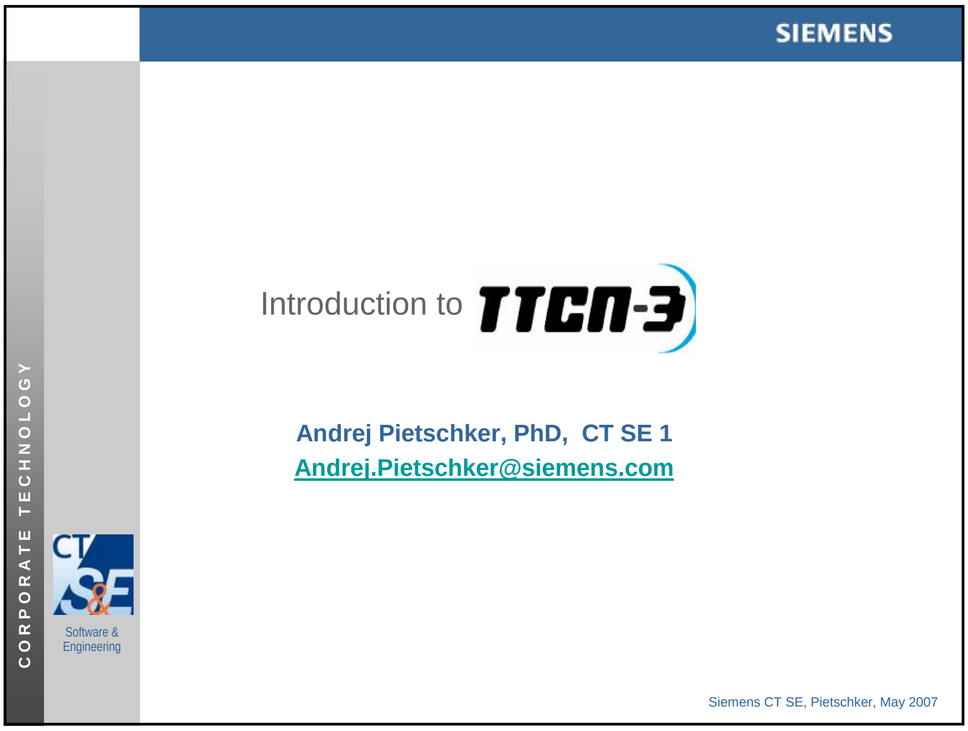


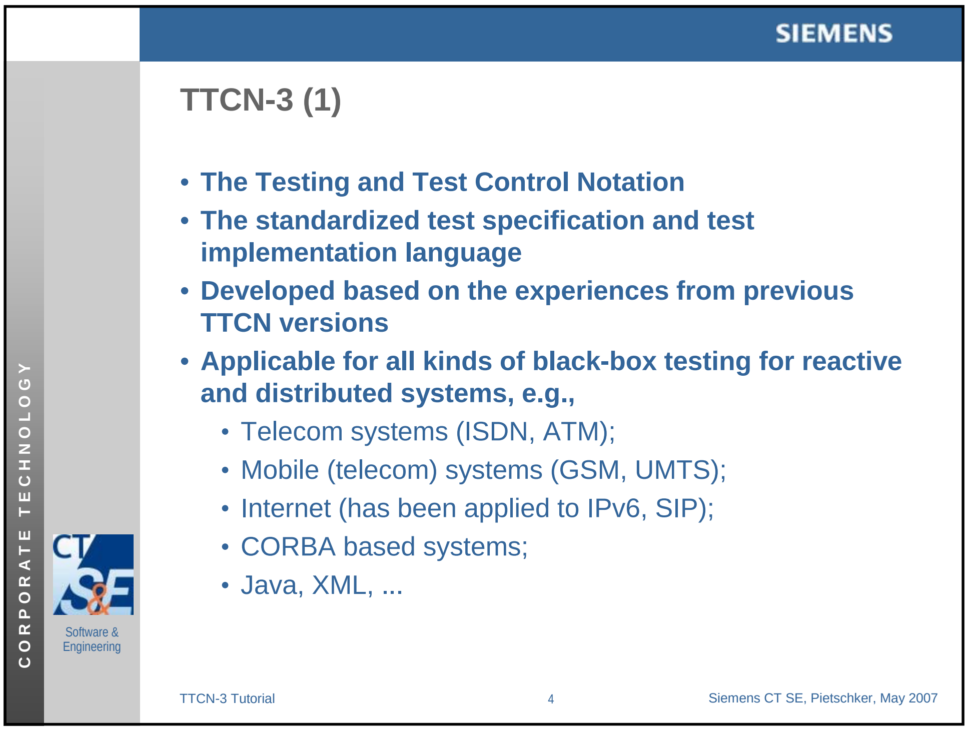
SIEMENS

CORPORATE TECHNOLOGY

CT S&E Software & Engineering

Introduction to TTCN-3

Andrej Pietschker, PhD, CT SE 1
Andrej.Pietschker@siemens.com



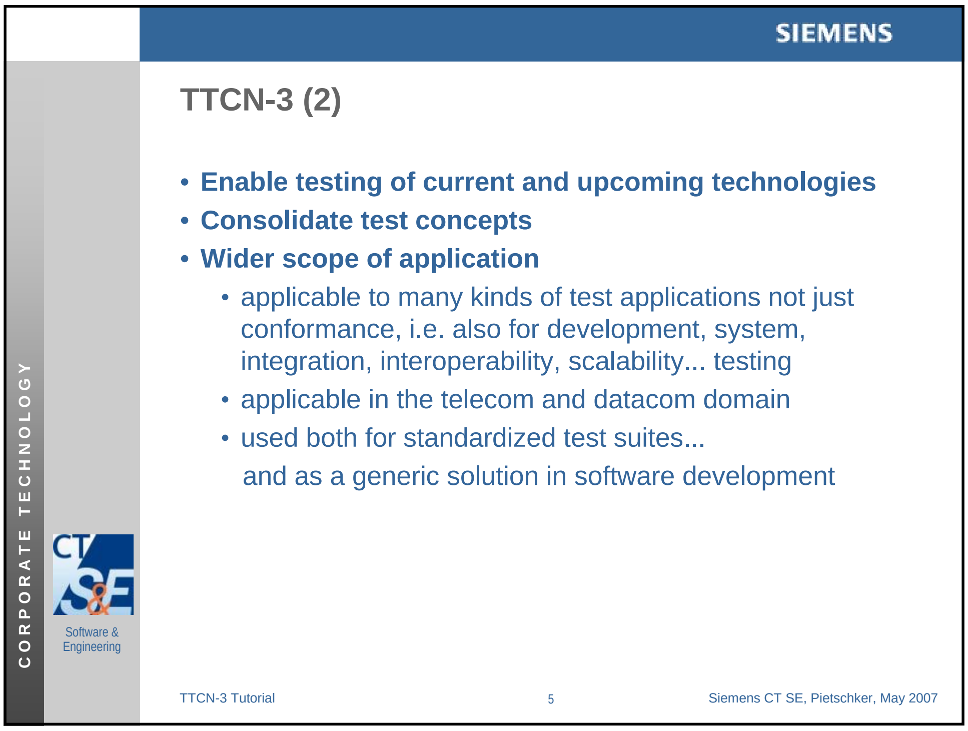
SIEMENS

CORPORATE TECHNOLOGY

CT S&E Software & Engineering

TTCN-3 (1)

- **The Testing and Test Control Notation**
- **The standardized 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, SIP);
  - CORBA based systems;
  - Java, XML, ...



SIEMENS

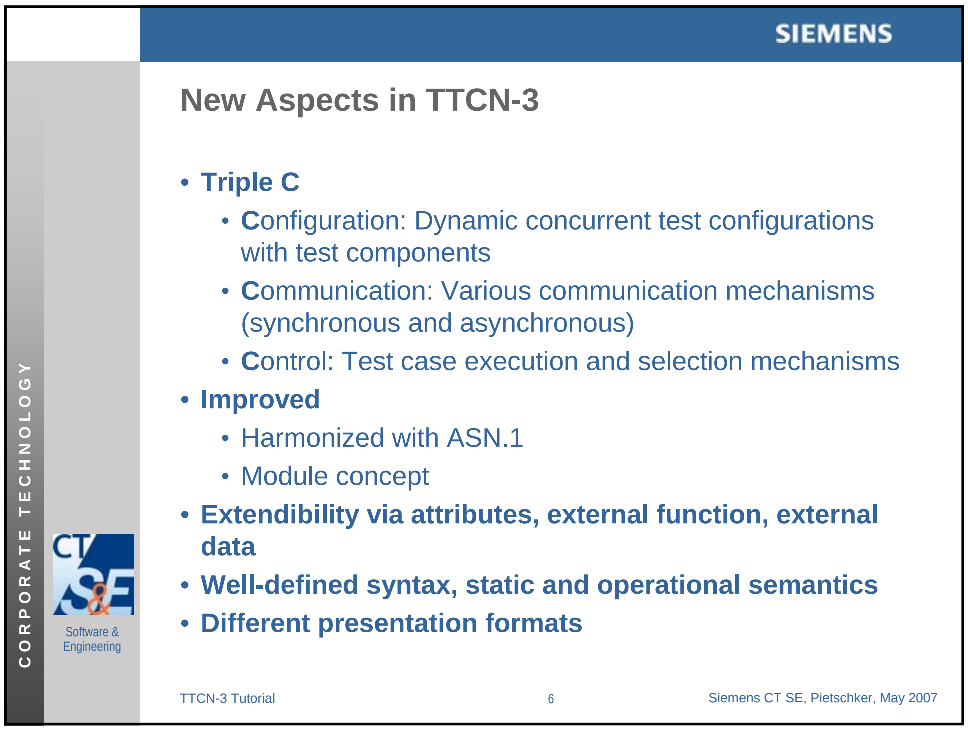
CORPORATE TECHNOLOGY

CT SE Software & Engineering

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



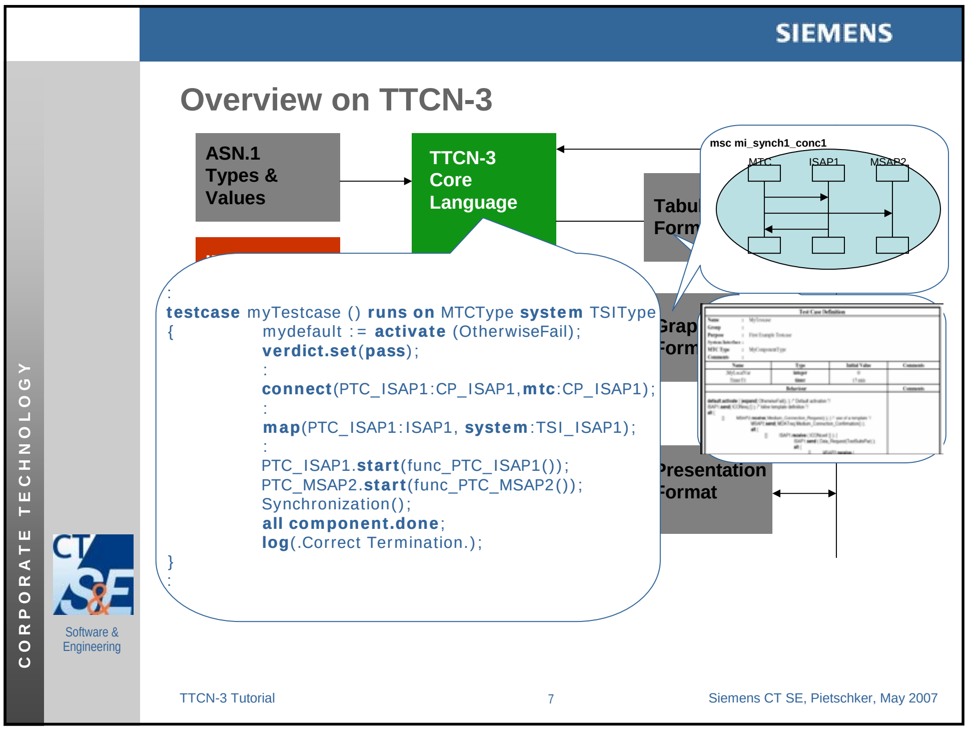
SIEMENS

CORPORATE TECHNOLOGY

CT SE Software & Engineering

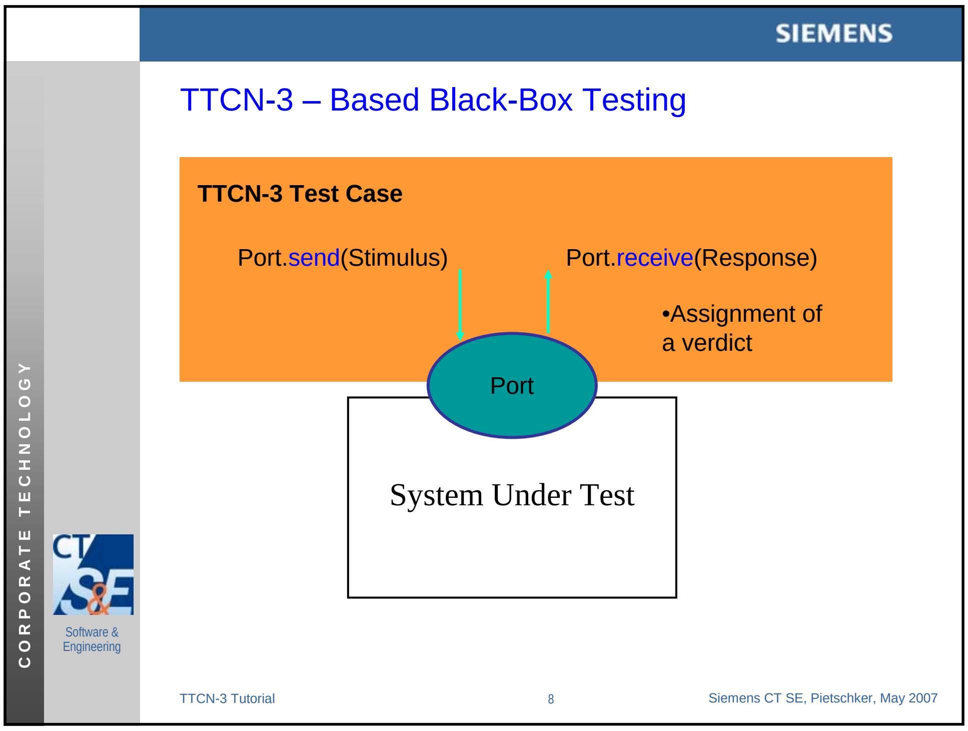
New Aspects in TTCN-3

- **Triple C**
  - **C**onfiguration: Dynamic concurrent test configurations with test components
  - **C**ommunication: Various communication mechanisms (synchronous and asynchronous)
  - **C**ontrol: 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**

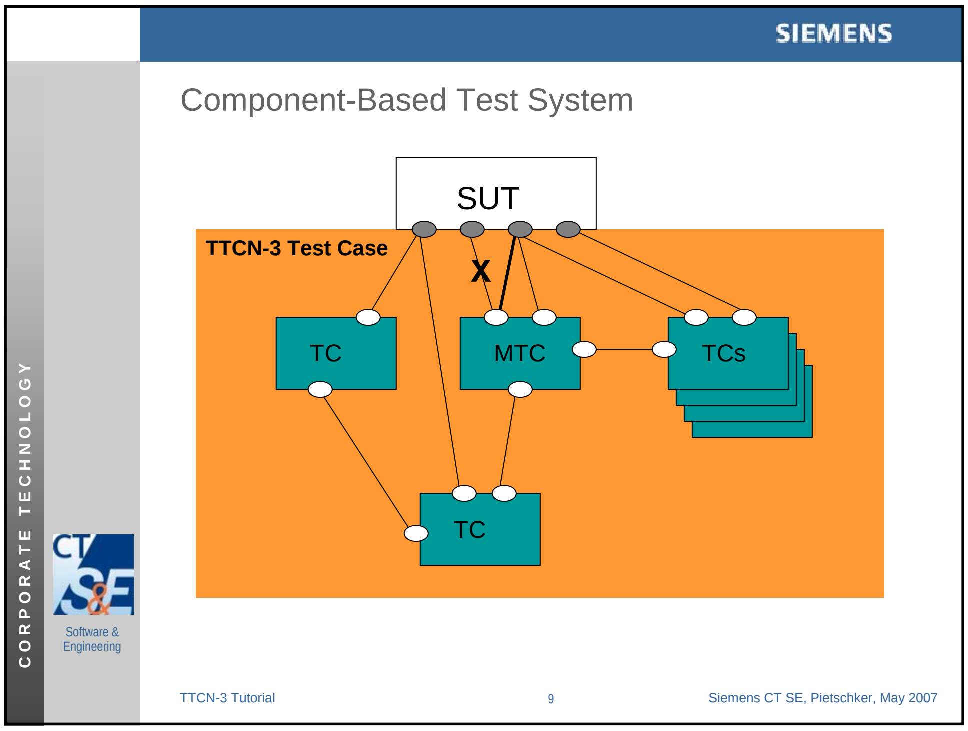


Overview on TTCN-3

```
:
testcase myTestcase () runs on MTCType system TSIType
{          mydefault := activate (OtherwiseFail);
           verdict.set(pass);
           :
           connect(PTC_ISAP1:CP_ISAP1,mtc:CP_ISAP1);
           :
           map(PTC_ISAP1:ISAP1, system:TSI_ISAP1);
           :
           PTC_ISAP1.start(func_PTC_ISAP1());
           PTC_MSAP2.start(func_PTC_MSAP2());
           Synchronization();
           all component.done;
           log(.Correct Termination.);
}
:
```



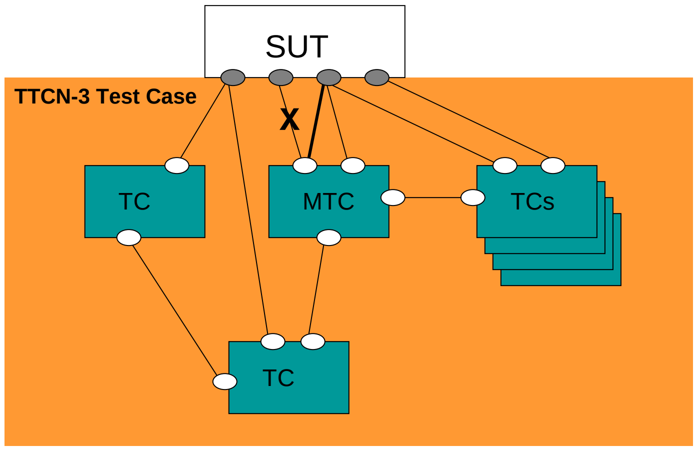
TTCN-3 – Based Black-Box Testing

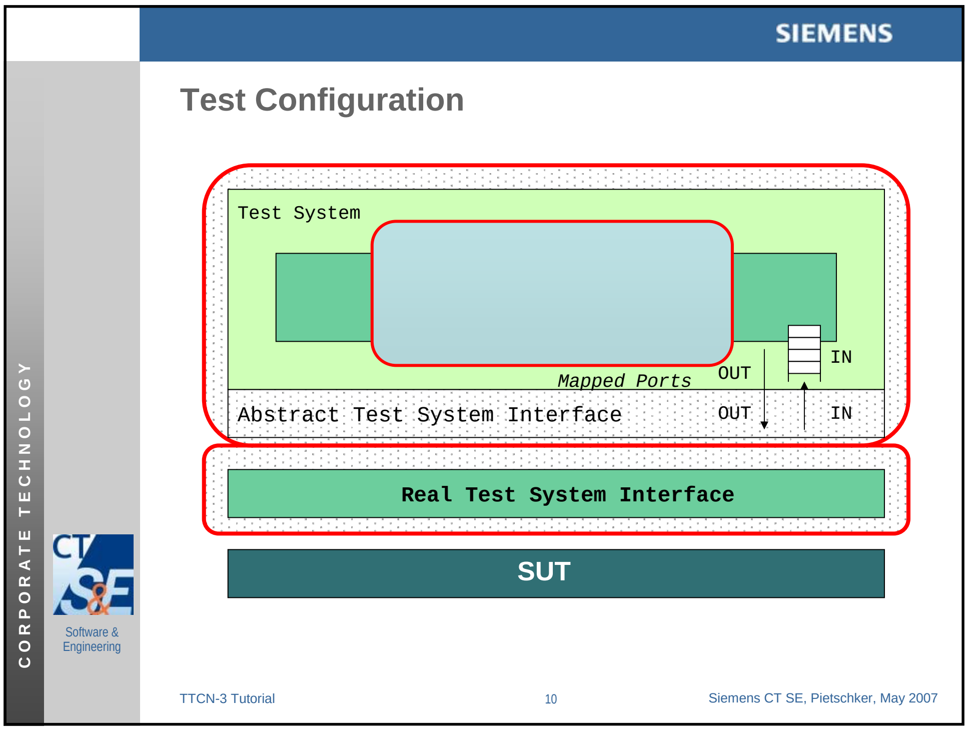


CORPORATE TECHNOLOGY

CT SE Software & Engineering

Component-Based Test System

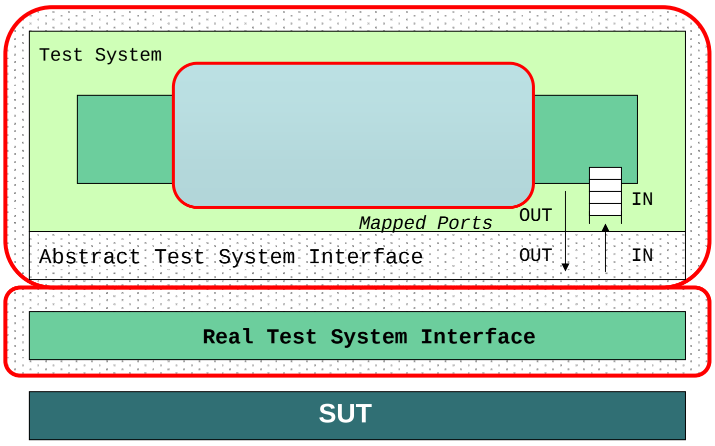


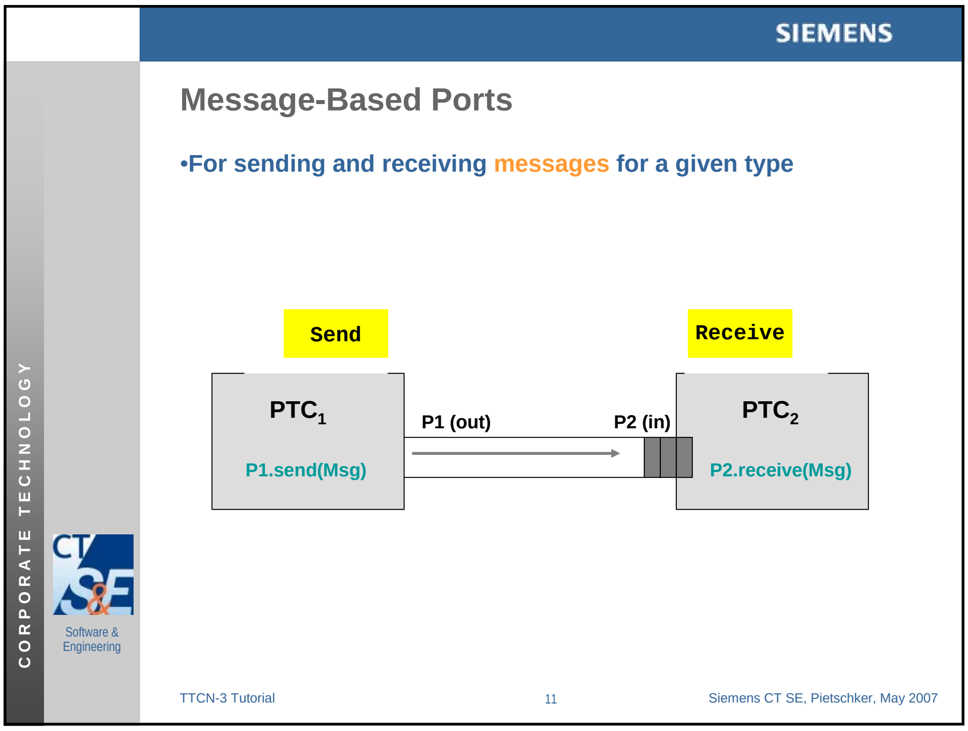


CORPORATE TECHNOLOGY

CT SE Software & Engineering

Test Configuration





SIEMENS

CORPORATE TECHNOLOGY

CT S&E Software & Engineering

Message-Based Ports

•For sending and receiving messages for a given type

Send

Receive

PTC$_1$

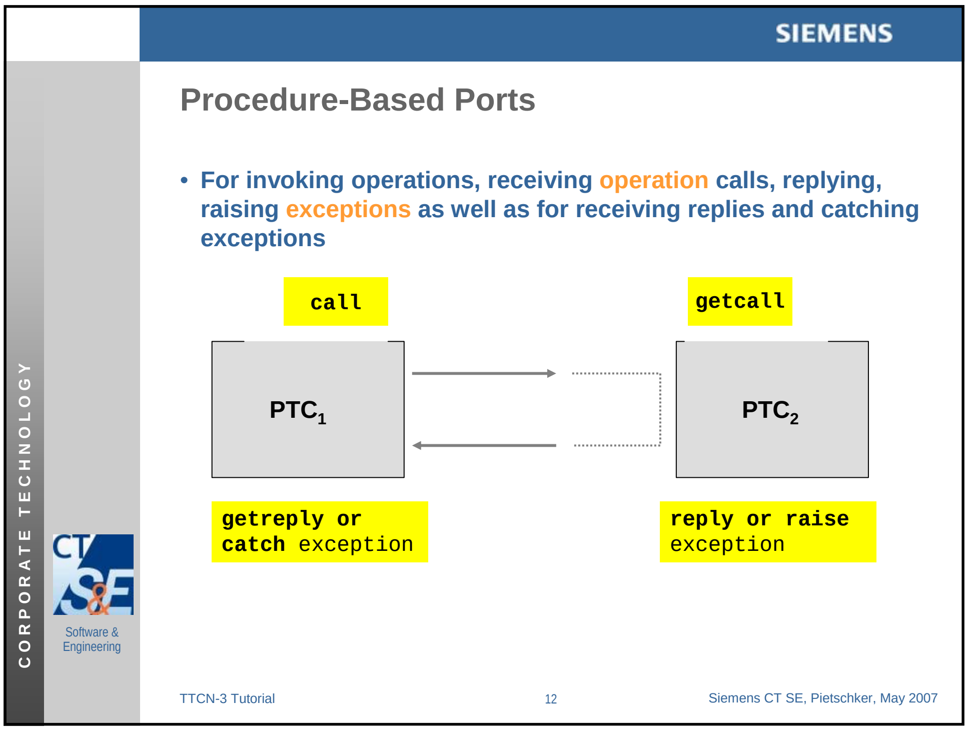
P1.send(Msg)

P1 (out)

P2 (in)

PTC$_2$

P2.receive(Msg)



SIEMENS

CORPORATE TECHNOLOGY

CT S&E Software & Engineering

Procedure-Based Ports

- For invoking operations, receiving operation calls, replying, raising exceptions as well as for receiving replies and catching exceptions

call

getcall

PTC$_1$

PTC$_2$

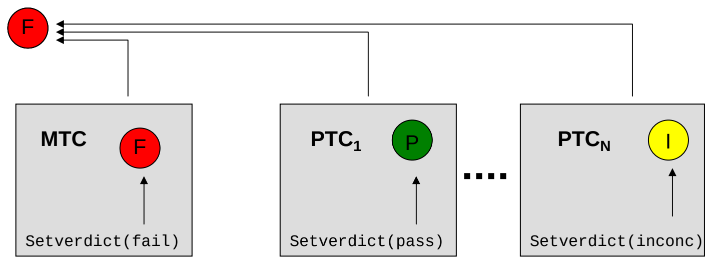
getreply or catch exception

reply or raise exception

Test Verdicts

- **Test verdicts: none < pass < inconc < fail < error**
- **Each test component has its own local verdict, which can be set (setverdict) and read (getverdict).**
- **A test case returns a global verdict**

Verdict returned by the test
case when it terminates



CORPORATE TECHNOLOGY

Software & Engineering

Basic Elements of TTCN-3

- 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 compileable programs
  - Share (type and data) information
- Test verdicts

CORPORATE TECHNOLOGY

Software & Engineering

SIEMENS

CORPORATE TECHNOLOGY

Software & Engineering

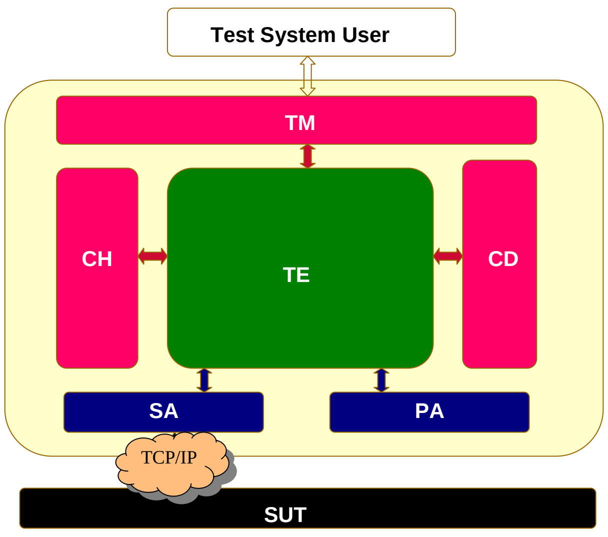
Demonstration

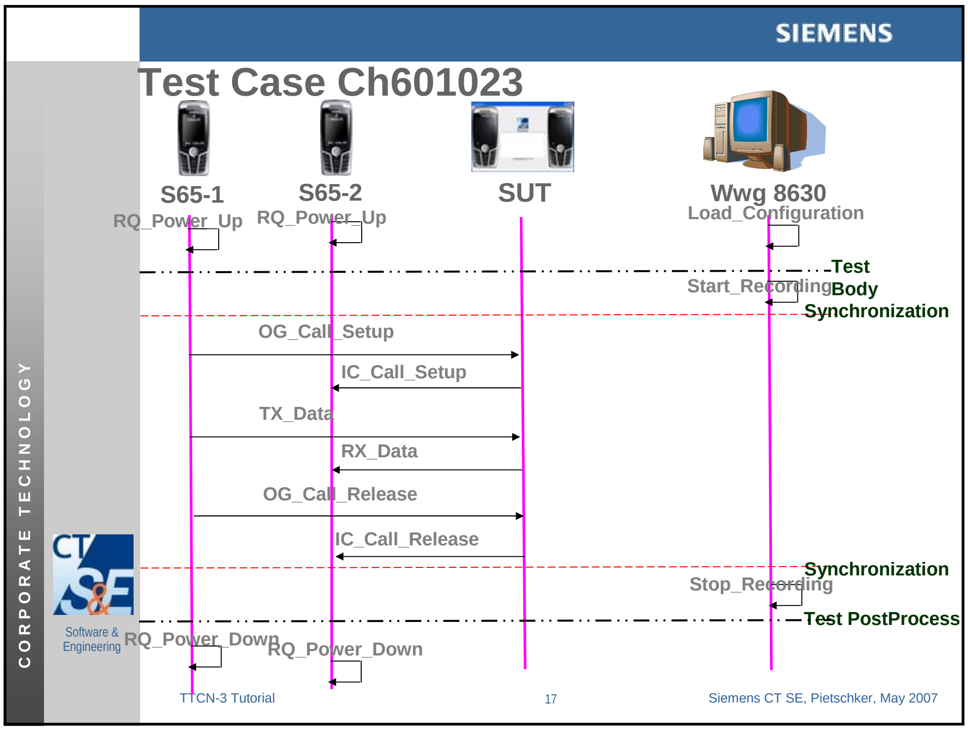
SIEMENS

CORPORATE TECHNOLOGY

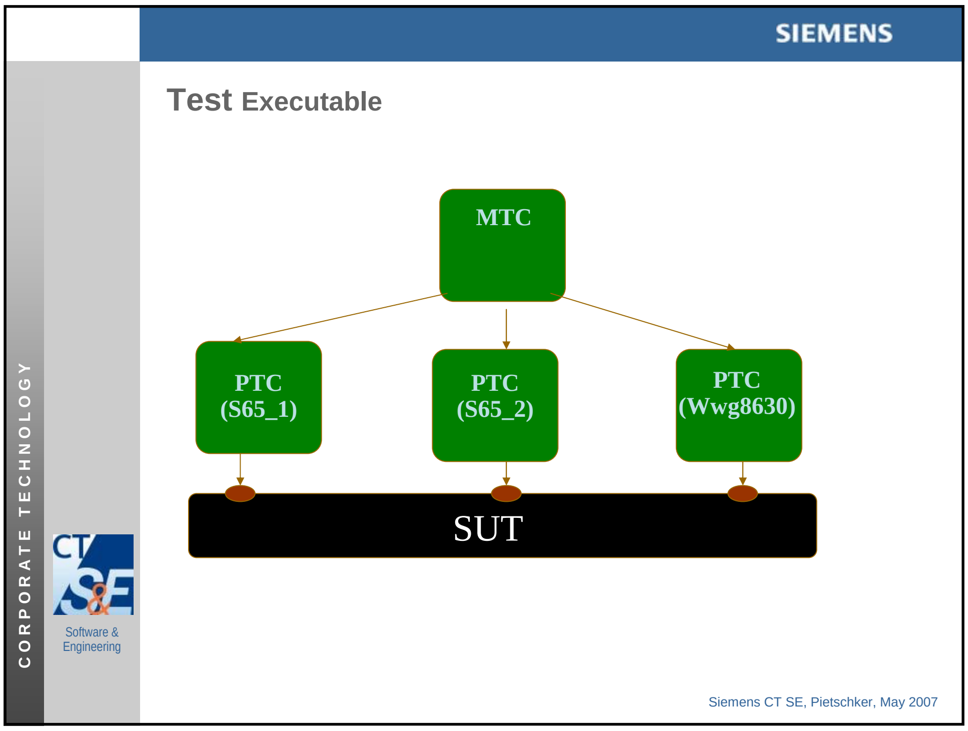
Software & Engineering

Test System





Test Case Ch601023



Test Executable

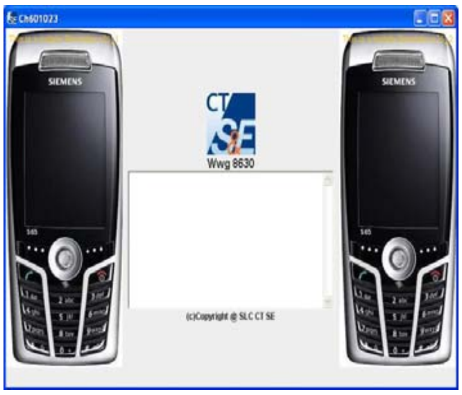
SIEMENS

CORPORATE TECHNOLOGY

CT S&E Software & Engineering

System Under Test

- GUI supported System to simulate two S65 mobile phones' communication working with WWG8630
- Work as server side over TCP/IP (be able to handle multiple clients)
- Accept the client side message and send the acknowledgement back to client



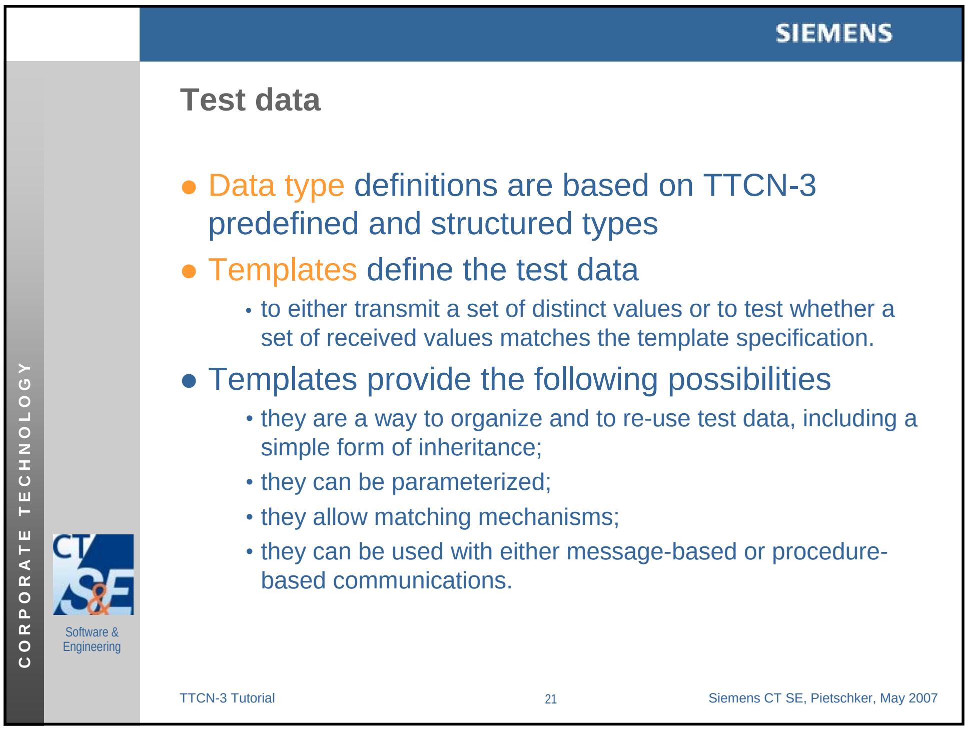
SIEMENS

CORPORATE TECHNOLOGY

CT S&E Software & Engineering

System Adapter

- Override the Test Runtime Interface (TRI) from ETSI and different tools vendor, such as TriMap, TriCall, etc.
- Work as multi clients communicate with SUT over TCP/IP
- TriMap – Create the corresponding socket for the Test Component
- TriSend – Send the correct message (object) to the SUT over the corresponding socket and get the acknowledgement back from the SUT over the same socket
- TriUnmap – Close the corresponding socket


SIEMENS

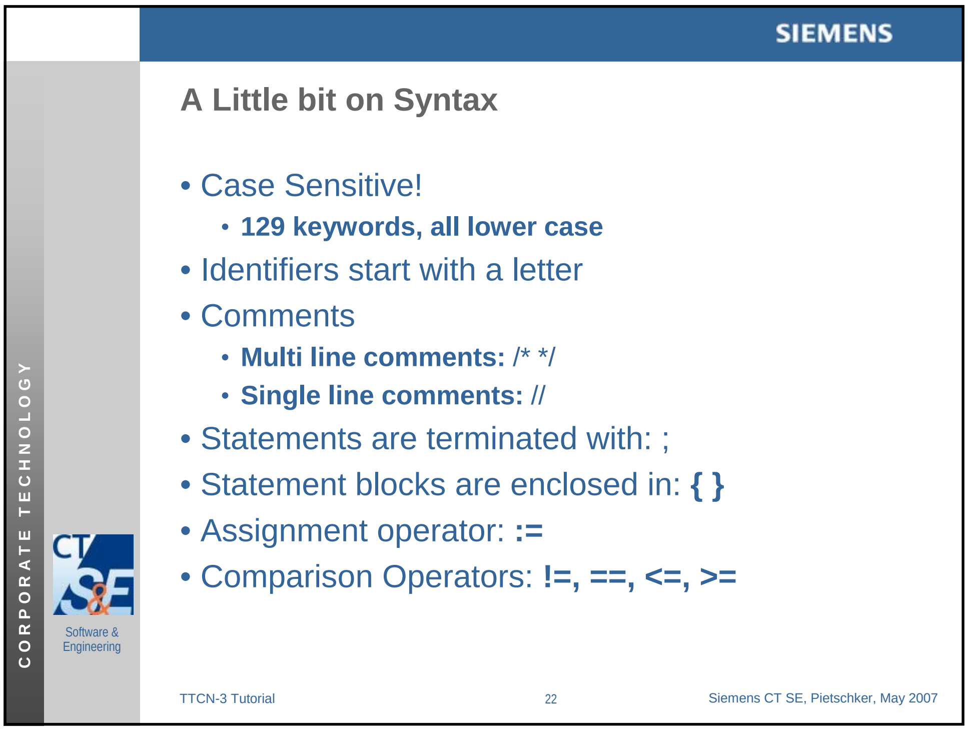
CORPORATE TECHNOLOGY

CT SE Software & Engineering

Test data

- Data type definitions are based on TTCN-3 predefined and structured types
- Templates define the test data
  - to either transmit a set of distinct values or to test whether a set of received values matches the template specification.
- Templates provide the following possibilities
  - they are a way to organize and to re-use test data, including a simple form of inheritance;
  - they can be parameterized;
  - they allow matching mechanisms;
  - they can be used with either message-based or procedure-based communications.

TTCN-3 Tutorial 21 Siemens CT SE, Pietschker, May 2007


SIEMENS

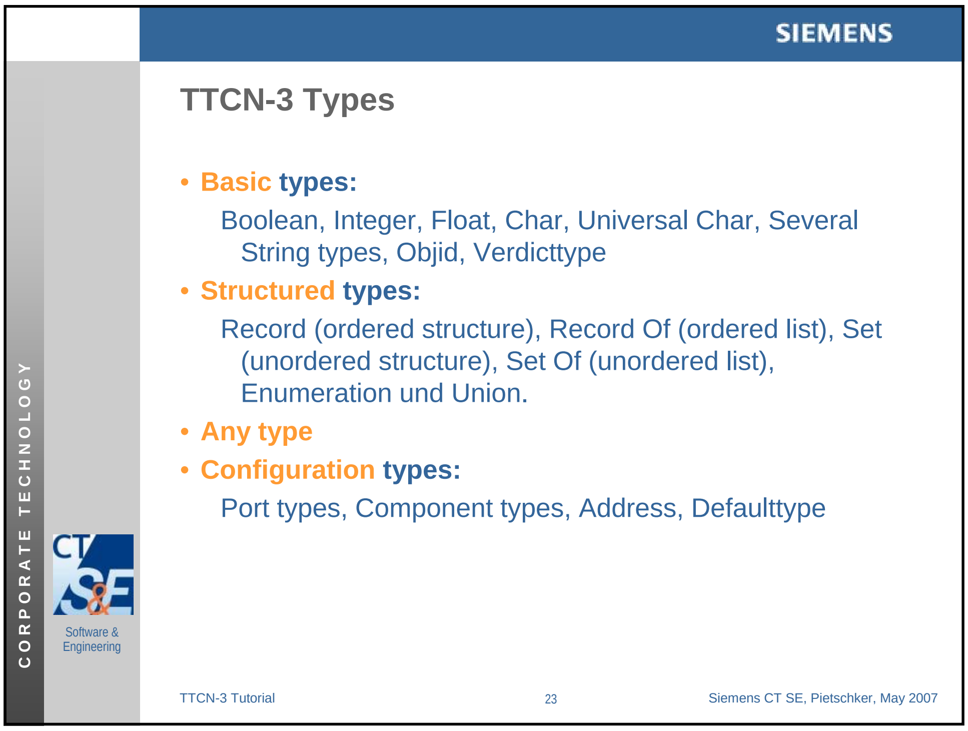
CORPORATE TECHNOLOGY

CT SE Software & Engineering

A Little bit on Syntax

- Case Sensitive!
  - **129 keywords, all lower case**
- Identifiers start with a letter
- Comments
  - **Multi line comments:** /* */
  - **Single line comments:** //
- Statements are terminated with: ;
- Statement blocks are enclosed in: **{ }**
- Assignment operator: **:=**
- Comparison Operators: **!=, ==, <=, >=**

TTCN-3 Tutorial 22 Siemens CT SE, Pietschker, May 2007


SIEMENS

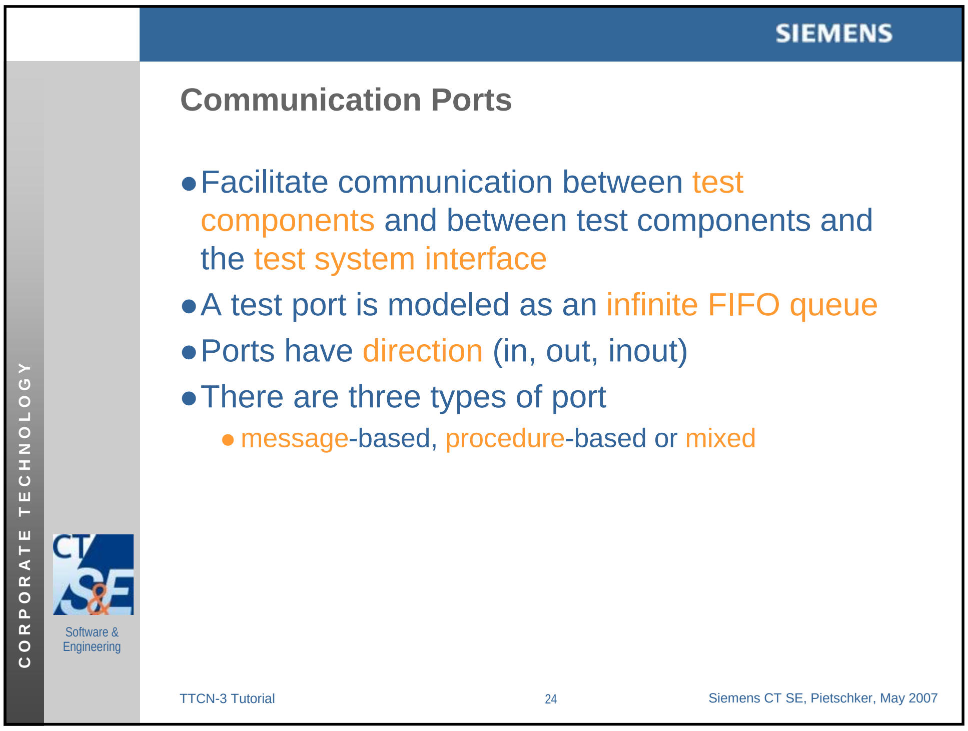
CORPORATE TECHNOLOGY

CT SE Software & Engineering

TTCN-3 Types

- **Basic types:**
  Boolean, Integer, Float, Char, Universal Char, Several String types, Objid, Verdicttype
- **Structured types:**
  Record (ordered structure), Record Of (ordered list), Set (unordered structure), Set Of (unordered list), Enumeration und Union.
- **Any type**
- **Configuration types:**
  Port types, Component types, Address, Defaulttype

TTCN-3 Tutorial 23 Siemens CT SE, Pietschker, May 2007


SIEMENS

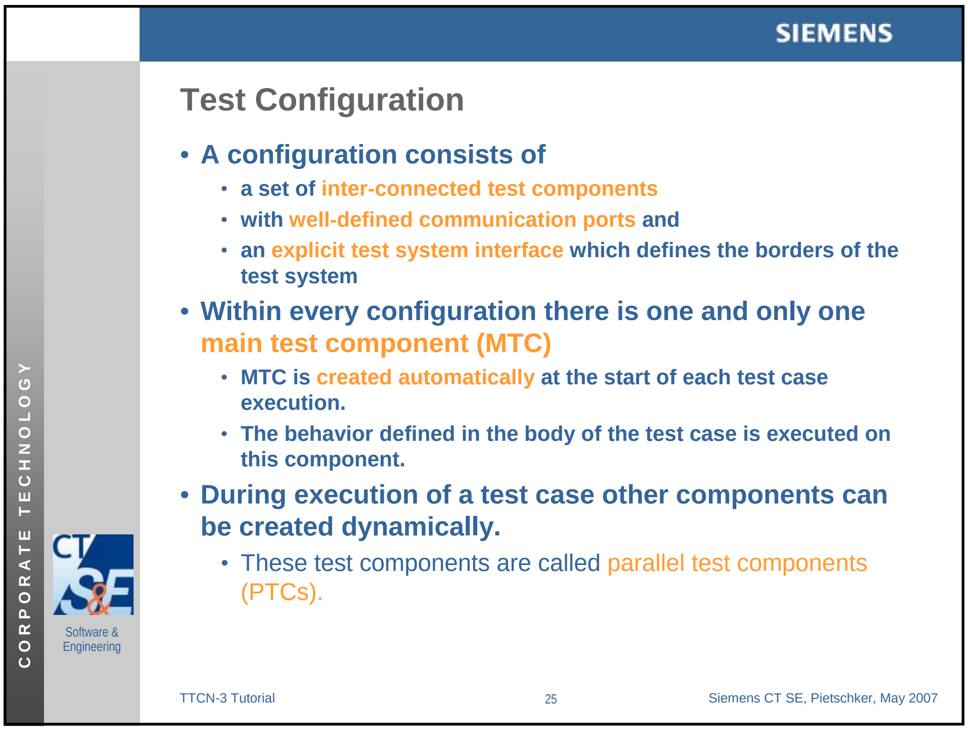
CORPORATE TECHNOLOGY

CT SE Software & Engineering

Communication Ports

- Facilitate communication between test components and between test components and the test system interface
- A test port is modeled as an infinite FIFO queue
- Ports have direction (in, out, inout)
- There are three types of port
  - message-based, procedure-based or mixed

TTCN-3 Tutorial 24 Siemens CT SE, Pietschker, May 2007


SIEMENS

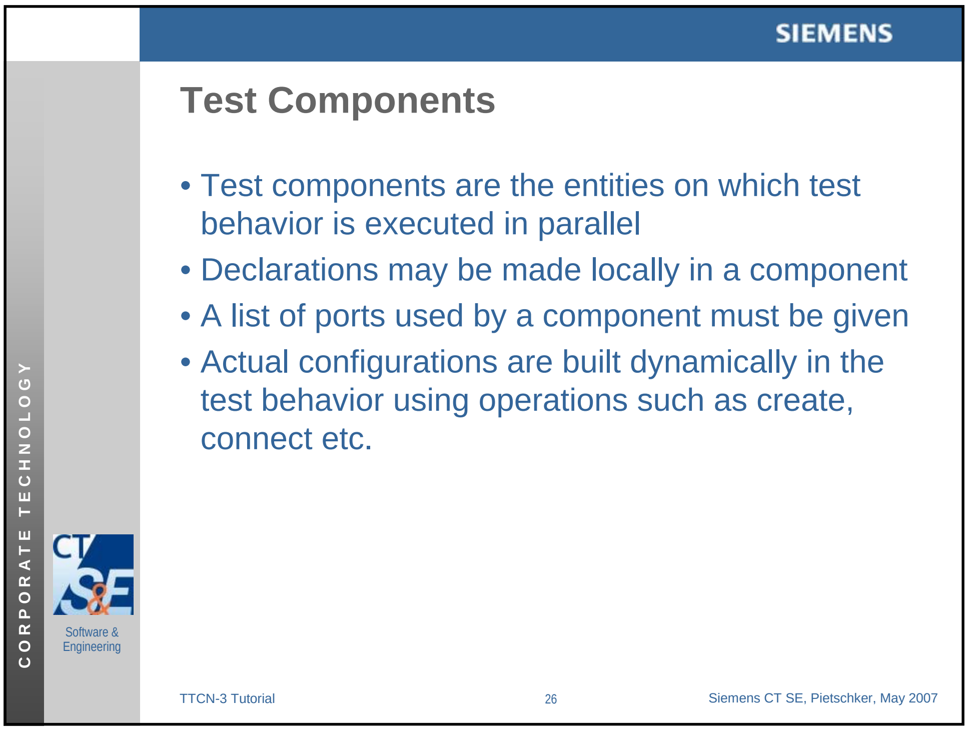
CORPORATE TECHNOLOGY

Software & Engineering

Test Configuration

- A configuration consists of
  - a set of inter-connected test components
  - with well-defined communication ports and
  - an explicit test system interface which defines the borders of the test system
- Within every configuration there is one and only one main test component (MTC)
  - MTC is created automatically at the start of each test case execution.
  - The behavior defined in the body of the test case is executed on this component.
- During execution of a test case other components can be created dynamically.
  - These test components are called parallel test components (PTCs).

TTCN-3 Tutorial 25 Siemens CT SE, Pietschker, May 2007


SIEMENS

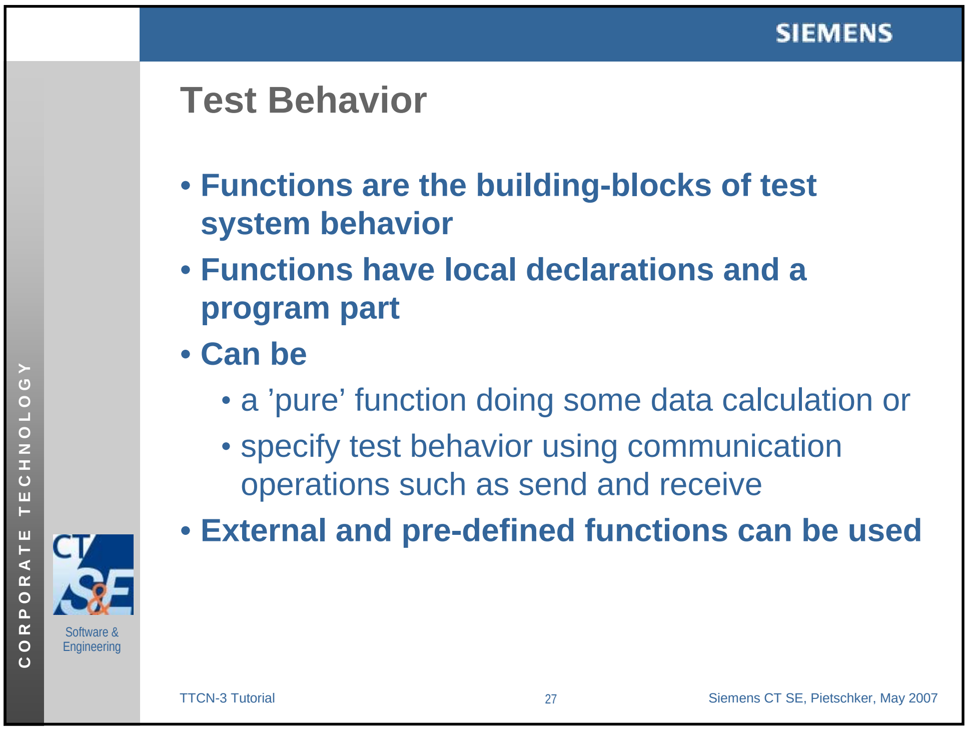
CORPORATE TECHNOLOGY

Software & Engineering

Test Components

- Test components are the entities on which test behavior is executed in parallel
- Declarations may be made locally in a component
- A list of ports used by a component must be given
- Actual configurations are built dynamically in the test behavior using operations such as create, connect etc.

TTCN-3 Tutorial 26 Siemens CT SE, Pietschker, May 2007



SIEMENS

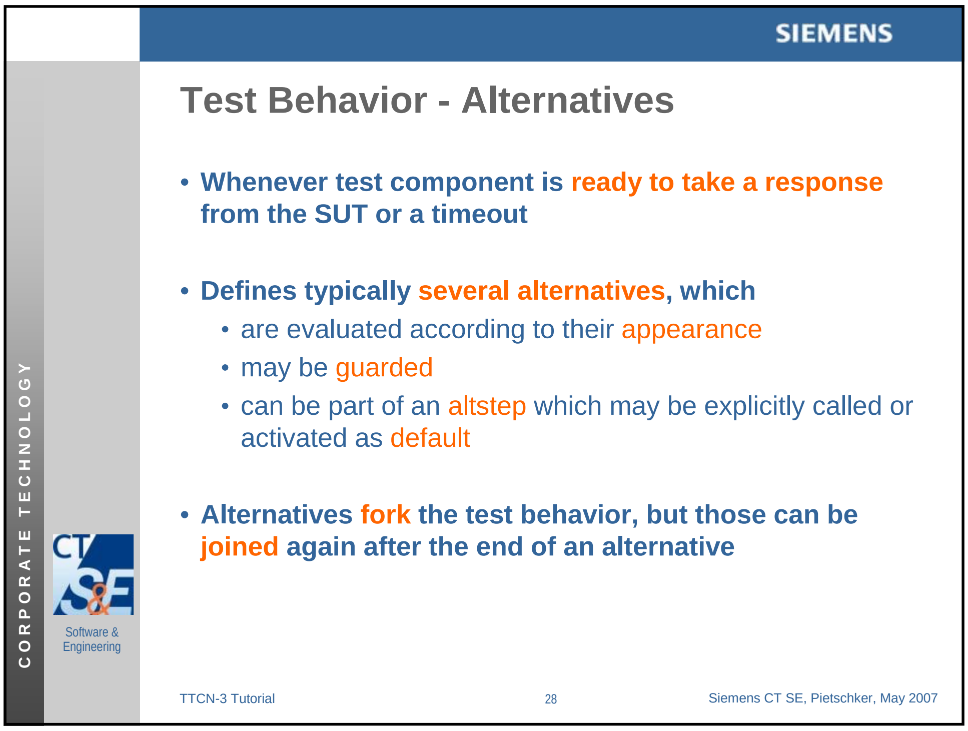
CORPORATE TECHNOLOGY

CT SE Software & Engineering

Test Behavior

- **Functions are the building-blocks of test system behavior**
- **Functions have local declarations and a program part**
- **Can be**
  - a 'pure' function doing some data calculation or
  - specify test behavior using communication operations such as send and receive
- **External and pre-defined functions can be used**

TTCN-3 Tutorial 27 Siemens CT SE, Pietschker, May 2007



SIEMENS

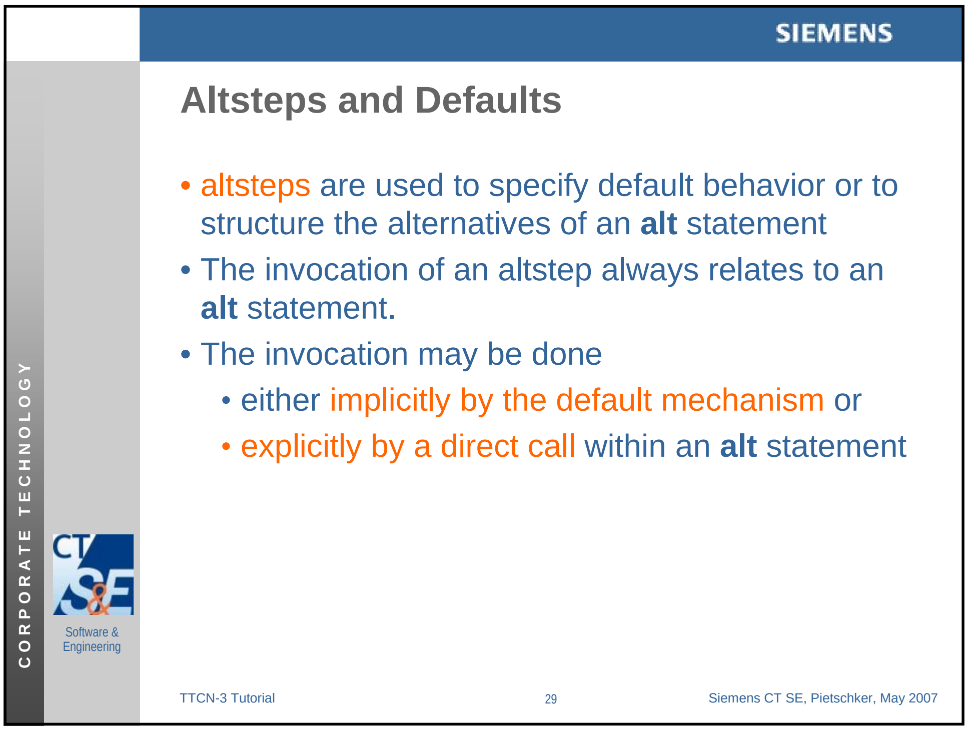
CORPORATE TECHNOLOGY

CT SE Software & Engineering

Test Behavior - Alternatives

- **Whenever test component is ready to take a response from the SUT or a timeout**
- **Defines typically several alternatives, which**
  - are evaluated according to their appearance
  - may be guarded
  - can be part of an altstep which may be explicitly called or activated as default
- **Alternatives fork the test behavior, but those can be joined again after the end of an alternative**

TTCN-3 Tutorial 28 Siemens CT SE, Pietschker, May 2007



SIEMENS

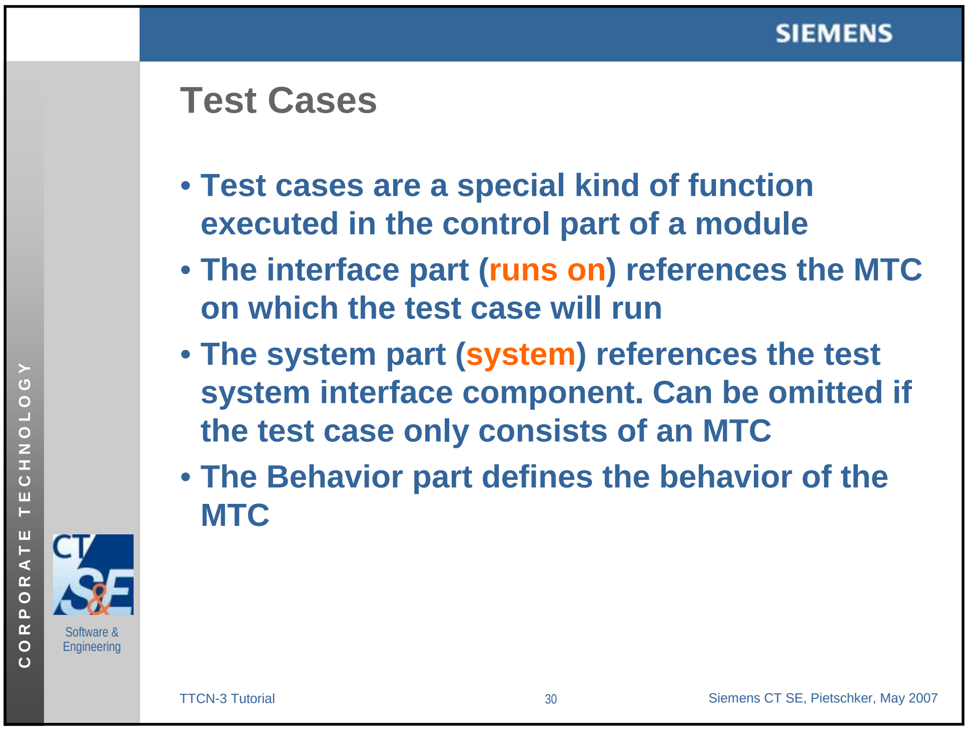
CORPORATE TECHNOLOGY

CT SE Software & Engineering

Altsteps and Defaults

- altsteps are used to specify default behavior or to structure the alternatives of an **alt** statement
- The invocation of an altstep always relates to an **alt** statement.
- The invocation may be done
  - either implicitly by the default mechanism or
  - explicitly by a direct call within an **alt** statement

TTCN-3 Tutorial 29 Siemens CT SE, Pietschker, May 2007



SIEMENS

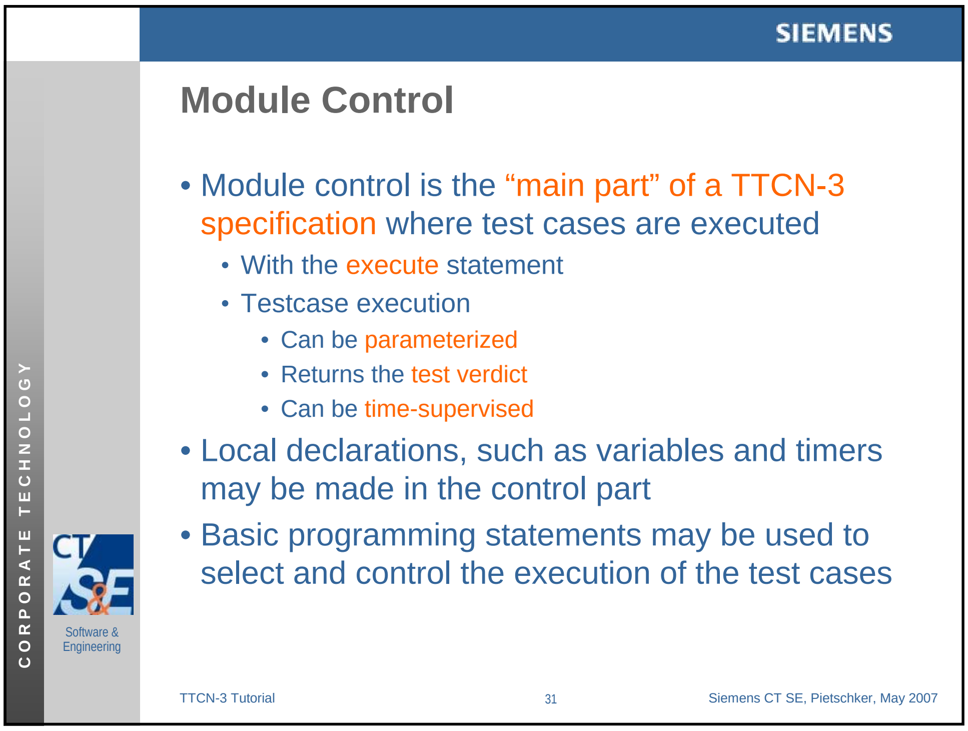
CORPORATE TECHNOLOGY

CT SE Software & Engineering

Test Cases

- **Test cases are a special kind of function executed in the control part of a module**
- **The interface part (runs on) references the MTC on which the test case will run**
- **The system part (system) references the test system interface component. Can be omitted if the test case only consists of an MTC**
- **The Behavior part defines the behavior of the MTC**

TTCN-3 Tutorial 30 Siemens CT SE, Pietschker, May 2007



SIEMENS

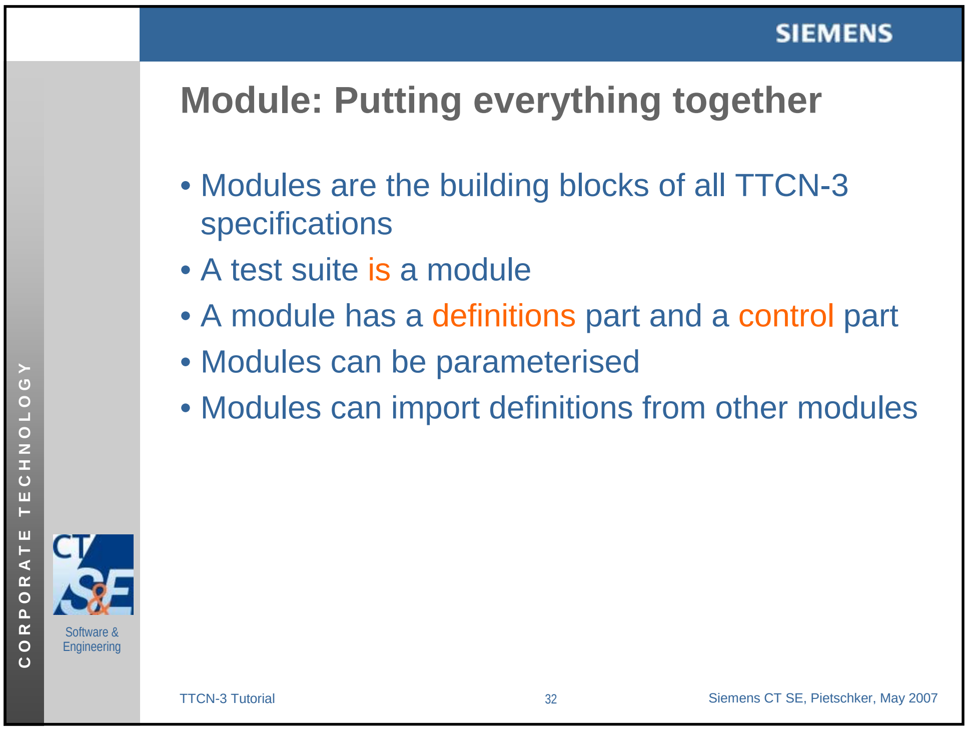
CORPORATE TECHNOLOGY

CT SE Software & Engineering

Module Control

- Module control is the “main part” of a TTCN-3 specification where test cases are executed
  - With the execute statement
  - Testcase execution
    - Can be parameterized
    - Returns the test verdict
    - Can be time-supervised
- Local declarations, such as variables and timers may be made in the control part
- Basic programming statements may be used to select and control the execution of the test cases

TTCN-3 Tutorial 31 Siemens CT SE, Pietschker, May 2007



SIEMENS

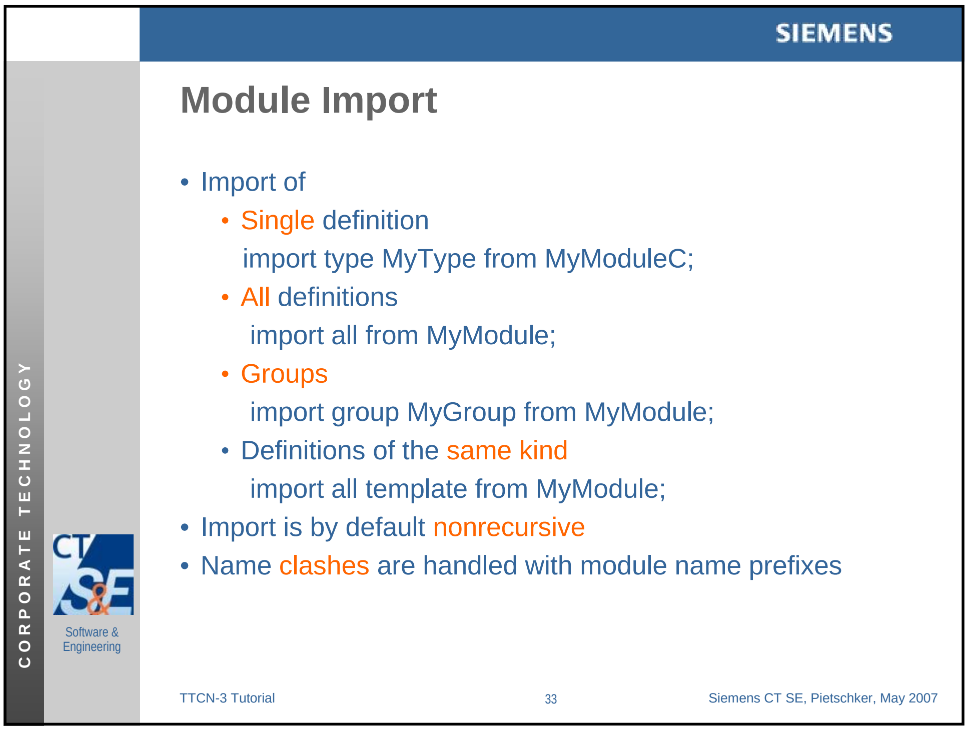
CORPORATE TECHNOLOGY

CT SE Software & Engineering

Module: Putting everything together

- Modules are the building blocks of all TTCN-3 specifications
- A test suite is a module
- A module has a definitions part and a control part
- Modules can be parameterised
- Modules can import definitions from other modules

TTCN-3 Tutorial 32 Siemens CT SE, Pietschker, May 2007


SIEMENS

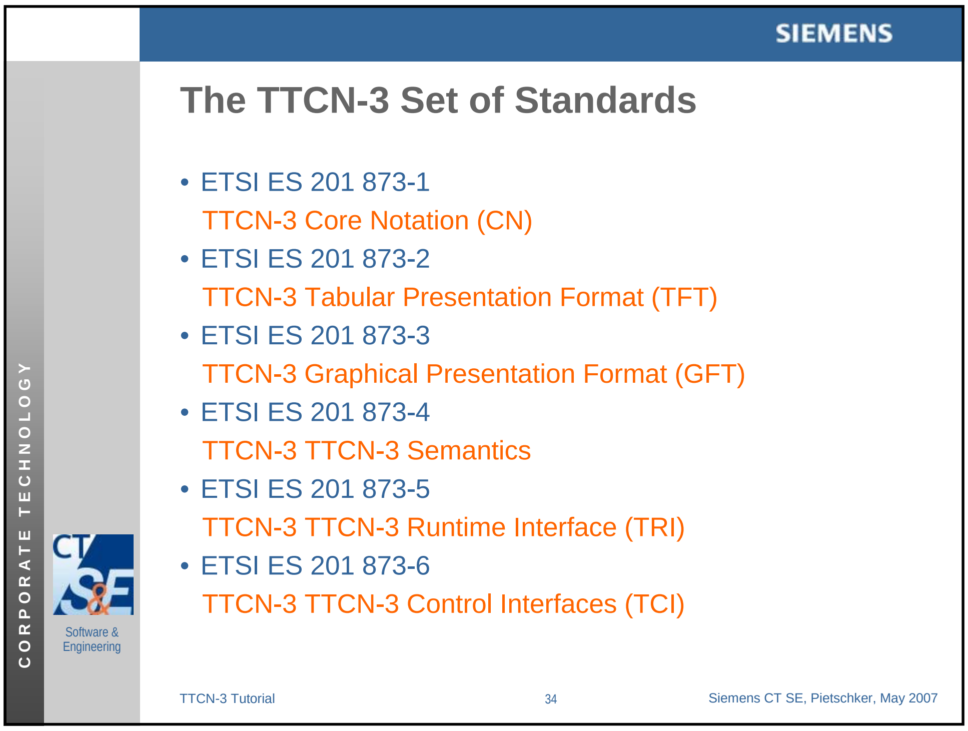
CORPORATE TECHNOLOGY

CT SE Software & Engineering

Module Import

- Import of
  - Single definition
    import type MyType from MyModuleC;
  - All definitions
    import all from MyModule;
  - Groups
    import group MyGroup from MyModule;
  - Definitions of the same kind
    import all template from MyModule;
- Import is by default nonrecursive
- Name clashes are handled with module name prefixes

TTCN-3 Tutorial 33 Siemens CT SE, Pietschker, May 2007


SIEMENS

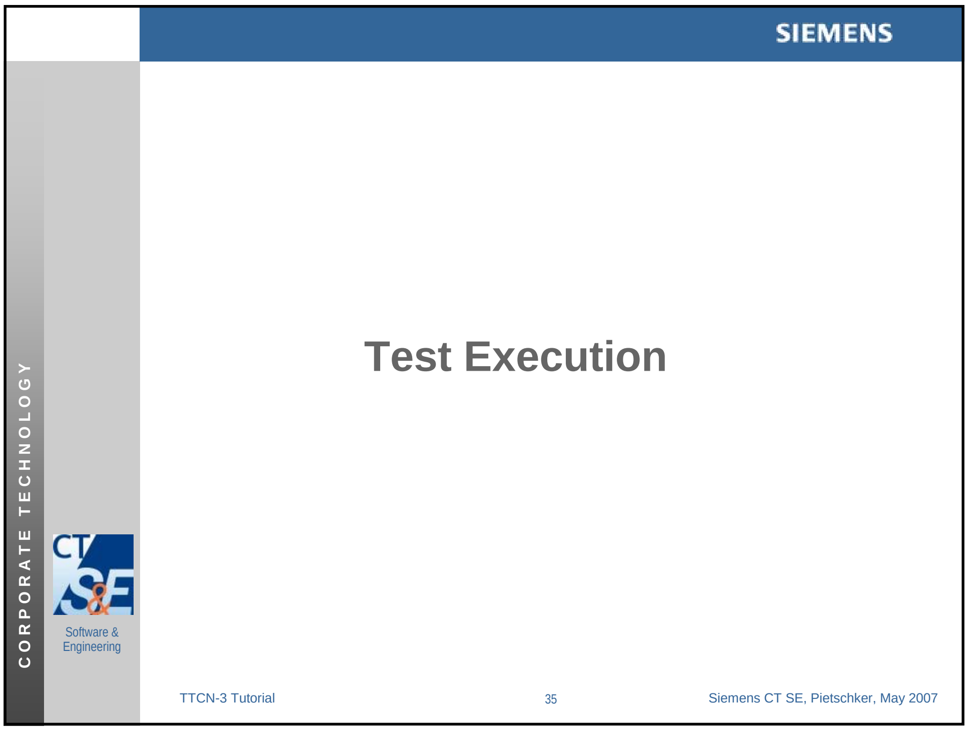
CORPORATE TECHNOLOGY

CT SE Software & Engineering

The TTCN-3 Set of Standards

- ETSI ES 201 873-1
  TTCN-3 Core Notation (CN)
- ETSI ES 201 873-2
  TTCN-3 Tabular Presentation Format (TFT)
- ETSI ES 201 873-3
  TTCN-3 Graphical Presentation Format (GFT)
- ETSI ES 201 873-4
  TTCN-3 TTCN-3 Semantics
- ETSI ES 201 873-5
  TTCN-3 TTCN-3 Runtime Interface (TRI)
- ETSI ES 201 873-6
  TTCN-3 TTCN-3 Control Interfaces (TCI)

TTCN-3 Tutorial 34 Siemens CT SE, Pietschker, May 2007

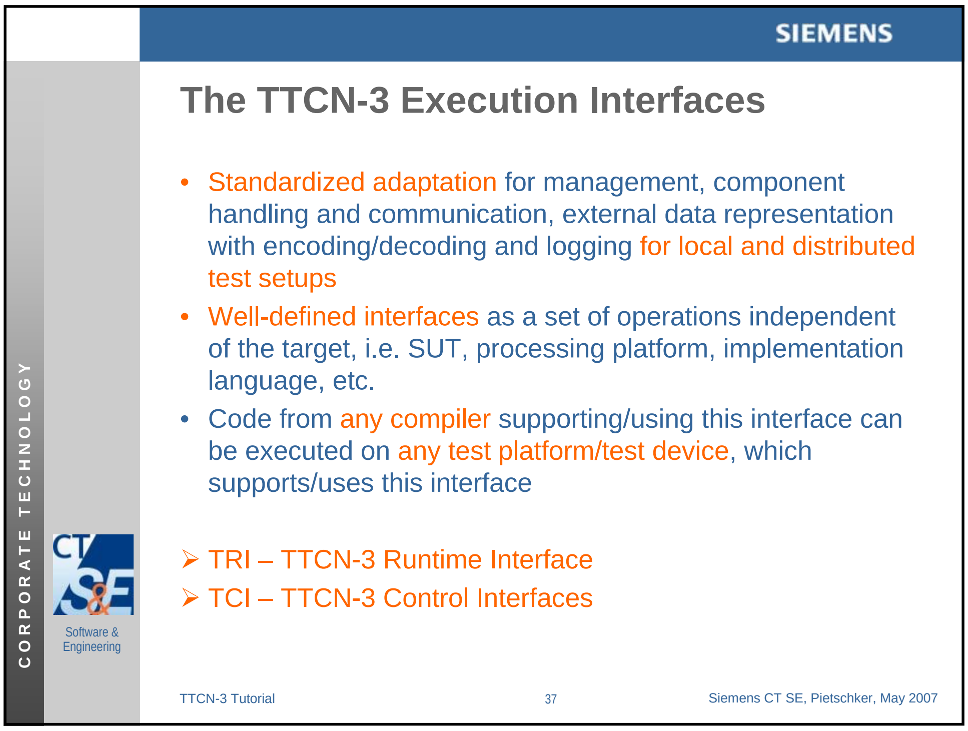


SIEMENS

CORPORATE TECHNOLOGY

CT S&E Software & Engineering

Test Execution



SIEMENS

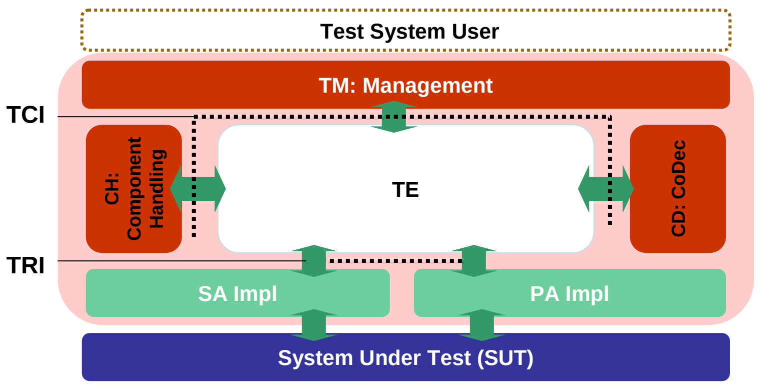
CORPORATE TECHNOLOGY

CT S&E Software & Engineering

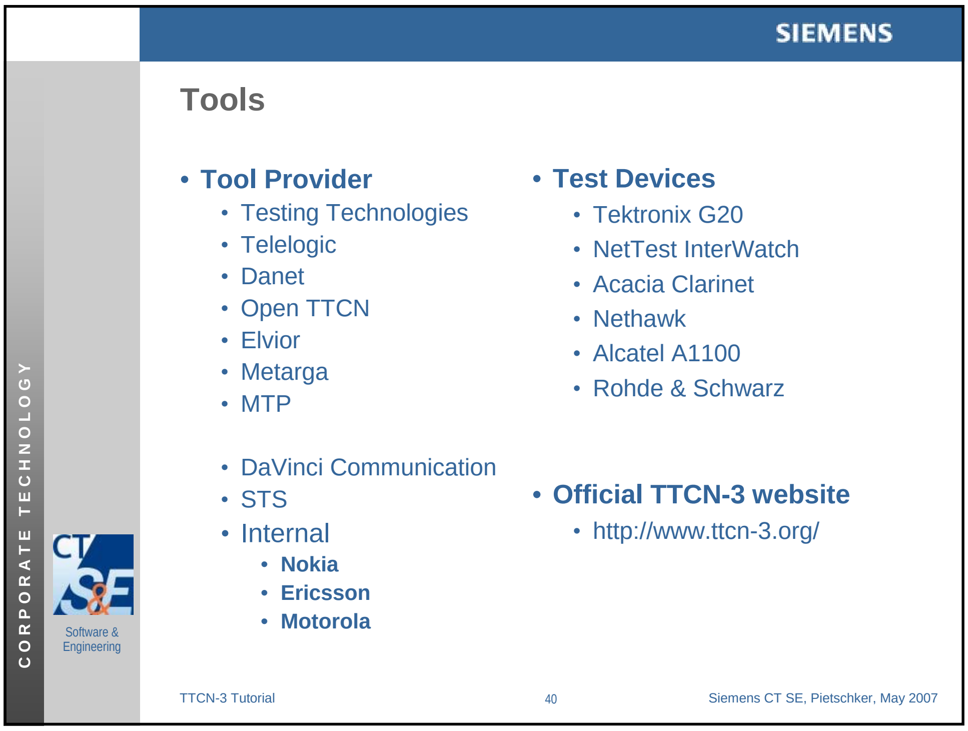
The TTCN-3 Execution Interfaces

- 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


Test System User
TM: Management
TCI
CH: Component Handling
TE
CD: CoDec
TRI
SA Impl
PA Impl
System Under Test (SUT)

TTCN-3 Tools



SIEMENS

CORPORATE TECHNOLOGY

CT S&E Software & Engineering

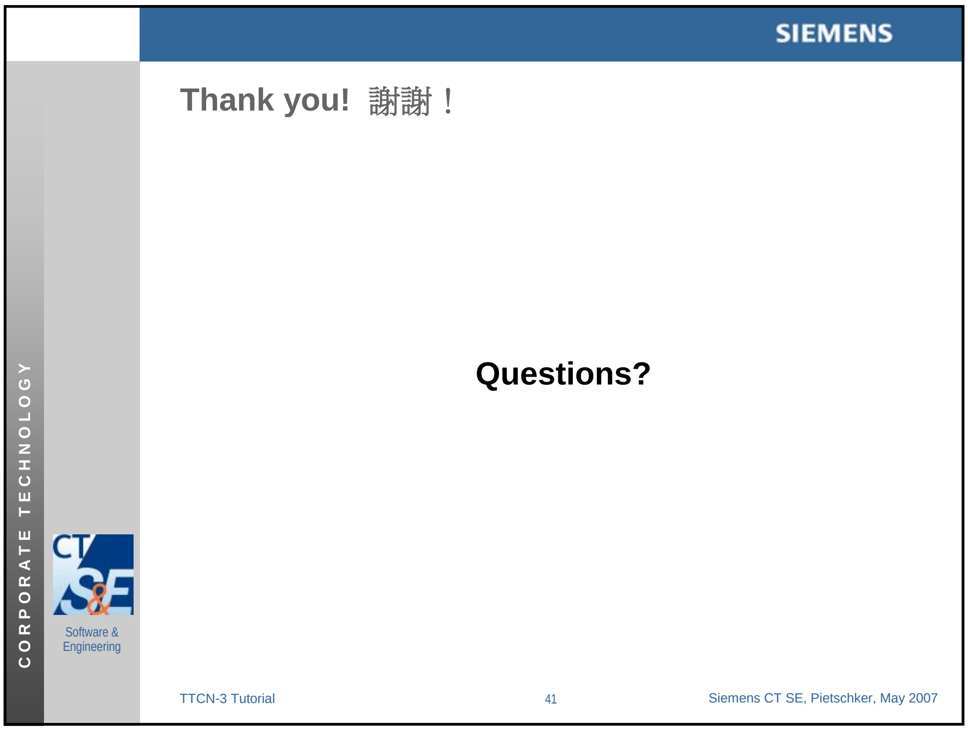
Tools

- **Tool Provider**
  - Testing Technologies
  - Telelogic
  - Danet
  - Open TTCN
  - Elvior
  - Metarga
  - MTP

  - DaVinci Communication
  - STS
  - Internal
    - **Nokia**
    - **Ericsson**
    - **Motorola**
- **Test Devices**
  - Tektronix G20
  - NetTest InterWatch
  - Acacia Clarinet
  - Nethawk
  - Alcatel A1100
  - Rohde & Schwarz

- **Official TTCN-3 website**
  - http://www.ttcn-3.org/

TTCN-3 Tutorial 40 Siemens CT SE, Pietschker, May 2007



SIEMENS

CORPORATE TECHNOLOGY

CT S&E Software & Engineering

Thank you! 謝謝！

Questions?

TTCN-3 Tutorial 41 Siemens CT SE, Pietschker, May 2007