

New TTCN-3 Test Adapter Framework with Capability Description

Yang LIU

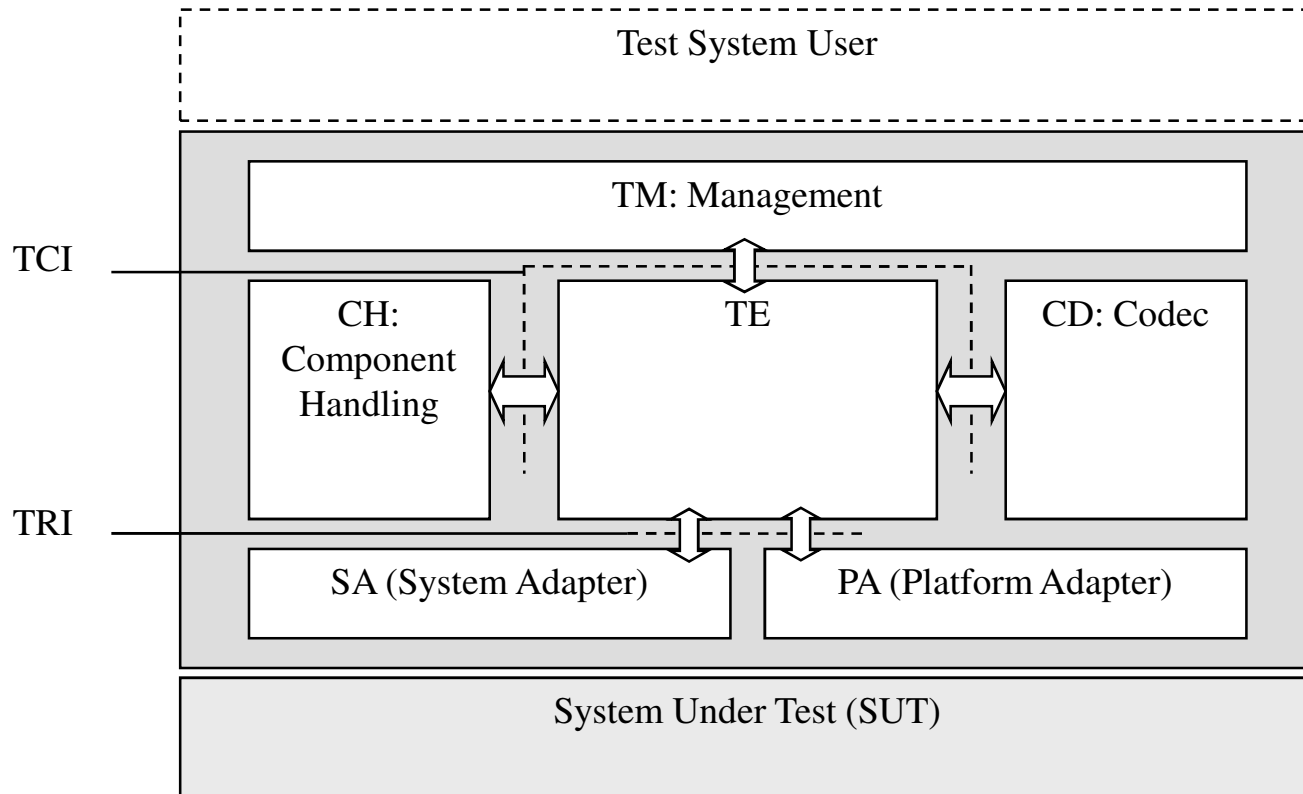
Bei Hang University

Outline

- Introduction
- The Problem
- The solution
- Test Adapter Framework
- Evaluation
- Conclusions

Introduction

The separation of test specification with test adapter



Outline

- Introduction
- **The Problem**
- The solution
- Test Adapter Framework
- Evaluation
- Conclusions

The Problem

- The dependencies among the abstract test case, CD and adapters (SA and PA) are obvious.
- There is no explicit semantic definition of the dependencies in TTCN-3 standard so far.

Outline

- Introduction
- The Problem
- **The solution**
- Test Adapter Framework
- Evaluation
- Conclusions

The solution

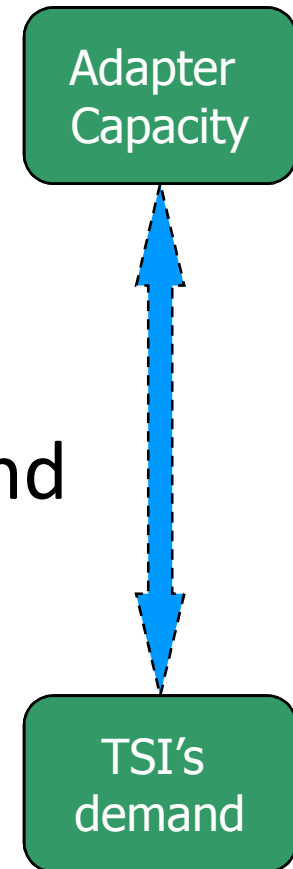
- Capability description language for specifying the dependencies among the abstract test case, CD, SA and PA
- A Test Adapter Framework
 - Automatic load the Adapter which comply with the requirements of test cases
 - Based on the description language
 - Transparent to the test system

Outline

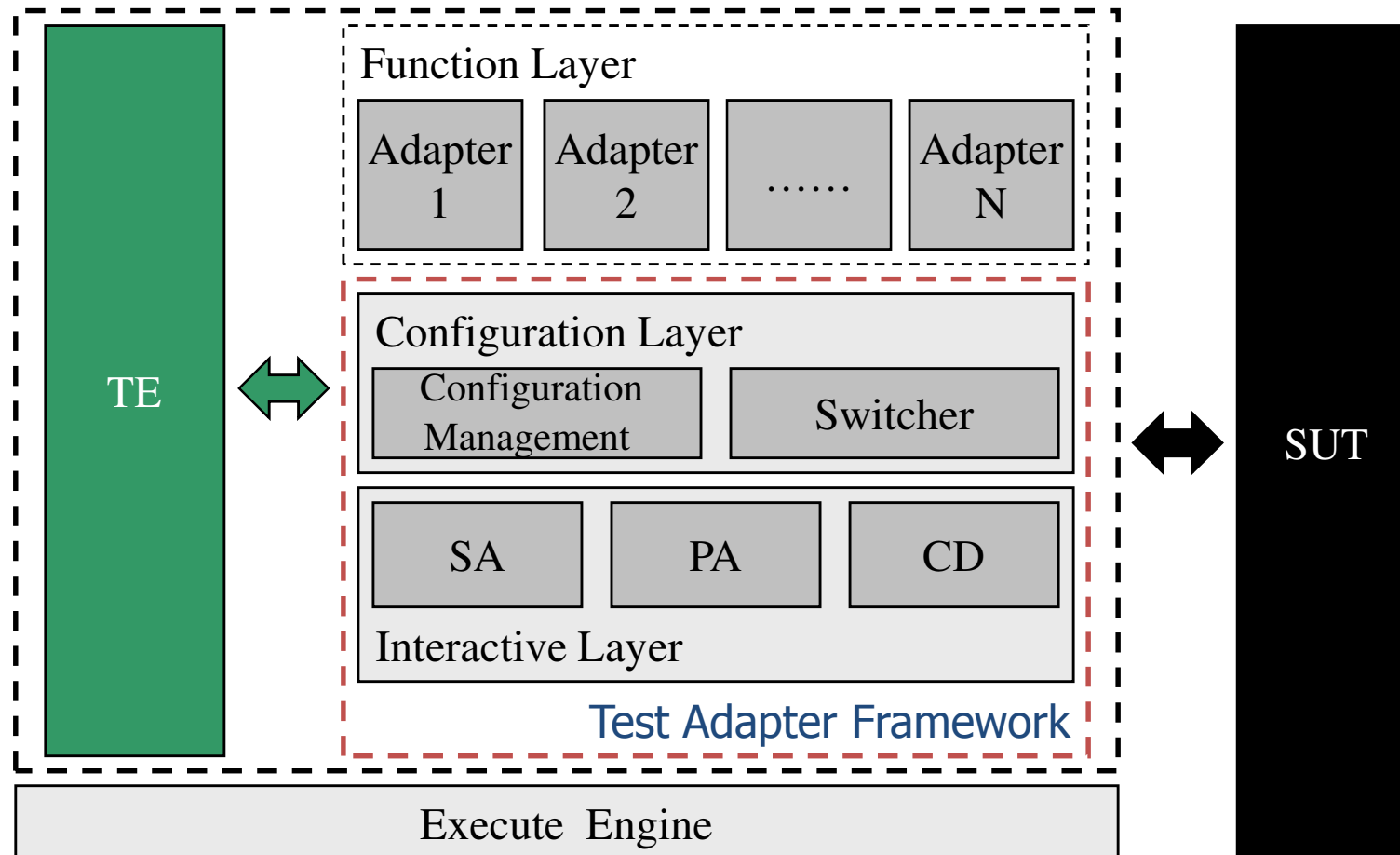
- Introduction
- The Problem
- The solution
- **Test Adapter Framework**
- Evaluation
- Conclusions

Mapping between the Adapter and TestCase

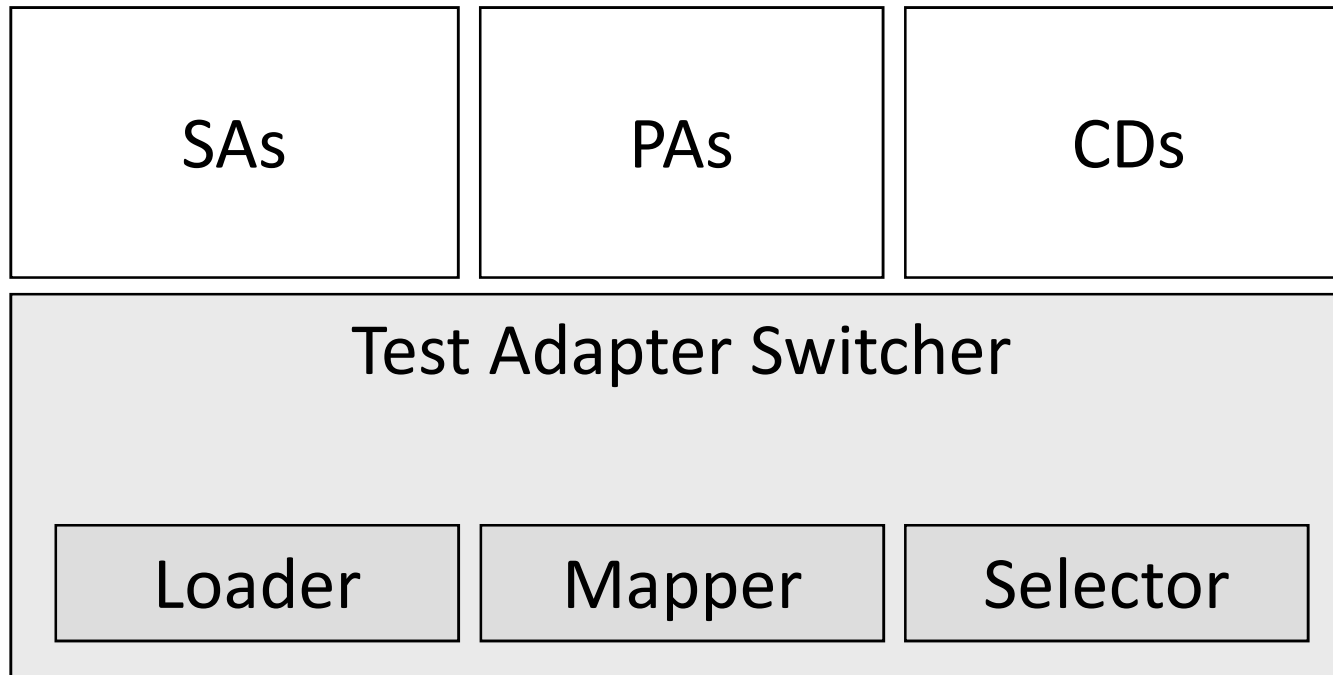
- Adapter Capacity
 - SA : Communication Adapter
 - PA : External method Adapter
 - CD (ECD) : Data Adapter
- Test System Interface(TSI)'s demand
 - Demand for communications capacity
 - Demand for External method
 - Input and output data types



Adapter Framework – Overall Architecture

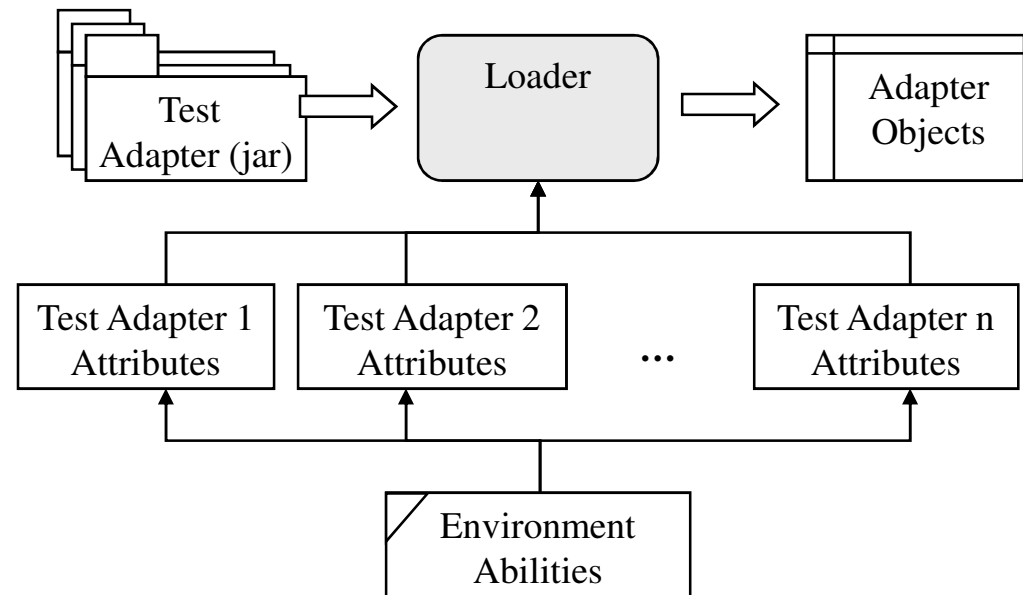


Adapter Framework – Switcher

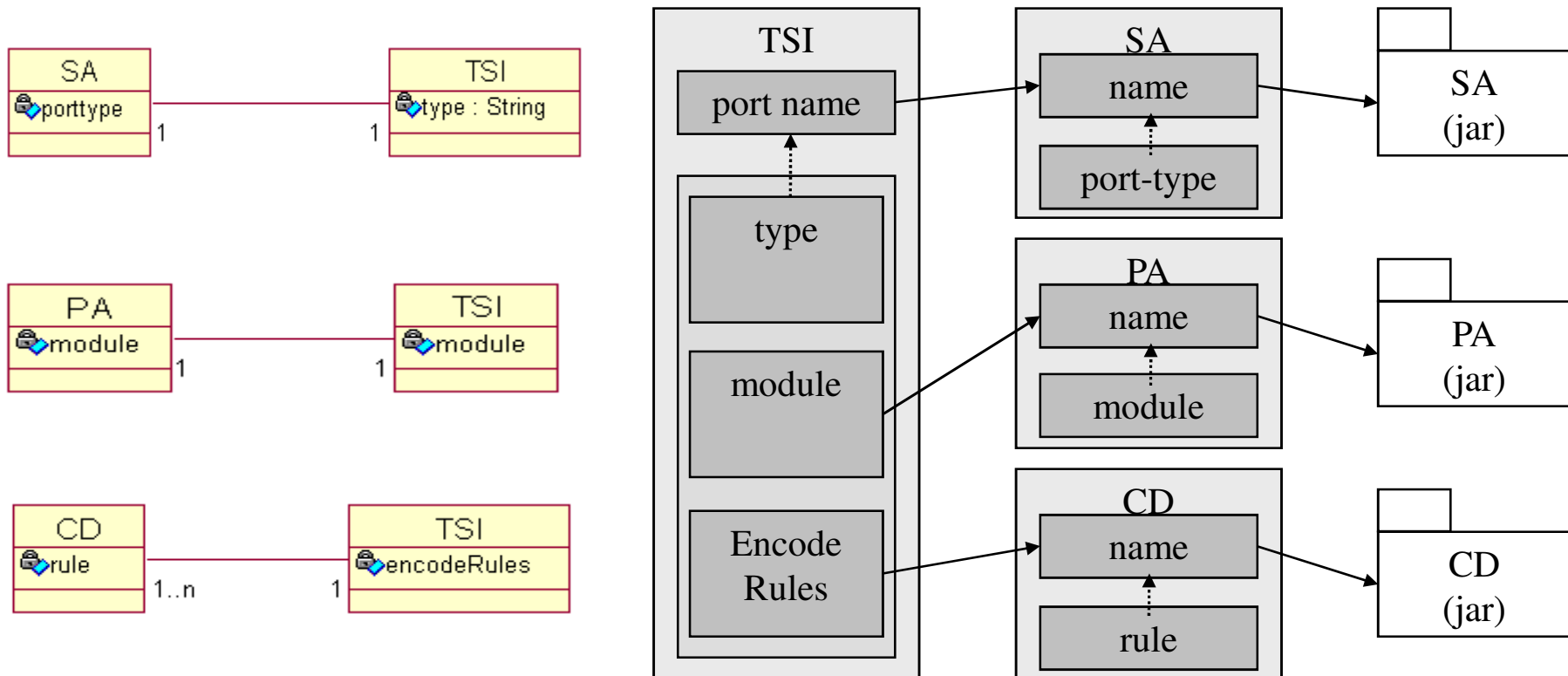


Adapter Framework - Loader

- Environment Description
 - OS
 - Environment Vars
- Test Adapter Description
 - SA
 - PA
 - CD



Adapter Framework - Mapper



example of capability matching

capability requirement
description of the test
case:

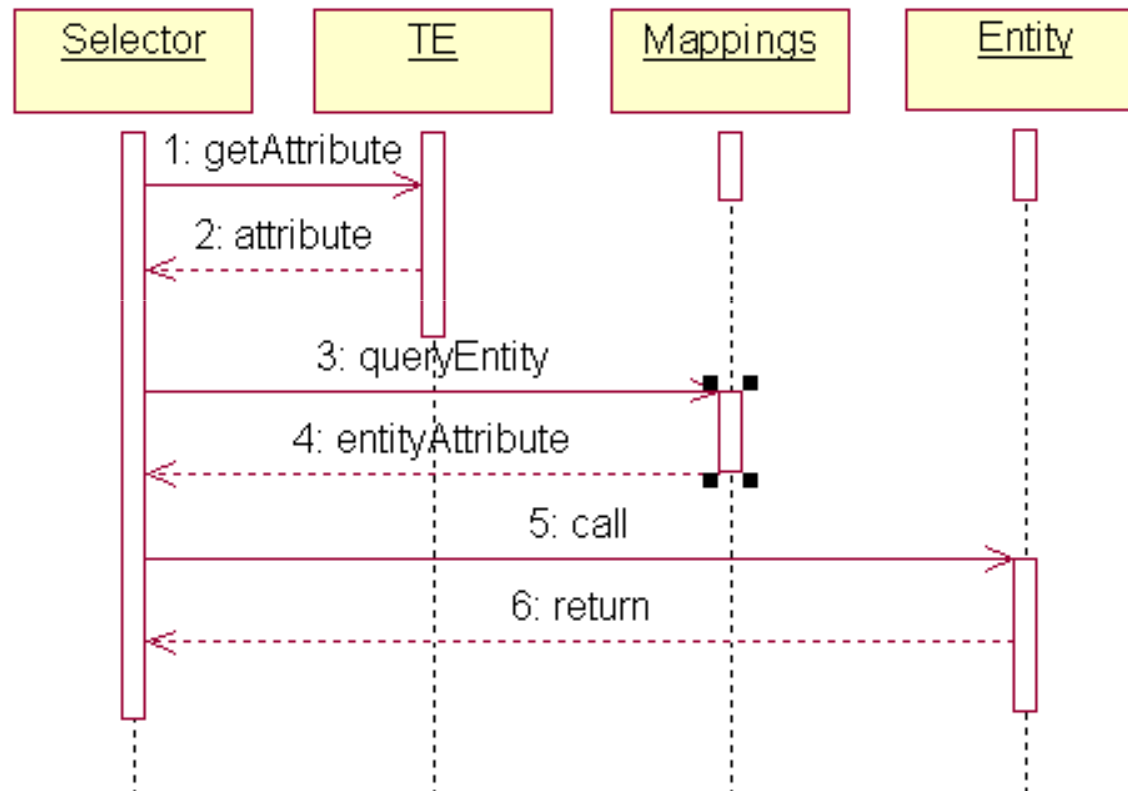
```
- <testcase Name="tc001" Module="TestCases">  
- <tsiPort>  
  <name>nameOfCA4thisTSI02</name>  
- <encodeRules>  
  <encodeRule name="encodeRuleName2" encode="encode">  
  </encodeRule>  
  <type>typeNameOfthisPort02</type>  
</tsiPort>  
</testcase>
```

capability provision
description of the CD & SA :

```
- <codec name="Codec_02">  
  <jarfile>cdJarFile2.jar</jarfile>  
  <class>Codec2.class</class>  
- <encoderule name="encodeRuleName2">  
  <encodemessagetype>MessageType2En</encodemessagetype>  
  <decodemessagetype>MessageType2De</decodemessagetype>  
</encoderule>  
- <lib>  
  <jarfile>LibFile4CD03.jar</jarfile>  
  <jarfile>LibFile4CD04.jar</jarfile>  
</lib>  
</codec>
```

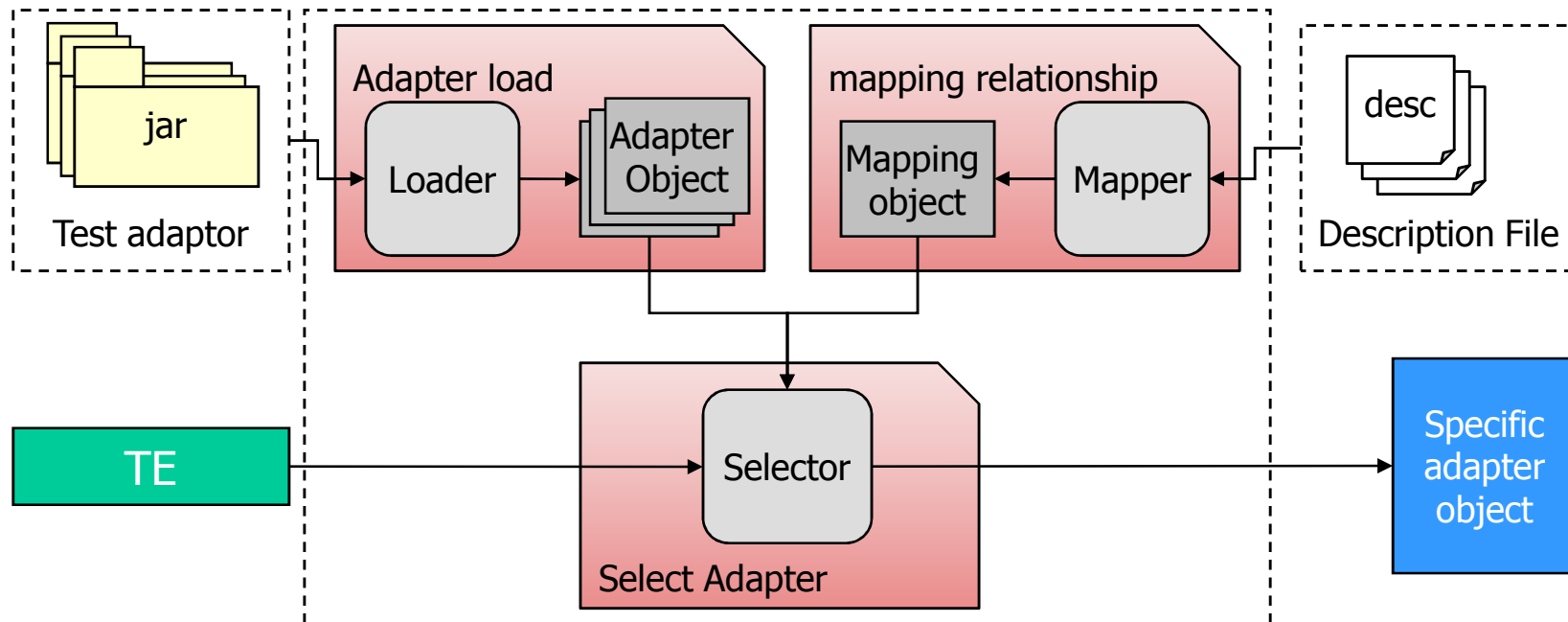
```
- <SA name="SA_02">  
  <jarfile>saJarFile2.jar</jarfile>  
  <class>saClass2.class</class>  
  <porttype>typeNameOfthisPort02</porttype>  
- <lib>  
  <jarfile>dependentJarFile.jar</jarfile>  
  <jarfile>dependentJarFile2.jar</jarfile>  
</lib>  
</SA>
```

Adapter Framework – Selector



Dynamic switcher

- Prerequisite :
 - TTCN-3 test script
 - SA , PA & CD
 - Test adaptor & TSI 's Description File

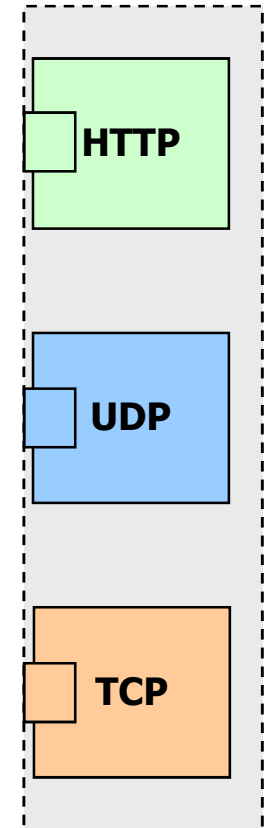


Outline

- Introduction
- The Problem
- The solution
- Test Adapter Framework
- **Evaluation**
- Conclusions

Evaluation

- Experiment:
 - SUT
 - Simulated heterogeneous distributed systems
 - Test system
 - SA and CD of Three test subsystems
 - test script
 - Support Tools
 - Compiler (TTthree)
 - Execution Engine (TTman)



Experiment with TAF

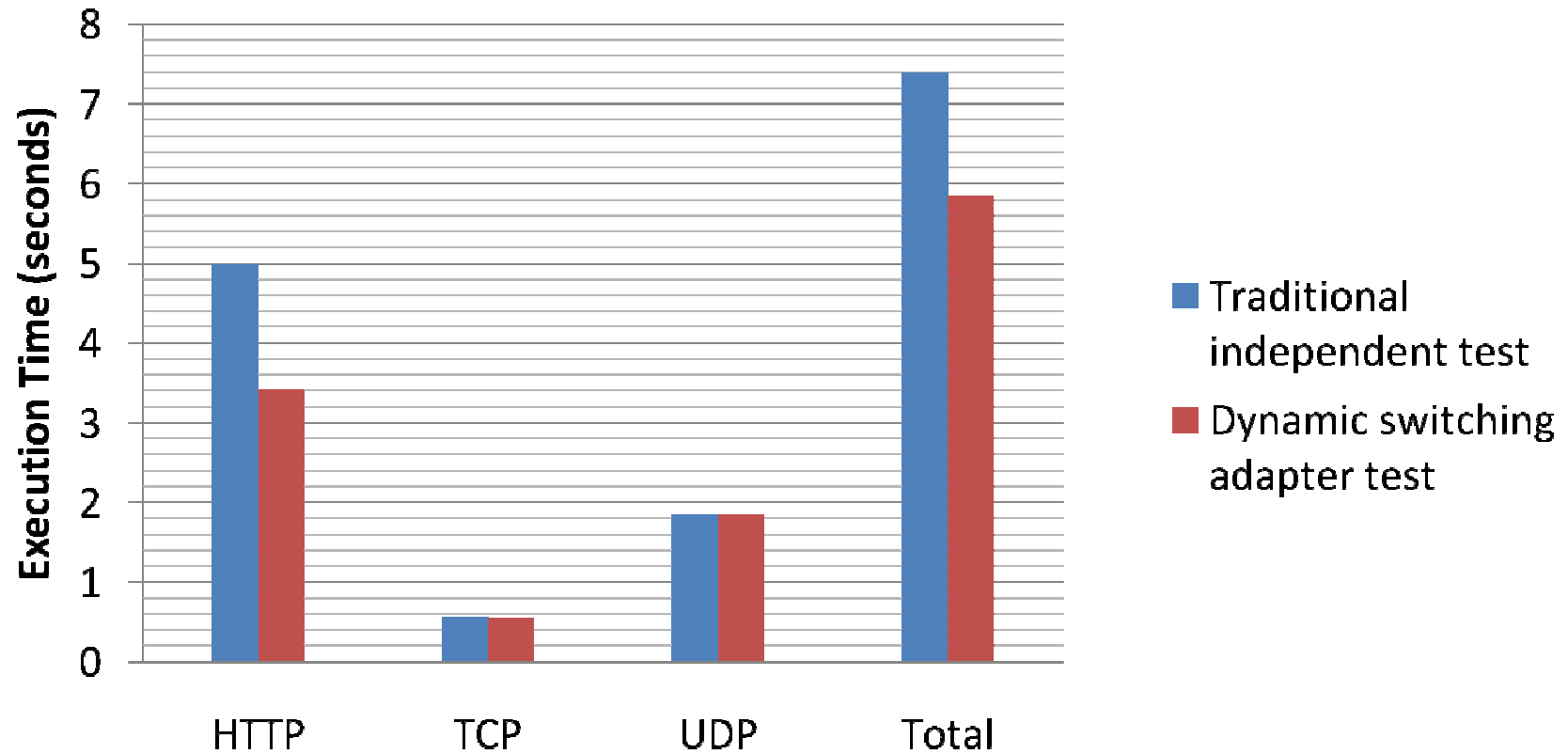
SUT	TCP	UDP	HTTP
module	TCPTest	UDPTest	HTTPTest
TSIPortName	tcpSystemPort	udpSystemPort	httpPortArray
type	TCP	UDP	HTTP
encodeRules	TCPCodec	UDPCodec	HTTPCodec
SA	TCP_TA.jar /TCPTestAdapter	UDP_TA.jar /UDPTestAdapter	HTTP_TA.jar /HTTPTestAdapter
CD	TCP_Codec.jar /TCPCodec	UDP_Codec.jar /UDPCodec	HTTP_Codec.jar /HTTPCodec

TAF –Test Result

```
21:09:38.781: parsing given loader file 'main.clf'
21:09:39.031: loading test adapter 'buaa.sei.mdt.ttcn3.framework.RealFramework'
21:09:39.062: starting test cases...
21:09:39.062: starting test case 'http_tc' -> 1
HTTPTestAdapter: Request <to: /127.0.0.1>
-----
http://localhost:8080/dinolist.xml
HTTPTestAdapter: Response
-----
File size: 3192
Warning: validation was turned on but an org.xml.sax.ErrorHandler was not set, which is probably not what is desired. Parser will use a default ErrorHandler to print the first 10 errors. Please call the setErrorHandler method to fix this
-
Error: URI = "null", Line = "42", : Document root element "dinolist", must match DOCTYPE root "null".
Error: URI = "null", Line = "42", : Document is invalid: no grammar found.
21:09:39.328: http_tc -> pass
21:09:39.343: starting test case 'tcp_tc' -> 2
MyTestAdapter: Sending <to: /127.0.0.1>
-----
0103ICP
MyTestAdapter: Received < 7 chars>
-----
0103ICP
21:09:39.406: tcp_tc -> pass
21:09:39.406: starting test case 'udp_tc' -> 3
UDPTestAdapter: Sending <to: /127.0.0.1>
-----
0x00 0x01 0x00 0x04
0x10 0x10 0x10 0x10
UDPTestAdapter: Received < 8 bytes>
-----
0x00 0x01 0x00 0x04
0x10 0x10 0x10 0x10
21:09:39.968: udp_tc -> pass

Total number of executed test cases: 3
none:    0
pass:    3
inconc:  0
fail:    0
error:   0
```

Performance comparison



Outline

- Introduction
- The Problem
- The solution
- Test Adapter Framework
- Evaluation
- **Conclusions**

Conclusions

- Why we design the test adapter framework
- Existing problems
 - Test Adapter Framework configuration is complex
 - The efficiency of loading the modules which will be dynamic switching is low

Conclusions

- Future work
 - Raise the ability to match to the semantic level
 - Use a general language of Capability Description
 - Set the general test adapter Standard

THANK YOU