PAMP: Categorized and Specialized Caching for SMPs

Erik Hagersten
Dept of Information Technology
Uppsala University
S-751 05 Uppsala, Sweden

1 Short summary

We propose a project within PAMP that aims at improved SMP cache behavior, especially for applications with large data sets in combination with real-time requirements. The focus of the Uppsala work will be on support for data with different behavior and requirements. We will develop algorithms that dynamically classifies data of different categories and configures the hardware to customize the treatment of each category. Especially, we will focus on support for requirements found in concurrent systems with a combination of a large data set and real-time requirement, such as the control processor found in telephony and/or process controll applications. The industrial partner is Ericsson Utvecklings AB and the contact people are Per Holmberg and Mike Williams.

This proposal has been formulated in close collaboration with Per Holmberg, processor architect at Ericsson Utvecklings AB.

2 Problem statement

Historically, high-performance control systems have been controlled by proprietary tailor-made central processors. Today, performance improvements of commodity servers and a desire to focus the development on core areas have created a desire to use Symmetric MultiProcessors (SMPs) to control such systems. However, SMPs banks heavily on the temporal and spatial locality frequently found in technical and commercial applications. The access pattern found in the application area of large concurrent control applications, such as telephony, have poor spatial and temporal locality. Further, such application have both soft and hard real-time requirements, not found in traditional SMP workloads. This is yet another example of a poor match for caches since worst-kind behavior must be assumed.

The proposed work will be divided into two subtasks, one exploring optimal cache organization and behavior for different categories of data accesses patterns, the other subtask will explore software-based algorithms for detecting certain categories of data accesses.

2.1 Subtask 1: Efficient caching of data categories

If we could anticipate how a particular piece of data would be used in the future and what its real-time requirement was we would be able to handle it in a more appropriate manner. This is especially applicable to caching, which is beneficial for some categories of data but could actually be a burden for others. There are many forms of caching known today and there are further unexplored ways to do caching. The goal of this subtask is to find the most appropriate and efficient ways of caching data of each category of data, e.g., real-time data, migratory data, stream data, and capacity-miss data. All these categories of data can be found in our target applications. One

example is the control programs of large switches and telephone exchanges, traditionally served by proprietary processors, such as Ericsson's APZ212-30. A first-order classification of such data would look like:

- RT: Low-level protocols communicating with other processors (RPs). The real-time requirements are today solved by a microcode-controlled co-processor (RPH) with hard real-time requirements.
- ERR: Error-handling software with hard real-time requirements that are run very rarely. These are often coded in microcode today.
- COMM: Frequent communication between parallel objects (blocks), causing producer/consumer and migratory sharing.
- RND: Random accesses to large "databases" holding subscriber data. Here, the re-use of data is highly unlikely
- DB: Maintenance streaming through the "databases"
- NORM: "Normal" data that may have both spatial and temporal locality.

Caching is very beneficial to some of these categories (e.g., NORM), could be a burden for others (e.g., COMM) and are neutral for some (e..g., DB, RND). Still, caching DB and RND data may cause replacement of NORM data, why the net effect could still be on the negative side.

While the specialized APZ212-30 provided software-controlled caches to support the different data behaviors, our goal is to find ways of supporting such behavior on existing commodity SMP architectures as well as suggesting simple extensions to SMP cache structures that would benefit such data usage.

2.2 Subtask 2: Dynamic detection and mapping of data categories

Given the result of Subtask 1, it would be beneficial for system properties and system performance if the data could be labeled according to its usage dynamically. For some applications, it can expected that the application programmer could provide hints about expected data usage. This could for example be true for some routines with real-time requirements. However, this can not be expected for the entire large and complex application. The goal of this subtask is to find algorithms that detects and classifies data of different types, e.g., migratory data, stream data, and capacity-miss data, and to find ways of implementing such algorithms on existing commodity SMP architectures as well as suggesting simple extensions to SMP cache structures that would make such algorithms more efficient.

The literature suggests integrated detection/optimizations hardware support for certain classes of access patterns, e.g., migratory sharing. Our goal is here to detect even more classes of data, while imposing small modifications to the SMP hardware. Ideally, the existing SMP architecture should be sufficient for implementing the algorithms. Contrary to the existing migratory sharing algorithms, implemented with a short response time supported by hardware, we instead propose a more sophisticated software-based approach that would be more appropriate for the our long-running applications, their need to detect more categories of access patterns and the urge to impose the smallest possible impact on the SMP architectures.

3 Main ideas

The main idea is to rely on the categorization of data to provide a better response time for large application with soft and hard real-time requirements.

3.1 Subtask 1: Efficient caching of data categories

This task assumes that data somehow is categorized and tries to find the most appropriate way of treating each category.

A first order idea to caching strategies for different categories is : RT: Guarantee space or pin down in cache, ERR: Guarantee space or pin down in cache, COMM: Exclusive read and/or write-through, RND: Do not cache, or cache temporarily, DB: Do not cache, and, NORM: Cache.

Our goal is to explore different cache implementations that can support the above behavior. The upper limit for the improved cache performance and overall system performance will be measured using the SIMICS simulator.

We will also explore solutions for commodity SMPs. Here, the categorization of data is done on a page-level granularity. The caches are divided into several so-called category segments (CS), e.g., RT segment, RND+DB segment, and NORM segments. A couple of the address bits (CS bits) used to index into the caches are used to distinguish between the segments. These CS bits are programmed as wild card address bits in the memory controller, i.e., their value will be neglected when the memory controller decodes an address. That way, there are many CS synonyms for the same physical page frame in memory that can cause the page frame to be mapped to different areas of the cache. The appropriate CS category for a page is selected by choosing the corresponding translation in the page table entry for the page. That way, data of different categories can not interfere with each other: RT data can be guaranteed to be in the cache and RND+DB will not be able to force replacement of NORM data. Another idea is to force the "prefetch cache", existing in some modern processors, to be used for the RND+DB categories. The are certain prefetch instructions (prefetch once) that will install a prefetched cache line only in the prefetch cache and not in the main cache, why unwanted data replacement in the main cache can be avoided.

We will use application examples provided from the industry as well as from the PAMP project "Simulation Concept to Model Real-Time Properties of SMPs"

3.2 Subtask 2: Dynamic detection and mapping of data categories

Also this subtask is searching for solutions in two levels, one for commodity SMPs and one a more limit-less approach for its implementation. A large amount of statistics is gathered in modern SMPs, e.g., frequency counts of certain hardware events such as accesses to the different cache states and the number of cache2cache transfers. Some of this statistics can be qualified with an address range. This is useful when studying the properties of a page. Further, very sparse address traces from the SMP interconnect can be sampled. The address tags of the caches are also accessible in some SMPs. We will search for algorithms that can classify pages using this kind of commodity SMP statistics.

The second part of this project aims at proposing a new set of hardware statistics that, if exposed to a user process, would allow for even more efficient and accurate detection of data categories. The goal is to find a solution with small complexity which would make inclusion in future SMP feasible.

We will use application examples provided by the industry as well as from the PAMP project "Simulation Concept to Model Real-Time Properties of SMPs"

4 Expected results and impacts

We expect to demonstrate a better cache utilization and real-time behavior for large real-time applications, such as switching software, running on SMPs. The results will be demonstrated on commodity SMPs. Further enhancements will be explored in SIMICS and schemes for building new caches and HW statistics will be proposed. If adopted by SMP vendors, this may change the applicability of SMPs in the real-time world.

5 Project plan

5.1 Subtask 1: Efficient caching of data categories

Phase 1 of this subtask will include the characterization of the different categories of data that have different caching needs. During this time a SIMICS simulation model will be developed that allows for a detailed study.

Phase 2 of this subtask will search for and implement of new caching strategy using available commodity SMP hardware (current plan: one E450, one 16-way E6000 follow-on and access to one 64-way WildFire follow-on). If this proves to be impossible, a strategy similar to the one outlined in the "Main Ideas" section will be implemented in the SIMICS simulator.

Phase 3 of this subtask will combine the results from phase 2 of this subtask with the results of phase 2 of the other subtask.

Phase 4 of this subtask will explore hardware extension to further support caching categories in with new cache structures and mechanisms.

Phase 5 of this subtask will propose an optimized system solution combining the results from phase 4 of this subtask with the results of phase 4 of the other subtask.

Milestones and deliverables:
Deliverable (1 year after start): Report on state-of-the-art.
Deliverable (2.5 years after start): Licentiate thesis.
Deliverable (5 years after start): Final report. Doctoral thesis.

5.2 Subtask 2: Dynamic detection and mapping of data categories

Phase 1 of this subtask will include the characterization of the different categories of data that have different caching needs. During this time a SIMICS simulation model will be developed that allows for a detailed study.

Phase 2 of this subtask will search for and implement new algorithms for detecting data categories using statistics available in commodity SMPs hardware (current plan: one E450, one 16-way E6000 follow-on and access to one 64-way WildFire follow-on). Depending on the success of phase 2 of the other subtask this phase may instead be performed using the SIMICS simulator.

Phase 3 of this subtask will combine the results from phase 2 of this subtask with the results of phase 2 of the other subtask.

Phase 4 of this subtask will explore the efficiency of new alternative hardware statistics for finding data categories.

Phase 5 of this subtask will propose an optimized system solution combining the results from phase 4 of this subtask with the results of phase 4 of the other subtask.

Milestones and deliverables:
Deliverable (1 year after start): Report on state-of-the-art.
Deliverable (2.5 years after start): Licentiate thesis.
Deliverable (5 years after start): Final report. Doctoral thesis.

6 Preliminary budget

The project will involve two Ph. D. students and one professor advisor. The activity level of these will be 80% and 20% respectively. Equipment depreciation cost is based on two workstations, software and regular maintenance costs, at a cost level of 60 kkr each year. Traveling cost estimate is based on internal Swedish travel trips motivated by the close collaboration with the other PAMP/ARTES partners and normal traveling for international conferences, workshops etc.

Item	Cost
Professor Erik Hagersten, 20%	100 kkr
PhD Student 80 %	163 kkr
PhD Student 80 %	163 kkr
Social costs (44 % of rows 1 – 3)	187 kkr
University OH (17.2 % of rows 1 – 4)	105 kkr
Equipment depreciation	60 kkr
Travel	70 kkr
Others	30 kkr
Office cost (12.7 % 1 – 8)	112 kkr
Dept. OH (14.7 % of rows 1 – 8)	129 kkr
Total	1119 kkr (Does not include Högskolemoms)

7 Related research

There have been many proposals for hardware algorithms detecting certain access patterns which result in some special treatment in their caching strategies, e.g., the detection and optimization of migratory sharing by Stenström et al. Software-controlled caching has been proposed in several IBM architectures as well as the Ericsson APZ212-30. Software-based detection and conversion between NUMA and S-COMA have also been proposed, e.g., the R-NUMA by Falsafi et al and the WildFire architecture described by Hagersten et al.

The research proposed here is unique in that we propose a software-based algorithm used to control the caching strategy used in the 2nd level caches of SMPs and that it addresses the application with large data sets and combined with some real-time routines.

8 Relation to the profile

The main objective of PAMP is to develop methods and tools for exploiting symmetric multiprocessors for real-time applications. This proposal includes tools for dynamically detecting certain access patterns in SMPs. We propose methods to provide more efficient use of the SMP caches as well as providing real-time properties in spite of the caches.

9 Industrial relevance

The proposed Uppsala work is targeting workloads commonly found in the Swedish Industry. This proposal has been formulated in close cooperation with one of the architects of Ericsson's APZ212-30. The problem statements have been taken directly from this kind of applications. Instead of proposing a completely new architecture, as was the case for APZ212-30, this proposal is centered around commodity SMP architectures and proposes two

level of optimizations. One using commodity SMPs of today and one relying on minor modifications to the hardware statistics and cache implementation hopefully found in future SMPs.

10 Relation to other SSF programs

There are two research programs supported by SSF that are to some extent related to our project proposal. PCC "Vetenskapliga beräkningar".

11 Context

11.1 The research group

This work will be performed in the new group built around the new Professor Chair in Computer Architecture recently created in the new Department for Information Technology at Uppsala University.

The goal is to create a group of about six PhD student.

11.2 Complementary activities and funding

Current funding: One professor, two PhD students and one post Doc financed by Uppsala University.

Current Infrastructure: Funding from the Goran Gustavsson fund and local funding will finance an SMP simulation server with four CPUs of 250 kkr.

Funding applied for from the industry: Two PhD students.

Infrastructure applied for from the industry: A 16-way SMP server and access to 25% time of a large DSM system with 64 CPUs.

11.3 Research cooperation

None so far. Based the background of professor Hagersten, it is expected that this group will form close ties to the Swedish Institute of Computer Science and Chalmers University in Sweden, and the University of Wisconsin and MIT in the US.

11.4 Industrial cooperation

None so far. Based the background of professor Hagersten, it is expected that this group will form close ties to Ericsson in Sweden, and Sun Microsystems in the US.