TECHNICAL REPORT

ISO/TR 17452

First edition
2007-04-15

Intelligent transport systems — Using UML for defining and documenting ITS/TICS interfaces

Systèmes intelligents de transport — Usage de UML pour définir et documenter les interfaces ITS/TICS

Reference number
ISO/TR 17452:2007(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

Page

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.

In exceptional circumstances, when a technical committee has collected data of a different kind from that which is normally published as an International Standard ("state of the art", for example), it may decide by a simple majority vote of its participating members to publish a Technical Report. A Technical Report is entirely informative in nature and does not have to be reviewed until the data it provides are considered to be no longer valid or useful.

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/TR 17452 was prepared by Technical Committee ISO/TC 204, *Intelligent transport systems*.

Introduction

ISO 14817 specifies the formats and procedures used to define information exchanges within the ITS/TICS sector. Such information arises through the development of the architecture for a particular application standard and the subsequent specification of the detailed data concept instances that arise in association with the architecture's interfaces. This Technical Report illustrates the steps involved in such development.

In the development of standards, it is often the case that working groups have a well-formed perception of the conceptual context within which their standard is to be applied. This is the case because many standards are the result of a refinement and consensus of requirements based on recent practice. The formal process for the identification of the requirements is streamlined to capitalize on this available body of knowledge.

For completeness, we begin with the capture of requirements. These requirements need be only those which directly affect the standard. The context of a real-world system that incorporates the standard would include a much wider range of requirements; however, we are focusing on that aspect of standards which produces data elements and other concept instances which will be registered in a data dictionary or registry. The methodology is derived from processes used in the development of software-intensive systems.

Intelligent transport systems — Using UML for defining and documenting ITS/TICS interfaces

1 Scope

This Technical Report gives guidelines for using the unified modelling language (UML) for defining and documenting interfaces between intelligent transport systems (ITS) and transport information and control systems (TICS). It presents these guidelines in the context of a case study for the creation of an ITS/TICS data dictionary and submissions to the ITS/TICS data registry.

In UML [6], an interface is a collection of operations that used to specify a service of a class or component.

The ITS/TICS data registry defined in ISO 14817 builds on this definition by mapping an operation to a message, and then it extends the definition of an interface to be a dialogue (i.e. a collection of messages within an implied protocol). This Technical Report conforms to these steps.

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 14817, *Transport information and control systems — Requirements for an ITS/TICS central Data Registry and ITS/TICS Data Dictionaries*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 14817 and the following apply.

3.1
automatic equipment identification
AEI
process of identifying equipment or entities that uses the surface transportation infrastructures by means of on-board equipment combined with the unambiguous data structure defined in ISO/TS 17261

NOTE "Equipment" indicates large equipment that is carried in, or forms an integral part of, a trailer or trailer-mounted unit.

3.2
automatic vehicle identification
AVI
process of identifying vehicles using on-board equipment combined with the unambiguous data structure defined in ISO/TS 17261

3.3
electronic data interchange
EDI
passing of a data message, or series of messages, between computers and/or between different software systems

NOTE 1 Within this context, an EDI message is normally compatible with the form specified in ISO 9897.

NOTE 2 EDI is an instance of an electronic data transfer transaction.

3.4
goods provider
party that provides goods to another party

NOTE A goods provider can be a manufacturer, trader, agent or individual.

3.5
information
data, documentation and other relevant knowledge organized to inform and describe

3.6
information manager
function of managing information in a system

NOTE The role of information manager can be provided by one or many actors. It can be performed internally by one or more of the system's principal actors, or can be formed commercially or altruistically by one or more third parties.

3.7
intermodal transport
movement of goods in one or more loading unit or vehicle which uses successively several modes of transport without handling of the goods themselves when changing modes

NOTE 1 Unlike multimodal transport (3.10), intermodal transport implies changing from one mode to another using the same form of loading unit.

NOTE 2 See ISO/TS 17262 and ISO/TS 17263.

3.8
journey
⟨AVI/AEI⟩ physical movement of goods from the goods provider (3.4) to the receiver (3.11)

3.9
load
that which is to be transported from the goods provider (3.4) to the receiver (3.11)

NOTE A load comprises the consignment, packaging, pallets and/or containers that are smaller than an ISO container.

3.10
multimodal transport
carriage of goods by at least two different modes of transport

cf. intermodal transport (3.7)

NOTE Multimodal transport implies that either there is more than one modal shift, or that loads may be broken into partial loads as part of a modal change.

3.11
receiver
⟨AVI/AEI⟩ one who receives goods as a result of a journey (3.8) from a goods provider (3.4)

3.12
returnables manager
function that manages the supply, maintenance and returns cycle of returnable units (3.13)

NOTE The returnables manager function may be performed by one or more of the system's principle actors or by an independent third party.

3.13
returnable unit
unit used as part of a load (3.9), which is returned to the goods provider (3.4) or to a returnables manager (3.12)

NOTE Pallets and trays are examples of a unit.

3.14
transponder
tag
electronic transmitter/responder which responds to the receipt of suitable modulated or unmodulated downlink signals and transmits predetermined information according to predefined protocols at a predetermined frequency

NOTE The transmissions can be powered from energy obtained from the downlink or can be assisted by an on-board power supply. Forms the core, but not necessarily the only, function of an item of on-board equipment. Within the AVI/AEI context, it is fitted to the vehicle or equipment. Its prime function is to provide the identity of the item, but it can also contain additional information. For some special purposes, transponders can be installed in fixed positions and read by mobile equipment.

3.15
transport
⟨AVI/AEI⟩ vehicles/aircraft/ships used to move a consignment from the goods provider (3.4) to the receiver (3.11) or to move returnable units (3.13) back through the system

3.16
transport means
vehicles, trailers, vessels, aircraft or combination thereof which perform the journey (3.8) to deliver the consignment to the receiver (3.11) or to return returnable units (3.13), together with the driver/pilot/crew physically conducting the journey

3.17
transport operator
function that owns and/or manages the transport means (3.16)

4 Abbreviated terms

AVI/AEI automatic vehicle and equipment identification

RCU returnable container unit (see also 3.13)

UML unified modelling language

VMS variable message sign

5 Example of automatic vehicle and equipment identification

To illustrate the steps in this tutorial, we use the example of AVI/AEI for intermodal goods transport as specified in ISO/TS 17261 [3] and ISO/TS 17262 [4]. The overall application information architecture is shown in Figure 1. The key entities involved in the architecture are defined in Clause 6.

The context of Figure 1 is the information associated with the journey of goods from the goods provider to the end user. In this journey, the goods will form a load. The load can pass through several transport mode changes and other handling processes. In each instance, ISO/TS 17262 is applicable to an automated handling process.

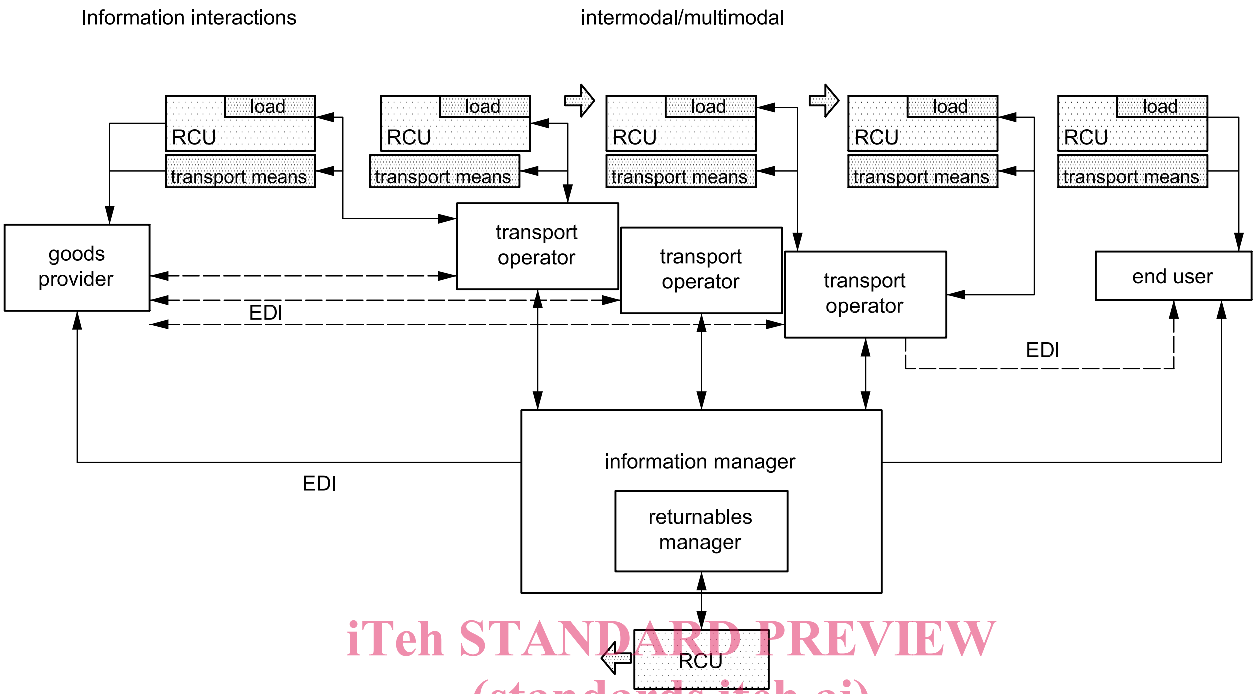

Figure 1 — Schematic diagram of the application information architecture for the journey of goods from goods provider to receiver

6 Developing the data concepts in an application standard

This tutorial employs UML [6] and illustrates how a process framed around UML (e.g. [7]) can be employed to develop the behavioural descriptions of an application architecture [7], necessary to capture the interface data concepts. In the case of standards development, some of these concepts will be the focus of an application standard.

The whole process can be broken into a sequence of steps (Figure 2), each of which has its own set of modelling artifices in UML. It is a straightforward matter to present an example in which the last step reveals a set of standard data elements. In practice the process is iterative, involving trial and error, and the steps are not always revisited in the same order.

The application information architecture shown in Figure 1 serves to justify the development of a standard because it identifies the widespread applicability of the standard. However, it is not sufficiently developed for defining the data concepts. This tutorial need only focus on a single goods-handling application function in order to illustrate the process of architecture development which culminates in data concept definition. The application is described in the first step of the process of Figure 2, in which the requirements are defined by a use case.

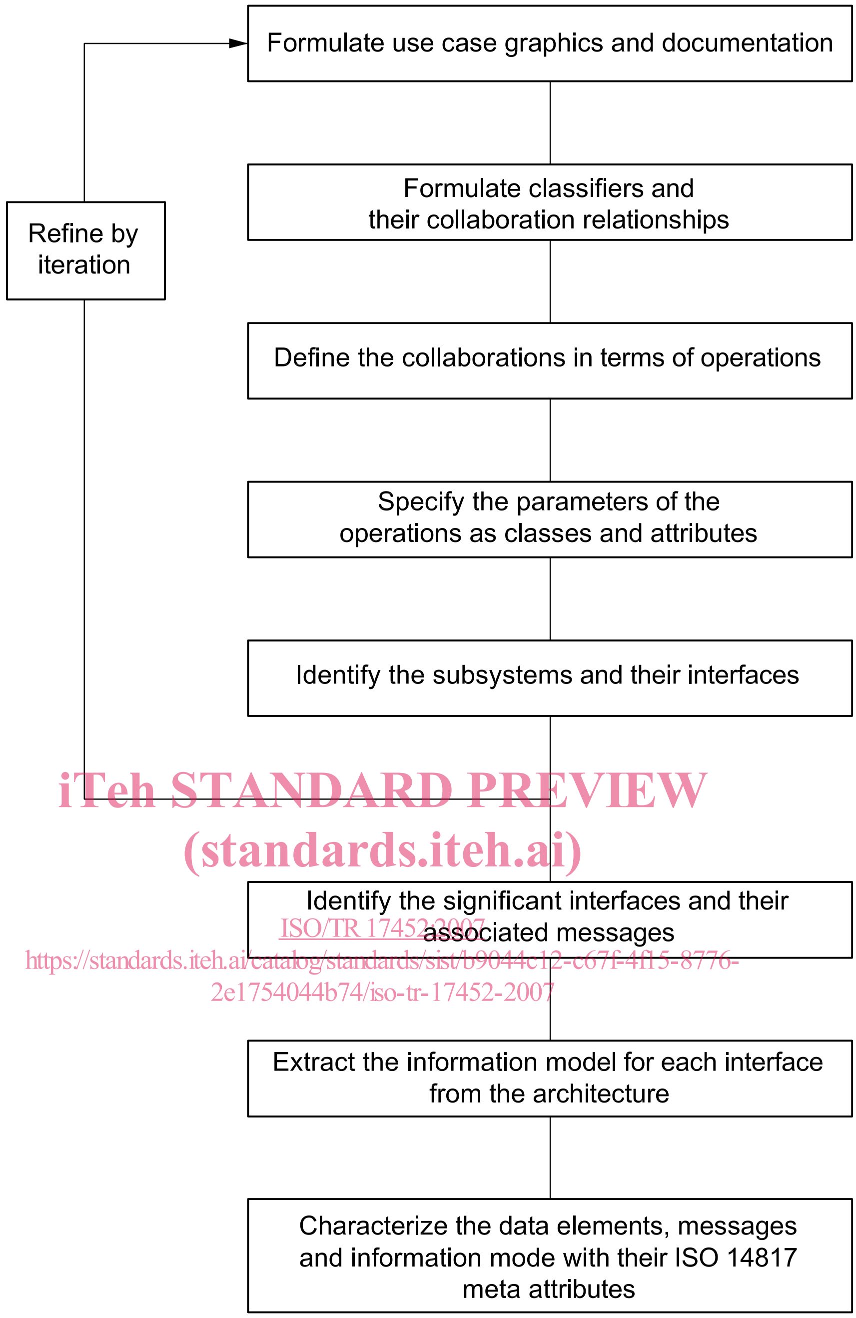

Figure 2 — Steps in the development of the architecture items and data concepts

6.1 Use case

As stated in [6],

> *The use case construct is used to define the behaviour of a system or other semantic entity without revealing the entity's internal structure. Each use case specifies a sequence of actions, including variants, that the entity can perform interacting with actors of the entity.*

A use case captures the essence of a service requirement. The user of the service (an actor) can be someone or it can be another system outside the target system. As stated in [6],

> *An actor defines a coherent set of roles that users of an entity can play when interacting with the entity. An actor may be considered to play a separate role with regard to each use case with which it communicates.*

In our example (Figure 3), the purpose of the use case is to provide instructions to the driver of the transport means for a load (e.g. at the marshalling yards where an intermodal change is to occur). The primary actor is the driver. Depending on your point of view, you might regard the transponders as within the system boundary or outside, but it is more meaningful here to describe them as actors that are necessarily involved in the functional requirement to instruct a driver where to proceed.

Depending on the scope of a system there can be many use cases, which would not be identified independently. Their specification affects the rest of the architecture and details of the architecture can in turn affect their specification.

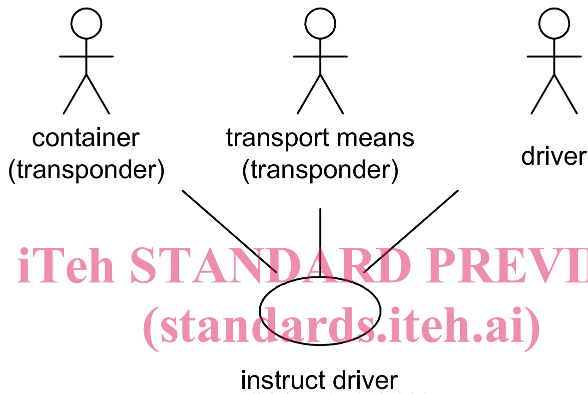

Figure 3 — Use-case diagram for "instruct driver"

The graphical view of Figure 3 alone is not a sufficient specification of the use case. Documentation must give details on the flow of events that delivers the result to the user and in particular the interactions with the actors. For example, when the vehicle approaches the entrance gate of the marshalling yards, the transponders on the vehicle and the vehicle load are interrogated for identification. The identifiers are processed by a control system that has access to database data about the load and data about the vehicle itinerary. The input of these data would be covered by other use cases that would involve the freight forwarder and the transport operator as actors. The control system uses special-purpose systems to determine where the load is to be unloaded and stacked. The location and path directions from the gate to the unload point are then output to the driver (e.g. using a VMS).

In the subsequent elaboration, we will only deal with the information and control for the load management. Similar elaboration would be required for the vehicle management (before and after unloading). However, all the necessary data concepts are identified in this restricted analysis.

The documentation of the use case sets the basis for all the remaining steps. Below is a complete sequence of steps that can be undertaken to ensure that all the situations have been covered. Pre-conditions and post-conditions might be too detailed for most architecture developments. The idea of extension points is to structure the collection of use cases, if possible. For example, in some circumstances a service might have to be expanded by actions that amount to the specification of a separate use case which is an extension of the basic service. The sequence is as follows:

a) use-case name;

b) brief description;

c) flow of events;

