

Building and Using a Russian Resource Grammar in GF

Janna Khegai and Arne Ranta

Department of Computing Science
Chalmers University of Technology and Gothenburg University
SE-41296, Gothenburg, Sweden
{janna, arne}@cs.chalmers.se

Abstract. Grammatical Framework (GF) [5] is a grammar formalism for describing formal and natural languages. An application grammar in GF is usually written for a restricted language domain, e.g. to map a formal language to a natural language. A resource grammar, on the other hand, aims at a complete description of a natural languages. The language-independent grammar API (Application Programmer's Interface) allows the user of a resource grammar to build application grammars in the same way as a programmer writes programs using a standard library. In an ongoing project, we have developed an API suitable for technical language, and implemented it for English, Finnish, French, German, Italian, Russian, and Swedish. This paper gives an outline of the project using Russian as an example.

1 The GF Resource Grammar Library

The Grammatical Framework (GF) is a grammar formalism based on type theory [5]. GF grammars can be considered as programs written in the GF grammar language, which can be compiled by the GF program. Just as with ordinary programming languages, the efficiency of programming labor can be increased by reusing previously written code. For that purpose standard libraries are usually used. To use the library a programmer only needs to know the type signatures of the library functions. Implementation details are hidden from the user.

The GF resource grammar library [4] is aimed to serve as a standard library for the GF grammar language. It aims at fairly complete descriptions of different natural languages, starting from the perspective of linguistics structure rather than the logical structure of applications. The current coverage is comparable with, but still smaller than, the Core Language Engine (CLE) project [2].

Since GF is a multilingual system the library structure has an additional dimension for different languages. Each language has its own layer, produced by visible to the linguist grammarian. What is visible to the application grammarian is a an API (Application Programmer's Interface), which abstracts away from linguistic details and is therefore, to a large extent, language-independent. The module structure of a resource grammar layer corresponding to one language is shown in Fig. 1. Arrows indicate the dependencies among the modules.