GridOps Management Suite 3.10

Network Import Notification Interface

Functional Specification

Document Version: 1.1

Updated: June, 2024

The information contained in this document is confidential, privileged and protected under the applicable laws. This document is only for the information of the intended recipient and may not be used, published, or redistributed without the prior written consent of Schneider Electric.

This document has undergone extensive technical review before being released. While every care has been taken in preparing these documents in order to keep the information herein as accurate and up to date as possible, neither Schneider Electric nor any of its affiliates or subsidiaries shall be responsible or liable for misuse of the information contained herein, nor for errors or omissions or for damages resulting from the use of the information contained herein.

The content of this document is subject to change without prior notice.

Life Is On

Table of Contents

Table of Figures

Table of Documents

No table of figures entries found.

1. REFERENCES

#	Title	Description
1.	EcoStruxure GridOps Management Suite 3.10 Enterprise Integration Platform - Functional Specification	The document represents a set of common integration principles applied to all baseline integration adapters.
2.	EcoStruxure GridOps Management Suite 3.10 Model Management - Functional Specification	The document describes the general procedure of creating, verifying and distributing network model data and associated changesets within an EcoStruxure GridOps system.
3.	EcoStruxure GridOps Management Suite 3.10 Network Data Integration - Functional Specification	The document describes the Network Data Integration (NDI) module of EcoStruxure GridOps which represents a set of functionalities designed to facilitate the data migration as well as the sustained data integration between the most commonly encountered external data sources (e.g. GIS, EAM, CIS/CRM, MDMS) and EcoStruxure GridOps Network Model data repositories.
4.	EcoStruxure GridOps Management Suite 3.10 Network Import Notification Interface	EcoStruxure GridOps Management Suite 3.10 Network Import Notifications Interface zip file contains essential configuration information, as well as web service definitions complemented with message examples.

2. ASSUMPTIONS AND PREREQUISITES

The Network Import Notifications (NIN) integration is designed under the following assumptions:

- Integration is implemented through the NIN Adapter.
- Out of the box integration is “one-way”, meaning that NIN Adapter will send necessary data to an externally hosted SOAP based Web Service and predefined shared location, when it is necessary.
- Both the SFTP and CIFS (SMB) are supported as shared locations.
- SFTP authentication can be done using username/password or private key file.
- Default encoding on SFTP server must be UTF-8.
- In case of unavailability of the web service hosted on the ESB, the NIN Adapter will preserve the data by not updating the last export time. That data will be queried again and sent in the next scheduled cycle.
- Utility is responsible for providing and maintenance of the File Share.
- Utility is responsible for defining folder structure on the File Share.
- NIN Adapter will never remove files from the SFTP, housekeeping of the File Share is Utility’s responsibility.
- Housekeeping of local file share which resides in the EcoStruxure GridOps environment is under the NIN Adapter’s responsibility.
- In case of the model promotion configuration change, when the state transitions or changeset types are modified, the adapter should be restarted.

3. INTRODUCTION

EcoStruxure GridOps Management Suite is a family of solutions designed to help electric utilities in the operations and management of their grid. It is offered as EcoStruxure ADMS, EcoStruxure Grid Operation, EcoStruxure DERMS or EcoStruxure Energy Transmission Operation solutions, which share the same technology platform.

NOTE: The functionality described in this document applies to all solutions.

NOTE: Most images presented in this document are related to the EcoStruxure ADMS solution and should be used as an example. The images for other solutions may differ slightly.

Purpose of this integration is reducing the unnecessary overhead in communication between utility' departments by providing network import notifications, thus increasing efficiency. Additionally, increase of situational and operational awareness is achieved by providing support for equipment state transitions during planned works execution.

The EcoStruxure GridOps receives network updates originated from external data sources such as GIS, CIS, MDSM, etc. in a form of the XML and CSV files – "extracts". When extract is sent to the EcoStruxure GridOps, it is processed, validated and transformed into the binary representation of a model change, which is afterwards stored inside the changeset repository database (CSRepo) in a form of a "changeset". A changeset is further manually verified and applied/promoted to the network model per defined model management process.

Extract and changesets, as main artifacts that participate in the model management process, have predetermined lifecycle per the state machine. During the state transitions, extract and changeset are applied/unapplied to the model, moved from the one system/zone to another or discarded as non-valid or obsolete.

Changesets can also be created manually, by using the Network Builder application. These types of changeset do not have any related extracts.

Users of external systems, in this case GIS, need to be aware of information regarding above-mentioned state transitions to fully support requirements for model management process. To realize this, after network updates are received and processed, the notification which contains processing result in a form of extract/changeset state change shall be sent back to the GIS.

The integration is achieved via NetworkImportNotifications Adapter which is responsible for monitoring these state changes and sending details about the desired state transitions to the web service hosted in the corporate zone.

Additionally, when failures are detected during import process, the adapter shall store textual import reports, with details of failures, on the SFTP or CIFS (SBM) file share inside corporate environment.

3.1. General Architecture

Described in the *EcoStruxure GridOps Management Suite 3.10 Enterprise Integration Platform - Functional Specification* document [1].

4. INTEGRATION OVERVIEW

The Network Import Notifications Integration is implemented through the NIN Adapter component. The aforementioned adapter implements the SOAP based Web Service Clients and File Upload component. Operations implemented within the NIN Adapter and corresponding web service client are listed below:

- SendNetworkImportNotificationsService – used for sending changesets' state transition notifications and uploading import reports to predefined shared location:
  - CreatedNetworkImportNotifications operation
  - CreatedImportReports operation

  Once files are uploaded to predefined shared location, the appropriate message is sent to dedicated Web Service hosted on the client side.
- SendEquipmentStatesService – used for sending information about network elements whose service state has transitioned to a new service state:
  - ChangedEquipmentStates operation

The following chapters provide more details regarding implementation (business process) covered by these web service clients, the appropriate web service operations, data mappings, error handling scenarios, etc. the use case diagram that represents common participants (actors) and users of the aforementioned interface in the NIN Integration is given in Figure 4.1.

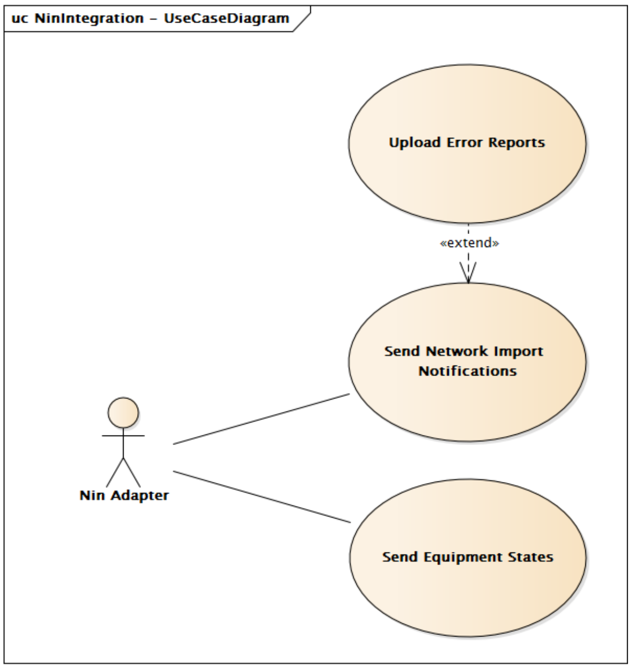
uc NinIntegration – UseCaseDiagram
Upload Error Reports
«extend»
Send Network Import Notifications
Nin Adapter
Send Equipment States

Figure 4.1 – The NIN Integration use case

5. INTEGRATION PROCESS

Information about extracts/changesets are stored inside the CSRepo database on a historical server. The CSRepo database is replicated between the systems (zones). The NIN adapter, running in the DMZ system is triggered at the predefined time interval and begins the process of data collection (step 1). State transitions for preconfigured set of extracts and changesets are collected through the CSS from CSRepo database residing in the DMZ system (step 2). The NIN Adapter collects only changes which occurred between the current and moment when the adapter was previously triggered.

In case when some of state changes imply that import of data was not successful, like when extracts are declared as invalid or changeset as rejected, textual reports of failure are uploaded to external shared location which resides in the corporate environment (step 3). After changes are collected, details about transitions are sent to the externally hosted web service within web service request (step 4 and 5).

Overview of the entire process is depicted in Figure 5.1.

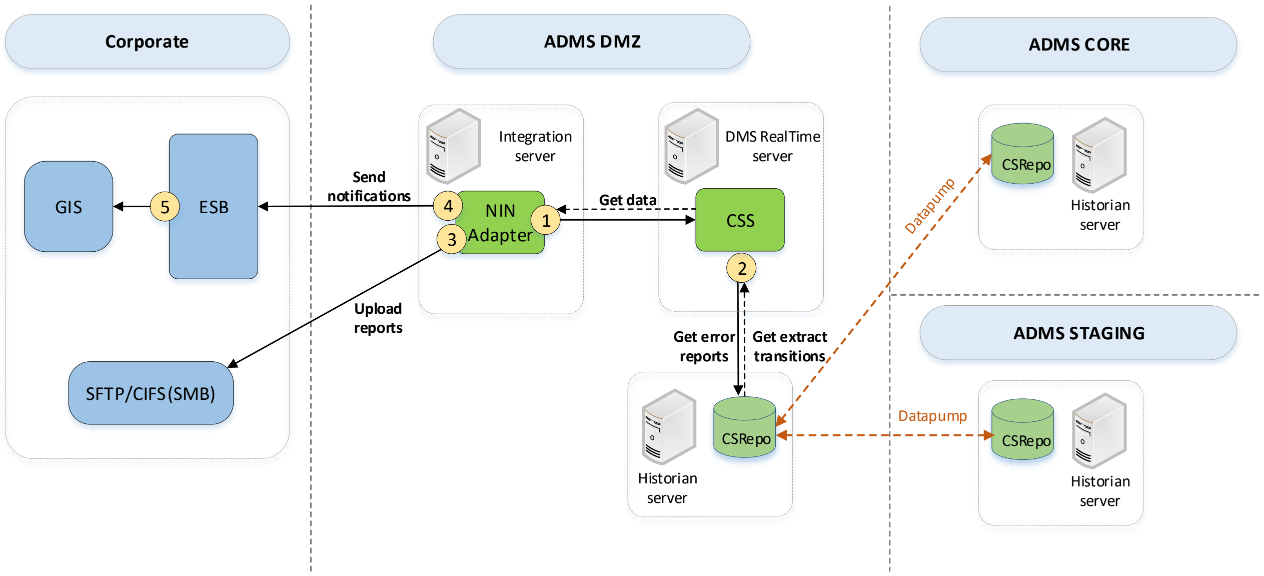

Figure 5.1 – Sending network import notifications and import reports

Besides network import notifications, a separate process dedicated for service state change notifications for planned states equipment exists. Adapter filters the CSRepo database for transitions of specific changeset, planned state changesets. Those changesets contain information about network elements whose state has transitioned from or to one of the following service states: “Decommissioned”, “InService”, “PlannedInstalled” and “PlannedUninstalled”. That information is formatted and propagated to corresponding external GIS system. Process diagram fully corresponds the one presented in Figure 5.1, except the SFTP/CIFS upload. All corresponding data is delivered via externally hosted SOAP based web service.

5.1. Extract and Changeset State Transitions

Since the NIN Adapter is highly configurable and implemented in a generic way, sending state transitions and appropriate import reports is supported for various extract and changeset types (both product and project specific). The configuration of the NIN Adapter is explained in Interface configuration.

Examples of state transitions for the MV/LV feeder extracts, related changesets and customer data extracts state transitions are depicted in Figure 5.2 and Figure 5.3.

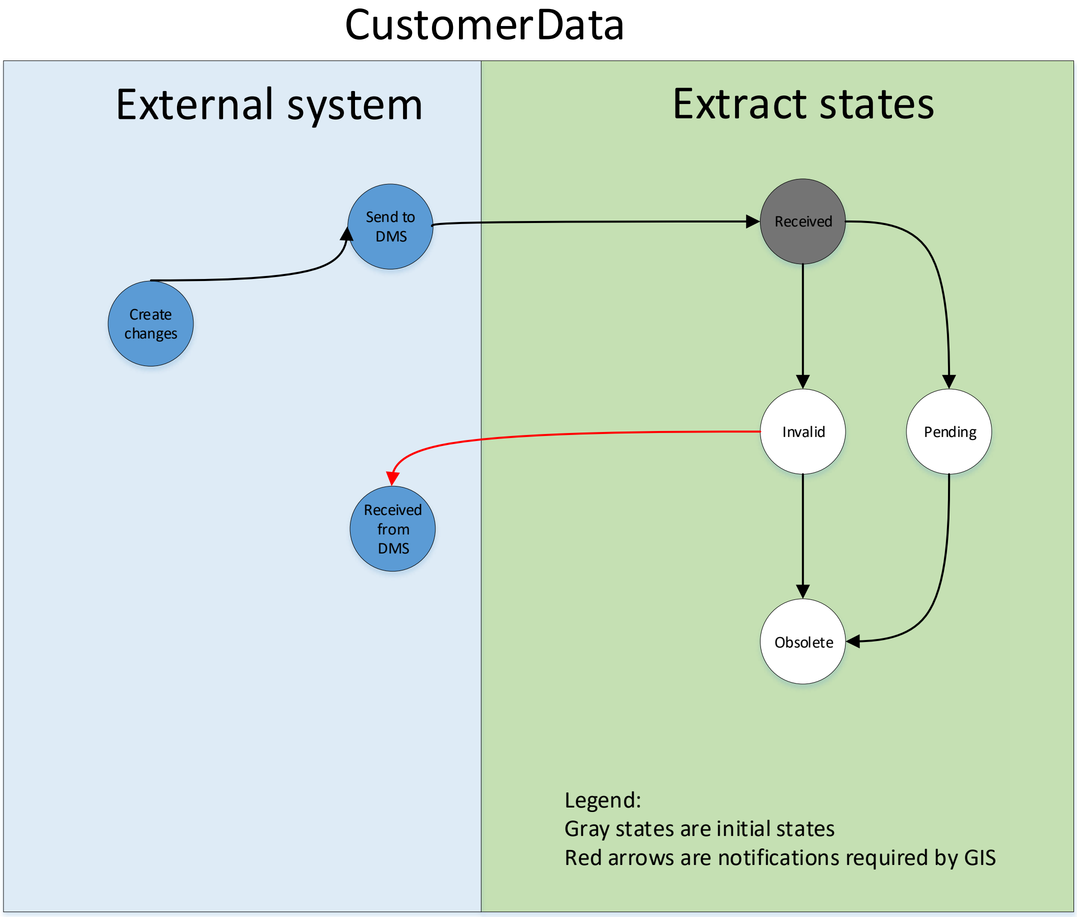

Figure 5.2 – The customer data extract state transitions

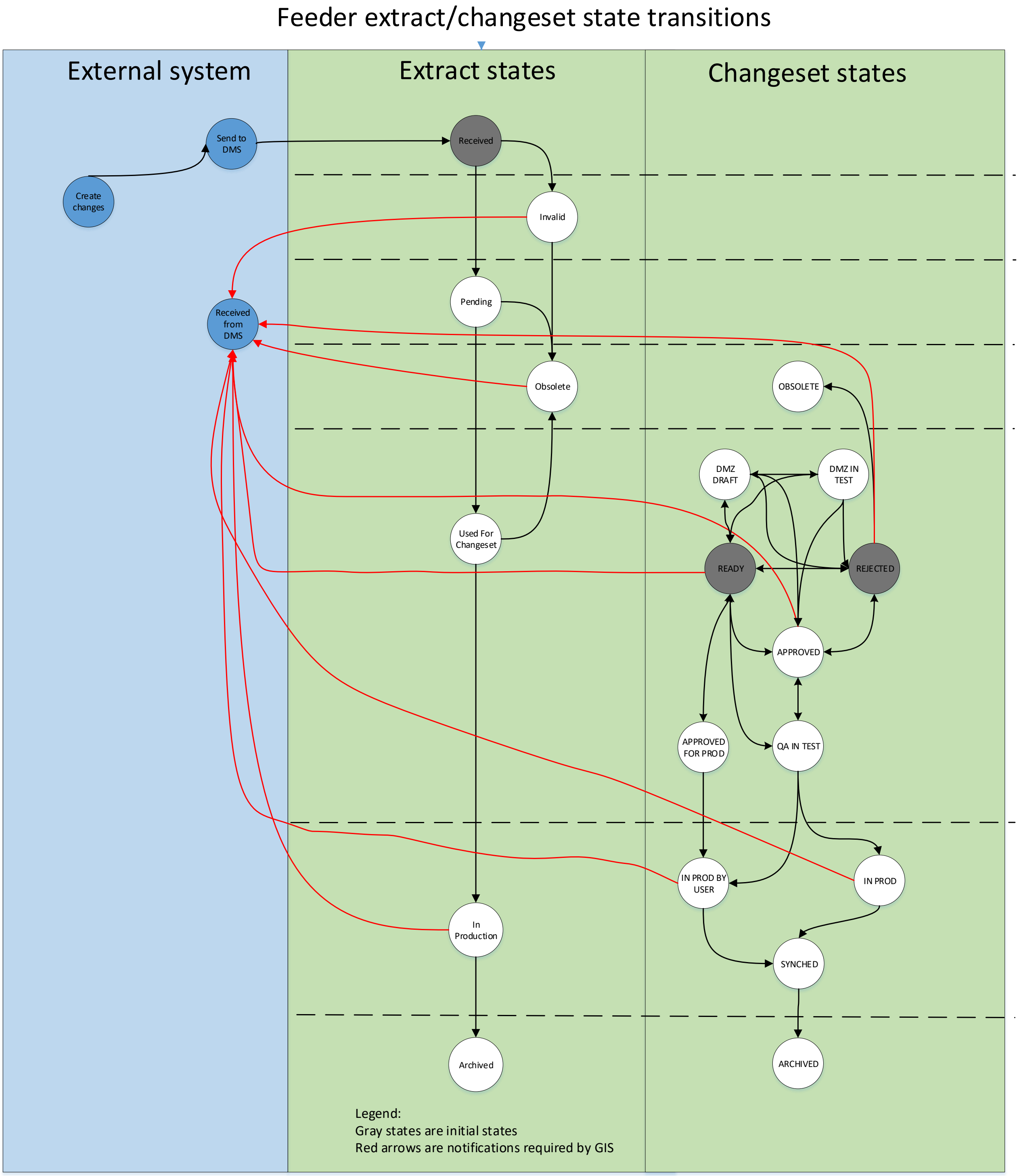

Figure 5.3 – The feeder extract and changeset state transitions

6. SENDNETWORKIMPORTNOTIFICATIONS SERVICE

6.1. CreatedNetworkImportNotifications Operation

6.1.1. Overview

As stated above, the NIN adapter provides functionality of sending a list of state transitions for processed extracts. On a preconfigured time, the NIN adapter initiates state transition check process. For preconfigured set of extract and changeset types adapter pulls state transitions. Only state transition data that is between the moment when the NIN Adapter was triggered and the last time when the NIN Adapter sent the message successfully to the Web Service is sent within the new message.

In case when the changeset is created out of extracts, one or many of them, information about state transitions are going to be grouped per extract. It means that if one changeset is generated out of multiple extracts, message for state transitions is going to contain state transitions for all related extracts. This applies to feeder or patch extracts/changesets. In other situations, when the changeset is not made from extract (e.g., “SubstationInternal” type of changeset), grouping will not be performed. Changes for all state transitions are send together inside the single message.

Additionally, for specific extract and changeset state transitions, Network Import service generates a report. NIN adapter pulls those import reports and stores them on a local file share (server where the NIN adapter is running). From there NIN adapter uploads import reports to external file share (SFTP or CIFS/SMB). Reports are uploaded in a form of textual files where single file per extract is generated, meaning that if a changeset affects multiple feeder extracts, for each feeder extract separate import report is generated. By doing that, collaborative cooperation between departments is increased thus increasing efficiency by reducing unnecessary need of other forms of communication.

Process of gathering, uploading and sending data to the web service is depicted in Figure 6.1.

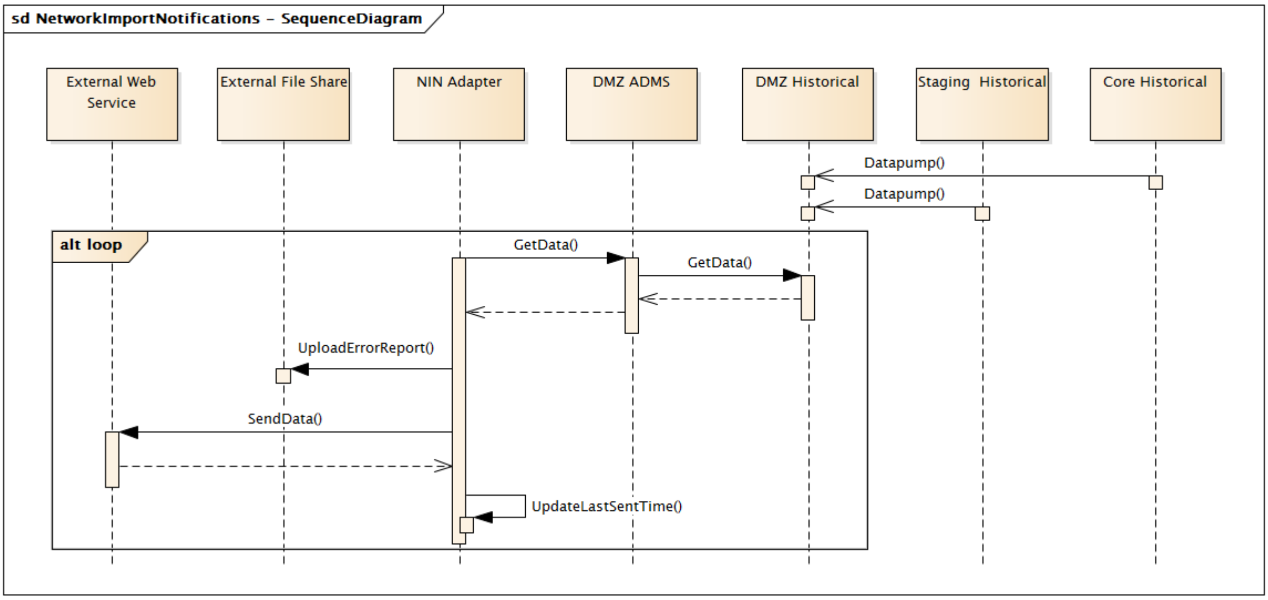

Figure 6.1 – The CreatedNetworkImportNotifications operation execution

In case when queried state transitions contain error report data, reports are generated and uploaded to a predefined file share, as stated above. In cases where a connection to a predefined file share cannot be made, the notification process is aborted also. That means that the next scheduled operation will query the data that was not processed in the previous iteration(s), together with the recent changes, and export them in the current iteration. Finally, timestamp that defines the start interval of each export iteration is updated with every successful notification. That timestamp is stored in a dedicated adapter element cache, which is persisted and replicated between adapter pairs. In case of failover event, newly promoted HOT adapter has the information about last successful export time. In cases when the web service is unavailable, timestamp will not be updated. Detailed explanation is located in the Use Cases.

6.1.2. Use Cases

The list of possible use cases and corresponding faults is given in Table 6.1.

Table 6.1 – The CreatedNetworkImportNotifications use cases

Use Case	Message Mapping			Action
	Property	Type	Value	
Local file share is not available	Result	String	FAILED	NIN Adapter cannot store import reports on the local file share. CreatedNetworkImportNotification process is aborted. Error is written to a log file and event is sent by NIN Adapter. LastSentTimestamp is not updated.
	Error.code	String	5.3	
	Error.level	String	FATAL	
	Error.reason	String	UnavailableLocalFileShare	
	Error.details	String	Could not copy file: {0} from local file share since it is unavailable.	
Failed to copy extract	Result	String	FAILED	NIN Adapter tries to copy import reports to external file share configurable number of times. After configurable number of attempts, CreatedNetworkImportNotification process is aborted. Error is written to a log file and event is created by NIN Adapter. LastSentTimestamp is not updated.
	Error.code	String	5.3	
	Error.level	String	FATAL	
	Error.reason	String	UnavailableExternalFileShare	
	Error.details	String	Could not copy file: {0} to: {1} since FS is unavailable.	
Web Service unavailable	Result	String	FAILED	The NIN Adapter pulls data from the CSRepo database and tries to send message to the external system configurable number of times. After it fails, an appropriate error is logged and event is created. The LastSentTimestamp is not updated.
	Error.code	String	5.3	
	Error.level	String	FATAL	
	Error.reason	String	UnavailableWebService	
	Error.details	String	Web service hosted on {0} is currently not available.	
Invalid response message	Result	String	OK	The NIN Adapter pulls data from the CSRepo database and sends message to the external system. If invalid response message is received from the external system, the NIN Adapter logs the error. The LastSentTimestamp is updated.
	Error.code	String	1.8	
	Error.level	String	WARNING	
	Error.reason	String	InvalidMessage	
	Error.details	String	Received message is invalid against XSD schema. Reason: {0}.	

Use Case	Message Mapping			Action
	Property	Type	Value	
Created Network Import Notifications are sent successfully	Result	String	OK	The NIN Adapter pulls data from the CSRepo database and sends message to the external system. The valid response message is sent by external system with the OK result. The LastSentTimestamp is updated.
	Error.code	String	N/A	
	Error.level	String	N/A	
	Error.reason	String	N/A	
	Error.details	String	N/A	

6.1.3. Triggers

Out of the box configuration implies that the NIN Adapter sends information about the feeder extracts, MV feeder changesets and customer data extracts (Table 6.2 and Table 6.3).

Additionally, there are two types of reports that importer generates and for both report types files are copied to SFTP:

- Invalid extract reports
- Rejected changeset reports

Out of the box configuration implies that error report data will be created and uploaded to a predefined SFTP server for invalid Feeder and Customer data extracts, as well as for the rejected MV Feeder changes.

Table 6.2 – The CreatedNetworkImportNotifications triggers – Extract Data

Triggers	
Source Data	**States**
Feeder	INVALID, OBSOLETE, IN PROD
Customer Data	INVALID

Table 6.3 – The CreatedNetworkImportNotifications triggers – Changeset Data

Triggers	
Type Data	**States**
MV Feeder	APPROVED, REJECTED

7. SENDEQUIPMENTSTATES SERVICE

7.1. ChangedEquipmentStates Operation

7.1.1. Overview

Besides network import notifications, which were described in the previous chapter, NIN adapter offers the functionality of subscribing to network element state changes. This is particularly important during the execution of planned works, where a certain part of the network is being commissioned/decommissioned. In order to increase situational and operational awareness EcoStruxure GridOps offers the planned (proposed) states feature. Proposed equipment is being imported through a regular model management procedure. Operators in the control room have better awareness about the upcoming work in the field so they can plan and schedule appropriate field crews through planned work. During the execution of planned work, where a part of network is being commissioned/decommissioned, operator can keep track of appropriate switching instructions and update the network model so it is aligned with the actual state in the field. Once dedicated part of the network is commissioned/decommissioned, a planned states changeset is created and stored to CSRepo database.

Service state mapping of values in network model and enum values "serviceState" in the notification message is given in Table 7.1.

Table 7.1 – Service state mapping

Network model	Notification message
Decommissioned	Decommissioned
InService	InService
PlannedInstalled	PlannedInstalled
PlannedUninstalled	PlannedUninstalled
Installed	InService
Uninstalled	Decommissioned

On a preconfigured time, the NIN adapter queries the CSRepo database for newly added planned state changesets. Only changesets that are added between the trigger moment and the last successful notification are processed and sent within the notification message. It is important to mention that last successful notification time is stored in the same adapter element cache, as explained in the Network Import Notifications interface. It is persisted and replicated between adapter pairs.

Data retrieved from planned states changes are filtered for data source. Only network elements that satisfy the filter will be processed and exported via corresponding notification message. That data source filter is defined within the adapter registry configuration file.

Process of gathering and sending data to the web service is depicted in Figure 7.1.

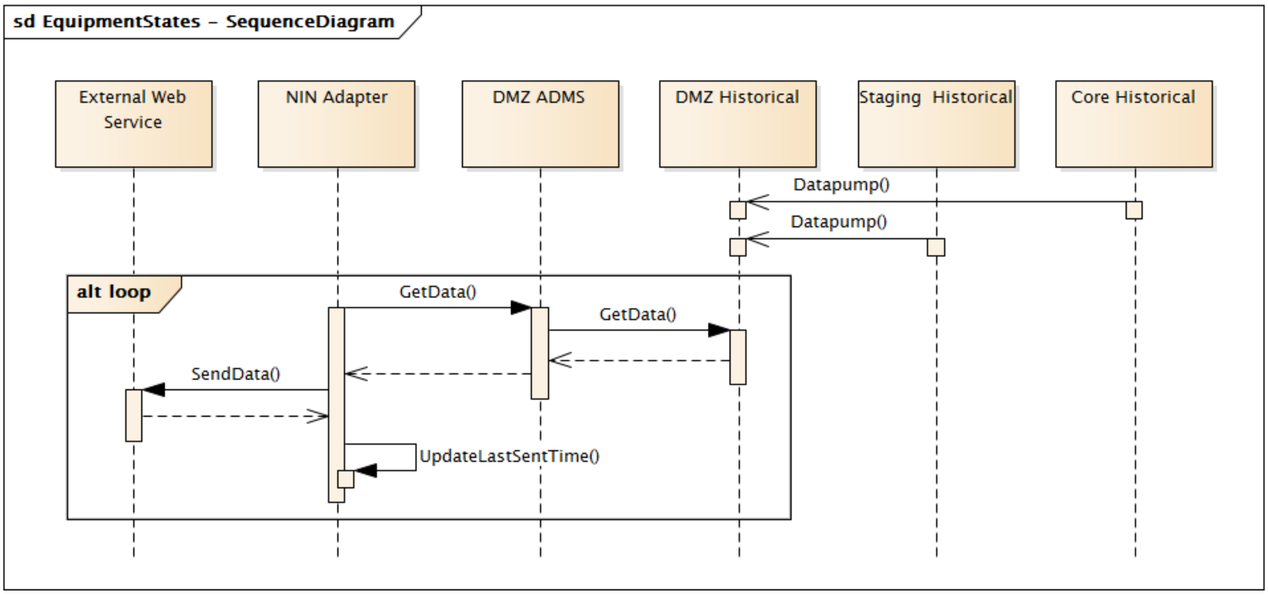

Figure 7.1 – The ChangedEquipmentStates operation execution

7.1.2. Use Cases

The list of possible use cases and corresponding faults is given in Table 7.2.

Table 7.2 – ChangedEquipmentStates operation use cases

Use Case	Message Mapping			Action
	Property	Type	Value	
Web Service unavailable	Result	String	FAILED	The NIN Adapter pulls data from the CSRepo database and tries to send message to the external system configurable number of times. After it fails, an appropriate error is logged and event is created. The LastSentTimestamp is not updated.
	Error.code	String	5.3	
	Error.level	String	FATAL	
	Error.reason	String	UnavailableWebService	
	Error.details	String	Web service hosted on {0} is currently not available.	
Invalid response message	Result	String	OK	The NIN Adapter pulls data from the CSRepo database and sends message to the external system. If invalid response message is received from the external system, the NIN Adapter logs the error. The LastSentTimestamp is updated.
	Error.code	String	1.8	
	Error.level	String	WARNING	
	Error.reason	String	InvalidMessage	
	Error.details	String	Received message is invalid against XSD schema. Reason: {0}.	
Changed Equipment States are sent successfully	Result	String	OK	The NIN Adapter pulls data from the CSRepo database and sends message to the external system. The valid response message is sent by external system with the OK result. The LastSentTimestamp is updated.
	Error.code	String	N/A	
	Error.level	String	N/A	
	Error.reason	String	N/A	
	Error.details	String	N/A	

8. MESSAGES

8.1. Common

8.1.1. Header

The header section is defined according to the IEC 61968-100 standard. Currently, there are two required fields that must be populated:

- **Verb** – to identify a specific action to be taken. There is an enumerated set of valid verbs, where commonly used values include "get", "create", "change", "cancel", "close", "execute" and "reply". Within the event notification messages "past tense" verbs are used, which can include "created", "changed", "canceled", "closed" and "executed". Implementations should treat deprecated verbs "update" and "updated" as synonyms to "change" and "changed".
- **Noun** – to identify the subject of the action and/or the type of the payload, such as the NetworkImportNotifications.

Field that can be optionally supplied include the following:

- Revision – to indicate the revision of the message definition. By default, this should be "1".
- ReplayDetection – this is a complex element with a timestamp and a nonce used to guard against replay attacks. The timestamp is generated by the source system to indicate when the message was created. The nonce is a sequence number or randomly generated string (e.g. UUID) that would not be repeated by the source system for at least a day. This serves to improve encryption.
- **Context** – a string that can be used to identify the context of the message. This can help provide an application level guard against incorrect message consumption in configurations where there may be multiple system environments running over the same messaging infrastructure. Some example values are PRODUCTION, TESTING, STUDY and TRAINING.
- Timestamp – an ISO 8601 compliant string that identifies the time the message was sent. This is analogous to the JMSTimestamp provided by JMS. Either Zulu ('Z') time or time with a time zone offset may be used.
- Source – identifying the source of the message, which should be the name of the system or organization.
- AsyncReplyFlag – the Boolean data type ("true" or "false" values) that indicates whether a reply message will be sent asynchronously. By default, replies are assumed to be sent synchronously,
- ReplyAddress – the address to which replies should be sent. This is typically used for asynchronous replies. This should take the form of a URL, topic name or queue name. This is analogous to the JMSReplyTo field provided by JMS. This is ignored when using unidirectional integration patterns (e.g., AckRequired=false). If the reply address is a topic, the topic name should be prefixed by "topic". If the reply address is a queue, the queue name should be prefixed by "queue". If the reply address is a web service, the reply address should be a URL beginning with "http://" or "https:/".
- AckRequired – the Boolean data type ("true" or "false" values) that indicates whether an acknowledgement is required. If false, this would indicate that a unidirectional integration pattern is being used for communicating transactional messages.

- User – a complex structure that identifies the user and associated organization. Should be supplied as it may be required for some interfaces, depending upon underlying implementations. This allows the UsersID string and optional the Organization string as sub-elements.
- MessageID – a string that uniquely identifies a message. Use of the UUID or sequence number is recommended. This is analogous to the JMSMessageID provided by JMS. A process should not issue two messages using the same MessageID value.
- CorrelationID – this is used to "link" messages together. This can be supplied on a request, so that the client can correlate a corresponding reply message. The server will place the incoming CorrelationID value as the CorrelationID on the outgoing reply. If not supplied on the request, the CorrelationID of the reply should be set to the value of the MessageID that was used on the request, if present. This is analogous to the use of the JMSCorrelationID provided by JMS. Given that the CorrelationID is used to 'link' messages together, it may be reused on more than one message. Use of a UUID or sequence number is recommended.
- Comment – any descriptive text, but shall never be used for any processing logic.
- Property – a complex type that allows the custom name/value pairs to be conveyed. The source and targets would need to agree upon usage. These are analogous to a Property as defined by JMS,
- Any – it can be used for custom extensions.

Figure 8.1 shows the graphical representation of the header field.

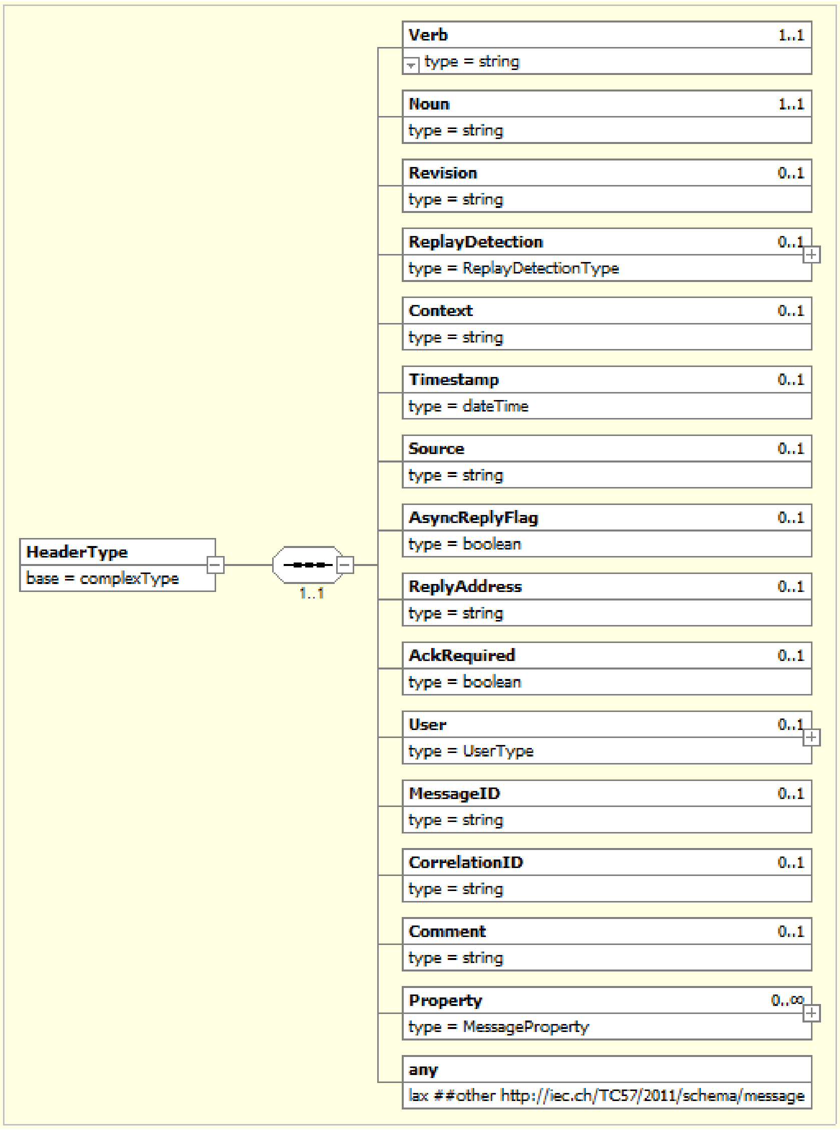

Figure 8.1 – The header field

8.1.2. Reply and Fault

The Reply.result value is an enumeration and would be populated in the following manner:

- "OK" – if there are no errors and all results have been returned. There is no requirement that a Reply.Error element be present.
- "PARTIAL" – if only a partial set of results has been returned, with or without errors. Existence of errors is indicated with one or more Reply.Error.code elements.
- “FAILED" – if no result can be returned due to one or more errors, indicated with one or more Reply.Error elements, each with a mandatory application level ‘code’.

If the result type is “PARTIAL” or “FAILED”, the **Error** field will be populated with the appropriate error description. The contents the **Reply** and **Error** fields are presented in Figure 8.2.

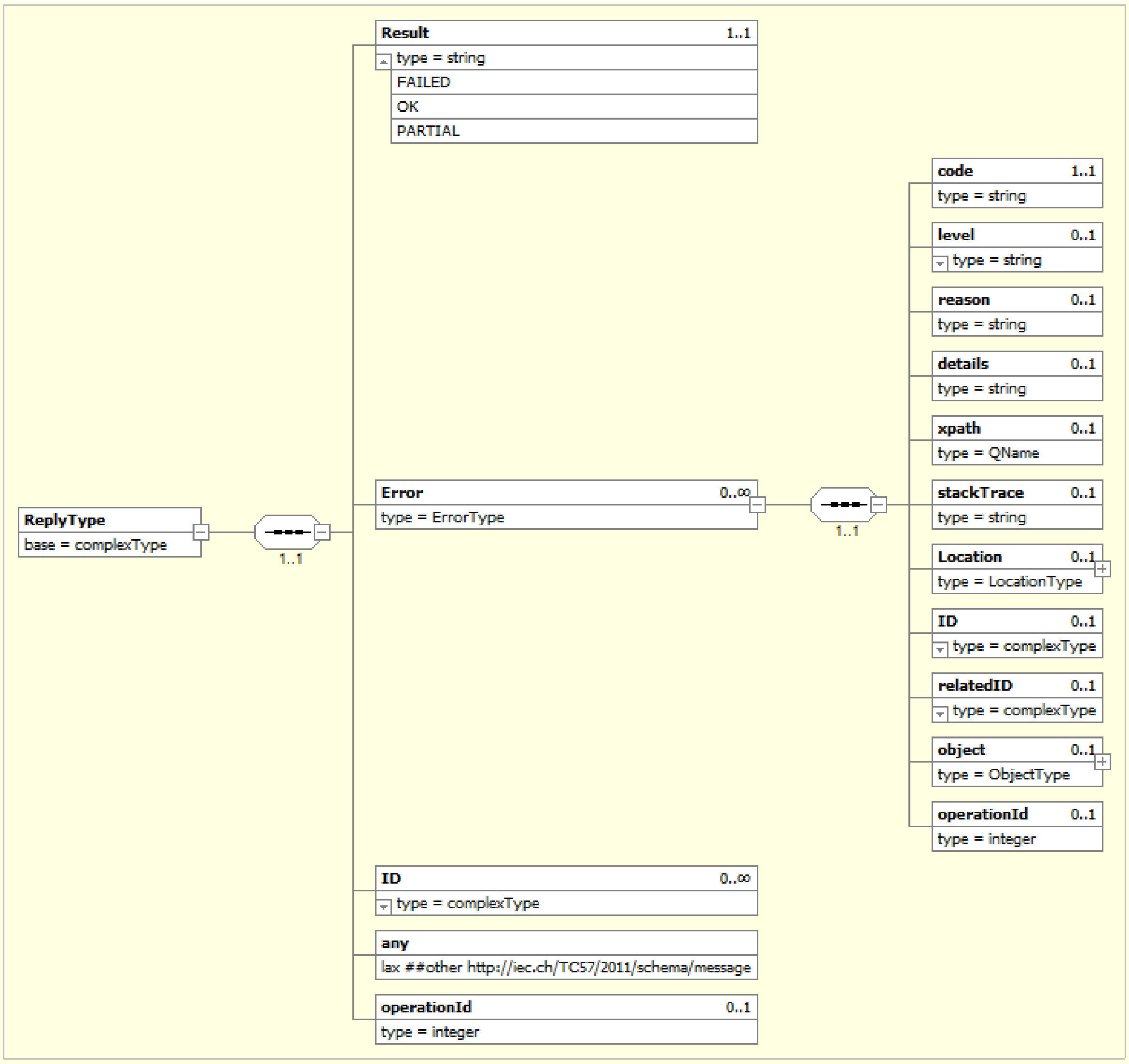

*Figure 8.2 – The **Reply** and **Error** field contents*

8.2. CreatedImportReports Operation

The operation definition:

CreatedImportReports(List<ImportReports>)

8.2.1. Request

Out of the box functionality implies that import reports will be sent for two types of extracts: the feeder and customer data. Such import reports will be uploaded to predefined folders on external file share:

- FEEDER_IMPORT_REPORTS,
- CUSTOMER_IMPORT_REPORTS.

Import report file names will be created from appropriate values extracted from the CSRepo database and corresponding tables (extract name and creation timestamp). All reports will have *.txt file extension. Depending on type of extract, generation of the report file name will be slightly different. The following table provides rules for naming import report files:

Table 8.1 – Import reports naming convention

Extract/Changeset Type	File name		
	Description	Rule	Example
Feeder extract/changeset	Extract file name where .xml extension is replaced with txt	**[EXTRACT_INFO]. [FILENAME] – file extension** + **“txt”**	FEEDER_01_20170728002608.txt
Customer	Extract file name where extension is replaced with txt and creation timestamp is added	**[EXTRACT_INFO]. [FILENAME] – file extension** + **[EXTRACT_INFO]. [CREATED_DATE]** + **“txt”**	CUSTOMER _20170728002608.txt

8.2.2. Response

There is no response message defined for this operation. Only indication whether files were successfully uploaded to the external file share.

8.2.3. Fault

There is no fault message defined for this operation Based on the upload result, the appropriate error will be logged and event sent.

8.3. CreatedNetworkImportNotifications Operation

The operation definition:

CreatedNetworkImportNotificationsResponse CreatedNetworkImportNotifications(CreatedNetworkImportNotificationsEvent request)

8.3.1. Request

The *CreatedNetworkImportNotifications* event message is defined according to the IEC 61968-100 and contains the following two sections:

- Header
- Payload

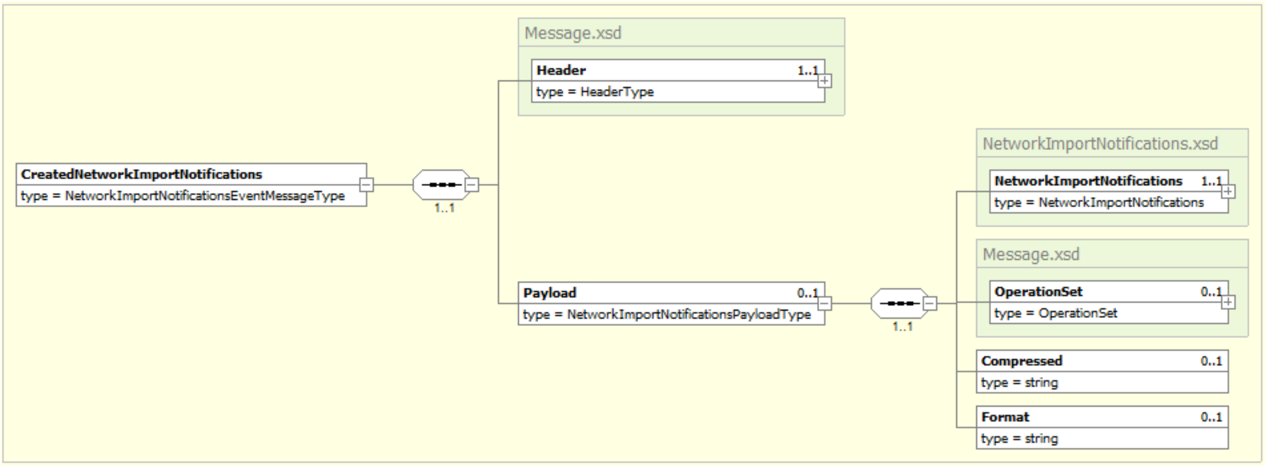
Message.xsd
Header 1..1
type = HeaderType
CreatedNetworkImportNotifications
type = NetworkImportNotificationsEventMessageType
1..1
NetworkImportNotifications.xsd
NetworkImportNotifications 1..1
type = NetworkImportNotifications
Message.xsd
OperationSet 0..1
type = OperationSet
Payload 0..1
type = NetworkImportNotificationsPayloadType
1..1
Compressed 0..1
type = string
Format 0..1
type = string

Figure 8.3 – The CreatedNetworkImportNotifications event message

The Payload sections carries the CIM defined profile (NetworkImportNotifications.xsd) for propagating extract’s status transitions to the external GIS system. The visual representation of the NetworkImportNotifications.xsd schema is given in Figure 8.4.

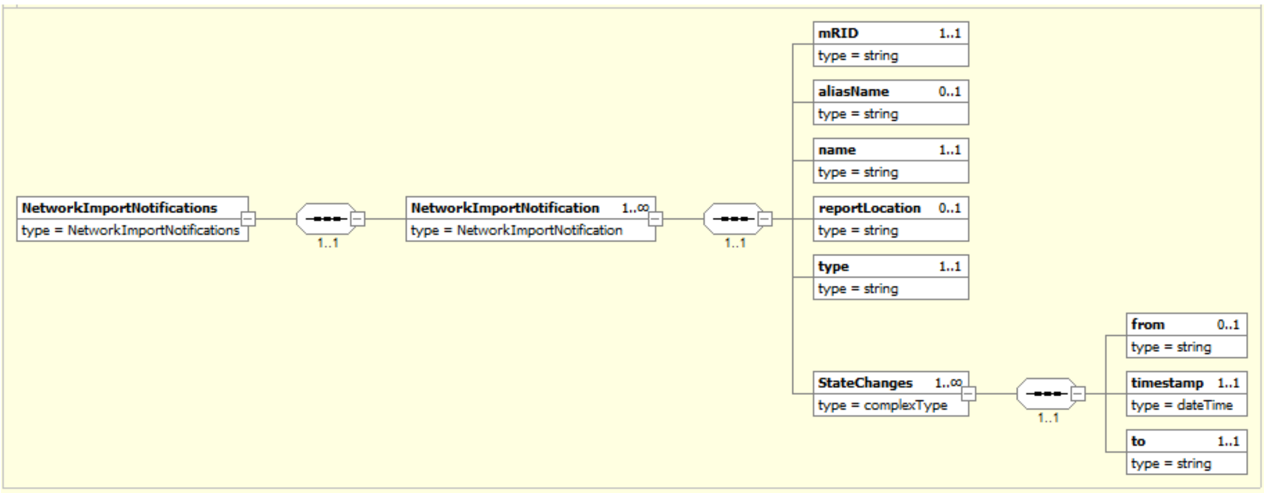

Figure 8.4 – NetworkImportNotifications.xsd

Table 8.2 defines the mapping between the *NetworkImportNotifications.xsd* and the appropriate entities in the CSRepo database.

Table 8.2 – The CreatedNetworkImportNotifications message → the CSRepo mapping

CreatedNIN Message			Description	CSRepo Database	
Section	Property	Type		Type	Table.Column
Header	**Verb**	String	Identifier for a specific action to be taken. Verb is created.	String	Populated by NIN Adapter
Header	**Noun**	String	Identifier for the subject of the action: NetworkImportNotifications.	String	Populated by NIN Adapter
Header	Revision	String	Revision of CIM standard used. Default value is 2.0.	String	Populated by NIN Adapter
Header	**Timestamp**	DateTime	Timestamp when message was produced.	DateTime	Populated by NIN Adapter
Header	Source	String	Source system that sends the message: EcoStruxure GridOps.	String	Populated by NIN Adapter
Header	**MessageID**	String	Unique message ID to be used for tracking messages.	String	Populated by NIN Adapter
Header	**CorrelationID**	String	Same as message ID.	String	Populated by NIN Adapter
Payload	**mRID**	String	Extract alias or changeset description.	String	[EXTRACT_INFO].[EXTRACT_ALIAS] [CHANGESET_INFO].[DESCRIPTION]
Payload	aliasName	String	Extract description. Empty for changeset	String	[EXTRACT_INFO].[DESCRIPTION]
Payload	**name**	String	Filename of the extract or changeset description.	String	[EXTRACT_INFO].[FILENAME] [CHANGESET_INFO].[DESCRIPTION]
Payload	**type**	String	Extract source name or changeset type name. Feeder, Customer, etc.	String	[EXTRACT_SOURCE].[SOURCE_NAME] [CHANGESET_TYPE].[TYPE_NAME]
Payload	**reportLocation**	String	Full path location of the report file on file share (SFTP or CIFS).	String	Populated by NIN Adapter
Payload	from	String	Source state of a transition.	String	[EXTRACT_HISTORY].[OLD_EXTRACT_STATE_ID] [CHANGESET_HISTORY].[OLD_CHANGESET_STATE_ID]
Payload	**timestamp**	DateTime	Date and time when extract changed its state.	DateTime	[EXTRACT_HISTORY].[TRANSITION_TIME] [CHANGESET_HISTORY].[TRANSITION_TIME]
Payload	**to**	String	Destination state of a transition.	String	[EXTRACT_HISTORY].[NEW_EXTRACT_STATE_ID] [CHANGESET_HISTORY].[NEW_CHANGESET_STATE_ID]

8.3.2. Response

The response message is depicted in Figure 8.5.

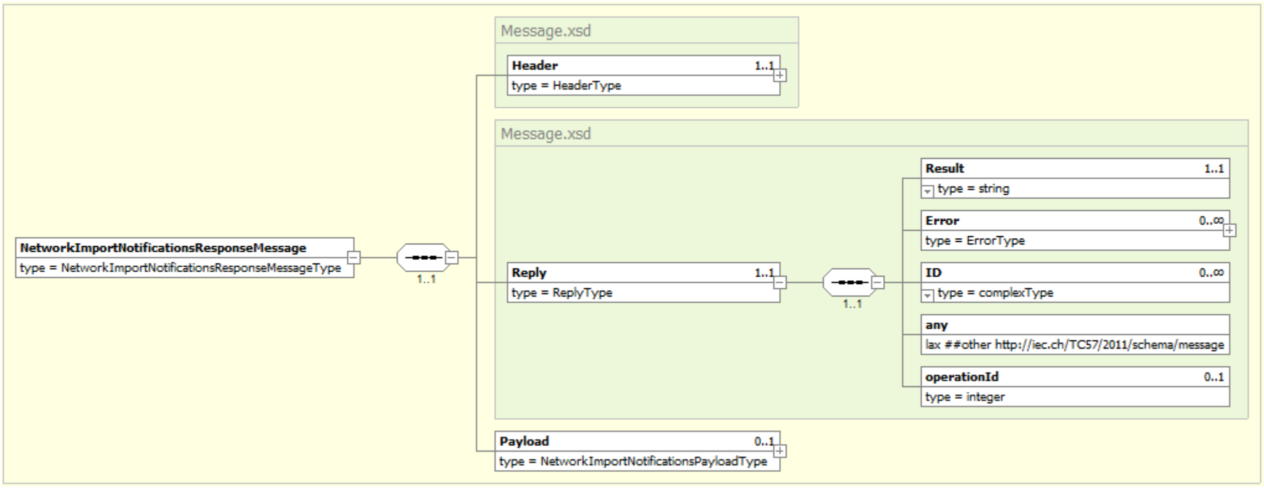

Figure 8.5 – The CreatedNetworkImportNotifications response message

8.3.3. Fault

The fault message is depicted in Figure 8.6.

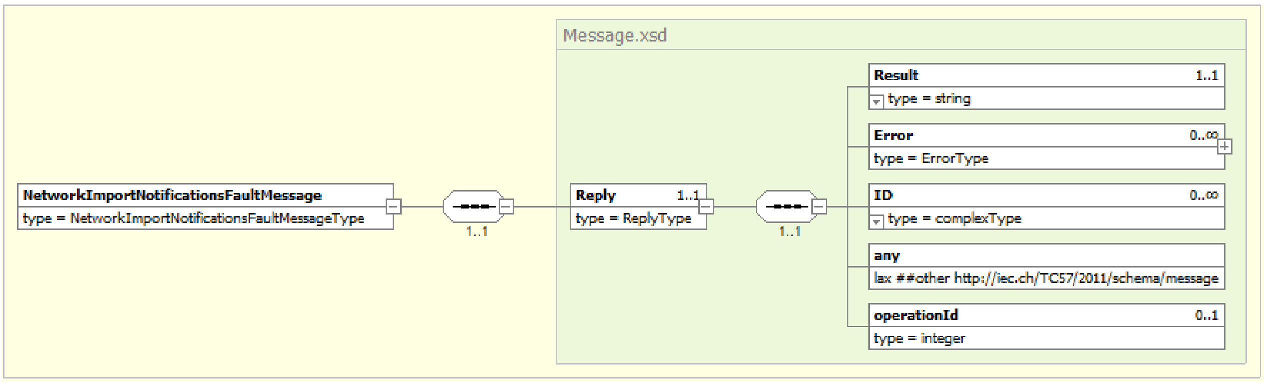

Figure 8.6 – The CreatedNetworkImportNotifications fault message

8.4. ChangedEquipmentStates Operation

The operation definition:

ChangedEquipmentStatesResponse ChangedEquipmentStates (ChangedEquipmentStatesEvent request)

8.4.1. Request

The *ChangedEquipmentStates* event message is defined according to the IEC 61968-100 and contains the following two sections:

- Header

- Payload

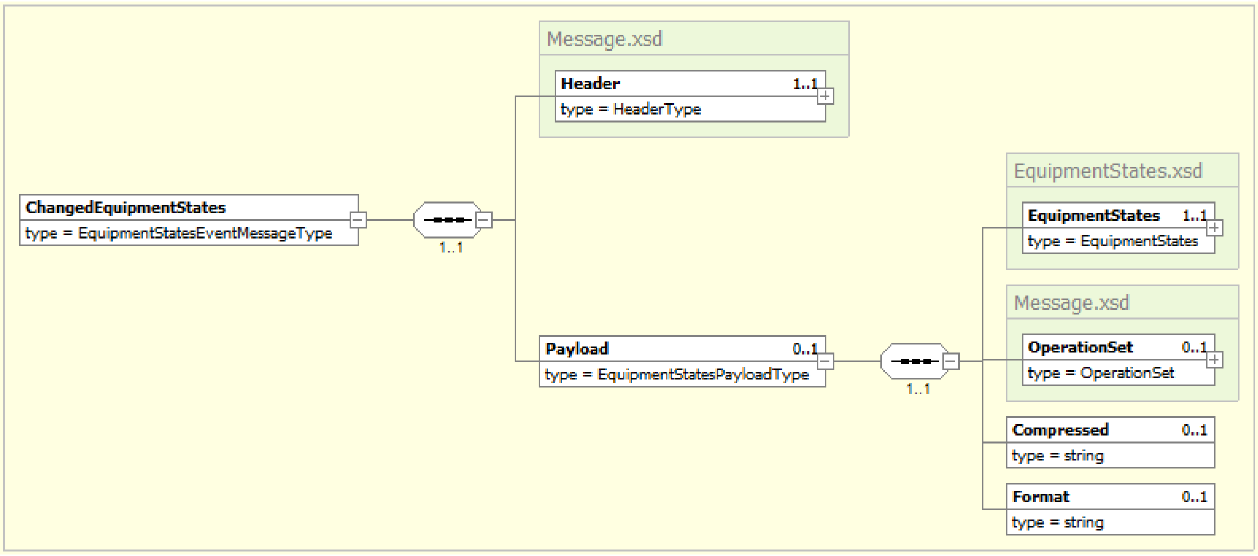

Figure 8.7 – The ChangedEquipmentStates message

The Payload sections carries the CIM defined profile (EquipmentStates.xsd) for propagating equipment state changes to the external GIS system. The visual representation of the EquipmentStates.xsd schema is given in Figure 8.8.

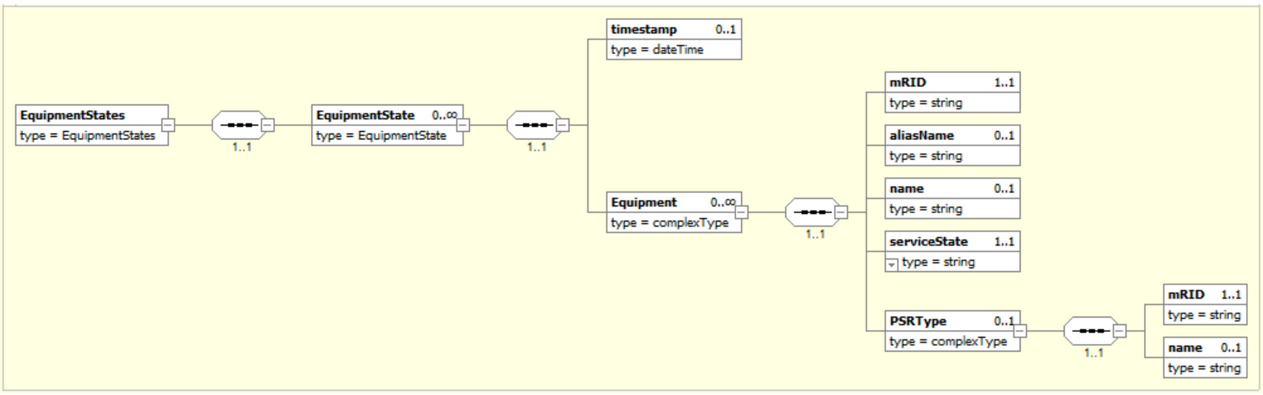

Figure 8.8 – EquipmentStates.xsd

Table 8.3 defines the mapping between the *EquipmentStates.xsd* and the appropriate attributes of Identified object.

Table 8.3 – ChangedEquipmentStates event message → model property mapping table

ChangedEquipmentStates Message			Description	CSRepo Database	
Section	Property	Type		Type	Table.Column
Header	**Verb**	String	Identifier for a specific action to be taken. Verb is created.	String	Populated by NIN Adapter
Header	**Noun**	String	Identifier for the subject of the action: EquipmentStates.	String	Populated by NIN Adapter
Header	Revision	String	Revision of CIM standard used. Default value is 2.0.	String	Populated by NIN Adapter
Header	**Timestamp**	DateTime	Timestamp when message was produced.	DateTime	Populated by NIN Adapter
Header	Source	String	Source system that sends the message: EcoStruxure GridOps	String	Populated by NIN Adapter
Header	**MessageID**	String	Unique message ID to be used for tracking messages.	String	Populated by NIN Adapter
Header	**CorrelationID**	String	Same as message ID.	String	Populated by NIN Adapter
Payload	timestamp	DateTime	Date and time when extract changed its state.	DateTime	[CHANGESET_HISTORY].[TRANSITION_TIME]
Payload	**Eqipment.mRID**	String	CustomId of the equipment whose service state has transitioned	String	IDOBJ_CUSTOMID
Payload	Eqipment.aliasName	String	Alias of the equipment whose service state has transitioned	String	IDOBJ_ALIAS
Payload	Eqipment.name	String	Name of the equipment whose service state has transitioned	String	IDOBJ_NAME
Payload	Eqipment.serviceState	String	New service state to which the equipment has transitioned	String	IDOBJ_SERVICESTATE
Payload	**Equipment.PSRType.mRID**	String	DMS type of the equipment whose service state has transitioned	String	IDOBJ_TYPE
Payload	Equipment.PSRType.name	String	Custom type of the equipment whose service state has transitioned	String	IDOBJ_CUSTOMTYPE

8.4.2. Response

The response message is depicted in Figure 8.9.

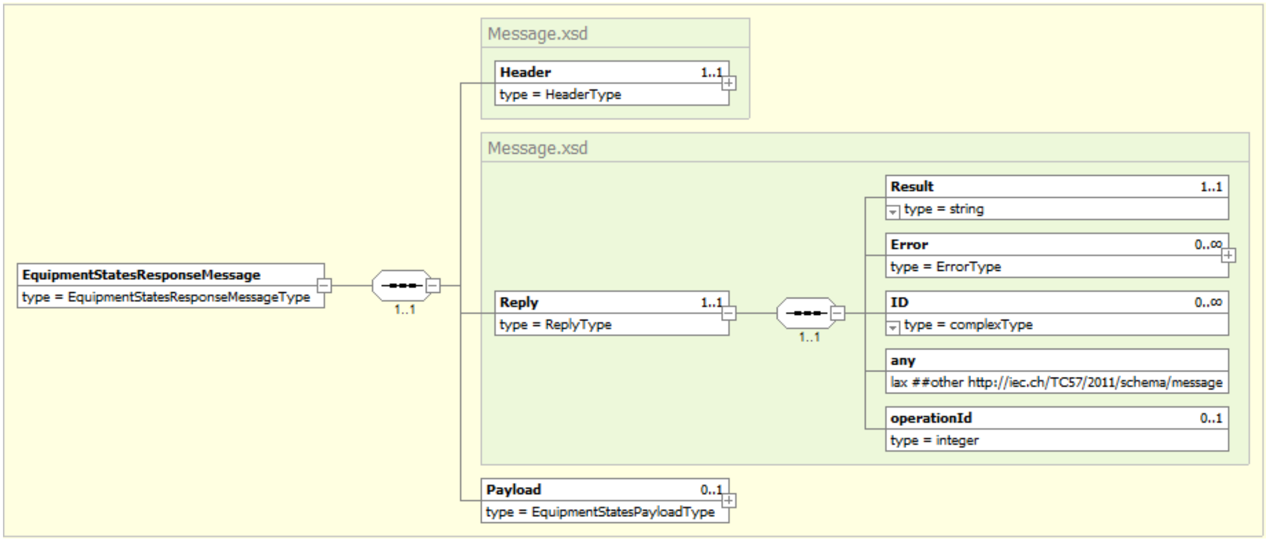

Figure 8.9 – The ChangedEquipmentStates response message

8.4.3. Fault

The fault message is depicted in Figure 8.10.

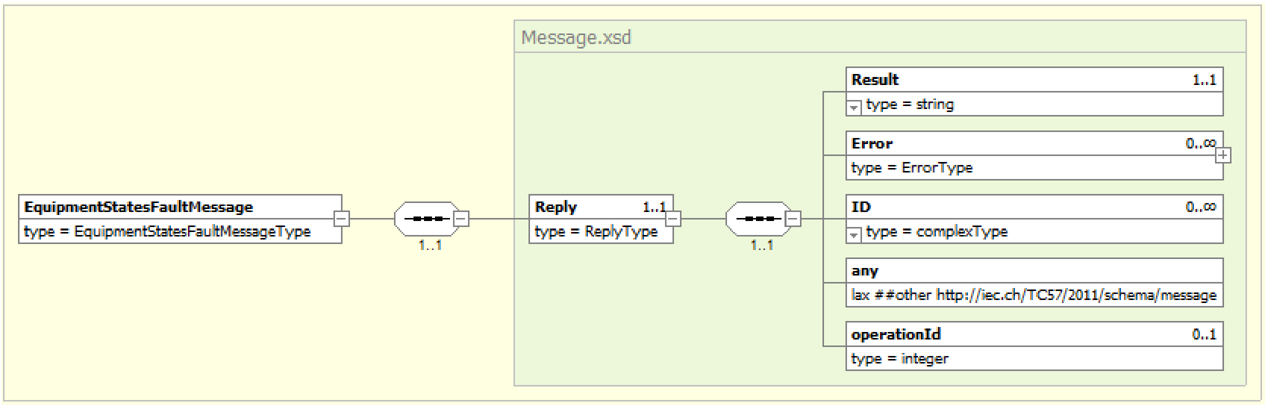

Figure 8.10 – The ChangedEquipmentStates fault message

9. DEPLOYMENT SPECIFICATION

Described in the *EcoStruxure GridOps Management Suite 3.10 Enterprise Integration Platform - Functional Specification* [1].

The deployment specification is provided in the following table:

Table 9.1 – The deployment specification

Deployment Specification	
Application	AdapterNIN
Critical process	Yes
OASyS service	OASyS DNA DMS_INTEGRATION Service
Servers	pdmz-int-1, pdmz-int-2, bdmz-int-1, bdmz-int-2
Zone	pdmz, bdmz
Installation Type	Product
Installation add-on name	Integration Adapters

Figure 9.1 depicts the standard deployment configuration for all NIN Integration participants.

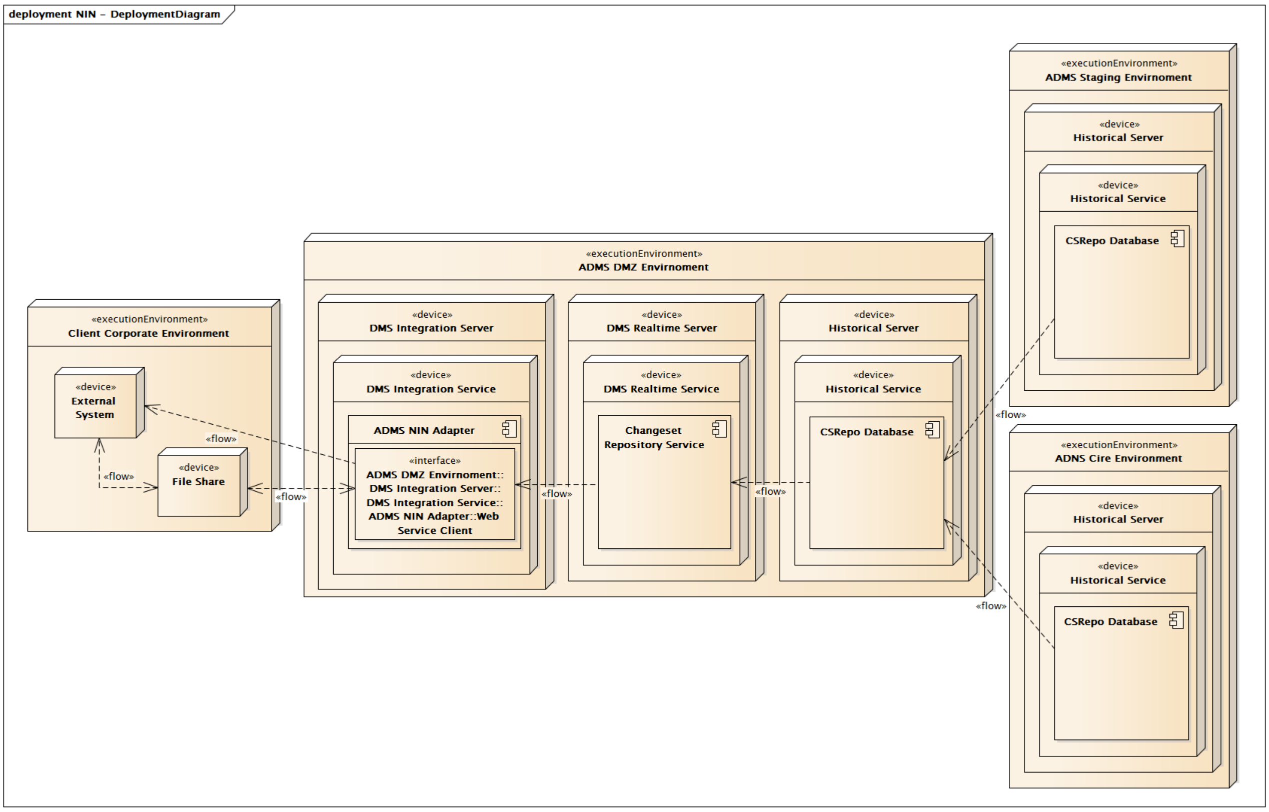

Figure 9.1 – The NIN Integration deployment diagram

10. INTERFACE CONFIGURATION

NIN adapter provides certain amount of configurability so that smaller adjustments in the functionality can be easily applied to the system, without interface down time. Such feature is provided through dedicated configuration files of the NIN adapter.

Table 10.1 – The configuration files specification

Name of the config file	Configuration File Description
AdapterNin	Registry configuration xml file
AdapterNin_WebServiceConfiguration	Web service configuration xml file
NINTriggersConfiguration	Filter configuration file for Network Import Notification process.
NINPlannedStatesTriggerConfiguration	Filter configuration file for Equipment States process.

Details about the structure and shared content of common interface configuration files are located in *EcoStruxure GridOps Management Suite 3.10 Enterprise Integration Platform - Functional Specification* [1]

Detailed content of above-mentioned configuration files is provided within the *Configuration* folder in the *EcoStruxure GridOps Management Suite 3.10 Network Import Notifications Interface.zip* file [4].

10.1. Triggers

As described throughout the document, the NIN Adapter is implemented in a generic way, meaning it can send state transitions for various types of extracts. Such functionality is enabled via appropriate configuration file where users can define for which type of extracts and their states, the message should be sent to external web service and import reports uploaded to external file share. Figure 10.1 shows XSD schema which describes the structure of the aforementioned XML configuration file.

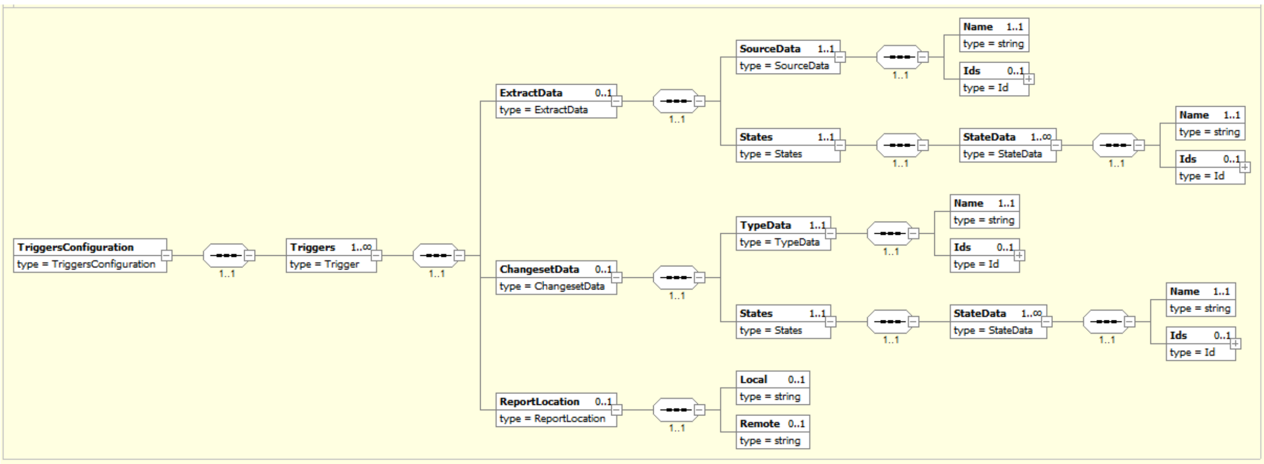

Figure 10.1 – Notification triggers configuration file XSD schema

Schema is split into three sections. Extract and Changeset data configuration, as well as report location. ExtractData contains information about extract types of interest, stored in the SourceData attribute, as well as extract states of interest (States attribute). ChangesetData contains information about changeset types of interest, stored in the TypeData attribute, as well as changeset states of interest. Report location stores information about local file path where import reports are stored upon creation and remote path where the import reports are uploaded.

Out of the box triggers configuration file is provided within the *Configuration* folder in the *EcoStruxure GridOps Management Suite 3.10 Network Import Notifications Interface.zip* file [4].

11. APPENDIX

11.1. WSDL

The WSDL file, XSD schemas and sample messages defined according to the IEC 61968-100 for the SendNetworkImportNotifications and SendEquipmentStates service is provided within the *Web Service Definitions* folder in the *EcoStruxure GridOps Management Suite 3.10 Network Import Notifications Interface.zip* file [4].

11.1. Message Examples

Message examples for several use cases are provided within the *Message Examples* folder in the *EcoStruxure GridOps Management Suite 3.10 Network Import Notifications Interface.zip* file [4].

12. RELEASE NOTES

The following new features related to Product File Monitoring Interface were introduced in the software, starting from version 3.8 SP1.

12.1. Software Version 3.8 SP1

Feature	Description
Report Location added to NetworkImportNotification message	Full path location to the report file was added to the NetworkImportNotification message so that claim check pattern can be utilized by the external system. In this way, external system can easily correlate report file with the state transition.
Planned Equipment States Notification	Nin adapter is extended with support for equipment service state transitions. Data is acquired from ChangesetRepository service, formatted accordingly and propagated to the dedicated ChangedEquipmentStates service hosted externally.

12.2. Software Version 3.9

Feature	Description
Triggering interval validation	Triggering time validation is added. If the user enters a value that is less than 1 min, the value will be defaulted to 1 min.

13. DEFINITIONS AND ABBREVIATIONS

Definition/Abbreviation	Description
ADMS	Advanced Distribution Management System
CIM	Common Information Model
Changeset	Changeset contains delta as a way to define what changes may be made to the model, but it also contains additional information needed for correct handling of those changes. Changeset can be created as a result of the import procedure, or manually by using Network Builder.
CSRepo	Storage of retrieved extracts and changesets created by one of the importer. CSRepo is present in DMZ, Staging (QA) and Core zones, and bidirectional data–pump replication is established between them.
CSS	Changeset Repository Service
DMD	Dynamic Mimic Diagram
DMZ	Demilitarized Zone (Access Service System)
ESB	Enterprise Service Bus
Extract	File (typically in CIM/XML format, in some cases .CSV) prepared by a source system to be used in EcoStruxure GridOps. File is retrieved from the distributed location and stored locally. When file is stored in CSRepo, additional information about the file is associated: extract type (GIS feeder, patch, EMS, ESCADA, etc.), time of storing, current status. In the context of CSRepo, extract represents not only content of the retrieved file, but also all additional info.
FTP	File Transfer Protocol
FS	File Share
LFS	Local File Share
NIS	Network Import Service
SFTP	SSH File Transfer Protocol
SOAP	Simple Object Access Protocol
WCF	Windows Communication Foundation
WS	Web Service
XML	Extensible Markup Language
XSD	XML Schema Definition