Contribution ID: **12**   Type: **not specified**

Throughput Computing in C++

Thursday, 6 November 2008 12:00 (40 minutes)

Power consumption is the ultimate limiter to current and future processor design, leading us to focus on more power efficient architectural features such as multiple cores, more powerful vector units, and use of hardware multi-threading (in place of relatively expensive out-of-order techniques). It is (increasingly) well understood that developers face new challenges with multi-core software development. The first of these challenges is a significant productivity burden particular to parallel programming. A big contributor to this burden is the relative difficulty of tracking down data races, which manifest non-deterministically. The second challenge is parallelizing applications so that they effectively scale with new core counts and the inevitable enhancement and evolution of the instruction set. This is a new and subtle qualifier to the benefits of backwards compatibility inherent in Intel® Architecture (IA): performance may not scale forward with new micro-architectures and, in some cases, actually regress. I assert that forward-scaling is an essential requirement for new programming models, tools, and methodologies intended for multi-core software development.

We are implementing a programming model called Ct (C for Throughput Computing) that leverages the strengths of data parallel programming to help address these challenges. Ct is a C++-hosted deterministic parallel programming model integrating the nested data parallelism of Blelloch and bulk synchronous processing of Valiant (with a dash of SISAL for good measure). Ct uses meta-programming and dynamic compilation to essentially embed a pure functional programming language in impure and unsafe C++. A key objective of the Ct project is to create both high-level and low-level abstractions that forward-scale across IA. I will describe the surface API and runtime architecture that we've built to achieve this, as well as some performance results

Summary

I'll present a 30-45 minute talk presenting Ct, a new programming model developed at Intel with architectures like Larrabee and future heterogeneous platforms in mind.

Primary author: Dr GHULOUM, Anwar (Intel Corporation)

Presenter: Dr GHULOUM, Anwar (Intel Corporation)

Session Classification: Thursday, 06 November 2008

Track Classification: 1. Computing Technology