

Network Working Group  
Internet-Draft  
Intended status: Informational  
Expires: 31 December 2025

C. Bormann  
Universitt Bremen TZI  
C. Amsss  
29 June 2025

CBOR: Generating Numeric Map Labels from Textual EDN  
draft-bormann-cbor-edn-mapkey-00

## Abstract

The Concise Binary Object Representation (CBOR, STD 94 == RFC 8949) is a data format whose design goals include the possibility of extremely small code size, fairly small message size, and extensibility without the need for version negotiation.

CBOR diagnostic notation (EDN) is widely used to represent CBOR data items in a way that is accessible to humans, for instance for examples in a specification. Complex examples often use nested maps, the map keys (labels) for each of which are often sourced from different specifications. While the e'' application extension provides a way to import data items, particularly constant values, from a CDDL model, it does not help with automatically selecting the right kind of map depending on its position in the nested maps.

// The present document is intended to capture ideas initially  
// discussed at the CBOR WG interim 2025-06-25 and demonstrate some  
// design alternatives. It is not ready for adoption yet in any way.

## About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at  
<https://cabo.github.io/mapkey/draft-bormann-cbor-edn-mapkey.html>.  
Status information for this document may be found at  
<https://datatracker.ietf.org/doc/draft-bormann-cbor-edn-mapkey/>.

Discussion of this document takes place on the cbor Working Group mailing list (<mailto:cbor@ietf.org>), which is archived at <https://mailarchive.ietf.org/arch/browse/cbor/>. Subscribe at <https://www.ietf.org/mailman/listinfo/cbor/>.

Source for this draft and an issue tracker can be found at <https://github.com/cabo/mapkey>.

## 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 31 December 2025.

## Copyright Notice

Copyright (c) 2025 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 (<https://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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

## Table of Contents

1. Introduction	3
2. The mapkey<>> application extension: importing map labels from CDDL	3
Problem	3
Solution	4
3. IANA Considerations	5
4. Security considerations	6
5. References	6
5.1. Normative References	6
5.2. Informative References	6
Acknowledgements	7
Authors' Addresses	7

## 1. Introduction

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

## 2. The mapkey&lt;&gt;&gt; application extension: importing map labels from CDDL

## Problem

In diagnostic notation examples, authors often would prefer to use text names in place of the integer map keys that are used in a protocol message for efficiency. While the e'' application extension provides a way to associate names with integer data items, the protocol designer must be very careful to use the right name for the kind of map that uses the integer key: Different specifications may use different integer numbers for a key with the same textual name, and even in a single specification there may be homonyms that resolve to different integer values in different kinds of maps (e.g., in [STD96], alg is represented by 1 in headers and by 3 in COSE\_Key).

For example, Figure 4 in Section 3.5.2 of [RFC9528] contains this example that employs nested maps:

```
{
  2 : "42-50-31-FF-EF-37-32-39",      /CCS/
  8 : {                                /sub/
    1 : {                              /cnf/
      1 : 1,                          /COSE_Key/
      2 : h'00',                      /kty/
      -1 : 4,                         /kid/
      -2 : h'b1a3e89460e88d3a8d54211dc95f0b90
          3ff205eb71912d6db8f4af980d2db83a' /crv/
    }
  }
}
```

To check this example, a human reviewer has to look up the integer labels in the specifications for the different maps employed and translate them to the names of the map keys defined for each type of map. The outer map is a CWT Claims Set (CCS), for which the labels 2 (sub) and 8 (cnf) are defined in [RFC8392] and [RFC8747], respectively. Within cnf, the label for COSE\_Key is also defined by [RFC8747], while the labels within that map are defined in Section 7 of RFC 9052 [STD96]. Map keys are also often an extension point, and obtaining their numeric values therefore also may require consulting an IANA registry.

The objective of the present proposal is that a specification writer could employ an EDN app-extension that allows the example to read a bit like:

```
mapkey<<"Claims-Set",
{
  "sub": "42-50-31-FF-EF-37-32-39"
  "cnf": {
    "COSE_Key": {
      "kty": "OKP"
      "kid": h'00'
      "crv": "X25519"
      "x": h'bla3e89460e88d3a8d54211dc95f0b90
          3ff205eb71912d6db8f4af980d2db83a'
    }
  }
}
```

Note that this example not only uses names for map keys, but also uses names for map values 1 ("OKP") and 4 ("X25519").

// Discussion: For use in EDN, the names need to be provided in some form that is a valid CBOR data item. In the example above, this is done in text strings; for increased clarity, it could be done in a more eye-catching way, e.g., as single-quoted (byte) strings or even in an e'...'-'like construct.

## Solution

In many cases, the constants needed to handle the numeric map labels in this example are already available in a CDDL model, or could be easily made available in this way.

For the example used in this section, [RFC9781] provides CDDL for [RFC8392] and [RFC8747], and [STD96] provides CDDL for its own data types. Note that, to remain useful with extension points where new map keys are defined regularly, there needs to be a way to reference IANA registries for the name-to-integer translation; this is a separate problem for which a potential solution is presented in Appendix A.2.1 of [I-D.bormann-cbor-cddl-2-draft].

This section needs to define where in the CDDL the names to be resolved are looked for; CDDL rule names are one obvious candidate, as are member names for group choices that are often employed for documentation (Figure 2 in Section 2 of [RFC9290] shows member names such as "title", "detail", etc.):

```

problem-details = non-empty<{
  ? &(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

```

This specification defines an app-extension mapkey<>> with two parameters:

- \* The name of the CDDL rule for the top level (here: Claims-Set), and
- \* an EDN depiction of the data structure where names can be used in place of numeric map keys and values.

Note that the app-extension does not itself define where the CDDL definitions it uses come from. This information needs to come from the context of the example, and there is probably value in establishing a convention.

### 3. IANA Considerations

IANA is requested to make the following two assignments in the CBOR Diagnostic Notation Application-extension Identifiers registry [IANA.cbor-diagnostic-notation] established by [I-D.ietf-cbor-edn-literals]:

+=====+=====+	
Application-extension Identifier	Description
+=====+=====+	
mapkey	import map labels
	from external CDDL
+-----+-----+	

Table 1: Additions to Application-extension Identifier Registry

All entries have the Change Controller "IETF" and the Reference "RFC-XXXX".

```

// RFC Editor: please replace RFC-XXXX with the RFC number of this
// RFC, [IANA.cbor-diagnostic-notation] with a reference to the
// registry group established by [I-D.ietf-cbor-edn-literals], and
// remove this note.

```

#### 4. Security considerations

The security considerations of [RFC8610], [STD94], [I-D.ietf-cbor-edn-literals] apply.

TBD.

#### 5. References

##### 5.1. Normative References

- [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](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>>.
- [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>>.

##### 5.2. Informative References

- [I-D.bormann-cbor-cddl-2-draft]  
Bormann, C., "CDDL 2.0 and beyond -- a draft plan", Work  
in Progress, Internet-Draft, draft-bormann-cbor-cddl-2-  
draft-06, 28 February 2025,  
<[https://datatracker.ietf.org/doc/html/draft-bormann-cbor-  
cddl-2-draft-06](https://datatracker.ietf.org/doc/html/draft-bormann-cbor-cddl-2-draft-06)>.
- [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-01, 29 December 2024,  
<[https://datatracker.ietf.org/doc/html/draft-ietf-cbor-  
edn-e-ref-01](https://datatracker.ietf.org/doc/html/draft-ietf-cbor-edn-e-ref-01)>.

- [RFC8392] Jones, M., Wahlstroem, E., Erdtman, S., and H. Tschofenig, "CBOR Web Token (CWT)", RFC 8392, DOI 10.17487/RFC8392, May 2018, <<https://www.rfc-editor.org/rfc/rfc8392>>.
- [RFC8747] Jones, M., Seitz, L., Selander, G., Erdtman, S., and H. Tschofenig, "Proof-of-Possession Key Semantics for CBOR Web Tokens (CWTs)", RFC 8747, DOI 10.17487/RFC8747, March 2020, <<https://www.rfc-editor.org/rfc/rfc8747>>.
- [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>>.
- [RFC9528] Selander, G., Preu Mattsson, J., and F. Palombini, "Ephemeral Diffie-Hellman Over COSE (EDHOC)", RFC 9528, DOI 10.17487/RFC9528, March 2024, <<https://www.rfc-editor.org/rfc/rfc9528>>.
- [RFC9781] Birkholz, H., O'Donoghue, J., Cam-Winget, N., and C. Bormann, "A Concise Binary Object Representation (CBOR) Tag for Unprotected CBOR Web Token Claims Sets (UCCS)", RFC 9781, DOI 10.17487/RFC9781, May 2025, <<https://www.rfc-editor.org/rfc/rfc9781>>.
- [STD96] Internet Standard 96, <<https://www.rfc-editor.org/info/std96>>.  
At the time of writing, this STD comprises the following:
- Schaad, J., "CBOR Object Signing and Encryption (COSE): Structures and Process", STD 96, RFC 9052, DOI 10.17487/RFC9052, August 2022, <<https://www.rfc-editor.org/info/rfc9052>>.
- Schaad, J., "CBOR Object Signing and Encryption (COSE): Countersignatures", STD 96, RFC 9338, DOI 10.17487/RFC9338, December 2022, <<https://www.rfc-editor.org/info/rfc9338>>.

## Acknowledgements

TBD.

## Authors' Addresses

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

Christian Amsss  
Austria  
Email: christian@amsuess.com