

Information technology — Syntactic Metalanguage — Extended BNF

ICS 35.060

National foreword

This British Standard reproduces verbatim ISO/IEC 14977 : 1996 and implements it as the UK national standard.

The UK participation in its preparation was entrusted to Technical Committee IST/5, Programming Languages, their environments and system software interfaces, which has the responsibility to:

- aid enquirers to understand the text;
- present to the responsible international/European committee any enquiries on the interpretation, or proposals for change, and keep the UK interests informed;
- monitor related international and European developments and promulgate them in the UK.

A list of organizations represented on this committee can be obtained on request to its secretary.

Cross-references

The British Standards which implement international or European publications referred to in this document may be found in the BSI Standards Catalogue under the section entitled 'International Standards Correspondence Index', or by using the 'Find' facility of the BSI Standards Electronic Catalogue.

A British Standard does not purport to include all the necessary provisions of a contract. Users of British Standards are responsible for their correct application.

Compliance with a British Standard does not of itself confer immunity from legal obligations.

Summary of pages

This document comprises a front cover, an inside front cover, the ISO/IEC title page, pages ii to viii, pages 1 to 12, an inside back cover and a back cover.

Amendments issued since publication

Amd. No.	Date	Text affected

This British Standard, having been prepared under the direction of the DISC Board, was published under the authority of the Standards Board and comes into effect on 15 February 1998

© BSI 1998

ISBN 0 580 29189 8

INTERNATIONAL STANDARD

ISO/IEC 14977

First edition
1996-12-15

Information technology — Syntactic metalanguage — Extended BNF

Technologies de l'information — Métalangage syntaxique — BNF étendu



Reference number
ISO/IEC 14977:1996(E)



Documents
Document Sharing Hub

Contents	Page
Foreword	iv
Introduction	v
1 Scope	1
2 Normative references	1
3 Definitions	1
4 The form of each syntactic element of Extended BNF	1
4.1 General	2
4.2 Syntax	2
4.3 Syntax-rule	2
4.4 Definitions-list	2
4.5 Single-definition	2
4.6 Syntactic-term	2
4.7 Syntactic exception	2
4.8 Syntactic-factor	2
4.9 Integer	2
4.10 Syntactic-primary	2
4.11 Optional-sequence	3
4.12 Repeated sequence	3
4.13 Grouped sequence	3
4.14 Meta-identifier	3
4.15 Meta-identifier-character	3
4.16 Terminal-string	3
4.17 First-terminal-character	3
4.18 Second-terminal-character	3
4.19 Special-sequence	3
4.20 Special-sequence-character	3
4.21 Empty-sequence	3
4.22 Further examples	3

5	The symbols represented by each syntactic element	3
5.1	General	3
5.2	Terminal-string	4
5.3	Meta-identifier	4
5.4	Grouped-sequence	4
5.5	Optional-sequence	4
5.6	Repeated-sequence	4
5.7	Syntactic-factor	4
5.8	Syntactic-term	4
5.9	Single-definition	5
5.10	Definitions-list	5
5.11	Special-sequence	5
5.12	Empty-sequence	5
6	Layout and Comments	5
6.1	General	5
6.2	Terminal-character	5
6.3	Gap-free-symbol	6
6.4	Gap-separator	6
6.5	Commentless-symbol	6
6.6	Comment-symbol	6
6.7	Bracketed-textual-comment	6
7	The representation of each terminal-character in Extended BNF	6
7.1	General	6
7.2	Letters and digits	6
7.3	Other terminal characters	6
7.4	Alternative representations	6
7.5	Other-character	7
7.6	Gap-separator	7
7.7	Terminal-characters represented by a pair of characters	8
7.8	Invalid character sequences	8
8	Examples	8
8.1	The syntax of Extended BNF	8
8.2	Extended BNF used to define itself informally	10
8.3	Extended BNF defined informally	10

Annexes

A	Two-level grammars	11
B	Bibliography	12

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75% of the national bodies casting a vote.

International Standard ISO/IEC 14977 was prepared by BSI (as BS 6154) and was adopted, under a special “fast-track procedure”, by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, in parallel with its approval by national bodies of ISO and IEC.

Annexes A and B of this International Standard are for information only.

Introduction

A syntactic metalanguage is an important tool of computer science. The concepts are well known, but many slightly different notations are in use. As a result syntactic metalanguages are still not widely used and understood, and the advantages of rigorous notations are unappreciated by many people.

Extended BNF brings some order to the formal definition of a syntax and will be useful not just for the definition of programming languages, but for many other formal definitions.

Since the definition of the programming language Algol 60 (Naur, 1960) the custom has been to define the syntax of a programming language formally. Algol 60 was defined with a notation now known as BNF or Backus-Naur Form. This notation has proved a suitable basis for subsequent languages but has frequently been extended or slightly altered. The many different notations are confusing and have prevented the advantages of formal unambiguous definitions from being widely appreciated. The syntactic metalanguage *Extended BNF* described in this standard is based on Backus-Naur Form and includes the most widely adopted extensions.

Syntactic metalanguages

A syntactic metalanguage is a notation for defining the syntax of a language by use of a number of rules. Each rule names part of the language (called a non-terminal symbol of the language) and then defines its possible forms. A terminal symbol of the language is an atom that cannot be split into smaller components of the language. A syntactic metalanguage is useful whenever a clear formal description and definition is required, e.g. the format for references in papers submitted to a journal, or the instructions for performing a complicated task.

A formal syntax definition has three distinct uses:

- a) it names the various syntactic parts (i.e. non-terminal symbols) of the language;
- b) it shows which sequences of symbols are valid sentences of the language;
- c) it shows the syntactic structure of any sentence of the language.

The need for a standard syntactic metalanguage

Without a standard syntactic metalanguage every programming language definition starts by specifying the metalanguage used to define its syntax. This causes various problems:

Many different notations — It is unusual for two different programming languages to use the same metalanguage. Thus human readers are handicapped by having to learn a new metalanguage before they can study a new language.

Concepts not widely understood — The lack of a standard notation hinders the use of rigorous unambiguous definitions.

Imperfect notations — Because a metalanguage needs to be defined for every programming language, almost inevitably, the metalanguage contains defects. For example errors occurred in the drafting of RTL/2 (BS5904) and CORAL 66 (BS5905) because the metalanguages could not be typed easily.

Special purpose notations — A metalanguage defined for a particular programming language is often simplified by taking advantage of special features in the language to be defined. However, the metalanguage is then unsuitable for other programming languages.

Few general syntax processors — The multiplicity of syntactic metalanguages has limited the availability of computer programs to analyse and process syntaxes, e.g. to list a syntax neatly, to make an index of the symbols used in the syntax, to produce a syntax-checker for programs written in the language.

In practice experienced readers have little difficulty in picking up and learning a new notation, but even so the differences obscure mutual understanding and hinder communication. A standard metalanguage enables more people to crystallize vague ideas into an unambiguous definition. It is also useful because other people needing to provide formal definitions no longer need to reinvent similar concepts.

The objectives to be satisfied

It is desirable that a standard syntactic metalanguage should be:

- a) concise, so that languages can be defined briefly and thus be more easily understood;
- b) precise, so that the rules are unambiguous;
- c) formal, so that the rules can be parsed, or otherwise processed, by a computer when required;
- d) natural, so that the notation and format are relatively simple to learn and understand, even for those who are not themselves language designers; (The meaning of a symbol should not be surprising. It should also be possible to define the syntax of a language in a way that helps to indicate the meaning of the constructions.)
- e) general, so that the notation is suitable for many purposes including the description of many different languages;
- f) simple in its character set and with a notation that avoids, as far as is practicable, using characters that are not generally available on standard keyboards (both typewriters and computer terminals) so that the rules can be typed and can be processed by computer programs;
- g) self describing, so that the notation is able to describe itself;