CoRE: CoRECONF

January 18, (Wednesday), 15:00-16:30 UTC
(16:00–17:30 CET, 07:00–08:30 PST)

- RFC 9254: YANG-CBOR
- In IESG: CORE-SID
- WGLC passed CORE-COMI
- WGLC passed CORE-YANG-LIBRARY

CORE-SID: PRs to be merged

Rob Wilton's DISCUSS (stale, on -16)
Remaining main issue was:

- document objectives of SID management

Was to be addressed by -19 (from PR #146)
➔ Feedback took a while

Requirement for a "stable" field in SID file (PR #141)
For SID files merged from stable and unstable,
enables use as further SID input
➔ Not yet supported by PYANG

Feedback from Rob Wilton

Came in 90 minutes ago :-)
Many useful improvements to be processed.

DISCUSS Preview:

What about obsolete SID assignments?
Do we need to carry these around in SID files?
(probably at least to prevent undesirable reallocation!)

PR #141

Add to SID file:

```
leaf status {
  type enumeration {
    enum stable {
      value 0;
      description "This SID allocation has been published in a stable catalog";
    }
    enum unstable {
      value 1;
      description "This SID allocation has been done during a
                   development process";
    }
    default stable;
  }
}
```

CORE-SID: Plan

Process Rob's feedback, extend #141 and #146, new issues

Editorial round needed after latest changes, e.g.:

- Issue #139:
  Remove remnants of rc:yang-data and RFC 8040 (RW)

Checking round:

- Issue #66: core-sid: May need to renumber YANG module
- Issue #88: review against COMI requirements

CORE-COMI

-11: Further work was waiting for yang-cbor can thaw now:

— technical issue: 'k' query parameter (key representation in GET URIs)

— various editorial nits

➔ PR in progress towards -12

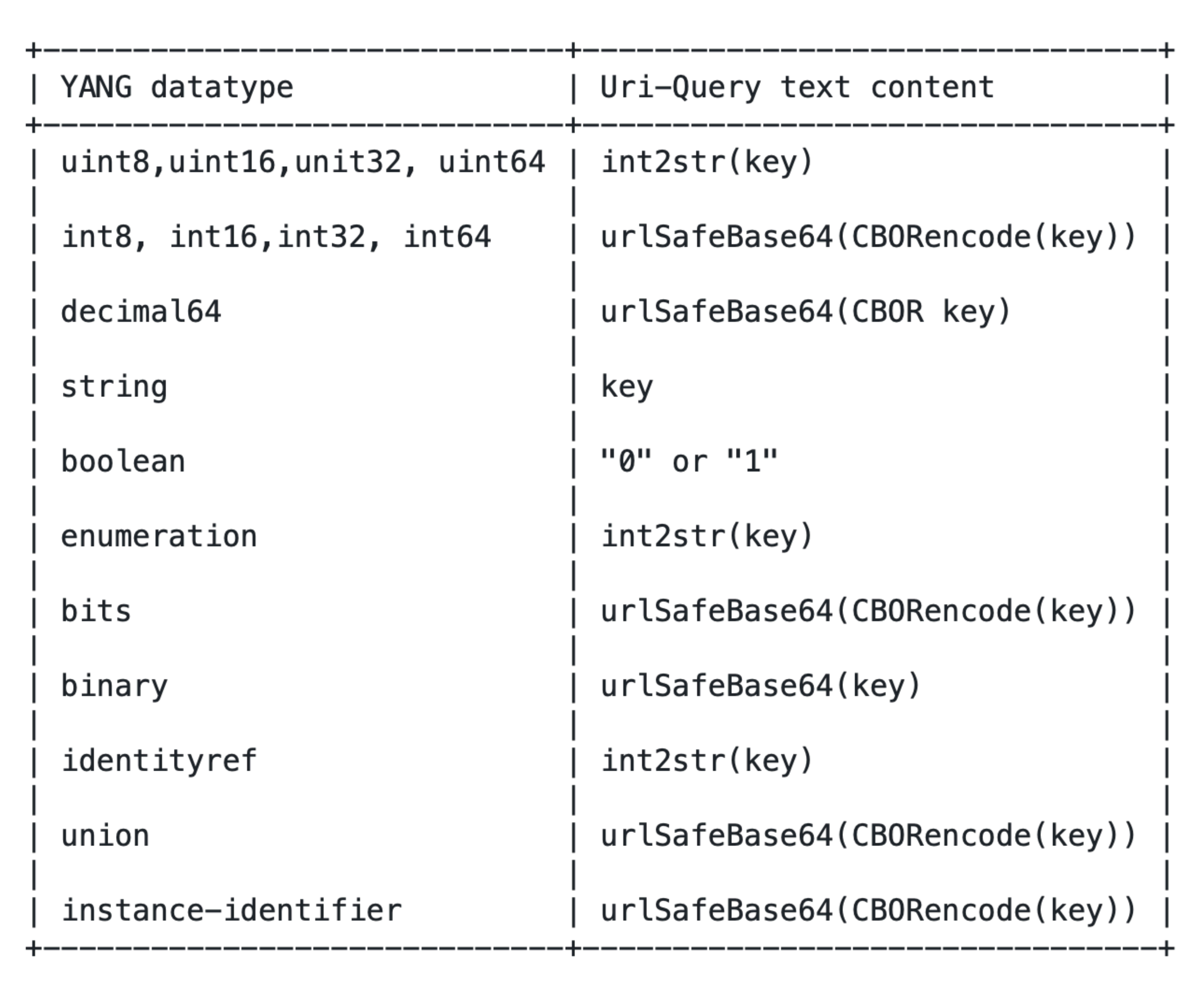
CORE-COMI: 'k' query parameter

key representation in GET URIs:

- highly complex
- surprising differences, brittle with changes in data type

Proposal: always use urlSafeBase64(CBOR)

Not done: add optimization for frequent string case?



YANG datatype	Uri-Query text content
uint8,uint16,unit32, uint64	int2str(key)
int8, int16,int32, int64	urlSafeBase64(CBORencode(key))
decimal64	urlSafeBase64(CBOR key)
string	key
boolean	"0" or "1"
enumeration	int2str(key)
bits	urlSafeBase64(CBORencode(key))
binary	urlSafeBase64(key)
identityref	int2str(key)
union	urlSafeBase64(CBORencode(key))
instance-identifier	urlSafeBase64(CBORencode(key))

CORE-COMI: 'k' query parameter — comma issue

```
4.1:
The SID in the URI is followed by the (?k=key1,key2,...).
```

This doesn't work where keys can contain commas!
(Fixed by base64url-encoding, but…)

Proposal:

- Instead, use a CBOR sequence (RFC 8742) of keys
- Encode the whole thing in base64url

Could also be done: Use CBOR sequence also in FETCH payload
(Note that this is a sequence of sid/key arrays!)

CORE-COMI: Plan

Process k simplification

Energy check -- who can drive finishing this?

- Technically stable after k simplification
- The usual editorial dance is needed

Feedback from WG needed for finishing drive