Stack Walking Breakout Report

On Wednesday afternoon nine workshop participants met to discuss open problems in call stack unwinding, an important operation for many performance analysis and debugging tools. We focused on three main topics: Technical challenges to call stack unwinding through signal handlers on BlueGene, incompatibilities between a "shadow stack" optimization and C++ exceptions, and mechanisms for avoiding disk accesses during call stack unwinding.

Several groups had an idea of maintaining a shadow stack, which allows faster call stack unwinding and function execution counting. Unfortunately, this shadow stack can cause incompatibilities with C++ exceptions. We identified the problem and brainstormed three possible solutions. None of the solutions are perfect--each involving trade-offs, so we left an action item of having someone explore this further.

We also discussed ways to prevent call stack from accessing disk for symbol and debug information. The HPCTookit project has already had success in this area, and we discussed ways to accomplish the same results in another tool, StackwalkerAPI. We identified several mechanisms that StackwalkerAPI could use and have an action item to explore these.

Group attendees were Madhavi Krishnan, Martin Schulz, Bronis de Supinski, Matthew LeGendre, Jeff Hollingsworth, Mike Fagan, Jim Galarowicz, Nick Rutar, and Heidi Poxon.