<testing_tech>

Tool Supported Test Frameworks in TTCN-3

Ina Schieferdecker, FOKUS
Theofanis Vassiliou-Gioles, Testing Technologies
Røros, June 2003

<testing_tech>

Outline

- Overview of TTCN-3
- Test Frameworks in TTCN-3
- Tooling Aspects

©I. Schieferdecker

<testing_tech>

TTCN-3 (1)

- Testing and Test Control Notation
- The standardised test specification and test implementation language
- Developed based on the experiences from previous TTCN versions
- Applicable for all kinds of black-box testing for reactive and distributed systems, e.g.,
  - Telecom systems (ISDN, ATM);
  - Mobile (telecom) systems (GSM, UMTS);
  - Internet (has been applied to IPv6);
  - CORBA based systems;
  - Java, XML, ...

©I. Schieferdecker

<testing_tech>

TTCN-3 (2)

- Enable testing of current and upcoming technologies
- Consolidate test concepts
- Wider scope of application
  - applicable to many kinds of test applications not just conformance, i.e. also for development, system, integration, interoperability, scalability … testing
  - applicable in the telecom and datacom domain
  - used both for standardized test suites … and as a generic solution in software development

©I. Schieferdecker

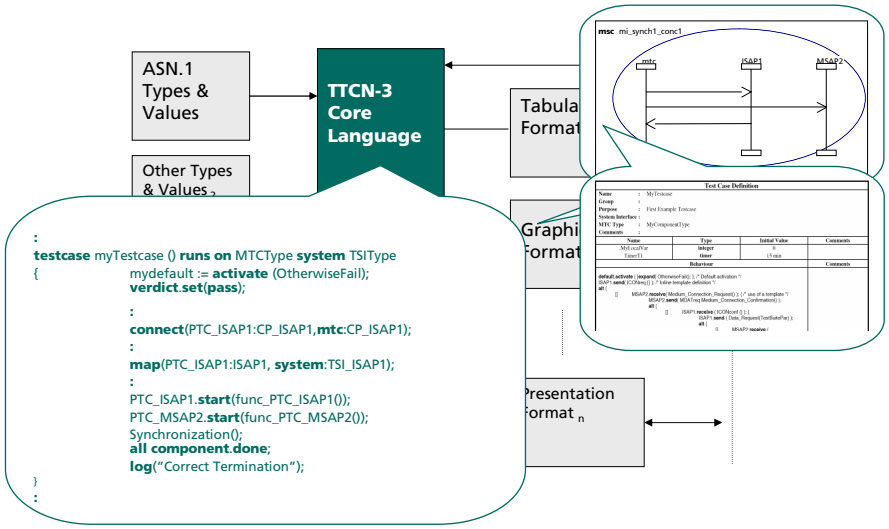
<testing_tech>

Main Aspects of TTCN-3

- Triple C
  - Configuration: Dynamic concurrent test configurations with test components
  - Communication: Various communication mechanisms (synchronous and asynchronous)
  - Control: Test case execution and selection mechanisms
- Improved
  - Harmonized with ASN.1
  - Module concept
- Extendibility via attributes, external function, external data
- Well-defined syntax, static and operational semantics
- Different presentation formats

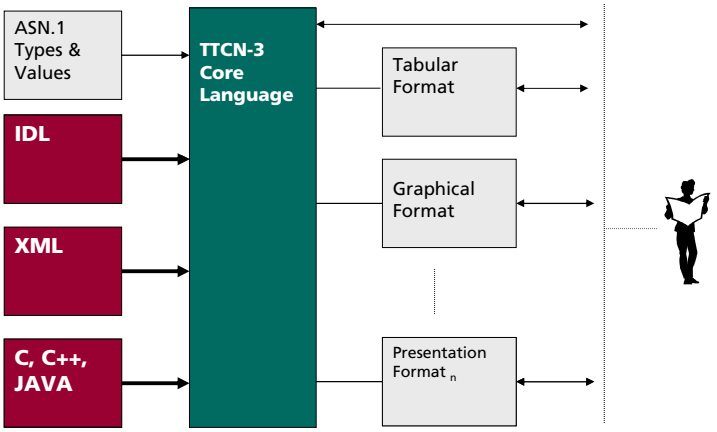
<testing_tech>

Overview on TTCN-3



<testing_tech>

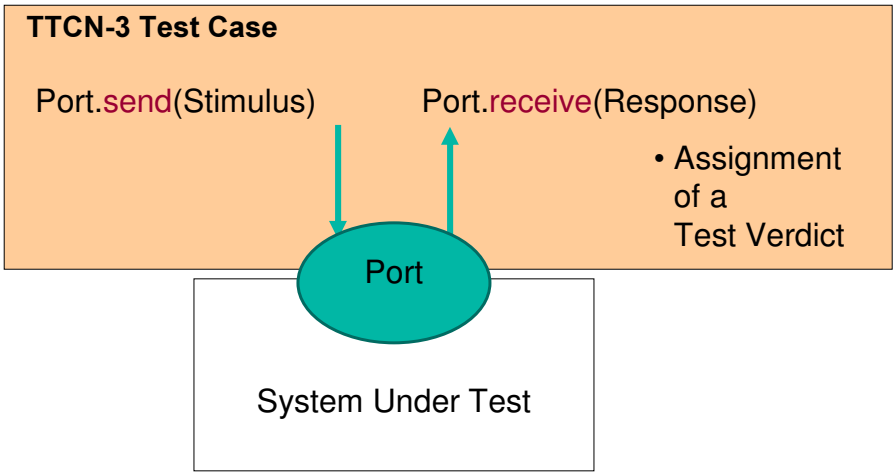
Overview on TTCN-3



©I. Schieferdecker

<testing_tech>

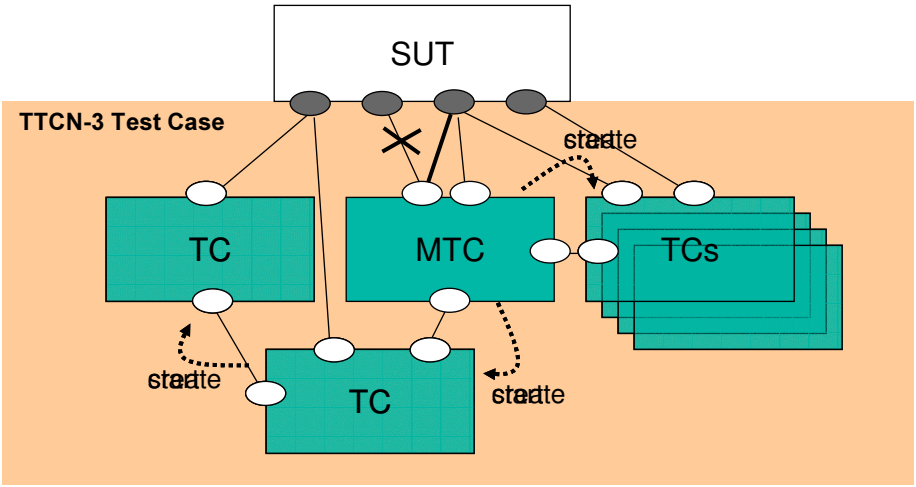
TTCN-3 – Based Black-Box Testing



©I. Schieferdecker

Component-Based Test System

<testing_tech>



©I. Schieferdecker

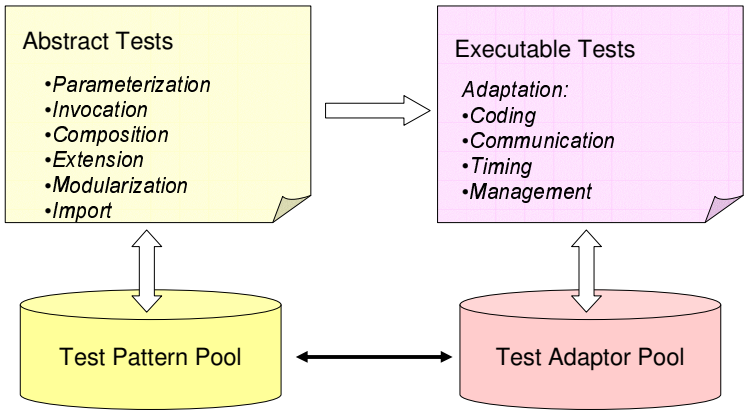
Basic Elements of TTCN-3

<testing_tech>

- Module covers declarations and control
- Templates (test data description) and matching mechanisms (pattern matching)
- Test configurations
  - Formally defined interfaces to the SUT
  - Dynamic creation of test component
  - Concurrency to describe distributed test setups
- Test cases
  - Small (complete) separate compilable programs
  - Share (type and data) information
- Test verdicts

©I. Schieferdecker

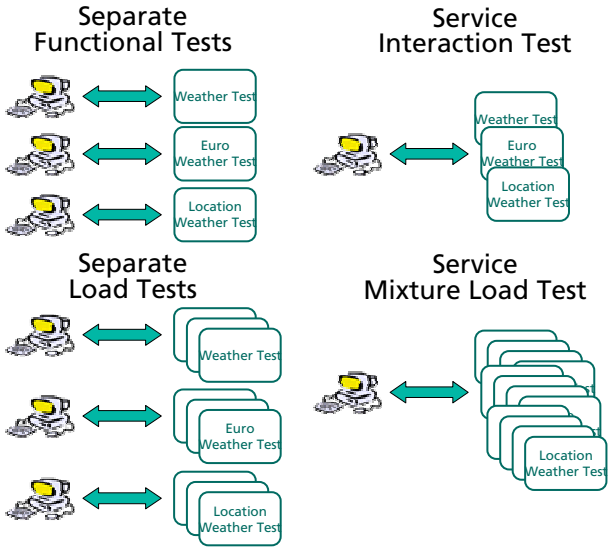
Test Frameworks



©I. Schieferdecker

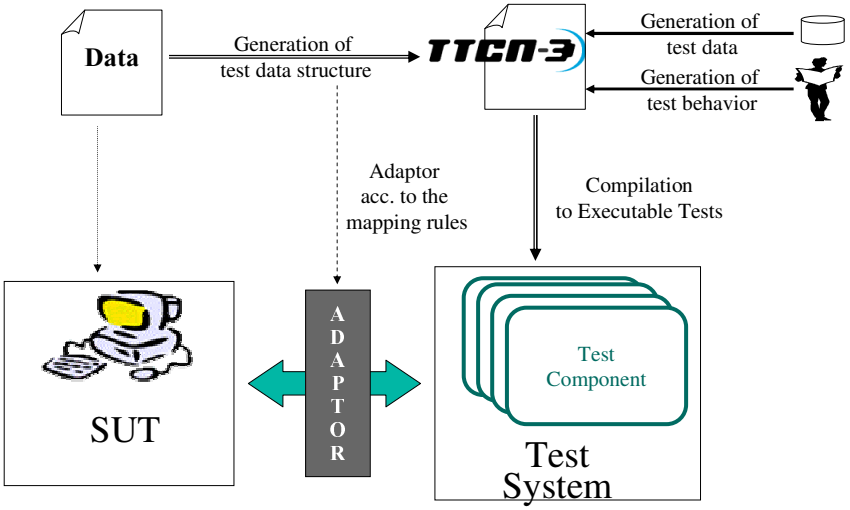
<testing_tech>

Test Frameworks



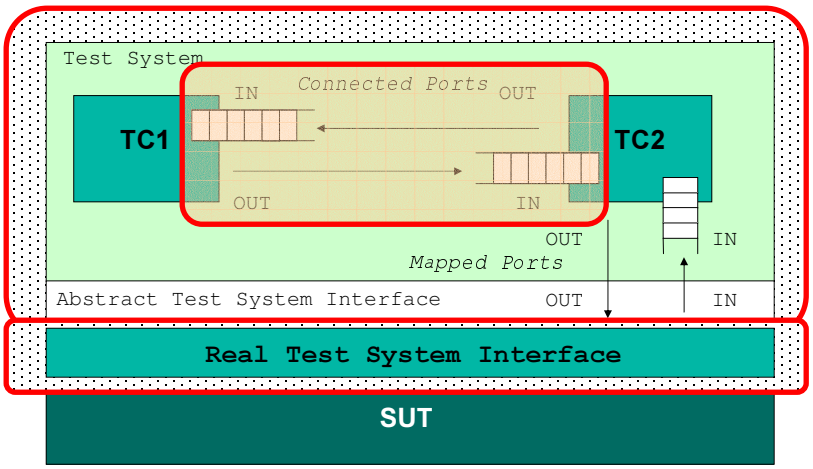
FMICS Workshop, June 2003: TTCN-3 Test Frameworks
©I. Schieferdecker

Overall Picture



©I. Schieferdecker

Test Configuration



©I. Schieferdecker

The TTCN-3 Execution Interfaces

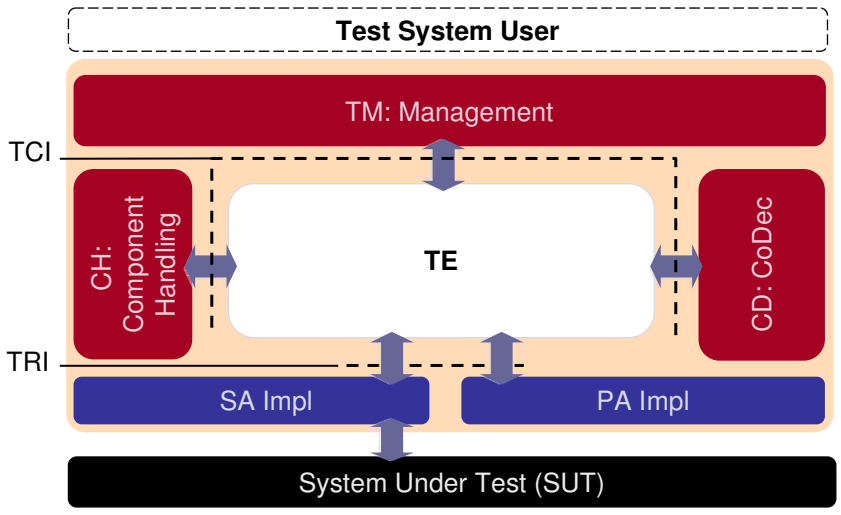
<testing_tech>

- Standardized adaptation for management, component handling and communication, external data representation with encoding/decoding and logging for local and distributed test setups
- Well-defined interfaces as a set of operations independent of the target, i.e. SUT, processing platform, implementation language, etc.
- Code from any compiler supporting/using this interface can be executed on any test platform/test device, which supports/uses this interface

➢ TRI – TTCN-3 Runtime Interface
➢ TCI – TTCN-3 Control Interfaces

©I. Schieferdecker

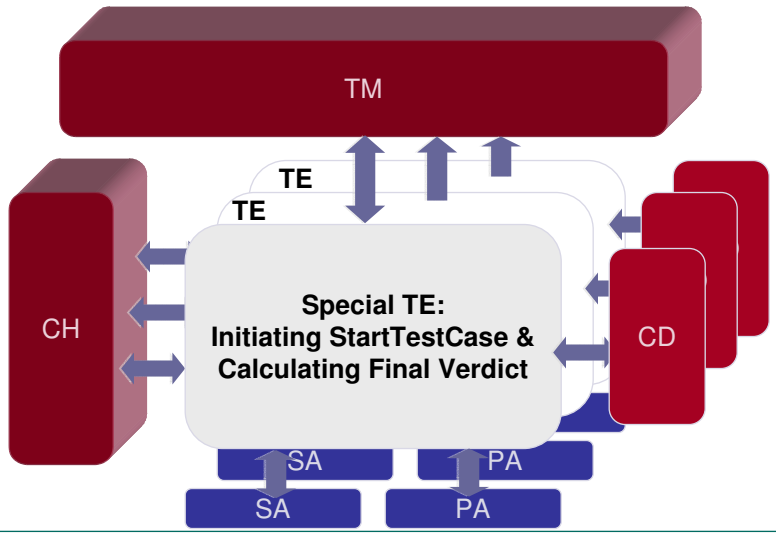
<testing_tech>



©I. Schieferdecker

Local and distributed test setups

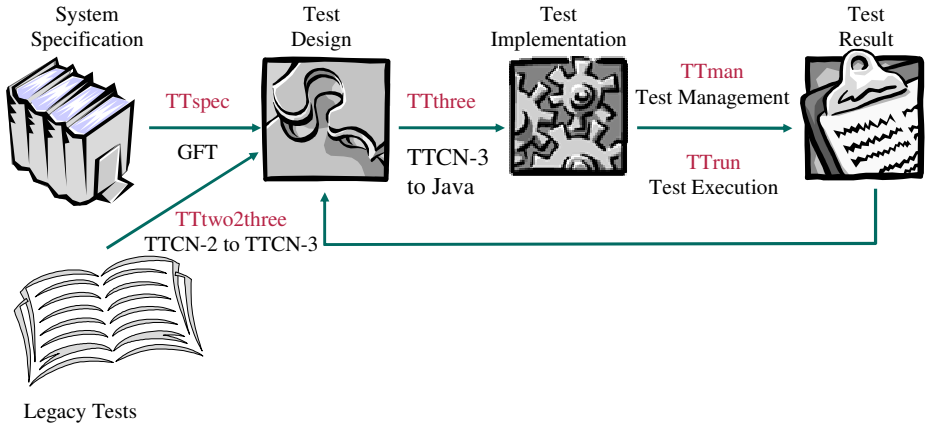
<testing_tech>



©I. Schieferdecker

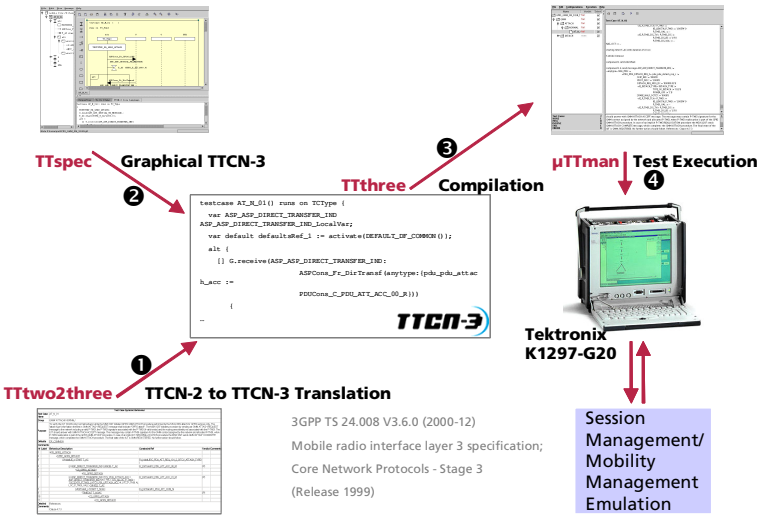
TTCN-3 Tool Chain

<testing_tech>



©I. Schieferdecker

An Example



©I. Schieferdecker

Conclusions

- TTCN-3 is a powerful test specification and implementation language
- Test frameworks enable
  - The customization and extension of test case collections
  - The reuse of test patterns in new and modified test contexts
- Test frameworks can be directly supported in TTCN-3 with test compiler and execution tools
- Further examples beyond GPRS to be studied

©I. Schieferdecker