OpenAirInterface Test Framework

Mobile Communications Department
Eurecom

Unleashing the potential of open-source in the 5G arena
Agenda

- Workflow and continuous integration
- Testbenches
  - Eurecom
  - TCL
- Testscripts
Gitlab basic workflow

- **Main branch: develop**
  - All branches (features, bugfixes, etc) need to branch off from and stay in sync with develop
  - **Weekly integration cycle:**
    - Tag current version
    - Collect all merge requests in integration branch
    - Test integration branch
    - If tests pass, accept merge requests, otherwise reject
Develop Branch

Feature Branch

Fix Branch

Integration Branch w15

Branch Freeze

Tag16.12: Stabe-version for w12

Tag16.13: Stabe-version for w13

Tag16.14: Stabe-version for w14

Commit 1

List of commits candidates should defined every Thursday with weekly basis

Auto – Test UE + ENB:
- test setup + branch merge should done Friday
- tests run during the weekend

→ Regression: remove suspicious commit (for example commit 1)

Re-Run Auto – Test UE + ENB:
→ Test Pass

→ Push only commit 2

Integration Branch deleted

Tag16.15: Stabe-version for w15
## Testbench @ Eurecom

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Nokia EPC</td>
<td>amerique</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OAI EPC</td>
<td>nano</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>eNB1</td>
<td>hutch</td>
<td>Intel Xeon, 4 cores</td>
<td>USRP B210</td>
<td>band 7 (duplexers)</td>
</tr>
<tr>
<td>eNB2</td>
<td>mozart</td>
<td>Intel Xenon, 10 cores</td>
<td>USRP X300</td>
<td>band 7 (duplexers),</td>
</tr>
<tr>
<td>eNB3</td>
<td>calisson</td>
<td>Intel Core-i5, 4 cores</td>
<td>USRP B210</td>
<td>band 38 (circulator)</td>
</tr>
<tr>
<td>RRU</td>
<td>superserver</td>
<td>Intel Atom, 8 cores</td>
<td>USRP B210</td>
<td>band 7 (duplexers)</td>
</tr>
<tr>
<td>RCC</td>
<td>starsky</td>
<td>Intel Xenon, 4 cores</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UEs</td>
<td>stevens &amp; mozart</td>
<td>Sony Experia M4 (x2), Bandrich FDD dongle, Huawei TDD dongle</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 RF cages</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Testbench @TCL

- **OAI eNB & UE: USRP B210**
  - Cabled, SISO
  - "noS1" test mode (w/o EPC) to test layer 3 throughput (iperf)
  - "phy-test" mode (w/o MAC & above) to test PHY throughput
  - Integrated in autotests

- **Amarisoft eNB & OAI UE,**
  - Cabled, 2x2 MIMO
  - Full connection procedure (EPC), layer 3 throughput (iperf)
  - Not integrated in autotests, for manual feature testing
Test files location (openairinterface5g)

- Wiki page for Test Framework
  - https://gitlab.eurecom.fr/oai/openairinterface5g/wikis/OAITestSet up

- README.txt for overview of all the tests
  - Openairinterface5g/cmake_targets/autotests/README.txt

- Test case classes:
  - Compilation: Test cases to test compilation
  - Execution: test cases to test oaisim, NAS securiry tests, dlsim, ulsim tests
  - Lte-softmodem: test cases for dynamic eNB/UE/EPC interaction
    - EPC can be OAI or Nokia
    - UE can be OAI or COTS
**Bash script tests**

- To be run on locally by dev
- For “compilation” and “execution” test
- Run from shell using `build_oai`
  
  ```sh
  ./build_oai -s --run-group "0101* 0102*"
  
  Run tests matching 0101* 0102*.
  ```

- Log/ dir for output of test results
  
  - Openairinterface5g/cmake_targets/autotests/log
Python script tests

- To be run only by Gitlab CI, on Eurecom testbench
- For all tests: “compilation,” “execution,” and “lte-softmodem”
- Test_case_list.xml for description of all the tests
  - Openairinterface5g/cmake_targets/autotests/test_case_list.xml
- cmake_targets/autotests/run_exec_lte-softmodem_tests.py
- Invoked manually by Eurecom on integration branch each weekend
  - Automatic trigger on each commit creates too much overhead
- Log/ dir for output of test results
  - On NFS share (not accessible from outside yet):
    /mnt/sradio/TEST_RESULTS/develop/<commit_hash>