INFORMATIK-KOLLOQUIUM
AB Programmiersprachen und Übersetzer
Institut für Computersprachen

Einladung

zum Informatik-Kolloquium des
AB Programmiersprachen und Übersetzer am
Dienstag, den 21. April 2009, um 11:00 Uhr s.t.
in der Bibliothek E185.1, Argentinierstr. 8, 4. Stock (Mitte)

Es spricht

Clément Ballabriga, M.Sc.

Université Paul Sabatier, TRACES/IRIT, Toulouse, France

über

An Introduction to OTAWA

Abstract: OTAWA is a freeware application dedicated to the computation of the WCET of programs. More precisely, OTAWA concerns the numerous WCET approaches based on static analyses of the executable program. Unlike many existing usual tools, the choice has been done to avoid specialisation and to produce, instead, a generic and open framework.

WCET computation has always required to pass flow fact information from the user to the computation analysis even with tools including loop bound analyses. From the start of OTAWA and overmore with the use of the oRange, OTAWA has developed a more and more extensible annotation system based on XML file. In the last project, this annotation system has been largely extended to embed timing information and user domain definitions and to locate accurately information in the program and in the execution time.

Biography: Clément Ballabriga is a PhD student supervised by Hugues Cassé since 2006 and works on cache behavior, partial analysis and abstract interpretation. He is a member of the TRACES team whose research interests include hardware issues of real-time embedded systems. The main goal is to guarantee that the execution time of an application code meets the system deadlines. We focus on characterizing the temporal properties of components off-the-shelves. Our target is to propose ways to use these components such that safe and tight worst-case execution time estimates can be computed. We also study architectural extensions that should improve the time predictability of the components. The estimation of the WCET requires three steps: a static analysis of the code identifies all the possible execution paths; the target hardware is modelled to determine the individual execution times of the basic blocks; then, the results of the previous steps are combined to evaluate an upper bound of the global execution time. Part of our work concerns the first and second steps. (`http://www.irit.fr/recherches/ARCHI/MARCH/index.php3`)

Zu diesem Vortrag lädt der *Arbeitsbereich für Programmiersprachen und Übersetzer am Institut für Computersprachen* herzlich ein.
Tee: 10:45 Uhr in der Bibliothek E185.1, Argentinierstr. 8, 4. Stock (Mitte).