Analyzing the Impact of Data Prefetching on Chip MultiProcessors

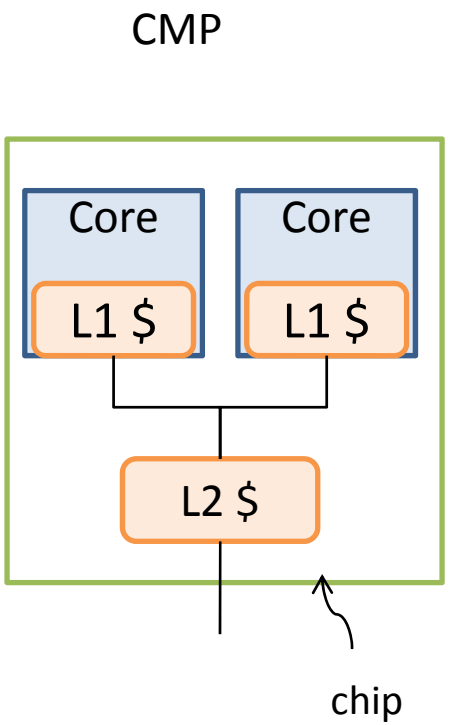
Naoto Fukumoto, Tomonobu Mihara,
Koji Inoue, Kazuaki Murakami

Kyushu University, Japan

ACSAC 13
August 4, 2008

Back Ground

- CMP (Chip MultiProcessor):
  - Several processor cores integrated in a chip
  - High performance by parallel processing
  - New feature: Cache-to-cache data transfer
- Limitation factor of CMP performance
  - Memory-wall problem is more critical
    - High frequency of off-chip accesses
    - Not scaling bandwidth with the number of cores



Data prefetching is more important in CMPs

Motivation & Goal

- Motivation
  - Conventional prefetch techniques have been developed for uniprocessors
  - Not clear that these prefetch techniques achieve high performance in even in CMPs
  - Is it necessary for the prefetch techniques to consider CMP features ?
  - Need to know the effect of prefetch on CMPs
- Goal
  - Analysis of the prefetch effect on CMPs

Outline

- Introduction
- Prefetch Taxonomy for multiprocessors
- Extension for CMPs
- Quantitative Analysis
- Conclusions

Classification of Prefetches According to Impact on Memory Performance

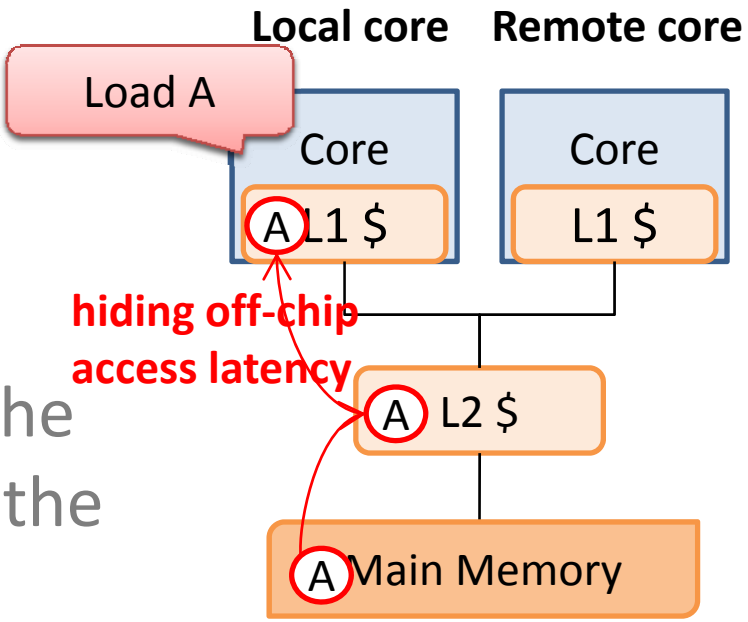
- Focusing on each prefetch
- Definition of the prefetch states
  - **Initial state**: the state just after a block is prefetched into cache memory
  - **Final State**: the state when the block is evicted from cache memory
  - The state transits based on ***Events*** during the life time of the prefetched block in cache memory

Definition of ***Events***

Event1. The prefetched block is accessed by the local core

Event2. The local core accesses the block which has evicted from the cache by the prefetch

Event3. The prefetch causes a downgrade followed by a subsequent upgrade in a remote core

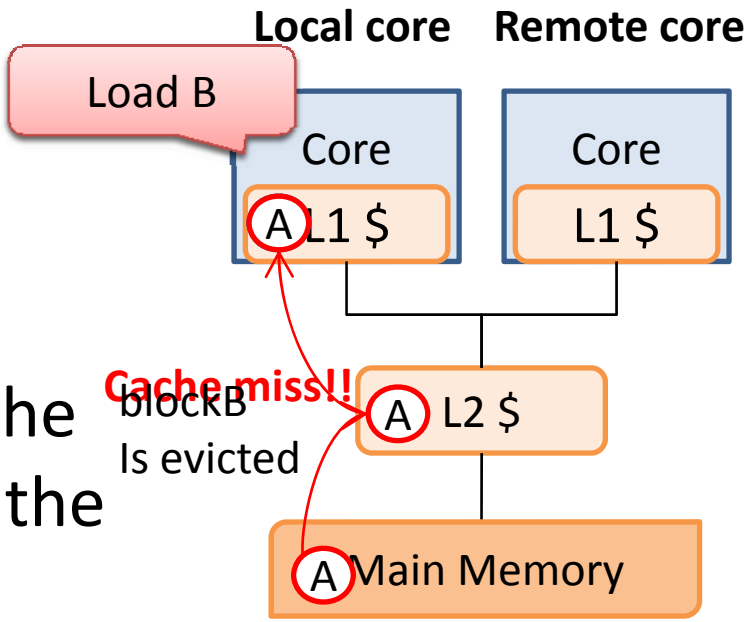


Definition of ***Events***

Event1. The prefetched block is accessed by the local core

Event2. The local core accesses the block which has evicted from the cache by the prefetch

Event3. The prefetch causes a downgrade followed by a subsequent upgrade in a remote core

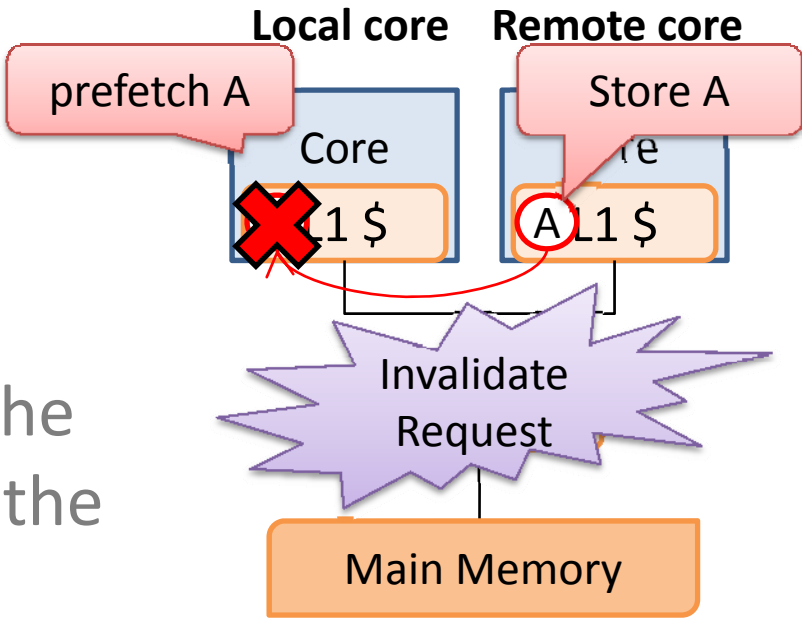


Definition of ***Events***

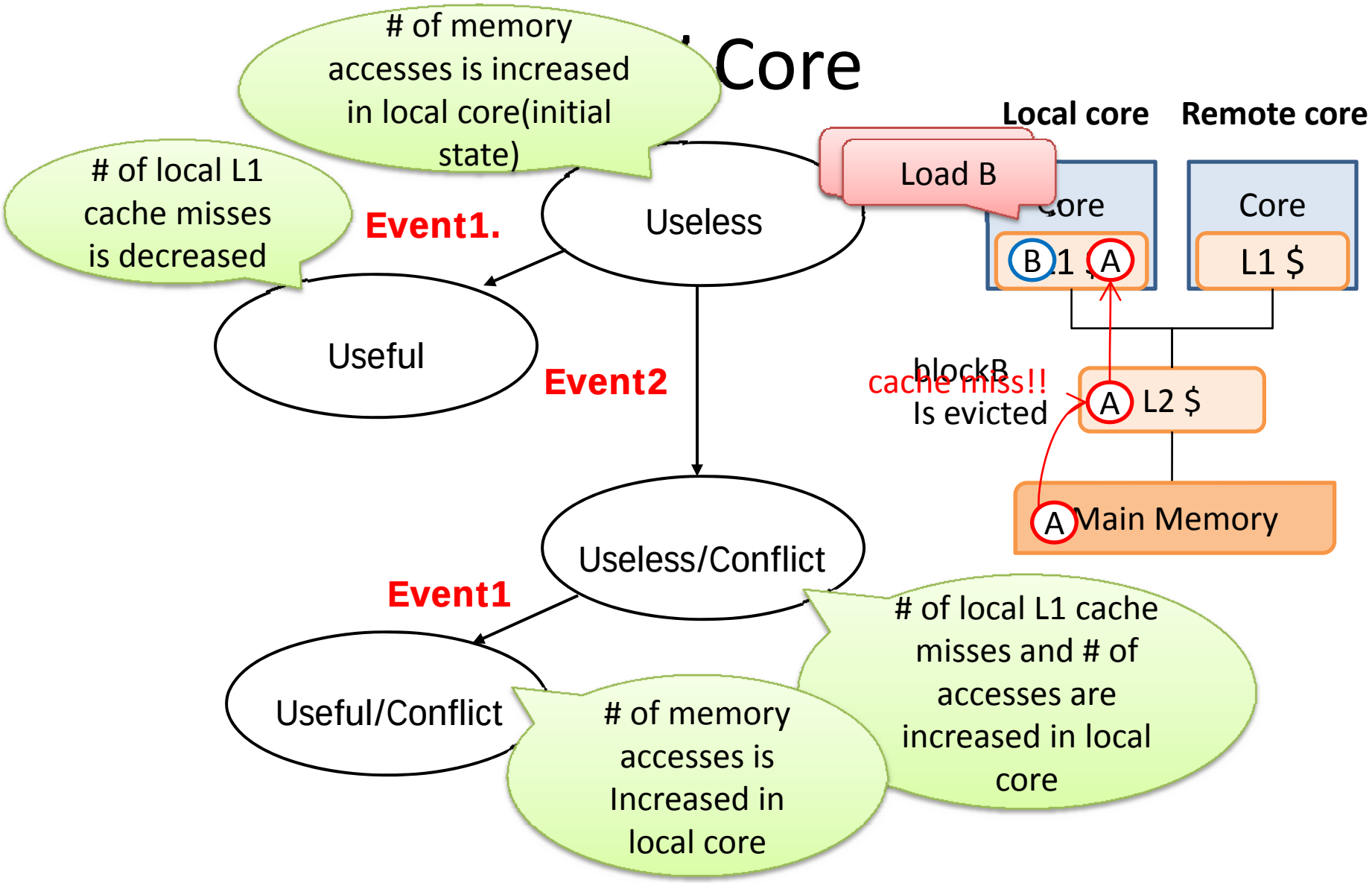
Event1. The prefetched block is accessed by the local core

Event2. The local core accesses the block which has evicted from the cache by the prefetch

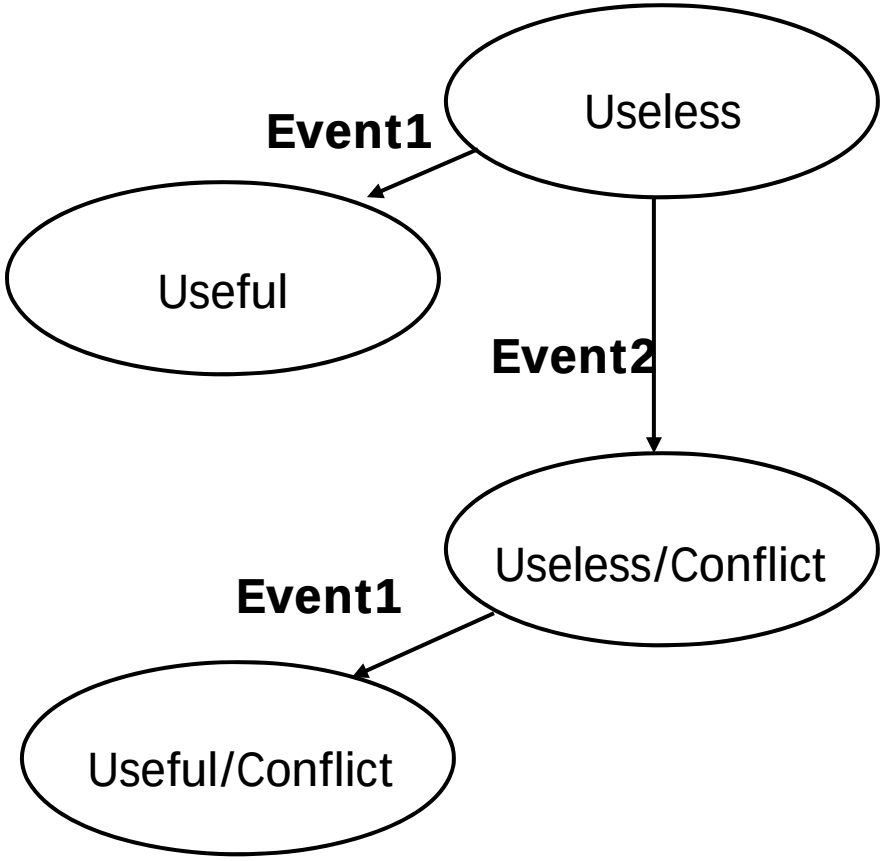
Event3. The prefetch causes a downgrade followed by a subsequent upgrade in a remote core



The State Transition of Prefetch in Core

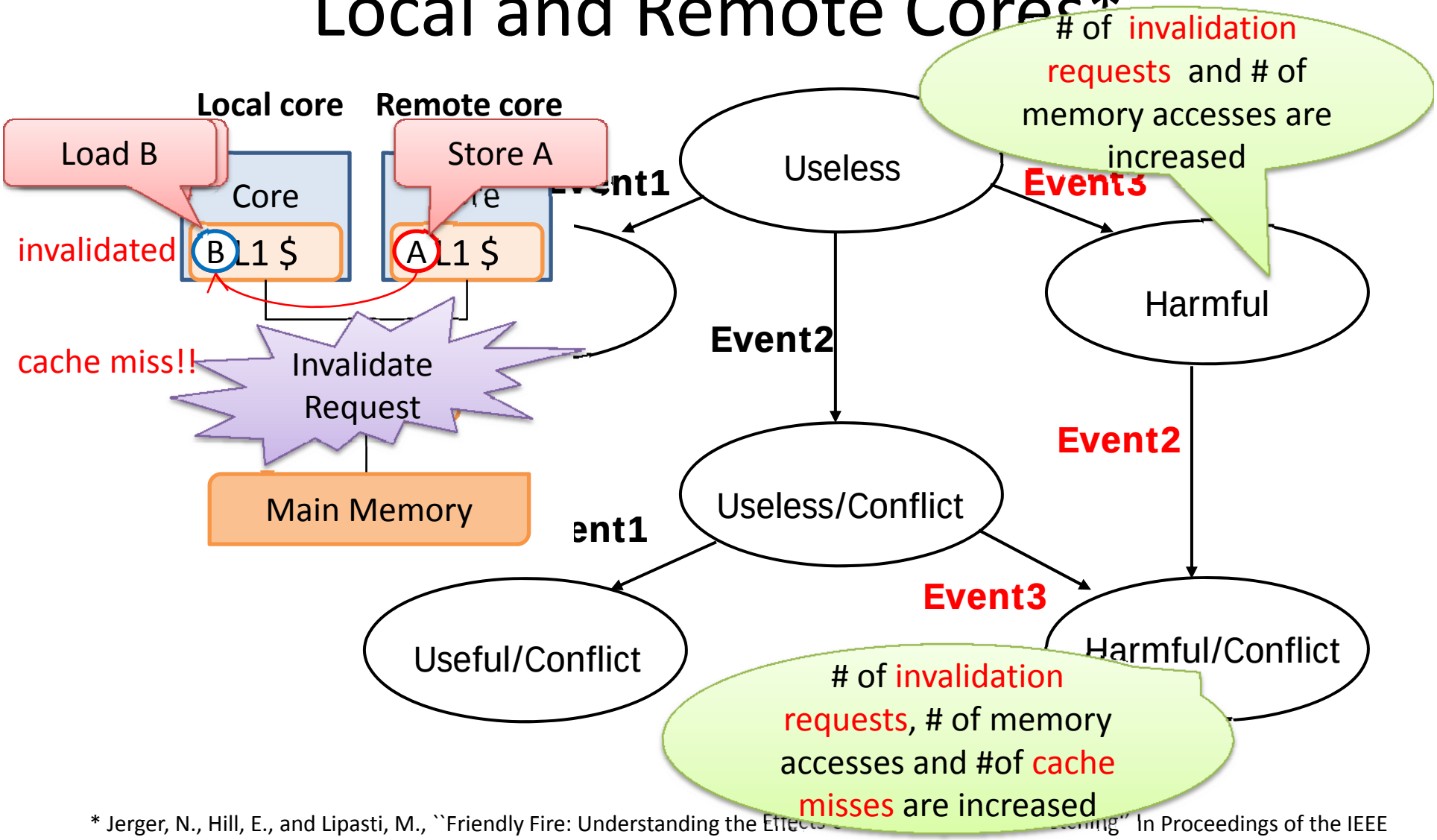


The State Transition of Prefetch in Local and Remote Cores*



* Jerger, N., Hill, E., and Lipasti, M., ``Friendly Fire: Understanding the Effects of Multiprocessor Prefetching'' In Proceedings of the IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS), 2006.

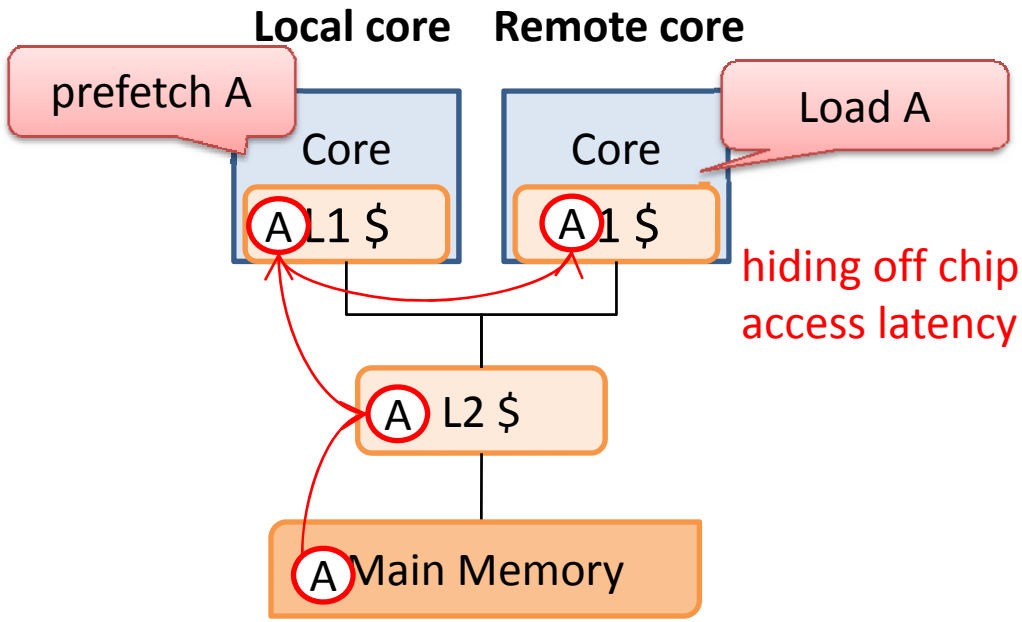
The State Transition of Prefetch in Local and Remote Cores*



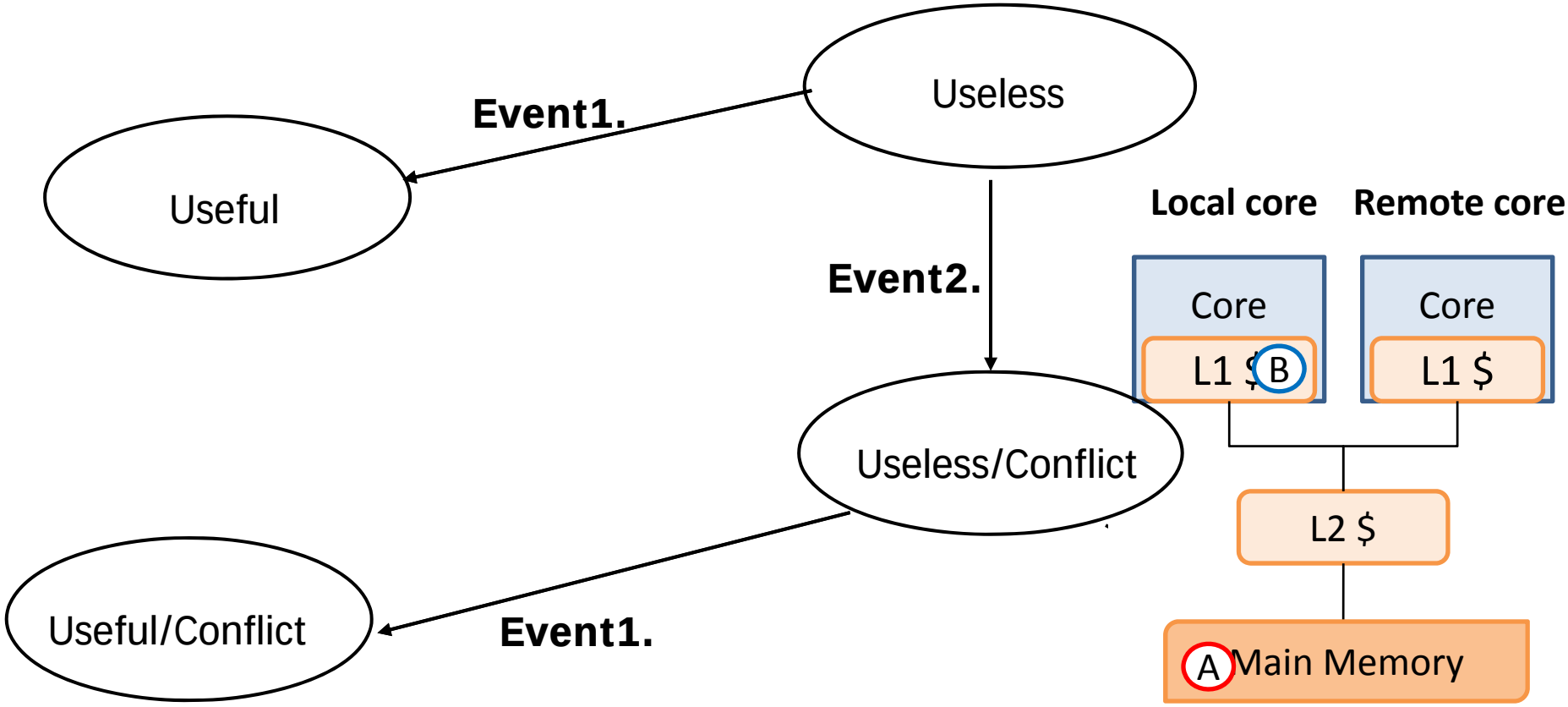
* Jerger, N., Hill, E., and Lipasti, M., ``Friendly Fire: Understanding the Effects ... ning'' In Proceedings of the IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS), 2006.

Considering Cache-to-Cache Data Transfer

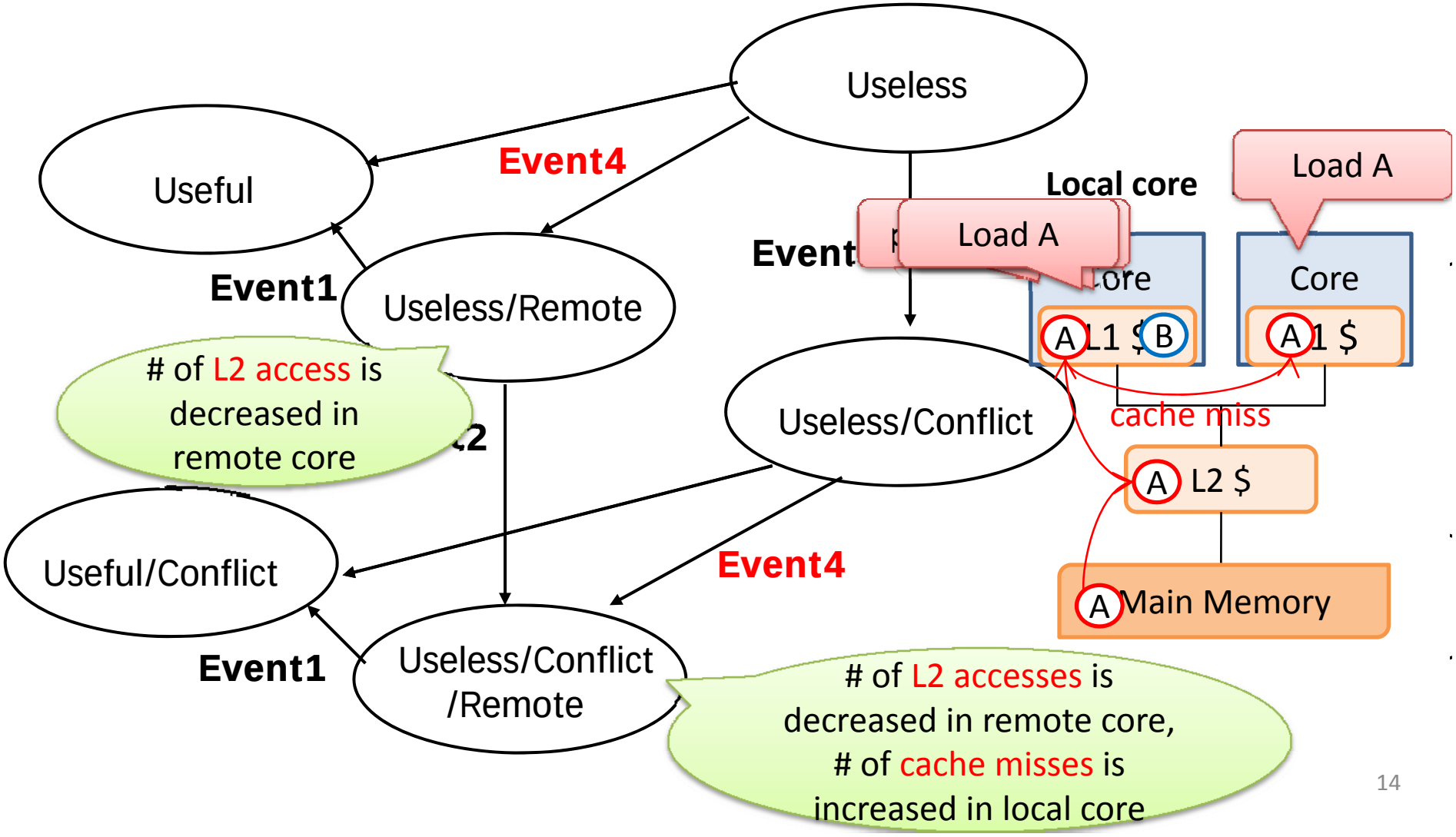
- ***Event4.*** The prefetched block loaded from L2 or main memory is accessed by a remote core



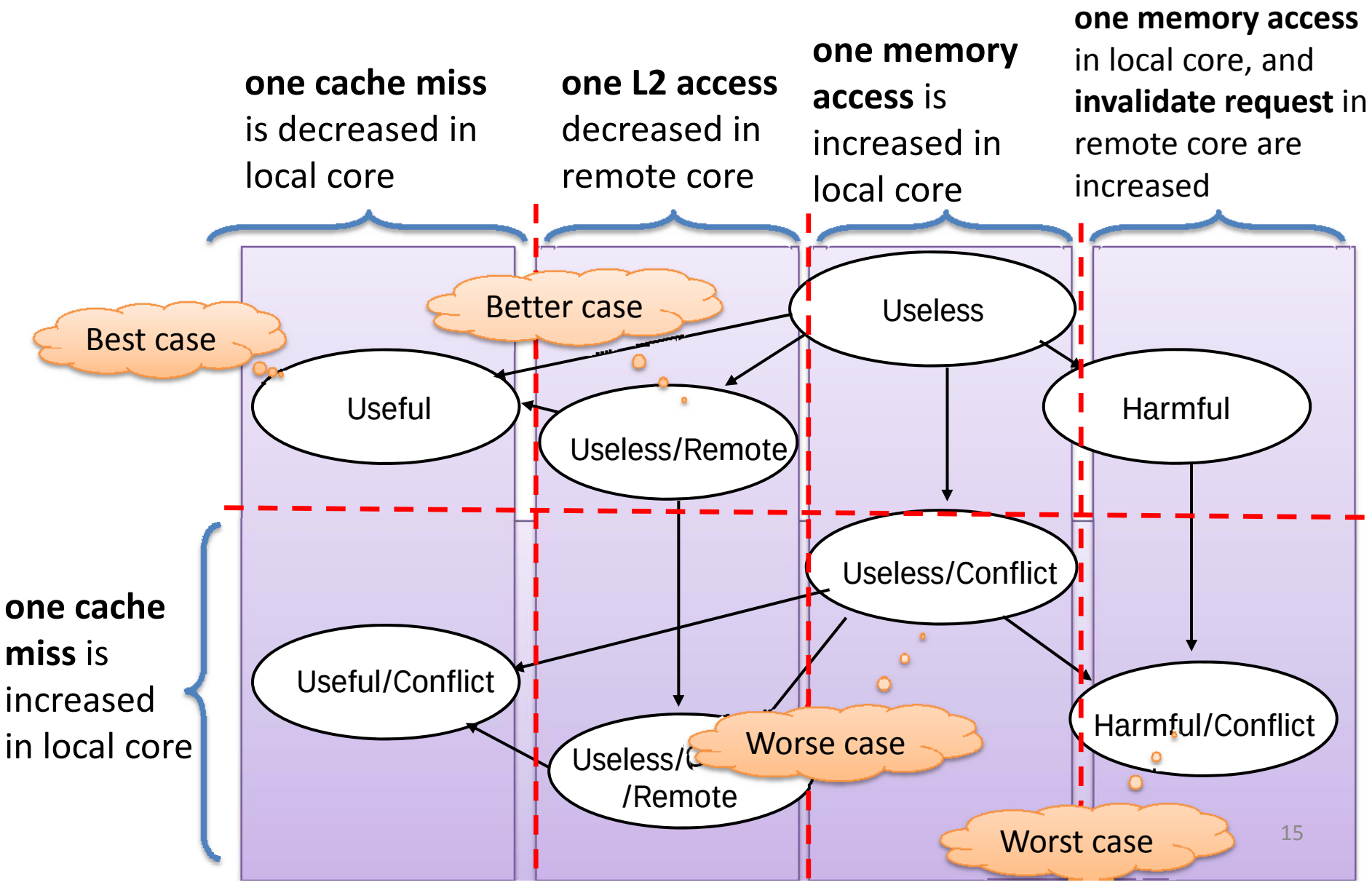
The State Transition in CMPs



The State Transition in CMPs



Classification of Prefetches in CMPs

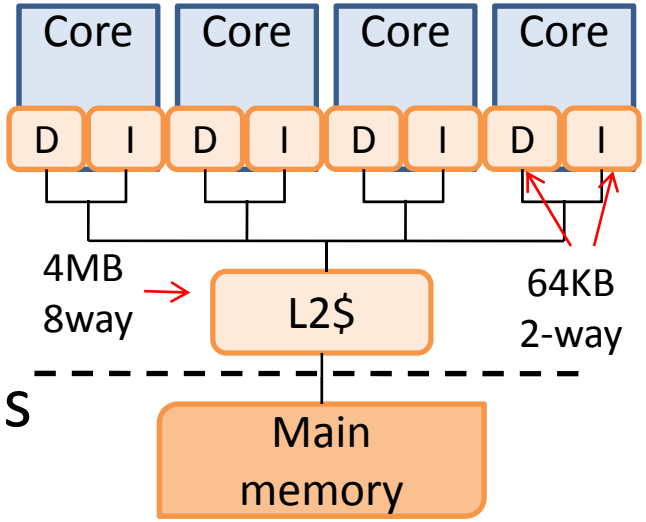


Outline

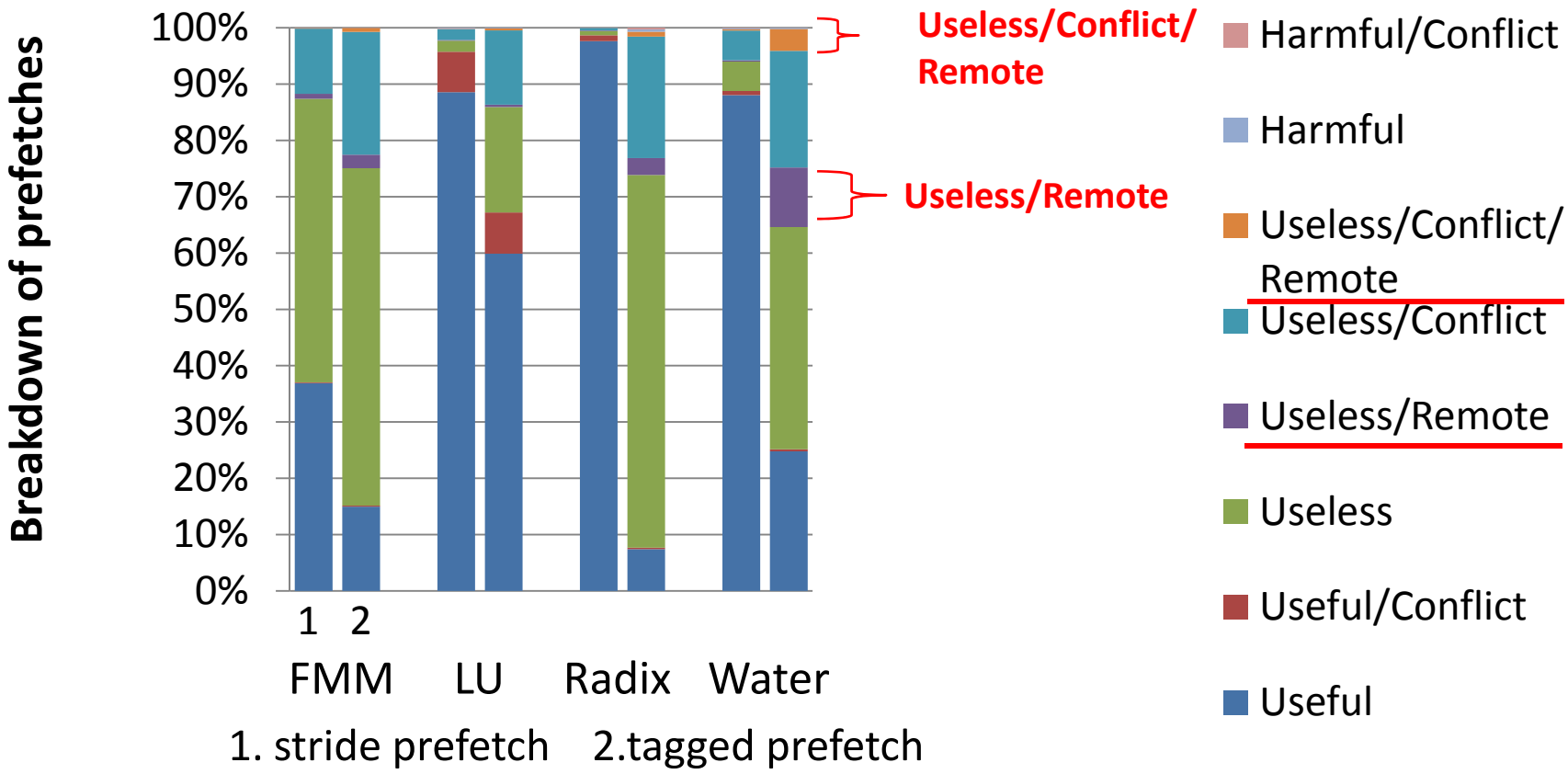
- Introduction
- Prefetch Taxonomy
  - for Multiprocessors
  - for CMP
- Quantitative Analysis
- Conclusions

Simulation Environment

- Simulator
  - M5: CMP simulator
    - Prefetch mechanism attached on L1 cache
    - Stride prefetch and tagged prefetch
    - MOESI coherence protocol
- Benchmark programs
  - SPLASH-2:
    Scientific computation programs



Can Conventional Prefetch Techniques Exploit Cache-to-Cache data transfer ?

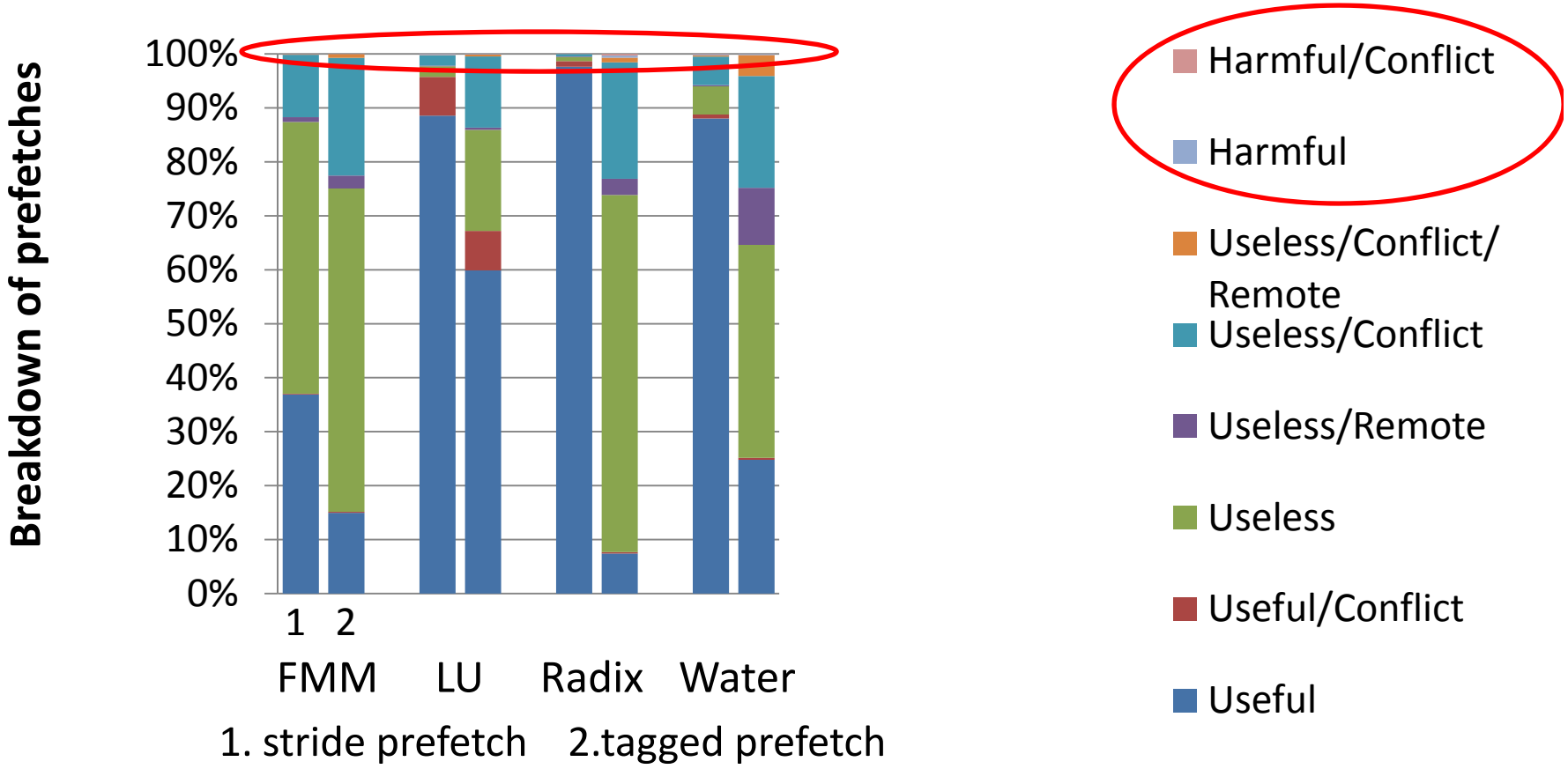


1. stride prefetch 2.tagged prefetch

•The percentage of Useless/Remote and Useless/Conflict/Remote prefetches is only 5%

➔ Conventional prefetch techniques do not exploit cache-to-cache data transfer effectively

Are the Prefetched-Block Invalidations Serious Problem for CMPs?



- Prefetches of Harmful and Harmful/Conflict are extremely few (average 0.2%)

➔ Invalidations of prefetched blocks are negligible

Multiprocessor vs. Chip Multiprocessor

- Harmful and Harmful/Conflict prefetches
  - 0.01~0.70% in CMP (tagged prefetch)
  - ➔ Small negative impact
  - 2~18% in MP* (sequential prefetch)
  - ➔ Large negative impact

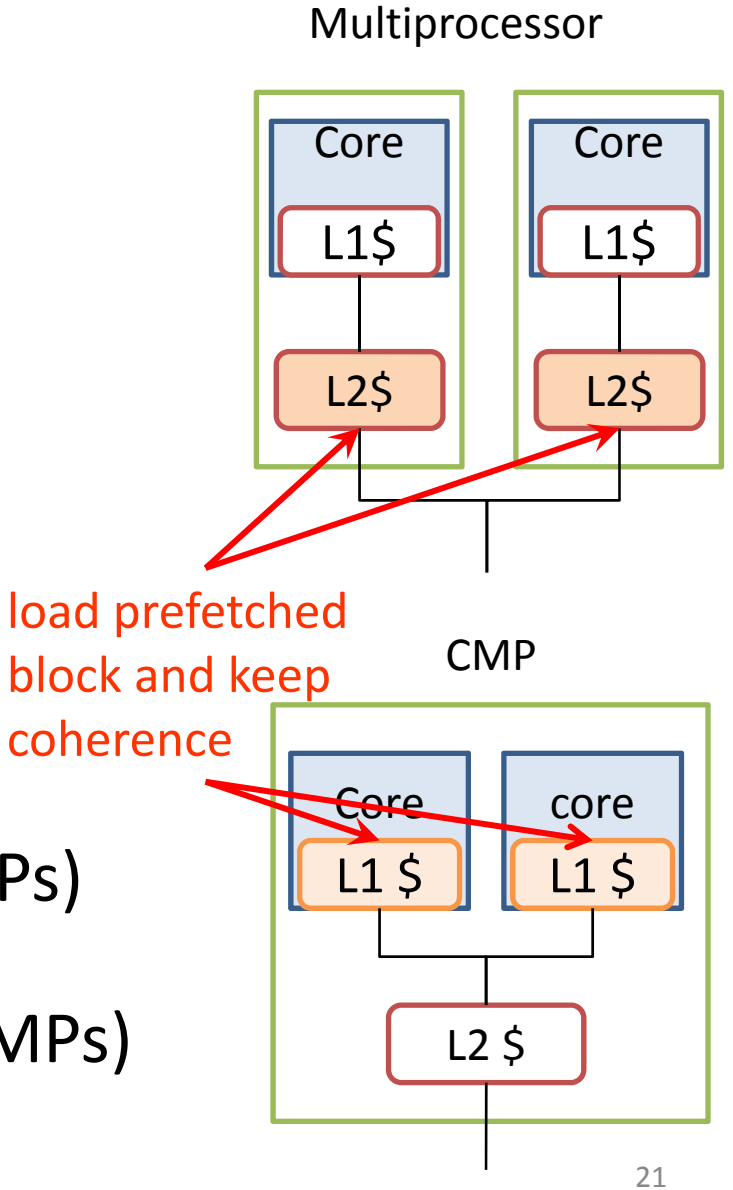
- Why does this difference occur ?

*Jerger, N., Hill, E., and Lipasti, M., Friendly Fire: Understanding the Effects of Multiprocessor Prefetching. In Proceedings of the IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS), 2006.

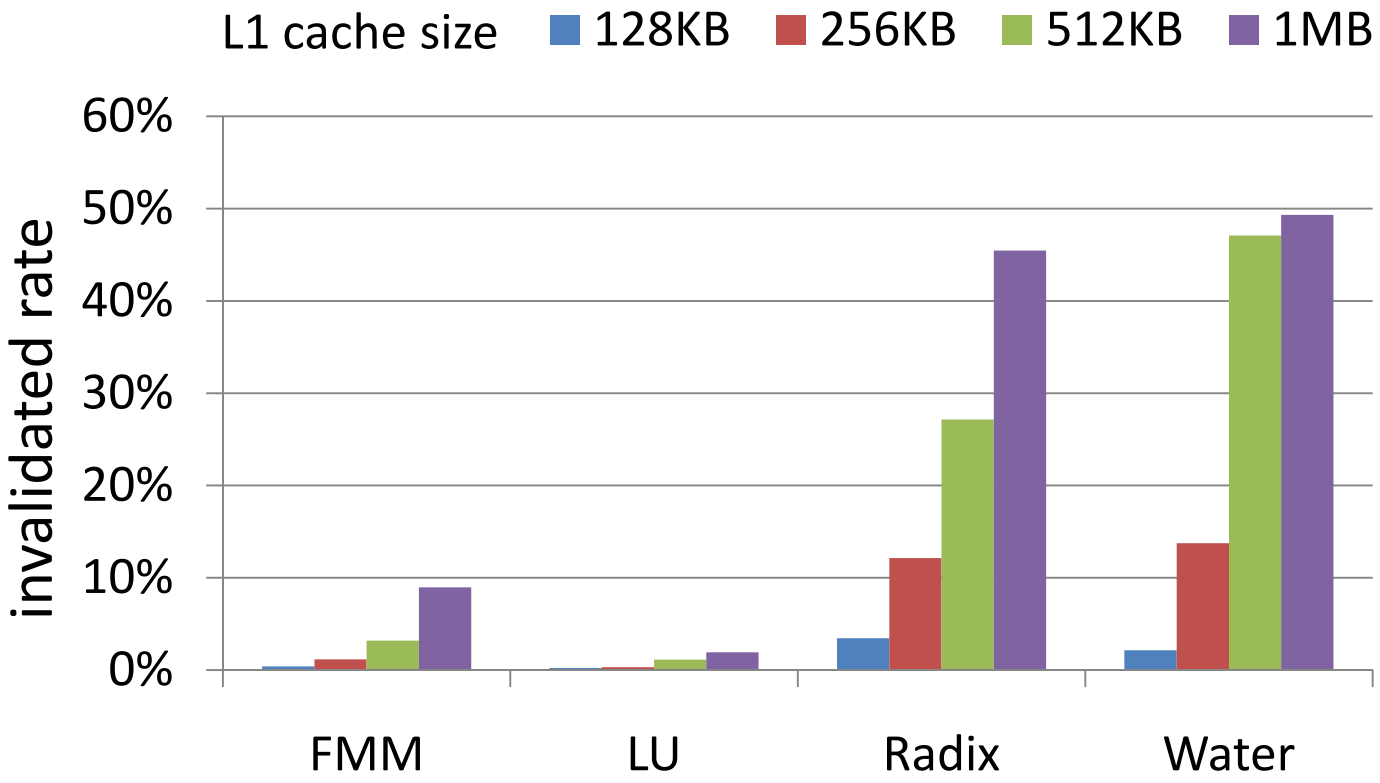
The Reason of Difference of Invalidation Rate

- Difference of the life time of prefetched blocks in cache
  - Long life time (large cache size)
    - ➔ High possibility of invalidation
  - Short life time (small cache size)
    - ➔ Low possibility of invalidation

- If the cache size is large, the negative impact is large(like MPs)
- If the cache size is small, the negative impact is small (like CMPs)



The Invalidation Rate of Prefetched Blocks with Varying L1 Cache Size (tagged prefetch)



Larger cache ➔ large negative impact (like MPs)
Smaller cache ➔ small negative impact (like CMPs)

Summary

- Contributions
  - New method to analyze prefetch effects on CMPs
  - Quantitative analysis for two types of prefetches
- Observations
  - Conventional prefetch techniques DO NOT exploit cache-to-cache data transfer effectively
  - Harmful prefetches are NOT *harmful* in CMPs
- Future work
  - Propose novel prefetch technique exploiting the features of CMPs

Thank you

Any Questions ?

~Please speak slowly~