**STP Demo: Summary**

STPs are used to mimic real-life scenarios. Using the software and the hardware we can create certain scenarios that could lead to break the connection between the tower and UE.

**(figure)**

The system shows the connection between the DU, RU, CCN, and UE.

* We use MOSHELL to communicate with the ENodeB (DU) and the Radio (RU).
  + For example, if there is a problem with the radio unit we use MOSHELL to communicate with the ENodeB to resolve the issue of the radio connection.
    - 5G technology, we can communicate with RU and DU separately( RU through telnet port, DU through moshell (although you can also communicate with DU through **telnet** but ideally we do it through moshell)
  + To communicate with the ENodeB, we use the IP of that EnodeB 🡪 SSH the IP address 🡪 go to that ENodeB.
    - Not quite, to log into DU (enb) we use moshell <IP>
    - **If it’s KI or LI node, we would log in to tgf worker and then moshell** 
      * **SSH -XY <TGF WORKER>**
      * Moshell <ENODEB IP>
  + Every EnodeB has its certain IP address and info
    - Correct. You can find them on ERIS ( which I’ll show you once you have you laptop)
  + To get the info of a certain enodeB we use WIKI… there we can find the IP and other important info.
    - Wiki should be updated with all the updated IP, however it is humanly manage resource so might or might not up to date. We should be using WIKI to find all the UP but if we want to verify if the test really uses those UP we can check the test screenlog (You can see the sample I sent you guys today)
* On the cells we have alarms (major and minor but we mostly care about major) … the alarm indicate if the cells are not working or there is an issue with it… again, we use MOSHELL to fix the issue.. if cant solve it, then make a ticket.
  + It does not have to be the cell, the alarms can be causes by any components of ENODEB, either the wire is loosened up, or MO (managed object) value is invalid or something, but cell is one of the common and major alarms we deal with
* QxDM: tool to create traffic modules, run traffic modules, or edit traffic modules.
  + We use QXDM to see the throughout of Qualcomm devices. It is a software provided by Qualcomm. Qualcomm are the actual UE we have set up in some of our labs
* When we do tests we are usually testing the traffic, handoffs, and capacity of cells.
  + Agree. More info on what we tested are provided on the wiki of every test case. To understand on what exact scenario the testcase is testing. But this is right.
* The UEs use Prisma or Viavi or other software.
  + Viavi or Prisma are UE simulator. UE does not use these simulators. Prisma or Viavi name of the software we use to simulate UEs (more 100 UEs). You can also see more about it on google. To understand what the product are.
* The UEs are ran on windows or OS.
  + Correct. We can access UE PC remotely as well.

**Some of the commands that were used on the VM**

An Overview on what TGF is:

TGF is a test execution engine. Supporting GTE and JCAT TAF in a pool environment with a global queue.  
A test scenario is described via the tgjob CLI, stored in the database, and picked up by the execution engine.  
Lab test equipment is grouped together as an "STP", and TGF launches one "worker" per STP.  
Read more about [Job\_Selection](https://wiki.lmera.ericsson.se/wiki/TGF/Using" \l "Job_Selection" \o "TGF/Using) from the [Queue](https://wiki.lmera.ericsson.se/wiki/TGF/Using#Queue).

On this wiki you can find information on how to [**use TGF**](https://wiki.lmera.ericsson.se/wiki/TGF/Using) and how to [**get help**](https://wiki.lmera.ericsson.se/wiki/TGF/Help).

**Where does TGF come from?**  
TGF was created in 2012 to tackle the following problems:

* Poor STP Utilization
* Faults in Everyday Repetitive Tasks i.e. Waste
* Poor Overview of Bottlenecks i.e Inefficiency
* No Test Scope Awareness i.e. Redundancy

Currently, more than 2000 STPs are included in TGF

**tgr** + **(name of moshell) - tgr <STP NAME>**

**tgw** + **(name of moshell) – tgw <STP NAME>**

**ssh -XY**  + **(IP)** 🡪 I think its used to go to the ip address that we provide

**Usually we use SSH to log in to TGF worker or to RU or DU.**

**Moshell** + **(IP)** 🡪 I think it’s the same as SSH

**Not same as ssh coz it’s mainly to log in to enodeb**

**Which Moshell**

**Tgq + (name of moshell) – tgq < STP NAME>**

**Tgqq** + **(name of moshell) – tgq <STP NAME>**

**clear**

**IMPORTANT COMMAND:**

**LT ALL (lt all)**🡪 which stands for Load **MO** all, load all allows us to play around within the enodeB features and MOs … since SSH only takes you to the enodeb that you address but does not allow you to edit or use its features**. It is not used to RESET**. (then how do you reset??)

**Do you mean restart?**

**MO or MOM:** which stands for Manage objects. Objects that are related to the enodeb. Not too sure ask.

**I’m not sure what do you mean here?**

**Format of Moshells name:** listpXXXX or KienbXXX or otenbXXXX

The first to letter is the ~~city~~ **lab site**(li, ki, Ot Ottawa), enb stands for EnodeB, then four digits.

**Not always have to four digits. For example: sekiitrbs00419 – lab in KISTA**

**Questions:**

* Command questions

Tgr

Tgw

Tgq

Tgqq

**Tglock force lock <STP>   
tglock force lock <STP> owner=ltegct**

Which moshell

* What is CTX?
* When do we use Lt all and when do we not use it?
  + **Lt all loads all the MO on the enobeb, you do it as soon you moshell into the enodeb. It’s the first command you do.**
* How do we reset the enodeb after a certain edit? Do we have to?
  + **What do you mean by after a certain edit? Sorry can you explain here?**
  + **This is more enodeb debugging or troubleshooting enodeb**
* How does this process relate to the delivery status page?
  + **It’s part of the STP that our flow uses. So our job is to make sure they’re correctly configured and works.**
* What exactly are MOs? Example ?

OTENB5111> **lt all**

181101-13:54:43 10.122.107.111 18.0j MSRBS\_NODE\_MODEL\_330.27706.46\_f74f\_TESTMOM stopfile=/tmp/22701

Connected to 10.122.107.111 (ManagedElement=1)

Last MO: 7176. Loaded 7176 MOs. Total: 7177 MOs.

OTENB5111> st field

181101-13:54:51 10.122.107.111 18.0j MSRBS\_NODE\_MODEL\_330.27706.46\_f74f\_TESTMOM stopfile=/tmp/22701

===================================================================================

Proxy Adm State Op. State MO

===================================================================================

208 1 (UNLOCKED) 1 (ENABLED) Equipment=1,FieldReplaceableUnit=1

225 1 (UNLOCKED) 1 (ENABLED) Equipment=1,FieldReplaceableUnit=2

232 1 (UNLOCKED) 1 (ENABLED) Equipment=1,FieldReplaceableUnit=3

===================================================================================

Total: 3 MOs

OTENB5111**> st sector**

181101-13:54:53 10.122.107.111 18.0j MSRBS\_NODE\_MODEL\_330.27706.46\_f74f\_TESTMOM stopfile=/tmp/22701

===================================================================================

Proxy Adm State Op. State MO

===================================================================================

186 1 (ENABLED) ENodeBFunction=1,SectorCarrier=1

187 1 (ENABLED) ENodeBFunction=1,SectorCarrier=2

251 1 (UNLOCKED) 1 (ENABLED) NodeSupport=1,SectorEquipmentFunction=1

252 1 (UNLOCKED) 1 (ENABLED) NodeSupport=1,SectorEquipmentFunction=2

===================================================================================

Total: 4 MOs

OTENB5111> **st cell**

181101-13:54:56 10.122.107.111 18.0j MSRBS\_NODE\_MODEL\_330.27706.46\_f74f\_TESTMOM stopfile=/tmp/22701

===================================================================================

Proxy Adm State Op. State MO

===================================================================================

32 1 (UNLOCKED) 1 (ENABLED) ENodeBFunction=1,EUtranCellFDD=1

74 1 (UNLOCKED) 1 (ENABLED) ENodeBFunction=1,EUtranCellFDD=2

===================================================================================

Total: 2 MOs

OTENB5111> **ala – alarm**

181101-13:54:57 10.122.107.111 18.0j MSRBS\_NODE\_MODEL\_330.27706.46\_f74f\_TESTMOM stopfile=/tmp/22701

Collecting Alarms...

====================================================================================================================

Sever Specific Problem MO (Cause/AdditionalInfo)

====================================================================================================================

Min Calendar Clock All NTP Servers Unavailable SysM=1 (No NTP server configured, or no NTP service available)

Warn Calendar Clock NTP Server Unavailable SysM=1,NtpServer=1 (Configured NTP server is not used as basis for time of day clock)

Alarm Id : 6

Event Time : 2018-10-31 22:35:57

Perceived Severity : Minor

Managed Object Class : SysM

Managed Object Instance : ManagedElement=1,SystemFunctions=1,SysM=1

Specific Problem : Calendar Clock All NTP Servers Unavailable

Probable Cause : ReceiveFailure

Additional Text : No NTP server configured, or no NTP service available

Acknowledged by :

Acknowledgement Time :

Acknowledgement State:

System DN :

Notification Id : 6

Additional Info :

Alarm Id : 5

Event Time: 2018-10-31 22:33:57

Perceived Severity: Warning

Managed Object Class: NtpServer

Managed Object Instance: ManagedElement=1, SystemFunctions=1, SysM=1, NtpServer=1

Specific Problem: Calendar Clock NTP Server Unavailable

Probable Cause: Unavailable

Additional Text: Configured NTP server is not used as basis for time of day clock

Acknowledged by:

Acknowledgement Time:

Acknowledgement State:

System DN:

Notification Id: 5

Additional Info :

>>> Total: 2 Alarms (0 Critical, 0 Major)

OTENB5111>