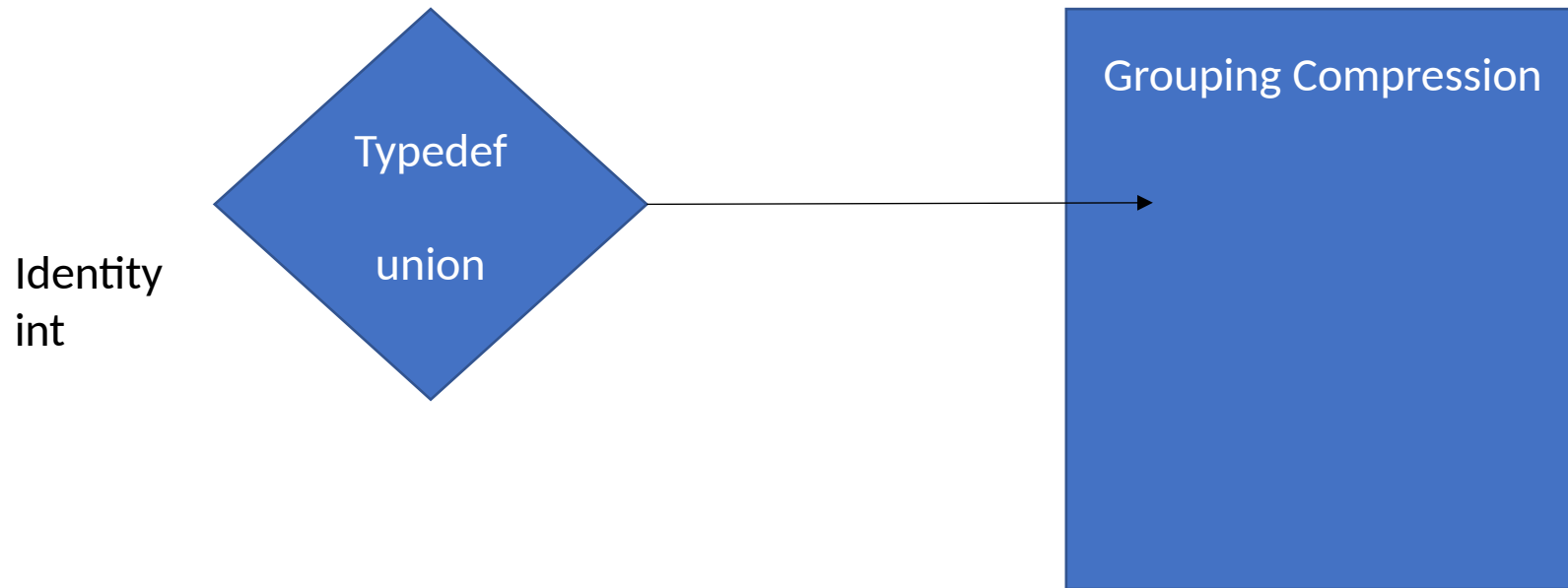


Data Model Improvements

Laurent Toutain

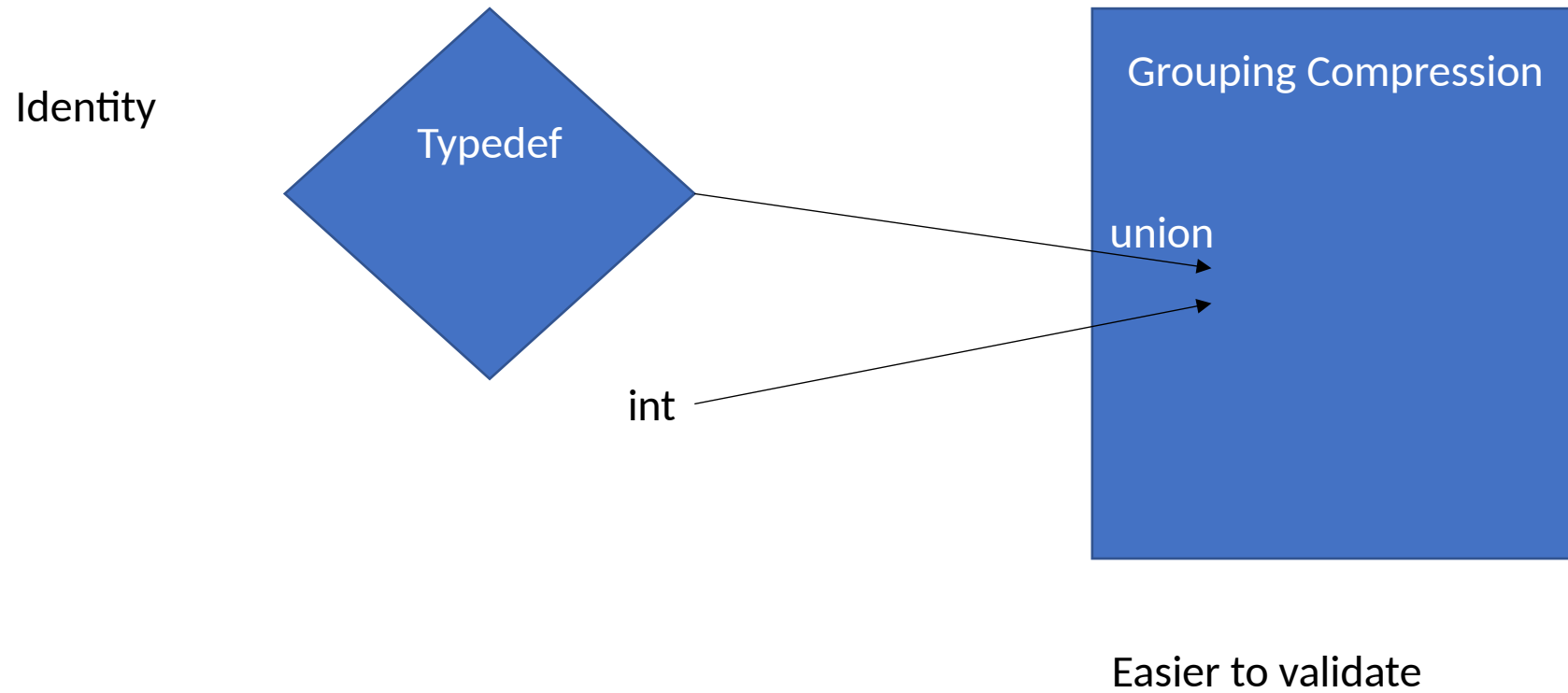
IMT Atlantique

What's wrong



Validation tool expects a string in JSON
No problem in XML, everything is ascii
<tag>10</tag> vs
'tag' : 10 or 'tag' : "10"

Proposal



changes

```
typedef fl-type {
    type union {
        type uint64 {
            range 1..max;
        }
        type identityref {
            base fl-base-type;
        }
    }
    description
        "Field length either a positive integer expressing the size in
        bits or a function defined through an identityref.";
    reference
        "RFC 8724 SCHC: Generic Framework for Static Context Header
        Compression and Fragmentation";
}
```

```
765 typedef fl-type {
766     type identityref {
767         base fl-base-type;
768     }
769     description
770         "Function used to indicate field length.";
771
772     reference
773         "RFC 8724 SCHC: Generic Framework for Static Context Header
774         Compression and Fragmentation";
775 }
```

```
1012 leaf field-length {
1013     type schc:fl-type;
1014
1015     mandatory true;
1016     description
1017         "Field Length, expressed in number of bits if the length is
1018         known when the Rule is created or through a specific
1019         function if the length is variable.";
1019 }
```

```
1014 leaf field-length {
1015     type union {
1016         type uint8;
1017         type schc:fl-type;
1018     }
1019     mandatory true;
1020     description
1021         "Field Length, expressed in number of bits if the length is
1022         known when the Rule is created or through a specific
1023         function if the length is variable.";
1024 }
```

And one more thing

324

```
identity fid-coap-option-if-match {
325     base fid-coap-base-type;
326     description
327         "CoAP option If-Match.";
328     reference
329         "RFC 7252 The Constrained Application Protocol (CoAP)";
330 }
331
332 identity fid-coap-option-uri-host {
333     base fid-coap-base-type;
334     description
335         "CoAP option URI-Host.";
336     reference
337         "RFC 7252 The Constrained Application Protocol (CoAP)";
338 }
339
340 identity fid-coap-option-etag {
341     base fid-coap-base-type;
342     description
343         "CoAP option Etag.";
344     reference
345         "RFC 7252 The Constrained Application Protocol (CoAP)";
346 }
347
348
```

324

```
identity fid-coap-option {
325     base fid-coap-base-type;
326     description
327         "Geeneric CoAP option.";
328     reference
329         "RFC 7252 The Constrained Application Protocol (CoAP)";
330 }
331
332 identity fid-coap-option-if-match {
333     base fid-coap-option;
334     description
335         "CoAP option If-Match.";
336     reference
337         "RFC 7252 The Constrained Application Protocol (CoAP)";
338 }
339
340 identity fid-coap-option-uri-host {
341     base fid-coap-option;
342     description
343         "CoAP option URI-Host.";
344     reference
345         "RFC 7252 The Constrained Application Protocol (CoAP)";
346 }
347
348 identity fid-coap-option-etag {
349     base fid-coap-option;
350     description
351         "CoAP option Etag.";
352     reference
353         "RFC 7252 The Constrained Application Protocol (CoAP)";
354 }
355
356
```

Access control

Model – all fields are rw

```
module: ietf-schc
+--rw schc
  +--rw rule* [rule-id-value rule-id-length]
    +--rw rule-id-value      uint32
    +--rw rule-id-length    uint8
    +--rw (nature)?
      +--:(fragmentation) {fragmentation}?
        +--rw fragmentation-mode      schc:fragmentation-mode-type
        +--rw l2-word-size?           uint8
        +--rw direction               schc:di-type
        +--rw dtag-size?              uint8
        +--rw w-size?                 uint8
        +--rw fcn-size                 uint8
        +--rw rcs-algorithm?          rcs-algorithm-type
        +--rw maximum-packet-size?    uint16
        +--rw window-size?           uint16
        +--rw max-interleaved-frames? uint8
        +--rw inactivity-timer
          | +--rw ticks-duration?    uint8
          | +--rw ticks-numbers?    uint16
        +--rw retransmission-timer
          | +--rw ticks-duration?    uint8
          | +--rw ticks-numbers?    uint16
        +--rw max-ack-requests?       uint8
        +--rw (mode)?
          +--:(no-ack)
          +--:(ack-always)
          +--:(ack-on-error)
            +--rw tile-size?         uint8
            +--rw tile-in-all-1?   schc:all-1-data-type
            +--rw ack-behavior?     schc:ack-behavior-type
          +--:(compression) {compression}?
            +--rw entry* [field-id field-position direction-indicator]
              +--rw field-id          schc:fid-type
              +--rw field-length      schc:fl-type
              +--rw field-position    uint8
              +--rw direction-indicator schc:di-type
              +--rw target-value* [index]
                | +--rw value?      binary
                | +--rw index      uint16
              +--rw matching-operator      schc:mo-type
              +--rw matching-operator-value* [index]
                | +--rw value?      binary
                | +--rw index      uint16
              +--rw comp-decomp-action      schc:cda-type
              +--rw comp-decomp-action-value* [index]
                +--rw value?      binary
                +--rw index      uint16
            +--:(no-compression)
          +--:(no-compression)
```

```

/ietf-schc:schc
rule
{1000094: {1: [
  {4:
    entry
    [
      field-id
      field-length
      field-position
      direction-indicator
      matching-operator
      comp-decomp-action
      target-value
      index
      value
      [{6: 1000067, 7: 4, 8: 1, 5: 1000018, 9: 1000082, 1: 1000015, 13: [{1: 0, 2: h'06'}]}],
      {6: 1000064, 7: 8, 8: 1, 5: 1000018, 9: 1000082, 1: 1000015, 13: [{1: 0, 2: h''}]},
      {6: 1000060, 7: 20, 8: 1, 5: 1000018, 9: 1000083, 1: 1000015, 13: [{1: 0, 2: h''}]},
      {6: 1000063, 7: 16, 8: 1, 5: 1000018, 9: 1000083, 1: 1000011},
      {6: 1000062, 7: 8, 8: 1, 5: 1000018, 9: 1000082, 1: 1000015, 13: [{1: 0, 2: h'3A'}]},
      {6: 1000061, 7: 8, 8: 1, 5: 1000018, 9: 1000083, 1: 1000015, 13: [{1: 0, 2: h'FF'}]},
      {6: 1000059, 7: 64, 8: 1, 5: 1000018, 9: 1000082, 1: 1000015, 13: [{1: 0, 2: h'200104701F2101D2'}]},
      {6: 1000058, 7: 64, 8: 1, 5: 1000018, 9: 1000082, 1: 1000015, 13: [{1: 0, 2: h'0000000000000001'}]},
      {6: 1000056, 7: 64, 8: 1, 5: 1000018, 9: 1000083, 1: 1000016},
      {6: 1000055, 7: 64, 8: 1, 5: 1000018, 9: 1000083, 1: 1000016}],
    34: 6,
    33: 3,
    35: 1000087},
    rule-id-value
    rule-id-length
    rule-nature
    nature-compression
  {34: 12,
    33: 11,
    35: 1000088,
    2: 1000020,
    29: 1000090,
    3: 2,
    20: 3,
    21: 1000080},
    rule-id-value
    rule-id-length
    rule-nature
    nature-fragmentation
    direction
    di-up
    rcs-algorithm
    rcs-RFC8724
    dtag-size
    fcn-size
    fragmentation-mode
    fragmentation-mode-no-ack
  {34: 100,
    33: 8,
    35: 1000089}
    rule-id-value
    rule-id-length
    rule-nature
    nature-no-compression
  ]}
}

```


Access control

- Add a new YANG field to indicate if an entry can be modified or not
- Add some property to a rule to add/delete entries
- Add some property to a rule to allow deletion
- Add some property to allow rule creation (fragmentation and compression)
- For fragmentation:
 - Allow timer changes (which range ?)
 - Allow max-interleave
 - Other values => another rule