OTX according to ISO 13209 – User Training

Automotive Diagnostic Systems

Dr.-Ing. Jörg Supke, emotive GmbH & Co. KG, Ostfildern-Stuttgart, Germany

- Founder and CEO of EMOTIVE GmbH & Co. KG

- Since 2015 responsible for the German ISO working group "Data Exchange Formats and Test Protocols" at VDA (OTX, ODX, MVCI)

  - Since 2024 convenor of WG 5

- Since 2016 member of ASAM technical steering committee

- Project leader of various OTX working groups at ASAM and ISO

About EMOTIVE

- EMOTIVE is an independent, German based expert for automotive diagnostic systems

- EMOTIVE sees itself as the technical market leader and driver for OTX software tools

- EMOTIVE is actively driving the standardization of OTX

- EMOTIVE is a founding member of the Automotive Engineering Tool Alliance (AETA)
  - RA Consulting, Intrepid Control Systems, CarMedialab, EMOTIVE
  - Over 350 employees and 300 customers worldwide

EMOTIVE Main Customers
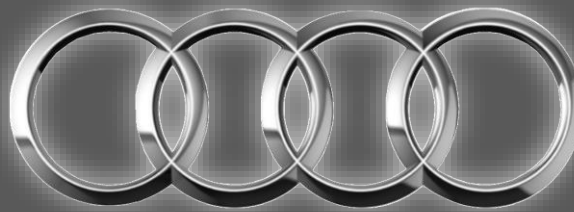
Automotive Diagnostic Systems - OTX User Training

Participants

- Introduce yourself

- Why do you want to use OTX?

- What do you expect from the seminar?

emotive

Agenda

OTX User Training

All

Agenda

First Day	Content	Second Day	Content
09:00 – 09:45	Introduction	09:00 – 10:30	Run and Test
09:45 – 10:30	Introduction in OTX	**10:30 – 10:45**	**Break**
10:30 – 10:45	**Break**	10:45 – 12:15	Advanced Basics OTX
10:45 – 12:15	OTX Projects	**12:15 – 12:45**	**Lunch break**
12:15 – 12:45	**Lunch break**	12:45 – 14:15	Advanced Basics OTF
12:45 – 14:15	OTF and OTX-Mapping	**14:15 – 14:30**	**Break**
14:15 – 14:30	**Break**	14:30 – 16:00	OTX-Runtime API
14:30 – 16:00	OTX as programing language		

Automotive Diagnostic Systems - OTX User Training

Goals of the Seminar

Overall overview and foundations of OTX for further in-depth knowledge

1. Understand OTX in the process

2. Use OTX software tools

3. Integrate OTX into target systems

4. Programming OTX

Legende:

Important!

For experts only!

ABC
123 Joint exercise

Roles
Within the OTX application process

| Role	Task
Checker	Checking an existing OTX sequence against a new Runtime-PDX
User	Execution of sequences on the vehicle
Specifier	Specification of OTX sequences
Target System Integrator	Integrating a newly programmed OTX sequence into a target environment
Bug Fixer	Implements small changes in OTX sequences without changing the basic test logic
Programmer	Programs new OTX sequences based on a specification and authoring guidelines
Target System Programmer	Integrates the OTX-RT API into a target system
Administrator	Generates OTX author guidelines and makes technical concepts for OTX development and application

Knowledge needed

Automotive Diagnostic Systems - OTX User Training

Content
OTX User Training

1. Introduction OTX
2. Basic working with OTX
3. OTX projects
4. OTX tools
5. ControlFile & OTX mapping
6. OTX mapping editor
7. Player projects
8. OTX as programing language
9. Advanced Basics OTX
10. Advanced Basics OTF
11. OTX runtime environment
12. Advanced basics OTX-Runtime
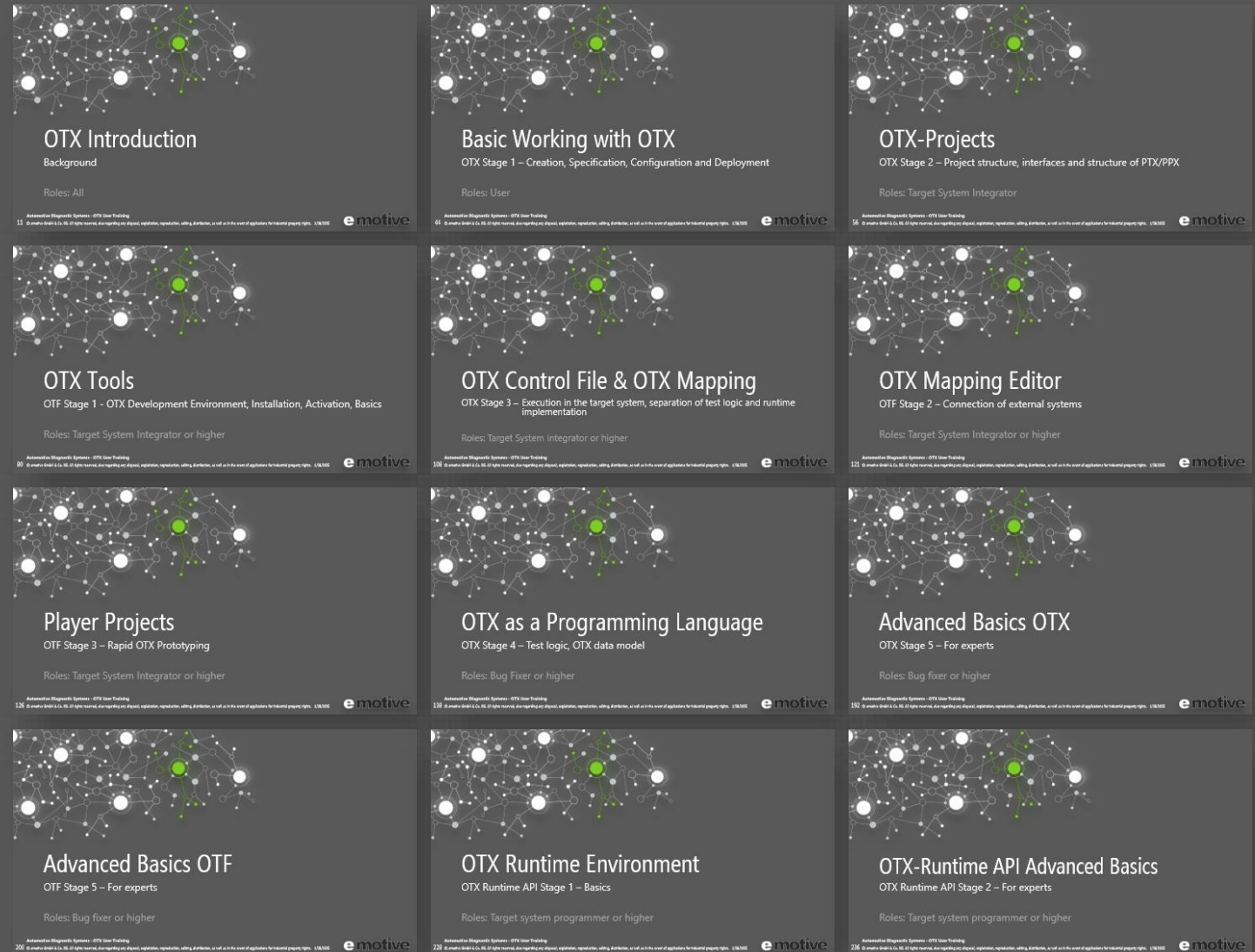
OTX Introduction

Background

1/28/2025

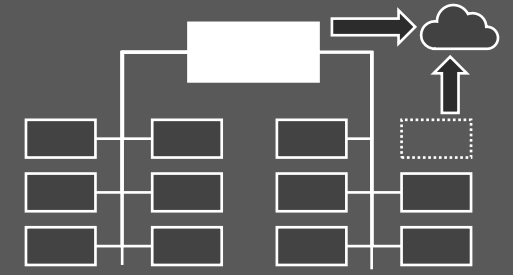**e motive**

Connected World
Over-The-Air

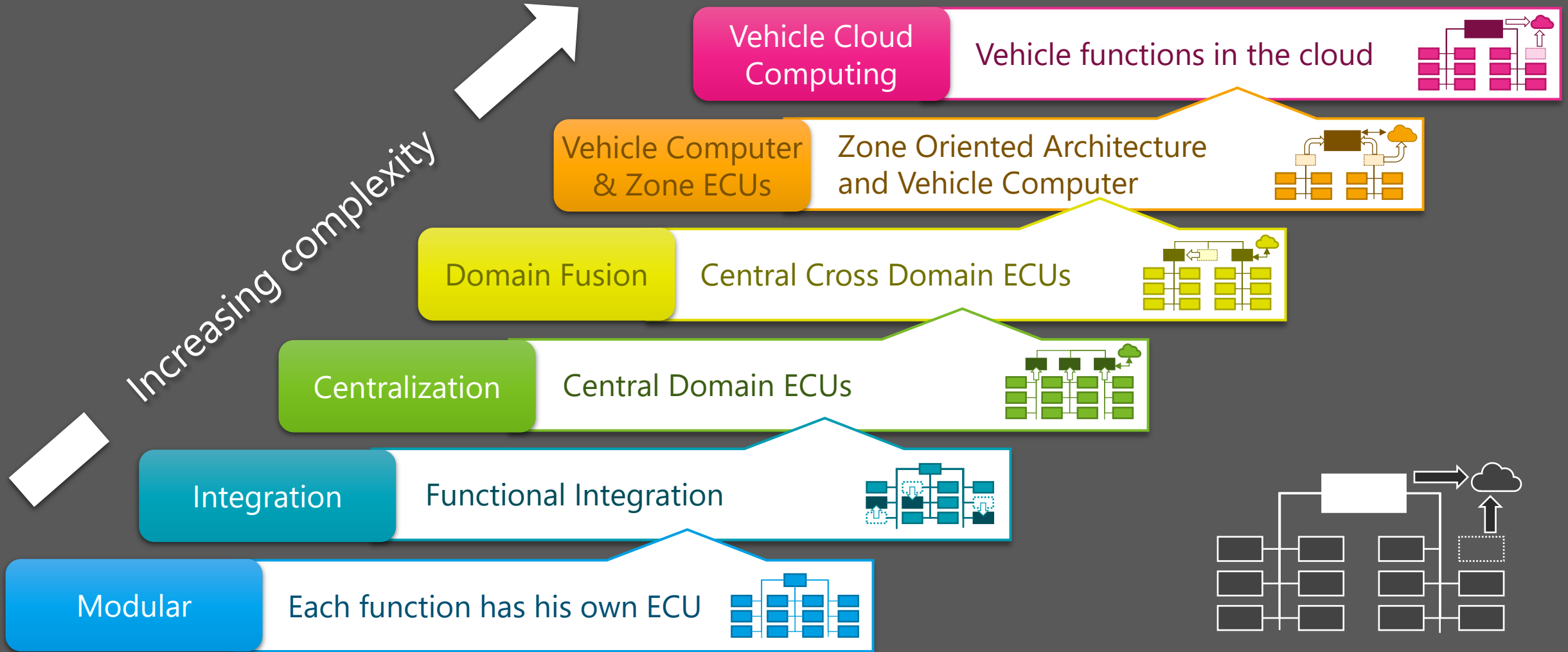1. Comprehensive availability of secure, broadband, cross-system communication

2. Ever more demanding customer requirements

3. Constantly increasing global competition

➡ Growing complexity and permanent pressure on all operational procedures and processes

Automotive Diagnostic Systems - OTX User Training

1/28/2025

„A holistic end-to-end process organization is the key to operational excellence"

emotive

Evolution of the E/E Architecture

Increasing complexity

Vehicle Cloud Computing — Vehicle functions in the cloud

Vehicle Computer & Zone ECUs — Zone Oriented Architecture and Vehicle Computer

Domain Fusion — Central Cross Domain ECUs

Centralization — Central Domain ECUs

Integration — Functional Integration

Modular — Each function has his own ECU

emotive

The challenge is mastering the constantly growing complexity

1/28/2025

„ Consistently standardized processes are a prerequisite for continuous improvement, consistency in the provision of services and the avoidance of redundancies"

Source: DIN Website

Automotive Diagnostic Systems - OTX User Training

1/28/2025

e motive

What is OTX?

International Organization for Standardization

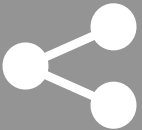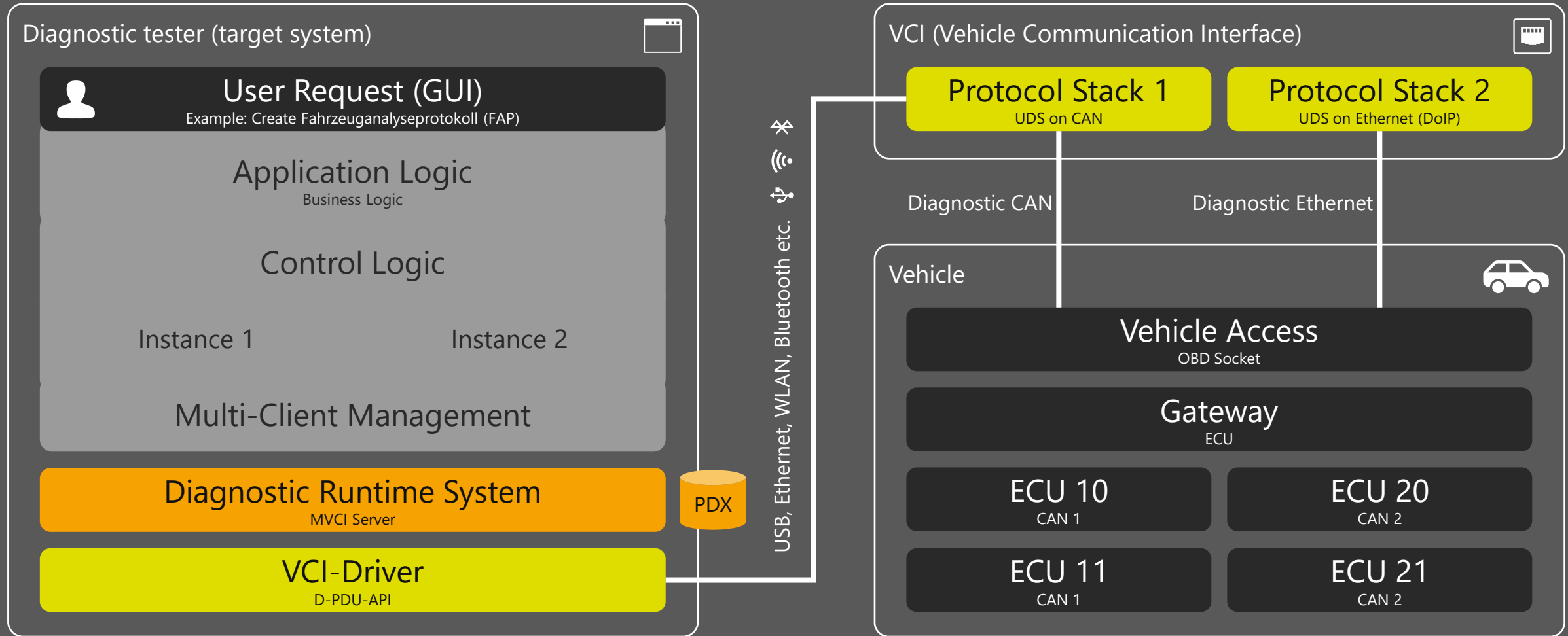|			
Harmonizing Effects	Programming Language for Testing Domain (DSL)	ISO Standard (ISO 13209)	Measurable Quality (Process Reliability)
Integrable	Executable	Graphically Expressible	Industry Proven
Open and Extensible	Platform Independent	Manufacturer Independent	Active Further Development

emotive

What is OTX?

OTX is a domain specific programming language (DSL) for process reliable description of exchangeable and executable test logic inside automotive industry.

What is OTX?

Software-technical structure of a diagnostic application without OTX

Diagnostic tester (target system)

- User Request (GUI)
  Example: Create Fahrzeuganalyseprotokoll (FAP)
- Application Logic
  Business Logic
- Control Logic
  - Instance 1
  - Instance 2
- Multi-Client Management
- Diagnostic Runtime System
  MVCI Server
- VCI-Driver
  D-PDU-API

PDX

USB, Ethernet, WLAN, Bluetooth etc.

VCI (Vehicle Communication Interface)

- Protocol Stack 1
  UDS on CAN
- Protocol Stack 2
  UDS on Ethernet (DoIP)

Diagnostic CAN   Diagnostic Ethernet

Vehicle

- Vehicle Access
  OBD Socket
- Gateway
  ECU
- ECU 10
  CAN 1
- ECU 20
  CAN 2
- ECU 11
  CAN 1
- ECU 21
  CAN 2

ISO
International Organization for Standardization

e motive

What is OTX?

Software-technical structure of a diagnostic application with OTX

Diagnostic tester (target system)

- **User Request (GUI)** — Example: Create Fahrzeuganalyseprotokoll (FAP)
- **Application Logic** — Business Logic
- **OTX-Control-File** — Independent, data-driven control logic — CTF
- **OTX Runner 1** — Encapsulated diagnostic test logic instance
- **OTX Runner 2** — Encapsulated diagnostic test logic instance — PTX
- **OTX DiagManager** — Modular multi-client management (also stand-alone)
- **Diagnostic Runtime System** — MVCI Server, SOVD, etc. — PDX
- **VCI Driver** — D-PDU-API

USB, Ethernet, WLAN, Bluetooth etc.

VCI (Vehicle Communication Interface)

- **Protocol Stack 1** — UDS on CAN
- **Protocol Stack 2** — UDS on Ethernet (DoIP)

Diagnose-CAN   Diagnose-Ethernet

Vehicle

- **Vehicle Access** — OBD Socket
- **Gateway** — ECU
- **ECU 10** — CAN 1
- **ECU 20** — CAN 2
- **ECU 11** — CAN 1
- **ECU 21** — CAN 2

International Organization for Standardization

Automotive Diagnostic Systems - OTX User Training

What is OTX?
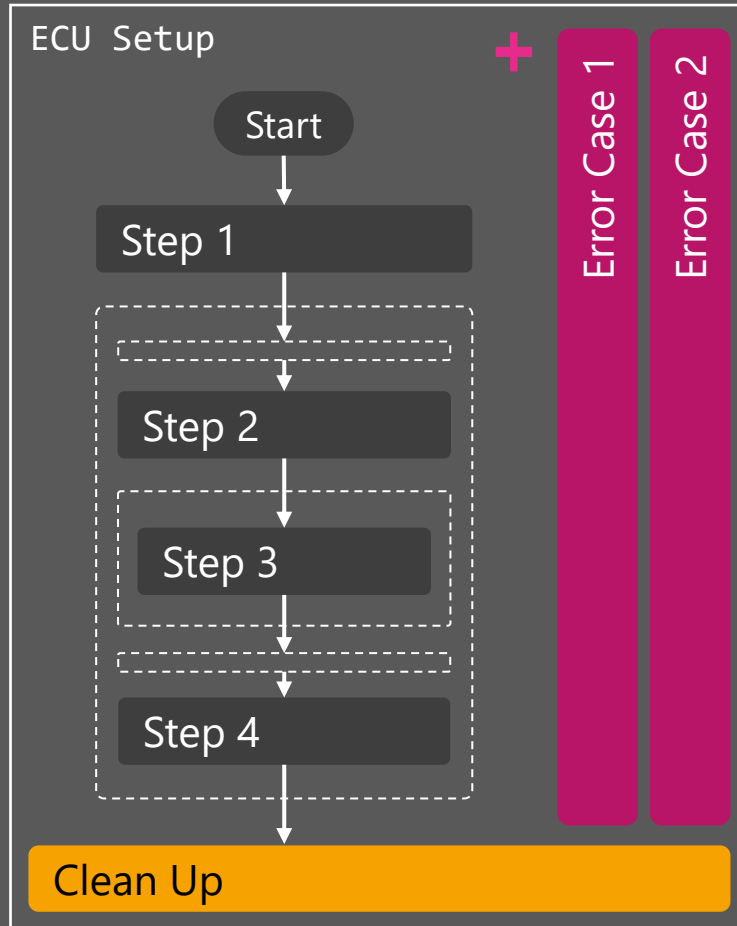
"Hello World" Example graphically, in OTL and in XML

```xml
<otx xmlns="http://iso.org/OTX/1.0.0" xmlns:hmi="http://iso.org/OTX/1.0.0/HMI"
id="id_5b910" name="Document1" package="HelloWorldPackage1" version="1.0.0.0"
timestamp="2021-10-12T21:34:34.0+02:00">
<procedures>
    <procedure id="id_46560" name="main" visibility="PUBLIC">
        <realisation>
            <flow>
                <action name="ConfirmDialog1" id="ConfirmDialog_179dd">
                    <realisation xsi:type="hmi:ConfirmDialog">
                        <hmi:message xsi:type="StringLiteral" value="Hello World" />
                    </realisation>
                </action>
            </flow>
        </realisation>
    </procedure>
</procedures>
</otx>
```

```
[#Name, Document1]
[#Version, 1.0.0.0]
[#Timestamp, 2021-10-12T21:34:34.099+02:00]
namespace HelloWorldPackage1
{
    public procedure main()
    {
        HMI.ConfirmDialog("Hello World");
    }
}
```
OTL

Hello World — OK

Start → ConfirmDialog (Parameters / Message "Hello World") → Stop

Automotive Diagnostic Systems - OTX User Training

emotive
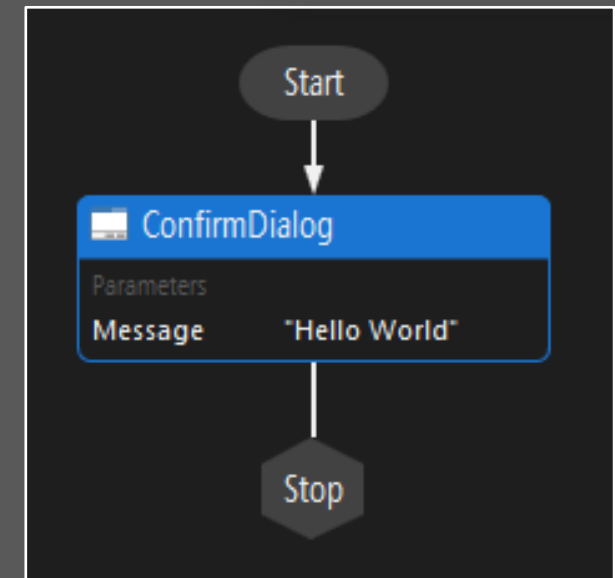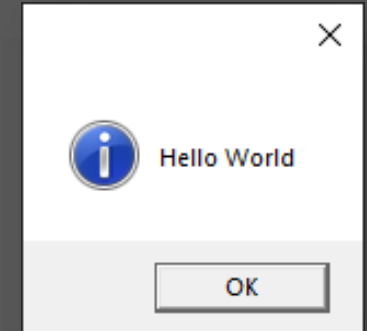
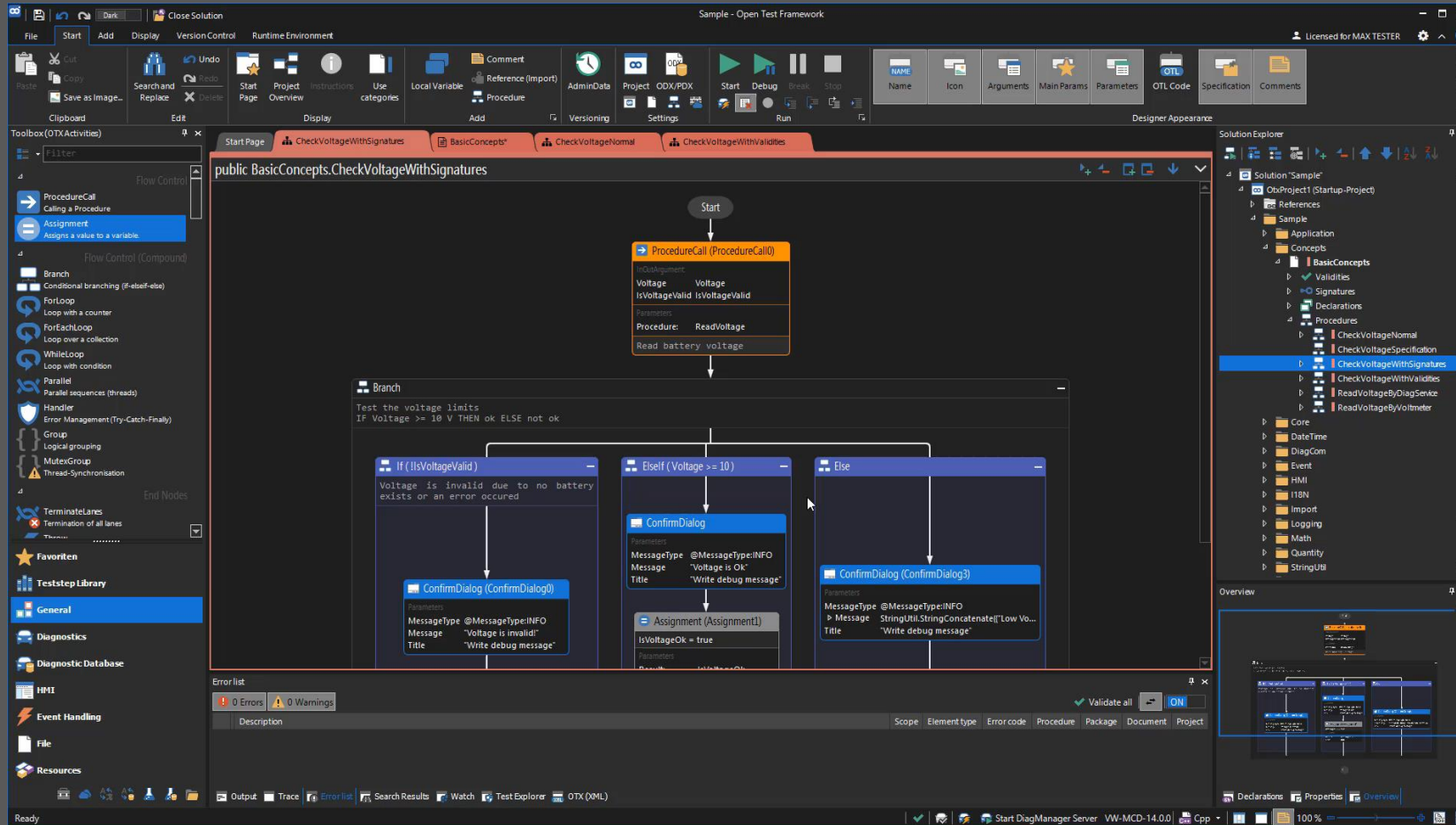# OTX in Process – 1ˢᵗ Specification

- Specification of test logic at the level of a diagnostic expert
- The diagnostic sequences are specified inside OTX single-source
- Role: Component manager (BTV), who does not need to have any programming knowledge
- Due to limited view of OTX
- Result: PTX file

OTX in Process – 2nd Creation

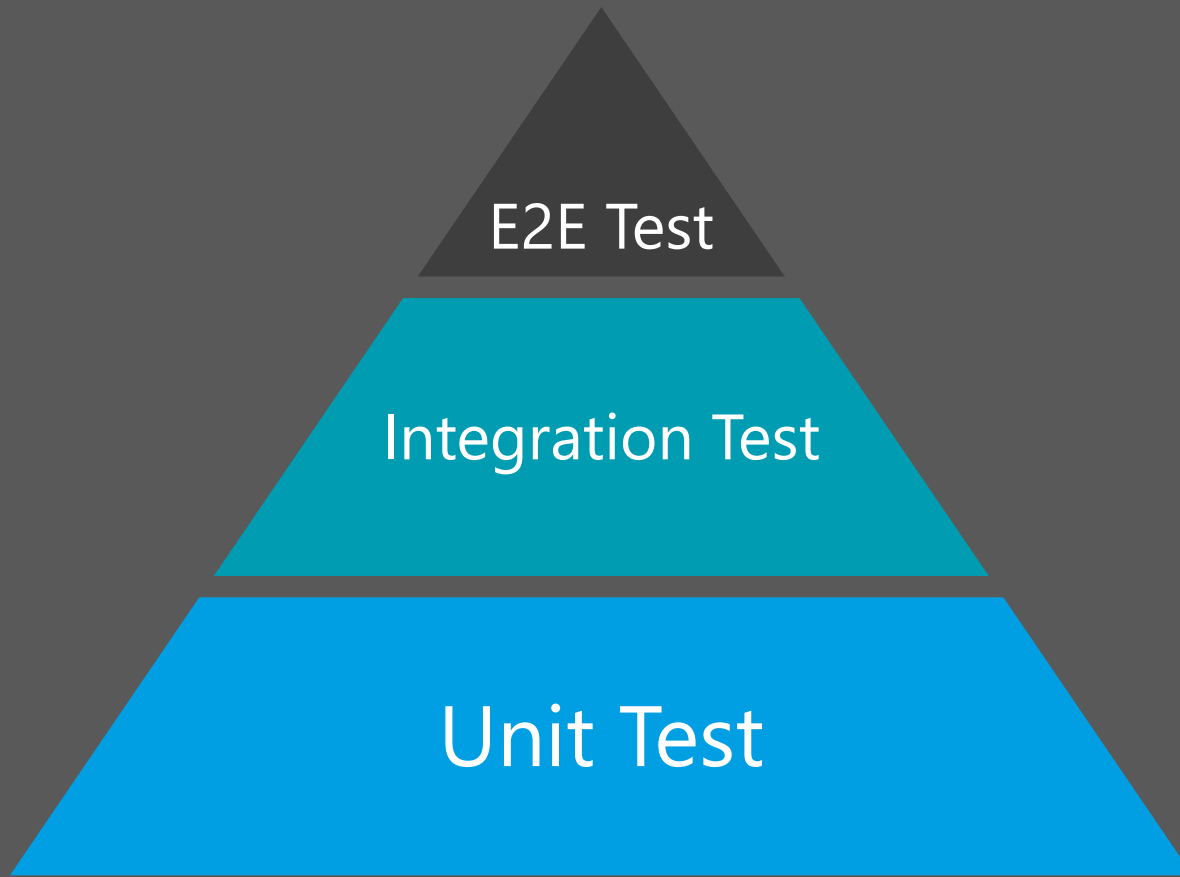- **Specification is implemented**
  - No change in specification!
- **By coding or graphic programming**
- **Role: OTX programmer**

Creation

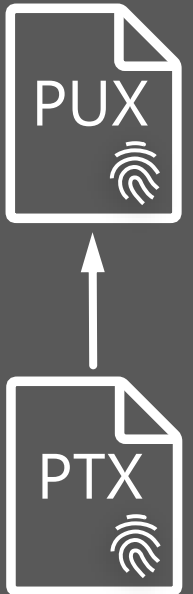# OTX in Process – 3rd Validation

E2E Test

Integration Test
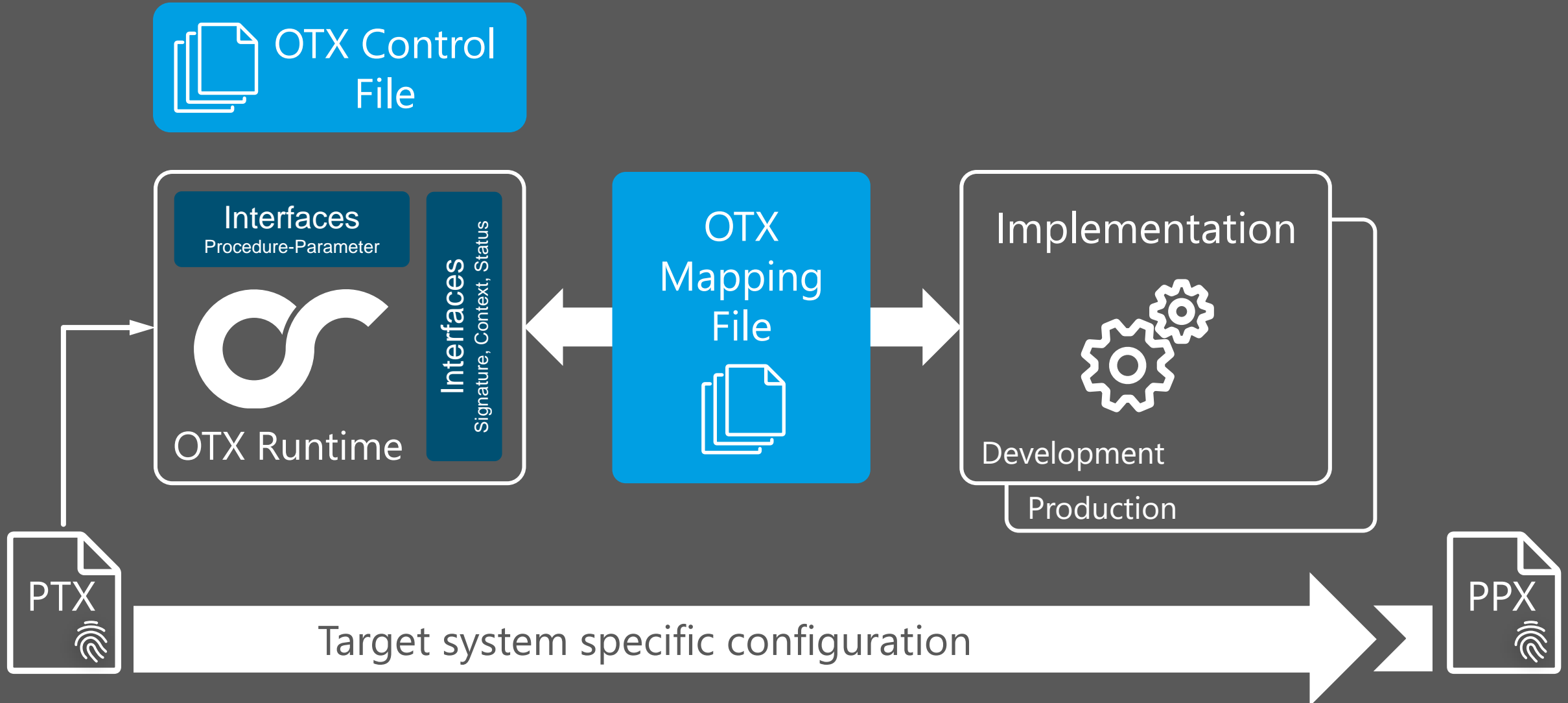
Unit Test

Validation

PUX

PTX

- Protection of the OTX test logic against expected behavior
  - Unit tests
  - Integration testing
- Role: OTX tester
- Result: PUX file
  - Contains executable test project and referenced PTX files

e motive

OTX in Process – 4th Configuration

OTX Control File

Interfaces
Procedure-Parameter

Interfaces
Signature, Context, Status

OTX Runtime

OTX Mapping File
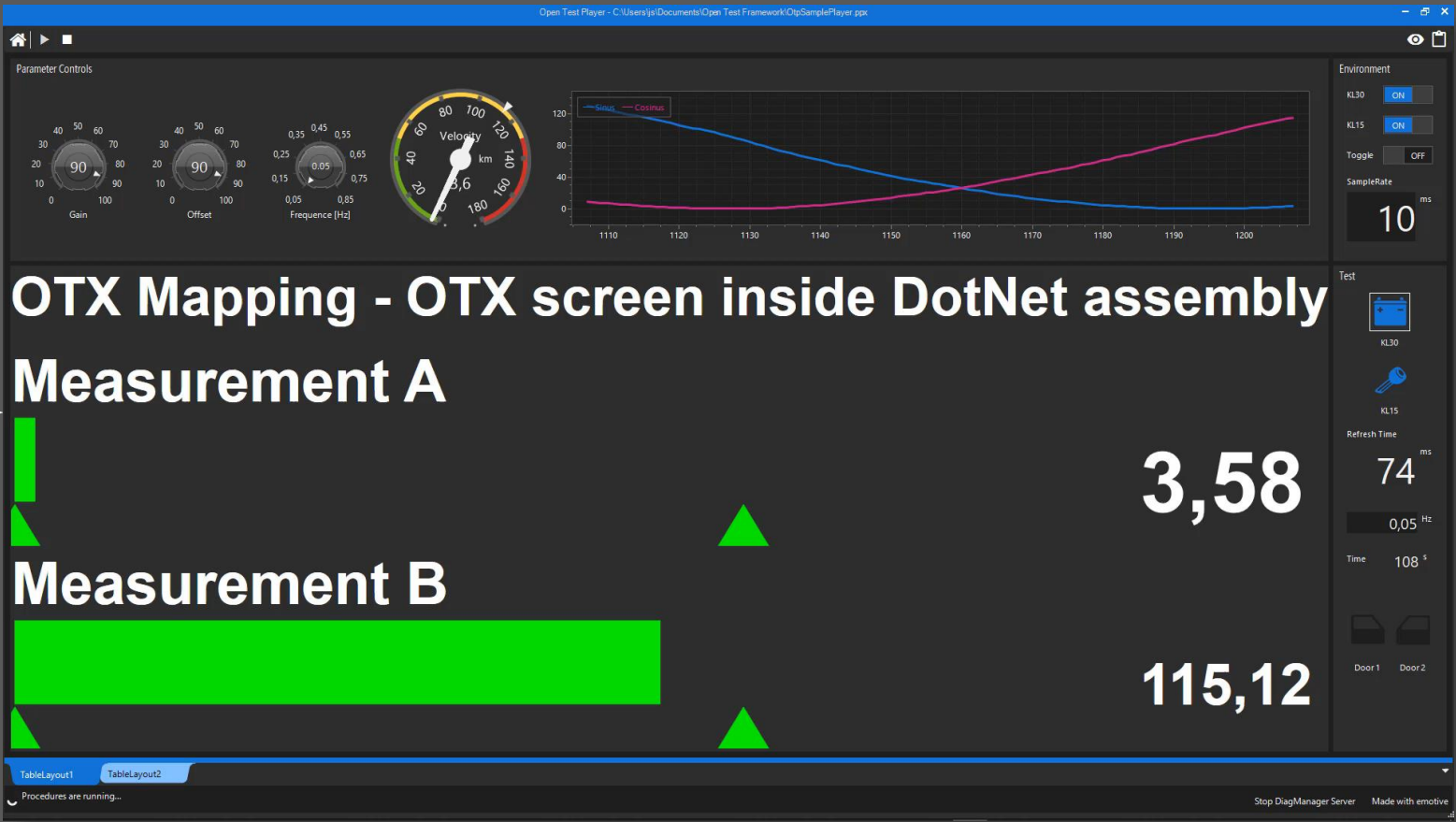
Implementation

Development

Production

PTX

PPX

Target system specific configuration

OTX in Process – 5th Target System Execution

Windows

Linux

Android
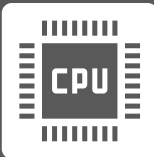
Embedded

Execution on Arbitrary Target Systems

Standard Compliant Exchangeability

Development

Production

After Sales

Customer (Inside the Vehicle)

Automotive Diagnostic Systems - OTX User Training

OTX can ensure that the same unchanged test logic can be executed in any target system at any time and comes to the same results.

OTX Application Areas

Test Bench Control	ECU Setup	Generic Diagnosis
Subjective Assessment	Flashing and Encoding	Service-Oriented Diagnosis*
Exchange of Test Logic	Status Documentation	Scenario-Based Testing*
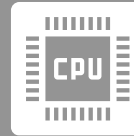
Automotive Diagnostic Systems - OTX User Training

OTX can be extended easily and in compliance with the standard for applications inside new areas outside of vehicle diagnostics.

Automotive Diagnostic Systems - OTX User Training

1/28/2025

Open Test Specification

Open Test Specification

OTX will be the base standard for the new ASAM standard Open Test Specification

OTX is expanded with new extensions for ASAM XIL and the OpenX standards

1/28/2025

emotive

OTX-Extension — Extensibility

Internalization

HMI

EcuConfiguration

DiagConfiguration

Flash+

Bus Monitoring

Diag Com+

DiagDataBrowsing+

ExternalService Provider

Quantities

Event

Event+

Measure

Core

DiagCom

DiagDataBrowsing

Flash

Job

ComInterface

Assertion

ResultHandling

ZIP

File

CommonDialogs

Math

StateVariable

Util

DataType

Black Box

StateMachine*

SQL

XML

Logging

Persistence

Time

String

37	Extensions
87	Actions
555	Terms
38	Data Types
11	Signatures (Interfaces)
150	Checker Rules

Automotive Diagnostic Systems - OTX User Training

Potential new OTX Extensions

UnitTest

Internalization

HMI

EcuConfiguration

DiagConfiguration

Flash+

Bus Monitoring

Diag Com+

DiagDataBrowsing+

SOVD

Range

ExternalService Provider

Quantities

Event

Event+

Measure

Core

DiagCom

DiagDataBrowsing

Flash

Job

ComInterface

OpenX

XIL

Assertion

StateVariable

DataType

ResultHandling

Util

Black Box

StateMachine*

ZIP

File

SQL

XML

CommonDialogs

Logging

Persistence

Math
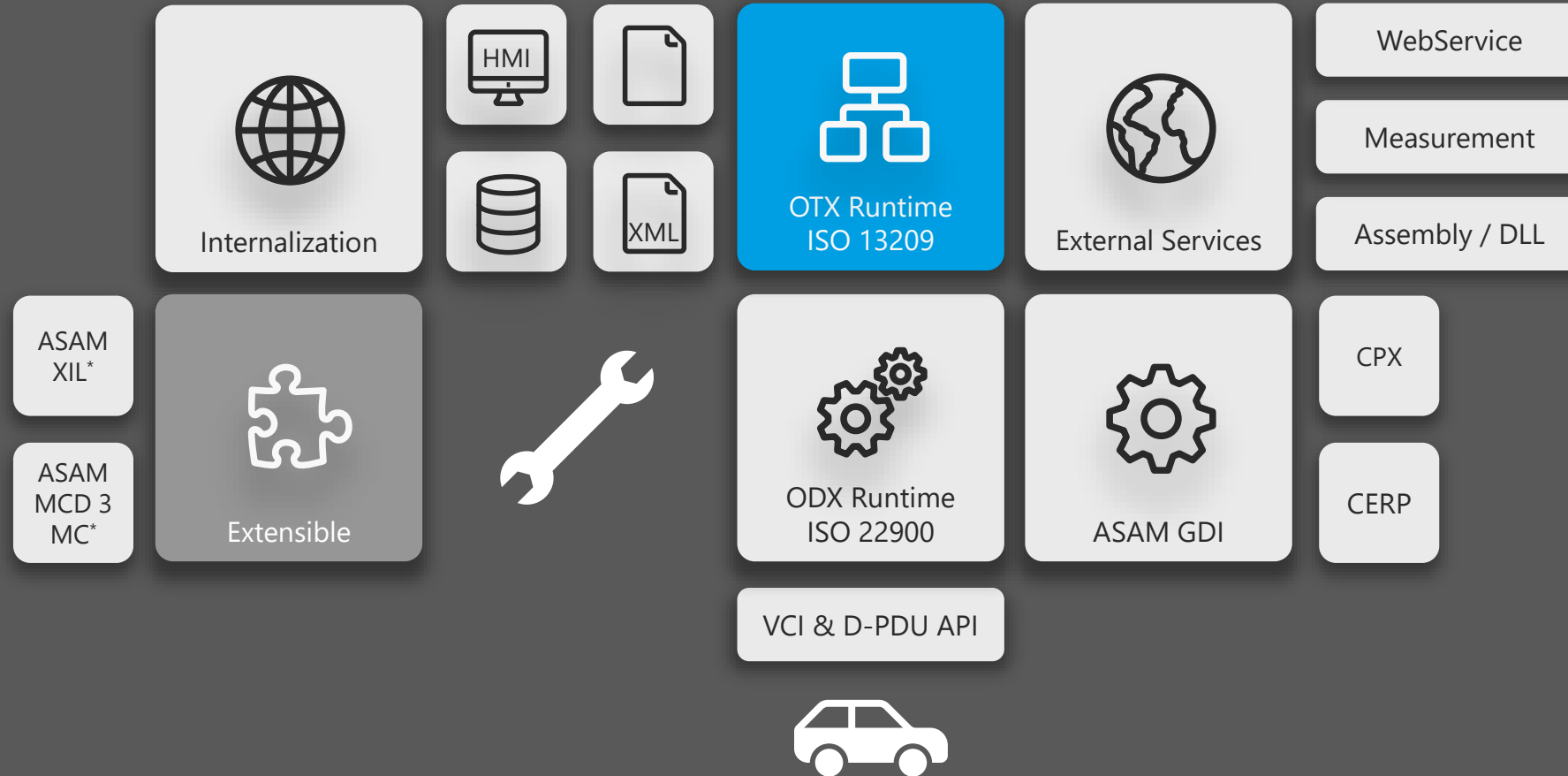
Time

String

AI

e motive

Integration into existing standards
Harmonizing, integrating and connecting standards

ISO — International Organization for Standardization

Internalization

HMI

XML

**OTX Runtime
ISO 13209**

External Services

WebService

Measurement

Assembly / DLL

ASAM XIL*

ASAM MCD 3 MC*

Extensible

ODX Runtime
ISO 22900

ASAM GDI

CPX

CERP

VCI & D-PDU API

emotive

Integration into existing standards
Harmonizing, integrating and connecting standards

Specific Environment	Interface to Environment (GUI, external Devices, Web-Service e.g.)
ISO 13209	Test Logic → OTX
ISO 22900-3 ISO 22901-1	Diagnostic Runtime System (MVCI-Server or D-Kernel) → ODX
ISO 22900-2	VCI Driver (D-PDU API)
ISO 14229 ISO 11898	Vehicle Communication Interface (VCI)
	ECUs

Integration into existing standards
Standardized Diagnostic Runtime System MVCI

4.

Cooling water temperature = 64 °C

How is the PDU converted to temperature?

Diagnostic Application

API

1.

What is the cooling water temperature?

What is the PDU for reading out the temperature?

D-Server API, MCD 3 (ISO 22900-3)

Return Temperature

ODX, MCD 2 (ISO 22901-1)

Modular Vehicle Communication Interface
(MVCI, ISO 22900)

ODX, MCD 2 (ISO 22901-1)

Request via ShortName

ODX/PDX
ECU diagnostic data description

Handover the PDU

ODX/PDX
ECU diagnostic data description

Return of PDU

D-PDU API, MCD 1 (ISO 22900-2)

Unpack the PDU from the response message

Vehicle Communication Interface – VCI

Bus configuration and Packaging the PDU into a message

Receiving a message on the bus (Response)

Bus Systems
ISO 11898 etc.

and Protocols
ISO 14229, ISO 15765 etc.

Sending a message on the bus (Request)

3.

ECU

2.

*PDU = **P**rotocoll **D**ata **U**nit

Automotive Diagnostic Systems - OTX User Training

OTX Standardization

- Since 2009 OTX standardization with approx. 10 man-years of expert knowledge in the field of testing (diagnostics) in the automotive industry

> **Excerpt from ISO 13209-1**
> This international standard specifies a standardized, tester-independent, XML-based data exchange format for the documentation and formal description of diagnostic test sequences. The format serves to support the requirements of transferring diagnostic test sequence logic between electronic system suppliers, vehicle manufacturers and service dealerships/repair shops.
>
> **General considerations**
> 1. Improvement of documentation quality
> 2. Refinement of diagnostic authoring processes
> 3. Achieving long-term availability of test sequences (Replace Java Jobs)
> 4. Setting up a uniform process chain
> 5. …

- Practical use since 2015

- Attendees:

Automotive Diagnostic Systems - OTX User Training

OTX Standardization
ISO 13209

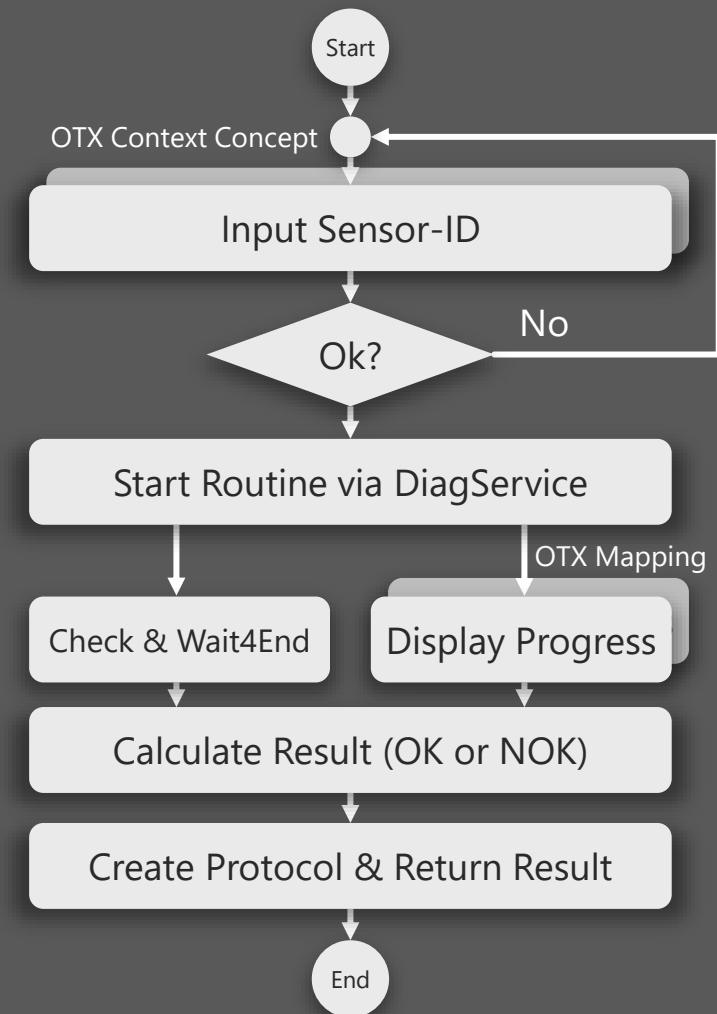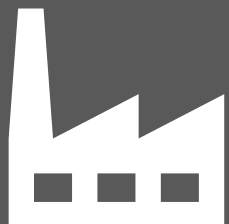|		
ISO 13209-1:2012	Part 1	Describes the motivation and objective of the standard as well as use cases
ISO 13209-2:2022 2nd Edition	Part 2	Core data model
ISO 13209-3:2022 2nd Edition	Part 3	13 + 1 Extensions e.g. for vehicle diagnostics, MMI etc.
ISO 13209-4:2021	Part 4	23 further extensions

Use Case Example

Tire Pressure Sensors Calibration (simplified)

1. Enter sensor ID for different environments
   - Production: Automated via test system
   - Development: Automated via barcode scanner
   - After Sales: Manual
2. Input validation
3. Start routine for calibrating the sensor
4. Monitoring routine (polling)
5. Parallel visualization calibration process
   - Production: Screen in the test system
   - Development: Generic Screen
   - After-sales: Screen in the infotainment system
6. Calculate result (OK or NOK)
7. Logging and output of the result

Start

OTX Context Concept

Input Sensor-ID

Ok? — No

Start Routine via DiagService

OTX Mapping

Check & Wait4End | Display Progress

Calculate Result (OK or NOK)

Create Protocol & Return Result

End

Automotive Diagnostic Systems - OTX User Training

emotive

Use Case Example

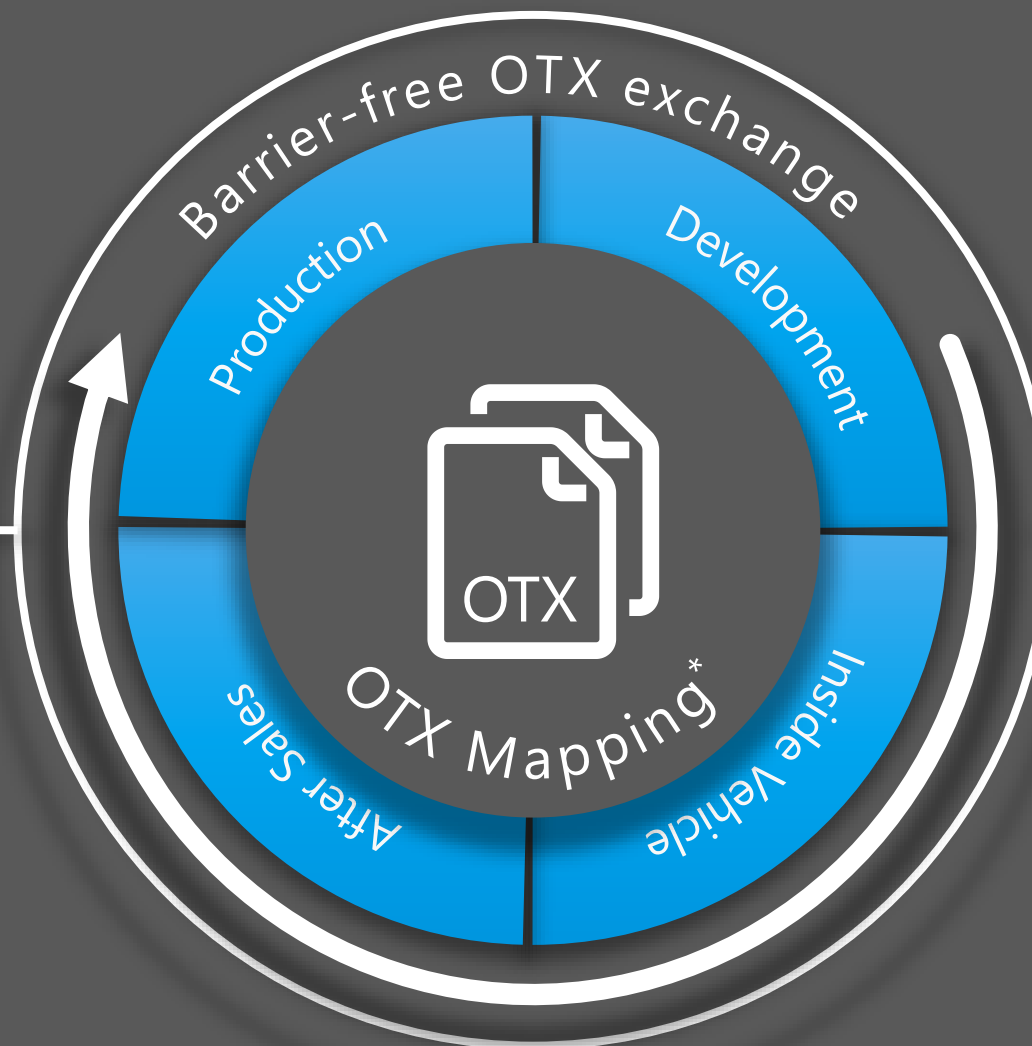Cross-departmental, barrier-free exchange of test knowledge

Production
Enrichment for plant-specific process environments

After Sales
Enrichment and adaptation for the control units commissioning

Development
Concept design and initial creation from OTX building blocks

Inside Vehicle
Enrichment and adaptation to vehicle-specific scopes

Barrier-free OTX exchange

Production

Development

After Sales

Inside Vehicle

OTX Mapping*

OTX

* Anpassung an unterschiedliche Umgebungen

Automotive Diagnostic Systems - OTX User Training

1/28/2025

e motive

OTX is an open domain specific programming language (DSL) for process reliable description of exchangeable and executable test logic inside automotive industry.

OTX can ensure, that the same unchanged test logic can be executed on any system at any time and comes to the same results.

Automotive Diagnostic Systems - OTX User Training

Content
OTX User Training

1. Introduction OTX
2. Basic working with OTX
3. OTX projects
4. OTX tools
5. ControlFile & OTX mapping
6. OTX mapping editor
7. Player projects
8. OTX as programing language
9. Advanced Basics OTX
10. Advanced Basics OTF
11. OTX runtime environment
12. Advanced basics OTX-Runtime