BS ISO 11783-10:2015

BSI Standards Publication

Tractors and machinery for agriculture and forestry — Serial control and communications data network

Part 10: Task controller and management information system data interchange

bsi.

...making excellence a habit.™

Document Sharing Hub

National foreword

This British Standard is the UK implementation of ISO 11783-10:2015. It supersedes BS ISO 11783-10:2009 which is withdrawn.

The UK participation in its preparation was entrusted to Technical Committee AGE/6, Agricultural tractors and forestry machinery.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

© The British Standards Institution 2015.
Published by BSI Standards Limited 2015

ISBN 978 0 580 78362 3

ICS 35.240.99; 65.060.01

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 30 September 2015.

Amendments/corrigenda issued since publication

Date	Text affected

Documents
Document Sharing Hub

Full document details: BS ISO 11783-10:2015

INTERNATIONAL STANDARD

ISO 11783-10

Second edition
2015-09-15

Tractors and machinery for agriculture and forestry — Serial control and communications data network —

Part 10: Task controller and management information system data interchange

Tracteurs et matériels agricoles et forestiers — Réseaux de commande et de communication de données en série —

Partie 10: Contrôleur de tâches et échange de données des systèmes d'information de gestion

Reference number
ISO 11783-10:2015(E)

Documents
Document Sharing Hub

© ISO 2015

Full document details: BS ISO 11783-10:2015

COPYRIGHT PROTECTED DOCUMENT

© ISO 2015, Published in Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub

Full document details: BS ISO 11783-10:2015

Contents

Page

Foreword ... **v**
Introduction ... **vii**
1 Scope ... **1**
2 Normative references ... **1**
3 Terms and definitions ... **2**
4 Abbreviations ... **3**
5 General description ... **4**
5.1 Task management ... 4
5.2 Task management on FMIS computer ... 5
5.3 Preselection and assignment of Devices ... 6
5.4 Task controller interface driver ... 6
5.5 Task controller user interface ... 6
5.6 Data logger function ... 7
6 Task controller requirements ... **7**
6.1 Task selection and execution ... 7
6.2 Logging of time and position ... 8
6.3 Logging parameters from parameter groups ... 8
6.4 Logging of task events ... 9
6.5 Language, formats, and measurement units selection ... 9
6.6 Connection management ... 10
6.6.1 Task controller initialization ... 10
6.6.2 Client initialization with the task controller ... 10
6.6.3 Connection maintenance ... 11
6.6.4 Task controller connection shutdown ... 12
6.7 Task controller number ... 14
6.7.1 Client initialization on networks with multiple task controllers ... 14
6.8 Data exchange on the network ... 15
6.8.1 Site-specific application ... 16
6.8.2 Data logging ... 19
6.8.3 Totals ... 20
6.8.4 Data log triggers ... 21
6.8.5 Peer Control ... 24
7 Data logger requirements ... **26**
7.1 General ... 26
7.2 Connection management ... 27
7.3 Measurements and totals ... 28
8 Data transfer ... **28**
8.1 General ... 28
8.2 Extensible Markup Language ... 28
8.3 Extensible schema definition ... 29
8.4 XML schema definition ... 30
8.4.1 Proprietary XML schema extensions ... 32
8.5 XML data transfer files ... 32
8.6 Binary data transfer files ... 35
8.6.1 General ... 35
8.6.2 Grid binary file structure ... 35
8.6.3 Log data binary file structure ... 38
8.6.4 Point data binary file structure ... 40
8.7 Device descriptor object pool ... 41
Annex A (normative) **Device descriptor objects** ... **45**

Documents
Document Sharing Hub

© ISO 2015 – All rights reserved

Full document details: BS ISO 11783-10:2015

Annex B (normative) **Message definitions** **52**
Annex C (normative) **XML elements relationship diagram** **71**
Annex D (normative) **XML elements and attributes** **74**
Annex E (normative) **Predefined ISO-11783 attachments** **156**
Annex F (normative) **TC Functionalities and Device Descriptor Object Pool definitions** **163**
Annex G (normative) **Task Based Time Registration** **202**
Bibliography **205**

© ISO 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

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.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

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. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the WTO principles in the Technical Barriers to Trade (TBT) see the following URL: Foreword - Supplementary information

The committee responsible for this document is ISO/TC 23, *Tractors and machinery for agriculture and forestry*, Subcommittee SC 19, *Agricultural electronics*.

This second edition cancels and replaces the first edition (ISO 11783-10:2009) which has been technically revised.

ISO 11783 consists of the following parts, under the general title *Tractors and machinery for agriculture and forestry — Serial control and communications data network*:

— *Part 1: General standard for mobile data communication*

— *Part 2: Physical layer*

— *Part 3: Data link layer*

— *Part 4: Network layer*

— *Part 5: Network management*

— *Part 6: Virtual terminal*

— *Part 7: Implement messages application layer*

— *Part 8: Power train messages*

— *Part 9: Tractor ECU*

— *Part 10: Task controller and management information system data interchange*

— *Part 11: Mobile data element dictionary*

— *Part 12: Diagnostics services*

— *Part 13: File server*

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

— *Part 14: Sequence control*

© ISO 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

Introduction

This International Standard specifies a communications system for agricultural equipment based on the ISO 11898-1 protocol. SAE J1939[1] documents, on which parts of ISO 11783 are based, were developed jointly for use in truck and bus applications and for construction and agriculture applications. Joint documents were completed to allow electronic units that meet the truck and bus SAE J1939[1] specifications to be used by agricultural and forestry equipment with minimal changes.

General information on this International Stamdard is to be found in ISO 11783-1. The purpose of this International Standard is to provide an open, interconnected system for on-board electronic systems. It is intended to enable electronic control units (ECUs) to communicate with each other, providing a standardized system.

The International Organization for Standardization (ISO) draws attention to the fact that it is claimed that compliance with this part of ISO 11783 may involve the use of a patent concerning the controller area network (CAN) protocol referred to throughout the document.

ISO takes no position concerning the evidence, validity, and scope of this patent.

The holder of this patent has ensured ISO that he is willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holder of this patent right is registered with ISO. Information may be obtained from:

Robert Bosch GmbH

Wernerstrasse 51

Postfach 30 02 20

D-70442 Stuttgart-Feuerbach

Germany

Attention is drawn to the possibility that some of the elements of this part of ISO 11783 may be the subject of patent rights other than those identified above. ISO shall not be held responsible for identifying any or all such patent rights.

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

Documents
Document Sharing Hub

INTERNATIONAL STANDARD ISO 11783-10:2015(E)

Tractors and machinery for agriculture and forestry — Serial control and communications data network —

Part 10: Task controller and management information system data interchange

1 Scope

This International Standard as a whole specifies a serial data network for control and communications on forestry or agricultural tractors and mounted, semi-mounted, towed, or self-propelled implements. Its purpose is to standardize the method and format of transfer of data between sensors, actuators, control elements, and information storage and display units, whether mounted on, or part of, the tractor or implement. This part of ISO 11783 describes the task controller applications layer, which defines the requirements and services needed for communicating between the task controller (TC) and electronic control units. The data format to communicate with the farm-management computer, the calculations required for control, and the message format sent to the control function are defined in this part of ISO 11783.

2 Normative references

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

ISO 11783-1, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 1: General standard for mobile data communication*

ISO 11783-3, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 3: Data link layer*

ISO 11783-5, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 5: Network management*

ISO 11783-6, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 6: Virtual terminal*

ISO 11783-7, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 7: Implement messages application layer*

ISO 11783-11, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 11: Mobile data element dictionary*

ISO 11783-12, *Tractors and machinery for agriculture and forestry — Serial control and communications data network — Part 12: Diagnostics services*

ISO 11898-1, *Road vehicles — Controller area network (CAN) — Part 1: Data link layer and physical signalling*

ISO/IEC 10646, *Information technology — Universal Coded Character Set (UCS)*

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 11783-1 apply.

3.1
client
control function (CF) that establishes a connection with a task controller (TC) or a data logger (DL) and provides data for data logging by the DL or TC and/or accepts control commands from the TC

Note 1 to entry: This term has been introduced in version 4 to replace the term working set for the CF that connects to a TC or DL. The multi member working set concept is not applied to TC or DL clients. The communication between a TC or DL and their clients is limited to transmission of messages between TC or DL and the working set master

3.2
coding data
data that changes infrequently, such as machine or chemical data, or data which is referenced in a task for the purpose of allocating tasks administratively

3.3
data dictionary entity
description of the data dictionary identifier, definition, value range, value resolution, and units specifications used by process data variables

3.4
data logger
DL
control function (CF) defined specifically to perform data logging functionality

3.5
data logger (DL) client
CF that establishes a connection with a data logger (DL) and provides data for data logging by the DL

3.6
data transfer file set
collection of files in the extensible markup language format and binary formats, which are used for the data transfer between the farm management information system and the task controller of an ISO 11783 network

3.7
device descriptor object pool
DDOP
collection of device-related objects and their relationships which together describe the functionality and structure of a device for the purposes of task control and data logging

3.8
device element
any addressable item on a device

EXAMPLE Nozzle on sprayer boom where the nozzle has individually addressable process data variables.

3.9
field
area of land managed by a farmer, represented by either a single partfield or a collection of more than one partfield

Note 1 to entry: The field is only of importance within the farm management information system for business management considerations and is not necessarily related to a single crop.

3.10
grid cell
rectangular areas defined by overlaying a grid on a partfield

© ISO 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

3.11
management computer gateway
CF that interfaces to the management computer system and to the ISO 11783 network

Note 1 to entry: A management computer gateway can store data for transmission at a later time.

3.12
partfield
area characterized by the cultivation of only one agricultural crop

Note 1 to entry: Partfield is the XML element to which tasks are allocated to obtain smallest granularity.

3.13
polygon
planar surface defined by one exterior boundary and by zero or more interior boundaries

Note 1 to entry: Each interior boundary describes a hole in the surface.

Note 2 to entry: A single or group of polygons can be used to define a treatment zone.

3.14
process data variable
element of information that has a value that describes the state of a process

Note 1 to entry: Process data variables consist of the attributes range, resolution, and units, as defined in the data dictionary.

3.15
setpoint value source
CF that can determine a setpoint value during a task for use by another CF and contains one or more Device Process Data object(s) with the property attribute set to the “control source” value

3.16
setpoint value user
CF that accepts a setpoint value from either the TC or another source to modify its real-time performance such as rate control and contains one or more Device Process Data object(s) with the property attribute set to the “settable” value

3.17
task controller (TC) client
CF that establishes a connection with a task controller (TC) and provides data for data logging and/or accepts control commands from the TC

3.18
task controller (TC) number
identification number that is derived from the function instance of the task controller (TC)

3.19
XML element
representation of the data of a domain object and consists at least of an opening tag, a number of attributes, and a closing tag

4 Abbreviations

For the purposes of this document, the abbreviations given in ISO 11783-1 apply.

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

5 General description

5.1 Task management

There are two main purposes of task management in the mobile implement control system.

The first is the management of the farm resources, including tractors, implements, sensor systems, workers, and the products used. It is possible for the farmer to plan and evaluate the use of the resources. He is then able to automatically control the use of his inventory of products and can keep track of the status and conditions of his machinery. Resource designators are globally transferred as coding data and are part of the data transfer file set as detailed in Clause 7.

The second purpose is the management of the farm activities carried out in the fields. These activities are described by tasks to distinguish all the work that is planned, is in execution, or has been done by the farmer or by a contractor for one customer in one partfield.

The data transfer is possible in two directions. The planned tasks are sent to the task controller (TC) on the mobile implement control system (MICS) and the results of the work are sent back to the farm-management information system (FMIS). Tasks can be generated both on the FMIS and on the MICS.

Task management has the following workflow.

a) Planning field tasks and maintaining coding data using the software of an FMIS computer operated by the farmers or contractors, as detailed in 5.2.

b) Converting the task data into XML format.

c) Assigning the task data produced by the planning software to the data required for the implements or sensor systems to be used to complete the planned tasks. This step is optional.

d) Transferring the task data from the FMIS system to the TC of the MICS, as detailed in 5.4.

e) The TC uses the task data to transmit process data messages to the ECUs on the implement.

f) The TC collects data according to the DataLogTriggers specified in the task data.

g) Transferring the collected data to the FMIS. Collected data can be in the XML format or in a proprietary format. When a proprietary format is used, this step involves converting the proprietary format into the XML format.

h) Reading the XML files and converting into the FMIS format for storage and evaluation of the data.

Figure 1 illustrates the interfaces between the software on the FMIS computer and the ECUs mounted on an ISO 11783-configured implement.

© ISO 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

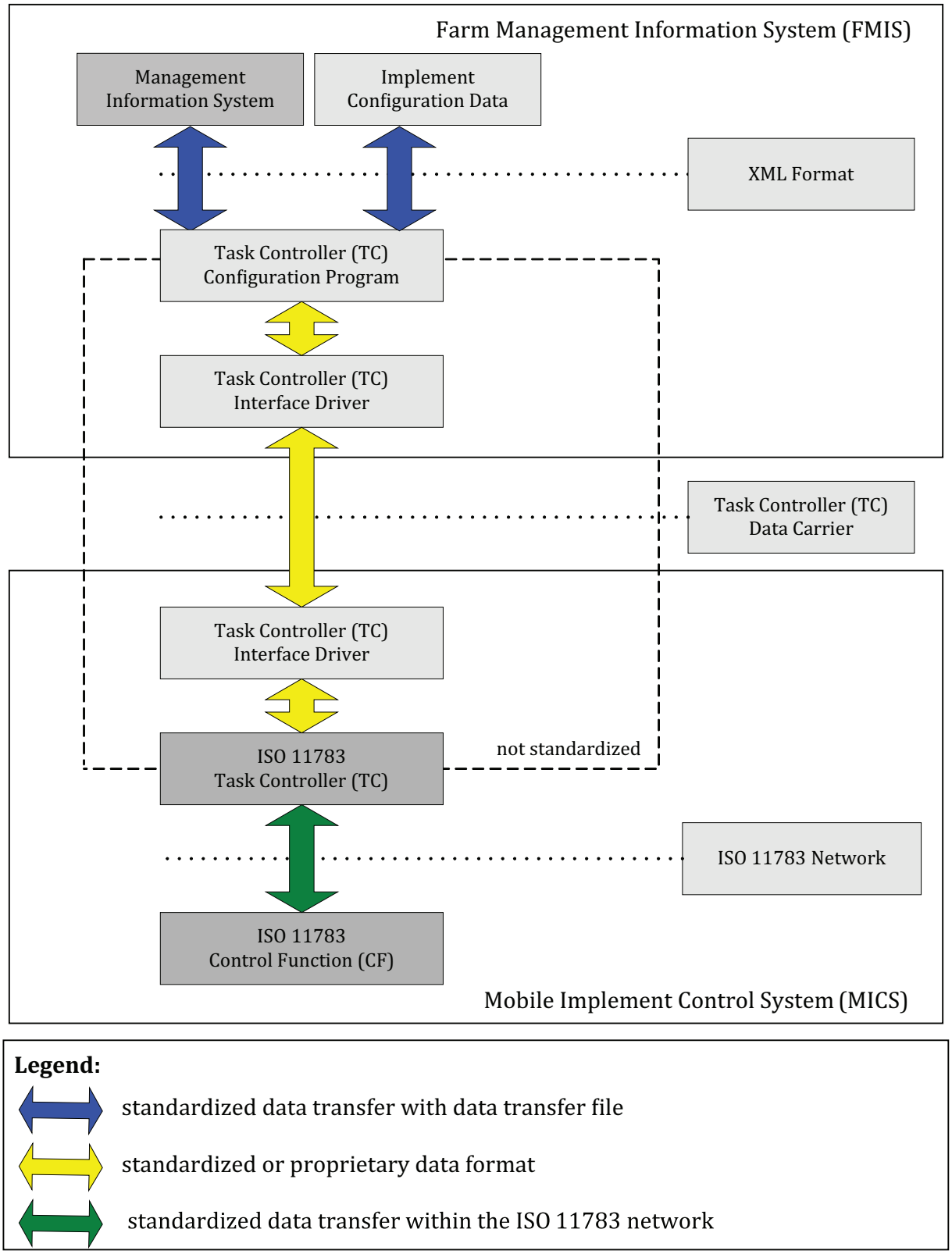


Figure 1 — Task management entities and interfaces

5.2 Task management on FMIS computer

Task management is defined as a part of an FMIS, responsible for planning and evaluation of field work. Tasks specify what, where, how, by whom, and when work is planned to be carried out.

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

The amount of data transferred between the FMIS and the MICS is dictated by the administrative requirements of a farming enterprise. For the recording of field activities only, the task management can be used to file the data in a working journal. For this purpose, only the coding data need be transferred from FMIS to MICS, and the tasks are created on the MICS by selection of the involved resources. In this case, only the data transfer file set from MICS to FMIS contain tasks. In enterprises where tasks are planned on the FMIS, these will be included, together with the coding data, in the data transfer file set from FMIS to MICS. These planned tasks can range from mere planned allocations of resources to geographical information for site-specific field operations.

5.3 Preselection and assignment of Devices

Any client device in the mobile system can only be identified uniquely by its CF NAME. For FMIS purposes, the client CF NAME needs to be unique worldwide. This implies that the device manufacturer has the responsibility to ensure that the Identity Number in combination with the other fields in the NAME uniquely identifies the device.

At the FMIS, the preselection of devices depends on the planned task. It can be necessary to assign a type of device or function, a specific device, or even the device of a particular manufacturer. The DeviceAllocation XML elements may include planned assignments about the devices to be used. This information ranges between specific and indistinct.

The XML attribute ClientNAMEValue contains the eight-byte NAME of a control function as it is defined in ISO 11783-5. Not all parts of a NAME need be specified and only certain elements of a NAME need to be defined to determine a device on the mobile network. Those parts of ClientNAMEValue containing information to be used on the mobile system to select a proper device are masked by a bitset structure stored in the XML attribute, ClientNAMEMask. All different combinations of elements of the NAME structure can be marked as valid for device selection (logical AND). On the FMIS, these masks could be coded as symbols. Once the preselection information is set in a task on the FMIS, it is not overwritten on the mobile system because the device that is used during task processing on the mobile system is stored as the XML attribute DeviceIdRef.

5.4 Task controller interface driver

After generating the interface files, the TC driver of the manufacturer of the TC is activated on the farm computer. This driver is responsible for the data transfer to the TC, which is part of the MICS using its proprietary data format or the ISO 11783-10 XML format and data carrier, such as any type of memory card or radio link. The translation of the data from the data transfer file set in messages on the ISO 11783 network, as well as the kind of transfer between the mobile system and the FMIS, is not subject to a standardized procedure as part of this International Standard. To make implement-specific set point data, this driver can add and use device descriptor data supplied by the manufacturers.

5.5 Task controller user interface

A TC can provide a means for the user to interact with the TC. User interaction can be through a virtual terminal (VT) or other interface. Operator interfaces can range from very simple to elaborate, depending on the designer's intent. For example, a simple TC that only supports single tasks that run automatically might not require any user interaction. More advanced TCs can offer additional operator interface capabilities, such as

— select a task from a list,

— start/stop/resume/complete a task,

— modify a task,

— create a task, and

— add new coding data.

© ISO 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

Through the user interface, an operator can react to special circumstances or events in order to execute tasks in a reasonable way. The operator can also be informed about the status and results of the tasks and their components. For example, the operator could print a work confirmation for a farmer.

5.6 Data logger function

A separate data logger (DL) can be installed on a network, e.g. to function as a telemetry data logger. The task controller protocol is used by such a telemetry data logger which enables this data logging only control function to use device descriptors and process data messages for collecting data in addition to logging data from any other parameter groups that are broadcast or can be requested on the ISO 11783 network.

This DL CF is identified on the network by a specific data logger function, as specificied in ISO 11783-1. The DL functionality is a subset of the TC functionality and uses the same connection mechanism as the TC control function. Reusing the TC to TC client communication protocol makes the data logging functionality and the process data definitions of the TC available for use with the DL CF, without interfering with the TC client controlling functionalities of the TC.

The data logger function can be used for general non-task-related data logging. The use of the data logger function is not restricted to a telemetry environment.

The data logger function is introduced in ISO 11783-10 version 4. The ISO 11783-10 version is communicated on the network in the Version message (B.5.3) and in the data transfer file set in the VersionMajor attribute of the ISO11783_TaskData XML element (D.32).

6 Task controller requirements

6.1 Task selection and execution

The TC can provide a mechanism for the selection and shall provide a mechanism for the execution of a task contained in the data transfer file set. Selection of an individual task can be done either by the operator through an operator interface or automatically by the TC. The means of task selection is not specified by this part of ISO 11783. The TC designer is free to decide how task selection is implemented. The method provided for starting and stopping a task is not subject to standardization. This functionality is also to be determined by the TC designer.

On a MICS, a task may or may not be selected. If a task has not been selected by the operator, then the TC may prompt the operator for a task selection or may select a task automatically.

The status of a task is defined in Table 1.

Table 1 — Task status

Planned	Task is prepared at FMIS or MICS but not yet processed on MICS.
Running	Task is currently being processed on MICS. Only one task can be active per TC at the same time.
Paused	Task was previously running, is not currently running, and is not yet completed. This state can be a task final state from a TC perspective. Tasks with this state in data transfer file set with a MICS origin are required to be processed by the FMIS.
Completed[a]	Task is finished. This status can only be set by the operator and cannot be set automatically by the MICS. Tasks with this state in data transfer file set with a MICS origin are required to be processed by the FMIS.
Template[b]	Task is a template task. When a template task is started, a copy of the template is made and the copy is started as a new task. Template tasks that are defined by the FMIS shall not be modified by the MICS. Template tasks that are created on the MICS shall be identified as new tasks in the data transfer file set. Once a template task that is created on the MICS is instantiated in a non-template task, then the template task shall not be modified.

© ISO 2015 – All rights reserved

AVA Documents Document Sharing Hub
Full document details: BS ISO 11783-10:2015

Table 1 *(continued)*

Canceled[b]	The purpose of this state is to be able to withdraw a task. The FMIS or MICS can use this status to signal the MICS or FMIS to withdraw a task.
[a] The completed task status is optional. Some TCs may not support this status.	
[b] This definition is introduced in ISO 11783-10 version 4.	

6.2 Logging of time and position

There can be a need to assign some optional information, such as date, time, and GPS position data, to events that occur during task processing. This could be an event reflecting the interconnection of XML elements, such as the allocation of GPS position-related comments and/or flags or the assignment or exchange of a worker, for example. Other events are based on logging of received process data variable values from task-related clients, which shall be supplied with information of date, time, and position.

The XML elements AllocationStamp and Time enable the allocation of time and date values to several XML elements where needed. These XML elements can be of the type planned or effective, for specifying whether an event was planned or accomplished. Additionally, at a task level within the Time XML element only, more detailed time types, such as preparation time, ineffective time, repair time, or clearing time, are defined. Prior to ISO 11783-10 version 4, all time and date values had to be specified in local time and date values inside this XML element. In version 4 and later, time zone information may be added to the time and data values.

Within a Task XML element, multiple instances of the Time XML element can be included. Examples of when this occurs are when a task with a Time XML element of type planned is started and a new Time XML element of type effective is added or when a task is resumed and a new Time XML element of type effective is added. Time XML elements shall only be added to a Task XML element upon task status changes, upon resource assignment changes to tasks, and upon the TC start up or shut down events. A product allocation is also an example of a resource assignment change. Previously recorded Time XML elements shall not be modified.

Optionally, the AllocationStamp and Time XML elements can include up to two position XML elements that contain GNSS-supplied information. The first position XML element is added when the AllocationStamp or Time XML element is created and the second position XML element is added when the AllocationStamp or Time XML element is completed. If there is only a single position XML element present, then it specifies the position at the start of the AllocationStamp or Time XML element. The intermediate positions along a continuous CommentAllocation are not specified in the AllocationStamp XML element itself but are recorded in the binary log files specified by the TimeLog XML element. See D.10 for the position recording definition of continuous comments.

To be able to allocate multiple process data values, the Time XML element can include several DataLogValue XML elements. The number of these DataLogValues stored inside the task shall be limited to totals or single instance values. For a large number of DataLogValues, the use of the TimeLog XML element and binary log file is defined.

All process data-related logging data can be stored in a binary format as a separate file. The reference to the binary file is to be set by the XML element TimeLog. More TimeLog XML elements can exist per task, referring to external files with unique names inside the name space of the set of tasks belonging to the data transfer file set. The uniqueness of file name prefixes shall be guaranteed by the TC. Each TimeLog file definition results in two separate files — one to contain the binary data and the other to contain the XML coded header structure of the binary data set. The header structure defines the maximum data per binary record and enables the correct interpretation of the binary data. The filename extension of the binary file shall be ".bin"; the filename extension of the XML header structure file shall be ".xml".

6.3 Logging parameters from parameter groups

In addition to the ProcessDataVariable values, the values or parameters from other parameter groups can be logged by a TC. The XML elements DataLogTrigger and DataLogValue contain attributes to specify from which parameter group a value is to be logged. These attributes are optional. When used

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 11783-10:2015

attributes are specified, the DataLogDDI attribute in DataLogTrigger or DataLogValue shall be set to the ParameterGroupNumberValue (DDI = $DFFE_{16}$). Each parameter group may contain multiple values, and therefore both a parameter group number and a start and stop bit to obtain a single value from the data field of the CAN data frame shall be specified when the TC logs data from parameter groups other than the ProcessDataVariable. The size of the value is at a maximum 32 bits and is stored as the DataLogValue attribute in the XML element DataLogValue.

When the TC logs data from parameter groups, a reference to a DeviceElement is required. If this log data originates from parameter groups other than the ProcessDataVariable, a device with a control function NAME with a filled-in ClientNAME XML attribute and a DeviceElement referring to this device shall be defined. These device and DeviceElement XML elements may be generated by the TC or can be supplied by the FMIS.

6.4 Logging of task events

The planned or effective allocation of resources like Workers, Devices, Products, Comments, and Controls to a Task XML element is specified through an allocation XML element pattern. This allocation pattern is instantiated in the XML elements WorkerAllocation, DeviceAllocation, ProductAllocation, CommentAllocation, GuidanceAllocation, and ControlAssignment. Within a single Task XML element, multiple instances of allocations of these resources can occur. Examples of when multiple allocations are recorded within a single task are when a planned allocation becomes effective upon a task start or when a resource is disconnected and reconnected to a task during execution of that task. Previously recorded resource allocations shall not be modified but new resource allocations can be added to a task.

6.5 Language, formats, and measurement units selection

On an ISO 11783 network, the language, formats, and measurement units which are set and used by a VT control function can be different from the locale settings used by TC or DL control functions. In order for clients to configure their device descriptors to the language, formats, and units used on the TC or DL user interfaces, clients can query these control functions for their standard setup and can adapt the device descriptor text attributes and value presentations to the language, formats, and units of the TC or DL. The device descriptor text attributes and DeviceValuePresentations shall be set such that the TC or the DL can use these values to display configuration and operating information about the client to the operator correctly on the user interface used by the TC or DL.

The following requirements are applicable to TCs and DLs that operate in an ISO 11783 network in which a VT is present.

a) The TC shall send the standard language, format, and measurement units messages defined in ISO 11783-7, hereafter referred to as “standard setups” (Note this term is defined in ISO 11783-6). The TC shall use the standard setup as defined and received from the primary connected VT (VT that will display values from the clients). If a standard setup has not been determined by a connection to a VT (as would be the case in a factory-new TC), the TC shall use its default standard setup defined by the manufacturer until a connection to the VT is established.

b) The TC shall report standard setup during client initialization and at any time when there is a change. The change may occur due to an operator change of the settings of the VT to which the TC is connected or the TC being connected to a different VT with a different standard setup. Standard device descriptor modification methods and rules shall apply to allow the client to modify its device descriptor to match the TCs VT operator-selected language — by updating text fields and DeviceValuePresentation (DVP) values.

c) The TC shall store the standard setups in non-volatile storage and restore the values during initialization prior to connection to a VT. Once a connection with the VT is established, then the TC follows the setup of the VT that it connects to.

d) The TC shall respond to ISO 11783-7 “Language Command” requests sent to the global address (GA). This requirement is introduced in ISO 11783-10 version 4.

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

e) The TC shall respond to ISO 11783-7 "Language Command" requests directed to the TC. This requirement is introduced in ISO 11783-10 version 4.

If the TC does not make use of a VT, then the TC shall provide a proprietary means to configure its language and units.

The following requirements are applicable to clients:

a) Clients shall configure their standard setup according to the TC to which they are publishing the device descriptor. This requirement is introduced in ISO 11783-10 version 4. ISO 11783-10 version 3 and prior clients followed the VT or T-ECU setup.

b) Clients shall use their default language if the selected language is not supported by the client.

6.6 Connection management

Upon power-up, a specific sequence of events shall occur, in order to ensure proper initialization of the TC and clients, as described in 6.6.1 and 6.6.2 and depicted in Figure 2.

6.6.1 Task controller initialization

The TC shall complete the following initialization.

a) The TC shall complete the address claim procedure in accordance with ISO 11783-5 and shall also send a request for an address claimed to the global destination address (255).

b) The TC shall wait for 6 s after completing the address claim procedure.

c) The TC shall begin transmission of the Task Controller Status message.

d) The TC shall allow clients to upload and initialize their device descriptor object pools.

e) The TC shall request the version of the client after a client has requested the version of the TC. This requirement has been introduced in version 4. Prior to version 4, the TC was not required to request the version of the client.

6.6.2 Client initialization with the task controller

The client shall complete the following initialization.

a) The client shall complete the address claim procedure in accordance with ISO 11783-5.

b) The client shall wait for 6 s after completing the address claim procedure.

c) The client shall wait until its selected TC begins transmission of the Task Controller Status message.

d) The client shall identify itself as a Working Set Master with its members to its selected TC, using the Working Set Master and Working Set Member messages given in ISO 11783-7. Prior to version 3, communication between a TC and a Working Set was not limited to TC to Working Set Master communication only. In version 4 and later, the client is still required to identify itself via the Working Set Master message but the communication with a TC is limited to the Working Set Master as the client CF only.

The client may send these messages for other purposes (e.g. VT initialization).

e) The client shall begin transmission of the Client Task message.

© ISO 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

f) A version 4 and later client shall query the version of its selected TC to determine its capabilities with a Request Version message (B.5.2). Prior to version 4, this request version step was recommended when the client intended to use version 3 or later features.

If the TC does not support the required ISO 11783-10 version or features, then the client shall limit its version and features to those that are available in the TC.

g) A version 4 and later client shall wait for and respond to a Request Version message transmitted by a version 4 and later TC. This enables the TC to process information from the client correctly.

h) The client may request the language and format messages from its selected TC.

i) The client shall query its selected TC to determine if its device descriptor object pool already exists.

j) The client shall either

1) activate the device descriptor object pool at the TC when the device descriptor object pool already exists in the TC, or

2) commence and complete a transfer of the device descriptor object pool to its selected TC and activate the device descriptor at its selected TC, accomplished using the messages defined in Annex B and either the transport protocol (see ISO 11783-3) or extended transport protocol (see ISO 11783-3), depending on the size of the device descriptor object pool.

6.6.3 Connection maintenance

The TC shall transmit a cyclic Task Controller Status message at an interval of 2 s. The TC shall also send a Task Controller Status message immediately whenever the task status changes or the value in any of the other Task Controller Status message bytes changes, although at least 200 ms shall elapse between Task Controller Status messages (the maximum transmit rate for Task Controller Status message is 5 Hz). This message includes an indication of the current task status and is sent to the destination-specific global address (GA). If the client does not receive this message for at least 6 s, it assumes a possible uncontrolled shutdown of the TC and stops sending the Client Task message. The client may re-establish connection to the TC by restarting the initialization procedure.

All clients that maintain a connection with a TC shall indicate their presence by transmitting to the TC a cyclic Client Task message at an interval of 2 s. The client shall wait at least 6 s after finishing the address claim procedure before transmitting the Client Task message. This timeout enables the TC to detect a restart of a client. If the TC does not receive this message for at least 6 s, it assumes a possible uncontrolled shutdown of the client.

All time values mentioned in ISO 11783-10, e.g. the 200 ms and 2 s intervals and 6 s time-out are the time values that shall be used for the implementation of this communication protocol. The accuracy of the implementation of these time values is subject to test requirements defined by AEF.[[2]]

The Task Controller Status message and the Client Task message are defined in Annex B.

When a client is restarted, or starts and connects to the TC during an active task, the TC shall accept the upload and activation of the client's device descriptor and issue the measurement commands applicable to this client.

The initialization and shutdown procedures of the connection with the TC are specified in Figure 2. The step "Request Localization Label" shall be skipped when the TC has indicated in the "Structure Label" answer that no device descriptor for the originating ECU is present in the TC.

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

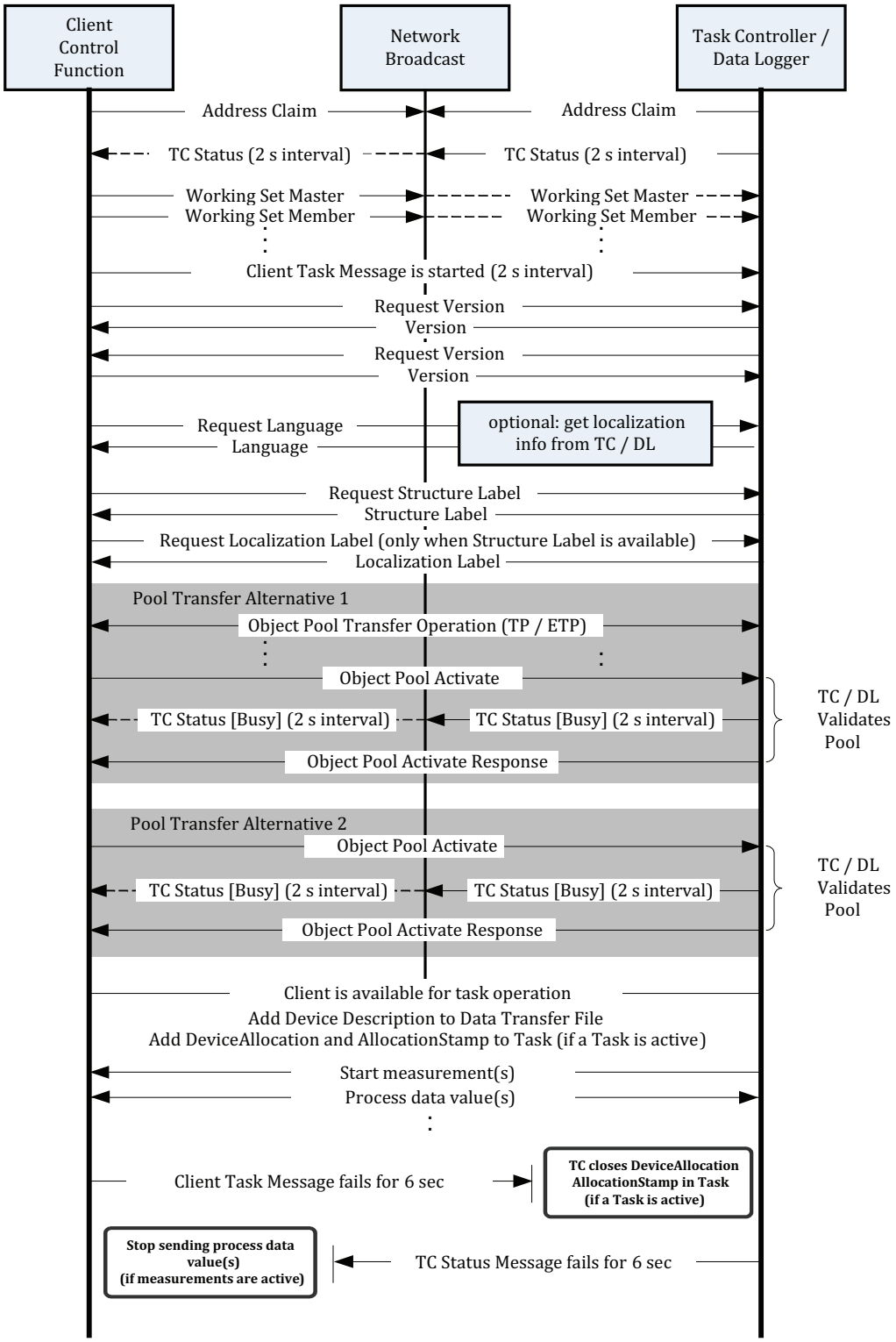


Figure 2 — Initialization and shutdown of connections

6.6.4 Task controller connection shutdown

The system shutdown is defined as the period of time when the Key Switch state indicates the tractor key is off and the ECU Power remains on. Actuator Power may or may not remain on concurrent with ECU Power (see ISO 11783-7).

© ISO 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

When the Key Switch state indicates that the tractor key has been turned off, and while ECU Power has not been terminated, the CFs in a system are able to transition to a shutdown state that is appropriate for that system. In some CFs, this can cause immediate termination of all network communications, where other CFs can request power to be maintained for a more orderly shutdown. Others could ignore the Key Switch state and continue normal operation until power is interrupted.

The following practices are introduced in version 4 and are mandatory for version 4 compatible CF implementations. Version 3 and prior CF implementations can be encountered in a system and these are identified and described in the following practices.

6.6.4.1 Task controller shutdown behaviour

The TC can expect that version 3 and prior clients can terminate communications without warning. The mandatory behaviour of the TC is to monitor the Key Switch state and take the following actions as a result of the transition from ”Key switch not Off” to “Key switch Off”.

a) The TC shall disable its “unexpected client shutdown detection logic” to avoid unnecessary notification to the operator as a result of one client shutting down immediately while another client maintains the ECU Power beyond the normal TC 6 s timeout (see 6.6.3).

b) The TC shall maintain services for a minimum of 2 s following the last “Maintain Power” request from those clients that have device descriptor object pools active in the TC. This ensures that data being collected from clients can be finalized correctly during the system shutdown period.

c) The TC shall continue to monitor the Key Switch state and shall reinitialize if “Key Switch Off” changes to ”Key Switch Not Off”. In this case, the TC shall ensure that the Task Controller Status message is discontinued and the standard Initialization process is performed (see 6.6.1).

NOTE ISO 11783-10 version 2 did not specify shutdown behaviour and ISO 11783-10 version 3 did not specify the mandatory requirements, therefore, a version 3 or prior TC may discontinue all communications with the network, including discontinuing the Task Controller Status message.

6.6.4.2 Client shutdown behaviour

Client behaviour can vary significantly depending on the design of the specific client. One variation of a client design implemented according to ISO 11783-10 version 3 and prior is not to monitor the Key Switch state and continue as normal until power is lost.

The mandatory behaviour of a client is to monitor the Key Switch state and take the following actions as a result of the transition from “Key switch not Off” to “Key switch Off”:

a) The client shall send a “Maintain Power“ message (see ISO 11783-7) to inform the system of the state of the client and can use this message as the means to request that ECU Power and/or Actuator Power be maintained.

b) The client monitors the “Maximum time of tractor power“ parameter (see Wheel-based speed and distance message in ISO 11783-7) and uses this information during any power management processes it executes.

c) The client shall send a Connection Deactivate command to the TC to eliminate the possibility of an unexpected shutdown indication (see B.6.10).

d) The client shall not consider the lack of the Task Controller Status message or other TC to client messages as an unexpected shutdown of the TC, and therefore shall not attempt a connection to any other TC as can be available.

e) The client shall continue to monitor the Key Switch state and reinitialize if it changes from ”Key Switch Off” to ”Key Switch Not Off”.

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

6.6.4.3 Task Resume upon Power cycle

When the TC starts after a power cycle, it shall resume the status that a Task was in before the power cycle occurred. It is recommended that all TC functions are resumed after the power cycle. Data logging and position-based control shall be activated automatically or the operator shall be informed to acknowledge further operation of these functions if these were active before the power cycle occurred.

If the TC implementation cannot resume the task functions automatically, it shall prompt the operator that a task functionality was active before power down and that reactivation is requested.

Note that the clients are responsible for allowing or disallowing execution of control commands.

6.7 Task controller number

The Connection Management rules require that there shall always be a TC with function instance zero (0) on the network if a TC is present on that network. TCs shall be factory set to function instance zero (0), but shall retain the function instance as configured by the operator. A mechanism is required to conveniently resolve conflicts in the case where there are multiple TCs with the same function instance and in the case where there is no TC with function instance zero connected to the network. The TC shall be responsible for providing a proprietary means for setting the function instance from the TC. This means shall ensure that duplicate function instances between TCs will not be created. A newly set function instance shall not be used until a reconnect of the TC to the network is performed.

The TC with function instance zero (0) is defined as the "primary TC". The proprietary means to set the function instance shall represent to the operator a TC number (See Clause 3). This ensures that TCs from all manufacturers will present a consistent numbering scheme for the operator to choose the primary (and secondary) TC(s). The TC number shall be in the range of 1 to 32, corresponding to function instances 0 to 31. TCs shall then be referenced as TC number 1, TC number 2, etc. A TC number equals TC function instance plus 1. The offset of 1 is for the support of operators, which may not be familiar with a zero-based numbering system.

The TC with function instance not zero (>0) shall follow the same connection procedure as specified in 6.6.

NOTE In case the TC is a component inside an ECU which has the ability to fulfil more than one control function, it is recommended that the TC control function can be configured by the operator to be active or inactive.

6.7.1 Client initialization on networks with multiple task controllers

Normally a client shall connect to a TC with function instance 0. If a client has a means to perform a "Move to another Task Controller" function on networks with multiple TCs, it can connect to a TC with a function instance >0. This function shall allow movement of the client to each of the available TCs in sequence. For example, this function could be accomplished with a "Next Task Controller" soft key or button in the user interface. The function behaves as follows:

a) "Move to another Task Controller" is enabled if the client detects more than one TC on the network.

b) When "Move to another Task Controller" is activated, the client shall perform the following:

   1) set to a safe state, or prevent activation of this feature unless it is in a safe state;

   2) send the Connection Deactivate message to the TC and wait for the response;

   3) stop sending the Client Task message to the TC;

   4) start the initialization process with another TC on the network.

The client shall save the new TC as the preferred TC for a next power cycle. If the preferred TC is not available within a certain time period after startup, the client can initialize connection to any other TC

© ISO 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

on the network. The client can provide a means for the operator to set the maximum wait time period or it may be obtained from the boot time specification in the "Version message" of the preferred TC.

NOTE A client shall connect only to a single TC and to a single DL at the same time. Connections from one client to multiple TCs or multiple DLs are not permitted in this edition of this part of ISO 11783.

6.8 Data exchange on the network

The TC converts data from the data transfer file set into process data messages to control the devices. These process data messages contain commands and values sent to the participating client control functions. The TC performs calculations to schedule the process data messages to be transmitted to the required addresses on the ISO 11783 network. An example of these calculations are site-specific applications such as when the TC looks up the position of an element of the client in the application grid, combines it with the operation delays of this client, and sends the relevant data to this client. In the opposite direction, the TC processes the process data messages sent by the participating clients and converts these process data variables to task data in order to return these to the data transfer file set.

All TC specific data exchange on the ISO 11783 network is based on process data messages.

It is recommended that control-related process data communication, e.g. section work status data, is triggered on change instead of using a fixed short time interval. The recommendation is to implement sending this type of data with an on change trigger as primary method in combination with a long time interval trigger as a fall-back mechanism. To avoid using an excessive amount of bandwidth on the network for process data communication and to ensure that process data requests and commands are acknowledged, the following process data exchange rules are defined.

a) A maximum of 10 process data messages per process data variable per second can be transmitted for each client to TC connection. This limitation addresses the distribution of process data message bandwidth over the different process data variables. This maximum is a generic requirement. Specific control scenarios that require a short burst such as controlling work states when entering or leaving a headland may exceed this maximum temporarily. These exceptions apply to specific DDIs such as the Setpoint Condensed Work State and the Actual Condensed Work State and are specified in the DDI definitions listed in the ISO 11783-11 data dictionary. Exceptions shall be explicitly listed in this data dictionary.

b) Requests that are transmitted by the TC shall be replied upon by the client before a next request is allowed to be transmitted by the TC to the same client. This synchronization of request-response process data message pairs has been introduced in ISO 11783-10 version 4. Prior to ISO 11783-10 version 4, a block of requests could be transmitted by the TC without waiting for individual responses throughout the block of requests.

c) All measurement commands (Command values 4 to 8) transmitted by the TC shall be acknowledged by the client with a PDACK message. The measurement command shall be replied to by the client before a next measurement command is allowed to be transmitted by the TC to the same client. The measurement command acknowledgement can be a positive acknowledgement when the measurement is supported and executed by the client or a negative acknowledgement when the measurement command is not accepted by the client. In case the measurement command is accepted by the client and the command starts a measurement, the client shall also transmit the value that the process data variable had at the time of starting the measurement for that process data variable to the TC. This ensures that the exact initial value of the process data variable measurement is communicated to the TC. The positive acknowledgement and the transmission of the initial process data value are requirements that have been introduced in ISO 11783-10 version 4. Prior to version 4, only a negative acknowledgement was required to be transmitted in case a measurement command could not be accepted by the client.

d) The PGN response time and time to wait before a request retry as specified in ISO 11783-3 also apply to process data communication. When a TC periodically sends a set of request process data value messages as a fallback method to acquire data when a client does not support a certain measurement, then the client shall respond per these requirements and a TC shall follow the specified retry procedure.

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

Clients shall initialize their internal process data variables to the correct operational values before activating a device descriptor object pool. For example, in case a client's internal data initialization process takes a certain amount of time to finalize before the correct device configuration values are available, then the device descriptor object pool shall not be activated prior to finalizing this data intialization. Thus, a TC requesting the device geometry upon activation of the device descriptor object pool will receive the correct operational values from the client.

The TC can generate process data messages containing process data variables that are not specified in a task data file. The process data messages control operation and data logging of the participating client control functions. The TC shall only send or request process data variables that are supported by the client control functions. Examples of this type of control are the use of sensor systems and the use of recorded spatial operation of client control functions to control client control functions.

Clients shall only transmit process data value commands to a TC that have been requested by

a) a request value command,

b) the default data logging method, and

c) individual measurement commands.

The interaction between the handling of measurement commands and the state changes of the "Task totals active" bit is as follows:

a) The Task Controller Status message "Task totals active" bit value does not put restrictions on the sending or responding of Process Data commands, except sending a default data logging command.

b) If the TC sends specific Process Data Measurement commands when the "Task totals active" bit value is set to 0, then the TC may also stop measurements by sending specific measurement commands to stop these measurements.

c) In case a TC has started measurements when the "Task totals active" bit is 0, then the TC shall not stop these measurements just before it changes the "Task totals active" bit from 0 to 1.

d) When the "Task totals active" bit goes from 0 to 1 or from 1 to 0, all measurements that were specified by the requesting CF that is transmitting this Task Controller Status message shall be stopped by the client. The TC may retrigger the measurements by sending specific Process Data Measurement commands or by sending the default data logging trigger.

In ISO 11783-10 version 3 and later, it is specified that a TC can start a measurement while a task is not active. In ISO 11783-10 version 2 and earlier, TC clients may or may not support the start of a measurement while a task is not active.

6.8.1 Site-specific application

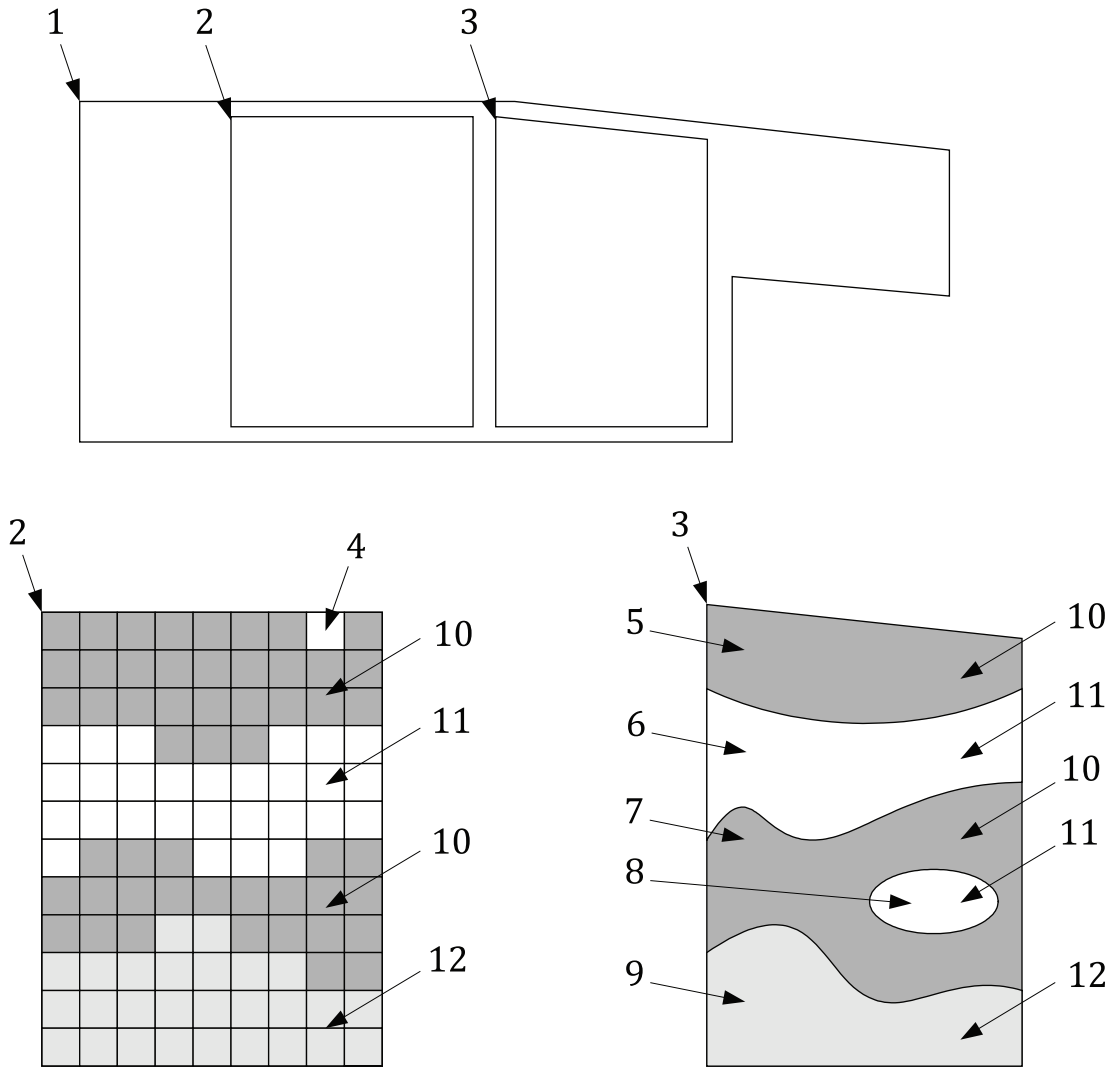
Site-specific applications require the TC to schedule sending of process data messages according to the actual location. For matching of the actual location with ProcessDataVariable XML element definitions, the geometry for site-specific process data has to be specified in the task data. The geometry definition is either a gridcell or a polygon and shall be labelled with a unique identification. Grid cells and polygons refer to a TreatmentZone to which the site-specific process data variable values are related. When the relevant DeviceElement enters a new TreatmentZone, the new set point values associated with that TreatmentZone are sent over the ISO 11783 network to the appropriate client.

Of the geometry definitions, gridcells have constant length and width dimensions. The gridcell location is relative to the origin of the grid to which a gridcell belongs. The structure and identification of gridcells are specified in the XML element Grid. Polygons can be used to define irregularly shaped TreatmentZones. The XML elements Polygon, LineString, and Point are used to define this type of TreatmentZone. Figure 3 presents a comparison of both types of TreatmentZone definition and Figure 4 the grid definition in more detail. On the right hand side in Figure 3, TreatmentZone 1 is represented by two Polygons that describe two single-bounded areas. The first of these two Polygons (Polygon 1) has a single Linestring bounding its area whereas the second of these two Polygons (Polygon 3) is defined

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 11783-10:2015

by two Linestrings, an outer boundary and an inner boundary. TreatmentZone 2 is also described by 2 Polygons, Polygon 2 and Polygon 4, both Polygons only have outer boundaries.

When using overlaying polygons, the TC shall always use the exterior polygon definition for a given location. Each interior boundary describes a hole in the exterior polygon.



Key

1	frame partfield	7	polygon 3
2	partfield A	8	polygon 4
3	partfield B	9	polygon 5
4	gridcell	10	treatment zone 1
5	polygon 1	11	treatment zone 2
6	polygon 2	12	treatment zone 3

Figure 3 — Frame partfield, partfield, gridcell, polygon, treatment zone relationship

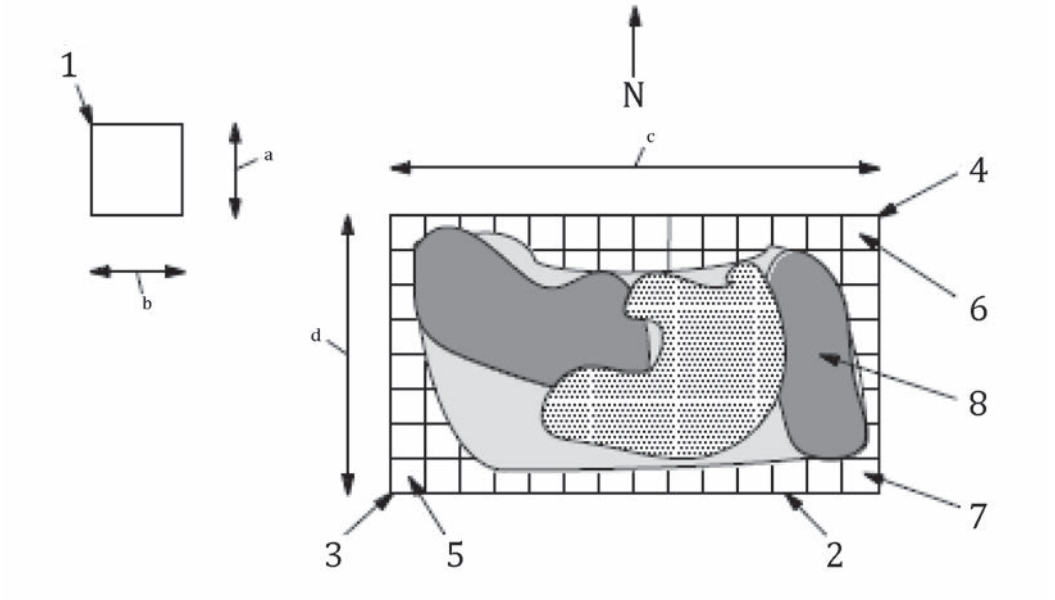
6.8.1.1 Grid based site-specific applications

Inside the XML data definitions, the Grid is defined by the GridMaximumColumns and the GridMaximumRows attributes, the GridMinimumNorthPosition and GridMinimumEastPosition, and the size of each grid cell. The grid cell ordering inside the binary file shall start with the GridMinimumNorthPosition/GridMinimumEastPosition, in ascending order of the columns (moving

© ISO 2015 – All rights reserved

AVA Documents Document Sharing Hub
Full document details: BS ISO 11783-10:2015

east), then proceeding in ascending order (moving north) on each row, beginning each time at the GridMinimumEastPosition column for each row. Figure 4 shows an example of a Grid and its grid cells.



Key

1	grid cell	7	fourteenth grid cell (row 0 and column 13)
2	frame partfield	8	treatment zone
3	origin (GridMinimumNorthPosition, GridMinimumEastPosition)	a	GridCellNorthSize.
4	FrameNorthMax, FrameEastMax	b	GridCellEastSize.
5	first grid cell (row 0 and column 0)	c	GridColumns.
6	last grid cell (row 7 and column 13)	d	GridRows.

Figure 4 — Definition of frame of grid for partfield

Grid cell definitions data are stored in binary format as separate files. The reference to the separate file is specified in the XML element Grid. There can only be a single Grid XML element per task, referring to both a binary and an XML file of a unique name prefix for all tasks of the data transfer file set. The uniqueness of grid file names shall be guaranteed by the FMIS. Each grid file definition results in two separate files — one to contain the binary data and the other the XML-coded header structure of the binary data set. The header structure defines the maximum data per binary record and enables the correct interpretation of the binary data. The binary file suffix is to be set to ".bin"; the suffix of the XML header structure file is to be set to ".xml".

See 8.6.2 for the encoding details of the XML header and binary data files.

6.8.1.2 Polygon based site-specific applications

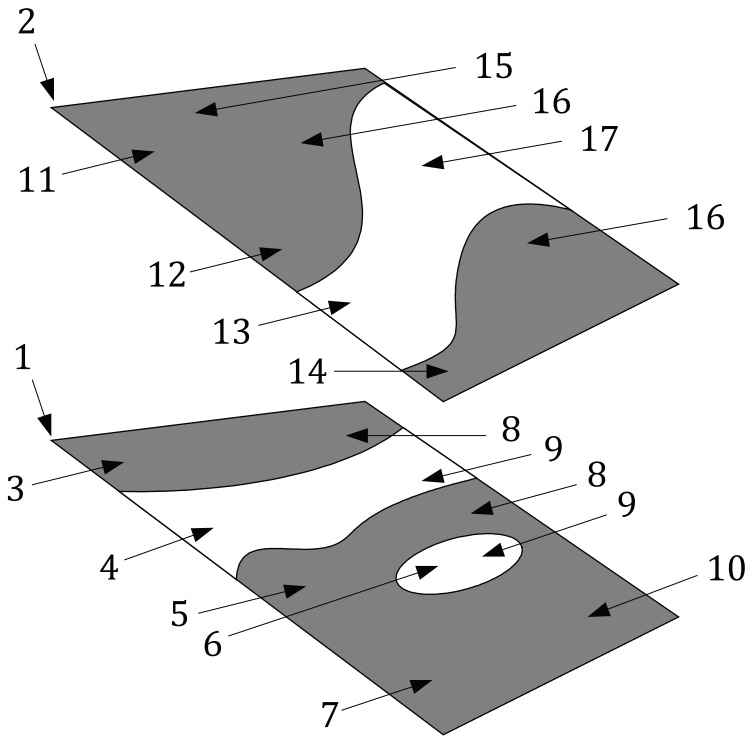
The geometry of the different TreatmentZone XML elements can also be specified by polygons. A separate set of polygons shall be defined for each site-specific operation of the task, i.e. each individual product or each individual spatially variable setpoint type has its own set of polygons that define the boundaries between the setpoint values for different treatment zones. For examples of single and multi-layer site-specific tasks with the geometry specified as polygons see D.41. An example of a dual-layer site-specific task is depicted in Figure 5 and the XML encoding for this example is provided in this section.

To transmit a setpoint value to a client, the TC shall determine for each device element that has a setpoint device process data assigned to the ProcessDataVariable XML elements of a variable rate layer whether it is located in one of the TreatmentZone XML elements of that layer.

© ISO 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

TreatmentZone XML elements may contain multiple polygons. Each polygon specifies an area bounded by one exterior boundary for which the ProcessDataVariable of that TreatmentZone is valid.

When polygons are used to define the geometry of the TreatmentZone XML element then only a single ProcessDataVariable XML element shall be included in that TreatmentZone. With this constraint, at maximum one variable rate is specified for the group of TreatmentZones and polygons that define one variable rate layer.



Key

1	layer 1	10	treatment zone 3
2	layer 2	11	polygon 6
3	polygon 1	12	polygon 7
4	polygon 2	13	polygon 8
5	polygon 3	14	polygon 9
6	polygon 4	15	treatment zone 4
7	polygon 5	16	treatment zone 5
8	treatment zone 1	17	treatment zone 6
9	treatment zone 2		

Figure 5 — Dual layer polygon based site-specific application

See D.50 example 2 for the XML encoding of this set of polygon-based treatment zones.

6.8.2 Data logging

The TC can also be used to log data for transfer back to the FMIS. For this purpose, process data variable values can be requested on the ISO 11783 network. Data log triggers can be specified by the FMIS in the data transfer file set to the TC. The TC is responsible for converting these data log triggers into process data measurement commands and for logging the replied process data variable values. The process data measurement commands for data logging are sent by the TC after starting and resuming a task. The values for the requested process data variables are provided by the connected clients, as long as the task is active. The sending of the requested values is stopped and measurements from the

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

TC are cancelled when the task is paused. When a client sends more data than requested by the TC for data logging, these additional data shall be ignored by the TC. The description of the XML element DataLogTrigger in Annex D provides more detail on the use of data log triggers.

Note that process data measurement commands can also be used by the TC to request process data when the "Task totals active" bit value is set to 0. This case is outlined in the introduction in 6.8.

6.8.3 Totals

Two different types of totals are distinguished: task totals and lifetime totals. A task total is controllable by the TC, i.e. when a task is started, the TC can set the value of this total to continue counting from. A lifetime total is not controllable by the TC and can only be requested and stored in the data transfer file set. Device process data objects that represent task totals shall have a data log trigger method of type "total" and a process data property bit that specifies that this process data object is settable. Device process data objects that represent lifetime totals shall have a data log trigger method of type "total" and the process data property bit "settable" shall be set to value 0.

A data log trigger with a data log method of type total shall be used to request the TC to store totals. Each total shall be stored once per Time XML element in a task. A task can contain multiple Time XML elements, e.g. in the case when a task is resumed, and each Time XML element can contain a set of totals. In addition to this, total values can also be stored more frequently in the TimeLog related data log files. When a task is resumed, the TC shall send the task totals stored in the most recent Time XML element to the client to continue counting from these task total values. When tasks are imported into the FMIS system, the most recent Time XML element shall contain all totals of the task. The total values may be requested by the TC from the client to obtain the latest total values when the task is stopped. Alternatively, when process data variables that represent total values are transferred as process data messages to the TC cyclically, such a query might not be necessary.

In the time period between the power up of a client and the first transmission of the "Task totals active" bit with value 1 the client shall keep its task total values initialized to a value of 0.

6.8.3.1 Handling of Task Totals

Task totals are required to be handled by TCs and may optionally be used by DLs. If a DL handles task totals, the requirements of handling task totals by a TC apply to that DL. Therefore, in this paragraph, the reference to a TC may be substituted by a DL if that DL supports task totals.

For task totals, it is the TCs responsibility to perform the following:

a) Set up an internal list of all task totals from all connected clients. To do so, the TC must parse the device descriptor object pools of all connected clients.

b) Request all those task totals from all connected clients if the task status changes from running to paused or completed and store them within a Time XML element.

c) Restore all those task totals to all connected clients if a task is resumed.

d) In order to avoid data loss in case of an unexpected shutdown, the TC shall request all task totals on a regular basis, e.g. at a 1 min interval. The TC may use a measurement to request these task totals on a regular basis. In order to support measurements for task totals, task totals are recommended to support the time and distance interval measurement methods.

e) The most recent Time XML element in a Task shall contain all task totals collected for that Task, even if some of the clients were not present during the most recent running period of a Task.

Task totals do not need to be in the default data set of a client because all task totals shall be handled by the TC. For this reason, task totals are recommended not to be added to the default data set of a client.

Definition of task totals behaviour regarding Start/Resume/Pause task events:

a) Task started: the client resets its task totals to zero.

© ISO 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

b) Task paused: the TC collects task totals from all clients, and all task totals retain their values.

c) Task resumed: this is to be handled similarly to the Start event for clients. After the task has been restarted, the TC sets all task totals to the previously stored task total values, sends out these values as process data variable values to all appropriate clients, and the clients count from these values onwards. (The TC is responsible for keeping track of previously collected task totals.)

The Task Controller Status message indicating the change of status of the task totals' active bit from inactive to active shall be sent before the task total values are set. The Task Controller Status message indicating the change of status of the task totals' active bit from active to inactive shall be sent before final task total values are requested.

It is the client's responsibility to default its task totals to a value of zero at client startup. The client logic will keep the task totals at zero as long as the task status is set to the counters are not active value (bit value = 0). Totals can be requested before a task is started; the value reported by the client will, in that case, be zero.

6.8.3.2 Handling of Lifetime Totals

Lifetime totals can both be requested and logged by TCs and DLs. When lifetime totals are handled by the TC or DL, it is the TC's or DL's responsibility to perform the following:

a) Set up an internal list of all lifetime totals from all connected clients. To do so, the TC or DL must parse the device descriptor object pools of all connected clients.

b) Request all those lifetime totals from all connected clients if the task status changes from running to paused or completed and store them within a Time XML element.

c) In order to avoid data loss in case of an unexpected shutdown, the TC or DL shall request all lifetime totals on a regular basis. The TC or DL may use a measurement to request these lifetime totals on a regular basis. In order to support measurements for lifetime totals, lifetime totals are recommended to support the time and distance interval measurement methods.

d) The most recent Time XML element in a Task shall contain all lifetime totals collected during that Task, even if some of the clients were not present during the most recent running period of a Task.

Lifetime totals are not recommended to be added to the default data set of a client. This enables TCs or DLs to handle the lifetime totals based upon the data collection requirements of the TC or DL.

Lifetime totals are not affected by the Start/Resume/Pause task events. The updating of lifetime totals is entirely controlled by the clients.

6.8.4 Data log triggers

Each process data variable of a certain device might be asked for, and can be logged, by the TC. The number and types of process data variables to be logged are specified by the XML element DataLogTrigger. The DataLogTrigger XML element can specify precisely which process data variables are required from certain device elements.

Alternatively, for a default data logging mechanism, a process data variable called RequestDefaultProcessData (DDI = $DFFF_{16}$) is specified. Within the default data logging mechanism, the clients are responsible to send the data that the clients want to be logged, at the interval and with the triggers that are specified by the client. Examples of cases for which this data logging mechanism is defined are the following:

a) A client has a large number of DeviceProcessData elements. The TC neither knows an appropriate selection nor a good interval for logging. By using the default data logging mechanism, the client itself decides which data are logged.

b) The FMIS can specify that it requires data logging to be activated by adding the RequestDefaultProcessData (DDI = $DFFF_{16}$) to each task.

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

In case no datalog triggers for specific DeviceProcessData are specified in a Task, the recommended practice is to include a DataLogTrigger XML element with the RequestDefaultProcessData DDI specified. This will ensure that at least the default data set from each client is logged. Due to that the default data logging can only be stopped by a transition of the "Task totals active" bit value from 1 to 0, the request default data command shall not be used by a TC when the "Task totals active" bit value is 0.

By using the identifier (DDI) of this process data variable in the DataLogTrigger XML element, the TC is commanded to request the specified devices to send all their default process data variable values from their default measurement method. The RequestDefaultProcessData DDI may only be requested from device element 0 of a device. The TC uses a request value command with DDI = $DFFF_{16}$ to request the sending of the default process data variables values. The set of process data variable values that belong to the default set of a device are specified in the device descriptor object pool. When the device descriptor object pool specifies certain DeviceProcessData objects to be part of the default set, a DeviceProcessData object with the RequestDefaultProcessData DDI shall be included in the device descriptor object pool of that device. The ProcessDataTriggerMethod attribute of this DeviceProcessData object shall be set to $1F_{16}$.

EXAMPLE 1 Device descriptor object pool containing the RequestDefaultProcessData DDI and a set of default data:

```
<DVC A="DVC1" B="Tiller" C="1.02*" D="A00484000B2CAF13" F="32A0FE34A56F00"
G="FF000000006E65">
    <DET A="DET1" B="1" C="1" E="0" F="0">
       <DOR A="2"/>
       <DOR A="3"/>
       <DOR A="4"/>
       <DOR A="5"/>
       <DOR A="6"/>
    </DET>
    <DPD A="2" B="0074" C="2" D="16" E="Area" F="9"/>
    <DPD A="3" B="0077" C="2" D="16" E="Time ON" F="7"/>
    <DPD A="4" B="008D" C="1" D="3" E="Work ON/OFF"/>
    <DPD A="5" B="0043" C="1" D="3" E="Width" F="8"/>
    <DPD A="6" B="DFFF" C="0" D="31"/>
    <DVP A="7" B="0" C="2.777778E-04" D="2" E="hr"/>
    <DVP A="8" B="0" C="1.000000E-03" D="1" E="m"/>
    <DVP A="9" B="0" C="1.000000E-04" D="2" E="ha"/>
</DVC>
```

In this example, the process data DDIs $8D_{16}$ and 43_{16} are part of the default set of data. This device supports the request default process data mechanism which is indicated by the DPD with object ID 6.

EXAMPLE 2 RequestDefaultProcessData DataLogTrigger in a task:

```
<TSK A="TSK1" E="PFD1" G="3">
    <DLT A="DFFF" B="31"/>
</TSK>
```

In this example, the task specifies the requesting of default data from connected clients that support the request default process data mechanism.

During data logging, there can be different frequencies of transmission of DDIs and also different time delays of replies to value request commands. The TC logs process data variable values at certain time intervals, and therefore the following rules apply.

a) Time and position data shall be logged at a maximum once per TimeLog instance.

b) Each process data variable value shall be logged at a maximum once per TimeLog instance. If data logging can only be performed at a low frequency because of TC capabilities, the value logged depends on the TC design.

c) If the TC is capable of data logging at the rate given by DDI with the highest update frequency, then a new data log record shall be started each time the next DDI with highest update frequency is

© ISO 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

received. All other process data values that are received between two values of the highest update frequency DDI are logged in the current record.

d) A received process data value shall be logged a maximum of once. If no new value is available for the next record, then no value shall be logged for that DDI.

Following these rules, logged data values in the same record can be separated by, at a maximum, the time difference that exists between two records. If several DDIs need to be grouped together, the process data variable "Log Count" (DDI = 0093_{16}) shall be used with the highest repetition rate to time tag the data logging of associated process data variable values.

EXAMPLE 3 Process data variable DDI "A" has the highest update frequency; process data variable DDIs "B" and "C" are added to current DataLog:

Client sends value of DDI A_1: TC stores value of A_1 in DataLog 1

Client sends value of DDI B_1: TC adds value of B_1 to DataLog 1

Client sends value of DDI A_2: TC closes DataLog 1; stores value of A_2 in DataLog 2

Client sends value of DDI C_1: TC adds value of C_1 to DataLog 2

Client sends value of DDI A_3: TC closes DataLog 2; stores value of A_3 in DataLog 3

Client sends value of DDI A_4: TC closes DataLog 3; stores value of A_4 in DataLog 4

EXAMPLE 4 Client desires to group process data variable DDIs: "A", "B", "C", and "D".

Client sends value of DDI (LogCount) LC_1: TC stores value of DDI LC_1 in DataLog 1

Client sends value of DDI A_1: TC adds value of A_1 to DataLog 1

Client sends value of DDI B_1: TC adds value of B_1 to DataLog 1

Client sends value of DDI C_1: TC adds value of C_1 to DataLog 1

Client sends value of DDI D_1: TC adds value of D_1 to DataLog 1

Client sends value of DDI LC_2: TC stores value of DDI LC_2 in DataLog 2

Client 1 sends value of DDI A_2: TC adds value of A_2 to DataLog 2

Client 1 sends value of DDI D_2: TC adds value of D_2 to DataLog 2

Client 1 sends value of DDI B_2: TC adds value of B_2 to DataLog 2

Client 1 sends value of DDI C_2: TC adds value of C_2 to DataLog 2

Client sends value of DDI LC_3: TC stores value of DDI LC_3 in DataLog 3

Client 1 sends value of DDI D_3: TC adds value of D_3 to DataLog 3

Client 1 sends value of DDI B_3: TC adds value of B_3 to DataLog 3

Client 1 sends value of DDI C_3: TC adds value of C_3 to DataLog 3

Client 1 sends value of DDI A_3: TC adds value of A_3 to DataLog 3

The FMIS can determine, based on the LogCounts, that DDIs A, B, C, and D all belong to the same group. There will always be a new DataLog started before the group is transmitted.

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

6.8.5 Peer Control

Peer Control is the mechanism by which any CF may be a setpoint value source to another CF as a setpoint value user such that the TC controls and records the assignment of sources to users.

Peer Control addresses the requirement for CFs in the ISO 11783 systems to accept setpoint values from sources that cannot be determined at the time that field operations are planned. These setpoint values need to be determined at run time and directed to a rate controller in a standardized manner without the specific knowledge of the setpoint value source or of the controller that is required to process these setpoint values. These sources of control information may include items such as on-the-go sensor systems. These systems may include determination of a rate directly by the setpoint value source or a modification of another rate such as that from a planned application map.

For this mechanism, both the setpoint value source and the setpoint value user shall be CFs that are TC "clients" as defined in this part of ISO 11783 and that upload and activate a device descriptor object pool. Similar to planned site specific applications, the setpoint value user device descriptor object pool shall include Device Process Data (DPD) object(s) that have their property defined as "Settable". The setpoint value source device descriptor object pool shall include a similar form of Device Process Data object with the property defined by a bit flag set as "Setpoint Source". The bits for Settable and for Setpoint Source are mutually exclusive. The setpoint value source may also include DPD objects defined as settable. The settable DPD in the setpoint value source can then be used in a "Map Overlay" function where a position dependent received setpoint value from the TC is modified by the setpoint value sources' specific algorithm and used as the setpoint value to be transferred to the final setpoint value user. The relations between the map source and peer source are depicted in Figure 6.

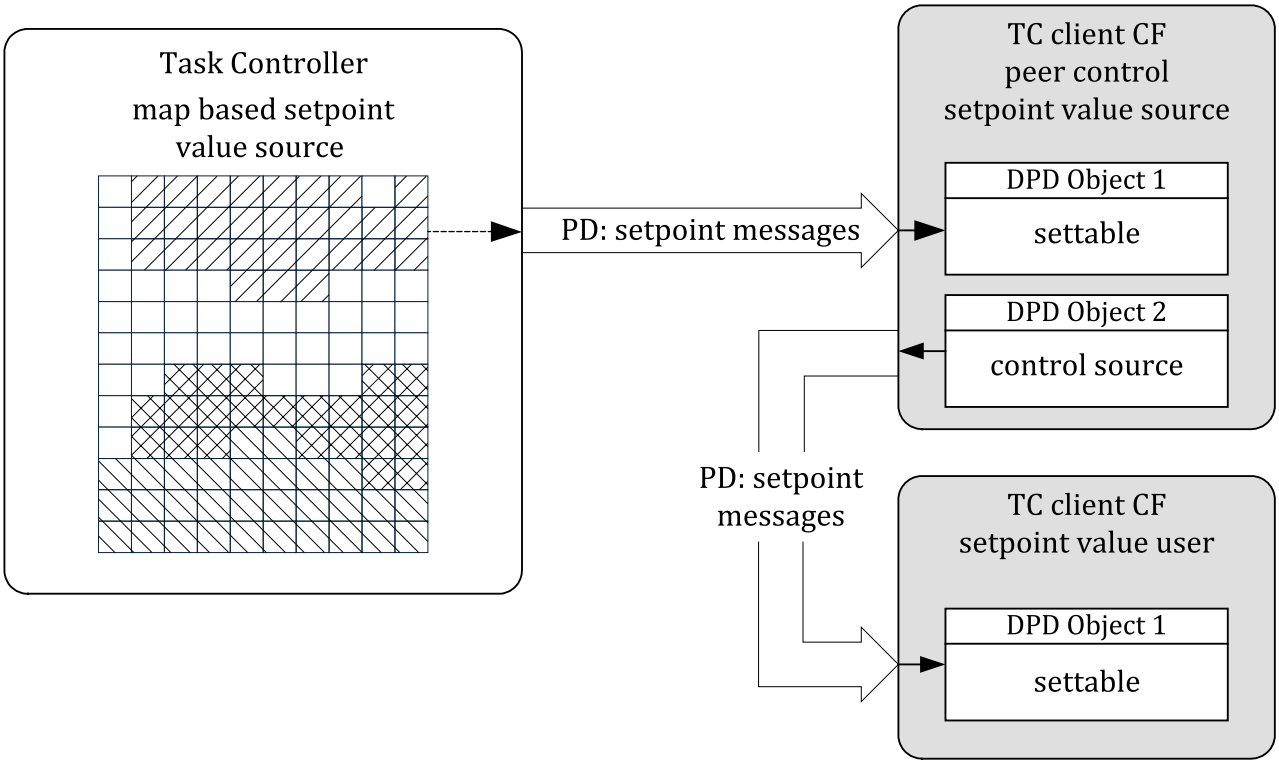


Figure 6 — Relations between a map-based setpoint value source, a peer control setpoint value source, and a setpoint value user

When a task is initiated in the TC, similar mechanisms are used as are available in ISO 11783-10 version 3 and prior TCs to automatically or manually assign controlling settable process data objects to setpoint data sources. In ISO 11783-10 version 3 and prior TCs, setpoint data sources include map layers from variable rate maps defined for a task or (optional) manually programmed control setpoints defined by the operator in the TC user interface. In ISO 11783-10 version 4 and later TCs, the setpoint data sources also include items from any CF that have a device process data element with the control source

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 11783-10:2015

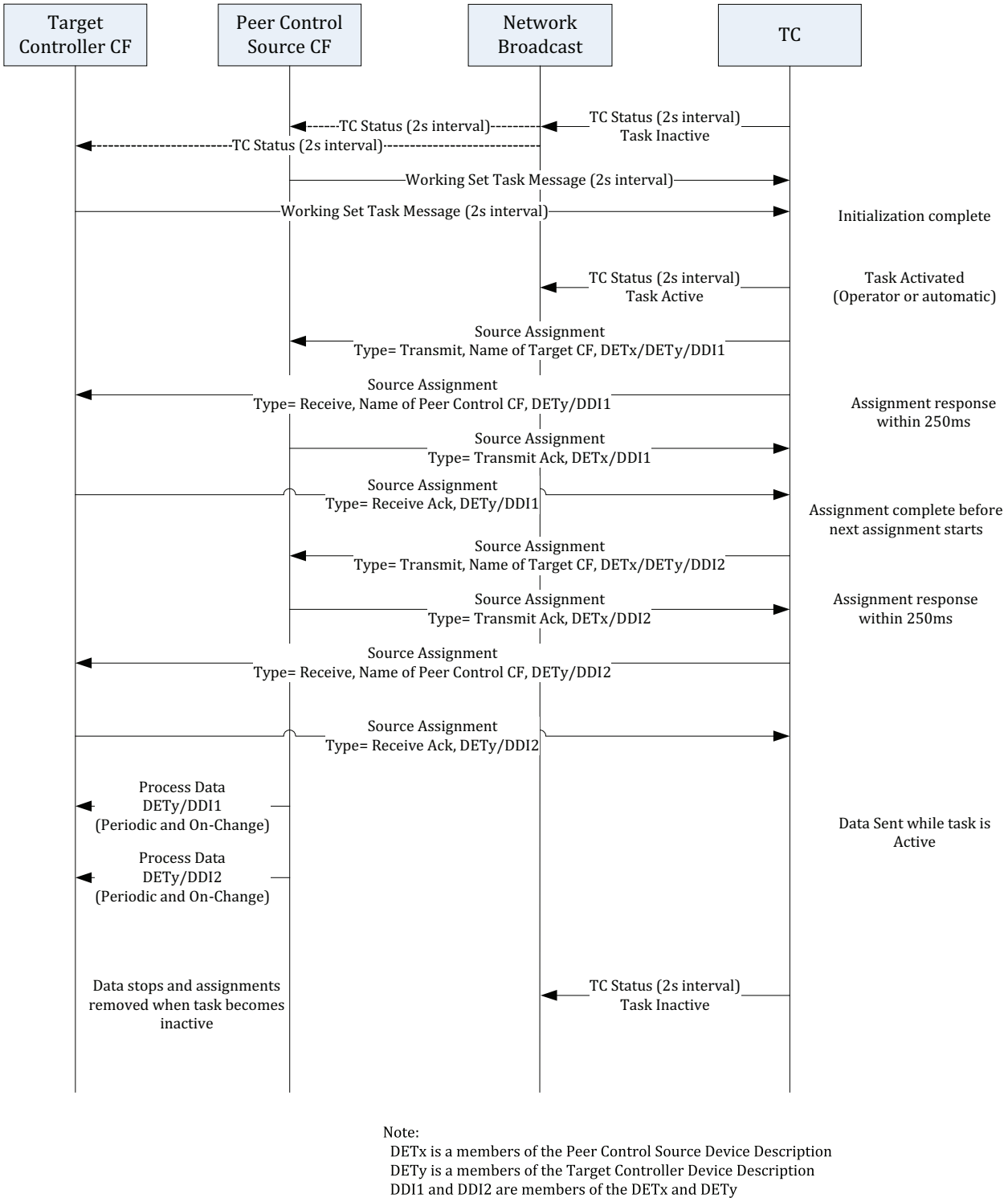
property set in their device descriptor object pool. All of the same assignment restrictions of DDIs and object properties apply to all setpoint data sources whether originating from an FMIS-planned source (TreatmentZone) or from another external source (CF with a device process data element with the control source property set). When the sources are assigned, the TC shall initiate assignment messages to both the setpoint value source and the setpoint value user. Each client shall respond to acknowledge or reject the assignment within 250 ms. For backwards compatibility, if the target controller does not send a response, the connection is treated as rejected. The assignments enable the setpoint value source to send the setpoint information directly to the setpoint value user without extra message latency and processing overhead of communicating through the TC. The assignment also enables the setpoint value user to qualify received control setpoint data as originating from its assigned setpoint value source CF. This assignment uses a command message from Process Data (PD) messaging. The message specifies the type of the connection (settable or control source property set), the Device Element, the DDI, and the NAME of the other CF. It is the responsibility of the assigned CF to track changes to the CF's SA's when new address claims occur. The assignment procedure is visually represented in Figure 7.

The assignment messages shall be sent following each change of task status from inactive to active. While the task is active, the control setpoint information from the setpoint value source CF shall be sent at a periodic rate of once per second and on-change with a maximum rate of 5Hz. If the control setpoint messages are not received for 3 s, the target controlling CF shall take action in the same manner as if the TC communications were lost. If the communications with the TC is lost (as determined by the periodic Task Controller Status message), both the setpoint value user and the setpoint value source take action based on lost TC communications and act as the task becomes inactive. When a task becomes inactive, the setpoint value user shall not accept setpoint messages, the setpoint value source shall stop sending messages and both CF's shall cancel the current assignments. A new set of assignment messages shall be sent at the event of a task becoming active for each device element assigned to an external source. If no assignment message is received during each active task session or the assignment is invalid, the setpoint value user shall default to only accepting setpoint messages from the TC while the task status is active.

An assignment may be made by the FMIS system and stored as part of the Task (TSK) element of the data transfer file set. If the assignment made by the FMIS is not available at the start of the task, the TC may make the assignment according to the same rules as defined for direct TC to controller usage. If the assignment is made by the TC the assignment shall be stored in the same manner such that the assignment is remembered for future start events of the same task and is available to the FMIS for reference. This assignment is stored by the ControlAssignment (CAT) XML element.

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015



Note:
DETx is a members of the Peer Control Source Device Description
DETy is a members of the Target Controller Device Description
DDI1 and DDI2 are members of the DETx and DETy

Figure 7 — Peer Control assignment procedure

7 Data logger requirements

7.1 General

A data logger (DL) can be installed on the network to perform non-task related data logging without the need for controlling a TC client. The DL is distinguished from a TC by a separate function definition (see ISO 11783-1).

© ISO 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

The DL uses a subset of the TC functionality. The DL has to fulfil the client connection mechanism requirements, the processing of device descriptor object pools, and the logging of data from process data as specified for a TC. In order to keep the CF responsibilities on the network clear, a DL shall not use all the TC functionalities that are described in Clause 6. The network communication rules specified in 6.8 do apply to DL-to-DL client communication.

The DL shall not

— use section control functionality,

— use position based control functionality,

— set up peer control assignments (note that a peer control CF can connect to a DL to supply values for data logging purposes), and

— send process data set command values for DDI's other than task totals to a DL client.

The DL may

— request process data from connected DL clients,

— issue measurement commands to each connected DL client,

— request default data logging from connected DL clients,

— request parameter groups to log parameters, and

— set and Request task totals and control accumulation of task totals via the "Task totals active" bit in its Task Controller Status message (the DL client shall maintain an individual set of task totals for the DL).

7.2 Connection management

The DL client supporting a connection to a DL shall initialize to a DL in the same way as described in 6.6.2. The DL client shall provide a device descriptor object pool to the DL. The device descriptor object pool of a client provided to a TC and the device descriptor object pool of the same client provided to a DL can differ. It is the client designer's responsibility to specify both device descriptor object pools and optimize the content for each purpose.

Multiple DLs can be present on a network. If multiple DLs are present on a network, each DL shall be able to be identified by a unique DL function instance. Configuring the function instance of a DL is, similar to configuring the TC function instance in case multiple TCs are present in a network, done via a proprietary means. The DL with function instance 0 does not have a role that is different from any other DL function instance on the network. In ISO 11783-10 version 4 and prior, a connection from a DL client to, at maximum, one DL is permitted.

The following network configurations are possible:

a) TC and TC client(s)

Every TC client supporting task controller functionality sets up a connection to the TC. This scenario has been available from ISO 11783-10 version 1 and onwards.

b) DL and DL client(s)

Every DL client supporting data logger functionality sets up a connection to the DL. This scenario is, connection wise, similar to scenario 1 and is introduced in ISO 11783-10 version 4.

c) TC and DL and TC/DL client(s)

Every CF supporting both task controller and data logger client functionalities connects to the TC and/or DL. The CF may support connecting to both the TC and DL servers simultaneously. This scenario is introduced in ISO 11783-10 version 4.

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

7.3 Measurements and totals

If a DL client transmits a device descriptor object pool containing process data values to a DL, it shall maintain a unique set of active measurement commands for the connected DL. In case a client connects to both a TC and a DL simulatenously, there shall be different sets of measurements commands active in parallel in one client.

If a DL client transmits a device descriptor object pool containing task totals to a DL it shall maintain a unique set of task totals for the connected DL. Thus, a client can have a set of task totals for a TC and a set of task totals for a DL, each accumulating in parallel. Each set of task totals may be set, reset, or triggered individually by the corresponding CF. There shall be no interference between these different sets of task totals. Note that even though a DL may not set or reset task totals often, this mechanism is useful for the DL to specify a start point for the accumulation of task total values and to avoid roll-over of task totals when a maximum value is reached.

8 Data transfer

8.1 General

The communication FMIS to MICS to FMIS is based on data transfer files. Each XML data transfer file is formatted according to XML definitions version 1.0. The XML files contain text only and are coded in UTF-8 (see ISO/IEC 10646). Optionally, binary-coded data files for gridcell definitions or logged process data can be part of the data transfer file set. All files shall be contained in the same directory.

Both coding data and task data are stored in the same set of XML files upon transmission from FMIS to the MICS. During processing of tasks by the TC, these files are likely to be modified and, when tasks are finished, can be transferred back to the FMIS.

8.2 Extensible Markup Language

XML is a language for the description of structured documents and data and represents a technological basis for data exchange. XML is a subset of SGML (standardized general markup language).

XML is a hierarchical set of XML elements. XML elements can contain one or more XML attributes. Inside the data transfer XML files, no text is allowed inside XML elements.

An XML element consists at least of an opening label, a number of attributes, and a closing label. For low memory and data transfer bandwidth requirements, abbreviations as specified in the description of the XML elements are used in the xml schema and in the xml files. The following line is an example of an XML element definition called *Worker*:

< WKR A = “WKR2” B = “Miller” >

< /WKR >

Opening and closing labels shall be of the same string, except that the closing label shall be preceded by a “/”. The case is significant in all uses of XML element labels, i.e. < worker > is not the same as < Worker >.

XML attributes carry information contained in XML elements. The syntax of an XML attribute is *attribute-name* = “*description*”.

The XML attribute label follows the same rules as an XML element label. The case is significant. The use of the equals sign is always required. The text string description shall be enclosed in quotes.

XML elements that do not contain child elements, but contain attributes, may be represented in a shortened form:

< WKR A = “WKR1” B = “Smith” / >

© ISO 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

If an XML file follows the rules of the XML syntax itself, it is said to be *well formed*. The elements of an XML document can be formally specified using a DTD. If a DTD exists for an XML document, XML parsers with the appropriate extensions can test the XML document against the DTD. A document that passes such a consistency and validation test is said to be *valid*.

XML data transfer files are always text files. Due to the hierarchical structure, XML files can only contain a single XML root element. The elements list of the root element specifies which XML elements might be used as primary elements. Primary elements consist of coding data elements and entities which are not related to a single task. The root element of a data transfer file set is named as ISO11783_TaskData. XML data transfer files shall always be well formed and valid; otherwise, the contents of the file cannot be processed.

8.3 Extensible schema definition

XML provides an application-independent way of sharing and transferring data. With a DTD, it is possible to verify that received data from the outside world is valid. Also, a DTD might be used to verify its own created data.

The purpose of a DTD is to define the legal building blocks of an XML document. It defines the document structure with a list of legal XML elements. A DTD can be declared inline in an XML document or as an external reference. An object-oriented language modelling approach is to use an XSD as DTD.

An XML schema

— defines elements that can appear in a document,

— defines attributes that can appear in a document,

— defines which elements are child elements,

— defines the order of child elements,

— defines the number of child elements,

— defines whether an element is empty or can include text,

— defines data types for elements and attributes, and

— defines default and fixed values for elements and attributes.

XML schemas are formatted in XML and well formed. A well-formed XML document is a document that conforms to the following XML syntax rules:

a) it shall begin with the XML declaration;

b) it shall have one unique root element;

c) all start tags shall match end tags;

d) XML tags are case-sensitive;

e) all elements shall be closed;

f) all elements shall be properly nested;

g) all attribute values shall be quoted.

Schemas support data types and namespaces. With the support for data types, it is possible to describe permissible document content, validate the correctness of data, work with data from a database, define data facets (restrictions on data) and data patterns (data formats), and convert data between different data types.

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

8.4 XML schema definition

The validity of XML data transfer files is defined by the schema ISO11783_TaskFile. The schema is based on http://www.w3.org/2001/XMLSchema definitions and data types. The ISO11783_TaskFile schema does not have an ISO 11783 specific namespace. The XML schemas are published at the following location: http://dictionary.isobus.net/isobus/file/supportingDocuments. Unique identifiers of XML elements and references to unique identifiers across XML elements are defined as data types ID and IDREF respectively. To create a unique namespace all over the data transfer file set, each XML element identifier shall be defined in its specific XML element namespace. Table 2 defines the valid namespaces of XML element identifiers.

The XML Schema definition has in its filename the version information of the ISO 11783-10 document. The naming convention for the XML Schema definition is: ISO11783_TaskFile_V[VersionMajor]-[VersionMinor].xml. For example, the XML Schema definition filename of the first published international standard is ISO11783_TaskFile_V2-0.xml; the XML Schema definition filename of the first revision to the first published international standard is ISO11783_TaskFile_V3-0.xml. The VersionMinor field in the ISO11783_TaskData XML element is used to enumerate revisions to the XML Schema definition within the same VersionMajor value. The XML Schema definition files are published on the Internet at URL http://dictionary.isobus.net/isobus/file/supportingDocuments. The XML element ISO11783_TaskData VersionMajor and VersionMinor attribute values are set to a fixed value in the corresponding XML Schema definition files.

Implementers of this standard are responsible for keeping their applications backwards compatible with older versions of the standard and their corresponding schemas.

The XML schema definition is to be applied to all XML data transfer files in both data transfer directions. All XML elements are considered as entities. Each entity shall have a unique identifier (if there is an identifier defined as the XML attribute for this element) over the whole data transfer file set. Each identifier starts with the appropriate namespace letters followed by, at a maximum, an 11-digit decimal number without leading zeroes. Unique identifiers are at least 4 bytes long and shall be provided by both the FMIS and the MICS. The values of 0 and –2147483648 for identifiers are reserved and shall not be used inside XML elements. Prior to ISO 11783-10 version 3, the value of 0 was not reserved and was assumed to be interpreted as a positive value identifying FMIS generated XML elements. In ISO 11783-10 version 3, the value of 0 has been introduced as reserved and shall not be used as XML element identifier.

On the MICS, no coding data XML element shall be changed or deleted. Non-coding data XML elements may be edited and changed. If the operator of the TC creates new entities of XML elements, then the TC is responsible for creating unique identifiers for all newly created entities of XML elements. To differentiate between those entities provided by the FMIS and those created on the mobile system, each newly created identifier shall have the appropriate namespace letters followed by a negative decimal number, e.g. “WKR–1”.

Table 2 — XML element types and acronyms

XML element designator	Coding data	XML element name
AllocationStamp		ASP
AttachedFile[a]	X	AFE
BaseStation[a]	X	BSN
CodedComment	X	CCT
CodedCommentGroup	X	CCG
CodedCommentListValue	X	CCL
ColourLegend	X	CLD
ColourRange	X	CRG

[a] This element is introduced in ISO 11783-10 version 4.

© ISO 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

Table 2 *(continued)*

XML element designator	Coding data	XML element name
CommentAllocation		CAN
Connection		CNN
ControlAssignment[a]		CAT
CropType	X	CTP
CropVariety	X	CVT
CulturalPractice	X	CPC
Customer	X	CTR
DataLogTrigger		DLT
DataLogValue		DLV
Device	X	DVC
DeviceAllocation		DAN
DeviceElement	X	DET
DeviceObjectReference	X	DOR
DeviceProcessData	X	DPD
DeviceProperty	X	DPT
DeviceValuePresentation	X	DVP
Farm	X	FRM
Grid		GRD
GuidanceAllocation[a]		GAN
GuidanceGroup[a]	X	GGP
GuidancePattern[a]	X	GPN
GuidanceShift[a]		GST
LineString		LSG
OperationTechnique	X	OTQ
OperationTechniqueReference	X	OTR
OperTechPractice		OTP
Partfield	X	PFD
Point		PNT
Polygon		PLN
Position		PTN
ProcessDataVariable		PDV
Product	X	PDT
ProductAllocation		PAN
ProductGroup	X	PGP
ProductRelation[a]	X	PRN
Task		TSK
TaskControllerCapabilities[a]		TCC
Time		TIM
TimeLog		TLG
TreatmentZone		TZN

[a] This element is introduced in ISO 11783-10 version 4.

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

Table 2 *(continued)*

XML element designator	Coding data	XML element name
ValuePresentation	X	VPN
Worker	X	WKR
WorkerAllocation		WAN
ExternalFileContents		XFC
ExternalFileReference		XFR
[a] This element is introduced in ISO 11783-10 version 4.		

Any XML attribute of type IDREF may only contain a single identifier reference.

8.4.1 Proprietary XML schema extensions

Manufacturer proprietary XML elements and XML attributes in the data transfer file set shall be preceded by a string consisting of a character "P", the decimal manufacturer code, and an underscore character "_". The manufacturer proprietary XML element names and tags shall be at maximum 16 characters in length. The data length of each manufacturer proprietary XML attribute shall not exceed 64 characters.

EXAMPLE 1 A proprietary XML element designated "MyElement" in the data transfer file set of a manufacturer with manufacturer code 500 receives the XML element name:

"P500_MyElement".

EXAMPLE 2 A proprietary XML attribute designated "MyAttribute" from the same manufacturer receives the XML attribute name:

"P500_MyAttribute".

The contents of the manufacturer proprietary XML elements and XML attributes shall be ignored when the data transfer file set is processed by a MICS or FMIS of another manufacturer. On the MICS, when the data transfer file set originating from the FMIS contains proprietary XML elements or XML attributes, these XML tags and their values may be omitted in the data transfer file set originating from the MICS back to the FMIS.

The XML element AttachedFile (AFE) is introduced in ISO 11783-10 version 4 to enable transfer of manufacturer proprietary data in separate files that are linked to the standard data transfer file set.

It is recommended that the amount of proprietary data are kept as low as possible. Reasons for keeping the amount of proprietary data low are that this data cannot be interpreted in multi-manufacturer systems and may not be retained. Proprietary data are allowed for reasons of being able to extend the standard for manufacturer internal use.

8.5 XML data transfer files

The name of the main XML coded data transfer file containing the root XML element ISO11783_TaskData shall be set to "TASKDATA.XML". When the data are transferred using a removable storage device, the MICS shall support access to this file in a directory named "TASKDATA" located in the root directory of the transfer medium. Both this directory name and the data transfer file names are case-sensitive and all the characters are in upper case. Any other medium used to transfer the data between FMIS and MICS is proprietary (see Figure 1).

The main data transfer file contains coding data and a number of tasks. Figure 8 illustrates the structure of XML elements in the data transfer file set. There can be references to other XML files containing coding data or tasks inside the main data transfer file. References to XML files are accomplished by use of the XML element XFR. XML file references cannot be nested. This means that only the main XML file can include XML elements XFR. There can be a single XML element in the referenced external XML file. The elements shall be embedded in an XML root element named XFC. A referenced XML file shall only

© ISO 2015 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO 11783-10:2015

contain top-level XML elements of the same type — for example, it may include definitions of customers but not of customers and of partfields together. Top-level XML elements may be defined both in the main and in the referenced XML files.

Each XML coded data transfer file shall start with an XML identification section and shall be well formed:

```
< ?xml version = ”1.0” encoding = ”UTF-8”? >
```

In the root data transfer file, all XML elements are always embedded in the global root structure:

```
< ISO11783_TaskData VersionMajor = ”...” VersionMinor = ”...”
TaskControllerManufacturer = ”...”
       TaskControllerVersion = ”...” ManagementSoftwareManufacturer = ”...”
ManagementSoftwareVersion = ”...”
       xmlns:xsi = ”http://www.w3.org/2001/XMLSchema-instance”
       xsi:noNamespaceSchemaLocation = ”… ” DataTransferOrigin = ”...” >
< /ISO11783_TaskData >
```

The number of coding data elements supported by the recipient of a data transfer file set shall, at a minimum, be 2 000 per element type, and the overall number of coding data elements supported by the recipient shall, at a minimum, be 20 000. A data transfer file set containing more than 2 000 coding data elements per element type or more than 20 000 coding data elements is at risk of not being able to be processed by a recipient. This supported data elements limitation does not apply to data elements that are not classified as coding data (see Table 2). An example of data elements not classified as coding data are the geometry elements, e.g. the number of Points in a boundary is not limited.

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

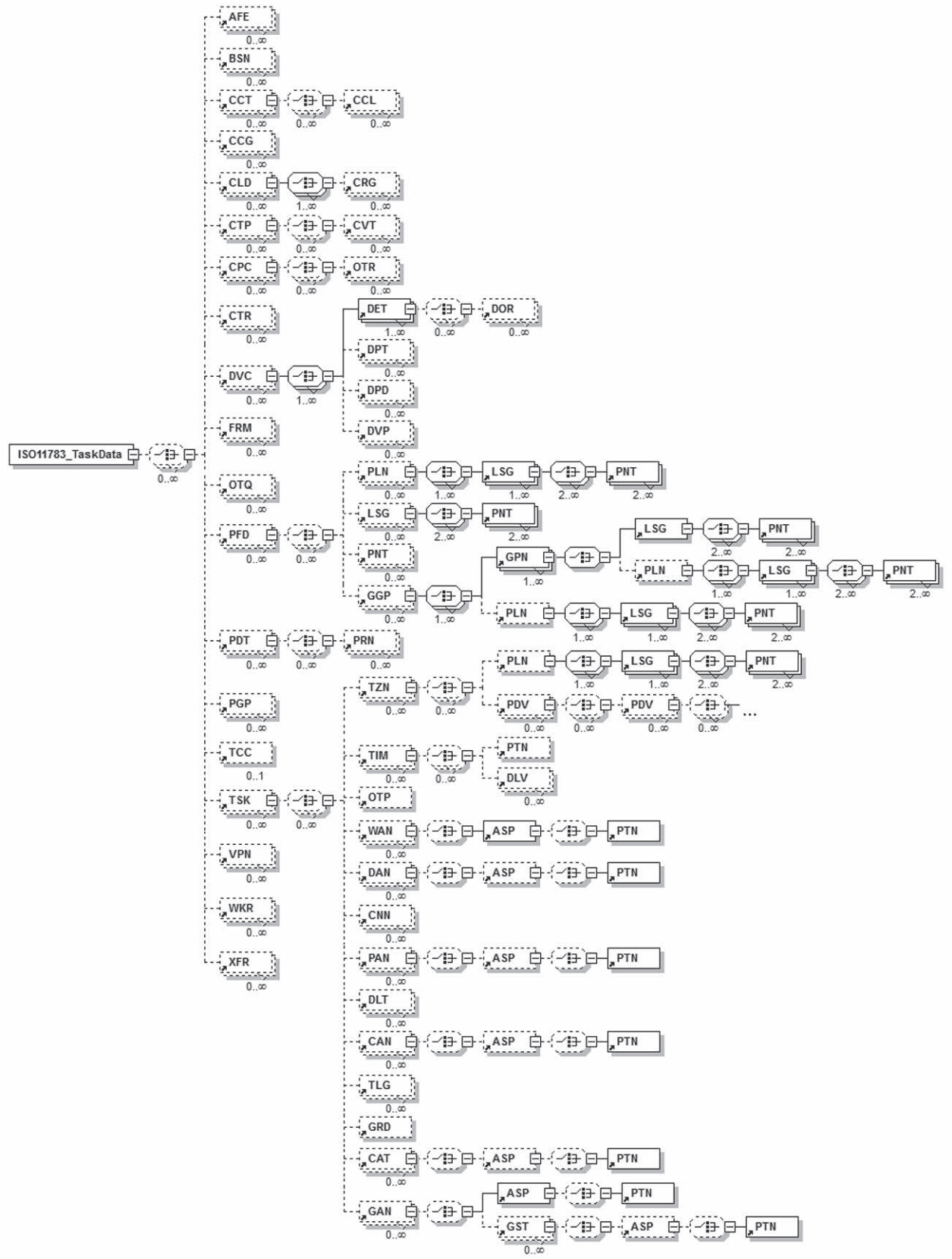


Figure 8 — Schematic structure of the data transfer file set

Figure 9 shows the usage of referenced XML files.

© ISO 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

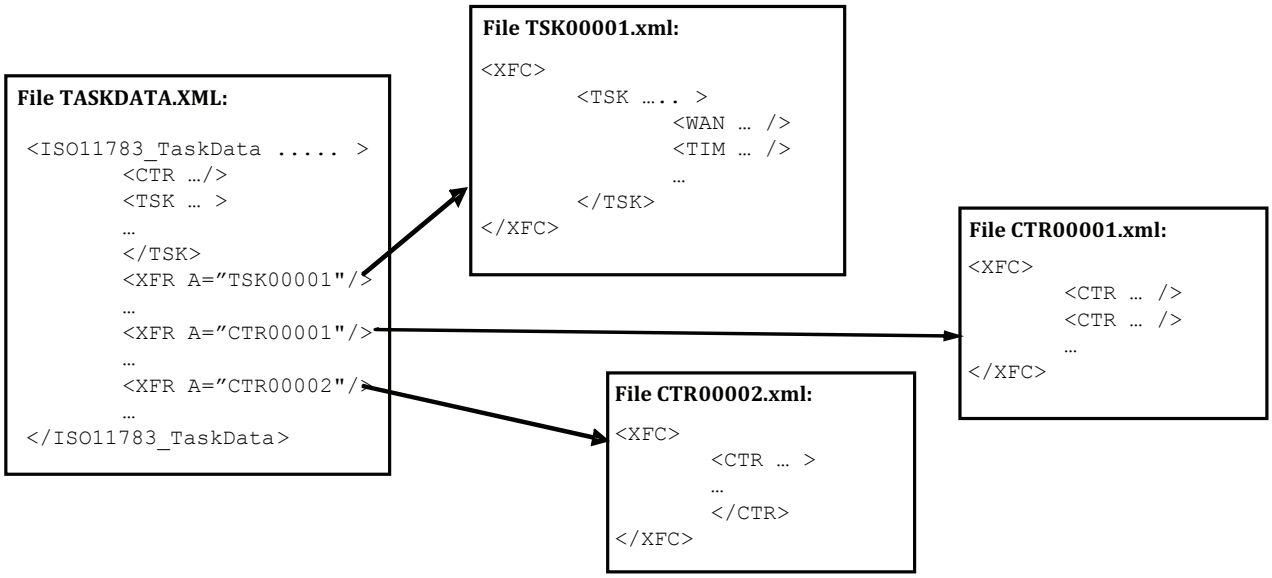


Figure 9 — XFR/XFC XML file references

The direction indication attribute in the XML header section (FMIS→MICS or MICS→FMIS) is to be handled and updated properly by both FMIS and MICS.

For an update of coding data, the MICS is allowed to add new instances of XML elements only. This means that modification of FMIS coding data can only be done by the FMIS.

The TC or TC configuration program shall be prepared to receive either a single XML data transfer file or set of XML data transfer files from the management information system. The management information system shall be prepared to receive either a single data transfer file or a set of XML data transfer files from the MICS, and both can refer to binary files.

8.6 Binary data transfer files

8.6.1 General

There are three types of binary file allowed to be used as part of the data transfer file set. One is to contain gridcell values. The second is to contain binary-coded log data. Both binary files belong to a task. The third binary file, introduced in ISO 11783-10 version 4, is used for the binary encoding of the Point geometry elements.

8.6.2 Grid binary file structure

The gridcells of a Grid can only be defined in a binary file. Two types of grid are supported: the first type of grid contains TreatmentZone codes; the second type of grid contains ProcessDataVariable values.

The first type is used when a limited number of TreatmentZones is defined and the grid functions as a lookup table to the TreatmentZones. In this case, each gridcell in the binary grid file contains a TreatmentZoneCode of the TreatmentZone to which that gridcell belongs. This grid type has at a maximum one value per gridcell, which is an unsigned 8-bit integer coded TreatmentZoneCode.

EXAMPLE 1 Grid that contains TreatmentZoneCodes:

The task data XML file contains a grid definition with an empty TreatmentZoneIdRef attribute:

< GRD A = "58.096653" B = "8.54321" C = "1.5E-3" D = "1.4E-3" E = "200" F = "300" G = "GRD00001" I = "1"/ >

The binary file contains: (TreatmentZoneCodes)

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

(1)(4)(3)(6)…

Each value inside brackets is a record and represents the binary-coded treatment zone code. The brackets are for better reading purposes in this example and are not part of the binary format. The relation between the binary file and the grid type 1 specification is shown in Figure 10.

File TASKDATA.XML:

```
<ISO11783_TaskData ..... >
     <CTR … />
     <TSK … >
          …
          <GRD A="58.096653" B="8.54321" C="1.5E-3" D="1.4E-3" E="200" F="300"
G="GRD00001" I="1"/>
          <TZN A="1" B="MidRate 1" C="2">
               <PDV A="0006" B="10000"/>
          </TZN>
          <TZN A="2" B="MidRate 2" C="3">
               <PDV A="0006" B="12000"/>
          </TZN>
          <TZN A="3" B="HighRate 3" C="4">
               <PDV A="0006" B="14000"/>
          </TZN>
          <TZN A="4" B="HighRate 4" C="5">
               <PDV A="0006" B="15000"/>
          </TZN>
          <TZN A="5" B="MaxRate 5" C="6">
               <PDV A="0006" B="17000"/>
          </TZN>
          <TZN A="6" B="MaxRate 6" C="7">
               <PDV A="0006" B="19000"/>
          </TZN>
          …
     </TSK>
</ISO11783_TaskData>
```

File GRD00001.bin:

Contents of the binary coded GridCell file GRD00001.bin:

(1)(4)(5)(1)……

(Each record represents a reference to the treatment zone for the grid cells in order of the first grid cell in the grid. So, grid cell 1 would reference treatment zone 1, grid cell 2 would reference 4, grid cell 3 would reference 5 and so on.)

Figure 10 — Grid type 1 binary file example

The second type of grid is used when the ProcessDataValue attribute of the ProcessDataVariable is not classified into a limited number of TreatmentZones. In this case, one TreatmentZone, possibly with multiple ProcessDataVariables, is defined as a template for the binary grid file. There are no values of the XML attributes "ProcessDataValue" in this template definition in the XML file specified and, instead, these values are coded in the binary grid file as signed 32-bit integers. For byte ordering of the 32-bit integer data type, the same rules apply as defined in ISO 11783-6.

EXAMPLE 2 Grid that contains ProcessDataVariable ProcessDataValues

The task data XML file contains a Grid definition with a TreatmentZoneCode attribute:

< GRD A = "58.096653" B = "8.54321" C = "1.5E-3" D = "1.4E-3" E = "200" F = "300" G = "GRD00001" I = "2" J = "6"/ >

and a TreatmentZone prototype without ProcessDataValue attributes:

< TZN A = "6" B = "Precision Farming" C = "2" >

© ISO 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

< PDV A = ”0001” B = ”0”/ >

< PDV A = ”0006” B = ”0”/ >

< PDV A = ”001A” B = ”0”/ >

< /TZN >

The binary file contains: (ProcessDataValue records [])

[(11100)(15000)(190)]

[(14000)(20000)(200)]

[(19000)(25000)(200)]……

where the first record represents the first gridcell with the following ProcessDataValues:

< PDV A = ”0001” B = ”11100”/ >

< PDV A = ”0006” B = ”15000”/ >

< PDV A = ”001A” B = ”190”/ >

Each value inside the square brackets in the binary file example is a record and represents the ProcessDataValues for one gridcell. The rounded brackets delimit the individual ProcessDataValue values when more than one ProcessDataVariable is present in the referenced TreatmentZone. The brackets are for better reading purposes in this example and are not part of the binary format. Figure 11 shows the binary file related to the grid type 2 specification.

File TASKDATA.XML:

```
<ISO11783_TaskData ..... >
     <CTR … />
     <TSK … >
          …
          <GRD A="58.096653" B="8.54321" C="1.5E-3" D="1.4E-3" E="200" F="300"
G="GRD00001" I=”2" J=”1"/>
          <TZN A=”1" B=”Precision Application”>
               <PDV A=”0001" B=”0"/>
               <PDV A=”0006" B=”0"/>
               <PDV A=”001A" B=”0"/>
          </TZN>
          …
     </TSK>
</ISO11783_TaskData>
```

File GRD00001.bin:

Contents of the binary coded GridCell file GRD00001.bin:

[(11100)(15000)(190)]
[(14000)(20000)(200)]
[(19000)(25000)(200)]

(Each record represents the values of the ProcessDataVariables in the grid referenced treatment zone. The first record contains the first grid cell in the
grid. So, the grid references treatment zone 1, grid cell 1 contains 3 values for the 3 processdatavariables, grid cell 2 contains 3 values for the 3 ProcessDataVariables and so on.)

Figure 11 — Grid type 2 binary file example

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

8.6.3 Log data binary file structure

The TimeLog XML element is used in the task data file to refer to a binary log file. In the task data file, a TimeLog XML element without referenced Time, Position, or DataLogValues shall be used. This TimeLog XML element refers to two files: a TimeLog XML header file and a binary log file. Inside the TimeLog XML header file, the XML elements shall include attributes without any values to define the record structure of the binary log file. All attributes with non-empty value definitions in the TimeLog XML header file contain fixed values which are valid for all binary-coded records in the binary log file. Only the values of the empty value attributes of the TimeLog XML header file are written in the binary log file. The empty value attributes of the Time, Position, and DataLogValue XML elements may be written in any order. The order of the corresponding values in the binary log file shall be according to the order listed in Table 3. An example of an XML-coded TimeLog file that specifies the structure of the binary TimeLog file is given by:

< TIM A = ”” D = ”4” >

< !– PositionNorth, PositionEast and PositionStatus are the required attributes of element Position. – >

< !– In this sample only these three position attributes are written to the binary TimeLog file. – >

< PTN A = ”” B = ”” D = ””/ >

< !– DLV 0 – >

< DLV A = ”0815” B = ”” C = ”DET1”/ >

< !– DLV 1 – >

< DLV A = ”4711” B = ”” C = ”DET2”/ >

< !– DLV 2 – >

< DLV A = ”4522” B = ”” C = ”DET3”/ >

< /TIM >

TimeLog binary data formats are shown in Table 3. Only the attributes that are listed in Table 3 can be used in the binary log file. The attributes from the Time, Position, and DataLogValue XML elements that are not listed in Table 3 shall not be used in a binary log file. The attributes shall be written in the order as listed in Table 3. Note that the local time zone information introduced in ISO 11783-10 version 4 in the Time XML element does not change the definition of the use of the TimeStart attribute in the binary logfile. The TimeStart attribute in the binary logfile shall continue to be expressed in local time.

Table 3 — Binary file record value definitions

Value	XML reference	Binary data type	Not Available value	Definition
TimeStart: time of day	TIM, A	Unsigned 32-bit integer	$FFFFFFFF_{16}$	Local time zone milliseconds since midnight
TimeStart: date	TIM, A	Unsigned 16-bit integer	$FFFF_{16}$	Local time zone days since 1980–01–01
PositionNorth	PTN, A	32-bit integer	$FFFFFFFF_{16}$	10^{-7} degrees WGS-84
PositionEast	PTN, B	32-bit integer	$FFFFFFFF_{16}$	10^{-7} degrees WGS-84
PositionUp	PTN, C	32-bit integer	$FFFFFFFF_{16}$	Millimetres relative to WGS-84 ellipsoid

© ISO 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

Table 3 *(continued)*

Value	XML reference	Binary data type	Not Available value	Definition
PositionStatus	PTN, D	Byte	FF_{16}	Position status. Definition references NMEA2000 MethodGNSS parameter. 0 = no GPS fix 1 = GNSS fix 2 = DGNSS fix 3 = Precise GNSS, no deliberate degradation (such as SA), and higher resolution code (P-code) and 2 frequencies are used to correct atmospheric delays 4 = RTK Fixed Integer 5 = RTK Float 6 = Est(DR)mode 7 = Manual input 8 = Simulate mode 9–13 = Reserved 14 = Error 15 = PositionStatus value not available
PDOP	PTN, E	Unsigned 16-bit integer	$FFFF_{16}$	10^{-1} PDOP quality information
HDOP	PTN, F	Unsigned 16-bit integer	$FFFF_{16}$	10^{-1} HDOP quality information
NumberOfSatellites	PTN, G	Byte	FF_{16}	Number of used satellites
GpsUtcTime	PTN, H	Unsigned 32-bit integer	$FFFFFFFF_{16}$	UTC milliseconds since midnight
GpsUtcDate	PTN, I	Unsigned 16-bit integer	$FFFF_{16}$	UTC days since 1980–01–01
#DLV		Byte	n.a.	Number of PDV to follow
DLVn		Byte	n.a.	Ordering number of PDV to follow, starting with 0 for first DataLogValue definition
ProcessDataValue	DLV, B	32-bit integer	n.a.	According to DDI

All attributes in the header file that contain values are specified as having these constant values for all records of the binary file. All DLV elements are indexed and referred to in the binary records according to their definitions order. The number of DataLogValues actually stored is defined as #DLV. An example of the binary log data file is shown in Figure 12, in which there are a total of three DLVs. To allow a dynamic set of DLVs in the binary record, each DLV is identified by its ordering number, DLVn, of definition in the XML file. The example of an XML-coded TimeLog file given above, just before Table 3, specifies that all following records consist of the following set of values:

(TimeStart,PositionNorth,PositionEast,PositionStatus,#DLV,DLV0,PDV0,DLV1,PDV1,DLV2,PDV2)

This means that a time entry can have a maximum of 255 PDVs.

© ISO 2015 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015

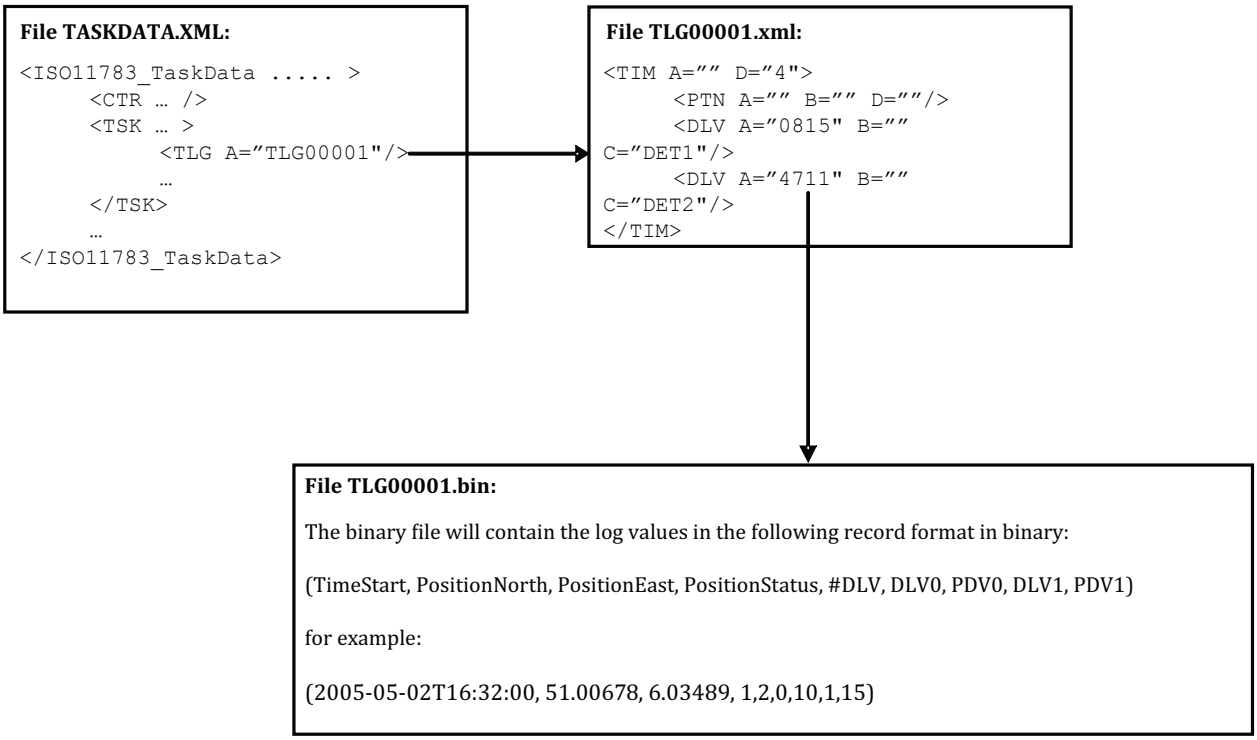


Figure 12 — Log data binary file example

8.6.4 Point data binary file structure

A series of Point XML elements can be transferred in a binary file as an alternative encoding to its XML definition. This data transfer method provides the possibility to use a higher range of accuracy. Within the binary format, the Position North and East are specified as 64-bit-integers. In the task data file, the Point XML element may refer to a binary file. In case of a reference to a binary file, the Point XML elements shall include attributes without any values to define the record structure of the binary file. All attributes with non-empty value definitions in the Point XML element definition contain fixed values which are valid for all binary-coded records in its referenced binary file. Only the values of the empty value attributes of the Point XML element are written in the binary file. The empty value attributes may be defined in the Point XML element in any order. The order in which the values for these attributes are written to or read from the binary file shall be according to the order of their listing in Table 4. An example of an XML-coded Point element that specifies the structure of the binary Point file is given by

```
< LSG A = "6" E = "1" B = "Line1" D = "2000" C = "20" >
   < PNT A = "2" C = "" D = "" E = "" F = "1" J = "PNT00001"/ >
< /LSG >
```

The attributes of the Point XML element that can be transferred via a binary file are specified in Table 4:

Table 4 — Binary Point file record value definitions

Value	XML reference	Binary Data Type	Not Available value	Definition
PointType	PNT, A	Unsigned 8-bit integer	FF_{16}	Enumeration of PointTypes
PositionNorth	PNT, C	64-bit integer	$FFFFFFFFFFFFFFFF_{16}$	10^{-16} degrees WGS-84
PositionEast	PNT, D	64-bit integer	$FFFFFFFFFFFFFFFF_{16}$	10^{-16} degrees WGS-84

© ISO 2015 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 11783-10:2015