

Using a TTCN-3 middleware in TTCN-3 based testing

Vikas Pratap Singh
- TTCN-3 User Conference Asia 2007 -

1

© Nokia Siemens Networks

Vikas Pratap Singh / TTCN-3 User conference Asia 2007 / Beijing, China



Overview

Introduction

Test Process

A TTCN-3 Middleware

TcSkeletonV3

Results

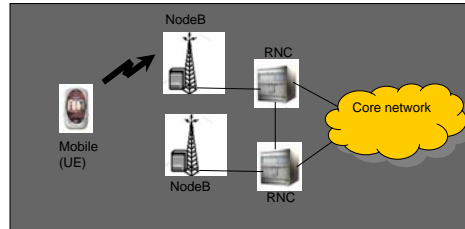
Things to remember



Introduction

UMTS

- Standards body : 3GPP
- Broadband mobile access for voice (TDM) and data (statistical multiplexing)
- Guaranteed QoS, can use IP for signaling

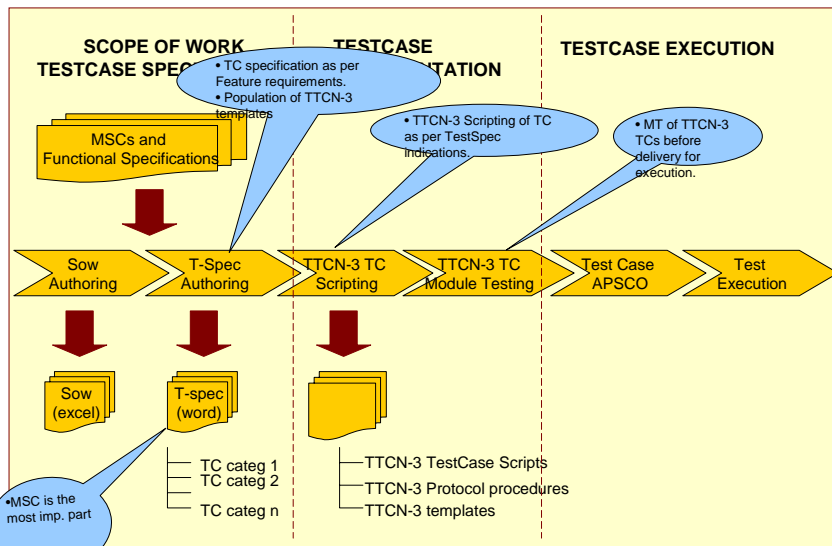


RNC: Radio Network Controller

- Owns and controls the radio resources (NodeB)
- Uses specialized procedures (e.g. handover control, admission control, power control...)



Test Process definition From SoW until TTCN-3 TC execution



Scripting Demands

- Script the data types.
- Script the templates.
- Design the data structure that might be needed by all the PTC's(e.g: ports and components etc)
- Create all the PTC's
- Connect them to the MTC and any other component as required.
- At the end of the testcase, tear down all the connections.
- Script different send and receive statements.
- Design an algorithm to assure that the exact sequence of the testcase is followed.
- Take care of the verdict handling
- Script the functions required to encode/decode incoming and outgoing data.
- Formulate the algorithm to handle various possible scenarios as, coregion, parallel behavior, alternative behavior etc.
- Etc
- Etc
- Etc.....



Remedy– A TTCN-3 middleware

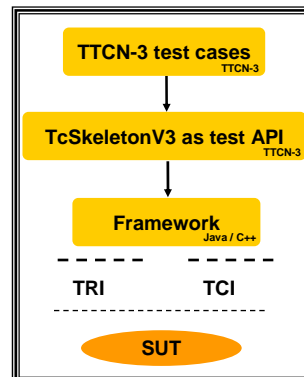
This middleware will be a library of functions written in TTCN-3 that would implement algorithms to handle the common behavior of the test system.

The middleware simplifies the task of the scripter massively in converting a test MSC into executable TTCN-3 scripts and helps him focusing on the correct translation of the test specification instead of spending time on implementing the test execution logic.

Remedy– A TTCN-3 middleware..... TcSkeletonV3

➤ This middleware was developed by a team with the following members:

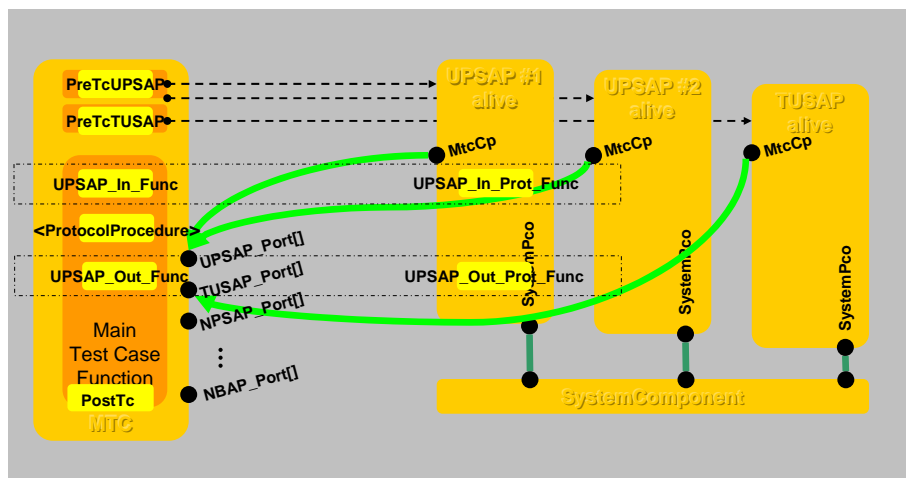
- Andy Rauland (Ulm)
- Andreas Ulrich (Munich)
- Zhang Chao (Beijing)
- Vikas Pratap Singh (Bangalore)



Aim of TcSkeletonV3

- To make a middleware fully compliant with TTCN-3 standard.
- To provide simple scripting interface to users.
- To provide easy mapping of test cases written in high-level MSC's to TcSkeletonV3 user functions.
- Make it efficient in terms of compilation time.
- Provide the support for alternatives, parallel behavior, etc by means of simple API.

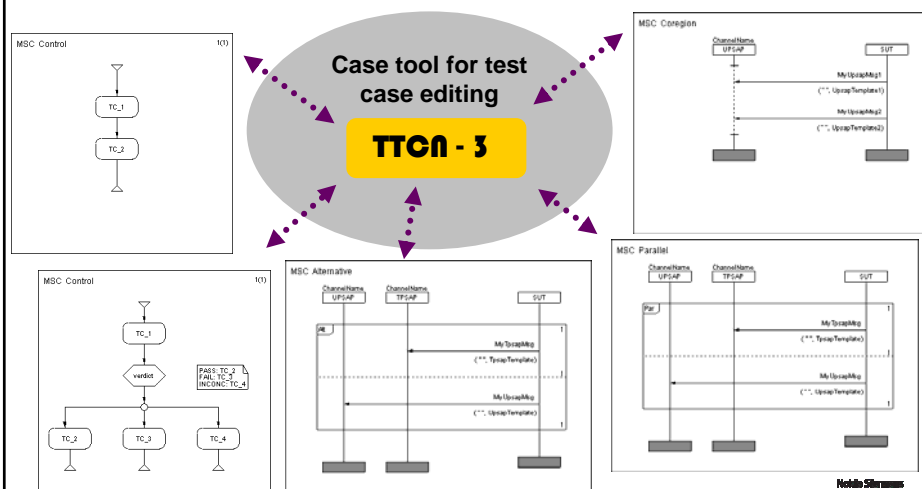
TcSkeletonV3-The Concept (Sequential behavior)



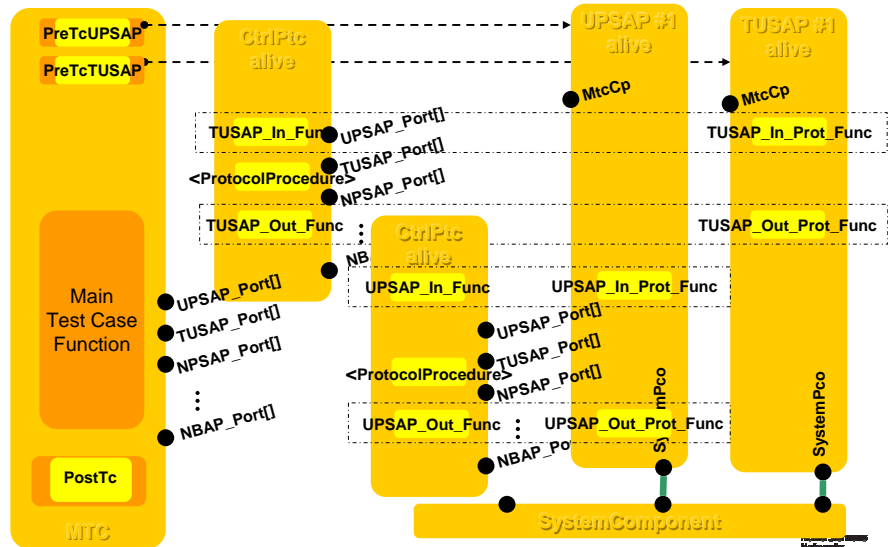
Some API's in detail

- PreTc<INTERFACE> (e.g: PreTcUPSAP)- Creates the PTC and connects it to the MTC.
- <INTERFACE> In Func(e.g:UPSAP In Func)- This function starts another function which receives, decodes and matches the incoming template .
- <ProtocolProcedure>: Here some intermediate functionality can be performed, like copying information from incoming message to outgoing template.
- <INTERFACE> Out Func(e.g:UPSAP Out Func)- This function starts another function which encodes the template and sends it out.
- PostTc: This function shall tear down all PTCs and check the queue towards SUT if empty

Features supported



TcSkeletonV3-The Concept (Parallel behavior)



13 © Nokia Siemens Networks Vikas Pratap Singh / TTCN-3 User conference Asia 2007 / Beijing, China

Results

- TcSkeletonV3 was used in our project by ~100 testers across 5 different sites and 3 different countries.
- This middleware reduced the complexity of scripting massively resulting in happy testers.
- Reusability: a high degree of reusability was achieved as the middleware itself is fully reusable.
- Another feature of TcSkeleton, called ANTI-TcSKELETON allowed the scripters to module test there testcases, thus reducing the scripting mistakes by about 95%.
- Currently this middleware is being used in LTE project.

14 © Nokia Siemens Networks Vikas Pratap Singh / TTCN-3 User conference Asia 2007 / Beijing, China

Things to remember

- Make sure that the scripting interface is simple, thus providing easy mapping of system requirements modeled in MSCs to middleware functions.
- Make sure that you stick to the latest edition TTCN-3 standards.
- Make sure that your middleware is as independent of any tooling as possible.
- Always keep automation in mind.
- Make sure that your middleware is optimized for compilation time.
- The middleware should be well documented.
 - A user's manual is always a good idea

Thank you for your attention