



World Class Standards

Tutorial: Automated Interoperability Testing Using TTCN-3

Thomas Rings
University of Göttingen
Germany

Stephan Schulz
Conformiq Software OY
Finland

Luca Serazio
Telecom Italia
Italy

TTCN-3 User Conference 2010, Beijing, China

© ETSI 2010. All rights reserved

Acknowledgements

- ❑ **Specialists Taskforce (STF) 370**
 - **Developed the presented materials**
 - **Team of 12 specialists with various background from industry and academia**
 - **Work started in February 2009 and ends in July 2010**
- ❑ **Committee for Methods for Testing and Specification (MTS) at the European Telecommunications Standards Institute (ETSI)**
 - **Initiation, guidance and monitoring of the project**
- ❑ **STF 370 has been funded in part by the European Commission (EC) 2008 ICT Standardisation Action Plan SA/ETSI/2008-10**

Outline

- 1. Methodology and Framework for Automated Interoperability Testing**
- 2. Development Process for an Interoperability Test System**

Interoperability & Interoperability Testing

- ❑ “Interoperability is the ability of two systems to interoperate using the same communication protocol”
- ❑ Context of
 - Complex and distributed systems
 - Multi-vendor, multi-network, multi-service environment
 - Interoperability test events (e.g. ETSI Plugtest)
- ❑ Goals of automated interoperability testing
 - Assure interoperability
 - Assess and validate that systems follow standards
 - Assess and validate standards

Motivation for a Methodology (1/2)

“Interoperability is the **ability of two systems to interoperate** using the same **communication protocol**”

Provide users a specific End to End (E2E) functionality/service

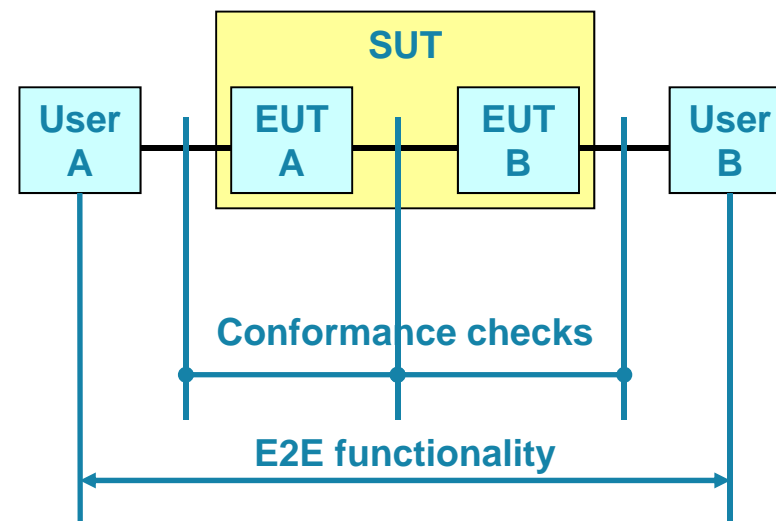
 ➤ E2E functional testing

Exchange information across standardized interfaces

 ➤ Conformance checks

Motivation for a Methodology (2/2)

E2E testing	Conformance testing
<ul style="list-style-type: none"> <input type="checkbox"/> may require complex set up <input type="checkbox"/> can verify correct service provision to end users <input type="checkbox"/> does not ensure adherence to standard specifications 	<ul style="list-style-type: none"> <input type="checkbox"/> can be expensive (lots of tests) <input type="checkbox"/> can not guarantee system interoperability especially for the application layer



EUT: Equipment Under Test
 SUT: System Under Test

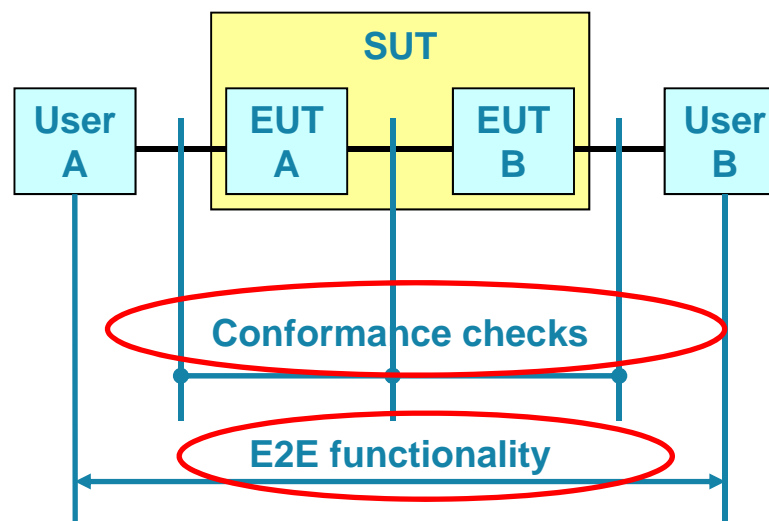
↓

Need for a methodology to enable automated interoperability testing with conformance checking

Verdicts in Interoperability Testing

- ❑ **E2E interoperability verdict**
 - **Pass:** all EUTs interoperate as required by the test
 - **Fail:** at least one EUT does not interoperate as required by the test

- ❑ **Conformance verdict**
 - **Pass:** Fulfils the conformance test objective
 - **Fail:** non-conformance to a normative requirement
 - **Inconclusive:** neither a pass nor a fail verdict can be given



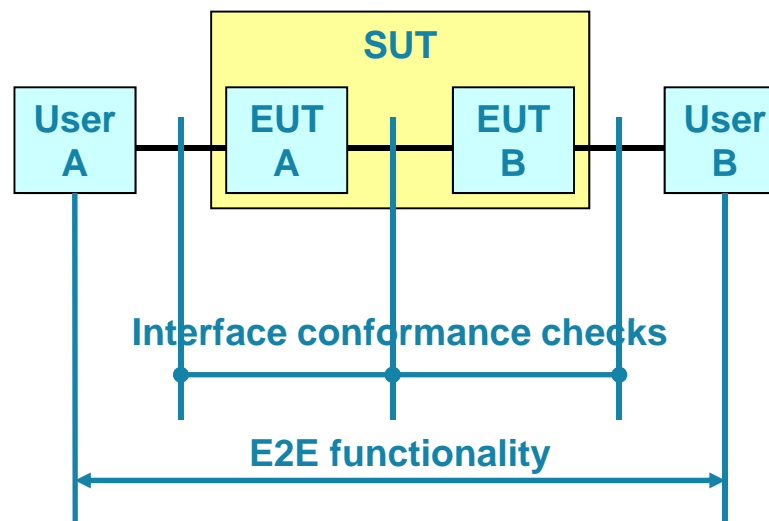
Automation of Interoperability Testing (1/2)

□ Testing tasks that can be automated

- Configuration of EUTs
- Monitoring of relevant standardized interfaces
- Emulation of equipment
- Test execution

□ Degree of possible automation

- Ideally all the tasks mentioned above should be automated
- Usually automation of a subset



EUT: Equipment Under Test

SUT: System Under Test

Automation of Interoperability Testing (2/2)

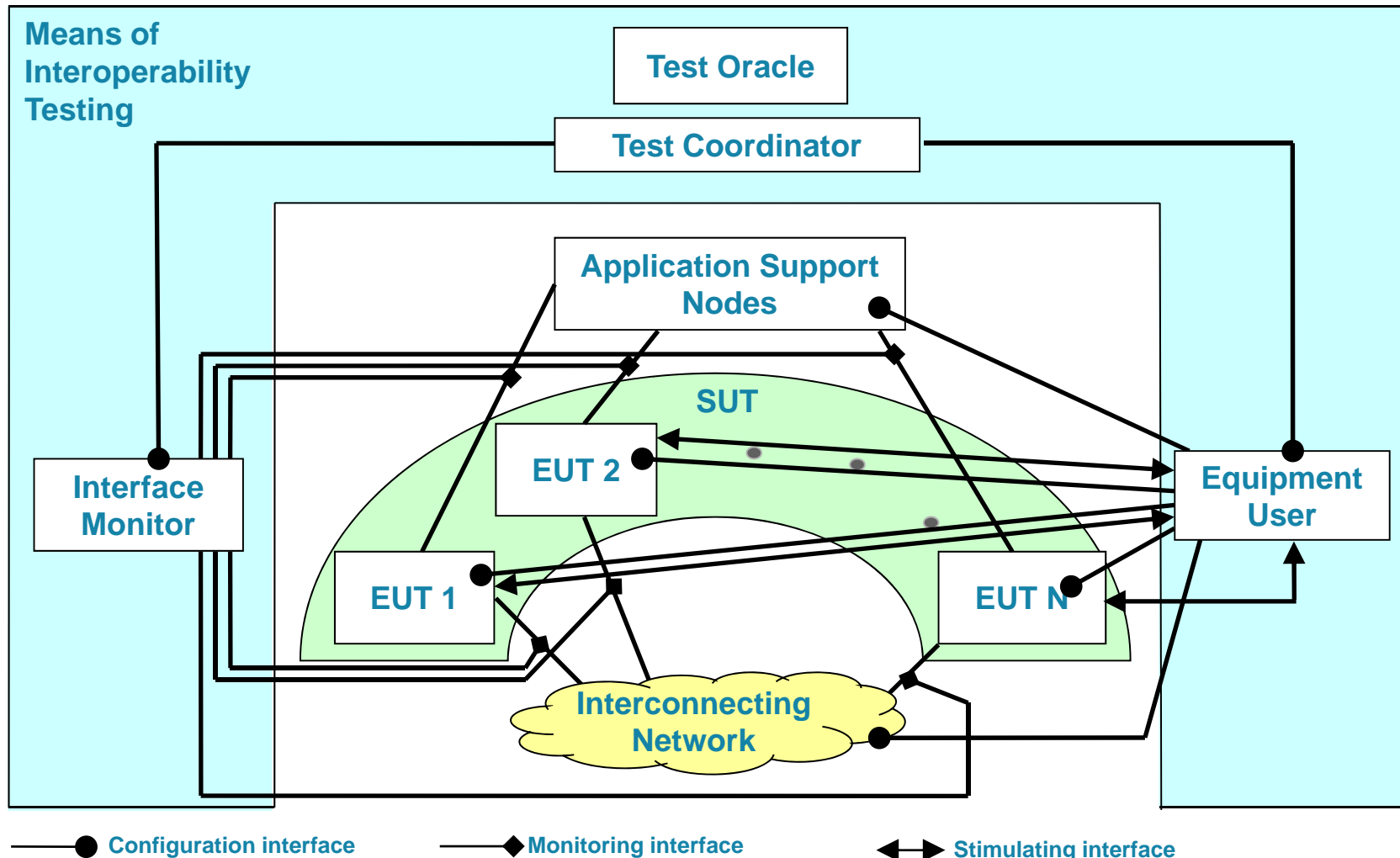
Possible limitations

- Accessibility of SUT interfaces
- Complexity or cost of implementing interface access and checks
- Unstable specifications

Benefits

- Reduces time needed for test execution and evaluation
- Avoids repetitive manual activity
- Saves costs related to human experts and test bed occupation
- Reduces time to market

A Framework for Automated Interoperability Testing

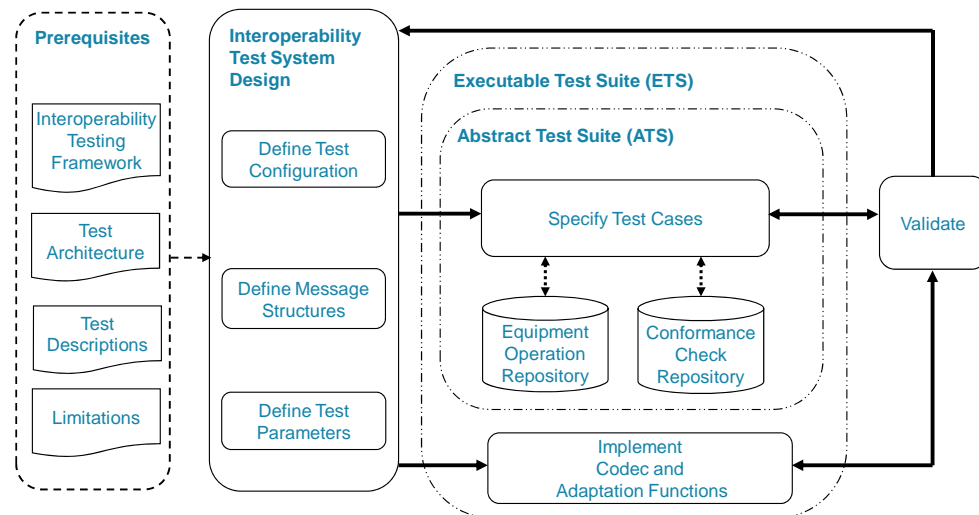


Outline

1. Methodology and Framework for Automated Interoperability Testing
2. **Development Process for an Interoperability Test System**

Development Process for an Interoperability Test System

- ❑ Language independent
- ❑ In this tutorial, explained and exemplified with specific technologies
 - IP Media Subsystem (IMS)
 - Testing and Test Control Notation Version 3 (TTCN-3)

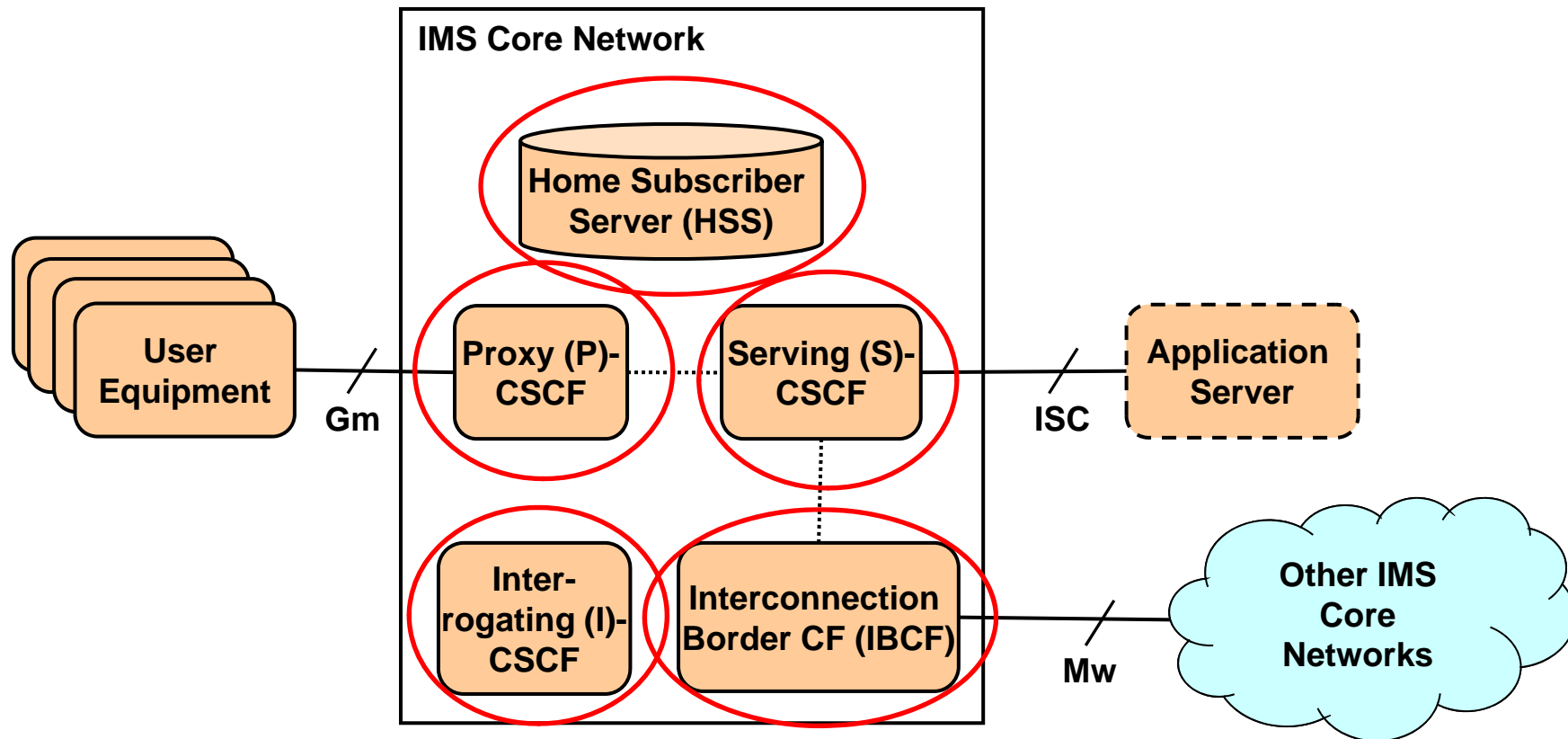


Introduction to IMS

- ❑ **IMS = IP Multimedia Subsystem**
 - 3GPP standard
 - One of the key enablers of next generation networks
- ❑ **Access independent platform**
- ❑ **A peer-to-peer architecture**
 - Can be split into user, control and service layer
 - Signalling mainly based on the Session Initiation Protocol (SIP)
- ❑ **Call Session Control Function (CSCF)**
 - SIP servers or proxies used to process SIP signaling packets in the IMS.

IMS Core Networks

- ❑ Interfaces between logical entities are standardized



Introduction to TTCN-3

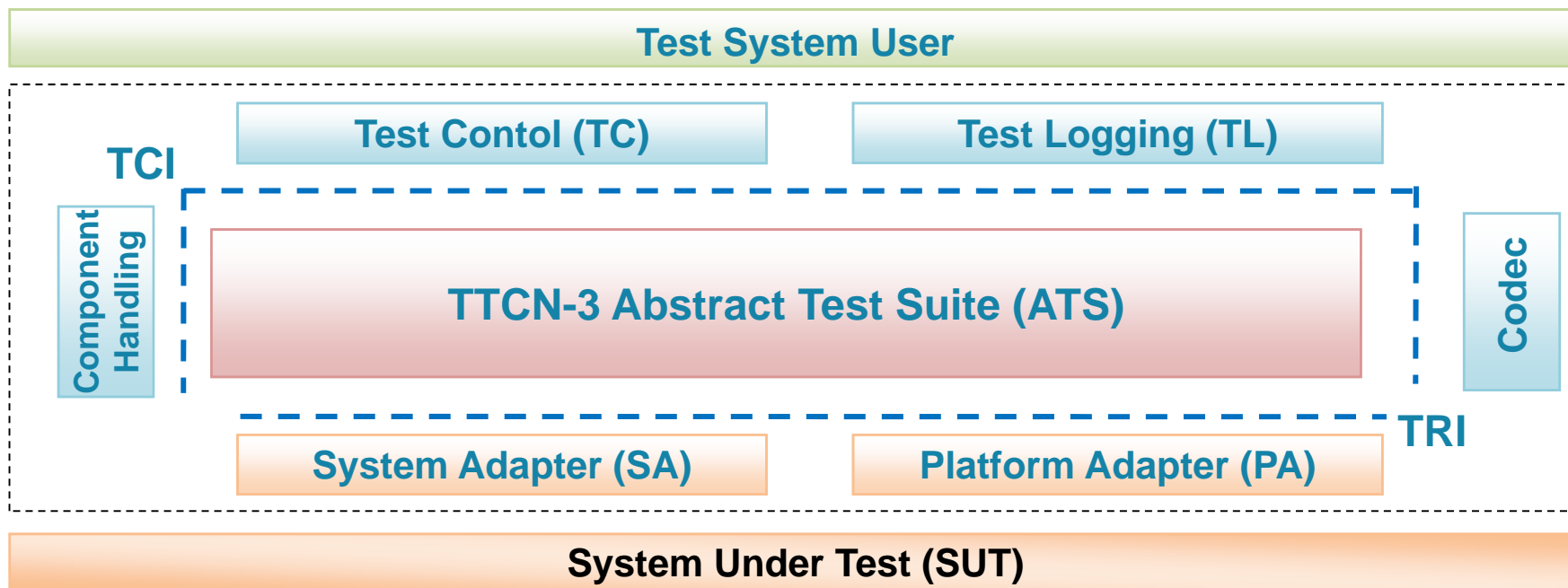
Source: www.ttcn-3.org

- ❑ **Internationally standardized testing language**
- ❑ **Developed and maintained by the ETSI Technical Committee MTS**
- ❑ **Specifically designed for testing and certification**
- ❑ **Can be applied to a variety of application domains and types of testing**
- ❑ **Proven to work in very large and complex industrial tests, e.g., of 3G network elements**
 - **TTCN-3 test suites for IMS, LTE and SIP**

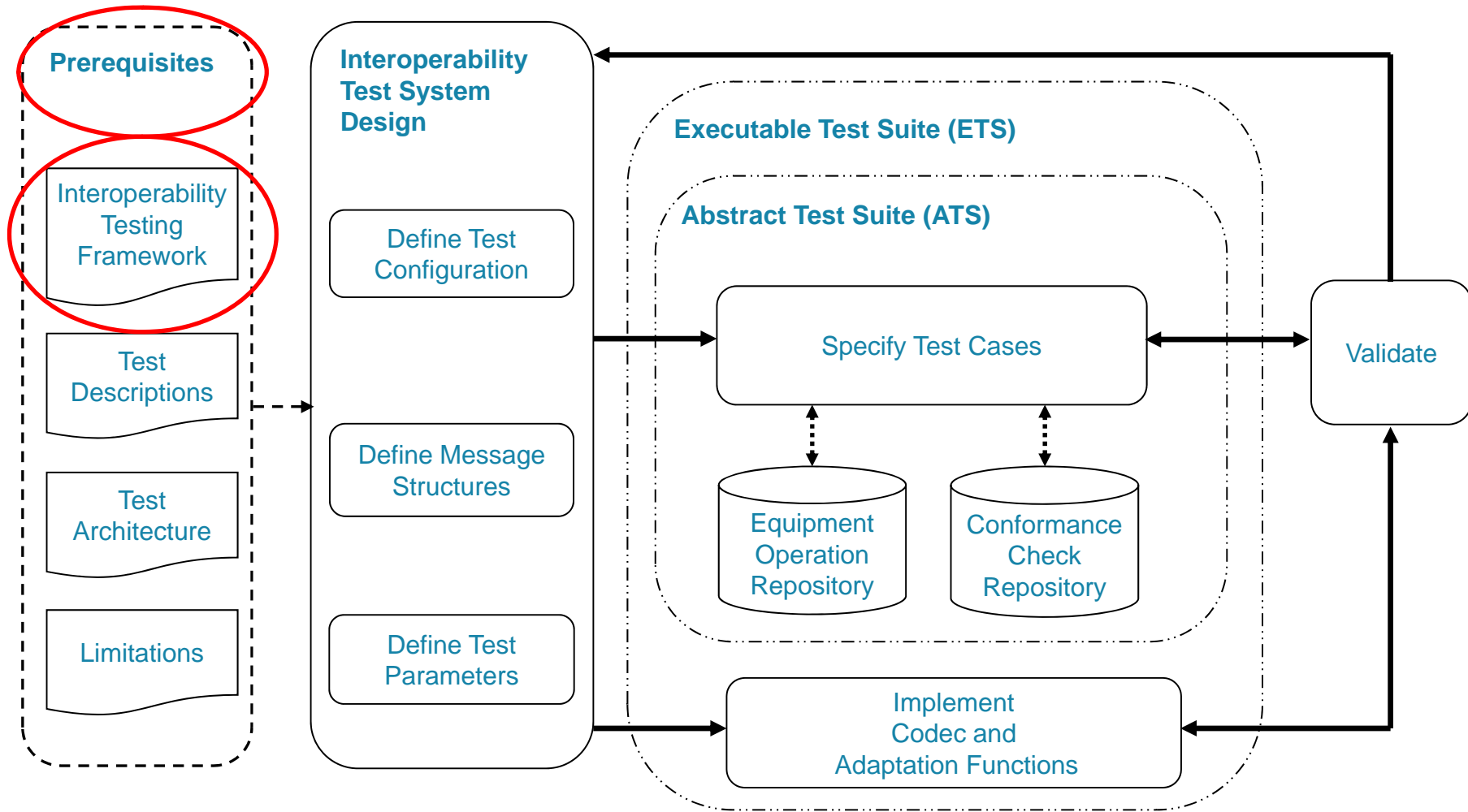
TTCN-3

Conceptual Test System Architecture

- The test system architecture is standardized via two interfaces
 - TTCN-3 Control Interface (TCI)
 - TTCN-3 Runtime Interface (TRI)



Development of Interoperability Test Systems in Practice





About ETSI's Interoperability Testing Framework

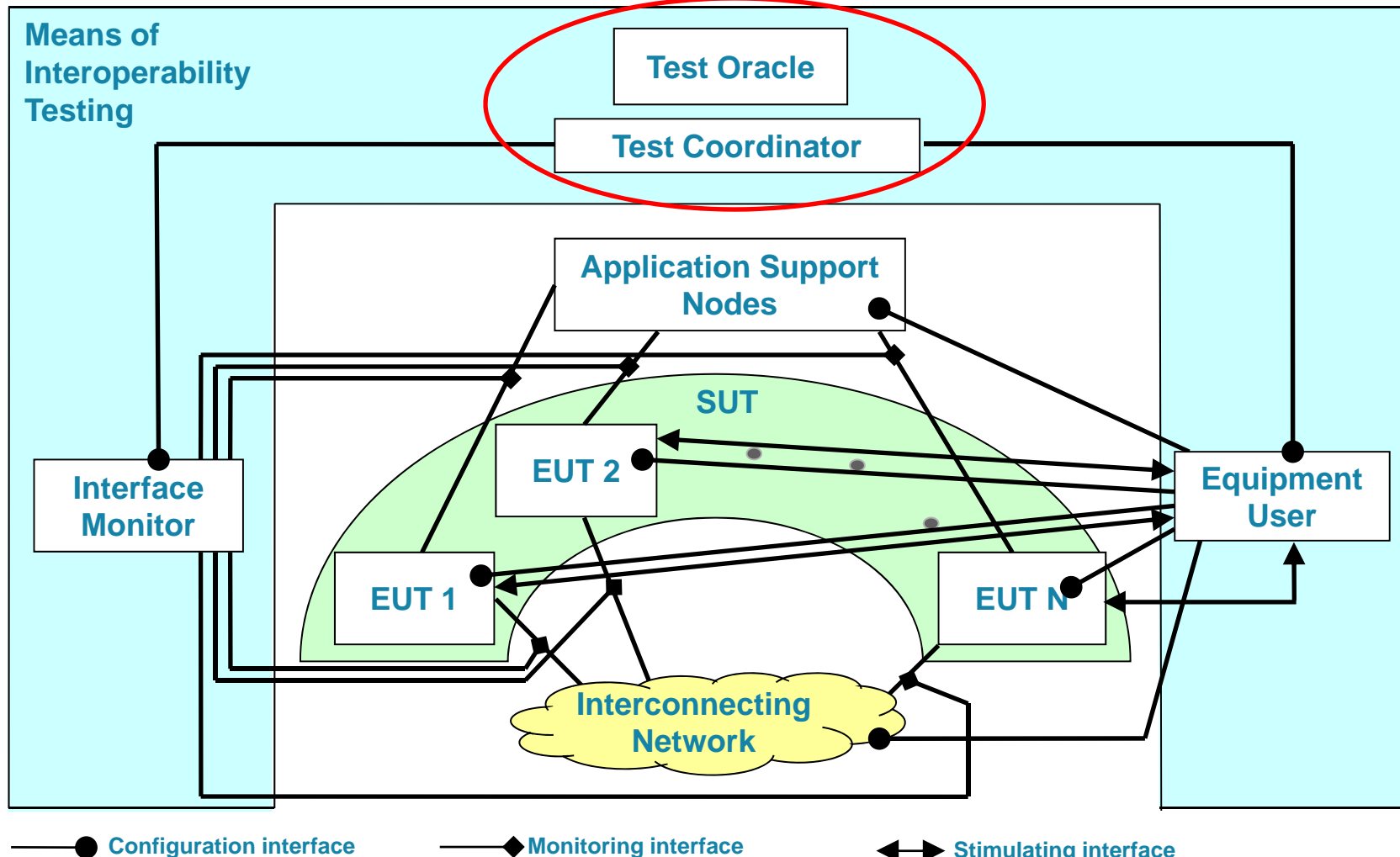
- ❑ A set of guidelines how to specify interoperability tests
- ❑ Realizes concepts and general architecture defined in the ETSI automated Interoperability Testing (IOT) methodology in TTCN-3
- ❑ Separates conformance and interoperability verdict management
- ❑ Supports live vs. offline interoperability test execution

Capture Mode	offline	live
Message checking	Performed on traffic capture files after testing	Performed on live capture during testing
Equipment operation	Disabled	manually or automatically

About ETSI's Interoperability Testing Framework

- ❑ Basis for test component type, test interface, and test parameter definition
- ❑ Can be implemented in a library based design
 - Libiot library
 - basic functionalities of IOT entities, i.e., test coordinator, equipment user, interface monitor, test oracle

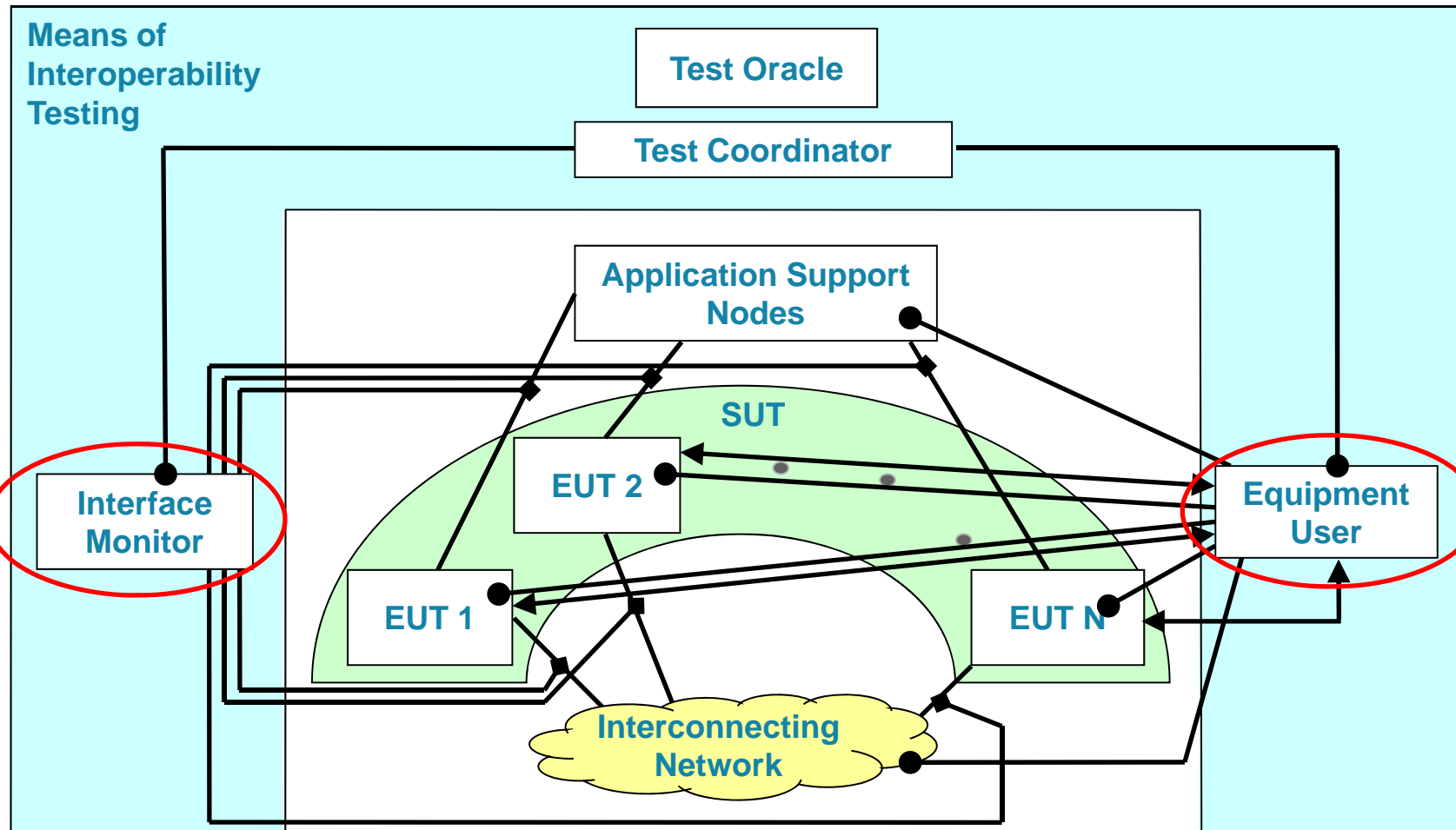
A Framework for Automated Interoperability Testing



Interoperability Testing Framework Test Entities Definitions

```
type component OracleServer extends ServerSyncComp {
    port VerdictPort vPort;
    var VerdictType vc_e3e_verdict := {none, "init"};
    var VerdictType vc_conf_verdict := {none, "init"};
}
type component OracleClient extends ClientSyncComp {
    port VerdictPort vPort;
}
type component TestCoordinator extends OracleServer {
    var ComponentIdList vc_compIds;
    port AdapterConfigPort acPort;
}
```

A Framework for Automated Interoperability Testing



Configuration interface
 Monitoring interface
 Stimulating interface

Interoperability Testing Framework Test Entities Definitions

```
type component EquipmentUser extends OracleClient {  
    port EquipmentAccessPort eaPort;  
    timer t_equipment;  
}
```

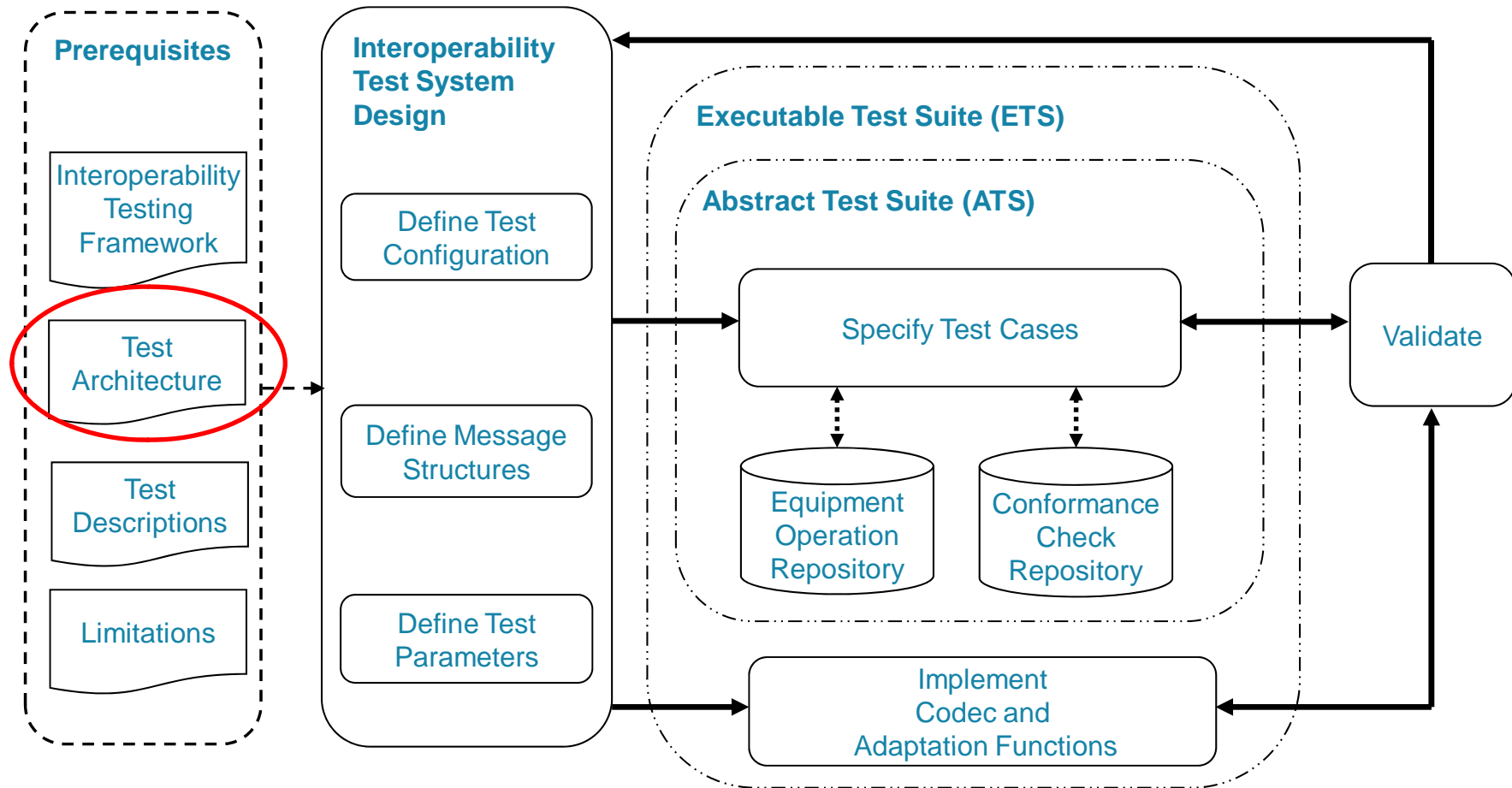
```
type component InterfaceMonitor extends OracleClient {  
    timer tc_wait := PX_MAX_MSG_WAIT;  
    var charstring vc_interfaceName := "UNDEFINED";  
    port AdapterConfigPort acPort;  
}
```

Interoperability Testing Framework Test Parameters

- ❑ Libraries are customized by setting test parameters
 - In the case of TTCN-3: module parameters
- ❑ Configuration information for each EUT
 - IP address(es), port(s), domain name
- ❑ Time related parameters
 - Timeout or duration time
- ❑ Test session pairing parameters
 - Test case specifications independence of specific EUT configuration information

Development of Interoperability Test Systems in Practice

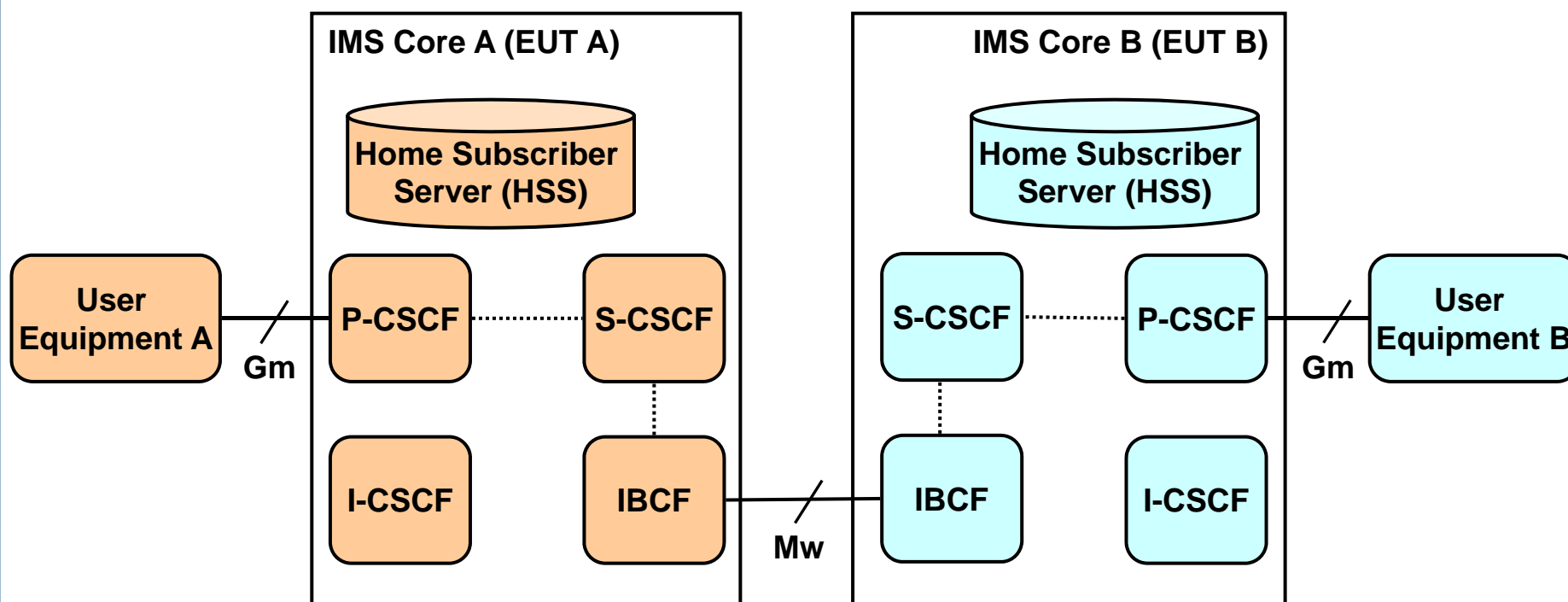
Prerequisites and Inputs



Test Architecture

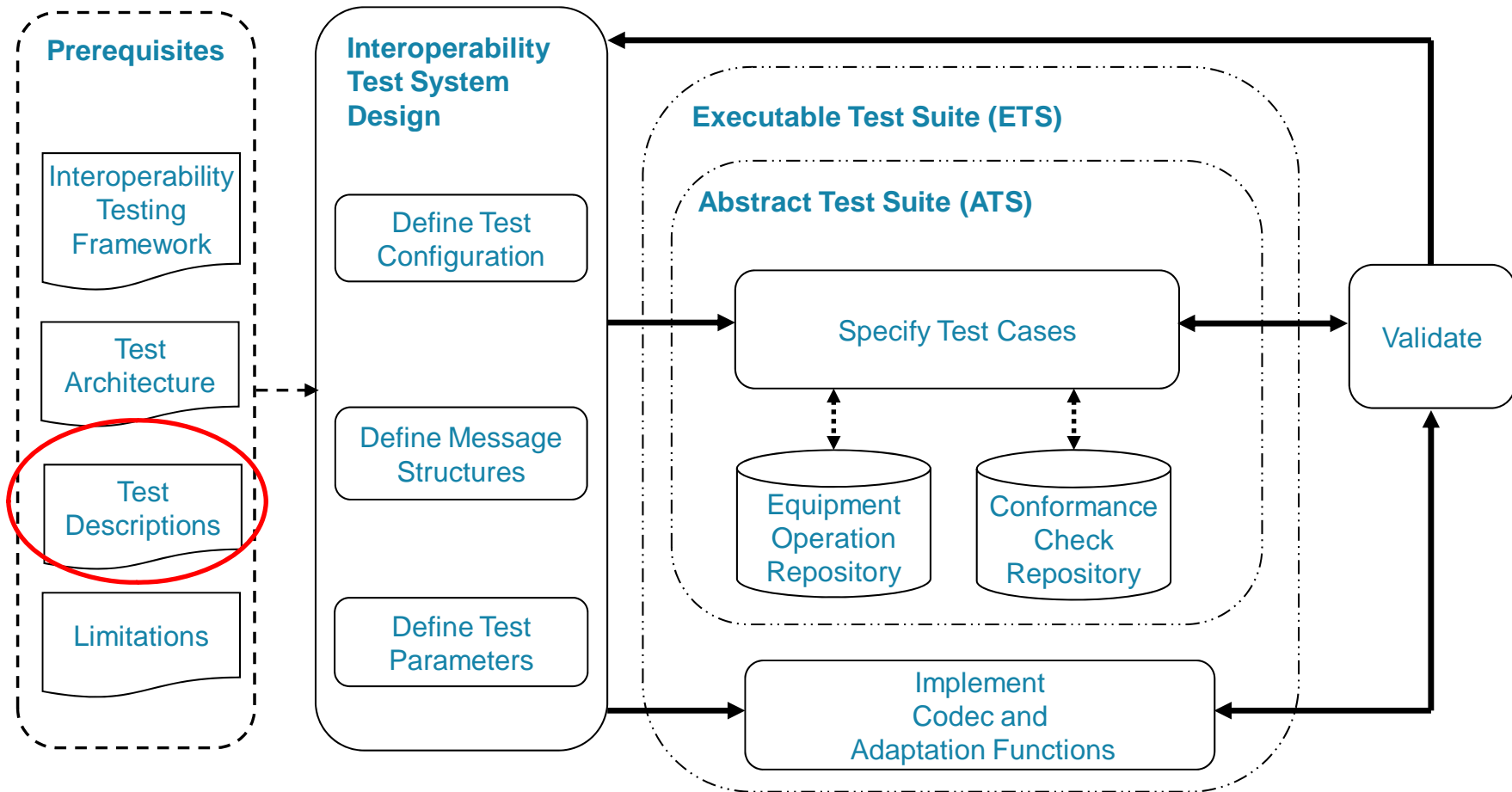
Example: Interworking IMS Core Networks

- A test architecture is an abstract description of logical entities as well as their interfaces and communication links involved in a test



Development of Interoperability Test Systems in Practice

Prerequisites and Inputs



Interoperability Test Descriptions (1/2)

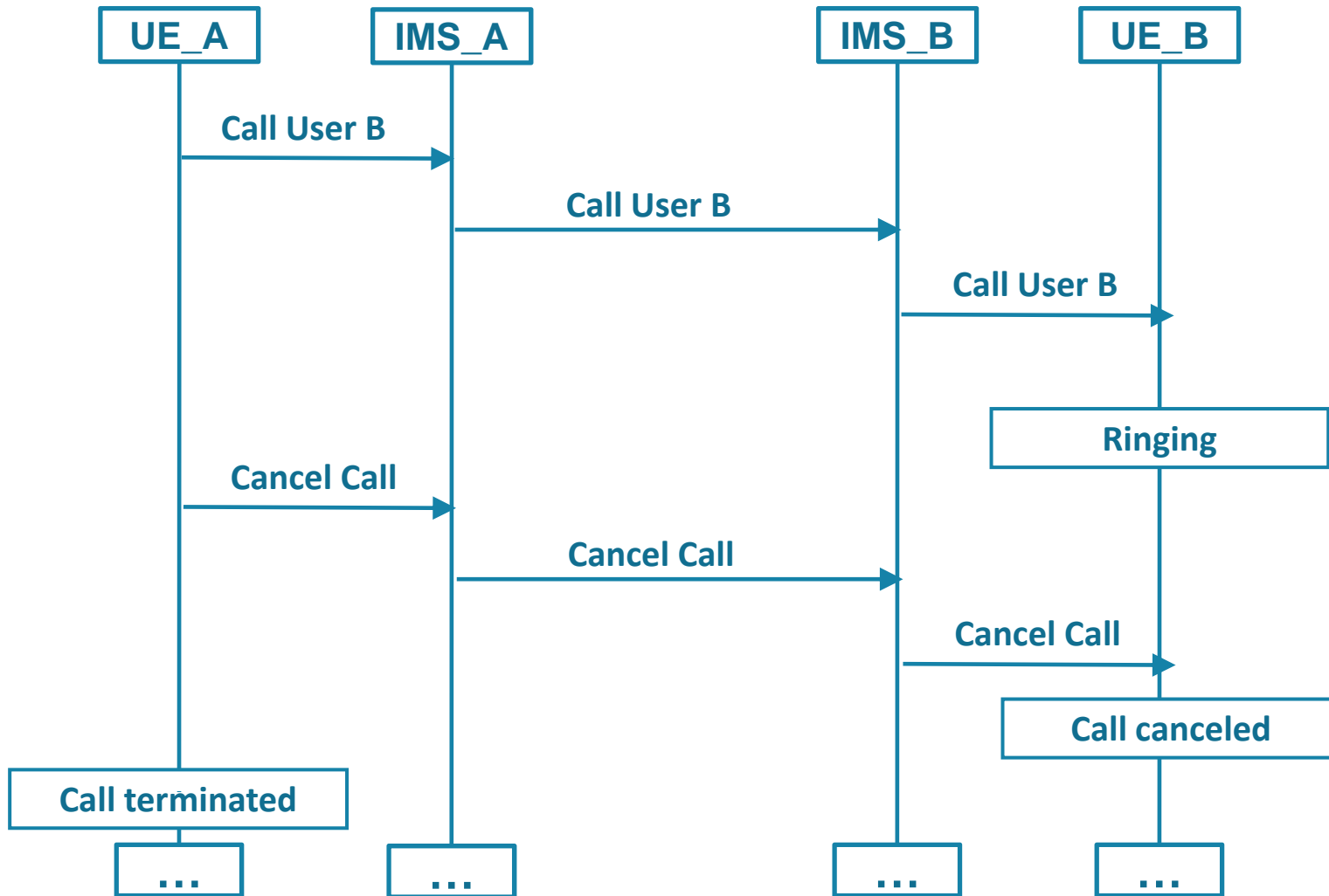
- ❑ Ideally unambiguous, informal descriptions of test cases usually written in English prose
- ❑ Capture
 - All equipment required for a test
 - Pre-conditions of a test
 - Equipment operation and observation during a test
 - Protocol messages or procedures to be checked
- ❑ An interoperability test description is derived from and includes an interoperability test purpose
 - Can also include conformance test purposes
 - Interoperability tests with conformance checks

Interoperability Test Descriptions

Example: IMS Call Cancellation by Calling User

Interoperability Test Description		
Identifier:	TD_IMS_CALL_0014	
TOP TP:	IMS network handles correctly cancels calling user call before its establishment	
Test Architecture:	Interworking IMS Core Networks (CF_INT_CALL)	
Specification Reference:	TS 124 229 [1], clause 5.4.3.2 ¶49	
Pre-test conditions:	<ul style="list-style-type: none"> • Home Subscriber Server (HSS) of IMS_A and of IMS_B is configured according to table 1 • UE_A is registered in IMS_A using any user identity • UE_B is registered in IMS_B using any user identity 	
Test Sequence:	Step	
	1	User A calls User B
	2	Verify that User B is informed of incoming call of User A
	3	Verify that User A is informed that UE_B is ringing
	4	User A cancels the call, before User B answers or before network timeout
	5	Verify that user B is informed that call has been cancelled
	6	Verify that user A is informed that call is terminated
Conformance Criteria:	Check	
	1	TP_IMS_5107_03 in CFW step 16 (CANCEL): ensure that { when { UE_A sends CANCEL to UE_B } then { IMS_B receives the CANCEL containing no Route_header indicating the S-CSCF_SIP_URI of IMS_A } }

Test Sequence



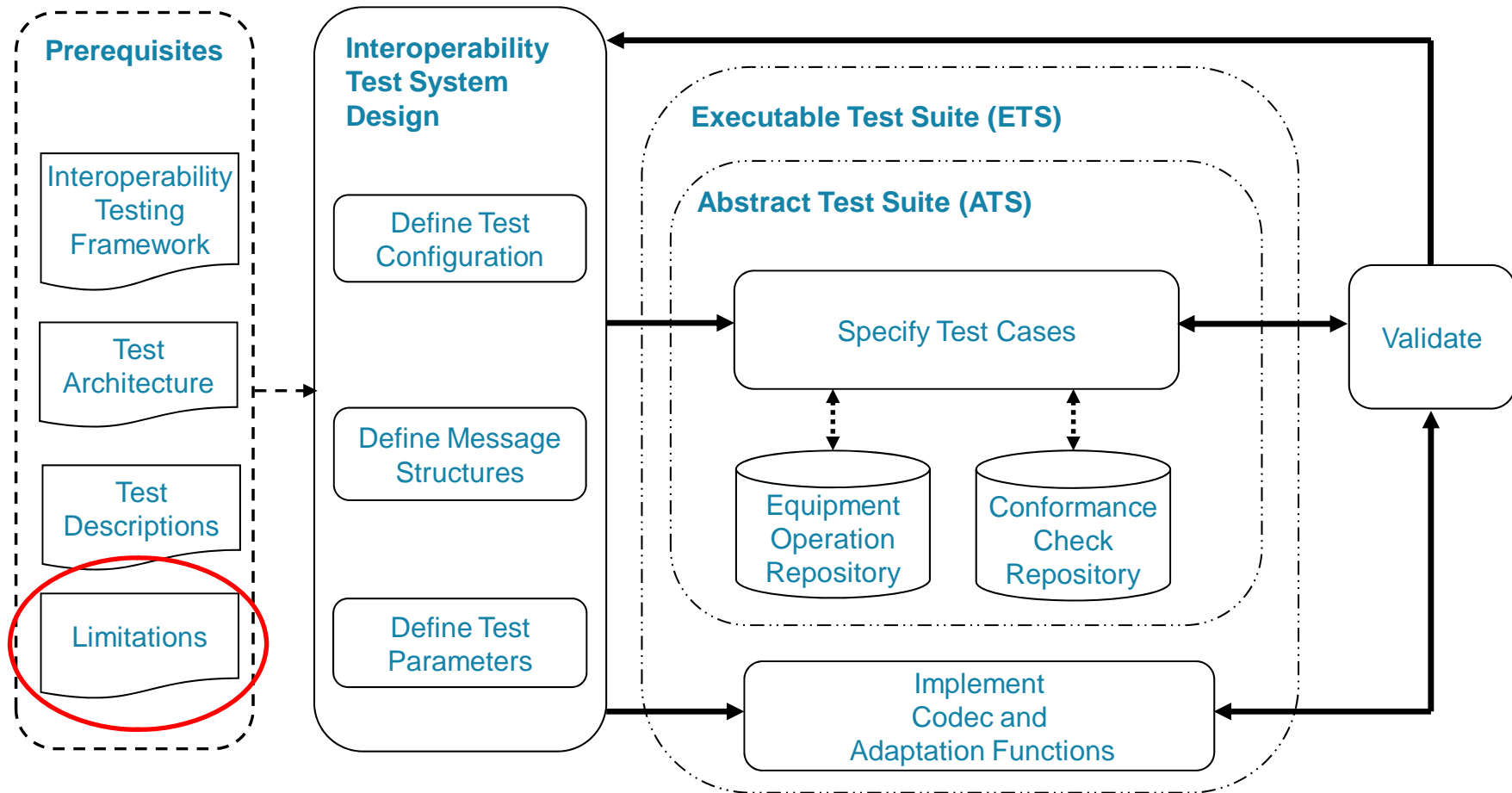
Interoperability Test Descriptions

Example: IMS Call Cancellation by Calling User

Interoperability Test Description		
Identifier:	TD_IMS_CALL_0014	
IOP TP:	IMS network handles correctly cancels calling user call before its establishment	
Test Architecture:	Interworking IMS Core Networks (CF_INT_CALL)	
Specification Reference	TS 124 229 [1], clause 5.4.3.2 ¶49	
Pre-test conditions:	<ul style="list-style-type: none"> • Home Subscriber Server (HSS) of IMS_A and of IMS B is configured according to table 1 • UE_A is registered in IMS_A using any user identity • UE_B is registered in IMS_B using any user identity 	
Test Sequence:	Step	
	1	User A calls User B
	2	Verify that User B is informed of incoming call of User A
	3	Verify that User A is informed that UE_B is ringing
	4	User A cancels the call, before User B answers or before network timeout
	5	Verify that user B is informed that call has been cancelled
	6	Verify that user A is informed that call is terminated
Conformance Criteria:	Check	
	1	TP_IMS_5107_03 in CFW step 16 (CANCEL): ensure that { when { UE_A sends CANCEL to UE_B } then { IMS_B receives the CANCEL containing no Route_header indicating the S-CSCF_SIP_URI of IMS_A } }

Development of Interoperability Test Systems in Practice

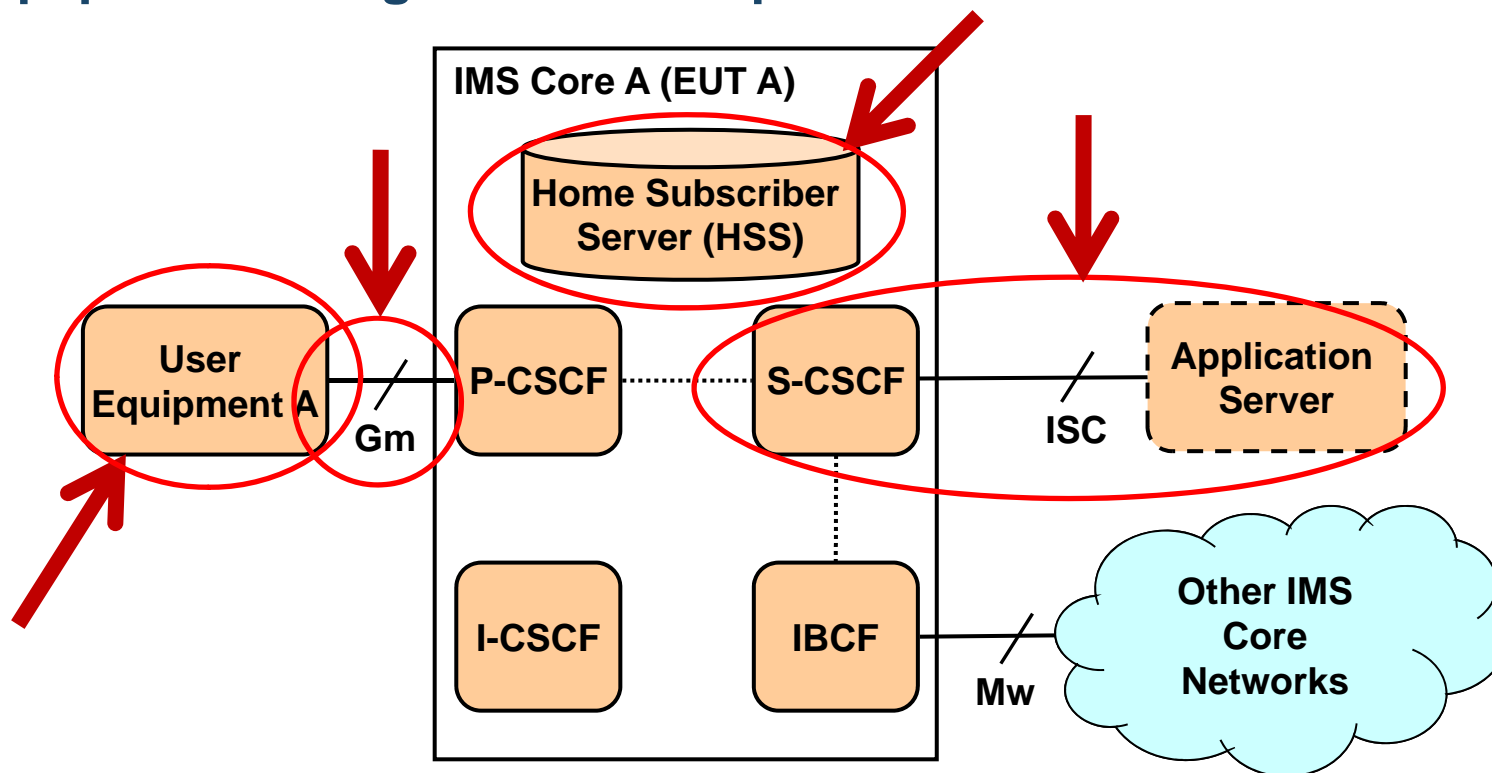
Prerequisites and Inputs



Limitations

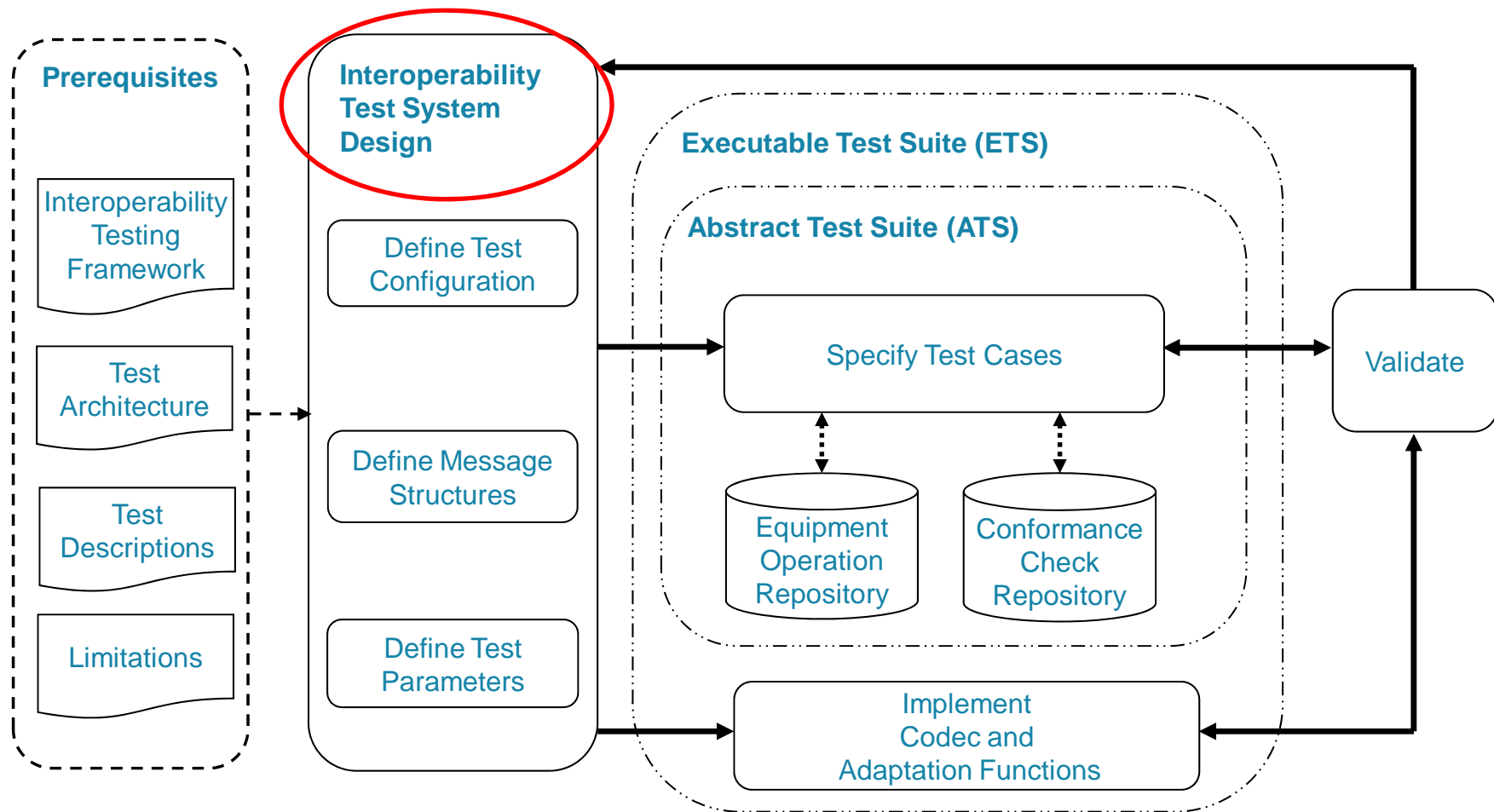
Example: IMS Core Networks

- ❑ Authentication and security
- ❑ Accessibility
- ❑ Equipment configuration and operation



Development of Interoperability Test Systems in Practice

Interoperability Test Design

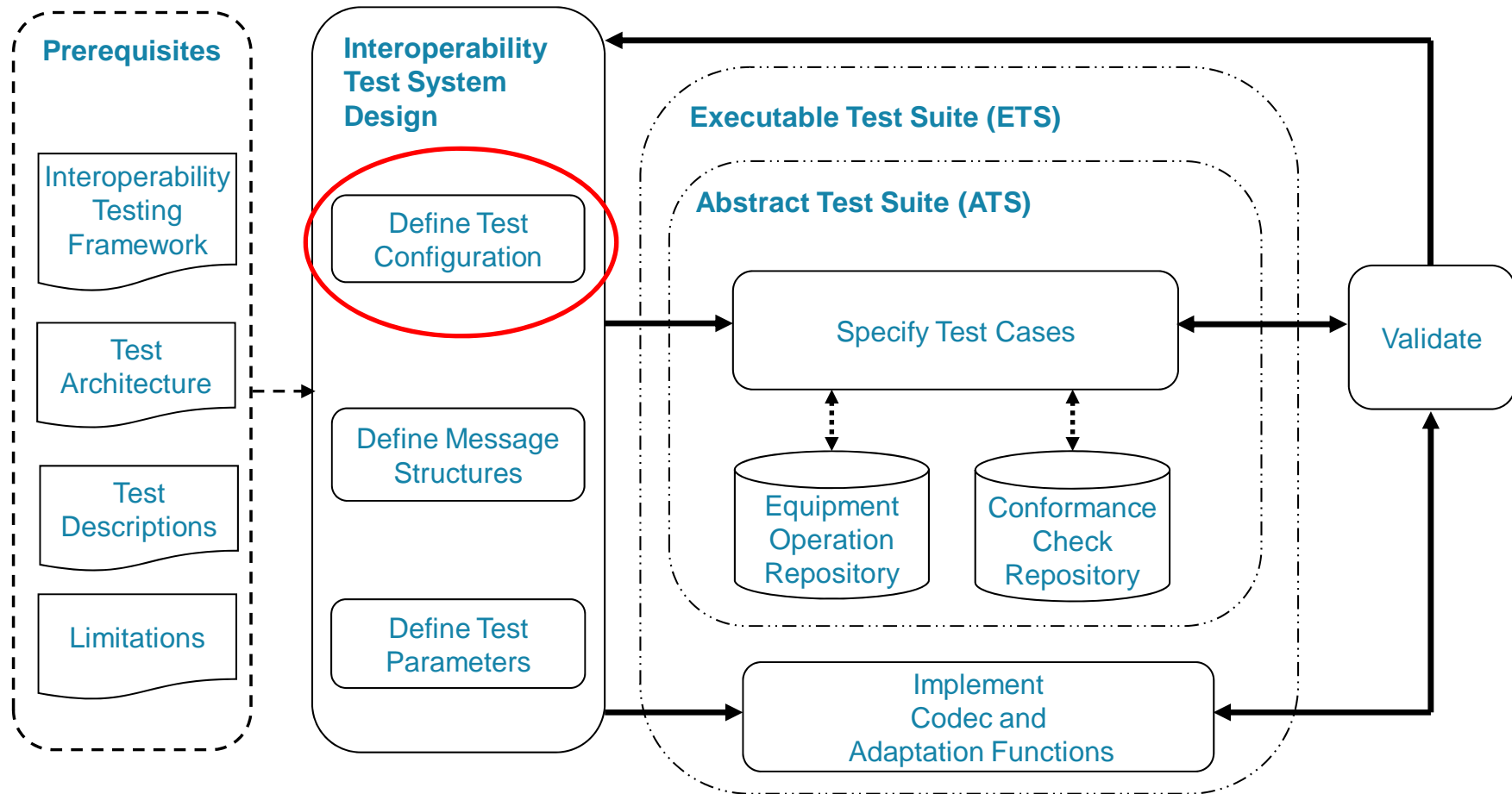


Interoperability Test Design General Guidelines

- Use naming conventions
- Use modularization → Library concept
- Develop function design
- Abstract handling of proprietary interfaces
- Configure message skipping in monitoring
- Manage of EUT interface information
- Document code

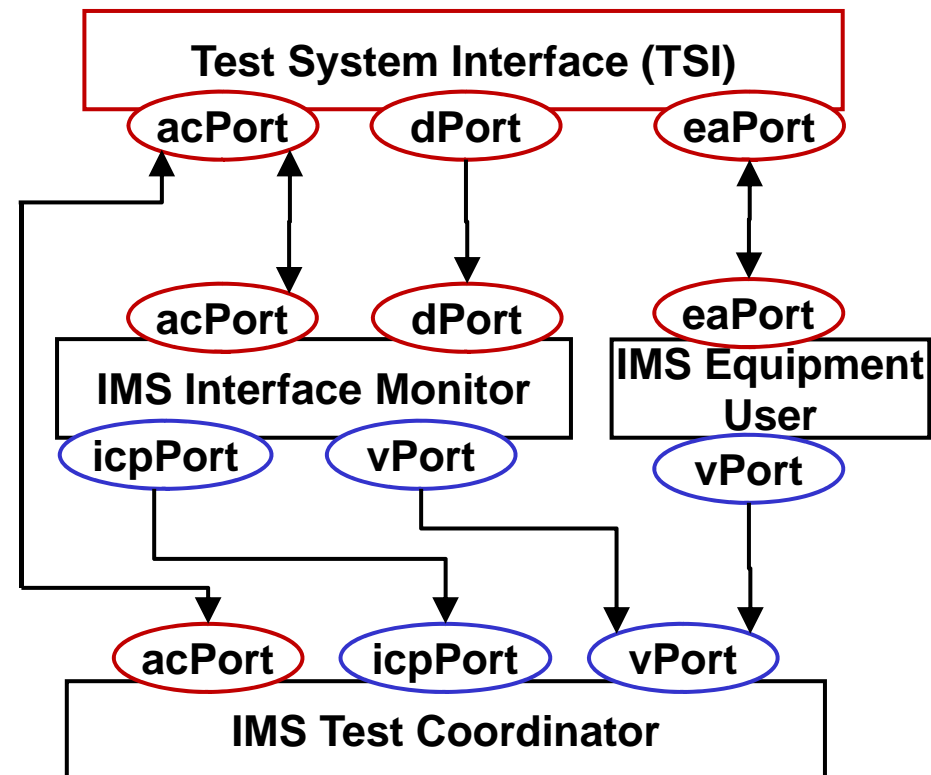
Development of Interoperability Test Systems in Practice

Interoperability Test Design



Test Configuration: Abstract Port Mappings/Connections

- ❑ Test System Interface (TSI)
 - Adapter configuration port (acPort)
 - Data port (dPort)
 - Equipment access port (eaPort)
- ❑ IMS Test Coordinator acts as Main Test Component (MTC)
- ❑ IMS coordination port (icpPort) for forwarding messages to MTC
- ❑ Verdict port (vPort) for communicating local verdicts

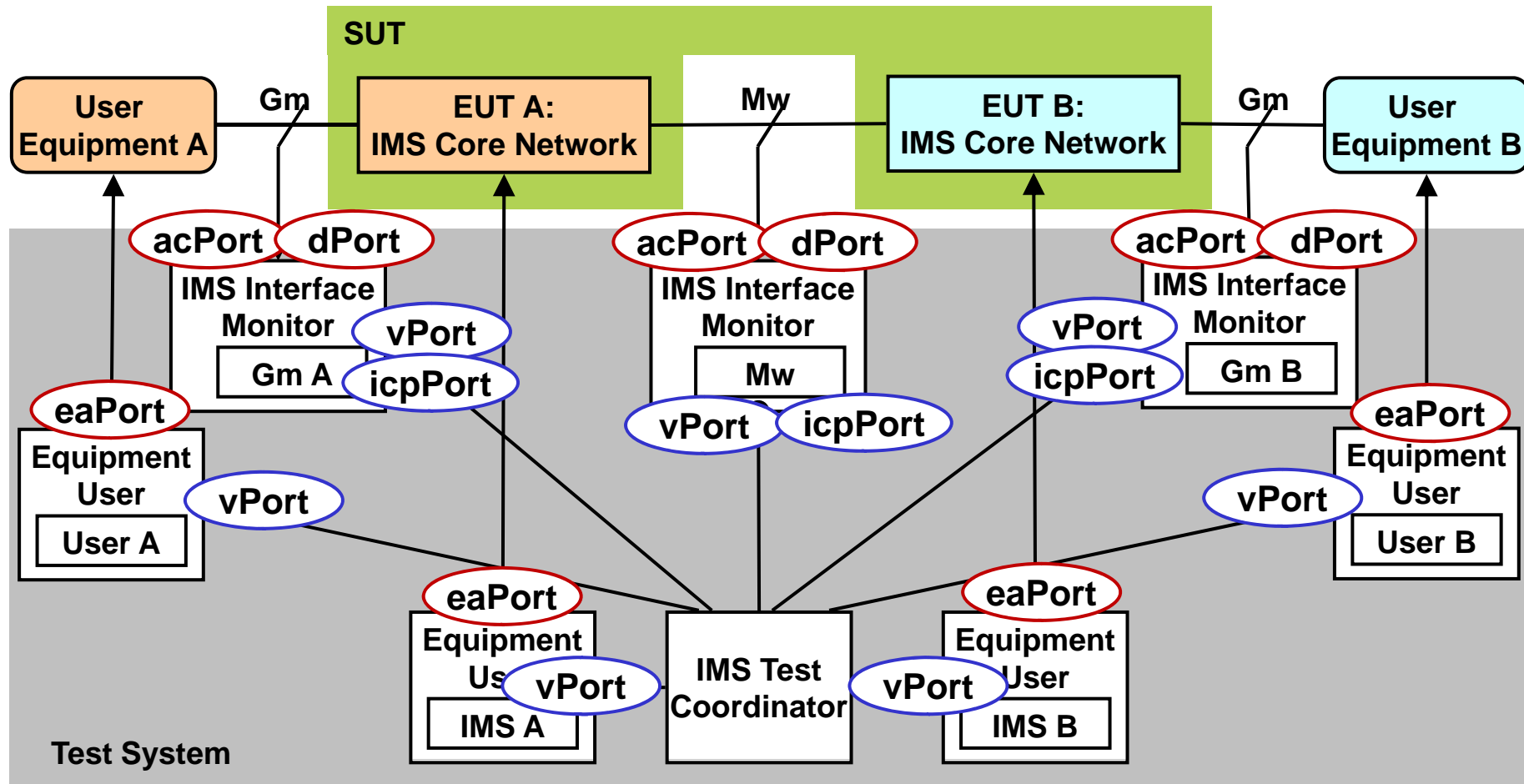


Test Configuration

- A concrete instance of a test architecture defined on the basis of test entities, communication links and their connection**
- Identifies EUTs, monitor points, and test entities**
- Defines entity instance which should realize the functionality of one or more test entity types**

Test Configuration

Example: Interworking IMS Core Networks



Legend:
 → Monitor interface
 ↔ Equipment operation interface
 —●— Control interface
 — SIP interface

Test Configuration Example: IMS IOT with TTCN-3

```
type component IotSystemInterface { // system component
  port DataPort dPort;
  port EquipmentAccessPort eaPort;
  port AdapterConfigPort acPort;
}
```

```
type component ImsInterfaceMonitor
extends InterfaceMonitor {
  port DataPort dPort;
  port ImsCoordinationPort icpPort;
}
```


Test Configuration Example: IMS IOT with TTCN-3

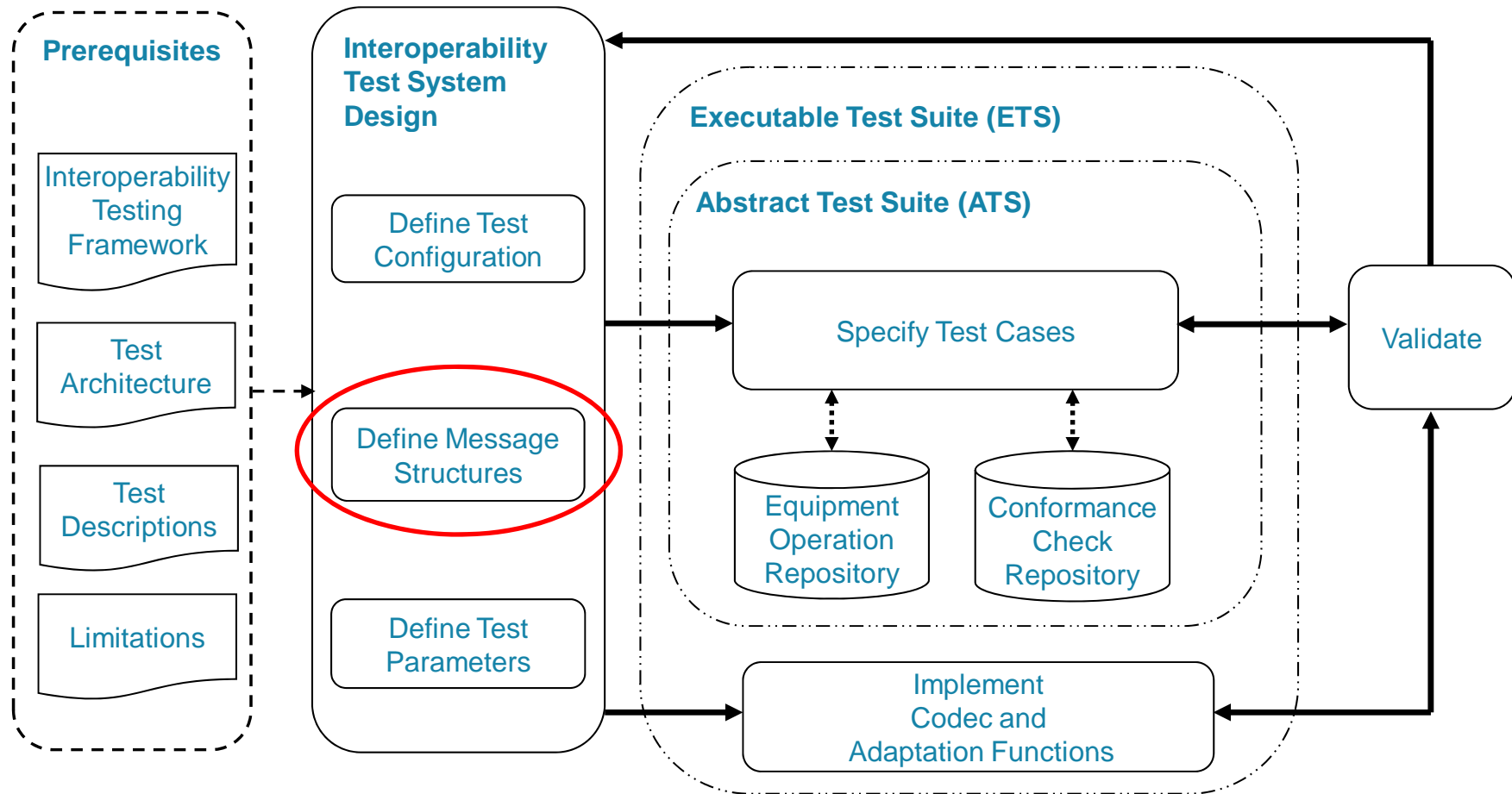
```
type port ImsCoordinationPort message {  
    inout SipMessage;  
}
```

```
// Test Configuration: Interworking IMS Core  
// Networks
```

```
type record CF_INT_CALL {  
    ImsInterfaceMonitor gmA,  
    ImsInterfaceMonitor mw,  
    ImsInterfaceMonitor gmB optional  
}
```

Development of Interoperability Test Systems in Practice

Interoperability Test Design



Message Structures

Example: TTCN-3 Types for a SIP Cancel

```

type record Request { //SIP Request from IETF RFC 3261
  RequestLine      requestLine,
  MessageHeader    msgHeader,
  MessageBody      messageBody optional,
  Payload          payload optional
}

```

```

type record RequestLine {
  Method          method,
  SipUrl          requestUri,
  charstring      sipVersion
}

```

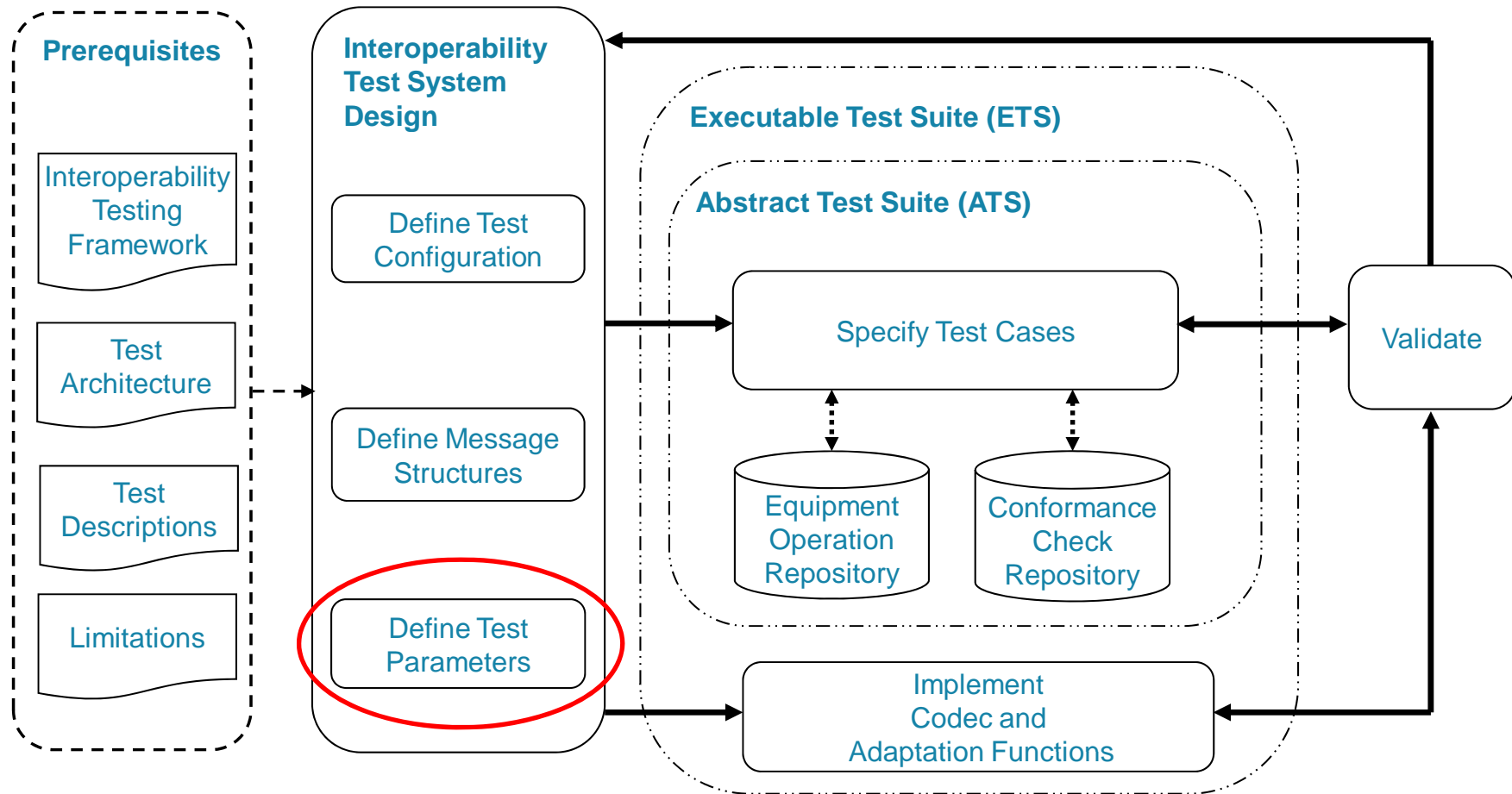
```

type enumerated Method {
  ACK_E,
  BYE_E,
  CANCEL_E,
  INVITE_E, // ...
}

```

Development of Interoperability Test Systems in Practice

Interoperability Test Design



Test Parameter Examples

- ❑ **Specify EUT interface information**
 - IP addresses and ports of interfaces, domain names, user identities
- ❑ **Specify time limit parameter**
- ❑ **Enable and disable observations of specific interfaces**
 - Interface(s) may not be accessible
 - Enable/disable execution of associated monitor components in test
- ❑ **Select capture mode for lower tester adapter, i.e., offline or live**
- ❑ **Configure general lower test adapter settings**
 - Dependent on capture mode selection

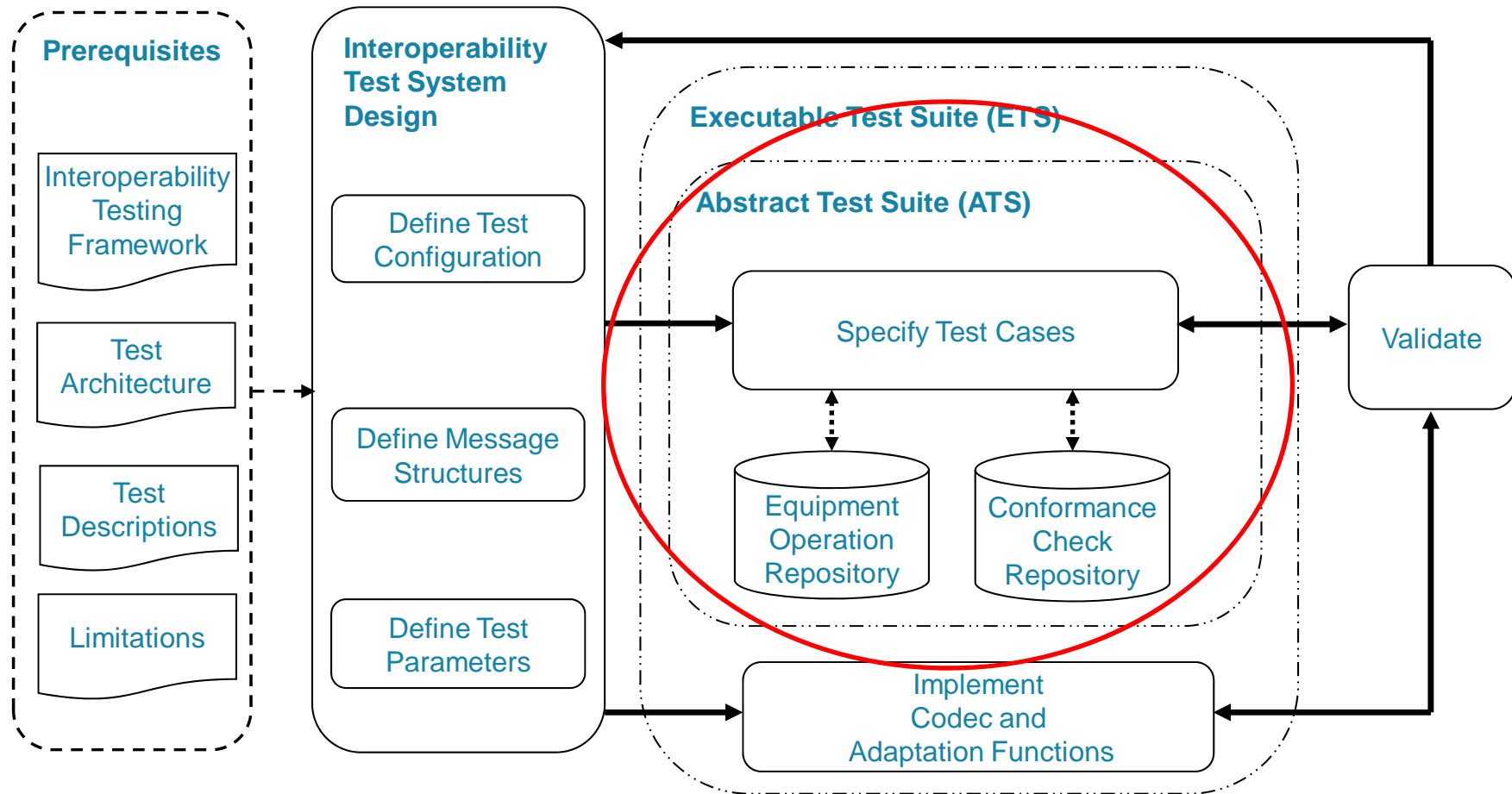
Test Parameters in TTCN-3

Configuration of EUT interface information

- A module parameter captures all information for each supported interface of each product participating in an interoperability event

```
modulepar ProductList PX_PRODUCTS := {
  { // productIndex = 0
    productName := "Super IMSCore",
    monitorInterfaces := {
      { interfaceName := "Mw",
        interfaceInfo := {
          IpInterfaceInfo := {
            { domainName := "pcscf.core.etsi",
              IPAddress := "192.86.1.97",
              portNumbers := {5060}
            },
            { domainName := "icscf.core.etsi",
              IPAddress := "192.86.1.98",
              portNumbers := {5060}
            }
          }
        }
      }
    }
  }
}
```

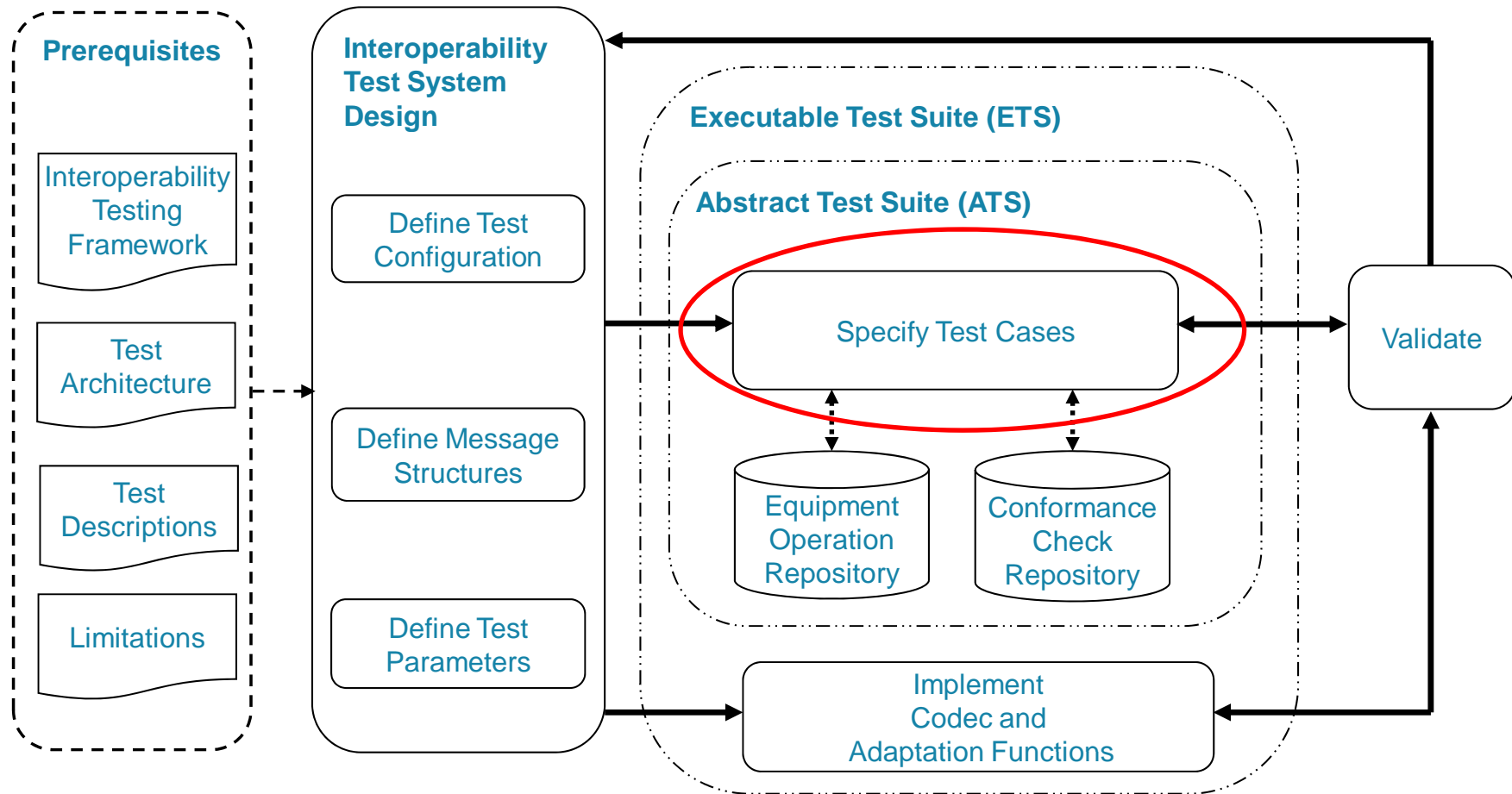
The Development of an Interoperability Test System in Practice – The Abstract Test Suite (ATS)



Abstract Test Suite (ATS) Specification

- ❑ **Specific TTCN-3 definitions**
 - test configuration management, test case statements, test purpose checking functions
- ❑ **Imports interoperability testing framework libraries**
- ❑ **Imports (or defines) technology specific libraries (e.g. protocols libraries)**
- ❑ **ATS consists of modules:**
 - TestConfiguration, TestSystem, TypesAndValues
 - Functions, Templates, TestSteps, TestCases, TestControl

The Development of an Interoperability Test System in Practice – The Abstract Test Suite (ATS)



Specification of Test Cases

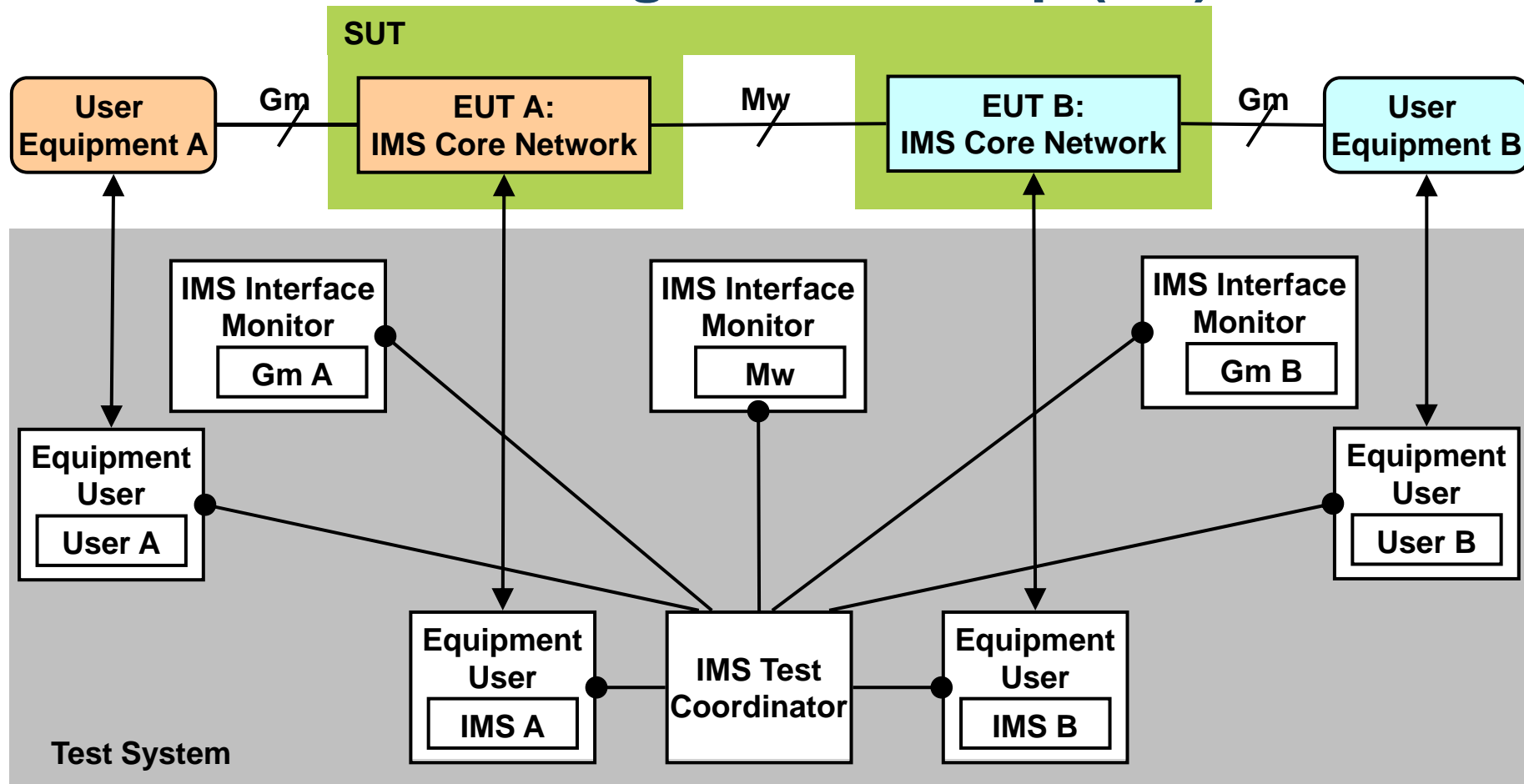
The TTCN-3 Testcase Statement

- ❑ Structure follows conventional test case implementation
 - `ImstestCoordinator` acts as MTC
 - `IotSystemInterface` represents the abstract interface to upper and lower test adapters

```
testcase TC_IMS_CALL_0014() runs on ImstestCoordinator
system IotSystemInterface {
  // 1. Test configuration setup
  // 2. Preamble
  // 3. Test body
  // 4. Postamble
  // 5. Tear down
}
```

Specification of Test Cases

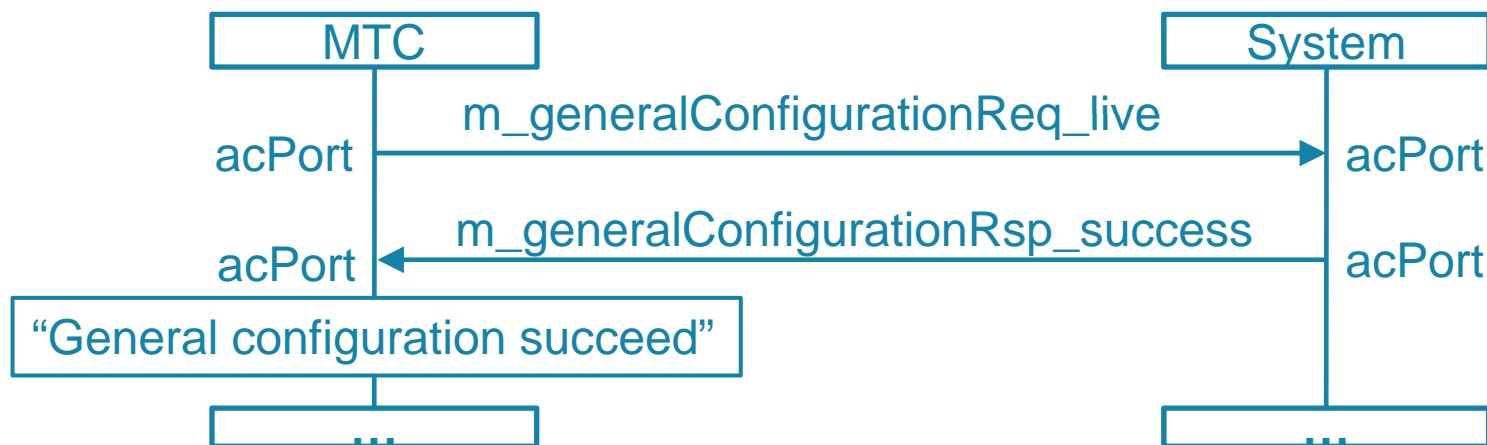
Test Configuration Setup (1/5)



Specification of Test Cases

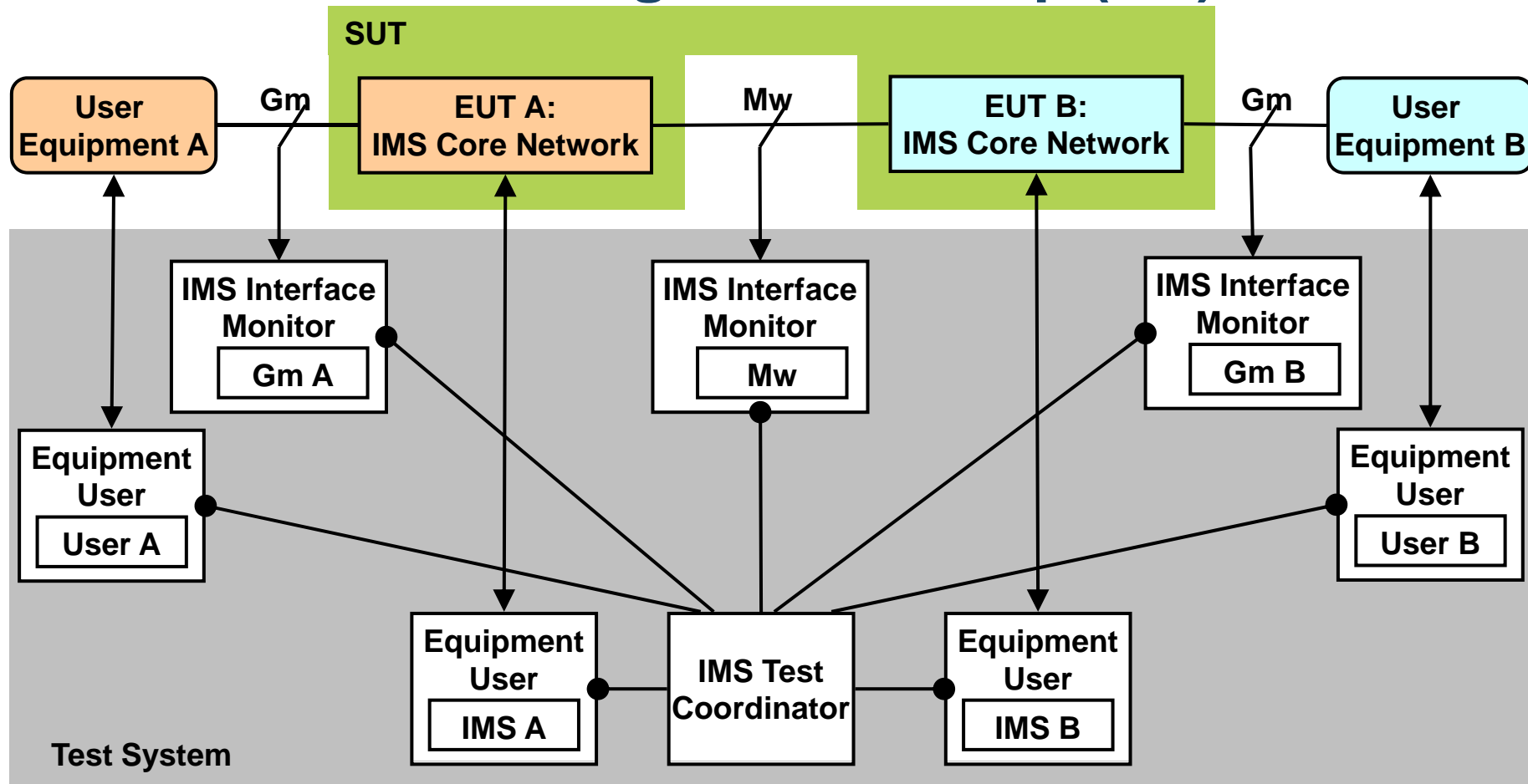
Test Configuration Setup (2/5)

- ❑ Lower test adapter is configured (e.g., for use of live mode)



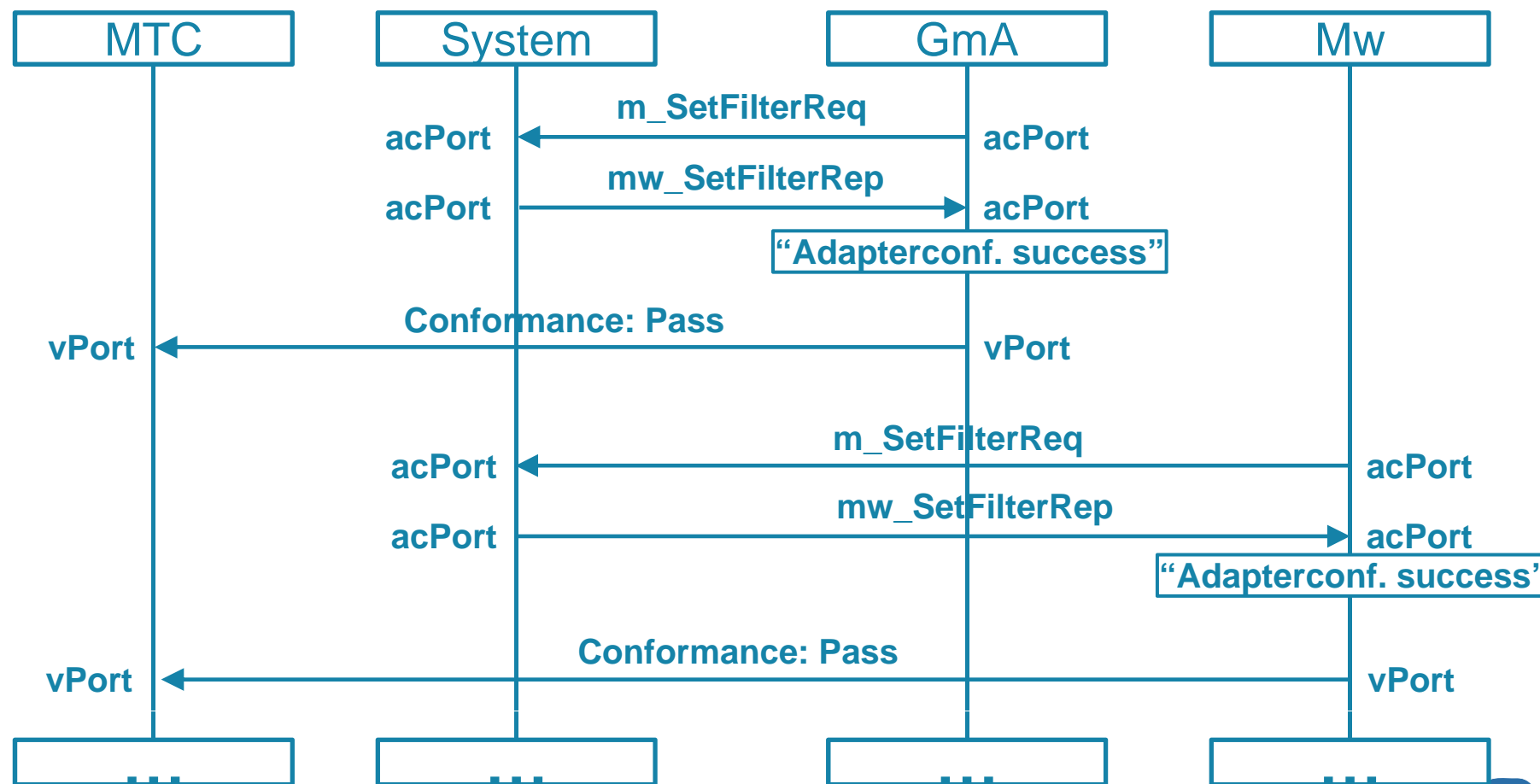
Specification of Test Cases

Test Configuration Setup (3/5)



Specification of Test Cases

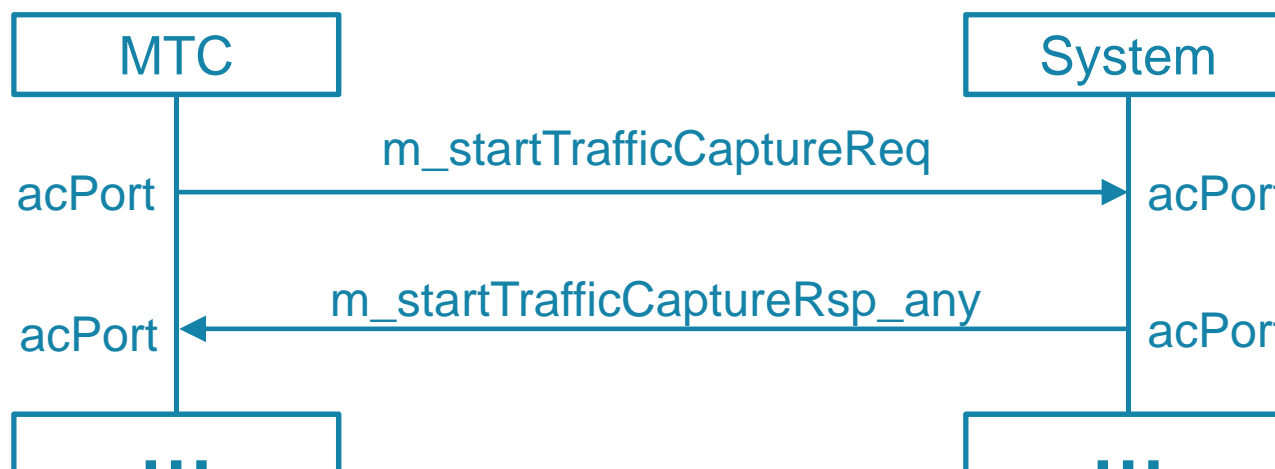
Test Configuration Setup (4/5)



Specification of Test Cases

Test Configuration Setup (5/5)

- ❑ After all lower test adapter configurations are completed, traffic capture processing is started by the MTC



Specification of Test Cases

Testcase Statement: Preamble

```
testcase TC_IMS_CALL_0014()  
runs on ImsTestCoordinator  
system IotSystemInterface {  
    // 1. Test configuration setup  
    // 2. Preamble  
    // 3. Test body  
    // 4. Postamble  
    // 5. Tear down  
}
```


Specification of Test Cases

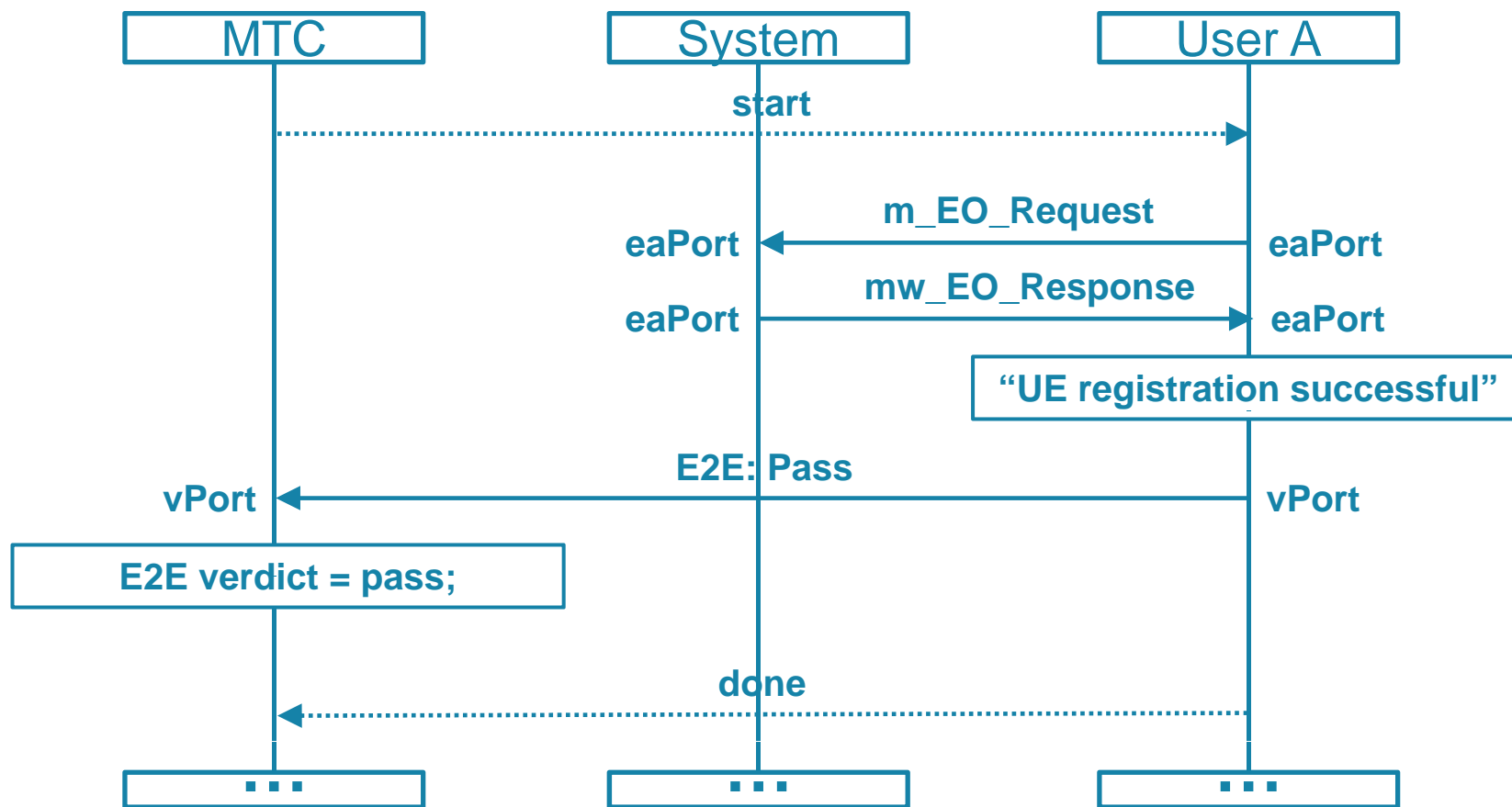
Interoperability Test Description - Preamble

Interoperability Test Description		
Identifier:	TD_IMS_CALL_0014	
IOP TP:	IMS network handles correctly cancels calling user call before its establishment	
Test Architecture:	Interworking IMS Core Networks	
Specification Reference:	TS 124 229 [1], clause 5.4.3.2 ¶49	
Pre-test conditions:	<ul style="list-style-type: none"> • HSS of IMS_A and of IMS_B is configured according to table 1 • UE_A is registered in IMS_A using any user identity • UE_B is registered in IMS_B using any user identity 	
Test Sequence:	Step	
	1	User A calls User B
	2	Verify that user B is informed of incoming call of User A
	3	Verify that user A is informed that UE_B is ringing
	4	User A cancels the call, before User B answers or before network timeout
	5	Verify that user B is informed that call has been cancelled
	6	Verify that user A is informed that call is terminated
Conformance Criteria:	Check	
	1	TP_IMS_5107_03 in CFW step 16 (CANCEL):
		ensure that {
		when { UE_A sends CANCEL to UE_B }
		then { IMS_B receives the CANCEL
	containing no Route_header	
	indicating the S-CSCF_SIP_URI of IMS_A }	
	}	

Specification of Test Cases

Preamble (1/2)

❑ Successful Registration of UE A in IMS A



Specification of Test Cases Preamble (2/2)

```
function f_userRegistration(  
    in charstring p_publicId, in charstring p_privateId,  
    in charstring p_pw)  
runs on EquipmentUser {  
    f_sendEquipmentCmd(valueof(  
        m_EO_Request(c_UE_REGISTRATION, {p_publicId,  
        p_privateId, p_pw}))); // LibIot function  
    }  
}
```

Specification of Test Cases

Testcase Statement: Test Body

```
testcase TC_IMS_CALL_0014()  
runs on ImsTestCoordinator  
system IotSystemInterface {  
    // 1. Test configuration setup  
    // 2. Preamble  
    // 3. Test body  
    // 4. Postamble  
    // 5. Tear down  
}
```

Specification of Test Cases

Interoperability Test Description – Test Body

Interoperability Test Description		
Identifier:	TD_IMS_CALL_0014	
IOP TP:	IMS network handles correctly cancels calling user call before its establishment	
Test Architecture:	Interworking IMS Core Networks	
Specification Reference	TS 124 229 [1], clause 5.4.3.2 ¶49	
Pre-test conditions:	<ul style="list-style-type: none"> • HSS of IMS_A and of IMS B is configured according to table 1 • UE_A is registered in IMS_A using any user identity • UE_B is registered in IMS_B using any user identity 	
Test Sequence:	Step	
	1	User A calls User B
	2	Verify that user B is informed of incoming call of User A
	3	Verify that user A is informed that UE_B is ringing
	4	User A cancels the call, before User B answers or before network timeout
	5	Verify that user B is informed that call has been cancelled
	6	Verify that user A is informed that call is terminated
Conformance Criteria:	Check	
	1	TP_IMS_5107_03 in CFW step 16 (CANCEL):
		ensure that {
		when { UE_A sends CANCEL to UE_B }
		then { IMS_B receives the CANCEL
	containing no Route_header	
	indicating the S-CSCF_SIP_URI of IMS_A }	
	}	

Specification of Test Cases

Test Body (1/9) – Equipment Operation

// 4. Test body

```
f_mtc_userInitiateCall(v_ueA, v_userInfoB); // Step 1
```

```
f_mtc_userCheckRinging(v_ueB); // Step 2
```

```
f_mtc_userCheckPeerIsRinging(v_ueA); // Step 3
```

```
f_mtc_userTriggerCancelCall(v_ueA); // Step 4
```

```
f_mtc_check_TP_IMS_5107_03_gm(v_gmA, false); // Check1
```

```
f_mtc_check_TP_IMS_5107_03_mw(v_mw, false); // Check1
```

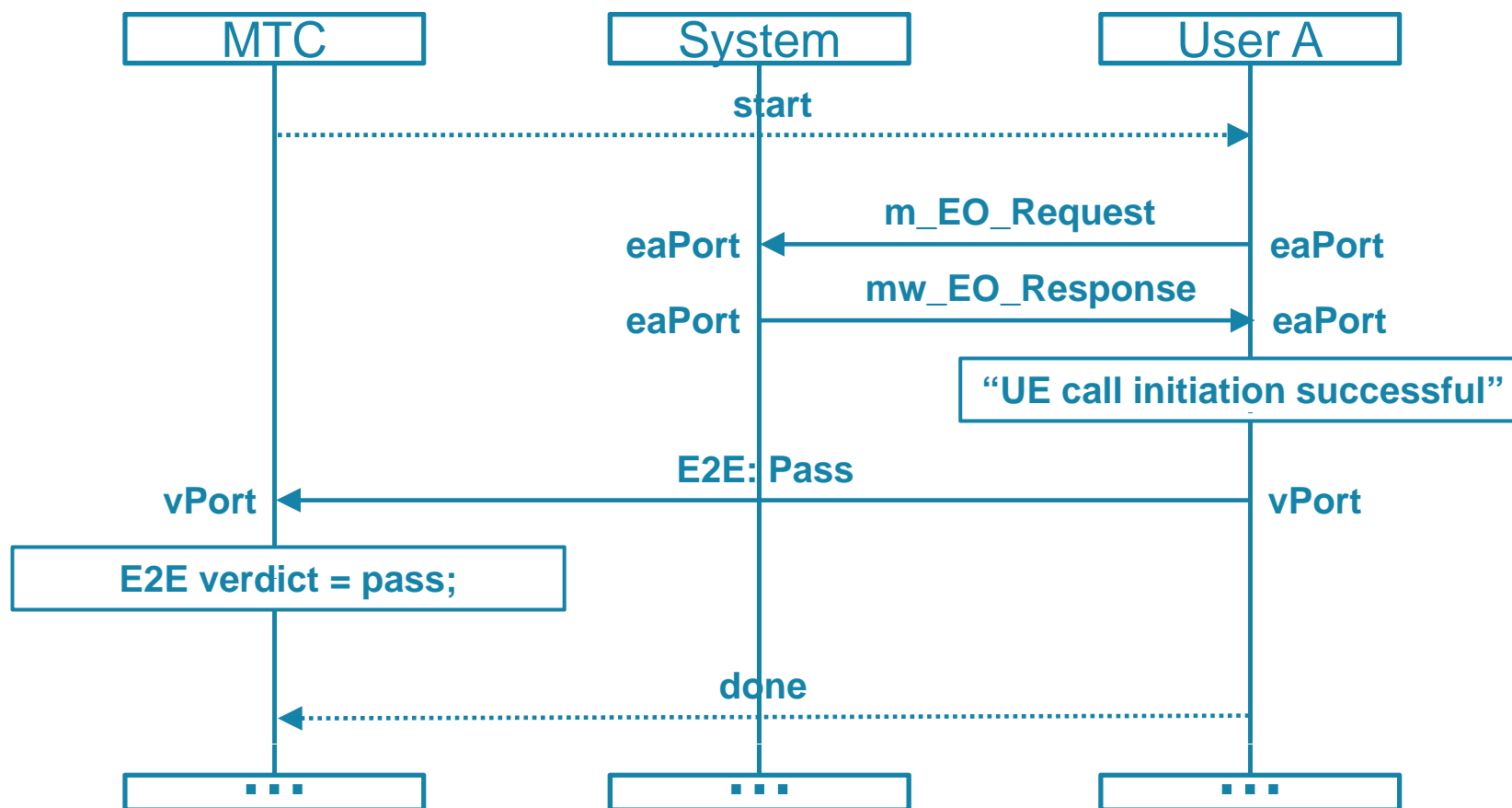
```
f_mtc_userCheckCallCancelled(v_ueB); // Step 5
```

```
f_mtc_userCheckCallEnded(v_ueA); // Step 6
```

Specification of Test Cases

Test Body (2/9) – Equipment Operation

- ❑ Successful call initiation by UE A



Specification of Test Cases

Test Body (3/9) – Equipment Operation

- Usage of a generic equipment operation function
 - Sends a IMS UE operation command
 - Sets local test component E2E verdict and reports it to MTC

```
function f_userCallInitiation(charstring p_publicID)
  runs on EquipmentUser {
  f_sendEquipmentCmd(valueof( // LibTot function
    m_EO_Request(c_UE_CALL_INITIATION, {p_publicID}))
  );
}
```


Specification of Test Cases

Test Body (4/9) – Conformance Checks

Interoperability Test Description		
Identifier:	TD_IMS_CALL_0014	
IOP TP:	IMS network handles correctly cancels calling user call before its establishment	
Test Architecture:	Interworking IMS Core Networks	
Specification Reference	TS 124 229 [1], clause 5.4.3.2 ¶49	
Pre-test conditions:	<ul style="list-style-type: none"> • HSS of IMS_A and of IMS_B is configured according to table 1 • UE_A is registered in IMS_A using any user identity • UE_B is registered in IMS_B using any user identity 	
Test Sequence:	Step	
	1	User A calls User B
	2	Verify that user B is informed of incoming call of User A
	3	Verify that user A is informed that UE_B is ringing
	4	User A cancels the call, before User B answers or before network timeout
	5	Verify that user B is informed that call has been cancelled
	6	Verify that user A is informed that call is terminated
Conformance Criteria:	Check	
	1	TP_IMS_5107_03 in CFW step 16 (CANCEL): ensure that { when { UE_A sends CANCEL to UE_B } then { IMS_B receives the CANCEL containing no Route_header indicating the S-CSCF_SIP_URI of IMS_A } }

Specification of Test Cases

Test Body (5/9) – Conformance Checks

// 4. Test sequence / test body

f_mtc_userInitiateCall(v_ueA, v_userInfoB); // Step 1

f_mtc_userCheckRinging(v_ueB); // Step 2

f_mtc_userCheckPeerIsRinging(v_ueA); // Step 3

f_mtc_userTriggerCancelCall (v_ueA); // Step 4

f_mtc_check_TP_IMS_5107_03_gm(v_gmA, false); // Check1

f_mtc_check_TP_IMS_5107_03_mw(v_mw, false); // Check1

f_mtc_userCheckCallCancelled(v_ueB); // Step 5

f_mtc_userCheckCallEnded(v_ueA); // Step 6

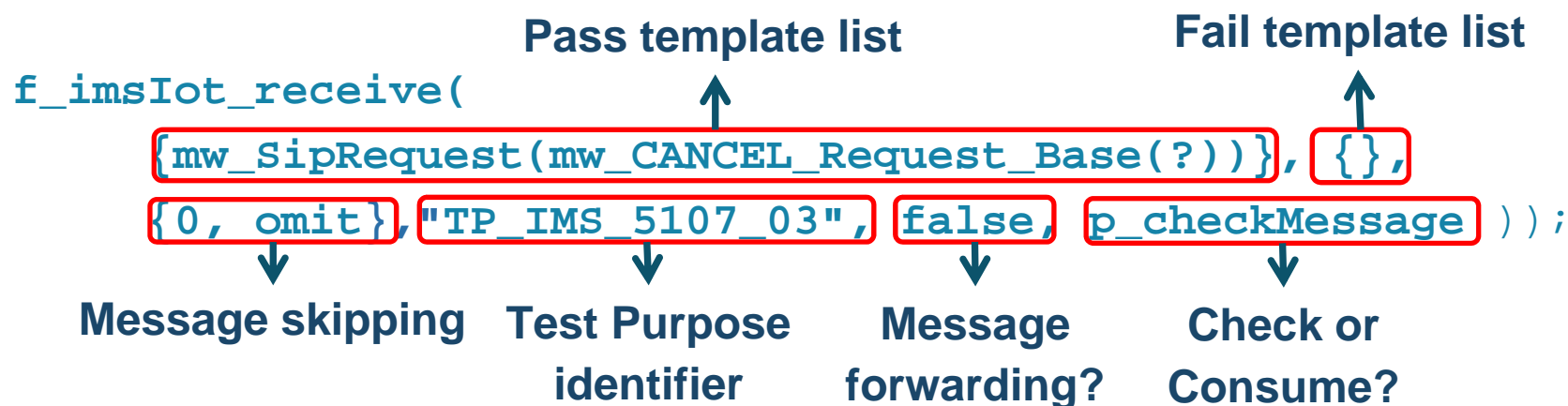
Specification of Test Cases

Test Body (6/9) – Gm Conformance Check

```
function f_mtc_check_TP_IMS_5107_03_gm(  
    ImsInterfaceMonitor p_monitorCompRef, boolean p_checkMessage )  
runs on ImsTestCoordinator {  
    p_monitorCompRef.start(  
        f_imsIot_receive(  
            {mw_SipRequest(mw_CANCEL_Request_Base(?))}, {},  
            {0, omit}, "TP_IMS_5107_03", false, p_checkMessage )  
        ),  
    p_monitorCompRef.done;  
}
```

Specification of Test Cases

Test Body (7/9) – Generic receive example



Template definitions:

- ❑ Base templates define very basic SIP message checks
 - Example: mw_CANCEL_Request_Base
- ❑ Complex checks are specified in templates derived from base templates
 - Example: mdw_TP_IMS_5107_03_mw (modifies mw_CANCEL_Request_Base)

Specification of Test Cases

Test Body (8/9) – Mw Conformance Check

```
function f_mtc_check_TP_IMS_5107_03_mw(  
    ImsInterfaceMonitor p_monitorCompRef, boolean  
    p_checkMessage )  
runs on ImsTestCoordinator {  
    var template SipUrl v_scscfImsAUrl :=  
        mw_SipUrl_Host(f_GetEUTScscfAddress(PX_EUT_A));  
    p_monitorCompRef.start(  
        f_imsIot_receive(  
            {mw SipRequest(  
                mdw_TP_IMS_5107_03_mw(?, v_scscfImsAUrl))},  
            {}, {0, omit}, "TP_IMS_5107_03", false, p_checkMessage  
        ));  
    p_monitorCompRef.done;  
}
```

Specification of Test Cases

Test Body (9/9) – Mw Conformance Check

❑ Message template for Mw conformance check of TP_IMS_5107_03

Conformance Criteria:	Check	
	1	TP_IMS_5107_03 in CFW step 16 (CANCEL): ensure that { when { UE_A sends CANCEL to UE_B } then { IMS_B receives the CANCEL containing no Route_header indicating the S-CSCF_SIP_URI of IMS_A } }

```

template CANCEL_Request mdw_TP_IMS_5107_03_mw (
    template CallId p_callId, template SipUrl p_SCSCF_SIP_URI)
modifies mw_CANCEL_Request_Base := {
    msgHeader := {
        route := (
            omit,
            { fieldName := ROUTE_E,
              routeBody :- {
                  *, complement(mw_routeBody(p_SCSCF_SIP_URI)), *
              }
            }
        )
    }
}
    
```

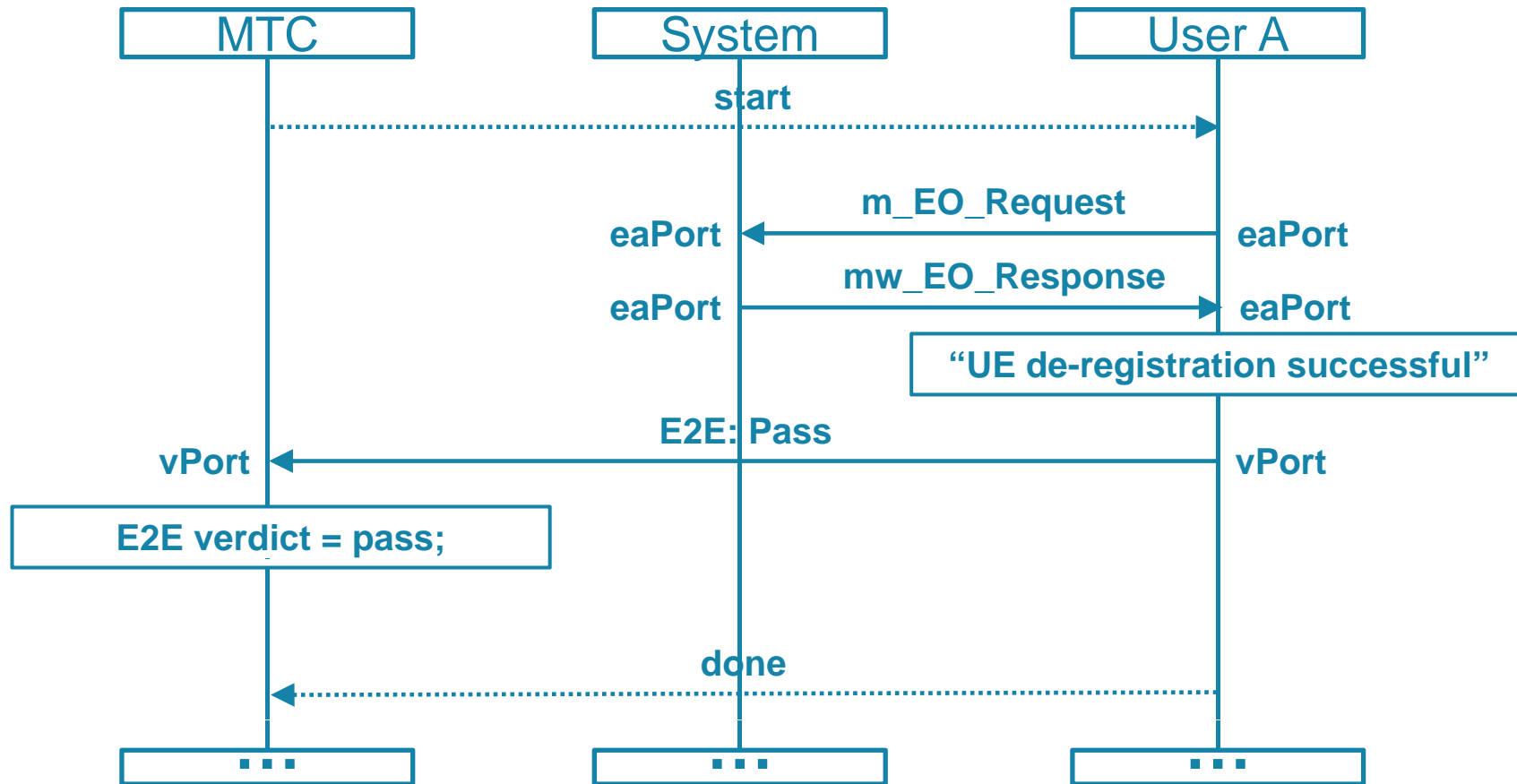
Specification of Test Cases

Testcase Statement – Postamble & Tear Down

```
testcase TC_IMS_CALL_0014()  
runs on ImsTestCoordinator  
system IotSystemInterface {  
    // 1. Test configuration setup  
    // 2. Preamble  
    // 3. Test body  
    // 4. Postamble  
    // 5. Tear down  
}
```

Specification of Test Cases

Postamble

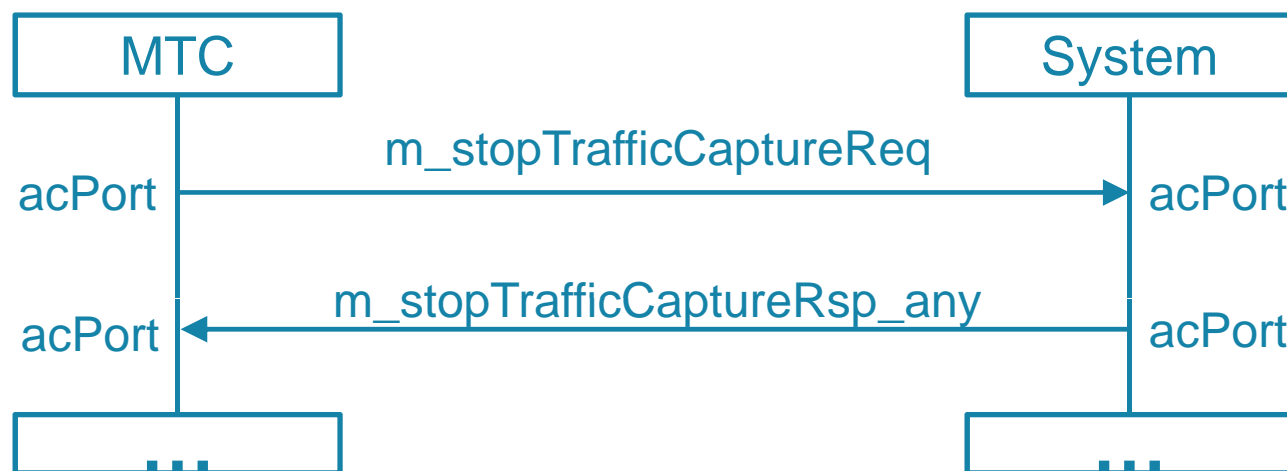


Specification of Test Cases

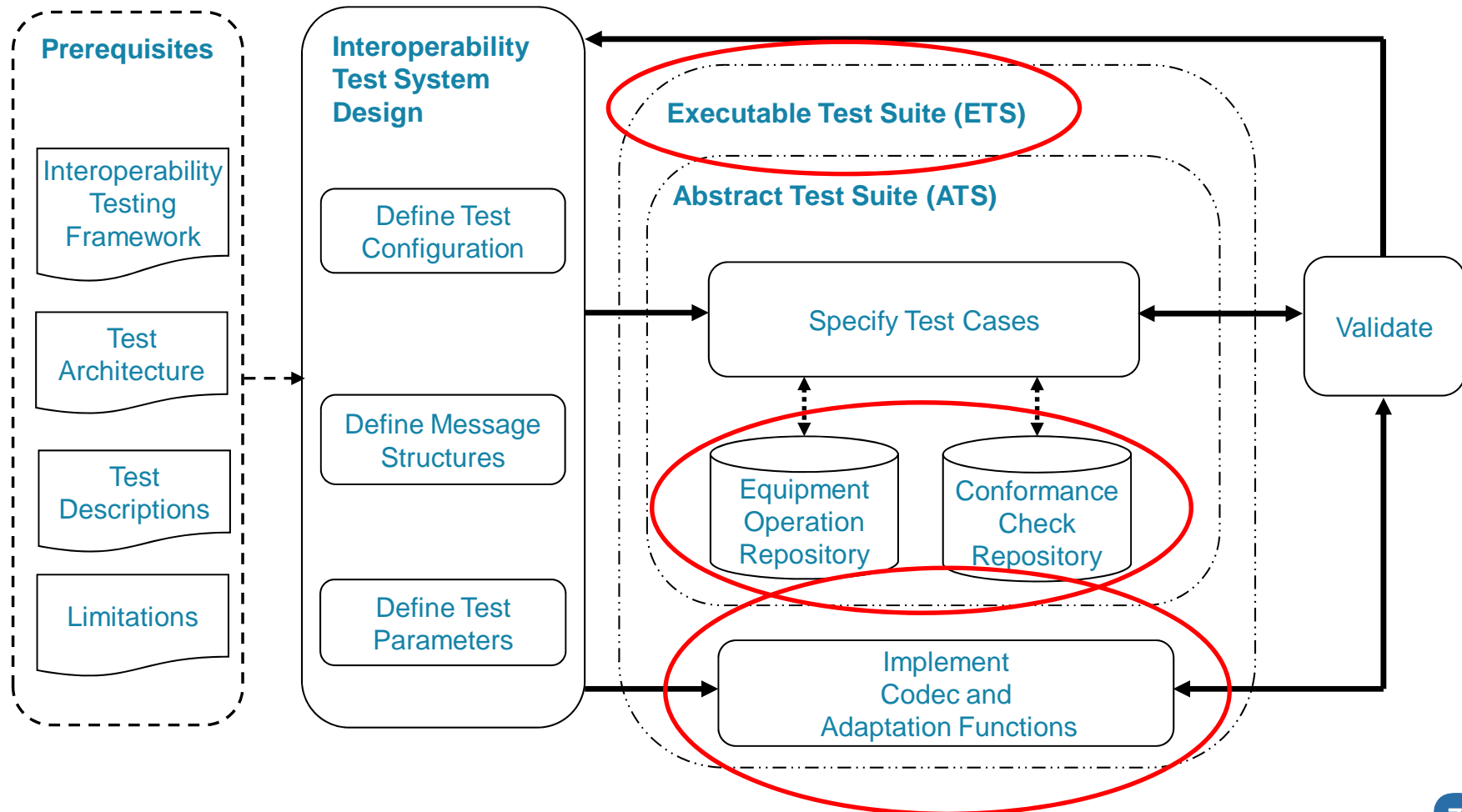
Tear down of test configuration

□ Re-establish initial state of the test environment

- Stop traffic capture in lower test adapter
- Disconnect and unmap test component ports
- Removes any selection requirements for next test to be executed

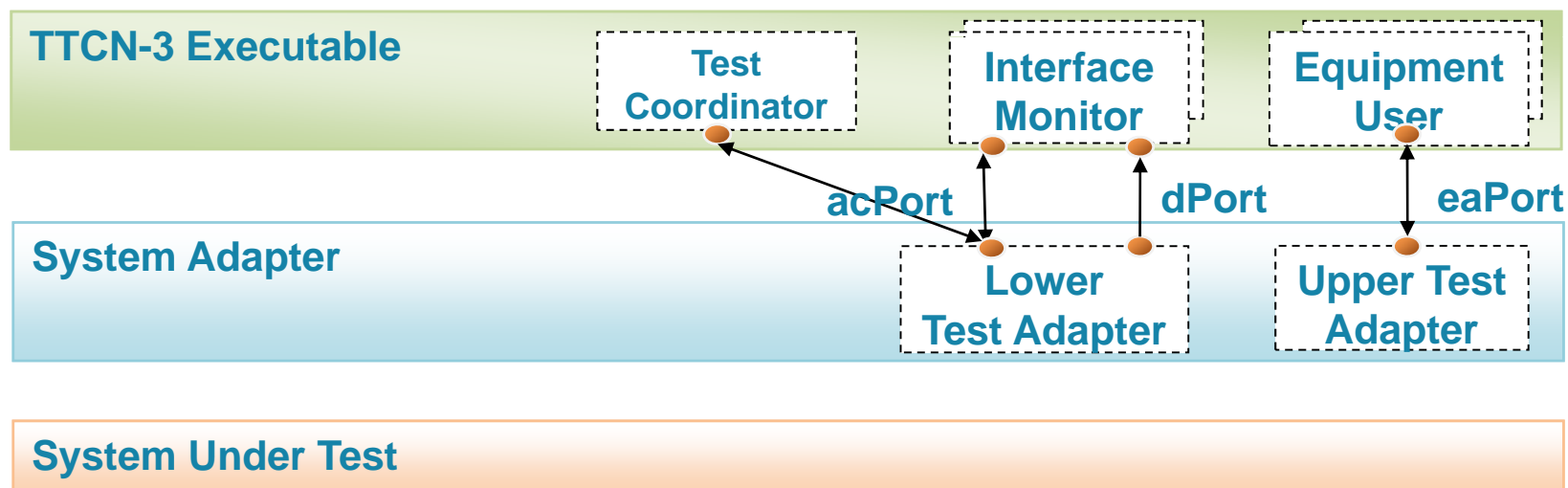


The Development of an Interoperability Test System in Practice – The Executable Test Suite (ETS)

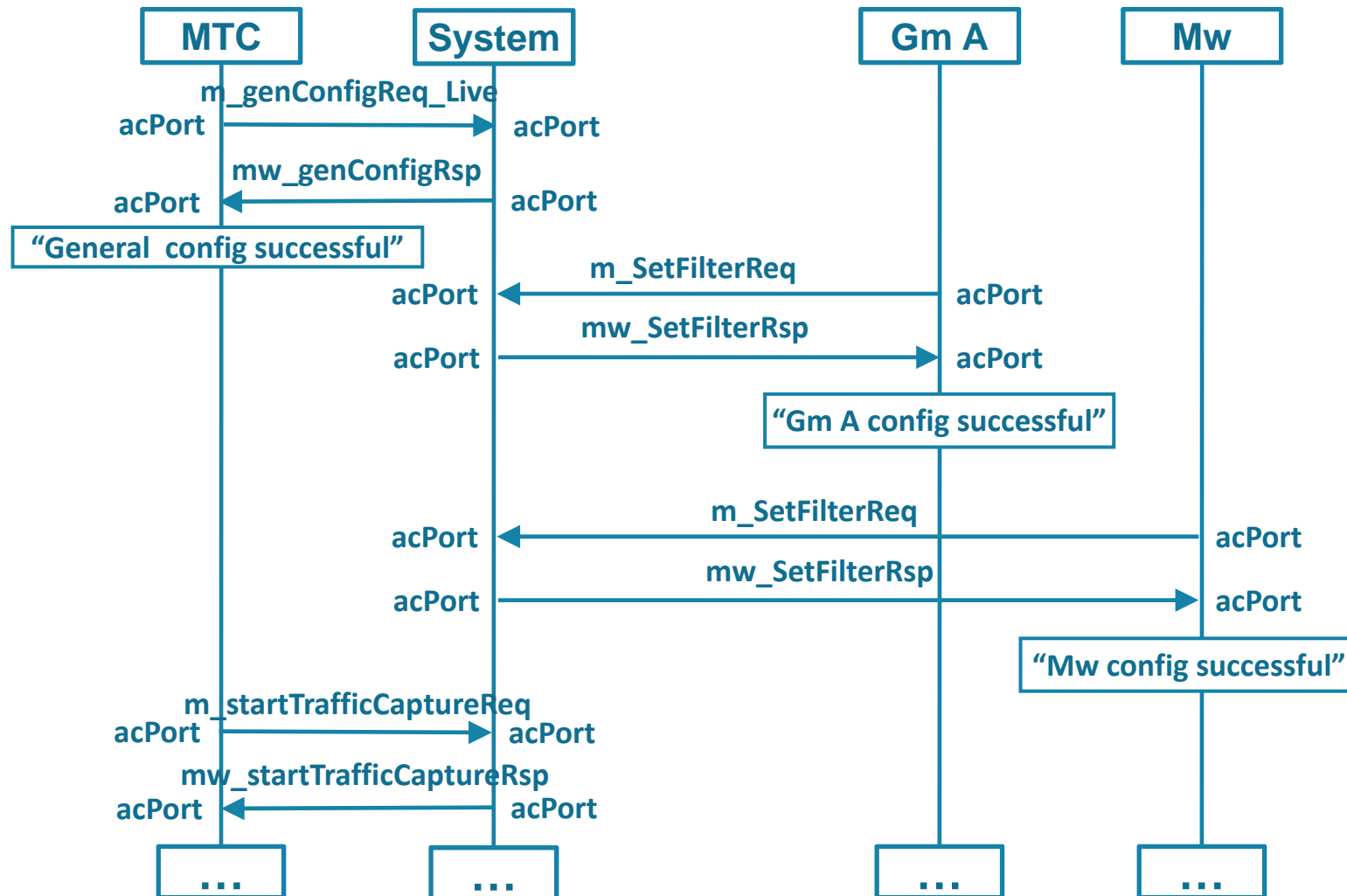


Executable Test Suite (ETS)

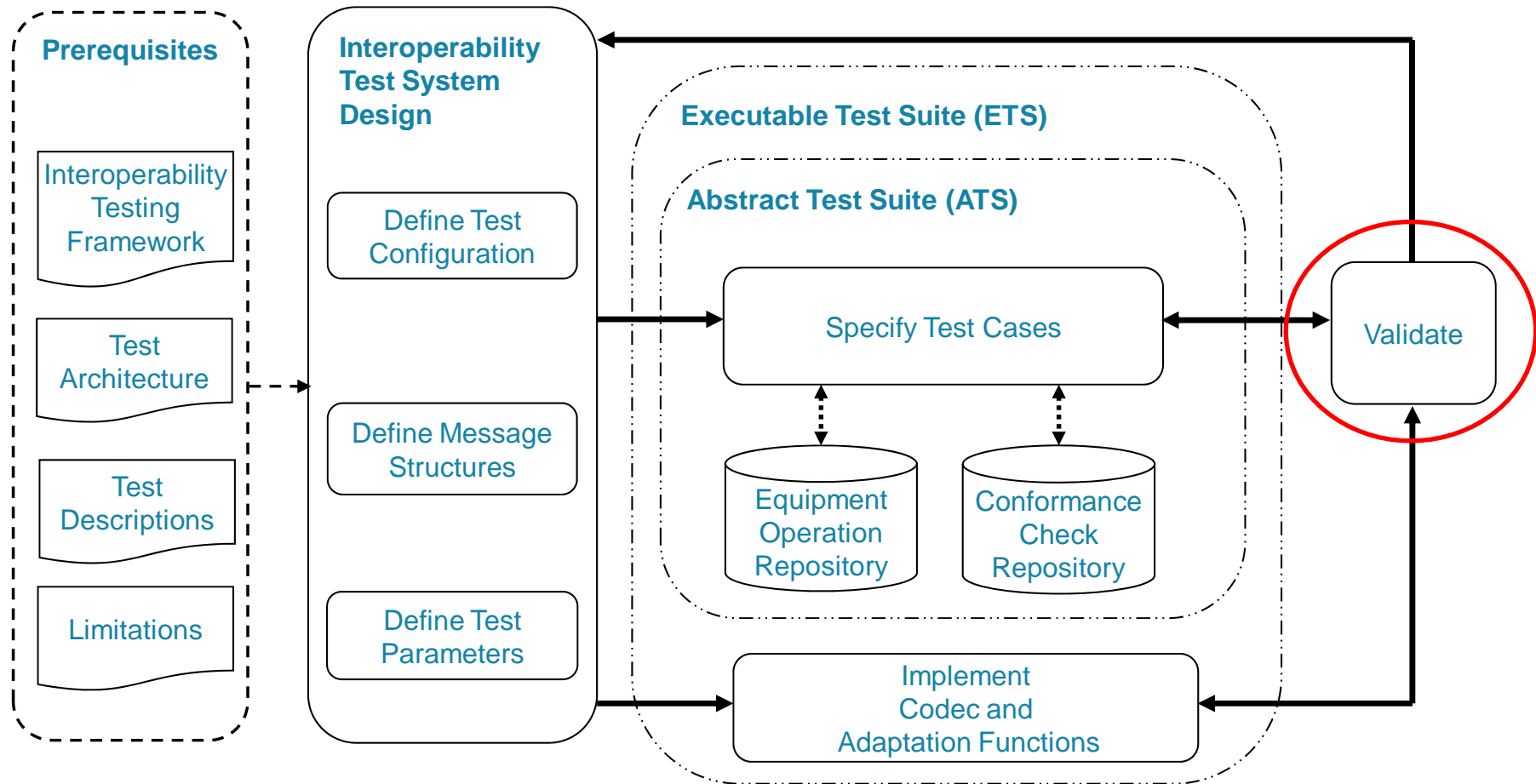
- ❑ Requires
 - Abstract Test Suite, Codecs, Adaptation functions
- ❑ The IMS interoperability test system follows the standardized architecture of a TTCN-3 test system
 - Codec for SIP and extensions required by IMS
 - Lower test adapter for capturing IP traffic
 - Upper test adapter for converting equipment operation requests into instructions for equipment operators



Example of Configuration of the Test Adapter



The Development of an Interoperability Test System in Practice – Validation of the Tests



Validation of the Test System

- ❑ Assures that the test system reproduces at its interface faithfully the behaviour specified for each test description
- ❑ Independent review against the test description
- ❑ Execution of the test system against a real SUT
 - Review of test results
- ❑ Validation results in improvements of the test specification and corrections of errors in the system adapter

Summary

- Methodology for automated interoperability testing of distributed systems**
- Process for the development of interoperability test systems**
- IMS Interoperability Test Suite has been applied successfully at the 3rd IMS Plugtest™ in 2009 and will be in future events**
- Application to other areas**
- Further details see the see references**

References

- [1] ETSI EG 202 568: "Methods for Testing and Specification (MTS); Internet Protocol Testing (IPT); Testing: Methodology and Framework"
- [2] ETSI ES 201 873-1: "Methods for Testing and Specification (MTS); The Testing and Test Control Notation version 3; Parts 1: TTCN-3 Core Language" (also published as ITU-T Recommendation series Z.140)
- [3] ETSI TS 123 228 (V7.15.0): "Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); LTE; IP Multimedia Subsystem (IMS); Stage 2 (3GPP TS 23.228 version 7.15.0 Release 7)"
- [4] M. Poikaselkä, G. Mayer, H. Khartabil, A. Niemi: "The IMS: IP Multimedia Concepts and Services", Wiley, 2004.
- [5] ETSI TS 102 351 (V2.1.1): "Methods for Testing and Specification (MTS); Internet Protocol Testing (IPT); IPv6 Testing: Methodology and Framework"
- [6] ETSI ES 201 873-6: "Methods for Testing and Specification (MTS); The Testing and Test Control Notation version 3; Part 6: TTCN-3 Control Interface (TCI)".
- [7] ETSI ES 201 873-5: "Methods for Testing and Specification (MTS); The Testing and Test Control Notation version 3; Part 5: TTCN-3 Runtime Interface (TRI)"
- [8] ETSI EG 202 810: "Methods for Testing and Specification (MTS); Automated Interoperability Testing; Methodology and Framework "
- [9] ETSI TR 102 788: "Methods for Testing and Specification (MTS); Automated Interoperability Testing; Specific Architectures "
- [10] ETSI TS 186 011-2: "Technical Committee for IMS Network Testing (INT); IMS NNI Interworking Test Specifications; Part 2: Test description for IMS NNI Interworking"



World Class Standards

Contact



Thomas Rings

Address

University of Göttingen

Institute of Computer Science

Goldschmidtstraße 7

37077 Göttingen

Email

rings@cs.uni-goettingen.de