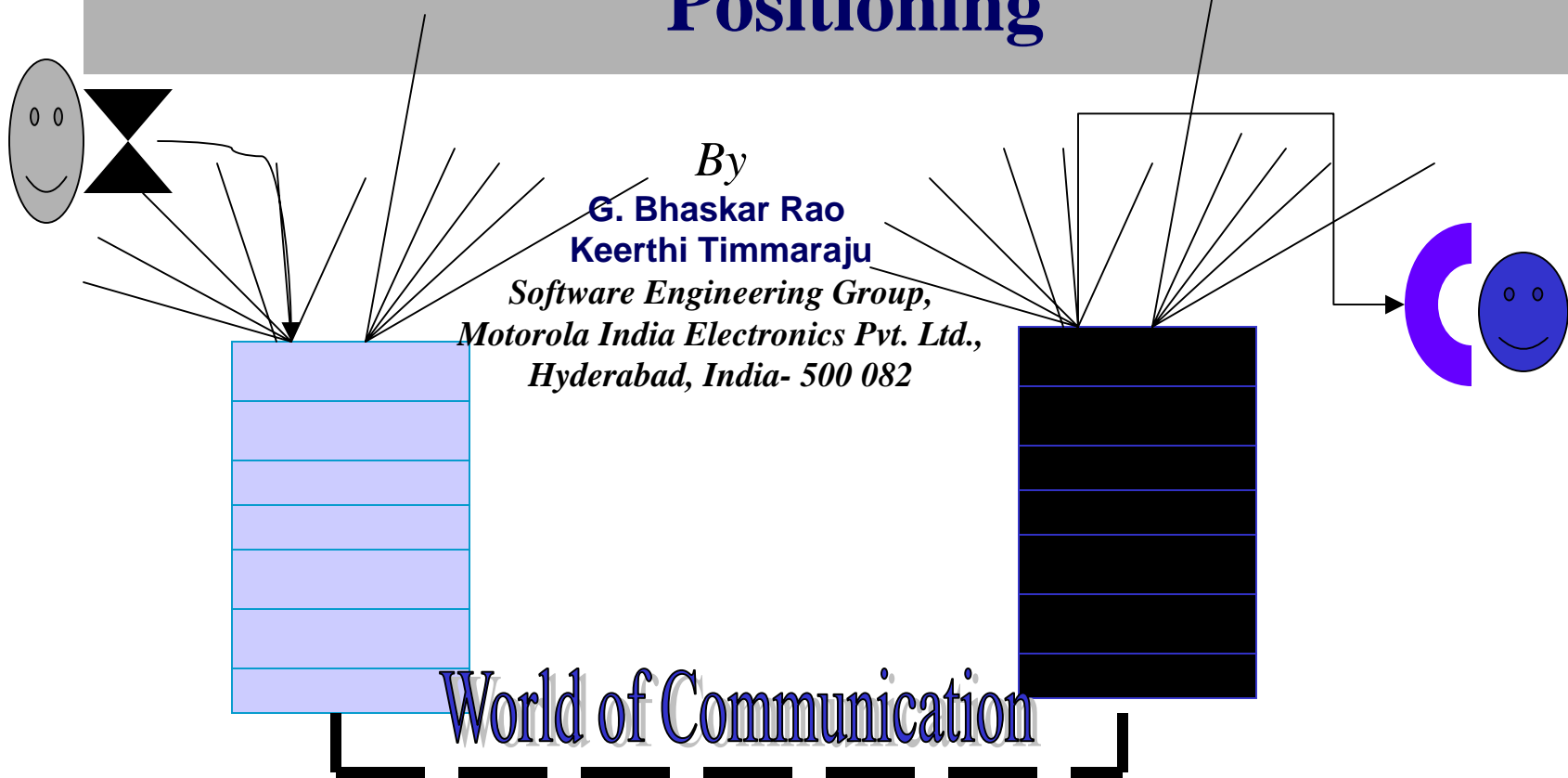


Load testing of MDA based HA Communication Network elements with TTCN-3: Approach, Benefits and Positioning



By
G. Bhaskar Rao
Keerthi Timmaraju
*Software Engineering Group,
Motorola India Electronics Pvt. Ltd.,
Hyderabad, India- 500 082*

World of Communication

Presentation outline

- MDE enabling benefits
- Network element architecture
- Testing of Network application
 - Integration, performance, load and etc. testing
- Benefits observed with TTCN-3
- Enhancements sought for TTCN-3
- 3GPP: TTCN-2 to TTCN-3 migration
- Conclusions

MDE Enablers and UML-2.0

- Model Driven processes, requirements, development and testing
 - Consistent and formal representation -Removes ambiguity
 - Graphical representation – clarity and understandability
 - Enables automation, maintenance, eases development
- UML-2.0
 - Motorola has used SDL (SDL influence can be seen in UML-2.0)
 - Has Model based support for
 - Business and Development processes
 - Requirements, Architecture and design
 - Verification and validation
 - Code generation

Integrated formal methods can give more benefits than conventional development

MDE vs Manual Development

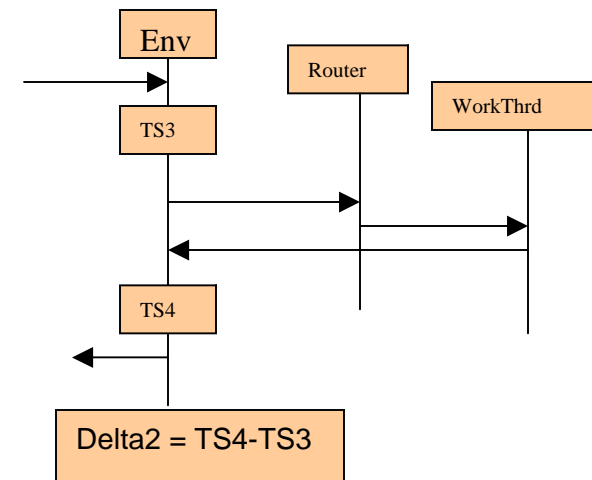
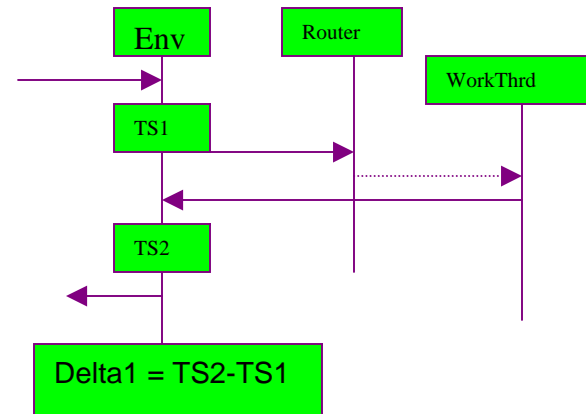
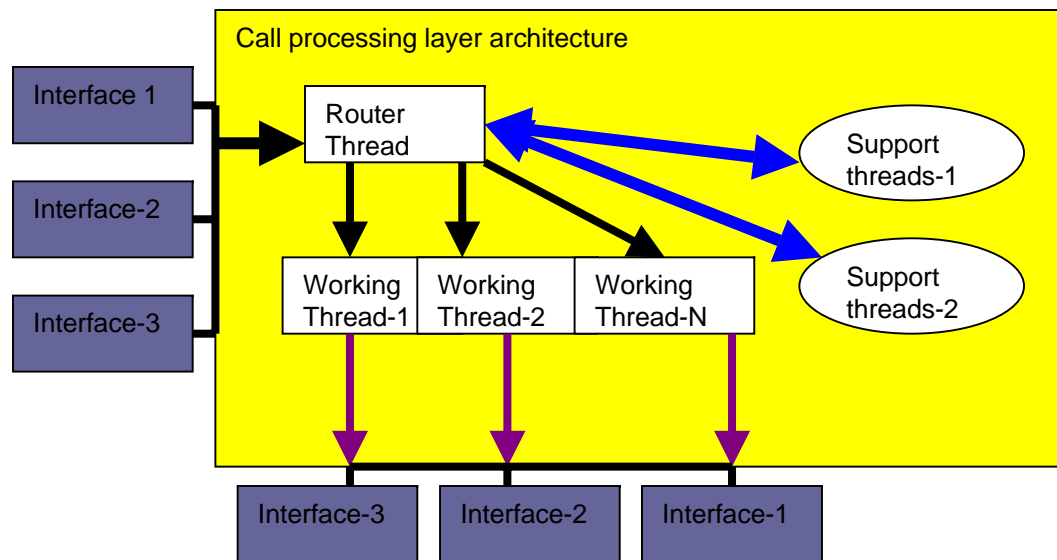
Attribute	Manual	With UML-II
Cycle Time	1	1.4X (less time)
In-process faults	1	2-3X (less faults)
Productivity	1	1.3 to 2X (increases)
Cost of Quality	1	2.2X (decreases)
Further automation capabilities	1	3X (increases)
Performance	1	Largely met
Memory consumption	1	20 to 50% overhead
Control over the process	Only during coding	Right from requirements
Reuse	Mostly code level	Right from requirements
Stability of the process	1	1.4X (increases)
Training	0	4 days
Practice	0	3 days

Networks conf 2005, India

Mobility Management Application :Requirements

- Critical network element
- Performance is important
- First time code generation from the UML-2.0 model
- Has to support configurable threads
- Benefits of MDE to be verified
- Prototype the approach to again confidence for the actual application development

MMII Prototype architecture

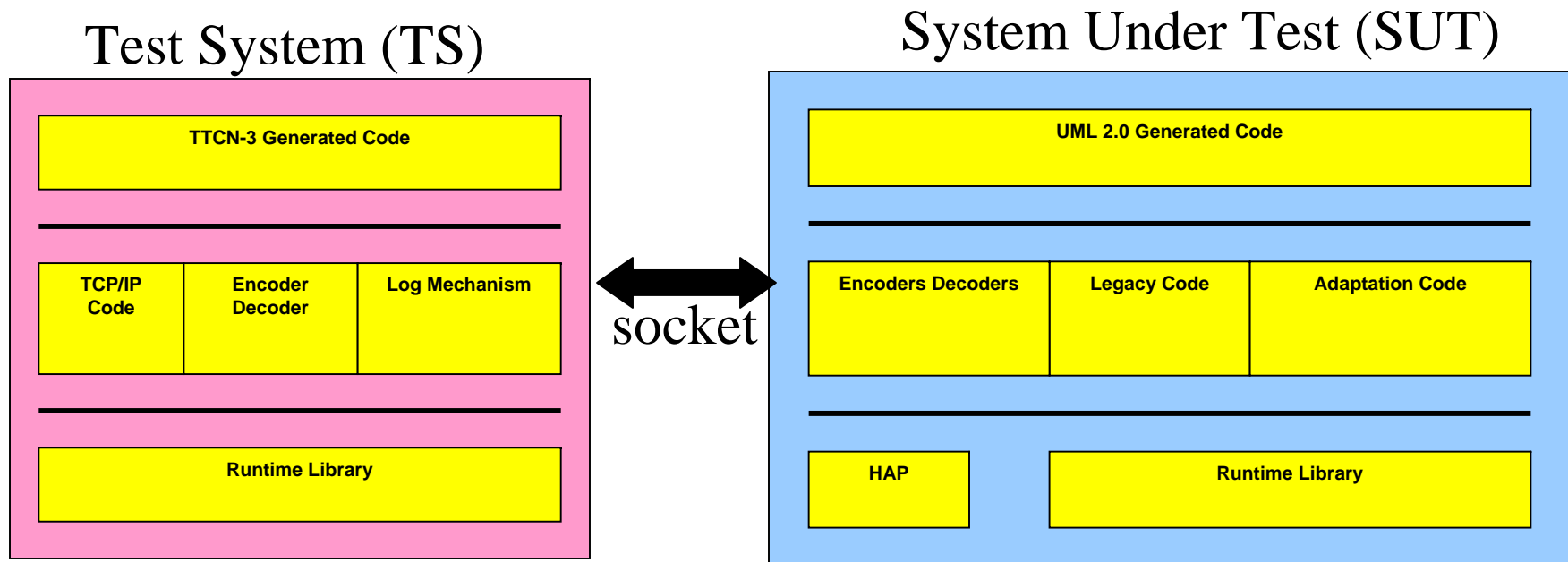


- Maximum of 6 threads
- Generated code efficiency indexes
Delta1 and Delta2 defined

Testing challenges

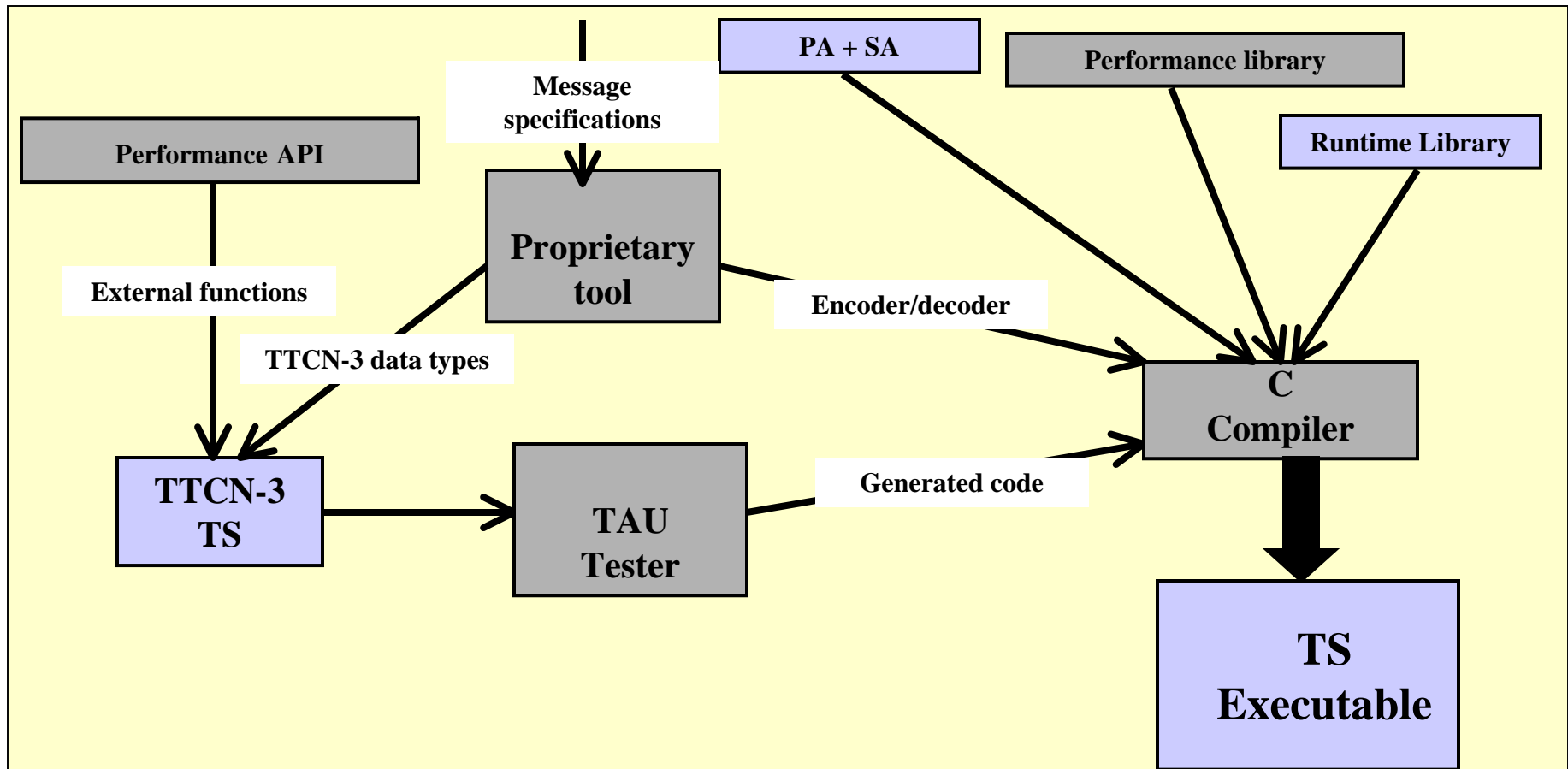
- Testing the prototype wrt its requirements (Delta1 and Delta2)
- Suitability to different types of testing
 - Performance
 - Reliability
 - Load
 - Portability (It needs to support Solaris and later migration to Linux)
- The impact of TTCN-3 on various business parameters such as productivity, quality, reuse and cycle time
- Further automation possibilities and integration with UML-2.0
- Tool maturity and support for testing

Basic test system architecture

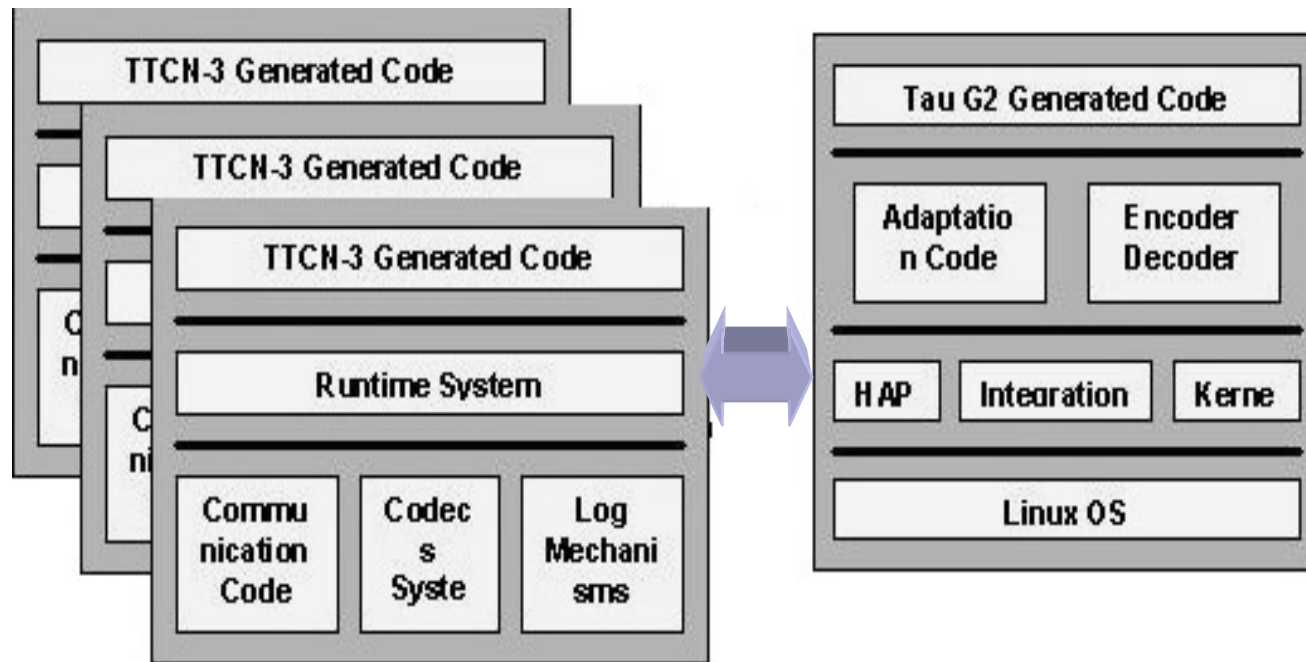


- Same architecture is used for integration, reliability and performance testing
- Simple TTCN-3 is used

Test system development



Load test system architecture



Multiple TSs are used to generate more load

Load test case with performance probing

```
/* Testcase to create PTCs for simulating several devices */
testcase load_perf_TC() runs on MTC system ATSI_
{
  map( self:OutPort, system:OutPort);
  loadGenTimer.start(loadRate); // generate as per the calibrated load
  alt {
    [ ] loadGenTimer.timeout {
      LoadGenArr [i] := PTC_.create;
      connect(self:mtc_ptcPort, LoadGenArr[i-1]:mtc_ptcPort);
      LoadGenArr [i].start(SimulateDevice(IDTYPE, DEVICE) ); // simulate a device and get time taken
      I := i+ 1 ;
      repeat;
    }
    [MAX_COMPONENTS > j] mtc_ptcPort.receive(sessTimeDiff_t?) -> value sessTimeDiff_var {
      timediff[sessTimeDiff_var.sessionId] := sessTimeDiff_var;
      j := j+1;
      repeat; // to be simulated upto max components
    }
    [MAX_COMPONENTS < j] mtc_ptcPort.receive(sessTimeDiff_t?) -> value sessTimeDiff_var{
      for(i:=0;i<j-1;i:=i+1){
        printTimeInfo(timediff[i].sessionId, timediff[i].timeDiff, timediff[i].cpuDiff); // logging of time info
      }
      setverdict(pass);
      stop;
    }
  }
}
```

18 May 2005

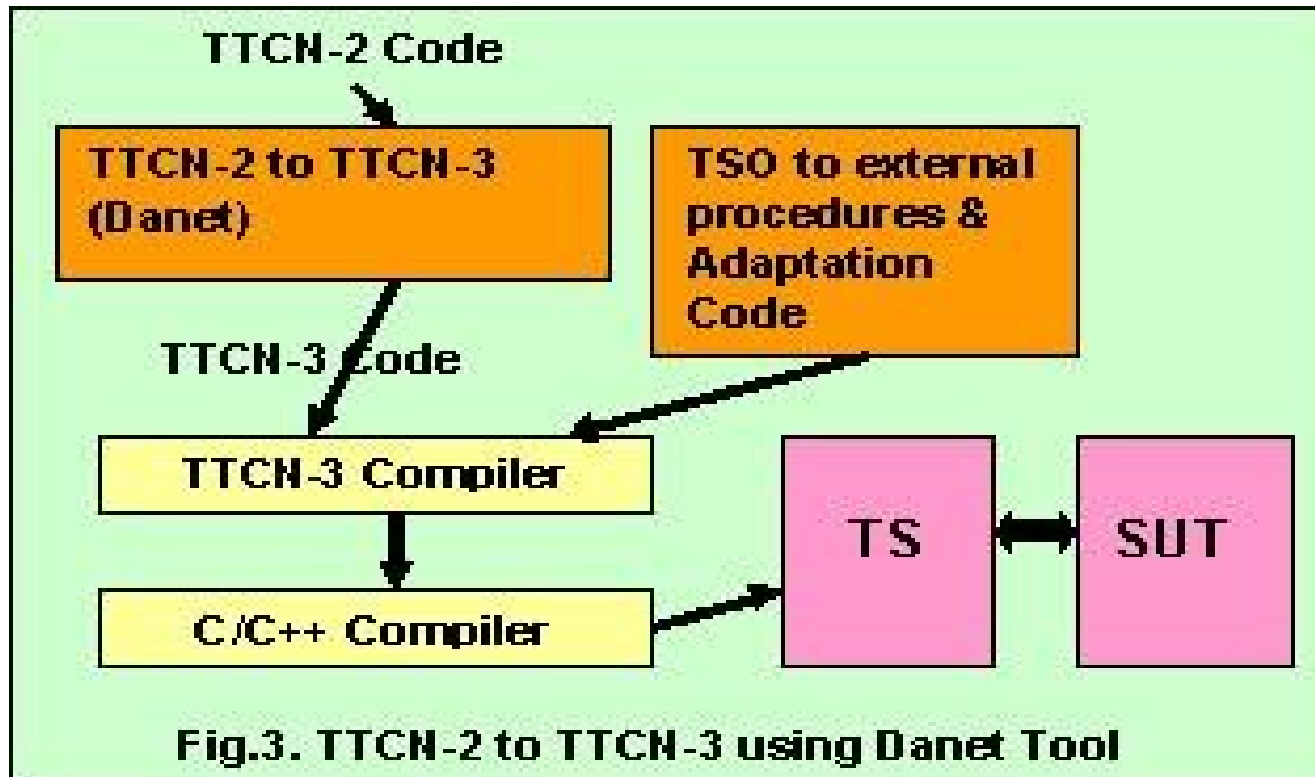
Benefits with TTCN-3

Business parameter	Conventional Testing	TTCN-3
Productivity	1X	2X (better)
Impact on Quality	1X	2X
Impact on CTR	1X	1.5X
Reuse	1X	2X
SUT coverage (same effort)	1X	2X

- Scaling of TTCN-3 to many integration test cases
- Reuse of the test system, communication and architecture across various types of testing
- Abstraction of communications ports will be handy

3GPP: TTCN-2 to TTCN-3 migration

3GPP: Migration to TTCN-3



Migration issues

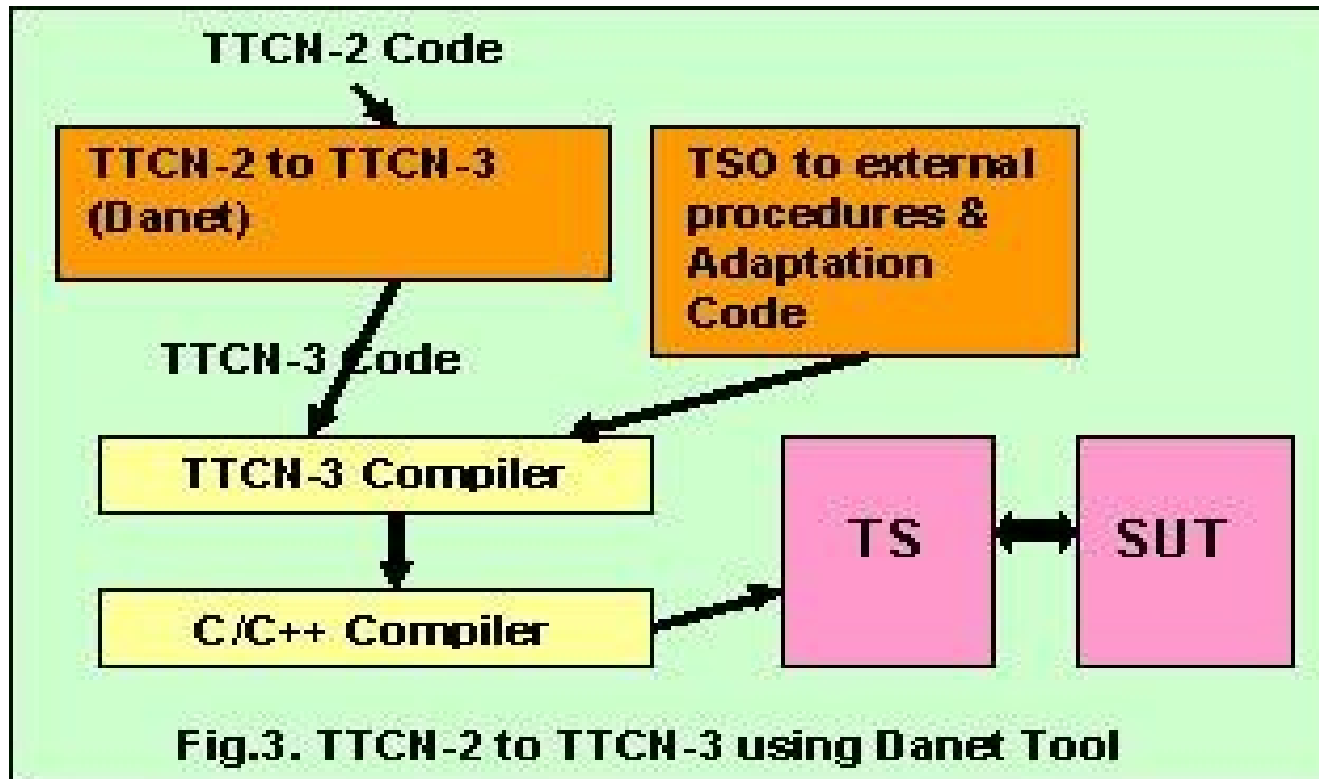
- Some issues were encountered
- Organization is not positive as there is no seamless transformation
 - Can ETSI support this?
- Conversation of the adaptation is huge
 - Many tools were developed for proprietary encoding/decoding
- Waiting for the 3GPP decision
- Tools were identified

Enhancements sought for TTCN-3

- Ability to control the timers of SUT from TS
- Un-necessary need to encode/decode inter PTC messages (co-ordination messages)
- Integer data type can be extended to have support for 1 byte, 2 bytes, 4 bytes and 8 bytes
- Shall have set of data types for which default encoding and decoding is defined

3GPP: TTCN-2 to TTCN-3 migration

3GPP: Migration to TTCN-3



Migration issues

- Some issues were encountered
- Organization is not positive as there is no seamless transformation
 - Can ETSI support this?
- Conversation of the adaptation is huge
 - Many tools were developed for proprietary encoding/decoding
- Waiting for the 3GPP decision
- Tools were identified

Conclusions

- The reuse was very good between performance, reliability, load and integration testing
- The environment has been reused for the actual product testing (integration and performance)
- In the beginning the encoder/decoders were written manually and later a tool was developed
- Internal training material has been prepared and thrust is being given for TTCN-3 usage in various domains
- Waiting for ETSI decision to migrate to TTCN-3 test cases (3GPP)

Q&A

EMAIL

Bhaskar Rao G. - bhaskarraog@motorola.com

Keerthi Timmaraju - keerthi.timmaraju@motorola.com