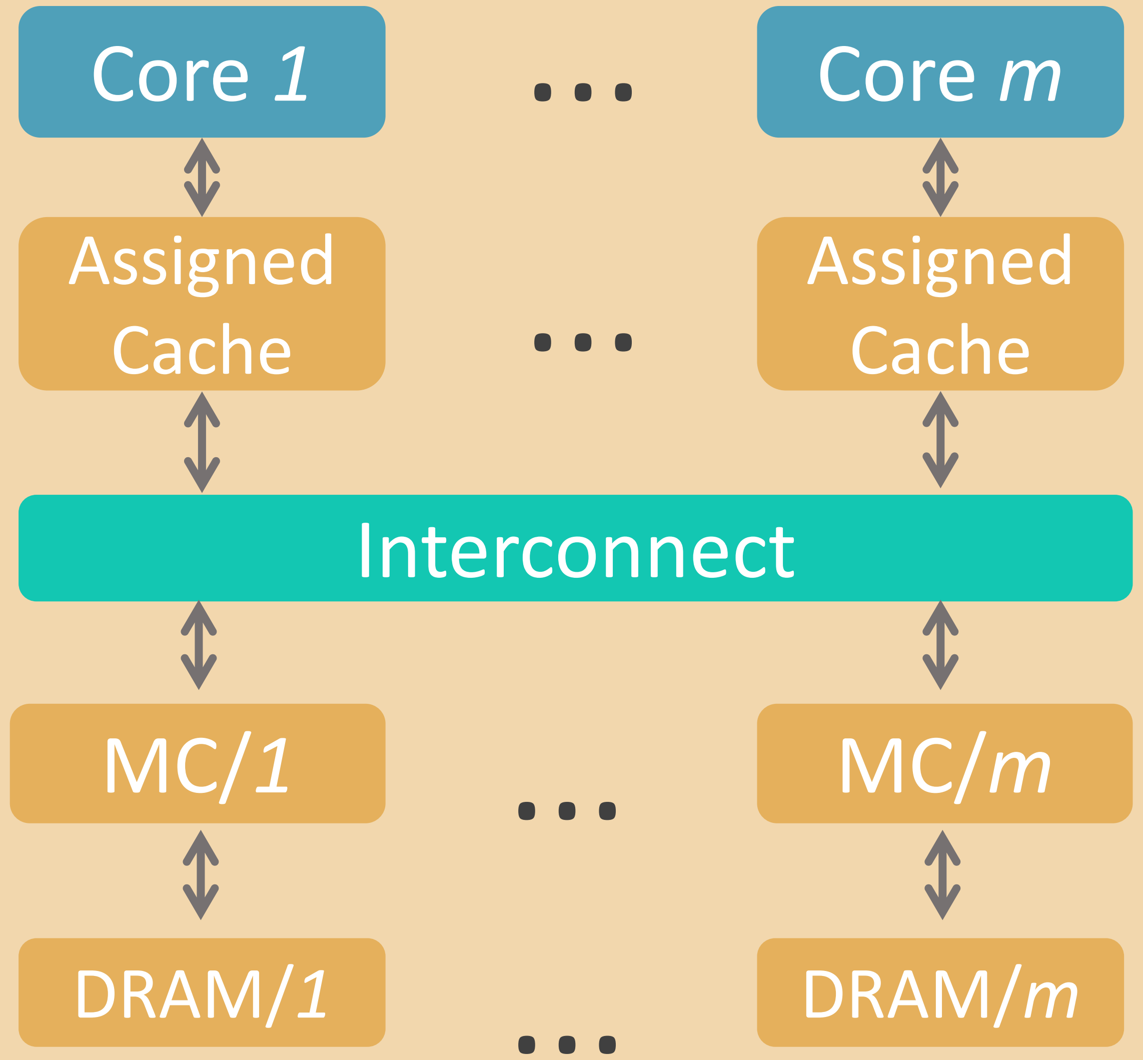
F	150 K
A	5 K
B	2 K
G	900
E	300

Cross-OS SCE



SCE

single-core equivalence



Core *1* … Core *m*

Assigned Cache … Assigned Cache

Interconnect

MC/*1* … MC/*m*

DRAM/*1* … DRAM/*m*

& SCE

single-core equivalence

[CAST-32A]

Minimal Multicore Avionics Certification Guidance

Rockwell Collins

WIND RIVER

Carnegie Mellon University

UNIVERSITY of York

AFRL

THE AIR FORCE RESEARCH LABORATORY

LEAD | DISCOVER | DEVELOP | DELIVER

ETRI

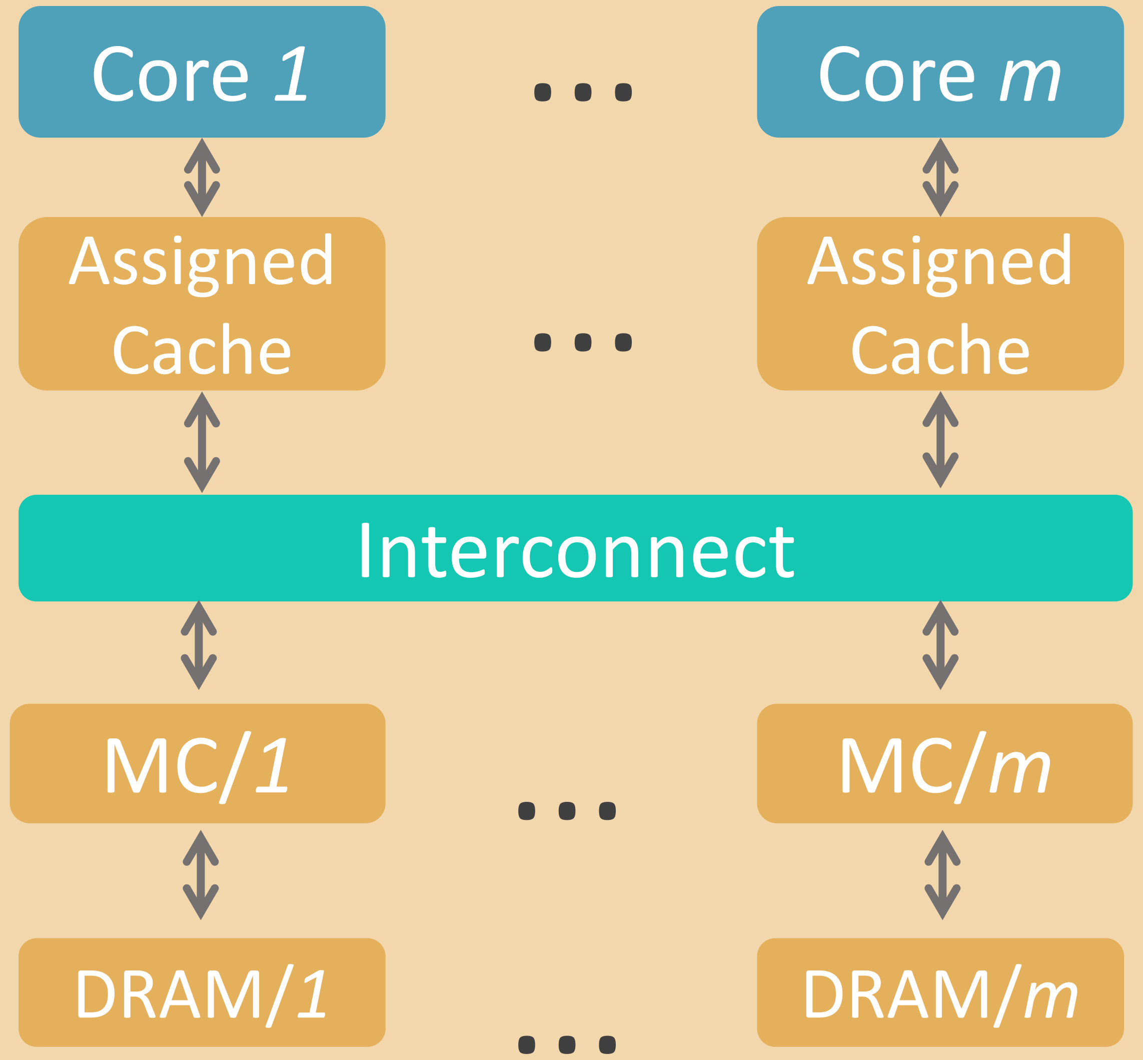
LOCKHEED MARTIN

GUIDANCE

STANDARD

SCE

single-core equivalence



& SCE

single-core equivalence

[CAST-32A]

Minimal Multicore Avionics Certification Guidance

Rockwell Collins

WIND RIVER

Carnegie Mellon University

UNIVERSITY of York

AFRL THE AIR FORCE RESEARCH LABORATORY LEAD | DISCOVER | DEVELOP | DELIVER

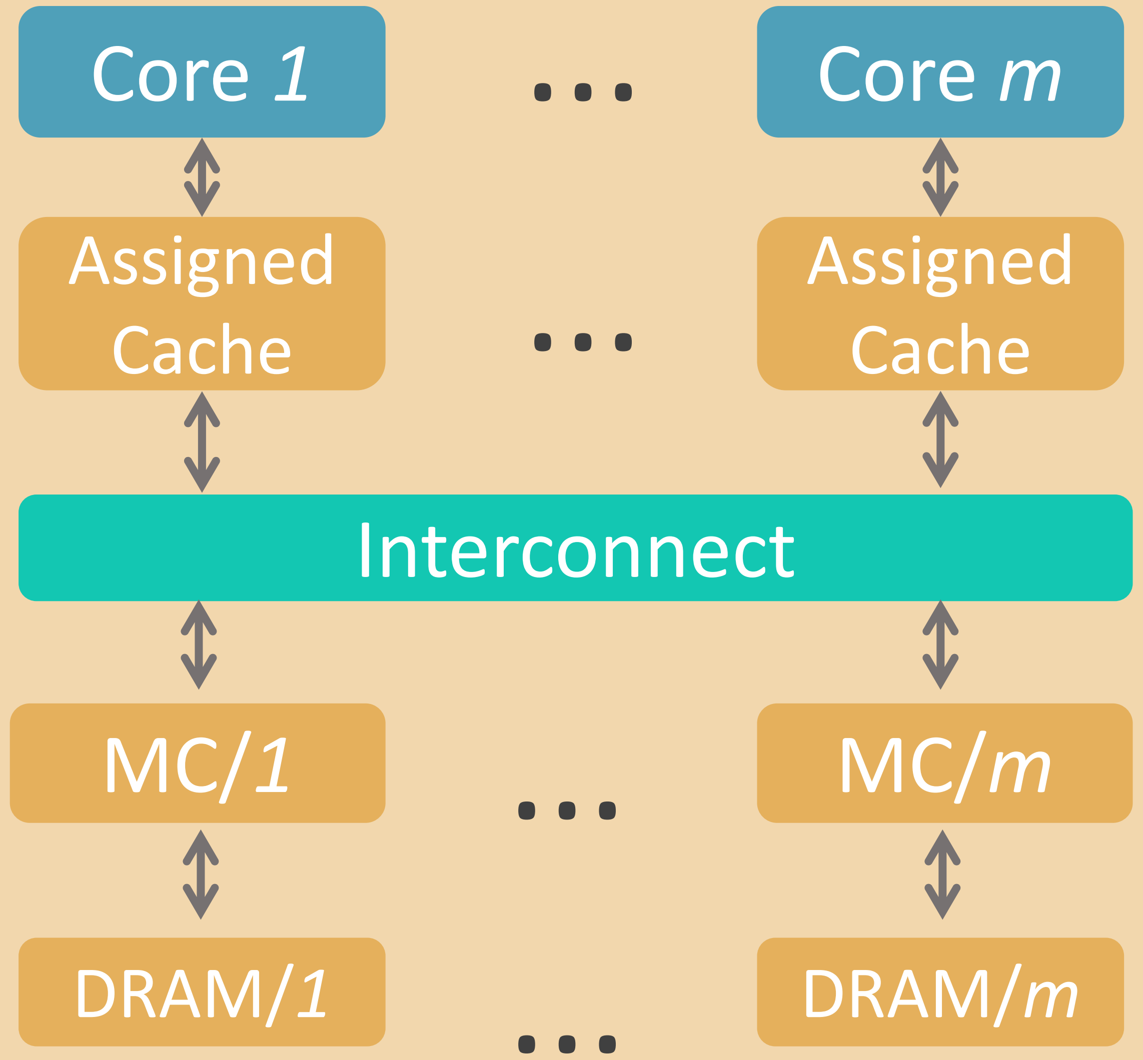
ETRI

LOCKHEED MARTIN

We want your contribute. **Get on board !**

SCE

single-core equivalence



& SCE

single-core equivalence

[CAST-32A]

Minimal Multicore Avionics Certification Guidance

Rockwell Collins WIND RIVER Carnegie Mellon University UNIVERSITY of York AFRL THE AIR FORCE RESEARCH LABORATORY LEAD I DISCOVER I DEVELOP I DELIVER

ETRI LOCKHEED MARTIN

We want your contribute. **Get on board !**

Q & A

THANKS FOR LISTENING

Renato Mancuso
rmancus2@illinois.edu