

CBOR@ 2023-11-07

WGLC edn-literals/update-cddl-grammar
cbor-packed
CDE (Common Deterministic Encoding Profile)

CBOR-associated languages

- **CBOR** = representation and interchange format (binary, concise, efficient)
 - low-level visualization in text as **cbor-pretty** (hex with comments)

Two associated textual languages:

- **EDN** (cbor-diag) → examples, diagnostics
 - Text form for single **instance** (item/sequence), convert back and forth (**cbor.me**)
 - Derived from **JSON**, made more useful for humans, added binary, tags, ...
- **CDDL** → specification, validation
 - Describe specific data **model** (grammar)
 - Inspired by **ABNF**, can describe JSON, CBOR, CSV*

WGLCs ended yesterday:

draft-ietf-cbor-edn-literals

Originally: Define EDN literals

Now really: EDN maintenance and extensions

draft-ietf-cbor-update-8610-grammar

Fixes to RFC 8610 ABNF for CDDL

Post-WGLC next steps

- Lots of feedback on edn-literals **before** WGLC
 - part of it based on a complete implementation
- Discussion of update-8610-grammar happened way before
- update-8610-grammar needed to proceed with CDDL 2 work
- limited urgency of edn-literals (but nice to stabilize now)

EDN: Random idea e''

People often write text strings into EDN when they actually mean integers in an enum

Fix: e''

```
{ /COSE Key/  
  e'x': ...  
  e'y': ...  
}
```

- Cannot be processed without schema information
- Still useful as a whiteboard/slide notation

CBOR-packed

- Not your garden variety compression:
Allow **in-place** use of packed CBOR data item
- Two Separate Items:
 1. Build reference table(s)
 2. Reference the table(s) from a "rump"

2. CBOR-packed reference set (~ ready)

- Shared items (complete data items)
- Argument items (for concatenation or other operations)

Simple values 0–15 ("shared")

Tag 6 ("shared" with int, "straight argument" otherwise)

Tags 224..255, 28704... (straight argument)

Tags 216..223, 27647... (inverted argument)

Function Tags (**extension point**): 105, 106 (ijoin/join)

1. CBOR-packed table building

"Batteries Included"

Tag 113: Simple basic table setup

Tag 1113: Split basic table setup

Larger variety of table building methods envisioned:

- optimized for specific application
- innovations (implicit, incremental)

Table building innovations

- **Implicit** Table Building:
Build the table from information already in the "rump"
- **Incremental** Table Building (serial):
Build the table so new entries are immediately usable **after** the primitive
 - Cf. Tags 256/25; problem → limited applicability:
 - CBOR maps don't have a defined "document order"

Plan

- Make use of more benchmarks such as `dns-in-cbor`
- Get more implementer feedback
- `timebox` this information acquisition period!

CCDE

Do **Common Deterministic Encoding** Profile now

Deterministic Encoding

Defined by Section 4.2 of RFC 8949
Some details (rightly) left to application

Problem:

- This hampers the development of **generic** encoders/decoders
- Users think that deterministic encoding isn't well defined

Common Deterministic Encoding Profile (CDE): Nail this down
Define "Application Profile" to work on top of this
"dCBOR" is one such Application Profile

Plan

Adopt CDE now → BCP or standards track?
Merge the two dCBOR documents, adopt?