

OsmoBSC - Feature #1602

BSC side of LCLS (local call local switching) as per the 3GPP specs

02/23/2016 03:56 PM - laforge

Status: Resolved	Start date: 02/23/2016
Priority: High	Due date:
Assignee: dexter	% Done: 100%
Category:	
Target version: 3GPP LCLS	
Spec Reference:	
Description Some deployments use OsmoBSC with its support to connect to multiple MSCs simultaneously in order to have local call switching. Let's study the 3GPP LCLS feature and see how that can help us in that regard, maybe it has a better/cleaner/more standard way to do it?	
Related issues:	
Related to OsmoMSC - Feature #2487: MSC side of LCLS (local call local switch...	Stalled 09/03/2017
Related to OsmoBSC - Feature #3586: support LCLS for inter-BSC handover	New 09/24/2018
Related to OsmoBSC - Bug #3659: LCLS directly between BTSs	Stalled 10/17/2018

History

#1 - 11/09/2016 09:47 AM - laforge

- Assignee set to wireless
- Priority changed from Low to Normal

please review the specs (or other information you can find online) regarding LCLS and try to summarize how it works in a small set of slides you can show to the team.

#2 - 12/29/2016 09:15 AM - wireless

- Status changed from New to In Progress

I have started to read and collect information about local call local switching.

#3 - 02/23/2017 08:18 PM - laforge

- Status changed from In Progress to New
- Assignee changed from wireless to laforge

#4 - 09/03/2017 05:35 PM - laforge

- Project changed from OsmoNITB to OsmoBSC
- Assignee deleted (laforge)

#5 - 09/03/2017 05:37 PM - laforge

- Related to Feature #2487: MSC side of LCLS (local call local switching) as per the 3GPP specs added

#6 - 10/06/2017 02:25 PM - laforge

- Subject changed from Investigate LCLS (local call local switching) as per the 3GPP specs to BSC side of LCLS (local call local switching) as per the 3GPP specs

#7 - 10/06/2017 02:26 PM - laforge

- Checklist item [] LCLS implementation in OsmoBSC added
- Checklist item [] test suite to test LCLS in OsmoBSC added
- Checklist item [] jenkins / CI integration of test suite added

#8 - 10/29/2017 07:00 PM - laforge

- Target version set to 3GPP LCLS

#9 - 01/30/2018 07:03 AM - laforge

- Priority changed from Normal to High

#10 - 02/19/2018 09:22 AM - laforge

- Assignee set to needs

#11 - 05/30/2018 02:49 PM - laforge

- Tags set to LCLS

#12 - 05/31/2018 09:00 AM - laforge

- Status changed from New to In Progress

- Assignee changed from needs to laforge

- % Done changed from 0 to 20

I've been making some headway on this in the laforge/lcls branches:

- <http://git.osmocom.org/osmo-bsc/log/?h=laforge/lcls> for the BSC, as well as
- <http://git.osmocom.org/osmo-tcn3-hacks/log/?h=laforge/lcls> on the test side.

both need quite some more work, but I think I've figured out the state machine with all of its transitions by now.

#13 - 06/01/2018 10:02 AM - laforge

I've made significant progress on the testing side, although I hit a Titan compiler bug (see <https://www.eclipse.org/forums/index.php/t/1093530/>) as well as some shortcomings in our *_Emulation components, see <https://gerrit.osmocom.org/#/c/osmo-tcn3-hacks/+9403/>

#14 - 06/01/2018 12:50 PM - laforge

- Checklist item [x] wireshark support added

- % Done changed from 20 to 30

during the first tests, it became apparent that wireshark wasn't able to decode LCLS specific PDUs. I've created a patch and submitted it at <https://code.wireshark.org/review/#/c/27941/>

#15 - 06/01/2018 11:18 PM - laforge

- Checklist item [] prevent LCLS enabling on calls of different codecs added

- % Done changed from 30 to 70

the tests for LCLS related signaling are rather complete now, and they all pass with my latest version of the OsmoBSC LCLS FSM.

The only missing bit now is to actually implement the body of the functions that enable/disable the local switching of the voice, i.e. that issue the MGCP commands.

Thinking about it:

- enable LCLS for a given call
 - Issue MGCP MDCX on MSC-side connection on MGCP endpoint of call A to point to IP/Port of call B
 - Issue MGCP MDCX on MSC-side connection on MGCP endpoint of call B to point to IP/Port of call A
- disable LCLS for a given call
 - Issue MGCP MDCX on MSC-side connection on MGCP endpoint of call A to point back to MSC/MGW
 - Issue MGCP MDCX on MSC-side connection on MGCP endpoint of call B to point back to MSC/MGW

So we have to cache the ip/port information of the MGW/MSC during ongoing local switching, as we don't receive that information [again] once we switch back.

Known limitation: Until the MGW can perform transcoding, calls with mis-matching codecs will break. We should add some provision into OsmoBSC that would prevent activating LCLS on two calls of different codecs until the MGW can transcode.

There will also be LCLS implications at the time we do inter-BSC hand-over. This will have to be looked into once inter-BSC HO is merged.

#16 - 06/01/2018 11:19 PM - laforge

laforge wrote:

during the first tests, it became apparent that wireshark wasn't able to decode LCLS specific PDUs. I've created a patch and submitted it at <https://code.wireshark.org/review/#/c/27941/>

wireshark patch is already accepted / merged to master.

#17 - 06/01/2018 11:21 PM - laforge

- Checklist item [] add vty/config command to enable/disable LCLS on per-MSB basis added
Checklist item [] logging of related BSSMAP messages, not just FSM auto-logging added

#18 - 06/02/2018 06:53 PM - laforge

- % Done changed from 70 to 90

<https://gerrit.osmocom.org/#/c/osmo-bsc/+9416> has been updated. It now actually performs local switching via MGCP MDCX re-configuration of OsmoMGW.

The test suite in <https://gerrit.osmocom.org/#/c/osmo-ttcn3-hacks/+9412> has equally been extended to verify the actual MGCP user plane switching functionality.

#19 - 06/02/2018 06:53 PM - laforge

- Checklist item [x] LCLS implementation in OsmoBSC set to Done
Checklist item [x] test suite to test LCLS in OsmoBSC set to Done
Checklist item [x] logging of related BSSMAP messages, not just FSM auto-logging set to Done

#20 - 06/03/2018 11:10 AM - laforge

- Checklist item [x] add vty/config command to enable/disable LCLS on per-MSB basis set to Done

bsc code + testsuite has been merged to respective master branch, and VTY config option to enable/disable LCLS has been added. Working on CI execution of tests now.

#21 - 06/03/2018 04:04 PM - laforge

- Checklist item [x] jenkins / CI integration of test suite set to Done

Jenkins integration of test suite: https://jenkins.osmocom.org/jenkins/view/TTCN3/job/ttcn3-bsc-test/test_results_analyzer/

#22 - 07/04/2018 12:47 PM - laforge

- Assignee changed from laforge to dexter

the only remaining bit is to ensure LCLS is not enabled on calls that have different codecs on both legs (at least until we have working transcoding in osmo-mgw).

As I'm leaving on holidays soon, it would be good if [dexter](#) could take care of this last missing bit, including related TTCN3 tests

#23 - 07/10/2018 01:41 PM - dexter

I have now added a check that tests the two call legs for different codec/rate. If codec/rate is different, then LCLS will be avoided. However, I am a bit unsure about the status codes here. At the moment the status code I see is LCLS_STS_not_yet_Is. I am not sure if this is right, but it must be the same as with the other conditions in lcls_enable_possible()

See also:

<https://gerrit.osmocom.org/#/c/osmo-bsc/+9940> lcls: do not LCLS call legs with different codecs
<https://gerrit.osmocom.org/#/c/osmo-ttcn3-hacks/+9941> BSC_Tests_LCLS: try call legs with different codec/rate

#24 - 07/16/2018 07:13 AM - dexter

- Checklist item [x] prevent LCLS enabling on calls of different codecs set to Done

#25 - 07/23/2018 10:29 AM - dexter

- % Done changed from 90 to 100

#26 - 08/08/2018 11:07 AM - laforge

- Status changed from In Progress to Resolved

if it's at 100%, it can be resolved.

#27 - 08/13/2018 08:17 AM - dexter

- % Done changed from 100 to 90

Its not fully done yet. There is one patch that is still in review: <https://gerrit.osmocom.org/#/c/osmo-bsc/+9940/>

#28 - 08/20/2018 07:31 PM - dexter

- % Done changed from 90 to 100

The patch is through. I think we are now done here.

#29 - 09/03/2018 07:40 AM - dexter

- Status changed from Resolved to In Progress

- % Done changed from 100 to 90

The TTCN3 test (TC_lcls_gcr_bway_codec_mismatch) that tests this still does not pass, we need to check why.

#30 - 09/12/2018 03:16 PM - dexter

The reason why TC_lcls_gcr_bway_codec_mismatch does not pass anymore is because there were API changes and LCLSs was using abandoned struct members. This is now fixed.

See also: <https://gerrit.osmocom.org/#/c/osmo-bsc/+10905>

#31 - 09/19/2018 09:49 AM - dexter

The patch is through, but on jenkins one can see that now the following LCLS related tests are failing:

TC_lcls_connect_break
TC_lcls_connect_clear
TC_lcls_gcr_bway_connect
TC_lcls_gcr_bway_connect_hr

This might be unrelated to this issue, but we certainly must check this. It is also very suspicious that just when TC_lcls_gcr_bway_codec_mismatch stops failing the others start to fail.

#32 - 09/24/2018 03:59 PM - neels

- Related to Feature #3586: support LCLS for inter-BSC handover added

#33 - 09/30/2018 08:50 AM - laforge

- Status changed from In Progress to Resolved

- % Done changed from 90 to 100

I'll close this issue as LCLS support is implemented. That fact that either our test suite or the implementation still has a bug doesn't change that fact.

#34 - 10/01/2018 07:55 PM - dexter

Yes, lets close it. The TTCN3 tests also pass fine now.

#35 - 10/22/2018 01:18 PM - msuraev

- Related to Bug #3659: LCLS directly between BTSs added