
**Mission:**
This Cost Action aims at consolidating European research on this important field, by coordinating the European research groups working on the development of complementary, interdisciplinary aspects of Transactional Memories, including theoretical foundations, algorithms, hardware and operating system support, language integration and development tools, and applications.

**Parallel programming (PP) used to be an area once confined to a few niches, such as scientific and high-performance computing applications. However, with the proliferation of multicore processors, and the emergence of new, inherently parallel and distributed deployment platforms, such as those provided by cloud computing, parallel programming has definitely become a mainstream concern.**

Transactional Memories (TMs) answer the need to find a better programming model for PP, capable of boosting developers' productivity and allowing ordinary programmers to unleash the power of parallel and distributed architectures avoiding the pitfalls of manual, lock based synchronization. It is therefore no surprise that TM has been subject to intense research in the last years.

This Cost Action aims at consolidating European research on this important field, by coordinating the European research groups working on the development of complementary, interdisciplinary aspects of Transactional Memories, including theoretical foundations, algorithms, hardware and operating system support, language integration and development tools, and applications.

**WG1: Cross-WG Activities, Showcases**
WG1 plays a special role within this Action by bundling cross-WG activities in order to ensure their cohesion and boost interdisciplinary collaborations. Specifically, this WG will be in charge of the integration of the remaining four WGs so that researchers in these different fields can share their common expertise, favouring cross-fertilization and minimizing fragmentation of efforts.

WG Leader: Prof. Luis Rodrigues

**WG2: Theoretical Foundations & Algorithms**
Unlike database transactions, in a TM transactions do not execute in a sand-boxed environment. This exposes TMs transactions to hazardous execution scenarios that would not be filtered out by the safety and liveness properties normally ensured in a DBMS environment.

WG Leader: Dr. Tim Harris

**WG3: Hardware's & Operating System's Supports**
Albeit TMs can be purely implemented in software, there is a growing consensus that some form of hardware support is desirable to improve performance. By providing hardware level support for conflict detection and version management, in fact, the bookkeeping overhead incurred by STMs can be drastically reduced, making TMs performance superior even to that of hand-crafted, fine grain locking. On the other hand, the implementa-
tion of these mechanisms in hardware is way more problematic than in software, as this can entail invasive, risky modifications of crucial components of existing processors such as cache, TLB and bus protocols.

WG Leader: Prof. Gilles Muller

**WG4: Language Integration & Tools**
There has been an increasing number of papers illustrating the potential of transactions in parallel programs: introducing TM implementation techniques, and proposing hardware mechanisms to accelerate common case behavior. Nevertheless, TMs are still not widely used by application developers as the TM technology has not yet been integrated with any of the widely used software development frameworks. It is unlikely that programmers will abandon the many advantages of mature frameworks in order to try this new technology. As a result, there are very few large applications that use transactions and there is no clear assessment on whether TM actually improves programmability.

WG Leader: Prof. Wolfgang Karl

**WG5: Applications & Performance Evaluation**
TMs appear to have a huge potential in simplifying the development of parallel applications. On the other hand, being the research field on TMs still in its infancy, the number of complex benchmarks for TMs currently available is still very limited, and the real-world applications pioneering the adoption of TMs are probably even less. This represents a major impediment not only for realistically evaluating the performance of the various TM solutions proposed in literature, but also for precisely assessing the usability of TMs in complex, large scale applications and across the many different application domains that could potentially benefit from their adoption (including web-based applications, video-games, CAD systems, stream processors, financial, healthcare, video-editing, navigation systems, simulators, graph analysis toolkits, just to mention a few).

WG Leader: Prof. Pascal Felber