Navithor Fleet Control - Communication specification between Navithor and MES

Navitec Systems

This document describes the interface between the Navithor Server and MES.

Version 2.92

Updated 05.03.2024

Table of contents

1. Version history

Document Version	Date	Author	Description
1.0	02.05.2016	Jouni Sievilä	First version
1.1	26.10.2016	Joni Sundholm	Added description texts to all messages
1.2	21.02.2017	Joni Sundholm	Added battery- and loadstatus-fields to AGVStatus-message and updated ProductionStatus-message. Added messages for handling of ProductionOrderLists. Navithor ver. 1.7 required for support
1.3	06.03.2017	Joni Sundholm	AsssingnedToMachine-field type changed from int16 to int32. DestinationIDs-field in message ID = 318 changed from int16 to uint16
1.4	11.05.2017	Joni Sundholm	Added messages TagStatus and TagStatusRequest
1.5	15.05.2017	Ville Rahikka	Major update to the chapter structure. Added RequestPickUp and RequestPickUpResult. Added ErrorIDs chapter
1.51	19.06.2017	Ville Rahikka	Minor spelling error
1.6	26.06.2017	Aleksi Ålander	Correction to SetStatusAtSymbolicPoint message
1.7	27.11.2017	Aleksi Ålander	Added messages 8.1.24 ClearProductionOrders and 8.1.25 ClearTransferRequests
1.71	08.12.2017	Aleksi Ålander	Corrected in 8.2.17. order status interrupted to be 4 and completed 5
1.72	16.01.2018	Tommi Kärki	Added data field AffixNumber to SetStatusAtSymbolicPoint
1.73	06.02.2018	Aleksi Ålander	Small fixes and notes to OutputAll and other Output messages
1.80	07.02.2018	Aleksi Ålander	Added TagType field to message 8.1.22. TagStatus
1.81	08.02.2018	Aleksi Ålander	Fixed the TagStatus format
1.9	28.02.2018	Aleksi Ålander	Corrected messages PickupReady, PickupAborted, EmptyReady, EmptyAborted
1.91	21.03.2018	Aleksi Ålander	Added 'cancelled' status to ProductionStatus message
2.0	21.08.2018	Joni Sundholm	DriveReady-messages (ID = 302) level-field is int32, not uint32 as previously in this document. Reflecting MES interface version 1.5 in Navithor Server
2.1	26.09.2018	Joni Sundholm	InitialSetup-messages (ID = 102) maximum data length is now 65535 bytes instead of 10000 Added message CancelActiveProductionOrderList (ID = 28). Supported by Navithor version 1.9.55.0 or later

Document Version	Date	Author	Description
2.2	09.11.2018	Aleksi Ålander	Added messages ResourceStatus (ID = 321), ReleaseHoldToCustomLoc (ID = 29) and SetDriveOffset (ID = 30). Messages supported by Navithor version 1.9.59.0 or later
2.3	18.02.2019	Joni Sundholm	Added messages RequestResourceStatusAtLocation (ID = 322) and AckRejRequest (ID = 201) Messages supported by Navithor version 1.9.59.0 or later
2.4	15.03.2019	Aleksi Ålander	New message TransferRequestStatus (ID = 323). Added new optional fields to TransferRequest (ID = 21) Added exectionStatus to ProductionStatus (ID = 313) Document version number matches from now on with MES interface version
2.5	09.04.2019	Joni Sundholm	Added RequestPickup-field to SetStatusAtSymbolicPoint-message. Supported in Navithor version 1.9.62.0 or later
2.6	28.05.2019	Samuli Vaara	Added message SymbolicPointStatus (ID = 324) and SetSymbolicPointStatus (ID = 34). Modified TransferRequest (ID = 21) to allow requesting groups
2.7	31.07.2019	Aleksi Ålander	Added chapter 8.4. Heartbeat with description about the optional heartbeat and added messages 8.4.1 Heartbeat (ID = 203) and 8.4.2 HeartbeatResponse (ID = 204)
2.8	03.10.2019	Aleksi Ålander	Added message FireAlarmSignal (ID = 36)
2.9	17.10.2019	Björn Eike Sopha	Editing ResponseId description of TransferRequest (ID = 21)
2.10	29.10.2019	Samuli Vaara	Added Active FireAlarmStatus (ID = 326), fixed some typos and heading numbering
2.11	13.12.2019	Samuli Vaara	Added ClearTransfRequestWithTransferID
2.12	19.12.2019	Tommi Kärki	A lot of corrections to message lengths in document. String max lengths specified. Array max length specified
2.13	23.01.2020	Aleksi Ålander	Fixed OutputAll, OutputFloat and OutputDigital messages to Navithor version 2.13.0.0 to work correctly
2.14	23.03.2020	Evgenii Kuzminskii	Modified Version history table layout. Corrected number for 8.2.30 Added DriveMachineToSymbolicPointWithStops (ID = 38), EstimateDriveToStartSymbolicPoint (ID = 41), StoppedAtStopPoint message (ID 328), EstimationOfDriveToStartSymbolicPoint (ID = 329)
2.15	28.04.2020	Evgenii Kuzminskii	Updated RequestPickUp (ID = 25) Added RackTransferRequest (ID = 39), SetRackStatus (ID = 40), GetRackSetup (ID = 43), DriveMachineToRack (ID = 44) Updated RequestPickUpResult (ID = 320) Added RackSetupReply (ID = 330), RackSetup (ID = 331)

Document Version	Date	Author	Description
2.16	29.04.2020	Tommi Kärki	Added missing SetResourcesAtMachine (ID = 42)
2.17	29.04.2020	Evgenii Kuzminskii	Updated SetResourcesAtMachine (ID = 42) and fixed numbering
2.18	29.04.2020	Evgenii Kuzminskii	Updated DriveMachineToRack (ID = 44)
2.19	06.05.2020	Aleksi Ålander	Clarify description for SetResourcesAtMachine (ID = 42), corrected header numbering for messages 8.1.41 and 8.1.42, updated Table of contents
2.20	14.05.2020	Samuli Vaara	Added optional flags for AddProductionOrder (ID = 16)
2.21	15.05.2020	Evgenii Kuzminskii	Added LoadUnloadMachineRequest (ID = 46)
2.22	28.07.2020	Aleksi Ålander	Modified InitialSetup ResourceTypeData struct's resource type ID datatype from uint16 to int32. Note: Before Navithor version 2.20.0.0 these resource types were not sent with InitialSetup, even though they should have been. This was fixed to release 2.20.0.0
2.23	31.08.2020	Björn Eike Sopha	Removed optional field that was not implemented for SetSymbolicPointAvailability (ID = 31)
2.24	02.09.2020	Antti Parkkonen	Added RequestZoneStatus (ID = 51) and ZoneStatusReply (ID = 333) messages to this document
2.25	15.09.2020	Aleksi Ålander	Added new messages GetSymbolicPointGroupSetup (ID = 46), SaveLogs (ID = 48), AckAlarms (ID = 49), SetBufferLaneStatus (ID = 50) and SymbolicPointGroupSetup (ID = 332). Added also option to extend existing TransferReqeustStatus (ID = 323) to include more information. This is optional and can be enabled from Navithor Server parameters, on default old shorter form is used
2.26	31.08.2020	Björn Eike Sopha	Fixing message SetSymbolicPointAvailability (ID = 31)
2.27	24.09.2020	Toni Liski	Added a new message ProductionAreaChanged (ID = 334)
2.28	24.09.2020	Samuli Vaara	Added a new message MesBatteryTemperatureWarning (ID = 334). Corrected previous message in doc 334 -> 335
2.29	01.10.2020	Tommi Kärki	Added new messages ID = 52 (GetProductionAreaInformation) and ID = 336-344 which are replies to ID = 52. Removed InitialSetup from Tips because it is optional
2.30	19.10.2020	Toni Liski	Add new messages SetResourceAtLocationWithExtendedInfo (ID = 53) and TransferRequestWithResourceIdentifier (ID = 54)
2.31	04.12.2020	Toni Liski	Add new messages RequestSPResourceStatus (ID = 55) and SymbolicPointResourceStatus (ID = 345)
2.32	10.12.2020	Evgenii Kuzminskii	Updated description for several fields of RackTransferRequest (ID = 39) message
2.33	17.11.2020	Tommi Kärki	Added optional field documentation for RackTransferRequest (ID = 39) message

Document Version	Date	Author	Description
2.34	15.01.2021	Tommi Kärki	Corrected TransferRequest wrongly mentioning AckOrRejRequest being sent as reply. It is actually AckOrRej. Changed RackTransferRequest TransferId to be uint32 instead of int32 to match TransferRequestStatus RequestId
2.35	29.01.2021	Evgenii Kuzminskii	Fixed document style after conversion from DOCX to MD. Merged documents for incoming and outgoing messages into this document with multiple minor fixes. Updated type of ProductionOrderID field in StoppedAtStopPoint message (ID = 328)
2.36	17.02.2021	Evgenii Kuzminskii	Added chapter with MES message example, updated CurrentStatus field description in ProductionOrderData struct of ProductionStatus message (ID = 313)
2.37	25.02.2021	Aleksi Ålander	Added option to send DistanceToTarget in AGVStatus message, if parameter MES_Protocol_Version_For_Status_Messages is at least 3.
2.38	15.03.2021	Aleksi Ålander	Added option to send CurrentDriveThroughPoint in AGVStatus message, if parameter MES_Protocol_Version_For_Status_Messages is at least 4.
2.39	12.04.2021	Tero Laakso	Add new message ErrorsV2
2.40	19.04.2021	Evgenii Kuzminskii	Corrected ErrorLevel description in ErrorsV2 specification
2.41	28.04.2021	Evgenii Kuzminskii	Updated description for Item type ID field of ShelfStatus struct for SetRackStatus message
2.42	26.05.2021	Evgenii Kuzminskii	Renamed Group to ErrorType field of ErrorDataV2 struct in ErrorsV2 specification and updated description for it and for multiple fields in ErrorsV2Request specification
2.43	11.06.2021	Aleksi Ålander	Clarified that StrictUnloadLoc in TransferRequest is obsolete. Renamed title, minor corrections to formatting
2.44	16.06.2021	Elena Sgonova	Fixed typos in ErrorDataV2 fields names. Added EntityID field description
2.45	17.06.2021	Toni Liski	Add new messages ElevatorStatus (ID = 351) and ElevatorRequest (ID = 352)
2.46	18.06.2021	Toni Liski	Add new Subscription mode ElevatorRequest to MESMessageSubscription
2.47	30.06.2021	Aleksi Ålander	Added option to send CurrentDriveThroughPoint, NextLevelChangePointId and DistanceToNextLevelChange in AGVStatus message, if parameter MES_Protocol_Version_For_Status_Messages is at least 5
2.48	02.07.2021	Evgenii Kuzminskii	Corrected header level and location of ErrorsV2 message (ID = 349) in the document
2.49	07.07.2021	Evgenii Kuzminskii	Changed value range for ElevatorStatus and DoorStatus fields in ElevatorStatus message (ID = 351)
2.50	03.08.2021	Aleksi Ålander	Changed RequestIdType in message 55 into RequestFlags and added new option of requesting ResourceCreationDate in SymbolicPointResourceStatus (ID = 345).

Document Version	Date	Author	Description
2.51	18.08.2021	Toni Liski	Corrected SPStatus values in message SymbolicPointResourceStatus (ID = 345).
2.52	06.09.2021	Tommi Kärki	Added new field to RackTransferRequest (ID = 39) message. Further clarified how message works.
2.53	29.09.2021	Benard Gathimba	Added DeleteProductionOrderWithId (ID = 353) message.
2.54	08.11.2021	Joni Sundholm	Clarified description for RequestResourceStatusAtLocation (ID = 322) message. Added more error- and reject-reasons to Error IDs-chapter
2.55	13.12.2021	Aleksi Ålander	Reverted change of message 345 minimum length done in protocol version 2.50.
2.56	15.12.2021	Aleksi Ålander	Added option to send LastSymbolPointDrivenOver in AGVStatus message, if parameter MES_Protocol_Version_For_Status_Messages is at least 6.
2.57	02.02.2022	Toni Liski	Added new optional flags for RequestSPResourceStatus message to request all resources and resource slot and shelf indices.
2.58	01.03.2022	Elena Sgonova	Corrected mistakes found by a customer (ID = 348, ID = 320), fixed typos found all over the document.
2.59	03.03.2022	Samuli Vaara	Added SetSymbolicPointPriority (ID = 355)
2.60	10.03.2022	Samuli Vaara	Added RackId extenstion for pickup and target in RackTransferRequest (ID = 39)
2.61	06.04.2022	Aleksi Ålander	Added option to send RequestFlags in SymbolicPointResourceStatus (ID = 345) in case correct override file is found.
2.62	06.04.2022	Samuli Vaara	Added TransferRequestReply (ID = 356), reorganize a bit.
2.63	25.04.2022	Tommi Kärki	Added InitPositionToSymbolicPoint.
2.64	18.05.2022	Aleksi Ålander	AddProductionOrder (ID = 16) extended with option to include racks in the order and new flags for giving only pickup or dropoff order.
2.65	19.05.2022	Aleksi Ålander	Added messages ValidateLoad (ID = 357) and ValidateLoadReply (ID = 358). Added a new flag to AddProductionOrder for requiring load validation.
2.66	31.05.2022	Tero Laakso-Pitkäkoski	Added message StartStopProductionBreak (ID = 359) to activate and stop production break.
2.67	19.05.2022	Aleksi Ålander	Added RequestID to the TransferRequestReply (ID = 356) and modified order of optional RackData in AddProductionOrder (ID = 16) according to customer feedback.
2.68	30.06.2022	Tero Laakso-Pitkäkoski	Length of Status field in TrafficLightStatus fixed to match field type.

Document Version	Date	Author	Description
2.69	14.07.2022	Aleksi Ålander	Updated DriveMachineToSymbolicPoint behaviour in case of sending machine ID -1.
2.70	30.08.2022	Toni Liski	Add MES Subscription for ProductionStatus (ID = 313) message.
2.71	05.09.2022	Samuli Vaara	Added NearSymbolicPoint for AGV status, requires protocol version 7
2.72	14.09.2022	Toni Liski	Added a new message ReturnToRoute (ID = 361).
2.73	15.09.2022	Toni Liski	Added a new message ReleaseStopAndHold (ID = 362).
2.74	19.09.2022	Samuli Vaara	AGVStatus NearSymbolicPoint value when not found to -1, heading parameter was wrong in docs as well
2.75	16.09.2022	Toni Liski	DriveMachineToSymbolicPoint (ID = 19) extended to contain OrderFlags.
2.76	30.09.2022	Aleksi Ålander	ErrorsV2 (ID = 349) sends ID as uint32, if MES_Protocol_Version_For_Status_Messages is set to at least 8.
2.77	30.11.2022	Toni Liski	Update maximum priority for certain production order and transfer request related messages.
2.78	10.01.2023	Simon Clement	Update terminology for symbolic point types
2.79	03.02.2023	Simon Clement	Changed behavior of SymbolicPointResourceStatus (ID = 345) to return quantity of items in each resource when RequestSPResourceStatus (ID = 55) is sent with bit 3 enabled in RequestFlags.
2.80	10.02.2023	Toni Liski	Add two new messages ActivateMissionTemplate (ID = 10008) and MissionTemplateActivationResult (ID = 10009).
2.81	10.03.2023	Toni Liski	Add the list of Navithor 3.0 supported MES messages.
2.82	28.03.2023	Samuli Vaara	Add support for getting rack side information when RequestSPResourceStatus (ID = 55) is sent with bit 4 enabled in RequestFlags.
2.83	24.05.2023	Toni Liski	Removed redundant messages LoadPickupAborted (ID = 305) and EmptyLoadAborted (ID = 308) and renamed other messages (IDs 303 - 307) to LoadPickupStarted, LoadPickupComplete, LoadDropoffStarted and LoadDropoffComplete.
2.84	26.05.2023	Toni Liski	Updated terminology: resource -> load.
2.85	26.05.2023	Evgenii Kuzminskii	Updated error level to include info value (0) in Errors message (ID = 309), ErrorsV2Request (ID = 56) and ErrorsV2 message (ID = 349).
2.86	06.07.2023	Evgenii Kuzminskii	Fixed "Max Length" of Errors (ID = 309), SymbolicPointResourceStatus (ID = 345) and ErrorsV2 (ID = 349) messages. Fixed "Min Length" of SymbolicPointResourceStatus (ID = 345) message.
2.87	07.08.2023	Aleksi Ålander	Added support for messages ClearTransferRequestsBySymbolicPointId (ID = 27) and ClearTransferRequestWithTransferId (ID = 37). Clarified that ProductionStatus (ID = 313) is still supported in Navithor 3.0.

Document Version	Date	Author	Description
2.88	01.11.2023	Aleksi Ålander	Added lists of currently not supported and obsolete messages and removed the list of supported messages.
2.89	14.12.2023	Toni Liski	Added support for negative levels in ElevatorStatus (ID = 351) message if StatusType is 1.
2.90	09.02.2024	Toni Liski	Added new messages PivotCommand (ID = 363) and PivotResult (364).
2.91	01.03.2024	Samuli Vaara	Updated ClearProductionOrders (ID = 26)
2.92	05.03.2024	Ramil Kadyrov	Fixed description for SymbolicPointGroupStatus (ID = 324). Navithor ver 3.x supports SetSymbolicPointGroupStatus (ID = 34), SymbolicPointGroupStatus (ID = 324), GetSymbolicPointGroupSetup (ID = 47), SymbolicPointGroupSetup (ID = 332)

2. Glossary

Term	Description
MES	Manufacturing execution system
Navithor	Navithor is fleet management software controlling a fleet of AGVs

3. General

This document describes the interface between the Navithor Server and MES.

4. Communication protocol

Communication between Navithor and MES client follows TCP/IP protocol where Navithor works as a TCP server. Listening port for communication establishment is defined in parameters. By default the listening port is 8015.

5. Endianness

Little-endian format is used (LSB first). Navithor does not support big-endian currently.

6. MES channel

MES channel uses TCP protocol. Navithor acts as a server, listens for connections and allows one or multiple PLCs to connect as a client. Messages can be sent in both directions in an MES channel, e.g. for receiving commands from MES and reporting Navithor and AGV statuses back to MES.

6.1. Versioning and identification

Communication protocol follows semantic versioning with a format XXX.YYY where XXX is major version and YYY is minor version.

Major number is incremented whenever backwards compatibility of the specification is compromised and all SW through the system should be updated. This kind of change in specification should be allowed only on very significant reasons. Severity of changing major number naturally depends also on the number of different machines and applications in the system. It is also possible to introduce new major number for special reasons even if the backwards compatibility has not been compromised. For example if it is desirable to force all SW in the system to be updated for some reason. As explained in the *"Backward/forward compatibility"* chapter, applications refuse to communicate with other parties if the major number of the communication specification does not match. Any HMI application should give a clear error in such a case.

Minor number is incremented whenever specification is extended or modified in backwards compatible manner but reflecting change in implementation. These changes require software update to MES only if the new features of the specification need to be supported. Whenever major number is increased, minor number is set to 0.

Every message includes identification information about its sender and receiver. For instance, while sending a message each machine and HMI adds its identification number to a message header as a sender ID. The same way an ID of machine or HMI the message is addressed to is included in the message header. Therefore, the receiver can check who has sent the message using the sender ID and reject the message if its own ID does not match the receiver ID included in the message.

6.2. Backward/forward compatibility

Rules:

1. When a message is defined the same message id cannot be used for any other message.
2. The meaning of message bytes cannot be changed.
3. The length of message may vary:
    - Messages can be of any length;
    - Frame tells the length of message;
    - Received message can be longer than expected. In this case only the known bytes will be parsed.
4. Messages can be extended as long as the meaning of the original bytes does not change.
5. Any number of new messages can be introduced.

7. Sending and receiving messages

Once connection between Navithor and MES has been successfully established Navithor will start sending following messages periodically:

- **AGVStatus** (ID = 310) - status information of the AGVs in the system;
- **InputValues** (ID = 312) - input data from the AGVs;
- **ProductionStatus** (ID = 313) - status of all production orders in system.

7.1. Message respond requirements

MES is not required to answer or acknowledge any way to messages received from Navithor.

Navithor replies with **AckOrReject** message (ID = 200) to each successfully parsed message from MES and replies with corresponding reply-message when request is handled in Navithor.

7.2. Tips for client software (MES)

- Send message **VersionInfo** (ID = 1) to verify that the communication matches with Navithor.
- Send message **SetTime** (ID = 3) when needing to have the AGVs set time to MES time.

8. Message structure

	Length(bytes)
Frame	9
Message Data	0-65535

8.1. Frame

Every message has a frame that consists of following data. Sender ID and Receiver ID are specified as follow:

Navithor: ID = 1000, MES clients: ID = 1001, 1002, ... For example when sending a message from MES to Navithor Sender ID = 1001 and Receiver ID = 1000.

Field	Type	Length (bytes)	Value	Description
Id	uint16	2	0-65535	Message ID
Sender id	uint16	2	0-65535	Unique ID of the sender
Receiver id	uint16	2	0 Any, 1-65535 receiver id	Unique ID of the receiver
Message type	byte	1	1 reply needed, 2 reply not needed	Tells if the message needs to be replied
Data length	uint16	2	0-65535	Message data length in bytes (0 means that no other data than frame is sent)

8.2. Message data

Message data consists of bytes that are defined in Messages section of this document.

9. Protocol version

The server configuration setting **MES_Protocol_Version_For_Status_Messages** defines the version of the MES protocol which should be used for the communication. By default the protocol version is 1, but if version is increased, some messages will behave differently. By increasing the protocol version additional fields, or fixed data lengths are available to certain messages.

In this section are listed the messages which will be affected by each change in protocol version setting. For example, if some changes enabled by the protocol version 8 are aimed, all messages which are in use must be checked if changing the protocol version affects them.

Protocol version	Changes
1	Default, all messages behave like described in this document.
2	Added several new fields to **TransferRequestStatus** (ID = 323) message.
3	*DistanceToTarget* field is added to **AGVStatus** (ID = 310) message.
4	*CurrentDriveThroughPoint* field is added to **AGVStatus** (ID = 310) message.
5	*NextLevelChangePointId* and *DistanceToNextLevelChange* fields added to **AGVStatus** (ID = 310) message.
6	*LastSymbolPointDrivenOver* field added to **AGVStatus** (ID = 310) message.
7	*NearSymbolicPoint* field added to **AGVStatus** (ID = 310) message.
8	**ErrorsV2** (ID = 349) sends ID as uint32.

10. Messages currently not supported

Navithor 3.X does not support all messages that have been supported in the past. Here is a list of currently not supported messages:

- ChargingRequest (ID = 311)
- DriveMachineToSymbolicPointWithStops (ID = 38)
- EstimateDriveToStartSymbolicPoint (ID = 41)
- GetRackSetup (ID = 43)
- RackSetup (ID = 331)
- RackSetupReply (ID = 330)
- SetDriveOffset (ID = 30)
- SetResourceAtLocationWithExtendedInfo (ID = 53)
- StartStopProductionBreak (ID = 359)
- StoppedAtStopPoint (ID = 328)
- ValidateLoad (ID = 357)
- ValidateLoadReply (ID = 358)

11. Obsolete messages

Some messages are obsolete and won't be supported anymore in the future. Obsolete messages either have no use anymore or have been replaced by other messages that can be used for a similar use case. Here is a list of messages that have been removed from the document as obsolete:

- AckAlarms (ID = 49)
- EmptyLoadAborted (ID = 308)
- EnableProductionOrder (ID = 17)
- LoadPickupAborted (ID = 305)
- TransferRequestV2 (ID = 45)

InitialSetup messages have been replaced with GetProductionAreaInformation (ID = 52) and related reply messages. Obsolete:

- GetInitialSetup (ID = 2)
- InitialSetup (ID = 102)

Production order lists are not supported anymore, so all messages related to them are considered as obsolete. Similar behaviour is still possible with Mission templates. Obsolete:

- ActivateProductionOrderList (ID = 22)
- ActiveProductionOrderListStatus (ID = 317)
- CancelActiveProductionOrderList (ID = 28)
- ProductionOrderListActivatedResult (ID = 315)
- ProductionOrderListCompleted (ID = 316)
- ProductionOrderListTemplateResult (ID = 318)
- RequestProductionOrderListTemplates (ID = 23)

12. Messages

Following message-tables contains required information to build the message data. "Min Length" and "Max Length" are byte counts containing bytes from message frame. Thus, the minimum length is always at least 9 bytes for all messages.

12.1. Messages from MES to Navithor

12.1.1. GetVersion (ID = 1)

Gets the communication version from Navithor.

Frame Length	Data Length	Min Length	Max Length
9	0	9	9

12.1.2. SetTime (ID = 3)

Sets the time on machine.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
Year	uint16	2	0-65536
Month	byte	1	1-12
Day	byte	1	1-31
Hour	byte	1	0-23
Minute	byte	1	0-59
Second	byte	1	0-59

Frame Length	Data Length	Min Length	Max Length
9	7	16	16

12.1.3. FSTOP (ID = 4)

Sending this message will stop a specific machine or all machines in the production system. This also takes the machine(s) out of production.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineID	int16	2	Machine ID, -1 for all machines

Frame Length	Data Length	Min Length	Max Length
9	2	11	11

12.1.4. ReleaseFSTOP (ID = 5)

Sending this message will release the brakes and FSTOP-state of Navitrol. If automatic start production is enabled in Navithor machines are taken back into production automatically after a specified timeout (system parameter, default 30 seconds).

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineID	int16	2	Machine ID, -1 for all machines

Frame Length	Data Length	Min Length	Max Length
9	2	11	11

12.1.5. InitializePosition (ID = 6)

Sending this message will trigger position initialization for the requested machine. Valid position initialization locations are Symbolic Points that define at least posInitId1, see GetProductionAreaInformation (ID = 52). To initialize position for any symbolic point see InitPositionToSymbolicPoint (ID = 59) instead.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineID	uint16	2	
SymbolPointPosInitId	uint16	2	posInitId1 or posInitId2 of the correct Symbolic Point

Frame Length	Data Length	Min Length	Max Length
9	4	13	13

12.1.6. ChargingCommand (ID = 9)

Sending this message will either trigger a charging request for a machine or release the charging request. Charging request from MES forces Navithor to find a valid charging location after the machine has executed its current production order. Charging request triggered by MES can only be released by MES (command = 0), i.e. machines will stay at charge as long as it has not been released by MES.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineID	uint16	2	
command	uint16	2	0 no charge command, 1 charge command

Frame Length	Data Length	Min Length	Max Length
9	4	13	13

12.1.7. OutputAll (ID = 10)

Sending this message will result Navithor to forward this data to the machine.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineId	uint16	2	
DigitalOutputLength	uint16	2	Number of digital outputs, from 0 to max 32
DigitalOutputs	byte[]	0...32	If DigitalOutputLength == 0, skip this field
IntegerOutputLength	uint16	2	Number of integer outputs, from 0 to max 32
IntegerOutputs	int[]	0...128	If IntegerOutputLength == 0, skip this field
FloatOutputLength	uint16	2	Number of float outputs, from 0 to max 32
FloatOutPuts	float[]	0...128	If FloatOutputLength == 0, skip this field

Frame Length	Data Length	Min Length	Max Length
9	8 + DigitalOutputLength + IntegerOutputLength * 4 + FloatOutputLength * 4	17	305

12.1.8. OutputDigital (ID = 11)

Sending this message will result Navithor to forward this data to the machine.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineID	uint16	2	
DigitalOutNumber	uint16	2	Starting from index 0, maximum number 31
DigitalOutValue	byte	1	

Frame Length	Data Length	Min Length	Max Length
9	5	14	14

12.1.9. OutputInteger (ID = 12)

Sending this message will result Navithor to forward this data to the machine.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineID	uint16	2	
IntegerOutNumber	uint16	2	Starting from index 0, maximum number 31
IntegerOut	int32	4	

Frame Length	Data Length	Min Length	Max Length
9	8	17	17

12.1.10. OutPutFloat (ID = 13)

Sending this message will result Navithor to forward this data to the machine.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineID	uint16	2	
FloatOutNumber	uint16	2	Starting from index 0, maximum number 31
FloatOut	float	4	

Frame Length	Data Length	Min Length	Max Length
9	8	17	17

12.1.11. StartStopProduction (ID = 14)

Sending this message will command Navithor to start or stop production with the selected machine or the whole fleet.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineID	int16	2	Machine ID, -1 for all machines
Start	byte	1	1 = Start, 0 = Stop

Frame Length	Data Length	Min Length	Max Length
9	3	12	12

12.1.12. EnableMachine (ID = 15)

Sending this message will enable or disable a selected machine or all machines in the system. Enable machine takes machine into use in Navithor. Disable machine removes the machine from Navithor. **NOTE! Navithor releases the traffic zones that machine reserved when machine is disabled.**

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineID	int16	2	Machine ID, -1 for all machines
Enable	byte	1	0 = Disable, 1 = Enable

Frame Length	Data Length	Min Length	Max Length
9	3	12	12

12.1.13. AddProductionOrder (ID = 16)

Sending this message will trigger Navithor to generate a production order for a specific machine. By default the target can be changed if unavailable. Enable strict dropoff bit to disable this behaviour.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineID	int16	2	Machine ID
productionOrderID	uint32	4	OrderID, unique for each order
fromSymbolicPoint	uint16	2	Symbolic point ID where order starts
toSymbolicPoint	uint16	2	Symbolic point ID where order ends
ItemsToDeliver	uint16	2	Number Of Items to deliver
ItemType	uint16	2	Item ID
StartTimeLength	uint16	2	If 0 start time not defined
StartTime	string	0 or 19	"yyyy.mm.dd.hh.mm.ss"
Priority	uint16	2	1 to 32767, where 1 has the lowest priority
OrderFlags	uint32	4	**Optional field, added in MES version 2.20** Bitmask to define behavior of the production order. Bits 1-6 affect load handling with multi load container. Check the table below for details.
RackData	struct	8...518	**Optional field, added in MES version 2.64.** Specifies shelf and rack IDs for pickup and dropoff locations. Check table below for data structure.

Frame Length	Data Length	Min Length	Max Length
9	18 + StartTimeLength + optional fields	27	568

OrderFlags bitmask

OrderFlags bits	Bitmask	Numeric value
None	0	0
Pickup whole stack	1 << 0	1
Dropoff whole stack	1 << 1	2
Pickup load from bottom of the stack	1 << 2	4
Pickup load from top of the stack	1 << 3	8
Dropoff to bottom of the stack	1 << 4	16
Dropoff to top of the stack	1 << 5	32
Strict dropoff (Navithor will not change target)	1 << 6	64
Remove load from the system after dropoff	1 << 7	128
After completing PO, take machine out of production	1 << 8	256
Ignore loads at pickup location	1 << 9	512
Ignore loads at target location	1 << 10	1024
Reserved for internal use	1 << 11 to 15	Do not use
Only pickup order	1 << 16	65536
Only dropoff order	1 << 17	131072
Rack data included in the packet	1 << 18	262144
Load needs to be validated before picking up	1 << 19	524288
A new production order is needed to complete PO	1 << 20	1048576

Optional data: Rack Data

Specifies shelf and rack IDs for pickup and dropoff locations. If rack ID is given, it will override the symbolic point ID and specify desired side of the rack for side loaders. For regular racks (not a sideloader) or non-rack locations, the PickupRackIdLength and DropoffRackIdLength should be given as 0.

FieldName	Type	Length	Description
PickupShelfID	uint16	2	Shelf ID for the height where the load needs to be picked up from (0 for non-rack symbolic points)
PickupRackIdLength	uint16	2	Length of the rack identifier
PickupRackId	string	0...255	Rack identifier, max 255 chars.
DropoffShelfID	uint16	2	Shelf ID for the height where the load needs to be moved to (0 for non-rack symbolic points)
DropoffRackIdLength	uint16	2	Length of the rack identifier
DropoffRackId	string	0...255	Rack identifier, max 255 chars.

Min Length	Max Length
8	518

12.1.14. ReleaseHold (ID = 18)

Sending this message will trigger Navithor to release a machine in the Symbolic Point from HOLD-state and generate a production order for the machine to the specified target location.

Note: HOLD feature is a separate feature from StopAndHold, so this message should not be mixed with *ReleaseStopAndHold (ID = 362)*.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
SymbolicPointId	uint16	2	Symbolic point ID where hold is released
ReleaseTarget	byte	1	0 = default, 1 = first alternative, 2 = second alternative, 3 = third alternative. Note: This is index for predefined hold-Release targets. Use ReleaseHoldToCustomTarget (ID = 29) instead to release by SymbolicPoint ID

Frame Length	Data Length	Min Length	Max Length
9	3	12	12

12.1.15. DriveMachineToSymbolicPoint (ID = 19)

Sending this message will result Navithor to either create a production order for the machine if MachineId is greater than 0. Using MachineId less than zero results Navithor to create a drive request to the Symbolic Point. Navithor will then reason the best possible machine to execute the drive order.

If drive request is given to a location that has Hold-rule enabled, machine goes to hold after arriving at the location. If location is a WaitForPickup symbolic point, machine goes to *Wait for production* task and can continue with any valid order from there. For other type of symbolic points the production order is completed, but the system does not guarantee that the AGV would stay at this location. If there are no unhandeld orders available and there is a reachable wait location available, the AGV will start driving towards the wait location.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineId	int16	2	Machine ID, use < 0 to allow Navithor to select the machine
productionOrderID	uint32	4	OrderID, unique for each order
toSymbolicPoint	uint16	2	Symbolic point ID where order ends
StartTimeLength	uint16	2	If 0 start time not defined
StartTime	string	0 or 19	"yyyy.mm.dd.hh.mm.ss"
Priority	uint16	2	1 to 32767, where 1 has the lowest priority
OrderFlags (optional)	uint32	4	**Optional field:** OrderFlags to define specific options for generated production order. Can be left out from the message. - With `NewOrderRequiredToComplete (1 << 20 = 1048576)` is required that a new production order is needed to complete DriveMachineToSymbolicPoint production order.

Frame Length	Data Length	Min Length	Max Length
9	12 + StartTimeLength	21	44

12.1.16. SetStatusAtSymbolicPoint (ID = 20)

Sending this message will either enable or disable and set load status to the Symbolic Point. Navithor will not order drives to disabled Symbolic Points.

AffixNumber was added in Navithor 1.9.26.0, the old message format without the field is still supported.

Note: AffixNumber is set to 0 if it is not included in the message. RequestPickup-field is supported in Navithor version 1.9.60.0 and above.

Navithor responses with a message **RequestPickupResult (ID = 320)** if `RequestPickup` field was set to `1` to request pickup for the added load.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
SymbolicPoint	uint16	2	Symbolic point ID
NumberOfItems	uint16	2	Number of items of given type at the symbolic point location
ItemTypeId	uint16	2	0 if no load at location, otherwise item type ID that will be set available at the location. ItemTypeID:s are defined in MachineTool
Enabled	byte	1	0 = Disabled, 1 = Enabled
AffixNumber	double	8	**Optional field, on default set to 0.** Number used for additional load info. Informational purpose only
RequestPickup	byte	1	**Optional field, on default set to 0.** If value 1 a pickup request is automatically triggered for the load that is set to be at the location. If Location is empty this field does not have affect

Frame Length	Data Length	Min Length	Max Length
9	7 + optional fields length	16	25

12.1.17. TransferRequest (ID = 21)

Sending this message will result Navithor to create a production source to the pick-up Symbolic Point. Navithor will assign machines that can carry the item type to the target Symbolic Point to fulfil the transfer request. TransferRequest message is replied with TransferRequestReply (ID = 356).

Used together with SetResourcesAtLocation-message. Start production from a symbolic with this message and inform the actual load to be available at the location with message SetResourcesAtLocation.

Fields "Priority" and "RequestID" are supported in Navithor version 1.9.62.X -> MES interface updated to version 1.7. Older versions support only shorter form of the message. Version 2.4 supports both.

Fields "PickupIDType" and "TargetIDType" are supported in Navithor Version 2.1.X.X. ->

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
PickupSymbolicPoint	uint16	2	Symbolic point ID
TargetSymbolicPoint	uint16	2	Symbolic point ID
ItemsToPickup	uint16	2	Number of items
ItemTypeId	uint16	2	
StrictDropoffLoc	byte	1	**Optional field** Obsolete field: If Transfer Request is given to a single symbolic point, Dropoff location is never changed (strict dropoff location rule). If given to symbolic point group instead, Dropoff location can be changed within this group.
Priority	byte	1	**Optional field** 1 to 127, where 1 has the lowest priority. Value 0 means Navithor default priority
RequestID	uint32	4	**Optional field** ID sent by the planner. TransferRequest is initially answered by AckOrReject (ID = 200) and later when production order is created for the TransferRequest, TransferRequestStatus message (ID = 323) is sent using this RequestID
PickupIDType	byte	1	**Optional field** 0 = PickupSymbolicPoint-field points to symbolic point ID 1 = PickupSymbolicPoint-field points to symbolic point group ID
TargetIDType	byte	1	**Optional field** 0 = TargetSymbolicPoint-field points to symbolic point ID 1 = TargetSymbolicPoint-field points to symbolic point group ID

Frame Length	Data Length	Min Length	Max Length
9	8 + optional fields length	17	25

12.1.18. TagStatus (ID = 24)

MES can set a status for defined tags. Following tag linking's are supported by Navithor:

- Symbolic Point of type 'Door'

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
TagID	uint32	4	Tag ID defined in Navithor Tools for Symbolic Point
TagStatus	int32	4	Status of the tag. Defined values: 0 = OFFLINE 1 = DOOR CLOSED 2 = DOOR CLOSING 3 = DOOR OPENING 4 = DOOR OPEN
TagType	byte	1	**Optional field**, Status type (introduced in version 1.80, defaults to 1, if not send in the message): 0 = NO ACTION 1 = DOOR STATUS

Frame Length	Data Length	Min Length	Max Length
9	8 + optional fields length	17	18

12.1.19. RequestPickUp (ID = 25)

Sending this message will result Navithor to create an unhandled pickup order and/or send back RequestPickUpResult message.

If symbolic point is a rack, then ShelfId field must be set to a positive value of valid shelf ID on the rack. If symbolic point is not a rack, then ShelfId field must be skipped or set to zero or any negative value.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
Symbolic Point ID	uint32	4	Symbolic Point ID defined in Navithor Tools for Symbolic Point
Shelf ID	int32	4	**Optional field**, Shelf ID if symbolic point is a rack

Frame Length	Data Length	Min Length	Max Length
9	4 + optional fields length	13	17

12.1.20. ClearProductionOrders (ID = 26)

Sending this message will command Navithor to clear missions assigned to the selected machine or the whole fleet.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineId	int16	2	Machine ID, -1 for all machines
Unused	byte	1	Reserved byte

Frame Length	Data Length	Min Length	Max Length
9	3	12	12

12.1.21. ClearTransferRequestsBySymbolicPointId (ID = 27)

Sending this message will command Navithor to clear all missions executed from the given symbolic point. This means that the mission to be aborted needs to have first step targeting this location and the first step has not been executed yet. If an AGV has already executed the first step, ie. picked up a load, the AGV will continue exceuting the Mission.

If the symbolic point ID is given as -1, all existing missions that have not been progressed past the first step are deleted. The missions that are past the first step are not deleted.

The logic resembles the behaviour with Transfer Requests in releases before Navithor 3.0, but it is good to be aware that Transfer Requests are now converted into Missions and in some scenarios the exact behaviour might be slightly different.

For each aborted mission, MissionAbortReply (MES ID: 10007) is sent as a reply.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
Symbolic Point ID	int32	4	Symbolic point ID, -1 for all symbolic points

Frame Length	Data Length	Min Length	Max Length
9	4	13	13

12.1.22. ReleaseHoldToCustomTarget (ID = 29)

Sending this message will trigger Navithor to release a machine in the Symbolic Point from HOLD-state and generate a production order for the machine to a custom target location, which is given in the message.

Note: HOLD feature is a separate feature from StopAndHold, so this message should not be mixed with *ReleaseStopAndHold (ID = 362)*.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
symbolicPointId	uint16	2	Symbolic point ID where hold is released
releaseTargetId	uint16	2	Target symbolic point ID for the next drive order

Frame Length	Data Length	Min Length	Max Length
9	4	13	13

12.1.23. SetDriveOffset (ID = 30)

The offset information will be sent to Navitrol and the machine should drive the route with the given offset. The offset resets after reaching the given symbolic point, if the target is defined.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineId	int32	4	Machine ID, -1 for all machines
toSymbolicPoint	int32	4	Symbolic point ID where offset resets, -1 to disable automatic reset
Transversal offset	float	4	Positive value to the left of the route, 0 = no offset
Longitudinal offset	float	4	Positive value for driving past the end point. 0 = no offset
Heading offset	float	4	Positive value CCW from route vehicle heading in degrees. 0 = no offset

Frame Length	Data Length	Min Length	Max Length
9	20	29	29

12.1.24. SetSymbolicPointAvailability (ID = 31)

Sets symbolic point to enabled or disabled state.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
SymbolicPointId	uint16	2	ID of symbolic point
Enabled	byte	1	0 = Disabled 1 = Enabled 2 = No change, use last enabled status 2-255 = Reserved

Frame Length	Data Length	Min Length	Max Length
9	3	12	12

12.1.25. SetResourcesAtLocation (ID = 32)

Sets loads to location. If same data is being sent the loads at location will not change. If order of MESResourceAtLocation is changed or if any of data is different from what Navithor thinks then the loads will be regenerated.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
SymbolicPointId	uint16	2	ID of symbolic point
LoadsAtLocationCount	uint16	2	Count of LoadAtLocation structures in array (from 0 to 1000)
LoadsAtLocation	LoadAtLocation[]	LoadAtLocationCount * LoadAtLocation length	Array of LoadAtLocation structures

Frame Length	Data Length	Min Length	Max Length
9	4 + LoadAtLocation length	13	8013

LoadAtLocation

LoadAtLocation struct.

FieldName	Type	Length	Description
LoadTypeId	int32	4	ID of load type
Quantity	int32	4	How many of given load type are at location

Min Length	Max Length
8	8

12.1.26. SetAreaForbidden (ID = 33)

This message can be used to forbid or allow AGV driving on specific route segments/areas. Multiple waypoints can be allowed or forbidden in a single message. Valid from Navithor version 2.0.1.8 forwards.

Note: Zone must be defined as FeatureZone to allow it to be controlled.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
Forbidden area ID	uint16	2	ID for the forbidden area
IsAllowed	byte	1	1 = Driving is allowed, 0 = driving not allowed

Frame Length	Data Length	Min Length	Max Length
9	3	12	12

12.1.27. SetSymbolicPointGroupStatus (ID = 34)

This message can be used to set load related status for a SymbolicPointGroup.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
GroupID	uint16	2	ID of the SymbolicPointGroup
GroupStatus	uint16	2	0 = disable all symbolic points in this group > 0 = enable all symbolic points in this group
RequestedLoadType	uint16	2	0 = set all symbolic points in this group as empty > 0 = set load count for to all points in the group
LoadCount	uint16	2	Number of loads for each point in this group. If value is higher than maximum amount of loads, then the maximum amount is applied
LoadFillMode	uint16	2	Strategy for filling symbolic point groups: 0 = Set "LoadCount" of loads to ALL symbolic points. Max total amount in group: LoadCount * SymPointGroupCount 1 = Set symbolic point loads in sorted order. Max total amount in group: LoadCount

Frame Length	Data Length	Min Length	Max Length
9	10	19	19

12.1.28. GroupedTagStatus (ID = 35)

MES can set statuses for an arbitrary number of tags in a single message.

- Symbolic Point of type 'Door'

Available from Navithor server versions 2.1.2.0 onwards.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumberOfTags	uint16	2	Number of tag datas to send (N). Max 100
TagDatas	TagData[]	4 * N	Tag ID defined in Navithor Tools for Symbolic Point

Frame Length	Data Length	Min Length	Max Length
9	2 + TagDatas length	17	417

TagData

Data structure for a single machine type.

Field Name	Type	Length	Description
TagID	uint16	2	Tag ID defined in Navithor Tools for Symbolic Point
TagStatus	uint16	2	Status of the tag. Defined values: 0 = OFFLINE 1 = DOOR CLOSED 2 = DOOR CLOSING 3 = DOOR OPENING 4 = DOOR OPEN

Min Length	Max Length
4	4

12.1.29. FireAlarmSignal (ID = 36)

This message can be used to activate fire alarm. AGVs inside the active fire alarm zone will drive away of the zone to the closest available symbolic point outside of the zone. During normal operation status is 1. When the fire alarm is active, Status 3 should be sent.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
FireAlarmId	uint32	4	ID for the fire alarm zone
Status	byte	1	0 = none, **1 = inactive**, 2 = deactivating, **3 = fire alarm active**

Frame Length	Data Length	Min Length	Max Length
9	5	14	14

12.1.30. ClearTransferRequestWithTransferID (ID = 37)

Sending this message will command Navithor to clear a mission with the specified TransferID, if it exists. This means that if the TransferID matches any mission with an external ID that can be defined by the user, that mission is aborted. If no matching mission was found, Navithor tries to abort a mission with identical internal ID.

If the TransferID is sent as -1, all existing missions that have not been progressed past the first step are deleted. The missions that are past the first step are not deleted.

The logic resembles the behaviour with Transfer Requests in releases before Navithor 3.0, but it is good to be aware that Transfer Requests are now converted into Missions and in some scenarios the exact behaviour might be slightly different.

For each aborted mission, MissionAbortReply (MES ID: 10007) is sent as a reply.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
TransferID	int32	4	TransferID, -1 for all transfers. TransferID is first compared against external ID of the mission and then to internal ID, if no mission was found

Frame Length	Data Length	Min Length	Max Length
9	4	13	13

12.1.31. DriveMachineToSymbolicPointWithStops (ID = 38)

The MES can send this message to Navithor. Navithor will acknowledge with AckOrReject message.

Sending this message will result Navithor to create a production order for the machine if MachineId is greater than 0.

If MES generates the order with the IsMachineReserved flag set to 1 then the machine executes the order and stays in stop and hold mode until released.

On each stopping point machine goes to stop and hold mode until released.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
ProductionOrderId	int32	4	Order ID, unique for each order
Priority	uint16	2	Order priority from 1 to 32767, where 1 is the lowest priority
MachineId	int32	4	Machine ID
StartSpId	uint16	2	Start symbolic point ID
EndSpId	uint16	2	End symbolic point ID
IsMachineReserved	uint16	2	Flag to tell if the machine is reserved until released: 0 = machine is **not** reserved 1 = machine is reserved
NumberOfStops	uint16	2	Number of stopping points, from 1 to 20. Each of the points has a distance to target, needs to be before the symbolic point
StoppingDistances	int32[]	NumberOfStops * 4	List of stopping points' distances, in millimeters, in decreasing order, one or more (negative values are invalid for now). Minimum decrease is 5 mm

Frame Length	Data Length	Min Length	Max Length
9	18 + StoppingDistances length	31	107

12.1.32. RackTransferRequest (ID = 39)

Sending this message will result Navithor to create a production source to the pick-up Symbolic Point. Navithor will assign machines that can carry the item type to the target Symbolic Point to fulfil the transfer request. RackTransferRequest message is replied with TransferRequestReply (ID = 356).

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
Pickup symbol ID	uint16	2	ID of the symbolic point where the load needs to be picked up from
Pickup shelf ID	uint16	2	Shelf ID for the height where the load needs to be picked up from (0 for non-rack symbolic points)
Target symbol ID	uint16	2	ID of the symbolic point where the load needs to be moved to
Target shelf ID	uint16	2	Shelf ID for the height where the load needs to be moved to (0 for non-rack symbolic points)
Item type ID	uint16	2	Type ID for the load
Priority	uint16	2	Priority for the transfer. The higher this number, the higher the priority. Lowest priority is 1, and 0 means to use default priority.
Ignore pickup status	uint16	2	0 = Transfer request is executed only when the pickup symbol actually has the load on the correct shelf 1 = The load with the item type ID given in this message will be created on the pickup symbol on the pickup shelf when this message is received. If there already is an item on the shelf the type will be converted to the one given in this message
Ignore target status	uint16	2	0 = Transfer request will not be executed until the target location and shelf are available 1 = Transfer request will be executed regardless of the target's load status. If Navithor thinks there is something on the target shelf, the information will be overridden when the AGV drops off its load on the target
Extra information flags	uint16	2	Mandatory bitmask field that is used to determine which optional fields are included in this message. Sending 0 means no additional data is included. Values larger than 0 mean that optional data is being included and must be parsed.
Optional data	byte[]	0...N	Optional data to be included.

Frame Length	Data Length	Min Length	Max Length
9	18 + OptionalDataLength	27	1465

ExtraInformation bitmask

ExtraInformation bits	Bitmask	Numeric value
None	0	0
Barcode	1 << 0	1
TransferId	1 << 1	2
BlockStacking	1 << 2	4
PickupRackId	1 << 3	8
TargetRackId	1 << 4	16

Combining multiple bits is allowed. To include both Barcode and TransferId the ExtraInformation should be set to 3 (1 + 2). To include Barcode and BlockStacking but not TransferId the value should be set to 5 (1 + 4).

The order of optional data MUST match the order of bits listed above. For example if both TransferId and BlockStacking are included then the data for TransferId MUST be included first and the data for BlockStacking MUST be included second.

Optional data: Barcode

Field name	Type	Length	Description
BarcodeLength	uint16	2	Length of the barcode identifier
Barcode	string	[0…N]	Barcode identifier

RESERVED, NOT IMPLEMENTED!

Specifies barcode or other unique identifier for load to be transferred. If this is not included then load will not have associated string identifier.

Optional data: TransferId

Field name	Type	Length	Description
TransferId	uint32	4	Numeric identifier for transfer request.

Specifies identifier for TransferRequest that will be included as 'RequestId' in TransferRequestStatus (ID = 323). If ID is not included then it will be generated automatically.

Optional data: BlockStacking

Enables block stacking (stacking on the top of another load) at the target buffer location. See separate buffer documentation for more details.

Only supported for buffer locations and will be ignored if target destination does not support stacking feature. If this packet is not included, then loads will not be stacked. Racks do not support stacking loads on top of each other, rather the loads will be placed in predefined shelves.

Field name	Type	Length	Description
Flags	uint16	2	Flags indicating how load can be stacked. 0 = No stacking allowed 1 << 0 = load can be stacked on top of previous load 1 << 1 = Another load can be stacked on the top of this load
MaximumStackingHeight	uint16	2	Allowed stacking height in millimeters. How high AGV forks can be lifted to stack the load on top of another. If 0, maximum reach of AGV actuator is used instead.
LoadHeight	uint16	2	Height of the load to be transferred (in millimeters). This data is stored to load object, and is used to speed up the stacking/destacking process at the load buffer. If height 0 is specified, load height can be updated automatically by the system, if available during the load transport process.
LoadsInStack	uint16	2	Allowed number of loads in the stack. If 0, maximum number of loads in the stack is limited only by the height. This information is only used by Navithor together with estimation of the existing loads in the buffer. Navithor can prevent stacking of loads, even the situation in the real buffer would allow stacking.

MaximumStackingHeight, LoadHeight, and LoadsInStack fields are also used to estimate if target load buffer still has free space to hold a new load.

Optional data: PickupRackId

Field name	Type	Length	Description
RackIdLength	uint16	2	Length of the rack identifier
RackId	string	[0…N]	Rack identifier, max 255 chars.

Specifies rackId, this will override the given pickup symbolic point id and specify desired side of the rack for side loaders.

Optional data: TargetRackId

Field name	Type	Length	Description
RackIdLength	uint16	2	Length of the rack identifier
RackId	string	[0…N]	Rack identifier, max 255 chars.

Specifies rackId, this will override the given target symbolic point id and specify desired side of the rack for side loaders.

12.1.33. SetRackStatus (ID = 40)

Equivalent to SetResourcesAtLocation for a warehouse rack.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
Symbol ID	int32	4	ID of the symbolic point where the status on shelves is set
NumOfShelves	uint16	2	Tells how many shelf statuses are set in this message (from 0 to 255)
Shelf Statuses	ShelfStatus[]	NumOfShelves * ShelfStatus length	

Frame Length	Data Length	Min Length	Max Length
9	6 + Shelf Statuses length	15	2055

ShelfStatus

FieldName	Type	Length	Description
Shelf ID	int32	4	ID of the shelf
Item type ID	int32	4	ID of the Item type (0 = remove load)

Min Length	Max Length
8	8

12.1.34. EstimateDriveToStartSymbolicPoint (ID = 41)

The MES can send this message to Navithor. If valid message sent, Navithor will reply to it with EstimationOfDriveToStartSymbolicPoint message. If message is invalid, Navithor will reply to it with AckOrReject message with reject reason.

In this message, both the StartSpId and EndSpId are needed because the drive direction needs to be determined by Navithor for the possible time and distance values to the StartSpId. This needs to be done while avoiding driving through the EndSpId.

Sending this message will result Navithor to reply with driving distances and times to the start symbolic point for each possible machine.

If end symbolic point ID is positive, it must be avoided when planning the route to maintain the correct direction.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
StartSpId	uint16	2	Start symbolic point ID
EndSpId	int32	4	End symbolic point ID. If positive, the route must avoid the symbolic point
SortingField	byte	1	Field for sorting the results: 0 = sort by machine ID 1 = sort by driving distance 2 = sort by driving time

Frame Length	Data Length	Min Length	Max Length
9	7	16	16

12.1.35. SetResourcesAtMachine (ID = 42)

Sets loads at machine. If same data is being sent the loads at machine will not change. If order of LoadAtMachine is changed or if any of data is different from what Navithor thinks then the loads will be regenerated.

When removing all loads from machine, send only MachineId and LoadCount as 0 (header + 4 bytes of data), so that all loads will be cleared.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineId	uint16	2	ID of machine
LoadCount	uint16	2	Number of elements in LoadList. Minimum is 0, maximum is 1000
LoadList	LoadAtMachine[]	LoadCount * LoadAtMachine length	List of LoadAtMachine structs

Frame Length	Data Length	Min Length	Max Length
9	4 + LoadAtMachineList length	13	8013

LoadAtMachine

Load at machine.

FieldName	Type	Length	Description
LoadTypeId	int32	4	ID of load type
Quantity	int32	4	How many of given load type are at location

Min Length	Max Length
8	8

12.1.36. GetRackSetup (ID = 43)

Sending this message will result Navithor to send back RackSetupReply message.

In addition, if MachineTypeId is positive, then for each rack with defined MachineTypeId RackSetup message will be sent back. If MachineTypeId is not positive, then for each rack and for each MachineTypeId defined on that rack RackSetup message will be sent back.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineTypeId	int32	4	ID of the machine type

Frame Length	Data Length	Min Length	Max Length
9	4	13	13

12.1.37. DriveMachineToRack (ID = 44)

Sending this message will result Navithor to create a production order for the machine if MachineId is greater than 0. Negative MachineId values are not supported.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineID	int16	2	Machine ID. Negative values are not supported
ProductionOrderId	uint32	4	OrderID, unique for each order
ToSymbolicPointId	uint16	2	Rack symbolic point ID where order ends
StartTimeLength	uint16	2	If 0 start time not defined
StartTime	string	0 or 19	"yyyy.mm.dd.hh.mm.ss"
Priority	uint16	2	1 to 32767, where 1 has the lowest priority
ToShelfId	uint16	2	Rack shelf ID

Frame Length	Data Length	Min Length	Max Length
9	14 + StartTimeLength	23	42

12.1.38. LoadUnloadMachineRequest (ID = 46)

Sending this message will result Navithor to make a machine do a pickup or dropoff operation.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineID	int32	4	Which machine should perform pickup/dropoff
LoadTypeId	int32	4	Which load type to pickup/dropoff if > 0. Otherwise assume load type from Container configuration
LoadQuantity	int32	4	Quantity of loads to pickup/dropoff if > 0. Otherwise max of given type
LoadType	int16	2	1 = Pickup, 2 = Dropoff. Sending anything else will result in rejected
PickupDropoffFlags	uint16	2	Special handling related to Pickup. 0 = no specials. Used for syncing load state only. Bit flags: 1 << 0 = Reject message if AGV not on symbolic point. 1 << 1 = Changes behavior so that message is not rejected if symbolic point is missing specified load 1 << 2 = Changes behavior so that load will be always removed from system after dropoff 1 << 3 = Source should use PickupFromBottom configuration. Rejected if given loadTypeId is not on bottom slot 1 << 4 = Source should use PickupFromTop configuration. Rejected if given loadTypeId is not on top slot. 1 << 8 = Target should be use DropoffToBottom configuration 1 << 9 = Target should use DropoffToTop configuration

Frame Length	Data Length	Min Length	Max Length
9	16	25	25

Message will be rejected with AckOrReject message if:

- AGV with given MachineId was not found;
- AGV is moving or in production or performing an order;
- Specified LoadTypeId is not valid;
- LoadType was not correct;
- Conflicting flags were being set for example both SourceLoadFromBottom and SourceLoadFromTop set;
- Pickup was given with LoadTypeId that is not found on symbolic point;
- It is not possible to pickup/dropoff specified load.

Warning in Client will be given if:

- Pickup/dropoff was given when AGV was not on symbolic point;
- Dropoff was given with LoadTypeId that machine is not carrying.

12.1.39. GetSymbolicPointGroupSetup (ID = 47)

Gets the initial setup of symbolic point groups from Navithor.

Frame Length	Data Length	Min Length	Max Length
9	0	9	9

12.1.40. SaveLogs (ID = 48)

Sending this message will trigger Navithor to save logs.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumOfMachinesInvolved	uint16	2	Number of machines involved. Can be set to 0, if no logs from the machine are needed. Maximum value is 1000 (N)
MachineIDs	uint16[]	2 * N	Machine IDs as an array. Navithor sends a command to dump a Navitrol log and tries to download it to be included on the log package
DescStringLength	uint16	2	Length of the description. Can be set to 0, if no additional description is needed. Max 1000 chars
Description	string	0...1000	Custom description that is saved to the log package as explanation of the issue
IncidentTimeLength	uint16	2	If 0, incident time not defined. Current system time is used
IncidentTime	string	0 or 19	"yyyy.mm.dd.hh.mm.ss"

Frame Length	Data Length	Min Length	Max Length
9	6 + MachineIDs length + DescStringLength + IncidentTimeLength	15	3034

12.1.41. SetBufferLaneStatus (ID = 50)

Sets symbolic point buffer lane status to full or to has space.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
SymbolicPointId	uint16	2	ID of symbolic point
BufferLaneStatus	uint16	2	0 = No change 1 = Buffer full 2 = Buffer has space

Frame Length	Data Length	Min Length	Max Length
9	4	13	13

12.1.42. RequestZoneStatus (ID = 51)

Gets the status of all zones or the status of a specified zone.

Valid from Navithor 2.20.0.0 forward.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
Zone ID	uint32	4	The ID of the zone whose status needs to be received. ID = 0 for requesting all statuses
Requested information	uint16	2	Determines what information is requested from the Server as a bit field. Existing fields: ZoneOccupancy status 1 << 0 AGVs inside zone (List of AGV ids) 1 << 1

Frame Length	Data Length	Min Length	Max Length
9	6	15	15

12.1.43. GetProductionAreaInformation (ID = 52)

Gets information about production area. Navithor will send a separate reply with requested information.

Valid from Navithor 2.21.0.0 forward.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
InfoType	uint16	2	Type of information to request: 1 = MachineTypeInformation (Reply ID = 336) 2 = MachineInformation (Reply ID = 337) 3 = SymbolicPointInformation (Reply ID = 338) 4 = SymbolicPointHoldInformation (Reply ID = 339) 5 = SymbolicPointSpatialInformation (Reply ID = 340) 6 = ResourceTypeInformation (Reply ID = 341) 7 = ResourceTypeOffsetInformation (Reply ID = 342) 8 = MachineLoadCapacityInformation (Reply ID = 343) 9 = SymbolicPointLoadCapacityInformation (Reply ID = 344)

Frame Length	Data Length	Min Length	Max Length
9	2	11	11

12.1.44. SetResourceAtLocationWithExtendedInfo (ID = 53)

Sets a load at a location with maximum of ten additional strings for extended information. Allows usage of transfer requests with a load identifier which is not load type ID (see message ID = 54).

ResContainerFlags can be used to determine behavior when same or different data is sent.

Supported from version 2.21.0.0.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
SymbolicPointId	uint16	2	ID of symbolic point
LoadTypeId	int32	4	ID of load type
Quantity	int32	4	How many of given load types are at location
LoadContainerFlags	uint16	2	Flags to determine rules for load container options. 0 to use global rules. Other flags currently not implemented
LoadIdStringLength	uint16	2	Length of load identifier
LoadIdentifier	string	0...100	A string that can be used to identify the load instead of using the load type ID
ExtInfoCount	uint16	2	Number of strings for extended info (from 0 to 10). Additional info shown only in Client
MESExtendedLoadInfos	MESExtendedLoadInfo[]	ExtInfoCount * MESExtendedLoadInfo length	MESExtendedLoadInfo structure list

Frame Length	Data Length	Min Length	Max Length
9	16 + LoadIdStringLength + MESExtendedLoadInfos length	25	1145

MESExtendedLoadInfo

MESExtendedLoadInfo structure

FieldName	Type	Length	Description
LoadInfoStringLength	uint16	2	Length of load info
LoadInfo	string	0...100	Additional info as string

Min Length	Max Length
2	102

12.1.45. TransferRequestWithResourceIdentifier (ID = 54)

Sending this message will result Navithor to create a production order from the pick-up Symbolic Point to the Target. Navithor will assign machines that can carry the item type to fulfil the transfer request. TransferRequestWithResourceIdentifier message is replied with TransferRequestReply (ID = 356).

This message supports creating a transfer request with some other load identifier besides load type ID. Should be used in combination with the SetResourceAtLocationWithExtendedInfo (ID = 53) message.

If both ItemTypeId and LoadIdentifier are defined, the load needs to fulfil both requirements for creating a valid order.

This message can be also used with only one of the ItemTypeId or LoadIdentifier defined, or even that neither of them are defined.

Supported from version 2.21.0.0.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
PickupSymbolicPoint	uint16	2	Symbolic point ID
TargetSymbolicPoint	uint16	2	Symbolic point ID
ItemsToPickup	uint16	2	Number of items
ItemTypeId	uint16	2	If 0, not defined, and only Load Identifier used
Priority	uint16	2	1 to 32767, where 1 has the lowest priority. Value 0 means Navithor default priority
RequestID	uint32	4	ID sent by the planner. TransferRequest is initially answered by AckOrRejRequest (ID = 201) and later when production order is created for the TransferRequest, TransferRequestStatus message (ID = 323) is sent using this RequestID
PickupIDType	byte	1	0 = PickupSymbolicPoint-field points to symbolic point ID 1 = PickupSymbolicPoint-field points to symbolic point group ID
TargetIDType	byte	1	0 = TargetSymbolicPoint-field points to symbolic point ID 1 = TargetSymbolicPoint-field points to symbolic point group ID
ResIdStringLength	uint16	2	Length of the load identifier. If 0, only load type ID is used. Max length is 100
LoadIdentifier	string	0...100	Load identifier as string. This can be used instead of, or in combination with ItemTypeId

Frame Length	Data Length	Min Length	Max Length
9	18 + ResIdStringLength	27	127

12.1.46. RequestSPResourceStatus (ID = 55)

Requests the status of all or a certain symbolic point. Answered with SymbolicPointResourceStatus (ID = 345).

Valid from Navithor 2.21.0.0. Rack side request flag added in Navithor 3.0.0.0.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
Identifier	uint32	4	ID of a single symbolic point or a group. If 0, status of all symbolic points will be sent
RequestFlags	uint16	2	Flags: bit 0: 0 = Identifier field points to a symbolic point ID 1 = Identifier field points to a symbolic point group ID bit 1: 1 = Send creation date of last load added to the symbolic point in addition to the default data bit 2: 1 = Send load slot and shelf information. bit 3: 1 = Send information of all loads at specific symbolic point or in a group. bit 4: 1 = Send load rack side information.

Frame Length	Data Length	Min Length	Max Length
9	6	15	15

12.1.47. ErrorsV2Request (ID = 56)

Requests the status of errors in the system. Answered with ErrorsV2 (ID = 349).

Valid from Navithor 2.21.0.0.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
ErrorType	byte	1	Apply exclusive filter by type of error: 0 = all errors 1 = system errors 2 = all entity errors 3 = machine errors (belong to entity errors) 4 = symbolic point errors (belong to entity errors) 5 = waypoint errors (belong to entity errors)
ErrorLevel	byte	1	Apply filter by minimum level of error: 0 = info (all) 1 = warning 2 = error 3 = fatal error
EntityID	uint32	4	Request errors for specific EntityId (0 = all errors matching the other filters are sent). This filter does not affect on system errors
Options	uint32	4	Reserved for later use (0 = not defined)

Frame Length	Data Length	Min Length	Max Length
9	10	19	19

12.1.48. RESERVED (ID = 57)

12.1.49. RESERVED (ID = 58)

12.1.50. InitPositionToSymbolicPoint (ID = 59)

Initializes AGV position to symbolic point with given heading in degrees. This works for any symbolic point unlike InitializePosition (ID = 6) which only works for symbolic points marked as position initialization locations.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
SymbolicPointId	int32	4	Id of symbolic point. X,Y position of symbolic point will be used when initializing position.
MachineId	int32	4	Id of machine.
HeadingDegrees	float	4	Heading of AGV in degrees. In coordinate system 0 degrees is map east (right) and 90 degrees is map north (upwards).

Frame Length	Data Length	Min Length	Max Length
9	12	21	21

12.1.51. TrafficLightStatus (ID = 347)

Message is sent by MES to inform current status of a traffic light.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
TrafficLightID	uint32	4	ID of the traffic light
Status	uint16	2	ID of the group containing the trafficlight

Frame Length	Data Length	Min Length	Max Length
9	6	15	15

12.1.52. MESMessageSubscription (ID = 348)

Message is sent by MES to subscribe to different messages.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
Subscriptions	Subscription[]	0...x	Multiple subscriptions in array

Frame Length	Data Length	Min Length	Max Length
9	0...65526	9	65535

Subscription

Field Name	Type	Length	Description
SubscriptionType	uint16	2	Type of the subscription
SubscriptionData	Opaque	2..x	Contains data of different subscriptions, data depends on type chosen

Min Length	Max Length
4	65526

Subscription Datas

Different subscriptions listed.

TrafficLight (Type = 1)

Field Name	Type	Length	Description
Subscribe	uint16	2	Unsubscribe when set 0, else subscribe
SubscriptionToGroup	uint16	2	Subscription to single trafficlight ID when set 0, else to group ID
ID	uint16	2	Group or trafficlight ID
IntervalGreen	uint16	2	Interval to send request while requested status is green.
IntervalNone	uint16	2	Interval to send request while not requesting a status

Min Length	Max Length
10	10

ElevatorRequest (Type = 3)

Field Name	Type	Length	Description
Subscribe	uint16	2	0 = unsubscribe, 1 = subscribe requests
ElevatorId	uint16	2	Subscription to single ElevatorID, or all requests if set to 0
StatusRequestInterval	uint16	2	Interval in seconds how often ElevatorRequest should be sent to MES client when no machines are reserving the Elevator. Value 0 means at every cycle (1 second cycle).
RequestToFloorInterval	uint16	2	Interval in seconds how often ElevatorRequest with type RequestToFloor should be sent to MES client when machine has reservation for Elevator. Value 0 means at every cycle (1 second cycle).

Min Length	Max Length
8	8

ProductionStatus (Type = 4)

This subscription type can be used to request of sending ProductionStatus (ID = 313) message, and override default parameter `Send_Production_Status_Interval`.

Parameters specified in this message are global, so if two clients subscribe the message, the later subscriber will define the parameters for sending the message.

If any client has subscribed the message, global interval parameter is overridden, and only subscribed client will receive the message. If all clients have unsubscribed the message, the interval defined in global parameter is used again.

Field Name	Type	Length	Description
Subscribe	uint16	2	Subscribe message when set to 1, unsubscribe when 0
SubscriptionType	uint16	2	Different subscription options: 1 = Sending of message is completely disabled. 2 = OnlyPeriodically - Message is sent by the given interval. 3 = OnlyWhenChanged - Message is only sent if there are changes in the content. 4 = WhenChangedOrPeriodically - Message is sent by the given interval or when there are changes in the content.
RequestInterval	uint16	2	Interval in milliseconds how often message is sent. The process cycle is 1000ms, so message can be sent every 1000ms or more seldom.

Min Length	Max Length
6	6

12.1.53. ElevatorStatus (ID = 351)

Message is sent by MES client to publish the status of the single elevator. This message is sent as reply to message **ElevatorRequest** (ID = 352).

If *ElevatorId* is not found, Navithor will reply with **AckOrReject** (ID = 200) message with reject reason **SymbolicPointNotFound (4)**. If floors defined in Navithor Tools are not matching with the data in the message, the reject reason will be **BadInputData (1)**.

Note: By setting **StatusType** to 1, level fields in this message are handled as a type int32 and they support negative elevator levels.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
StatusType	uint16	2	Type of status message. 0 = default, 1 = use int32 fields instead.
ElevatorId	uint16	2	ID of the elevator (symbolic point group ID)
ElevatorStatus	uint16	2	1 = automatic, 2 = automatic and moving, 3 = manual, 4 = out of service
CurrentLevel	uint16/int32	2/4	Level where elevator is currently, or was previously if currently moving
DoorStatus	uint16	2	Door status at **CurrentLevel**: 1 = closed, 2 = open, 3 = broken and cannot be used (disabled)
TargetLevel	uint16/int32	2/4	Level where elevator is currently going to. Must equal to **CurrentLevel**, if not moving or target is not known.
TargetReachTime	uint16	2	Estimation to reach target level in seconds, or 0 if not moving. 0 if no estimation available.

Frame Length	Data Length	Min Length	Max Length
9	14/18	23/27	23/27

12.1.54. DeleteProductionOrder (ID = 353)

Sending this message with a mission external ID will delete the given mission. Same as MissionAbort.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MissionId	int32	4	Mission External ID, unique for each order

Frame Length	Data Length	Min Length	Max Length
9	4	13	13

12.1.55. SetSymbolicPointPriority (ID = 355)

Sending this message will set the priority of the symbolic point.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
SymbolicPointID	int32	4	ID of the symbolic point
Priority	int16	2	Desired priority between 1 and 8. Priority 1 is lowest.

Frame Length	Data Length	Min Length	Max Length
9	6	15	15

12.1.56. ValidateLoadReply (ID = 358)

Message used to reply to message ValidateLoad (ID = 357).

Field Name	Type	Length	Description
MessageContents	uint32	4	Bit flags of included message contents 1 << 0 = LoadValidationDataWithBarcode 1 << 1 = ValidationResult 1 << 2 = NewTransferTarget Bits 4 to 32 reserved for later use.

LoadValidationDataWithBarcode

Should be identical to the data sent in the ValidateLoad (ID = 357) request. Helps to determine which barcode is validated by the reply.

Field Name	Type	Length	Description
ProductionOrderID	uint32	4	OrderID, unique for each order. 0 if no linked order.
RequestingAgvID	uint16	2	ID of the AGV requesting the validation of the load.
SymbolicPointID	uint16	2	Symbolic point ID of the location of the load.
LoadTypeID	int32	4	ID of the load type.
BarcodeStringLength	uint16	2	Barcode string length.
Barcode	string	0...255	Barcode of the load. If not available, nothing is sent. Minimum length is 0, maximum 255.

ValidationResult

Field Name	Type	Length	Description
Result	uint32	4	Result of the validation: 0 = Success, continue with the pickup task. 1 = Error: Invalid barcode. 2 = Error: Invalid load type.

NewTransferTarget

Gives a new transfer target for the load that is being validated. Typically WMS decides the end target for each load. Production Order of the AGV is updated accordingly. If not given, AGV either continues to the old target, if such was given, or completes the pickup order and waits for a new order.

Field Name	Type	Length	Description
SymbolicPointId	uint16	2	ID of the symbolic point where the load needs to be transferred to (can be 0, if RackId is given instead)
ShelfId	uint16	2	Shelf ID for the height where the load needs to be transferred to (0 for non-rack symbolic points)
RackIdLength	uint16	2	Length of the rack identifier. 0 if no rack ID given.
RackId	string	0...255	Rack identifier, max 255 chars. Specifies rackId. If given, this will override the given symbolic point id and specifies desired side of the rack for side loaders.

Frame Length	Data Length	Min Length	Max Length
9	4 + Packet Lengths	13	547

12.1.57. StartStopProductionBreak (ID = 359)

Message is used to control production break feature of the Fleet Control. Activation of production break will make stop normal production activity and make the machines to drive into locations configured as production break locations.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
ProductionBreakMode	byte	1	0 = Normal mode, production break feature not active 1 = Production break active
ActivationReasonLength	uint16	2	Length of reason for activation, can be set to 0 when no reason specified
ActivationReason	string	0...200	Reason for break mode activation.

Frame Length	Data Length	Min Length	Max Length
9	3	12	212

12.1.58. ReturnToRoute (ID = 361)

Message is used to request machine to return to the route, if it is nearby the route. Successful action requires that the machine is currently either in STBY or IDLE state.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineId	int32	4	Machine ID to request to return to the route.

Frame Length	Data Length	Min Length	Max Length
9	4	13	13

12.1.59. ReleaseStopAndHold (ID = 362)

Message is used to release machine from StopAndHold symbolicpoint. This message can be used with either machine ID or symbolicpoint ID, both are not required.

Note: StopAndHold feature is a separate feature from HOLD, so this message should not be mixed with *ReleaseHold (ID = 18)*.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineId	int32	4	Machine ID to release, or `0` if symbolicpointId is used.
SymbolicpointId	int32	4	The ID of the symbolicpoint where machine is at StopAndHold state, or `0` if machineId is used.

Frame Length	Data Length	Min Length	Max Length
9	8	17	17

12.1.60. PivotCommand (ID = 363)

Message is used to request machine to pivot in place. Machine must be stationary in and STBY mode, but it does not need to be at the pivot point.

Message will be responded with PivotResult (ID = 364) once the message is forwarded to machine (Started) and once machine has completed the pivoting (Success).

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineId	uint16	2	Machine ID to pivot.
Direction	uint16	2	Direction of the rotation: 0 = Shortest angle 1 = Clockwise 2 = Counterclockwise
TargetHeading	float	4	Target heading in degrees after the pivoting. Global heading in AGV world is defined in range [-180, 180], so that zero-heading is pointing to right.

Frame Length	Data Length	Min Length	Max Length
9	8	17	17

12.1.61. ActivateMissionTemplate (ID = 10008)

Message is used to activate existing mission templates from the `MissionTemplates` folder at the Navithor Server root. Templates can be reloaded into server memory using **Reload overrides** functionality.

Message is responsed with a MissionTemplateActivationResult (ID = 10009) message.

Note: ExternalId and MachineId values provided in the message will overwrite the values in the mission template, which is loaded into memory.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
String Length	int16	2	
TemplateName	string	0..255	Template filename with or without `.json` extension.
String Length	int16	2	
ExternalId	string	0..255	External ID for mission to be created or empty string if value from mission template should be used.
MachineId	int16	2	Machine ID which should execute the mission or `0` if value from mission template should be used.

Frame Length	Data Length	Min Length	Max Length
9	8	15	525

12.2. Messages from Navithor to MES

12.2.1. VersionInfo (ID = 101)

Returns the version number of the communication protocol. Version of this document reflects the version of the communication protocol.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
InterfaceVersionMajor	uint16	2	Major number of the protocol version. When major number is changed the earlier protocol versions will be incompatible
InterfaceVersionMinor	uint16	2	Minor number of the protocol version
String Length	uint16	2	
software_version	string	0...100	Navithor software version (eg. "2.3.10.0"). Major.Minor.Branch.Revision

Frame Length	Data Length	Min Length	Max Length
9	6...106	15	115

12.2.2. TimeSet (ID = 103)

Time is set message.

Frame Length	Data Length	Min Length	Max Length
9	0	9	9

12.2.3. FSTOPDone (ID = 104)

Machine(s) have stopped and are in FSTOP-state.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineId	int16	2	-1 for all

Frame Length	Data Length	Min Length	Max Length
9	2	11	11

12.2.4. FSTOPReleased (ID = 105)

Machine(s) have returned to normal operation mode.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineId	int16	2	-1 for all

Frame Length	Data Length	Min Length	Max Length
9	2	11	11

12.2.5. DriveReady (ID = 302)

Machine has stopped driving.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineID	uint16	2	
X	double	8	
Y	double	8	
Heading	double	8	
Level	int32	4	
SymbolicPointId	uint16	2	If driving has finished at symbolic point SymbolicPointId > 0
ProductionOrderID	uint32	4	ID of the production order, or 0 when drive task was not from production order

Frame Length	Data Length	Min Length	Max Length
9	36	45	45

12.2.6. LoadPickupStarted (ID = 303)

Machine has started pickup task.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineID	uint16	2	

Frame Length	Data Length	Min Length	Max Length
9	2	11	11

12.2.7. LoadPickupComplete (ID = 304)

Machine has completed pickup task.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineID	uint16	2	

Frame Length	Data Length	Min Length	Max Length
9	2	11	11

12.2.8. LoadDropoffStarted (ID = 306)

Machine has started dropoff task.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineID	uint16	2	

Frame Length	Data Length	Min Length	Max Length
9	2	11	11

12.2.9. LoadDropoffComplete (ID = 307)

Machine has completed dropoff task.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineID	uint16	2	

Frame Length	Data Length	Min Length	Max Length
9	2	11	11

12.2.10. Errors (ID = 309)

This message is sent periodically from Navithor to MES. Message contains current error status in the system.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumberOfErrors	uint16	2	Max 1000 entries
ErrorDataList	ErrorData[]	NumberOfErrors * ErrorData length	ErrorData-struct below

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	65544

ErrorData

Error Data struct.

Field Name	Type	Length	Description
DescriptionStringLength	uint16	2	Length of the following Name-field. Limited by Server parameter: MES_MesMaxMsgInfoStringLen
Description	string	0...X	Error description with following format: ERROR_SOURCE.ENTITY.ERROR_TEXT Alarm source can be: "FleetControl", "NavigationSoftware" or "Supervisor" Entity is either name of the AGV that the error is related to or "SystemAlarm".
Id	uint16	2	Error ID
Value	uint16	2	Error level: 0 = info, 1 = warning, 2 = error, 3 = fatal error

Min Length	Max Length
6	6 + MES_MesMaxMsgInfoStringLen

12.2.11. AGVStatus (ID = 310)

Periodic status message from a single machine in the system. Sending interval can be changed from Navithor Server parameters: **Interval_To_Send_AGV_Status_To_MES_When_AGV_Enabled** and **Interval_To_Send_AGV_Status_To_MES_When_AGV_Disabled**. Parameter **MES_Disable_Outbound_Message_Groups_Rules** can be used to disable message completely.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineId	uint16	2	Machine ID
X	double	8	X-coordinate location of the machine
Y	double	8	Y-coordinate location of the machine
H	double	8	Machine heading
Level	int16	2	Current level of the machine
PositionConfidence	byte	1	Position confidence level of the machine (0-100)
SpeedNavigationPoint	double	8	Current speed of the machine
State	byte	1	Navitrol state (e.g. STBY = 2, AUTO = 3)
BatteryLevel	double	8	Current battery level of the machine (0-100)
AutoOrManual	byte	1	Control status (0=manual control active, 1 = automatic control active)
PositionInitialized	byte	1	1 = position is initialized, 0 = position is not initialized
LastSymbolPoint	int32	4	ID of the symbolic point where machine previously visited
MachineAtLastSymbolPoint	byte	1	1 = machine is at "LastSymbolPoint"-symbolic point, 0 = machine is not at the location
TargetSymbolPoint	int32	4	ID of the symbolic point that is the current target for the machine
MachineAtTarget	byte	1	1 = machine is at "TargetSymbolPoint", 0 = machine is not at the location
Operational	byte	1	1 if no errors, 0 if errors are stopping AGV
InProduction	byte	1	1 if in production, 0 if not in production
LoadStatus	byte	1	0 = unknown, 1 = empty, 2 = empty started, 3 = pickup started, 4 = full
Battery voltage	double	8	Battery voltage reported by machine
ChargingStatus	byte	1	0 = not charging, 1 = charging requested, 2 = charging
DistanceToTarget	float	4	Distance in meters to machine's current end target. **(Optional: enable by setting parameter MES_Protocol_Version_For_Status_Messages >= 3)**
CurrentDriveThroughPoint	int32	4	Current drive through point of the machine. If drive is cut to either level change, DecisionPoint or traffic light, reports the symbolic point ID of that location. If no drive through point is set, reports -1. **(Optional: enable by setting parameter MES_Protocol_Version_For_Status_Messages >= 4)**
NextLevelChangePointId	int32	4	Symbolic point ID of next level change point along the route. If no level change point along the route, -1 is sent. **(Optional: enable by setting parameter MES_Protocol_Version_For_Status_Messages >= 5)**

Field Name	Type	Length	Description
DistanceToNextLevelChange	float	4	Distance to next level change point. If no level change point assigned, 0 is sent. **(Optional: enable by setting parameter MES_Protocol_Version_For_Status_Messages >= 5)**
LastSymbolPointDrivenOver	int32	4	ID of the symbolic point that was last driven over, regardless if it was a target or not. **(Optional: enable by setting parameter MES_Protocol_Version_For_Status_Messages >= 6)**
NearSymbolicPoint	int32	4	ID of the symbolic point that we are currently within **Distance_For_AGV_To_Be_Considered_At_Target** from. AGV must not be moving and heading difference between AGV and symbolic point must also be within **Max_Agv_Heading_Difference_To_Starting_Segment**. Value -1 if not set. **(Optional: enable by setting parameter MES_Protocol_Version_For_Status_Messages >= 7)**

Frame Length	Data Length	Min Length	Max Length
9	70 [+ max 24]	79	103

12.2.12. ChargingRequest (ID = 311)

This message is sent when machine charging request status is changed (e.g. ChargeCommand-message (ID = 9) triggers this message with RequestType = 3).

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineId	uint16	2	Machine ID
RequestType	uint16	2	Charging request type: 0 = no request active, 1 = rules request, 2 = Navithor Client operator request, 3 = MES or PLC request (force), 4 = machine request, 5 = idle rule request

Frame Length	Data Length	Min Length	Max Length
9	4	13	13

12.2.13. InputValues (ID = 312)

Current input values sent by the machine.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineId	uint16	2	
DigitalInputlength	uint16	2	
DigitalInputs	byte[]	0...n	
IntegerInputLength	uint16	2	
IntegerInputs	int[]	0...n	
FloatInputLength	uint16	2	
FloatInputs	float[]	0...n	

Frame Length	Data Length	Min Length	Max Length
9	8 + Inputs	17	1000

12.2.14. ProductionStatus (ID = 313)

Periodic production status message sent by Navithor. Message is sent periodically defined by the interval parameter `Send_Production_Status_Interval`.

Message can be also subscribed using **MESMessageSubscription** (ID = 348) message with subscription type 4 (ProductionStatus). This allows only specific clients to receive the message with specific interval, or when content of the message has changed.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
numberOfProductionOrders	uint16	2	Number of ProductionOrderData-structures in this message. Max 200
ProductionOrderData	ProductionOrderData[]	numberOfProductionOrders * ProductionOrderData length	Production Order Data structure below

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	44411

Production Order Data

Production order data struct containing information from a single production order.

Field Name	Type	Length	Description
NameStringLength	uint16	2	Max 200 characters
Name	string	0...200	Production Order Name
Id	uint32	4	Production order ID
TargetSymbol	int32	4	
AssignedToMachineId	uint16	2	0 if not assigned, machine ID if assigned machine
PickupSymbol	int32	4	-1 if no pickup symbol
ItemTypeToDeliver	int32	4	-1 if no items
CurrentStatus	byte	1	new_order = 0, loaded = 1, waiting = 2, executing = 3, interrupted = 4, completed = 5, cancelled = 6, paused = 7
ExecutionStatus	byte	1	none = 0, drive to pickup = 1, at pickup = 2, picking up = 3, picked up = 4, driving to target = 5, at target = 6, dropping off = 7, dropped off = 8, at hold = 9

Min Length	Max Length
22	222

12.2.15. ProductionOrderCompleted (ID = 314)

This message is sent by Navithor when machine has completed a production order.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NameStringLength	uint16	2	Max 200 characters
Name	string	0...200	Production Order Name
ID	uint32	4	Unique ID of the production order

Frame Length	Data Length	Min Length	Max Length
9	4 + Name length	15	215

12.2.16. ProductionOrderListActivatedResult (ID = 315)

Navithor replies to message ActivateProductionOrderList with this message.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
POLId	uint32	4	ID of the Production Order List template that was activated
OrderID	uint32	4	ID of the created active POL -> 0 if failed
MachineID	int32	4	ID of the machine that is requested to execute the order or less than zero when any machine was requested

Frame Length	Data Length	Min Length	Max Length
9	12	21	21

12.2.17. ProductionOrderListCompleted (ID = 316)

Navithor replies to message ActivateProductionOrderList with this message.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
OrderID	uint32	4	ID of the completed Production Order List
ExecutionTime	uint32	4	Seconds
TotalDist	double	8	Meters travelled during execution of this order

Frame Length	Data Length	Min Length	Max Length
9	16	25	25

12.2.18. ActiveProductionOrderListStatus (ID = 317)

This message is sent when changes in statuses are reported or periodically if no changes detected.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumberOfActiveLists	uint16	2	Number of currently active (waiting for assign, assigned to machine) ProductionOrderLists in Navithor. Max 1000
POLDataList	POLData[]	NumberOfActiveLists * POLData length	List of ProductionOrderList-data structures

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	49011

POLData

Production Order List Data structure containing information of a single POL (active).

Field Name	Type	Length	Description
NameStringLength	uint16	2	Max 30 characters
POLName	string	0...30	Production Order List Name (max length = 30 chars)
POLId	uint32	4	Production order List ID
OrderID	uint32	4	OrderID generated by Navithor when a template list was activated
AssignedToMachineId	int32	4	-1 if not assigned, machine ID if assigned machine
POLStatus	byte	1	0 = none, 1 = loaded (waiting), 2 = active, 3 = cancelled, 4 = completed
CurrentTargetIndex	int16	2	-1 if not started, 0-> index of the Production Order List location that is the current target
NumberOfTargets	uint16	2	Total number of locations in the POL

Min Length	Max Length
19	49

12.2.19. ProductionOrderListTemplateResult (ID = 318)

This message is as a reply to RequestProductionOrderListTemplates.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumberOfTemplates	uint16	2	Number of templates loaded in Navithor. Max 100
POLTemplateData	POLTemplateData[]	NumberOfTemplates * POLTemplateData length	List of ProductionOrderList-template data structures

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	23811

POLTemplateData

Production Order List Template Data structure containing information of a single POL Template.

Field Name	Type	Length	Description
NameStringLength	uint16	2	Max 30 characters
POLName	string	0...30	Production Order List Name (max length = 30 chars)
POLId	uint32	4	Production order List ID
NumberOfDestinations	uint16	2	Number of destinations in execution order. Max 100
DestinationID	uint16[]	NumberOfDestinations * 2	SymbolicPointID of the destination

Min Length	Max Length
8	238

12.2.20. TagStatusRequest (ID = 319)

Navithor can request a certain status for a tag that is attached to production area entity. Currently supported entities are:

- Symbolic Point of type 'Door'

When controlled tags are defined for the production system Navithor sends this message with regular interval and when status change is required. MES should answer to this message with TagStatus (ID = 24).

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
TagID	uint32	4	Tag ID defined in Navithor Tools for Symbolic Point
RequiredStatus	int32	4	Required status for entity linked to the tag. Defined values: 0 = OFFLINE (not requested) 1 = DOOR CLOSED (default request) 2 = DOOR CLOSING (not requested) 3 = DOOR OPENING (not requested) 4 = DOOR OPEN (when machine needs pass-through)

Frame Length	Data Length	Min Length	Max Length
9	8	17	17

12.2.21. RequestPickUpResult (ID = 320)

Navithor's answer to RequestPickUp message (ID = 25).

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
Symbolic Point ID	uint32	4	Symbolic point ID defined in Navithor Tools for Symbolic Point
Result	uint32	4	Result of RequestPickUp message handling. Possible results: 0 = no error 1 = bad input data, invalid combination of Symbolic Point ID and Shelf ID 2 = no load at the specified location 4 = symbolic point with specified ID was not found Error ID (See chapter 9.5 "Error IDs")

Frame Length	Data Length	Min Length	Max Length
9	8	17	17

12.2.22. MESResourceStatus (ID = 321)

Periodic load status message sent by Navithor.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumberOfLoads	uint16	2	Number of LoadData-structures in this message. Max 500
LoadDataList	LoadData[]	NumberOfLoads * LoadData length	LoadData structure below

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	10011

LoadData

Load data struct containing information from a single load

Field Name	Type	Length	Description
LoadId	int32	4	Unique ID for the load
LoadTypeId	int32	4	Load type ID
LoadQuantity	uint16	2	Load quantity
LoadStatus	uint16	2	0 = not defined, 1 = at symbolic point, 2 = onboard of AGV
LocationID	int32	4	-1 if not defined, otherwise either SymbolicPointId or MachineId depending on the load status given above
ProductionOrderId	int32	4	-1 if no production order linked to the load

Min Length	Max Length
20	20

12.2.23. RequestResourceStatusAtLocation (ID = 322)

Navithor sends this message to request a specific load status to a location. The request can be:

- Navithor asks for a specific load type to be added to a specific symbolic point
  - Use case: Transfer request is linked to a pickup location but there is no load at the pickup location. The transfer request cannot proceed until the correct load is added to the symbolic point
- Navithor requests that a load must be removed from a specific symbolic point
  - Use case 1: dropoff location needs to be set empty so that next AGV arriving to the location can drop off its load
  - Use case 2: pickup location has wrong load type compared to the load type of transfer request

This message is not automatically sent by default. Enabling requires a mapping file configuration for Navithor to understand the correct MES client IDs that handles the request(s). Mapping file:

- File 'LocationMESRequestLinkings.txt' must be found in 'Overrides'-subfolder of Navithor Server's running folder
- One configuration per line (Symbolic Point ID:MES ID). E.g:
  - 1:1001
    - Symbolic Point 1 is monitored for required load status and RequestResourceStatusAtLocation-message will be sent to MES Client that is identified as 1001

Parameters **Request_Load_At_Location_Interval** and **Request_Load_At_Location_Acknowledged_Interval** affect the periodic interval of sending this message. The request is also sent in case the data in the message gets changed.

MES should reply with message ID = 201 AckOrRejRequest. Acknowledging the request does not create any changes in the production. The request can be fulfilled only by changing the load status at needed symbolic point(s). This can be done with messages SetStatusAtSymbolicPoint (ID = 20), SetResourceAtLocationWithExtendedInfo (ID = 53) or SetResourcesAtLocation (ID = 32).

MES Client can also reject the request with AckOrRejRequest. Rejecting a request will either cause the related transfer request to be postponed or cancelled. See section *Error IDs* for information about reason codes. Postponing means the transfer request will unlink from the symbolic point and become inactive for a specified time (parameter **Transfer_Request_Timeout_After_Postpone**) until it can link to a symbolic point again.

This message is supported in Navithor version 1.9.59.X -> MES interface updated to version 1.6.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
RequestID	uint16	2	A running number to identify reply to this specific request
SymbolicPointID	uint16	2	ID of the symbolic point
CurrLoadTypeAtLoc	int32	4	Type ID of the load currently at symbolic point. 0 if location is empty
CurrLoadCountAtLoc	uint16	2	Number of loads currently at the location. 0 if location is empty
CurrPickUpReqStatus	uint16	2	Current Pickup request-status: 0 = No request 1 = Pickup request active 2 = Pickup request active and AGV routed to location
RequestedLoadTypeID	int32	4	Type ID of the load needed at the location
RequestedLoadCount	uint16	2	Number of loads requested to be at location
RequestReason	uint16	2	Navithor has reason for the request: 0 = Production Request. E.g. transfer request received but no load at pickup location 1 = AGV is currently picking up current load 2 = AGV is currently dropping a load to this location 100 = Error – wrong load type at symbolic point 200 = Error – delivery of the current load is not possible 300 = Error – AGV fails to pick up the current load at the location 301 = Error – AGV fails to drop off the load to a location
ReqResIdentifierLen	uint16	2	Length of following ReqResIdentifier-string field. Max len = 256 characters
ReqLoadIndentifier	string	0...255	Request_Load_At_Location_Acknowledged_Interval identifier number or code
RequestInfoStrLen	uint16	2	Length of following RequestInformation-string field. Max len = 256 characters
RequestInformation	string	0...255	Information from Navithor about the request

Frame Length	Data Length	Min Length	Max Length
9	24 + ReqLoadIdentifierLen + RequestInfoStrLen	33	543

12.2.24. TransferRequestStatus (ID = 323)

If TransferRequest message was sent with a RequestID, Navithor sends this message with the same RequestID when the status of TransferRequest changes. The message is sent first time, when production order is created and linked to the transfer request.

This message is supported in Navithor version 2.0.2.X -> MES interface updated to version 2.4.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
RequestID	uint32	4	ID sent by the planner in TransferRequest message to identify reply to that specific request
ProductionOrderID	uint32	4	Production order ID linked to the transfer request
TransferStatus	uint16	2	None = 0, waiting pickup = 1, assigned to machine = 2, transporting = 3, dropped off = 4, cancelled = 6
MachineID	uint32	4	0 = not assigned to any machine, otherwise tells the machine ID
PickupLocationID	uint16	2	ID of the pickup location (symbolic point or group). **(Optional: enable by setting parameter MES_Protocol_Version_For_Status_Messages >= 2)**
TargetLocationID	uint16	2	ID of the current target location (symbolic point or group). **(Optional: enable by setting parameter MES_Protocol_Version_For_Status_Messages >= 2)**
PickupIDType	byte	1	0 = PickupLocationID-field points to symbolic point ID 1 = PickupLocationID-field points to symbolic point group ID **(Optional: enable by setting parameter MES_Protocol_Version_For_Status_Messages >= 2)**
TargetIDType	byte	1	0 = TargetLocationID-field points to symbolic point ID 1 = TargetLocationID-field points to symbolic point group ID **(Optional: enable by setting parameter MES_Protocol_Version_For_Status_Messages >= 2)**
OrderedAmount	uint16	2	Total amount of loads to be delivered in the order. **(Optional: enable by setting parameter MES_Protocol_Version_For_Status_Messages >= 2)**
AGVsTransporting	uint16	2	Total amount of AGVs currently transporting loads. **(Optional: enable by setting parameter MES_Protocol_Version_For_Status_Messages >= 2)**
DeliveredCount	uint16	2	Amount of loads delivered so far. **(Optional: enable by setting parameter MES_Protocol_Version_For_Status_Messages >= 2)**

Frame Length	Data Length	Min Length	Max Length
9	14 [+ 12]	23	35

12.2.25. SymbolicPointGroupStatus (ID = 324)

This message is sent periodically if enabled by the server parameter Send_Symbolic_Point_Group_Status_Interval.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
SympointGroupID	uint16	2	ID of the SymbolicPointGroup
NumberOfSyms	uint16	2	Number of enabled Symbolic Points in this group
NumberOfAvailableSyms	uint16	2	Number of Symbolic Points in this group that are available for loads
MaxLoadCount	uint16	2	Total number of loads possible for Symbolic Points in this group
CurrLoadCount	uint16	2	Current number of loads at Symbolic Points of this group

Frame Length	Data Length	Min Length	Max Length
9	10	19	19

12.2.26. GroupedTagStatusRequest (ID = 325)

Navithor can request a certain status for a tag that is attached to production area entity. Currently supported entities are:

- Symbolic Point of type 'Door'

When controlled tags are defined for the production system Navithor sends this message with regular interval and when status change is required. MES should answer to this message with TagStatus (ID = 24) or with GroupedTagStatus (ID = 35).

This message is only sent if **Enable_Grouping_Tag_Statuses_To_Single_Message** parameter is set to true. Available from Navithor server versions 2.1.2.0 onwards.

Field Name	Type	Length	Description
Number of tags	uint16	2	Number of tags in this request
Requests for tags	TagData[]	Number of tags * TagData length	

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	4011

TagData

Data structure for a single machine type.

Field Name	Type	Length	Description
TagID	uint16	2	Tag ID defined in Navithor Tools for Symbolic Point
TagStatus	uint16	2	Status of the tag. Defined values: 0 = OFFLINE 1 = DOOR CLOSED 2 = DOOR CLOSING 3 = DOOR OPENING 4 = DOOR OPEN

Min Length	Max Length
4	4

12.2.27. ActiveFireAlarmStatus (ID = 326)

Navithor sends this message periodically for each active fire alarm.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
FireAlarmID	uint32	4	ID of the active fire alarm
MachineCount	uint16	2	Number of machines that are still in the zones belonging to the alarm

Frame Length	Data Length	Min Length	Max Length
9	6	15	15

12.2.28. StoppedAtStopPoint (ID = 328)

Machine has stopped driving.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineID	uint16	2	
StopPointIndex	int16	2	The index of the current point where AGV stopped. Might be used for sending notification when light curtain triggered with negative index
ProductionOrderID	int32	4	ID of the production order, or 0 when drive task was not from production order

Frame Length	Data Length	Min Length	Max Length
9	8	17	17

12.2.29. EstimationOfDriveToStartSymbolicPoint (ID = 329)

Navithor will reply with this message to valid EstimateDriveToStartSymbolicPoint message.

Navithor replies with driving distances and times in the ascending order of the SortingField specified in EstimateDriveToStartSymbolicPoint. The distances and times are calculated to the start symbolic point for each possible machine.

Estimation of the driving distances and times are only possible for enabled machines in production. If some machine is not enabled, or is not in production, or is executing another order, it will not appear in the estimation results.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumberOfMachines	uint16	2	Number of machines, from 0 to 200
MachineIds	uint16[]	NumberOfMachines * 2	Array of machine IDs
DrivingDistances	float[]	NumberOfMachines * 4	Estimated driving distances in meters
DrivingTimes	uint16[]	NumberOfMachines * 2	Estimated driving times in seconds

Frame Length	Data Length	Min Length	Max Length
9	2 + NumberOfMachines * 8	11	1611

12.2.30. RackSetupReply (ID = 330)

This message will be sent by Navithor back to the same MES client as a reply to its GetRackSetup message.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumberOfSps	int32	4	If GetRackSetup message had positive MachineTypeId, then this field contains the number of symbolic points which are racks with defined MachineTypeId. Else, this field contains the number of all symbolic points which are racks

Frame Length	Data Length	Min Length	Max Length
9	4	13	13

12.2.31. RackSetup (ID = 331)

This message will be sent by Navithor back to the same MES client as an additional part of the reply to its GetRackSetup message.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
SpId	int32	4	Symbolic point ID which is a rack with defined MachineTypeId
MachineTypeId	int32	4	Machine type ID
NumberOfShelves	uint16	2	Number of shelves in this rack (maximum is 100)
Shelves	<int32, int32> []	0...800	Shelves passed as list of tuples containing <Shelf ID, Actuator height (mm)>. Number of the list elements is equal to NumberOfShelves

Frame Length	Data Length	Min Length	Max Length
9	10 + length of shelves	19	819

12.2.32. SymbolicPointGroupSetup (ID = 332)

This message will be sent by Navithor back to the same MES client as an answer to the GetSymbolicPointGroupSetup message.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumOfGroups	uint16	2	Number of symbolic point groups in the system
SymbolicPointGroups	SymbolicPointGroupData[]	NumOfGroups * SymbolicPointGroupData length	Array of SymbolicPointGroupData structs containing information about each group. Data defined below

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	6011

SymbolicPointGroupData

Data structure for a single symbolic point group.

Field Name	Type	Length	Description
Group ID	uint16	2	Symbolic point group ID
NameStringLength	uint16	2	Max 100 characters
Name	string	0...100	Name of the symbolic point group
NumberOfSPs	uint16	2	Number of Symbolic Points in this group (SPN)
SympointID	uint16[]	2 * SPN	Symbolic point IDs as array

Min Length	Max Length
6	600

12.2.33. ZoneStatusReply (ID = 333)

Reply to the Request zone status message (ID = 51).

Valid from Navithor 2.20.0.0 forward.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
Zone ID	uint32	4	The ID of the zone
Requested information	uint16	2	Flags that were set in the request so that the Client can read the message: ZoneOccupancy status 1 << 0 AGVs inside zone (List of AGV ids) 1 << 1
Zone occupancy status	uint16	2	1 if occupied and 0 if no AGVs inside the zone. **(Optional: is only sent when Requested information flags have first bit set to 1 (1 << 0))**
Number of AGVs inside zone	uint16	2	How many AGVs are inside the zone. Denoted as N **(Optional: is only sent when Requested information flags have second bit set to 1 (1 << 1))**
AGV IDs of the AGVs inside	uint32[]	N * 4	The IDs of the AGVs inside the zone. **(Optional: is only sent when Requested information flags have second bit set to 1 (1 << 1))**

Frame Length	Data Length	Min Length	Max Length
9	6 [+ 2] [+ 2 + N * 4]	15	1617

12.2.34. MesBatteryTemperatureWarning (ID = 334)

This message is sent when machine battery reports too high temperature.

Field Name	Type	Length	Description
MachineID	int32	4	MachineID
PositionX	float	4	Machine X position
PositionY	float	4	Machine Y position

Frame Length	Data Length	Min Length	Max Length
9	12	21	21

12.2.35. ProductionAreaChanged (ID = 335)

Message will be sent by Navithor when production area has been changed or reloaded. Will be also sent at startup.

Available from Navithor server versions 2.21.0.0 onwards.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
ProductionAreaID	uint16	2	The ID of the production area
ProductionAreaVersion	uint16	2	The version number of the production area
NameStringLength	uint16	2	Length of the Production Area Name. Max 128 characters
ProductionAreaName	string	0...128	Production Area Name. Max length = 128 characters

Frame Length	Data Length	Min Length	Max Length
9	6 + ProductionAreaName length	15	143

12.2.36. MachineTypeInformationReply (ID = 336)

Message sent as reply to GetProductionAreaInformation (ID = 52). Contains information about currently active machine types.

Available from Navithor server versions 2.21.0.0 onwards.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumberOfMachineTypes	uint16	2	How many MachineTypeInfo data structures are included
Information	MachineTypeInfo[]	NumberOfMachineTypes * MachineTypeInfo length	Machine type information

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	65535

MachineTypeInfo

Machine type information.

Field Name	Type	Length	Description
ID	int32	4	ID of machine type
NameStringLength	uint16	2	Length machine type name
Name	string	0...50	Name of machine type

Min Length	Max Length
6	56

12.2.37. MachineInformationReply (ID = 337)

Message sent as reply to GetProductionAreaInformation (ID = 52). Contains information about currently active machines.

Available from Navithor server versions 2.21.0.0 onwards.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumberOfMachineEntries	uint16	2	How many MESMachineInformation data structures are included
Information	MESMachineInformation[]	NumberOfMachineEntries * MESMachineInformation length	Machine information

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	65535

MESMachineInformation

Machine information.

Field Name	Type	Length	Description
ID	int32	4	ID of machine
TypeId	int32	4	ID of machine type
NameStringLength	uint16	2	Length machine name
Name	string	0...50	Name of machine

Min Length	Max Length
10	60

12.2.38. SymbolicPointInformationReply (ID = 338)

Message sent as reply to GetProductionAreaInformation (ID = 52). Contains information about currently active symbolic points.

Available from Navithor server versions 2.21.0.0 onwards.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumberOfSymbolicpoints	uint16	2	How many MESSymbolicPointInformation data structures are included
Information	SPInformation[]	NumberOfSymbolicpoints * SPInformation length	Symbolic point information

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	65535

SPInformation

Symbolic point information. Max length can be determined from 62 + (number of machine types in production area * 4).

Field Name	Type	Length	Description
ID	int32	4	ID of symbolic point
Type	int32	4	Type of symbolic point
AvailableForMachineTypesCount	uint16	2	How many machine types are valid for this symbolic point
AvailableForMachineTypes	int32[]	AvailableForMachineTypesCount * 4	ID's of machine types that are valid for this symbolic point
NameStringLength	uint16	2	Length of name
Name	string	0...50	Name of symbolic point

Min Length	Max Length
12	102

12.2.39. SymbolicPointHoldInformationReply (ID = 339)

Message sent as reply to GetProductionAreaInformation (ID = 52). Contains hold information of currently active symbolic points.

Available from Navithor server versions 2.21.0.0 onwards.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumberOfHoldEntries	uint16	2	How many HoldInformation data structures are included
Information	HoldInformation[]	NumberOfHoldEntries * HoldInformation length	Symbolic point hold information

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	65535

HoldInformation

Symbolic point hold information.

Field Name	Type	Length	Description
Id	int32	4	ID of symbolic point
MachineTypeId	int32	4	For which machine type this hold information is for
DefaultReleaseLocation	int32	4	ID of default destination for hold-release. -1 if not defined
Alt1Location	int32	4	ID of first alternative location where AGV will be released to if default location is not available. -1 if not defined
Alt2Location	int32	4	ID of second alternative location where AGV will be release to if first alternative is not available. -1 if not defined
Alt3Location	int32	4	ID of third alternative location where AGV will be released to if second alternative is not available. -1 if not defined
ReleaseTimeOutSecond	uint32	4	How many seconds before AGV will be automatically released from hold location. If not set = 4294967295 (uint32 max value)

Min Length	Max Length
28	28

12.2.40. SymbolicPointSpatialInformationReply (ID = 340)

Message sent as reply to GetProductionAreaInformation (ID = 52). Contains spatial information of currently active symbolic points.

Available from Navithor server versions 2.21.0.0 onwards.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumberOfEntries	uint16	2	How many SymbolicPointSpatialInformation data structures are included
Information	SymbolicPoint SpatialInformation[]	NumberOfEntries * SymbolicPoint SpatialInformation length	Symbolic point spatial information

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	65535

SymbolicPointSpatialInformation

Symbolic point spatial information.

Field Name	Type	Length	Description
Id	int32	4	ID of symbolic point
X	double	8	X coordinate of symbolic point
Y	double	8	Y coordinate of symbolic point
Z	double	8	Z coordinate of symbolic point Not in use
Level	int32	4	Which level/floor this symbolic point is located at. Not in use

Min Length	Max Length
32	32

12.2.41. ResourceTypeInformationReply (ID = 341)

Message sent as reply to GetProductionAreaInformation (ID = 52). Contains information about currently active load types.

Available from Navithor server versions 2.21.0.0 onwards.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumberOfEntries	uint16	2	How many LoadTypeInformation data structures are included
Information	LoadType Information[]	NumberOfEntries * LoadTypeInformation length	Load type information

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	65535

LoadTypeInformation

Load type information.

Field Name	Type	Length	Description
Id	int32	4	ID of load type
NameStringLength	uint16	2	Length of name
Name	string	0...50	Name of load type

Min Length	Max Length
6	56

12.2.42. ResourceTypeOffsetInformationReply (ID = 342)

Message sent as reply to GetProductionAreaInformation (ID = 52). Contains offset information of currently active load types.

Available from Navithor server versions 2.21.0.0 onwards.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumberOfOffsetEntries	uint16	2	How many MESLoadTypeOffsetInformation data structures are included
Information	MESLoadType OffsetInformation[]	NumberOfOffsetEntries * MESLoadType OffsetInformation length	Load type offset information

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	65535

MESLoadTypeOffsetInformation

Load type offset information.

Field Name	Type	Length	Description
Id	int32	4	ID of load type
AppliesForMachineType	int32	4	ID of machine type these offsets are for
OffsetX	double	8	X offset
OffsetY	double	8	Y offset
OffsetZ	double	8	Z offset. Not used
RotationX	double	8	Rotation around X axis. Not used
RotationY	double	8	Rotation around Y axis Not used
RotationZ	double	8	Rotation around Z axis
ScaleX	double	8	Scale on X axis
ScaleY	double	8	Scale on Y axis
ScaleZ	double	8	Scale on Z axis

Min Length	Max Length
80	80

12.2.43. MachineLoadCapacityInformationReply (ID = 343)

Message sent as reply to GetProductionAreaInformation (ID = 52). Contains load capacity information for currently active machines.

Available from Navithor server versions 2.21.0.0 onwards.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumberOfEntries	uint16	2	How many MachineLoadCapacityInformation data structures are included
Information	Machine LoadCapacityInformation[]	NumberOfEntries * Machine LoadCapacityInformation length	Machine load capacity information

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	65535

MachineLoadCapacityInformation

Machine load capacity information.

Field Name	Type	Length	Description
MachineId	int32	4	ID of machine
LoadtypeId	int32	4	ID of load type
Capacity	int32	4	How many of given load type AGV is allowed to carry at once

Min Length	Max Length
12	12

12.2.44. SymbolicPointLoadCapacityInformationReply (ID = 344)

Message sent as reply to GetProductionAreaInformation (ID = 52). Contains load capacity information for currently active symbolic points.

Available from Navithor server versions 2.21.0.0 onwards.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumberOfEntries	uint16	2	How many MachineLoadCapacityInformation data structures are included
Information	SymbolicPoint LoadCapacityInformation[]	NumberOfEntries * SymbolicPoint LoadCapacityInformation length	Symbolic point load capacity information

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	65535

SymbolicPointLoadCapacityInformation

Symbolic point load capacity information.

Field Name	Type	Length	Description
SymbolicPointId	int32	4	ID of symbolic point
LoadtypeId	int32	4	ID of load type
Capacity	int32	4	How many of given load type symbolic point is allowed to hold at once

Min Length	Max Length
12	12

12.2.45. SymbolicPointResourceStatus (ID = 345)

Reply to the RequestSPResourceStatus message (ID = 55).

Valid from Navithor 2.21.0.0.

Included since Navithor version 2.51.0.0, if you create an empty text file *Navithor Server/Overrides/SendRequestFlagsInResourceStatusReply.txt* and reload the override files, this message will contain *RequestFlags* field as two extra bytes after the frame. These flags match with the requested flags sent in the request this message is replying to. In this case minimum length of the message is 13 bytes.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
SymbolicPointCount	uint16	2	Count of for how many symbolic points the status will be returned. Maximum amount is 400.
LoadData	MESLoadData WithIdentifier[]	SymbolicPointCount * MESLoadData WithIdentifier length	MESLoadDataWithIdentifier data structure

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	65544

MESLoadDataWithIdentifier

MESLoadDataWithIdentifier.

Field Name	Type	Length	Description
SymbolicPointId	uint32	4	Symbolic point ID where status was requested
SPStatus	uint16	2	Status of the symbolic point. 0 = Disabled, 1 = Enabled
NoOfLoads	uint16	2	The value of this field depends on the bits enabled in the RequestFlags field of the RequestSPResourceStatus message. - Bit 3 = 0: The value is the total number of loads at the location. - Bit 3 = 1: The value is the number of items for the individual load.
LoadType	uint16	2	Load type ID. Returned as 0, if no loads at SP. If multiple load types, only the last type is returned
LoadIdentifierLength	uint16	2	Length of the load identifier. 0 if load identifier is not defined. If multiple load identifiers found, only the last is returned.
LoadIdentifier	string	0...ResIdentifierLength	Load identifier. Maximum length is limited to 100.
DateStringLength	uint16	2	**Optional field, send only if requested** Length of the date string. If no load present, length is 0, otherwise 14
LoadCreationDate	String	0-14	**Optional field, send only if requested** Format: Year, month, day, hours, minutes, seconds (yyyymmddhhmmss) using 24-hour UTC time (not adjusted to local time zone). If location has multiple loads, the creation date of last load added to the symbolic point is given.
LoadShelf	uint16	2	**Optional field, send only if requested**. Load shelf index, if load is located in rack or in load buffer. In buffer lane, the shelf index represents the load y-position from the ground, so that loads on the ground have shelf index 0, and if load is stacked, it will have value 1 or higher. To be used with a requestFlag *AllLoads*.
LoadSlot	uint16	2	**Optional field, send only if requested**. Load slot index, if load is located on buffer lane. Slot index 0 is closest to symbolic point. To be used with a requestFlag *AllLoads*.
LoadSide	uint16	2	**Optional field, send only if requested**. Load side, if load is located on rack with different sides. 0 is no side defined, 1 is left side, 2 is right side. To be used with a requestFlag *Side*.

Min Length	Max Length
12	134

12.2.46. TrafficLightStatusRequest (ID = 346)

Message is sent by Navithor to inform requested status for a trafficlight.

Field Name	Type	Length	Description
TrafficLightID	uint32	4	ID of the traffic light
GroupID	uint32	4	ID of the group containing the trafficlight
RequestedStatus	uint16	2	0 = None, 1 = Green, 2 = Yellow, 3 = Red
Priority	int16	2	Highest priority of the AGVs in queue
QueueCount	uint16	2	Amount of AGVs queued behind this light within the trafficlight distance
PassCount	uint16	2	Amount of AGVS passing the intersection that have passed this light.

Frame	Data Length	Min Length	Max Length
9	16	25	25

12.2.47. ErrorsV2 (ID = 349)

This message is sent as a response to ErrorsV2 request (ID = 56).

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NumberOfErrors	uint16	2	Max 1000 entries, or number of entries that can fit the message frame
ErrorData	ErrorDataV2[]	NumberOfErrors * ErrorDataV2 length	ErrorDataV2-struct below. Length of individual data entry will vary

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	65544

ErrorDataV2

Error Data V2 struct.

Field Name	Type	Length	Description
Id	uint16/uint32	2-4	Error ID. NOTE: **Error ID is not globally unique**, instead it depends on the source of the error. Meaning that error from the Supervisor can have the same ID as error from the Fleet Control. **By default ID is sent as uint16. If MES_Protocol_Version_For_Status_Messages is set to at least 8, ID is sent as uint32.**
NameStringLength	uint16	2	Length of the following Name-field. Limited by Server parameter: MESStringFieldFillMaxLen
Name	string	0...X	Error name / identifier
ErrorType	uint16	2	Unknown = 0, SystemAlarm = 1, EntityAlarm = 2, MachineAlarm = 3, SymbolicPointAlarm = 4, WaypointAlarm = 5
EntityID	uint32	4	0 = not defined, otherwise tells entity ID. EntityID reflects the ID of the machine (ErrorType = 3) or symbolic point (ErrorType = 4), depending on what the errorType is. If errorType is System alarm, then EntityID is 0.
Source	byte	1	Unknown = 0, FleetControl = 1, Machine = 2, Supervisor = 3, Component = 4, MES = 5
Level	byte	1	0 = info, 1 = warning, 2 = error, 3 = fatal error
Value	uint16	2	Error value
Priority	int16	2	Error priority

Min Length	Max Length
16	18 + MESStringFieldFillMaxLen

12.2.48. ElevatorRequest (ID = 352)

Message is sent by Navithor to specific MES client to request **ElevatorStatus** (ID = 351) message or to request elevator to specific floor. Both requestTypes should be always replied with **ElevatorStatus** message.

MES Client which is responsible for controlling the specific elevator should subscribe these messages by sending **MESMessageSubscription** (ID = 348) message to Navithor once connection is established.

Field Name	Type	Length	Description
RequestType	uint16	2	Type of the request: 0 = status request, 1 = request to drive to given *RequestToFloor*.
ElevatorId	uint16	2	ID of the elevator (symbolic point group ID)
RequestId	uint16	2	ID of the request to distinguish different requests (currently always 0).
-	-	-	Following data is sent only if RequestType = 1.
MachineId	uint16	2	ID of the machine which is going to use elevator first.
RequestToFloor	uint16	2	Floor ID where elevator is required. ID should match to ones defined in the Navithor Tools.
MachineTargetFloor	uint16	2	Target floor ID which machine is supposed to reach. If elevator has only two levels, this can be ignored, as machine will always change the level.
Priority	uint16	2	Priority of the request, value between 1 and 7. Default value is MED priority (4).

Frame	Data Length	Min Length	Max Length
9	6-14	15	23

12.2.49. TransferRequestReply (ID = 356)

Response that is sent after receiving a transfer request message to signal success of creation.

More specific statuses describing reason for failure may be implemented at a later date.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
RequestID	uint32	4	ID sent by the planner in TransferRequest message to identify reply to that specific request. 0 if ID was not given.
Status	uint16	2	1 = Success, 2 = Failure

Frame Length	Data Length	Min Length	Max Length
9	6	15	15

12.2.50. ValidateLoad (ID = 357)

Message is used by Fleet Control to request load verification from the MES Client (ie. WMS of the host).

The message includes the data fields that needs to be matched with the load to be picked up.

There can be multiple sources for this data, ie. MES interface or configuration file in Navithor or Navitrol/Supervisor if sent by Navitrol to Navithor.

Navithor expects to receive the message ValidateLoadReply (ID = 358) in response to this message. If no reply is received, Navithor periodically once per second requests the validation again.

The response can either indicate success or error. A successful respond can also contain a new transfer target for the load.

Field Name	Type	Length	Description
MessageContents	uint32	4	Bit flags of included message contents 1 << 0 = RequestLoadValidationWithBarcode 1 << 1 = RackData (sent if the load is on a rack) Bits 3 to 32 reserved for later use.

RequestLoadValidationWithBarcode

Field Name	Type	Length	Description
ProductionOrderID	uint32	4	OrderID, unique for each order. 0 if no linked order.
RequestingAgvID	uint16	2	ID of the AGV requesting the validation of the load.
SymbolicPointID	uint16	2	Symbolic point ID of the location of the load.
LoadTypeID	int32	4	ID of the load type.
BarcodeStringLength	uint16	2	Barcode string length.
Barcode	string	0...255	Barcode of the load. If not available, nothing is sent. Minimum length is 0, maximum 255.

RackData

Field Name	Type	Length	Description
ShelfId	uint16	2	Shelf ID of the pickup location.
RackIdLength	uint16	2	Length of the rack identifier. 0 if no rack ID given.
RackId	string	0...255	Rack identifier, max 255 chars. Sent if the pickup location has any rack ID defined.

Frame Length	Data Length	Min Length	Max Length
9	4 + Packet Lengths	13	541

12.2.51. PivotResult (ID = 364)

Message is sent as a response to PivotCommand (ID = 363).

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
MachineId	uint32	4	Machine ID to pivot.
Result	uint32	4	Result of the pivoting. See list below.

Frame Length	Data Length	Min Length	Max Length
9	8	17	17

Result	Description
0	Success
1	Bad input
2	Wrong state
4	Control disabled, or machine does not support the message
8	Drive rejected
9	Error active
10	Pivoting aborted
100	Started (command sent to machine)
101	Unknown (unknown response from machine)

12.2.52. MissionTemplateActivationResult (ID = 10009)

Message is sent as a response to ActivateMissionTemplate (ID = 10008) message.

FieldName	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
Success	int16	2	Result of the activation. `1` = success, `0` = failure.
InternalId	int32	4	Internal ID generated for the activated mission. The mission internal ID might change during the area reload.
String Length	int16	2	
ExternalId	string	0..255	External ID which was given for the activated mission.
String Length	int16	2	
Description	string	0..255	Description explaining the activation result.

Frame Length	Data Length	Min Length	Max Length
9	8	19	529

12.3. Message acknowledgement

These messages can be sent either end of the communication to verify message receival. Navithor replies with message ID = 200 to all successfully parsed messages.

12.3.1. AckOrReject (ID = 200)

This message can be sent either by Navithor or MES Client as a reply to all received messages. Navithor sends automatically AckOrReject (acknowledging) after receiving a correctly formatted message. Navithor can still later send another AckOrReject-message (rejecting) in case the received message cannot be fulfilled for some reason. In case MES Client does not receive an AckOrReject-message after successfully sending a message to Navithor, then the message data is not in correct format or network is blocking the message.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
AckReject	byte	1	0 for Ack, 1-254 rejection reason. See section ErrorIDs for information about possible reasons for rejected message
MessageID	uint16	2	Acked/Rejected message ID
ResponseID	uint16	2	0 do not wait, 1-65535 response ID
ResponseTimeOut	uint32	4	Amount of time to wait for response (ms)

Frame Length	Data Length	Min Length	Max Length
9	9	18	18

12.3.2. AckOrRejRequest (ID = 201)

This message should be send as a reply to request messages that have a RequestID-field specified. For example MES client is required to reply to RequestResourceStatusAtLocation-message with this message.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
AckReject	byte	1	0 for Ack, 1-254 rejection reason. See section ErrorIDs for information about possible rejection reasons
MessageID	uint16	2	Acked/Rejected message ID
RequestID	uint16	2	Running ID number copied from the received request message

Frame Length	Data Length	Min Length	Max Length
9	5	14	14

12.4. Heartbeat

These messages can be sent between Navithor Server and MES Clients to verify communication socket is still open. On default heartbeat is not required, but it can be activated with parameters from Navithor:

`MES_Heartbeat_Required` : Signals whether MES communication requires periodic heartbeat messages to clients.

`MES_Heartbeat_Interval` : Interval to send heartbeat in seconds, received heartbeat is valid for three times this interval. Once heartbeat is no longer valid, MES connection will be closed to the client. *HeartbeatRequired* must be set true for this to apply.

12.4.1. Heartbeat (ID = 203)

This is sent as a heartbeat from MES to all connected Clients. It has to be answered with message ID = 204 HeartbeatResponse, or the communication will close after a time.

Field Name	Type	Length	Description
Frame	header	9	Every message has a frame that consists of the same data
NavithorStatus	uint16	2	Flags: Bit 0: AreaOK Bit 1: DatabaseOK Bit 2: TrafficControllerOk Bit 3: AGVCommsOk All OK = 15 in decimal format
Count	uint16	2	Running index for heartbeat, starts from 0 for each Client and increases by one for each heartbeat. Each Client has its own heartbeat counter

Frame Length	Data Length	Min Length	Max Length
9	4	13	13

12.4.2. HeartbeatResponse (ID = 204)

This is sent as a reply to Heartbeat message (ID = 203). No additional data in the message.

Frame Length	Data Length	Min Length	Max Length
9	0	9	9

12.5. Error IDs

12.5.1. Rejecting a received message with AckOrReject-message

Following error IDs can be informed to MES Client when rejecting a received message.

Error ID	Description
1	Bad input
2	Load not at the location
3	Machine was not found
4	Symbolic point with specified ID was not found
5	Database error
6	MES Client connection not found
7	Load not valid for pickup
8	Received message not supported
9	Machine is disabled
10	Symbolic Point is disabled
11	Order is not found
12	Bad state
254	Logic error

12.5.2. Rejecting a request-type message with AckOrRejRequest-message

Following reject reasons are handled by Navithor when Navithor is sending a request-message to MES Client(s). MES Client can reject the request with following reasons in AckOrRejRequest-message:

**Reject reason	Description
4	Request cannot be fulfilled at the moment. Postpone related order (Transfer request)
5	Request cannot be fulfilled at all. Cancel related order

13. Building a data message in Navithor-MES interface - Example

In this chapter you will find an example of communication between Navithor Server and Manufacturing Execution System (MES). All values follow little-endian format (LSB first). First, the overall structure is shown and then same thing byte by byte written to TCP/IP socket.

The example is represented with Production status message (ID = 313). Server sends Production status message periodically. The message includes both Frame and Message Data.

13.1. Structure

	Length
Frame	9
Message Data	0-65535

13.1.1. Frame

Field	Type	Length (bytes)	Value	Description
ID	uint16	2	0-65535	Message ID
Sender ID	uint16	2	0-65535	Unique ID of the sender
Receiver ID	uint16	2	0 Any, 1-65535 receiver ID	Unique ID of the receiver
Message type	byte	1	1 reply needed, 2 reply not needed	Tells if the message needs to be replied
Data length	uint16	2	0-65535	Message data length in bytes (0 means that no other data than frame is sent)

13.1.2. Message Data

Field Name	Type	Length	Description
numberOfProductionOrders	uint16	2	Number of ProductionOrderData-structures in this message. Max 200
ProductionOrderData	ProductionOrderData[]	numberOfProductionOrders * ProductionOrderData length	Production Order Data structure below

Frame Length	Data Length	Min Length	Max Length
9	2 + structs	11	44411

Production Order Data

Production order data struct containing information from a single production order.

Field Name	Type	Length	Description
NameStringLength	uint16	2	Max 200 characters
Name	string	0...200	Production Order Name
Id	uint32	4	Production order ID
TargetSymbol	int32	4	
AssignedToMachineId	uint16	2	0 if not assigned, machine ID if assigned machine
PickupSymbol	int32	4	-1 if no pickup symbol
ItemTypeToDeliver	int32	4	-1 if no items
CurrentStatus	byte	1	new_order = 0, loaded = 1, waiting = 2, executing = 3, interrupted = 4, completed = 5, cancelled = 6, paused = 7
ExecutionStatus	byte	1	none = 0, drive to pickup = 1, at pickup = 2, picking up = 3, picked up = 4, driving to target = 5, at target = 6, dropping off = 7, dropped off = 8, at hold = 9

Min Length	Max Length
22	222

13.2. Byte by byte

The message example is represented with the following structure:

```
<Index of byte value in message> : <Byte value in HEX>    Description
```

Empty lines are used to differentiate each item/variable in the message.

Production status (Frame + MessageData):

```
00 : 39
01 : 01    Message ID = 313

02 : E8
03 : 03    Sender ID = 1000

04 : E9
05 : 03    Receiver ID = 1001

06 : 01    Message type = 1 (this can be ignored)

07 : 23
08 : 00    Data Length = 36 bytes

09 : 01
10 : 00    Number of production orders = 1

11 : 0C
12 : 00    Name string length = 12

13 : 4D    'M'
14 : 61    'a'
```

```
15 : 6E    'n'
16 : 75    'u'
17 : 61    'a'
18 : 6C    'l'
19 : 20    ' '
20 : 6F    'o'
21 : 72    'r'
22 : 64    'd'
23 : 65    'e'
24 : 72    'r' (Production Order Name = "Manual order")

25 : D9
26 : 80
27 : 00
28 : 00    Production order ID = 32985

29 : 13
30 : 00
31 : 00
32 : 00    Target symbol = 19

33 : F5
34 : 06    Assigned to machine ID = 1781

35 : FF
36 : FF
37 : FF
38 : FF    Pickup symbol = -1 (no pickup symbol target)

39 : FF
40 : FF
41 : FF
42 : FF    Item type to deliver = -1 (no items)

43 : 03    Current status = 3 (executing)

44 : 05    Execution status = 5 (driving to target)
```