In libosmocore, we have enum CELL_IDENT, which gets used in both of Cell Identifier IE (3GPP TS 48.008 3.2.2.17) and Cell Identifier List IE (3GPP TS 48.008 3.2.2.27).

The similar meaning and structuring of these two IEs suggests that the List IE is simply a bunch of the plain Cell Identifier. However, if you read in 3GPP TS 48.008, comparing 3.2.2.17 and 3.2.2.27, there is this odd difference:

3.2.2.17, Cell Identifier IE:

0000 CGI
  0001 LAC+CI
  0010 CI
  0011 No cell

1000 PLMN+LAC+RNC (-> E-UTRAN)
  1001 RNC (-> E-UTRAN)
  1010 LAC+RNC (-> E-UTRAN)
  1011 SAI
  1100 LAC+RNC+CI

3.2.2.27a, Cell Identifier List Segment:

0000 CGI
  0001 LAC+CI
  0010 CI
  0011 No cell

0100 LAI
  0101 LAC
  0110 entire BSS
  0111 MCC+MNC

I noticed this because I was trying to check for the correct target cell identifier in the BSSMAP Handover Request message; we send a LAI cell identification in the Cell Identifier IE. wireshark shows the LAI cell identifier the same way that osmo-msc intends to encode it, but the TTCN3 BSSAP_Templates fail to parse this LAI cell identifier.

So I added this cell id type to BSSMAP_IE_CellIdentifier in ./deps/titan.ProtocolModules.BSSMAP/src/BSSAP_Types.ttcn. and just to confirm that I'm right i took a quick look in the spec, and then found that the ttcn definition is in fact on par with the spec: the LAI cell id is not present in 3.2.2.17 at all!

Now, our libosmocore gsm/protocol/gsm_08_08.h and gsm0808_utils.h are based on the assumption that the same cell identifiers are used in both of these IEs.

The osmo-bsc and osmo-msc neighbor config for inter-BSC handover is built assuming that we can send the 3.2.2.27a cell identifiers in the Handover
Request's 'Cell Identifier (Target)' IE.

Apparently the spec intends otherwise.

The effect of this goes across several osmo cn components:

- We define enum CELL_IDENT in libosmocore.
- osmo-bsc sends the Handover Request using the 'Cell Identifier (Serving)' and 'Cell Identifier (Target)' IEs.
- osmo-msc parses these, possibly forwards in inter-MSC handover procedure.

I'm not sure how to resolve this.

- We could just accept that we're using the 3.2.2.27a identifiers also in 3.2.2.17. It makes an awful lot of sense in fact.
  The wireshark implementation seems to agree with this point, though of course that's not the benchmark.
  It might pose an incompatibility with strictly spec conforming cn implementations.
  And we would need to extend the TTCN BSSAP_Types.ttcn with values that aren't strictly present in 3.2.2.17,
  if we want to test these cell identifier types in our test suite.

- or we could endeavour to fix the identifiers that we use in the BSSMAP protocol according to spec, in osmo-bsc and osmo-msc
  either we introduce two distinct enums -- that could ripple through a lot of the cell identifier API, ugly.
  or we still use the same enum for both, but making distinctions in the implementations which values are used for which.

To enforce that, we may have to adjust which cell identifier types can be used in the neighbor configuration of osmo-bsc and
osmo-msc,
or we find ways to, for example, convert all cell identifier types to a supported one.
I guess easiest would be to enforce full CGI everywhere.

Yet easier would be to highlight in the documentation that osmocom is capable of nonstandard cell identifiers,
and to conform to specs users should simply use the full CGI everywhere in the neighbor config.

Any insights or opinions on this?

---

**History**

#1 - 06/23/2021 10:10 AM - laforge

On Wed, Jun 23, 2021 at 12:43:19AM +0000, neels [REDMINE] wrote:

> Any insights or opinions on this?

The A interface is the most important point of interop between RAN and CN, and
it is important for many users that osmo-bsc interoperates with different CN out there. So I don't think we can deviate here from what the spec permits.

For sure we can optionally extend AoIP if we know osmo-msc is on the other side, but we always must have a spec compliant use of AoIP as a base line.

#2 - 06/23/2021 05:06 PM - neels

On Wed, Jun 23, 2021 at 10:10:15AM +0000, laforge [REDMINE] wrote:

> For sure we can optionally extend AoIP if we know osmo-msc is on the other side, but we always must have a spec compliant use of AoIP as a base line.

As long as the neighbors in the osmo-bsc.cfg are configured using CGI or only-LAC, the Handover Request will be compliant. So instead of enforcing (or converting), we could tell the users to take care with their neighbor config...

Using other cell identifiers could be seen as a feature. The feature being, osmo-msc could translate other cell id kinds to CGI before sending in the HO Request. For that osmo-msc.cfg must also provide the full CGI of neighbors.

so step one would be informing the users to rather use full CGI everywhere.
Apart from that we can analyse how to enforce a valid cell id type in osmo-msc.

#3 - 06/24/2021 07:30 AM - laforge

On Wed, Jun 23, 2021 at 05:06:21PM +0000, neels [REDMINE] wrote:

> As long as the neighbors in the osmo-bsc.cfg are configured using CGI or
only-LAC, the Handover Request will be compliant. So instead of enforcing (or
converting), we could tell the users to take care with their neighbor config...

I would think the default in osmo-bsc should be to only accept spec compliant
neighbor configs. The additional options should be enabled/permited
only by another vty command like "osmocom-aolp-extensions enabled"

Does that make sense?