



MECHATROLINK-4

Specifications for MECHATROLINK Network Information(MNI) file



MECHATROLINK Members Association

MANUAL NO. MMA TDEP 047A

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, mechanical, electronic, photocopying, recording, or otherwise, without the prior written permission of MMA*. No patent liability is assumed with respect to the use of the information contained herein. Moreover, because MMA is constantly striving to improve its high-quality products, the information contained in this manual is subject to change without notice. Every precaution has been taken in the preparation of this manual. Nevertheless, MMA assumes no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of the information contained in this publication.

* MMA: MECHATROLINK Members Association

[Organization of the Manuals]

■ Scope of this Manual

This manual describes the specifications for the MECHATROLINK network information (MNI = MECHATROLINK Network Information) file that is used in the network configuration of the MECHATROLINK-4 high-speed field network.

For related documents, see "Organization of the Manuals" below.

■ Organization of the Manuals

MECHATROLINK-4 related manuals are organized as follows.

1) For general users

(1) MECHATROLINK-4 Installation Manual

Describes the cable routing for MECHATROLINK-4 devices and the procedures for connecting and installing peripheral devices.

2) For manufacturers developing devices

(2) MECHATROLINK-4 Protocol Manual

Describes the physical layer, data link layer and application layer of MECHATROLINK-4.

Detailed instruction manuals for each profile supplementing the Protocol Manual

- MECHATROLINK-4 Command Manual for Standard Servo Profile
- MECHATROLINK-4 Command Manual for Standard I/O Profile
- MECHATROLINK-4 Command Manual for Standard Stepping Motor Drive Profile
- MECHATROLINK-4 Command Manual for Standard Inverter Profile
- MECHATROLINK-4 Command Manual for Event-driven Communication ID Information Acquisition Profile
- MECHATROLINK-III Command Manual for Standard Servo Profile

(3) MECHATROLINK-4 Command Manual for Message Communication

Describes the communication commands in MECHATROLINK-4 message communication.

(4) MECHATROLINK-4 Device Information (MDI: MECHATROLINK Device Information) File Manual

Describes the MECHATROLINK-4 device information files.

(5) MECHATROLINK-4 Network Information (MNI: MECHATROLINK Network Information) File Manual

Describes the file specifications required for MECHATROLINK-4 network information.

(This manual)

Contents

1 Overview	5
2 Purpose of Adopting MNI	7
3 Usage Methods	8
3.1 Creation	8
3.2 Usage	9
3.3 Confirming and Updating Content	10
3.4 Version Control Rules	11
3.4.1 MNI Schema Version Control Rules	11
3.4.2 MNI Version Control Rules	12
4 Format	13
4.1 Data	13
4.2 Schema	13
5 Data Types	14
6 MNI Data Structure	15
6.1 Network	16
6.1.1 Topology	17
6.1.2 Parameter	20
6.2 DomainList	22
6.2.1 Member	23
6.2.2 Parameter	24
6.3 NodeList	33
6.3.1 MDeviceList	34
6.3.2 HubList	45
APPENDIX A Notation	47
A.1 Element Structure	47
A.2 Paths	48
A.3 Support Requirements	49
Revision History	50

1 Overview

MECHATROLINK-4 uses the MECHATROLINK device information (MDI = MECHATROLINK Device Information)^(*)file, MECHATROLINK network information (MNI = MECHATROLINK Network Information) file, and network configuration tool to provide a quick and easy configuration environment with high portability. An overall view of the typical configuration flow for MECHATROLINK-4 is shown in Fig 1-1.

The MNI file is created and exported using the network configuration tool in this configuration flow, and it contains all the information of network setting for the MECHATROLINK-4 system.

The purpose, usage methods, format, and data structure of the MNI file are described in later chapters.

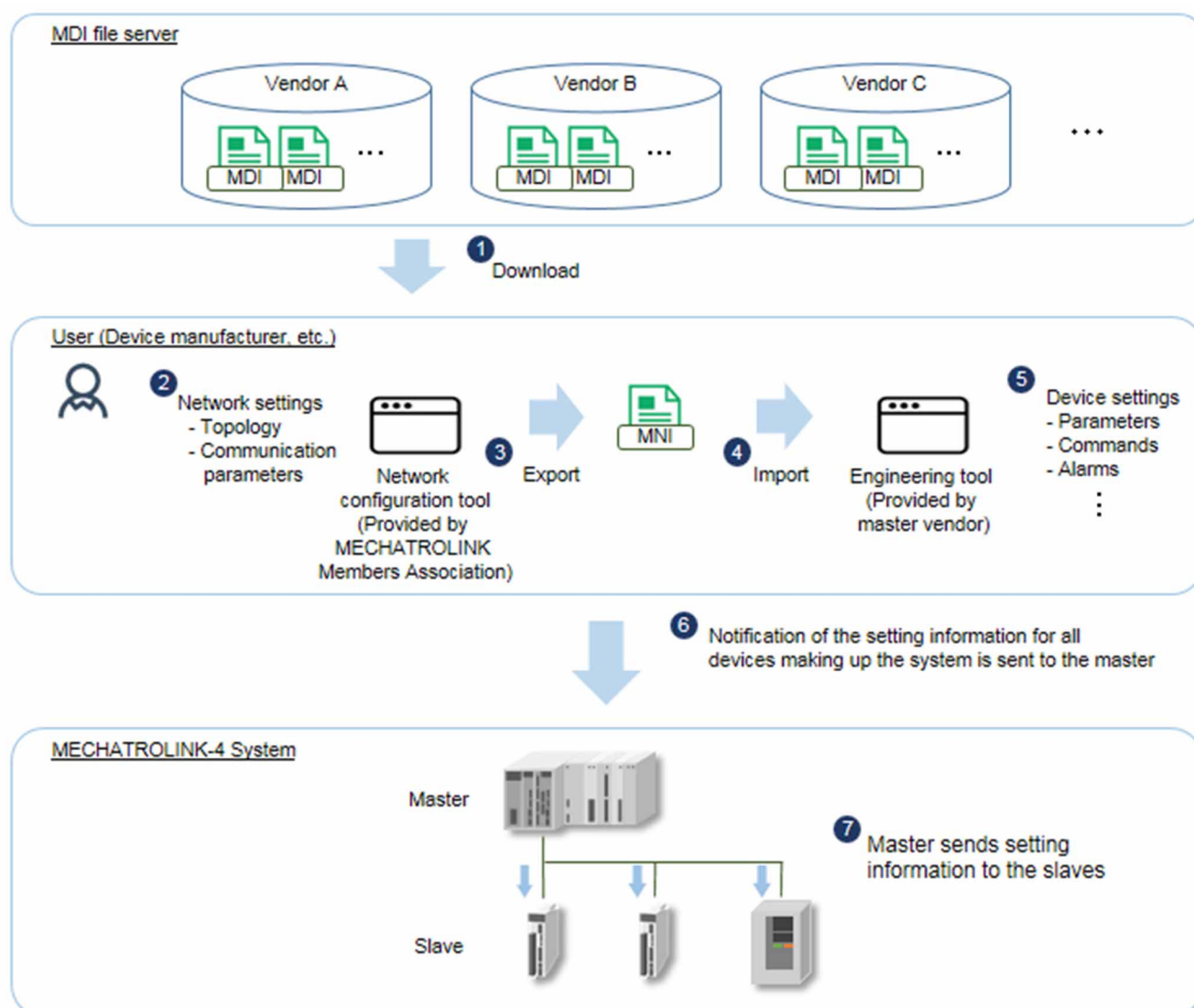


Fig 1-1 Overview of MDI/MNI File Usage

- *1. The MDI (MECHATROLINK Device Information) file contains the MECHATROLINK device specifications information. Because MDI definitions are used to standardize device specifications information, this enables simplification of specifications information input to various tools for quick configuration.

2 Purpose of Adopting MNI

The MNI file and the network configuration tool provide the following advantages.

(1) Quick and easy configuration

The MNI file, network configuration tool, and engineering tool can be used to standardize network configuration procedures and the network information formats between tools for enabling quicker and easier configuration.

(2) Improved portability

The MNI enables easy portability of system network information, such as when replacing the master in an existing system with a new master.

3 Usage Methods

3.1 Creation

An overview of MNI creation is provided below.

- (1) The MDI files are obtained from the vendors of the devices making up the MECHATROLINK-4 system.
- (2) The obtained MDI files are imported to the configuration tool.
- (3) The user makes the MECHATROLINK-4 communication settings based on the specifications information of the devices specified in the imported MDI files.
- (4) The system network information and MNI files are exported.

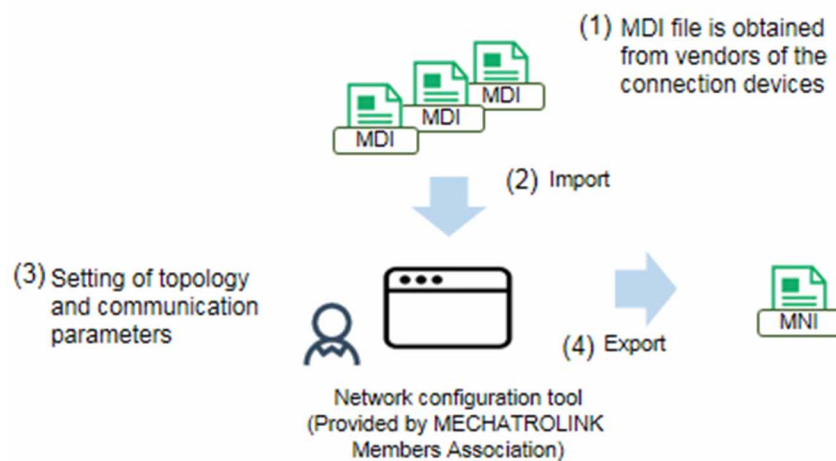


Fig 3-1 Creation of MNI File

3.2 Usage

An overview of MNI usage is provided below.

- (1) The MNI files are imported to the engineering tool of the master.
- (2) The engineering tool is used to set the parameters for each device, and all setting information is sent to the master.
- (3) The master uses the setting information received from the engineering tool to initialize MECHATROLINK-4 communication.

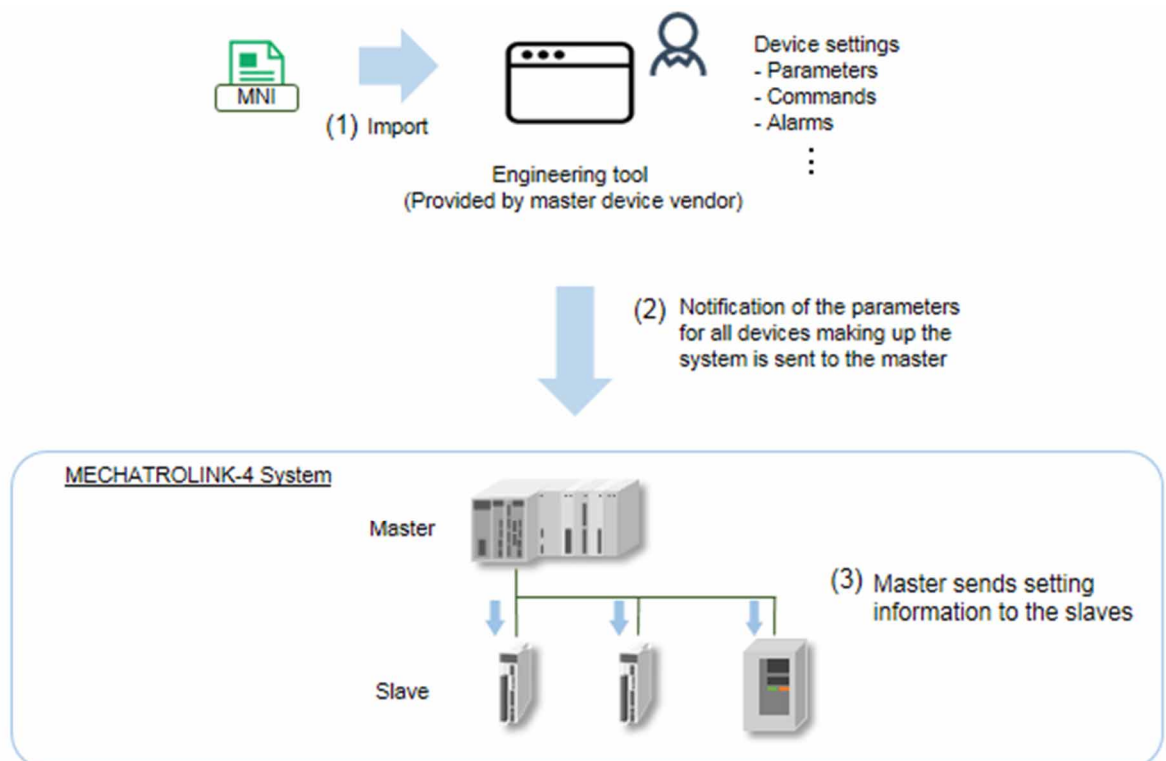


Fig 3-2 Usage of MNI File

3.3 Confirming and Updating Content

An overview of the procedure for confirming and updating MNI content is provided below.

- (1) The MNI file is imported to the configuration tool.
- (2) The network configuration tool is used to confirm and update the setting information content.
- (3) If necessary, the MNI file is exported.

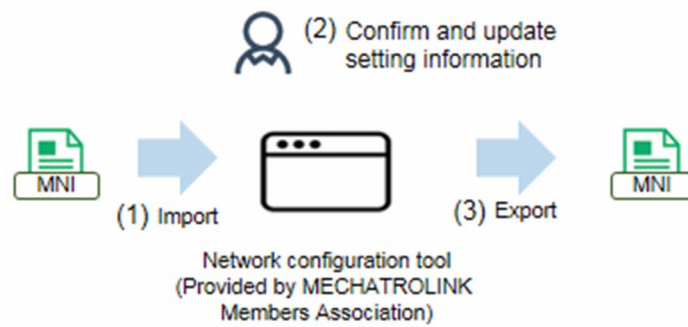


Fig 3-3 Checking and Updating MNI File Content

3.4 Version Control Rules

3.4.1 MNI Schema Version Control Rules

The MNI schema version is controlled by the MECHATROLINK Members Association.

The MMA increases the MNI schema version according to the following table when a change occurs in the MNI schema.

Details								
Description: The MNI schema version has the specifications shown below.								
Bit No.	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
Version Information	Minor version							
Bit No.	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
Version Information	Major version							
bit16 to bit31 : Reserved(0)								
Major version: When significant changes are made to the MNI schema, such as those that accompany a change in the MNI data structure.								
Minor version: When changes are made to the MNI schema, such as those that accompany an addition or change in minor elements and attributes.								

3.4.2 MNI Version Control Rules

The MNI version is controlled by the user.

The user increases the MNI version according to the following table when a change occurs in the information contained in the MNI.

Details								
Description: The MNI version has the specifications shown below.								
Bit No.	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
Version Information	Minor version							
Bit No.	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
Version Information	Major version							
bit16 to bit31 : Reserved(0)								
Major version: Usage method is established by the user.								
Minor version: Usage method is established by the user.								

4 Format

4.1 Data

In the following explanations, the file containing the MNI that was set by the user (MNI file) will be referred to as the "MNI data file" below. The format of the MNI data file is based on the XML specifications (<https://www.w3.org/TR/xml/>) defined by the W3C (World Wide Web Consortium). For the file extension, ".xml" is added to the file to indicate that the file is an XML file.

4.2 Schema

The file where the MNI data structure is defined will be called the "MNI schema file" below. The format of the MNI schema file is based on the XML schema specifications (<https://www.w3.org/XML/Schema>) defined by the W3C (World Wide Web Consortium). The MNI schema file uses mni.xsd, which is issued by the MECHATROLINK Members Association (Table 4-1). This schema file can be used to verify the validity of the MNI data file.

Table 4-1 Schema

Schema	Issued by	Explanation
mni.xsd	MECHATROLINK Members Association	This defines the MNI data structure. This file is used by the configuration tool to verify the validity of the MNI file that was set by the user.

5 Data Types

The XML data types used in the MNI data structure are shown in Table 5-1. The information content is based on the XML data types defined by the W3C (<https://www.w3.org/TR/xmlschema11-2/>).

Table 5-1 XML Data Type

Data Type	Explanation	Example
xs: boolean	This is a logical value.	True, False, 1, 0
xs: hexBinary	This is binary data that has been coded in hexadecimal format. It is not case-sensitive.	0A6B, 0a6b
xs: unsignedLong	This is a 64-bit unsigned interger (0 to 8,446,744,073,709,551,615).	0, 522240000000
xs: unsignedInt	This is a 32-bit unsigned integer (0 to 4,294,967,295).	0, 1267884
xs: unsignedShort	This is a 16-bit unsigned integer (0 to 65,535).	0, 45238
xs: unsignedByte	This is an 8-bit unsigned integer (0 to 255).	0, 126
xs: date	This is the date (YYYY-MM-DD).	1999-08-07
xs: string	This is a character string.	This network has two nodes.
xs: ID	This is an identifier, and it must be unique within the data file.	node001, domain02
xs: IDREF	This is a reference to an element that has an ID attribute matching the defined ID.	—

6 MNI Data Structure

This chapter describes the data structure and elements and attributes for MNI files. The structure of the Mni element, which is the root node of this data structure, is shown in Fig 6-1. For details on the notation used in this chapter, see APPENDIX A.

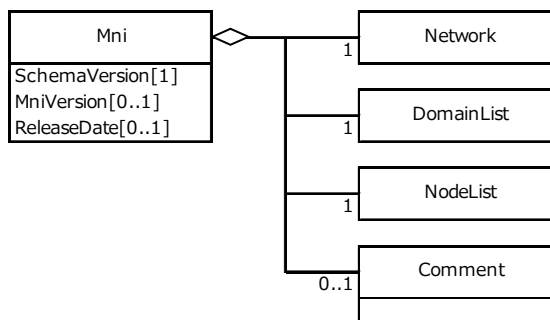


Fig 6-1 Structure of Mni Element

Details on the component elements of the Mni element are shown in Table 6-1.

Table 6-1 Component Elements of Mni Element

Element	Data Type	Support	Explanation
Network	—	M	This is common information within the network. This specifies the topology and common parameters for all nodes. (See 6.1)
DomainList	—	M	This is information specific to all control domains within the network. (See 6.2)
NodeList	—	M	This is information specific to all nodes within the network. This is information specific to each MECHATROLINK device and each hub. (See 6.3)
Comment	xs: string	O	This contains comment information if needed.

Details on the MNI element and its component element attributes are shown in Table 6-2.

Table 6-2 MNI Element and Component Element Attributes

Attribute	Data Type	Support	Explanation
@SchemaVersion	xs: hexBinary	M	This is the version of the supported schema.
@MniVersion	xs: hexBinary	O	This is the version of the MNI file.
@ReleaseDate	xs: date	O	This is the date that the MDI file was released.

6.1 Network

The Network element holds common network information. The structure of the Network element is shown in Fig 6-2.

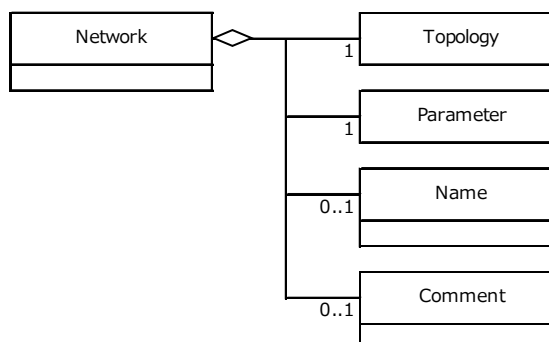


Fig 6-2 Structure of Network Element

Details on the component elements of the Network element are shown in Table 6-3.

Table 6-3 Component Elements of Network Element

Element	Data Type	Support	Explanation
Topology	—	M	This is the topology information of the network. This specifies the connection relationship for all nodes in the network, cable length, and other details. (See 6.1.1)
Parameter	—	M	This is common parameter information for all nodes within the network. This specifies the assignment of manager functions and various bandwidth information. (See 6.1.2)
Name	xs: string	O	This is the network name.
Comment	xs: string	O	This contains comment information if needed.

6.1.1 Topology

The Topology element holds the topology information of the network. The structure of the Topology element is shown in Fig 6-3.

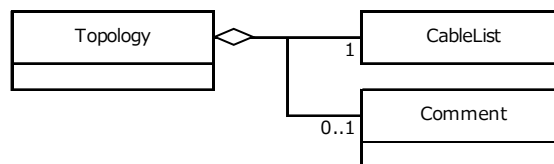


Fig 6-3 Structure of Topology Element

Details on the component elements of the Topology element are shown in Table 6-4.

Table 6-4 Component Elements of Topology Element

Element	Data Type	Support	Explanation
CableList	—	M	This is length information for all cables within the network. (See 6.1.1.1)
Comment	xs: string	O	This contains comment information if needed.

6.1.1.1 CableList

The CableList element holds the cable information within the topology information of the network. The structure of the CableList element is shown in Fig 6-4.

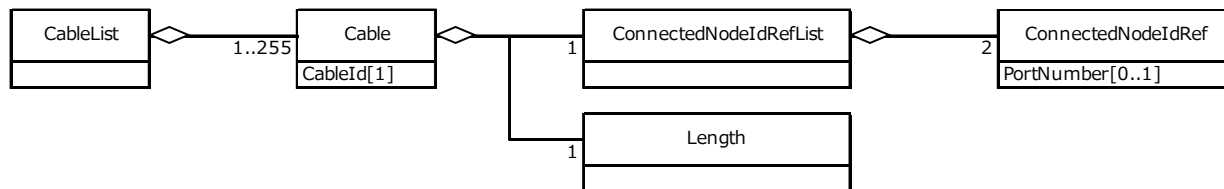


Fig 6-4 Structure of CableList Element

Details on the component elements of the CableList element are shown in Table 6-5.

Table 6-5 Component Elements of CableList Element

Element	Data Type	Support	Explanation
Cable	-	M	This is the cable information.
Cable/ConnectedNodeIdRefList	-	M	This is the information for the node where the cable is connected.
Cable/ConnectedNodeIdRefList/ConnectedNodeIdRef	xs: IDREF	M	This is the identifier for the node where the cable is connected. Setting range: Value specified from the elements below "Mni/NodeList/MDeviceList/MDevice/@NodeId" or "Mni/NodeList/HubList/Hub/@NodeId"
Length	xs: unsignedInt	M	This is the cable length. Setting range: 0 to 100,000 (mm)

Details on the CableList element and its component element attributes are shown in Table 6-6.

Table 6-6 CableList Element and Component Element Attributes

Attribute	Data Type	Support	Explanation
Cable/@CableId	xs: ID	M	This is the cable identifier. This must be set uniquely for each cable in the network. Setting range: cable000 to cable253
Cable/ConnectedNodeId RefList/ConnectedNodeId Ref/@PortNumber	xs: unsignedByte	C	This is a port number where the cable is connected. This is used when the node where the cable is connected is a MECHATROLINK device. This element is used only when various tools are used to set the values obtained from devices online. Setting range: 0 to 15

6.1.2 Parameter

The Parameter element holds common network information. This information is not dependent on the control domain or node. The structure of the Parameter element is shown in Fig 6-5.

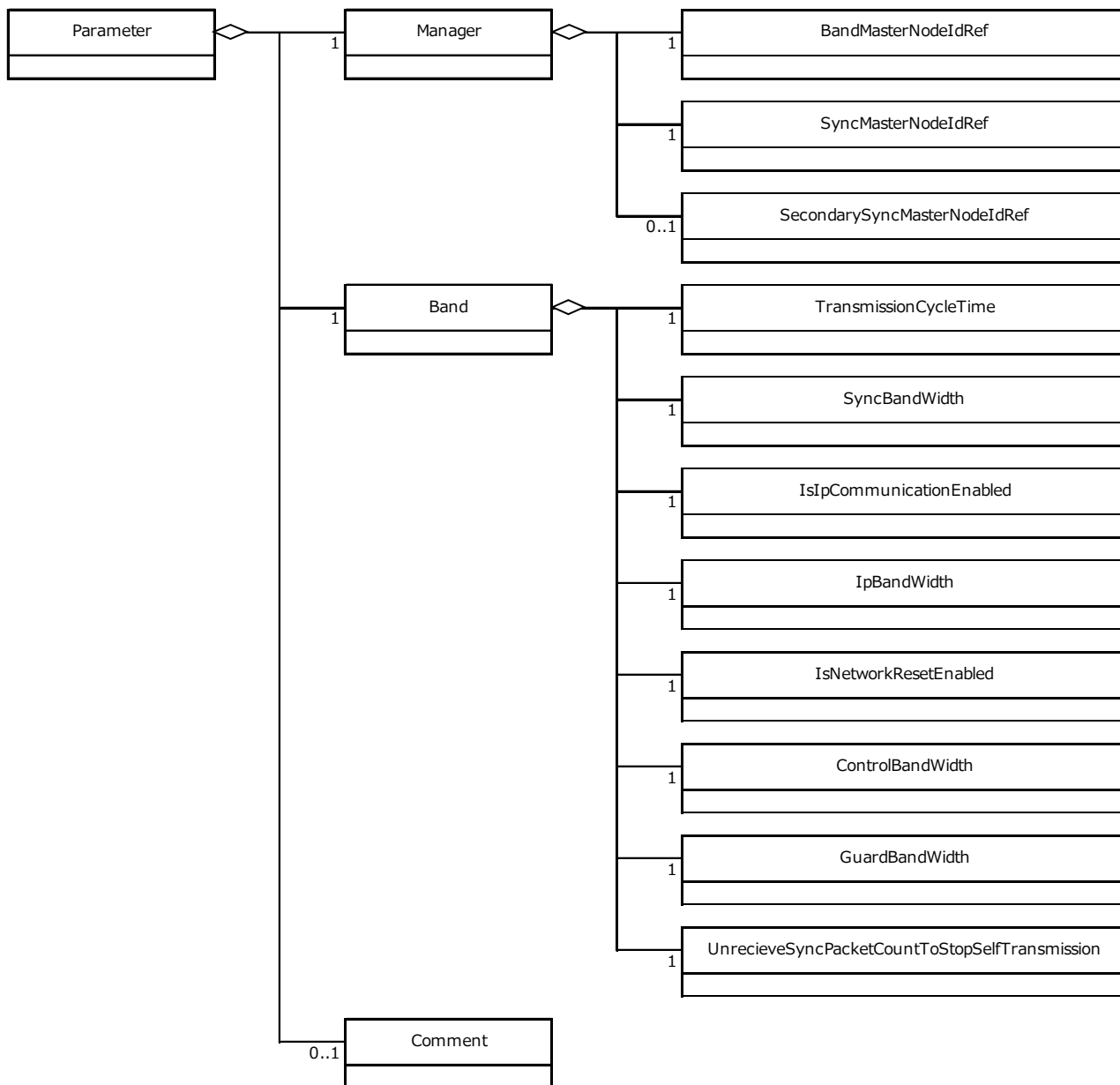


Fig 6-5 Structure of Parameter Element

Details on the component elements of the Parameter element are shown in Table 6-7.

Table 6-7 Component Elements of Parameter Element

Element	Data Type	Support	Explanation
Manager	—	M	This is the manager information of the network.
Manager/ BandMasterNodeIdRef	xs: IDREF	M	This is a node identifier of the band master. Setting range: Value specified from the elements below "Mni/NodeList/MDeviceList/MDevice/@NodeId"
Manager/SyncMasterNodeId Ref	xs: IDREF	M	This is a node identifier of the sync master. Setting range: Value specified from the elements below "Mni/NodeList/MDeviceList/MDevice/@NodeId"
Manager/SecondarySync MasterNodeIdRef	xs: IDREF	C	This is a node identifier of the secondary sync master. This element is used only when the sync master switching function is used. Setting range: Value specified from the elements below "Mni/NodeList/MDeviceList/MDevice/@NodeId"
Band	—	M	This is bandwidth information of the network.
Band/TransmissionCycleTime	xs: unsignedInt	M	This is the MECHATROLINK transmission cycle. Setting range: 15,625 to 64,000,000 (ns)
Band/SyncBandWidth	xs: unsignedInt	M	This is the bandwidth where a sync master broadcasts sync notification packets at the start of the transmission cycle. Setting range: 0 to 4,000,000 (ns)
Band/ IsIpCommunicationEnabled	xs: boolean	M	This is the IP communication usage flag. Setting range: 0: Not used 1: Usage
Band/IpBandWidth	xs: unsignedInt	M	This is the IP communication bandwidth. Setting range: 0 to 63,999,999 (ns)
Band/IsNetworkResetEnabled	xs: boolean	M	This is the network reset usage flag. Setting range: 0: Not used 1: Usage
Band/ControlBandWidth	xs: unsignedInt	M	This is the control bandwidth. Setting range: 0 to 63,999,999 (ns)
Band/GuardBandWidth	xs: unsignedShort	M	This is the guard bandwidth. Setting range: 0 to 10,000 (ns)
Band/UnrecieveSyncPacket CountToStopSelfTransmission	xs: unsignedByte	M	This is the maximum number of consecutive failures for receiving sync packets until the local node stops transmission. Setting range: 0 – 255
Comment	xs: string	O	This contains comment information if needed.

6.2 DomainList

The DomainList element holds the control domain-dependent information on the network. The structure of the DomainList element is shown in Fig 6-6.

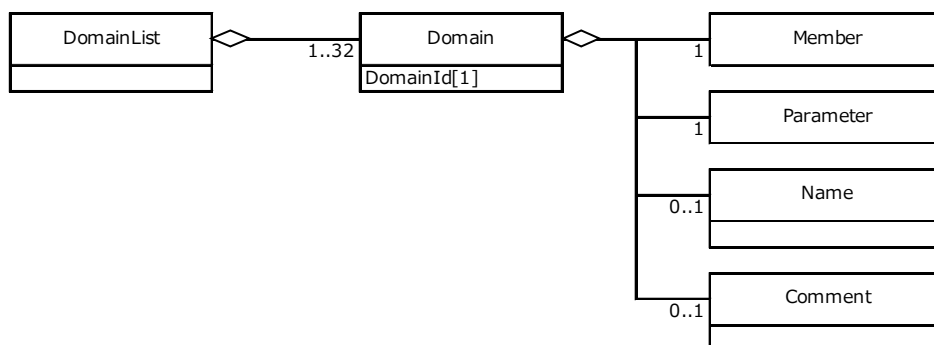


Fig 6-6 Structure of DomainList Element

Details on the component elements of the DomainList element are shown in Table 6-8.

Table 6-8 Component Elements of DomainList Element

Element	Data Type	Support	Explanation
Domain	—	M	This is control domain information.
Domain/Member	—	M	This is member node information for the control domain. (See 6.2.1)
Domain/Parameter	—	M	This is parameter information dependent on the control domain. (See 6.2.2)
Name	xs: string	O	This is the domain name.
Comment	xs: string	O	This contains comment information if needed.

Details on the DomainList element and its component element attributes are shown in Table 6-9.

Table 6-9 DomainList Element and Component Element Attributes

Attribute	Data Type	Support	Explanation
Domain/@DomainId	xs: ID	M	This is the identifier of the control domain. This must be set uniquely in the network. Setting range: domain00 to domain31

6.2.1 Member

The Member element holds node identifier information for the masters and slaves of the control domain within the control domain-dependent information. The structure of the Member element is shown in Fig 6-7.

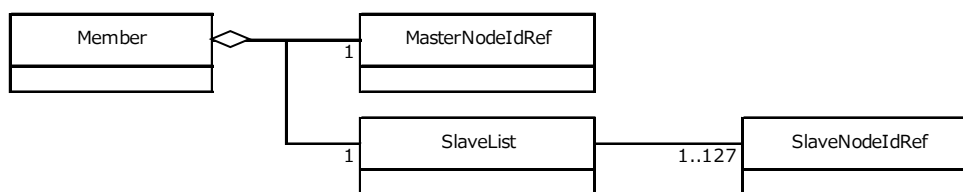


Fig 6-7 Structure of Member Element

Details on the component elements of the Member element are shown in Table 6-10.

Table 6-10 Component Elements of Member Element

Element	Data Type	Support	Explanation
MasterNodeIdRef	xs: IDREF	M	This is the node identifier of the control domain master. Setting range: Value specified from the elements below "Mni/NodeList/MDeviceList/MDevice/@NodeId"
SlaveList	—	M	This is information on all slaves of the control domain.
SlaveList/SlaveNodeIdRef	xs: IDREF	M	This is the node identifier for the slave. Setting range: Value specified from the elements below "Mni/NodeList/MDeviceList/MDevice/@NodeId"

6.2.2 Parameter

The Parameter element holds parameter information for the control domain within the control domain-dependent information. The structure of the Parameter element is shown in Fig 6-8.

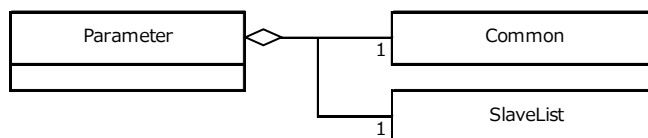


Fig 6-8 Structure of Parameter Element

Details on the component elements of the Parameter element are shown in Table 6-11.

Table 6-11 Component Elements of Parameter Element

Element	Data Type	Support	Explanation
Common	—	M	This is common information for all nodes of the control domain. (See 6.2.2.1)
SlaveList	—	M	This is information dependent on each slave of the control domain. (See 6.2.2.2)

6.2.2.1 Common

The Common element holds common parameter information for the control domain. This information is dependent on the control domain and is common with the nodes in the control domain. The structure of the Common element is shown in Fig 6-9.

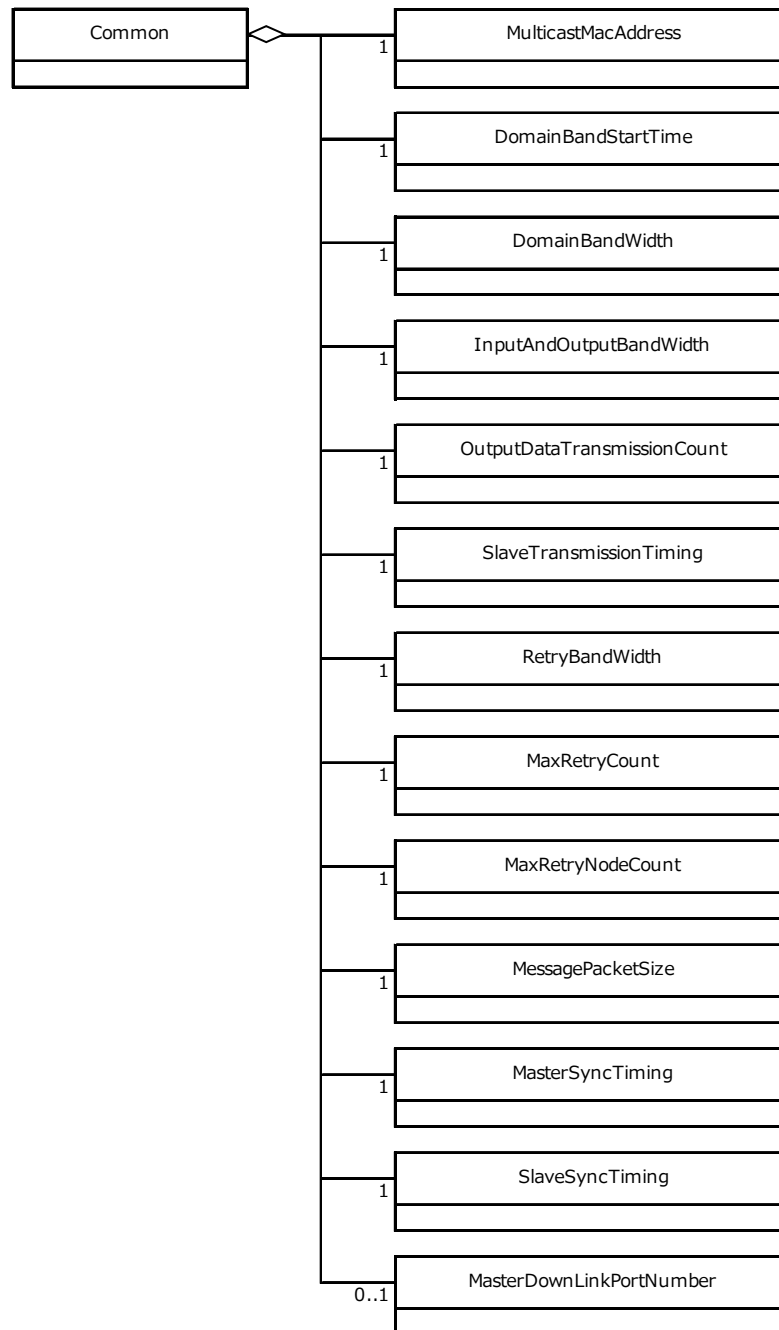


Fig 6-9 Structure of Common Element

Details on the component elements of the Common element are shown in Table 6-12.

Table 6-12 Component Elements of Common Element

Element	Data Type	Support	Explanation
MulticastMacAddress	xs: hexBinary	M	This is the multicast MAC address for simultaneous reporting to all nodes within the control domain. This must be set uniquely in the network. Setting range: 01:20:B5:00:04:20 to 01:20:B5:00:04:3F
DomainBandStartTime	xs: unsignedInt	M	This is the start time of the control domain transmission band. This is set as the offset time from the transmission cycle start. Setting range: 0 to 63,999,999 (ns)
DomainBandWidth	xs: unsignedInt	M	This is the transmission bandwidth of the control domain. Setting range: 0 to 63,999,999 (ns)
InputAndOutputBandWidth	xs: unsignedInt	M	This is the input/output data transmission bandwidth. Setting range: 0 to 63,999,999 (ns)
OutputDataTransmissionCount	xs: unsignedByte	M	This is the count for sending output data in the control domain. Setting range: 1 to 4
SlaveTransmissionTiming	xs: unsignedInt	M	This is the time that the slave starts transmission of input data within the input/output data transmission band. This is set as the offset time from the control domain start time. Setting range: 0 to 63,999,999 (ns)
RetryBandWidth	xs: unsignedInt	M	This is the input data retransmission bandwidth. Setting range: 0 to 63,999,999 (ns)
MaxRetryCount	xs: unsignedByte	M	This is the maximum value for the number of retry requests in the input data retry band. Setting range: 0 to 127
MaxRetryNodeCount	xs: unsignedByte	M	This is the maximum value for the number of nodes that is the target for a single retry request. Setting range: 0 to 127
MessagePacketSize	xs: unsignedShort	M	This is the data length for packets during message communication. Message data that exceeds the message packet length is sent by dividing into parts. This parameter is used only when message communication is used. Setting range: 0 to 1,490 (bytes)

Element	Data Type	Support	Explanation
MasterSyncTiming	xs: unsignedInt	M	This is the synchronous interrupt occurrence time for the master. This is set by specifying the offset time from the control domain start time. Setting range: 0 to 63,999,999 (ns)
SlaveSyncTiming	xs: unsignedInt	M	This is the synchronous interrupt occurrence time for the slave. This is set by specifying the offset time from the control domain start time. Setting range: 0 to 63,999,999 (ns)
MasterDownLinkPortNumber	xs: unsignedShort	C	This is the downstream port setting of the master. Downstream ports refer to the ports connected to a route where there is a slave to be managed in the control domain. One port is assigned to 1 bit, and the correspondence relation is as follows. This element is used only when various tools are used to set the values obtained from devices online.

bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
port	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Setting range:
0: Disabled
1: Enabled

6.2.2.2 SlaveList

The SlaveList element holds parameter information dependent on the slave within the parameters dependent on the control domain. The structure of the SlaveList element is shown in Fig 6-10.

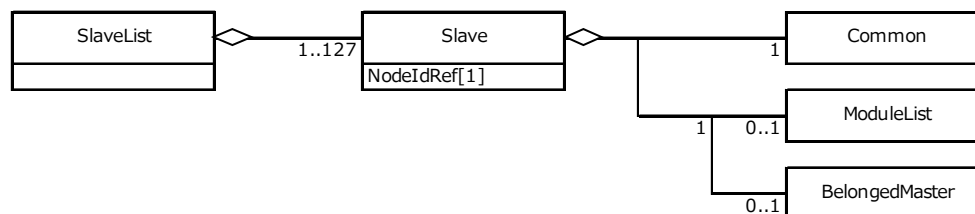


Fig 6-10 Structure of SlaveList Element

Details on the component elements of the SlaveList element are shown in Table 6-13.

Table 6-13 Component Elements of SlaveList Element

Element	Data Type	Support	Explanation
Slave	—	M	This holds parameter information dependent on each slave.
Slave/Common	—	M	This holds parameter information common to the modules. (See 6.2.2.2.1)
Slave/ModuleList	—	C	This holds parameter information dependent on the module. (See 6.2.2.2.2)
Slave/BelongedMaster	—	C	This holds parameter information dependent on the master that belongs to the control domain. This element is used only when the device node type is master. (See 6.2.2.2.3)

Details on the SlaveList element and its component element attributes are shown in Table 6-14.

Table 6-14 SlaveList Element and Component Element Attributes

Attribute	Data Type	Support	Explanation
Slave/@NodeIdRef	xs: IDREF	M	This is the node identifier of the slave that is referenced. Setting range: Value specified from the elements below "Mni/DomainList/Domain/Member/SlaveList/SlaveNodeIdRef"

6.2.2.2.1 Common

The Common element holds parameter information common to the modules within the parameter information dependent on the slave. The structure of the Common element is shown in Fig 6-11.

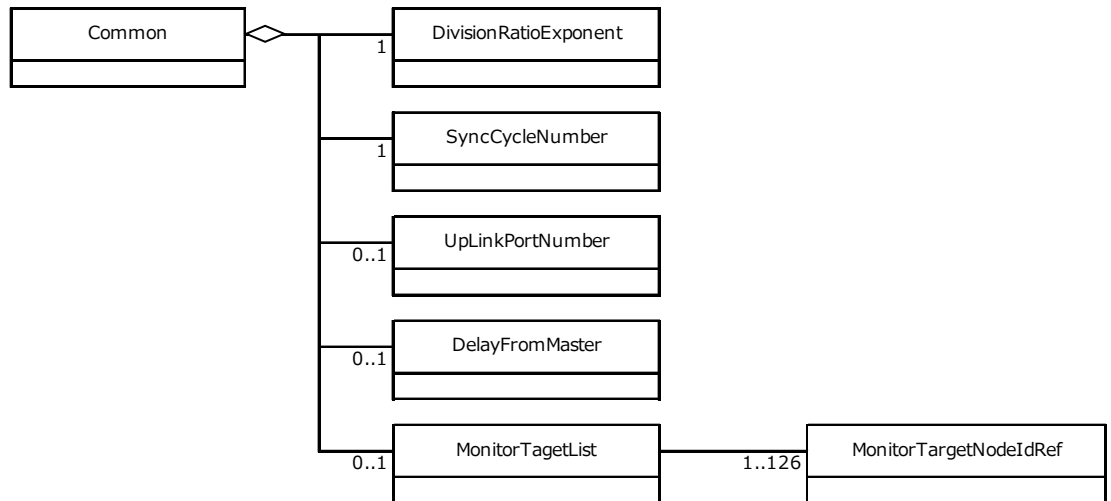


Fig 6-11 Structure of Common Element

Details on the component elements of the Common element are shown in Table 6-15.

Table 6-15 Component Elements of Common Element

Element	Data Type	Support	Explanation
DivisionRatioExponent	xs: unsignedByte	M	This is the exponent of the division ratio. The division ratio is expressed as 2^n (where n is an exponent). Setting range: 0 to 5
SyncCycleNumber	xs: unsignedByte	M	This is the transmission cycle number where the control data is changed. Setting range: 0 to (division ratio - 1)
UpLinkPortNumber	xs: unsignedByte	C	This is the upstream port number. This element is used only when various tools are used to set the values obtained from devices online. Setting range: 0 to 15
DelayFromMaster	xs: unsignedInt	C	This is the transmission delay time from the control domain master. This element is used only when various tools are used to set the values obtained from devices online. Setting range: 0 to 4,000,000 (ns)
MonitorTargetList	-	C	This is the monitor target node information. This element is used only when monitoring other slaves within the same control domain.
MonitorTargetList/ MonitorTargetNodeIdRef	xs: IDREF	M	This is the node identifier of the slave that is the monitor target. Setting range: Value specified from the elements below "Mni/DomainList/Domain/Member/SlaveList/SlaveNodeIdRef"

6.2.2.2.2 ModuleList

The ModuleList element holds parameter information dependent on the module within the parameter information dependent on the slave. The structure of the ModuleList element is shown in Fig 6-12.

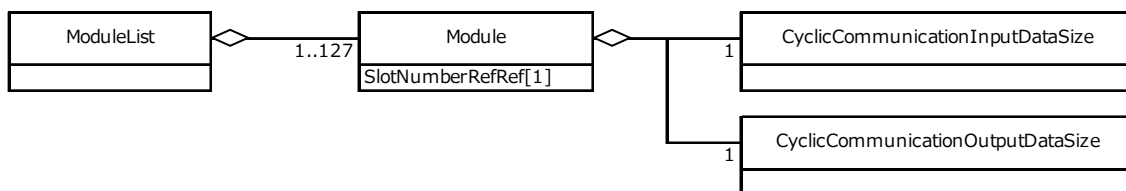


Fig 6-12 Structure of ModuleList Element

Details on the component elements of the ModuleList element are shown in Table 6-16.

Table 6-16 Component Elements of ModuleList Element

Element	Data Type	Support	Explanation
Module	—	M	This holds parameter information dependent on the respective module. Only modules used in the control domain are defined in this element.
Module/ CyclicCommunicationInput DataSize	xs: unsignedShort	M	This is the input data size of the module in cyclic communication. Setting range: 0 to 1,492 (bytes)
Module/ CyclicCommunication OutputDataSize	xs: unsignedShort	M	This is the output data size of the module in cyclic communication. Setting range: 0 to 1,492 (bytes)

Details on the ModuleList element and its component element attributes are shown in Table 6-17.

Table 6-17 ModuleList Element Attributes

Attribute	Data Type	Support	Explanation
Module/@SlotNumberRefRef	xs: unsignedByte	M	This is the slot number where the module is installed. Setting range: Value specified from the elements below "Mni/NodeList/MDeviceList/SlotList/Slot/@SlotNumberRef"

6.2.2.2.3 BelongedMaster

The BelongedMaster element holds parameter information dependent on the master that belongs to the control domain within the parameter information dependent on the slave. The structure of the BelongedMaster element is shown in Fig 6-13.

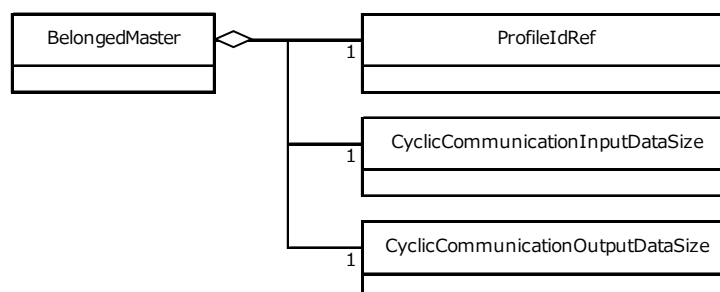


Fig 6-13 Structure of BelongedMaster Element

Details on the component elements of the BelongedMaster element are shown in Table 6-18.

Table 6-18 Component Elements of BelongedMaster Element

Element	Data Type	Support	Explanation
ProfileIdRef	xs: string	M	This is the identifier for the profile used by the master that belongs to the control domain. Various tools use this information to reference the profile information in the MDI. This information specifies the profile identifier contained in the MDI.
CyclicCommunicationInputDataSize	xs: unsignedShort	M	This is the input data size of the master that belongs to the control domain in cyclic communication. Setting range: 0 to 8,192 (bytes)
CyclicCommunicationOutputDataSize	xs: unsignedShort	M	This is the output data size of the master that belongs to the control domain in cyclic communication. Setting range: 0 to 8,192 (bytes)

6.3 NodeList

The NodeList element holds specific information for all nodes. This information is not dependent on the control domain and corresponds uniquely to each node. The structure of the NodeList element is shown in Fig 6-14.

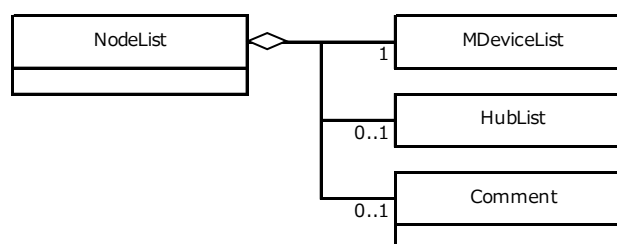


Fig 6-14 Structure of NodeList Element

Details on the component elements of the NodeList element are shown in Table 6-19.

Table 6-19 Component Elements of NodeList Element

Element	Data Type	Support	Explanation
MDeviceList	—	M	This holds node-specific information for all MECHATROLINK devices. (See 6.3.1)
HubList	—	C	This holds node-specific information for all hubs. This element is available only for topologies with hubs. (See 6.3.2)
Comment	xs: string	O	This contains comment information if needed.

6.3.1 MDeviceList

The MDeviceList element holds information specific to all MECHATROLINK devices. This information is not dependent on the control domain and corresponds uniquely to each node. The structure of the MDeviceList element is shown in Fig 6-15.

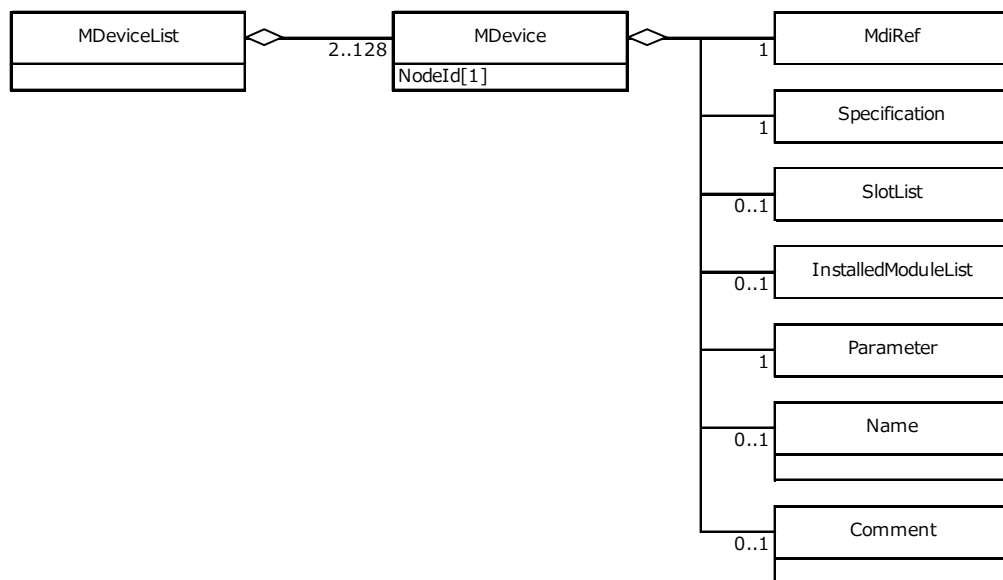


Fig 6-15 Structure of MDeviceList Element

Details on the component elements of the MDeviceList element are shown in Table 6-20.

Table 6-20 Component Elements of MDeviceList Element

Element	Data Type	Support	Explanation
MDevice	—	M	This holds the node-dependent information.
MDevice/MdiRef	—	M	This holds the reference information for the MDI. (See 6.3.1.1)
MDevice/Specification	—	M	This is the node specifications information. (See 6.3.1.2)
MDevice/SlotList	—	C	This holds the information for all slots belonging to a node. This element is used only when the device node type is slave. (See 6.3.1.3)
MDevice/InstalledModuleList	—	C	This is the information for all modules installed in a slot. This element is used only when the device node type is slave. (See 6.3.1.4)
MDevice/Parameter	—	M	This is the node parameter information. (See 6.3.1.5)
MDevice/Name	xs: string	O	This is the node name.
MDevice/Comment	xs: string	O	This contains comment information if needed.

Details on the MDeviceList element and its component element attributes are shown in Table 6-21.

Table 6-21 MDeviceList Element and Component Element Attributes

Attribute	Data Type	Support	Explanation
MDevice/@NodeId	xs: ID	M	This is node identifier of the MECHATROLINK device. Setting range: mDevice000 to mDevice127

6.3.1.1 MdiRef

The MdiRef element holds the MDI reference information within the node-dependent information. The structure of the MdiRef element is shown in Fig 6-16.

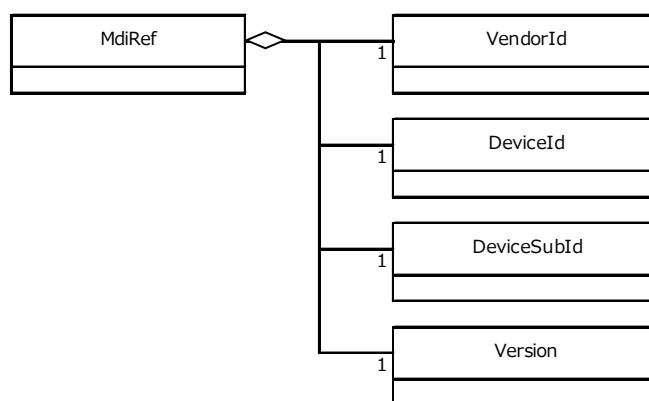


Fig 6-16 Structure of MdiRef Element

Details on the component elements of the MdiRef element are shown in Table 6-22.

Table 6-22 Component Elements of MdiRef Element

Element	Data Type	Support	Explanation
VendorId	xs: hexBinary	M	This is the vendor ID code registered with the MECHATROLINK Members Association. This information holds the same value as the vendor ID code contained in the MDI.
DeviceId	xs: hexBinary	M	This is the device code. This information holds the same value as the device code contained in the MDI.
DeviceSubId	xs: hexBinary	M	This is the device subcode. This information holds the same value as the device subcode contained in the MDI.
Version	xs: hexBinary	M	This is the version of the MDI file. This information holds the same value as the MDI file version contained in the MDI.

6.3.1.2 Specification

The Specification element holds the node specifications information within the node-dependent information. The structure of the Specification element is shown in Fig 6-17.

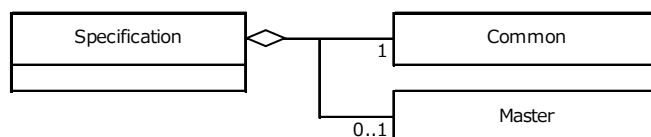


Fig 6-17 Structure of Specification Element

Details on the component elements of the Specification element are shown in Table 6-23.

Table 6-23 Component Elements of Specification Element

Element	Data Type	Support	Explanation
Common	—	M	This is specifications information that is common between the master and slave. The value obtained from the MDI is used in this information. (See 6.3.1.2.1)
Master	—	C	This is specifications information when functioning as a master. The value obtained from the MDI is used in this information. This element is used only when the device node type is master. (See 6.3.1.2.2)

6.3.1.2.1 Common

The Common element holds information common to the master and slave within the node specifications information. The structure of the Common element is shown in Fig 6-18.

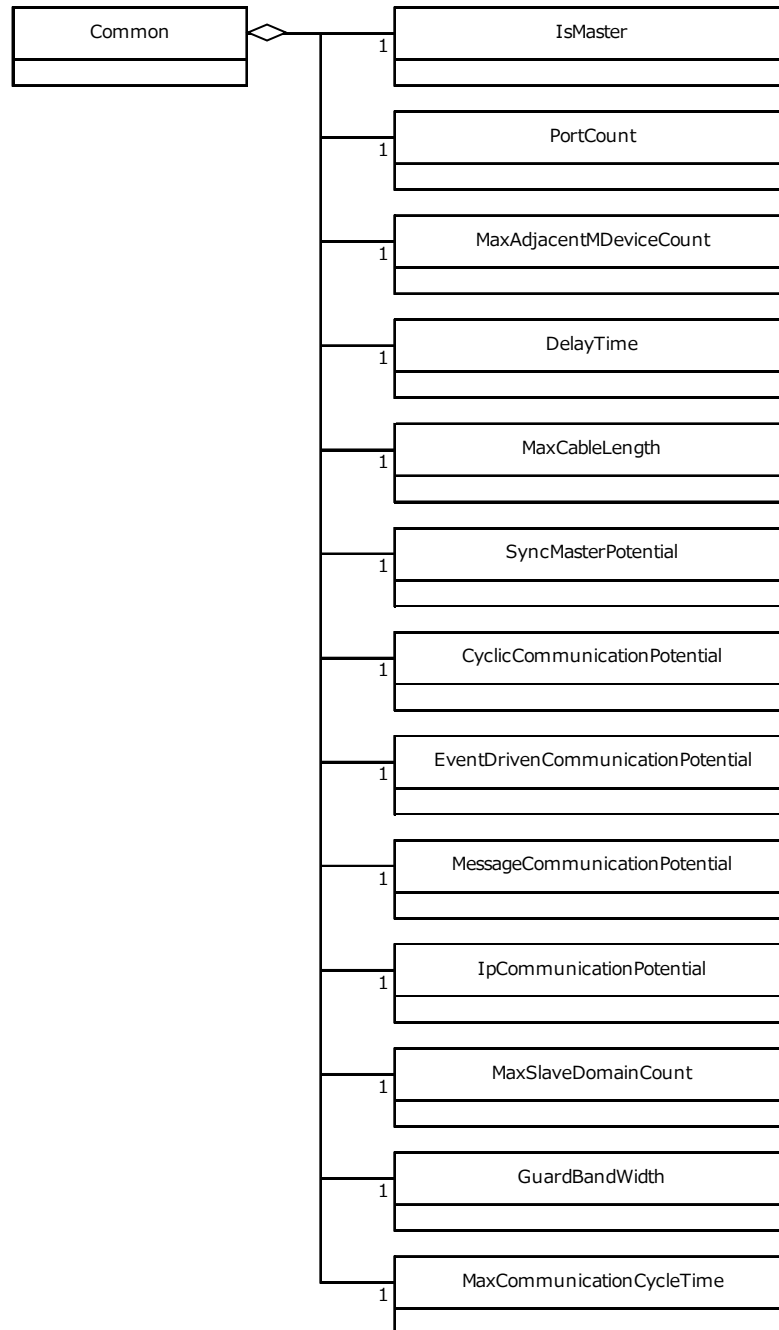


Fig 6-18 Structure of Common Element

Details on the component elements of the Common element are shown in Table 6-24.

Table 6-24 Component Elements of Common Element

Element	Data Type	Support	Explanation
IsMaster	xs: boolean	M	This is the type of the node. Setting range: 0: Slave 1: Master
PortCount	xs: unsignedByte	M	This is the number of ports that a node has. Setting range: 1 to 16
MaxAdjacentMDeviceCount	xs: unsignedByte	M	This is the maximum number of adjacent MECHATROLINK devices. This indicates the maximum number of MECHATROLINK devices that can be adjacent in this device. This also includes MECHATROLINK devices connected through a hub. Setting range: 1 to 127
DelayTime	xs: unsignedInt	M	This is the relay delay time for a node. This indicates the delay time required when relaying packets received from an adjacent node. Setting range: 0 to 4,000,000 (ns)
MaxCableLength	xs: unsignedInt	M	This is the maximum value for the cable length supported by a node. Setting range: 0 to 100,000 (mm)
SyncMasterPotential	xs: boolean	M	This indicates whether the device functions as a sync master. Setting range: 0: No 1: Yes
CyclicCommunicationPotential	xs: boolean	M	This indicates whether cyclic communication is supported. Setting range: 0: Not supported 1: Supported
EventDrivenCommunication Potential	xs: boolean	M	This indicates whether event-driven communication is supported. Setting range: 0: Not supported 1: Supported
MessageCommunication Potential	xs: boolean	M	This indicates whether message communication is supported. Setting range: 0: Not supported 1: Supported
IpCommunicationPotential	xs: boolean	M	This indicates whether IP transmission is supported. Setting range: 0: Not supported 1: Supported

Element	Data Type	Support	Explanation
MaxSlaveDomainCount	xs: unsignedByte	M	This is the maximum value for the number of domains that a node can belong to. Setting range: 0 to 4
GuardBandWidth	xs: unsignedShort	M	This is the guard bandwidth. Setting range: 0 to 10,000 (ns)
MaxCommunicationCycleTime	xs: unsignedLong	M	This is the maximum value of the communications cycle. Setting range: 15,625 to 522,240,000,000 (ns)

6.3.1.2.2 Master

The Master element holds specifications information of the master within the node specifications information. The structure of the Master element is shown in Fig 6-19.

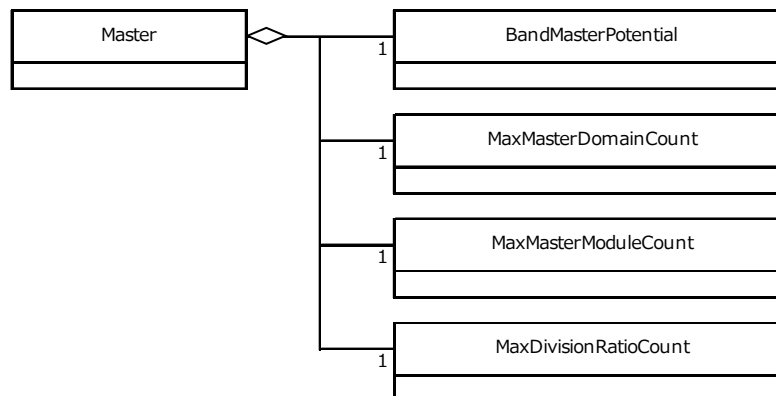


Fig 6-19 Structure of Master Element

Details on the component elements of the Master element are shown in Table 6-25.

Table 6-25 Component Elements of Master Element

Element	Data Type	Support	Explanation
BandMasterPotential	xs: unsignedByte	M	This is information when functioning as a band master. Setting range: 0: Both band master and non-band master are supported 1: Band master only is supported 2: Non-band master only is supported
MaxMasterDomain Count	xs: unsignedByte	M	This is the maximum number of domains that a master can manage. Setting range: 1 to 4
MaxMasterModuleCount	xs: unsignedByte	M	This is the maximum number of modules that a master can manage. The number of slaves that a master can manage is this value or less. Setting range: 1 to 127
MaxDivisionRatioCount	xs: unsignedByte	M	This is the maximum number of division ratios that a master can manage. Setting range: 1 to 6

6.3.1.3 SlotList

The SlotList element holds the information for all slots belonging to a device within the node-dependent information. The structure of the SlotList element is shown in Fig 6-20.

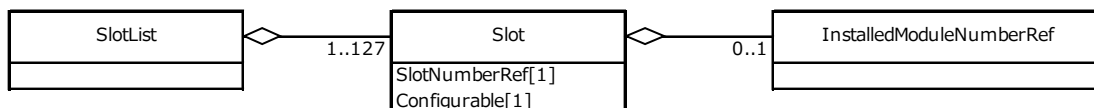


Fig 6-20 Structure of SlotList Element

Details on the component elements of the SlotList element are shown in Table 6-26.

Table 6-26 Component Elements of SlotList Element

Element	Data Type	Support	Explanation
Slot	—	M	This is the information for each slot belonging to a device.
Slot/InstalledModuleNumberRef	xs: unsignedByte	C	This is the number of the module installed in the slot. This element is used when installing a module to a slot. Setting range: Value specified from the elements below "Mni/NodeList/MDeviceList/MDevice/InstalledModuleList/InstalledModule/@InstalledModuleNumber"

Details on the SlotList element and its component element attributes are shown in Table 6-27.

Table 6-27 SlotList Element and Component Element Attributes

Attribute	Data Type	Support	Explanation
Slot/@SlotNumberRef	xs: unsignedByte	M	This is the slot number. This information specifies the slot number contained in the MDI. Various tools use this information to reference the slot information in the MDI.
Slot/@Configurable	xs: boolean	M	This information indicates whether the module can be installed and removed to and from the slot. This information holds the same value as the module install/remove allowed information contained in the MDI.

6.3.1.4 InstalledModuleList

The InstalledModuleList element holds information for all modules installed in a device slot within the node-dependent information. The structure of the InstalledModuleList element is shown in Fig 6-21.

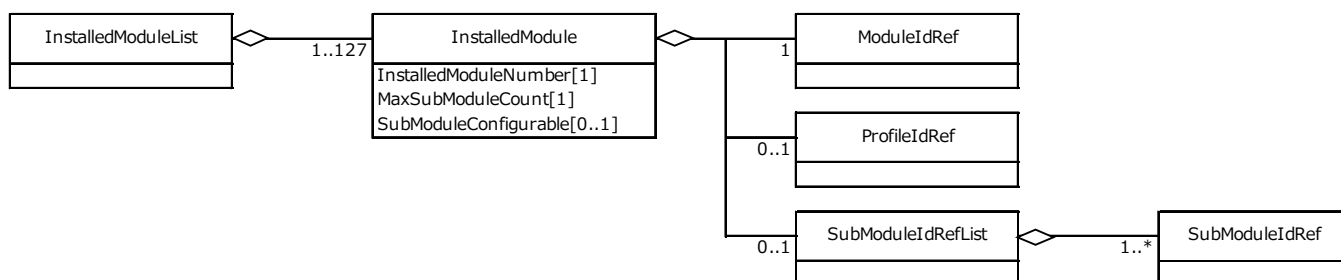


Fig 6-21 Structure of InstalledModuleList Element

Details on the component elements of the InstalledModuleList element are shown in Table 6-28.

Table 6-28 Component Elements of InstalledModuleList Element

Element	Data Type	Support	Explanation
InstalledModule	—	M	This is information on the module installed in the slot.
InstalledModule/ ModuleIdRef	xs: string	M	This is the module identifier. This information specifies the module identifier contained in the MDI. Various tools use this information to reference the module information in the MDI.
InstalledModule/ ProfileIdRef	xs: string	C	This is identifier for the profiles used by the module. Various tools use this information to reference the profile information in the MDI. This information specifies the profile identifier contained in the MDI. This element is used only when this module is used within the system.
InstalledModule/ SubModuleIdRefList	—	C	This is the identifier information for all sub-modules installed in a module. This information specifies the sub-module identifier contained in the MDI. This element is used only when a sub-module is installed.
InstalledModule/ InstalledModuleIdRef/ SubModuleIdRef	xs: string	M	This is the identifier of the sub-module installed in a module. This information specifies the sub-module identifier contained in the MDI. Various tools use this information to reference the sub-module information in the MDI.

Details on the InstalledModuleList element and component element attributes are shown in Table 6-29.

Table 6-29 InstalledModuleList Element and Component Element Attributes

Attribute	Data Type	Support	Explanation
InstalledModule/@InstalledModuleNumber	xs: unsignedByte	M	This is the module number. This information is a unique value within the device. Various tools use this information as a key for referencing information of the module installed in the slot. Setting range: 0 to 127
InstalledModule/@MaxSubModuleCount	xs: unsignedByte	M	This is maximum value for the number of sub-modules supported by the module. This information holds the same value as the maximum number of sub-modules contained in the MDI.
InstalledModule/@SubModuleConfigurable	xs: boolean	C	This information indicates whether sub-module installation/removal is allowed. Setting range: 0: Installation/removal not allowed 1: Installation/removal allowed This element is used only when there are 1 or more of the following attributes. "Mni/NodeList/MDeviceList/MDevice/InstalledModule/@MaxSubModuleCount"

6.3.1.5 Parameter

The Parameter element holds node parameter information within the node-dependent information. The structure of the Parameter element is shown in Fig 6-22.

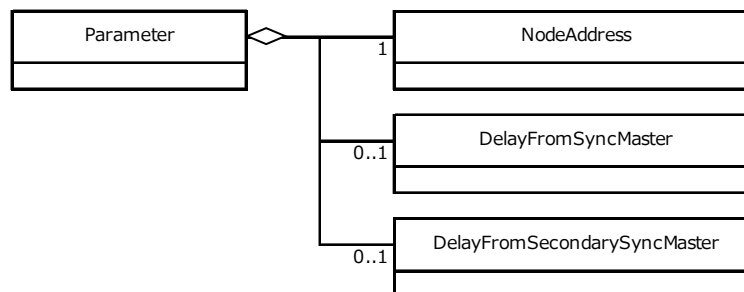


Fig 6-22 Structure of Parameter Element

Details on the component elements of the Parameter element are shown in Table 6-30.

Table 6-30 Component Elements of Parameter Element

Element	Data Type	Support	Explanation
NodeAddress	xs: hexBinary	M	This is the node station address. Setting range: 0001 to FFEF
DelayFromSyncMaster	xs: unsignedInt	C	This is the transmission delay time from the sync master. This element is used only when various tools are used to set the values obtained from devices online. Setting range: 0 to 4,000,000 (ns)
DelayFromSecondary SyncMaster	xs: unsignedInt	C	This is the transmission delay time from the secondary sync master. This element is used only when various tools are used to set the values obtained from devices online. Setting range: 0 to 4,000,000 (ns)

6.3.2 HubList

The HubList element holds information specific to all hubs. This information is not dependent on the control domain and corresponds uniquely to each hub. The structure of the HubList element is shown in Fig 6-23.

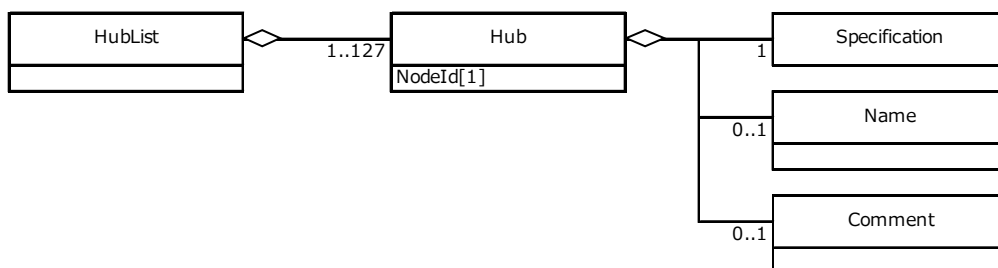


Fig 6-23 Structure of HubList Element

Details on the component elements of the HubList element are shown in Table 6-31.

Table 6-31 Component Elements of HubList Element

Element	Data Type	Support	Explanation
Hub	—	M	This holds the hub-dependent information.
Hub/Specification	—	M	This is the hub specifications information. (See 6.3.2.1)
Hub/Name	xs:string	O	This is hub name.
Hub/Comment	xs:string	O	This contains comment information if needed.

Details on the HubList element and its component element attributes are shown in Table 6-32.

Table 6-32 HubList Element and Component Element Attributes

Attribute	Data Type	Support	Explanation
Hub/@NodeId	xs: ID	M	This is the node identifier for the hub. Setting range: hub000 to hub126

6.3.2.1 Specification

The Specification element holds the hub specifications information within the hub-dependent information. The structure of the Specification element is shown in Fig 6-24.

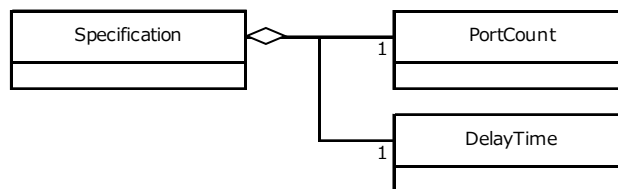


Fig 6-24 Structure of Specification Element

Details on the component elements of the Specification element are shown in Table 6-33.

Table 6-33 Component Elements of Specification Element

Element	Data Type	Support	Explanation
PortCount	xs:unsignedByte	M	This is the number of ports that a hub has. Setting range: 2 to 64
DelayTime	xs:unsignedInt	M	This is the repeat delay time for a hub. Setting range: 0 to 4,000,000 (ns)

APPENDIX A Notation

A.1 Element Structure

The notation for the element structure is based on the UML specifications (<https://www.omg.org/spec/UML/>) described in the OMG (Object Management Group). An example of the element structure notation is shown in Fig A-1. The element name is contained in the top section of the box, and the attribute name is contained in the bottom section of the box. Elements that have a link relationship have a parent-child or brother-sister relationship, and in this document, a parent-child relationship is indicated using an aggregation symbol (blank diamond shape). Also, the number in square brackets after the attribute name indicates the number of attributes, and the multiplicity shown to the left of the top section of the box indicates the number of elements.

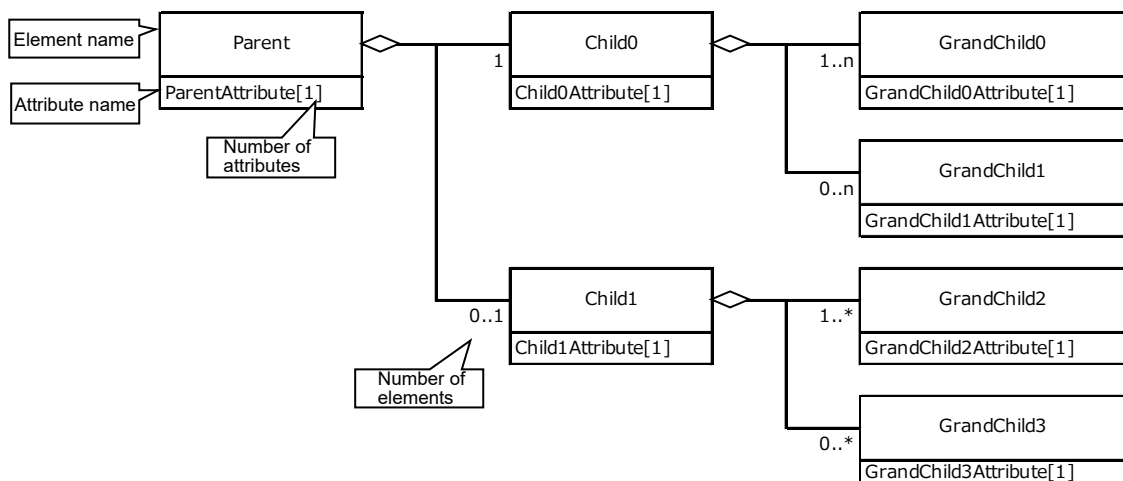


Fig A-1 Example of Element Structure Notation

A.2 Paths

The path notation for elements and attributes is based on the XPath specifications (<https://www.w3.org/TR/xpath/all/>) defined by the W3C (World Wide Web Consortium). However, this omits paths from the root node of the MNI data structure to the highest node within the scope described in each section. Examples of the path notation for component elements of the parent element and the attributes for various elements are shown in Table A-1.

Table A-1 Examples of Path Notation for Elements and Attributes

Element/Attribute	Name	Path Notation
Element	Child0	Child0
	Child1	Child1
	GrandChild0	Child0/GrandChild0
	GrandChild1	Child0/GrandChild1
	GrandChild2	Child1/GrandChild2
	GrandChild3	Child1/GrandChild3
	Attribute	ParentAttribute
Child0Attribute		Child0/@Child0Attribute
Child1Attribute		Child1/@Child1Attribute
GrandChild0Attribute		Child0/GrandChild0/@GrandChild0Attribute
GrandChild1Attribute		Child0/GrandChild0/@GrandChild0Attribute
GrandChild2Attribute		Child0/GrandChild0/@GrandChild0Attribute
GrandChild3Attribute		Child0/GrandChild0/@GrandChild0Attribute

A.3 Support Requirements

Support requirements for elements and attributes are divided into Mandatory, Optional, and Conditional, and Mandatory is abbreviated as "M", Optional as "O", and Conditional as "C". The relationship between the support requirements and multiplicity is shown in Table A-2. The multiplicity for Mandatory is 1 or more, and the multiplicity for Optional is 0 or more.

Table A-2 Support Notation

Support Requirements	Support Notation	Multiplicity
Mandatory	M	1
		1..n
		1..*
Optional	O	0..1
		0..n
		0..*
Conditional	C	0..1
		0..n
		0..*

Revision History

Date of Publication	Rev. No.	Section	Description of Revision
January 2022	<2>	—	Deleted page numbers
		3	Revision: Added section 3.4
		4	Revision: Name of schema file
		6.2.2.2	Revision: Fig 6-10, Table 6-13 Added section 6.2.2.2.3
April 2020	<1>	6.1.2	Addition: the Parameter element
September 2019	—	—	Initial version

