

Aufgabenstellung Internship

Benchmarking Incremental Dynamic Attribute Evaluation

RACR/JastAdd Performance Case Study in the Domain of Invasive Software Composition

Student: Марта Тасић
Matrikelnummer: 3929351
Beginn: 01.01.2014
Einzureichen am: 30.06.2014

Attribute grammars and graph rewriting are well-known compiler construction concepts for the development of semantic analysers, code generators and optimizers. Typically, both are independently applied in different, but consecutive compilation phases. To investigate advantages of a more tightly integration of attribute grammars and rewriting, a new technique called attribute grammar controlled rewriting and its reference implementation *RACR*¹ has been developed at the software technology chair. *RACR* is a Scheme library providing a set of functions that can be used to specify abstract syntax tree schemes and their attribution and construct respective trees, query their attributes and node information and annotate and rewrite them. *RACR*'s attribute evaluator is demand-driven and applies a new incremental attribute evaluation technique based on dynamic attribute dependency analyses. Compared to static attribute dependency analysis, dynamic attribute dependency analyses permit more precise attribute cache flushing and therefore better incremental evaluation.

Objective of the internship is an empirical evaluation of the demand-driven, dynamic incremental attribute evaluation technique emphasised by *RACR* - in particular regarding performance compared to demand-driven non-incremental evaluation techniques as employed e.g., by the *JastAdd* attribute grammar system. The evaluation has to be conducted in form of a case study in the area of invasive software composition. As basis for the case study *SkAT*², an invasive software composition system for *Java* developed at the software technology chair, can be used. *SkAT* has been developed using *JastAdd*. Invasive software composition is suffering from performance issues since composition rewrites prevent attribute caching. If well-formedness or other static semantics are reused for composition, the efficient evaluation of the attributes representing such information therefore is performance critical. The tasks to realise a comparative case study between incremental *RACR* and non-incremental *JastAdd* attribute evaluation are:

¹ <https://code.google.com/p/racr/>

² <https://fusionforge.zih.tu-dresden.de/projects/skat/>

- Design of a component and composition language with emphasises on interesting interactions between semantic analyses and composition rewrites
- Development of composition fragments and recipes well-suited to evaluate performance
- Implementation of the languages in *RACR* and *JastAdd* using comparable attributions and attribute equations
- Benchmarking of the implementations
- Reasoning about and illustration of benchmarking results

The solution must be programmed in *Scheme*; its source code and the software artefacts the code depends on will be published under terms of the *MIT License* (*X11 License*).

Dipl.-Inf. Christoff Bürger (Supervisor)
Dresden, 19.12.2013

Мартин Тасић (Student)
Dresden, 19.12.2013