

Research Review 2019

# Using All Processor Cores While Being Confident about Timing

Bjorn Andersson

Software Engineering Institute  
Carnegie Mellon University  
Pittsburgh, PA 15213

Copyright 2019 Carnegie Mellon University. All Rights Reserved.

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

The view, opinions, and/or findings contained in this material are those of the author(s) and should not be construed as an official Government position, policy, or decision, unless designated by other documentation.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

This material may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other use. Requests for permission should be directed to the Software Engineering Institute at [permission@sei.cmu.edu](mailto:permission@sei.cmu.edu).

DM19-0879

# RESEARCH REVIEW 2019

Using All Processor Cores While  
Being Confident about Timing

Bjorn Andersson

# DoD Systems Interact with Their Physical Environment



# DoD Systems Include Software



# DoD Systems Include Software that Interacts with the Physical Environment



# DoD Systems Include Software that Has Real-Time Requirements



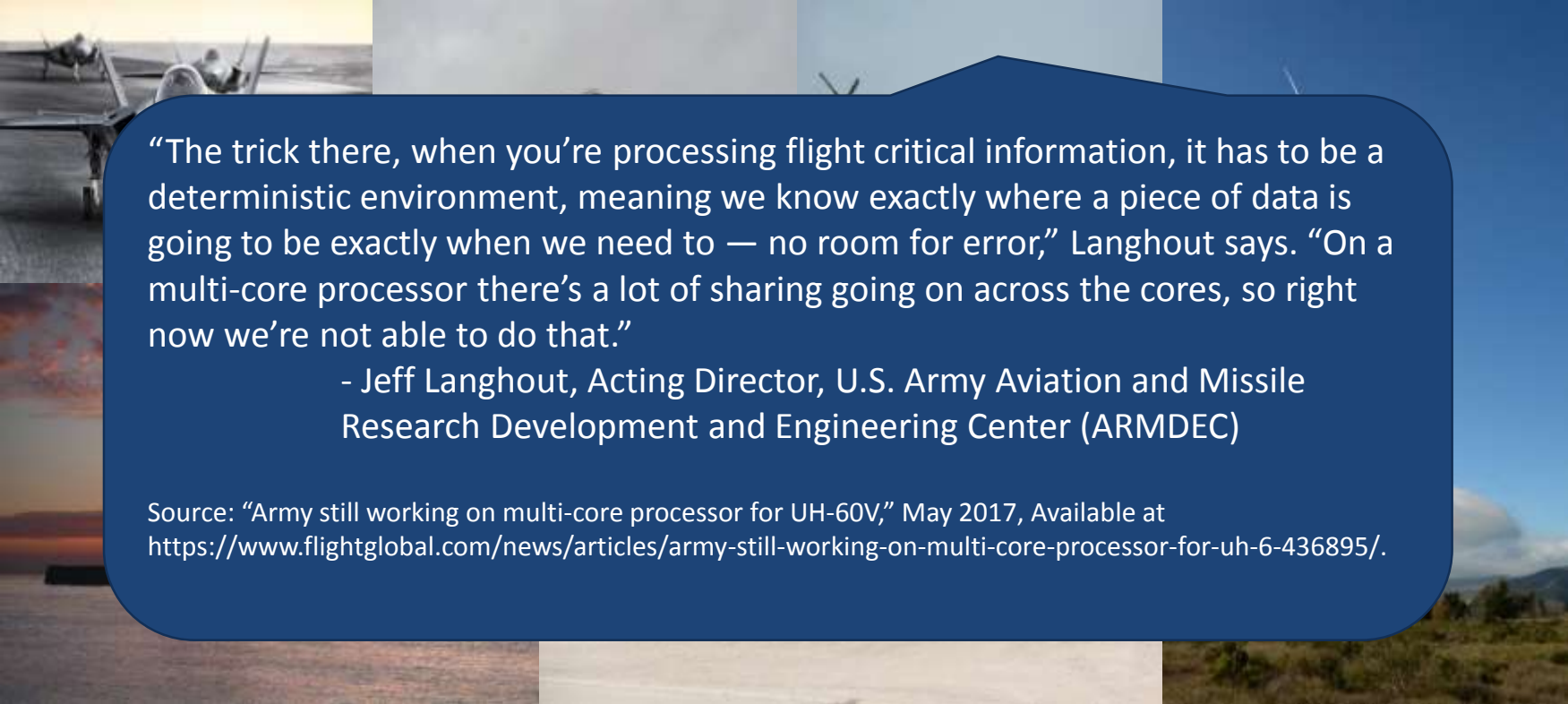
# Satisfying Real-Time Requirements Is a Challenge for the DoD in General



# Satisfying Real-Time Requirements Is Challenging for Upgrading Blackhawk UH-60 Helicopter



# Satisfying Real-Time Requirements Is Challenging for Upgrading the Blackhawk UH-60 Helicopter



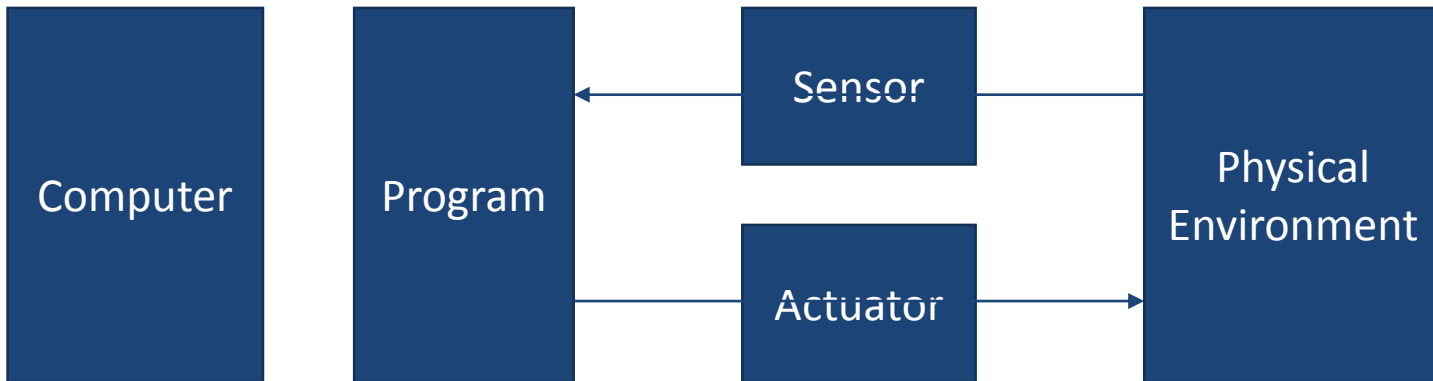
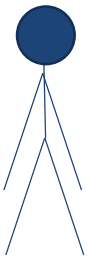
“The trick there, when you’re processing flight critical information, it has to be a deterministic environment, meaning we know exactly where a piece of data is going to be exactly when we need to — no room for error,” Langhout says. “On a multi-core processor there’s a lot of sharing going on across the cores, so right now we’re not able to do that.”

- Jeff Langhout, Acting Director, U.S. Army Aviation and Missile Research Development and Engineering Center (ARMDEC)

Source: “Army still working on multi-core processor for UH-60V,” May 2017, Available at <https://www.flightglobal.com/news/articles/army-still-working-on-multi-core-processor-for-uh-6-436895/>.

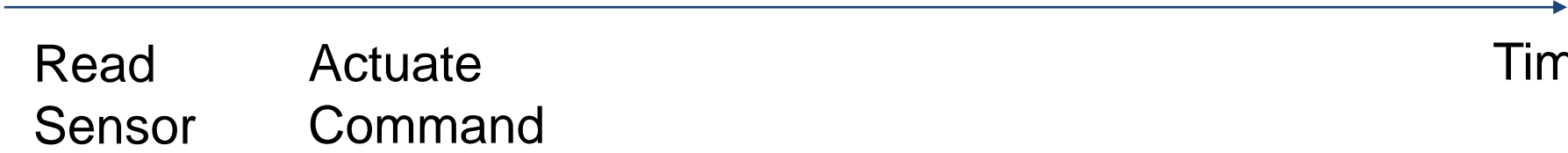
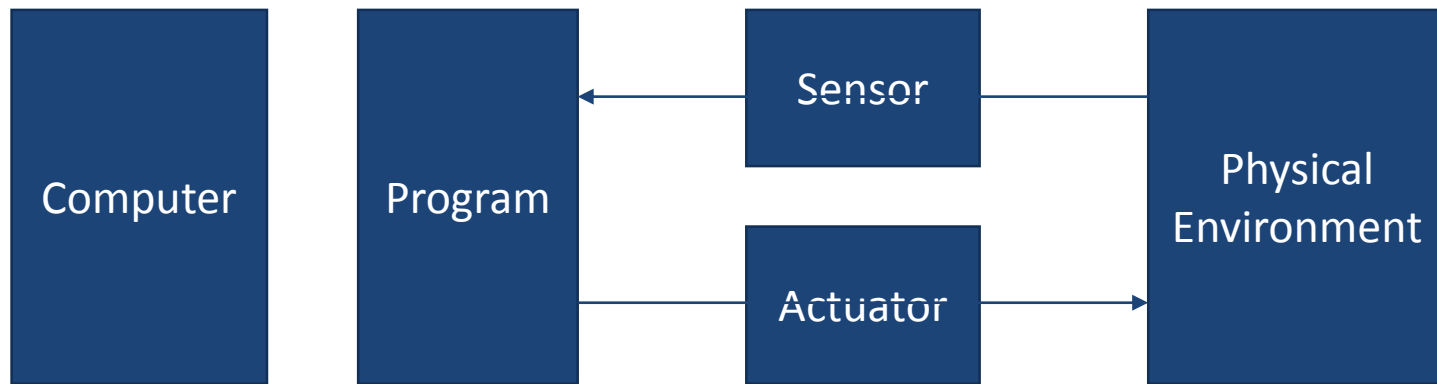
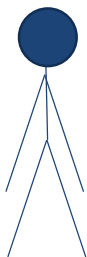
# Commonality of DoD Systems

Human  
Operator



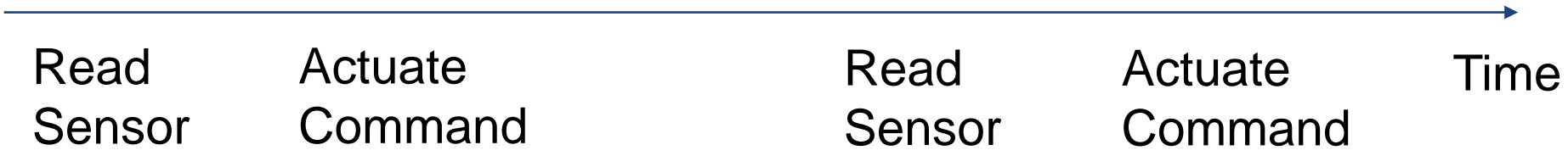
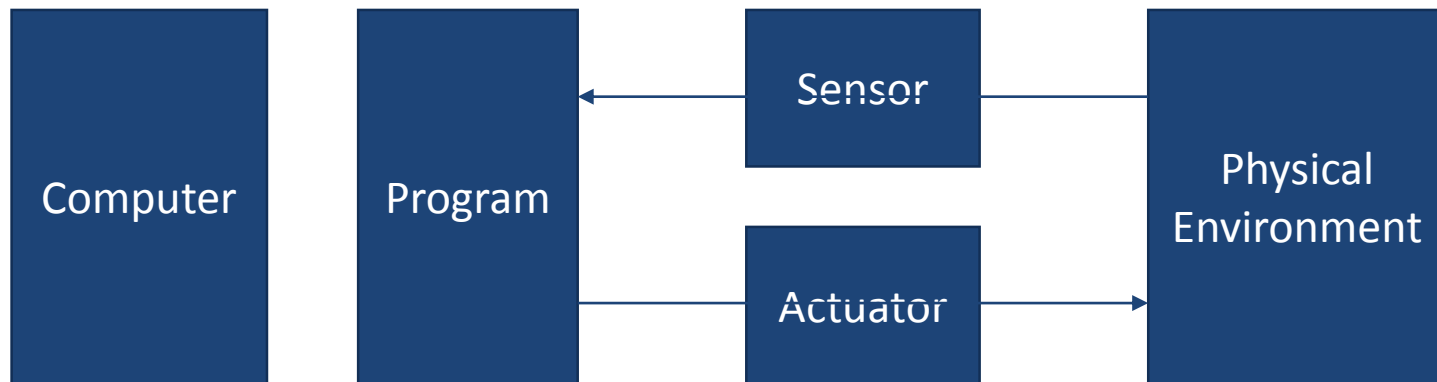
# Commonality of DoD Systems

Human operator



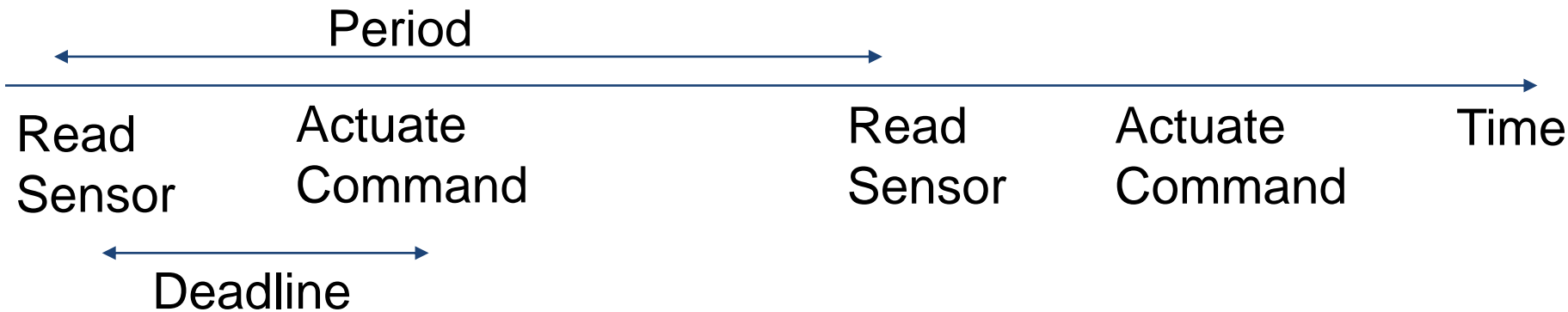
# Commonality of DoD Systems

Human Operator



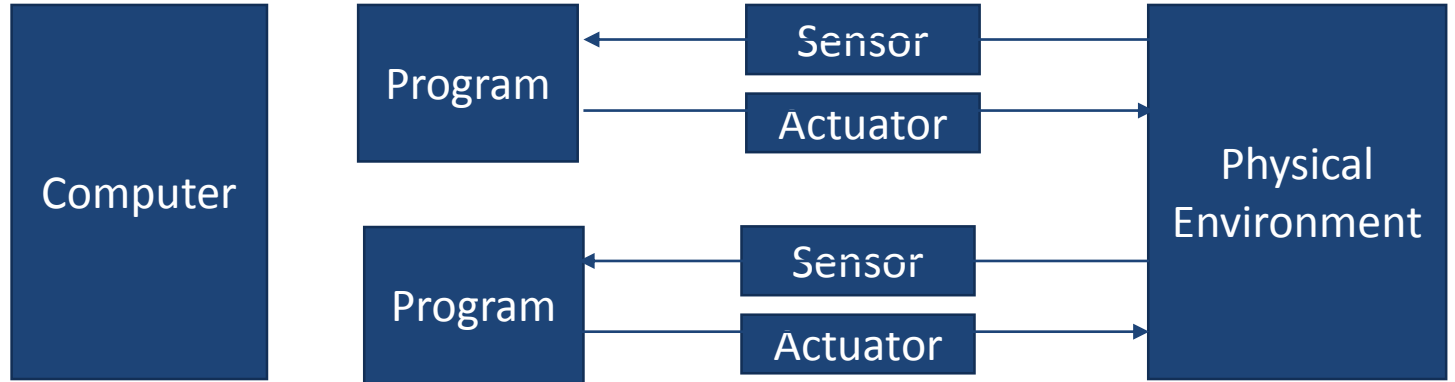
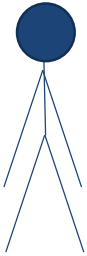
# Commonality of DoD Systems

Human operator



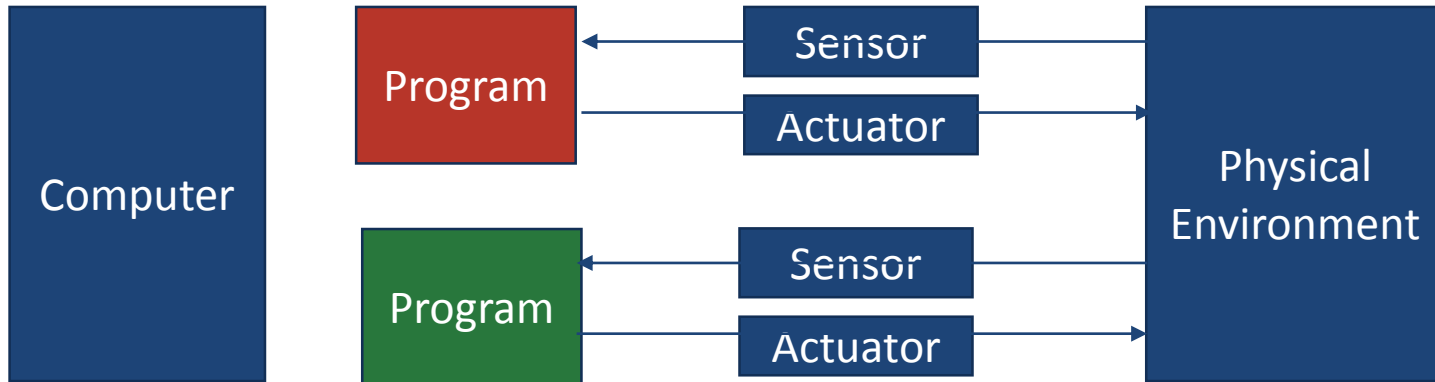
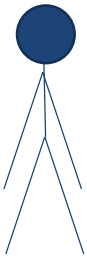
# Commonality of DoD Systems

Human Operator



# Commonality of DoD Systems

Human operator



Read  
Sensor

Actuate  
Command

Read  
Sensor

Actuate  
Command

Time →

Read  
Sensor

Actuate  
Command

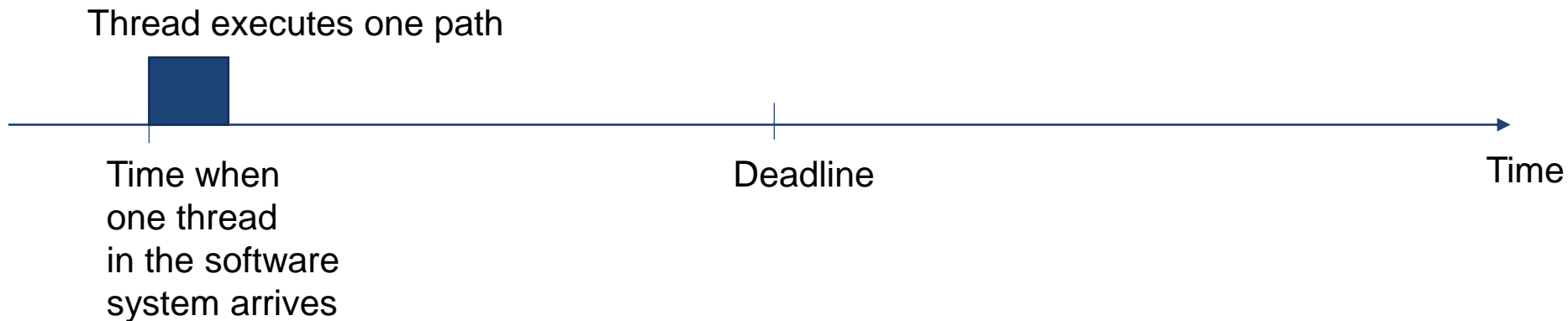
Read  
Sensor

Actuate  
Command

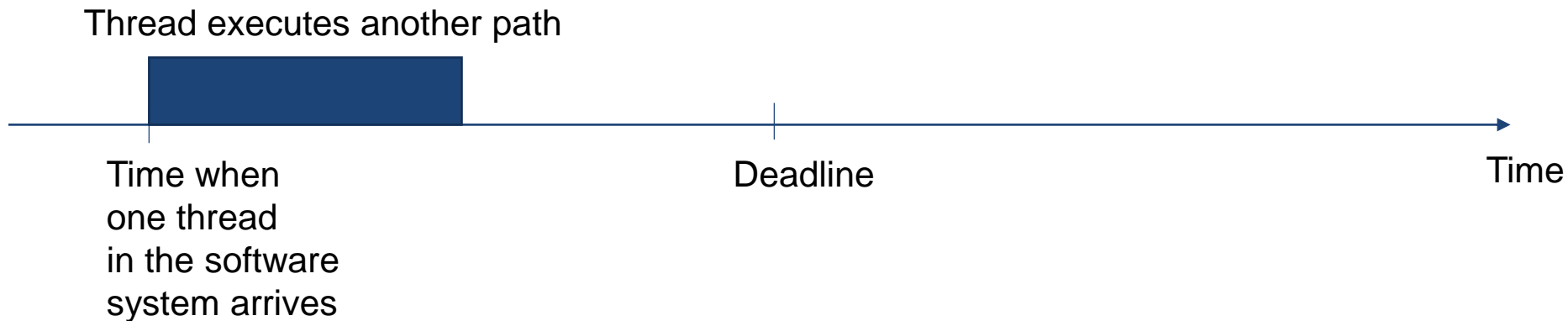
# What Makes it Challenging to Satisfy Real-Time Requirements?

# What Causes the Delay of Software?

# What Causes the Delay of Software?

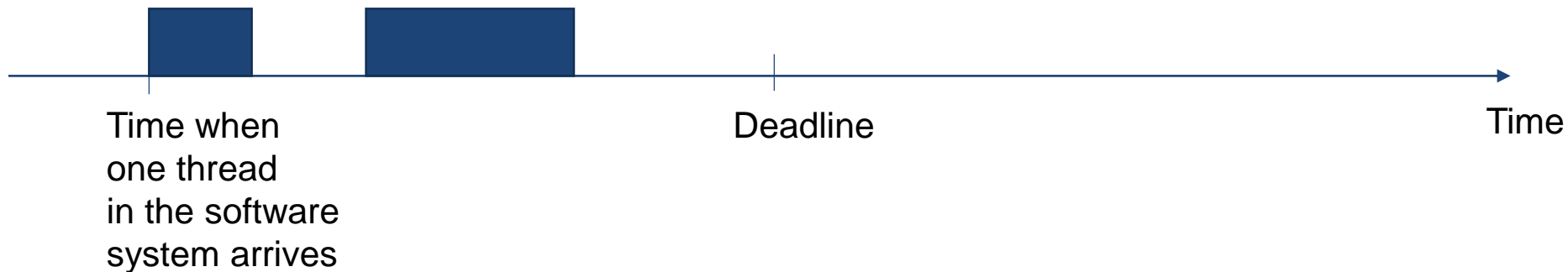


# What Causes the Delay of Software?



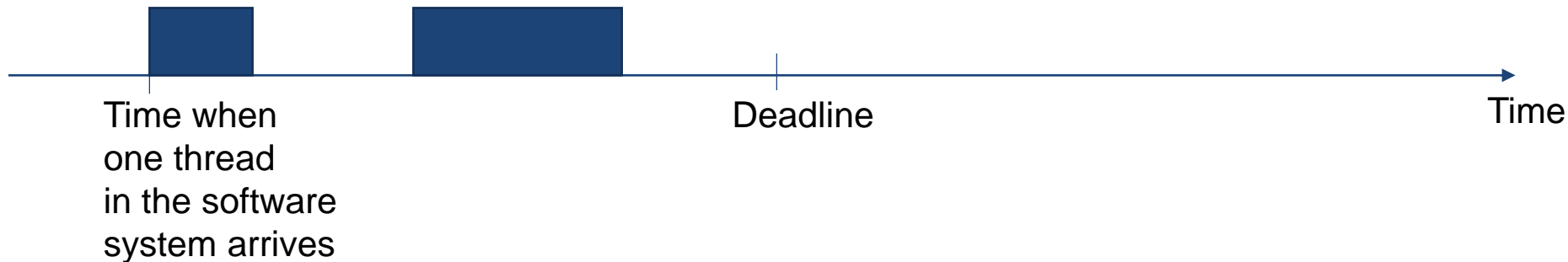
# What Causes the Delay of Software?

Preemption.  
Another  
thread  
uses  
the  
processor

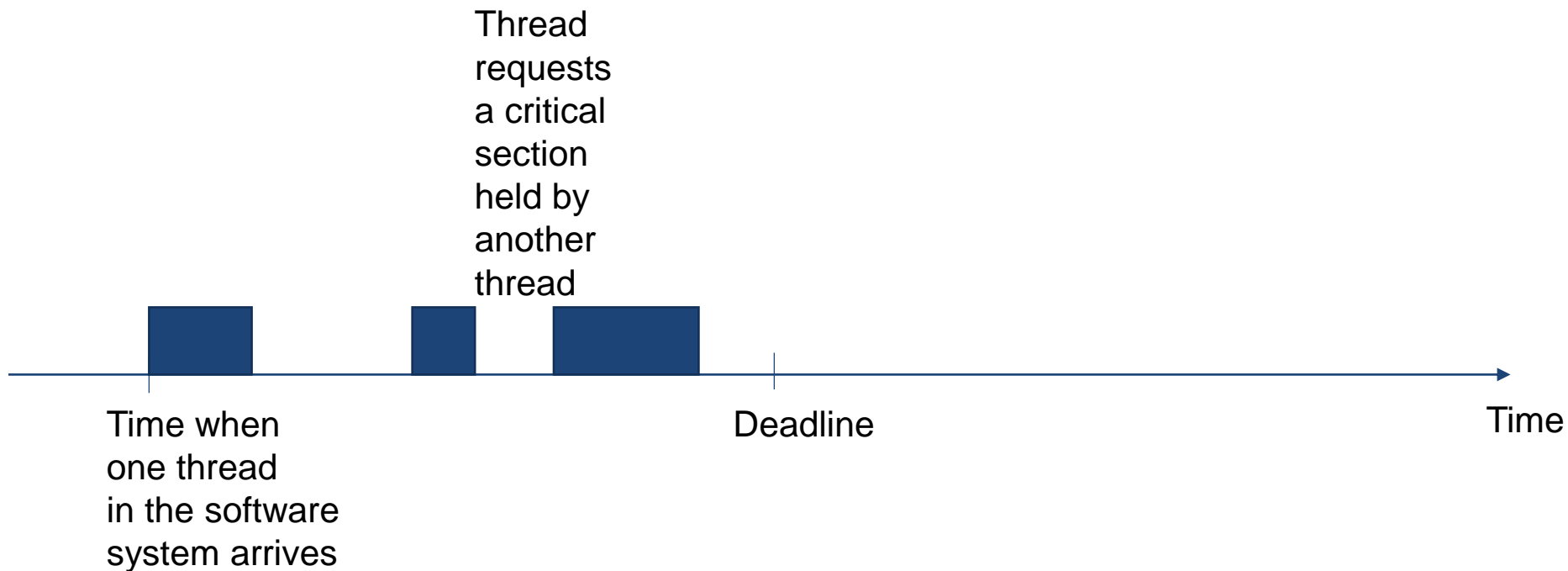


# What Causes the Delay of Software?

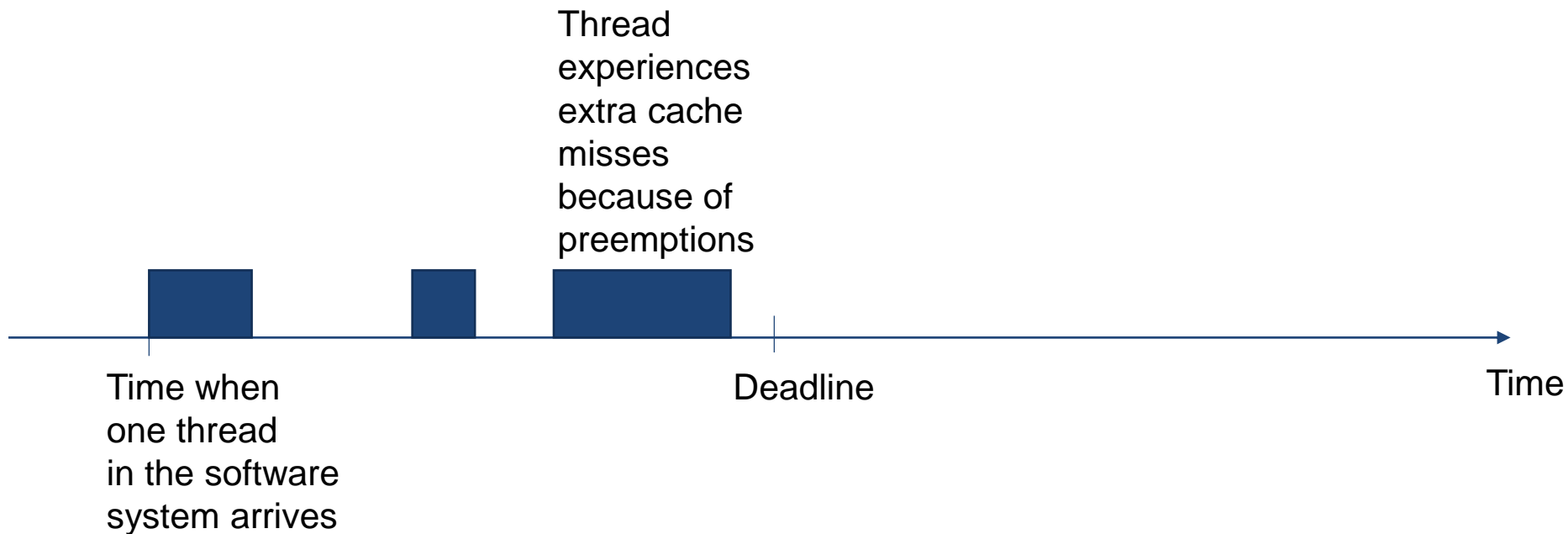
Interrupt  
service  
routine  
executes



# What Causes the Delay of Software?



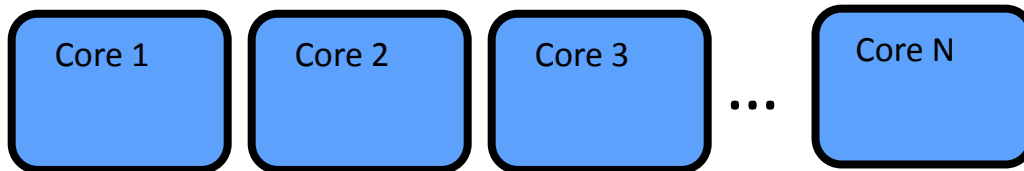
# What causes the delay of software?



# Real-time Requirements of Software Executing on a Multicore Processor

## Hardware Trends

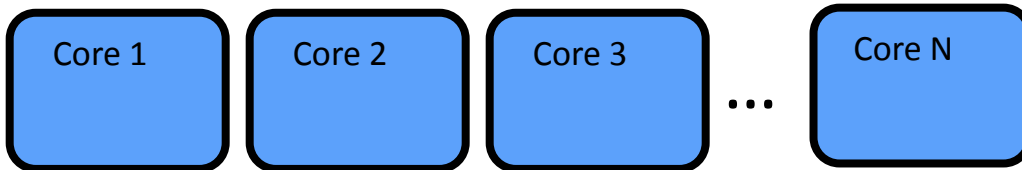
- *All computers are multicores.*



# Real-Time Requirements of Software Executing on a Multicore Processor

## Hardware Trends

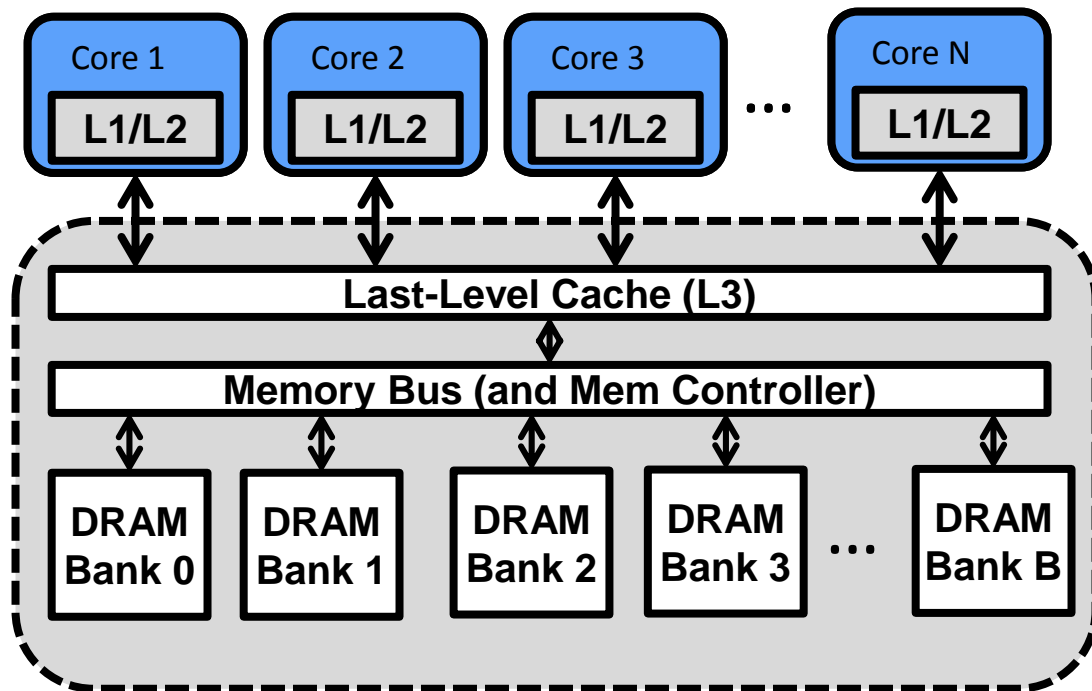
- *All computers are multicores.*
- *Most chip makers do not offer single core.*

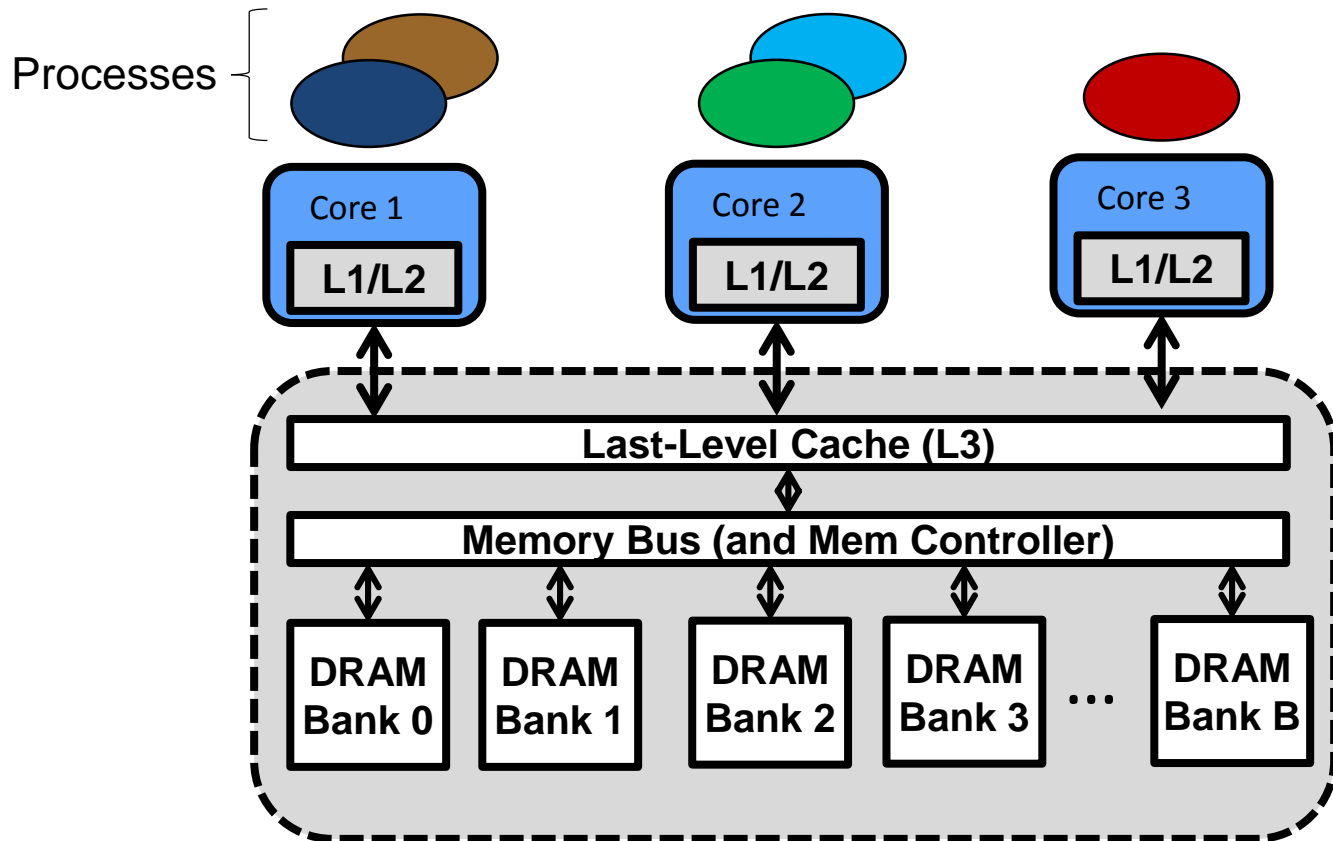


# Real-Time Requirements of Software Executing on a Multicore Processor

## Hardware trends

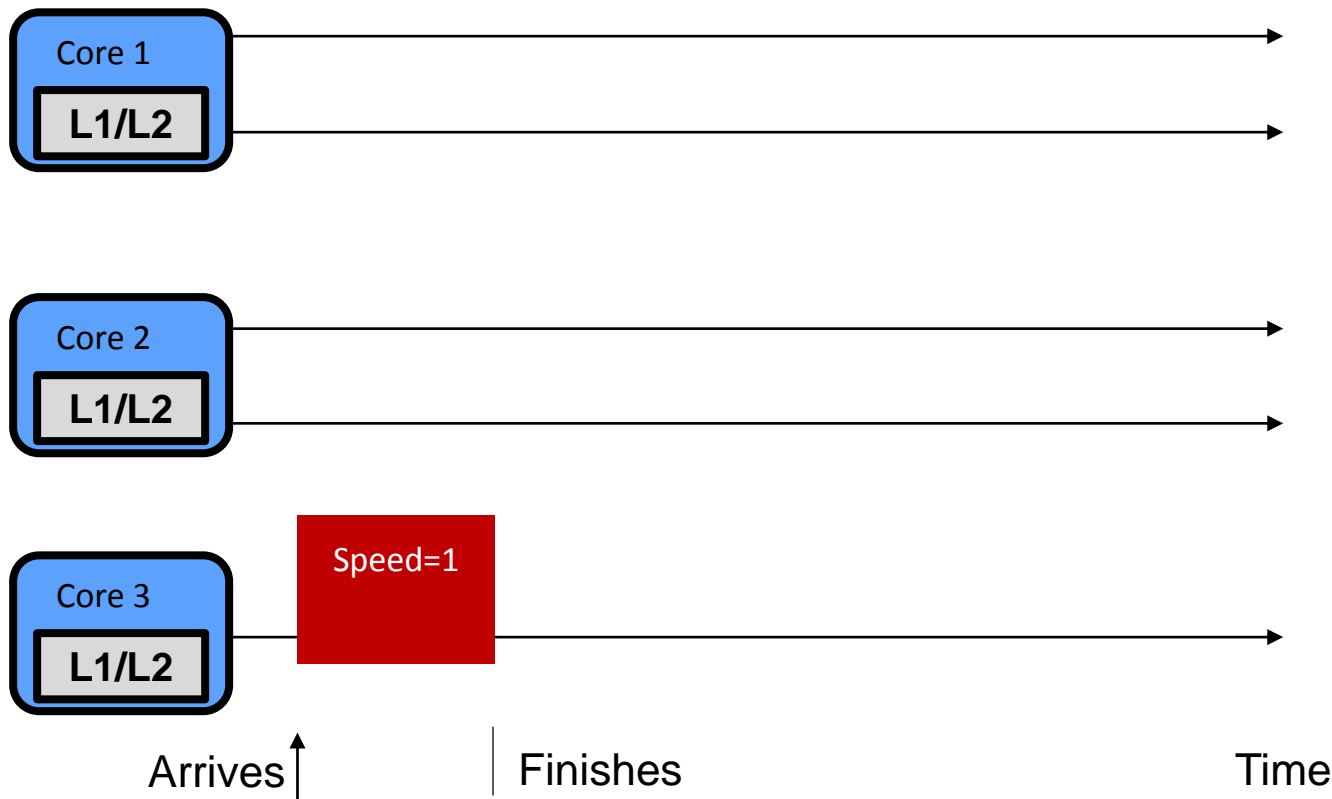
- *All computers are multicores.*
- *Most chip makers do not offer single core.*
- *Most multicores have shared memory.*



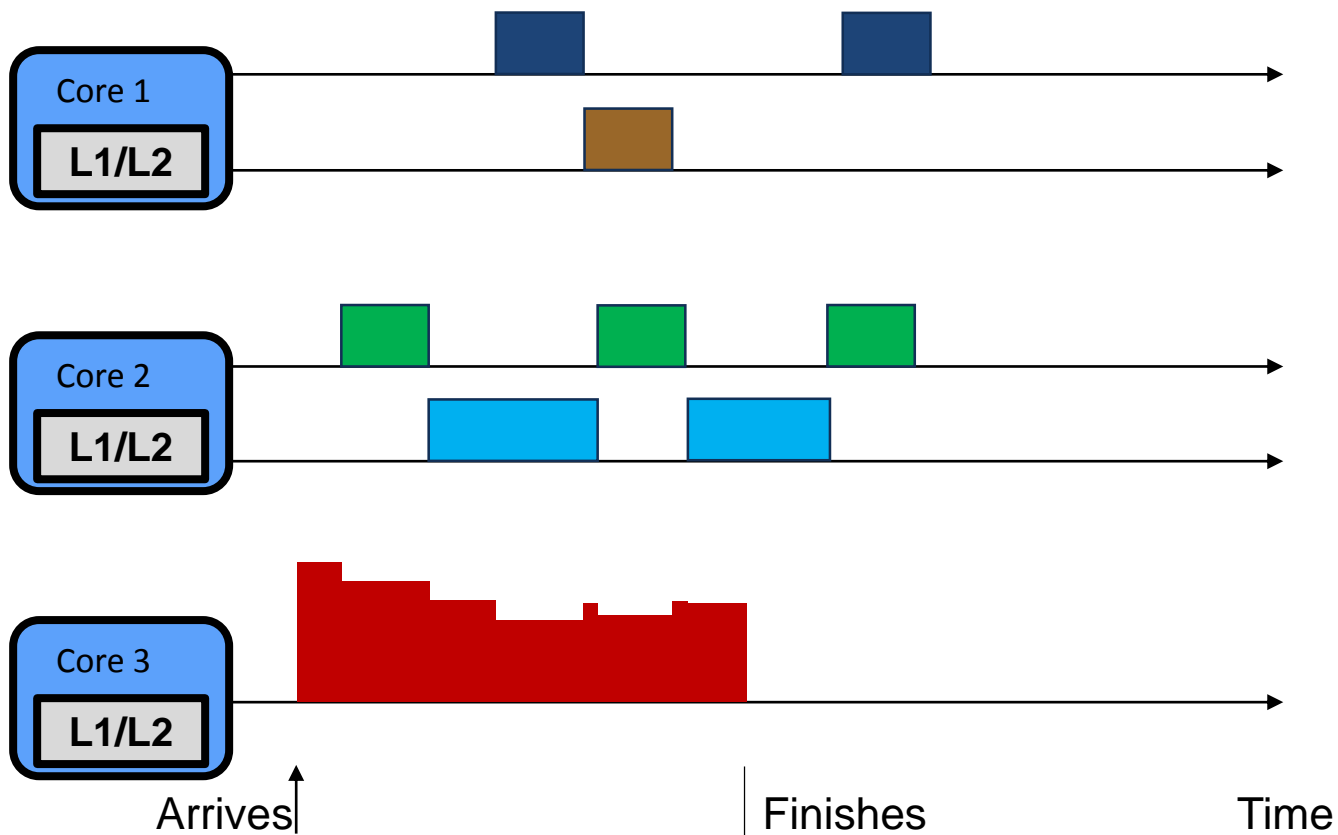


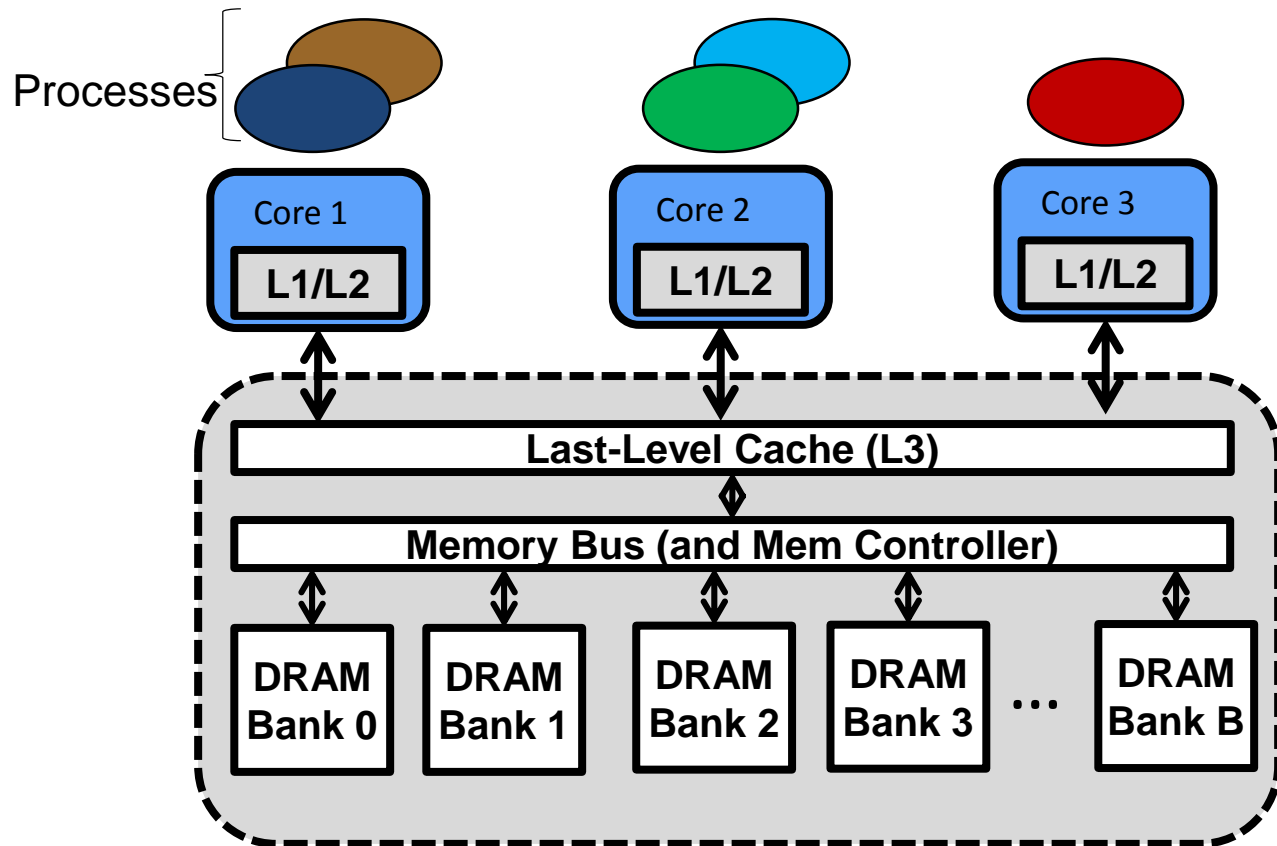
**Problem:** For each process, compute an upper bound on its response time.

# How Co-Runners Impact Speed of Execution



# How Co-Runners Impact Speed of Execution

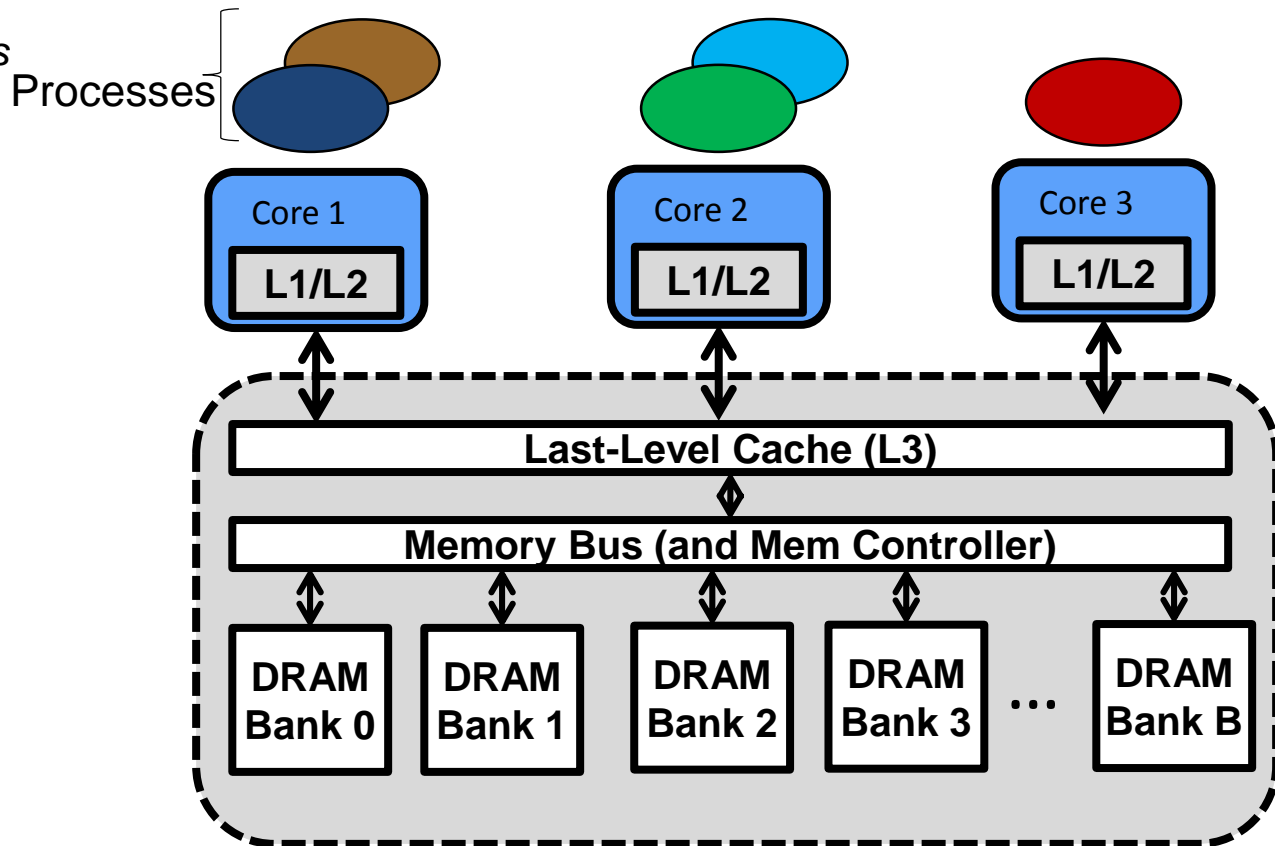




**Problem:** For each process, compute an upper bound on its response time.

## Issues

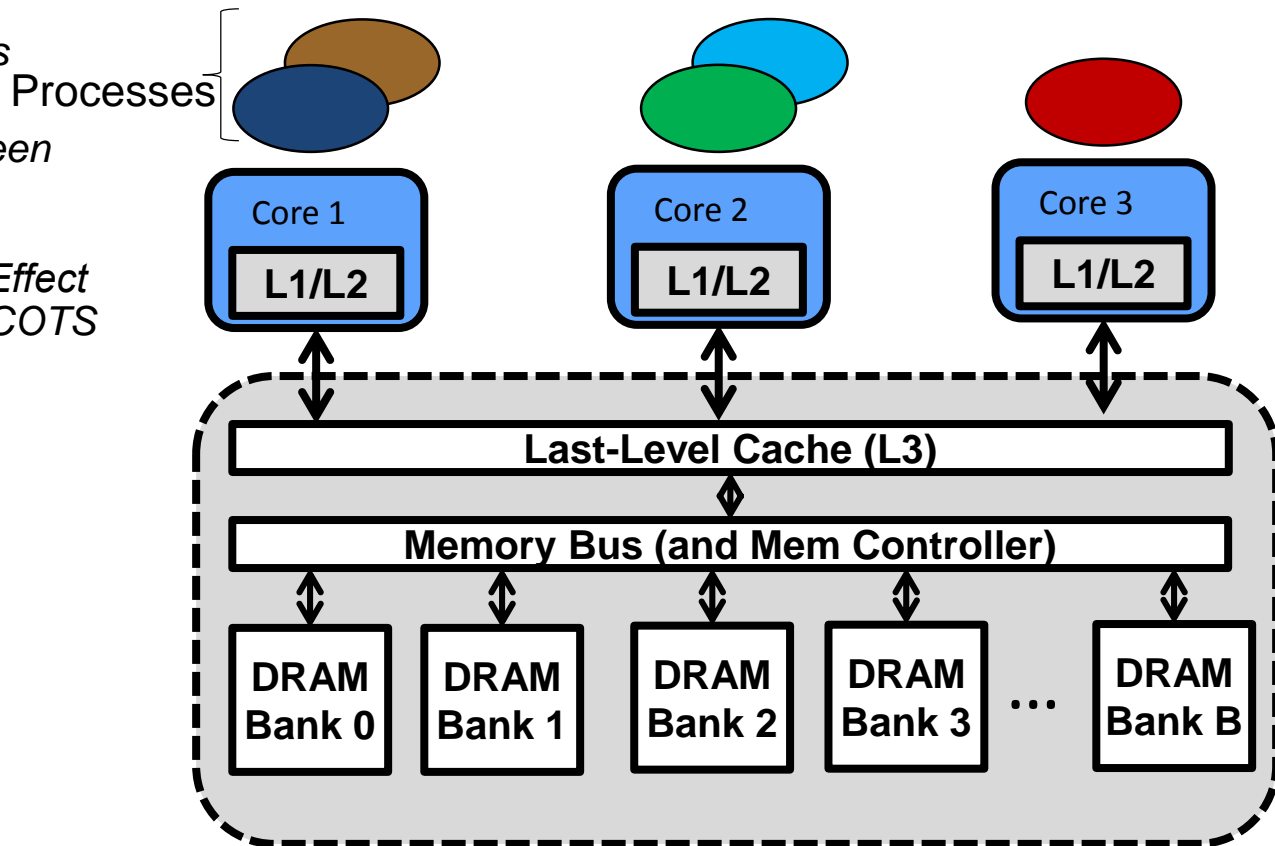
- *Shared hardware resources impact timing.*



## Issues

- *Shared hardware resources impact timing.*
- *103 times slowdown has been observed*

*H. Yun and P. K. Valsan,  
“Evaluating the Isolation Effect  
of Cache Partitioning on COTS  
Multicore Platforms,”  
OSPERT, 2015.*

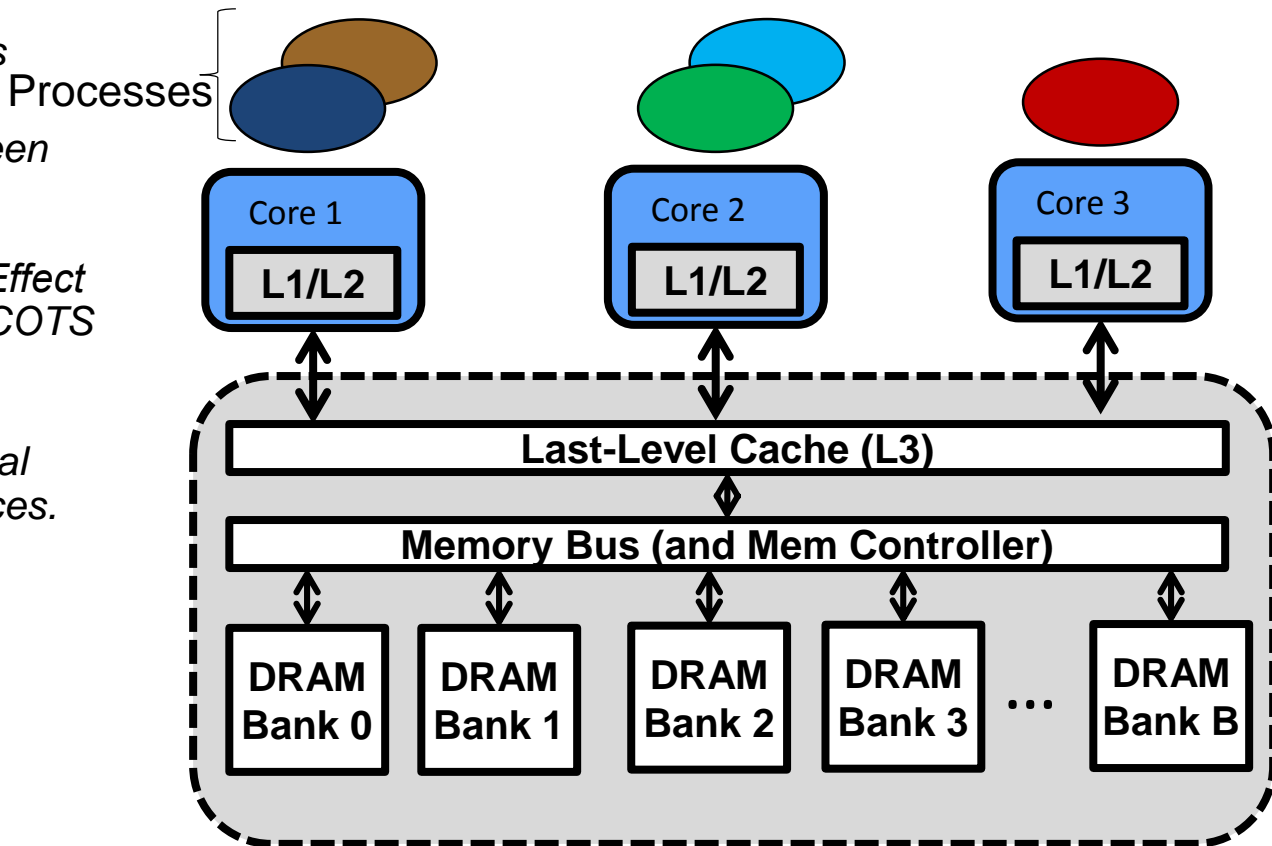


## Issues

- *Shared hardware resources impact timing.*
- *103 times slowdown has been observed*

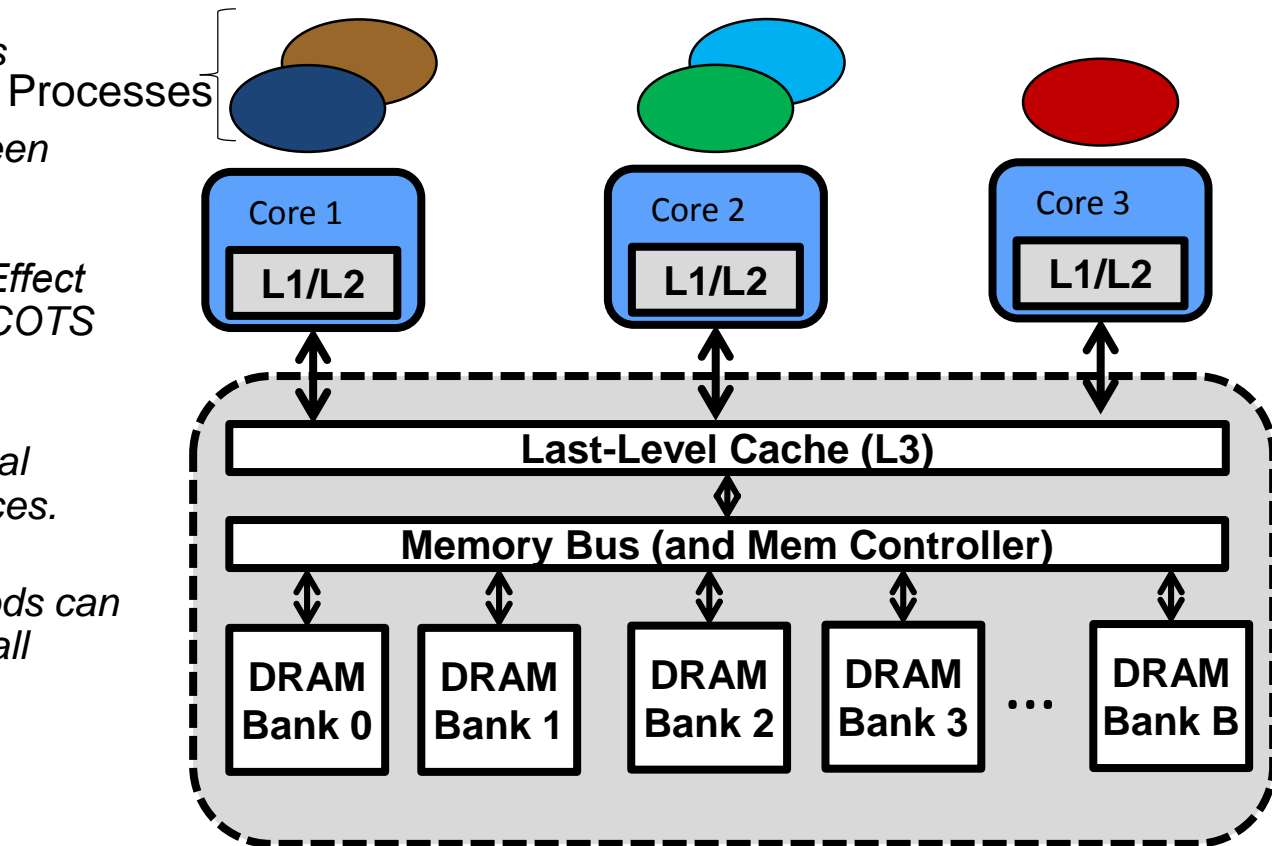
*H. Yun and P. K. Valsan,  
“Evaluating the Isolation Effect  
of Cache Partitioning on COTS  
Multicore Platforms,”  
OSPERT, 2015.*

- *Current methods cannot deal  
with undocumented resources.*



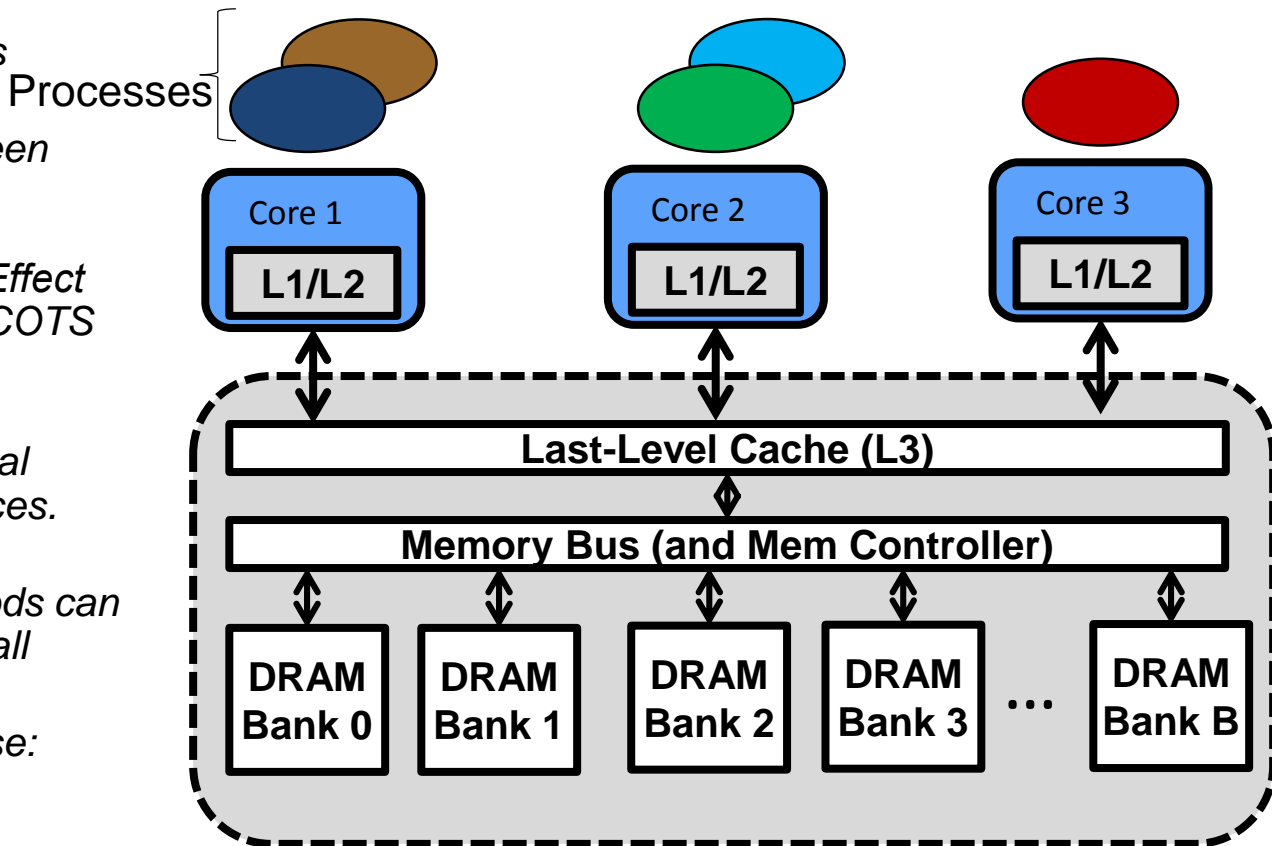
## Issues

- *Shared hardware resources impact timing.*
- *103 times slowdown has been observed*  
*H. Yun and P. K. Valsan,*  
*“Evaluating the Isolation Effect of Cache Partitioning on COTS Multicore Platforms,”*  
*OSPERT, 2015.*
- *Current methods cannot deal with undocumented resources.*
- *Even when resources are documented, current methods can only analyze/manage a small set of them.*



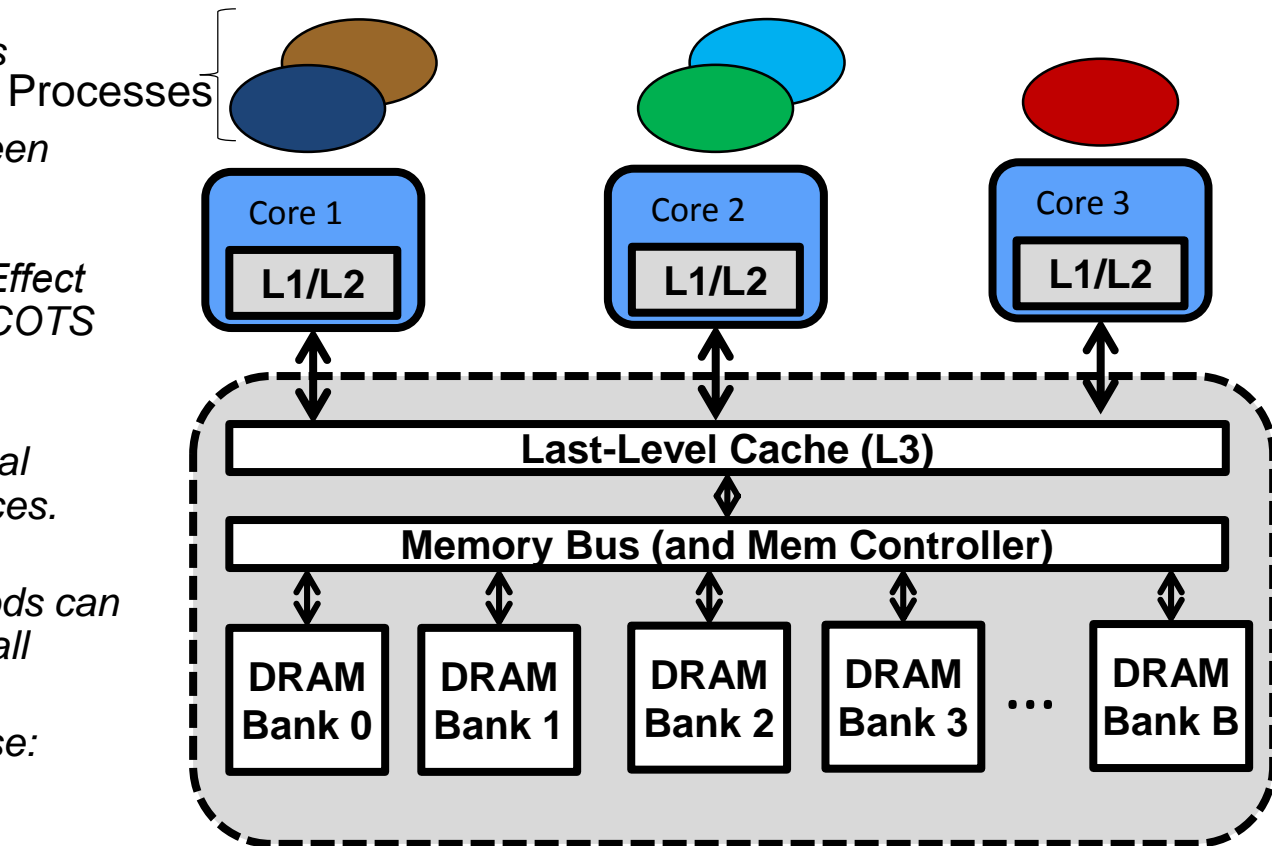
## Issues

- *Shared hardware resources impact timing.*
- *103 times slowdown has been observed*  
*H. Yun and P. K. Valsan,*  
*“Evaluating the Isolation Effect of Cache Partitioning on COTS Multicore Platforms,”*  
*OSPERT, 2015.*
- *Current methods cannot deal with undocumented resources.*
- *Even when resources are documented, current methods can only analyze/manage a small set of them.*
- *The problem is getting worse:*
  - \* *Slowdown increasing*
  - \* *More undocumented h/w*



## Issues

- *Shared hardware resources impact timing.*
- *103 times slowdown has been observed*  
*H. Yun and P. K. Valsan,*  
*“Evaluating the Isolation Effect of Cache Partitioning on COTS Multicore Platforms,”*  
*OSPERT, 2015.*
- *Current methods cannot deal with undocumented resources.*
- *Even when resources are documented, current methods can only analyze/manage a small set of them.*
- *The problem is getting worse:*
  - \* *Slowdown increasing*
  - \* *More undocumented h/w*



We need a new method to compute response times of processes.

# Project Objective

Many real-time systems within the DoD have all processor cores except one disabled in order to be confident about timing.

Therefore, the objective of this project is to develop a solution to overcome this obstacle.

# Thanks!