A graphical depiction of the layers of encoding and field values using CBOR terminology for layers:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Sequence** | Item #1 | | | … | Item #N | | |
| **Headers** | Head | | Content | Head | | Content |
| **Fields** | Major Type | Argument | Byte String | Major Type | Argument | Byte String |
| **Values** | 2 | Non-zero length | Non-empty string | 2 | Non-zero length | Non-empty string |

The block content can be defined as a CDDL schema from [RFC 8610] to formalize the definition in a machine-readable way:

; The LTP block is the non-empty CBOR sequence (unwrapped array)

; of ~cla\_seq

cla\_seq = [1\* cla\_item]

; Each item is a non-empty byte string which contains

; a serialized BPv7 bundle

cla\_item = bstr .cbor bundle

; The symbol “bundle” is defined in Appendix B of RFC 9171

Another way is to include an example in CBOR Extended Diagnostic Notation from Appendix G of [RFC 8610] where two indefinite-length arrays use elided placeholder tags for real BPv7 blocks, which would be considerably longer:

<<[\_ 888(null), 888(null), 888(null)]>>,<<[\_ 888(null), 888(null)]>>

Which encodes to the following hexadecimal string, using indentation to separate CBOR heads from content:

4E

9FD90378F6D90378F6D90378F6FF

4A

9FD90378F6D90378F6FF