Value Specialization using PLTO

Tal Shaked

May 1, 2002

Abstract

It is quite common in programs for some variables and registers to take on a small set of values throughout the course of execution. A variable that takes on only one value during execution is referred to as a runtime constant. Compilers cannot optimize runtime constants since these are not known at compile time. Value specialization is the process of optimizing runtime constants as well as other variables that take on a specific value most of the time. It involves instrumenting an executable in order to keep track of the values of different variables and to find those with skewed distributions. These program points can then be specialized to run more quickly for the common value. Although the idea of value specialization is not new, implementing it for the Pentium architecture has posed some interesting challenges. This thesis describes the details involved in augmenting PLTO (Pentium Link-time Optimizer) with value specialization.