Network Working Group
Internet-Draft
Intended status: Informational
Expires: 8 January 2026

C. Bormann
Universit辰t Bremen TZI

7 July 2025

Managing CBOR codepoints in Internet-Drafts
draft-bormann-cbor-draft-numbers-06

Abstract

CBOR-based protocols often make use of numbers allocated in a registry. During development of the protocols, those numbers may not yet be available. This impedes the generation of data models and examples that actually can be used by tools.

This short draft proposes a common way to handle these situations, without any changes to existing tools. Also, in conjunction with the application-oriented EDN literal e'', a further reduction in editorial processing of CBOR examples around the time of approval can be achieved.

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 https://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 8 January 2026.

Copyright Notice

Table of Contents

1. Introduction

(Please see abstract.) [STD94] [I-D.ietf-cbor-edn-e-ref]

This document uses "EDN" (extended diagnostic notation) as a shorthand for "CBOR diagnostic notation" as defined in Section 8 of RFC 8949 [STD94], extended in Appendix G of [RFC8610], and updated in [I-D.ietf-cbor-edn-literals].

2. The Problem

A CBOR-based protocol might want to define a structure using the Concise Data Definition Language (CDDL) [RFC8610][RFC9682][RFC9165][RFC9741], like that in Figure 1 (based on [RFC9290]):

```
problem-details = {
  ? &(title: -1) => oltext
  ? &(detail: -2) => oltext
  ? &(instance: -3) => ~uri
  ? &(response-code: -4) => uint .size 1
  ? &(base-uri: -5) => ~uri
  ? &(base-lang: -6) => tag38-ltag
  ? &(base-rtl: -7) => tag38-direction
  / ... /
  * (uint .feature "extension") => any
}
```

Figure 1: CDDL data model, final form

The key numbers shown in this structure are likely to be intended for allocation in an IANA section.

The key numbers will be used in an example in the specification such as shown in Figure 2.

```
{
  / title /         -1: "title of the error",
  / detail /        -2: "detailed information about the error",
  / instance /      -3: "coaps://pd.example/FA317434",
  / response-code / -4: 128, / 4.00 /
  4711: {
     / ... /
  }
}
```

Figure 2: EDN example, final form

However, during development, these numbers are not yet fixed; they are likely to move around as parts of the specification are added or deleted.

3. The Anti-Pattern

What not to do during development:

```
problem-details = {
  ? "title" => oltext
  ? "detail" => oltext
  ? "instance" => ~uri
  ? "response-code" => uint .size 1
  ? "base-uri" => ~uri
  ? "base-lang" => tag38-ltag
  ? "base-rtl" => tag38-direction
  / ... /
  * (uint .feature "extension") => any
}
```

Figure 3: CDDL data model, muddled form

```
{
  "title": "title of the error",
  "detail": "detailed information about the error",
  "instance-code": "coaps://pd.example/FA317434",
  "response-code": 128, / 4.00 /
  4711: {
     / ... /
  }
}
```

Figure 4: EDN example, muddled form

This makes the model and the examples compile/check out even before having allocated the actually desired numbers, but it also leads to several problems:

* It becomes hard to assess what the storage/transmission cost of these structures will be.

* What is being checked in the CI (continuous integration) for the document is rather different from the final form.

* Draft implementations trying to make use of these provisional structures have to cater for text strings, which may not actually be needed in the final form (which might expose specification bugs once numbers are used, too late in the process).

* The work needed to put in the actual numbers, once allocated, is significant and error-prone.

* It is not certain the CI system used during development can interact with the RFC editor’s way of editing the document for publication.

4. What to do during spec development

To make the transition to a published document easier, the document is instead written with the convention demonstrated in the following example:

```
// This document uses the keys for a map as an example.  Other such
// constructs involving assigned numbers might also require temporary
// values for exposition in a specification, e.g., CBOR tags.  For
// the sake of keeping this document short, examples for these are
// not given.


// Including examples of other things that generate the need for
// temporary numbers, like tags, would be good.

problem-details = {
  ? &(title-CPA: -1) => oltext
  ? &(detail-CPA: -2) => oltext
  ? &(instance-CPA: -3) => ~uri
  ? &(response-code-CPA: -4) => uint .size 1
  ? &(base-uri-CPA: -5) => ~uri
  ? &(base-lang-CPA: -6) => tag38-ltag
  ? &(base-rtl-CPA: -7) => tag38-direction
  / ... /
  * (uint .feature "extension") => any
}
```

Figure 5: CDDL data model, development form

CPA is short for "code point allocation", and is a reliable search key for finding the places that need to be updated after allocation.

```
// An earlier concept for this draft used TBD in place of CPA, as do
// many draft specifications being worked on today.  TBD is better
// recognized than CPA, but also could be misunderstood to mean
// further work by the spec developer is required.  A document
// submitted for publication should not really have "TBD" in it.
```

In the IANA section, the table to go into the registry is prepared as follows:

Key value	Name	CDDL Type	Brief description	Reference
CPA-1	title	text / tag38	short, human-readable summary of the problem shape	RFC XXXX
CPA-2	detail	text / tag38	human-readable explanation specific to this occurrence of the problem	RFC XXXX
CPA-3	instance	~uri	URI reference identifying specific occurrence of the problem	RFC XXXX
CPA-4	response-code	uint .size 1	CoAP response code	RFC XXXX
CPA-5	base-uri	~uri	Base URI	RFC XXXX
CPA-6	base-lang	tag38-ltag	Base language tag (see tag38)	RFC XXXX
CPA-7	base-rtl	tag38-direction	Base writing direction (see tag38)	RFC XXXX

Table 1: IANA table, development form

The provisionally made up key numbers will then be used in an example in the specification such as:

```
{
  / title-CPA /         -1: "title of the error",
  / detail-CPA /        -2: "detailed information about the error",
  / instance-CPA /      -3: "coaps://pd.example/FA317434",
  / response-code-CPA / -4: 128, / 4.00 /
  4711: {
     / ... /
  }
}
```

Figure 6: EDN example, development form

A "removeInRFC" note in the draft points the RFC editor to the present document so the RFC editor knows what needs to be done at which point. In the publication process, it is easy to remove the -CPA suffixes and CPA prefixes for the RFC editor while filling in the actual IANA allocated numbers and removing the note.

Note that in Table 1, the first column uses the name "CPA-1" for a value that in the rest of the document is assumed to be "-1" (and indicating a preference by the document author for this number); IANA as well as the designated experts involved are expected by the present document to decode this notation.

A "removeInRFC" note to the RFC Editor for Table 1 could have this approximate contents: This document uses the CPA (code point allocation) convention described in [I-D.bormann-cbor-draft-numbers]. For each entry, please remove the prefix "CPA" from the indicated value of the column <REG_COLUMN>, and replace the residue with the value assigned by IANA; perform the same substitution for all other occurrences of the prefix "CPA" in the document. Finally, please remove this note.

A "removeInRFC" note to the RFC Editor for Figure 6 could have this approximate contents: This document uses the CPA (code point allocation) convention described in [I-D.bormann-cbor-draft-numbers]. For each item whose key textual identifier has suffix "-CPA", please remove the suffix. Then, consider the residue of the suffix removal, and replace the key numeric identifier with the value assigned by IANA in the <REG_COLUMN_1> of the registry <REG_NAME>, for the entry where the value in the <REG_COLUMN_2> is equal to the residue. Finally, please remove this note.

The RFC editor with IANA would then execute these instructions as shown in Table 2 and Figure 7 (assuming the unlikely case that all numbers allocated are ten times the number proposed):

Key value	Name	CDDL Type	Brief description	Reference
-10	title	text / tag38	short, human-readable summary of the problem shape	RFC XXXX
-20	detail	text / tag38	human-readable explanation specific to this occurrence of the problem	RFC XXXX
-30	instance	~uri	URI reference identifying specific occurrence of the problem	RFC XXXX
-40	response-code	uint .size 1	CoAP response code	RFC XXXX
-50	base-uri	~uri	Base URI	RFC XXXX
-60	base-lang	tag38-ltag	Base language tag (see tag38)	RFC XXXX
-70	base-rtl	tag38-direction	Base writing direction (see tag38)	RFC XXXX

Table 2: IANA table, final form

```
{
  / title /         -10: "title of the error",
  / detail /        -20: "detailed information about the error",
  / instance /      -30: "coaps://pd.example/FA317434",
  / response-code / -40: 128, / 4.00 /
  4711: {
     / ... /
  }
}
```

Figure 7: EDN example, final form

4.1. Documents with Significant Generated Content Depending on Assignments

Many documents have examples (which might even involve signatures over the contents) that depend on the assignments in more than the trivial way shown above, and regenerating them may not be easy for the RFC editor to do.

Therefore, for these documents we need another step involving the authors:

Immediately after allocation, but before the RFC-Editor EDIT step, the authors need to regenerate these examples and other generated content depending on the exact allocations.

In the current process, allocation is usually done after IESG approval, after IANA action, so we would need to halt the EDIT step for this regeneration.

Alternatively, we could be more aggressive in invoking some kind of IANA Early Allocation process, near the end of the IESG review. One way to do this with current tooling and process is to perform a late form of actual IANA "Early" Allocation. Or we could amend [BCP9] and/or [BCP100] in a more fundamental way.

// We probably need an indicator in addition to CPA that signifies an
// example or other text must be regenerated (vs. simply be updated
// by IANA) when proposed numbers are updated by IANA.

4.2. Reducing the editorial workload with CDDL definitions

[I-D.ietf-cbor-edn-e-ref] defines a EDN application extension that allows EDN to reference constants defined in a CDDL model, the e'' application extension.

If the draft contains a CDDL model that includes definitions of constants that may then be used in EDN, the use of e'' constant references makes it unnecessary to change the constant value in the example when final values are defined for these constants. (This application extension also can make the EDN more readable and less distracting, replacing constructs such as

/ title-CPA / -1

by

e'title'

which removes the need to mention "CPA" and to provide a potentially distracting copy of the value assignment in the example.)

The document using the e'' application extension may want to provide a CDDL file with provisional assignments, as in:

```
; CPA: not yet assigned by IANA, subject to change during allocation
title = -1
```

This file should be clearly labeled as CPA, i.e., not yet assigned and subject to change during allocation.

5. IANA Considerations

This document makes no requests of IANA. However, it specifies a procedure that can be followed during draft development that has a specific role for IANA and the interaction between RFC editor and IANA at important points during this development. This procedure is intended to be as little of an onus as possible, but that is the author's assessment only. IANA feedback is therefore requested.

6. Security considerations

The security considerations of [RFC8610] and [STD94] apply.

7. References

7.1. Normative References

[I-D.bormann-cbor-draft-numbers]
Bormann, C., "Managing CBOR codepoints in Internet-Drafts", Work in Progress, Internet-Draft, draft-bormann-cbor-draft-numbers-05, 1 March 2025, <https://datatracker.ietf.org/doc/html/draft-bormann-cbor-draft-numbers-05>.

[I-D.ietf-cbor-edn-e-ref]
Bormann, C., "External References to Values in CBOR Diagnostic Notation (EDN)", Work in Progress, Internet-Draft, draft-ietf-cbor-edn-e-ref-02, 2 July 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-cbor-edn-e-ref-02>.

[I-D.ietf-cbor-edn-literals]
Bormann, C., "CBOR Extended Diagnostic Notation (EDN)", Work in Progress, Internet-Draft, draft-ietf-cbor-edn-

literals-17, 12 May 2025,
<https://datatracker.ietf.org/doc/html/draft-ietf-cbor-edn-literals-17>.

[RFC8610] Birkholz, H., Vigano, C., and C. Bormann, "Concise Data Definition Language (CDDL): A Notational Convention to Express Concise Binary Object Representation (CBOR) and JSON Data Structures", RFC 8610, DOI 10.17487/RFC8610, June 2019, <https://www.rfc-editor.org/rfc/rfc8610>.

[RFC9165] Bormann, C., "Additional Control Operators for the Concise Data Definition Language (CDDL)", RFC 9165, DOI 10.17487/RFC9165, December 2021, <https://www.rfc-editor.org/rfc/rfc9165>.

[RFC9682] Bormann, C., "Updates to the Concise Data Definition Language (CDDL) Grammar", RFC 9682, DOI 10.17487/RFC9682, November 2024, <https://www.rfc-editor.org/rfc/rfc9682>.

[RFC9741] Bormann, C., "Concise Data Definition Language (CDDL): Additional Control Operators for the Conversion and Processing of Text", RFC 9741, DOI 10.17487/RFC9741, March 2025, <https://www.rfc-editor.org/rfc/rfc9741>.

[STD94] Internet Standard 94,
<https://www.rfc-editor.org/info/std94>.
At the time of writing, this STD comprises the following:

Bormann, C. and P. Hoffman, "Concise Binary Object Representation (CBOR)", STD 94, RFC 8949, DOI 10.17487/RFC8949, December 2020, <https://www.rfc-editor.org/info/rfc8949>.

7.2. Informative References

[BCP100] Best Current Practice 100,
<https://www.rfc-editor.org/info/bcp100>.
At the time of writing, this BCP comprises the following:

Cotton, M., "Early IANA Allocation of Standards Track Code Points", BCP 100, RFC 7120, DOI 10.17487/RFC7120, January 2014, <https://www.rfc-editor.org/info/rfc7120>.

[BCP9] Best Current Practice 9,
<https://www.rfc-editor.org/info/bcp9>.
At the time of writing, this BCP comprises the following:

Bradner, S., "The Internet Standards Process -- Revision 3", BCP 9, RFC 2026, DOI 10.17487/RFC2026, October 1996, <https://www.rfc-editor.org/info/rfc2026>.

Dusseault, L. and R. Sparks, "Guidance on Interoperation and Implementation Reports for Advancement to Draft Standard", BCP 9, RFC 5657, DOI 10.17487/RFC5657, September 2009, <https://www.rfc-editor.org/info/rfc5657>.

Housley, R., Crocker, D., and E. Burger, "Reducing the Standards Track to Two Maturity Levels", BCP 9, RFC 6410, DOI 10.17487/RFC6410, October 2011, <https://www.rfc-editor.org/info/rfc6410>.

Resnick, P., "Retirement of the "Internet Official Protocol Standards" Summary Document", BCP 9, RFC 7100, DOI 10.17487/RFC7100, December 2013, <https://www.rfc-editor.org/info/rfc7100>.

Kolkman, O., Bradner, S., and S. Turner, "Characterization of Proposed Standards", BCP 9, RFC 7127, DOI 10.17487/RFC7127, January 2014, <https://www.rfc-editor.org/info/rfc7127>.

Dawkins, S., "Increasing the Number of Area Directors in an IETF Area", BCP 9, RFC 7475, DOI 10.17487/RFC7475, March 2015, <https://www.rfc-editor.org/info/rfc7475>.

Halpern, J., Ed. and E. Rescorla, Ed., "IETF Stream Documents Require IETF Rough Consensus", BCP 9, RFC 8789, DOI 10.17487/RFC8789, June 2020, <https://www.rfc-editor.org/info/rfc8789>.

Rosen, B., "Responsibility Change for the RFC Series", BCP 9, RFC 9282, DOI 10.17487/RFC9282, June 2022, <https://www.rfc-editor.org/info/rfc9282>.

[RFC9290] Fossati, T. and C. Bormann, "Concise Problem Details for Constrained Application Protocol (CoAP) APIs", RFC 9290, DOI 10.17487/RFC9290, October 2022, <https://www.rfc-editor.org/rfc/rfc9290>.

Acknowledgements

This document was motivated by the AUTH48 experience for RFC 9200..RFC 9203. Then, Jaime Jim辿nez made me finally write this document. Marco Tiloca provided useful comments on an early presentation of this idea. Michael Richardson pointed out the issues that led to Section 4.1. Carl Wallace provided further comments shining light on the practical aspects of the proposals.

Author’s Address

Carsten Bormann
Universit辰t Bremen TZI
Postfach 330440
D-28359 Bremen
Germany
Phone: +49-421-218-63921
Email: cabo@tzi.org