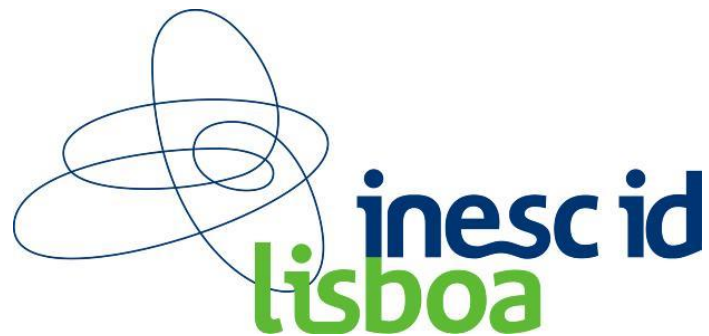
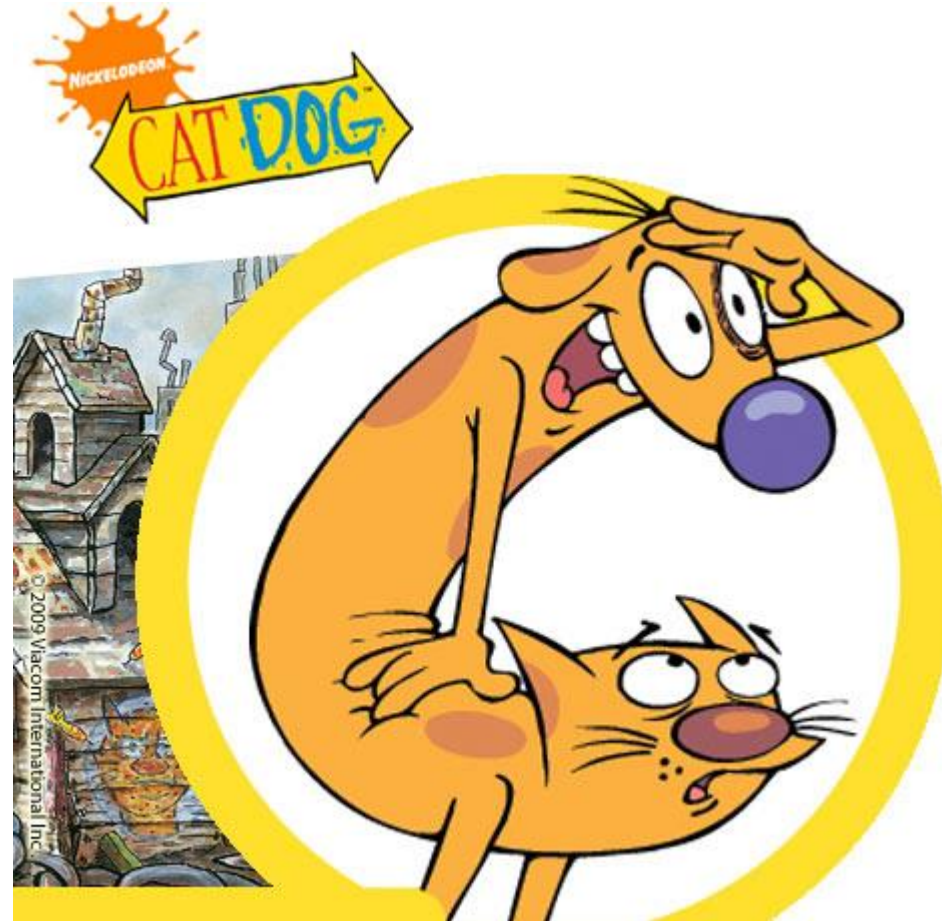


# The hardships of TLSTM

Ricardo Filipe, João Barreto

<http://www.gsd.inesc-id.pt/project-pages/specSTM>



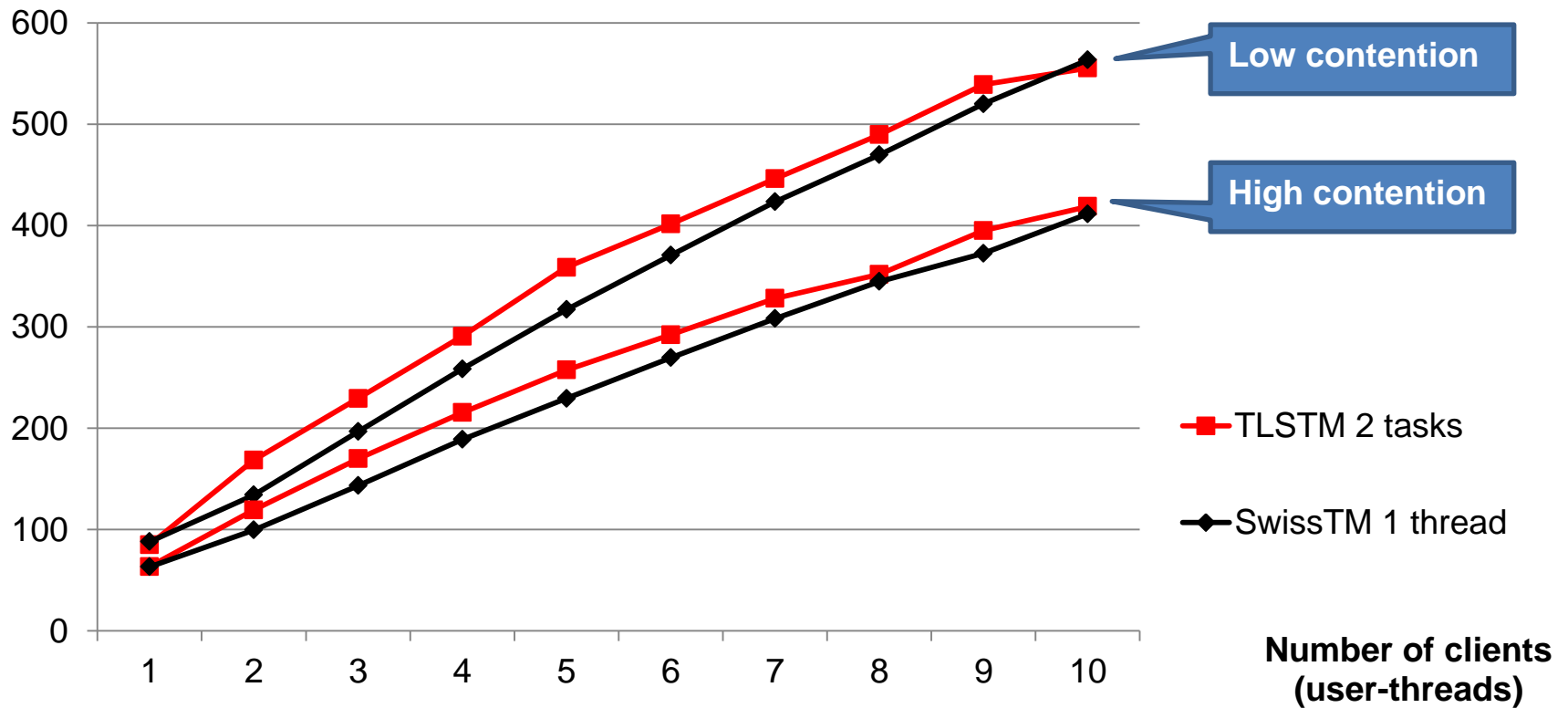


# Unified TLS+STM runtime: Vacation Benchmark results

technology  
from seed



## Throughput operations/ms



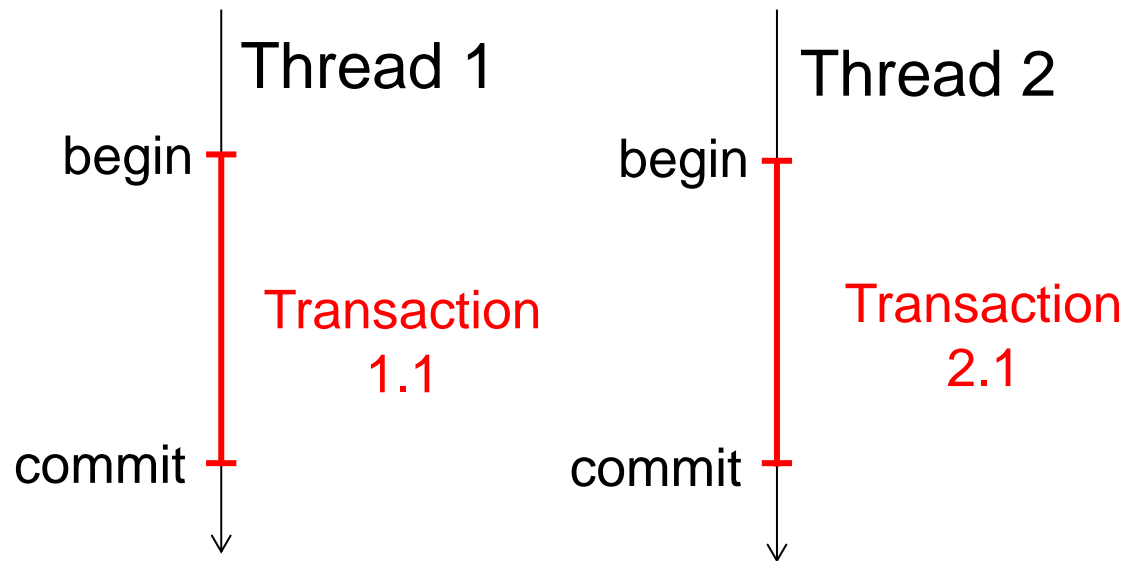
# Two approaches for the same problem

technology  
from seed

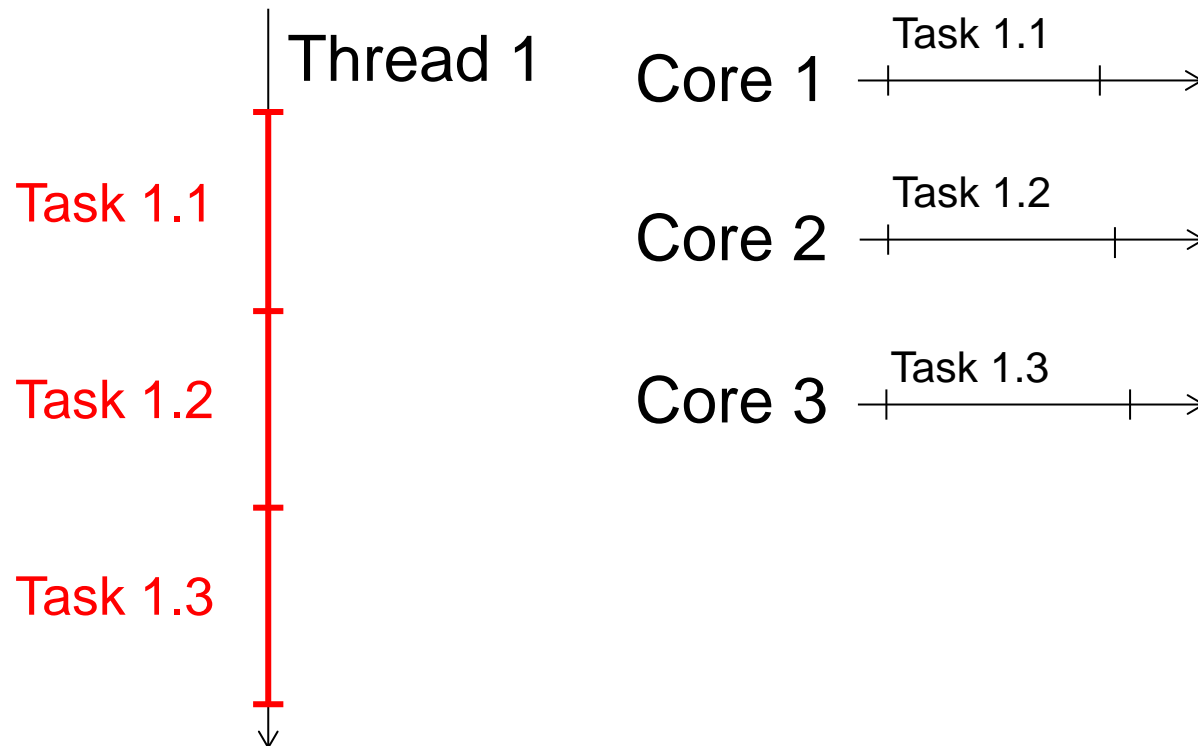


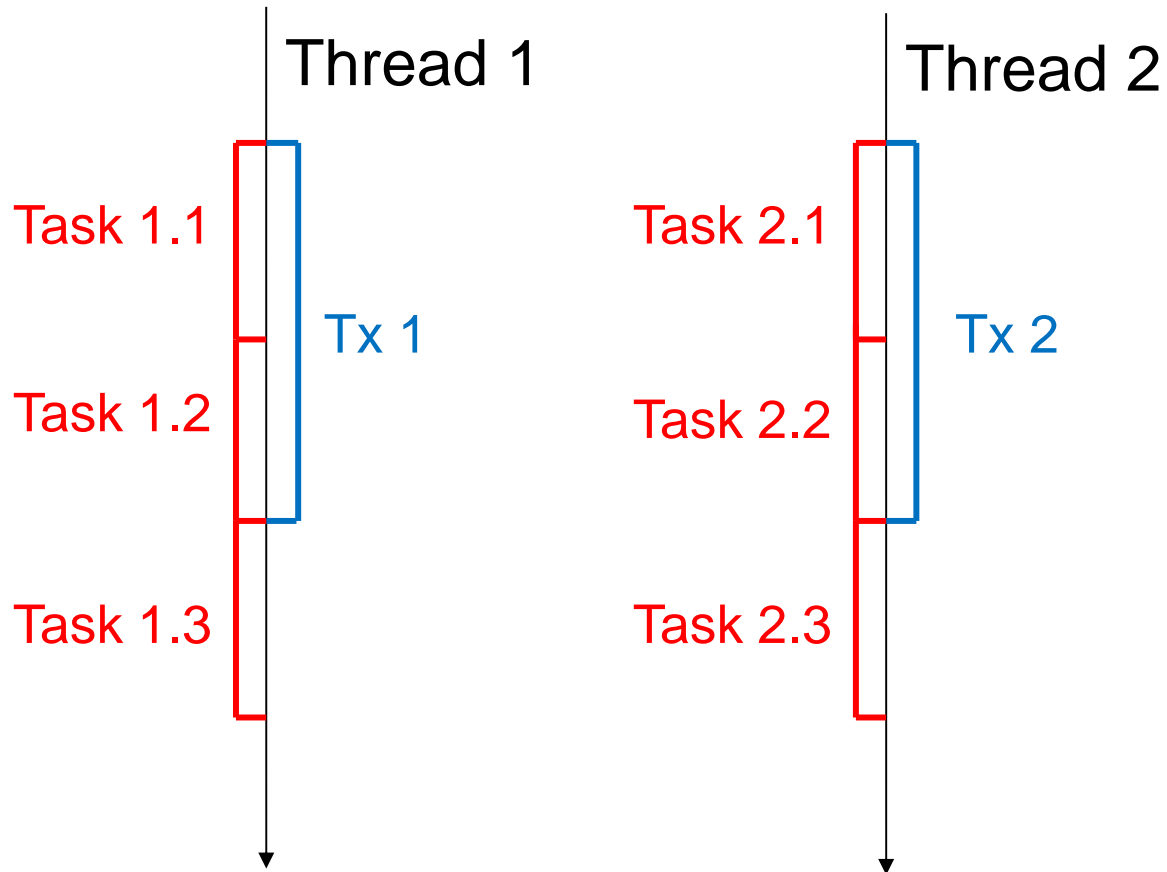
- Increase the level of parallelism of an application

# One approach: Transactional Memory



# Another approach: Thread Level Speculation





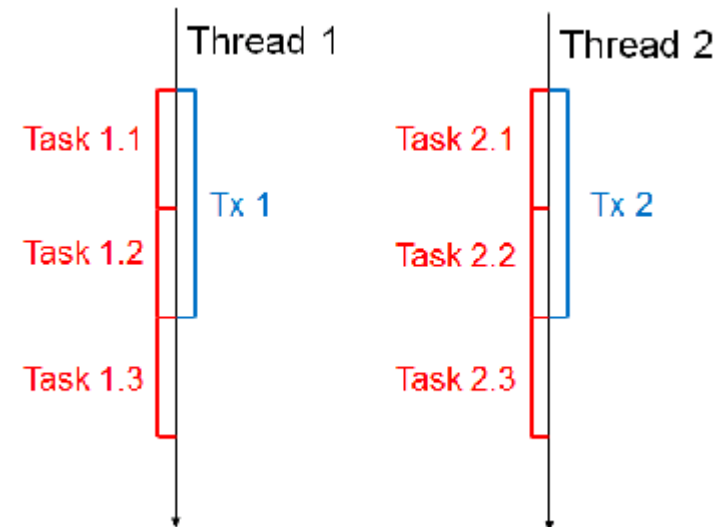
# Sample execution: 4 core processor

Core 1    Task 1.1 begins & Tx 1 begins  
+————→

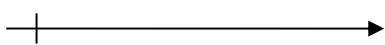
Core 2    Task 1.2 begins  
+————→

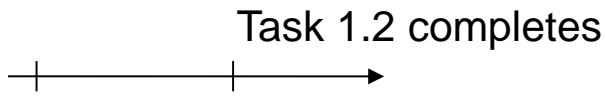
Core 3    Task 2.1 begins & Tx 2 begins  
+————→

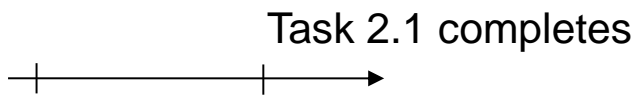
Core 4    Task 2.2 begins  
+————→

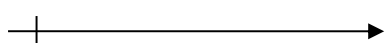


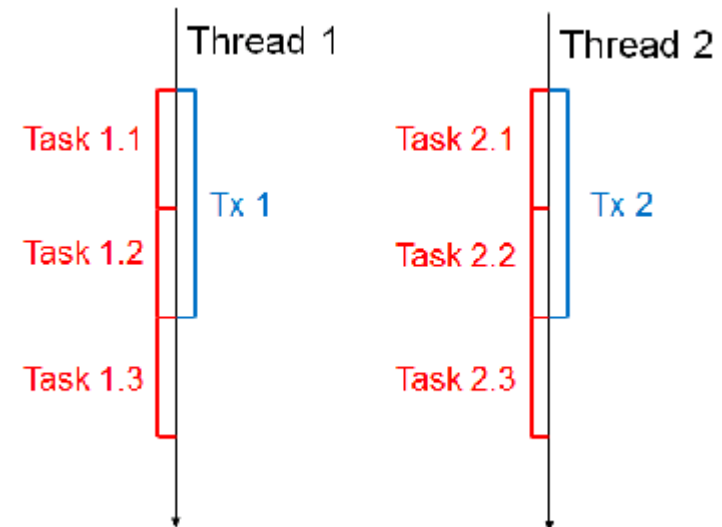


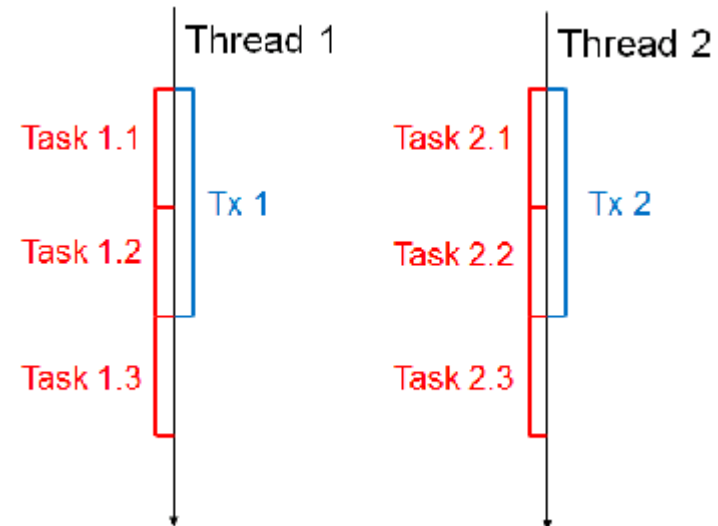
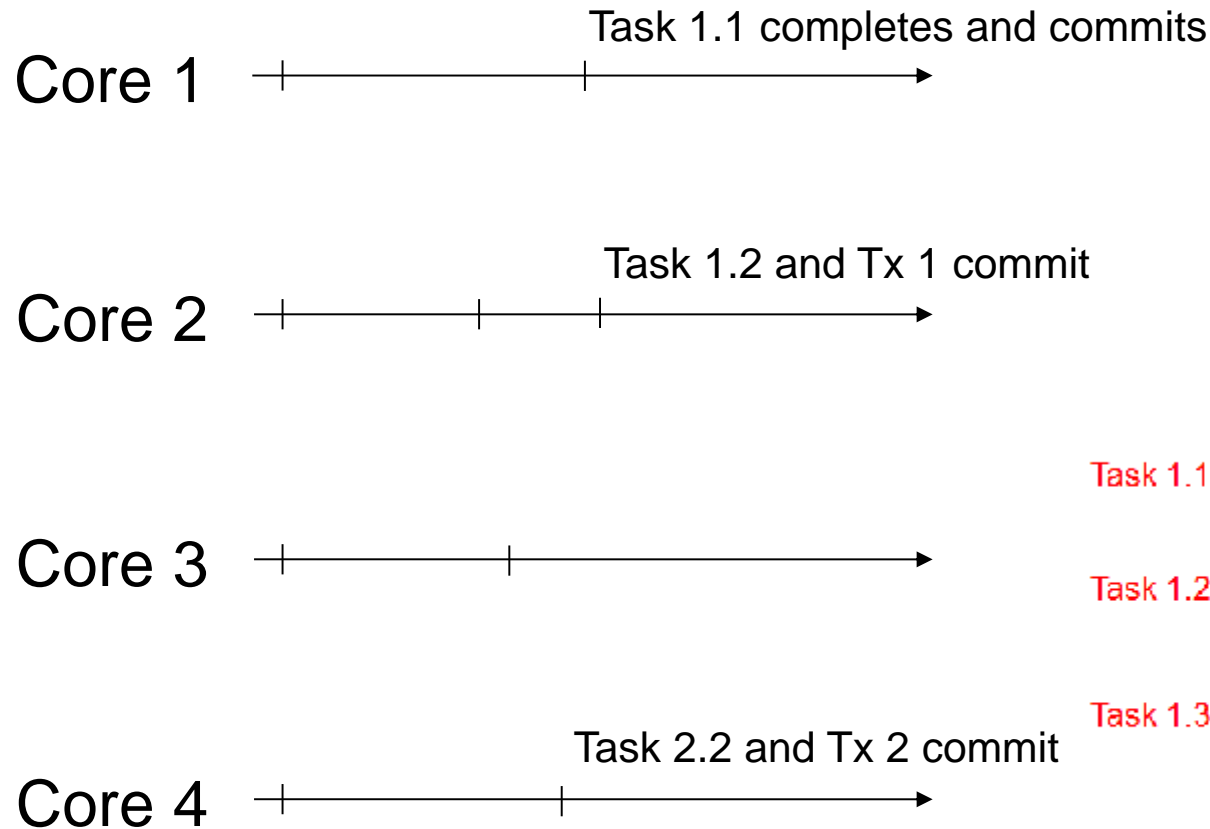
Core 1 

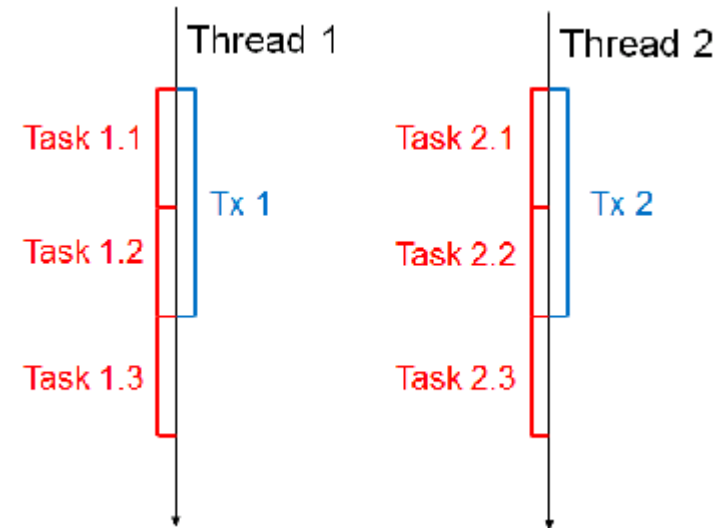
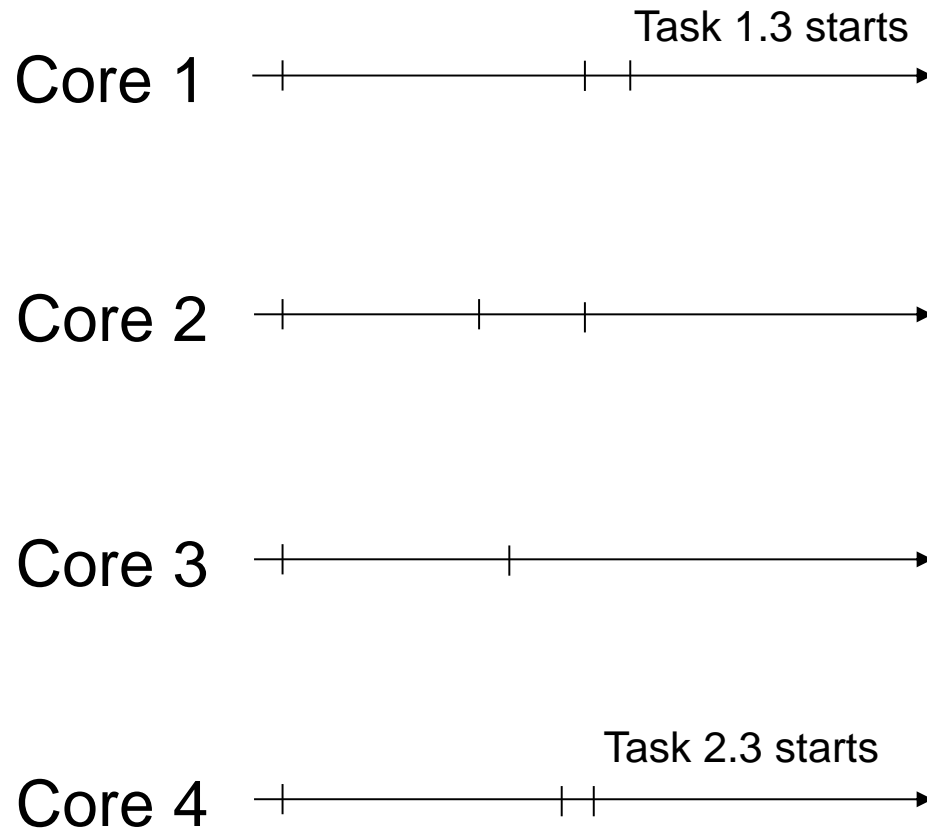
Core 2 

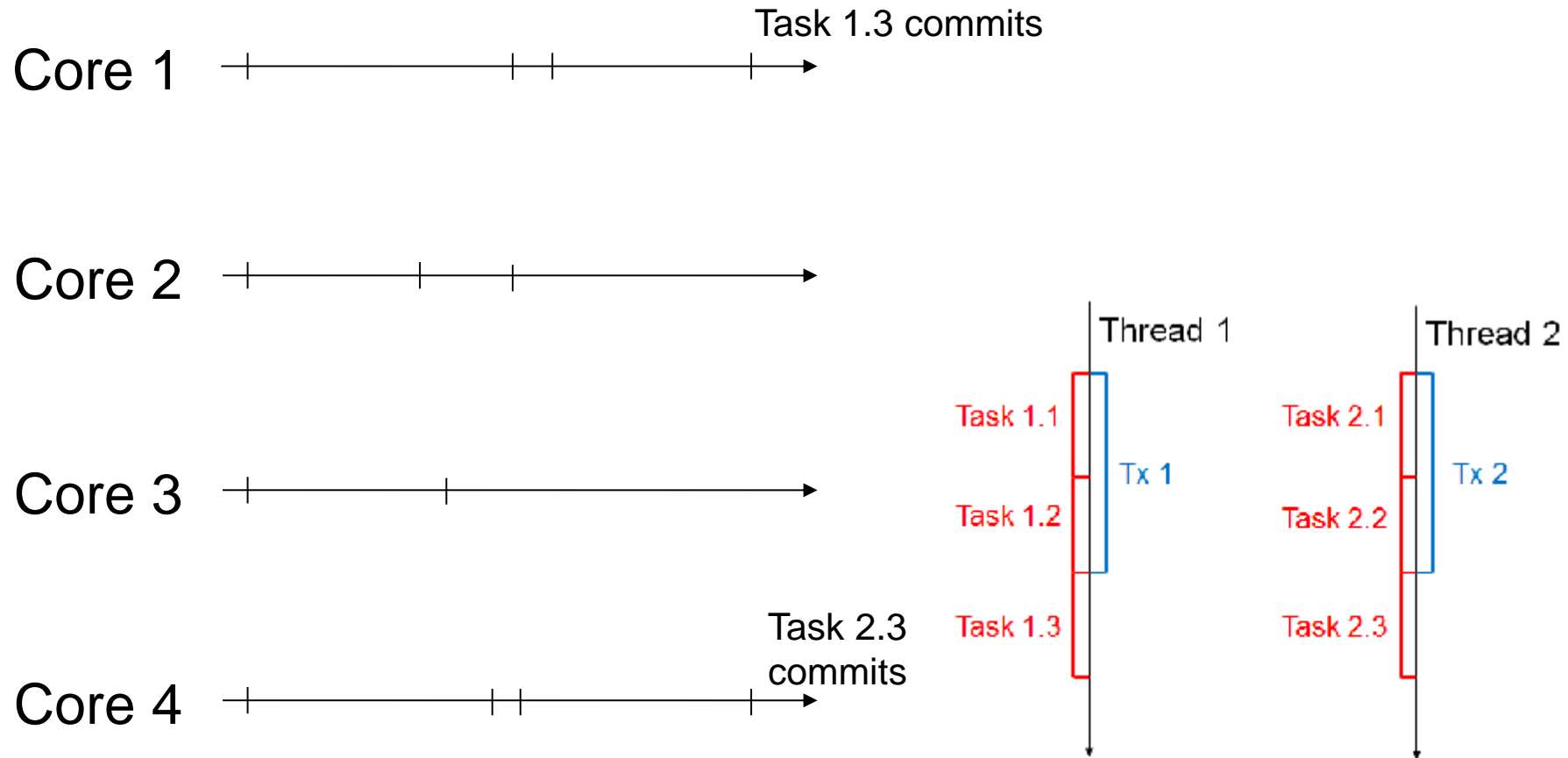
Core 3 

Core 4 

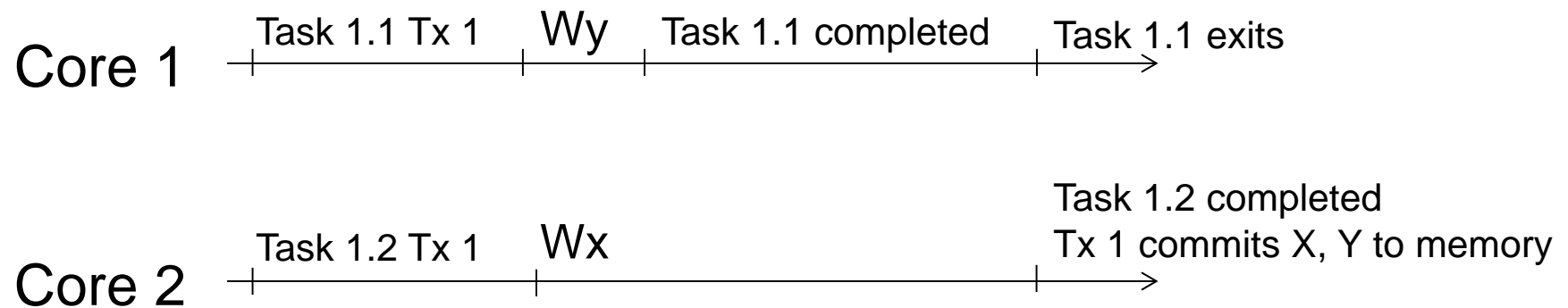






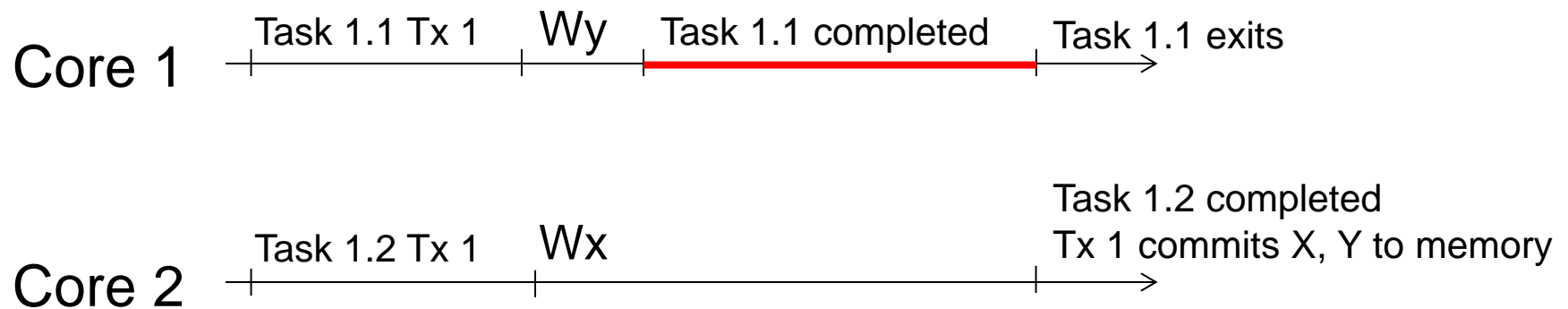


- Leveraging SwissTM with TLS features
  - Transaction commit takes place in the last task of the transaction, after all of its tasks have completed



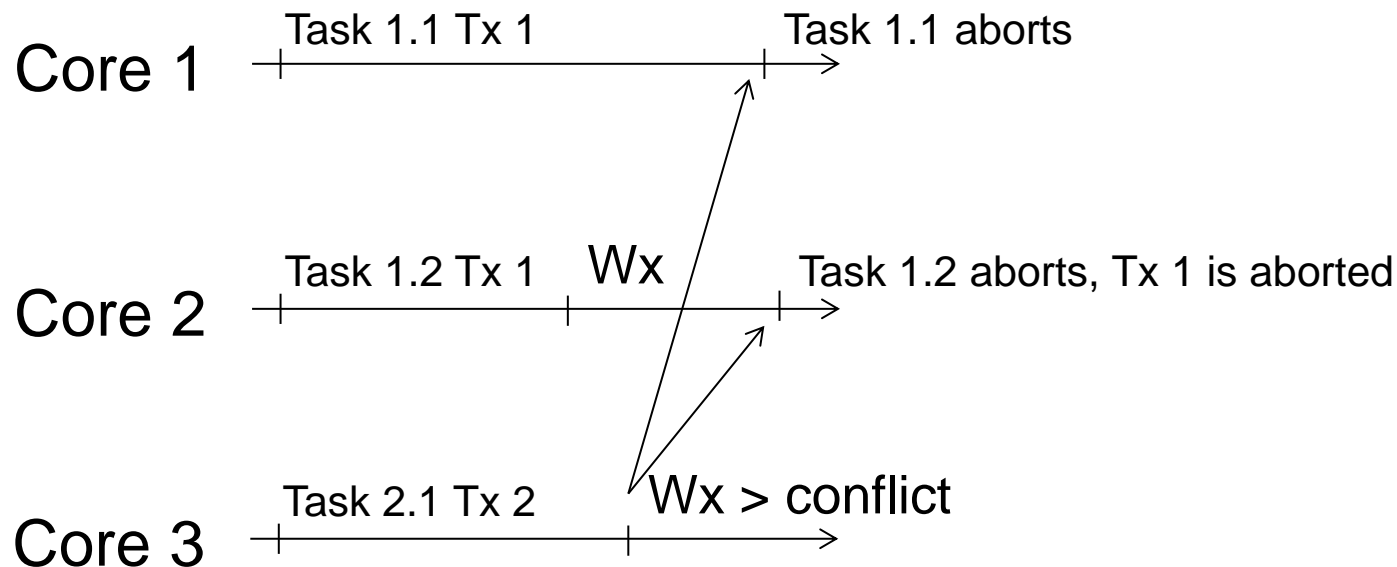
## •Leveraging SwissTM with TLS features

- Transaction commit takes place in the last task of the transaction, after all of its tasks have completed
- Core 1 could be working on another task/transaction, but it is busy waiting until it can exit the current task



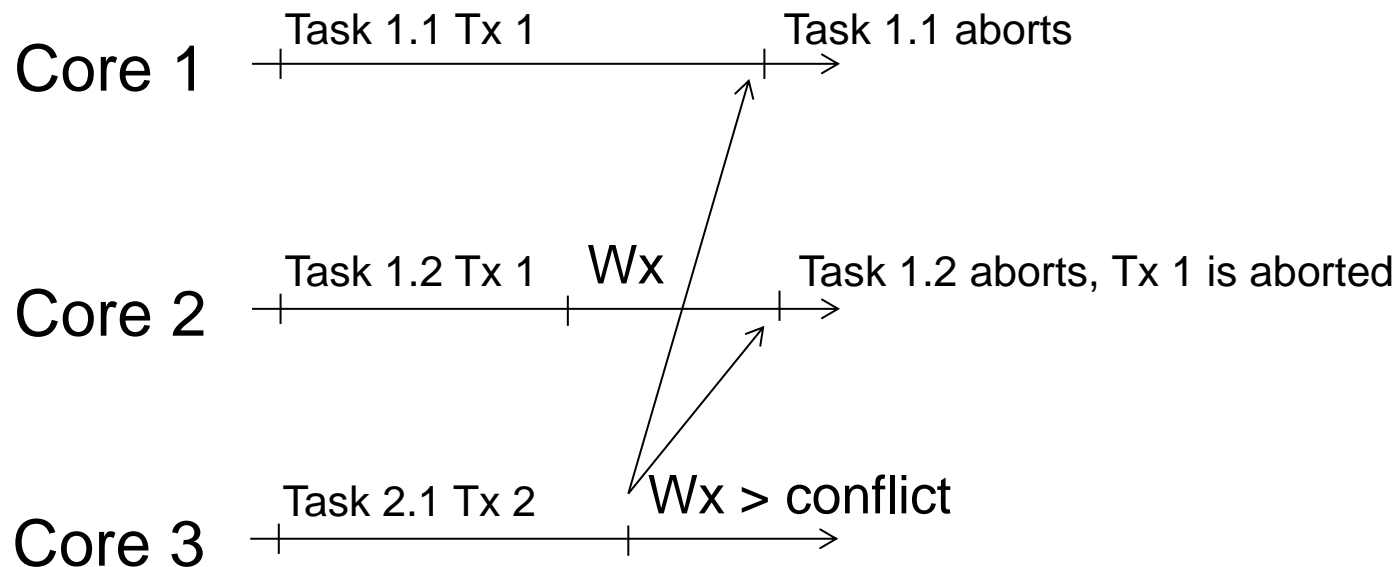
# The hardships of TLSTM: Atomic operations on several tasks

- Leveraging SwissTM with TLS features
  - Transaction abort synchronizes all of its tasks to abort atomically



# The hardships of TLSTM: Atomic operations on several tasks

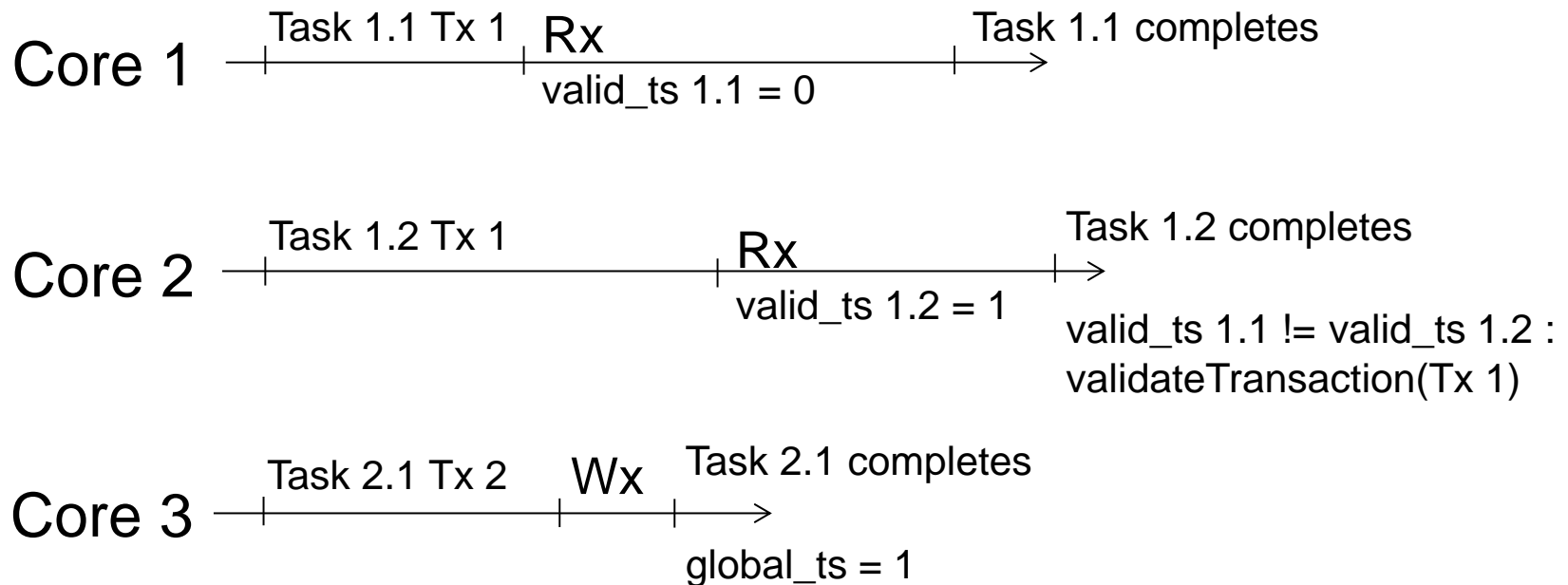
- Leveraging SwissTM with TLS features
  - Transaction abort synchronizes all of its tasks to abort atomically
  - Not all tasks of the aborting transaction need to abort





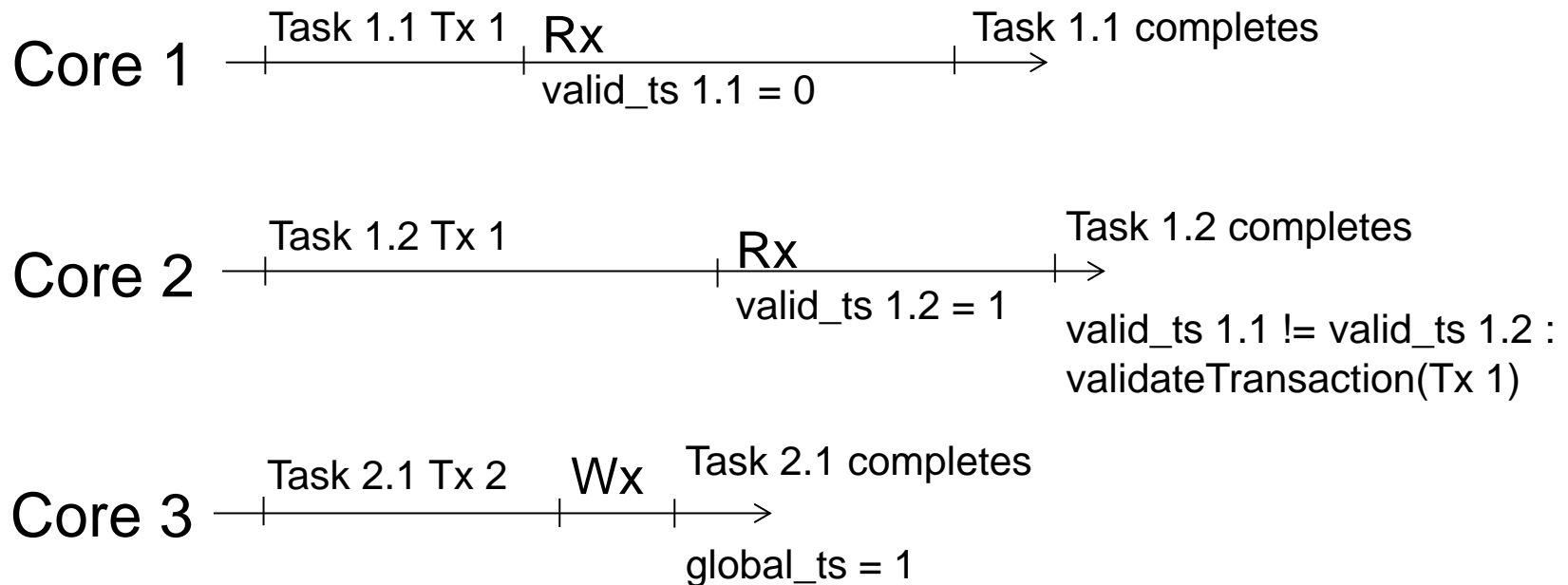
# The hardships of TLSTM: Atomic operations on several tasks

- Leveraging SwissTM with TLS features
  - Transaction validation is done separately by each task



# The hardships of TLSTM: Atomic operations on several tasks

- Leveraging SwissTM with TLS features
  - Transaction validation is done separately by each task
  - Transaction validation was not needed in SwissTM at commit time



- There are many open problems in TM+TLS
- We have identified 3 major problems with TLSTM:
- Cores become idle too much time
- Transaction abort requires that all tasks of the transaction abort
- Transaction validation is not done atomically

# Questions?

**technology  
from seed**



<http://www.gsd.inesc-id.pt/project-pages/specSTM>