

Network Working Group
Internet-Draft
Updates: [5234](#) (if approved)
Intended Status: Standards Track
Expires: April 1, 2017

S. Leonard
Penango, Inc.
September 28, 2016

Comprehensive Core Rules and References for ABNF
draft-seantek-abnf-more-core-rules-07

Abstract

This document extends the base definition of ABNF (Augmented Backus-Naur Form) to include a reference syntax, along with core rules that provide comprehensive support for certain symbols related to ASCII.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 1, 2017.

Copyright Notice

Copyright (c) 2016 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

1. Introduction

Augmented Backus-Naur Form (ABNF) [[RFC5234](#)] is a formal syntax that is popular among many Internet specifications. Many Internet documents employ this syntax along with the Core Rules defined in [Appendix B.1 of \[RFC5234\]](#). However, the Core Rules do not specify many symbols in the ASCII range that are also needed by these relying documents, forcing document authors to define them as local rules. Sometimes different documents define these common symbols in different ways, resulting in confusion or incompatibility when the rules are misread or are combined with other sets of rules. Furthermore, [[RFC5234](#)] does not clarify whether referencing [[RFC5234](#)] for ABNF automatically defines its Core Rules.

[RFC5234] also lacks a syntax for referring to rules from other specifications. Instead, authors have been required to name the rules and sources in the specification prose. While this method has served authors well, it has hampered machine-readable ABNF efforts for services such as syntax highlighting, automatic grammar checking, and compiling into target computer languages.

This document addresses these problems by introducing a reference syntax for rules taken from other ABNF grammars, as well as an enhanced set of "Core Rules" based on ASCII that are usable without needing to be referenced.

2. Comprehensive Core Rule Update

This document provides Core Rules that include comprehensive support for certain symbols, namely DELETE (DEL) and the C0 controls in [[ASCII86](#)], which for purposes of this document is equivalent to [[RFC0020](#)]. The Comprehensive Core Rules are listed in [Appendix A](#) as a drop-in replacement for the Core Rules of [[RFC5234](#)].

3. Reference Syntax

The purpose of reference syntax is to provide a uniform way to refer to rules in other ABNF grammars, without needing to "import", "recognize", or "take" all of the rules from those ABNF grammars into the subject grammar. The syntax in this section essentially replaces the verbiage: "{RULE} is taken from {[RFCXXXX]}" in text that describes the ABNF. This verbiage traditionally has appeared in the specification prose adjacent to ABNF, in ABNF prose-val productions, or in ABNF comments. The varying verbiage has made it difficult for both human readers and machine parsers to validate the ABNF. At the same time, the presence of such verbiage in the vast majority of published ABNF specifications in the RFC series demonstrates the need for a general-purpose referencing facility.

To reference a rule in another ABNF grammar, use the syntax `rulename@REF`. The referenced rule resolves to terminal values in the context of the referenced ABNF grammar. The following enhancement to [\[RFC5234\]](#) permits this referenced-rule syntax as a change to the `<element>` production:

```
element      = rulename [ "@" ruleref ] / group / option
               char-val / num-val / prose-val

ruleref      = ref-doc / ref-path

ref-doc      = "[" 1*(SP / %x21-5A / %x5C / %x5E-7E) "]"
               ; bracketed string of SP / VCHAR without [ or ]
               ; TODO: are leading and trailing SP ok?

ref-path     = "<" 1*(SP / %x21-3B / "<" / "=" / ">" / %x3F-7E) ">"
               ; angle-bracketed string of SP / VCHAR;
               ; < and > permitted with prefix
```

In a referenced-rule production the `<rulename>` production preceding the `"@"` specifies the name of the rule in the reference containing ABNF. The `<ruleref>` production following the `"@"` specifies the reference containing the rule. This specification does not define the semantics if a rule is found in a grammar that is not ABNF. (This limitation is because rule names in ABNF are case-insensitive and drawn from a limited character repertoire. Some rule names in other BNFs may be unreachable or ambiguous, even though the productions named by the rules are linguistically compatible.)

The `<ref-doc>` production is a document reference of a resource containing ABNF. The term "document reference" refers to "the document containing this ABNF (i.e., the instance of these production rules)". In IETF-related publications, `ref-doc` conveniently is of the same form as document references, such as "[\[RFC1605\]](#)". [[NB: in this draft:]] Arbitrary spaces (not tabs) are allowed; interior brackets `"["` and `"]"` are prohibited.

The `<ref-path>` production is a path to a resource containing ABNF. The ABNF might be in a text file or MIME entity, for example. The intent of this production is to accommodate file paths and Uniform Resource Identifiers [\[RFC3986\]](#) (including fragment identifier components), but this specification imposes no requirement to validate conformance to those syntaxes. If the characters `"<"` or `">"` are present in the path, they are syntactically distinguishable from the `ref-path` terminators by being escaped with a preceding backslash. The assumption is that `ref-doc` rather than `ref-path` productions will be used in published standards documents.

[[NB: in this draft:]] This document only proposes referenced-rule syntax in <element> productions, that is, on the right-hand side of a rule definition. The referenced-rule syntax is not proposed to appear on the left-hand side at this time.

Stylistically, authors are encouraged to put reference syntax at the top of a list of rules, and to limit usage of the reference syntax to the single element of a rule definition. For example:

```
You      = Edward@[FFIV]
spoony   = spoony@[FFIV]
bard     = bard@[FF-JOB-CLASS]
chara    = Tellah@[FFIV]

insult   = chara ":" You spoony bard "!"
```

[Appendix B](#) provides some tips on how to think about combining referenced ABNF rules with the subject ABNF grammar of a specification.

5. Effects on [RFC 5234](#)

Formally, this document updates [\[RFC5234\]](#) but does not modify it in situ. Authors need to reference this document if they want to include these enhancements; bare references to [\[RFC5234\]](#) do not include this specification (or, for that matter, [\[RFC7405\]](#)). This directive follows a model whereby document authors can choose whether to invoke particular enhancements to ABNF. As time goes on, the IETF can determine how often these enhancements are invoked, and can decide whether to include them as part of a revision to the base [\[RFC5234\]](#).

A bare reference to this document invokes the reference syntax enhancement and the Core Rules of [Appendix A](#) (i.e., the Core Rules do not have to use reference syntax).

[Appendix A](#) of this document is meant to mirror [Appendix B.1 of \[RFC5234\]](#); therefore, concurrently referencing [Appendix B.1 of \[RFC5234\]](#) is redundant yet harmless. Document authors who reference this document should use the rules of [Appendix A](#), and should not attempt to redefine or provide incremental alternatives to them (except for backwards compatibility with prior documents).

6. IANA Considerations This document implies no IANA considerations.

7. Security Considerations

While the Core Rules themselves may not be security-relevant, the use of such control characters could very well be security-relevant,

because they may trigger special functions on various devices, while being invisible in other contexts.

Unfortunately security is relevant to the reference syntax in this document. Using the reference syntax facilitates automated processing of ABNF. A malicious source could supply different ABNF as an attack vector on a compiled program. Furthermore, referring to a mutable resource (e.g., a document series such as BCP) permits the resource to change its contained ABNF, which may be well-intentioned but have side-effects when combined with the referring ABNF. Authors should stick to persistent, durable references, whose integrity can be validated easily.

8. Acknowledgements

The author wishes to thank Paul Kyzivat and Chris Newman for ongoing discussion and comments during the development of this draft.

9. References

9.1. Normative References

- [ASCII86] American National Standards Institute, "Coded Character Set -- 7-bit American Standard Code for Information Interchange", ANSI X3.4, 1986.
- [RFC0020] Cerf, V., "ASCII format for network interchange", [RFC 20](#), October 1969.
- [RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, [RFC 5234](#), January 2008.

9.2. Informative References

- [UNICODE] The Unicode Consortium, "The Unicode Standard, Version 9.0.0", The Unicode Consortium, August 2016.
- [RFC1345] Simonsen, K., "Character Mnemonics and Character Sets", [RFC 1345](#), June 1992.
- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, [RFC 3986](#), January 2005.
- [RFC5198] Klensin, J. and M. Padlipsky, "Unicode Format for Network Interchange", [RFC 5198](#), March 2008.

[Appendix A](#). Comprehensive Core Rules

Certain basic rules are in uppercase, such as SP, HTAB, CRLF, DIGIT, ALPHA, etc.

ALPHA	=	%x41-5A / %x61-7A ; A-Z / a-z
BIT	=	"0" / "1"
CHAR	=	%x01-7F ; any 7-bit US-ASCII character, ; excluding NUL
CR	=	%x0D ; carriage return
CRLF	=	CR LF ; Internet standard newline
CTL	=	%x00-1F / %x7F ; controls
DIGIT	=	%x30-39 ; 0-9
DQUOTE	=	%x22 ; " (Double Quote)
HEXDIG	=	DIGIT / "A" / "B" / "C" / "D" / "E" / "F"
HTAB	=	%x09 ; horizontal tab
LF	=	%x0A ; linefeed
LWSP	=	*(WSP / CRLF WSP) ; Use of this linear-white-space rule ; permits lines containing only white ; space that are no longer legal in ; mail headers and have caused ; interoperability problems in other ; contexts. ; Do not use when defining mail ; headers and use with caution in ; other contexts.
OCTET	=	%x00-FF ; 8 bits of data
SP	=	%x20
VCHAR	=	%x21-7E

; visible (printing) characters

WSP	=	SP / HTAB	
		; white space	
NUL	=	%d0	
SOH	=	%d1	
STX	=	%d2	
ETX	=	%d3	
EOT	=	%d4	
ENQ	=	%d5	
ACK	=	%d6	
BEL	=	%d7	
BS	=	%d8	
HT	=	%d9	; also defined as HTAB
VT	=	%d11	
FF	=	%d12	; (literally used in every RFC)
S0	=	%d14	
SI	=	%d15	
DLE	=	%d16	
DC1	=	%d17	
DC2	=	%d18	
DC3	=	%d19	
DC4	=	%d20	
NAK	=	%d21	
SYN	=	%d22	
ETB	=	%d23	
CAN	=	%d24	
EM	=	%d25	
SUB	=	%d26	
ESC	=	%d27	
FS	=	%d28	
GS	=	%d29	
RS	=	%d30	
US	=	%d31	
DEL	=	%d127	
ASCII	=	%x00-7F	
C0	=	%x00-1F	
G0	=	VCHAR	; 94-set

[Appendix B](#). Guidance for Automated Referenced Rule Conversion

ABNF is a formal notation for describing the syntax of languages used in (Internet-connected) computing. Emphasis is therefore placed on human interpretation of ABNF grammars in the context of prose specifications, over formal computer languages that require machine tools to interpret. Nevertheless, as a formal syntactic metalanguage, tools can interpret ABNF grammars and validate conformance of grammars to ABNF as well as conformance of language instances to ABNF-defined grammars. This informative appendix provides guidance on how an automated tool might convert between referenced rules and terminal values.

[[TODO: Discuss and put content here.]]

Assume the existence of an "ABNF extractor", a tool that takes as input a document, and provides as output a stream of ABNF conforming to the <rulelist> production of ABNF.

Extract the document reference from the <refrule>.

Match the document reference to a reference in the References section of an RFC or conforming Internet-Draft.

Parse the reference for an identifier that can be dereferenced, e.g., a file path or URI.

Dereference the identifier.

Use the ABNF extractor to extract ABNF from the dereferenced document.

Identify the <rulename> that matches the <rulename> from the <refrule>.

If the ABNF in the dereferenced document is resolved to terminal values, it is resolved in its own context, not in the context of the original <refrule>'s ABNF.

Author's Address

Sean Leonard
Penango, Inc.
5900 Wilshire Boulevard
21st Floor
Los Angeles, CA 90036
USA

EMail: dev+ietf@seantek.com

URI: <http://www.penango.com/>