The GENTLE Compiler Construction System

Mathias Peters & Eik Villwock

Allgemeines:

Gentle ist eine problemorientierte Programmiersprache zum Schreiben von Compilern. Die Sprache wurde 1989 von F.W. Schröder (GMD) entwickelt. Sie fasst die wichtigsten Methoden und Werkzeuge im Rahmen des Compilerbaus zusammen.

Gentle ist eine deklarative Compilerbeschreibungssprache, die auf logischer Programmierung beruht. Sie basiert auf rekursiven Definitionen und strukturierten Ersetzungen. Ein Programm besteht im Wesentlichen aus Regeln die Prädikate beschreiben. Es können aber auch Konstrukte imperativer Programmiersprachen wie globale Variablen oder Arrays benutzt werden.

Gentle deckt das gesamte Spektrum der Übersetzung ab. So kann ein Compiler mit all seinen Phasen vollständig auf einer einheitlichen und hohen Abstraktionsebene spezifiziert werden. Gentle erzeugt effizienten lex, yacc und C-Code. Es besteht aber auch die Möglichkeit Teile des Compilers direkt in C zu implementieren.

Es gibt sechs unterschiedlichen Arten von Prädikaten. Mit den Prädikaten 'TOKEN' und 'NONTERM' wird die kontextfreie Grammatik beschrieben. Termumformungen werden mit den Prädikaten 'ACTION' und 'CONDITION' spezifiziert. Das Prädikat 'SWEEP' kann zum Durchschreiten komplexer Strukturen dienen. In Gentle ist auch eine optimale Codeauswahl mit dem 'CHOICE'-Prädikat möglich.

Installation:

Der Download von Gentle kann von folgender Adresse erfolgen:

<http://www.first.gmd.de/gentle/>

Anschließend ist die Datei `gentle-97.tar.gz` zu entpacken und durch das Ausführen der Datei `build` in den Verzeichnissen `gentle`, `lib`, `reflex` zu kompilieren. Hierzu wird ein C-Compiler benötigt.

Gentle ist lauffähig unter allen verbreiteten UNIX-Systemen und DOS.

Zur Compilererstellung muss ein Scanner (`flex`, `lex`) und ein Parser (`bison`, `yacc`) auf dem Rechner vorhanden sein.

Gentle kann kostenlos für Ausbildung genutzt werden. Für kommerzielle Anwendungen ist ein Lizenzvertrag zu schließen. Das Gentle System ist in der Industrie und Forschung im Einsatz.

Vergleich Gentle-Prolog:

Gentle besitzt im Gegensatz zu Prolog verschiedene Prädikate. Parameter von Prädikaten müssen als Eingabeparameter oder Ausgabeparameter festgelegt werden. Ein weiterer Unterschied besteht im verwendeten Wiederaufsetzungsverfahren. Gentle benutzt ein flaches Wiederaufsetzen (`shallow backtracking`) anstelle von dem in Prolog benutzten tiefem Wiederaufsetzen (`deep backtracking`). Die einzelnen Prädikate werden auf unterschiedliche Weise ausgeführt.

Fazit:

Gentle bietet sich zur Compilererstellung an, wenn ein logischer Programmierstil bevorzugt wird. Die Beschreibung findet somit auf einem hohen Abstraktionsniveau statt.

Es werden schnell und effiziente Compiler erzeugt. Durch Verwendung von anderen Werkzeugen wie `lex` und `yacc` werden aber auch die Eigenschaften dieser Tools in den mit Gentle erstellten Compilern sichtbar.

Links zu weiteren Informationen:

Hauptseite zu Gentle:

<http://www.first.gmd.de/gentle/>

Weiterführende Informationen:

<http://www.informatik-vollmer.de/publications/gentle-manual.pdf>

<http://www.tfh-berlin.de/~grude/SkriptGentle.pdf>

<http://www.tfh-berlin.de/~grude/CBmitGentle.pdf>

Seminarseite:

<http://informatik.hu-berlin.de/~kunert/lehre/compilergeneratoren/>