SWAT Toolchanin achievements and results

SWAT provides a set of models and a fully automated methodology for characterization, analysis and execution time, energy consumption estimation and optimization.

The **characterization flow** allows automatically building the **target microprocessor models** needed to support the estimation flow. The process is based on static code analysis and requires limited data on the target processor.

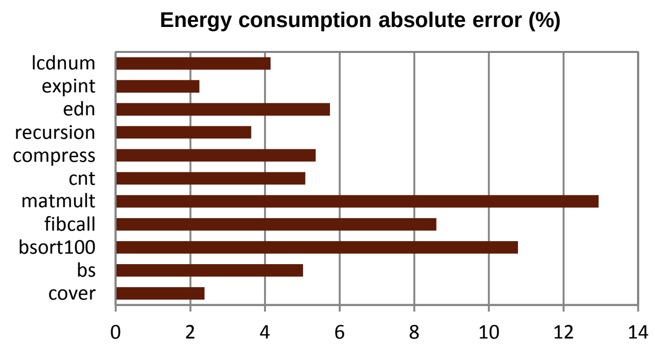
The **analysis flow** is based on a **static representation** of the source code in LLVM pseudo-assembly language, and on **profiling data** derived from code executions, either on a generic host machine or on the target platform. The analyses performed range from **source code structure** (functions, static call graph, basic blocks, CDFG, ...) to **IR statistics** (instructions, classes of instructions, load/store counts, ...) to more informative and **complex analyses** such as inlining candidated, type promotion/demotion, memory pressure, stack size, basic-block and function level energy and execution time distributions. The SWAT flow allows to back-annotate energy and timing figures onto the source code with a very high accuracy.

Different tools are provided by the **optimization flow**:

- Optimal transformation recipe derived through DSE
- High-level parametric DSE
- Abstract and application-wide optimization hints

Currently the flows and models have been validated for the STMicroelectronic ultra low power ReISC III core using instruction-set simulation results as reference.

Accuracy of the results is promising and the run-time of the estimation toolchain is more 400 times faster than ISS.



Main reference:

C. Brandolese, S. Corbetta and W. Fornaciari, **"Software Energy Estimation Based on Statistical Characterization of Intermediate Compilation Code",** ISLPED'11.

SWAT Development Team

Politecnico di Milano – DEI
Via Ponzio 34/5
20133 Milano, Italy

Project Manager and Contact

Prof. William Fornaciari
fornacia@elet.polimi.it
http://home.dei.polimi.it/fornacia
phone: +39 02 2399 3504

Chief Developer

Prof. Carlo Brandolese
carlo.brandolese@polimi.it

Complex Project

OFFIS e.V.
philipp.hartmann@offis.de
http://complex.offis.de

SWAT

SoftWare Analysis Toolset

Developed at
Politecnico di Milano

POLITECNICO MILANO

Within the 7th Framework IP Project

COMPLEX

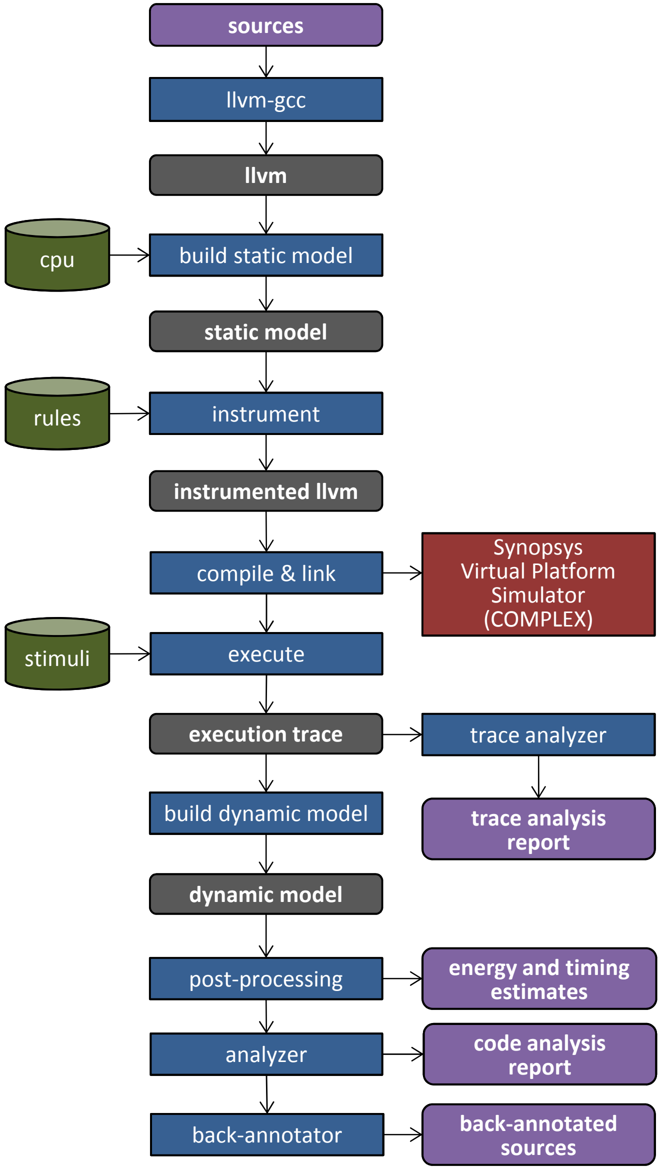
Visit us at DATE 2012 - Booth EP8
Thursday, March 15, 10:00-12:30

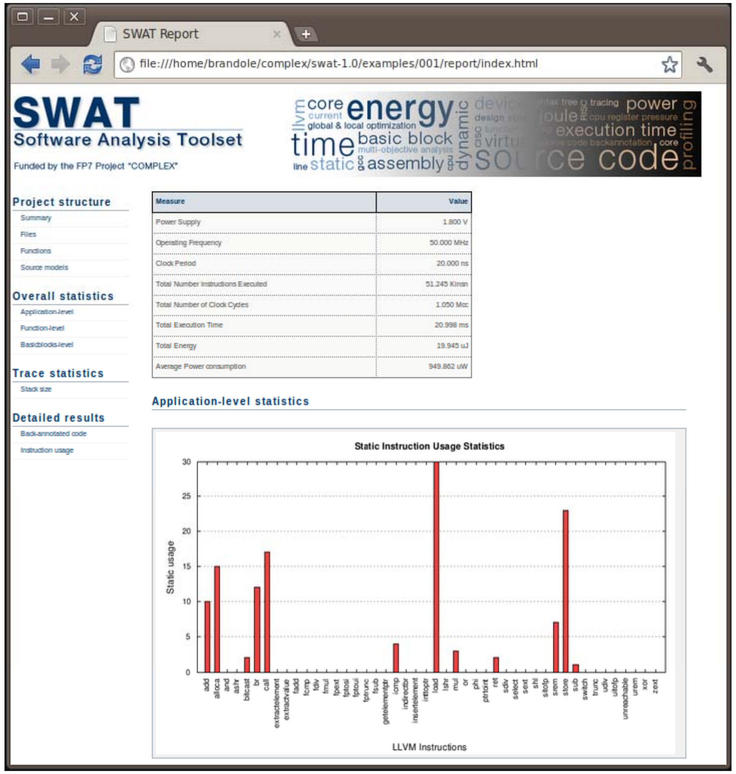
SWAT analysis and estimation toolchain

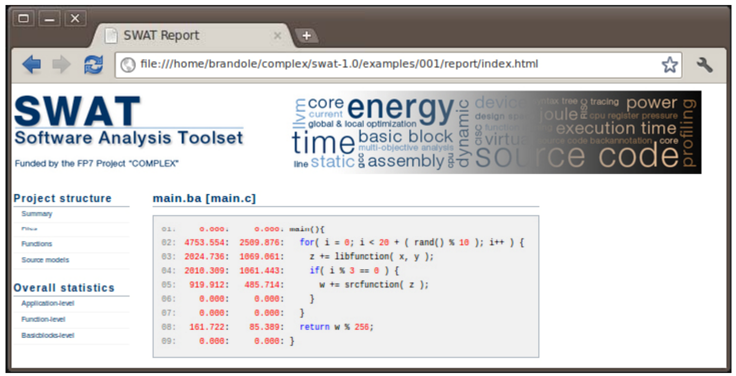
The SWAT analysis and estimation toolchain provides support for early inspection of the non-functional properties of an application starting from its **C source code**. The key idea behind the flow is decoupling the underlying models into:

- A **static model of the source code** structure that only accounts for the semantics of the application
- A target **characterization library** specifying the elementary timing and power figures of the **target processor**
- A **data dependent model of the application behaviour** constituted by an **LLVM basic-block profiling**

The **back-end of the flow**, constituted by a post-processor, two analyzers and a back-annotator, combines such models to derive a **complete static and dynamic characterization of the application**. The toolchain produces a detailed, fine-grained easy-to-navigate report collecting all the main analysis results.

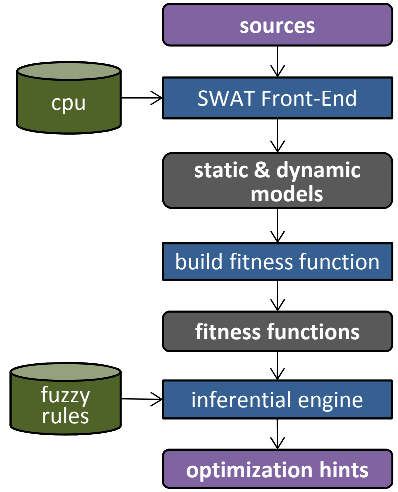






SWAT optimization toolchain

Based on the same front-end of the estimation flow and using the results extracted from the dynamic models and the execution traces, the main optimization flow apply a set of fuzzy rules on selected portions of the application to suggest the most promising transformations to apply.



A second optimization flow explores compiler optimization options in order to determine the transformation mix that best fits the specific applicatio. To this purpose, the MOST design space exploration engine is used to generate sets of transformation mixes for the LLVM optimizer until the best optimization recipe is found.