

ETAS

LiveDevices  
ETAS Group

Vetronix  
ETAS Group



Frank Tränkle and Wolfgang Eismann / ETAS GmbH

# LABCAR-AUTOMATION

## Open Interfaces for Test Automation

TestingExpo 2005, Stuttgart

# Outline

- Evolution of Test Automation
- Systematic Approach for Test Automation Framework
- Open Interfaces
- Summary

# LABCAR-AUTOMATION

## Motivation

### **Challenge at Customer Robert Bosch:**

- More than 1000 automated system tests per year and per division
- Need to improve productivity
- Replace heterogeneous tool environment, standardize processes and tools

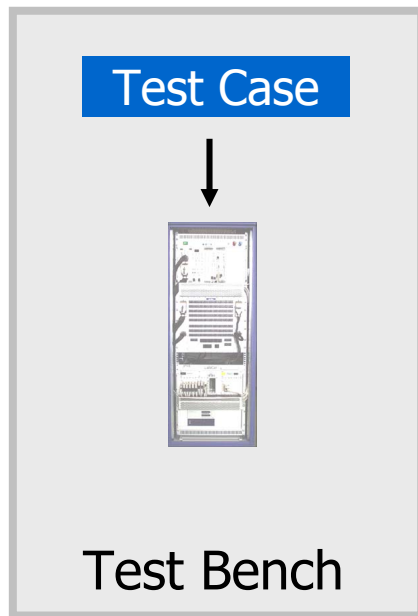
The Bosch logo, consisting of the word "BOSCH" in a bold, red, sans-serif font.

### **Solution: Provide new test automation tool, that ...**

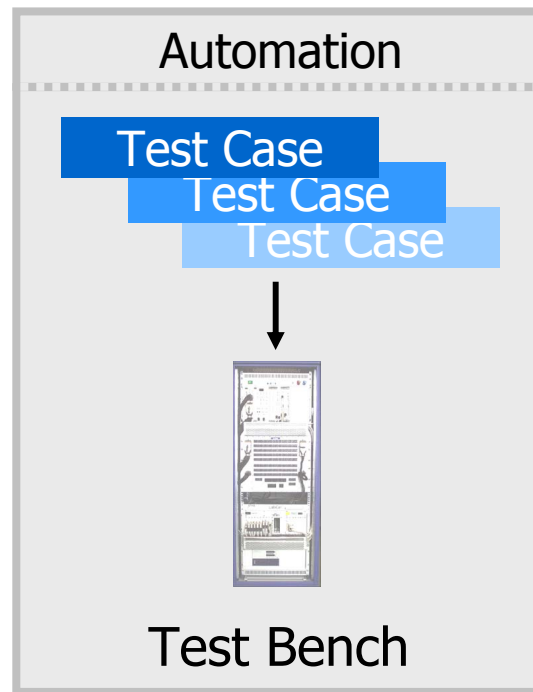
- Increases reusability of test cases
- Enables more transparent, manageable operation of test systems
- Offers high flexibility, so that it can be used as standardization platform for test automation

# Evolution of Automated Testing

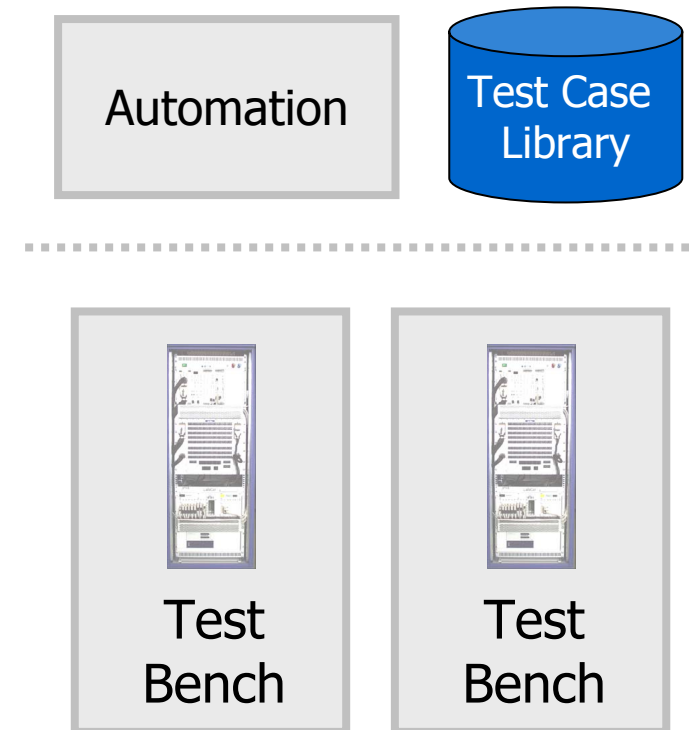
## Interactive Testing



## "Typical Automation"



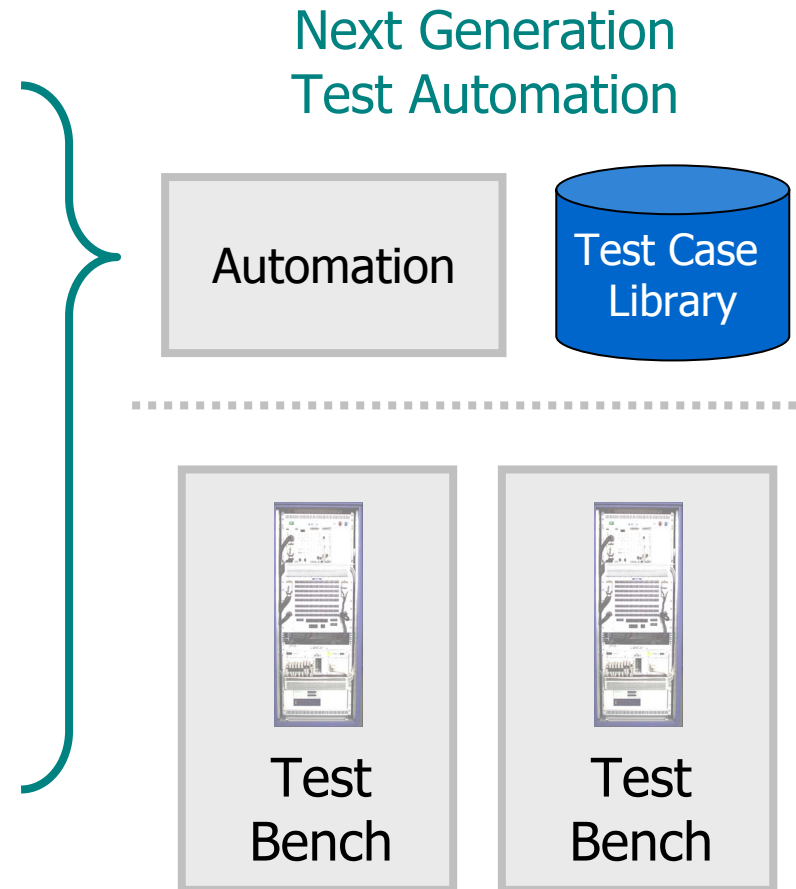
## Next Generation Test Automation



# Evolution of Automated Testing

## LABCAR-AUTOMATION provides:

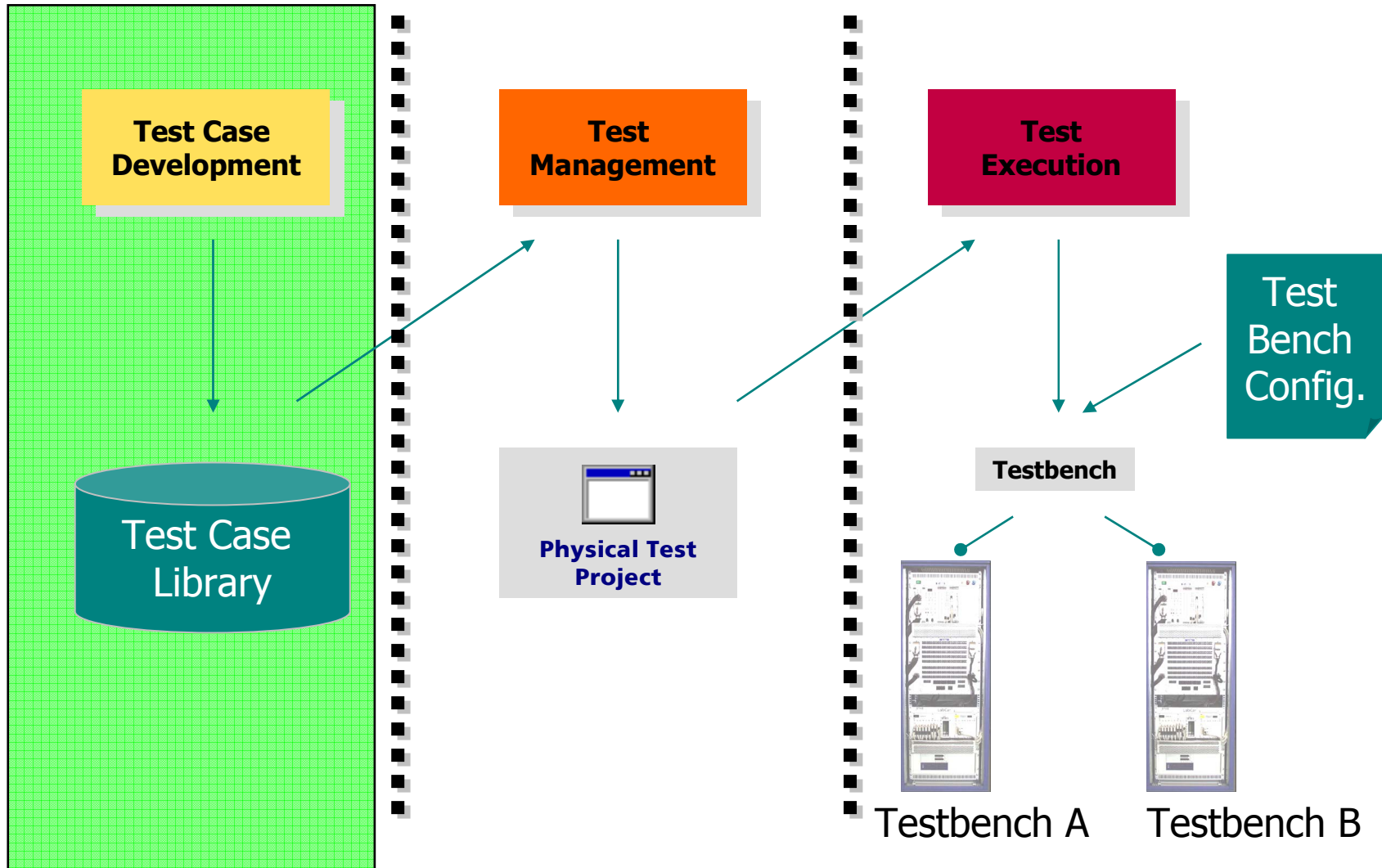
- Efficient development, deployment, and execution of reusable test cases
- Abstraction layer between test cases and test benches (test bench independence)
- 3rd party tool integration (open interfaces)
- Test design tool independence (matching various test case developer skills)



# Outline

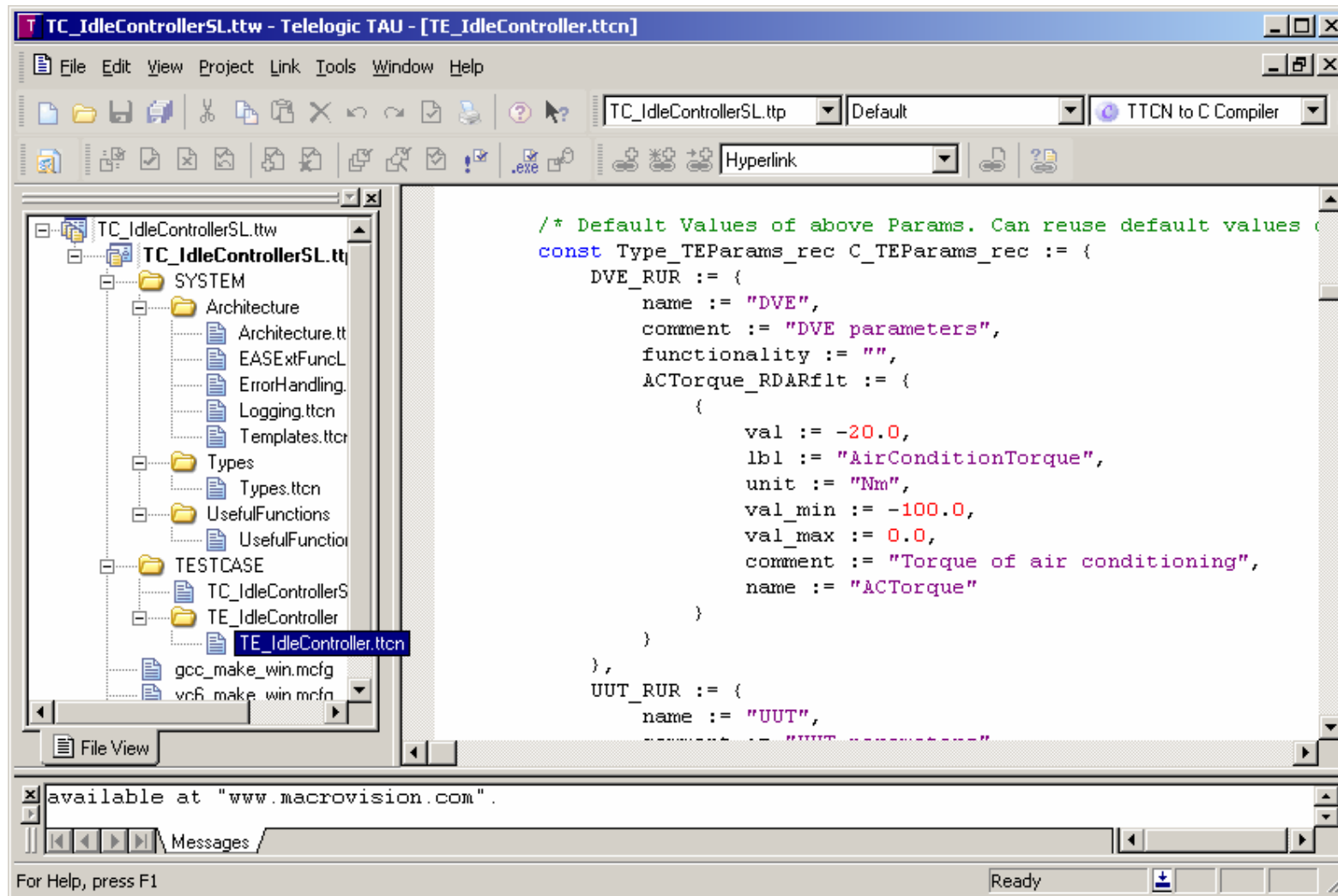
- Evolution of Test Automation
- Systematic Approach for Test Automation Framework
- Open Interfaces
- Summary

# Test Case Development

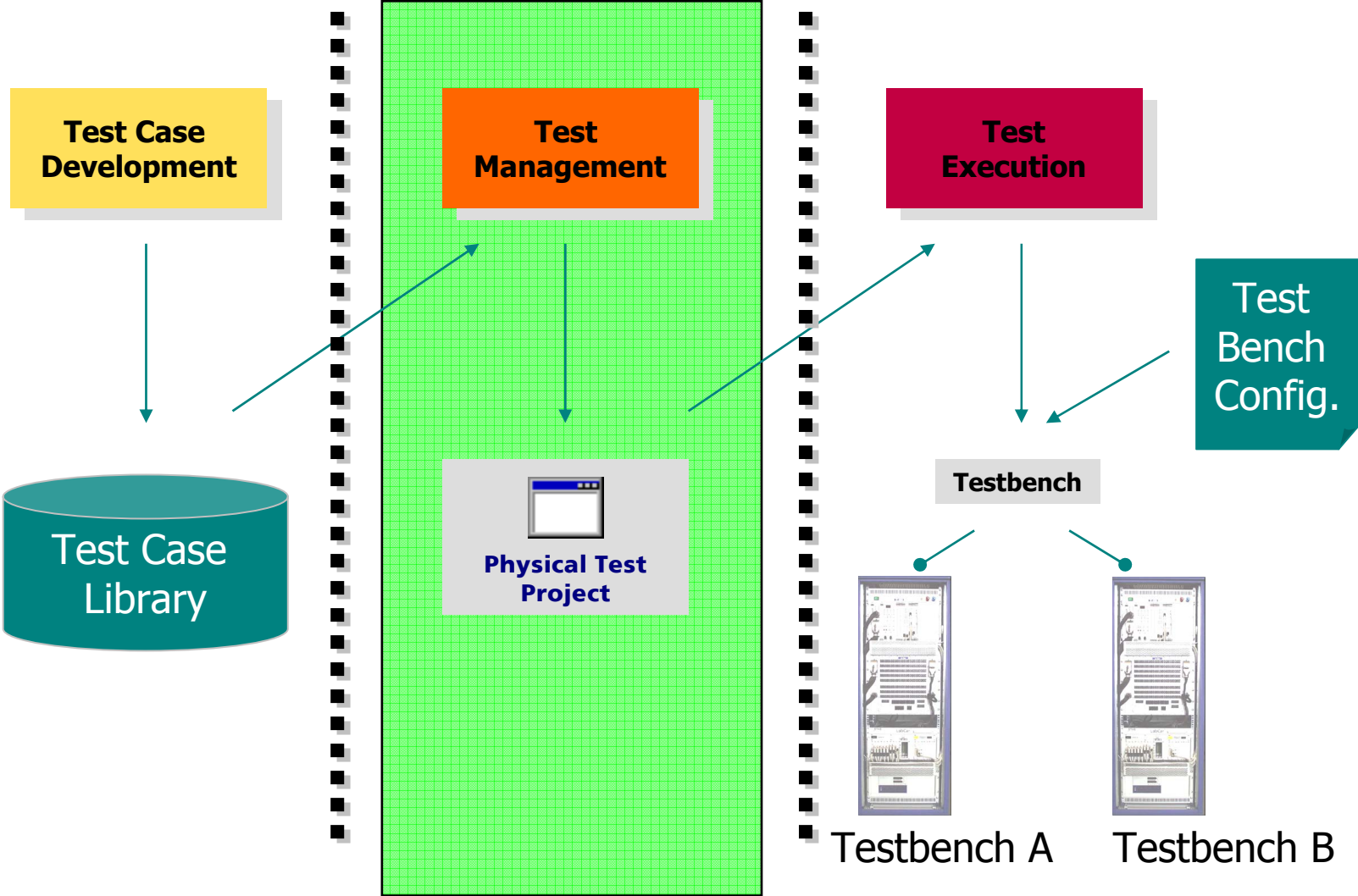


# Test Case Development

## Example: TTCN-3 Development with TAU/Tester

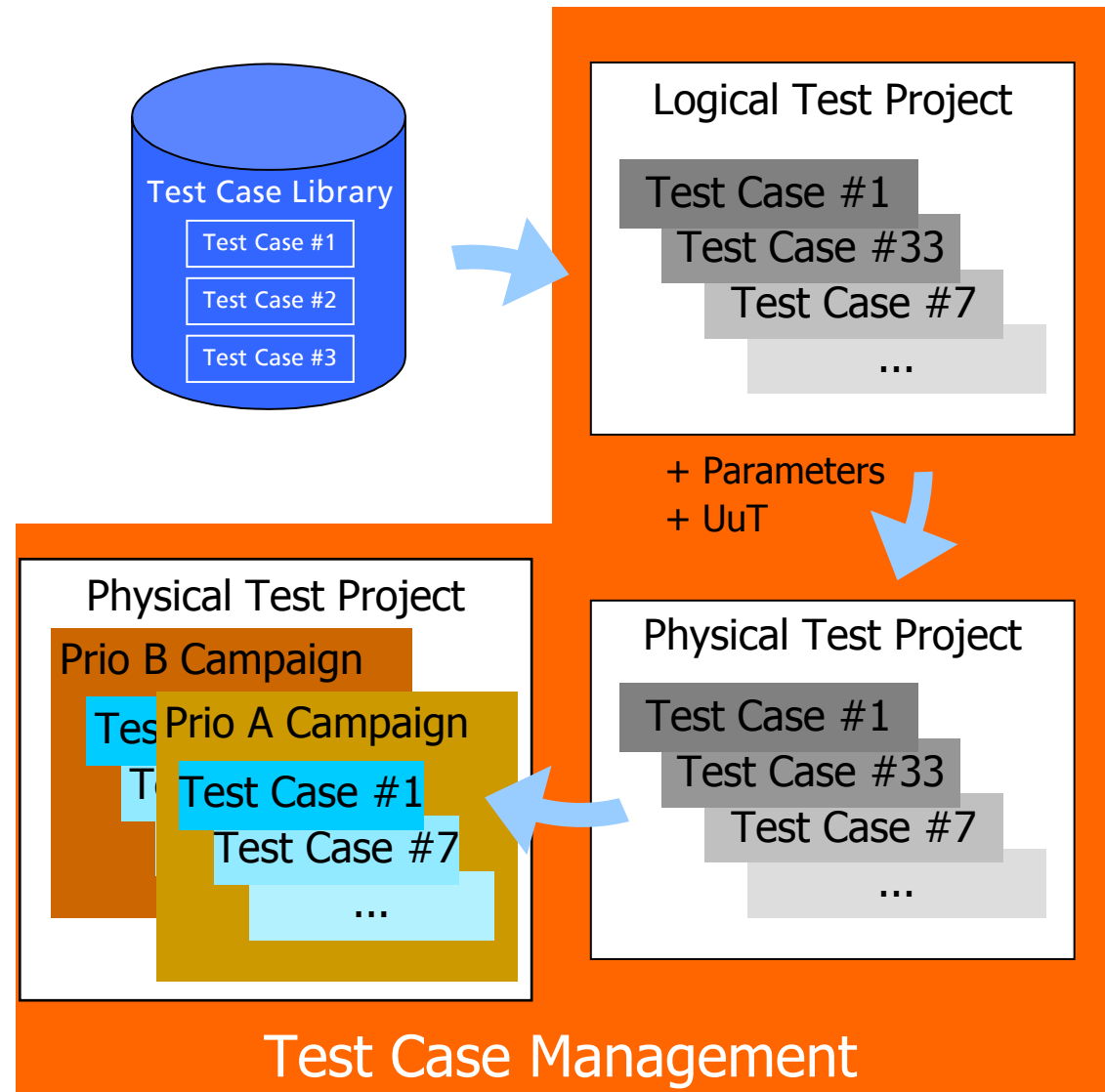


# Test Management



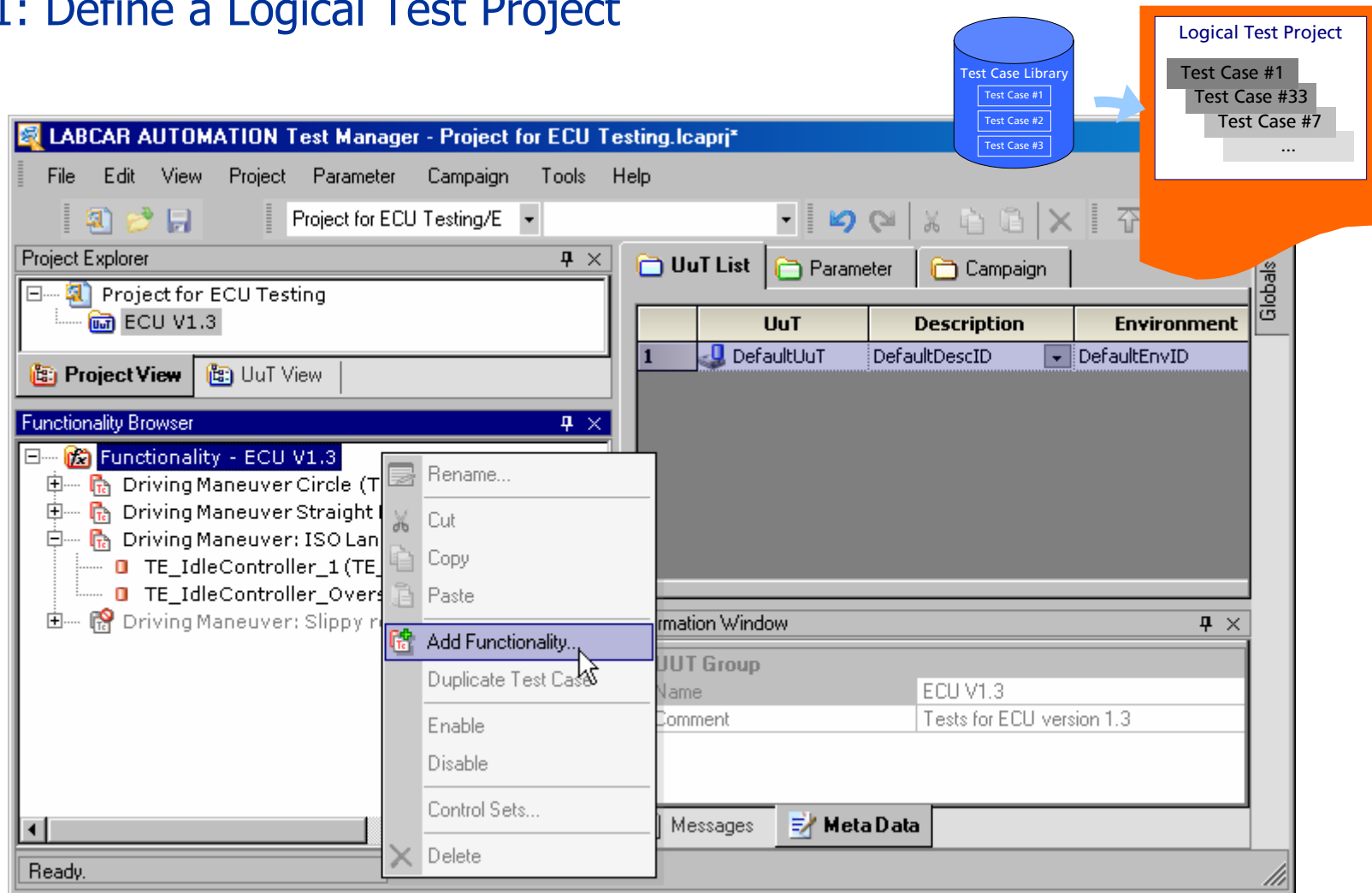
# Test Management

- Set up logical test project by selecting test cases from library
- Convert logical test project into physical test project by assigning test parameters and ECU variants
- Define reproducible test runs for a test project
- Test bench configured at execution time of test case / test campaign



# Test Management

## Step 1: Define a Logical Test Project



# Test Management

## Step 2: Adjust Test Case Parameters

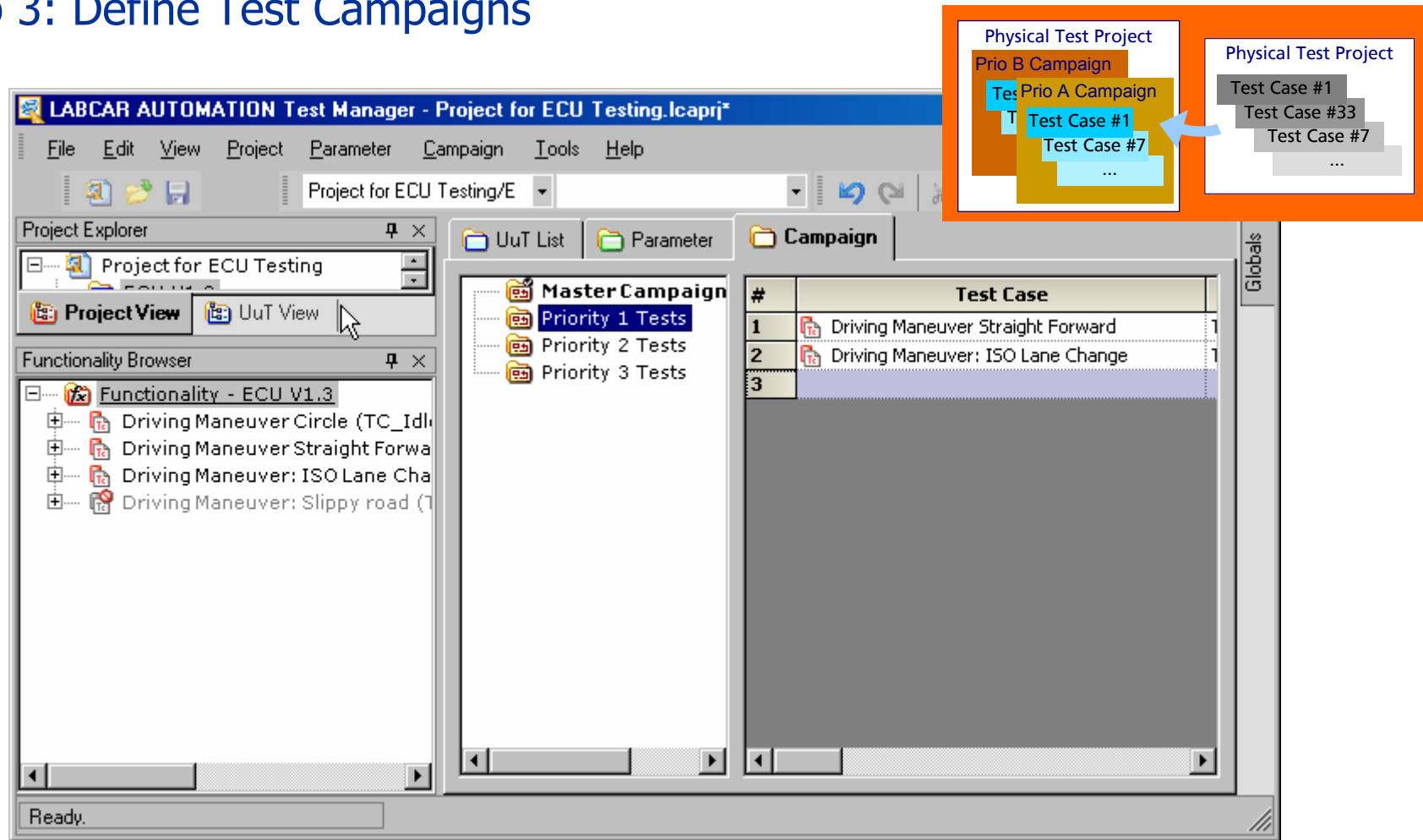
The screenshot shows the LABCAR AUTOMATION Test Manager interface. The main window displays a table of parameters for a test case. The table has columns for Name, Value, U, Te, and Comment. The parameters listed include StartAirCondition, Minimum Torque, StartEngine, modpar\_IdleController\_de..., StartTorque, StartTorque\_0, Idle, Pdle, Data Log File, StartEngine\_0, StartIgnition, Maximum Torque, Engine, and another modpar\_IdleController\_de... entry.

Name	Value	U	Te	Comment
StartAirCondition	On		St	Switch the air condition On
Minimum Torque	600		Mi	The minimum Torque
StartEngine	Off		St	Switch the engine Off
modpar_IdleController_de...				
StartTorque	10		St	Start Torque
StartTorque_0	-10		AI	Start Torque
Idle	0,0001		II	I
Pdle	0,001		PI	P
Data Log File	C:\ETAS\Tes ...		File	The File for the Datalogger
StartEngine_0	On		St	Switch the engine On
StartIgnition	On		St	Switch the ignition On
Maximum Torque	800		M	The maximum Torque
Engine	0		En	The measurement Engine
modpar_IdleController_de...				

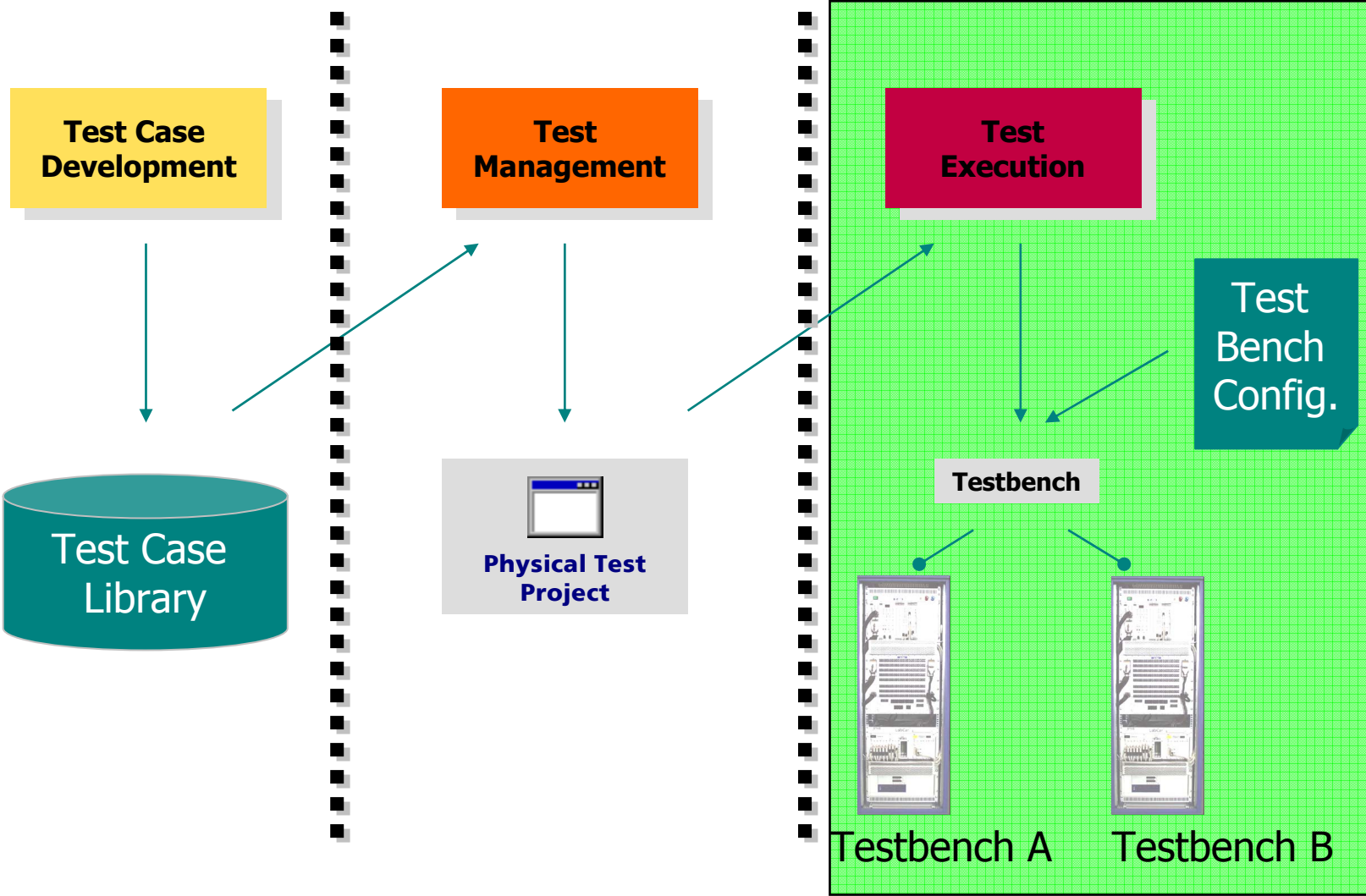
On the right side, there is a callout box with an orange border. It contains two sections: 'Logical Test Project' and 'Physical Test Project'. Both sections list 'Test Case #1', 'Test Case #33', and 'Test Case #7'. Below the 'Logical Test Project' section, there are two buttons: '+ Parameters' and '+ UuT'. A blue arrow points from the '+ Parameters' button to the 'Physical Test Project' section.

# Test Management

## Step 3: Define Test Campaigns



# Test Execution



# Test Execution

**Test Handler**

File Edit Execution Testbench

IUT

- ME701
  - Turbo Engines
  - Natural Aspirant Engines
    - G1
      - Master Campaign ✓
      - Quick And Dirty
      - Plaus\_no CarbCRC
      - Full\_Execution
        - Report\_FirstRun
        - Report\_2
      - StressTest
        - Report\_1
        - Report\_2
        - Report\_3
      - Plaus\_US1 ✓
    - G3A
      - Master Campaign ✓
      - Plausibility
      - InitCampaign
    - Small Natural Aspirant Engines
      - G2
      - G4B

		Test Case	Status	Start	Stop	Time	Module Name	Executable
1	1	Testcase_A	pass	12:10:04	12:17:35	00:07:31	Initialization	testcase_a.exe
2	1	Testcase_Y	fail	12:17:51	12:22:25	00:04:34	Plausability	testcase_y.exe
3	0	Testcase_H	skipped				Idle Speed Check	testcase_h.exe
4	1	Testcase_J	pass	12:23:22	12:24:11	00:00:49	Engine Check	testcase_j.exe
5	2	Testcase_X	running	12:24:13			Input Test 1	testcase_x.exe
6	0	Testcase_Z					Input Test 2	testcase_z.exe
7	7	Testcase_F					Output Test	testcase_f.exe
8	1	Testcase_K					Finalization	testcase_k.exe

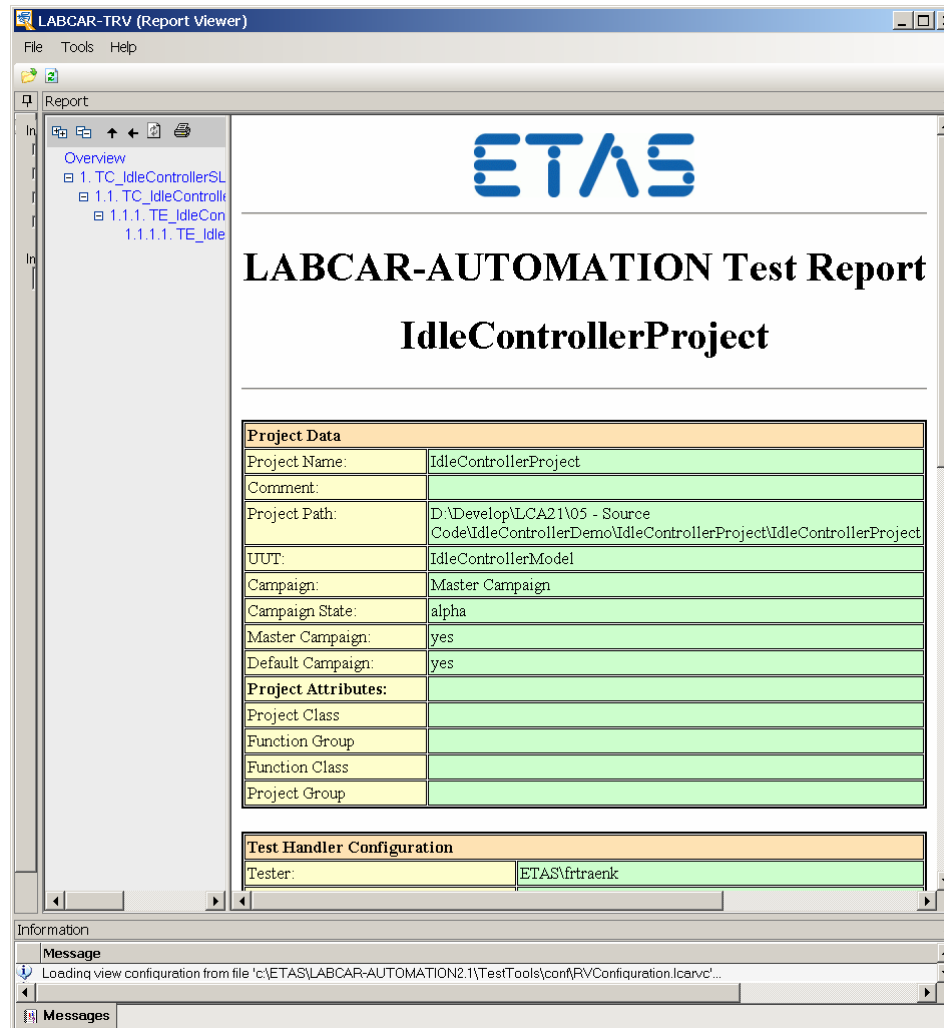
Info

- Note: IUT Group loaded.
- Error 156: TBC file is missing a port definition for P\_DIAG1!
- Warning: No report dirctory selected!
- Test run complete. Report was created.

Messages | Metadata

# Test Execution

## Hierarchical HTML View on XML Reports



# Product Features Overview

## Test Case Development Tools

- Open APIs for integration of 3rd party test case development tools and languages (e.g. Python)
- Integration package for Telelogic TAU/Tester development tool (TTCN-3 support)

## Test Management Tools

- Project Management
- Parameter Management
- Campaign Management

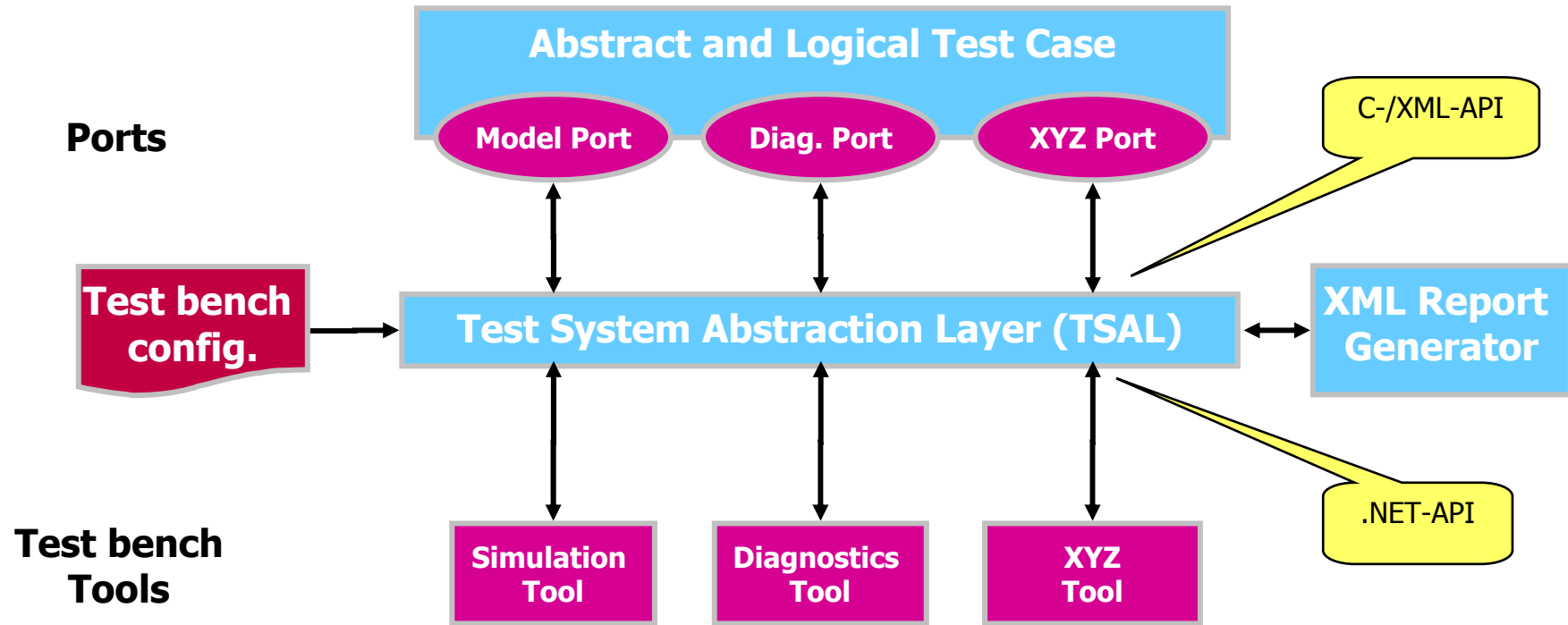
## Test Execution Tools

- Test Handler
- Report Viewer
- Support for HiL, MiL, SiL scenarios
- Open APIs for flexible integration of test bench tools
- Integration package for LABCAR / ETAS test bench tools

# Outline

- Evolution of Test Automation
- Systematic Approach for Test Automation Framework
- Open Interfaces
- Summary

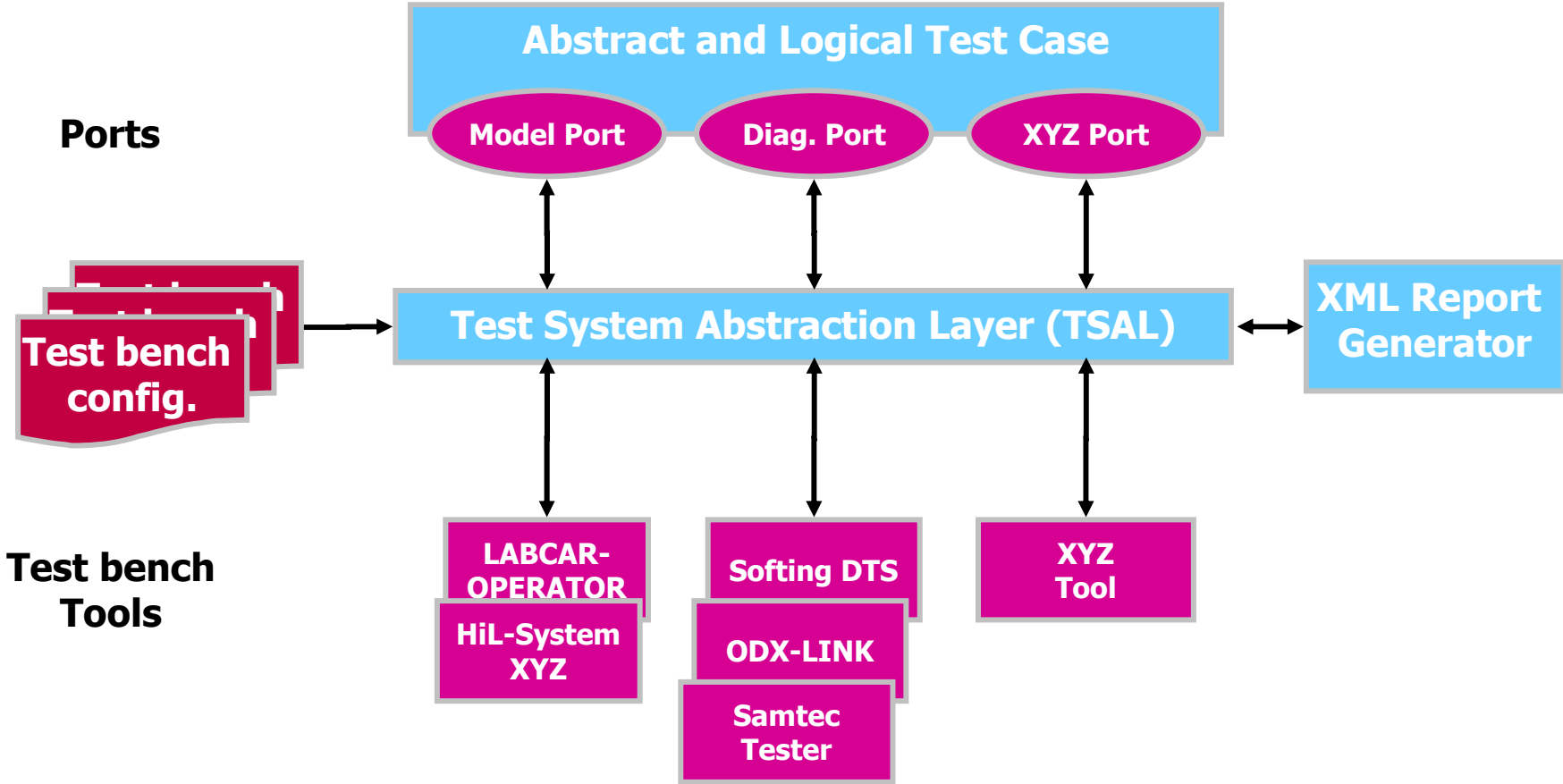
# Open Interfaces



- Test-bench-independent ports for developing abstract and logical test cases
- TSAL provides open .NET API to connect to test bench tools
- TSAL provides open C-/XML-API to connect to test languages and test design tools

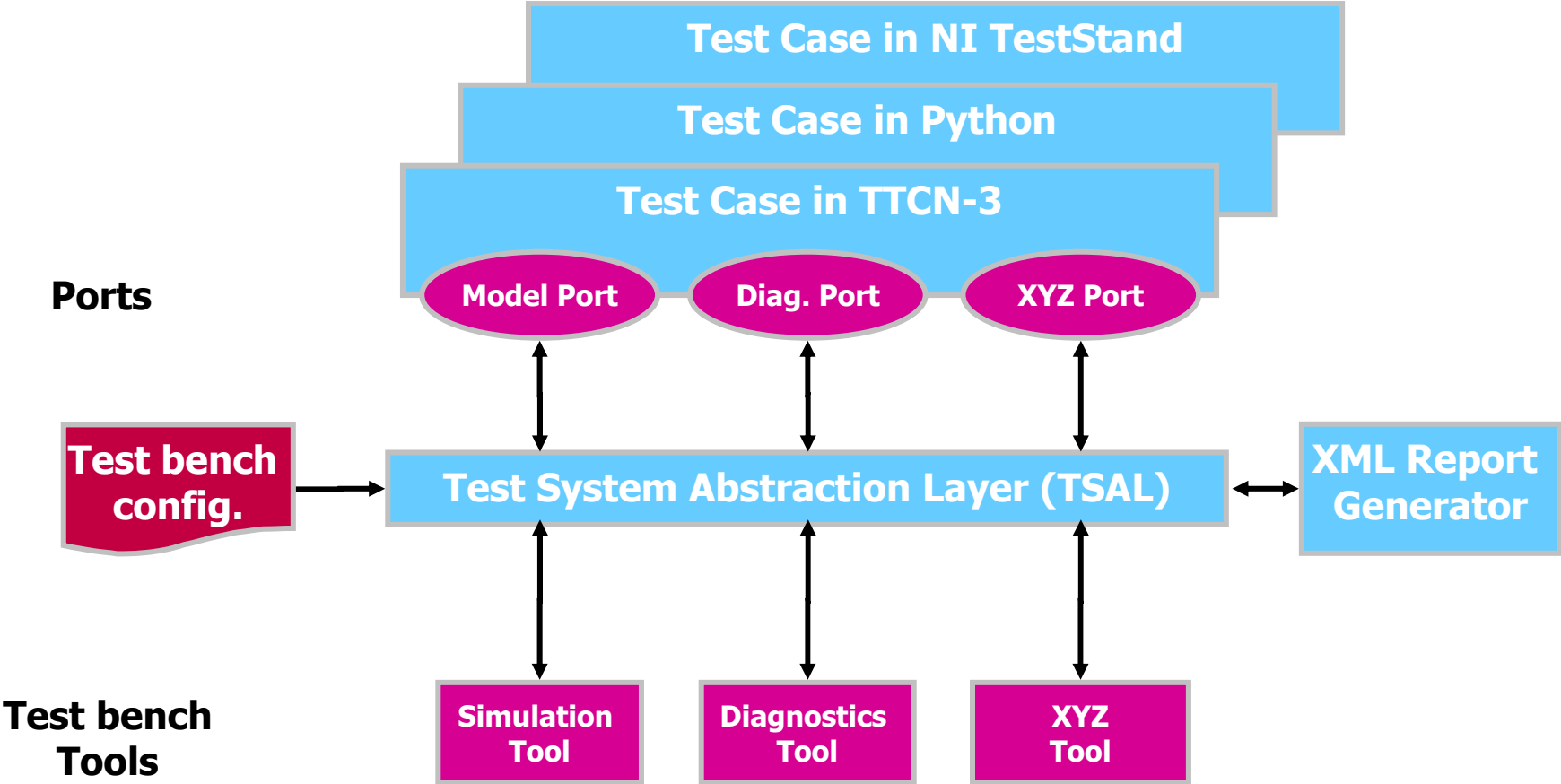
# Open Interfaces

## Dynamic Test Bench Configuration at Test Execution Time



# Open Interfaces

Open C-/XML-API to Test Languages and Test Design Tools



# Outline

- Evolution of Test Automation
- Systematic Approach for Test Automation Framework
- Open Interfaces
- Summary

# Summary

## LABCAR-AUTOMATION

- Graphical user front ends for
  - test case development,
  - test management and
  - test execution
  - **Process-safe test management and execution**
- Test bench- and UuT- independent test cases
  - **Improved reuse for test cases**
- Open Interfaces for
  - connection to test bench tools
  - connection to test languages and test design tools
  - **Investment protection for test case development**
  - **Investment protection for test bench tools (software + hardware)**

Thank you for your attention!

Your questions are welcome!

**2005-01-1040**