INTERNATIONAL STANDARD

ISO 13209-1

First edition
2011-11-01

Road vehicles — Open Test sequence eXchange format (OTX) —

Part 1: **General information and use cases**

Véhicules routiers — Format public d'échange de séquence-tests (OTX) —

Partie 1: Information générale et cas d'utilisation

Reference number
ISO 13209-1:2011(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO 13209-1 was prepared by Technical Committee ISO/TC 22, *Road vehicles*, Subcommittee SC 3, *Electrical and electronic equipment*.

ISO 13209 consists of the following parts, under the general title *Road vehicles — Open Test sequence eXchange format (OTX)*:

— *Part 1: General information and use cases*

— *Part 2: Core data model specification and requirements*

— *Part 3: Standard extensions and requirements*

Introduction

Diagnostic test sequences are utilized whenever automotive components or functions with diagnostic abilities are being diagnosed, tested, reprogrammed or initialized by off-board test equipment. Test sequences define the succession of interactions between the user (i.e. workshop or assembly line staff), the diagnostic application (the test equipment) and the vehicle communication interface as well as any calculations and decisions that have to be carried out. Test sequences provide a means to define interactive, guided diagnostics or similar test logic.

Today, the automotive industry mainly relies on paper documentation and/or proprietary authoring environments to document and to implement such test sequences for a specific test application. An author who is setting up engineering, assembly line or service diagnostic test applications needs to implement the required test sequences manually, supported by non-uniform test sequence documentation, most likely using different authoring applications and formats for each specific test application. This redundant effort can be greatly reduced if processes and tools support the OTX (Open Test sequence eXchange) concept.

ISO 13209 proposes an open and standardized format for the human- and machine-readable description of diagnostic test sequences. The format supports the requirements of transferring diagnostic-test-sequence logic uniformly between electronic system suppliers, vehicle manufacturers and service dealerships/repair shops.

ISO 13209 (also referred to as the OTX standard) is comprised of three parts:

Part 1: General information and use cases

> This provides a general overview over the individual parts. It documents use cases that were considered during the standardization and which are derived from real world scenarios as found in the automotive industry. It also provides the rationale for proposing the OTX standard, explaining the considerations that went into the development of that standard and giving an overview of the structure of the concepts and documents related to ISO 13209.

Part 2: Core data model specification and requirements

> This provides the data model specification of the core part of the OTX test sequence description language in the form of UML design diagrams, XML Schema definitions and descriptive documentation. The core describes the basic structure underlying every OTX document. This comprises detailed data model definitions of all required control structures by which test sequence logic is described, as well as definitions of the outer, enveloping document structure in which test sequence logic is embedded. A tool implementing the OTX standard has to implement all definitions within Part 2 to be considered compliant with the OTX standard.

Part 3: Standard extensions and requirements

> This provides specifications for ubiquitous functionalities that may, to various extents, be used by every OTX-based environment. The core data model extensions defined in Part 3 makes use of the extension mechanisms provided by the OTX language to provide interface definitions for feature sets like HMI (Human-Machine Interface), internationalization or diagnostic vehicle communication. A tool implementing the OTX standard does not have to implement all (or any) of the extension definitions within Part 3 to be considered compliant with the OTX standard.

Road vehicles — Open Test sequence eXchange format (OTX) —

Part 1:
General information and use cases

1 Scope

This part of ISO 13209 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.

This part of ISO 13209 provides an introduction to the rationale behind ISO 13209. It gives an overview of the document set and structure along with the use case definitions and a common set of resources (definitions, references) for use by all subsequent parts.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 14229 (all parts), *Road vehicles — Unified diagnostic services (UDS)*

ISO 22900 (all parts), *Road vehicles — Modular vehicle communication interface (MVCI)*

ISO 22901 (all parts), *Road vehicles — Open diagnostic data exchange (ODX)*

3 Terms, definitions and abbreviated terms

3.1 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

3.1.1
aftermarket
part of the automotive industry concerned with manufacturing, remanufacturing, distribution, retailing and installation of all vehicle parts, chemicals, tools, equipment and accessories for light and heavy vehicles, after the sale of the automobile by the original equipment manufacturer (OEM) to the consumer

3.1.2
after sales
after-sales department
department of an automotive OEM which is concerned with the distribution, retailing, servicing, repair and installation of vehicles

3.1.3
engineering
engineering department
department of an automotive OEM which is concerned with the design, development, integration and testing of vehicles

3.1.4
manufacturing
manufacturing department
department of an automotive OEM which is concerned with the production and end-of-line testing of vehicles

3.1.5
original equipment manufacturer
OEM
automotive company that engineers, manufactures, sells and services vehicles

3.1.6
OTX Core
most generic and stand-alone part of the overall OTX data model which describes the basic structure underlying every OTX document and comprises detailed data model definitions of all required control structures (loops, branches, etc.) by which test sequence logic is described, and definitions of the outer, enveloping document structure in which test sequence logic is embedded

3.1.7
OTX Extension
OTX Standard Interface Definition
otxIFD
set of OTX data type-, action-, term- and signature-definitions that are tailored for a specific area of application and that are defined outside of the OTX Core

NOTE OTX Extension model data types, actions and terms needed for communication with systems are usually hidden behind diverse interfaces (e.g. a MVCI, HMI, GDI, etc.). Through these interfaces, calls can be performed to external systems whose internal behaviours do not have to be known to the (client) OTX test sequence/runtime. The system-side interface (server-side) can be proprietary because the adapter design pattern is applied.

3.1.8
test procedure
procedure
stand-alone, configurable flow of OTX actions that can be executed separately by a diagnostic application or be called from other OTX procedures

3.1.9
test sequence
main procedure
test procedure defining a full test

NOTE A test sequence is a procedure, but not all procedures are test sequences. By using procedures, a test sequence can be split into several procedure modules. An adequately assembled set of frequently needed procedures can serve as a library which provides procedures that are callable from any other (client) procedure or test sequence.

3.2 Abbreviated terms

API	Application programming interface
CM	Configuration management
GDI	Generic device interface
HMI	Human-machine interface
JRE	Java runtime environment
MVCI	Modular vehicle communication interface
OEM	Original equipment manufacturer
OTX	Open test sequence eXchange

UML	Unified modeling language
XML	Extensible markup language
XSD	XML Schema Definition

4 Document overview and structure

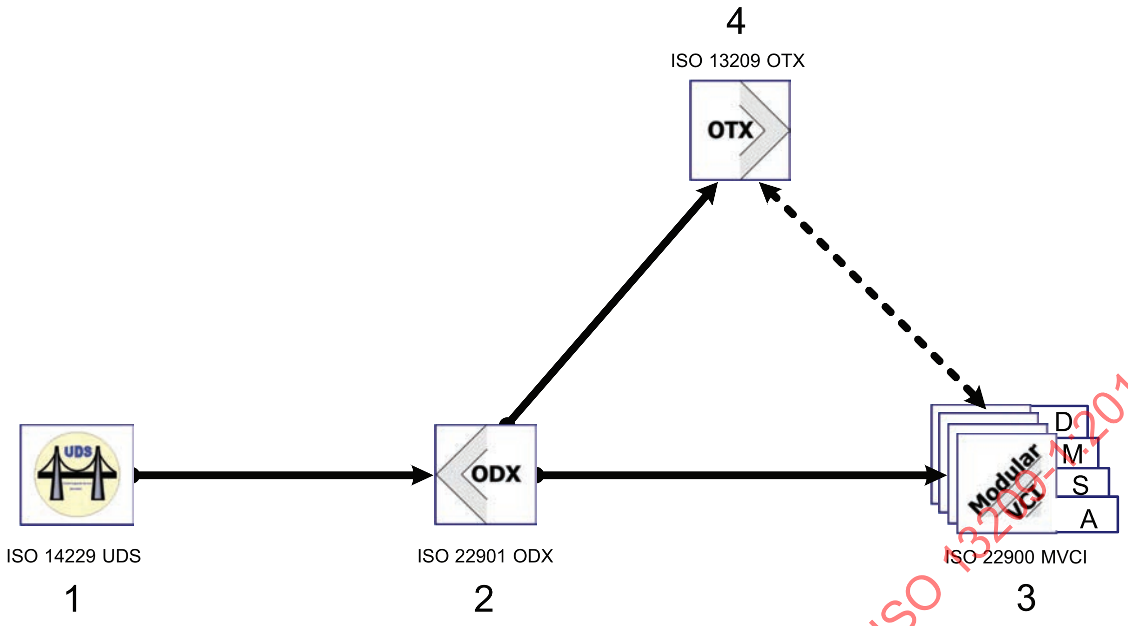
This part of ISO 13209 is structured into three main sections.

- Clauses 1 to 3 and the Introduction provide an overview of the concepts that ISO 13209 aims to cover, define the scope of what is being standardized and provide the basic terms, definitions and abbreviated terms.
- Clause 5 details the general considerations that went into the standardization effort, illustrates usage scenarios on a high level and introduces the design of OTX-based processes and systems.
- Clauses 6 and 7 define the use cases that were considered during the development of the OTX standard.

5 General considerations

5.1 Integration of OTX with existing standards

An overview of how OTX fits into the structure of existing diagnostic ISO standards is given in Figure 1. Please note that although this overview depicts OTX in the context of ISO 22900 (MVCI) and ISO 22901 (ODX) standards, **OTX is by no means designed to be used only in conjunction with these standards**. The use of any other equivalent symbolic data description is also supported. It is an explicit design goal of OTX that the OTX data model can be used within any system context.



Key

1 ISO 14229 (all parts), *Road vehicles — Unified diagnostic services (UDS)*

2 ISO 22901 (all parts), *Road vehicles — Open diagnostic data exchange (ODX)*

3 ISO 22900 (all parts), *Road vehicles — Modular vehicle communication interface (MVCI)*

4 ISO 13209 (all parts), *Road vehicles — Open Test sequence eXchange format (OTX)*

D OEM development

M OEM manufacturing

S OEM service/aftersales

A aftermarket

solid arrows “used by”

dotted arrows “interacts with”

Figure 1 — Integration of OTX with existing standards

5.2 Improvement of documentation quality

At present, documentation of diagnostic test sequences is done in various ways. Documentation exists in the form of text documents, flow charts, tables, etc. The transporting media are non-uniform paper printouts or computer files/database entries that are based on proprietary formats. Moreover, some of the diagnostic test sequence knowledge is transported by verbal communication only.

Since neither the storage format nor the human-readable presentation is standardized, test sequences have to be re-implemented frequently; the various media used for documenting the related knowledge lead to misunderstandings and interpretation gaps. Even if the content (the diagnostic expert knowledge) of documentation is of high quality, the quality and usability of the actual documentation tend to be rather poor.

One of the goals of ISO 13209 is to provide a formal, machine-readable, uniform documentation format. OTX shall also be designed for easy visualization of diagnostic test sequences. This allows for providing diagnostic test sequences in a formal, machine-readable format that can at the same time be easily visualized to be comprehensible to human readers.

5.3 Refinement of diagnostic authoring processes

Diagnostic test sequences exist in a lot of different formats produced by various proprietary authoring tools. This results in a tight interlocking of authoring tools, data formats and runtime applications. The effect is poor to non-existent interchangeability of authoring tools, diagnostic applications and of diagnostic test sequences themselves. As a consequence, there is a high degree of redundancy within OEM processes, as test sequences have to be re-modelled multiple times to make them work with different diagnostic applications, e.g. in engineering, manufacturing or after sales. This reduces productivity and, at the same time, quality.

As shown in Figure 2, OTX is a means to improve the exchangeability of diagnostic test sequences. It enables OEMs to base their OTX-specific processes on the single source principle, where OTX sequences are the root of the diagnostic-test-sequence-authoring process starting at the supplier and extending through engineering, manufacturing and after sales to aftermarket and legislative applications.

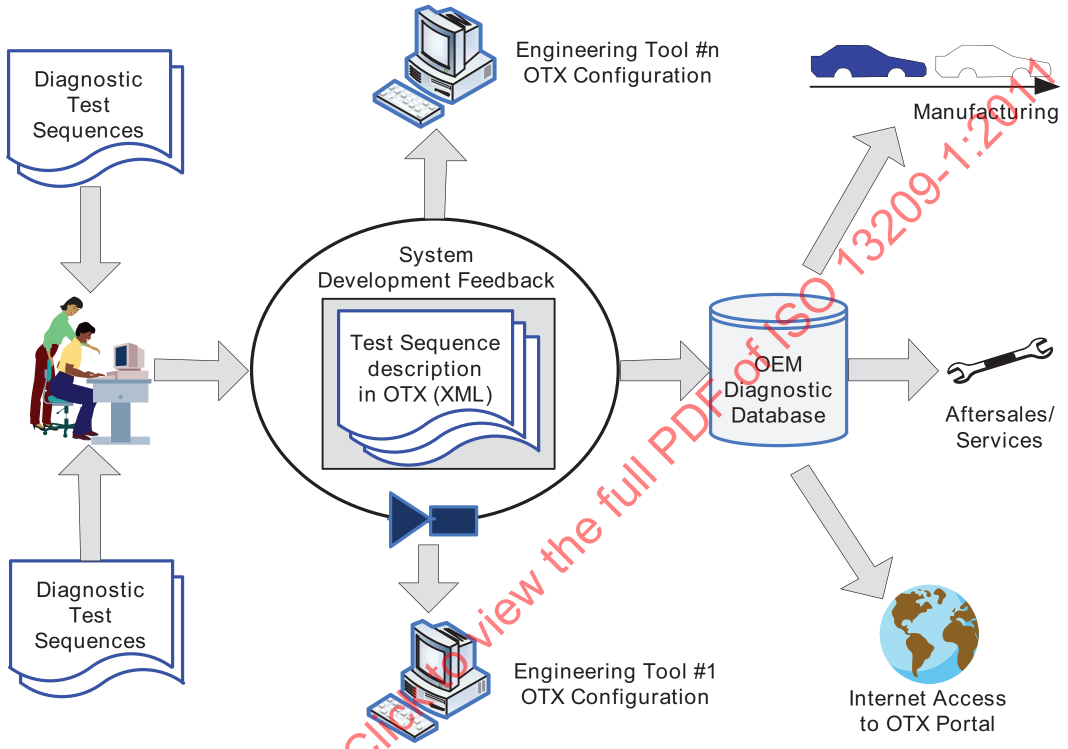

Diagnostic Test Sequences
Engineering Tool #n OTX Configuration
Manufacturing
System Development Feedback
Test Sequence description in OTX (XML)
OEM Diagnostic Database
Aftersales/ Services
Diagnostic Test Sequences
Engineering Tool #1 OTX Configuration
Internet Access to OTX Portal

Figure 2 — Usage of OTX data in the ECU life cycle

Although the adaptation of a test sequence to a specific runtime context might require a certain amount of manual implementation work in some cases, the OTX diagnostic test sequence format makes it possible to automatically convert OTX documentation into proprietary data formats. This can be done by implementing converters and code generators for target authoring tools and diagnostic applications.

5.4 Achieving long-term availability of test sequences

Because of the reasons outlined in 5.3, automotive OEMs are often tied to their proprietary authoring-system/data-format/diagnostic-application tool chain. Vast diagnostic test sequence libraries that were accumulated over years eventually become unusable when migrating to a new diagnostic application that doesn’t support the old, proprietary sequence formats. Being a standardized data format, OTX offers a higher degree of independency from proprietary tools. At the same time, suppliers are freed from inventing and maintaining customer-specific solutions for the representation of the test sequences that are used by their tools, enabling them to provide OTX-based products instead of customer-tailored solutions. Companies competing on this level can focus on providing real added value for their authoring and runtime product offerings, the differentiator being tool features, usability aspects and integration concepts rather than subtle intricacies of highly customer-specific peculiarities of underlying data formats.

A rather specific example where OTX improves the availability of test sequences is the Java job concept that is part of the ISO 22900/ISO 22901 standards. ODX Java jobs are used to describe vehicle-oriented diagnostic sequences that are executed by a MVCI D-server component. To this end, Java jobs use the D-Server API standardized by ISO 22900. Since the D-Server API is subject to change and since there is no guarantee that successive versions of java runtime environments (JRE) used by D-server components support java byte code generated for previous JRE-versions, long-term compatibility of Java job code is not assured. This issue is of particular interest to OEM after-sales departments that have to provide long-term support for their vehicles in the field.

As the OTX standard provides an XML description of diagnostic test sequences, and as XML (derived from ISO 8879 SGML; forms a subset of the SGML standard) is fully platform- and runtime-environment independent, the Java job compatibility issue described above can be alleviated. OTX sequences that are the source for Java job business logic shall use the D-Server API (ISO 22900-2) via an abstraction layer which increases the independence of test sequences from changes in the D-Server API.

5.5 Setting up a uniform process chain

This section introduces a recommended process to be implemented to exchange diagnostic test sequences across development, production and service/after-sales departments.

With the increasing complexity of electric/electronic systems in road vehicles, diagnostic requirements grow at the same rate. Diagnostic test sequences are one major part of this emerging complexity, creating the need for

- a uniform way to document diagnostic test sequences in development, manufacturing and after-sales departments that are reusable and available in the long term,
- a formal, machine-readable test sequence storage format that supports user-friendly, graphical authoring environments for both viewing and manipulation of test sequences, and
- exchangeability of diagnostic test sequences between OEM-internal business units, suppliers and other OEMs.

To help address these requirements, OTX provides a common ground for data exchange processes between system suppliers, OEM departments (engineering, manufacturing, after sales/service) and the independent aftermarket automotive service industry.

The process example given in Figure 3 shows the common source concept of test sequence data, including a verification and feedback mechanism with distribution to end users. Engineering, manufacturing and after-sales/service departments specify which test sequences are required to diagnose the vehicle components and functions. This information is documented in a structured format utilizing the OTX standard and an appropriate OTX authoring system/tool. Once all quality goals are met the OTX file(s) is published to an OEM database. Diagnostic test sequence information is now available to manufacturing, service, OEM franchised dealers and aftermarket repair shops via intranet and internet channels.

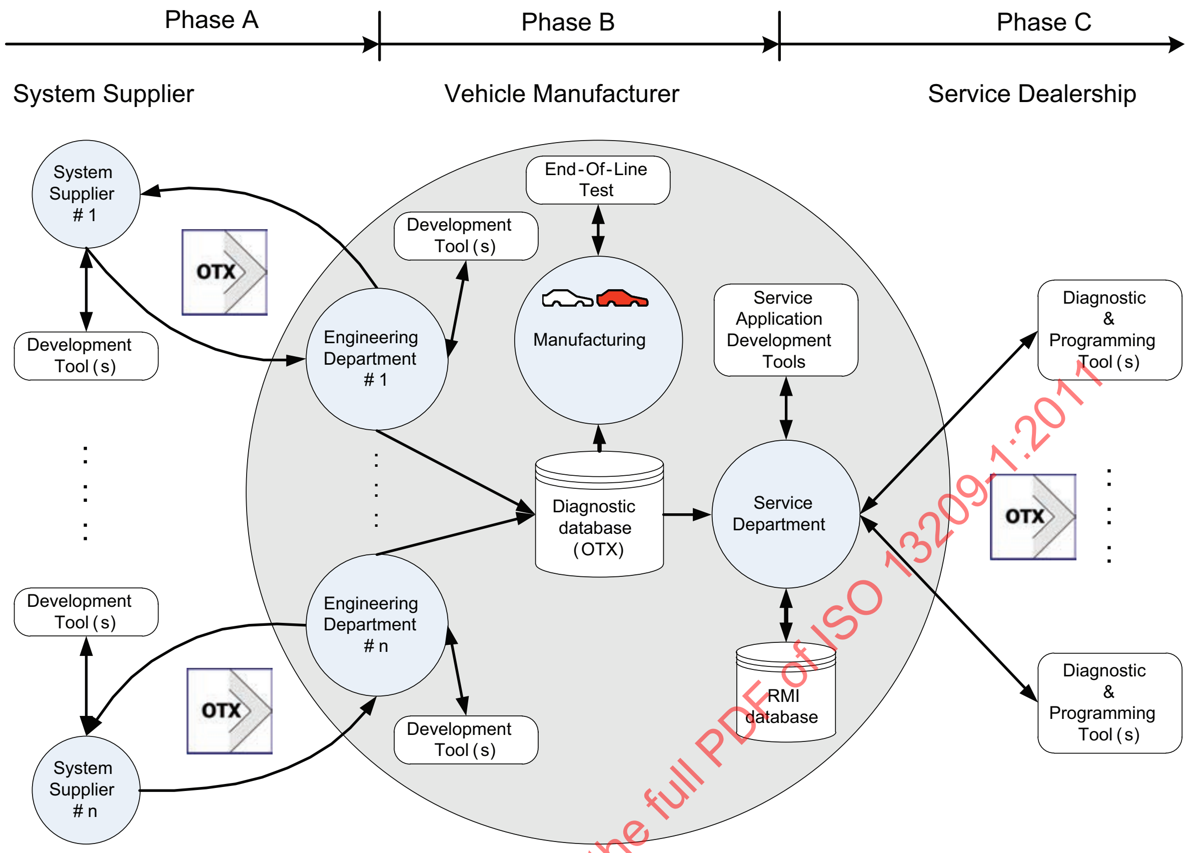


Figure 3 — Example of an OTX process chain

5.6 OTX authoring and impact on Modular VCI software architecture

5.6.1 Non-OTX-compliant authoring

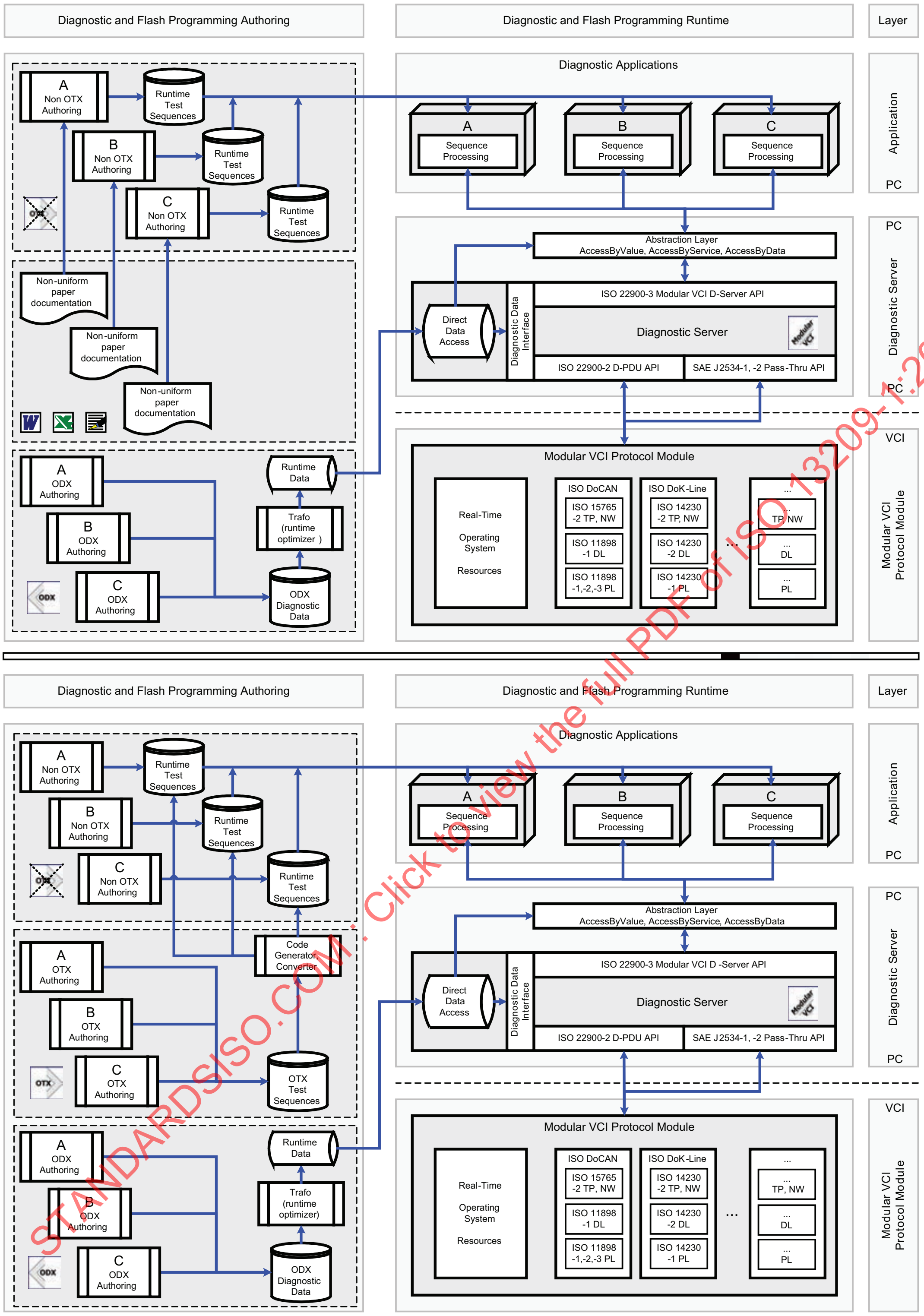
The upper diagram in Figure 4 illustrates the current process of diagnostic test sequence authoring and its use in diagnostic applications in combination with an ISO 22900 Modular VCI compliant runtime system framework.

Tool-manufacturer-specific non-OTX-compliant authoring tools are used to generate runtime test sequences for their specific runtime sequence processors (A, B and C). The content of the test sequence specification usually derives from non-uniform paper documentation.

5.6.2 OTX-compliant authoring

OTX sequences serve as a formal, uniform documentation for diagnostic test sequences. As such, they may complement or even replace the heterogeneous prose documentation which is usually used for test sequence description and documentation.

Refer to the diagrams in Figure 4 which illustrate how to expand the authoring process in order to support a single source test sequence on process using the OTX exchange format.



Key

A, B, C represent tool-manufacturer-specific authoring systems and runtime formats

Figure 4 — Non-OTX-compliant authoring versus OTX-compliant authoring

5.6.3 OTX scenarios

There are several coexisting scenarios (two are shown in the above figures) which are compliant to the OTX standard:

— **Scenario A — Non-OTX compliant tools**

Proprietary test sequence authoring tools can import and extend/adapt single source OTX test sequences. This shall be possible through the implementation of "OTX-to-proprietary" converters or code generators. These allow using one single OTX test sequence as the source for implementation of the sequence in multiple proprietary diagnostic applications. In this scenario, today's diagnostic applications and authoring tools do not need to be changed in order to support OTX.

— **Scenario B — OTX-compliant tools**

Future diagnostic applications can be designed to support OTX directly. This results in the need for an OTX interpreter engine in the runtime application itself. This scenario does not require a proprietary authoring tool for creating diagnostic test sequences anymore.

— **Scenario C — Mixed operation of OTX-compliant and legacy tools**

Obviously, it is possible to mix scenarios A and B, with some legacy tools requiring OTX-to-proprietary converters, while others natively support OTX. Support for legacy data formats and OTX sequences at the same time might even be directly supported by a single tool, to allow interaction of OTX sequences with existing, more specialized functionality. It is very likely that this is the scenario that will be implemented by most OEMs, to allow for a smooth migration to an OTX-based diagnostic process.

5.7 OTX-based runtime system architecture

As has already been discussed in previous sections, the OTX data model is structured into a language core and various functional extensions. It is important to note that the OTX Core, as well as the extensions defined by ISO 13209, do only provide interface definitions for the elements of test sequences. The detailed definition of actual runtime behaviour is not part of the OTX standard. This is necessary to allow for the necessary freedom when implementing OTX runtime components to fit specific use cases. For example, it would be impractical for the OTX standard to define the specific implementation of an HMI library, as the requirements and backend frameworks for HMI functionality differ for each test environment.

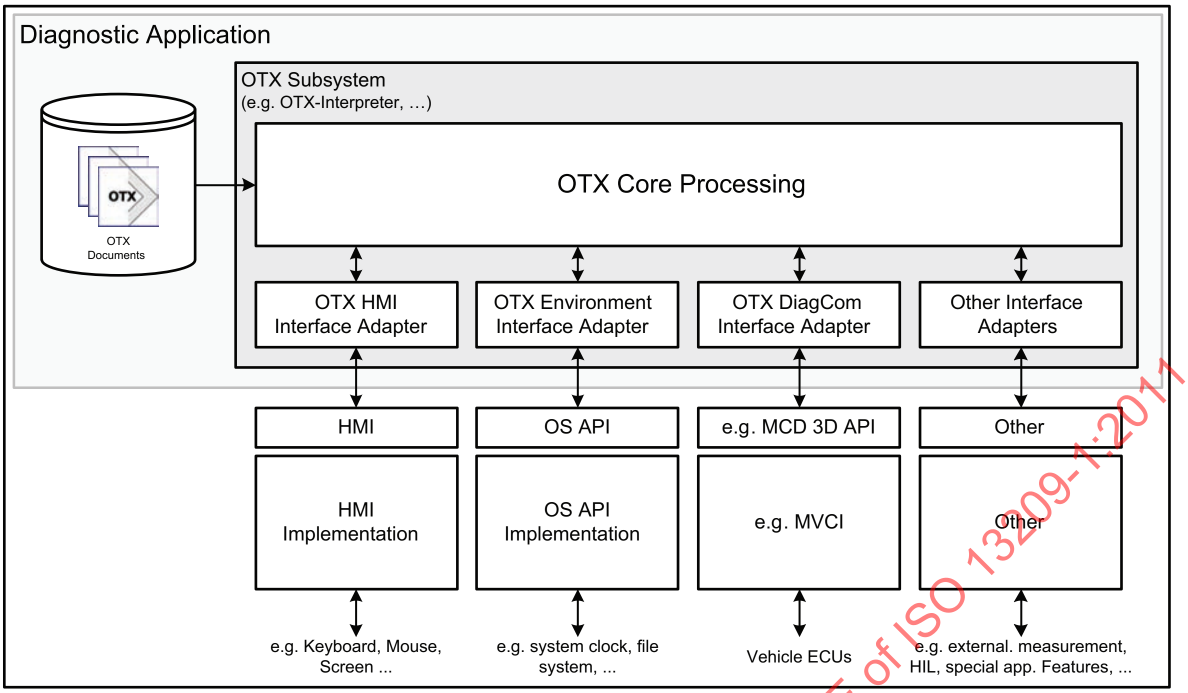

Diagnostic Application
OTX Documents
OTX Subsystem
(e.g. OTX-Interpreter, …)
OTX Core Processing
OTX HMI Interface Adapter
OTX Environment Interface Adapter
OTX DiagCom Interface Adapter
Other Interface Adapters
HMI
OS API
e.g. MCD 3D API
Other
HMI Implementation
OS API Implementation
e.g. MVCI
Other
e.g. Keyboard, Mouse, Screen ...
e.g. system clock, file system, ...
Vehicle ECUs
e.g. external. measurement, HIL, special app. Features, ...

Figure 5 — OTX-based runtime system architecture

This separation of OTX structure (intent) and runtime implementation (execution) allows for OTX to be used in the manner envisioned in the previous clause: as a test sequence exchange format that can be deployed throughout an entire organization, or even across company boundaries. Figure 5 illustrates the architecture of an OTX runtime tester application. The OTX Subsystem that is part of the tester application is interpreting OTX sequences which make use of OTX Core data model constructs, as well as of various extension interfaces (HMI, system-specifics, vehicle communications and other proprietary functionality).

The actual runtime system implementation knows how to map e.g. a HMI extension action within an OTX sequence to the GUI framework that is used by the tester application. For example, an after-sales tester application might map an OTX “User Choice” construct to a Java Swing dialog entity that allows interaction with the workshop mechanic. The same OTX sequence could also be executed on a manufacturing tester station, where the tester application implementation would map the “User Choice” construct to a state machine connected to manufacturing line measurement equipment instead of a GUI interface.

5.8 OTX benefit examples

5.8.1 ECU system supplier

The following benefits are applicable to ECU system suppliers:

- documentation can be generated from OTX/XML test sequence exchange format;
- development testers can be automatically configured to verify ECU diagnostic behaviour;
- the OTX/XML data format provides machine-readable information to import into supplier test sequence database;
- test sequences for a test-equipment-specific application framework can be automatically generated.

5.8.2 Vehicle manufacturer engineering

The following benefits are applicable to vehicle manufacturer engineering departments:

- reduction of test sequence authoring effort;

- various development testers can be supported with “single source” test sequences;
- one single file format for import and export to/from OTX database.

5.8.3 Vehicle manufacturer production

The following benefits are applicable to vehicle manufacturer production departments:

- reduced effort for test sequence verification, because verification needs to be performed only once;
- reuse of verified test sequences;
- fewer errors because of fewer manual process steps;
- manufacturing tester can use the same basic test sequences as engineering diagnostic tester.

5.8.4 Vehicle manufacturer service departments and repair shops

The following benefits are applicable to vehicle manufacturer service departments and repair shops:

- more convenient reuse of verified test sequences;
- lower cost to distribute test sequences;
- “pull” (via Intranet/Internet) test sequences, for example from a portal into a diagnostics application versus “push” (e.g. send CD ROMs);
- one single format for various diagnostic service systems.

5.8.5 Test equipment manufacturer

The following benefits are applicable to test equipment manufacturers:

- less effort needed to implement high-quality scan tool software by using a standardized test sequence approach;
- focus on “rich diagnostic application(s)” versus “bits & bytes”;
- more convenient reuse of test sequences verified by vehicle manufacturers.

5.8.6 Franchise and aftermarket dealerships

The following benefits are applicable to authorized and independent repairers:

- more convenient reuse of test sequences verified by vehicle manufacturers;
- tester configuration by data download instead of software modification;
- download on demand versus buying tester software update upfront.

5.8.7 Legal authorities

The following benefits are applicable to legal authorities:

- standardized description format for enhanced test sequences documentation;
- enablement of roadside scan tools and I/M (Inspection/Maintenance) tools to be vehicle-manufacturer independent;
- fulfilled requirement to make enhanced test sequences available to independent aftermarket.

6 Use case overview and principles

6.1 Basic principles for use case definition

The following basic principles have been established as a guideline to define the OTX use cases.

a) OTX use cases describe the interaction between automotive system suppliers, departments (OEM engineering, OEM manufacturing and OEM after sales) and authorized and independent repairers.

b) All stakeholders (system suppliers, OEMs and tool suppliers) which offer diagnostic test sequences are expected to implement and follow the requirements of this part of ISO 13209.

c) The use cases as specified in this part of ISO 13209 define a common way to organize the exchange of test sequences.

d) The content of instance OTX documents and the quality of the information is the responsibility of the originator.

6.2 Use case clustering

Table 1 provides an overview of the main categories of OTX use cases. An OTX use case category may have one or more use cases.

Table 1 — Main use case clusters

#	Main title of use case cluster	Brief description
1	Documentation and specification	Use cases within this cluster describe the formal documentation and specification of test sequences.
2	Exchange and reusability	OTX sequences shall be exchangeable and reusable. Use cases describing the different scenarios of test sequence exchange and reuse are described in this cluster.
3	Extensibility	Use cases describing the use of sequence components that are defined within a distinct extension (e.g. legacy diagnostic system, measurement systems, etc.).
4	Localization	Use case describing support for string internationalization as well as physical unit conversion, comparison and unit localization.
5	Ability to be executed	The use of OTX sequences within a runtime environment is described in this cluster.

6.3 Actors

Table 2 gives a definition of the actors that were considered for modelling of the use cases.

Table 2 — Use case actors

#	Role	Brief description
1	Vehicle system engineer	Engineer involved in the development of vehicle systems, at the supplier or the OEM.
2	Test engineer	Engineer involved in the diagnostic testing of vehicles, in vehicle production, engineering, after sales or aftermarket.
3	Test sequence author	Creates, modifies and maintains test sequences for vehicle diagnostics.
4	Configuration manager	Provides the overall configuration management (CM) infrastructure and environment.

7 Use cases

7.1 Cluster 1: Documentation and specification

UC_DocSpec1 — High-level test sequence design

- **Rationale:** Describe overall flow of a test without the need for details relating to actual test execution.
- **Description:** In this use case, a system expert describes the overall flow and sequence of a test without the need for precise details such as communication parameters and test limits. The high-level sequence of steps is described using a specialized editing tool that emits a machine-readable exchange format. This sequence can be represented graphically by tools that import the exchange format without the need for further manual input.
- **Actors:** Vehicle system engineer.
- **Input:** Vehicle system specification.
- **Output:** High-level formal test sequence that can be visualized graphically.
- **Typical example:** A vehicle system engineer describes "how to test" a system at a functional level, before details of communication services and vehicle integration are known. A test engineer then uses the sequence as input to the process of creating a detailed executable sequence for production or after sales testing.

UC_DocSpec2 — Vehicle system engineer creates an executable test sequence

- **Rationale:** Allow a vehicle system expert to create test sequences using e.g. ODX-based communication for use in vehicle engineering, production or after sales without the need for conventional programming skills.
- **Description:** The vehicle system engineer creates a test sequence to be executed in a production or after-sales test system.
- **Actors:** Vehicle system engineer.
- **Input:** Vehicle system specification.
- **Output:** Machine-readable test sequence for execution in a test system that can be viewed graphically.
- **Typical example:**
  - A new electronic system is introduced into a vehicle and tests are required in production and after sales.
  - A production test engineer inspects graphical test sequence to diagnose production issues.
- **Comment:** Vehicle system specialists have the knowledge about how a system is intended to work, and how it is integrated into a vehicle. These specialist engineers may not have conventional programming skills, and require a GUI-centred way of capturing the test sequences for the test system.

UC_DocSpec3 — Test engineer updates a test sequence

- **Rationale:** Allow a test engineer who may not have programming skills to modify an existing test sequence because of a requirements change.
- **Description:** The test engineer updates a test sequence for a production or after sales test system. production/after sales environments may impose requirements (for example legislative, timing, parameter change) that require modification to a test sequence. These modifications may be carried out by test engineers who do not have access or skills to change a test system. The test engineer changes the test sequence using graphical tools.
- **Actors:** Test engineer.

- **Input:** Existing test sequence, test requirements.
- **Output:** Machine-readable test sequence for execution in a test system.
- **Typical examples:**
  - Electronic system updated in a vehicle.
  - Production test requirements change.
  - After-sales test requirements change.

7.2 Cluster 2: Exchange and reusability

UC_ExchRe1 — Test sequence exchange (engineering, manufacturing, after sales)

- **Rationale:** Exchange of test sequences between engineering, manufacturing and after sales to increase productivity and quality; delivery of test sequences by ECU and system suppliers; know-how transfer and feedback between engineering, manufacturing and after sales using test sequences as a standardized documentation.
- **Description:** Test sequence exchange between engineering, manufacturing and after sales as well as between different OEM departments and ECU/system suppliers:
  - Engineering defines the test requirements of new ECUs in test sequences.
  - Manufacturing and after sales use and adapt the test sequences for their purpose.
  - Engineering uses the feedback of the other company parts to improve the test sequences.
- **Actors:** Test sequence author.
- **Input:** Diagnostic know-how; feedback of users.
- **Output:** Optimized and adapted test sequence; documentation.
- **Typical examples:**
  - Development of a new ECU, creating a basic test sequence by the engineering department and adapted by manufacturing and after-sales departments.
  - ECU supplier develops and delivers test sequences.

UC_ExchRe2 — Reusing test sequences

- **Rationale:** Reusing of verified test sequences increases productivity and quality; reusing of test sequences allows centralized changes; reusing of test sequences encourages team working and shared developing.
- **Description:** With reuse of test sequences, large test applications can be built composed of test sequences of different task levels. Generic test sequences help to adapt a test sequence to different ECUs, ECU variants and different vehicle configurations.
- **Actors:** Test sequence author.
- **Input:** (generic)Test sequence.
- **Output:** Large test application composed of test sequences.
- **Typical examples:**
  - Library of ECU-dedicated test sequences to use in function-oriented test sequences.