Towards Incremental Compilation for Stratego

**Whole-program compilation model**

*Stratego is a DSL for program transformation based on term rewriting with programmable rewriting strategies (www.metaborg.org). Named rules and strategies can be defined in multiple modules. Multiple definitions with the same name are merged as different options of the rule or strategy. This is an extensibility mechanism of the language.*

Whole program compilation takes in all relevant files with Stratego modules. It parses all and builds a single internal model of the program. This is then used to generate a Java class for each strategy, and to generate two shared classes.

**Dynamic linking compilation model**

With a dynamic linking approach we run the compiler once for each module. This means that if multiple modules define the same strategy, we now have duplicate modules. These classes need to be merged (linked) at run-time.

**NEW: Static linking compilation model**

To achieve incremental compilation with static linking, we split up the compiler into a front-end and a back-end.

In this model, a change to a single strategy definition in a single module will result in some checks by the incremental system and the execution of one front-end task for the module and one back-end task for the changed strategy.

---

Jeff Smits, Eelco Visser

Programming Languages Research Group — Delft University of Technology