



Doctoral Thesis

A new approach to formal language definition and its application to Oberon

Author(s):

Odersky, Martin

Publication Date:

1989

Permanent Link:

<https://doi.org/10.3929/ethz-a-000522620> →

Rights / License:

[In Copyright - Non-Commercial Use Permitted](#) →

This page was generated automatically upon download from the [ETH Zurich Research Collection](#). For more information please consult the [Terms of use](#).

Diss. ETH No. 8938

A New Approach to Formal Language Definition
and its Application to Oberon

A dissertation submitted to the
SWISS FEDERAL INSTITUTE OF TECHNOLOGY ZURICH
(ETH Zürich)

for the degree of
Doctor of Technical Sciences

presented by
MARTIN ODERSKY
Dipl. Math. LMU München
born September 5, 1958
citizen of the Federal Republic of Germany

accepted on the recommendation of
Prof. Dr. N. Wirth, examiner
Prof. Dr. H.P. Mössenböck, co-examiner

Zürich 1989

Abstract

This thesis presents a new method and notation for the definition of the syntax of programming languages. With syntax, we mean the set of all rules which determine whether a text belongs to a given language. A subset of these rules is commonly and successfully defined with context-free grammars. For other areas, like scope- or type-rules, there does not yet exist a satisfactory formalism.

Our method, CADET, is based on the idea that predicate calculus can be employed for the characterization of legal derivation trees. A context-free grammar is used for defining a superset of a given language, which is then restricted by imposing legality criteria on derivation trees. The criteria are expressed with logic predicates which we call context-rules. A text belongs to the defined language if and only if its derivation tree is a model of all context-rules.

A CADET language definition is precise and often considerably shorter than definitions with other formal methods. Its structure can be closely related to descriptions formulated in natural language.

The first part of this thesis contains a detailed introduction into the CADET notation and a critical comparison with other methods. In the second part, we present a commented CADET definition of the programming language Oberon [Wirth 88]. The results obtained from this first large application of our method are quite encouraging: the complete Oberon definition has a size of approximately 700 lines.

In the final part of the thesis we discuss a rule checking program which generates from a CADET specification the analysis part of a compiler. The program is used for determining whether test-inputs belong to the defined language. Test runs can be a valuable help in locating and correcting errors in a definition.

Zusammenfassung

In dieser Arbeit stellen wir eine neue Methode und Notation zur Definition der Syntax von Programmiersprachen vor. Unter Syntax verstehen wir alle Regeln, die festlegen, ob ein Text zu einer gegebenen Sprache gehört. Eine Untermenge dieser Regeln wird schon seit langem mit kontextfreien Grammatiken erfolgreich beschrieben. Für andere Gebiete, wie zum Beispiel Deklarations- und Typenregeln, existiert bis jetzt noch kein allseits befriedigender Formalismus.

Unsere Methode, CADET, basiert auf der Überlegung, dass das Prädikatenkalkül zur Charakterisierung legaler Ableitungsbäume herangezogen werden kann. Eine kontextfreie Grammatik beschreibt zunächst eine Obermenge einer gegebenen Sprache. Diese Obermenge wird dann eingeschränkt, indem Legalitätskriterien an Ableitungsbäume gestellt werden. Die Kriterien werden mit Logik-Prädikaten, die wir Kontextregeln nennen, formuliert. Ein Text gehört zu der definierten Sprache genau dann wenn sein Ableitungsbaum ein Modell aller Kontextregeln ist.

Eine in CADET formulierte Sprachdefinition ist exakt und oft wesentlich kürzer als Definitionen mit anderen formalen Methoden. Sie hat zudem den Vorteil, dass sie in ihrer Struktur eng an eine natürlichsprachige Beschreibung angelehnt sein kann.

Der erste Teil dieser Arbeit enthält eine detaillierte Einführung in die CADET Notation und einen kritischen Vergleich mit anderen formalen Methoden. Im zweiten Teil stellen wir eine kommentierte Definition der Programmiersprache Oberon [Wirth 88] vor. Die Resultate, die aus dieser ersten grossen Anwendung unserer Methode gewonnen wurden, sind recht ermutigend: Die vollständige Oberon-Definition umfasst circa 700 Zeilen.

Der abschliessende Teil dieser Arbeit befasst sich mit einem Regelprüf-Programm, das aus einer CADET Spezifikation den Analyseteil eines Compilers erzeugt. Das Programm dient dazu, Testeingaben auf Zugehörigkeit zur definierten Sprache zu überprüfen. Diese Testläufe können dazu beitragen, Fehler in einer Spezifikation zu erkennen und zu korrigieren.