Anhang B

Kochrezept: Attributierte Grammatiken

Der Entwurf von Attributierten Grammatiken läßt sich in folgende Schritte einteilen:

Umformen der Grammatik: Grammatiken in EBNF müssen in BNF umgewandelt werden.

Ermittlung der benötigten Informationen: Zusätzlich zu den zu erzeugenden Informationen (z.B. Codesequenz) werden meist noch Hilfsinformationen (z.B. Codeadressen) gebraucht.

Festlegung des Datenflusses: In diesem Schritt wird ermittelt, wo die Information gebraucht wird, und wo sie produziert werden kann. Daraus ergibt sich die Zuordnung von Attributen zu den Grammatiksymbolen und die Einteilung in ererbte und synthetisierte Attribute.

Erstellen der Attributierungsregeln: Die Attributierungsregeln können wiederum eingeteilt werden in solche, die die eigentlichen Berechnungen durchführen und solche, die nur dem Datenfluß dienen.

Umformen in Aktionen (optional): Ist ein LL(1)-Übersetzungsschema gesucht, müssen die Attributierungsregeln an den entsprechenden Stellen der Grammatikregeln untergebracht werden. Und zwar dürfen synthetisierte Attribute erst nach dem Nonterminal verwendet werden, das sie synthetisiert, die Zuweisung an ererbte Attribute muß vor dem Nonterminal erfolgen. Der Datenfluß erfolgt also nur von links nach rechts.

Zum besseren Verständnis kann man die Grammatikregeln für ein Nonterminal als Prozedur betrachten, ererbte Attribute als Eingangsparameter, und synthetisierte Attribute als Ergebnisparameter (LL(1)-Übersetzungsschemata werden oft tatsächlich so implementiert). Die Datenübergabe ist nur über die Parameter (also die Attribute) möglich (also nicht über z.B. nichtlokale Variable). Jedem Attribut kann nur einmal ein Wert zugewiesen werden (single-assignment-Eigenschaft), Überschreiben ist nicht möglich, ähnlich wie in funktionalen und Logik-Programmiersprachen. Kombinierte Ein/Ausgabe-Parameter sind zu ersetzen durch ein ererbtes und ein synthetisiertes Attribut. In Leer-Regeln muß dann eine Zuweisung vom ererbten zum synthetisierten Attribut gemacht werden.