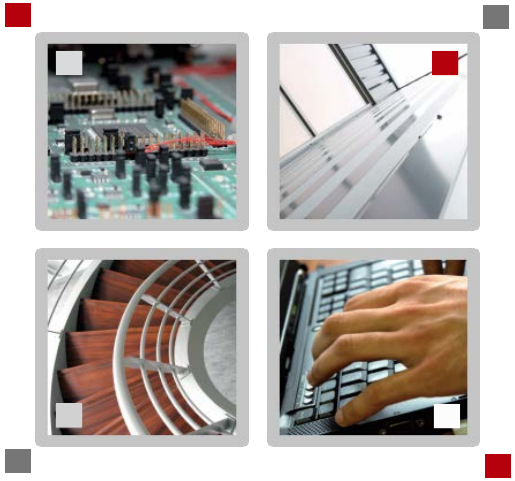
Interoperability tests for components based architectures.



22 May 2018

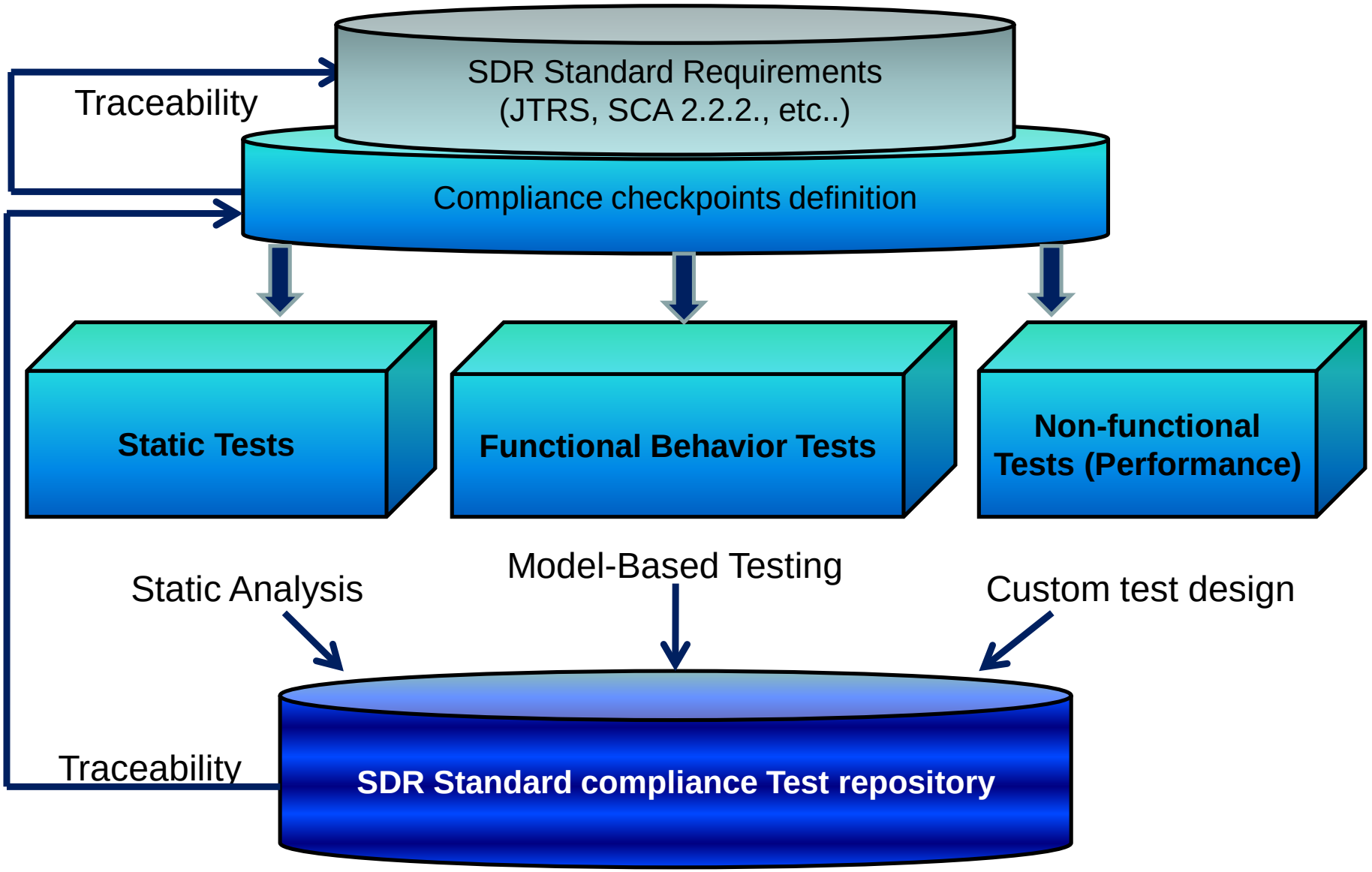
Agenda

- Test Strategy
  - Objectives
  - Overview
  - Test Design Process
  - Compliance checkpoints definition
  - Modeling
  - Test generation
- Test Bench modularity & automation
  - Objectives
  - Overview
  - Test execution software capabilities.
  - Real implementation on JTRS Audio Device
- Test Bench Dataflow
  - Example on startTone() function test.

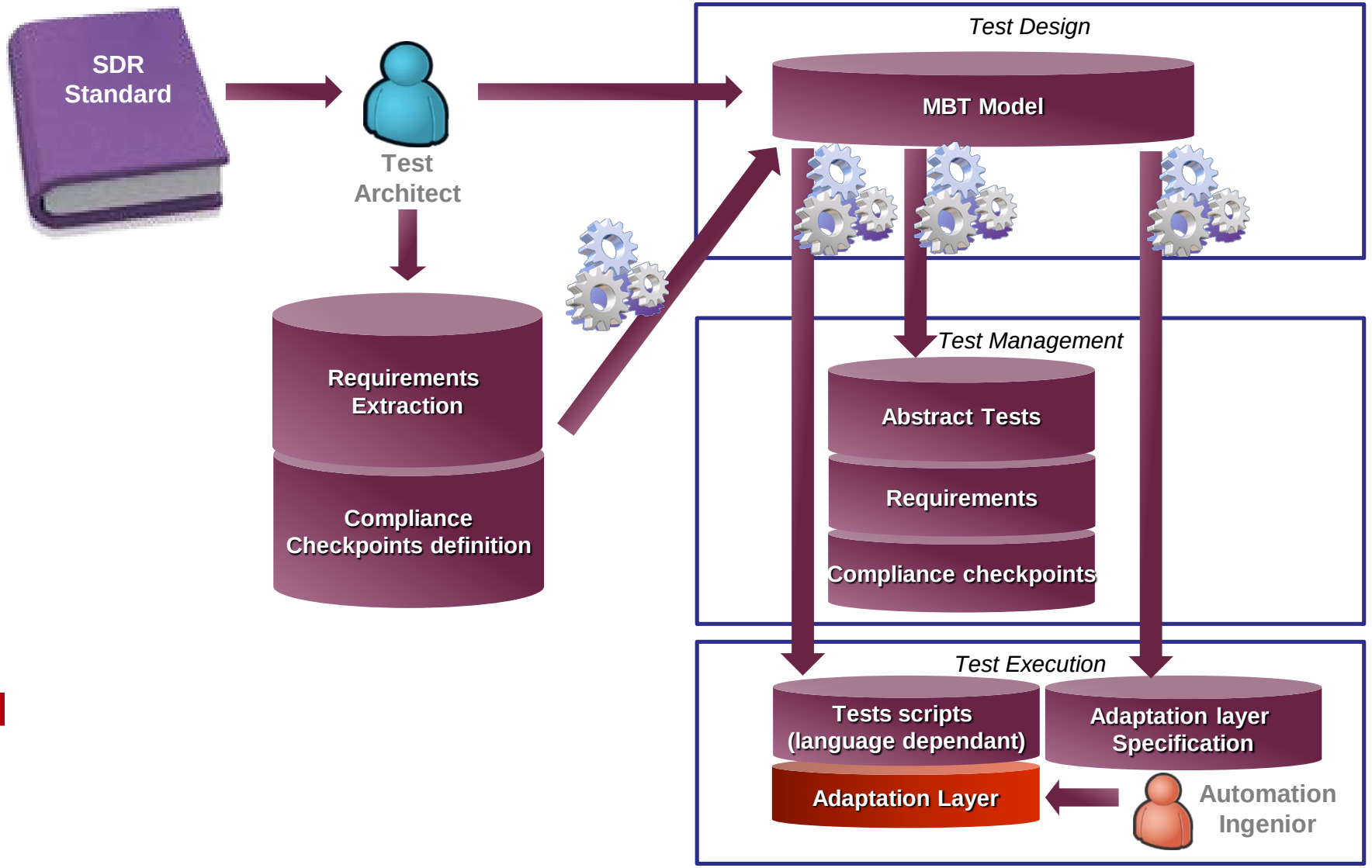
Test Strategy
Objectives

- Generic approach for components based architectures by interfaces testing.
- Test design based on the behavior of the system under Test (Model Based Testing).
- Tests should be exported into several formats and programming languages.

Test Strategy
Overview



Test Strategy
Functional Test design Process



Test Strategy
Compliance checkpoints definition

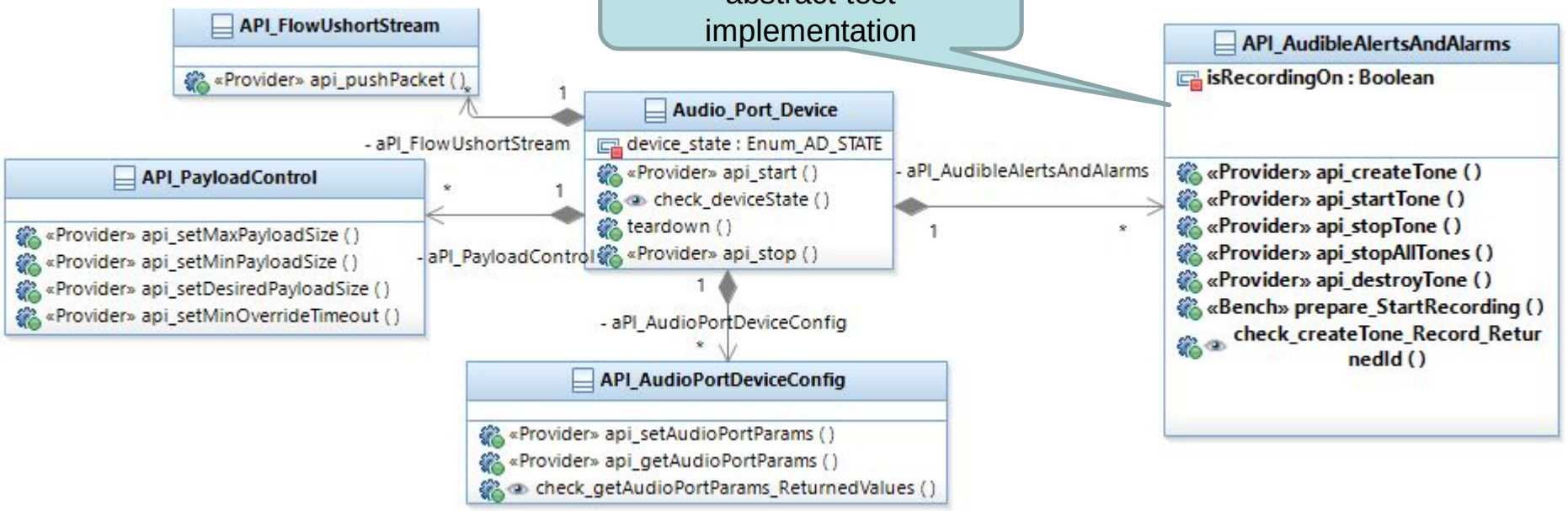
- **Compliance checkpoint defines the test objectives**
  - **Success case(s) or Error case(s) definition**
  - **Definition of test success criteria.**
  - **Definition of the applicability of the test.**

Sample on the startTone function of Audio Device

Requirement Identifier	Requirement Text	PDC Identifier	PDC Applicab	Compo nent	PDC Description
JTRS_AD_PROVIDE_START_TONE	The startTone operation provides the user the ability to start the generation of a previously created tone/beep to the device user. - Synopsis: void startTone(in unsigned short toneId) raises(InvalidToneId); - Return Value: None - State: ENABLED CF::Device::operationalState. - Originator: Service User	-	-	-	-
JTRS_AD_PROVIDE_START_TONE		JTRS_AD_PROVIDE_ST ART_TONE_SUCCESS_0 01	Platform	GPP	* Success case * the tone or beep identification number is valid * Check the tone is started
JTRS_AD_PROVIDE_START_TONE_EXCEPTIO N_InvalidToneId	InvalidToneId (see A.5.3.2) A CORBA exception is raised when the tone/beep identification number is invalid.	-	-	-	-
JTRS_AD_PROVIDE_START_TONE_EXCEPTIO N_InvalidToneId		JTRS_AD_PROVIDE_ST ART_TONE_EXCEPTIO N_InvalidToneId_001	Platform	GPP	* Check an exception: InvalidToneId is raised * Not existing Tone Id

Test Strategy
Modeling

UML Class diagrams for abstract test implementation



Functions to call on set up before the test body

Test body

Functions to call on Tear down to return to initial state

Test edition

```
aPI_AudioPortDeviceConfig.api_setAudioPortParams(Enum_Audio_Params_Valid_1)
aPI_AudibleAlertsAndAlarms.api_createTone(Enum_Audio_Channel_2, Enum_Tone_Profile_Multi_Tone_Valid_withOneTone_1)
aPI_AudibleAlertsAndAlarms.prepare_StartRecording()
aPI_ChannelAudioConfig.api_getOutputGain(Enum_Audio_Channel_2)
aPI_ChannelAudioConfig.api_setOutputGain(Enum_Audio_Channel_2, Enum_Output_Gain_Valid_1)
common._body()
aPI_AudibleAlertsAndAlarms.api_startTone(Enum_Audio_Channel_2, Enum_Tone_Id_1)
aPI_ChannelAudioConfig.api_setOutputGain(Enum_Audio_Channel_2, Enum_Output_Gain_Default)
aPI_AudibleAlertsAndAlarms.api_destroyTone(Enum_Audio_Channel_2, Enum_Tone_Id_1)
aPI_AudibleAlertsAndAlarms.api_stopAllTones(Enum_Audio_Channel_2)
```

Test Strategy
Test generation: Abstract Tests

- **Example of C++ test with the startTone function of JTRS Audio Device**
  - Each generated function is a single test step.
  - Each test will be an assembly of single steps.

```
bool JTRS_AD_PROVIDE_START_TONE_1::setUp()
{
current_result = m_adapter->api_setAudioPortParams(<params>);
current_result = m_adapter->api_getOutputGain(<params>);
current_result = m_adapter->check_getOutputGain_Record_DefaultOutputGain(<params>);
current_result = m_adapter->prepare_StartRecording(<params>);
current_result = m_adapter->api_createTone(<params>);
current_result = m_adapter->check_createTone_Record_ReturnedId(<params>);
return current_result;
}

bool JTRS_AD_PROVIDE_START_TONE_1::test()
{
current_result = m_adapter->api_startTone(<params>);
current_result = m_adapter->check_ToneStatusForChannel(<params>);
return current_result;
}

bool JTRS_AD_PROVIDE_START_TONE_1::tearDown()
{
current_result = m_adapter->api_setOutputGain(<params>);
current_result = m_adapter->api_destroyTone(<params>);
current_result = m_adapter->api_stopAllTones(<params>);
current_result = m_adapter->bench_tearDown();
return current_result;
}
```

API — Call SUT interface

prepare — Prepare measurement tools

Check — Compare received value with expected value

Bench — Specific actions on Test Bench

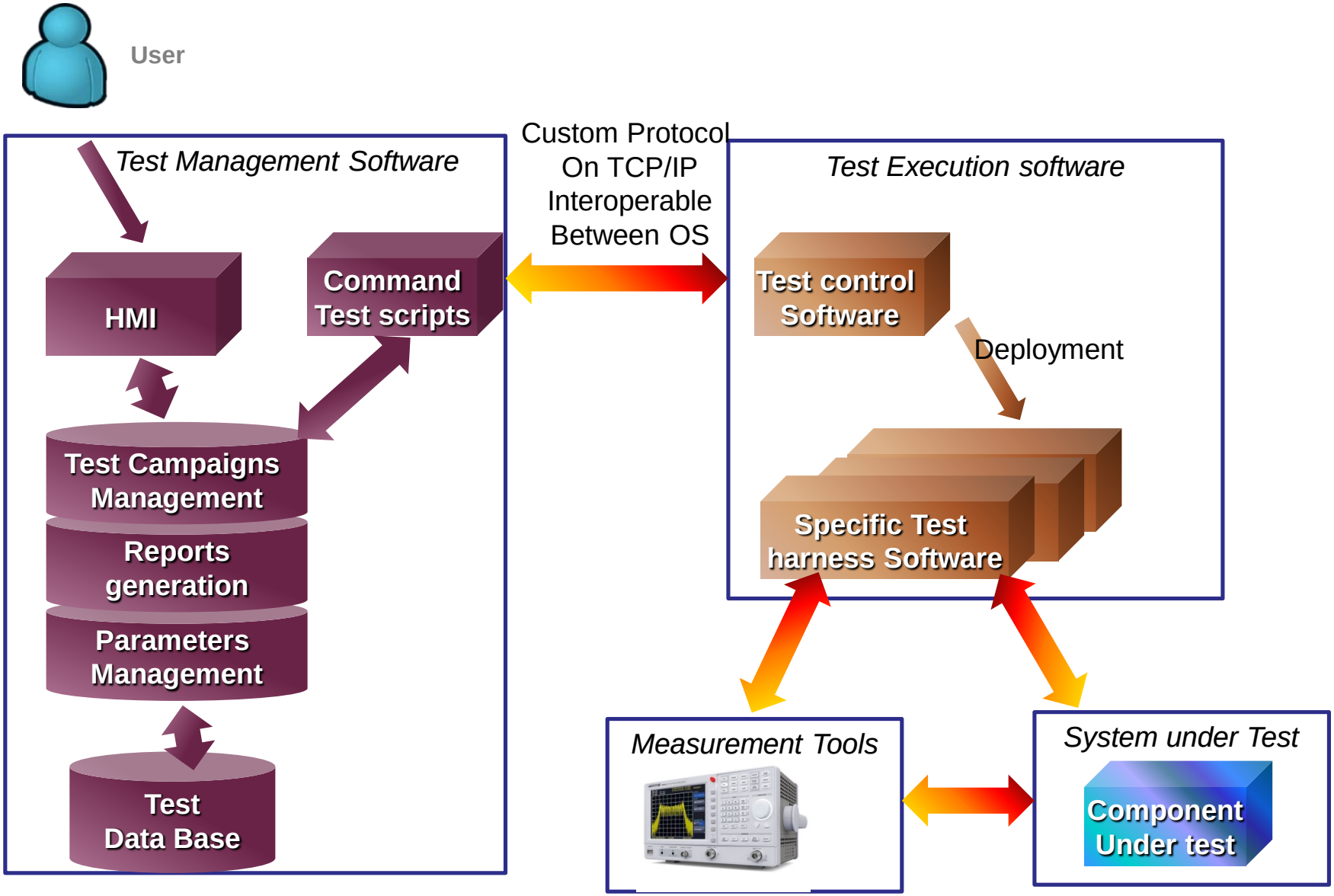
Agenda

- Test Strategy
  - Objectives
  - Overview
  - Test Design Process
  - Compliance checkpoints definition
  - Modeling
  - Test generation
- Test Bench modularity & automation
  - Objectives
  - Overview
  - Test execution software capabilities.
  - Real implementation on JTRS Audio Device
- Test Bench Dataflow
  - Example on startTone() function test.

Test Bench modularity & Automation
Objectives

- Maximum automation of test campaigns
- In the same spirit automatic management of measurement tools
- No dependency between Test management software and test execution software.
- Portability of test execution software to address several systems under test.

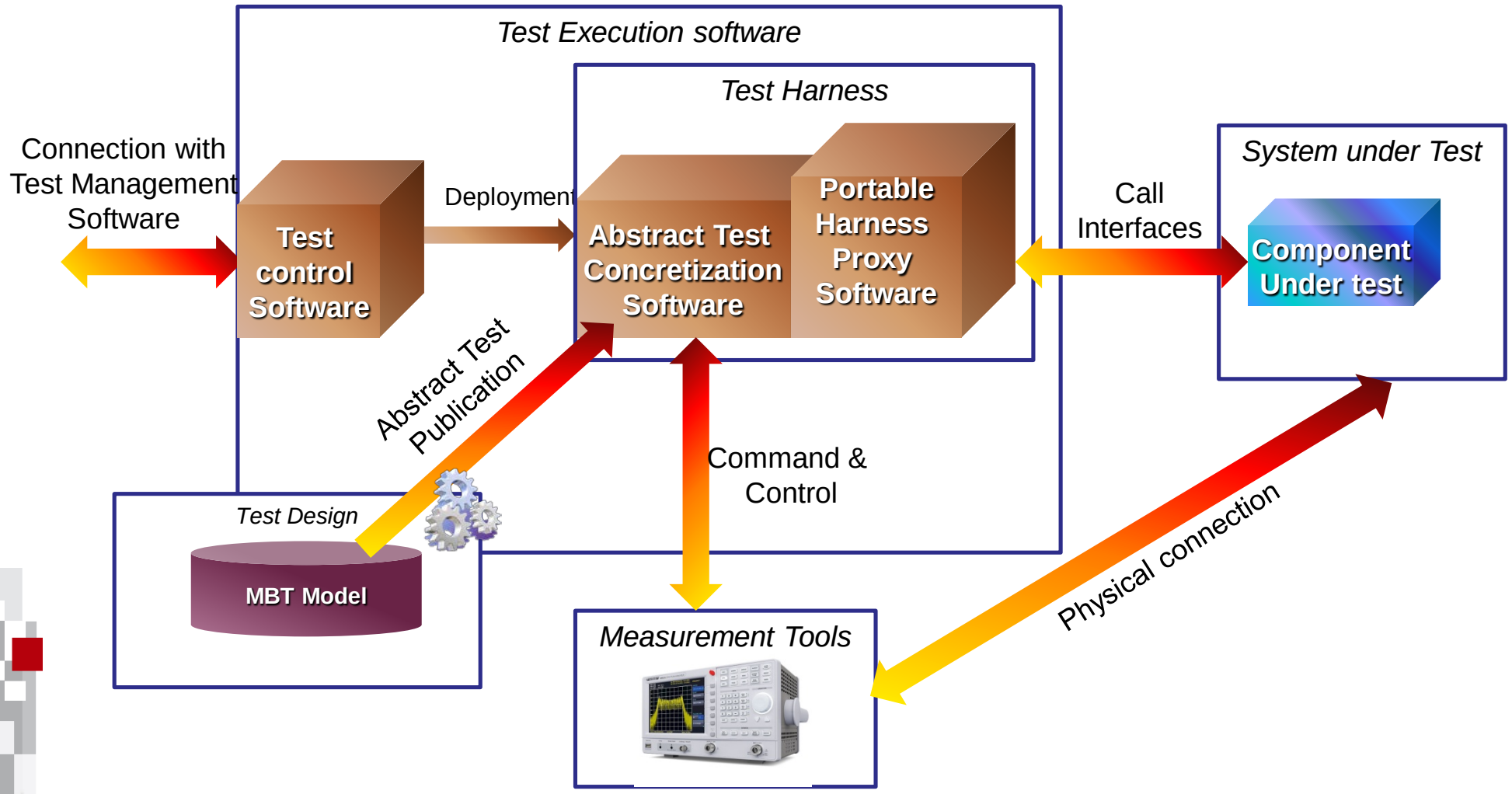
Test Bench modularity & Automation
Overview



Test Bench modularity & Automation

Test execution software capabilities.

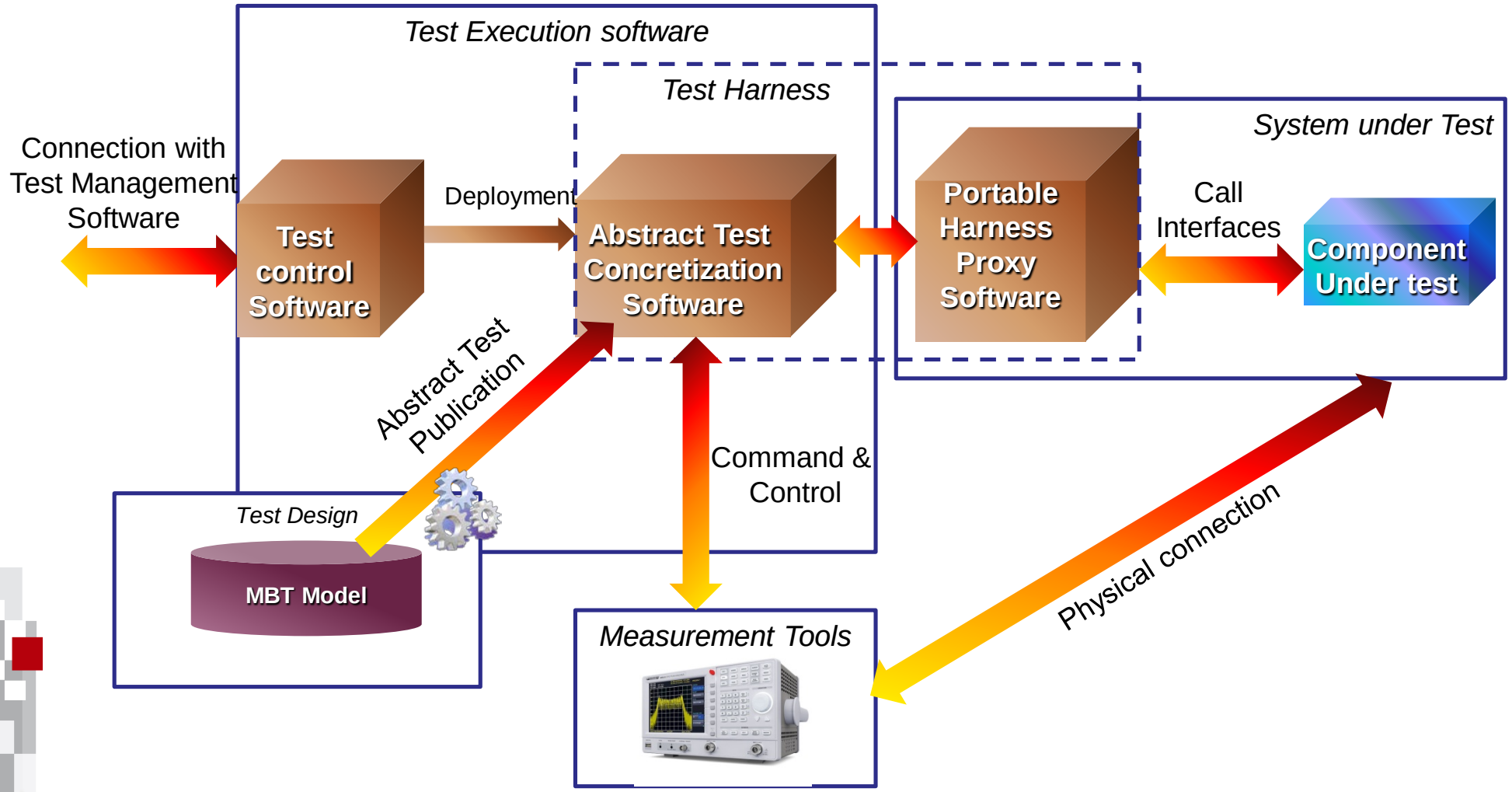
- **Case n°1: Interfaces of the component under test are reachable from outside (CORBA bus available for example)**



Test Bench modularity & Automation

Test execution software capabilities.

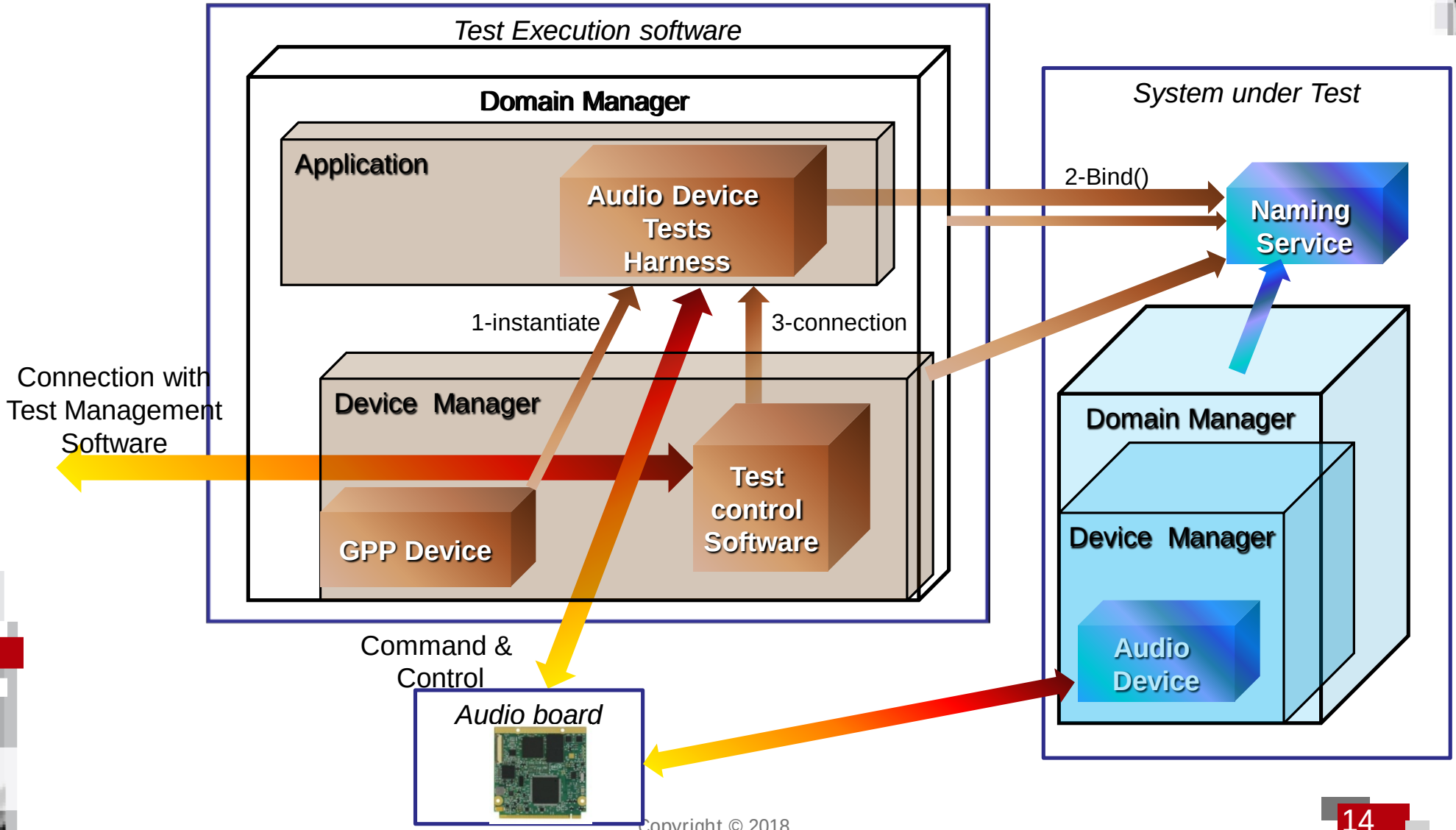
- **Case n°2: Interfaces of the component under test are not reachable from outside**



Test Bench modularity & Automation

Real implementation on JTRS Audio Device

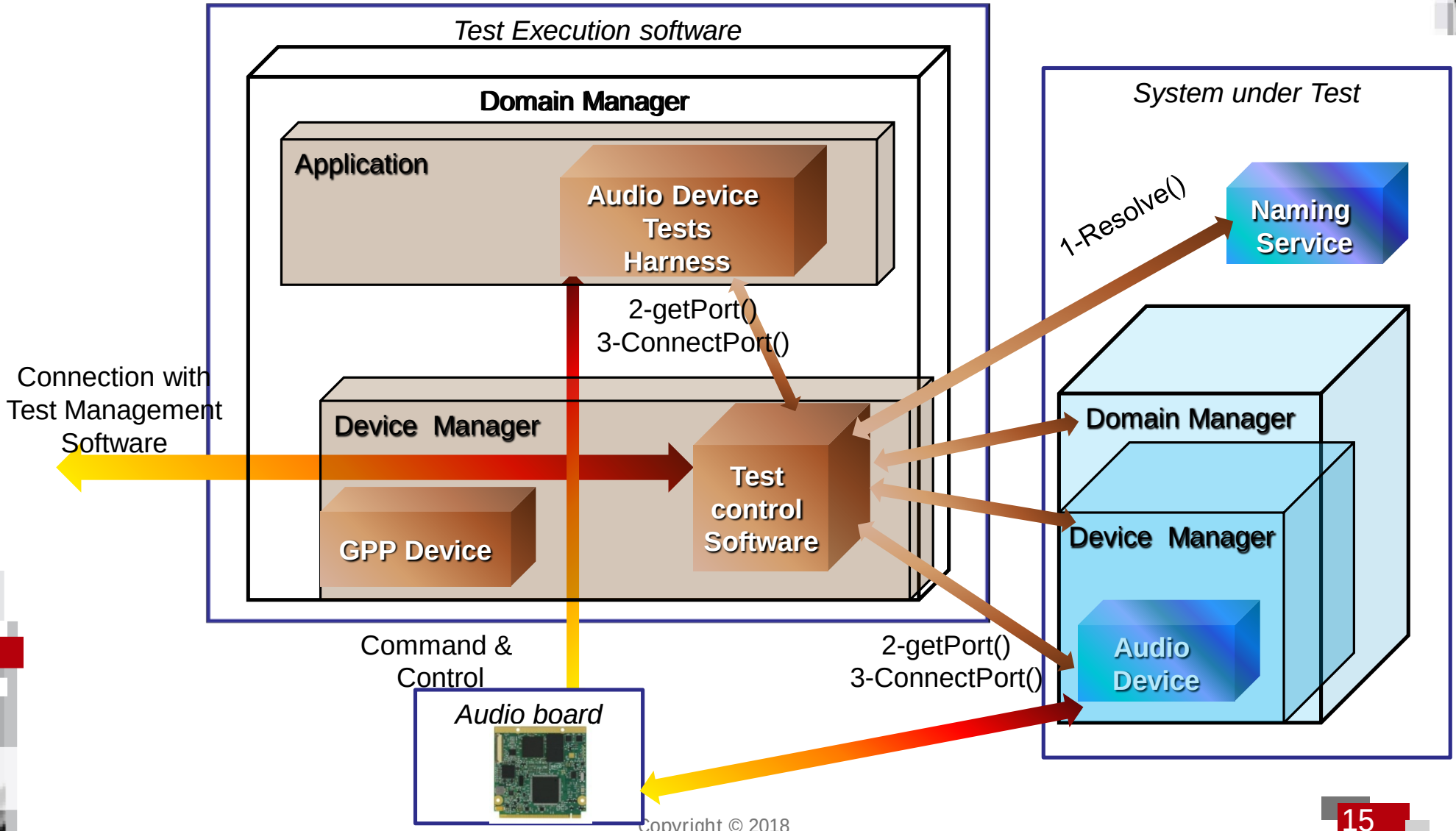
- **Deployment phase by using SCA 2.2.2.**



Test Bench modularity & Automation

Real implementation on JTRS Audio Device

- **Connection phase using SCA 2.2.2. capabilities**



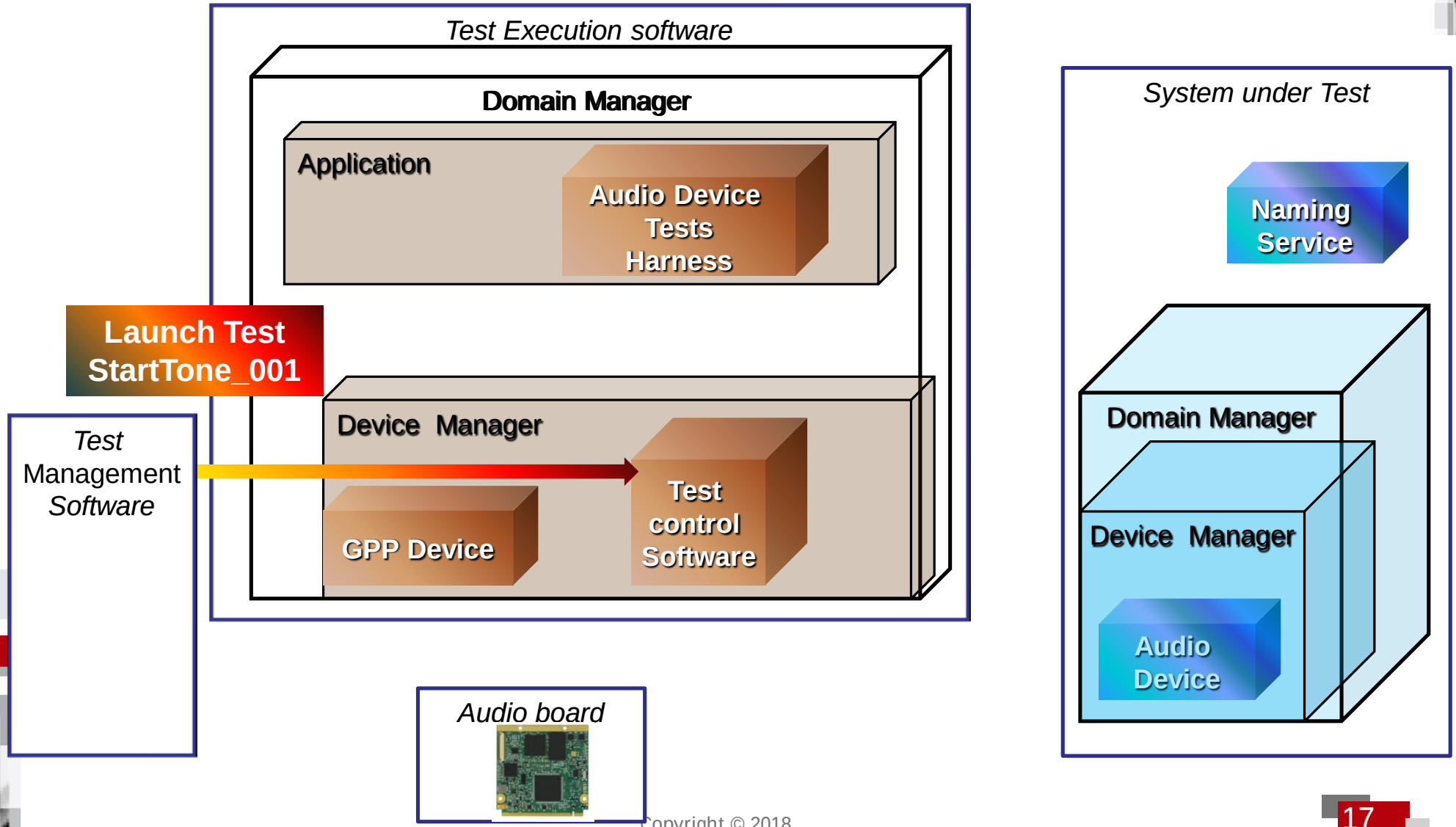
Agenda

- Test Strategy
  - Objectives
  - Overview
  - Test Design Process
  - Compliance checkpoints definition
  - Modeling
  - Test generation
- Test Bench modularity & automation
  - Objectives
  - Overview
  - Test execution software capabilities.
  - Real implementation on JTRS Audio Device
- Test Bench Dataflow
  - Example on startTone() function test.

Test Bench Data flow

Example on startTone() function test.

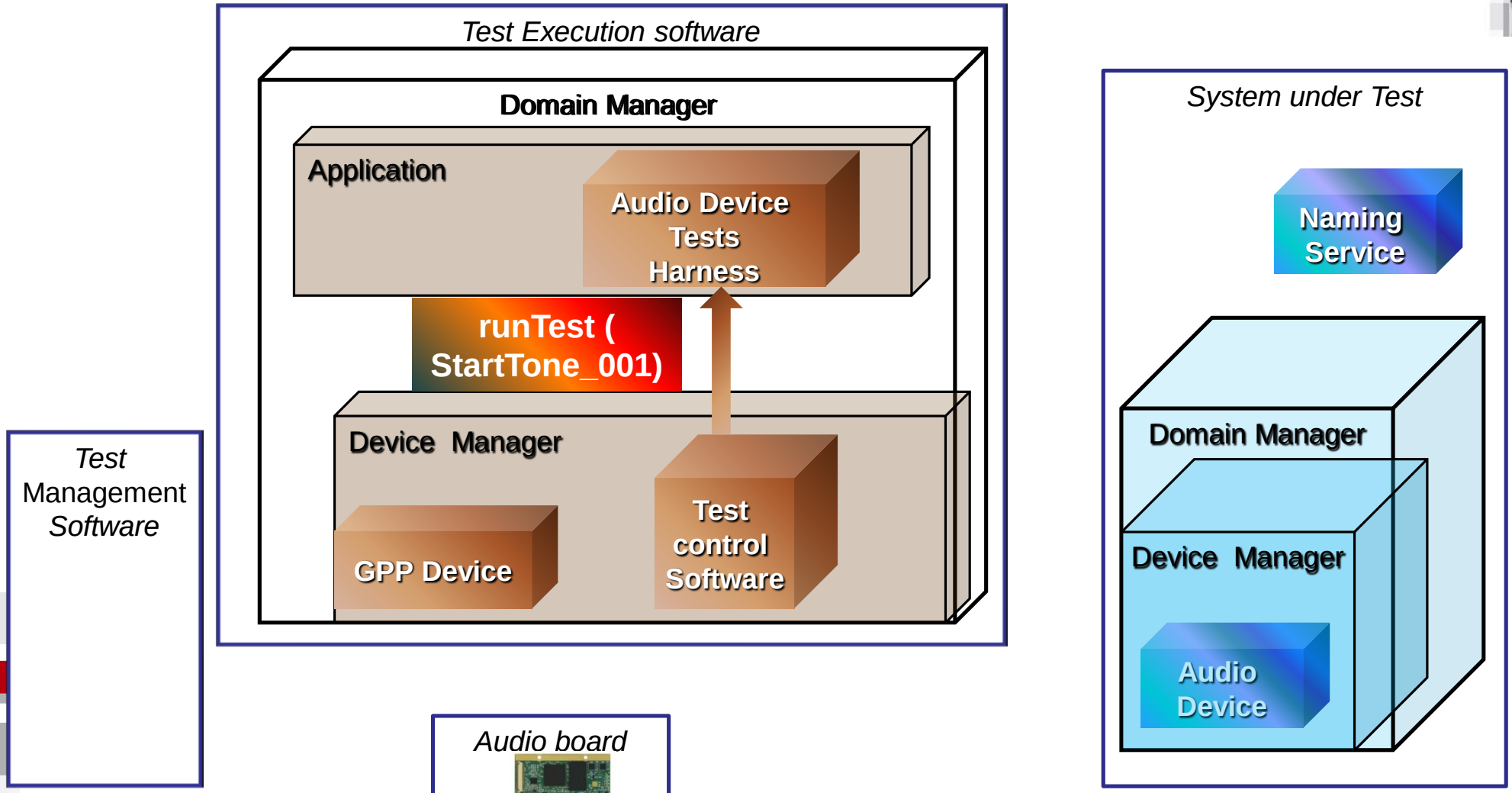
- **Test request from Test management Software**



Test Bench Data flow

Example on startTone() function test.

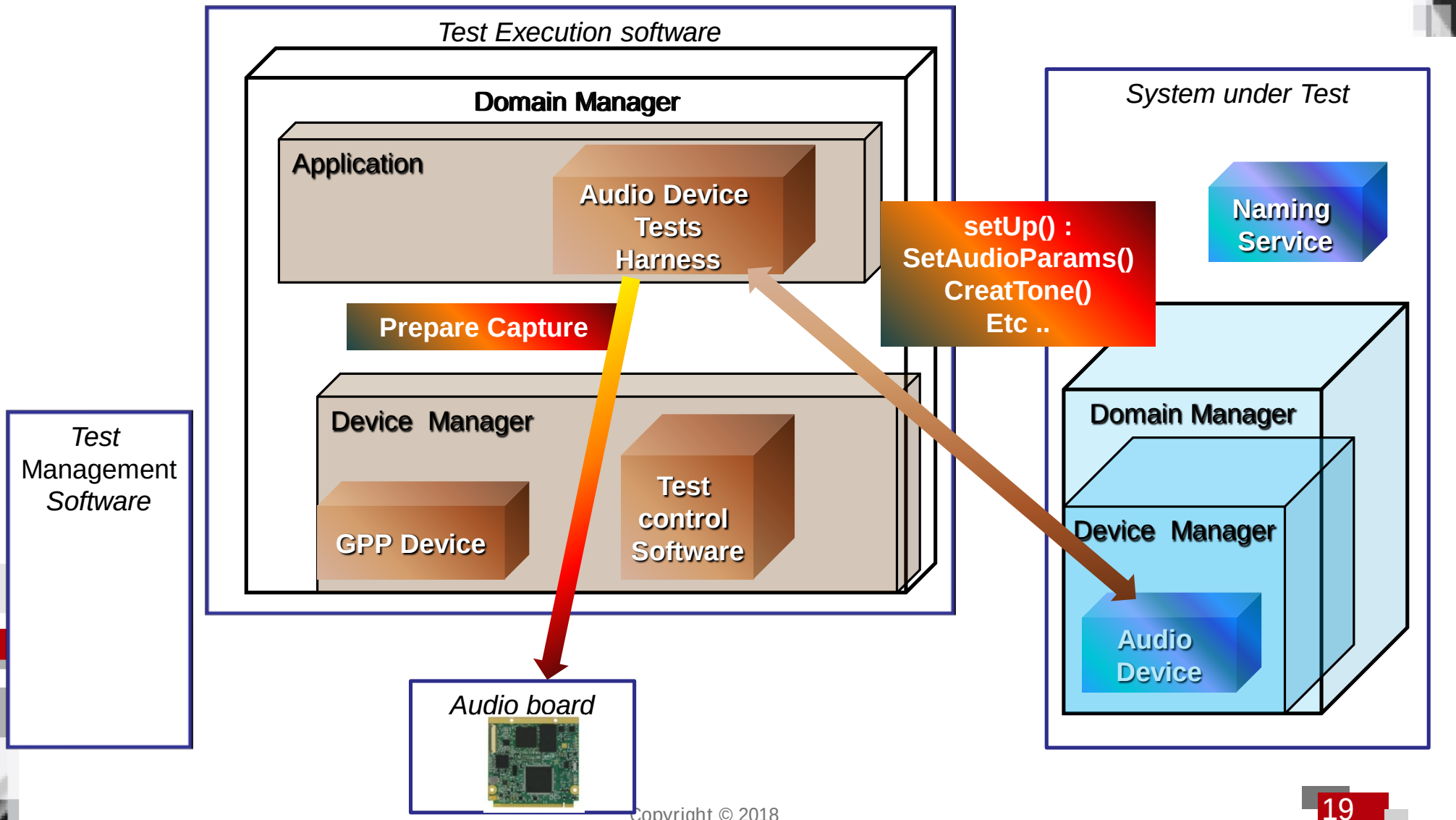
- **Test Request on Audio Device Harness**



Test Bench Data flow

Example on startTone() function test.

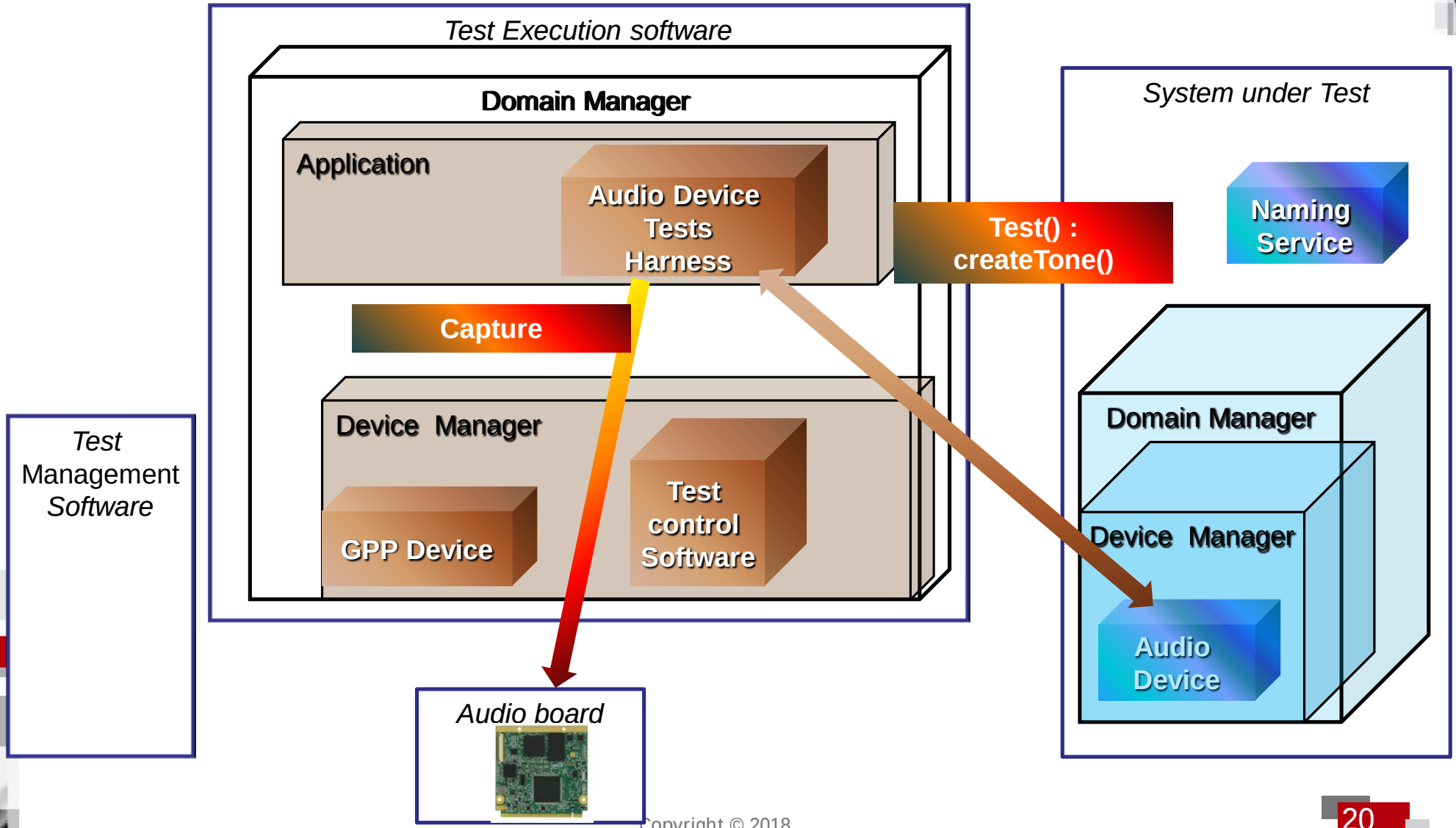
- **Test Set Up**



Test Bench Data flow

Example on startTone() function test.

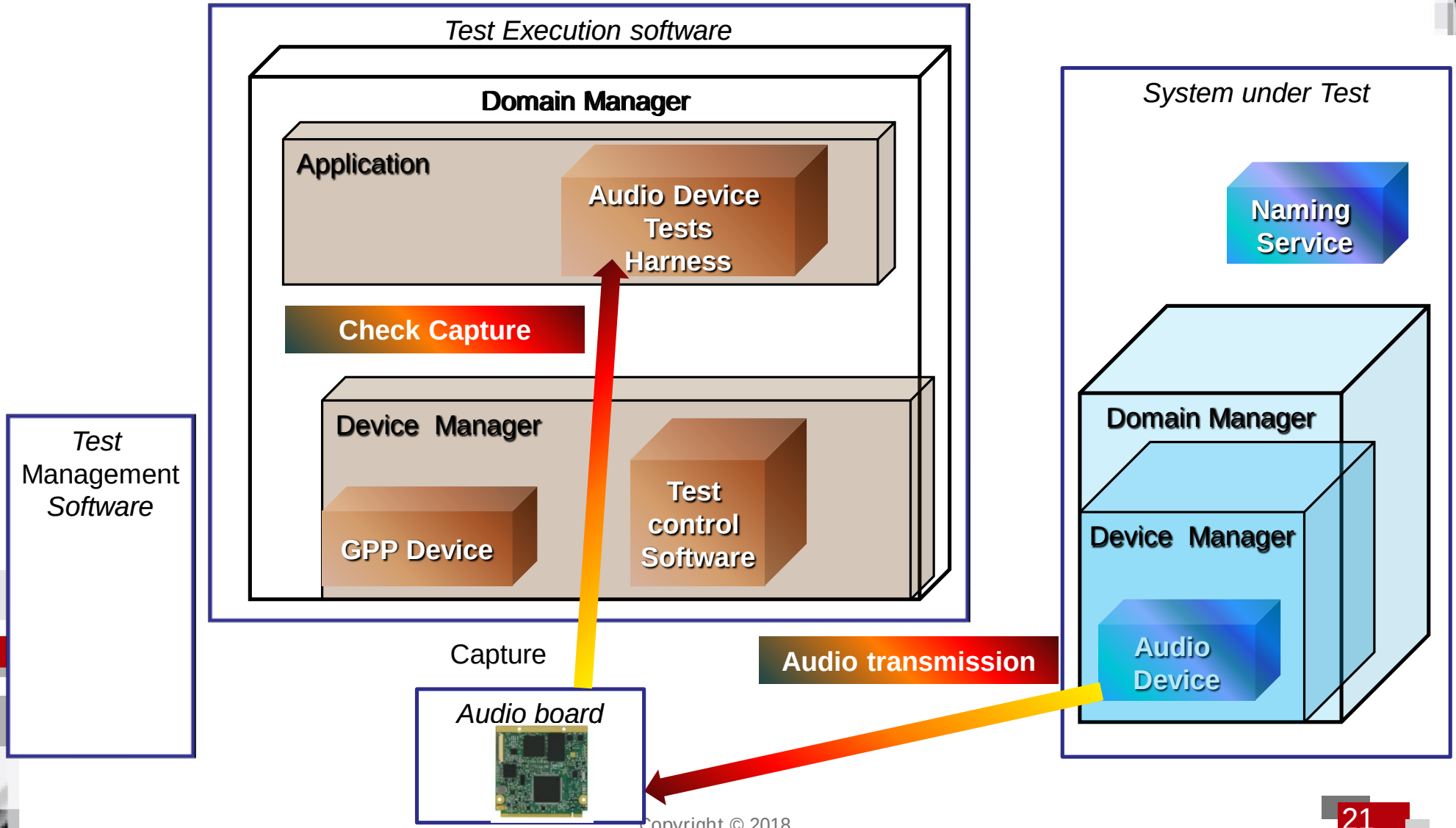
- **Test main initial phase**



Test Bench Data flow

Example on startTone() function test.

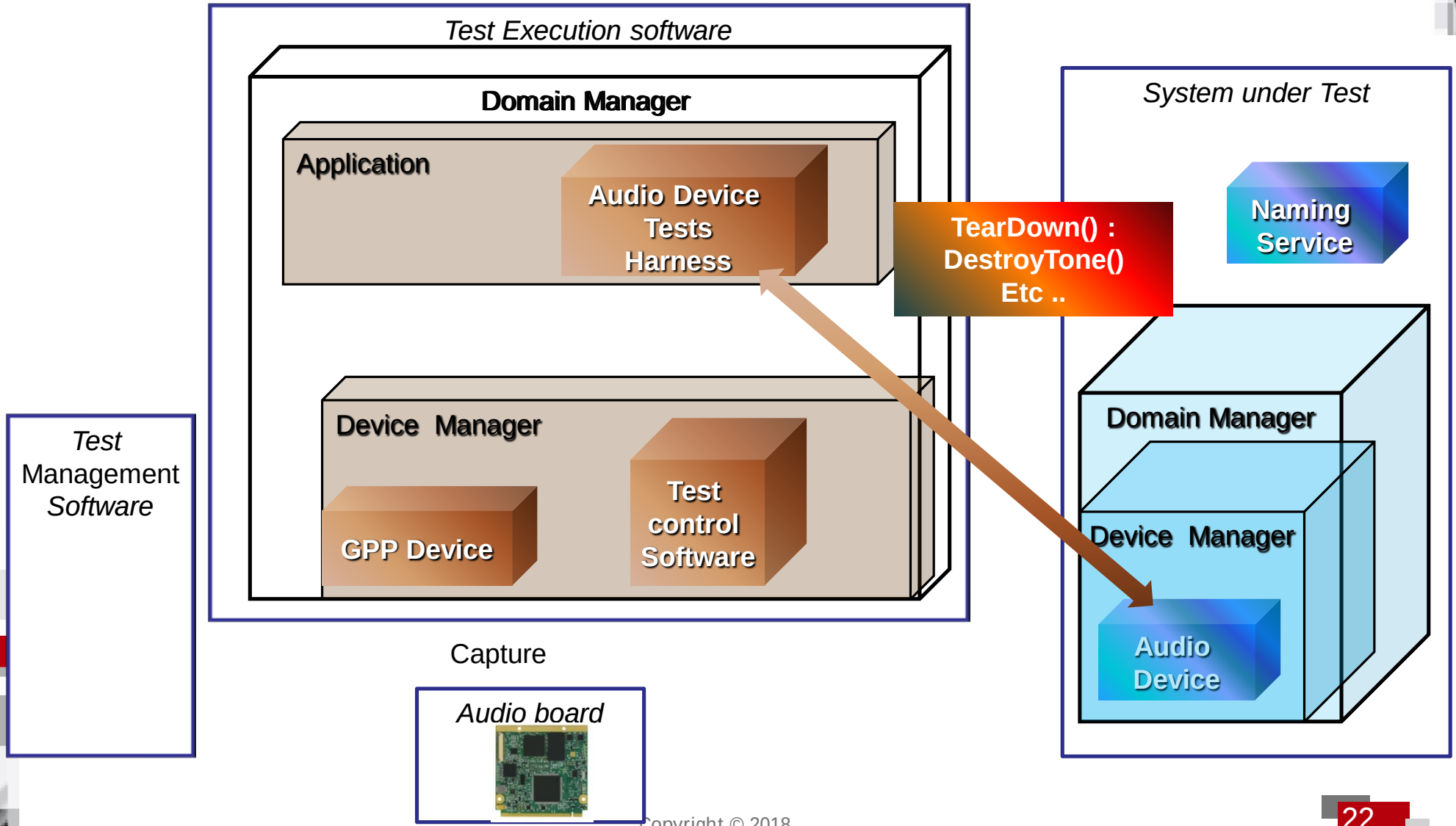
- **Test main capture phase**



Test Bench Data flow

Example on startTone() function test.

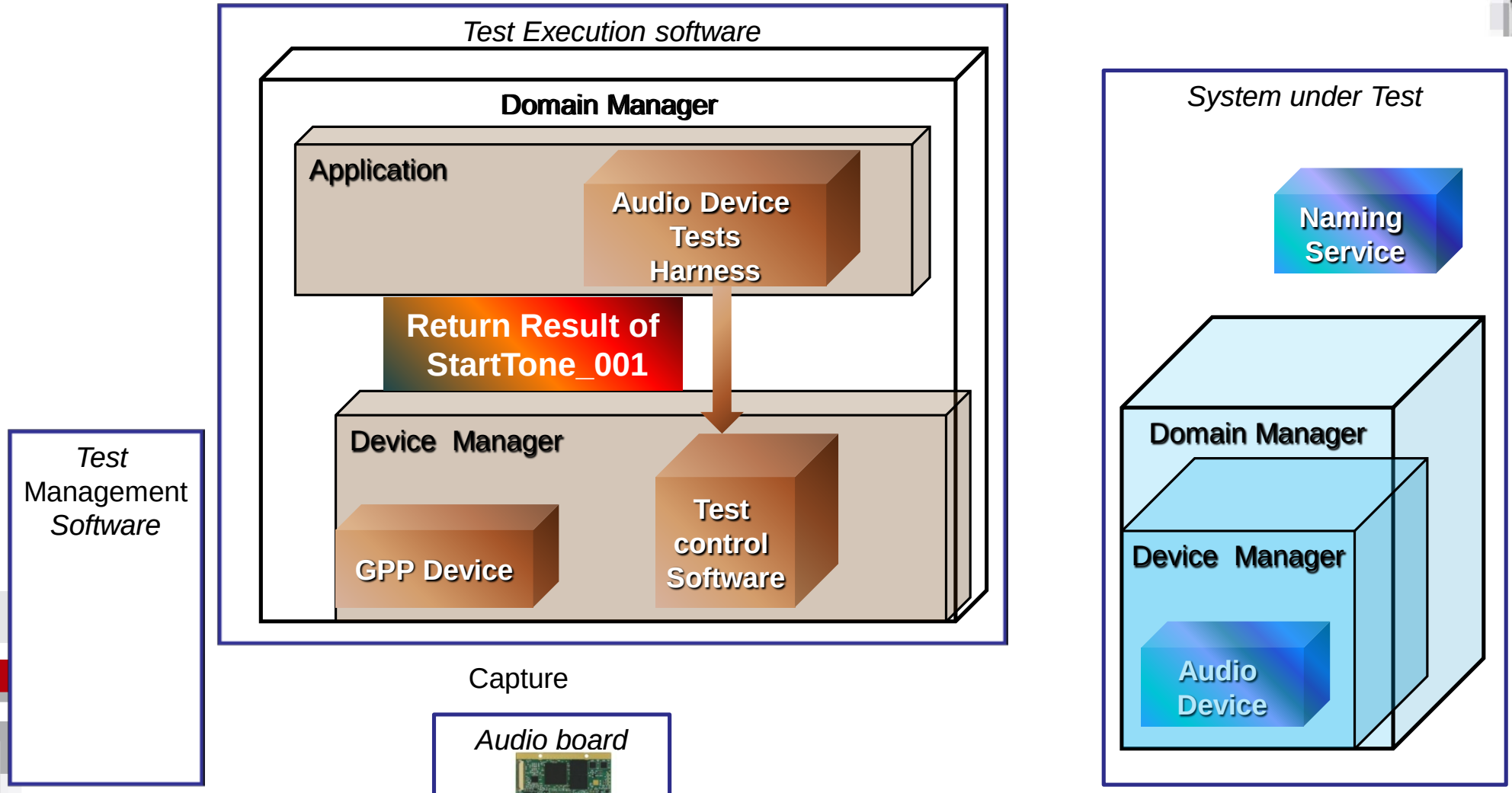
- **Test Tear Down**



Test Bench Data flow

Example on startTone() function test.

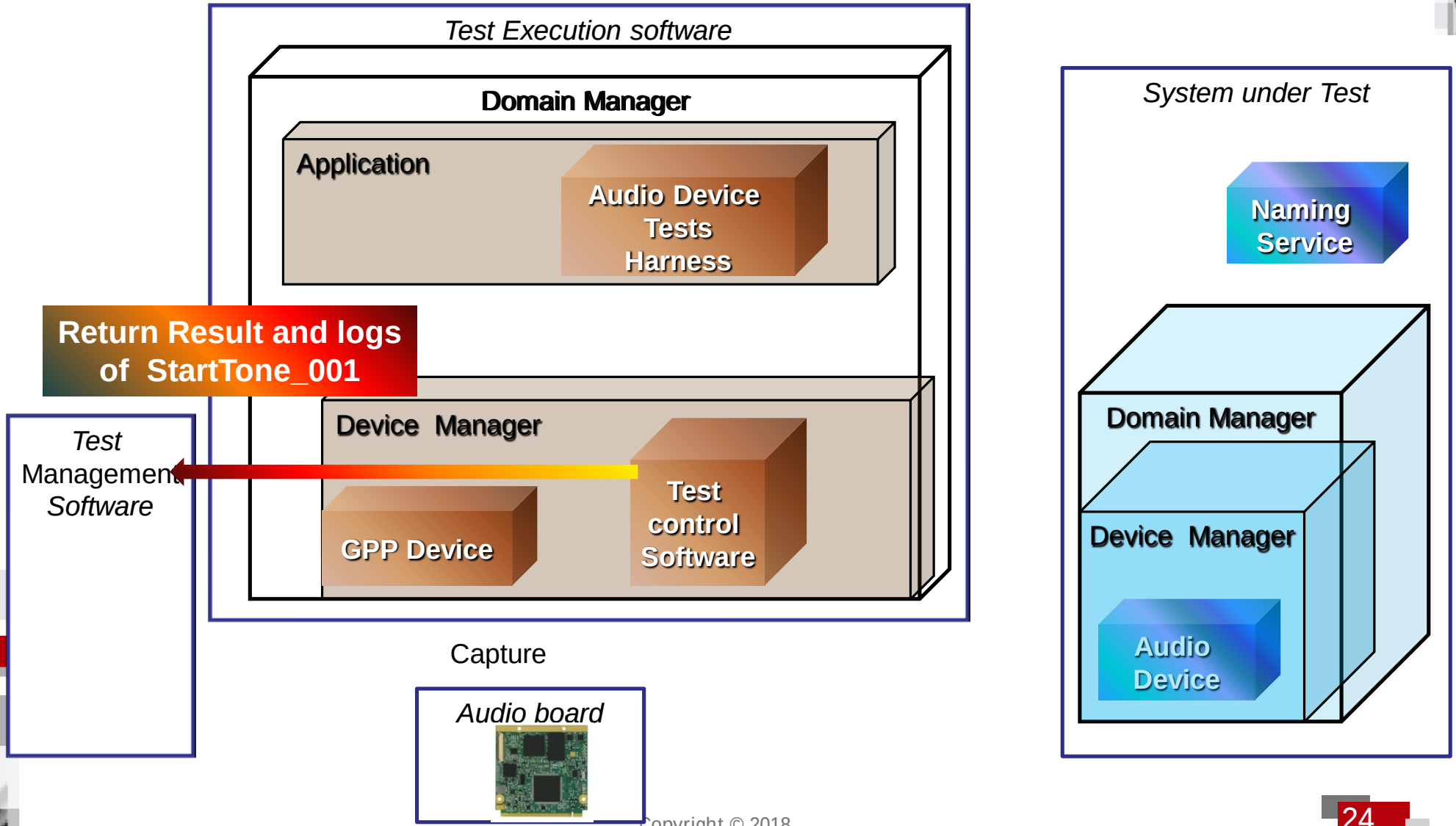
- **Result Return from Harness**



Test Bench Data flow

Example on startTone() function test.

- **Result return to Test Management Software**



Questions