

INTERNATIONAL STANDARD

ISO/IEC
9314-6

First edition
1998-08

Information technology – Fibre distributed data interface (FDDI) –

Part 6: Station Management (SMT)



Reference number
ISO/IEC 9314-6:1998(E)

INTERNATIONAL STANDARD

ISO/IEC
9314-6

First edition
1998-08

Information technology – Fibre distributed data interface (FDDI) –

Part 6: Station Management (SMT)

© ISO/IEC 1998

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the publisher.

ISO/IEC Copyright Office • Case postale 56 • CH-1211 Genève 20 • Switzerland



PRICE CODE **XD**

For price, see current catalogue

CONTENTS

	Page
FOREWORD	v
INTRODUCTION	vi
Clause	
1 Scope.....	1
2 Normative references	2
3 Definitions	2
4 Conventions and abbreviations	4
4.1 Conventions	4
4.1.1 State machines.....	5
4.1.2 Default and initial values	5
4.2 Abbreviations.....	6
5 General description.....	7
5.1 Definition of an FDDI node	7
5.2 Definition of an FDDI network	9
5.2.1 Physical topology	10
5.2.2 Logical topology	11
5.2.3 Physical media topology.....	11
5.2.4 FDDI connection rules.....	12
5.3 Overview of SMT functions	13
6 Services	13
6.1 SMT-to-MAC services	14
6.1.1 SM_MA_INITIALIZE_PROTOCOL.request	14
6.1.2 SM_MA_INITIALIZE_PROTOCOL.confirmation	15
6.1.3 SM_MA_CONTROL.request	15
6.1.4 SM_MA_STATUS.indication	16
6.1.5 SM_MA_UNITDATA.request	17
6.1.6 SM_MA_UNITDATA.indication	18
6.1.7 SM_MA_UNITDATA_STATUS.indication	19
6.1.8 SM_MA_TOKEN.request	19
6.2 SMT-to-PHY services.....	20
6.2.1 SM_PH_LINE-STATE.request	20
6.2.2 SM_PH_STATUS.indication.....	21
6.2.3 SM_PH_CONTROL.request	21
6.3 SMT-to-PMD services	22
6.3.1 SM_PM_CONTROL.request.....	22
6.3.2 SM_PM_BYPASS.request	22
6.3.3 SM_PM_SIGNAL.indication	23
6.4 SMT services to systems management	23
6.4.1 Overview of SMT management services	23
6.4.2 SMT-Management agent process local service primitives	24
6.4.3 Management information base (MIB) structure	24
6.4.4 Integrity of MIB state	25
6.4.5 Management information definitions	26
6.4.5.1 MIB summary.....	26
6.4.5.2 Managed object class templates.....	30
6.4.5.3 Attribute group templates	41
6.4.5.4 Attribute templates	44
6.4.5.5 Action templates	63
6.4.5.6 Notification templates.....	64
6.4.5.7 ASN.1 definitions	67
6.4.5.8 Name binding.....	78

7	Facilities.....	80
7.1	SMT frame format	80
7.1.1	SMT frame contents	80
7.1.2	SMT header.....	80
7.1.3	SMT InfoField	82
7.1.4	SMT encoding rules	83
7.1.5	Byte ordering in multibyte fields	85
7.1.6	Addressing	85
7.1.7	Frame validity	85
7.2	SMT frames.....	85
7.2.1	Neighbour Information Frame (NIF)	86
7.2.2	Status Information Frames (SIF).....	87
7.2.3	ECHO Frame (ECF).....	88
7.2.4	Resource Allocation Frame (RAF) OPTIONAL	89
7.2.5	Request Denied Frame (RDF)	91
7.2.6	Extended Service Frame (ESF) OPTIONAL	91
7.2.7	Status Report Frame (SRF)	92
7.2.8	Parameter Management Frames (PMF)	92
7.3	SMT_Parameters	94
7.3.1	General parameters	94
8	Frame-based management protocols	102
8.1	Frame processing	102
8.1.1	Request–response protocols	102
8.1.2	Announcement protocols	103
8.1.3	SMT header processing.....	103
8.2	Neighbour Notification	104
8.2.1	Neighbour information polling.....	104
8.2.2	Facilities	104
8.2.3	Neighbour Notification transmitter operation	106
8.2.4	Neighbour Notification receiver operation	109
8.3	Status Report protocol.....	109
8.3.1	Overview	109
8.3.2	Facilities	111
8.3.3	Status Report transmitter operation	114
8.4	Parameter Management protocol.....	118
8.4.1	Overview	118
8.4.2	Operation.....	118
8.5	Station Status polling.....	121
8.5.1	Overview	121
8.5.2	Operation.....	121
8.6	Echo protocol	122
8.6.1	Overview	122
8.6.2	Operation.....	122
8.7	Synchronous Bandwidth Allocation.....	122
8.7.1	Overview	122
8.7.2	Operation.....	122
8.7.3	Synchronous bandwidth management process	125
8.8	Extended Service protocol OPTIONAL	126
9	Connection Management.....	126
9.1	Overview	126
9.2	Organization	127
9.3	Connection Management structure	127
9.4	Facilities	127
9.4.1	Variables	127
9.4.2	Signals.....	130
9.4.3	Flags.....	131
9.4.4	Timers	133
9.4.5	Line states	138
9.4.6	Link Confidence Test (LCT)	139

9.4.7	Link Error Monitor (LEM)	140
9.4.8	Path Test	142
9.4.9	Trace function	142
9.5	Entity Coordination Management (ECM)	143
9.5.1	ECM functional description	143
9.5.2	Detailed ECM description	143
9.6	Physical Connection Management (PCM)	147
9.6.1	PCM functional description	147
9.6.2	Detailed PCM description	151
9.6.3	PCM signalling	154
9.7	Configuration Management (CFM)	157
9.7.1	CFM functional description	157
9.7.2	Paths	157
9.7.3	Configuration Control Element (CCE)	165
9.7.4	Station and concentrator structure	166
9.7.5	Configuration element considerations	166
9.7.6	Detailed Configuration Management (CFM) description for Ports	170
9.7.7	Detailed Configuration Management (CFM) description for MACs	177
10	Ring Management	179
10.1	Concepts	179
10.2	Facilities	180
10.2.1	Flags	180
10.2.2	Timer	180
10.3	Operation	183
10.3.1	Overview	183
10.3.2	Detailed description	185

Tables

Table 1 – Summary of SMT frames	81
Table 2 – Station topology matrix	97

Figures

Figure 1 – Example Single Attachment Station (SAS)	8
Figure 2 – Example Dual Attachment Station (DAS)	9
Figure 3 – Example Dual Attachment Concentrator (DAC)	10
Figure 4 – Ring of trees topology	11
Figure 5 – SMT management model	24
Figure 6 – FDDI naming tree	79
Figure 7 – Neighbour Notification transmitter state diagram	108
Figure 8 – Status Report transmitter state diagram	116
Figure 9 – Connection Management structure	128
Figure 10 – Entity Coordination Management (ECM) state diagram	144
Figure 11 – Physical Connection Management (PCM) state diagram	148
Figure 12 – Configuration Control Element (CCE) interfaces	167
Figure 13 – DAS configuration examples	168
Figure 14 – Port Configuration Management (CFM)	172
Figure 15 – MAC Configuration Management (CFM)	178
Figure 16 – Ring Management (RMT) state diagram	188

Annex A (informative) Addressing	190
--	-----

FOREWORD

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

International Standard ISO/IEC 9314-6 was prepared by Joint Technical Committee ISO/IEC JTC 1 *Information technology*, Subcommittee SC 25, *Interconnection of information technology equipment*.

ISO/IEC 9314 consists of the following parts, under the general title *Information technology – Fibre Distributed Data Interface (FDDI)*:

- *Part 1: Token Ring Physical Layer Protocol (PHY) (1989)*
- *Part 2: Token Ring Media Access Control (MAC) (1989)*
- *Part 3: Physical Layer Medium Dependent (PMD) (1990)*
- *Part 4: Single Mode Fibre Physical Layer Medium Dependent (SMF-PMD) ¹⁾*
- *Part 5: Hybrid Ring Control (HRC) (1995)*
- *Part 6: Station Management (SMT)*
- *Part 7: Physical Layer Protocol (PHY-2)*
- *Part 8: Media Access Control-2 (MAC-2)*
- *Part 9: Low-Cost Fibre – Physical Medium Dependent (LCF-PMD) (under consideration)*
- *Part 10: Token Ring Twisted Pair Physical layer Medium Dependent (TP-PMD) (under consideration)*
- *Part 13: Conformance Test Protocol Implementation Conformance Statement Proforma (CT-PICS)*
- *Part 20: Physical Medium Dependent Conformance Testing (PMD-ATS) (under consideration)*
- *Part 21: Physical Layer Protocol Conformance Testing (PHY-ATS) (under consideration)*
- *Part 25: Abstract Test Suite for FDDI – Station Management Conformance Testing (SMT-ATS)*
- *Part 26: Media Access Control Conformance Testing (MAC-ATS) (under consideration)*

¹⁾ To be published.

INTRODUCTION

The Fibre Distributed Data Interface (FDDI), ISO/IEC 9314, is intended for use in a high-performance general purpose multi-node network and is designed for efficient operation with a peak data rate of 100 Mbit/s. It uses a Token Ring architecture with optical fibre as the transmission medium. FDDI provides for hundreds of nodes operating over an extent of tens of kilometres.

Station Management (SMT) specifies the local portion of the system management application process for FDDI, including the control required for proper operation of a node in an FDDI ring. SMT provides services such as connection management, station insertion and removal, station initialization, configuration management, fault isolation and recovery, communications protocol for external authority, scheduling policies, and collection of statistics.

When the set of basic FDDI standards, ISO/IEC 9314, is completed it will include the following standards:

- a) A Media Access Control (MAC), which specifies the lower sublayer of the Data Link Layer of ISO/IEC 9314,
- b) A Physical Layer Media Dependent (PMD), which specifies the lower sublayer of the Physical Layer of ISO/IEC 9314,
- c) A Physical Layer Protocol (PHY), which specifies the upper sublayer of the Physical Layer of ISO/IEC 9314.

A number of extensions to ISO/IEC 9314 are completed or in process. One extension, ISO/IEC 9314-5, for Hybrid Ring Control (HRC), commonly known as FDDI-II, extends the capability of FDDI to handle isochronous data streams at a multiplicity of data rates. Another extension, ISO/IEC 9314-4, provides for a single-mode optical fibre version of PMD (SMF-PMD) and will permit optical links of up to 60 km.

Other extensions, addressing alternate PMDs, will provide low-cost attachments for use in concentrator-to-workstation environments.

This part of ISO/IEC 9314 for SMT represents the final standard in the set of basic FDDI standards. SMT is a sophisticated document specifying many critical aspects of interoperability in a multi-vendor FDDI network and, as such, has proved to be by far the most difficult of the set of FDDI standards to complete. The successful completion of the work on SMT is the result of a high degree of cooperation between competing manufacturers of FDDI equipment.

INFORMATION TECHNOLOGY — FIBRE DISTRIBUTED DATA INTERFACE (FDDI) —

Part 6: Station Management (SMT)

1 Scope

This part of ISO/IEC 9314 specifies the Station Management (SMT) for the Fibre Distributed Data Interface (FDDI).

FDDI provides a high bandwidth (100 megabits per second) general purpose interconnection among computers and peripheral equipment using optical fibre as the transmission medium in a ring configuration. FDDI can be configured to support a sustained transfer rate of approximately 80 megabits (10 megabytes) per second. The use of dual attachment stations with dual MACs allows these rates to be doubled under the circumstance of a fault-free FDDI ring.

FDDI establishes the connection among many stations (nodes) distributed over distances of several kilometres in extent. Default values for FDDI were calculated on the basis of 1 000 physical connections and a total fibre path length of 200 km.

The FDDI consists of

- a) A Physical Layer (PL), which provides the medium, connectors, optical bypassing, and driver/receiver requirements. PL also defines encode/decode and clock requirements as required for framing the data for transmission on the medium or to the higher layers of the FDDI. For the purposes of this part of ISO/IEC 9314, references to the PL are made in terms of the Physical Layer protocol (PHY) and the Physical Layer Media Dependent (PMD) entities which are the upper and lower sublayers of PL, respectively.
- b) A Data Link Layer (DLL) which controls the accessing of the medium and the generation and verification of frame check sequences to assure the proper delivery of valid data to the higher layers. DLL also concerns itself with the generation and recognition of device addresses and the peer-to-peer associations within the FDDI network. For the purposes of this part of ISO/IEC 9314, references to the DLL are made in terms of the Media Access Control (MAC) entity which is the lowest sublayer of DLL.
- c) A Station Management (SMT) standard, this part of ISO/IEC 9314, which provides the control necessary at the station (node) level to manage the processes underway in the various FDDI layers such that a station may work cooperatively as a part of an FDDI network. SMT shall provide services such as connection management, station insertion and removal, station initialization, configuration management, fault isolation and recovery, communications protocol for external authority, scheduling policies, and collection of statistics.

The definition of SMT as contained herein includes the set of services that it provides for, and receives from, the other entities that are contained within a node. Within SMT resides both knowledge of the uniqueness of this node and the current network structure to the extent that this node's function is affected.

The set of International Standards for FDDI, ISO/IEC 9314, specifies the interfaces, functions and operations necessary to insure interoperability between conforming FDDI implementations. This part of ISO/IEC 9314 is a functional description. Conforming implementations may employ any design technique which does not violate interoperability.

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this part of ISO/IEC 9314. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this part of ISO/IEC 9314 are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO/IEC 7498-4(1989) *Information processing systems – Open Systems Interconnection – Basic Reference Model – Part 4: Management framework*

ISO 9314-1: 1989, *Information processing systems – Fibre Distributed Data Interface (FDDI) – Part 1: Token Ring Physical Layer Protocol (PHY)*

ISO 9314-2: 1989, *Information processing systems – Fibre Distributed Data Interface (FDDI) – Part 2: Token Ring Media Access Control (MAC)*

ISO/IEC 9314-3(1990) *Information processing systems – Fibre Distributed Data Interface (FDDI) – Part 3: Physical Layer Medium Dependent (PMD)*

ISO/IEC TR3 8802-1(1997) *Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 1: Overview of Local Area Network Standards*

ISO/IEC 8824: 1990, *Information technology – Open Systems Interconnection – Specification of Abstract Syntax Notation One (ASN.1)*

ISO 8825: 1990, *Information technology – Open Systems Interconnection – Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)*

ISO/IEC 10165-4: 1992, *Information technology – Open Systems Interconnection – Structure of management information – Part 4: Guidelines for the definition of managed objects*

3 Definitions

For the purposes of this part of ISO/IEC 9314, the following definitions apply. In some cases these definitions may duplicate those contained in other parts of ISO/IEC 9314. Such definitions are included for completeness and to improve readability. In certain cases, definitions herein may slightly update those contained in the earlier published parts of ISO/IEC 9314 to improve their clarity. Other parts of ISO/IEC 9314, e.g. FDDI MAC, PHY and PMD, may however, contain additional definitions of interest.

3.1 attachment: The capability of a station or concentrator for connection into an FDDI network. Stations and concentrators are classified as dual attachment, single attachment or null attachment.

3.2 bypass: The ability of a node to optically isolate itself from the FDDI network while maintaining the continuity of the cable plant.

3.3 concentrator: An FDDI node that has additional Ports beyond those required for its own attachment to an FDDI network. These additional Ports (type M – see 5.2.4) are for attaching other FDDI nodes (including other concentrators) in a tree topology.

3.4 counter-rotating: An arrangement whereby two signal paths, one in each direction, exist in a ring topology.

3.5 Dual Attachment Concentrator (DAC): A concentrator that offers a dual attachment to the FDDI network and is capable of accommodating a dual (counter-rotating) ring.

3.6 Dual Attachment Station (DAS): A station that offers a dual attachment to the FDDI network and is capable of accommodating a dual (counter-rotating) ring.

3.7 dual ring (FDDI dual ring): A pair of counter-rotating logical rings.

- 3.8 entity:** An active service or management element within an Open Systems Interconnection (OSI) layer, or sublayer.
- 3.9 fibre optic cable:** A cable containing one or more optical fibres.
- 3.10 Local Path:** A Local Path represents the segment(s) of ring(s) other than the primary ring and secondary ring that pass through the node.
- 3.11 logical ring:** The set of MACs serially connected to form a single ring. A fault-free FDDI network provides two logical rings.
- 3.12 Media Interface Connector (MIC):** A mated connector pair that provides an attachment between an FDDI node and a fibre optic cable plant. The MIC consists of two parts: a MIC plug and a MIC receptacle.
- 3.13 MIC plug:** The male part of the MIC which terminates a fibre optical cable.
- 3.14 MIC receptacle:** The female part of the MIC which is contained in an FDDI node.
- 3.15 network (FDDI network):** A collection of FDDI nodes interconnected to form a trunk, or a tree, or a trunk with multiple trees. This topology is sometimes called a dual ring of trees.
- 3.16 node:** A generic term applying to an active element in an FDDI network (station or concentrator).
- 3.17 Null Attachment Concentrator (NAC):** A concentrator that does not contain an A, B, or S Port.
- 3.18 Path:** A Path represents the segment(s) of a logical ring that pass through a node.
- 3.19 Physical Connection:** The full-duplex physical layer association between adjacent PHY entities (in adjacent nodes) in an FDDI network, i.e. a pair of Physical Links.
- 3.20 Physical Link:** The simplex path (via PMD and attached medium) from the transmit function of one PHY entity to the receive function of an adjacent PHY entity (in adjacent nodes) in an FDDI network.
- 3.21 Port:** A PHY entity and a PMD entity in a node, together creating a PHY/PMD pair, that may connect to the fibre media and provide one end of a physical connection with another node.
- 3.22 Primary Path:** A Primary Path represents, to the best of a node's knowledge, the segment(s) of the primary ring that pass through the node. Conditions may exist in parts of the network which may cause the Path to be in a different ring (e.g. Secondary Path instead of Primary Path).
- 3.23 primitive:** An element of the services provided by one entity to another.
- 3.24 receiver (optical):** An opto-electronic circuit that converts an optical signal to an electrical logic signal.
- 3.25 repeater:** A physical-layer relay in an FDDI network. A repeater is not further defined in this International Standard.
- 3.26 ring:** A set of nodes wherein information is passed sequentially between nodes, each node in turn examining or copying the information, finally returning it to the originating node.
- 3.27 rooted node:** A node that does not have any active A, B, or S Ports in tree mode.
- 3.28 Secondary Path:** A Secondary Path represents, to the best of a node's knowledge, the segment(s) of the secondary ring that pass through the node. Conditions may exist in parts of the network which may cause the Path to be in a different ring (e.g. Primary Path instead of Secondary Path).
- 3.29 services:** The services provided by one entity to another. Data services are provided to a higher layer entity; management services are provided to a management entity in the same or another layer.
- 3.30 Single Attachment Concentrator (SAC):** A concentrator that offers a single attachment to the FDDI network.

3.31 Single Attachment Station (SAS): A station that offers a single attachment to the FDDI network.

3.32 station: An addressable node on an FDDI network capable of transmitting, repeating and receiving information. A station has exactly one SMT, at least one MAC, at least one PHY, and at least one PMD.

3.33 symbol: The smallest signalling element used by the Data Link Layer (DLL). The symbol set consists of 16 data symbols and eight control symbols.

3.34 transmitter (optical): An opto-electronic circuit that converts an electrical logic signal to an optical signal.

3.35 trunk: A physical loop topology, either open or closed, employing two optical fibre signal paths, one in each direction (i.e. counter-rotating), forming a sequence of peer connections between FDDI nodes. When the trunk forms a closed loop it is sometimes called a trunk ring.

3.36 tree: A physical topology consisting of a hierarchy of master-slave connections between a concentrator and other FDDI nodes (including subordinate concentrators).

4 Conventions and abbreviations

4.1 Conventions

The terms SMT, MAC, LLC, PHY, and PMD when used without modifiers, refer specifically to the local entities.

The terms node, station, concentrator and repeater are used as follows in this part of ISO/IEC 9314. The term node is used as a generic term to denote any active element in an FDDI network. Station is used to denote a node that has at least one MAC. Concentrator is used to denote any node that has concentrator capability. The terms station and concentrator thus overlap in such a way that some nodes may be referred to by either term. In this case, the term actually used will be dependent upon the context of the usage. The term repeater is used only to denote a physical-layer relay in an FDDI network and is not further defined.

This International Standard when referring to any of the PMD components assumes the multimode optical fibre PMD. This does not preclude the use of other PMD media types.

Low lines (e.g. requested_service_class) are used as a convenience to form the name of signals, functions, etc., which might otherwise be misinterpreted as independent individual words if they were to appear in text.

In 6.4, hyphens are used in place of low lines. This is done to maintain compatibility of the encoding of the names of attributes with the standard on ASN.1 (see ISO 8824) which does not allow the use of low lines. No difference in meaning is implied. Thus T-Max in 6.4 is exactly equivalent to T_Max as used in clause 10 of this International Standard and in ISO/IEC 9314-2.

The use of a period (e.g. SM_MA_UNITDATA.request) is equivalent to the use of a low line except that a period is used as an aid to distinguish modifier words appended to an antecedent expression.

The use of an asterisk (e.g. *:SM_PM_CONTROL.request) indicates that the primitive is to be sent to all of the entities of the type that receive the primitive. Thus, issuing *:SM_PM_CONTROL.request will send a SM_PM_CONTROL.request to all the PMD entities under control of this SMT.

In the presentation of diagrams, dashed lines are used to indicate optional entities, data paths, transitions and states. Dotted lines are used to indicate a functional unit that may be broken into other functional units.

Timers are given by a name of the form TXX, where XX are two capital letters. An example is the PCM timer, TPC.

This part of ISO/IEC 9314 on FDDI SMT contains four kinds of documentation as follows:

- a) Narrative text, including text associated with state machines,
- b) State machine diagrams, including associated footnotes,

- c) Pseudo code,
- d) Examples, which are specifically noted as such.

If any discrepancies exist between the above, the following precedence shall be used to resolve those discrepancies and determine conformance:

- a) State machine diagrams,
- b) Pseudo code,
- c) Narrative text.

Examples are provided only for clarification and shall not be used for determining conformance.

4.1.1 State machines

SMT operation is defined using cooperating state machines. It is assumed that time elapses only in discrete states, with instantaneous transitions between the states.

State diagrams are expressed using vertical staffs to represent states and horizontal arrows to represent transitions.

Transitions are illustrated with the triggering condition located above the horizontal arrow and any actions on transition located below the transition arrow. Transition actions are performed while remaining in the previous state, before entry into the new state.

The state name appears above the vertical staff representing state and entry actions in the state, if any, appear directly below the state name. Entry actions in a state are executed on any transition entering a state, including those transitions exiting and entering that same state.

The following event-processing sequence is assumed:

- a) Evaluate all transition conditions from the current state;
- b) If a transition condition is satisfied, then
 - 1) perform the associated transition actions in the current state,
 - 2) enter the new state,
 - 3) perform entry actions, if any, for the new state,
 - 4) if a transition condition from this new state is satisfied, then repeat the transition sequence steps 1 to 4;
- c) Perform any instate actions when their associated conditions are satisfied.

Logical symbols are represented in state machines and pseudo code using | to represent the 'or' operator and & to represent the 'and' operator.

Footnotes are used in state diagrams to give precise detail on transition conditions, transitions actions, entry actions, and instate actions

4.1.2 Default and initial values

Default values are defined for many of the attributes and parameters in this part of ISO/IEC 9314. These default values provide correct protocol operation and interoperability of equipment over a broad range of network configurations. Implementers may use alternate values than the defaults specified. The effects of using an alternate value are not directly described.

An initial value is the value assumed by the attribute or parameter as the result of an implementer defined causative action (e.g. power-on). When an initial value is specified in this part of ISO/IEC 9314, the station shall be capable of assuming that value.

4.2 Abbreviations

A	Port type A
ALS	Active Line State
ASN.1	Abstract Syntax Notation One
B	Port type B
BER	Basic Encoding Rules
CCE	Configuration Control Element
CEM	Configuration Element Manager
CF_State	Configuration state
CFM	Configuration Management
CMT	Connection Management
DA_Flag	Duplicate Address flag
DAC	Dual Attachment Concentrator
DAS	Dual Attachment Station
DM-DAS	Dual-MAC Dual Attachment Station
DNA	Downstream Neighbour Address
ECF	ECHO Frame
ECM	Entity Coordination Management
ESF	Extended Service Frame
GDMO	Guidelines for the Definition of Managed Objects
HLS	Halt Line State
ID	Identifier (field)
ILS	Idle Line State
LCT	Link Confidence Test
LEM	Link Error Monitor
LER	Link Error Rate
LS_Flag	Line State flag
LSU	Line State Unknown
M	Port type M
MIB	Management Information Base
MLS	Master Line State
MSB	Most Significant Bit
N	PC_Mode is None
NAC	Null Attachment Concentrator
NIF	Neighbour Information Frame
NLS	Noise Line State
NSA	Next Station Addressing
P	Mode is Peer
Path_Test	Path Test state variable
PC_Withhold	PCM Connection Withhold variable
PCM	Physical Connection Management
PDR	PHY Data Request
PDU	Protocol Data Unit
PMD	Physical Media Dependent
PMF	Parameter Management Frame
QLS	Quiet Line State

R_Val	PCM signalling Received Value array
RAF	Resource Allocation Frame
RDF	Request Denied Frame
RMT	Ring Management
S	Port type S
SAC	Single Attachment Concentrator
SAS	Single Attachment Station
SBA	Synchronous Bandwidth Allocation
SIF	Status Information Frame
SM-DAS	Single-MAC Dual Attachment Station
SRF	Status Report Frame
T	Mode is Tree
T_Val	PCM signalling Transmitted Value array
TEC	Timer, Entity Coordination
TID	Timer, Idle Detection
TNE	Timer, Noise Events
TNN	Timer, Neighbour Notification
TPC	Timer, Physical Connection
TRM	Timer, Ring Management
TSR	Timer, Status Reporting
TVD	Timer, Valid Downstream Neighbour
TVU	Timer, Valid Upstream Neighbour
UNA	Upstream Neighbour Address
UNDA_Flag	Upstream Neighbour Duplicate Address flag
WC_Flag	Withhold Connection flag

5 General description

An FDDI network consists of a set of nodes logically connected. Information is transmitted sequentially, as a stream of suitably encoded symbols, from one active node to the next. Each node generally regenerates and repeats each symbol and serves as the means for attaching one or more devices to the network for the purpose of communicating with other devices on the network. The method of actual physical attachment to an FDDI network may vary and is dependent on specific application requirements as described herein.

The basic building block of an FDDI network is the Physical Connection which consists of paired Physical Layer entities in two adjacent nodes on the FDDI network connected with a transmission medium. Connection to the physical medium, specified in PMD, is controlled by the station insertion and removal algorithms of Station Management (SMT) which are contained herein.

SMT specifies the local functions within an FDDI node necessary to manage the FDDI network. This clause describes the physical and logical topologies of an FDDI network and specifies examples of allowable node configurations used for specification of SMT protocols.

5.1 Definition of an FDDI node

A variety of internal node configurations are possible. However, a node shall have one, and only one, SMT entity. It may, however, have multiple instances of MACs, PHYs, and PMDs, with the actual number, within bounds, being implementer defined.

Several internal node configurations are defined within SMT with state machines provided to specify their operation. These fall into the general classifications of single attachment nodes and dual attachment nodes. FDDI trunk rings are normally composed of dual attachment nodes which have two Ports (each consisting of one PHY and one PMD) to accommodate the dual (counter-rotating) rings.

Concentrators, which may be single attachment, dual attachment, or form the root of a tree of other FDDI nodes (null attachment), are also defined. Concentrators provide for the connection of single attachment nodes (either stations or concentrators) which have only one Port and therefore cannot directly attach to the dual ring. A dual attachment station or concentrator may also be connected to a concentrator for use as the functional equivalent of the corresponding single attachment node (or nodes if Port A and Port B are both connected to concentrators).

A single attachment node requires a minimum of one Port. A dual attachment node requires a minimum of two Ports. To be designated a station, a node requires a minimum of one MAC. Stations may have additional MACs, with MACs inserted in each of the dual rings in the case of dual attachment stations. A concentrator requires one additional Port [type M (see 5.2.4)] for each tree connection that is to be provided. Different concentrator types may or may not have any MACs. The absence of a MAC may limit the functionality of the concentrator. If it has at least one MAC, a MAC shall be located at the exit Port(s) (i.e. a MAC follows the stations inserted on its M Ports). In general, no maximum is placed on the number of MACs that may be contained within a station or concentrator.

The six specific station and concentrator types listed below are used in describing SMT configuration and topology protocols. Figures 1 to 3 depict examples of some of these. Other internal node configurations that are interoperable with these six node types are not precluded by this part of ISO/IEC 9314.

- Single MAC – Dual Attachment Station (SM-DAS)
- Dual MAC – Dual Attachment Station (DM-DAS)
- Single Attachment Station (SAS)
- Dual Attachment Concentrator (DAC)
- Single Attachment Concentrator (SAC)
- Null Attachment Concentrator (NAC)

FDDI nodes are differentiated by PMD entities. Each PHY/PMD pair, designated a Port, belongs to one of four types: A, B, M, or S. The type of a PMD may be indicated by the keying of the MIC receptacle (see the appropriate part of ISO/IEC 9314 on PMD).

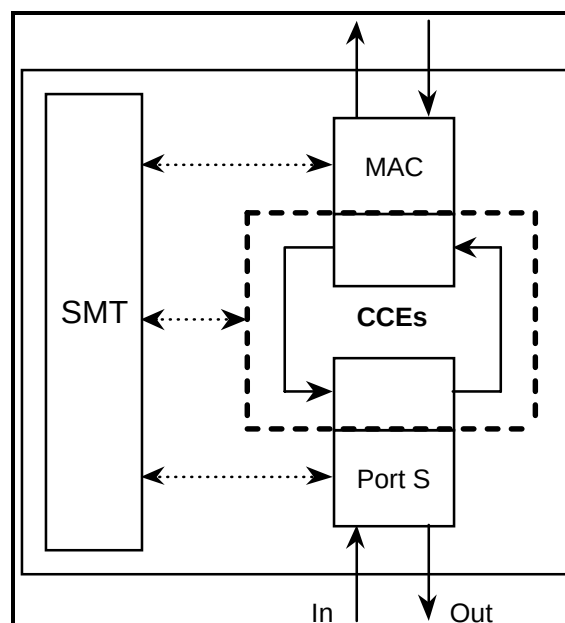


Figure 1 – Example Single Attachment Station (SAS)

Dual attachment nodes (DAS and DAC) may attach directly to the trunk ring. Each dual attachment node contains two Ports designated as A and B. Port A is intended to be connected to the primary ring on the incoming fibre and the secondary ring on the outgoing fibre. Similarly, Port B is intended to be connected to the incoming fibre of the secondary ring and the outgoing fibre of the primary ring. Therefore, a properly formed trunk ring is composed of a set of stations with the Port A of one station connected to the Port B of the neighbouring station.

Concentrator nodes (DAC, SAC, and NAC) contain one or more Ports of type M to provide connections within the concentrator tree.

A single attachment node (SAS or SAC) has a Port of type S which is intended to be attached to a Port of type M within a concentrator tree.

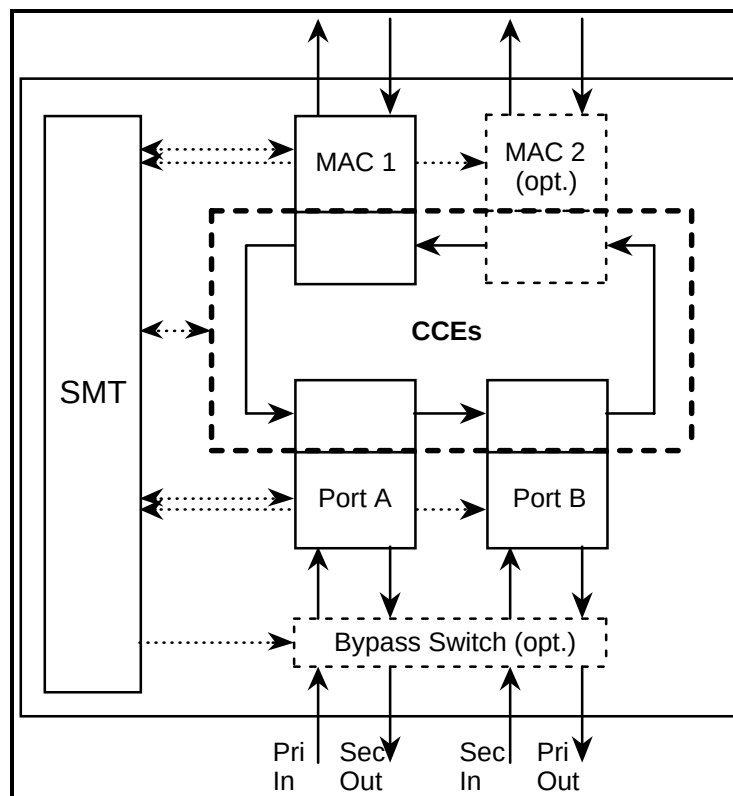


Figure 2 – Example Dual Attachment Station (DAS)

Given that a node may have a variety of internal configurations, with multiple MACs, PHYs, and PMDs, consistent external behaviour shall be required to claim conformance with ISO/IEC 9314. At the discretion of an implementer, a node may be considerably more complex in its internal structure than the example configurations.

This part of ISO/IEC 9314 does not specify implementation. Any node, regardless of its internal implementation, that conforms at its external interfaces to the requirements defined in this part of ISO/IEC 9314 is permitted.

5.2 Definition of an FDDI network

The FDDI network topology may be viewed at two distinct levels: physical and logical. Physical topology describes the arrangement and interconnection of nodes with physical connections. In contrast, logical topology describes the paths by which tokens and information flow through the network between MACs.

An FDDI network forms a dual ring of trees or a subset of a dual ring of trees physical topology. The implication on the logical topology is that at most two logical sets of MAC entities (i.e. two independent token/data Paths) exist in a single, fault-free FDDI network. A set of MAC entities that are in the same token/data Path is called a logical ring hereafter. The valid physical topologies are also constrained to those where physical connections and internal configurations in nodes form at most two logical rings.

The logical and physical topologies of an FDDI network are not necessarily the same. The tree structure provided through concentrators can have the token Path entering and exiting the concentrator many times on the same ring, where logically, the concentrator may only appear on the ring once. Also, the number of MACs and attachments in a station need not be equal; so, a station in the trunk ring is physically in both rings but may be in only one of the logical rings.

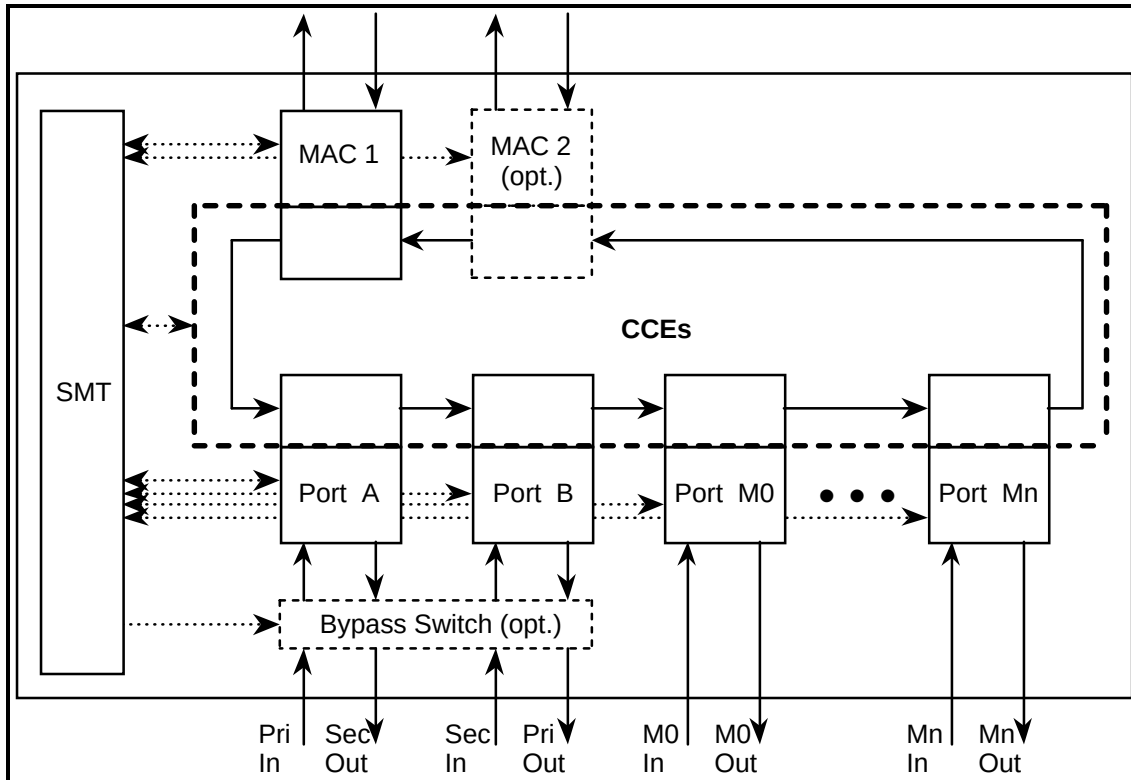


Figure 3 – Example Dual Attachment Concentrator (DAC)

5.2.1 Physical topology

All physical connections in an FDDI topology are duplex links. In a fully connected trunk ring, a duplex link supports counter-rotating rings; and in a tree, the duplex link provides transmit and receive paths for one of the dual rings. Combining these two structures in an FDDI topology, as shown in figure 4, produces a dual ring of trees in a fully connected network. The nodes in the network are interconnected so that at most one dual ring exists.

By removing physical connections and nodes from any legal FDDI topology, one or more legal FDDI topologies are formed. Subsets of legal topologies are legal topologies, so that failure of nodes or connections in an FDDI network produce legal topologies.

A subset of any legal topology is also a legal topology. An FDDI network can be configured in several different physical topologies:

- A dual ring with trees
- A dual ring without trees
- A wrapped ring with trees
- A wrapped ring without trees
- A single tree

5.2.2 Logical topology

At the logical level, a single legal FDDI topology consists of at most two separate logical rings, designated as the primary and secondary rings. The two logical rings are formed from the pairs of opposing physical links that make up the Physical Layer connections. A set of Dual Attachment Stations connected into a closed loop form two counter-rotating rings, referred to as a dual ring. Each of these counter-rotating rings is the basis of a logical ring. If a closed loop of physical connections is not present (i.e. the trunk ring is wrapped) in an FDDI topology, then only one logical ring is present.

Recovery from a single fault on the trunk ring is achieved by joining the two rings in the two nodes adjacent to the fault to create a single logical ring. This wrapped configuration with a single logical ring is also a legal topology. If only one logical ring is present in an FDDI topology, it is designated as the primary ring. Multiple non-adjacent faults on the trunk ring produce multiple FDDI topologies, each with a disjoint logical ring.

An optional recovery process from a single fault, called Global Hold, may be employed. Global Hold is a policy whereby, if both logical rings are operational and a fault is detected in one of the logical rings, then the current configuration is held and all data traffic may be switched to the fault-free logical ring.

A single concentrator may connect one, or two in the case of a DAC, trees into a logical ring. This allows either the primary, the secondary, or both logical rings to be extended to include a tree.

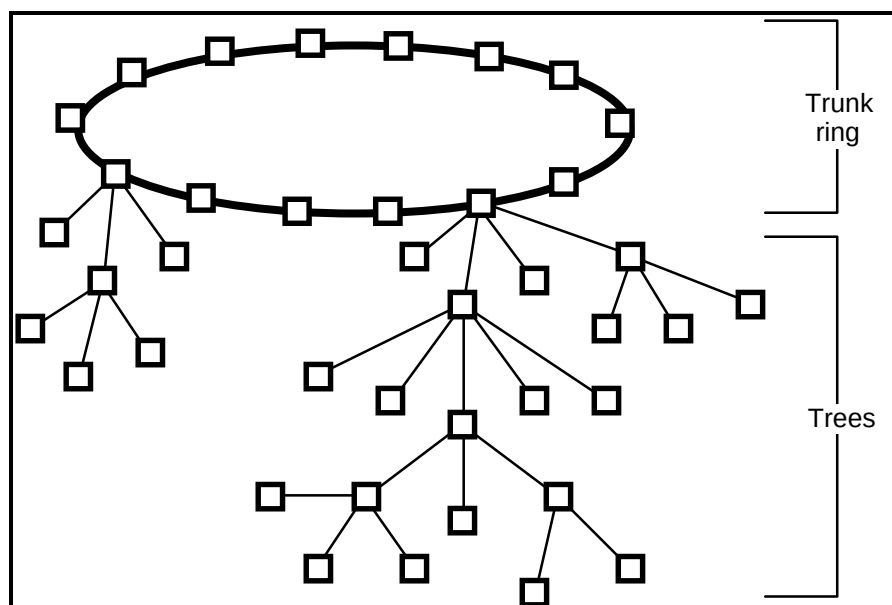


Figure 4 – Ring of trees topology

5.2.3 Physical media topology

The Physical Media topology may contain more than one dual ring. For example, additional connections may exist to provide redundancy within the concentrator trees. These redundant connections are not included in the token Path, which is limited to a single dual ring of trees, but are available as backup links. The construction of networks with redundant tree connections is allowed by ISO/IEC 9314, but the rules for efficient utilization of these connections are not described herein.

Legal FDDI topologies may be constructed with the FDDI Physical Media by following the connection rules set for Ports when connecting FDDI nodes together. This allows legal FDDI topologies to be constructed with the FDDI Physical Media.

5.2.4 FDDI connection rules

The types of Ports (A, B, M, or S) at both ends of a physical connection determine the characteristics of that physical connection. These characteristics include whether the connection will be accepted or rejected, whether SMT will be notified of potential connection problems, and the connection mode that will be established. Connections may be rejected to prevent the establishment of illegal or undesirable topologies. A connection may also be rejected by a neighbouring node because of that node's inability to support the connection. The connection rules matrix that follows summarizes the validity of and the action to be taken for each type of connection:

		Other Port			
		A	B	S	M
This Port	A	V,U	V	V,U	V,W
	B	V	V,U	V,U	V,W
	S	V,U	V,U	V	V
	M	V	V	V	I,U

where

V indicates a valid connection

I indicates an illegal connection

U indicates an undesirable connection with notification to SMT required

W indicates, if Active, prevent THRU in CFM and Port B takes precedence (with defaults)

The detailed connection rules for a specific Port (this Port) to other Ports are:

- A to A Undesirable peer connection that creates twisted primary and secondary rings, notify SMT.
- A to B Normal trunk ring peer connection.
- A to S Undesirable peer connection that creates a wrapped ring, notify SMT.
- A to M Tree connection with possible redundancy. Node shall not go to THRU state in CFM. My Port B shall have precedence (with defaults) for connecting to a Port M in a single MAC node.
- B to A Normal trunk ring peer connection.
- B to B Undesirable peer connection that creates twisted primary and secondary rings, notify SMT.
- B to S Undesirable peer connection that creates a wrapped ring, notify SMT.
- B to M Tree connection with possible redundancy. Node shall not go to THRU state in CFM. My Port B shall have precedence (with defaults) for connecting to a Port M in a single MAC node.
- S to A Undesirable peer connection that creates a wrapped ring, notify SMT.
- S to B Undesirable peer connection that creates a wrapped ring, notify SMT.
- S to S Connection that creates a single ring of two slave stations.
- S to M Normal tree connection.
- M to A Tree connection that provides possible redundancy.
- M to B Tree connection that provides possible redundancy.
- M to S Normal tree connection.
- M to M Illegal connection that creates a tree of rings topology.

When either Port A or Port B is connected to a Port M and the other Port is connected to a Port A, a Port B, or a Port S, then either connection may be withheld in a deterministic manner as specified herein (see clause 9). This avoids a dual attachment node attempting to form part of a tree at one of its Ports and part of a dual ring at the other Port.

Whether to accept or reject the connection depends upon the connection policies adopted by the nodes involved in the connection. The rule is to accept the connection if at least one node's policy is to allow such a connection, and to reject the connection if both nodes have a policy that disallows such a connection.

5.3 Overview of SMT functions

SMT is described herein in terms of various subcomponent functions. These are summarized, on a clause-by-clause basis, as follows.

Clause 6 specifies the services between SMT and other entities. Subclauses 6.1 to 6.3 specify the SMT services with the MAC, PHY, and PMD entities within the same node, whereas 6.4 specifies the services provided by SMT to System Management, and includes specification of the Management Information Base (MIB) of SMT.

Clause 7 specifies the frame formats and protocols used to manage stations in an FDDI ring on a peer-to-peer basis. Included, for instance, is the Neighbour Information Frame (NIF) that is used to periodically announce basic station description information, and the Status Information Frame (SIF) that is used to request and provide a station's operating and configuration information.

Also included in clause 7 are Status Report Frame (SRF) and Parameter Management Frame (PMF) types which are used in conjunction with the protocols specified in clause 8 to provide for remote management of nodes. Complete access to the MIB with read/write capabilities matching those locally available are specified.

Clause 9 specifies the Connection Management (CMT) for a node. CMT supports the wide variety of physical and logical topologies described earlier in this clause. CMT controls the establishment of a media attachment to the FDDI network, the connections with other nodes in the ring, and the internal configuration of the various entities within a node. CMT includes provision for a link confidence test and also specifies a Link Error Monitor (LEM) which monitors active links, on a link-by-link basis, to ensure that failing links are detected and, if required, removed from the network.

Clause 10 specifies the Ring Management (RMT) function which monitors MAC operation and takes actions necessary to aid in achieving an operational ring. RMT occurs on a per MAC basis and aids in the detection and resolution of failures, such as stuck beaconing and the presence of duplicate addresses that can have a global significance.

6 Services

This clause specifies the services provided by SMT and the services required by SMT. The services as defined herein do not imply any particular implementation, or any interface. Services described are

- a) Services between the local MAC entity and SMT (indicated by SM_MA_ prefix);
- b) Services between the local PHY entity and SMT (indicated by SM_PH_ prefix);
- c) Services between the local PMD entity and SMT (indicated by SM_PM_ prefix);
- d) SMT services provided to System Management.

6.1 SMT-to-MAC services

This clause specifies the services provided at the interface between the station management (SMT) entity and MAC. This interface is used by SMT to monitor and control the operation of a local MAC entity. Additional detail is provided in MAC concerning conditions that generate these primitives and MAC actions upon receipt of SMT generated primitives.

The following primitives are defined:

SM_MA_INITIALIZE_PROTOCOL.request
 SM_MA_INITIALIZE_PROTOCOL.confirmation
 SM_MA_CONTROL.request
 SM_MA_STATUS.indication
 SM_MA_UNITDATA.request
 SM_MA_UNITDATA.indication
 SM_MA_UNITDATA_STATUS.indication
 SM_MA_TOKEN.request

Each primitive includes the information that is passed between the MAC and the SMT entities.

6.1.1 SM_MA_INITIALIZE_PROTOCOL.request

This primitive has local significance and is used by SMT to reset MAC and optionally change operational parameters of MAC.

Semantics of the primitive:

```
SM_MA_INITIALIZE_PROTOCOL.request    (
                                     individual_MAC_address (16 bit),
                                     individual_MAC_address (48 bit),
                                     group_MAC_addresses,
                                     T_Min_value,
                                     T_Max_value,
                                     TVX_value,
                                     T_Req_value,
                                     T_Pri_value,
                                     indicate_for_own_frame,
                                     indicate_for_rcv_only_good_frame
                                     )
```

All parameters of this primitive are optional. If a parameter is omitted, MAC shall use the most recently provided value for this parameter or, if no value has been previously provided, the default value for the parameter. References to the attributes contained in 6.4 for the individual parameters are provided in parentheses.

The individual_MAC_address is the octet string that MAC shall use as its individual address. Both a 16-bit address and a 48-bit address may be supplied (see 6.4 for MIB definitions).

Each group_MAC_address is an octet string that MAC shall use as a group address.

T_Min_value specifies the minimum target Token rotation time (TTRT) to be supported.

T_Max_value specifies the maximum target Token rotation time (TTRT) (ref. fddiMACT-Max).

TVX_value specifies the value of TVX to be used (ref. fddiMACTvxValue).

T_Req_value specifies the requested TTRT for asynchronous traffic (ref. fddiMACT-Req).

T_Pri_value specifies a set of priority Token rotation time thresholds (ref. fddiMACT-Pri0 – fddiMACT-Pri6).

The indicate_for_own_frame parameter specifies that MAC shall also receive and indicate to SMT frames that it has transmitted.

The indicate_for_rcv_only_good_frames parameter is the value MAC shall use to decide whether to generate MA_UNITDATA.indication and SM_MA_UNITDATA.indication primitives only on frames that are good, or alternatively on all frames.

When generated:

This primitive shall be generated by SMT whenever it requires MAC to reconfigure.

Effect upon receipt:

Receipt of this primitive causes MAC to establish the values of its addresses, timers, and other initialization parameters. Upon completion of this primitive, MAC generates a SM_MA_INITIALIZE_PROTOCOL.confirmation.

6.1.2 SM_MA_INITIALIZE_PROTOCOL.confirmation

This primitive is used by the MAC to inform SMT that the SM_MA_INITIALIZE_PROTOCOL.request is complete.

Semantics of the primitive:

```
SM_MA_INITIALIZE_PROTOCOL.confirmation    (
                                          status
                                          )
```

The status parameter indicates the success or failure of the SM_MA_INITIALIZE_PROTOCOL.request.

When generated:

This primitive is generated by MAC upon completion of a SM_MA_INITIALIZE_PROTOCOL.request.

Effect upon receipt:

The receipt of this primitive by SMT shall signify to SMT that the desired reconfiguration has been accomplished.

6.1.3 SM_MA_CONTROL.request

This primitive has local significance and is used by SMT to control the operation of a local MAC entity.

Semantics of the primitive:

```
SM_MA_CONTROL.request    (
                          control_action,
                          beacon_information,
                          requested_status,
                          requested_condition
                          )
```

The control_action parameter shall include the following: Reset, Beacon, Present_Status, Reset_Counters, interrupt_upon_condition, send bad FCS, or optionally Claim.

The beacon_information parameter shall contain the DA field and the Information field for use in the Beacon Frame if the control_action parameter specifies Beacon.

The requested_status parameter shall include the following: current values of all counters, current values of all timers, current receiver state machine state, and current transmitter state machine state.

The requested_condition parameter shall include: capture of Token, receipt of frame.

When generated:

This primitive is generated by SMT to cause MAC to take the action specified by the control_action parameter.

Effect upon receipt:

Receipt of this primitive by MAC causes it to take the following actions:

- a) If the control_action parameter is Reset or Beacon then MAC shall:
 - generate the MAC_Reset signal
 - enter receiver state R0
 - enter transmitter state T0;

- b) If the control_action parameter is Beacon, then following the above, State T5 shall be entered in the transmitter state machine. The Beacon Frames shall be constructed using the beacon_information parameter;
- c) If the control_action parameter is Present_Status, then MAC shall present the status to SMT as indicated by the requested_status parameter;
- d) If the control_action parameter is Reset_Counters, then MAC shall reset all counters;
- e) If the control_action parameter is interrupt_upon_condition, then MAC shall signal SMT when the requested_condition is detected;
- f) If the control_action parameter is send bad FCS, then MAC shall send bad FCS with a specific frame to be transmitted.

6.1.4 SM_MA_STATUS.indication

This primitive is used by MAC to inform SMT of errors and significant status changes.

Semantics of the primitive:

```
SM_MA_STATUS.indication      (
                                status_report
                                )
```

The status_report parameter specifies appropriate status including the following:

- a) any change of Ring_Operational status;
- b) receipt of My_Claim, Higher_Claim or Lower_Claim;
- c) receipt of My_Beacon or Other_Beacon;
- d) expiration of TVX;
- e) expiration of TRT and Late_Ct _ 0;
- f) receipt of PH_Invalid causing the receiver to enter State R0;
- g) receipt of MAC_Reset causing the receiver to enter State R0;
- h) overflow of Frame_Ct, Error_Ct or Lost_Ct;
- i) receipt of a frame with the Destination Address equal to this station's individual address and the received A Indicator set;
- j) recognition that this station's transmitted frame was lost in the course of its circulation around the ring;
- k) anytime the R_Flag goes from reset to set;
- l) anytime a station sets Ax but does not set Cx;
- m) receipt of a Token if the MAC transmitter is in State T2 or T3;
- n) expiration of TRT in MAC transmitter State T4 or T5;
- o) occurrence of any condition specified in a previously received SM_MA_CONTROL.request (interrupt_upon_condition);

When generated:

This primitive is generated by MAC when any of the listed reportable conditions are detected.

Effect upon receipt:

The receipt of this primitive shall cause SMT to take the actions specified in this International Standard.

6.1.5 SM_MA_UNITDATA.request

This primitive defines the transfer of one or more SMT Service Data Units (SDU) from SMT. Any defined PDU format may be sent.

Semantics of the primitive:

```

SM_MA_UNITDATA.request
(
    FC_value (1),
    destination_address (1),
    M_SDU (1),
    requested_service_class (1),
    stream (1),
    FC_value (2),
    destination_address (2),
    M_SDU (2),
    requested_service_class (2),
    stream (2),
    .
    .
    .
    FC_value (n),
    destination_address (n),
    M_SDU (n),
    requested_service_class (n),
    stream (n),
    Token_class
)

```

Each set of FC_value, destination_address, M_SDU, requested_service_class and stream parameters specifies one frame for transmission and is referred to as a subrequest.

The FC_value parameter supplies the FC field to be transmitted as part of the frame.

The destination_address parameter may specify either an individual or a group MAC address. It shall contain sufficient information to create the DA field that is appended to the frame by MAC. Address length is determined by the L bit of the associated FC_value parameter.

Each M_SDU parameter specifies an SMT service data unit as received at the MAC interface to be transmitted by MAC. There is sufficient information associated with the M_SDU for MAC to determine the length of the service data unit. Associated with each M_SDU is a requested_service_class parameter.

Requested_service_class may be either synchronous, asynchronous or immediate. If asynchronous the priority level may also be specified. If not specified the default value is assumed. Immediate, if specified, shall cause the frame to be sent immediately without waiting for receipt of a Token.

NOTE 1 -The use of synchronous as the requested_service_class is dependent upon allocation of the required synchronous bandwidth.

Stream is a parameter which, if set, shall cause multiple M_SDUs to be transmitted as a result of the same SM_MA_UNITDATA.request. Stream, when reset, indicates that this SDU is the last one associated with this SM_MA_UNITDATA.request. The frames shall be transmitted in the order presented by this primitive regardless of the associated requested_service_class. If TRT has expired (Late_Ct not= 0) or if a frame is encountered which cannot be transmitted because of its associated requested_service_class and the current value of THT, then transmission is terminated and a Token is issued as defined by the Token_class parameter. A SM_MA_UNITDATA_STATUS.indication is subsequently returned to SMT. If the transmission_status is successful, then MAC may initiate transmission of the remaining frames on the next permitted access opportunity or alternatively, MAC may require a new SM_MA_UNITDATA.request.

Token_class specifies the class of Token MAC shall issue following the transmission of the associated SDUs if no other request that can be honoured is pending. It may specify either none, restricted or non-restricted. If no SDUs were specified by the SM_MA_UNITDATA.request, then MAC shall issue the requested class of Token.

When generated:

This primitive shall be generated by SMT whenever data is to be transferred to a peer SMT entity or entities or a Token is to be generated.

Effect of Receipt:

The receipt of this primitive causes MAC to append all MAC specific fields, including DA, SA, and any fields that are unique to the medium access method, and pass the properly formed frames to the lower layers of protocol for transfer to the peer MAC entity or entities.

NOTE 2 – The capture of a Token is implicit in this primitive and therefore it is not necessary to issue an SM_MA_TOKEN.request primitive in conjunction with it.

6.1.6 SM_MA_UNITDATA.indication

This primitive defines the transfer of data from a local MAC entity to SMT. This primitive shall be able to report any SMT or MAC frame addressed to the station.

Semantics of the primitive:

```
SM_MA_UNITDATA.indication
(
    FC_value,
    destination_address,
    source_address,
    M_SDU,
    reception_status
)
```

The FC_value parameter specifies the value for the frame's FC field. The destination_address parameters may be either an individual or a group address as specified by the DA field of the incoming frame. The source_address parameter is an individual address as specified by the SA field of the incoming frame. The M_SDU parameter shall specify the MAC service data unit as received by the local MAC entity.

The reception_status parameter indicates the success or failure of the incoming frame. It consists of the following elements:

- a) frame validity: FR_GOOD, FR_BAD
 - If a FR_BAD is reported, the reason for the error shall also be reported. The reason shall be one of the following:
 - 1) Invalid FCS – calculated FCS does not match the received FCS
 - 2) Length error – the frame did not have a valid data length
 - 3) Internal error – an internal error has occurred which prevents MAC from transferring to SMT a frame that has been acknowledged by the setting of the A and C indicators;
- b) frame status:
 - The received E, A, C, and, optionally, any other indicator values.

When generated:

The SM_MA_UNITDATA.indication primitive shall be generated by MAC to indicate the arrival of a MAC or SMT frame addressed to this station at the local MAC entity. If so initialized (ref. SM_MA_INITIALIZE.PROTOCOL.request), the MAC shall also generate this primitive upon receipt of frames that it has transmitted. This allows examination of elements of the reception_status for a transmitted frame following circulation around the ring.

Effect of Receipt:

The receipt of this primitive shall cause SMT to take the actions specified in this International Standard.

6.1.7 SM_MA_UNITDATA.STATUS.indication

This primitive shall provide an appropriate response to the SM_MA_UNITDATA.request primitive generated by SMT signifying the success or failure of the request.

Semantics of the primitive:

```
SM_MA_UNITDATA.STATUS.indication    (
                                     number_of_SDUs,
                                     transmission_status,
                                     provided_service_class
                                     )
```

The number_of_SDUs parameter reports the number of M_SDUs transmitted on a given access opportunity as a result of this request.

The transmission_status parameter shall be used to pass information back to the local requesting SMT entity. It shall be used to indicate the success or failure of the previous associated SM_MA_UNITDATA.request. If the SM_MA_UNITDATA.request primitive specified more than one M_SDU then the transmission_status parameter shall apply to all of the SDUs transmitted indicating if all were acknowledged, via the A and C indicators, by a peer MAC entity.

The provided_service_class parameter specifies the service class that was provided for the transfer.

When generated:

This primitive is generated by the MAC entity in response to an SM_MA_UNITDATA.request primitive from SMT.

Effect of Receipt:

The receipt of this primitive by SMT shall indicate to SMT the success or failure of the request.

NOTE 3 – In the event of multiple outstanding requests, additional information may be required in order for SMT to associate the response with the appropriate request. The association may be implied by the requests being serviced in a first-in-first-out manner. Alternatively, MAC may maintain multiple queues of requests, at least one for each class_of_service implemented, servicing each of these queues in a first-in-first-out manner. It is assumed that if the addressed peer MAC entity, or entities, acknowledge receipt of the frame(s) (by setting the A and C indicators), then either the frame(s) will be delivered, or an internal error will be reported, to the corresponding SMT entity, or entities, via an SM_MA_UNITDATA.indication.

6.1.8 SM_MA_TOKEN.request

This primitive is used by SMT to request the capture of the next Token.

Semantics of the primitive:

```
SM_MA_TOKEN.request    (
                        requested_Token_class
                        )
```

Requested_Token_class may be either restricted or non-restricted. The priority level may also be specified if multiple levels of priorities are implemented.

When generated:

This primitive may be generated by the local SMT entity when data of a time critical nature is to be transferred.

Effect of Receipt:

The receipt of this primitive causes MAC to capture the next usable Token based on the requested_Token_class parameter. MAC then enters State T2 and transmits Idle symbols until an SM_MA_UNITDATA.request primitive is received from SMT unless TRT expires first, in which case MAC shall issue another Token of the same Token_class as was captured.

NOTE 4 – This primitive may be used for time critical operations to minimize the effects of ring latency. This mode of operation may cause longer than usual preambles preceding a frame and may lock out other stations from gaining fair access to transmit data. It is an additional mode of operation which results in the capture of an appropriate Token following the receipt of SM_MA_UNITDATA.request and therefore should not be used for non-time critical transfers of data on the FDDI ring.

6.2 SMT-to-PHY services

The services supplied by PHY allow SMT to control the operation of PHY. Additional detail is provided in PHY concerning conditions that generate these primitives and PHY actions upon receipt of SMT generated primitives. The following primitives are defined:

SM_PH_LINE-STATE.request
SM_PH_STATUS.indication
SM_PH_CONTROL.request

All primitives described in this subclause are mandatory. Each primitive includes the information that will be passed between PHY and SMT.

6.2.1 SM_PH_LINE-STATE.request

This primitive is generated by SMT to request PHY to send a continuous stream of similar symbols.

Semantics of the primitive:

```
SM_PH_Line-State.request      (
                                Line-State_action
                                )
```

The Line-State_action parameter shall be one of the following:

TRANSMIT_QUIET – This parameter is asserted to cause PHY to send a continuous stream of Quiet symbols. In this condition the medium transmitter generates no transitions.

NOTE 5 – It is recommended that the transmitter be inactive (i.e. no light output).

TRANSMIT_HALT – This parameter is asserted to cause PHY to send a continuous stream of Halt symbols.

TRANSMIT_IDLE – This parameter is asserted to cause PHY to send a continuous stream of Idle symbols.

TRANSMIT_MASTER – This parameter is asserted to cause PHY to send a continuous stream of alternating Halt and Quiet symbol pairs.

TRANSMIT_PDR – This parameter is asserted to cause PHY to send the stream of symbols presented to the PH_UNITDATA.request interface.

When generated:

These primitives shall be generated by SMT as part of station insertion or removal sequences.

Effect upon receipt:

PHY sends a continuous stream of the commanded symbol or symbol pair out the outbound medium.

6.2.2 SM_PH_STATUS.indication

This primitive is generated by PHY to inform SMT of Line-State activity and status changes. The specific items reported are defined in the following.

Semantics of the primitive:

```
SM_PH_STATUS.indication      (
                               status_report
                              )
```

The status_report parameter shall be one of the following:

QUIET_LINE-STATE_RECEIVED (QLS) – This parameter is asserted by PHY when the Quiet_Line-State is received on the inbound medium.

HALT_LINE-STATE_RECEIVED (HLS) – This parameter is asserted by PHY when the Halt_Line-State is received on the inbound medium.

MASTER_LINE-STATE_RECEIVED (MLS) – This parameter is asserted by PHY when the Master_Line-State is received on the inbound medium.

IDLE_LINE-STATE_RECEIVED (ILS) – This parameter is asserted by PHY when the Idle_Line-State is received on the inbound medium.

ACTIVE_LINE-STATE_RECEIVED – This parameter is asserted by PHY when the Active_Line-State (ALS) is entered.

NOISE_LINE-STATE_RECEIVED – This parameter is asserted by PHY when the Noise_Line-State (NLS) is entered.

LINE-STATE_UNKNOWN – This parameter is asserted by PHY when any of the defined line states are exited and the entry conditions to a new line state have not yet been satisfied. Also included is an indication of the most recently known line state.

When generated:

These primitives are generated by PHY to signal the occurrence of the indicated condition.

Effect upon receipt:

The receipt of this primitive shall cause SMT to take the actions specified in the state machines contained herein.

6.2.3 SM_PH_CONTROL.request

This primitive has local significance and is used by SMT to control the operation of PHY.

Semantics of the primitive:

```
SM_PH_CONTROL.request      (
                             Control_Action,
                             Requested_Status
                            )
```

The Control_Action parameter shall include the following: Reset, Present_Status, Begin_Loopback, or Cancel_Loopback.

The Requested_Status parameter causes PHY to report its status which includes the current line state. If the current line state is unknown, then LINE-STATE_UNKNOWN is reported along with the most recent known line state.

When generated:

This primitive shall be generated by SMT to cause PHY to take the action specified by the Control_Action parameter.

Effect upon receipt:

PHY takes the requested action.

6.3 SMT-to-PMD services

The services supplied by PMD allow SMT to control the operation of PMD. The PMD performs the requested SMT services preemptively over any requested PHY services. Additional detail is provided in PMD concerning conditions that generate these primitives and PMD actions upon receipt of SMT generated primitives.

The following primitives are defined:

SM_PM_CONTROL.request
SM_PM_BYPASS.request
SM_PM_SIGNAL.indication

Each primitive includes the information that is passed between the PMD and SMT entities.

6.3.1 SM_PM_CONTROL.request

This primitive is generated by SMT and asserted to PMD to force the transmit function to place a logic "0" optical signal on the outbound medium.

Semantics of the primitive:

```
SM_PM_CONTROL.request      (
                             Control_Action
                             )
```

The Control_Action parameter shall include the following: Transmit_Enable and Transmit_Disable.

When generated:

SMT shall generate this primitive whenever it wants to enable or disable the PMD optical transmitter.

Effect upon receipt:

Receipt of this primitive by PMD with a Control_Action parameter of Transmit_Disable causes PMD to transmit a logic "0" optical signal (i.e. low light) preemptively over the PM_UNITDATA.request primitive.

Receipt of this primitive by PMD with a Control_Action parameter of Transmit_Enable causes PMD to transmit the optical signal requested by the PM_UNITDATA.request primitive.

6.3.2 SM_PM_BYPASS.request

This primitive is generated by SMT and asserted to PMD to indicate that SMT wants to join or leave the FDDI network.

Semantics of the primitive:

```
SM_PM_BYPASS.request      (
                             Control_Action
                             )
```

The Control_Action parameter shall include the following: Insert, Deinsert.

When generated:

SMT shall generate this primitive whenever it wants to activate or deactivate the optical switch(es).

Effect upon receipt:

Upon receipt of this primitive with a Control_Action parameter of Insert, PMD activates the optical switch such that the MIC inbound optical signal from the cable plant is directed to the optical receiver. The output of the optical transmitter shall be directed to the MIC output to the cable plant.

Upon receipt of this primitive with a `Control_Action` parameter of `Deinsert`, PMD deactivates the optical switch such that the MIC inbound optical signal from the cable plant is directed through the switch to the MIC output to the cable plant. The output of the optical transmitter shall be directed through the optical switch to the input of the optical receiver. This state is called the bypassed mode.

NOTE 6 – Optical bypass switches are optional in an FDDI network. Stations that do not employ optical switches do not require this service.

6.3.3 SM_PM_SIGNAL.indication

This primitive is generated by PMD and asserted to SMT to indicate the status of the optical signal level being received by PMD.

Semantics of the primitive:

```
SM_PM_SIGNAL.indication      (
                               Signal_Detect(status)
                              )
```

The `Signal_Detect(status)` parameter indicates if the inbound optical signal level is above (status = on) or below (status = off) the optical signal detection threshold.

When generated:

PMD generates this primitive to indicate the status of the `Signal_Detect`.

Effect upon receipt:

SMT shall use this primitive to determine the status of the optical signal level being received by PMD and shall take actions as specified in the state machines contained herein.

6.4 SMT services to systems management

This subclause provides the specification of management information related to the operation of the SMT. SMT management information is defined by specifying:

- the syntax and semantics of the information elements;
- the relationship of the information elements to the operation of SMT and to other information elements of SMT;
- the operations on the information elements that are available to Systems Management.

6.4.1 Overview of SMT management services

The SMT management information that is supplied to systems management is represented using an object oriented approach consistent with the approach taken by OSI Management Standards. See ISO/IEC 7498-4:1989 and ISO/IEC 10165-4:1992. Station Management Information is described in terms of Managed Objects. Figure 5 depicts the SMT Object Model. It shows the relationship of managed objects within an FDDI station and the components of SMT: the SMT frame services (see clause 8), Connection Management (see clause 9), and Ring Management (see clause 10).

This part of ISO/IEC 9314 defines four Managed Object Classes. The SMT managed object class models the management information for an FDDI station (node). The MAC managed object class models the management information required for the MAC entities within a station. The Path managed object class models the management information required for the management of configuration Paths within a station. The Port object class models the management information required for the PHY/PMD entities within a station.

A particular instance of a managed object class is called a managed object. A particular FDDI station can have multiple MAC, Path, and Port managed objects but only one SMT managed object. The management of these managed objects is accomplished through notifications and operations.

Notifications are unsolicited reports generated by a managed object. Operations are functions performed by a management agent process on a managed object. Get and Set operations are defined in this part of ISO/IEC 9314.

6.4.2 SMT-Management agent process local service primitives

The definition of the local primitives or service interfaces, that are used to read and write attributes, or to cause an action or report an event, is beyond the scope of this part of ISO/IEC 9314.

NOTE 7 – SMT uses the MSB address form in its Management Information Base. Some standard management agent processes use the Canonical representation for interchange of addresses. The implementer should be aware that in this case, conversion between MSB representation and Canonical representation is required when the management agent process accesses a station's MIB either locally, or remotely through SMT frames.

6.4.3 Management information base (MIB) structure

The SMT managed object classes are defined using the formats defined in the ISO 10165-4, hereafter referred to as GDMO.

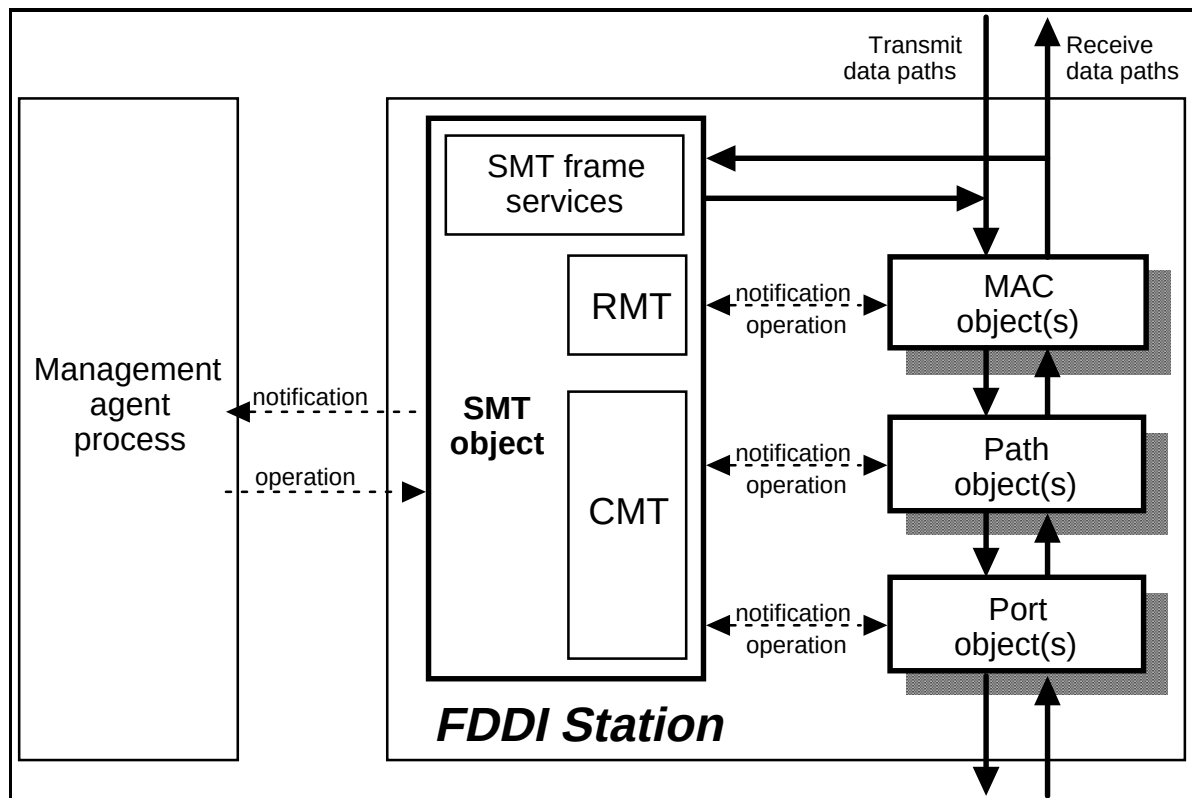


Figure 5 – SMT management model

GDMO presents specific structures, called templates, in a formal description language, that provides the information necessary for the application of object oriented design principles in the definition of OSI management for Open Systems, for protocol layers, and for communication components.

These templates serve to describe the managed object classes in a precise manner. Each managed object class is defined using templates that describe the behaviour, attributes, notifications, and actions for that class.

The data structures required to support the GDMO templates are defined using ISO 8824. ASN.1 is a formal notation for documenting data structures. The templates for the four SMT-defined managed object classes are contained in the management information definitions (see 6.4.5). This part of ISO/IEC 9314 uses a template format that places GDMO templates within templates. This use of "inline templates" improves the readability of this part of ISO/IEC 9314. A summary of the characteristics of the object classes is found in the MIB Summary (see 6.4.5.1).

The managed object class templates (see 6.4.5.2) contain the definition of the object classes. This definition includes the specification of the following characteristics.

- a) Object class behaviour;
- b) Mandatory object class attributes;
- c) Mandatory object class notifications;
- d) Mandatory object class actions;
- e) Conditional object class attributes;
- f) Conditional object class notifications;
- g) Conditional object class actions.

The conditional aspects of a managed object are represented in the object class template as a conditional package. The conditional packages define the conditions under which conditional attributes, notifications, and actions are included in an object instance.

The attribute group templates (see 6.4.5.3) contain the definitions of attribute groups within an object class. These templates specify the attributes that are operated on when group operations are supported by a station. This part of ISO/IEC 9314 specifies the PMF GET as the only group operation.

The attribute templates (see 6.4.5.4) define the attributes of an object class. These templates define the behaviour and registration of a particular attribute. The "WITH ATTRIBUTE SYNTAX" clause indicates the ASN.1 definition of the attribute.

The action templates (see 6.4.5.5) define the actions of an object class. These templates define the behaviour, confirmation mode, and registration of a particular action. The "WITH INFORMATION SYNTAX" clause indicates the ASN.1 definition of the action. The "WITH REPLY SYNTAX" clause indicates the ASN.1 definition of the reply to the action.

The notification templates (see 6.4.5.6) define the notifications of an object class. These templates define the behaviour, confirmation mode, and registration of a particular notification. The "WITH INFORMATION SYNTAX" clause indicates the ASN.1 definition of the notification. All notifications in this part of ISO/IEC 9314 are unconfirmed.

The ASN.1 definitions (see 6.4.5.7) contain the formal representation of the data structure of each attribute, action, and notification.

The name binding (see 6.4.5.8) contains a formal description of the naming scheme associated with the object classes. The name binding templates indicate how instances of managed object classes are named within a station.

6.4.4 Integrity of MIB state

The MAC, Path and Port objects may have multiple instances within the MIB. In order to support implementations with flexible internal configurations, a MAC or Port object instance in the MIB may not necessarily be associated with a physical resource of that type within the station. Each MAC and Port object instance and its associated attributes are identified by a resource index within the object type. These indexes, assigned by the implementer, shall be non-zero.

An implementation is not required to maintain the state of any MIB object across discontinuities in the management of that object. Permitted discontinuities include SMT initialization, power off/on transitions, self-test, and, for MAC and Port objects, unavailable/available transitions of the fddiMACHardwarePresent or fddiPORTHardwarePresent attributes. It is also possible that detection of a major inconsistency in the internal state of SMT may cause a discontinuity.

An implementation is not required to maintain the state of any object across discontinuities in the instantiation of that object. The mandatory package shall be included in all instantiations of that object class. Any discontinuity shall cause the object to be re-instantiated. A conditional package shall be included in an instantiation of an object class when the conditions for its presence are satisfied.

After any discontinuity in the instantiation of an object class its state components (attributes, actions, and notifications) shall be deterministically set to their initial values, with the following exceptions.

- a) A resource index attribute may assume any prior value, if known; otherwise it shall be assigned an initial value. There may be less impact to management applications if a prior value is used.
- b) A statistical counter attribute may assume its latest prior value, if known, and if the resource index for its associated object is known not to have changed; otherwise it shall be reset to its initial value. The value of statistical information is generally enhanced if the continuity of counters is maintained.
- c) A time stamp or set count attribute may assume any value greater than or equal to its latest prior value, if known, and if the StationId is known not to have changed; otherwise it shall be reset to its initial value. The value of statistical information is generally enhanced if, when the continuity of counters is maintained, the continuity of time stamps is also maintained.

When a request to alter a MIB attribute is honoured, either from the local systems management entity or via the Parameter Management protocol, SMT shall set the value of the attribute exactly to the requested value.

When a MIB attribute is altered or a MIB action is invoked, either from the local systems management entity or via the Parameter Management protocol, SMT shall ascertain whether the resulting MIB state is at variance with the current operation of the node. If there is a variance, SMT shall act to restore consistency in a timely and deterministic manner.

For those attributes designated read-write for which a specific range of values is defined in the MIB, the entire range of values shall be supported, unless otherwise specified in the attribute behaviour description.

6.4.5 Management information definitions

6.4.5.1 MIB summary

The following sections summarize the FDDI Management Information Base. Each group, attribute, action and notification is listed by its name, access (G = GET, GR = GET-REPLACE, S = PMF-Set, NA = Not Accessible), status (M = MANDATORY, MP = MANDATORY if hardware present, O = OPTIONAL, OP = OPTIONAL but present only when hardware present), Parameter_Type, and abbreviated registration information. This summary is for information purposes only; if any differences exist, the formal definitions have precedence.

6.4.5.1.1 SMT attributes

fddiSMTStationIdGrp	G	M	10 0A	fddiSMT 10
fddiSMTStationId	G	M	10 0B	fddiSMT 11
fddiSMTOpVersionId	G	M	10 0D	fddiSMT 13
fddiSMTHiVersionId	G	M	10 0E	fddiSMT 14
fddiSMTLoVersionId	G	M	10 0F	fddiSMT 15
fddiSMTManufacturerData	G	O	10 10	fddiSMT 16
fddiSMTUserData	GR	M	10 11	fddiSMT 17
fddiSMTMIBVersionId	G	M	10 12	fddiSMT 18
fddiSMTStationConfigGrp	G	M	10 14	fddiSMT 20
fddiSMTMac-Ct	G	M	10 15	fddiSMT 21
fddiSMTNonMaster-Ct	G	M	10 16	fddiSMT 22
fddiSMTMaster-Ct	G	M	10 17	fddiSMT 23
fddiSMTAvailablePaths	G	M	10 18	fddiSMT 24
fddiSMTConfigCapabilities	G	M	10 19	fddiSMT 25
fddiSMTConfigPolicy	GR	M	10 1A	fddiSMT 26

fddiSMTConnectionPolicy	GR	M	10 1B	fddiSMT 27
fddiSMTT-Notify	GR	M	10 1D	fddiSMT 29
fddiSMTStatRptPolicy	GR	M	10 1E	fddiSMT 30
fddiSMTTrace-MaxExpiration	GR	M	10 1F	fddiSMT 31
fddiSMTPORTIndexes	G	M	10 20	fddiSMT 32
fddiSMTMACIndexes	G	M	10 21	fddiSMT 33
fddiSMTBypassPresent	G	M	10 22	fddiSMT 34
fddiSMTStatusGrp	G	M	10 28	fddiSMT 40
fddiSMTECMState	G	M	10 29	fddiSMT 41
fddiSMTCF-State	G	M	10 2A	fddiSMT 42
fddiSMTHoldState	G	O	10 2B	fddiSMT 43
fddiSMTRemoteDisconnectFlag	G	M	10 2C	fddiSMT 44
fddiSMTStationStatus	G	M	10 2D	fddiSMT 45
fddiSMTPeerWrapFlag	G	M	10 2E	fddiSMT 46
fddiSMTMIBOperationGrp	G	M	10 32	fddiSMT 50
fddiSMTTimeStamp	G	M	10 33	fddiSMT 51
fddiSMTTransitionTimeStamp	G	M	10 34	fddiSMT 52
fddiSMTSetCount	G	O	10 35	fddiSMT 53
fddiSMTLastSetStationId	G	O	10 36	fddiSMT 54
Not Grouped				
fddiSMTVendorAttrib	GR	O	10 FF	fddiSMT 255

6.4.5.1.2 MAC attributes

fddiMACCapabilitiesGrp	G	MP	20 0A	fddiMAC 10
fddiMACFrameStatusFunctions	G	MP	20 0B	fddiMAC 11
fddiMACBridgeFunctions	G	OP	20 0C	fddiMAC 12
fddiMACT-MaxCapability	G	MP	20 0D	fddiMAC 13
fddiMACTVXCapability	G	MP	20 0E	fddiMAC 14
fddiMACConfigGrp	G	M	20 14	fddiMAC 20
fddiMACAvailablePaths	G	MP	20 16	fddiMAC 22
fddiMACCurrentPath	G	MP	20 17	fddiMAC 23
fddiMACUpstreamNbr	G	MP	20 18	fddiMAC 24
fddiMACDownstreamNbr	G	MP	20 19	fddiMAC 25
fddiMACOldUpstreamNbr	G	MP	20 1A	fddiMAC 26
fddiMACOldDownstreamNbr	G	MP	20 1B	fddiMAC 27
fddiMACDupAddressTest	G	MP	20 1D	fddiMAC 29
fddiMACRequestedPaths	GR	M	20 20	fddiMAC 32
fddiMACDownstreamPORTType	G	MP	20 21	fddiMAC 33
fddiMACIndex	G	M	20 22	fddiMAC 34
fddiMACAddressGrp	G	MP	20 28	fddiMAC 40
fddiMACSMTAddress	G	MP	20 29	fddiMAC 41
fddiMACLongGrpAddress	G	OP	20 2C	fddiMAC 44
fddiMACShortGrpAddress	G	OP	20 2D	fddiMAC 45
fddiMACOperationGrp	G	MP	20 32	fddiMAC 50
fddiMACT-Req	G	MP	20 33	fddiMAC 51
fddiMACT-Neg	G	MP	20 34	fddiMAC 52
fddiMACT-Max	G	MP	20 35	fddiMAC 53
fddiMACTvxValue	G	MP	20 36	fddiMAC 54
fddiMACT-Pri0	G	OP	20 38	fddiMAC 56
fddiMACT-Pri1	G	OP	20 39	fddiMAC 57
fddiMACT-Pri2	G	OP	20 3A	fddiMAC 58
fddiMACT-Pri3	G	OP	20 3B	fddiMAC 59
fddiMACT-Pri4	G	OP	20 3C	fddiMAC 60
fddiMACT-Pri5	G	OP	20 3D	fddiMAC 61
fddiMACT-Pri6	G	OP	20 3E	fddiMAC 62

fddiMACCountersGrp	G	MP	20 46	fddiMAC 70
fddiMACFrame-Ct	G	MP	20 47	fddiMAC 71
fddiMACCopied-Ct	G	MP	20 48	fddiMAC 72
fddiMACTransmit-Ct	G	MP	20 49	fddiMAC 73
fddiMACToken-Ct	G	OP	20 4A	fddiMAC 74
fddiMACError-Ct	G	MP	20 51	fddiMAC 81
fddiMACLost-Ct	G	MP	20 52	fddiMAC 82
fddiMACTvxExpired-Ct	G	OP	20 53	fddiMAC 83
fddiMACNotCopied-Ct	G	OP	20 54	fddiMAC 84
fddiMACLate-Ct	G	OP	20 55	fddiMAC 85
fddiMACRingOp-Ct	G	OP	20 56	fddiMAC 86
fddiMACFrameErrorConditionGrp	G	M	20 5A	fddiMAC 90
fddiMACFrameErrorThreshold	GR	M	20 5F	fddiMAC 95
fddiMACFrameErrorRatio	G	MP	20 60	fddiMAC 96
fddiMACNotCopiedConditionGrp	G	O	20 64	fddiMAC 100
fddiMACNotCopiedThreshold	GR	O	20 67	fddiMAC 103
fddiMACNotCopiedRatio	G	OP	20 69	fddiMAC 105
fddiMACStatusGrp	G	M	20 6E	fddiMAC 110
fddiMACRMTState	G	MP	20 6F	fddiMAC 111
fddiMACDA-Flag	G	MP	20 70	fddiMAC 112
fddiMACUNDA-Flag	G	MP	20 71	fddiMAC 113
fddiMACFrameErrorFlag	G	MP	20 72	fddiMAC 114
fddiMACNotCopiedFlag	G	OP	20 73	fddiMAC 115
fddiMACMA-UnitdataAvailable	G	MP	20 74	fddiMAC 116
fddiMACHardwarePresent	G	M	20 75	fddiMAC 117
fddiMACMA-UnitdataEnable	GR	M	20 76	fddiMAC 118
Not Grouped				
fddiMACVendorAttrib	GR	O	20 FF	fddiMAC 255
6.4.5.1.3 PATH attributes				
fddiPATHConfigGrp	G	M	32 0A	fddiPATH 10
fddiPATHIndex	G	M	32 0B	fddiPATH 11
fddiPATHRingLatency	GR	O	32 0D	fddiPATH 13
fddiPATHTraceStatus	G	O	32 0E	fddiPATH 14
fddiPATHSbaPayload	GR	O	32 0F	fddiPATH 15
fddiPATHSbaOverhead	GR	O	32 10	fddiPATH 16
fddiPATHConfiguration	G	M	32 12	fddiPATH 18
fddiPATHT-Rmode	GR	O	32 13	fddiPATH 19
fddiPATHSbaAvailable	GR	O	32 14	fddiPATH 20
fddiPATHTVXLowerBound	GR	M	32 15	fddiPATH 21
fddiPATHT-MaxLowerBound	GR	M	32 16	fddiPATH 22
fddiPATHMaxT-Req	GR	M	32 17	fddiPATH 23
Not Grouped				
fddiPATHVendorAttrib	GR	O	32 FF	fddiPATH 255
6.4.5.1.4 PORT attributes				
fddiPORTConfigGrp	G	M	40 0A	fddiPORT 10
fddiPORTMy-Type	G	M	40 0C	fddiPORT 12
fddiPORTNeighbourType	G	MP	40 0D	fddiPORT 13
fddiPORTConnectionPolicies	GR	M	40 0E	fddiPORT 14
fddiPORTMACIndicated	G	MP	40 0F	fddiPORT 15
fddiPORTCurrentPath	G	MP	40 10	fddiPORT 16
fddiPORTRequestedPaths	GR	M	40 11	fddiPORT 17
fddiPORTMACPlacement	G	MP	40 12	fddiPORT 18
fddiPORTAvailablePaths	G	MP	40 13	fddiPORT 19

fddiPORTMACLoop-Time	GR	O	40 15	fddiPORT 21
fddiPORTPMDCClass	G	MP	40 16	fddiPORT 22
fddiPORTConnectionCapabilities	G	MP	40 17	fddiPORT 23
fddiPORTIndex	G	M	40 1D	fddiPORT 29
fddiPORTOperationGrp	G	MP	40 1E	fddiPORT 30
fddiPORTMaint-LS	GR	O	40 1F	fddiPORT 31
fddiPORTBS-Flag	G	MP	40 21	fddiPORT 33
fddiPORTPC-LS	G	OP	40 22	fddiPORT 34
fddiPORTErrorsCtrsGrp	G	MP	40 28	fddiPORT 40
fddiPORTEBError-Ct	G	OP	40 29	fddiPORT 41
fddiPORTLCTFail-Ct	G	MP	40 2A	fddiPORT 42
fddiPORTLerGrp	G	M	40 32	fddiPORT 50
fddiPORTLer-Estimate	G	MP	40 33	fddiPORT 51
fddiPORTLem-Reject-Ct	G	MP	40 34	fddiPORT 52
fddiPORTLem-Ct	G	MP	40 35	fddiPORT 53
fddiPORTLer-Cutoff	GR	M	40 3A	fddiPORT 58
fddiPORTLer-Alarm	GR	M	40 3B	fddiPORT 59
fddiPORTStatusGrp	G	M	40 3C	fddiPORT 60
fddiPORTConnectState	G	MP	40 3D	fddiPORT 61
fddiPORTPCMState	G	MP	40 3E	fddiPORT 62
fddiPORTPC-Withhold	G	MP	40 3F	fddiPORT 63
fddiPORTLerFlag	G	MP	40 40	fddiPORT 64
fddiPORTHardwarePresent	G	M	40 41	fddiPORT 65
Not Grouped				
fddiPORTVendorAttrib	GR	O	40 FF	fddiPORT 255
6.4.5.1.5 Actions				
SMT Actions				
fddiSMTStationAction	S	M	10 3C	fddiSMT 60
fddiSMTVendorAction	S	O	10 FE	fddiSMT 254
MAC Actions				
fddiMACVendorAction	S	O	20 FE	fddiMAC 254
PATH Actions				
fddiPATHVendorAction	S	O	32 FE	fddiPATH 254
PORT Actions				
fddiPORTAction	S	M	40 46	fddiPORT 70
fddiPORTVendorAction	S	O	40 FE	fddiPORT 254
6.4.5.1.6 Notifications				
SMT Notifications				
fddiSMTHoldCondition	NA	O	10 47	fddiSMT 71
fddiSMTPeerWrapCondition	NA	M	10 48	fddiSMT 72
fddiSMTVendorNotification	NA	O	10 FC	fddiSMT 252
MAC Notifications				
fddiMACDuplicateAddressCondition	NA	M	20 8C	fddiMAC 140
fddiMACFrameErrorCondition	NA	M	20 8D	fddiMAC 141
fddiMACNotCopiedCondition	NA	O	20 8E	fddiMAC 142
fddiMACNeighbourChangeEvent	NA	M	20 8F	fddiMAC 143
fddiMACPathChangeEvent	NA	M	20 90	fddiMAC 144
fddiMACVendorNotification	NA	O	20 FC	fddiMAC 252
PATH Notifications				
fddiPATHVendorNotification	NA	O	32 FC	fddiPATH 252

PORT Notifications

fddiPORTLerCondition	NA	M	40 50	fddiPORT 80
fddiPORTUndesiredConnectionAttemptEvent	NA	M	40 51	fddiPORT 81
fddiPORTEBErrorCondition	NA	O	40 52	fddiPORT 82
fddiPORTPathChangeEvent	NA	M	40 53	fddiPORT 83
fddiPORTVendorNotification	NA	O	40 FC	fddiPORT 252

6.4.5.2 Managed object class templates

6.4.5.2.1 SMT object class

fddiSMT

MANAGED OBJECT CLASS

DERIVED FROM "Rec.X.71|ISO/IEC 10165-2":top;
CHARACTERIZED BY

fddiSMTBase

PACKAGE

BEHAVIOUR

fddiSMTBaseBhv BEHAVIOUR

DEFINED AS "The fddiSMT object class provides the support necessary at the station (node) level to manage the processes underway in the various FDDI layers such that a station (node) may work cooperatively as a part of an FDDI network. The fddiSMT object class provides services such as connection management, station insertion and removal, station initialization, configuration management, fault isolation and recovery, communication protocol for external authority, scheduling policies, and collection of statistics.";;

ATTRIBUTES

fddiSMTStationId	GET,
fddiSMTOpVersionId	GET,
fddiSMTHiVersionId	GET,
fddiSMTLoVersionId	GET,
fddiSMTUserData	GET-REPLACE,
fddiSMTMIBVersionId	GET,
fddiSMTMAC-Ct	GET,
fddiSMTNonMaster-Ct	GET,
fddiSMTMaster-Ct	GET,
fddiSMTAvailablePaths	GET,
fddiSMTConfigCapabilities	GET,
fddiSMTConfigPolicy	GET-REPLACE,
fddiSMTConnectionPolicy	GET-REPLACE,
fddiSMTT-Notify	GET-REPLACE,
fddiSMTStatRptPolicy	GET-REPLACE,
fddiSMTTrace-MaxExpiration	GET-REPLACE,
fddiSMTPORTIndexes	GET,
fddiSMTMACIndexes	GET,
fddiSMTBypassPresent	GET,
fddiSMTTECMState	GET,
fddiSMTCF-State	GET,
fddiSMTRemoteDisconnectFlag	GET,
fddiSMTStationStatus	GET,
fddiSMTPeerWrapFlag	GET,
fddiSMTTimeStamp	GET,
fddiSMTTransitionTimeStamp	GET;

ATTRIBUTE GROUPS

fddiSMTStationIdGrp,
fddiSMTStationConfigGrp,
fddiSMTStatusGrp,
fddiSMTMIBOperationGrp;

ACTIONS

fddiSMTStationAction;

NOTIFICATIONS

fddiSMTPeerWrapCondition;;;

CONDITIONAL PACKAGES

manufacturerdata	PACKAGE
BEHAVIOUR	
manufacturerdataBhv BEHAVIOUR	
DEFINED AS "This package provides support for the Manufacturer data attribute of a node.";;	
ATTRIBUTES	
fddiSMTManufacturerData	GET;
ATTRIBUTE GROUPS	
fddiSMTStationIdGrp	
fddiSMTManufacturerData;	
REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiSMTpkg(1) manufacturerdata(1)};	
PRESENT IF	"Manufacturer Data Supported",
parametermanagement	PACKAGE
BEHAVIOUR	
parametermanagementBhv BEHAVIOUR	
DEFINED AS "This package provides support for the parameter management capability of a node.";;	
ATTRIBUTES	
fddiSMTSetCount	GET,
fddiSMTLastSetStationId	GET;
ATTRIBUTE GROUPS	
fddiSMTMIBOperationGrp	
fddiSMTSetCount	
fddiSMTLastSetStationId;	
REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiSMTpkg(1) parametermanagement(2)};	
PRESENT IF	"Parameter Management Frame Set operations are supported",
hold	PACKAGE
BEHAVIOUR	
holdBhv BEHAVIOUR	
DEFINED AS "This package provides support for the Hold policy. The Hold policy only applies to dual attach, dual MAC nodes.";;	
ATTRIBUTES	
fddiSMTHoldState	GET;
ATTRIBUTE GROUPS	
fddiSMTStatusGrp	
fddiSMTHoldState;	
NOTIFICATIONS	
fddiSMTHoldCondition;	
REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiSMTpkg(1) hold(3)};	
PRESENT IF	"Hold policy supported",
smtvendorspecific	PACKAGE
BEHAVIOUR	
smtvendorspecificBhv BEHAVIOUR	
DEFINED AS "This package provides support for the extension of the MIB for the use of specific vendors.";;	
ATTRIBUTES	
fddiSMTVendorAttrib	GET-REPLACE;
ACTIONS	
fddiSMTVendorAction;	
NOTIFICATIONS	
fddiSMTVendorNotification;	

REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiSMTpkg(1)
smtvendorspecific(4)};
PRESENT IF "Vendor Extensions Supported";
REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiSMT(16)};

6.4.5.2.2 MAC object class

fddiMAC

DERIVED FROM "Rec.X.71|ISO/IEC 10165-2":top;
CHARACTERIZED BY

MANAGED OBJECT CLASS

fddiMACBase

BEHAVIOUR

fddiMACBaseBhv BEHAVIOUR

DEFINED AS "The fddiMAC object provides support for the access control of the medium and generation and verification of frame check sequences to assure the proper delivery of valid data to the higher layers. It also concerns itself with the generation and recognition of device addresses and the peer-to-peer associations within the FDDI network.";;

ATTRIBUTES

fddiMACIndex	GET,
fddiMACRequestedPaths	GET-REPLACE,
fddiMACFrameErrorThreshold	GET-REPLACE,
fddiMACHardwarePresent	GET,
fddiMACMA-UnitdataEnable	GET-REPLACE;

ATTRIBUTE GROUPS

fddiMACConfigGrp,
fddiMACFrameErrorConditionGrp,
fddiMACStatusGrp;

NOTIFICATIONS

fddiMACDuplicateAddressCondition,
fddiMACFrameErrorCondition,
fddiMACNeighbourChangeEvent,
fddiMACPathChangeEvent;;;

CONDITIONAL PACKAGES

machardwarerepresent

BEHAVIOUR

machardwarerepresentBhv BEHAVIOUR

DEFINED AS "This package contains attributes which support the management of the MAC entity within a station. This package shall be included in every MAC object where the underlying hardware for the MAC entity is present.";;

ATTRIBUTES

fddiMACFrameStatusFunctions	GET,
fddiMACT-MaxCapability	GET,
fddiMACTVXCapability	GET,
fddiMACAvailablePaths	GET,
fddiMACCurrentPath	GET,
fddiMACUpstreamNbr	GET,
fddiMACDownstreamNbr	GET,
fddiMACOldUpstreamNbr	GET,
fddiMACOldDownstreamNbr	GET,
fddiMACDupAddressTest	GET,
fddiMACDownstreamPORTType	GET,
fddiMACSMTAddress	GET,
fddiMACT-Req	GET,
fddiMACT-Neg	GET,
fddiMACT-Max	GET,
fddiMACTvxValue	GET,

fddiMACFrame-Ct	GET,
fddiMACCopied-Ct	GET,
fddiMACTransmit-Ct	GET,
fddiMACError-Ct	GET,
fddiMACLost-Ct	GET,
fddiMACFrameErrorRatio	GET,
fddiMACRMTState	GET,
fddiMACDA-Flag	GET,
fddiMACUNDA-Flag	GET,
fddiMACFrameErrorFlag	GET,
fddiMACMA-UnitdataAvailable	GET;

ATTRIBUTE GROUPS

```

fddiMACCapabilitiesGrp
  fddiMACFrameStatusFunctions
  fddiMACT-MaxCapability
  fddiMACTVXCapability,
fddiMACConfigGrp
  fddiMACAvailablePaths
  fddiMACCurrentPath
  fddiMACUpstreamNbr
  fddiMACDownstreamNbr
  fddiMACOldUpstreamNbr
  fddiMACOldDownstreamNbr
  fddiMACDupAddressTest
  fddiMACDownstreamPORTType,
fddiMACAddressGrp
  fddiMACSMTAddress,
fddiMACOperationGrp
  fddiMACT-Req
  fddiMACT-Neg
  fddiMACT-Max
  fddiMACTvxValue,
fddiMACCountersGrp
  fddiMACFrame-Ct
  fddiMACCopied-Ct
  fddiMACTransmit-Ct
  fddiMACError-Ct
  fddiMACLost-Ct,
fddiMACFrameErrorConditionGrp
  fddiMACFrameErrorRatio,
fddiMACStatusGrp
  fddiMACRMTState
  fddiMACDA-Flag
  fddiMACUNDA-Flag
  fddiMACFrameErrorFlag
  fddiMACMA-UnitdataAvailable;

```

REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiMACpkg(2)
machardwarerepresent(1)};

PRESENT IF

"Underlying hardware support of the MAC
object is present.",

framenotcopied**PACKAGE**

BEHAVIOUR

framenotcopiedBhv BEHAVIOUR

DEFINED AS "This package provides support for the detection and reporting of
the Frame Not Copied condition of the fddiMAC object.";

ATTRIBUTES
 fddiMACNotCopied-Ct GET,
 fddiMACNotCopiedRatio GET,
 fddiMACNotCopiedFlag GET;
ATTRIBUTE GROUPS
 fddiMACNotCopiedConditionGrp
 fddiMACNotCopiedRatio,
 fddiMACCountersGrp
 fddiMACNotCopied-Ct,
 fddiMACStatusGrp
 fddiMACNotCopiedFlag;
NOTIFICATIONS
 fddiMACNotCopiedCondition;
REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiMACpkg(2)
 framenotcopied(2)};
PRESENT IF "Frame not copied is supported AND
underlying hardware support is present.",

framenotcopiedmanagement **PACKAGE**
BEHAVIOUR
 framenotcopiedmanagementBhv BEHAVIOUR
 DEFINED AS "This package provides support for the detection and reporting of
the Frame Not Copied condition of the fddiMAC object.";;
ATTRIBUTES
 fddiMACNotCopiedThreshold GET-REPLACE;
ATTRIBUTE GROUPS
 fddiMACNotCopiedConditionGrp
 fddiMACNotCopiedThreshold;
REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiMACpkg(2)
 framenotcopiedmanagement(3)};
PRESENT IF "Frame not copied is supported.",

prioritythreshold **PACKAGE**
BEHAVIOUR
 prioritythresholdBhv BEHAVIOUR
 DEFINED AS "This package provides support for the priority thresholding
capabilityof the fddiMAC object.";;
ATTRIBUTES
 fddiMACT-Pri0 GET,
 fddiMACT-Pri1 GET,
 fddiMACT-Pri2 GET,
 fddiMACT-Pri3 GET,
 fddiMACT-Pri4 GET,
 fddiMACT-Pri5 GET,
 fddiMACT-Pri6 GET;
ATTRIBUTE GROUPS
 fddiMACOperationGrp
 fddiMACT-Pri0
 fddiMACT-Pri1
 fddiMACT-Pri2
 fddiMACT-Pri3
 fddiMACT-Pri4
 fddiMACT-Pri5
 fddiMACT-Pri6;
REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiMACpkg(2)
 prioritythreshold(4)};
PRESENT IF "Priority Thresholding supported AND
underlying hardware support is present",

tvxexpired	PACKAGE
BEHAVIOUR	
tvxexpiredBhv BEHAVIOUR	
DEFINED AS "This package provides support for monitoring the expiration of TVX.";;	
ATTRIBUTES	
fddiMACTvxExpired-Ct	GET;
ATTRIBUTE GROUPS	
fddiMACCountersGrp	
fddiMACTvxExpired-Ct;	
REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiMACpkg(2) tvxexpired(5)};	
PRESENT IF	"TVX Expiration monitored AND underlying hardware support is present",
latecount	PACKAGE
BEHAVIOUR	
latecountBhv BEHAVIOUR	
DEFINED AS "This package provides support for monitoring the number of TRT expirations since MAC was reset or a token was received.";;	
ATTRIBUTES	
fddiMACLate-Ct	GET;
ATTRIBUTE GROUPS	
fddiMACCountersGrp	
fddiMACLate-Ct;	
REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiMACpkg(2) latecount(6)};	
PRESENT IF	"Late Counter monitored AND underlying hardware support is present",
ringopcount	PACKAGE
BEHAVIOUR	
ringopcountBhv BEHAVIOUR	
DEFINED AS "This package provides support for monitoring the number of times the ring has entered the 'Ring_Operational' state from the 'Ring Not Operational' state.";;	
ATTRIBUTES	
fddiMACRingOp-Ct	GET;
ATTRIBUTE GROUPS	
fddiMACCountersGrp	
fddiMACRingOp-Ct;	
REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiMACpkg(2) ringopcount(7)};	
PRESENT IF	"RingOp Counter monitored AND underlying hardware support is present",
longgrpaddress	PACKAGE
BEHAVIOUR	
longgrpaddressBhv BEHAVIOUR	
DEFINED AS "This package provides support for 48 bit group addresses for the MAC.";;	
ATTRIBUTES	
fddiMACLongGrpAddress	GET;
ATTRIBUTE GROUPS	
fddiMACAddressGrp	
fddiMACLongGrpAddress;	
REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiMACpkg(2) longgrpaddress(8)};	
PRESENT IF	"Long Group Addresses supported AND underlying hardware support is present.",

shortgrpaddress **PACKAGE**
 BEHAVIOUR
 shortgrpaddressBhv BEHAVIOUR
 DEFINED AS "This package provides support for 16 bit group addresses for the
 MAC.";;
 ATTRIBUTES
 fddiMACShortGrpAddress GET;
 ATTRIBUTE GROUPS
 fddiMACAddressGrp
 fddiMACShortGrpAddress;
 REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiMACpkg(2)
 shortgrpaddress(9)};
 PRESENT IF "Short Group Addresses supported AND
 underlying hardware support is present.",

tokencounter **PACKAGE**
 BEHAVIOUR
 tokencounterBhv BEHAVIOUR
 DEFINED AS "This package provides support for counting the tokens received
 by a station.";;
 ATTRIBUTES
 fddiMACToken-Ct GET;
 ATTRIBUTE GROUPS
 fddiMACCountersGrp
 fddiMACToken-Ct;
 REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiMACpkg(2)
 tokencounter(10)};
 PRESENT IF "Token Counting supported AND underlying
 hardware support is present",

bridgefunctions **PACKAGE**
 BEHAVIOUR
 bridgefunctionsBhv BEHAVIOUR
 DEFINED AS "This package provides support for a MAC performing bridge
 functions.";;
 ATTRIBUTES
 fddiMACBridgeFunctions GET;
 ATTRIBUTE GROUPS
 fddiMACCapabilitiesGrp
 fddiMACBridgeFunctions;
 REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiMACpkg(2)
 bridgefunctions(11)};
 PRESENT IF "Bridge Functions supported AND underlying
 hardware support is present",

macvendorspecific **PACKAGE**
 BEHAVIOUR
 macvendorspecificBhv BEHAVIOUR
 DEFINED AS "This package provides support for the extension of the MIB for
 the use of specific vendors.";;
 ATTRIBUTES
 fddiMACVendorAttrib GET-REPLACE;
 ACTIONS
 fddiMACVendorAction;
 NOTIFICATIONS
 fddiMACVendorNotification;
 REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiMACpkg(2)
 macvendorspecific(12)};
 PRESENT IF "Vendor Extensions supported";;

;
 REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiMAC(32)};

6.4.5.2.3 PATH object class**fddiPATH****MANAGED OBJECT CLASS**

DERIVED FROM "Rec.X.71|ISO/IEC 10165-2":top;
CHARACTERIZED BY

fddiPATHBase**PACKAGE****BEHAVIOUR****fddiPATHBaseBhv BEHAVIOUR**

DEFINED AS "The fddiPATH object provides for the management of the Path entity. A station may have multiple Paths. Each instance of this class models the configuration of a station's internal resources (e.g. MACs and PORTs) as they relate to an external ring (see 9.7.2).";

ATTRIBUTES

fddiPATHIndex	GET,
fddiPATHConfiguration	GET,
fddiPATHTVXLowerBound	GET-REPLACE,
fddiPATHT-MaxLowerBound	GET-REPLACE,
fddiPATHMaxT-Req	GET-REPLACE;

ATTRIBUTE GROUPS

fddiPATHConfigGrp;;

CONDITIONAL PACKAGES**synchbandwidthrequestor****PACKAGE****BEHAVIOUR****synchbandwidthrequestorBhv BEHAVIOUR**

DEFINED AS "This package provides support for the requisition of synchronous bandwidth.";

ATTRIBUTES

fddiPATHSbaPayload	GET-REPLACE,
fddiPATHSbaOverhead	GET-REPLACE;

ATTRIBUTE GROUPS

fddiPATHConfigGrp
fddiPATHSbaPayload
fddiPATHSbaOverhead;

REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiPATHpkg(3)
synchbandwidthrequestor(1)};

PRESENT IF

"Synchronous service supported",

synchbandwidthmgt**PACKAGE****BEHAVIOUR****synchbandwidthmgtBhv BEHAVIOUR**

DEFINED AS "This package provides support for the allocation of synchronous bandwidth.";

ATTRIBUTES

fddiPATHSbaAvailable	GET-REPLACE;
----------------------	--------------

ATTRIBUTE GROUPS

fddiPATHConfigGrp
fddiPATHSbaAvailable;

REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiPATHpkg(3)
synchbandwidthmgt(2)};

PRESENT IF

"Allocating Synchronous bandwidth",

ringlatency**PACKAGE****BEHAVIOUR****ringlatencyBhv BEHAVIOUR**

DEFINED AS "This package provides support for measuring the total accumulated latency of the ring associated with this path.";

ATTRIBUTES

fddiPATHRingLatency	GET-REPLACE;
---------------------	--------------

ATTRIBUTE GROUPS
 fddiPATHConfigGrp
 fddiPATHRingLatency;
REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiPATHpkg(3)
 ringlatency(3)};

PRESENT IF "Latency measurement supported",

tracestatus **PACKAGE**
 BEHAVIOUR
 tracestatusBhv BEHAVIOUR
 DEFINED AS "This package provides status from the trace function.";;
 ATTRIBUTES
 fddiPATHTraceStatus GET;
 ATTRIBUTE GROUPS
 fddiPATHConfigGrp
 fddiPATHTraceStatus;
REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiPATHpkg(3)
 tracestatus(4)};

PRESENT IF "Trace Status is supported",

restrictedtoken **PACKAGE**
 BEHAVIOUR
 restrictedtokenBhv BEHAVIOUR
 DEFINED AS "This package along with the RMT optional protocol provides
 support for monitoring the restricted token service.";;
 ATTRIBUTES
 fddiPATHT-Rmode GET-REPLACE;
 ATTRIBUTE GROUPS
 fddiPATHConfigGrp
 fddiPATHT-Rmode;
REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiPATHpkg(3)
 restrictedtoken(5)};

PRESENT IF "Restricted Token Monitoring is supported",

pathvendorspecific **PACKAGE**
 BEHAVIOUR
 pathvendorspecificBhv BEHAVIOUR
 DEFINED AS "This package provides support for the extension of the MIB for
 the use of specific vendors.";;
 ATTRIBUTES
 fddiPATHVendorAttrib GET-REPLACE;
 ACTIONS
 fddiPATHVendorAction;
 NOTIFICATIONS
 fddiPATHVendorNotification;
REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiPATHpkg(3)
 pathvendorspecific(6)};

PRESENT IF "Vendor Extensions Supported";;

;

REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiPATH(50)};

6.4.5.2.4 PORT object class

fddiPORT **MANAGED OBJECT CLASS**
 DERIVED FROM "Rec.X.71|ISO/IEC 10165-2":top;
 CHARACTERIZED BY

fddiPORTBase	PACKAGE
BEHAVIOUR	
fddiPORTBaseBhv BEHAVIOUR	
DEFINED AS "The fddiPORT object provides support for the medium, connector, bypassing, and driver/receiver requirements. It also defines encode/decode and clock requirements as required for framing the data for transmission on the medium by the higher layers of the FDDI.";	
ATTRIBUTES	
fddiPORTMy-Type	GET,
fddiPORTConnectionPolicies	GET-REPLACE,
fddiPORTRequestedPaths	GET-REPLACE,
fddiPORTIndex	GET,
fddiPORTLer-Cutoff	GET-REPLACE,
fddiPORTLer-Alarm	GET-REPLACE,
fddiPORTHardwarePresent	GET;
ATTRIBUTE GROUPS	
fddiPORTConfigGrp,	
fddiPORTLerGrp,	
fddiPORTStatusGrp;	
ACTIONS	
fddiPORTAction;	
NOTIFICATIONS	
fddiPORTLerCondition,	
fddiPORTUndesiredConnectionAttemptEvent,	
fddiPORTPathChangeEvent;;;	

CONDITIONAL PACKAGES

porthardwarepresent	PACKAGE
BEHAVIOUR	
porthardwarepresentBhv BEHAVIOUR	
DEFINED AS "This package contains attributes which support the management of the PORT entity within a station. This package shall be included in every PORT object where the underlying hardware for the PORT entity is present.";	
ATTRIBUTES	
fddiPORTNeighbourType	GET,
fddiPORTMACIndicated	GET,
fddiPORTCurrentPath	GET,
fddiPORTMACPlacement	GET,
fddiPORTAvailablePaths	GET,
fddiPORTPMDClass	GET,
fddiPORTConnectionCapabilities	GET,
fddiPORTBS-Flag	GET,
fddiPORTLCTFail-Ct	GET,
fddiPORTLer-Estimate	GET,
fddiPORTLem-Reject-Ct	GET,
fddiPORTLem-Ct	GET,
fddiPORTConnectState	GET,
fddiPORTPCMState	GET,
fddiPORTPC-Withhold	GET,
fddiPORTLerFlag	GET;
ATTRIBUTE GROUPS	
fddiPORTConfigGrp	
fddiPORTNeighbourType	
fddiPORTMACIndicated	
fddiPORTCurrentPath	
fddiPORTMACPlacement	
fddiPORTAvailablePaths	
fddiPORTPMDClass	
fddiPORTConnectionCapabilities,	

```

fddiPORTOperationGrp
  fddiPORTBS-Flag,
fddiPORTErrorsCtrsGrp
  fddiPORTLCTFail-Ct,
fddiPORTLerGrp
  fddiPORTLer-Estimate
  fddiPORTLem-Reject-Ct
  fddiPORTLem-Ct,
fddiPORTStatusGrp
  fddiPORTConnectState
  fddiPORTPCMState
  fddiPORTPC-Withhold
  fddiPORTLerFlag;
REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiPORTpkg(4)
  porthardwarepresent(1)};
PRESENT IF
    "Underlying hardware support of the port
    object is present.",

eberrorcounter                                PACKAGE
  BEHAVIOUR
    eberrorcounterBhv BEHAVIOUR
      DEFINED AS "This package provides support for counting elasticity buffer errors
      in the fddiPORT object.";;
  ATTRIBUTES
    fddiPORTEBError-Ct                                GET;
  ATTRIBUTE GROUPS
    fddiPORTErrorsCtrsGrp
    fddiPORTEBError-Ct;
  NOTIFICATIONS
    fddiPORTEBErrorCondition;
  REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiPORTpkg(4)
    eberrorcounter(2)};
PRESENT IF
    "EB Error Tracking supported AND
    underlying hardware support is present.",

maclocalloop                                PACKAGE
  BEHAVIOUR
    maclocalloopBhv BEHAVIOUR
      DEFINED AS "This package provides support for the time for the optional MAC
      Local Loop.";;
  ATTRIBUTES
    fddiPORTMACLoop-Time                                GET-REPLACE;
  ATTRIBUTE GROUPS
    fddiPORTConfigGrp
    fddiPORTMACLoop-Time;
  REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiPORTpkg(4)
    maclocalloop(3)};
PRESENT IF
    "MAC Local Loop supported",

maint-LS                                    PACKAGE
  BEHAVIOUR
    maint-LSBhv BEHAVIOUR
      DEFINED AS "This package provides support for identifying the state of the
      line.";;
  ATTRIBUTES
    fddiPORTPC-LS                                GET;
  ATTRIBUTE GROUPS
    fddiPORTOperationGrp
    fddiPORTPC-LS;
  REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiPORTpkg(4)
    maint-LS(4)};

```

PRESENT IF "Maint_LS is supported AND underlying hardware support is present.",

maint-LSmanagement

PACKAGE

BEHAVIOUR

maint-LSmanagementBhv BEHAVIOUR

DEFINED AS "This package provides support for identifying the state of the line.";;

ATTRIBUTES

fddiPORTMaint-LS

GET-REPLACE;

ATTRIBUTE GROUPS

fddiPORTOperationGrp

fddiPORTMaint-LS;

REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiPORTpkg(4) maint-LSmanagement(5)};

PRESENT IF

"Maint_LS is supported.",

portvendorspecific

PACKAGE

BEHAVIOUR

portvendorspecific Bhv BEHAVIOUR

DEFINED AS "This package provides support for the extension of the MIB for the use of specific vendors.";;

ATTRIBUTES

fddiPORTVendorAttrib

GET-REPLACE;

ACTIONS

fddiPORTVendorAction;

NOTIFICATIONS

fddiPORTVendorNotification;

REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiPORTpkg(4) portvendorspecific(6)};

PRESENT IF

"Vendor Extensions Supported";;

; REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiPORT(64)};

6.4.5.3 Attribute group templates

6.4.5.3.1 SMT attribute groups

fddiSMTStationIdGrp

ATTRIBUTE GROUP

GROUP ELEMENTS

fddiSMTStationId,

fddiSMTOpVersionId,

fddiSMTHiVersionId,

fddiSMTLoVersionId,

fddiSMTUserData,

fddiSMTMIBVersionId;

DESCRIPTION

The identification group includes the attributes that identify the station for its operation in FDDI.;

REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiSMT(16) fddiSMTStationIdGrp(10)};

fddiSMTStationConfigGrp

ATTRIBUTE GROUP

GROUP ELEMENTS

fddiSMTMAC-Ct,

fddiSMTNonMaster-Ct,

fddiSMTMaster-Ct,

fddiSMTAvailablePaths,

fddiSMTConfigCapabilities,

fddiSMTConfigPolicy,

fddiSMTConnectionPolicy,

fddiSMTT-Notify,
fddiSMTStatRptPolicy,
fddiSMTTrace-MaxExpiration,
fddiSMTPORTIndexes,
fddiSMTMACIndexes,
fddiSMTBypassPresent;

DESCRIPTION

The configuration group includes the attributes that characterize the station's configuration.;

REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiSMT(16)
fddiSMTStationConfigGrp(20)};

fddiSMTStatusGrp

ATTRIBUTE GROUP

GROUP ELEMENTS

fddiSMTECMState,
fddiSMTCF-State,
fddiSMTRemoteDisconnectFlag,
fddiSMTStationStatus,
fddiSMTPeerWrapFlag;

DESCRIPTION

The status group includes attributes that define the general station state.;

REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiSMT(16) fddiSMTStatusGrp(40)};

fddiSMTMIBOperationGrp

ATTRIBUTE GROUP

GROUP ELEMENTS

fddiSMTTimeStamp,
fddiSMTTransitionTimeStamp;

DESCRIPTION

The MIB operation group includes the attributes that are associated with MIB accesses. The last three attributes in the group, the TransitionTimeStamp, SetCount, and LastSetStationId are used in the Parameter Management Frame (PMF) protocol (see 8.4).;

REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiSMT(16)
fddiSMTMIBOperationGrp(50)};

6.4.5.3.2 MAC attribute groups

fddiMACCapabilitiesGrp

ATTRIBUTE GROUP

DESCRIPTION

The MAC capabilities group includes attributes that describe the frame status handling, bridge, and timer bounds capabilities.;

REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiMAC(32)
fddiMACCapabilitiesGrp(10)};

fddiMACConfigGrp

ATTRIBUTE GROUP

GROUP ELEMENTS

fddiMACIndex;

DESCRIPTION

The MAC configuration group includes the Path configuration attributes as well as neighbour attributes that give a topological characterization of the MAC in the station and in the network.;

REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiMAC(32) fddiMACConfigGrp(20)};

fddiMACAddressGrp

ATTRIBUTE GROUP

DESCRIPTION

The address group includes all of the local address attributes for the MAC.;

REGISTERED AS {iso(1) standard(0) iso9314(9314) fddiMIB(1) fddiMAC(32)
fddiMACAddressGrp(40)};