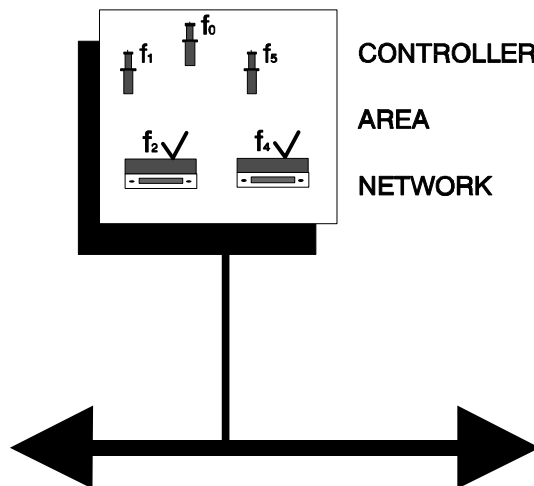


**CAN in Automation (CiA)
International Users and Manufacturers Group e.V.**



**CAN Application Layer for Industrial Applications
CiA/DS203-1
February 1996**

NMT Service Specification

1. SCOPE

This document contains the Network Management Service Specification. This document is part of a set of documents that standardize the CAN Application Layer for Industrial Applications.

2. REFERENCES

/1/: CiA/DS201, CAN Reference Model

/2/: CiA/DS203-2, NMT Protocol Specification

/3/: CiA/DS207, Application Layer Naming Conventions

3. GENERAL DESCRIPTION

3.1 NMT PERSPECTIVE

NMT is one of the application layer entities in the CAN Reference Model, see /1/.

The NMT aids in the development of distributed applications. Due to the fact that an application is distributed, certain events have to be handled (e.g failures of parts of the applicaton) that would not occur if the same application had not been distributed.

The application has to deal with these network management aspects, although they have nothing to do with the goal of the application (e.g controlling a process). These aspects are the consequence of building a distributed application and must be compared to the advantages of building a distributed application.

3.2 NMT Objects and Services

A CAN network consists of modules that are connected by one physical CAN bus. The NMT uses three objects to model a CAN network:

- **the network object.** The network object represents the set of all modules in a CAN network. A network can contain at most 255 modules. The network object may exist on one module only, called the NMT Master.

NMT Service Specification

- **the remote node object.** Each module in the network that is managed by the NMT services is represented by a remote node object on the NMT Master.
- **the node object.** Each module that is managed by the NMT services is represented by a node object on that module (including the NMT Master). A module where a node object exists is called an NMT Slave.

Each NMT Slave and its node object is uniquely identified in the network by its NMT Address. The syntax of an NMT Address is defined in /3/. The NMT Address of an NMT Slave cannot be changed by the NMT services but can be configured via the LMT Service Element (see /1/) or in an application specific way that does not fall within the scope of the CiA Standard on the CAN Application Layer for Industrial Applications.

For each NMT Slave there must exist one remote node object with the same NMT Address on the NMT Master. A node object and the remote node object that have the same NMT Address are called peers. Each remote node object communicates with its peer via the NMT Protocol as defined in /2/. The NMT Protocol uses a Node-ID to address an NMT Slave. The syntax of a Node-ID is defined in /3/. A unique Node-ID is assigned to the node object of each NMT Slave and its peer by the NMT Master. Peers have the same Node-ID. The NMT model of a CAN network is depicted in fig. 1. Note that it is possible that a module is an NMT Master and an NMT Slave at the same time.

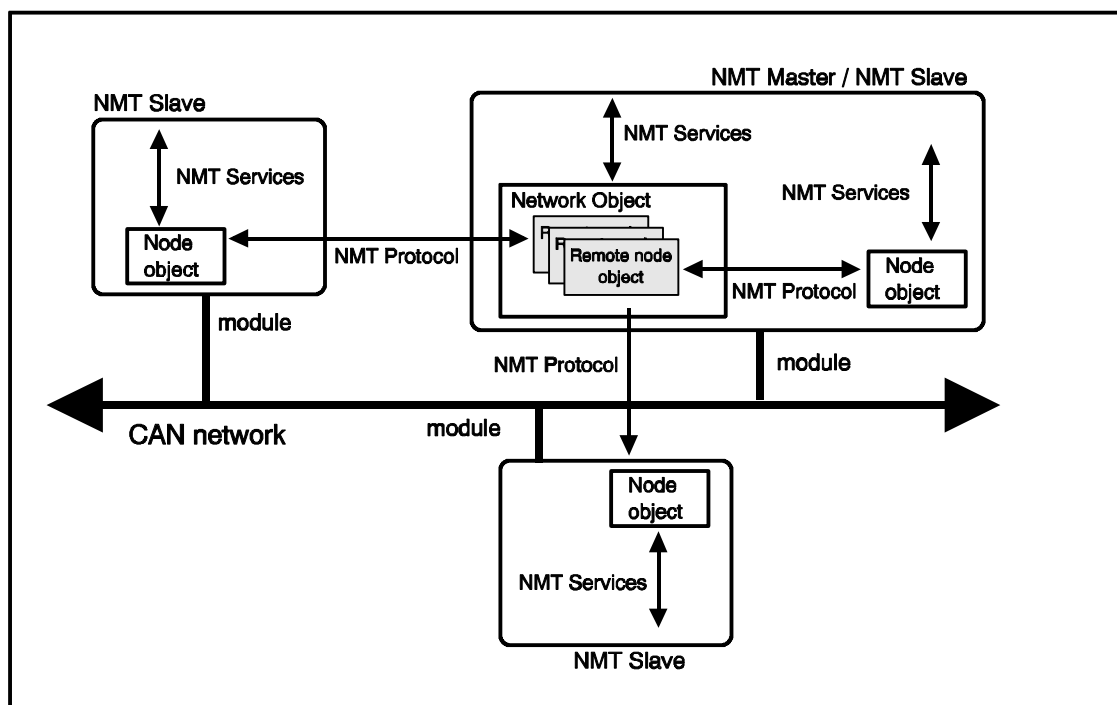


Fig. 1: The NMT Model

The NMT offers the following services:

- **Module Control Services:** through these services, the NMT Master initializes NMT Slaves that want to take part in the distributed application and allows them to communicate with each other through the CMS Service Element (see /1/). To this purpose, an NMT Slave has to define all the CMS objects it needs. The COB's required by the CMS protocol for these CMS objects have to obtain COB identifiers and inhibit-times. These can be assigned statically by the application or dynamically by the Distributor Service Element (see /1/). Once the COB identifiers have been obtained, the NMT Master can indicate the NMT Slave that it may or may not access the network through the CMS Service Element. Through the Module Control Services the NMT Master controls the sequence of these actions for each NMT Slave. Through the Module Control services, the NMT Master and NMT Slave also negotiate about parameters for the NMT Protocol.
- **Error Control Services:** through these services, the NMT detects failures in a CAN network. Local failures are caused by errors detected in the Data Link or Physical Layer (see /1/) of a module or by other application specific conditions on that module that prevent it from taking part in the distributed application. These conditions do not fall within the scope of the CiA Standard on the CAN Application Layer for Industrial Applications. Remote failures are failures detected by the Node Guarding Protocol (see section 6 of this document).
- **Configuration Control Services:** through these services, the NMT can up and download configuration data from respectively to a module in the CAN network. The meaning of the configuration data that is up- or downloaded is application specific and does not fall within the scope of the CiA Standard on the CAN Application Layer for Industrial Applications. It may be executable code, parameters, data or anything else.

3.3 NMT Capabilities

NMT capabilities indicate categories of network management functionality that may or may not be present in the network. Capabilities that affect all modules in a network are called network capabilities and can only be configured on the NMT Master. Capabilities that affect only one module in the network are called node capabilities and can only be configured on an NMT Slave. The following capabilities are defined:

- Network Management capability. This network capability implements the mandatory module control services on the NMT Master. These services can only be executed with NMT Slaves that have the:

NMT Service Specification

- **Node Management capability.** This node capability implements the mandatory module control services on an NMT Slave. These services can only be executed if the NMT Master has the Network Management capability.
- **Network Error capability.** This network capability implements the mandatory error control services on the NMT Master. These services can only be executed with NMT Slaves that have the:
- **Node Error capability.** This node capability implements the mandatory error control services on an NMT slave. These services can only be executed if the NMT Master has the Network Error capability
- **Network Configuration capability.** This network capability implements the mandatory configuration control services on the NMT Master. These services can only be executed with NMT Slaves that have the:
- **Node Configuration capability.** This node capability implements the mandatory configuration control services on an NMT Slave. These services can only be executed if the NMT Master has the Network Configuration capability

How to configure NMT capabilities on an NMT Master and NMT Slave does not fall within the scope of the CiA Standard on the CAN Application Layer for Industrial Applications.

3.4 Network Classes

The network class indicates the network capabilities that have been configured on the NMT Master:

- **Class 0:** no Network Management capability. As a consequence, module control, error control, configuration control, and dynamic identifier/inhibit time distribution is not possible for any of the modules in the network.
- **Class 1:** Network Management capability, no Network Error capability, no Network Configuration capability. This is a network where error control and configuration control is not possible for any of the modules in the network. Module control and dynamic identifier/inhibit-time distribution is possible.
- **Class 2:** Network Management capability, Network Error capability, no Network Configuration capability. This is a network where module control, error control, and dynamic identifier/inhibit-time distribution is possible. Configuration control is not possible for any of the modules in the network.

- **Class 3:** Network Management capability, no Network Error capability, Network Configuration capability. This is a network where module control, configuration control, and dynamic identifier/inhibit-time distribution is possible. Error control is not possible for any of the modules in the network.
- **Class 4:** Network Management capability, Network Error capability, Network Configuration capability. This is a network where module control, error control, configuration control and dynamic identifier/inhibit-time distribution is possible.

3.5 Node Classes

The node class indicates the node capabilities that have been configured on an NMT Slave:

- **Class 0:** no Node Management capability. This is a module that cannot be managed through the module control services of the NMT Master. As a consequence, error control, configuration control, and dynamic identifier/inhibit-time distribution is not possible for this module.
- **Class 1:** Node Management capability, no Node Error capability, no Node Configuration capability. This is a module that can be managed through the module control services of the NMT Master and for which dynamic identifier/inhibit-time distribution is possible. Error control and configuration control is not possible for this module.
- **Class 2:** Node Management capability, Node Error capability, no Node Configuration capability. This is a module that can be managed through the module and error control services of the NMT Master and for which dynamic identifier/inhibit-time distribution is possible. Configuration control is not possible for this module.
- **Class 3:** Node Management capability, no Node Error capability, Node Configuration capability. This is a module that can be managed through the module and configuration control services of the NMT Master and for which dynamic identifier/inhibit-time distribution is possible. Error control is not possible for this module.
- **Class 4:** Node Management capability, Node Error capability, Node Configuration capability. This is a module that can be managed through the module-, error-, and configuration control services of the NMT Master and for which dynamic identifier/inhibit-time distribution is possible.

3.6 NMT Service Descriptions

The NMT services are described in a tabular form that contains the parameters of each service primitive that is defined for that service. The primitives that are defined for a particular service determine the service type (e.g unconfirmed, confirmed, etc.). How to interpret the tabular form and what service types exist is defined in /1/. In the service descriptions, [a, b] denotes the range of integers from a to b with a and b included. If a > b, the range is empty.

4. NMT OBJECTS

4.1 Network Object

Network Attributes:

- **remote node set:** the set of remote node objects that form the network
- **class:** the network class. A value in the range [0, 4].

4.2 Remote Node Object

Remote Node Attributes

- **NMT Address:** see /3/. This attribute uniquely identifies the remote node object in the remote node set.
- **state:** one of the values {DISCONNECTED, CONNECTED, PREPARED, OPERATIONAL}. This attribute indicates the state of the remote node object. The state is controlled by the NMT services according to the state diagrams in section 6 of this document.
- **Node-ID:** a value in the range [1, 255]. This attribute uniquely identifies the remote node object in the remote node set of the network object if and only if the state of the remote node object is not DISCONNECTED. It is identical to the Node-ID attribute of its peer.

4.3 Node Object

Node Attributes

- **NMT Address:** see /3/. This attribute uniquely identifies the NMT Slave and its node object in the network.
- **state:** one of the values {DISCONNECTED, CONNECTING, PREPARING, PREPARED, OPERATIONAL}. This attribute indicates the state of the NMT

Slave and its node object. The state is controlled by the NMT services according to the state diagrams in the section 6 of this document.

- **Node-ID:** a value in the range [1, 255]. This attribute uniquely identifies the NMT Slave and its node object in the network if and only if the state of the node object is neither DISCONNECTED nor CONNECTING. It is identical to the Node-ID attribute of its peer.
- **class:** the node class. A value in the range [0, 4].

In the remainder of this document, the attributes of a node object of an NMT Slave are also considered to be attributes of the NMT Slave. E.g the state of an NMT Slave denotes the state of its node object.

5. NMT SERVICES

There can be at most one confirmed NMT service outstanding in the complete network.

5.1 Module Control Services

The mandatory module control services need to be implemented on the NMT Master if and only if the Network Management capability has been configured on the NMT Master. The mandatory module control services need to be implemented on an NMT Slave if and only if the Node Management capability has been configured on that NMT Slave.

Create Network

<i>Parameter</i>	<i>Request</i>
Argument class	Mandatory mandatory

Through this service the NMT Master creates a network object with the requested attributes. The service will only be executed if no network object exists. After completion of the service, the remote node set will be empty. The service is local and mandatory.

Add Remote Node

<i>Parameter</i>	<i>Request</i>
Argument NMT_Address	Mandatory mandatory

Through this service the NMT Master creates a remote node object with the requested attributes and inserts it in the remote node set of the network object. The service will only be executed if a network object exists and if there are less than 255 remote node objects. After completion of the service, the state of the remote node object will be DISCONNECTED. The service is local and mandatory.

Remove Remote Node

Through this service the NMT Master removes the remote node object identified by NMT_Address from the remote node set of the network object. The service will only be

NMT Service Specification

<i>Parameter</i>	<i>Request</i>
Argument NMT_Address	Mandatory mandatory

executed if NMT_Address identifies a remote node object whose state is DISCONNECTED. The service is local and mandatory.

Create Node

<i>Parameter</i>	<i>Request</i>
Argument NMT_Address class	Mandatory mandatory mandatory

Through this service an NMT Slave creates a node object with the requested attributes. The service will only be executed if no node object already exists. After completion of the service, the state of the NMT Slave will be DISCONNECTED. The service is local and mandatory.

Delete Node

<i>Parameter</i>	<i>Request</i>
Argument	Mandatory

Through this service an NMT Slave deletes its node object. The service will only be executed if the state of the NMT Slave is DISCONNECTED. The service is local and mandatory.

Identify Remote Nodes

Through this service the NMT Master requests all NMT Slaves whose NMT Address meets the NMT_Address_selection and whose state is CONNECTING, to identify themselves through the 'Identify Node' service. The service is unconfirmed and optional.

NMT Service Specification

<i>Parameter</i>	<i>Request/Indication</i>
Argument NMT_Address_selection	Mandatory mandatory

Identify Node

<i>Parameter</i>	<i>Request/Indication</i>
Argument	Mandatory

Through this service an NMT Slave indicates the NMT Master that there is an NMT Slave whose state is CONNECTING. The service will only be executed if the state of the NMT Slave is CONNECTING. The service is unconfirmed and optional.

Connect Node

<i>Parameter</i>	<i>Request</i>
Argument download	Mandatory optional

Through this service the NMT Slave sets its state from DISCONNECTED to CONNECTING. The NMT Master may optionally be requested to download a configuration to the NMT Slave. The service will only be executed if the state of the NMT Slave is DISCONNECTED. The service is local and mandatory.

Connect Remote Node

Through this service the NMT Master sets the state of the NMT Slave identified by NMT_Address from CONNECTING to PREPARING. The service will only be executed if NMT_Address identifies a remote node object whose state is DISCONNECTED.

The service is confirmed and mandatory. The Remote Result parameter will confirm the success or failure. If the state of the NMT Slave is not CONNECTING the service will fail. In case of success the following holds:

<i>Parameter</i>	<i>Request/Indication</i>	<i>Response/Confirm</i>
Argument NMT_Address Remote Result success download failure reason	Mandatory mandatory	Mandatory selection optional selection optional

- the state of the remote node object identified by NMT_Address will be CONNECTED
- the state of the NMT Slave identified by NMT_Address will be PREPARING.
- the Node-ID attribute of the remote node object identified by NMT_Address and its peer have been assigned a value.

In case of success it will be confirmed whether the NMT Slave needs a configuration to be downloaded. In case of a failure, the state of the remote node object identified by NMT_Address and its peer will be DISCONNECTED and optionally the reason may be confirmed.

Prepare Remote Node

Through this service the NMT Master sets the state of the NMT Slave identified by Node_ID from PREPARING to PREPARED. The service will only be executed if Node_ID identifies a remote node object whose state is CONNECTED.

Prior to the state transition, the NMT Slave is allowed to obtain identifiers and inhibit times from the DBT Service Element (see /1/) for the COB's required by the CMS protocol for the CMS objects as defined on that NMT Slave. The NMT Master may optionally request that previously obtained identifiers and inhibit times are discarded. If this is not requested, the NMT Slave itself may decide whether or not to discard them. If the NMT Slave does not obtain identifiers and inhibit times from the DBT Service Element, this parameter must be ignored by the NMT Slave.

The service is confirmed and mandatory. The Remote Result parameter will confirm the success or failure of the request. If the state of the NMT Slave is not PREPARING the

NMT Service Specification

<i>Parameter</i>	<i>Request/Indication</i>	<i>Response/Confirm</i>
Argument Node_ID discard	Mandatory mandatory optional	Mandatory selection selection optional
Remote Result success failure reason		

service will fail. If the NMT Slave does not obtain identifiers and inhibit times from the DBT Service Element, the service will succeed. In case of success, the state of the remote node object identified by Node-ID and its peer will be PREPARED. In case of a failure, the state of the remote node object identified by Node_ID and its peer will be DISCONNECTED and optionally the reason may be confirmed.

Start Remote Node

<i>Parameter</i>	<i>Indicatio/Requestn</i>
Argument Node_ID all	Mandatory selection selection

The service will only be executed for the selected remote node objects whose state is PREPARED. Through this service the NMT Master sets the state of the selected NMT Slaves from PREPARED to OPERATIONAL. Only NMT Slaves whose state is OPER ATIONAL may execute services of the CMS Service Element (see /1/).

The service is unconfirmed and mandatory. If the state of an NMT Slave is not PREPARED no state transition will occur on the NMT Slave. After completion of the service, the state of the selected remote node objects will be OPERATIONAL.

Stop Remote Node

The service will only be executed for the selected remote node objects whose state is OPERATIONAL. Through this service the NMT Master sets the state of the selected NMT

Slaves from OPERATIONAL to PREPARED. NMT Slaves whose state is not OPERATIONAL may not execute services of the CMS Service Element (see /1/).

<i>Parameter</i>	<i>Request/Indication</i>
Argument Node_ID all	Mandatory selection selection

The service is unconfirmed and mandatory. If the state of an NMT Slave is not OPERATIONAL no state transition will occur on the NMT Slave. After completion of the service, the state of the selected remote node objects will be PREPARED.

Disconnect Remote Node

<i>Parameter</i>	<i>Request/Indication</i>
Argument Node_ID all	Mandatory selection selection

The service will only be executed for the selected remote node objects. Through this service the NMT Master sets the state of the selected NMT Slaves to DISCONNECTED independent of their present state and undefines their Node-ID attribute.

The service is unconfirmed and mandatory. After completion of the service, the state of the selected remote node objects will be DISCONNECTED and their Node-ID attributes are undefined.

Disconnect Node

<i>Parameter</i>	<i>Request</i>
Argument	Mandatory

Through this service, the NMT Slave sets the state of the node object to DISCONNECTED independent of its present state and undefines its Node-ID attribute. The service will only be executed if a node object exist. The service is local and mandatory.

NMT Service Specification

5.2 Error Control Services

The mandatory error control services need to be implemented on the NMT Master if and only if the Network Error capability has been configured on the NMT Master. The mandatory error control services need to be implemented on an NMT Slave if and only if the Node Error capability has been configured on that NMT Slave.

Network Event

<i>Parameter</i>	<i>Indication</i>
Argument	Mandatory
remote_error	selection
Node_ID	mandatory
local error	selection
state	mandatory
occurred	selection
resolved	selection

The service is provider initiated and mandatory and does not affect the state of the network object. A network object must exist. Through this service, the NMT service provider on the NMT Master indicates that one of the following has occurred:

- a remote error occurred or has been resolved for the remote node object identified by Node_ID and its peer.
- a local error occurred or has been resolved on the NMT Master.

Node Event

The service is provider initiated and mandatory and does not affect the state of the network object. A node object must exist. Through this service, the NMT service provider on an NMT Slave indicates that one of the following has occurred:

- a remote error occurred or has been resolved for the NMT Master.
- a local error occurred or has been resolved on the NMT Slave.

<i>Parameter</i>	<i>Indication</i>
Argument remote_error local_error state occurred resolved	Mandatory selection selection mandatory selection selection

5.3 Configuration Control Services (non-segmented)

When using these services the NMT is responsible for the segmentation of the configuration. All non-segmented configuration control services are optional.

Configuration Download

<i>Parameter</i>	<i>Request/Indication</i>	<i>Response/Confirm</i>
Argument Node_ID data size Remote Result success failure reason	Mandatory mandatory mandatory optional	Mandatory selection selection optional

Through this service the NMT Master downloads configuration data from the NMT Master to the NMT Slave identified by Node_ID. The data and optionally its size are indicated.

The service will only be executed if Node_ID identifies a remote node object. The service is confirmed and optional. The Remote Result parameter will indicate the success or failure of the request. In case of a failure, optionally the reason may be indicated.

Configuration Upload

Through this service the NMT Master uploads configuration data from the the NMT Slave identified by Node_ID to the NMT Master.

NMT Service Specification

<i>Parameter</i>	<i>Request/Indication</i>	<i>Response/Confirm</i>
Argument Node_ID Remote Result success data size failure reason	Mandatory mandatory	Mandatory selection mandatory optional selection optional

The service will only be executed if Node_ID identifies a remote node object. The service is confirmed and optional. The Remote Result parameter will indicate the success or failure of the request. In case of success, the data and optionally its size are confirmed. In case of a failure, optionally the reason may be confirmed.

5.4 Configuration Control Services (segmented)

When using these services the application is responsible for the segmentation of the configuration. The mandatory segmented configuration control services need to be implemented on the NMT Master if and only if the Network Configuration capability has been configured on the NMT Master. They need to be implemented on an NMT Slave if and only if the Node Configuration capability has been configured on that NMT Slave.

Initiate Configuration Download

<i>Parameter</i>	<i>Request/Indication</i>	<i>Response/Confirm</i>
Argument Node_ID size Remote Result success failure reason	Mandatory mandatory optional	Mandatory selection selection optional

Through this service the NMT Master prepares the NMT Slave identified by Node_ID for downloading a configuration from the NMT Master. Optionally the size of the configuration to be downloaded may be indicated.

The service will only be executed if Node_ID identifies a remote node object. The service is confirmed and mandatory. The Remote Result parameter will indicate the success or failure of the request. In case of a failure, optionally the reason may be confirmed.

Initiate Configuration Upload

<i>Parameter</i>	<i>Request/Indication</i>	<i>Response/Confirm</i>
Argument Node_ID	Mandatory mandatory	
Remote Result success size failure reason		Mandatory selection optional selection optional

Through this service the NMT Master prepares the NMT Slave identified by Node_ID for uploading a configuration to the NMT Master. The service will only be executed if Node_ID identifies a remote node object. The service is confirmed and mandatory. The Remote Result parameter will indicate the success or failure of the request. In case of success, optionally the size of the configuration to be uploaded is confirmed. In case of a failure, optionally the reason may be confirmed.

Download Configuration Segment

Through this service the NMT Master transfers the data of the next segment to the NMT Slave identified by Node_ID. The data and optionally its size are indicated. The continue parameter indicates whether there are still more segments to be downloaded or that this was the last segment to be downloaded. A successful 'Initiate Configuration Download' service must have been executed prior to this service.

The service will only be executed if Node_ID identifies a remote node object. The service is confirmed and mandatory. The Remote Result parameter will indicate the success or failure of the request. In case of a failure, optionally the reason is confirmed. In case of success, the NMT Slave identified by Node_ID has accepted the segment data and is ready to accept the next segment.

NMT Service Specification

<i>Parameter</i>	<i>Request/Indication</i>	<i>Response/Confirm</i>
Argument Node_ID data size continue more last Remote Result success failure reason	Mandatory mandatory mandatory optional mandatory selection selection	Mandatory selection selection optional

Upload Configuration Segment

Parameter	Request/Indication	Response/Confirm
Argument Node_ID Remote Result success data size continue more last failure reason	Mandatory mandatory	Mandatory selection mandatory optional mandatory selection selection selection optional

Through this service the NMT Master requests the NMT Slave identified by Node_ID to supply the data of the next segment to the NMT Master. A successful 'Initiate Configuration Upload' service must have been executed prior to this service.

The service will only be executed if Node_ID identifies a remote node object. The service is confirmed and mandatory. The Remote Result parameter will indicate the success or failure of the request. In case of a failure, optionally the reason may be confirmed. In case

of success, the data and optionally its size are confirmed. The continue parameter confirms whether there are still more segments to be uploaded or that this was the last segment to be uploaded.

Abort Configuration Transfer

This service aborts the up- or download of a configuration to or from the NMT Slave identified by Node_ID. The service may be executed at any time by the NMT Master. An NMT Slave will only execute this service as a response to any of the other configuration control services. Optionally the reason may be indicated. On the NMT Master the service will only be executed if Node_ID identifies a remote node object. On the NMT Slave the service will only be executed if a node object exists and the Node_ID parameter must be ignored.

<i>Parameter</i>	<i>Request/Indication</i>
Argument Node_ID reason	Mandatory mandatory optional

The service is unconfirmed and mandatory. If the NMT Master has a confirmed configuration service outstanding for the NMT Slave identified by Node_ID, the Abort Indication is taken to be the Confirm of that service.

Verify Configuration

<i>Parameter</i>	<i>Request/Indication</i>	<i>Response/Confirm</i>
Argument Node-ID check_sum Remote Result success failure reason	Mandatory mandatory mandatory	Mandatory selection selection optional

Through this service the NMT Master requests the NMT Slave identified by Node_ID to verify if check_sum matches with the last configuration that was successfully downloaded. The value of the check_sum is application specific and does not fall within the scope of the CiA Standard on the CAN Application Layer for Industrial Applications.

NMT Service Specification

The service will only be executed if Node_ID identifies a remote node object. The service is confirmed and optional. The Remote Result parameter will confirm the success or failure of the verification. In case of a failure optionally the reason may be confirmed.

6. STATE TRANSITION DIAGRAMMS

Each remote node object on the NMT Master and its peer on an NMT Slave maintain a state transition diagram, see fig.2 and fig.3. The state transitions in these diagrams are caused by the NMT services as indicated in these diagrams. Some services can only cause a state transition in one diagram, while others may cause a state transition in both diagrams. Services that do not cause a state transition are not drawn in the state transition diagrams. An error response/confirm indicates that for the service involved, the 'failure' selection in the service specification will be selected.

Depending on the NMT services that were executed, the NMT Master assumes that the node object of an NMT Slave is in a certain state. To detect if this assumption is true, the NMT Master regularly retrieves the state of an NMT Slave and compares it to the state of its peer. This mechanism is called Node Guarding and the protocol the Node Guarding Protocol. If the comparison fails or if the state of an NMT Slave could not be retrieved at all, this is indicated to the NMT Master through the 'Network Event' service as a remote error. If the node state of an NMT Slave has not been retrieved during a certain period of time by the NMT Master, this is indicated to the NMT Slave through the 'Node Event' service as a remote error. Note that the 'Network Event' and 'Node Event' services do not cause a state transition.

The Node Guarding Protocol is active if and only if the NMT Master has the Network Error capability, the NMT Slave has the Node Error capability, and if the (remote) node state is not DISCONNECTED or CONNECTING, see fig.2 and fig.3. Note that the 'Disconnect Remote Node' service causes the Node Guarding Protocol to stop for that NMT Slave, since the state of the remote node object becomes DISCONNECTED.

If the Node Guarding Protocol functions again normally after a remote error has been reported and no state transition has occurred, this is indicated to the NMT Master and NMT Slave through the 'Network Event' and the 'Node Event' service respectively.

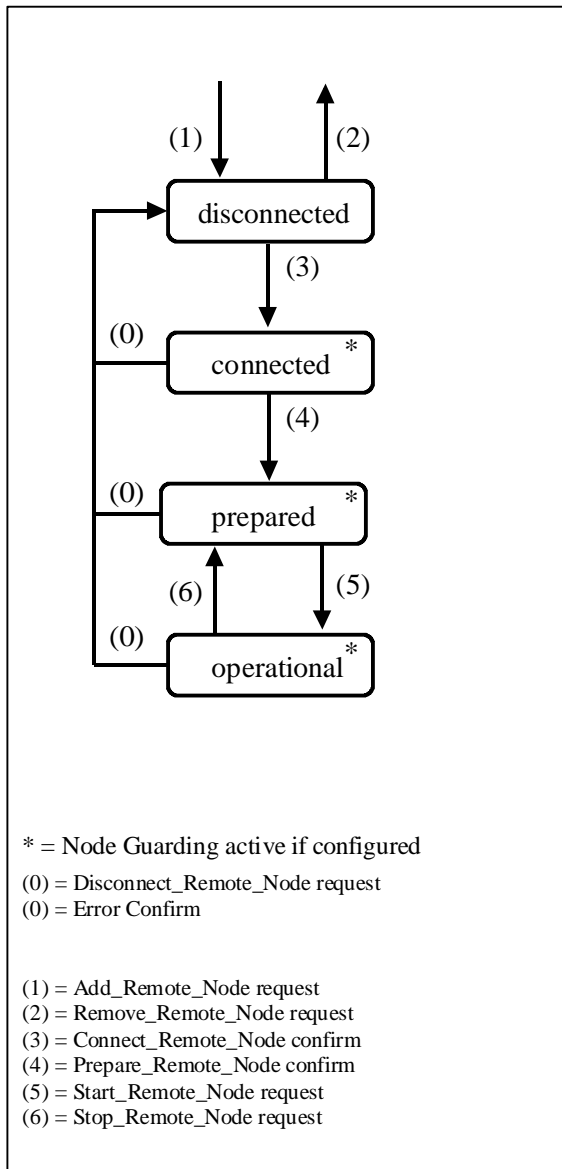


Fig. 2: Remote Node State Diagram

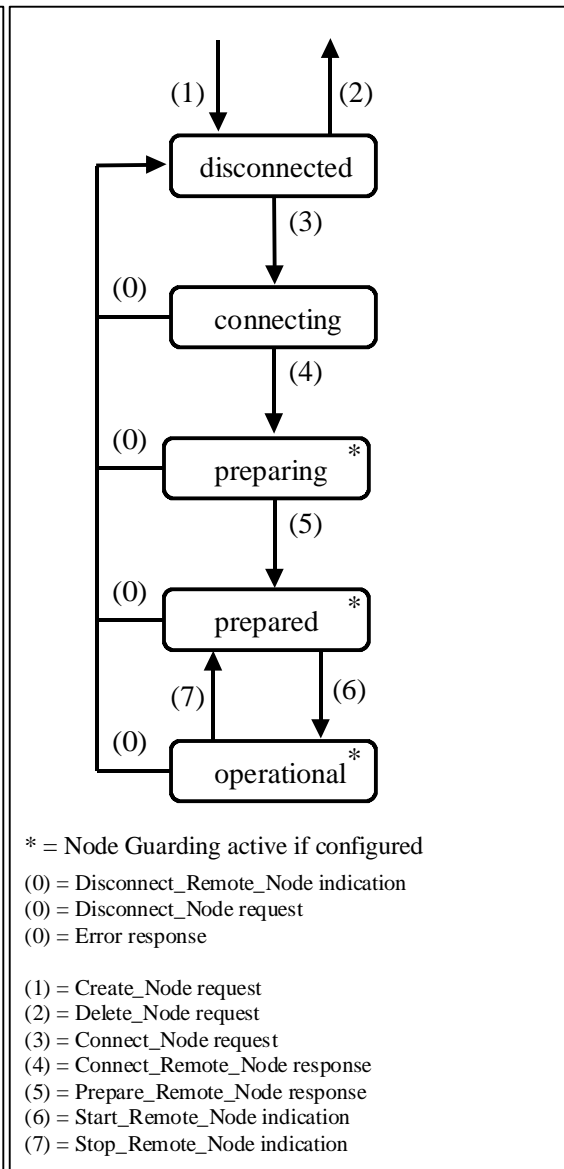


Fig. 3: Node State Diagram