

# SATIrE within ALL-TIMES

Improving Timing Technology with Source Code Analysis

Gergö Barany

Institute of Computer Languages, Vienna University of Technology

15. Kolloquium Programmiersprachen und  
Grundlagen der Programmierung  
Maria Taferl

# Timing Analysis

Knowledge of each task's worst-case execution time (WCET) is needed to ensure correct timing of safety-critical hard real-time systems.

Design decisions:

- Static/dynamic analysis
- Source-based/binary analysis
- Analysis algorithm
- ...
- Various combinations are possible.

The ALL-TIMES project's goal is to integrate timing tools that use different approaches.

# Timing Analysis

Knowledge of each task's worst-case execution time (WCET) is needed to ensure correct timing of safety-critical hard real-time systems.

Design decisions:

- **Static**/dynamic analysis
- **Source-based**/binary analysis
- Analysis algorithm
- ...
- Various combinations are possible.

The ALL-TIMES project's goal is to integrate timing tools that use different approaches.

# Coming up...

- 1 The SATIrE Framework
- 2 The ALL-TIMES Project
- 3 Tool Connections in ALL-TIMES

# SATIrE: Static Analysis Tool Integration Engine

SATIrE is a framework for integrating source code analysis tools for C.

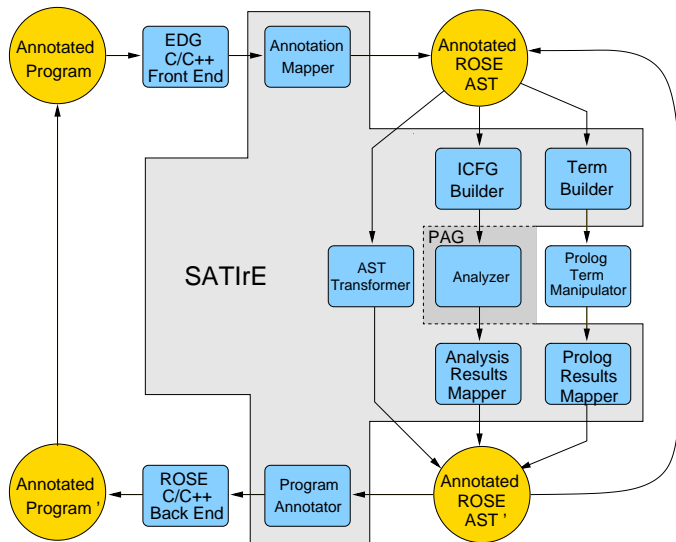
Currently integrated:

- ROSE: source-to-source transformation framework
- PAG: data-flow analyzer generator
- Termite: Prolog library for high-level program manipulation

The integration provides tools and representations for:

- Data-flow analysis of C programs (source code level)
- Source-to-source transformations
- Source code annotations

# SATIrE Architecture



# SATIrE's Analyzers

Analyzers provided by SATIrE:

**Points-to analysis:** Flow-insensitive unification-based (Steensgaard-style) almost linear analysis; with context-sensitive extensions.

**Interval analysis:** Abstract interpretation to determine possible value ranges of integer variables; arbitrary context-sensitivity.

**Loop bounds analysis:** Constructs and solves systems of inequalities describing loop behavior.

# ALL-TIMES: Integrating European Timing Technology

ALL-TIMES is a European project (12/2007–02/2010) aimed at integrating timing analysis tools.

Six partners: Four companies, two university groups.

Partners contribute a wide range of analysis techniques and expertise.

**SATIrE** is involved in three tool connections to provide/exchange information derived from source-based analysis.



# Source-Level Timing Analysis

**Traditionally:** Programmers annotate program with high-level information.

- Tedious and error-prone

**SATIrE's role in ALL-TIMES:** Generate annotations automatically

Advantages of source-code analysis for users:

- Reduce user's annotation burden
- User-checkable analysis information

Advantages of source-code vs. binary analysis:

- Symbolic pointer analysis: names, not numeric addresses
- Compound type information available

# Overview of SATIrE's Connections (1/2)

To RapiTime (dynamic analyzer):

- Export function pointer information.
- **Benefit:** Confirm or correct dynamic information.

To aiT (static binary analyzer):

- Export function and data pointer information.
- **Benefit:** Improve flow and value analysis.

# Overview of SATIrE's Connections (2/2)

Two-way connection to **SWEET** (static flow analyzer):

- Export program representation (ALF).
- Export pointer, interval, loop bound information.
- Import flow constraints.
- **Benefits:** Improve SWEET's analyses, communicate its flow information to other tools.

# Conclusions

- ALL-TIMES integrates European timing analysis technology.
- Integrations combine strengths of various approaches:
  - dynamic vs. static analysis
  - binary vs. source code analysis
  - abstract execution vs. abstract interpretation
  - ...
- SATIrE contributes source-based pointer, interval, and loop bounds analyses.

This work is supported by the research project “Integrating European Timing Analysis Technology” (ALL-TIMES) under contract No. 215068 funded by the 7th EU R&D Framework Programme. See <http://www.all-times.org> for more information about the ALL-TIMES project.

# Conclusions

- ALL-TIMES integrates European timing analysis technology.
- Integrations combine strengths of various approaches:
  - dynamic vs. static analysis
  - binary vs. source code analysis
  - abstract execution vs. abstract interpretation
  - ...
- SATIrE contributes source-based pointer, interval, and loop bounds analyses.

Thank you for your attention! Questions?

This work is supported by the research project “Integrating European Timing Analysis Technology” (ALL-TIMES) under contract No. 215068 funded by the 7th EU R&D Framework Programme. See <http://www.all-times.org> for more information about the ALL-TIMES project.