Влияние КЭШа на скорость работы программы

Анна Субботина

14 Сентября 2016

CACHE

- промежуточный буфер с быстрым доступом, содержащий информацию, которая может быть запрошена с наибольшей вероятностью

- доступ к данным в кэше осуществляется быстрее, чем выборка исходных данных из более медленной памяти или удаленного источника, однако её объём существенно меньше памяти

CACHE

- состоит из набора записей
- каждая запись имеет идентификатор соответствия между элементами данных в кэше и их копиями в памяти
- попадание/промах кэша

CACHE

- При модификации данных в кэше происходит обновление данных в памяти это управляется политикой записи

- немедленная запись

- отложенная (обратная) запись - при вытеснении, на данных хранится флаг “dirty”, промах вызывает двойное обращение к памяти

CACHE когерентность

- Кэш центрального процессора разделён на несколько уровней
- Кэш-память уровня N+1, как правило, больше по размеру и медленнее по скорости доступа и передаче данных, чем кэш-память уровня N

CACHE когерентность

- L1 cache
- является неотъемлемой частью процессора, поскольку расположен на одном с ним кристалле и входит в состав функциональных блоков
- в современных процессорах обычно L1 разделен на два кэша — кэш команд (инструкций) и кэш данных (Гарвардская архитектура)
- L1 работает на частоте процессора, и, в общем случае, обращение к нему может производиться каждый такт

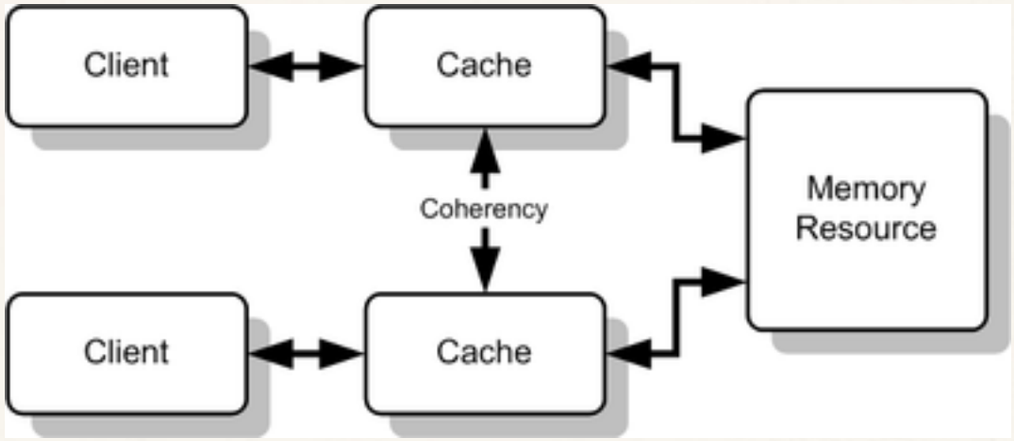
CACHE когерентность

- L2 cache
- тоже расположен на одном кристалле с процессором
- объём L2 от 128 кбайт до 1–12 Мбайт
- кэш второго уровня, находясь на том же кристалле, является памятью раздельного пользования — при общем объёме кэша в n Мбайт на каждое ядро приходится по n/c Мбайта, где c — количество ядер процессора

CACHE когерентность

- L2 cache
- кэш третьего уровня наименее быстродействующий, но он может быть очень большим — более 24 Мбайт
- L3 медленнее предыдущих кэшей, но всё равно значительно быстрее, чем оперативная память
- вмногопроцессорных системах находится в общем пользовании и предназначен для синхронизации данных различных L2

CACHE когерентность



- части кэша минимального размера - *линии кэша* (8 - 512 байт)
- протоколы взаимодействия между кэшами, которые сохраняют согласованность данных - *протоколы когерентности кэшей*

CACHE когерентность

- алгоритм кэш-когерентности MESI (x86)

- к каждой линии кэша приписываются 2 бита принимающие значения: modified, exclusive, shared, invalid

CACHE когерентность

- Modified - *линия* присутствует только в текущем кэше (dirty-bit)
- Exclusive - *линия* присутствует только в текущем кэше и соответствует своей копии в памяти
- Shared - *линия* может присутствовать в других кэшах и соответствует своей копии в памяти
- Invalid - *линия* недействительна

CACHE когерентность

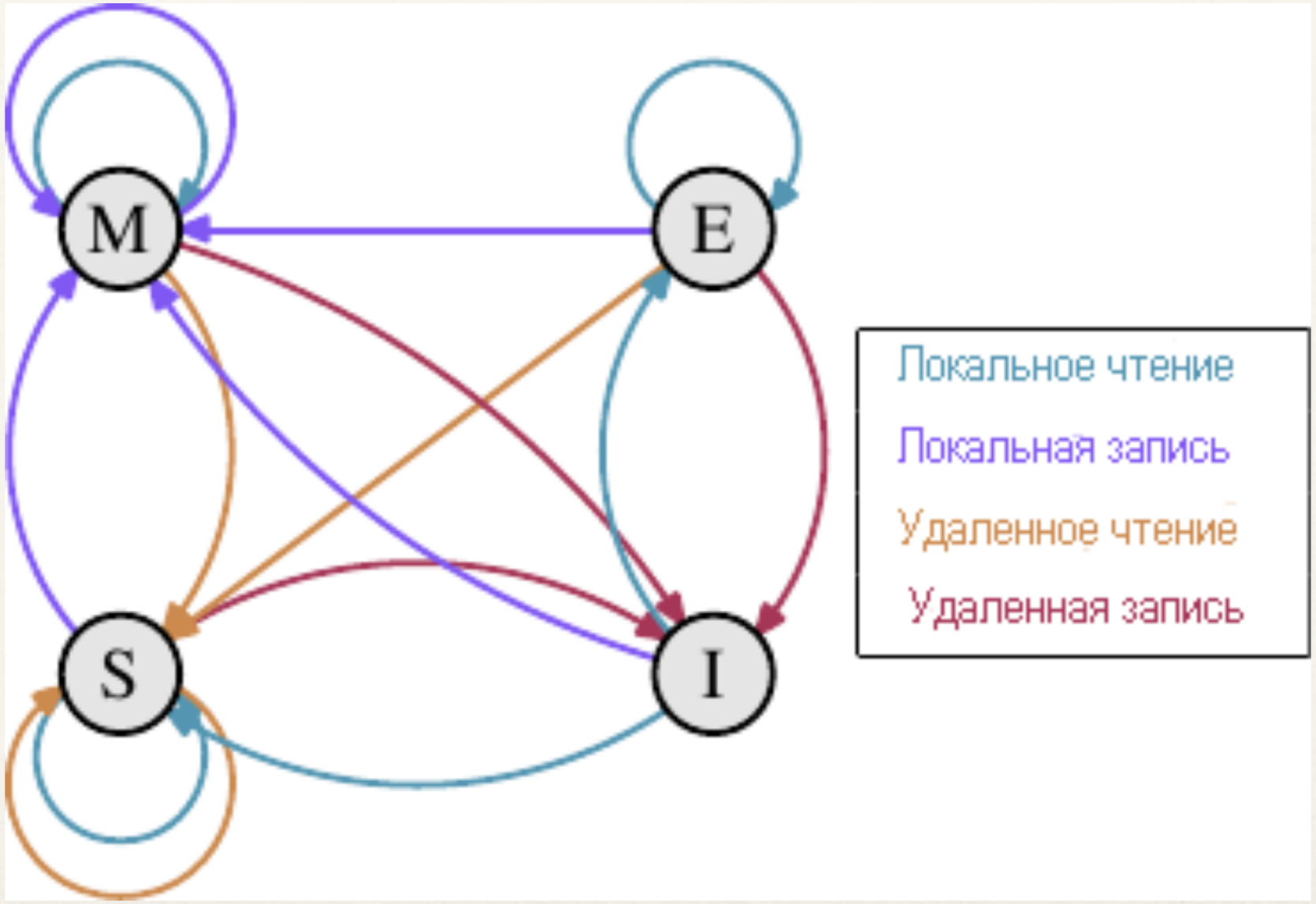
- операция чтения может быть из *линии* кэша в *любом* состоянии, кроме invalid

- операция записи может быть из кэша в состоянии exclusive *или* shared (*если* shared, то *остальные* надо пометить invalid)

- *линию* в состоянии invalid можно *удалить*

- *линию* в состоянии modified надо записать в память

CACHE когерентность



CACHE когерентность

- Фальшивое разделение
- Пример: область памяти в одну линию кэша, два потока
- один читает, другой модифицирует -> пересчитывание данных

CACHE

- Общие правила при написании программ:
- конверсии из формата в формат и многоуровневых структур следует избегать
- разделяемые данные допустимы (чтение через кэш)
- изменяемы данные допустимы при условии локальности
- изменяемых и разделяемых данных следует избегать

ЗАДАНИЕ

- Написать программу блочного перемножения матриц
- Измерить время выполнения перемножения для разных величин блоков (256, 512, 1024, 1025)

Вопросы?