Tera Multithreaded Multiprocessor

- Provides hardware support for multithreading
- Uses shared memory with full/empty bits for synchronization
- Hardware supports switching between many thread contexts on a single core
- Operating system handles allocation of contexts and scheduling of collections of threads
Cray XMT: handle long memory access latency by providing hardware support for large numbers of lightweight threads

This paper shows that it is possible to implement a similar model on commodity hardware using software-managed coroutines for concurrency

The paper includes performance results showing little runtime overhead
Questions

- How did the complexity of building a compiler for the Tera architecture compare to building a compiler for a VLIW architecture?
- How would the architecture from the second paper perform on other (non-graph processing) types of applications?