



Core Network Testing

Service Testing independent from Radio Problems

Testing without Radio Access

In some cases the active test approach of using test mobiles on an air interface can be counterproductive because radio specific problems can disturb results without adding any value. A typical example is roaming testing, where the problems come from the core network. For such cases there is benefit in testing the services by means of simulating the radio access part, connecting directly to the core components such as the MSC, SGSN etc. without involving the radio access components.

Regression Testing

Another instance where connection direct to core network components is required is where tests need to be executed in a very short time. This is typically the case when network components like the MSC or SGSN have been updated or where the influence of a high number of concurrently connected subscribers has to be checked immediately and automatically. Such kinds of regression testing can usually be executed in less than 20 minutes whilst still covering all services and subscriber profiles.



Inside out – core network testing can save money

Typical use cases for testing without radio access part:

- Interactive Voice Response incl. interactive announcement tests
- Testing of Supplementary Services
- Testing of IN Services
- Inbound Roaming tests
- Load Tests on Core Network components
- Billing Testing via CDR tests
- Data Content Testing
- Handset Simulation
- etc.

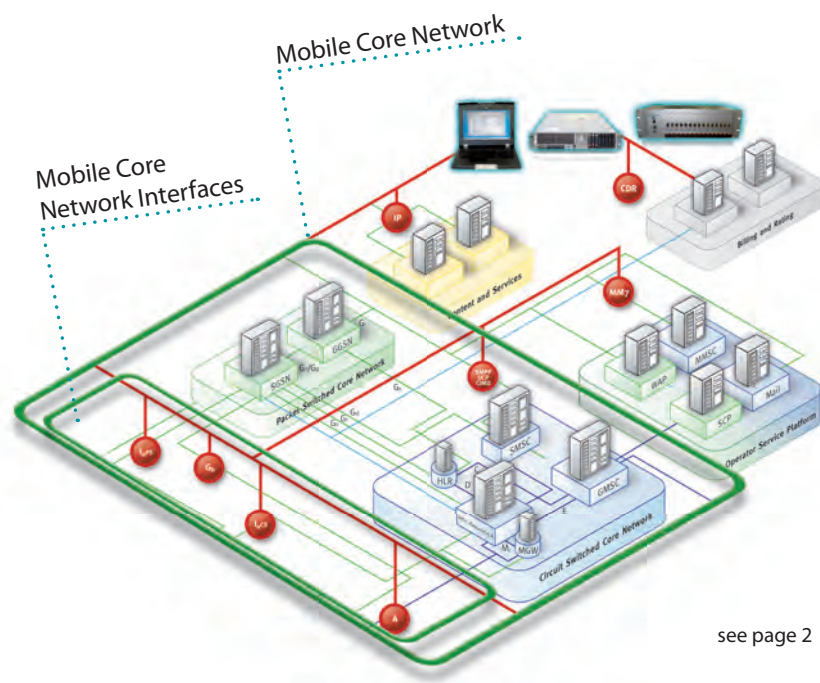
Typical use cases for regression testing without radio access part:

Automated regression testing after upgrade/updates on:

- MSC (via GSM A)
- MSC and fixed network switches (via ISUP)
- SGSN (2.5G) via Gb
- SGSN via I_uCS and I_uPS
- SMSC, MMSC
- WAP-, Web-, Mail-Server
- SCP testing including IN services and prepaid services
- etc.

Typical advantages of core network testing:

- Not influenced by radio specific problems
- Fast execution of mass tests (e.g. thousands of tests in an hour)
- Quick results
- Detailed trace analyses with protocol stack information



see page 2