|  |  |
| --- | --- |
| 3GPP TS 28.532 V18.4.0 (2024-09) | |
| Technical Specification | |
| 3rd Generation Partnership Project;  Technical Specification Group Services and System Aspects;  Management and orchestration;  Generic management services  (Release 18) | |
|  | |
|  |  |
|  | |
| The present document has been developed within the 3rd Generation Partnership Project (3GPP TM) and may be further elaborated for the purposes of 3GPP. The present document has not been subject to any approval process by the 3GPPOrganizational Partners and shall not be implemented. This Specification is provided for future development work within 3GPPonly. The Organizational Partners accept no liability for any use of this Specification. Specifications and Reports for implementation of the 3GPP TM system should be obtained via the 3GPP Organizational Partners' Publications Offices. | |

|  |
| --- |
|  |
| ***3GPP***  Postal address  3GPP support office address  650 Route des Lucioles - Sophia Antipolis  Valbonne - FRANCE  Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16  Internet  https://www.3gpp.org |
| ***Copyright Notification***  No part may be reproduced except as authorized by written permission. The copyright and the foregoing restriction extend to reproduction in all media.  © 2024, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).  All rights reserved.  UMTS™ is a Trade Mark of ETSI registered for the benefit of its members  3GPP™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners LTE™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners  GSM® and the GSM logo are registered and owned by the GSM Association |

Contents

Foreword 15

1 Scope 17

2 References 17

3 Definitions and abbreviations 19

3.1 Definitions 19

3.2 Abbreviations 19

4 Overview 19

5 Void 19

6 Void 19

7 Void 19

8 Void 19

9 Void 19

10 Void 20

11 Management services – Stage 2 20

11.1 Generic provisioning management service 20

11.1.0 Introduction 20

11.1.1 Operations and notifications 20

11.1.1.1 createMOI operation 20

11.1.1.1.1 Description 20

11.1.1.1.2 Input parameters 21

11.1.1.1.3 Output parameters 21

11.1.1.1.4 Results 21

11.1.1.2 getMOIAttributes operation 22

11.1.1.2.1 Definition 22

11.1.1.2.2 Input Parameters 22

11.1.1.2.3 Output Parameters 23

11.1.1.2.4 Results 24

11.1.1.3 modifyMOIAttributes operation 24

11.1.1.3.1 Description 24

11.1.1.3.2 Input parameters 25

11.1.1.3.3 Output parameters 25

11.1.1.3.4 Results 26

11.1.1.4 deleteMOI operation 26

11.1.1.4.1 Description 26

11.1.1.4.2 Input parameters 26

11.1.1.4.3 Output parameters 26

11.1.1.4.4 Results 26

11.1.1.4a changeMOIs operation 26

11.1.1.4a.2 Input parameters 27

11.1.1.4a.3 Output parameters 27

11.1.1.5 Void 27

11.1.1.6 Void 27

11.1.1.7 Notification notifyMOICreation 28

11.1.1.7.1 Definition 28

11.1.1.7.2 Input parameters 29

11.1.1.7.3 Triggering event 30

11.1.1.7.3.1 From-state 30

11.1.1.7.3.2 To-state 30

11.1.1.8 Notification notifyMOIDeletion 30

11.1.1.8.1 Definition 30

11.1.1.8.2 Input parameters 31

11.1.1.8.3 Triggering event 32

11.1.1.8.3.1 From-state 32

11.1.1.8.3.2 To-state 32

11.1.1.9 Notification notifyMOIAttributeValueChanges 32

11.1.1.9.1 Definition 32

11.1.1.9.2 Input parameters 33

11.1.1.9.3 Triggering event 35

11.1.1.9.3.1 From-state 35

11.1.1.9.3.2 To-state 35

11.1.1.10 Notification notifyEvent 35

11.1.1.10.1 Definition 35

11.1.1.10.2 Input parameters 36

11.1.1.11 Notification notifyMOIChanges 36

11.1.1.11.1 Definition 36

11.1.1.11.2 Input parameters 38

11.1.2 Managed Information 41

11.1.2.1 ManagedEntity << ProxyClass>> 41

11.1.2.1.1 Definition 41

11.2 Generic fault supervision management service 41

11.2.1 Operations and notifications 41

11.2.1.1 Fault supervision data report 41

11.2.1.1.1 subscribe 41

11.2.1.1.1.1 Definition 41

11.2.1.1.1.2 Input parameters 41

11.2.1.1.1.3 Output parameters 41

11.2.1.1.1.4 Pre-condition 42

11.2.1.1.1.5 Post-condition 42

11.2.1.1.1.6 Exceptions 42

11.2.1.1.2 unsubscribe 43

11.2.1.1.2.1 Definition 43

11.2.1.1.2.2 Input parameters 43

11.2.1.1.2.3 Output parameters 43

11.2.1.1.2.4 Pre-condition 43

11.2.1.1.2.5 Post-condition 43

11.2.1.1.2.6 Exceptions 44

11.2.1.1.3 getAlarmList 44

11.2.1.1.3.1 Definition 44

11.2.1.1.3.2 Input parameters 44

11.2.1.1.3.3 Output parameters 45

11.2.1.1.3.4 Exceptions and constraints 48

11.2.1.1.4 notifyNewAlarm 48

11.2.1.1.4.1 Definition 48

11.2.1.1.4.2 Input parameters 48

11.2.1.1.4.2a Input parameters for notifications related to security alarms 50

11.2.1.1.4.3 Triggering event 50

11.2.1.1.4.3.1 From-state 50

11.2.1.1.4.3.2 To-state 51

11.2.1.1.5 notifyChangedAlarm 51

11.2.1.1.5.1 Definition 51

11.2.1.1.5.2 Input parameters 51

11.2.1.1.5.3 Triggering event 51

11.2.1.1.5.3.1 From-state 51

11.2.1.1.5.3.2 To-state 52

11.2.1.1.6 notifyAlarmListRebuilt 52

11.2.1.1.6.1 Definition 52

11.2.1.1.6.2 Input parameters 52

11.2.1.1.6.3 Triggering event 53

11.2.1.1.6.3.1 From-state 53

11.2.1.1.6.3.2 To-state 53

11.2.1.1.7 notifyCorrelatedNotificationChanged 53

11.2.1.1.7.1 Definition 53

11.2.1.1.7.2 Input parameters 53

11.2.1.1.7.3 Triggering event 53

11.2.1.1.7.3.1 From-state 53

11.2.1.1.7.3.2 To-state 54

11.2.1.1.8 getAlarmCount 54

11.2.1.1.8.1 Definition 54

11.2.1.1.8.2 Input parameters 54

11.2.1.1.8.3 Output parameters 55

11.2.1.1.8.4 Pre-condition 55

11.2.1.1.8.5 Post-condition 55

11.2.1.1.8.6 Exceptions 55

11.2.1.1.9 setComment 55

11.2.1.1.9.1 Definition 55

11.2.1.1.9.2 Input parameters 56

11.2.1.1.9.3 Output Parameters 56

11.2.1.2 Fault supervision data control 56

11.2.1.2.1 acknowledgeAlarms 56

11.2.1.2.1.1 Definition 56

11.2.1.2.1.2 Input parameters 56

11.2.1.2.1.3 Output parameters 57

11.2.1.2.1.4 Exceptions and constraints 57

11.2.1.2.2 unacknowledgeAlarms 57

11.2.1.2.2.1 Definition 57

11.2.1.2.2.2 Input parameters 58

11.2.1.2.2.3 Output parameters 58

11.2.1.2.2.4 Exceptions and constraints 59

11.2.1.2.3 clearAlarms 59

11.2.1.2.3.1 Definition 59

11.2.1.2.3.2 Input parameters 59

11.2.1.2.3.3 Output parameters 59

11.2.1.2.3.4 Exceptions and constraints 59

11.2.1.2.4 notifyClearedAlarm 59

11.2.1.2.4.1 Definition 59

11.2.1.2.4.2 Input parameters 60

11.2.1.2.4.3 Triggering event 60

11.2.1.2.4.3.1 From-state 60

11.2.1.2.4.3.2 To-state 60

11.2.1.2.5 notifyAckStateChanged 61

11.2.1.2.5.1 Definition 61

11.2.1.2.5.2 Input parameters 61

11.2.1.2.5.3 Triggering event 61

11.2.1.2.5.3.1 From-state 61

11.2.1.2.5.3.2 To-state 61

11.2.1.2.6 notifyComments 61

11.2.1.2.6.1 Definition 61

11.2.1.2.6.2 Input parameters 62

11.2.1.2.6.3 Trigger event 62

11.2.1.2.6.3.1 From-state 62

11.2.1.2.6.3.2 To-state 62

11.2.1.2.7 notifyPotentialFaultyAlarmList 62

11.2.1.2.7.1 Definition 62

11.2.1.2.7.2 Input parameters 63

11.2.1.2.7.3 Trigger event 63

11.2.1.2.7.3.1 From-state 63

11.2.1.2.7.3.2 To-state 63

11.2.1.2.8 notifyChangedAlarmGeneral 64

11.2.1.2.8.1 Definition 64

11.2.1.2.8.2 Input parameters for notifications related to non-security alarms 64

11.2.1.2.8.3 Input parameters for notifications related to security alarm 64

11.2.1.2.8.4 Trigger event 65

11.2.1.2.8.4.1 From-state 65

11.2.2 Managed information 66

11.2.2.1 Alarm information, alarm state change and Information Object Classes 66

11.2.2.1.1 Imported information entities and local labels 66

11.2.2.1.2 Class diagram 66

11.2.2.1.2.1 Introduction 66

11.2.2.1.2.2 Attributes and relationships 67

11.2.2.1.3 Information Object Class Definitions 67

11.2.2.1.3.1 AlarmInformation 67

11.2.2.1.3.1.1 Definition 67

11.2.2.1.3.1.2 Attribute 68

11.2.2.1.3.1.3 State diagram 68

11.2.2.1.3.2 AlarmList 70

11.2.2.1.3.2.1 Definition 70

11.2.2.1.3.2.2 Attribute 70

11.2.2.1.3.3 FSMnSProducer 71

11.2.2.1.3.3.1 Definition 71

11.2.2.1.3.3.2 Attribute 71

11.2.2.1.3.3.3 Notification Table 71

11.2.2.1.3.4 Comment 71

11.2.2.1.3.4.1 Definition 71

11.2.2.1.3.4.2 Attribute 71

11.2.2.1.3.5 CorrelatedNotification 71

11.2.2.1.3.5.1 Definition 71

11.2.2.1.3.5.2 Attribute 72

11.2.2.1.3.6 MonitoredEntity 72

11.2.2.1.3.6.1 Definition 72

11.2.2.1.3.6.2 Attribute 72

11.2.2.1.4 Information relationships definition 73

11.2.2.1.4.1 relation-FSMnSProducer-AlarmList (M) 73

11.2.2.1.4.1.1 Definition 73

11.2.2.1.4.1.2 Role 73

11.2.2.1.4.1.3 Constraint 73

11.2.2.1.4.2 relation-AlarmList-AlarmInformation (M) 73

11.2.2.1.4.2.1 Definition 73

11.2.2.1.4.2.2 Role 73

11.2.2.1.4.2.3 Constraint 73

11.2.2.1.4.3 relation-AlarmInformation-Comment (M) 73

11.2.2.1.4.3.1 Definition 73

11.2.2.1.4.3.2 Role 73

11.2.2.1.4.3.3 Constraint 73

11.2.2.1.4.4 relation-AlarmInformation-CorrelatedNotification (M) 73

11.2.2.1.4.4.1 Definition 73

11.2.2.1.4.4.2 Role 74

11.2.1.4.4.3 Constraint 74

11.2.2.1.4.5 relation-AlarmedObject-AlarmInformation (M) 74

11.2.2.1.4.5.1 Definition 74

11.2.2.1.4.5.2 Role 74

11.2.2.1.4.5.3 Constraint 74

11.2.2.1.4.6 relation-backUpObject-AlarmInformation (O) 74

11.2.2.1.4.6.1 Definition 74

11.2.2.1.4.6.2 Role 74

11.2.2.1.4.6.3 Constraint 74

11.2.2.1.5 Information attribute definition 75

11.2.2.1.5.1 Definition and legal values 75

11.2.2.1.5.2 Constraints 78

11.2.2.2 Subscription information, subscription state and Information Object Classes 79

11.2.2.2.1 Imported information entities and local labels 79

11.2.2.2.2 Class Diagram 79

11.2.2.2.2.1 Attributes and relationships 79

11.2.2.2.2.2 Inheritance 79

11.2.2.2.3 Information object classes definition 80

11.2.2.2.3.1 NtfSubscriber 80

11.2.2.2.3.1.1 Definition 80

11.2.2.2.3.1.2 Attributes 80

11.2.2.2.3.2 NtfSubscription 80

11.2.2.2.3.2.1 Definition 80

11.2.2.2.3.2.2 Attributes 80

11.2.2.2.3.2.3 Void 80

11.2.2.2.3.3 NotificationIRP 80

11.2.2.2.3.3.1 Definition 80

11.2.2.2.4 Information relationship definitions 81

11.2.2.2.4.1 relation-ntfSubscriber-ntfSubscription (M) 81

11.2.2.2.4.1.1 Definition 81

11.2.2.2.4.1.2 Roles 81

11.2.2.2.4.1.3 Constraints 81

11.2.2.2.4.2 relation-ntfIRP-ntfSubscriber (M) 81

11.2.2.2.4.2.1 Definition 81

11.2.2.2.4.2.2 Roles 81

11.2.2.2.4.2.3 Constraints 81

11.2.2.2.5 Information attribute definitions 82

11.2.2.2.5.0 Introduction 82

11.2.2.2.5.1 Definitions and legal values 82

11.2.2.2.5.2 Constraints 82

11.3 Performance assurance 82

11.3.1 Operations and notifications 82

11.3.1.1 Void 82

11.3.1.2 Void 82

11.3.1.3 Notification notifyThresholdCrossing 82

11.3.1.3.1 Definition 82

11.3.1.3.2 Notification information 83

11.3.2 Managed information 83

11.3.2.1 Performance data file 83

11.3.2.1.1 Void 83

11.3.2.1.2 Performance data file content description 83

11.3.2.1.3 Void 85

11.3.2.1.3.1 Void 85

11.3.2.1.3.2 Void 85

11.3.2.1.4 Performance data file naming convention 85

11.3.2.1.4 Void 86

11.4 Heartbeat notification 86

11.4.1 Operations and notifications 86

11.4.1.1 Notification notifyHeartbeat 86

11.4.1.1.1 Definition 86

11.4.1.1.2 Input parameters 87

11.4.1.1.3 Triggering event 87

11.4.1.1.3.1 From-state 87

11.4.1.1.3.2 To-state 87

11.5 Streaming data reporting service 87

11.5.1 Operations and notifications 87

11.5.1.1 establishStreamingConnection operation (M) 87

11.5.1.1.1 Definition 87

11.5.1.1.2 Input parameters 88

11.5.1.1.3 Output parameters 88

11.5.1.1.4 Exceptions 89

11.5.1.2 terminateStreamingConnection operation (M) 89

11.5.1.2.1 Definition 89

11.5.1.2.2 Input parameters 89

11.5.1.2.3 Output parameters 89

11.5.1.2.4 Exceptions 89

11.5.1.3 reportStreamData operation (M) 89

11.5.1.3.1 Definition 89

11.5.1.3.2 Input parameters 89

11.5.1.3.3 Output parameters 90

11.5.1.3.4 Exceptions 90

11.5.1.4 addStream operation (M) 90

11.5.1.4.1 Definition 90

11.5.1.4.2 Input parameters 91

11.5.1.4.3 Output parameters 92

11.5.1.4.4 Exceptions 92

11.5.1.5 deleteStream operation (M) 93

11.5.1.5.1 Definition 93

11.5.1.5.2 Input parameters 93

11.5.1.5.3 Output parameters 93

11.5.1.5.4 Exceptions 93

11.5.1.6 getConnectionInfo operation (M) 93

11.5.1.6.1 Definition 93

11.5.1.6.2 Input parameters 93

11.5.1.6.3 Output parameters 94

11.5.1.6.4 Exceptions 94

11.5.1.7 getStreamInfo operation (M) 94

11.5.1.7.1 Definition 94

11.5.1.7.2 Input parameters 94

11.5.1.7.3 Output parameters 95

11.5.1.7.4 Exceptions 97

11.6 File data reporting service 97

11.6.1 Operations and notifications 97

11.6.1.1 Notification notifyFileReady 97

11.6.1.1.1 Definition 97

11.6.1.1.2 Input parameters 98

11.6.1.2 Notification notifyFilePreparationError 100

11.6.1.2.1 Definition 100

11.6.1.2.2 Input parameters 101

11.6.1.3 Operation subscribe 101

11.6.1.3.1 Definition 101

11.6.1.3.2 Input parameters 101

11.6.1.3.3 Output parameters 102

11.6.1.3.4 Exceptions 102

11.6.1.4 Operation unsubscribe 102

11.6.1.4.1 Definition 102

11.6.1.4.2 Input parameters 102

11.6.1.4.3 Output parameters 102

11.6.1.4.4 Exceptions 102

11.6.1.5 Operation listAvailableFiles 103

11.6.1.5.1 Definition 103

11.6.1.5.2 Input parameters 103

11.6.1.5.3 Output parameters 103

11.6.1.5.4 Exceptions 103

11.6.2 File transfer protocols 103

12 Management services – Stage 3 104

12.1 Generic provisioning management service 104

12.1.1 RESTful HTTP-based solution set 104

12.1.1.1 Mapping of operations 104

12.1.1.1.1 Introduction 104

12.1.1.1.2 Operation createMOI 104

12.1.1.1.3 Operation getMOIAttributes 104

12.1.1.1.4 Operation modifyMOIAttributes 105

12.1.1.1.4.1 Mapping to HTTP PUT 105

12.1.1.1.4.2 Mapping to HTTP PATCH 106

12.1.1.1.5 Operation deleteMOI 106

12.1.1.1.6 Void 107

12.1.1.1.7 Void 107

12.1.1.2 Mapping of notifications 107

12.1.1.2.1 Introduction 107

12.1.1.2.2 Notification notifyMOICreation 107

12.1.1.2.3 Notification notifyMOIDeletion 108

12.1.1.2.4 Notification notifyMOIAttributeValueChanges 108

12.1.1.2.5 Notification notifyMOIChanges 108

12.1.1.3 Resources 108

12.1.1.3.1 Resource structure 108

12.1.1.3.1.2 Resource structure on the MnS consumer 109

12.1.1.3.2 Resource definitions 109

12.1.1.3.2.1 Resource "…/{className}={id}" 109

12.1.1.3.2.1.1 Description 109

12.1.1.3.2.1.2 URI 109

12.1.1.3.2.1.3 HTTP methods 110

12.1.1.3.2.2 Void 112

12.1.1.3.2.3 Void 112

12.1.1.3.2.4 Resource "{notificationTarget}" 112

12.1.1.3.2.4.1 Description 112

12.1.1.3.2.4.2 URI 112

12.1.1.3.2.4.3 HTTP methods 112

12.1.1.4 Data type definitions 113

12.1.1.4.1 General 113

12.1.1.4.1a Structured data types 114

12.1.1.4.1a.1 Type Resource 114

12.1.1.4.1a.2 Type Scope 114

12.1.1.4.1a.3 Type CorrelatedNotification 114

12.1.1.4.1a.4 Type MoiChange 115

12.1.1.4.1a.5 Type NotifyMoiCreation 119

12.1.1.4.1a.6 Type NotifyMoiDeletion 120

12.1.1.4.1a.7 Type NotifyMoiAttributeValueChanges 121

12.1.1.4.1a.8 Type NotifyMoiChanges 122

12.1.1.4.1a.9 Type PatchItem 123

12.1.1.4.2 Void 123

12.1.1.4.3 Void 123

12.1.1.4.4 Simple data types and enumerations 123

12.1.1.4.4.7 Enumeration PatchOperation 124

12.1.2 RESTful HTTP-based solution set for integration with ONAP VES API 124

12.1.2.1 Mapping of operations 124

12.1.2.2 Mapping of notifications 125

12.1.2.2.1 Introduction 125

12.1.2.2.1.1 General 125

12.1.2.2.1.2 Void 125

12.1.2.2.2 Notification notifyMOICreation 125

12.1.2.2.3 Notification notifyMOIDeletion 125

12.1.2.2.4 Notification notifyMOIAttributeValueChange 125

12.1.2.2.5 Notification notifyMOIChanges 125

12.1.2.3 Resources 125

12.1.2.3.1 Resource structure 125

12.1.2.3.2 Resource definitions 126

12.1.2.4 Data type definitions 126

12.1.3 YANG/Netconf-based solution set 126

12.1.3.1 Mapping of operations 126

12.1.3.1.1 Introduction 126

12.1.3.1.2 Operation createMOI 126

12.1.3.1.3 Operation getMOIAttributes 127

12.1.3.1.4 Operation modifyMOIAttributes 129

12.1.3.1.4a Operation changeMOIs 130

12.1.3.1.5 Operation deleteMOI 131

12.1.3.2 Mapping of notifications 131

12.1.3.2.1 Introduction 131

12.1.3.2.5 Notification notifyMOIChanges 132

12.1.3.3 Netconf Server behavior 136

12.1.3.3.1 Introduction 136

12.1.3.3.2 Implement IETF RFC 6243: “With-defaults Capability for NETCONF” 136

12.2 Generic fault supervision management service 136

12.2.1 RESTful HTTP-based solution set 136

12.2.1.1 Mapping of operations 136

12.2.1.1.1 Introduction 136

12.2.1.1.2 Operation getAlarmList 136

12.2.1.1.3 Operation getAlarmCount 137

12.2.1.1.4 Operation setComment 138

12.2.1.1.5 Operation acknowledgeAlarms 138

12.2.1.1.6 Operation unacknowledgeAlarms 140

12.2.1.1.7 Operation clearAlarms 141

12.2.1.1.8 Operation subscribe 142

12.2.1.1.9 Operation unsubscribe 143

12.2.1.2 Mapping of notifications 143

12.2.1.2.1 Introduction 143

12.2.1.2.2 Notification notifyNewAlarm (non-security alarm) 144

12.2.1.2.3 Notification notifyNewAlarm (security alarm) 144

12.2.1.2.4 Notification notifyAckStateChanged 144

12.2.1.2.5 Notification notifyClearedAlarm 145

12.2.1.2.6 Notification notifyAlarmListRebuilt 145

12.2.1.2.7 Notification notifyChangedAlarm 145

12.2.1.2.8 Notification notifyComments 146

12.2.1.2.9 Notification notifyPotentialFaultyAlarmList 146

12.2.1.2.10 Notification notifyCorrelatedNotificationChanged 146

12.2.1.2.11 Notification notifyChangedAlarmGeneral (non-security alarm) 147

12.2.1.2.12 Notification notifyChangedAlarmGeneral (security alarm) 147

12.2.1.3 Resources 148

12.2.1.3.1 Resource structure 148

12.2.1.3.1.2 Resource structure on the MnS consumer 148

12.2.1.3.2 Resource definitions 149

12.2.1.3.2.1 Resource "…/alarms" 149

12.2.1.3.2.1.1 Description 149

12.2.1.3.2.1.2 URI 149

12.2.1.3.2.1.3 HTTP methods 149

12.2.1.3.2.2 Resource "…/alarms /{alarmId}" 150

12.2.1.3.2.2.1 Description 150

12.2.1.3.2.2.2 URI 150

12.2.1.3.2.2.3 HTTP methods 150

12.2.1.3.2.3 Resource "…/alarms/alarmCount" 151

12.2.1.3.2.3.1 Definition 151

12.2.1.3.2.3.2 URI 151

12.2.1.3.2.3.3 HTTP methods 151

12.2.1.3.2.4 Resource "…/alarms/{alarmId}/comments" 152

12.2.1.3.2.4.1 Definition 152

12.2.1.3.2.4.2 URI 152

12.2.1.3.2.4.3 HTTP methods 152

12.2.1.3.2.5 Resource "…/comments/{commentId}" 153

12.2.1.3.2.5.1 Definition 153

12.2.1.3.2.5.2 URI 153

12.2.1.3.2.5.3 HTTP methods 153

12.2.1.3.2.6 Resource "…/subscriptions" 153

12.2.1.3.2.6.1 Description 153

12.2.1.3.2.6.2 URI 153

12.2.1.3.2.6.3 HTTP methods 153

12.2.1.3.2.7 Resource "…/subscriptions/{subscriptionId}" 154

12.2.1.3.2.7.1 Description 154

12.2.1.3.2.7.2 URI 154

12.2.1.3.2.7.3 HTTP methods 154

12.2.1.3.2.8 Resource "{notificationTarget}" 155

12.2.1.3.2.8.1 Description 155

12.2.1.3.2.8.2 URI 155

12.2.1.3.2.8.3 HTTP methods 155

12.2.1.4 Data type definitions 155

12.2.1.4.1 General 155

12.2.1.4.1a Structured data types 157

12.2.1.4.1a.1 Type ThresholdHysteresis 157

12.2.1.4.1a.2 Type ThresholdLevelInd 157

12.2.1.4.1a.3 Type ThresholdInfo 158

12.2.1.4.1a.4 Type CorrelatedNotification 158

12.2.1.4.1a.5 Type AlarmRecord 159

12.2.1.4.1a.6 Type AlarmCount 161

12.2.1.4.1a.7 Type Comment 161

12.2.1.4.1a.8 Type Subscription 161

12.2.1.4.1a.9 Type MergePatchAcknowledgeAlarm 161

12.2.1.4.1a.10 Type MergePatchClearAlarm 161

12.2.1.4.1a.11 Type FailedAlarm 162

12.2.1.4.1a.12 Type NotifyNewAlarm 162

12.2.1.4.1a.13 Type NotifyNewSecAlarm 163

12.2.1.4.1a.14 Type NotifyClearedAlarm 163

12.2.1.4.1a.15 Type NotifyChangedAlarm 164

12.2.1.4.1a.16 Type NotifyChangedAlarmGeneral 164

12.2.1.4.1a.17 Type NotifyChangedSecAlarmGeneral 165

12.2.1.4.1a.18 Type NotifyCorrelatedNotificationChanged 165

12.2.1.4.1a.19 Type NotifyAckStateChanged 166

12.2.1.4.1a.20 Type NotifyComments 166

12.2.1.4.1a.21 Type NotifyPotentialFaultyAlarmList 166

12.2.1.4.1a.22 Type NotifyAlarmListRebuilt 167

12.2.1.4.2 Void 167

12.2.1.4.3 Void 167

12.2.1.4.4 Simple data types and enumerations 167

12.2.1.4.4.1 General 167

12.2.1.4.4.2 Simple data types 167

12.2.1.4.4.3 Enumeration AlarmAckState 167

12.2.1.4.4.4 Enumeration AckState 168

12.2.1.4.4.5 Enumeration AlarmListAlignmentRequirement 168

12.2.1.4.4.6 Enumeration AlarmType 168

12.2.1.4.4.7 Enumeration ProbableCause 169

12.2.1.4.4.8 Enumeration AlarmNotificationTypes 169

12.2.1.4.4.9 Enumeration PerceivedSeverity 169

12.2.1.4.4.10 Enumeration TrendIndication 169

12.2.2 RESTful HTTP-based solution set for integration with ONAP VES API 170

12.2.2.1 Mapping of operations 170

12.2.2.2 Mapping of notifications 170

12.2.2.2.1 Introduction 170

12.2.2.2.1.1 General 170

12.2.2.2.1.2 Void 170

12.2.2.2.2 Notification notifyNewAlarm (non-security alarm) 170

12.2.2.2.3 Notification notifyNewAlarm (security alarm) 170

12.2.2.2.4 Notification notifyAckStateChanged 170

12.2.2.2.5 Notification notifyClearedAlarm 170

12.2.2.2.6 Notification notifyAlarmListRebuilt 170

12.2.2.2.7 Notification notifyChangedAlarm 170

12.2.2.2.8 Notification notifyComments 170

12.2.2.2.9 Notification notifyPotentialFaultyAlarmList 171

12.2.2.2.10 Notification notifyCorrelatedNotificationChanged 171

12.2.2.2.11 Notification notifyChangedAlarmGeneral (non-security alarm) 171

12.2.2.2.12 Notification notifyChangedAlarmGeneral (security alarm) 171

12.2.2.3 Resources 171

12.2.2.3.1 Resource structure 171

12.2.2.3.2 Resource definitions 171

12.2.2.4 Data type definitions 171

12.3 Generic performance assurance management service 172

12.3.1 RESTful HTTP-based solution set 172

12.3.1.1 Void 172

12.3.1.2 Performance threshold monitoring service 172

12.3.1.2.1 Mapping of operations 172

12.3.1.2.2 Mapping of notifications 172

12.3.1.2.2.1 Introduction 172

12.3.1.2.2.2 Notification notifyThresholdCrossing 172

12.3.1.2.3 Resources 172

12.3.1.2.3.1 Resource structure 172

12.3.1.2.3.2 Resource definitions 173

12.3.1.2.3.2.1 Resource "/notificationSink" 173

12.3.1.2.4 Data type definitions 173

12.3.1.2.4.1 General 173

12.3.1.2.4.2 Structured data types 174

12.3.1.2.4.2.1 Type NotifyThresholdCrossing 174

12.3.1.2.4.4 Void 174

12.3.1.2.4.5 Void 174

12.3.1.2.4.6 Simple data types and enumerations 174

12.3.1.2.4.6.1 General 174

12.3.1.2.4.6.2 Simple data types 175

12.3.1.2.4.6.3 Enumeration PerfNotificationTypes 175

12.3.1.2.4.6.4 Enumeration PerfMetricDirection 175

12.3.2 Performance data XML file format definition 175

12.3.2.1 Introduction 175

12.3.2.2 Mapping table 175

12.3.2.3 Void 176

12.3.2.3.1 Void 176

12.3.2.3.2 Void 176

12.3.2.4 XML schema 176

12.4 Heartbeat 179

12.4.1 RESTful HTTP-based solution set 179

12.4.1.1 Mapping of operations 179

12.4.1.2 Mapping of notifications 179

12.4.1.2.1 Introduction 179

12.4.1.2.2 Notification "notifyHeartbeat" 179

12.4.1.3 Usage of HTTP 179

12.4.1.4 Resources 179

12.4.1.5 Data type definitions 179

12.4.1.5.1 General 179

12.4.1.5.2 Structured data types 180

12.4.1.5.3 Simple data types and enumerations 180

12.4.1.5.3.1 General 180

12.4.1.5.3.2 Simple data types 180

12.4.1.5.3.3 Enumeration HeartbeatNotificationTypes 180

12.4.2 RESTful HTTP-based solution set for integration with ONAP VES API 180

12.4.2.1 Mapping of operations 180

12.4.2.2 Mapping of notifications 180

12.4.2.2.1 Introduction 180

12.4.2.2.1.1 General 180

12.4.2.2.1.2 Notification parameter mapping principles 181

12.4.2.2.2 Notification notifyHeartbeat 181

12.5 Streaming data reporting service 181

12.5.1 RESTful HTTP-based solution set 181

12.5.1.1 Mapping of operations 181

12.5.1.1.1 Introduction 181

12.5.1.1.2 Operation "establishStreamingConnection" 181

12.5.1.1.3 Operation "terminateStreamingConnection" 184

12.5.1.1.4 Operation "reportStreamData" 184

12.5.1.1.5 Operation "addStream" 185

12.5.1.1.6 Operation "deleteStream" 185

12.5.1.1.7 Operation "getConnectionInfo" 186

12.5.1.1.8 Operation "getStreamInfo" 186

12.5.1.2 Mapping of notifications 187

12.5.1.3 Resources 187

12.5.1.3.1 Resources structure 187

12.5.1.3.2 Resources definitions 187

12.5.1.4 Data type definitions 194

12.5.1.4.1 General 194

12.5.1.4.2 Query, message body and resource data types 195

12.5.1.4.3 Simple data types and enumerations 196

12.6 File data reporting service 197

12.6.1 RESTful HTTP-based solution set 197

12.6.1.1 Mapping of operations 197

12.6.1.1.1 Introduction 197

12.6.1.1.2 Operation listAvailableFiles 197

12.6.1.1.3 Operation subscribe 198

12.6.1.1.4 Operation unsubscribe 198

12.6.1.2 Mapping of notifications 198

12.6.1.2.1 Introduction 198

12.6.1.2.2 Notification notifyFileReady 198

12.6.1.2.3 Notification notifyFilePreparationError 198

12.6.1.3 Resources 199

12.6.1.3.1 Resource structure 199

12.6.1.3.1.1 Resource structure on the MnS producer 199

12.6.1.3.1.2 Resource structure on the MnS consumer 199

12.6.1.3.2 Resource definitions 199

12.6.1.4 Data type definitions 203

12.6.1.4.1 General 203

12.6.1.4.2 Structured data types 203

12.6.1.4.3 Void 204

12.6.1.4.4 Void 204

12.6.1.4.5 Void 204

12.6.1.4.6 Simple data types and enumerations 204

Annex A (normative): OpenAPI specification 206

A.0 Introduction 206

A.1 Provisioning management service 206

A.1.0 Introduction 206

A.1.1 OpenAPI document "TS28532\_ProvMnS.yaml" 206

A.1.2 Integration with ONAP VES 213

A.2 Generic fault supervision management service 213

A.2.0 Introduction 213

A.2.1 OpenAPI document "TS28532\_FaultMnS.yaml" 213

A.2.2 Integration with ONAP VES 227

A.3 Void 227

A.4 Generic performance assurance management service 227

A.4.1 Void 227

A.4.2 OpenAPI document "TS28532\_PerfMnS.yaml" 227

A.4.3 Integration with ONAP VES 228

A.5 Heartbeat 228

A.5.0 Introduction 228

A.5.1 OpenAPI document "TS28532\_HeartbeatNtf.yaml" 228

A.5.2 Integration with ONAP VES 229

A.6 Streaming data reporting management service 229

A.6.1 Introduction 229

A.6.2 OpenAPI document "TS28532\_StreamingDataMnS.yaml" 229

A.7 File data reporting management service 235

A.7.1 Introduction 235

A.7.2 OpenAPI document "TS28532\_FileDataReportingMnS.yaml" 235

A.7.3 Integration with ONAP VES 238

Annex B (Informative): Guidelines for the integration of 3GPP MnS notifications with ONAP VES 239

Annex C (informative): Change history 240

# Foreword

This Technical Specification has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

In drafting the TS/TR, pay particular attention to the use of modal auxiliary verbs! TRs shall not contain any normative provisions.

In the present document, modal verbs have the following meanings:

**shall** indicates a mandatory requirement to do something

**shall not** indicates an interdiction (prohibition) to do something

The constructions "shall" and "shall not" are confined to the context of normative provisions, and do not appear in Technical Reports.

The constructions "must" and "must not" are not used as substitutes for "shall" and "shall not". Their use is avoided insofar as possible, and they are not used in a normative context except in a direct citation from an external, referenced, non-3GPP document, or so as to maintain continuity of style when extending or modifying the provisions of such a referenced document.

**should** indicates a recommendation to do something

**should not** indicates a recommendation not to do something

**may** indicates permission to do something

**need not** indicates permission not to do something

The construction "may not" is ambiguous and is not used in normative elements. The unambiguous constructions "might not" or "shall not" are used instead, depending upon the meaning intended.

**can** indicates that something is possible

**cannot** indicates that something is impossible

The constructions "can" and "cannot" are not substitutes for "may" and "need not".

**will** indicates that something is certain or expected to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

**will not** indicates that something is certain or expected not to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

**might** indicates a likelihood that something will happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

**might not** indicates a likelihood that something will not happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

In addition:

**is** (or any other verb in the indicative mood) indicates a statement of fact

**is not** (or any other negative verb in the indicative mood) indicates a statement of fact

The constructions "is" and "is not" do not indicate requirements.

# 1 Scope

The present document specifies the stage 2 and stage 3 of generic management services for mobile network.

# 2 References

- The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non‑specific.

- For a specific reference, subsequent revisions do not apply.

- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2] Void

[3] 3GPP TS 28.541: "Management and orchestration ; 5G Network Resource Model (NRM); Stage 2 and stage3".

[4] ITU-T Recommendation X.733 (02/92): "Information technology - Open Systems Interconnection - Systems Management: Alarm reporting function".

[5] 3GPP TS 28.531: "Management and orchestration ; Provisioning; ".

[6] 3GPP TS 28.554: "Management and orchestration ; 5G end to end Key Performance Indicators (KPI)".

[7] Void

[8] Void

[9] Void

[10] Void

[11] 3GPP TS 28.622: "Telecommunication management; Generic Network Resource Model (NRM) Integration Reference Point (IRP); Information Service (IS)".

[12] Void

[13] 3GPP TS 28.533: "Management and orchestration; Architecture framework"

[14] Void

[15] 3GPP TS 32.158: "Management and orchestration; Design rules for REpresentational State Transfer (REST) Solution Sets (SS)".

[16] Void

[17] Void

[18] 3GPP TS 28.552: "Management and orchestration; 5G performance measurements".

[19] 3GPP TS 32.401: "Telecommunication management; Performance Measurement (PM); Concept and requirements".

[20] ISO 8601:2004: "Data elements and interchange formats – Information interchange – Representation of dates and times".

[21] Void.

[22] Void.

[23] Void.

[24] Void.

[25] 3GPP TS 32.300: "Telecommunication management; Configuration Management (CM); Name convention for Managed Objects ".

[26] W3C REC-xmlschema-0-20010502: "XML Schema Part 0: Primer".

[27] W3C REC-xmlschema-1-20010502: "XML Schema Part 1: Structures".

[28] W3C REC-xmlschema-2-20010502: "XML Schema Part 2: Datatypes".

[29] W3C REC-xml-names-19990114: "Namespaces in XML".

[30] Void.

[31] Void.

[32] IETF RFC 6241 "Network Configuration Protocol (NETCONF)".

[33] 3GPP TS 32.160 " Management and orchestration; Management service template ".

[34] IETF RFC 7950 "The YANG 1.1 Data Modeling Language".

[35] Void

[36] IETF RFC 6902: "JavaScript Object Notation (JSON) Patch".

[37] IETF RFC 7396: "JSON Merge Patch".

[38] 3GPP TS 32.422: "Telecommunication management; Subscriber and equipment trace; Trace control and configuration management".

[39] 3GPP TS 32.423: "Telecommunication management; Subscriber and equipment trace; Trace data definition and management".

[40] IETF RFC 6455: "The WebSocket Protocol".

[41] IETF RFC 793: "Transmission Control Protocol".

[42] 3GPP TS 28.550: "Management and orchestration; Performance assurance".

[43] Void

[44] 3GPP TS 28.623: "Telecommunication management; Generic Network Resource Model (NRM) Integration Reference Point (IRP); Solution Set (SS) definitions".

[45] Text Attribution: Creator: ONAP, under Creative Commons Attribution 4.0 International License, https://creativecommons.org/licenses/by/4.0/, URI to access the text: <https://github.com/onap/vnfrqts-requirements/blob/05f26fac2b941513a7d0e856b99fd8c61d688299/docs/Chapter8/ves7_1spec.rst#resource-structure>.

[46] Void

[47] 3GPP TS 32.404: "Performance Management (PM); Performance measurements; Definitions and template".

[48] Void

[49] IETF RFC 8040: "RESTCONF protocol".

[50] IETF RFC 7951: " JSON Encoding of Data Modeled with YANG".

[51] IETF RFC 6243: "With-defaults Capability for NETCONF".

[52] IETF RFC 3339: " Date and Time on the Internet: Timestamps".

[53] 3GPP SA5 FORGE OpenAPI definitions: <https://forge.3gpp.org/rep/sa5>

[54] 3GPP TS 28.111: "Management and orchestration; Study on Network Slice Management Enhancement".

[55] 3GPP TS 33.210: "Network Domain Security (NDS); IP network layer security"

# 3 Definitions and abbreviations

## 3.1 Definitions

For the purposes of the present document, the terms and definitions given in TR 21.905 [1] apply.

## 3.2 Abbreviations

For the purposes of the present document, the abbreviations given in TR 21.905 [1] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in TR 21.905 [1].

MnS Management Service

# 4 Overview

The generic management services concept follows the management service concepts as defined in TS 28.533 [13].

# 5 Void

# 6 Void

# 7 Void

# 8 Void

# 9 Void

# 10 Void

# 11 Management services – Stage 2

## 11.1 Generic provisioning management service

### 11.1.0 Introduction

This clause provides the stage 2 definitions of Create, Read, Update and Delete (CRUD) operations for managing managed objects. According to clause 4.2.2 of TS 28.533 [13], these CRUD operations are the MnS component type A. The operations specified in this clause in combination with a NRM (MnS component type B) constitute a MnS, as defined in clause 4.3 of TS 28.533 [13] providing generic provisioning services for supported NRM (MnS component type B) of all MnS.

In addition, notifications to report changes related to managed objects and their attributes are specified.

### 11.1.1 Operations and notifications

#### 11.1.1.1 createMOI operation

##### 11.1.1.1.1 Description

This operation is invoked by MnS consumers to request a MnS producer to create a (single) managed object instance on the MnS producer.

The "managedObjectClass" parameter in the request specifies the class name and the "managedObjectInstance" parameter the instance name of the object to be created. Both parameters shall be included in the request.

The MnS consumer shall generate the instance name by first assigning a value to the naming attribute of the new instance, and then constructing a DN according to TS 32.300 [25].

The MnS consumer shall provide in "attributeListIn" none, some or all values for the attributes specified by the managed object class definition of the class to be created. The MnS producer shall not update attribute values or remove attribute values, that are provided in the request, before creating the object and returning the "createMOI" response.

The properties of an attribute determine if attribute values shall, shall not or may be provided in the "createMOI" request. If no value is provided and a default value is specified for the attribute, the MnS producer shall set the attribute value to the default value. For further information on attribute properties and their impact on the presence or absence of attribute values in object creation requests and MnS producer behaviour, see TS 32.156 [x], clause 5.2.1 and annex B.

When the MnS producer assigns values, that are not known to the MnS consumer, to one or more attributes for which no value is included in the "createMOI" request, the MnS producer shall include "attributeListOut" in the "createMOI" response, otherwise "attributeListOut" may be omitted.

In case of a successful operation, the object shall be created immediately upon reception of the "createMOI" request, and the "createMOI" response shall be returned immediately after the creation of the object. The MnS producer shall not wait with the creation of the object or returning the response until some other potentially long-lasting process or activity, that might be triggered by the reception of the request or the creation of the object, has completed.

Only objects, whose parent exists, can be created (directly under that parent). The MnS producer shall consider an attempt to create an object whose parent object does not exist as an error.

The model state after applying the "createMOI" request shall fulfill all model constraints such as cardinality, multiplicity, allowed values, or data types, otherwise the operation shall fail.

Note that stage 3 protocols may represent missing values for attributes, that are defined by the object class, in the "createMOI" request in different ways. For some protocols just the attribute name may be present, without an attribute value. For other protocols, the complete attribute name/value pair may be absent.

Some stage 3 protocols do not support returning "attributeListOut". In this case, the MnS producer shall not modify the attribute list provided in the request before creating the object. If required by TS 32.156 [x], clause 5.2.1 and annex B, the MnS producer shall assign default values to attributes only after returning the "createMOI" response. Attribute value change notifications may be used to notify MnS consumers about the changes. Only default values, that have a specific definitive value may be assigned upon object creation. This is because the MnS consumer knows the MnS producer will assign this value when required according to TS 32.156 [x], clause 5.2.1 and annex B. Default values that are determined by the MnS producer based on standardized or proprietary selection methods are typically not known to MnS consumers.

##### 11.1.1.1.2 Input parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter Name | S | Information Type / Legal Values | Comment |
| managedObjectClass | M | string | Class name of the managed object to be created. |
| managedObjectInstance | M | DN | Distinguished Name of the managed object to be created. |
| attributeListIn | M | LIST OF SEQUENCE< attribute name, attribute value> | List of attribute name/value pairs of the managed object to be created. |

##### 11.1.1.1.3 Output parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | S | Matching Information / Legal Values | Comment |
| attributeListOut | O | LIST OF SEQUENCE< attribute name, attribute value> | List of attribute name/value pairs of the created object.  The parameter shall be present if the MnS producer assigns values, that are not known to the MnS consumer, to one or more attributes, otherwise it may be absent. |
| status | M | ENUM (OperationSucceeded, OperationFailed) |  |

##### 11.1.1.1.4 Results

In case of success, the ManagedEntity instance has been created with the supplied DN. In case of failure, indication of the failure is provided in the Output parameters.

#### 11.1.1.2 getMOIAttributes operation

##### 11.1.1.2.1 Definition

This operation is invoked by MnS consumer to request the retrieval of management information (Managed Object attribute names and values) from the MIB maintained by MnS producer. One or several Managed Objects may be retrieved - based on the containment hierarchy.

The operation allows a MnS consumer to specify the data nodes to be returned with two optional methods. The first method allows to select objects with the "scope" and "filter" parameters. The second method uses the "dataNodeSelector" to specify the data nodes to be returned. All kinds of data nodes (i.e. objects, attributes, attribute fields and attribute elements) can be selected. The selection may be based on conditions. The value of "dataNodeSelector" is an expression constructed based on a SS specific grammar.

Note that the functionality of the "scope" and "filter" parameters is fully covered by the functionality of the "dataNodeSelector" parameter. Therefore, a MnS producer supporting the "dataNodeSelector" parameter is not expected to support the "filter" parameter.

A SS may choose to split this operation in several operations (e.g. operations to get "handlers" or "iterators" to Managed Objects fulfilling the scope/filter criteria and other operations to retrieve attribute names/values from these "handlers").

##### 11.1.1.2.2 Input Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Name | S | Information Type | Comment |
| baseObjectInstance | M | DN | This parameter specifies the base object instance.  If the "scope" parameter is absent, then either only the base object or the complete subtree below and including the base object shall be selected. The default behaviour is protocol specific. |
| CHOICE 1.1 scope | O | n/a | This parameter specifies the scope. It is a structured parameter and consists of the sub-parameters "scopeType" and "scopeLevel". The scope describes which object instances are selected with respect to a base object instance. The base object instance needs to be specified using a dedicated attribute. |
| CHOICE 1.2 > scopeType | O | ENUM {  BASE\_ONLY,  BASE\_ALL  } | If the optional "scopeLevel" parameter is not supported or absent, allowed values of "scopeType" are "BASE\_ONLY" and "BASE\_ALL".  The value "BASE\_ONLY" indicates only the base object is selected.  The value "BASE\_ALL" indicates the base object and all of its subordinate objects (incl. the leaf objects) are selected.  This parameter is redundant and can be omitted when confirming only the protocol specific default behaviour. |
|  |  | ENUM {  BASE\_NTH\_LEVEL,  BASE\_SUBTREE  } | If the "scopeLevel" parameter is supported and present, allowed values of "scopeType" are "BASE\_NTH\_LEVEL" and "BASE\_SUBTREE".  The value "BASE\_NTH\_LEVEL" indicates all objects on the level, which is specified by the "scopeLevel" parameter, below the base object are selected. The base object is at "scopeLevel" zero.  The value "BASE\_SUBTREE" indicates the base object and all of its subordinate objects down to and including the objects on the level, which is specified by the "scopeLevel" parameter, are selected. The base object is at "scopeLevel" zero. |
| CHOICE\_1.3 > scopeLevel | O | Integer | See definition of "scopeType" parameter. |
| CHOICE\_1.4 filter | O | See Comment. | This parameter defines filter criteria to be applied to the objects selected by the "baseObjectInstance", "scope" and "scopeLevel" parameters.  The actual syntax and capabilities of the filter is SS specific. However, each SS should support a filter consisting of one or several assertions that may be grouped using the logical operators AND, OR and NOT.  Each assertion is a logical expression of attribute existence, attribute value comparison ("equal to X, less than Y" etc.) and MO Class. |
| CHOICE\_1.5 attributeListIn | O | LIST OF attribute name. | This parameter identifies the attributes to be returned by this operation. If the parameter is absent or empty all attributes shall be returned. |
| CHOICE\_2.1 dataNodeSelector | O | string | This parameter contains an expression allowing to conditionally select data nodes. The expression semantic and syntax is SS specific. |

##### 11.1.1.2.3 Output Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Name | S | Matching Information | Comment |
| managedObjectClass | M | ManagedEntity class | For each returned MO: The class of the MO. |
| managedObjectInstance | M | ManagedEntity DN | For each returned MO: The name of the MO. This is a full DN according to TS 32.300 [5]. |
| attributeListOut | M | LIST OF SEQUENCE< attribute name, attribute value > | For each returned MO: A list of name/value pairs for MO. |
| status | M | ENUM (OperationSucceeded, OperationFailed) | An operation may fail because of a specified or unspecified reason. |

##### 11.1.1.2.4 Results

In case of success, all of the ManagedEntity instances selected for retrieval are returned. In case of failure, a specified or unspecified reason may be provided in the Output parameters.

#### 11.1.1.3 modifyMOIAttributes operation

##### 11.1.1.3.1 Description

This operation is invoked by MnS consumers to request a MnS producer to modify one or more attributes of one or more managed objects on that MnS producer. More specifically, this operation allows to modify (replace, add, remove) complete attributes, attribute fields and attribute elements.

The selection of the objects to be modified is achieved with the parameters "baseObjectInstance", "scopeType", "scopeLevel" and "filter". If no "scopeType" is specified, the value defaults to "BASE\_ONLY". Also, when no scoping is supported by the MnS producer, the value in "baseObjectInstance" identifies the object to be modified.

A specific protocol solution may choose to split the selection of objects with scoping and filtering and the modification of the attributes of the selected objects into different operations.

The modifications to be applied to the selected objects are described in the "modificationList" parameter. This is a multi-valued parameter. Each value is a structure composed of "modifyOperator", "nodeIdentifier" and "nodeValue". The values of "modificationList" are ordered and shall be applied in the sequence as they occur in the list.

The parameter "nodeIdentifier" is used to identify the attribute, attribute field or attribute element to which the modification shall be applied. Attributes within one managed object can be identified with their name only. Unambiguous identification of attribute fields is not possible with their names only, because multiple attribute fields within one object may have the same name. Therefore, the identification needs to be based e.g. on global or local identifiers, or on the specification of a path allowing to navigate to the attribute field. Details are protocol specific.

For the modification of multi-valued attributes or multi-valued attribute fields two cases are distinguished:

- All values (elements) are considered as a single value. Elements cannot be addressed individually. If replaced, all existing elements are replaced with the new elements received in the modification request. If deleted, all existing elements are deleted. Addition of new elements to existing elements is not possible.

- Each element can be addressed individually. Single elements can be added, replaced, and deleted.

The first case does not need any further considerations. The second case requires the identification of attribute elements and attribute field elements in the modification request, when replacing and deleting elements in ordered and unordered lists, and when adding elements to ordered lists. Details are not defined at stage 2. They are protocol specific and include identification by the element value, identification by a positional index or identification by an auxiliary key added at stage 3. Note that the concept of element keys is not defined at stage 2. Identification of elements may be provided also by the "nodeIdentifier" parameter.

The "modifyOperator" parameter specifies the modification to be applied to the value of the attribute or attribute field, or the attribute element or attribute field element identified by the "nodeIdentifier". The parameter can have the values "replace", "add", "remove" or "setToDefault":

- For "replace", the "nodeValue" specifies the attribute value, attribute field value, attribute element or attribute field element that shall replace the existing value.

- For "add", the "nodeValue" specifies the attribute value or attribute field value to be added to an attribute or attribute field without value, or the new attribute element or attribute field element to be added to a multi-valued attribute.

- For "remove", the "nodeValue" is absent when an attribute value or attribute field value is removed. When an attribute element or attribute field element is removed, "nodeValue" may carry the element to be removed, depending on how on protocol level attribute elements and attribute field elements are identified.

- For "setToDefault", the "nodeValue" is absent.

Attributes and attribute fields without value can be represented in different protocol specific ways, for example by an attribute name without attribute value, by an absent attribute name/value pair, or by a specific attribute value (such as "null" or "nil").

The "modifyMOIAttributes" operation allows to modify one or more attributes in one or more objects. When not all attribute modifications can be applied successfully, the MnS producer has different options how to proceed. He may not perform any of the modifications and roll back to the state at the reception of the modification request. He may apply the changes that can be applied, so that some of the requested modifications are applied and some are not applied. He may stop processing the modification request when the first error occurs. The stage 2 definition of this operation does not include any provisions on how to proceed in case an error occurs. These provisions are left to stage 3.

The model state after applying the " modifyMOIAttributes " request shall fulfill all model constraints such as cardinality, multiplicity, allowed values, or data types, otherwise the operation shall fail.

##### 11.1.1.3.2 Input parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter Name | S | Matching Information / Legal Values | Comment |
| baseObjectInstance | M | ManagedEntity.objectInstance | Base object used for scoping the target objects of the operation. If no scoping is applied, the base object is the only target object. |
| scopeType | O | See corresponding parameter in "getMOIAttributes". | See corresponding parameter in "getMOIAttributes". |
| scopeLevel | O | See corresponding parameter in "getMOIAttributes". | See corresponding parameter in "getMOIAttributes". |
| filter | O | See corresponding parameter in "getMOIAttributes". | See corresponding parameter in "getMOIAttributes". |
| modificationList | M | LIST OF SEQUENCE <  nodeIdentifier  modifyOperator,  nodeValue  > | Set of sub-operations to be applied to attributes and attribute fields of the target objects.  The "nodeIdentifier" specifies the target attribute or target attribute field of the sub-operation.  The "modifyOperator" specifies the operation to be applied to the target attribute or target attribute field . The parameter can have the values "replace", "add", "remove" or "setToDefault".  The "nodeValue" specifies the value used by the sub-operation. This parameter is absent for "remove" operations. |

##### 11.1.1.3.3 Output parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | S | Matching Information / Legal Values | Comment |
| modificationsOut | O | LIST OF SEQUENCE <  objectInstance DN,  objectClass string,  LIST OF SEQUENCE<  attribute name,  attribute value >  > | Provides for each object, that is selected by the request, the object name, the object class, and a list of name/value pairs with the values of *all* attributes after modification.  If all requested modifications are applied, the parameter may be absent.  If no requested modification is applied and an error response is returned, the parameter may be absent, too.. |
| status | M | ENUM (  SUCCEEDED,  PARTIALLY\_FAILED,  FAILED  ) | Indicates if all, some or none of the requested modifications were applied. Details on the error, such as which modification could not be applied and the corresponding reason, may be returned as well. |

##### 11.1.1.3.4 Results

In case of success, all of the ManagedEntity instances selected for modification are modified. In case of failure, a specified or unspecified reason may be provided in the Output parameters.

#### 11.1.1.4 deleteMOI operation

##### 11.1.1.4.1 Description

This operation is invoked by MnS consumer to request the deletion of one or more Managed Object instances in the MIB maintained by the MnS producer.

##### 11.1.1.4.2 Input parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter Name | S | Information Type / Legal Values | Comment |
| baseObjectInstance | M | DN | The MO instance that is to be used as the starting point for the selection of managed objects to which the filter (when supplied) is to be applied. This is a full DN according to TS 32.300 [5]. |
| scopeType | O | See corresponding parameter in getMOIAttributes. | See corresponding parameter in getMOIAttributes. |
| scopeLevel | O | See corresponding parameter in getMOIAttributes. | See corresponding parameter in getMOIAttributes. |
| filter | O | See comment. | See corresponding parameter in getMOIAttributes. |

##### 11.1.1.4.3 Output parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | S | Matching Information / Legal Values | Comment |
| deletionList | M | LIST OF SEQUENCE< ManagedEntity DN, ManagedEntity class name> | If the base object alone is specified, then this parameter is optional; otherwise it contains a list of managedObjectInstance/managedObjectClass pairs identifying the managed objects deleted. |
| status | M | ENUM (OperationSucceeded, OperationFailed, OperationPartiallySucceeded) | An operation may fail because of a specified or unspecified reason. The operation is partially successful if some, but not all, objects selected to be deleted are actually deleted. |

In lieu of a synchronization parameter, best effort synchronization will apply; that is, all managed objects selected for this operation will perform the operation if possible regardless of whether some managed objects fail to perform it.

##### 11.1.1.4.4 Results

In case of success, all of the ManagedEntity instances selected for deletion are deleted. In case of failure, a specified or unspecified reason may be provided in the Output parameters.

#### 11.1.1.4a changeMOIs operation

11.1.1.4a.1 Definition

This operation is invoked by MnS consumers to request a MnS producer to create, delete, and update one or more objects using a single request. The request contains an ordered set of sub-operations. Each sub-operation creates an object, deletes an object, or updates attribute or attribute field values. Sub-operations should be executed in the order they appear in the request.

The "baseObjectInstance" parameter is common for all sub-operations and identifies the root of the object tree where changes can be made. Each sub-operation is defined by the "path", "modifyOperator" and "nodeValue" parameters. The "path" parameter specifies the offset from the root object to the target object, the target attribute or the target attribute field of the sub-operation. The "modifyOperator" specifies the operation to be applied. Valid values are "replace", "add", remove, and for attributes and attributes fields also the value "setToDefault".

The "nodeValue" provides the value for the sub-operation. The parameter shall be absent for "remove" operations.

For operations on attribute values or attribute field values the same provisions as in clause 11.1.1.3 apply.

When adding (creating) objects, the "nodeValue" contains the object representation.

The model state after applying the "changeMOIs" request shall fulfill all model constraints such as cardinality, multiplicity, allowed values, or data types, otherwise the operation shall fail.

Note that the parameters introduced and used in this clause just serve the purpose of explaining the functionality. Specific stage 3 solutions may implement the functionality in very different ways.

##### 11.1.1.4a.2 Input parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter Name | S | Matching Information / Legal Values | Comment |
| baseObjectInstance | M | ManagedEntity.objectInstance | Identifies the base object, that together with the "path" identifies the nodes to be modified. |
| modificationsIn | M | LIST OF SEQUENCE <  path,  modifyOperator,  nodeValue  > | Set of sub-operations to be applied to the target node.  The "nodeIdentifier" specifies the target node.  The "modifyOperator" specifies the operation to be applied to the target attribute node. The parameter can have the values "replace", "add", "remove" or "setToDefault". The value "replace" is not applicable, when the target node is an object. The value "SetToDefault" is applicable only to attributes and attribute fields.  The "nodeValue" specifies the value for the sub-operation. This parameter is absent for "remove" operations. |

##### 11.1.1.4a.3 Output parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter name | S | Matching Information / Legal Values | Comment |
| modificationsOut | O | LIST OF SEQUENCE <  objectInstance DN,  objectClass string,  LIST OF SEQUENCE<  attribute name,  attribute value >  > | Provides for each object, that is modified, the object name, the object class, and a list of name/value pairs with the values of *all* attributes after modification.  If all requested modifications are applied, the parameter may be absent.  If no requested modification is applied and an error response is returned, the parameter may be absent, too. |
| status | M | ENUM (  SUCCEEDED,  PARTIALLY\_FAILED,  FAILED  ) | Indicates if all, some or none of the requested modifications were applied. Details on the error, such as which modification could not be applied and the corresponding reason, may be returned as well. |

#### 11.1.1.5 Void

#### 11.1.1.6 Void

#### 11.1.1.7 Notification notifyMOICreation

##### 11.1.1.7.1 Definition

This notification notifies the subscribed consumers that a new Managed Object Instance has been created.

##### 11.1.1.7.2 Input parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter Name | S | Information Type / Legal Values | Comment |
| objectClass | M | ManagedEntity.objectClass | It specifies the class name of the IOC. A network event has occurred in an instance of this class. |
| objectInstance | M | ManagedEntity.objectInstance | It specifies a new instance of the above IOC in which the network event related to by carrying the Distinguished Name (DN) for the instance. |
| notificationId | M | This is an identifier for the notification, which may be used to correlate notifications. | The identifier of the notification shall be chosen to be unique across all notifications of a particular managed object instance throughout the time that correlation is significant, it uniquely identifies the notification from other notifications generated by the subject MOI. |
| notificationType | M | It specifies the type of provisioning management services related notifications. The value “notifyMOICreation” shall be carried. | It specifies the type of notification. |
| eventTime | M | It indicates the MOICreation event time. | See RFC 3339 [52] section 5.6 for details. |
| systemDN | M | It shall carry the DN of management service providers. | - |
| correlatedNotifications | CM | It specifies a set of notifications that are correlated to the subject notification. | The condition is that the MnS producer support the correlation of notifications |
| additionalText | O | It can contain further information in text on the event of the ManagedEntity(s). | - |
| sourceIndicator | O | ENUM(  Resource\_operation,  Management\_operation,  SON\_operation,Unknown) | This parameter, when present, indicates the source of the operation that led to the generation of this notification. It can have one of the following values:  1. resource operation: The notification was generated in response to an internal operation of the resource;  2. management operation: The notification was generated in response to a management operation applied across the managed object boundary external to the managed object;  3. SON operation: The notification was generated as result of a SON (Self Organising Network) process like self-configuration, self-optimization, self-healing etc. .  4. unknown: It is not possible to determine the source of the operation.  Remark: A provisioning MnS provider may not in any case be aware that SON operation lead to the generation of this generation. In this case another value than SON\_operation for sourceIndicator might be sent. |
| attributeList | O | LIST OF SEQUENCE <AttributeName, AttributeValue> | The attributes (name/value pairs) of the created MOI. |

##### 11.1.1.7.3 Triggering event

###### 11.1.1.7.3.1 From-state

stateBeforeObjectCreation.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| stateBeforeObjectCreation | The number of instances of the IOC ManagedEntity is equal to N. |

###### 11.1.1.7.3.2 To-state

stateAfterObjectCreation.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| stateAfterObjectCreation | The number of instances of the IOC ManagedEntity is equal to N + 1. |

#### 11.1.1.8 Notification notifyMOIDeletion

##### 11.1.1.8.1 Definition

This notification notifies the subscribed consumers that an existing Managed Object Instance has been deleted.

##### 11.1.1.8.2 Input parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter Name | S | Information Type / Legal Values | Comment |
| objectClass | M | ManagedEntity.objectClass | It specifies the class name of the IOC. A network event has occurred in an instance of this class. |
| objectInstance | M | ManagedEntity.objectInstance | It specifies an existing instance of the above IOC in which the network event related to by carrying the Distinguished Name (DN) for the instance. |
| notificationId | M | This is an identifier for the notification, which may be used to correlate notifications. | The identifier of the notification shall be chosen to be unique across all notifications of a particular managed object throughout the time that correlation is significant, it uniquely identifies the notification from other notifications generated by the subject MOI. |
| notificationType | M | It specifies the type of provisioning management services related notifications. The value “notifyMOIDeletion” shall be carried. | It specifies the type of notification. |
| eventTime | M | It indicates the MOIDeletion event time. | See RFC 3339 [52] section 5.6 for details. |
| systemDN | M | It shall carry the DN of management service providers. | - |
| correlatedNotifications | CM | It specifies a set of notifications that are correlated to the subject notification. | The condition is that the MnS producer support the correlation of notifications |
| additionalText | O | It can contain further information in text on the event of the ManagedEntity(s). | - |
| sourceIndicator | O | ENUM(  Resource\_operation,  Management\_operation,  SON\_operation,Unknown) | This parameter, when present, indicates the source of the operation that led to the generation of this notification. It can have one of the following values:  1. resource operation: The notification was generated in response to an internal operation of the resource;  2. management operation: The notification was generated in response to a management operation applied across the managed object boundary external to the managed object;  3. SON operation: The notification was generated as result of a SON (Self Organising Network) process like self-configuration, self-optimization, self-healing etc. .  4. unknown: It is not possible to determine the source of the operation.  Remark: A provisioning MnS provider may not in any case be aware that SON operation lead to the generation of this generation. In this case another value than SON\_operation for sourceIndicator might be sent. |
| attributeList | O | LIST OF SEQUENCE <AttributeName, AttributeValue> | The attributes (name/value pairs) of the deleted MOI. |

##### 11.1.1.8.3 Triggering event

###### 11.1.1.8.3.1 From-state

stateBeforeObjectDeletion.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| stateBeforeObjectDeletion | The number of instances of the IOC ManagedEntity is equal to N. |

###### 11.1.1.8.3.2 To-state

stateAfterObjectDeletion.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| stateAfterObjectDeletion | The number of instances of the IOC ManagedEntity is equal to N - 1. |

#### 11.1.1.9 Notification notifyMOIAttributeValueChanges

##### 11.1.1.9.1 Definition

This notification notifies the subscribed MnS consumers that changes of one or several attributes of a Managed Object Instance in the NRM.

##### 11.1.1.9.2 Input parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter Name | S | Information Type / Legal Values | Comment |
| objectClass | M | ManagedEntity.objectClass | It specifies the class name of the IOC. A network event has occurred in an instance of this class. |
| objectInstance | M | ManagedEntity.objectInstance | It specifies the existing instance of the above IOC in which the network event related to by carrying the Distinguished Name (DN) for the instance. |
| notificationId | M | This is an identifier for the notification, which may be used to correlate notifications. | The identifier of the notification shall be chosen to be unique across all notifications of a particular managed object throughout the time that correlation is significant, it uniquely identifies the notification from other notifications generated by the subject Information Object. |
| notificationType | M | It specifies the type of provisioning management services related notifications. The value “notifyMOIAttributeValueChange” shall be carried. | It specifies the type of notification. |
| eventTime | M | It indicates the MOIAttributeValueChange event time. | See RFC 3339 [52] section 5.6 for details. |
| systemDN | M | It shall carry the DN of management service providers. | - |
| correlatedNotifications | CM | It specifies a set of notifications that are correlated to the subject notification. | The condition is that the MnS producer support the correlation of notifications |
| additionalText | O | It can contain further information in text on the event of the ManagedEntity(s). | - |
| sourceIndicator | O | ENUM(  Resource\_operation,  Management\_operation,  SON\_operation,Unknown) | This parameter, when present, indicates the source of the operation that led to the generation of this notification. It can have one of the following values:  1. resource operation: The notification was generated in response to an internal operation of the resource;  2. management operation: The notification was generated in response to a management operation applied across the managed object boundary external to the managed object;  3. SON operation: The notification was generated as result of a SON (Self Organising Network) process like self-configuration, self-optimization, self-healing etc. .  4. unknown: It is not possible to determine the source of the operation.  Remark: A provisioning MnS provider may not in any case be aware that SON operation lead to the generation of this generation. In this case another value than SON\_operation for sourceIndicator might be sent. |
| attributeValueChange | M | LIST OF SEQUENCE <AttributeName, NewAttributeValue,  CHOICE [NULL, OldAttributeValue]> | The changed attributes (name/value pairs) of the MOI (with both new and, optionally, old values). |

##### 11.1.1.9.3 Triggering event

###### 11.1.1.9.3.1 From-state

stateBeforeAttributeValueChange.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| stateBeforeAttributeValueChange | The subject attribute has a value at time T1. |

###### 11.1.1.9.3.2 To-state

stateAfterAttributeValueChange.

|  |  |  |
| --- | --- | --- |
| Assertion Name | Definition | |
| stateAfterAttributeValueChange | | The subject attribute has been changed to a value other than the value at time T1. |

#### 11.1.1.10 Notification notifyEvent

##### 11.1.1.10.1 Definition

This notification notifies the MnS consumer, who has a subscription receiving this type of notification, that certain network events has occurred with potential service impact, for example, system restart and system redundancy shift (backup).

This notification definition is generic in the sense that the specific types of network event are not defined.

##### 11.1.1.10.2 Input parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter Name | S | Information Type / Legal Values | Comment |
| objectClass | M | ManagedEntity.objectClass | -- |
| objectInstance | M | ManagedEntity.objectInstance | -- |
| notificationId | M | It carries the identifier for the subject notification. | See Note 1. |
| eventTime | M | It indicates the time of the event. | See RFC 3339 [52] section 5.6 for details. |
| systemDN | M | It carries the DN of producer of the notification. | -- |
| notificationType | M | "notifyEvent" | -- |
| specificProblem | M | It indicates a problem detected. | -- |
| additionalText | O | It carries additional information. | -- |
| additionalInformation | O | It carries additional information. | -- |
| NOTE 1: If consumer receives notifications from one producer, consumer can use the notificationId and the objectInstance to uniquely identify all received notifications.  If consumer receives notifications from multiple producers and notifications of each objectInstance are reported to at most by one producer, consumer can use the notificationId and objectInstance to uniquely identify all received notifications.  If consumer receives notifications from multiple producers and notifications of one or more objectInstance(s) are reported byo two or more producers, consumer can use the notificationId together with objectInstance and the identity of producer (systemDN), to uniquely identify all received notifications. If the information systemDN is absent, consumer needs other means, which are outside the scope of this TS, to determine the identity of producer.  How notificationId of notifications are re-used to correlate notifications is outside of the scope of this specification. | | | |

#### 11.1.1.11 Notification notifyMOIChanges

##### 11.1.1.11.1 Definition

This notification reports NRM updates to subscribed MnS consumers. It can report multiple NRM updates that happen at the same time. All possible NRM updates can be reported:

- Creation and deletion of an object.

- Creation and deletion of an attribute, attribute field, attribute element and attribute field element.

- Replacement of an attribute value, attribute field value, attribute element and attribute field element.

The MnS producer decides whether to send notifications of type notifyMOICreation, notifyMOIDeletion or notifyMOIAttributesValueChange, or a single notifyMOIChanges reporting all changes in a single notification. The MnS producer should take subscription information into account when deciding the notification types to be sent, and not try to send notifications that the MnS consumer did not subscribe to.

The notification header includes a notificationId. This identifier shall not be used in the parameter correlatedNotifications potentially carried in other notifications. The notificationId in mOIChanges shall be used instead. This is because the latter notification id is associated to a single MOI only, whereas the former notification id can be associated to changes of multiple MOIs. The correlatedNotifications associates to a single MOI one or more notification ids identifying notifications reporting events for that MOI.

The scope of the subscription for this notification may specify managed objects, attributes, attribute fields or attribute elements. This allows for example to create subscriptions for notifyMOIChanges notifications that report attribute value changes of one attribute only.

##### 11.1.1.11.2 Input parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter Name | S | Information Type / Legal Values | Comment |
| objectClass | M | See clause 11.1.1.7.2 | Identifies the classe name of a common ancestor object of the objects for which changes are reported. A MnS producer may set this parameter always to the class name of the parent of the local root object in the MIB. |
| objectInstance | M | See clause 11.1.1.7.2 | Identifies the instance of a common ancestor object of the objects for which changes are reported. A MnS producer may set this parameter always to the instance of the parent of the local root object in the MIB. |
| notificationId | M | See clause 11.1.1.7.2 | See clause 11.1.1.7.2 |
| notificationType | M | const string “notifyMOIChanges” | See clause 11.1.1.7.2 |
| eventTime | M | See clause 11.1.1.7.2 | See clause 11.1.1.7.2 |
| systemDN | M | See clause 11.1.1.7.2 | See clause 11.1.1.7.2 |
| moiChanges | M | SEQUENCE OF SET {  notificationId (M),  correlatedNotifications (O),  additionalText (O),  sourceIndicator (O),  op (M),  path (M),  value (M) ,  oldValue (O)  } | This parameter describes the reported NRM updates. It is a list of items; each item reports a single NRM update. The "notificationId" identifies an item.  The NRM update itself is described by the parameters "op", "path", "value" and "oldValue". The parameters "correlatedNotifications", "additionalText" and "sourceIndicator " provide context information.  The parameter "op" specifies the type of operation reporting the NRM update. Valid values are "add", "remove" and "replace". The operation describes what has conceptually happened to the NRM on the MnS producer. The operation applied to the NRM by the MnS producer and causing the reported NRM update can be different.  "add" shall be used for reporting the creation of an object, attribute, attribute field or multi-value attribute element.  "remove" shall be used for reporting the deletion of an object, attribute, attribute field or multi-value attribute element.  "replace" shall be used for reporting the replacement of an existing attribute value, attribute field value or multi-value attribute element.  The "path" and "objectInstance" identify the object, attribute, attribute field or multi-value attribute element, that was created, deleted or replaced.  If an object creation is reported with "add", the "value" shall carry a complete representation of the created object. If an object deletion is reported with "remove", the "value" shall be absent. It may optionally carry a complete representation of the deleted object.  If an attribute, attribute field or multi-value attribute element creation is reported with "add", the "value" shall carry the value of the created attribute, attribute field or multi-value attribute element.  If an attribute, attribute field or multi-value attribute element deletion is reported with "remove", the "value" shall be absent. It may optionally carry the old value of the deleted attribute, attribute field or multi-value attribute element.  If the replacement of an attribute, attribute field or multi-value attribute element value is reported with "replace", the "value" shall carry the new value of the attribute, attribute field or multi-value attribute element. The "oldValue" may optionally carry the old value of the attribute, attribute field or multi-value attribute element before the replacement.  If multiple objects are created, the creation of parent objects shall be reported before the creation of the child objects. Vice versa, when the deletion of multiple objects is reported, the deletion of child objects shall be reported before the deletion of the parent objects. |

### 11.1.2 Managed Information

#### 11.1.2.1 ManagedEntity << ProxyClass>>

##### 11.1.2.1.1 Definition

The ProxyClass ManagedEntity represents the role that can be played by an instance of an IOC defined in NRMs, e.g. Generic NRM, NR and NG-RAN NRM, or 5GC NRM. ManagedEntity is used in the specification of provisioning operations and notifications to represent an instance of an IOC defined in these NRMs.

## 11.2 Void

## 11.2a Generic fault supervision management service

See TS 28.111 [54] for more information on Generic fault supervision management service.

## 11.3 Performance assurance

### 11.3.1 Operations and notifications

#### 11.3.1.1 Void

#### 11.3.1.2 Void

#### 11.3.1.3 Notification notifyThresholdCrossing

##### 11.3.1.3.1 Definition

A MnS producer sends this notification to subscribed MnS consumers when a "ThresholdMonitor" (TS 28.622 [11]) on that MnS producer detects the threshold crossing of a monitored performance metric.

##### 11.3.1.3.2 Notification information

| **Parameter Name** | **S** | **Information Type** | **Comment** |
| --- | --- | --- | --- |
| objectClass | M | ManagedEntity.objectClass | Class of the managed object, where the threshold crossing occurred. |
| objectInstance | M | ManagedEntity.objectInstance | Instance of the managed object, where the threshold crossing occurred. |
| notificationId | M | -- |  |
| notificationType | M | "notifyThresholdCrossing" |  |
| eventTime | M | -- | Time when the threshold crossing occurred. |
| systemDN | M | MnSAgent.objectInstance |  |
| observedPerfMetricName | M | ThresholdMonitor.thresholdInfoList[11].\  performanceMetrics[y] | Name of the performance metric that has crossed the threshold. |
| observedPerfMetricValue | M | -- | Value of the performance metric, that has crossed the threshold, when the threshold crossing was observed |
| observedPerfMetricDirection | M | -- | Direction ("UP" or "DOWN") of the performance metric, when the threshold crossing was observed |
| thresholdValue | M | ThresholdMonitor.thresholdInfoList[11].\  thresholdvalue | Threshold value of the triggered threshold |
| hysteresis | O | ThresholdMonitor.thresholdInfoList[11].\  hysteresis | Hysteresis of the triggered threshold |
| monitorGranularityPeriod | M | ThresholdMonitor.monitorGranularityPeriod | Granularity period of the threshold monitor |
| additionalText | O | -- | Vendor specific information |

### 11.3.2 Managed information

#### 11.3.2.1 Performance data file

##### 11.3.2.1.1 Void

##### 11.3.2.1.2 Performance data file content description

Table 11.3.2.1.2-1 provides the content definition of a performance data file.

Table 11.3.2.1.2-1: Performance data file content description

| File content item | Description |
| --- | --- |
| measDataFile | Top-level tag indicating the file contains performance metrics. Each file includes a header ("measFileHeader"), a collection of information elements with produced performance metrics and associated meta data ("measData") and a footer ("measFileFooter"). |
| measFileHeader | File header including the file format version, information about the sending node (DN, type and vendor) and a time stamp indicating the begin of the first granularity period contained in the file ("collectionBeginTime"). |
| measData | Information element containing the DN of the common root of the measured object instances ("measObjRootDn ") included in that information element, followed by a list of information elements containing the produced performance metrics and associated meta data ("measInfo"). A "MeasDataFile" contains zero, one or more "measData" elements. |
| measFileFooter | File footer with a time stamp indicating the end of the last granularity period contained in the file ("collectionEndTime"). |
| fileFormatVersion | File format version applied by the sender as indicated by the specific format version identifier provided for each version. |
| senderName | DN of the entity, that generated and sent the file. The entity is either a managed element represented by a "ManagedElement" or a management node represented by a "ManagementNode" |
| senderType | Type of the entity, that generated and sent the file, as defined in TS 28.620 [y]. The type of a management node is "MANAGEMENT\_NODE". |
| vendorName | Vendor of the the entity, that generated and sent the file. |
| collectionBeginTime | Time stamp indicating the begin of the first granularity period for which performance metrics are stored in the file. |
| measObjRootDn | DN of the measured object root. The measured object root is the first common object name-containing all objects that the metrics in one "measData" element are related to. When the metrics are produced by a managed element, the root object is the "ManagedElement" representing this managed element. When (aggregated) metrics are produced by a management node (based on input metrics from managed elements), such as metrics for sub-networks or network slices, the root object is the root "SubNetwork" of this management node. |
| measObjRootUserLabel | User label of the measured object root. |
| measObjRootSwVersion | Software version of the measured object root, allowing post-processing systems to take care of vendor specific performance metrics. It is either the software version of a managed element or of a management node. |
| measInfo | Information element added to "measData" for each expired granularity period, containing information on the produced performance metrics, starting with a time stamp ("measTimeStamp"), the granularity period ("granularityPeriod") and reporting period ("reportingPeriod") that are associated to the following performance metrics ("measValues"), for which is indicated the performance metric name, the measured or computed performance metric value and the object instance to which the performance metric is related to. |
| measInfoId | Identifier of a "measInfo". |
| jobId | Job identifier of the related "PerfMetricJob" in this "measInfo". |
| reportingPeriod | Period used for performance metric reporting in this "measInfo". Unit is seconds |
| granularityPeriod | Period used for performance metric production in a "measInfo". Unit is seconds. |
| measTimeStamp | End time of the granularity period in a "measInfo". |
| measTypes | Performance metric names in a "measInfo" |
| measValues | Performance metric values in a "measInfo". Each item in this list includes the LDN of the object the metrics are related to ("measObjLdn"), the measured or computed values of the metrics ("measResults") and a flag that indicates whether the metrics are reliable ("suspectFlag"). |
| measObjLdn | Local distinguished name (LDN) of the object the performance metrics are related to (measured object) within the scope defined by the "measObjRootDn". The concatenation of the "measObjRootDn" and the "measObjLdn" is the DN of the measured object. The "measObjLdn" is therefore empty if the "measObjRootDn" already specifies completely the DN of the measured object, which is the case for metrics associated to "ManagedElement" or the root "SubNetwork".  For example, if the measured object is a "ManagedElement" representing RNC "RNC-Gbg-1", then the "measObjRootDn" may look like  "DC=a1.operatorNN.com,SubNetwork=CountryNN,ManagedElement=RNC-Gbg-1"  and the "measObjLdn" is empty. However, if the measured object is an "UtranCell" representing cell "Gbg-997" managed by that RNC, then the "measObjRootDn" is the same as above, i.e.  "DC=a1.companyNN.com,SubNetwork=CountryNN,ManagedElement=RNC-Gbg-1"  and the "measObjLdn" is  "RncFunction=RF-1,UtranCell=Gbg-997".  The class of the measured object is defined in item f) of measurement definitions (TS 32.404 [47], TS 28.552 [18]) and in item d) of KPI definitions (TS 28.554 [6]). |
| measResults | List of result values for the observed or computed performance metrics. The "measResults" sequence shall have the same number of elements and follow the same order as the "measTypes" sequence. The NULL value is reserved to indicate that the performance metric is not applicable or could not be produced for the object instance. |
| suspectFlag | Reliability of the performance metrics. FALSE means the metrics are reliable, TRUE means they are not reliable. The default value is "FALSE". |
| collectionEndTime | Time stamp indicating the end of the last granularity period for which performance metrics are stored in the file. |

The representation of all timestamps in PM files shall follow the representations allowed by the ISO 8601 [20].   
The precise format for timestamp representation shall be determined by the technology used for encoding the PM file (e.g. ASN.1, XML DTD, and XML Schema). The choice of technology should ensure that this representation is derived from ISO 8601 [20]. Based on the representation used, the timestamp shall refer to either UTC time or local time or local time with offset from UTC.

##### 11.3.2.1.3 Void

###### 11.3.2.1.3.1 Void

###### 11.3.2.1.3.2 Void

##### 11.3.2.1.4 Performance data file naming convention

This clause defines a rule that shall be applied for constructing names for files containing performance data.

<Type><Startdate>.<Starttime>-[<Enddate>.]<Endtime>[\_-<jobIdList>][\_<UniqueIdList>][\_-\_<RC>]

1) The "Type" field indicates if the file contains measurement results for single or multiple measured objects and/or granularity periods where:

- "A" means single measured object, single granularity period (this is used when granularity period is equal to reporting period);

- "B" indicates multiple measured objects, single granularity period (this is used when granularity period is equal to reporting period);

- "C" signifies single measured object, multiple granularity periods (this is used when reporting period is multiples of the granularity period and will contain multiple measurement reports);

- "D" stands for multiple measured objects, multiple granularity periods (this is used when reporting period is multiples of the granularity period and will contain multiple measurement reports).

2) The "Startdate" field indicates the date when the granularity period began if the "Type" field is set to A or B. If the "Type" field is either "C" or "D" then "Startdate" contains the date when the first granularity period of the measurement results contained in the file started. The "Startdate" field is of the form YYYYMMDD, where:

- YYYY is the year in four-digit notation;

- MM is the month in two digit notation (01 - 12);

- DD is the day in two-digit notation (01 - 31).

3) The "Starttime" field indicates the time when the granularity period began if the "Type" field is set to A or B. If the "Type" field is either "C" or "D" then "Starttime" contains the time when the first granularity period of the measurement results contained in the file began. The "Starttime" field is of the form HHMMshhmm, where:

- HH is the two-digit hour of the day (local time), based on 24-hour clock (00 - 23);

- MM is the two digit minute of the hour (local time), based on 60-minutes clock (00 - 59);

- s is the sign of the local time differential from UTC (+ or -), in case the time differential to UTC is 0 then the sign may be arbitrarily set to "+" or "-";

- hh is the two-digit number of hours of the local time differential from UTC (00-23);

- mm is the two digit number of minutes of the local time differential from UTC (00-59).

4) The "Enddate" field shall only be included if the "Type" field is set to "C" or "D", i.e. measurement results for multiple granularity periods are contained in the file. It identifies the date when the last granularity period of these measurements ended, and its structure corresponds to the "Startdate" field.

5) The "Endtime" field indicates the time when the granularity period ended if the "Type" field is set to A or B. If the "Type" field is either "C" or "D" then "Endtime" contains the time when the last granularity period of the measurement results contained in the file ended. Its structure corresponds to the "Starttime" field.

6) The "UniqueIdList" field indicates the DNs of the measured objects.

7) The "RC" field is a running count, starting with the value of "1", and shall be appended only if the filename is otherwise not unique, i.e. more than one file is generated and all other parameters of the file name are identical. Therefore it may only be used by the EM, since the described situation cannot occur with NE generated files. Note that the delimiter for this field, \_-\_, is an underscore character (\_), followed by a minus character (-), followed by an underscore character (\_).

8) The "jobIdList" indicates the measurement job id(s) that the performance data file is associated with.

Some examples describing file-naming convention:

1) file name: A20000626.2315+0200-2330+0200\_gNBId,   
meaning: file produced for gNB <gNBId> on June 26, 2000, granularity period 15 minutes from 23:15 local to 23:30 local, with a time differential of +2 hours against UTC.

2) file name: B20021224.1700-1130-1705-1130\_-job10\_S-NSSAI,  
meaning: file containing results for multiple measured objects, generated for measurement job job10, produced for NSI <S-NSSAI> on December 24, 2002, granularity period 5 minutes from 17:00 local to 17:05 local, with a time differential of –11:30 hours against UTC.

3) file name: D20050907.1030+0000-20050909.1500+0000\_SubnetworkId\_-\_2,  
meaning: file containing results subnetwork <SubnetworkId>, start of first granularity period 07 September 2005, 10:30 local, end of last granularity period 09 September 2005, 15:00 local, with a time differential of 0 against UTC. This is the second file for this subnetwork/granularity period combination.

4) file name: C20050907.1030+0000-20050909.1500+0000\_gNBId,  
meaning: file produced for the gNB <gNBId>, start of first granularity period 07 September 2005, 10:30 local, end of last granularity period 09 September 2005, 15:00 local, with a time differential of 0 against UTC.

#### 11.3.2.1.4 Void

## 11.4 Heartbeat notification

### 11.4.1 Operations and notifications

#### 11.4.1.1 Notification notifyHeartbeat

##### 11.4.1.1.1 Definition

This notification allows a MnS producer to send heartbeats to consumer(s) when the MnS producer heartbeat period has expired or when a MnS consumer requests the emission of an immediate heartbeat notification.

The emission of heartbeat notifications is controlled by the HeartbeatControl IOC (TS 28.622 [11]).

##### 11.4.1.1.2 Input parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter Name | S | Information Type / Legal Values | Comment |
| objectClass | M | HeartbeatControl.objectClass |  |
| objectInstance | M | HeartbeatControl.objectInstance | Instance controlling the emission of this notifyHeartbeat notification. |
| notificationId | M | -- |  |
| notificationType | M | "notifyHeartbeat" |  |
| eventTime | M | -- | Time at which the notification is emitted.  See RFC 3339 [52] section 5.6 for details. |
| systemDN | M | MnSAgent.objectInstance |  |
| heartbeatNtfPeriod | M | HeartbeatControl.heartbeatNtfPeriod |  |

##### 11.4.1.1.3 Triggering event

###### 11.4.1.1.3.1 From-state

stateBeforeHeartbeatNotification1 OR stateBeforeHeartbeatNotification2.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| stateBeforeHeartbeatNotification1 | The internal countdown timer of the MOI emitting the notifyHeartbeat notification has reached the value ‘0’ (zero). |
| stateBeforeHeartbeatNotification2 | The value of the attribute triggerHeartbeatNtf of the MOI emitting the notifyHeartbeat notification is TRUE. |

###### 11.4.1.1.3.2 To-state

stateAfterOHeartbeatNotification1 OR stateAfterOHeartbeatNotification2.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| stateAfterHeartbeatNotification1 | If From-state is stateBeforeHeartbeatNotification1 then:  the internal countdown timer of the MOI is reset to the value of its heartbeatNtfPeriod attribute. |
| stateAfterHeartbeatNotification2 | If From-state is stateBeforeHeartbeatNotification2 then:  the value of the internal countdown timer of the MOI is not affected. |

## 11.5 Streaming data reporting service

### 11.5.1 Operations and notifications

#### 11.5.1.1 establishStreamingConnection operation (M)

##### 11.5.1.1.1 Definition

This operation enables the MnS producer to establish a connection to the MnS consumer (i.e. streaming target). The connection establishement includes the exchange of meta-data (producer informs consumer about its own identity and the nature of the data to be reported via streaming) phase and the actual connection (a data pipe for streaming) establishment.

Established connection supports stream multiplexing (one connection supports one or more reporting streams simultaneously).

Upon successful connection establishment, the MnS consumer is aware of the MnS producer's identity, the list of reporting streams and the nature of data being reported on each of the streams.

The established connection may be kept "alive" either by built-in functionality of the solution set or by periodic reporting of empty stream data.

##### 11.5.1.1.2 Input parameters

| Parameter Name | S | Information type | Comment |
| --- | --- | --- | --- |
| producerId | M | The identity of the producer requesting the connection establishment. | DN of the MnS producer. If the MnS producer is not modeled as 3GPP NRM MOI, an alternative identifer other than DN may be used. |
| streamInfoList | M | List of StreamInfo | This parameter contains the list of meta-data about each reporting stream.  For streaming trace reporting each StreamInfo includes:  - StreamType carrying the value "TRACE";  - SerializationFormat carrying the value "GPB" or "ASN1";  - streamId globally unique stream identifier;  - Trace Reference (see clause 5.6 of TS 32.422 [38]) as stream identifier;  - list of Trace Reference (see clause 5.6 of TS 32.422 [38]) for signaling based trace  - list of tuple of <Trace Reference (see clause 5.6 of TS 32.422 [38]), jobId (see clause 4.3.30 of TS 28.622 [11]) providing the id of the job for the configuration> for management based trace  For streaming performance data reporting each StreamInfo includes:  - StreamType carrying the value "PERFORMANCE";  - SerializationFormat carrying the value "GPB" or "ASN1";  - streamId globally unique stream identifier;  - measObjDn: the DN of the measured object instance;  - performanceMetrics: a list of performance metric names whose values are to be reported by the Performance Data Stream Units (see Annex C of TS 28.550 [42]) via this stream. Performance metrics include measurement and KPI;  - either:  - jobId defined in the PerfMetricJob MOI (see clause 4.3.31 of TS 28.622 [11]) for which the data is being reported;  - or:  - jobId globally unique identifier of a measurement job (see TS 28.550 [42]).  For streaming analytics reporting each StreamInfo includes:  - StreamType carrying the value "ANALYTICS";  - SerializationFormat carrying the value "GPB" or "ASN1";  - streamId globally unique stream identifier;  - AnalyticsInfo providing the details about the analytics activity for which the data is being reported.  For proprietary data streaming reporting each StreamInfo includes:  - StreamType carrying the value "PROPRIETARY";  - streamId globally unique stream identifier;  - VsDataContainer (see clause 4.3.9 of TS 28.622 [11]) providing the details about the data being reported. |

##### 11.5.1.1.3 Output parameters

| Parameter Name | S | Matching Information | Comment |
| --- | --- | --- | --- |
| connectionId | M | Identifier of the established streaming connection. | It identifies the established streaming connection. The format may have dependency on the solution set. |
| status | M | ENUM (Success, Failure) | An operation may fail because of a specified or unspecified reason. |

##### 11.5.1.1.4 Exceptions

| Exception Name | Definition |
| --- | --- |
| unexpectedStreams | **Condition:** Some information in the list of streamInfo was unexpected by the MnS consumer.  **Returned Information:** Name of the exception; status is set to "Failure". |

#### 11.5.1.2 terminateStreamingConnection operation (M)

##### 11.5.1.2.1 Definition

This operation enables the MnS producer to terminate the connection to theMnS consumer (i.e. streaming target).

Upon successful termination of the streaming connection, the MnS producer stops reporting data to the MnS consumer on this connection.

##### 11.5.1.2.2 Input parameters

| Parameter Name | S | Information type | Comment |
| --- | --- | --- | --- |
| connectionId | M | See clause 11.5.1.1.3 | It identifies the streaming connection being terminated. The format may have dependency on the solution set. |

##### 11.5.1.2.3 Output parameters

| Parameter Name | S | Matching Information | Comment |
| --- | --- | --- | --- |
| status | M | ENUM (Success, Failure) | An operation may fail because of a specified or unspecified reason. |

##### 11.5.1.2.4 Exceptions

| Exception Name | Definition |
| --- | --- |
| unknownConnection | **Condition:** the connectionId is invalid.  **Returned Information:** Name of the exception; status is set to "Failure". |

#### 11.5.1.3 reportStreamData operation (M)

##### 11.5.1.3.1 Definition

This operation enables the MnS producer to send a unit of streaming data to the MnS consumer.

##### 11.5.1.3.2 Input parameters

| Parameter Name | S | Information type | Comment |
| --- | --- | --- | --- |
| connectionId | M | See clause 11.5.1.1.3 | It identifies the streaming connection on which the reported data are being sent. The format may have dependency on the solution set. |
| streamingData | M | Unit of streaming data | This parameter contains the actual data (payload) being reported via stream.  For streaming trace reporting each streamingData is encoded according to the format specified in the clause 5 of TS 32.423 [39].  For streaming performance data reporting each streamingData is encoded according to the format specified in the Annex C of TS 28.550 [42].  For proprietary data streaming reporting each streamingData is encoded according to the format specified in the product documentation. |

##### 11.5.1.3.3 Output parameters

| Parameter Name | S | Matching Information | Comment |
| --- | --- | --- | --- |
| status | M | ENUM (Success, Failure) | An operation may fail because of a specified or unspecified reason. |

##### 11.5.1.3.4 Exceptions

| Exception Name | Definition |
| --- | --- |
|  |  |

#### 11.5.1.4 addStream operation (M)

##### 11.5.1.4.1 Definition

This operation allows the MnS producer to add one or more reporting streams to an already established streaming connection.

##### 11.5.1.4.2 Input parameters

| Parameter Name | S | Information type | Comment |
| --- | --- | --- | --- |
| connectionId | M | See clause 11.5.1.1.3 | It identifies the streaming connection to which new reporting streams are being added. The format may have dependency on the solution set. |
| streamInfoList | M | List of StreamInfo | This parameter contains the list of meta-data about each reporting stream being added to the already established connection.  For streaming trace reporting each StreamInfo includes:  - StreamType carrying the value "TRACE";  - SerializationFormat carrying the value "GPB" or "ASN1";  - streamId globally unique stream identifier  - list of Trace Reference (see clause 5.6 of TS 32.422 [38]) for signaling based trace  - list of tuple of <Trace Reference (see clause 5.6 of TS 32.422 [38]), jobId (see clause 4.3.30 of TS 28.622 [11]) providing the id of the job for the configuration> for management based trace  For streaming performance data reporting each StreamInfo includes:  - StreamType carrying the value "PERFORMANCE";  - SerializationFormat carrying the value "GPB" or "ASN1";  - streamId globally unique stream identifier;  - measObjDn: the DN of the measured object instance;  - performanceMetrics: a list of performance metric (i.e. measurement or KPI) names whose values are to be reported by the Performance Data Stream Units (see Annex C of TS 28.550 [42]) via this stream;  - either:  - jobId defined in the PerfMetricJob MOI (see clause 4.3.31 of TS 28.622 [11]) for which the data is being reported;  - or:  - jobId globally unique identifier of a measurement job (see TS 28.550 [42]).  For streaming analytics reporting each StreamInfo includes:  - StreamType carrying the value "ANALYTICS";  - SerializationFormat carrying the value "GPB" or "ASN1";  - streamId globally unique stream identifier;  - AnalyticsInfo providing the details about the analytics activity for which the data is being reported.  For proprietary data streaming reporting each StreamInfo includes:  - StreamType carrying the value "PROPRIETARY";  - streamId globally unique stream identifier;  - VsDataContainer (see clause 4.3.9 of TS 28.622 [11]) providing the details about the data being reported. |

##### 11.5.1.4.3 Output parameters

| Parameter Name | S | Matching Information | Comment |
| --- | --- | --- | --- |
| streamInfoList | M | List of StreamInfo | This parameter contains the list of meta-data about each reporting stream that has been successfully added as a result of this operation.  For streaming trace reporting each StreamInfo includes:  - StreamType carrying the value "TRACE";  - SerializationFormat carrying the value "GPB" or "ASN1";  - streamId globally unique stream identifier  - list of Trace Reference (see clause 5.6 of TS 32.422 [38]) for signaling based  - list of tuple of <Trace Reference (see clause 5.6 of TS 32.422 [38]), jobId (see clause 4.3.30 of TS 28.622 [11]) providing the id of the job for the configuration> for management based trace  For streaming performance data reporting each StreamInfo includes:  - StreamType carrying the value "PERFORMANCE";  - SerializationFormat carrying the value "GPB" or "ASN1";  - streamId globally unique stream identifier;  - measObjDn: the DN of the measured object instance;  - performanceMetrics: a list of performance metric names whose values are to be reported by the Performance Data Stream Units (see Annex C of TS 28.550 [42]) via this stream. Performance metrics include measurement and KPI;  - either:  - jobId defined in the PerfMetricJob MOI (see clause 4.3.31 of TS 28.622 [11]) for which the data is being reported;  - or:  - jobId globally unique identifier of a measurement job (see TS 28.550 [42]).  For streaming analytics reporting each StreamInfo includes:  - StreamType carrying the value "ANALYTICS";  - SerializationFormat carrying the value "GPB" or "ASN1";  - streamId globally unique stream identifier;  - AnalyticsInfo providing the details about the analytics activity for which the data is being reported.  For proprietary data streaming reporting each StreamInfo includes:  - StreamType carrying the value "PROPRIETARY";  - streamId globally unique stream identifier;  - VsDataContainer (see clause 4.3.9 of TS 28.622 [11]) providing the details about the data being reported. |
| status | M | ENUM (Success, Failure, PartialSuccess) | An operation may fail because of a specified or unspecified reason. |

##### 11.5.1.4.4 Exceptions

| Exception Name | Definition |
| --- | --- |
| duplicateStream | **Condition:** One or more of stream identifiers in the streamInfoList already exist on this connection.  **Returned Information:** Name of the exception; status is set to "Failure" or "PartialSuccess". |
| unexpectedStreams | **Condition:** Some information in the list of streamInfo was unexpected by the MnS consumer.  **Returned Information:** Name of the exception; status is set to "Failure". |
| unknownConnection | **Condition:** the connectionId is invalid.  **Returned Information:** Name of the exception; status is set to "Failure". |

#### 11.5.1.5 deleteStream operation (M)

##### 11.5.1.5.1 Definition

This operation allows the MnS producer to remove one or more reporting streams from an already established streaming connection.

##### 11.5.1.5.2 Input parameters

| Parameter Name | S | Information type | Comment |
| --- | --- | --- | --- |
| connectionId | M | See clause 11.5.1.1.3 | It identifies the streaming connection from which the reporting streams are being removed. The format may have dependency on the solution set. |
| streamIdList | M | List of stream identifiers | This parameter contains the list of identifiers for streams being removed from the already established connection.  For streaming trace reporting streamId globally unique stream identifier and Trace Reference (see clause 5.6 of TS 32.422 [38]).  For streaming performance data reporting streamId globally unique stream identifier.  For streaming analytics reporting streamId globally unique stream identifier.  For proprietary data streaming reporting streamId globally unique stream identifier. |

##### 11.5.1.5.3 Output parameters

| Parameter Name | S | Matching Information | Comment |
| --- | --- | --- | --- |
| status | M | ENUM (Success, Failure, PartialSuccess) | An operation may fail because of a specified or unspecified reason. |

##### 11.5.1.5.4 Exceptions

| Exception Name | Definition |
| --- | --- |
| unknownStreamId | **Condition:** One or more of stream identifiers in the streamIdList does not exist on this connection.  **Returned Information:** Name of the exception; status is set to "Failure" or "PartialSuccess". |
| unknownConnection | **Condition:** the connectionId is invalid.  **Returned Information:** Name of the exception; status is set to "Failure". |

#### 11.5.1.6 getConnectionInfo operation (M)

##### 11.5.1.6.1 Definition

This operation enables the MnS producer to obtain information about one or more streaming connections from the MnS consumer.

##### 11.5.1.6.2 Input parameters

| Parameter Name | S | Information type | Comment |
| --- | --- | --- | --- |
| connectionIdList | M | List of streaming connection identifiers | This parameter contains the list of streaming connection identifiers for which the stream information is to be returned.  The empty list indicates the stream information for all connections are to be returned. |

##### 11.5.1.6.3 Output parameters

| Parameter Name | S | Matching Information | Comment |
| --- | --- | --- | --- |
| connectionInfoList | M | List of <connectionId, streamReporter, streamIdList> tuples | This parameter contains the list of meta-data about each streaming connection requested by this operation. Each entry in this list is a tuple of connectionId, streamReporter and streamIdList.  For streaming trace reporting:  - streamReporter is the identity of the streaming data reporting MnS producer reporting data for this connectionId;  - streamIdList is the list of streamId globally unique stream identifiers.  For streaming performance data reporting:  - streamReporter is the identity of the streaming data reporting MnS producer reporting data for this connectionId;  - streamIdList is the list of streamId globally unique stream identifiers.  For streaming analytics reporting:  - streamReporter is the identity of the streaming data reporting MnS producer reporting data for this connectionId;  - streamIdList is the list of streamId globally unique stream identifiers.  For streaming proprietary data reporting:  - streamReporter is the identity of the streaming data reporting MnS producer reporting data for this connectionId;  - streamIdList is the list of streamId globally unique stream identifiers. |
| status | M | ENUM (Success, Failure, PartialSuccess) | An operation may fail because of a specified or unspecified reason. |

##### 11.5.1.6.4 Exceptions

| Exception Name | Definition |
| --- | --- |
| unknownConnectionId | **Condition:** One or more of connection identifiers in the connectionIdList is not known to this MnS consumer.  **Returned Information:** Name of the exception; status is set to "Failure" or "PartialSuccess". |

#### 11.5.1.7 getStreamInfo operation (M)

##### 11.5.1.7.1 Definition

This operation enables theMnS producer to obtain information about one or more reporting streams the MnS consumer.

##### 11.5.1.7.2 Input parameters

| Parameter Name | S | Information type | Comment |
| --- | --- | --- | --- |
| streamIdList | M | List of stream identifiers | This parameter contains the list of stream identifiers for which the stream information is to be returned.  The empty list indicates the stream information for all streams are to be returned.  For streaming trace reporting streamId globally unique stream identifier.  For streaming performance data reporting streamId globally unique stream identifier.  For streaming analytics reporting streamId globally unique stream identifier.  For proprietary data streaming reporting streamId globally unique stream identifier. |

##### 11.5.1.7.3 Output parameters

| Parameter Name | S | Matching Information | Comment |
| --- | --- | --- | --- |
| streamInfoSumList | M | List of <StreamInfo, StreamReporters> tuples | This parameter contains the list of meta-data about each reporting stream requested by this operation. Each entry in this list is a tuple of StreamInfo and StreamReporters.  For streaming trace reporting each StreamInfo includes:  - StreamType carrying the value "TRACE";  - SerializationFormat carrying the value "GPB" or "ASN1";  - streamId globally unique stream identifier  - list of Trace Reference (see clause 5.6 of TS 32.422 [38]) for signaling based  - list of tuple of <Trace Reference (see clause 5.6 of TS 32.422 [38]), jobId (see clause 4.3.30 of TS 28.622 [11]) providing the id of the job for the configuration> for management based trace  For streaming trace the StreamReporters is a list of the identities of the streaming data reporting MnS producer(s) reporting data for this Trace Reference to this MnS consumer.  For streaming PM reporting each StreamInfo includes:  - StreamType carrying the value "PERFORMANCE";  - SerializationFormat carrying the value "GPB" or "ASN1";  - streamId globally unique stream identifier;  - measObjDn: the DN of the measured object instance;  - performanceMetrics: a list of performance metric names whose values are to be reported by the Performance Data Stream Units (see Annex C of TS 28.550 [42]) via this stream. Performance metrics include measurement and KPI;  - either:  - jobId defined in the PerfMetricJob MOI (see clause 4.3.31 of TS 28.622 [11]) for which the data is being reported;  - or:  - jobId globally unique identifier of a measurement job (see TS 28.550 [42]).  For streaming performance data the StreamReporters is a list of the identities of the streaming data reporting MnS producer(s) reporting data for this streamId to this MnS consumer.  For streaming analytics reporting each StreamInfo includes:  - StreamType carrying the value "ANALYTICS";  - SerializationFormat carrying the value "GPB" or "ASN1";  - streamId globally unique stream identifier;  - AnalyticsInfo providing the details about the analytics activity for which the data is being reported.  For streaming analytics the StreamReporters is a list of the identities of the streaming data reporting MnS producer(s) reporting data for this streamId to this MnS consumer.  For proprietary data streaming reporting each StreamInfo includes:  - StreamType carrying the value "PROPRIETARY";  - streamId globally unique stream identifier;  - VsDataContainer (see clause 4.3.9 of TS 28.622 [11]) providing the details about the data being reported.  For proprietary data streaming the StreamReporters is a list of the identities of the streaming data reporting MnS producer(s) reporting data for this streamId to this MnS consumer. |
| status | M | ENUM (Success, Failure, PartialSuccess) | An operation may fail because of a specified or unspecified reason. |

##### 11.5.1.7.4 Exceptions

| Exception Name | Definition |
| --- | --- |
| unknownStreamId | **Condition:** One or more of stream identifiers in the streamIdList is not known to this MnS consumer.  **Returned Information:** Name of the exception; status is set to "Failure" or "PartialSuccess". |

## 11.6 File data reporting service

### 11.6.1 Operations and notifications

#### 11.6.1.1 Notification notifyFileReady

##### 11.6.1.1.1 Definition

A MnS producer sends this notification to subscribed MnS consumers when a new file becomes ready (available) on the MnS producer for upload by MnS consumers. The "fileInfoList" parameter provides information (meta data) about the new file and optionally, in addition to that, information about all other files, which became ready for upload earlier and are still available for upload when the notification is sent.

The "objectClass" and "objectInstance" parameters of the notification header identify the object representing the function (process) making the file available for retrieval, such as the "PerfMetricJob" or the "TraceJob" defined in TS 28.622 [11]. When no dedicated object is standardized or instantiated, the "ManagedElement", where the file is processed, shall be used. For the case that the file is processed on a mangement node, the "ManagementNode", where the file is processed, shall be used instead.

##### 11.6.1.1.2 Input parameters

| Parameter Name | S | Information Type | Comment |
| --- | --- | --- | --- |
| objectClass | M | Entity.objectClass | See clause 11.6.1.1.1 for the definition of Entity |
| objectInstance | M | Entity.objectInstance | See clause 11.6.1.1.1 for the definition of Entity |
| notificationId | M | -- |  |
| notificationType | M | "notifyFileReady" |  |
| eventTime | M | -- | Time when the file, that triggered this notification, was ready for upload. |
| systemDN | M |  |  |
| fileInfoList | M | List of struct  <  fileLocation (M),  fileCompression (M),  fileSize (O),  fileDataType (M),  fileFormat (M),  fileReadyTime (O),  fileExpirationTime (O),  …jobId (CO)  >  Each element is defined as following:  - "fileLocation": Location of the file. The location may be a directory path or a URL, for example  "\\202.112.101.1\D:\user\Files\<xxx>", or  "ftp://nms.telecom\_org.com/datastore/<xxx>,  where <xxx> is the filename.  - "fileCompression": Name of the algorithm used for compressing the file. An empty or absent "fileCompression" parameter indicates the file is not compressed. The MnS producer selects the compression algorithm. It is encouraged to use popular algorithms such as GZIP.  - "fileSize": Size of the file. Its value is a non negative integer. The unit is byte.  - "fileDataType": Type of the management data stored in the file. Allowed values are:  - "PERFORMANCE"  - "TRACE"  - "ANALYTICS"  - "PROPRIETARY"  The value "PERFORMANCE" refers to measurements and KPIs.  - "fileFormat": Identifier of the XML or ASN.1 schema (incl. its version) used to produce the file content.  - "fileReadyTime": Date and time when the file was closed (the last time) and made available on the MnS producer. The file content will not be changed anymore.  - "fileExpirationTime": Date and time after which the file may be deleted. It shall not be empty and shall be later than "fileReadyTime".  - "jobId": Job identifier of the "PerfMetricJob" (TS 28.622 [11]) or "TraceJob" (TS 28.622 [11]) that produced the file. This parameter should be present, when the file is related to a job and that job is represented by a "PerfMetricJob" or "TraceJob". Multiple jobs may share the same job identifier. This may for example be the case for jobs collecting measurements to compuate a KPI or for jobs related to a specific task in some analytics application. Note that a specific job is identified by the objectClass/objectInstance parameters of the notification header. | Information (meta data) about the new file, that became ready for upload and triggered this notification, and information about files, which became ready for upload earlier and are still available for upload when the notification is sent. |
| additionalText | O | -- | Allows a free form text description to be reported as defined in ITU-T Rec. X. 733 [4] |

#### 11.6.1.2 Notification notifyFilePreparationError

##### 11.6.1.2.1 Definition

A MnS producer sends this notification to subscribed MnS consumers when an error occurs while preparing a file. For many error reasons, such as low memory or hard disk full, it is very likely that all ongoing file preparation processes fail at the same time. For that reason, it is possible to report with this notification that multiple file preparation processes failed.

In case the MnS producer keeps the file, where an error occurred during preparation, the "fileInfoList" parameter contains a list item with information about that file, otherwise, if the file is deleted or not created at all, the "fileInfoList" parameter has no list item related to that file.

##### 11.6.1.2.2 Input parameters

| Parameter Name | S | Information Type | Comment |
| --- | --- | --- | --- |
| objectClass | M | Entity.objectClass. | See clause 11.6.1.1.1 for the definition of Entity |
| objectInstance | M | Entity.objectInstance | See clause 11.6.1.1.1 for the definition of Entity. |
| notificationId | M | -- | See Table 11.6.1.1.2-1. |
| notificationType | M | "notifyFilePreparationError" |  |
| eventTime | M | -- | Time when the file preparation error occured |
| systemDN | M |  |  |
| fileInfoList | M | See Table 11.6.1.1.2-1. | Each list item contains information about a file where a file preparation error occurred and that is kept on the MnS producer. Files, that are deleting or not created at all, have no list item. |
| reason | M | -- | Detailed error reason, including  - errorInPreparation  - hardDiskFull  - hardDiskFailure  - tooManyFiles  - collectionTimeOut  - incompleteTruncatedFile  - corruptedFile  - lowMemory  - dataNotAvailable |
| additionalText | O | -- | Allows a free form text description to be reported as defined in ITU-T Rec. X. 733 [4] |

#### 11.6.1.3 Operation subscribe

##### 11.6.1.3.1 Definition

This operation allows a MnS consumer to subscribe to the notifications of the file data reporting service producer.

##### 11.6.1.3.2 Input parameters

| Parameter Name | S | Information Type | Comment |
| --- | --- | --- | --- |
| consumerReference | M | Reference (address) of the MnS consumer to which the notifications shall be sent. |  |
| timeTick | O | Initial value of a timer held by the MnS producer. This value defines the time window within which the MnS consumer intends to invoke the "subscribe" operation again to confirm its subscription. The value "0" shall indicate infinity. In this case the subscription is not terminated by the MnS producer.  Unit is minutes |  |
| filter | O | Filter constraint that the MnS producer shall use to filter notifications. The filter can be applied to all parameters of a notification  The filter constraint grammar is solution set dependent |  |

##### 11.6.1.3.3 Output parameters

| **Parameter Name** | **S** | **Matching Information** | **Comment** |
| --- | --- | --- | --- |
| subscriptionId | M | Unambiguous identity of this subscription. |  |
| status | M | ENUM (OperationSucceeded, OperationFailedExistingSubscription, OperationFailed) | If subscription is successfully created, status = OperationSuceeded.  If subscription is not created because it is duplicated or conflict with existing subscription(s), status = OperationFailedExistingSubscription  If the operation is failed for any other reason than being duplicated or conflict with existing subscription(s), status = OperationFailed. |

##### 11.6.1.3.4 Exceptions

|  |  |
| --- | --- |
| **Name** | **Definition** |
| operation\_failed\_existing\_subscription | **Condition:** The subscription is duplicated or conflict with existing subscription(s)  **Returned Information:** The output parameter status |
| operation\_failed | **Condition:** The operation failed for any other reason than being duplicated or conflict with subscription(s)  **Returned Information:** The output parameter status |

#### 11.6.1.4 Operation unsubscribe

##### 11.6.1.4.1 Definition

This operation allows a MnS consumer to cancel subscription(s) at a MnS producer.

A MnS consumer can cancel one subscription made with a "consumerReference" by providing the corresponding "subscriptionId" or all subscriptions made with the same "consumerReference" by leaving the "subscriptionId" parameter absent.

##### 11.6.1.4.2 Input parameters

| **Parameter Name** | **S** | **Information Type** | **Comment** |
| --- | --- | --- | --- |
| consumerReference | M | Reference of the MnS consumer whose subscriptions are to be cancelled. | The format of the reference may have dependency on the solution set. |
| subscriptionId | O | Subscription id returned in the subscribe operation response | If this parameter is absent, all subscriptions made with the same "consumerReference" shall be cancelled. |

##### 11.6.1.4.3 Output parameters

| **Parameter Name** | **S** | **Matching Information** | **Comment** |
| --- | --- | --- | --- |
| status | M | ENUM (OperationSucceeded, OperationFailed) | If subscription(s) as identified in the input parameter are cancelled, status = OperationSucceeded.  If the operation is failed, status = OperationFailed. |

##### 11.6.1.4.4 Exceptions

|  |  |
| --- | --- |
| **Name** | **Definition** |
| operation\_failed | **Condition:** the operation is failed  **Returned Information:** The output parameter status |

#### 11.6.1.5 Operation listAvailableFiles

##### 11.6.1.5.1 Definition

This operation allows a MnS consumer to retrieve a list of files available for upload on a MnS producer. The request message contains the file data type of the files, that shall be listed in the response. In addition to that it is possible to specify that only files shall be included in the response whose file ready time falls into a specific time window defined by the "beginTime" and "endTime" input parameters.

##### 11.6.1.5.2 Input parameters

| **Parameter Name** | **S** | **Information type** | **Comment** |
| --- | --- | --- | --- |
| fileDataType | M | It specifies the type of the management data stored in the file. | For performance data (including measurement data and KPI) files, the value is assigned to "PERFORMANCE".  For trace data files, the value is assigned to "TRACE".  For analytic data files, the value is assigned to "ANALYTICS".  For proprietary data files, the value is assigned to "PROPRIETARY". |
| beginTime | M | The consumer requests to list information about the available file(s) whose ready time(s) are later or equal to this time.  This parameter is expressed in UTC time. | This parameter indicates date and time.  If this parameter is empty or absent, no restriction on begin time is applied on the file ready time. |
| endTime | M | The consumer requests to list information about the available file(s) whose ready time(s) are earlier than this time.  This parameter is expressed in UTC time. | This parameter indicates date and time.  If this parameter is empty or absent, no restriction on end time is applied on the file ready time. |

##### 11.6.1.5.3 Output parameters

| **Parameter Name** | **S** | **Matching Information** | **Comment** |
| --- | --- | --- | --- |
| fileInfoList | M | See "fileInfoList" defined in notifyFileReady notification (clause 11.6.1.1.1) |  |
| status | M | ENUM (Success, Failure) |  |

##### 11.6.1.5.4 Exceptions

| **Exception Name** | **Definition** |
| --- | --- |
| invalidTimes | **Condition:** Either "beginTime" or "endTime" is invalid.  **Returned information:** output parameter status is set to Failure. |

### 11.6.2 File transfer protocols

The MnS producer shall support at least one of the following file transfer protocols:

- SFTP;

- FTPES,

- HTTPS.

The MnS producer shall always act as the server while the MnS consumer shall always act as the initiator (client) of file transfer actions.

# 12 Management services – Stage 3

## 12.1 Generic provisioning management service

### 12.1.1 RESTful HTTP-based solution set

#### 12.1.1.1 Mapping of operations

##### 12.1.1.1.1 Introduction

The IS operations are mapped to SS equivalents according to table 12.1.1.1.1-1.

Table 12.1.1.1.1-1: Mapping of IS operations to SS equivalents

|  |  |  |  |
| --- | --- | --- | --- |
| **IS operation** | **HTTP Method** | **Resource URI** | **S** |
| createMOI | PUT | {MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{className}={id} | M |
| getMOIAttributes | GET | {MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{className}={id} | M |
| modifyMOIAttributes | PUT  PATCH | {MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{className}={id} | M |
| deleteMOI | DELETE | {MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{className}={id} | M |
| changeMOIs | PATCH | {MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{className}={id} | M |

##### 12.1.1.1.2 Operation createMOI

This operation creates a single resource representing a managed object instance.

Table 12.1.1.1.2-1: Mapping of IS operation input parameters to SS equivalents (HTTP PUT)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| managedObjectClass  managedObjectInstance | path | …/{className}={id} | className: string  id: string | M |
| attributeListIn | request body | n/a | Resource | M |

Note 1: Void.

Table 12.1.1.1.2-2: Mapping of IS operation output parameters to SS equivalents (HTTP PUT)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IS parameter name | SS parameter location | SS parameter name | SS parameter type | S |
| attributeListOut | response body | n/a | Resource | M |
| status | response status codes | n/a | n/a | M |
| response body | error | ErrorResponseDefault | O |

Further details on creating a resource with HTTP PUT are provided in TS 32.158 [15], clause 5.1.2.

##### 12.1.1.1.3 Operation getMOIAttributes

This operation retrieves one or multiple resources representing managed object instances.

Table 12.1.1.1.3-1: Mapping of IS operation input parameters to SS equivalents (HTTP GET)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| baseObjectInstance | path | /{className}={id} | className: string  id: string | M |
| scope | query | scope | Scope  style: form  explode: true | O |
| filter | query | filter | Filter | O |
| attributeListIn | query | attributes | array(string)  style: form  explode: false | O |
| fields | array(string)  style: form  explode: false | O |
| dataNodeSelector | query | dataNodeSelector | Filter | O |

Note 1: Void.

Note 2: Void.

Table 12.1.1.1.3-2: Mapping of IS operation output parameters to SS equivalents (HTTP GET)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| attributeListOut | response body | n/a | Resource or  array(Resource) | M |
| status | response status codes | n/a | n/a | M |
| response body | error | ErrorResponseGet | O |

Further details on reading resources with HTTP GET are provided in TS 32.158 [15], clause 5.2.

Further details on the SS parameters "scope" and "filter" are provided in TS 32.158 [15], clause 6.1.

Further details on the SS parameters "attributes" and "fields" are provided in TS 32.158 [15], clause 6.2.

Further details on the SS parameter "dataNodeSelector" is provided in TS 32.158 [15], clause 6.2a.

##### 12.1.1.1.4 Operation modifyMOIAttributes

###### 12.1.1.1.4.1 Mapping to HTTP PUT

HTTP PUT is used for a full update of a single resource.

Table 12.1.1.1.4.1-1: Mapping of IS operation input parameters to SS equivalents (HTTP PUT)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| baseObjectInstance | path | /{className}={id} | className: string  id: string | M |
| scope | n/a | n/a | n/a | n/a |
| filter | n/a | n/a | n/a | n/a |
| modificationList | request body | n/a | Resource | M |

The IS parameters "scope" and "filter" have no meaning when targeting a single resource with the target URI and are not mapped.

Table 12.1.1.1.4.1-2: Mapping of IS operation output parameters to SS equivalents (HTTP PUT)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| attributeListOut | response body | n/a | Resource | O |
| status | response status codes | n/a | n/a | M |
| response body | error | ErrorResponseDefault | O |

Further details on updating a resource with HTTP PUT are provided in TS 32.158 [15], clause 5.3.

###### 12.1.1.1.4.2 Mapping to HTTP PATCH

HTTP PATCH is used to create, update or delete one or multiple resources.

Table 12.1.1.1.4.2-1: Mapping of IS operation input parameters to SS equivalents (HTTP PATCH)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| baseObjectInstance | path | …/{className}={id} | className: string  id: string | M |
| scope | n/a | n/a | n/a | n/a |
| filter | n/a | n/a | n/a | n/a |
| modificationList | request body | n/a | Resource, or  array(PatchItem) | M |

Four patch media types are available for the request message body. They are listed below together with their request body data types:

- "application/merge-patch+json" (RFC 7396 [37]), request body type: Resource

- "application/vnd.3gpp.merge-patch+json" (TS 32.158 [15]), request body type: Resource

- "application/json-patch+json" (RFC 6902 [36]), request body type: array(PatchItem)

- "application/vnd.3gpp.json-patch+json" (TS 32.158 [15]), request body type: array(PatchItem)

If the MnS producer cannot honoura patch request for some reason, such as malformed requests or unsupported patch operations, an error response with an appropriate error response code such as "400 Bad Request" shall be returned.

The patch operations "copy" and "move" have no corresponding definition in stage 2. Support for these operations is optional.

The IS parameters "scope" and "filter" have no SS equivalents in the present document.

Table 12.1.1.1.4.2-2: Mapping of IS operation output parameters to SS equivalents (HTTP PATCH)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| attributeListOut | response body | n/a | Resource | M |
| status | response status codes | n/a | n/a | M |
| response body | error | ErrorResponseDefault, or  ErrorResponsePatch | O |

For JSON Merge Patch and 3GPP JSON Merge Patch the type "ErrorResponseDefault" is used.

For JSON Patch and 3GPP JSON Patch the type "ErrorResponsePatch" is used.

Further details on updating resources with HTTP PATCH and JSON Merge Patch are provided in TS 32.158 [15], clause 6.3.2.

Further details on updating resources with HTTP PATCH and 3GPP JSON Merge Patch are provided in TS 32.158 [15], clause 6.4.2.

Further details on updating resources with HTTP PATCH and JSON Patch are provided in TS 32.158 [15], clause 6.3.3.

Further details on updating resources with HTTP PATCH and 3GPP JSON Patch are provided in TS 32.158 [15], clause 6.4.3.

Note 1: Void.

##### 12.1.1.1.5 Operation deleteMOI

This operation deletes a single resource representing a managed object instance

Table 12.1.1.1.5-1: Mapping of IS operation input parameters to SS equivalents (HTTP DELETE)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| baseObjectInstance | path | /{className}={id} | className: string  id: string | M |
| scope | n/a | n/a | n/a | n/a |
| filter | n/a | n/a | n/a | n/a |

Note 1: Void.

Note 2: Void.

Table 12.1.1.1.5-2: Mapping of IS operation output parameters to SS equivalents (HTTP DELETE)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| deletionlist | n/a | n/a | n/a | n/a |
| status | response status codes | n/a | n/a | M |
| response body | error | ErrorResponseDefault | O |

Further details on deleting a resource with HTTP DELETE are provided in TS 32.158 [15], clause 5.4.

##### 12.1.1.1.6 Void

##### 12.1.1.1.7 Void

##### 12.1.1.1.8 Operation changeMOIs

This operation creates, deletes, and updates one or more objects using a single request.

Table 12.1.1.1.8-1: Mapping of IS operation input parameters to SS equivalents (HTTP PATCH)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| baseObjectInstance | path | /{className}={id} | className: string  id: string | M |
| modificationsIn | query | n/a | Resource, or  array(PatchItem) | M |

Two patch media types are available for the request message body. They are listed below together with their request body data types:

- "application/vnd.3gpp.merge-patch+json" (TS 32.158 [15]), request body type: Resource

- "application/vnd.3gpp.json-patch+json" (TS 32.158 [15]), request body type: array(PatchItem)

If the MnS producer cannot honour a patch request for some reason, such as malformed requests or unsupported patch operations, an error response with an appropriate error response code such as "400 Bad Request" shall be returned.

The patch operations "copy" and "move" have no corresponding definition in stage 2. Support for these operations is optional.

Table 12.1.1.1.8-2: Mapping of IS operation output parameters to SS equivalents (HTTP PATCH)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| modificationsOut | response body | n/a | Resource | M |
| status | response status codes | n/a | n/a | M |
| response body | n/a | ErrorResponseDefault, or  ErrorResponsePatch | O |

For JSON Merge Patch and 3GPP JSON Merge Patch the "ErrorResponseDefault" is used.

For JSON Patch and 3GPP JSON Patch the "ErrorResponsePatch" is used.

Further details on updating resources with HTTP PATCH and 3GPP JSON Merge Patch are provided in TS 32.158 [15], clause 6.4.2.

Further details on updating resources with HTTP PATCH and 3GPP JSON Patch are provided in TS 32.158 [15], clause 6.4.3.

Further details on the error response formats are provided in TS 32.158 [15], clause 6.6.

#### 12.1.1.2 Mapping of notifications

##### 12.1.1.2.1 Introduction

The IS notifications are mapped to SS equivalents according to table 12.1.1.2.1-1.

**Table 12.1.1.2.1-1: Mapping of IS notifications to SS equivalents**

|  |  |  |  |
| --- | --- | --- | --- |
| **IS notification** | **HTTP Method** | **Resource URI** | **S** |
| notifyMOICreation | POST | {notificationTarget} | M |
| notifyMOIDeletion | POST | {notificationTarget} | M |
| notifyMOIAttributeValueChanges | POST | {notificationTarget} | M |
| notifyMOIChanges | POST | {notificationTarget} | M |
| notifyEvent | POST | {notificationTarget} | M |

##### 12.1.1.2.2 Notification notifyMOICreation

The IS notification parameters are mapped to SS equivalents according to table 12.1.1.2.2-1.

Table 12.1.1.2.2-1: Mapping of IS notification input parameters to SS equivalents (HTTP POST)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| objectClass | request body | href | Uri | M |
| objectInstance |
| notificationId | request body | notificationId | NotificationId | M |
| notificationType | request body | notificationType | NotificationType | M |
| eventTime | request body | eventTime | DateTime | M |
| systemDN | request body | systemDN | SystemDN | M |
| correlatedNotifications | request body | correlatedNotifications | array(CorrelatedNotification) | O |
| additionalText | request body | additionalText | AdditionalText | O |
| sourceIndicator | request body | sourceIndicator | SourceIndicator | O |
| attributeList | request body | attributeList | AttributeNameValuePairSet | O |

##### 12.1.1.2.3 Notification notifyMOIDeletion

The IS notification parameters are mapped to SS equivalents according to table 12.1.1.2.3-1.

Table 12.1.1.2.3-1: Mapping of IS notification input parameters to SS equivalents (HTTP POST)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| objectClass | request body | href | Uri | M |
| objectInstance |
| notificationId | request body | notificationId | NotificationId | M |
| notificationType | request body | notificationType | NotificationType | M |
| eventTime | request body | eventTime | DateTime | M |
| systemDN | request body | systemDN | SystemDN | M |
| correlatedNotifications | request body | correlatedNotifications | array(CorrelatedNotification) | O |
| additionalText | request body | additionalText | AdditionalText | O |
| sourceIndicator | request body | sourceIndicator | SourceIndicator | O |
| attributeList | request body | attributeList | AttributeNameValuePairSet | O |

##### 12.1.1.2.4 Notification notifyMOIAttributeValueChanges

The IS notification parameters are mapped to SS equivalents according to table 12.1.1.2.4-1.

Table 12.1.1.2.4-1: Mapping of IS notification input parameters to SS equivalents (HTTP POST)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| objectClass | request body | href | Uri | M |
| objectInstance |
| notificationId | request body | notificationId | NotificationId | M |
| notificationType | request body | notificationType | NotificationType | M |
| eventTime | request body | eventTime | DateTime | M |
| systemDN | request body | systemDN | SystemDN | M |
| correlatedNotifications | request body | correlatedNotifications | array(CorrelatedNotification) | O |
| additionalText | request body | additionalText | AdditionalText | O |
| sourceIndicator | request body | sourceIndicator | SourceIndicator | O |
| attributeListValueChanges | request body | attributeListValueChange | AttributeValueChangeSet | M |

##### 12.1.1.2.5 Notification notifyMOIChanges

The IS notification parameters are mapped to SS equivalents according to table 12.1.1.2.5-1.

Table 12.1.1.2.5-1: Mapping of IS notification input parameters to SS equivalents (HTTP POST)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| objectClass | request body | href | Uri | M |
| objectInstance |
| notificationId | request body | notificationId | NotificationId | M |
| notificationType | request body | notificationType | NotificationType | M |
| eventTime | request body | eventTime | DateTime | M |
| systemDN | request body | systemDN | SystemDN | M |
| moiChanges | request body | mOIChanges | array(MoiChange) | M |

12.1.1.2.6 Notification notifyEvent

The IS notification parameters are mapped to SS equivalents according to table 12.1.1.2.6-1.

**Table 12.1.1.2.6-1: Mapping of IS notification input parameters to SS equivalents (HTTP POST)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| objectClass | request body | href | Uri | M |
| objectInstance |
| notificationId | request body | notificationId | NotificationId | M |
| notificationType | request body | notificationType | NotificationType | M |
| eventTime | request body | eventTime | DateTime | M |
| systemDN | request body | systemDN | SystemDN | M |
| specificProblem | request body | specificProblem | SpecificProblem | M |
| additionalText | request body | additionalText | string | O |
| additionalInformation | request body | additionalInformation | AttributeNameValuePairSet | O |

#### 12.1.1.3 Resources

##### 12.1.1.3.1 Resource structure

12.1.1.3.1.1 Resource structure on the MnS producer

Figure 12.1.1.3.1.1-1 shows the resource structure of the Provisioning MnS on the MnS producer.



Figure 12.1.1.3.1.1-1: Resource URI structure of the Provisioning MnS on the MnS producer

Table 12.1.1.3.1.1-1 provides an overview of the resources and applicable HTTP methods.

Table 12.1.1.3.1-1: Resources and methods overview

|  |  |  |  |
| --- | --- | --- | --- |
| Resource name | Resource URI | HTTP method | Description |
| MOI | …/{className}={id} | PUT | Create a resource representing a managed object instance |
| MOI | …/{className}={id} | GET | Retrieve one or multiple resources representing managed object instances |
| MOI | …/{className}={id} | PATCH | Modifiy one or multiple resources representing managed object instances |
| MOI | …/{className}={id} | DELETE | Delete one or multiple resources representing managed object instances |

###### 12.1.1.3.1.2 Resource structure on the MnS consumer

Figure 12.1.1.3.1.2-1 shows the resource structure of the Provisioning MnS on the MnS consumer.



Figure 12.1.1.3.1.2-1: Resource URI structure of the Provisioning MnS on the MnS consumer

Table 12.1.1.3.1.2-1 provides an overview of the resources and applicable HTTP methods.

Table 12.1.1.3.1.2-1: Resources and methods overview

|  |  |  |  |
| --- | --- | --- | --- |
| Resource name | Resource URI | HTTP method | Description |
| Notification Target | {notificationTarget} | POST | Send a notification to the notification target |

##### 12.1.1.3.2 Resource definitions

###### 12.1.1.3.2.1 Resource "…/{className}={id}"

12.1.1.3.2.1.1 Description

This resource represents a managed object instance.

12.1.1.3.2.1.2 URI

Resource URI: {MnSRoot}/ProvMnS/{MnSVersion}/{URI-LDN-first-part}/{className}={id}

The resource URI variables are defined in table 12.1.1.3.2.1.2-1.

Table 12.1.1.3.2.1.2-1: URI variables

|  |  |
| --- | --- |
| **Name** | **Definition** |
| MnSRoot | See clause 4.4.2 of TS 32.158 [15] |
| MnSVersion | See clause 4.4.2 of TS 32.158 [15] |
| URI-LDN-first-part | See clause 4.4.2 of TS 32.158 [15] |
| className | Class name of the targeted resource |
| id | Identifier of the targeted resource |

12.1.1.3.2.1.3 HTTP methods

12.1.1.3.2.1.3.1 HTTP PUT

This method shall support the URI query parameters specified in the following table.

Table 12.1.1.3.2.1.3.1-1: URI query parameters supported by the PUT method on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data type** | **Description** | **S** |
| n/a | n/a | n/a | n/a |

This method shall support the request data structures, the response data structures and response codes specified in the following table.

Table 12.1.1.3.2.1.3.1-2: Data structures supported by the PUT request body on this resource

|  |  |  |
| --- | --- | --- |
| **Data type** | **Description** | **S** |
| Resource | Resource representation of the resource to be created or replaced | M |

Table 12.1.1.3.2.1.3.1-3: Data structures supported by the PUT Response Body on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| **Data type** | **Response codes** | **Description** | **S** |
| Resource | 200 OK | Status code returned when the resource is replaced, and when the replaced resource representation is not identical to the resource representation in the request.  This status code may be retourned when the resource is updated and when the updated resource representation is identical to the resource representation in the request.  The representation of the updated resource is returned in the response message body. | M |
| Resource | 201 Created | Status code returned when the resource is created. The representation of the created resource is returned in the response message body. | M |
| n/a | 204 No Content | Status code that may be returned only when the replaced resource representation is identical to the representation in the request. The response has no message body. | M |
| ErrorResponse | 4xx/5xx | Returned in case of an error | O |

12.1.1.3.2.1.3.2 HTTP GET

This method shall support the URI query parameters specified in the following table.

Table 12.1.1.3.2.1.3.2-1: URI query parameters supported by the GET method on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data type** | **Description** | **S** |
| scope | Scope  style: form  explode: true | Extends the set of targeted resources beyond the base resource identified with the authority and path component of the URI. | O |
| filter | Filter | Reduces the targeted set of resources by applying a filter to the scoped set of resource representations. Only resources representations for which the filter construct evaluates to "true" are targeted. | O |
| attributes | array(string)  style: form  explode: false | Attributes of the scoped resources to be returned. The value is a comma-separated list of attribute names. | O |
| fields | array(string)  style: form  explode: false | Attribute fields of the scoped resources to be returned. The value is a comma-separated list of JSON pointers to the attribute fields. | O |

This method shall support the request data structures, the response data structures and response codes specified in the following tables.

Table 12.1.1.3.2.1.3.2-2: Data structures supported by the GET request body on this resource

|  |  |  |
| --- | --- | --- |
| **Data type** | **Description** | **S** |
| n/a | n/a | n/a |

Table 12.1.1.3.2.1.3.2-3: Data structures supported by the GET response body on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| **Data type** | **Response codes** | **Description** | **S** |
| Resource | 200 OK | Resources identified in the request for retrieval. In case the attributes or fields query parameters are used, only the selected attributes or sub-attributes are returned. The response message body is constructed according to the hierarchical response construction method (TS 32.158 [15]) | M |
| ErrorResponse | 4xx/5xx | Returned in case of an error | M |

12.1.1.3.2.1.3.3 HTTP PATCH

This method shall support the URI query parameters specified in the following table.

Table 12.1.1.3.2.1.3.3-1: URI query parameters supported by the PATCH method on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data type** | **Description** | **S** |
| n/a | n/a | n/a | n/a |

This method shall support the request data structures, the response data structures and response codes specified in the following tables.

Table 12.1.1.3.2.1.3.3-2: Data structures supported by the PATCH request body on this resource

|  |  |  |
| --- | --- | --- |
| **Data type** | **Description** | **S** |
| Resource, or  array(object) | Patch document describing the set of modifications to be applied to the targeted resources.  The following patch media types are available:  - "application/merge-patch+json" (RFC 7396 [37])  - "application/3gpp-merge-patch+json" (TS 32.158 [15])  - "application/json-patch+json" (RFC 6902 [36])  - "application/3gpp-json-patch+json" (TS 32.158 [15]) | M |

Table 12.1.1.2.1.1.3.3-3: Data structures supported by the PATCH response body on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| **Data type** | **Response codes** | **Description** | **S** |
| ErrorResponse | 4xx/5xx | Returned in case of an error | M |

12.1.1.3.2.1.3.4 HTTP DELETE

This method shall support the URI query parameters specified in the following table.

Table 12.1.1.3.2.1.3.4-1: URI query parameters supported by the DELETE method on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data type** | **Description** | **S** |
| scope | Scope  style: form  explode: true | Extends the set of targeted resources beyond the base resource identified with the authority and path component of the URI. | O |
| filter | Filter | Reduces the targeted set of resources by applying a filter to the scoped set of resource representations. Only resources representations for which the filter construct evaluates to "true" are targeted. | O |

This method shall support the request data structures, the response data structures and response codes specified in the following tables.

Table 12.1.1.3.2.1.3.4-2: Data structures supported by the DELETE request body on this resource

|  |  |  |
| --- | --- | --- |
| **Data type** | **Description** | **S** |
| n/a | n/a | n/a |

Table 12.1.1.3.2.1.3.4-3: Data structures supported by the DELETE response body on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| **Data type** | **Response codes** | **Description** | **S** |
| array(Uri) | 200 OK | Status code returned, when query parameters are present in the request and one or multiple resources are deleted. The URIs of the deleted resources are returned in the response message body. | M |
| n/a | 204 No Content | Status code returned, when no query parameters are present in the request and only one resource is deleted. The message body is empty. | M |
| ErrorResponse | 4xx/5xx | Returned in case of an error | M |

###### 12.1.1.3.2.2 Void

###### 12.1.1.3.2.3 Void

###### 12.1.1.3.2.4 Resource "{notificationTarget}"

12.1.1.3.2.4.1 Description

This resource represents a notification target on the MnS consumer.

12.1.1.3.2.4.2 URI

Resource URI: {notificationTarget}

The resource URI variables are defined in table 12.1.1.3.2.4.2-1.

Table 12.1.1.3.2.4.2-1: URI variables

|  |  |
| --- | --- |
| Name | Definition |
| notificationTarget | URI of the notification target on the MnS consumer, contained in the notification subscription, see notificationRecipientAddress defined in clause 4.3.22.2 in TS 28.622 [11]. |

12.1.1.3.2.4.3 HTTP methods

12.1.1.3.2.4.3.1 POST

This method shall support the URI query parameters specified in table 12.1.1.3.2.4.3.1-1.

Table 12.1.1.3.2.4.3.1-1: URI query parameters supported by the POST method on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data type | Description | S |
| n/a | n/a | n/a | n/a |

This method shall support the request data structures specified in table 12.1.1.3.2.4.3.1-2 and the response data structures and response codes specified in table 12.1.1.3.2.4.3.1-3.

Table 12.1.1.3.2.4.3.1-2: Data structures supported by the POST Request Body on this resource

|  |  |  |
| --- | --- | --- |
| Data type | Description | S |
| NotifyMOICreation | Type for a notifyMOICreation notification | M |
| NotifyMOIDeletion | Type for a notifyMOIDeletion notification | M |
| NotifyAttributeValueChanges | Type for a notifyAttributeValueChanges notification | M |
| NotifyMoiChanges | Type for a notifyMOIChanges notification | M |
| NotifyEvent | Type for a notifyEvent notification | O |

Table 12.1.1.3.2.4.3.1-3: Data structures supported by the POST Response Body on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| Data type | Response codes | Description | S |
| n/a | 204 No Content | In case of success no message body is returned | M |
| ErrorResponse | 4xx/5xx | In case of failure the error object is returned. | M |

#### 12.1.1.4 Data type definitions

##### 12.1.1.4.1 General

This clause defines the data types used by the Provisioning MnS. Table 12.1.1.4.1-1 specifies the data types defined in the present document and Table table 12.1.1.4.1-2 the data types imported.

Table 12.1.1.4.1-1: Data types defined in this specification

|  |  |  |
| --- | --- | --- |
| Data type | Reference | Description |
| CmNotificationTypes | 12.1.1.4.4.3 | Notification type (notifyMOICreation, etc.) |
| SourceIndicator | 12.1.1.4.4.4 | Indicates the source of the operation that led to the generation of the notification. |
| ScopeType | 12.1.1.4.4.5 | Scope type of a scope |
| Operation | 12.1.1.4.4.6 | Enum with "create", "delete" and "replace" |
| Insert | 12.1.1.4.4.8 | Enum with "before" and "after" |
| PatchOperation | 12.1.1.4.4.7 | Enum with "add", "replace", "remove", "copy", "move" and "test" |
| Resource | 12.1.1.4.1a.1 | Used for resource representations |
| Scope | 12.1.1.4.1a.2 | Used in the query part of HTTP GET and HTTP DELETE to extend the set of targeted resources beyond the base resource identified with the authority and path component of the URI |
| CorrelatedNotification | 12.1.1.4.1a.3 | Describes the correlated notifications of a single source |
| MoiChange | 12.1.1.4.1a.4 | Single MOI change reported by notifyMOIChanges |
| NotifyMOICreation | 12.1.1.4.1a.5 | Used in the request body of HTTP POST for the notification type notifyMOICreation |
| NotifyMOIDeletion | 12.1.1.4.1a.6 | Used in the request body of HTTP POST for the notification type notifyMOIDeletion |
| NotifyMOIAttributeValueChanges | 12.1.1.4.1a.7 | Used in the request body of HTTP POST for the notification type notifyMOIAttributeValueChanges |
| NotifyMOIChanges | 12.1.1.4.1a.8 | Used in the request body of HTTP POST for the notification type notifyMOIChanges |
| NotifyEvent | 12.1.1.4.1a.10 | Used in the request body of HTTP POST for the notification type notifyEvent |
| PatchItem | 12.1.1.4.1a.9 | Specifies a patch item of a patch document |

Table 12.1.1.4.1-2: Data types imported

|  |  |  |
| --- | --- | --- |
| Data type | Reference | Description |
| DateTime | TS 28.623 [44] | Date and time |
| Dn | TS 28.623 [44] | DN type |
| SystemDN | TS 28.623 [44] | systemDN type |
| Uri | TS 28.623 [44] | URI type |
| AttributeNameValuePairSet | TS 28.623 [44] | Set of attribute name/value pairs |
| AttributeValueChangeSet | TS 28.623 [44] | Set of attribute names with their old and new values |
| Filter | TS 28.623 [44] | Filter type |
| NotificationId | TS 28.623 [44] | Notification identifier as defined in ITU-T Rec. X. 733 [4] |
| NotificationType | TS 28.623 [44] | Notification type |
| NotificationHeader | TS 28.623 [44] | Notification header |
| ErrorResponse | TS 28.623 [44] | Used in the response body of multiple HTTP methods in case of error |

##### 12.1.1.4.1a Structured data types

###### 12.1.1.4.1a.1 Type Resource

**Table 12.1.1.4.1a.1 -1: Definition of type Resource**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Description** | **S** |
| id | string | Identifier of the resource object | M |
| objectClass | string | Object class of the resource object | O |
| objectInstance | Dn | Object instance of the resource object | O |
| attributes | object | "attributes" (JSON) object whose members are the IOC attributes (except for "id”, "objectClass” and "objectInstance"). | M |
| n/a | map(array(object)) | Name contained objects | M |

This definition of "Resource" does not specify any attributes or name contained objects. Resource representations with specific attributes and name contained objects are contained in the NRM definitions. These definitions should be used in implementations of the Provisioning MnS instead of this generic definition.

###### 12.1.1.4.1a.2 Type Scope

Table 12.1.1.4.1a.2-1: Definition of type Scope

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute name | Data type | Description | S |
| scopeType | ScopeType | Used in the query component of HTTP GET and HTTP DELETE together with scopeLevel to extend the set of targeted resources beyond the base resource identified with the authority and path component of the URI | M |
| scopeLevel | integer | Used in the query component of HTTP GET and HTTP DELETE together with scopeType to extend the set of targeted resources beyond the base resource identified with the path component of the URI | M |

###### 12.1.1.4.1a.3 Type CorrelatedNotification

Table 12.1.1.4.1a.3 -1: Definition of type CorrelatedNotification

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute name | Data type | Description | S |
| source | Dn | Source of the correlated notifications | M |
| notificationIds | array(NotificationId) | Notification identifiers of correlated notifications of that source | M |

###### 12.1.1.4.1a.4 Type MoiChange

Table 12.1.1.4.1a.4 -1: Definition of type MoiChange

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute name | Data type | Description | S |
| notificationId | NotificationId | Notification identifier as defined in ITU-T Rec. X. 733 [4] | M |
| correlatedNotifications | array(CorrelatedNotification) | Set of all notifications to which this notification is considered to be correlated as defined in ITU-T Rec. X. 733 [4] | O |
| additionalText | string | Allows a free form text description to be reported as defined in ITU-T Rec. X. 733 [4] | O |
| sourceIndicator | SourceIndicator | Indicates the source of the operation that led to the generation of this notification. | O |
| op | Operation | Operation associated to the reported change ("add", "remove , "replace"). | M |
| path | string | URI path component segments specifying when appended to "href" the created, deleted or updated resource or secondary resource | M |
| insert | Insert | Indicates whether the new attribute element was added before or after the attribute element specified by "path", only valid for attributes with the property isOrdered=True. It can take the values "before" and "after". If missing, it defaults to "before".  The "insert" attribute shall be supported only when changes from YANG defined NRMs are reported. For JSON defined NRMs the attribute shall not be supported. | CM |
| value | any type | New value of the created or updated resource or secondary resource. Optional old value of the deleted resource or secondary resource | M |
| oldValue | any type | Old value of the updated secondary resource | O |

The properties "op", "path" and "value" shall use the 3GPP JSON Patch format (TS 32.158 [15]) for reporting NRM changes. The "merge" operation specified by 3GPP JSON Patch is not supported in "notifyMOIChanges". The "move", "copy" and "test" operations specified by JSON Patch are not supported either.

The "oldValue" is an optional extension for "notifyMOIChanges" allowing to report also the value that the attribute had before replacing the value with the new value, that is contained in "value".

The following example notification (where JSON is expressed in YAML notation) reports an object creation

href: https://example.com/3gpp

...

moiChanges

- notificationId: 123456789

op: add

path: /ClassA=1

value:

id: 1,

objectClass: ClassA,

attributes:

attrA: 123

attrB:

subAttrB1: ABC

subAttrB2: 56

The following example reports the deletion of that object.

href: https://example.com/3gpp

...

moiChanges

- notificationId: 123456789

op: remove

path: /ClassA=1

The following example reports the addition of a new attribute "attrC".

href: https://example.com/3gpp

...

moiChanges

- notificationId: 123456789

op: add

path: /ClassA=1#/attributes/attrC

value: xyz

The following example reports the deletion of the attribute "attrC".

href: https://example.com/3gpp

...

moiChanges

- notificationId: 123456789

op: remove

path: /ClassA=1#/attributes/attrC

The following example reports a value change for the simple attribute "attrA".

href: https://example.com/3gpp

...

moiChanges

- notificationId: 123456789

op: replace

path: /ClassA=1#/attributes/attrA

value: 456

When the old value is reported as well, the notification looks like.

href: https://example.com/3gpp

...

moiChanges

- notificationId: 123456789

op: replace

path: /ClassA=1#/attributes/attrA

value: 456

oldValue: 123

The following example reports a value change for the complex attribute "attrB".

href: https://example.com/3gpp

...

moiChanges

- notificationId: 123456789

op: replace

path: /ClassA=1#/attributes/attrB

value:

subAttrB1: abc

subAttrB2: 78

The previous two notifications can be combined into a single notification as follows.

href: https://example.com/3gpp

...

moiChanges

- notificationId: 123456789

op: replace

path: /ClassA=1#/attributes/attrA

value: 456

- notificationId: 123456789

op: replace

path: /ClassA=1#/attributes/attrB

value:

subAttrB1: abc

subAttrB2: 78

Note the operation "replace" has replace semantics and not merge semantics. The following notification reports the value change of the attribute field "attrB:subAttrB1" to "def" and the deletion of the attribute field "attrB:subAttrB2".

href: https://example.com/3gpp

...

moiChanges

- notificationId: 123456789

op: replace

path: /ClassA=1#/attributes/attrB

value:

subAttrB1: def

The value change of the attribute field "attrA:subAttrB1" is reported as follows.

href: https://example.com/3gpp

...

moiChanges

- notificationId: 123456789

op: replace

path: /ClassA=1#/attributes/attrA/subAttrB1

value: def

Assume "attrD" is a JSON array with simple elements, then the creation of this multi-valued attribute is reported as follows.

href: https://example.com/3gpp

...

moiChanges

- notificationId: 123456789

op: add

path: /ClassA=1#/attributes/attrD

value:

- 1

- 2

- 3

Its deletion is reported by the following notification.

href: https://example.com/3gpp

...

moiChanges

- notificationId: 123456789

op: remove

path: /ClassA=1#/attributes/attrD

The complete replacement of the array is reported by the following notification.

href: https://example.com/3gpp

...

moiChanges

- notificationId: 123456789

op: add

path: /ClassA=1#/attributes/attrD

value:

- 11

- 21

- 31

The following example reports the second item in the array changed to "22".

href: https://example.com/3gpp

...

moiChanges

- notificationId: 123456789

op: replace

path: /ClassA=1#/attributes/attrD/1

value: 22

Note the array index of the second item is "1".

Assume now "attrE" is a JSON array with complex array items, for example.

[{subItemE1: 11, subItemD2: abc}, {subItemE1: 21, subItemE2: def}, {subItemE1: 31, subItemE2": ghi}.

A value change to

[{subItemE1: 11, subItemE2: abc}, {subItemE1: 21, subItemE2: xyz}, {subItemE1: 31, subItemE2": ghi}.

is reported by

href: https://example.com/3gpp

...

moiChanges

- notificationId: 123456789

op: replace

path: /ClassA=1#/attributes/attrE/1/subItemE2

value: xyz

When "subItemE2" is defined as array item key at stage 2, then "attrE" should contain a JSON map.

attrE:

11:

subItemE2: abc

21:

subItemE2: def

31:

subItemE2: ghi

The same change as above is now reported by the notification.

href: https://example.com/3gpp

...

moiChanges

- notificationId: 123456789

op: replace

path: /ClassA=1#/attributes/attrE/21/subItemD2

value: xyz

When all attributes of an object have been updated with a new value, the MnS producer may use a compact format reporting that the "attributes" container was updated completely.

|  |
| --- |
| POST /3gpp-management/cm-notification-sink HTTP/1.1  Host: example.com  Content-Type: application/json  {  "href": "http://example.com/3gpp",  "notificationId": 123456789,  "notificationType": "notifyMOICreation",  "eventTime": "Tue, 06 Aug 2019 16:50:26 GMT",  "systemDN":"DC=example.com,ManagedElement=ME1,MnsAgent=MA1",  "moiChanges": [  {  "notificationId": 123,  "op": "replace",  "path": "/ClassA=1#/attributes",  "value": {  "attrA": "newValueAttrA",  "attrB": "newValueAttrB"  }  }  ]  } |

Note that clause 4.3 of IETF RFC 6902 [13] does not consider it as an error if an attribute value is replaced with exactly the same value. For that reason, it would not be an error if in the example above an attribute value is included in the "value" property that did not change value. A MnS producer may consider this compact format hence also for the case that not all attributes of an object have been updated.

|  |
| --- |
| POST /3gpp-management/cm-notification-sink HTTP/1.1  Host: example.com  Content-Type: application/json  {  "href": "http://example.com/3gpp",  "notificationId": 123456789,  "notificationType": "notifyMOICreation",  "eventTime": "Tue, 06 Aug 2019 16:50:26 GMT",  "systemDN":"DC=example.com,ManagedElement=ME1,MnsAgent=MA1",  "moiChanges": [  {  "notificationId": 123,  "op": "replace",  "path": "/ClassA=1#/attributes",  "value": {  "attrA": "newValueAttrA",  "attrB": "oldValueAttrB"  }  }  ]  } |

To allow the MnS consumer to understand which attributes have been updated, the MnS producer may decide to send the following notification.

|  |
| --- |
| POST /3gpp-management/cm-notification-sink HTTP/1.1  Host: example.com  Content-Type: application/json  {  "href": "http://example.com/3gpp",  "notificationId": 123456789,  "notificationType": "notifyMOICreation",  "eventTime": "Tue, 06 Aug 2019 16:50:26 GMT",  "systemDN":"DC=example.com,ManagedElement=ME1,MnsAgent=MA1",  "moiChanges": [  {  "notificationId": 123,  "op": "replace",  "path": "/ClassA=1#/attributes",  "value": {  "attrA": "newValueAttrA",  "attrB": "oldValueAttrB"  },  "oldValue": {  "attrA": "oldValueAttrA",  "attrB": "oldValueAttrB"  }  ]  } |

###### 12.1.1.4.1a.5 Type NotifyMoiCreation

Table 12.1.1.4.1a.5 -1: Definition of type NotifyMoiCreation

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Description** | **S** |
| href | Uri | URI of the resource where the event (alarm) occurred | M |
| notificationId | NotificationId | Notification identifier as defined in ITU-T Rec. X. 733 [4] | M |
| notificationType | NotificationType | Notification type ("notifyMOICreation") | M |
| eventTime | DateTime | Event (MOI creation) occurrence time | M |
| systemDN | SystemDN | System DN | M |
| correlatedNotifications | array(CorrelatedNotification) | Set of all notifications to which this notification is considered to be correlated as defined in ITU-T Rec. X. 733 [4] | O |
| additionalText | string | Allows a free form text description to be reported as defined in ITU-T Rec. X. 733 [4] | O |
| sourceIndicator | SourceIndicator | Indicates the source of the operation that led to the generation of this notification. | O |
| attributeList | AttributeNameValuePairSet | The attributes (name/value pairs) of the created MOI. | O |

The following example shows a notification reporting the creation of an object with two attributes "attrA" and "attrB". Note that the notification includes the name/value pairs representing the attributes of the created object only and not the complete object representation.

|  |
| --- |
| POST /3gpp-management/cm-notification-sink HTTP/1.1  Host: example.com  Content-Type: application/json  {  "href": "http://example.com/ManagedElement=ME1/ClassA=CA1",  "notificationId": 123456789,  "notificationType": "notifyMOICreation",  "eventTime": "Tue, 06 Aug 2019 16:50:26 GMT",  "systemDN":"DC=example.com,ManagedElement=ME1,MnsAgent=MA1",  "attributeList":  {  "attrA": "valueAttrA",  "attrB": "valueAttrB"  }  } |

The creation of an empty object not containing any attribute values is reported as follows.

|  |
| --- |
| POST /3gpp-management/cm-notification-sink HTTP/1.1  Host: example.com  Content-Type: application/json  {  "href": "http://example.com/ManagedElement=ME1/ClassA=CA1",  "notificationId": 123456789,  "notificationType": "notifyMOICreation",  "eventTime": "Tue, 06 Aug 2019 16:50:26 GMT",  "systemDN":"DC=example.com,ManagedElement=ME1,MnsAgent=MA1",  "attributeList":  {  }  } |

###### 12.1.1.4.1a.6 Type NotifyMoiDeletion

Table 12.1.1.4.1a.6 -1: Definition of type NotifyMoiDeletion

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Description** | **S** |
| href | Uri | URI of the resource where the event (alarm) occurred | M |
| notificationId | NotificationId | Notification identifier as defined in ITU-T Rec. X. 733 [4] | M |
| notificationType | NotificationType | Notification type ("notifyMOIDeletion") | M |
| eventTime | DateTime | Event (MOI deletion) occurrence time | M |
| systemDN | SystemDN | System DN | M |
| correlatedNotifications | array(CorrelatedNotification) | Set of all notifications to which this notification is considered to be correlated as defined in ITU-T Rec. X. 733 [4] | O |
| additionalText | string | Allows a free form text description to be reported as defined in ITU-T Rec. X. 733 [4] | O |
| sourceIndicator | SourceIndicator | Indicates the source of the operation that led to the generation of this notification. | O |
| attributeList | AttributeNameValuePairSet | Attributes (name/value pairs) of the deleted MOI. | O |

The following example demonstrates the deletion of an object. The message body includes the name/value pairs representing the attributes of the deleted object.

|  |
| --- |
| POST /3gpp-management/cm-notification-sink HTTP/1.1  Host: example.com  Content-Type: application/json  {  "href": "http://example.com/ManagedElement=ME1/ClassA=CA1",  "notificationId": 123456789,  "notificationType": "notifyMOIDeletion",  "eventTime": "Tue, 06 Aug 2019 16:50:26 GMT",  "systemDN":"DC=example.com,ManagedElement=ME1,MnsAgent=MA1"  } |

The message body may include the name/value pairs representing the attributes of the deleted object.

|  |
| --- |
| POST /3gpp-management/cm-notification-sink HTTP/1.1  Host: example.com  Content-Type: application/json  {  "href": "http://example.com/ManagedElement=ME1/ClassA=CA1",  "notificationId": 123456789,  "notificationType": "notifyMOIDeletion",  "eventTime": "Tue, 06 Aug 2019 16:50:26 GMT",  "systemDN":"DC=example.com,ManagedElement=ME1,MnsAgent=MA1",  "attributeList":  {  "attrA": "valueAttrA",  "attrB": "valueAttrB"  }  } |

###### 12.1.1.4.1a.7 Type NotifyMoiAttributeValueChanges

Table 12.1.1.4.1a.7 -1: Definition of type NotifyMoiAttributeValueChanges

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Description** | **S** |
| href | Uri | URI of the resource where the event (alarm) occurred | M |
| notificationId | NotificationId | Notification identifier as defined in ITU-T Rec. X. 733 [4] | M |
| notificationType | NotificationType | Notification type ("notifyMOIAttributeValueChanges") | M |
| eventTime | DateTime | Event (MOI attribute value changes) occurrence time | M |
| systemDN | SystemDN | System DN | M |
| correlatedNotifications | array(CorrelatedNotification) | Set of all notifications to which this notification is considered to be correlated as defined in ITU-T Rec. X. 733 [4] | O |
| additionalText | string | Allows a free form text description to be reported as defined in ITU-T Rec. X. 733 [4] | O |
| sourceIndicator | SourceIndicator | Indicates the source of the operation that led to the generation of this notification. | O |
| attributeListValueChanges | AttributeValueChangeSet | List with names of changed attributes, together with new value and optionally old value | M |

The following example notification reports the modification of the attribute values for "attrA" and "attrB".

|  |
| --- |
| POST /3gpp-management/cm-notification-sink HTTP/1.1  Host: example.org  Content-Type: application/json  {  "href": "http://example.com/ManagedElement=ME1/ClassA=CA1",  "notificationId": 123456789,  "notificationType": "notifyMOIAttributeValueChanges",  "eventTime": "Tue, 06 Aug 2019 16:50:26 GMT",  "systemDN":"DC=example.com,ManagedElement=ME1,MnsAgent=MA1",  "attributeListValueChanges": [  {  "attrA": "newValueAttrA",  "attrB": "newValueAttrB"  }  ]  } |

In addition to the new values, the old values may be included in the notification.

|  |
| --- |
| POST /3gpp-management/cm-notification-sink HTTP/1.1  Host: example.org  Content-Type: application/json  {  "href": "http://example.com/ManagedElement=ME1/ClassA=CA1",  "notificationId": 123456789,  "notificationType": "notifyMOIAttributeValueChanges",  "eventTime": "Tue, 06 Aug 2019 16:50:26 GMT",  "systemDN":"DC=example.com,ManagedElement=ME1,MnsAgent=MA1",  "attributeListValueChanges": [  {  "attrA": "newValueAttrA",  "attrB": "newValueAttrB"  },  {  "attrA": "oldValueAttrA",  "attrB": "oldValueAttrB"  }  ]  } |

In the example above the attribute values are of simple type. For attributes of structured type, the question is if all attribute fields of an attribute, where at least one attribute field changed its value, need to be reported or only those attribute fields that changed value. Assume an attribute field that did not change value is not included in the notification. In this case the MnS consumer receiving the notification cannot tell if the attribute field was deleted or if the attribute field did not change value. It is not possible to distinguish these two cases based on the information in the notification. For that reason, always all attribute fields of an attribute need to be included in the notification. For structured attributes with many attribute fields this may not be very efficient.

In the next example "attrA" is a structured attribute with the attribute fields "attrFieldAA" and "attrFieldAB". The attribute field "attrFieldAA" changed value, the attribute field "attrFieldAB" did not change value. The attribute value change notification reporting this change may look as follows.

|  |
| --- |
| POST /3gpp-management/cm-notification-sink HTTP/1.1  Host: example.org  Content-Type: application/json  {  "href": "http://example.com/ManagedElement=ME1/ClassA=CA1",  "notificationId": 123456789,  "notificationType": "notifyMOIAttributeValueChanges",  "eventTime": "Tue, 06 Aug 2019 16:50:26 GMT",  "systemDN":"DC=example.com,ManagedElement=ME1,MnsAgent=MA1",  "attributeListValueChanges": [  {  "attrA": {  "attrFieldAA": "newValueAttrFieldAA",  "attrFieldAB": "oldValueAttrFieldAB"  }  },  {  "attrA": {  "attrFieldAA": "oldValueAttrFieldAA",  "attrFieldAB": "oldValueAttrFieldAB"  }  }  ]  } |

Note also that for multi-valued attributes all attribute elements of the new value need to be reported. It is not possible to report only added, deleted, or modified attribute elements. Furthermore, the notification does not allow reporting of deleted attributes. When this is required the MnS producer needs to include always all attributes of the object in the notification. It is outside the present document how the MnS producer signals to the MnS consumer if all attributes or only the changed ones are included in the attribute value change notification. The notification itself does not include this information.

###### 12.1.1.4.1a.8 Type NotifyMoiChanges

**Table 12.1.1.4.1a.8 -1: Definition of type NotifyMoiChanges**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Description** | **S** |
| href | Uri | URI of a common ancestor resource (object) of the resources for which changes are reported. A MnS producer may set this attribute always to the parent of the root resource in the MIB. | M |
| notificationId | NotificationId | Notification identifier as defined in ITU-T Rec. X. 733 [4]. | M |
| notificationType | NotificationType | Notification type (notifyMOIChanges) | M |
| eventTime | DateTime | Event (NRM updates) occurrence time | M |
| systemDN | SystemDN | System DN | M |
| moiChanges | array(MoiChange) | MOI changes to be reported | M |

###### 12.1.1.4.1a.9 Type PatchItem

Table 12.1.1.4.1a.9 -1: Definition of type PatchItem

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute name | Data type | Description | S |
| op | PatchOperation | Patch operation. | M |
| from | string | Present only for "copy" and "move" operations, identifies the value to be copied or moved to the location specified by path. | M |
| path | string | Path specifying the patched value. | M |
| value | any type | New value for the resource identified by "path". | M |

12.1.1.4.1a.10 Type NotifyMoiEvent

**Table 12.1.1.4.1a.10 -1: Definition of type NotifyEvent**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Description** | **S** |
| href | Uri | URI of the resource where the event (alarm) occurred | M |
| notificationId | NotificationId | Notification identifier as defined in ITU-T Rec. X. 733 [4] | M |
| notificationType | NotificationType | Notification type ("notifyEvent") | M |
| eventTime | DateTime | Date and time of the event | M |
| systemDN | SystemDN | It carries the DN of producer of the notification. | M |
| specificProblem | SpecificProblem | It indicates a problem detected | M |
| additionalText | string | It carries additional information. | O |
| additionalInformation | AttributeNameValuePairSet | It carries additional information. | O |

The following is an example of the notifyEvent notification.

|  |
| --- |
| POST /3gpp-management/cm-notification-sink HTTP/1.1  Host: myMns.mytelecom.com  Content-Type: application/json  {  "href": "http://myNode.com/ManagedElement=ME1 ",  "notificationId": 123456789,  "notificationType": "notifyEvent",  "eventTime": "Tue, 06 Aug 2019 16:50:26 GMT",  "systemDN":"DC=myNode.com,ManagedElement=ME1,MnsAgent=MA1",  "specificProblem": "Restart",  "additionalText": "Restart due to overheating",  "additionalInformation":  {  "temperature": "94.7",  "trendIndication": "MORE\_SEVERE"  }  } |

##### 12.1.1.4.2 Void

##### 12.1.1.4.3 Void

##### 12.1.1.4.4 Simple data types and enumerations

12.1.1.4.4.1 General

This clause defines simple data types and enumerations that are used by the data structures defined in the previous clauses.

12.1.1.4.4.2 Simple data types

Table 12.1.1.4.3.2-1: Simple data types

|  |  |  |
| --- | --- | --- |
| Type name | Type definition | Description |
| n/a | n/a | n/a |

12.1.1.4.4.3 Enumeration CmNotificationTypes

Table 12.1.1.4.4.3-1: Enumeration CmNotificationTypes

|  |  |
| --- | --- |
| Enumeration value | Description |
| notifyMOICreation | Notification type is notifyMOICreation |
| notifyMOIDeletion | Notification type is notifyMOIDeletion |
| notifyMOIAttributeValueChanges | Notification type is notifyMOIAttributeValueChange |
| noitifyMOIChanges | Notification type is notifyMOIChanges |
| notifyEvent | Notification type is notifyEvent |

12.1.1.4.4.4 Enumeration SourceIndicator

Table 12.1.1.4.4.4-1: Enumeration SourceIndicator

|  |  |
| --- | --- |
| Enumeration value | Description |
| RESOURCE\_OPERATION | The notification was generated in response to an internal operation of the resource. |
| MANAGEMENT\_OPERATION | The notification was generated in response to a management operation applied across the managed object boundary external to the managed object |
| SON\_OPERATION | The notification was generated as result of a SON (Self Organising Network) process like self-configuration, self-optimization, self-healing etc. . |
| UNKNOWN | It is not possible to determine the source of the operation. |

12.1.1.4.4.5 Enumeration ScopeType

Table 12.1.1.4.4.4.1-1: Enumeration ScopeType

|  |  |
| --- | --- |
| Enumeration value | Description |
| BASE\_ONLY | Selects only the base resource. The "scopeLevel" parameter shall be absent or ignored if present. |
| BASE\_ALL | Selects the base resource and all of its subordinate resources (incl. the leaf resources). The "scopeLevel" parameter shall be absent or ignored if present. |
| BASE\_NTH\_LEVEL | Selects all resources on the level, which is indicated by the "scopeLevel" parameter, below the base resource. The base resource is at "scopeLevel" zero. |
| BASE\_SUBTREE | Selects the base resource and all of its subordinate resources down to and including the resources on the level indicated by the "scopeLevel" parameter. The base resource is at "scopeLevel" zero. |

12.1.1.4.4.6 Enumeration Operation

Table 12.1.1.4.4.4.6-1: Enumeration Operation

|  |  |
| --- | --- |
| Enumeration value | Description |
| add | Create operation |
| remove | Delete operation |
| replace | Replace operation |

###### 12.1.1.4.4.7 Enumeration PatchOperation

Table 12.1.1.4.4.4.7-1: Enumeration PatchOperation

|  |  |
| --- | --- |
| Enumeration value | Description |
| add | Add operation |
| replace | Replace operation |
| remove | Remove operation |
| copy | Copy operation |
| move | Move operation |
| test | Test operation |

12.1.1.4.4.8 Enumeration Insert

Table 12.1.1.4.4.4.8-1: Enumeration Insert

|  |  |
| --- | --- |
| Enumeration value | Description |
| before | Specifies the new attribute element is inserted before the attribute element identified by the "path" attribute of "MoiChange". |
| after | Specifies the new attribute element is inserted after the attribute element identified by the "path" attribute of "MoiChange". |

### 12.1.2 RESTful HTTP-based solution set for integration with ONAP VES API

#### 12.1.2.1 Mapping of operations

NOTE: this mapping is not part of the present document.

#### 12.1.2.2 Mapping of notifications

##### 12.1.2.2.1 Introduction

###### 12.1.2.2.1.1 General

The 3GPP IS notifications are mapped to SS euivalents according to table 12.1.2.2.1.1-1.

Table 12.1.2.2.1.1-1: Mapping of 3GPP IS notifications to SS equivalents

|  |  |  |  |
| --- | --- | --- | --- |
| **3GPP IS notifications** | **HTTP Method** | **Resource URI** | **S** |
| notifyMOICreation | POST | /eventListener | M |
| notifyMOIDeletion | POST | /eventListener | M |
| notifyMOIAttributeValueChanges | POST | /eventListener | M |
| notifyMOIChanges | POST | /eventListener | M |
| notifyEvent | POST | /eventListener | M |

###### 12.1.2.2.1.2 Void

##### 12.1.2.2.2 Notification notifyMOICreation

See clause 12.1.1.2.2..

##### 12.1.2.2.3 Notification notifyMOIDeletion

See clause 12.1.1.2.3.

##### 12.1.2.2.4 Notification notifyMOIAttributeValueChange

See clause 12.1.1.2.4.

##### 12.1.2.2.5 Notification notifyMOIChanges

See clause 12.1.1.2.5.

##### 12.1.2.2.6 Notification notifyEvent

See clause 12.1.1.2.6.

#### 12.1.2.3 Resources

##### 12.1.2.3.1 Resource structure

Figure 12.1.2.3.1-1 shows the resource structure of the provisioning MnS in the context of its integration with VES Event Listener 7.1.1 [45].

A screen shot of a computer program

Description automatically generated with low confidence

Figure 12.1.2.3.1-1: Resource URI structure of the provisioning MnS for integration with ONAP VES Event Listener 7.1.1 (Resource structure section) [45]

Table 12.1.2.3.1-1 provides an overview of the resources and applicable HTTP methods.

Table 12.1.2.3.1-1: Resources and methods overview

|  |  |  |  |
| --- | --- | --- | --- |
| Resource name | Resource URI | HTTP method | Description |
| eventListener | /eventListener | POST | Send notifications |

##### 12.1.2.3.2 Resource definitions

See Resource structure section in [45].

#### 12.1.2.4 Data type definitions

See clause 12.1.1.4.

### 12.1.3 YANG/Netconf-based solution set

#### 12.1.3.1 Mapping of operations

##### 12.1.3.1.1 Introduction

The YANG/Netconf based solution set is based on the TS 32.160 [33] clause 6.2 and the IETF RFC 6241 [32] including the Xpath capability.

NOTE: The clauses below omit namespaces for brevity. In NETCONF operations namespaces are included following [34]

##### 12.1.3.1.2 Operation createMOI

The operation is mapped to a NETCONF <edit-config> operation, with XML elements representing the DN path to the MOI, the MOI itself, its id/key and its attributes.

The NETCONF operation attribute on the list representing the newly created MOI should be set to ‘create’.

The default-operation parameter of the <edit-config> operation should be set to none.

The IS operation parameters are mapped to SS equivalents according to table 12.1.3.1.2-1 and table 12.1.3.1.2-2.

Table 12.1.3.1.2-1: Mapping from IS createMOI input parameters to SS equivalents

|  |  |  |  |
| --- | --- | --- | --- |
| IS operation parameter name | SS parameter name | S | Remark |
| managedObjectClass | config | M | XML element’s name inside the <config> element. |
| managedObjectInstance | config | M | A sequence of embedded XML elements inside the <config> element. XML elements for all containing MOIs and their ids(keys) shall be included together wilt the XML elements representing the to be created MOI and its key. |
| attributeListIn | config | M | The key leaf, the “attributes container” and leaf, leaf-list or list entries of YANG models representing the attributes. |

Table 12.1.3.1.2-2: Mapping from IS createMOI output parameters to SS equivalents

|  |  |  |  |
| --- | --- | --- | --- |
| IS operation parameter name | SS parameter name | S | Remark |
| attributeListOut | no corresponding SS parameter | M | Not supported.  (note 1) |
| status | - | M | OperationSucceeded if NETCONF rpc-reply contains <ok> element.  OperationFailed if NETCONF-reply contains <rpc-error>. |

NOTE 1: Successful Netconf <edit-config> operations only return an <ok> element. Therefore, the attributeListOut can be retrieved via a separate <get-config> operation.

**Examples**

Create ManagedElement=myNode, GNBDUFunction=1

<rpc message-id="101">

<edit-config>

<target>

<running/>

</target>

<default-operation>none</default-operation>

<config>

< ManagedElement>

<id>myNode</id>

<GNBDUFunction operation=”create”>

<id>1</id>

<attributes>

<gNBIdLength>25</gNBIdLength>

<gNBId>357</gNBId>

<priorityLabel>1</priorityLabel>

<gNBDUName>du-south-1</gNBDUName>

<!-- other attributes --->

</attributes>

</GNBDUFunction>

</ManagedElement>

</config>

</edit-config>

</rpc>

<!-- createMO Response -->

<rpc-reply message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">

<ok/>

</rpc-reply>

##### 12.1.3.1.3 Operation getMOIAttributes

This IS operation is mapped to NETCONF <get> or <get-config> operation, depending on whether all configuration and state information is to be retrieved, or configuration data only. (In the next paragraphs only <get> operation is mentioned but <get-config> is always an alternative).

The IS operation patameters baseObjectInstance , (3GPP-)filter, scope, level, dataNodeSelector and attributeListIn are all combined and mapped into the Netconf-filter element. The scopes BASE\_ONLY and BASE\_ALL can be mapped to both subtree and Xpath filtering. The scopes BASE\_NTH\_LEVEL and BASE\_SUBTREE can only be mapped to Xpath filtering.

The IS operation parameters are mapped to SS equivalents according to table 12.1.3.1.3-1 and table 12.1.3.1.3-2.

Table 12.1.3.1.3-1: Mapping of IS getMOIAttributes input parameters to SS equivalents

|  |  |  |  |
| --- | --- | --- | --- |
| IS operation parameter name | SS parameter name | S | Remark |
| baseObjectInstance | filter | M | Initial part of the filter element.  For subtree filter this is a set of XML element representing lists containing MOIs together with the leafs representing key values for these MOIs from the root MOI (e.g. ManagedElement) to the baseObjectInstance.  For Xpath filter it is the initial parts of the Xpath expression representing the same information. |
| scope | filter | M | BASE\_ONLY and BASE\_ALL realized by the initial XML elements of the <get> operation. BASE\_SUBTREE and BASE\_NTH\_LEVEL is encoded in the Xpath filter. |
| level | filter | M | Included in the Xpath filter, see examples. (If level is used Xpath filtering must be used.  For BASE\_SUBTREE the levels number is transformed into a number of filter sub-expressions joined by the OR operator.  For BASE\_NTH\_LEVEL included in the Xpath expression as a sequence of ‘\*’ parts (descendant axis) The number of ‘\*’ correspond to the number of levels. |
| filter | filter | M | Netconf Subtree or Xpath filter |
| attributeListIn | filter | M | add the attributes to the subtree or Xpath filter |
| dataNodeSelector | filter | M | Included in the Xpath filter |

Table 12.1.3.1.3-2: Mapping of IS getMOIAttributes output parameters to SS equivalents

|  |  |  |  |
| --- | --- | --- | --- |
| IS operation parameter name | SS parameter name | S | Remark |
| managedObjectClass | data | M | Can be extracted from the NETCONF <rpc-reply> <data> elements |
| managedObjectInstance | data | M | Can be extracted from the NETCONF <rpc-reply> <data> elements |
| attributeListOut | data | M | Can be extracted from the NETCONF <rpc-reply> <data> elements |
| status | data | M | rpc-reply or rpc-error indicates general status. |

If scope is ***BASE\_ONLY*** the <get> shall be directed against the “attributes” container of the baseObjectInstance.

**Example 1**

A getMOIAttributes for base object ManagedElement=myNode, scope = BASE\_ONLY, filter=none, attributesListIn=empty is mapped into the following <get-config> operation -

<rpc message-id="101"

xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">

<get-config>

<source>

<running/>

</source>

<filter type="subtree">

<ManagedElement>

<id>myNode</id>

<attributes/>

</ManagedElement>

</filter>

</get-config>

</rpc>

If scope is ***BASE\_ALL*** the <get> shall be directed against the list representing the baseObjectInstance.

**Example 2**

A getMOIAttributes for base object ManagedElement=myNode, scope = BASE\_ALL, filter=, MeasurementControl.pMAdministrativeState=UNLOCKED, attributesListIn=empty.

<rpc message-id="101"

xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">

<get>

<source>

<running/>

</source>

<filter type="subtree">

<ManagedElement>

<id>myNode</id>

<MeasurementControl>

<pMAdministrativeState>

UNLOCKED

</pMAdministrativeState>

</MeasurementControl>

</ManagedElement>

</filter>

</get>

</rpc>

If scope is ***BASE\_SUBTREE*** the <get> shall be directed against the list representing the baseObjectInstance. The Xpath filter expression will need a sub-expression for each level joined by the OR operator.

**Example 3**

A getMOIAttributes for base object ManagedElement=me1, scope = BASE\_ SUBTREE, level=2, filter=none, attributesListIn=empty.

<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="101">

<get>

<source>

<running/>

</source>

<filter type="xpath"

select="/me3gpp:ManagedElement[id='me1']/attributes |

/me3gpp:ManagedElement[id='me1']/\*/attributes |

/me3gpp:ManagedElement[id='me1']/\*/\*/attributes" />

</get>

</rpc>

If scope is ***BASE\_NTH\_LEVEL*** the <get> shall be directed against the list representing classes at the N*th* level under the baseObjectInstance. The number of ‘\*’ parts (descendant axis) will correspond to the number of levels.

**Example 4**

A getMOIAttributes for base object ManagedElement=myNode, scope = BASE\_NTH\_LEVEL, level=2, filter=none, attributesListIn=empty.

<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="101">

<get>

<source>

<running/>

</source>

<filter type="xpath"

select="/me3gpp:ManagedElement[id='me1']/\*/\*/attributes"/>

</get>

</rpc>

##### 12.1.3.1.4 Operation modifyMOIAttributes

This IS operation modifies one or multiple managed object instances. It is mapped to the NETCONF <edit-config> operation. The NETCONF <edit-config> operation can modify attributes in a given MOI or set of MOIs but only indirectly supports scope or filtered sets of MOIs that are part of the modifyMOIAttributes 3GPP operation specification. <edit-config> needs a config block, containing the explicit config changes to be made for each MOI.

The default-operation parameter should be set to none.

The Netconf operation attribute on the list representing modified MOI(s) should be set to create, replace or delete according to the ENUM in the modificationList.

The IS operation parameters are mapped to SS equivalents according to table 12.1.3.1.4-1 and table 12.1.3.1.4-2.

Table 12.1.3.1.4-1: Mapping of IS modifyMOIAttributes input parameters to SS equivalents

|  |  |  |  |
| --- | --- | --- | --- |
| IS operation parameter name | SS parameter name | S | Remark |
| baseObjectInstance | config | M | A sequence of embedded XML elements inside the <config> element. XML elements for all containing MOIs and their ids(keys) shall be included together with the XML elements representing the to be modified MOI and its key. |
| scope | config | M | BASE\_ONLY supported as default. Multiple MOIs can be specified in the same operation, emulating other scopes. |
| filter | config | M | Multiple MOIs can be specified in the same operation, emulating filtering. |
| modificationList | config | M | The “attributes container” and leaf, leaf-list or list entries representing the attributes. |

Table 12.1.3.1.4-2: Mapping of IS modifyMOIAttributes output parameters to SS equivalents

|  |  |  |  |
| --- | --- | --- | --- |
| **IS operation parameter name** | **SS parameter name** | **S** | **Remark** |
| modificationListOut | no corresponding SS parameter | M | Not supported.  (note 1) |
| status | - | M | rpc-reply or rpc-error indicates general status.  The following elements give detailed error information:  <error-tag>  <error-path> |

Note 1: Successful Netconf <edit-config> operations only return an <ok> element. Therefore, the attributeListOut can be retrieved via a separate <get-config> operation.

##### 12.1.3.1.4a Operation changeMOIs

The operation is mapped to a NETCONF <edit-config> operation, with XML elements representing the DN path to the MOI and any attributes or attribute fields.

The default-operation parameter of the <edit-config> operation should be set to none.

The IS operation parameters are mapped to SS equivalents according to table 12.1.3.1.4a-1 and table 12.1.3.1.4a-2.

The detailed semantics is specified by the Netconf protocol and the related YANG models.

Table 12.1.3.1.4a-1: Mapping from IS changeMOIs input parameters to SS equivalents

|  |  |  |  |
| --- | --- | --- | --- |
| IS operation parameter name | SS parameter name | S | Remark |
| baseObjectInstance | config | M | A sequence of embedded XML elements inside the <config> element. XML elements for all containing MOIs and their ids(keys) shall be included together wilt the XML elements representing the to be created MOI and its key. |
| modificationsIn | config | M | Path and nodeValue are represented by XML elements inside the <config> element.  modifyOperator is represented by the Netconf operation parameter. |

Table 12.1.3.1.4a-2: Mapping from IS changeMOIs output parameters to SS equivalents

|  |  |  |  |
| --- | --- | --- | --- |
| IS operation parameter name | SS parameter name | S | Remark |
| attributeListOut | - | O | Not supported.  (note 1) |
| status | - | M | SUCCEEDED if NETCONF rpc-reply contains an <ok> element.  FAILED if NETCONF-reply contains an <rpc-error>.  Support for PARTIALLY\_FAILED depends on the NETCONF error-option. It is recommended to always use the error-option=rollback-on-error as semantics for stop-on-error, and continue-on-error is not well defined. |

NOTE 1: The attributeListOut can be retrieved via a separate <get-config> operation.

##### 12.1.3.1.5 Operation deleteMOI

This IS operation deletes one or multiple managed object instances. It is mapped to the NETCONF <edit-config> operation. <edit-config> can delete one or more specific MOIs but only indirectly supports scope or filtered sets of MOIs that are part of the generic deleteMOI 3GPP operation specification. <edit-config> uses a config block, indicating the MOI(s) to be deleted.

The Netconf operation attribute on the list representing the baseObjectInstance should be set to delete or remove.

The default-operation parameter should be set to none.

The IS operation parameters are mapped to SS equivalents according to table 12.1.3.1.5-1 and table 12.1.3.1.5-2.

Table 12.1.3.1.5-1: Mapping of IS deleteMOI input parameters to SS equivalents

|  |  |  |  |
| --- | --- | --- | --- |
| **IS operation parameter name** | **SS parameter name** | **S** | **Remark** |
| baseObjectInstance | config | M | A sequence of embedded XML elements inside the <config> element. XML elements for all containing MOIs and their ids(keys) shall be included together wilt the XML elements representing the to be deleted MOI and its key. |
| scope | config | M | BASE\_ONLY supported as default. Multiple MOIs can be specified in the same operation, emulating other scopes. |
| filter | config | M | Multiple MOIs can be specified in the same operation, emulating filtering. |

Table 12.1.3.1.5-2: Mapping of IS deleteMOI output parameters to SS equivalents

|  |  |  |  |
| --- | --- | --- | --- |
| **IS operation parameter name** | **SS parameter name** | **S** | **Remark** |
| deletionList | no corresponding SS parameter | M | Not supported.  (note 1) |
| status | - | M | rpc-reply or rpc-error indicates general status.  The following elements give detailed error information:  <error-tag>  <error-path> |

NOTE 1: Successful Netconf <edit-config> operations only return an <ok> element. Therefore, the deletionList can be retrieved via a separate <get-config> operation.

#### 12.1.3.2 Mapping of notifications

##### 12.1.3.2.1 Introduction

The notifications "notifyMOICreation", "notifyMOIDeletion" and "notifyMOIAttributeValueChanges" should not be used in the YANG\_Netconf solution set as "notifyMOIChanges" provides the same functionality.

12.1.3.2.2 Notification notifyMOICreation

The notification is not mapped to the NETCONF/YANG solution.

12.1.3.2.3 Notification notifyMOIDeletion

The notification is not mapped to the NETCONF/YANG solution.

12.1.3.2.4 Notification notifyMOIAttributeValueChange

The notification is not mapped to the NETCONF/YANG solution.

##### 12.1.3.2.5 Notification notifyMOIChanges

The NETCONF/YANG solution set uses the same mapping as the RESTful HTTP-based solution set as described in clause 12.1.1.2.5 with the changes and additions described below.

- Any changes reported are based on the YANG NRM definitions, even though the RESTful notification mapping is reused.

- The media type as specified by the "Content-Type" header in the HTTP POST request shall be "application/yang-data+json". If the ONAP VES API integration is used the "Content-Type" shall be set to *application/json as dictated by the VES specification[45].*

- The value of "href" shall be set to the FQDN or IP address identifying the NETCONF server.

- The value of "path" shall be a RESTCONF data resource identifier (RFC 8040 [49], clause 3.5.3).

- The "path" includes the YANG module name.

- The "#" character before "/attributes" in "path" is not present. NETCONF/YANG does not differentiate between the stage 2 concepts of object and attribute, hence there is no need for a delimiter.

- The value of "value" shall follow the JSON encoding of YANG (RFC 7951 [50]).

- Attribute elements are identified by their value (in case of a YANG "leaf-list") or by the values of keys (in case of a YANG "list"). In JSON Patch, attribute elements are identified based on their index, i.e. based on the position in the array.

- In case no key is defined for a YANG "list" it is not possible to report the creation, deletion or replacement of individual list entries. In this case, whenever the list is modified, the replacement of the complete attribute or attribute field (the complete list with all list enties) shall be reported.

- Similarly if an attribute(field) is mapped to a YANG leaf-list with non-unique values it is not possible to report the creation, deletion or replacement of an individual value. In this case, whenever the leaf-list is modified, the replacement of the complete attribute or attribute field (the complete leaf-list; all values) shall be reported.

- YANG default values shall be handled as follows:

- Attributes with default values, for which no value is specified in the object creation request, shall be included in the object creation report with their default values.

- Attributes with default values, that are deleted and consequently set to their default value, shall be included in attribute replacement reports.

Note all following use-cases use JSON expressed in YAML notation.

Case 1: Creation of an MOI is reported with:

- operation: add

- path: YANG resource identifier pointing to the list entry representing the MOI

- value: a complete MOI representation, represented by the "id" node and the "attributes" container but exluding the list entry itself encoded according to RFC7951 [50].

For example, the following instance of a "moiChanges" array item reports an object creation:

href: node1.lichtenberg.de

…

notificationId: 123456001

path: "/\_3gpp-common-managed-element:ManagedElement=node3/PerfMetricJob=job1"

operation: add

value:

id: job1

attributes:

jobId: 9865

fileReportingPeriod: 30

Case 2: Deletion of an MOI is reported with:

- operation: remove

- path: YANG resource identifier pointing to the list entry representing the MOI

- value: not present

For example, the following instance of a "moiChanges" array item reports an object deletion:

href: node1.charlottenburg.de

…

notificationId: 123456002

path: "/\_3gpp-common-managed-element:ManagedElement=node3/PerfMetricJob=job1"

operation: remove

Case 3: Creating a (complete) attribute is reported as follows. (Setting the value(s) of an attribute that had no value before the change):

- operation: add.

- path: YANG resource identifier pointing to the leaf/leaf-list/container/list representing the attribute. If the attribute is represented by a list or leaf-list, then for this last data node the equal sign, the key value(s) or leaf-list value is omitted, only the list/leaf-list name shall be present.

- value: the content of the leaf/leaf-list entry(s)/container/list entry(s) representing the created attribute encoded according to RFC7951 [50]. In case of attribute represented by a container/list the child data nodes are encoded only,the container/list itself is not.

For example, the following instance of a "moiChanges" array item reports setting the values of the performanceMetrics simple, multivalue attribute:

href: node1.spandau.de

…

notificationId: 123456003

path: "/\_3gpp-common-managed-element:ManagedElement=node3/PerfMetricJob=job1/attributes/

performanceMetrics"

operation: add

value:

- inOctets

- inPackets

- outPackets

Case 4: Deleting all values of a complete attribute is reported with

- operation: remove.

- path: Same as in case 3.

- value: not present.

For example, the following instance of a "moiChanges" array item reports deleting all values of the performanceMetrics attribute:

href: node1.pankow.de

…

notificationId: 123456004

path: "/\_3gpp-common-managed-element:ManagedElement=node3/PerfMetricJob=job1/attributes/

performanceMetrics"

operation: remove

Case 5: Replacing a (complete) attribute is reported as follows. (Removing all previous values of the attribute and setting new value(s)):

- operation: replace.

- path: Same as in case 3.

- value: Same as in case 3.

Case 6: Adding a new value to a multivalue attribute (an attribute with multiplicity upper bound greater than 1) is reported as follows. (This does not imply any change to exisiting values):

- operation: add/

- path: YANG resource identifier pointing to a leaf-list/list entry representing an attribute element(value). In case of adding a new element to an attribute with the property isOrdered=True the new element/value is inserted before the pointed element(value), unless the "insert" subparameter specifies differently.

- value: the leaf-list/list entry representing the new attribute value encoded according to RFC7951 [50]. In case of a list the child data nodes are encoded the list-entry itself is not.

- insert: an additional input subparameter is added to the moiChange input parameter. This indicates whether the new element/value was added before or after the element/value specified in path. The subparameter is only valid in case of attributes with the property isOrdered=True. It can take the values "before", "after". If missing it defaults to "before".

For example, the following instance of a "moiChanges" array item reports adding a new element/value to the "performanceMetrics" attribute before the outPackets element.:

notificationId: 123456006

path: "/\_3gpp-common-managed-element:ManagedElement=node3/PerfMetricJob=job1/attributes/

performanceMetrics/performanceMetrics=outPackets"

operation: add

insert: before

value: outOctets

Case 7: Deleting a single element/value from a multivalue attribute is reported as follows. (This does not imply any change to any other elements):

- operation: remove.

- path: Same as case 6.

- value: not present.

Case 8: Replacement of a single value for a multivalue attribute is reported as follows. This implies removing the old value; in case of a structured attribute removal all its subparts. This does not imply any change to any other values:

- operation: replace.

- path: Same as case 6.

- value: Same as case 6.

For example, the following instance of a "moiChanges" array item reports replacing an element/value of the "thresholdInfoList" structured attribute:

notificationId: 123456008

path: /\_3gpp-common-managed-element:ManagedElement=node3/ThresholdMonitor=job1/attributes/

thresholdInfoList=thr1

operation: replace

value:

- idx: thr1

thresholdDirection: UP

thresholdValue: '4.5'

Case 9: Adding afield (subpart) of an attribute value is reported as follows (only used for structured attributes represented by a list or container in YANG):

- operation: add.

- path: YANG Resource Identifier pointing to the leaf/leaf-list/container/list representing the attribute field. If the attribute field is represented by a list or leaf-list, the field has multiplicity upper bound greater than 1, with the property isOrdered=True the new element/value is inserted before the pointed element(value), unless the "insert" subparameter specifies differently.

- value: the leaf/leaf-list/container/list representing the new attribute field values encoded according to RFC7951. In case of a list/container representing the attribute field, value shall contain only the child data nodes, but not the container/list-entry itself.

- insert: In case the field has multiplicity upper bound greater than 1 and has the property isOrdered=True, the subparameter is used similarly as in case 6.

For example, the following instance of a "moiChanges" array item reports adding a value to the " hysteresis " attribute subpart:

notificationId: 123456009

path: /\_3gpp-common-managed-element:ManagedElement=node3/ThresholdMonitor=job1/attributes/

thresholdInfoList=thr1/hysteresis

operation: add

value: '10'

Case 10: Deleting a field (subpart) of an attribute is reported as follows. (only used for structured attributes represented by a list or container in YANG):

- operation: remove.

- path: Same as case 9.

- value: Not present.

For example, the following instance of a "moiChanges" array item reports deleting all values of the "hysteresis" attribute field:

notificationId: 123456010

path: /\_3gpp-common-managed-element:ManagedElement=node3/ThresholdMonitor=job1/attributes/

thresholdInfoList=thr1/hysteresis

operation: remove

Case 11: Replacement of a field (subpart) of an attribute is reported as follows. This implies removing previous value(s). (only used for structured attributes represented by a list or container in YANG):

- operation: replace.

- path: Same as case 9.

- value: Same as case 9.

12.1.3.2.6 Notification notifyEvent

The NETCONF/YANG solution set uses the same mapping as the RESTful HTTP-based solution set. See clause 12.1.1.2.6.

#### 12.1.3.3 Netconf Server behavior

##### 12.1.3.3.1 Introduction

The Netconf server implementing the MnS provider shall implement some basic capabilities.

##### 12.1.3.3.2 Implement IETF RFC 6243: “With-defaults Capability for NETCONF”

The Netconf server (MnS producer) shall be compliant to RFC 6243[51] and implement the Netconf urn:ietf:params:netconf:capability:with-defaults:1.0 capability. The ‘report-all’ retrieval mode shall be supported. Other retrieval modes may be supported. The basic mode should be ‘report-all’.

## 12.2 Void

## 12.3 Generic performance assurance management service

### 12.3.1 RESTful HTTP-based solution set

#### 12.3.1.1 Void

#### 12.3.1.2 Performance threshold monitoring service

##### 12.3.1.2.1 Mapping of operations

None.

##### 12.3.1.2.2 Mapping of notifications

###### 12.3.1.2.2.1 Introduction

The IS notifications are mapped to SS equivalents according to table 12.3.1.2.2.1-1.

**Table 12.3.1.2.2.1-1: Mapping of IS notifications to SS equivalents**

|  |  |  |  |
| --- | --- | --- | --- |
| **IS notifications** | **HTTP Method** | **Resource URI** | **S** |
| notifyThresholdCrossing | POST | /notificationSink | M |

###### 12.3.1.2.2.2 Notification notifyThresholdCrossing

The IS notification parameters are mapped to SS equivalents according to table 12.3.1.2.2.2-1.

Table 12.3.1.2.2.2-1: Mapping of IS notification input parameters to SS equivalents (HTTP POST)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| objectClass | request body | href | Uri | M |
| objectInstance |
| notificationId | request body | notificationId | NotificationId | M |
| notificationType | request body | notificationType | NotificationType | M |
| eventTime | request body | eventTime | DateTime | M |
| systemDN | request body | systemDN | SystemDN | M |
| observedPerfMetricName | request body | observedPerfMetricName | string | M |
| observedPerfMetricValue | request body | observedPerfMetricValue | PerfMetricValue | M |
| observedPerfMetricDirection | request body | observedPerfMetricDirection | PerfMetricDirection | M |
| thresholdValue | request body | thresholdValue | PerfMetricValue | M |
| hysteresis | request body | hysteresis | PerfMetricValue) | M |
| monitorGranularityPeriod | request body | monitorGranularityPeriod | integer | M |
| additionalText | request body | additionalText | string | O |

##### 12.3.1.2.3 Resources

###### 12.3.1.2.3.1 Resource structure

Table 12.3.1.2.3.1-1 provides an overview of the resources and applicable HTTP methods.

Table 12.3.1.2.3.1-1: Resources and methods overview

|  |  |  |  |
| --- | --- | --- | --- |
| Resource name | Resource URI | HTTP method | Description |
| notificationSink | /notificationSink | POST | Send notifications |

###### 12.3.1.2.3.2 Resource definitions

12.3.1.2.3.2.1 Resource "/notificationSink"

12.3.1.2.3.2.1.1 Description

This resource represents a resource on a MnS consumer to which notifications are sent to.

12.3.1.2.3.2.1.2 URI

The resource URI is provided by the notification subscription.

12.3.1.2.3.2.1.3 HTTP methods

12.3.1.2.3.2.1.3.1 POST

This method shall support the URI query parameters specified in table 12.3.1.2.3.2.1.3.1-1.

Table 12.3.1.2.3.2.1.3.1-1: URI query parameters supported by the POST method on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data type | Description | S |
| n/a | n/a | n/a | n/a |

This method shall support the request data structures specified in table 12.3.1.2.3.2.1.3.1-2 and the response data structures and response codes specified in table 12.3.1.2.3.2.1.3.1-3.

Table 12.3.1.2.3.2.1.3.1-2: Data structures supported by the POST Request Body on this resource

|  |  |  |
| --- | --- | --- |
| Data type | Description | S |
| NotifyThresholdCrossing | Type in case a notifyThresholdCrossing notification is sent | M |

Table 12.3.1.2.3.2.1.3.1-3: Data structures supported by the POST Response Body on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| Data type | Response  codes | Description | S |
| n/a | 204 No Content | In case of success no message body is returned | M |
| Error-Response | 4xx/5xx | In case of failure the error object is returned. | M |

##### 12.3.1.2.4 Data type definitions

###### 12.3.1.2.4.1 General

Table 12.3.1.2.4.1-1: Data types defined in this specification

|  |  |  |
| --- | --- | --- |
| **Data type** | **Reference** | **Description** |
| NotifyThresholdCrossing | 12.3.1.2.4.2.1 | Used in the request body of HTTP POST for the notification type notifyThresholdCrossing |
| PerfNotificationTypes | 12.3.1.2.4.6.4 | Performance notification types (notifyThresholdCrossing) |

Table 12.3.1.1.4.1-2: Data types imported

|  |  |  |
| --- | --- | --- |
| **Data type** | **Reference** | **Description** |
| DateTime | TS 28.623 [44] | Date and time |
| Float | TS 28.623 [44] | Float type |
| Uri | TS 28.623 [44] | URI type |
| SystemDN | TS 28.623 [44] | systemDN type |
| NotificationId | TS 28.623 [44] | Notification identifier as defined in ITU-T Rec. X. 733 [4] |
| NotificationHeader | TS 28.623 [44] | Notification header |
| ErrorResponse | TS 28.623 [44] | Used in the response body of multiple HTTP methods in case of error |

###### 12.3.1.2.4.2 Structured data types

12.3.1.2.4.2.1 Type NotifyThresholdCrossing

Table 12.3.1.2.4.2.1-1: Definition of type NotifyThresholdCrossing

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Description** | **S** |
| href | Uri | URI of the resource where the event (threshold crossing) occurred | M |
| notificationId | NotificationId | Notification identifier as defined in ITU-T Rec. X. 733 [4] | M |
| notificationType | NotificationType | Notification type (notifyThresholdCrossing) | M |
| eventTime | DateTime | Event (threshold crossing) occurrence time | M |
| systemDN | SystemDN | System DN | M |
| observedPerfMetricName | string | Name of the performance metric that has crossed the threshold | M |
| observedPerfMetricValue | PerfMetricValue | Value of the performance metric, that has crossed the threshold, when the threshold crossing was observed | M |
| observedPerfMetricDirection | PerfMetricDirection | Direction ("UP" or "DOWN") of the performance metric, when the threshold crossing was observed | M |
| thresholdValue | PerfMetricValue | Threshold value of the triggered threshold | M |
| hysteresis | PerfMetricValue | Hysteresis of the triggered threshold | M |
| monitorGranularityPeriod | integer | Granularity period of the threshold monitor | M |
| additionalText | string | Vendor specific information | O |

12.3.1.2.4.3 Void

###### 12.3.1.2.4.4 Void

###### 12.3.1.2.4.5 Void

###### 12.3.1.2.4.6 Simple data types and enumerations

12.3.1.2.4.6.1 General

This clause defines simple data types and enumerations that are used by the data structures defined in the previous clauses.

12.3.1.2.4.6.2 Simple data types

Table 12.3.1.2.4.6.2-1: Simple data types

|  |  |  |
| --- | --- | --- |
| Type name | Type definition | Description |
| PerfMetricValue | Union(integer, Float) | The type of a performance metric is either integer or Float |

12.3.1.2.4.6.3 Enumeration PerfNotificationTypes

Table 12.3.1.2.4.6.3-1: Enumeration PerfNotificationTypes

|  |  |
| --- | --- |
| Enumeration value | Description |
| notifyThresholdCrossing | Notification type is notifyThresholdCrossing |

12.3.1.2.4.6.4 Enumeration PerfMetricDirection

Table 12.3.1.2.4.6.4-1: Enumeration PerfMetricDirection

|  |  |
| --- | --- |
| Enumeration value | Description |
| UP | Performance metric values are going up |
| DOWN | Performance metric values are going down |

### 12.3.2 Performance data XML file format definition

#### 12.3.2.1 Introduction

This clause describes the format of performance data file. The XML file format definition is based on XML schema ([26], [27], [28] and [29]).

#### 12.3.2.2 Mapping table

Table 12.3.2.2-1 maps the file content items in the clause 11.3.2.1.2 to those used in the XML schema based file format definitions. XML attributes are useful where data values bind tightly to its parent XML element. They have been used where appropriate.

Table 12.3.2.2-1: Mapping of File Content Items to XML tags

| File Content Item | XML schema based XML tag | Description |
| --- | --- | --- |
| measDataFile | XML element: measDataFile | Document element |
| measFileHeader | XML element: fileHeader |  |
| measData | XML element: measData |  |
| measFileFooter | XML element: fileFooter |  |
| fileFormatVersion | XML element: fileHeader  XML attribute: fileFormatVersion |  |
| senderName | XML element: fileHeader  XML attribute: dnPrefix  XML element: fileHeader:fileSender  XML attribute: senderName | The DN of the sender is split into the DN prefix contained in "dnPrefix" and the Local DN (LDN) contained in "senderName". |
| senderType | XML element fileHeader:fileSender  XML attribute: senderType |  |
| vendorName | XML element fileHeader  XML attribute vendorName |  |
| collectionBeginTime | XML element: fileHeader:measData  XML attribute beginTime |  |
| measObjRootDn | XML element fileHeader  XML attribute dnPrefix  XML element measData:measEntity  XML attribute localDn | The DN of the root object is split into the DN prefix contained in "dnPrefix" and the Local DN (LDN) contained in "localDn". |
| measObjRootUserLabel | XML element: measData:measEntity  XML attribute: userLabel |  |
| measObjRootSwVersion | XML element: measData:measEntity  XML attribute: swVersion |  |
| measInfo | XML element measInfo | An instance of this XML element is added for each expired granularity period. |
| measInfoId | XML element measData:measInfo  XML attribute measInfoId |  |
| jobId | XML element measData:measInfo:job  XML attribute jobId |  |
| reportingPeriod | XML element measData:measInfo:repPeriod  XML attribute duration | The XML attribute "duration" shall use the truncated representation for duration "PT*n*S" (see [28]). |
| granularityPeriod | XML element measData:measInfo:granPeriod  XML attribute duration | The XML attribute "duration" shall use the truncated representation for duration "PT*n*S" (see [28]). |
| measTimeStamp | XML element measData:measInfo:granPeriod  XML attribute endTime |  |
| measTypes | XML element measData:measInfo:measTypes  or  XML element measData:measInfo:measType  XML attribute p | Depending on sender's choice for optional positioning presence, either XML element "measTypes" or XML elements "measType" will be used. |
| measValues | XML element measData:measInfo:measValue |  |
| measObjLdn | XML element measData:measInfo:measValue  XML attribute measObjLdn |  |
| measResults | XML element measData:measInfo:measValue:measResults  or, when the positioning option is used,  measData:measInfo:measValue:r | Depending on sender's choice for optional positioning, either XML element "measResults" or XML elements "r" is used. |
| suspectFlag | XML element measData:measInfo:measValue:suspect |  |
| collectionEndTime | XML element fileFooter:measData  XML attribute endTime |  |
| There is no corresponding File Content Item. | XML element measType  XML attribute p | Only for the positioning option: XML attribute "p" of XML element "measType", used to link the performance metric type specified in "measType" to the result value. Its value is a positive integer (excl. zero) and shall be unique for each instance of "measType" in a file. |
| There is no corresponding File Content Item. | XML element r  XML attribute p | Only for the positioning option: XML attribute "p" of the XML element "r", used to link the result value in "r" to its performance metric type in "measType". The value of "p" shall match the value of the XML attribute "p" in the corresponding XML element "measType". |

#### 12.3.2.3 Void

##### 12.3.2.3.1 Void

##### 12.3.2.3.2 Void

#### 12.3.2.4 XML schema

This clause specifies the XML schema that shall be used for XML files containing performance data.

Name: measData.xsd

Version: 2.0.0

Identifier: measData.xsd-v2.0.0

<?xml version="1.0" encoding="UTF-8"?>

<!--

TS 28.532 Performance data XML file format definition

measData.xsd-v2.0.0

-->

<schema

xmlns="http://www.w3.org/2001/XMLSchema"

xmlns:md="http://www.3gpp.org/ftp/specs/archive/28\_series/28.532#measData"

targetNamespace="http://www.3gpp.org/ftp/specs/archive/28\_series/28.532#measData"

elementFormDefault="qualified">

<element name="measDataFile">

<complexType>

<sequence>

<element name="fileHeader">

<complexType>

<sequence>

<element name="fileSender">

<complexType>

<attribute name="senderName" type="string" use="optional"/>

<attribute name="senderType" type="string" use="optional"/>

</complexType>

</element>

<element name="measData">

<complexType>

<attribute name="beginTime" type="dateTime" use="required"/>

</complexType>

</element>

</sequence>

<attribute name="fileFormatVersion" type="string" use="required"/>

<attribute name="vendorName" type="string" use="optional"/>

<attribute name="dnPrefix" type="string" use="optional"/>

</complexType>

</element>

<element name="measData" minOccurs="0" maxOccurs="unbounded">

<complexType>

<sequence>

<element name="measEntity">

<complexType>

<attribute name="localDn" type="string" use="optional"/>

<attribute name="userLabel" type="string" use="optional"/>

<attribute name="swVersion" type="string" use="optional"/>

</complexType>

</element>

<element name="measInfo" minOccurs="0" maxOccurs="unbounded">

<complexType>

<sequence>

<element name="job" minOccurs="0">

<complexType>

<attribute name="jobId" type="string" use="required"/>

</complexType>

</element>

<element name="granPeriod">

<complexType>

<attribute name="duration" type="duration" use="required"/>

<attribute name="endTime" type="dateTime" use="required"/>

</complexType>

</element>

<element name="repPeriod" minOccurs="0">

<complexType>

<attribute name="duration" type="duration" use="required"/>

</complexType>

</element>

<choice>

<element name="measTypes">

<simpleType>

<list itemType="Name"/>

</simpleType>

</element>

<element name="measType" minOccurs="0" maxOccurs="unbounded">

<complexType>

<simpleContent>

<extension base="Name">

<attribute name="p" type="positiveInteger" use="required"/>

</extension>

</simpleContent>

</complexType>

</element>

</choice>

<element name="measValue" minOccurs="0" maxOccurs="unbounded">

<complexType>

<sequence>

<choice>

<element name="measResults">

<simpleType>

<list itemType="md:measResultType"/>

</simpleType>

</element>

<element name="r" minOccurs="0" maxOccurs="unbounded">

<complexType>

<simpleContent>

<extension base="md:measResultType">

<attribute name="p" type="positiveInteger" use="required"/>

</extension>

</simpleContent>

</complexType>

</element>

</choice>

<element name="suspect" type="boolean" minOccurs="0"/>

</sequence>

<attribute name="measObjLdn" type="string" use="required"/>

</complexType>

</element>

</sequence>

<attribute name="measInfoId" type="string" use="optional"/>

</complexType>

</element>

</sequence>

</complexType>

</element>

<element name="fileFooter">

<complexType>

<sequence>

<element name="measData">

<complexType>

<attribute name="endTime" type="dateTime" use="required"/>

</complexType>

</element>

</sequence>

</complexType>

</element>

</sequence>

</complexType>

</element>

<simpleType name="measResultType">

<union memberTypes="integer float string">

<simpleType>

<restriction base="string">

<enumeration value="NULL"/>

</restriction>

</simpleType>

</union>

</simpleType>

</schema>

## 12.4 Heartbeat

### 12.4.1 RESTful HTTP-based solution set

#### 12.4.1.1 Mapping of operations

N/A

#### 12.4.1.2 Mapping of notifications

##### 12.4.1.2.1 Introduction

The IS notifications are mapped to SS equivalents according to table 12.4.1.2.1-1.

**Table 12.4.1.2.1-1: Mapping of IS notifications to SS equivalents**

|  |  |  |  |
| --- | --- | --- | --- |
| **IS notifications** | **HTTP Method** | **Resource URI** | **S** |
| notifyHeartbeat | POST | /notificationSink | M |

##### 12.4.1.2.2 Notification "notifyHeartbeat"

The IS notification parameters are mapped to SS equivalents according to table 12.4.1.2.2-1.

Table 12.4.1.2.2-1: Mapping of IS notification input parameters to SS equivalents (HTTP POST)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS operation parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| objectClass | request body | href | Uri | M |
| objectInstance |
| notificationId | request body | notificationId | NotificationId | M |
| notificationType | request body | notificationType | NotificationType (notifyHeartbeat) | M |
| eventTime | request body | eventTime | DateTime | M |
| systemDN | request body | systemDN | systemDN | M |
| heartbeatNtfPeriod | request body | heartbeatNtfPeriod | integer | M |

#### 12.4.1.3 Usage of HTTP

N/A.

#### 12.4.1.4 Resources

N/A.

#### 12.4.1.5 Data type definitions

##### 12.4.1.5.1 General

Table 12.4.1.5.1-1: Data types defined in the present document

|  |  |  |
| --- | --- | --- |
| Data type | Reference | Description |
| HeartbeatNotificationTypes | 12.4.1.4.2.2 | Haertbeat notification types |

Table 12.4.1.5.1-2: Data types imported

|  |  |  |
| --- | --- | --- |
| Data type | Reference | Description |
| Uri | TS 28.623 [44] | URI type |
| NotificationId | TS 28.623 [44] | Notification identifier as defined in ITU-T Rec. X. 733 [4] |
| NotificationType | TS 28.623 [44] | Notification type |
| DateTime | TS 28.623 [44] | Date and time |
| SystemDN | TS 28.623 [44] | systemDN type |
| NotificationHeader | TS 28.623 [44] | Notification header |

##### 12.4.1.5.2 Structured data types

None.

##### 12.4.1.5.3 Simple data types and enumerations

###### 12.4.1.5.3.1 General

This subclause defines simple data types and enumerations that are used by the data structures defined in the previous subclauses.

###### 12.4.1.5.3.2 Simple data types

Table 12.4.1.4.3.2-1: Simple data types

|  |  |  |
| --- | --- | --- |
| Type Name | Type Definition | Description |
|  |  |  |

###### 12.4.1.5.3.3 Enumeration HeartbeatNotificationTypes

Table 12.4.1.4.3.3-1: Enumeration HeartbeatNotificationTypes

|  |  |
| --- | --- |
| Enumeration value | Description |
| notifyHeartbeat | Notification type is notifyHeartbeat |

### 12.4.2 RESTful HTTP-based solution set for integration with ONAP VES API

NOTE: Void.

#### 12.4.2.1 Mapping of operations

See clause 12.1.1.1.

#### 12.4.2.2 Mapping of notifications

##### 12.4.2.2.1 Introduction

###### 12.4.2.2.1.1 General

The 3GPP IS heartbeat notifications are mapped to SS equivalents according to table 12.4.2.2.1.1-1.

Table 12.4.2.2.1.1-1: Mapping of 3GPP IS notifications to SS equivalents

|  |  |  |  |
| --- | --- | --- | --- |
| **3GPP IS notifications** | **HTTP Method** | **Resource URI** | **S** |
| notifyHeartbeat | POST | /eventListener | M |

###### 12.4.2.2.1.2 Notification parameter mapping principles

3GPP IS fault supervision alarm notification parameters are mapped to solution set equivalent as follows:

##### 12.4.2.2.2 Notification notifyHeartbeat

See clause 12.4.1.2.2.

## 12.5 Streaming data reporting service

### 12.5.1 RESTful HTTP-based solution set

#### 12.5.1.1 Mapping of operations

##### 12.5.1.1.1 Introduction

The IS operations are mapped to SS equivalents according to table 12.5.1.1.1-1. The Streaming data reporting MnS shall use TLS as specified in TS 33.210 [55].

Table 12.5.1.1.1-1: Mapping of IS operations to SS equivalents

|  |  |  |  |
| --- | --- | --- | --- |
| **IS operation** | **Method/frame** | **Resource/URI** | **S** |
| establishStreamingConnection | HTTP POST (see NOTE) | /connections | M |
| HTTP GET (Upgrade, see NOTE) | /connections/{connectionId} | M |
| terminateStreamingConnection | WebSocket Close frame sent (frame with opcode of 0x8), and  WebSocket Close frame received (frame with opcode of 0x8 for successful case) | /connections/{connectionId} | M |
| reportStreamData | WebSocket Data frame sent (frame with opcode of 0x2) | /connections/{connectionId} | M |
| addStream | HTTP POST | /connections/{connectionId}/streams | M |
| deleteStream | HTTP DELETE | /connections/{connectionId}/streams | M |
| getConnectionInfo | HTTP GET | /connections | M |
| HTTP GET | /connections/{connectionId} | M |
| getStreamInfo | HTTP GET | /connections/{connectionId}/streams | M |
|  | HTTP GET | /connections/{connectionId}/streams/{streamId} | M |
| Note: the establishStreamingConnection is mapped to a HTTP POST operation followed by a HTTP GET operation. The HTTP POST operation is to provide the information in streamInfoList parameter to the consumer and receive the connectionId assigned by the consumer, while the HTTP GET (Upgrade) operation is to establish the WebSocket connection. | | | |

##### 12.5.1.1.2 Operation "establishStreamingConnection"

The IS operation parameters are mapped to SS equivalents according to the tables 12.5.1.1.2-1 through 12.5.1.1.2-4.

Table 12.5.1.1.2-1: Mapping of IS operation input parameters to SS equivalents (HTTP POST)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IS operation parameter name | SS parameter location | SS parameter name | SS parameter type | S |
| producerId | request body | producerId | String | M |
| streamInfoList | request body | streamInfoList | array(streamInfo-Type) | M |

Table 12.5.1.1.2-2: Mapping of IS operation output parameters to SS equivalents (HTTP POST)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IS operation parameter name | SS parameter location | SS parameter name | SS parameter type | S |
| connectionId | location header | n/a | uri-Type | M |
| status | response status codes  response body | n/a  error | n/a  error-ResponseType | M |

Table 12.5.1.1.2-3: Mapping of IS operation input parameters to SS equivalents (HTTP GET (Upgrade))

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS operation parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| connectionId | Headers | Request-URI | String | n/a |
| -- | HTTP-Version (Request-Line) | -- | String (see Note 1) | M |
| -- | Upgrade Header | -- | Constant string: websocket | M |
| -- | Connection Header | -- | Constant string: Upgrade | M |
| -- | Sec-WebSocket-Key Header | -- | String (see Note 2) | M |
| -- | Sec-WebSocket-Version Header | -- | String (see Note 3) | M |
| -- | See Note 4. | | | |
| NOTE 1: The HTTP version shall be not earlier than HTTP/1.1.  NOTE 2: The valid value needs to be assigned according to WebSocket protocol (see IETF RFC 6455 [40]).  NOTE 3: The valid value needs to be assigned according to WebSocket protocol (see IETF RFC 6455 [40]).  NOTE 4: Other SS parameters (not listed in this table) independent from the Stage 2 may be used, according to the WebSocket protocol (see IETF RFC 6455 [40]). | | | | |

Table 12.5.1.1.2-4: Mapping of IS operation output parameters to SS euivalents (HTTP GET (Upgrade))

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS operation parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| connectionId | n/a | -- | n/a | n/a |
| status | HTTP-Version (Response-Line) | -- | String (see Note 1) | M |
| Status-Code | -- | String |
| response body | error | error-ResponseType |
| -- | Upgrade Header |  | Constant string: websocket | M |
| -- | Connection Header | -- | Constant string: Upgrade | M |
|  | Sec-WebSocket-Accept Header | -- | String (see Note 2) | M |
| -- | See Note 3. | | | |
| NOTE 1: The HTTP version shall be not earlier than HTTP/1.1.  NOTE 2: The valid value needs to be assigned according to WebSocket protocol (see IETF RFC 6455 [40]).  NOTE 3: Other SS parameters (not listed in this table) independent from the Stage 2 may be used, according to the WebSocket protocol (see IETF RFC 6455 [40]). | | | | |



Figure 12.5.1.1.2-1: Message flow for establishing a streaming connection

The message flow for establishing a streaming connection illustrated on Figure 12.5.1.1.2-1 is as follows:

1. The MnS producer sends a HTTP POST request to theMnS consumer.

- The URI identifies the "…/connections" collection resource.

- The request message body carries the information about the connecting producer identity via parameter "producerId" and about streams supported by the new connection via parameter "StreamInfoList".

2. The MnS consumer sends a HTTP POST response to the MnS producer.

- On success "201 Posted" shall be returned with the identifier of a newly created ".../connections/{connectionId}" resource.

- On failure, an appropriate error code shall be returned. The response message body may carry an error object.

3. If step 2 is successful, the MnS producer sends a HTTP GET (upgrade) request to theMnS consumer to establish the WebSocket connection.

- The URI identifies the ".../connections/{connectionId}" resource with the /secure/flag;

- The HTTP-version in the Request-line indicates the HTTP version which is no earlier than HTTP/1.1;

- The Upgrade header is with value "websocket";

- The Connection header is with value "Upgrade";

- The Sec-WebSocket-Key header is with a valid value according to IETF RFC 6455 [40].

- The Sec-WebSocket-Version header is with a valid according to IETF RFC 6455 [40].

4. The MnS consumer sends a HTTP GET (Upgrade) response to the MnS producer.

- On success, "101 Switching Protocols" shall be returned;

- On failure, an appropriate error code shall be returned. The response message body may carry an error object.

- The HTTP-version in the Response-line indicates the HTTP version which is no earlier than HTTP/1.1;

- The Upgrade header is with value "websocket";

- The Connection header is with value "Upgrade";

- The Sec-WebSocket-Accept header is with a valid value according to IETF RFC 6455 [40].

##### 12.5.1.1.3 Operation "terminateStreamingConnection"

The IS operation parameters are mapped to SS equivalents according to the tables 12.5.1.1.3-1 and 12.5.1.1.3-2.

Table 12.5.1.1.3-1: Mapping of IS operation input parameters to SS equivalents (WebSocket Close frame sent)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS operation parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| connectionId | n/a | -- | n/a | n/a |
| -- | Opcode (see clause 5 of IETF RFC 6455 [40]) | -- | Constant value: 0x8 | M |

Table 12.5.1.1.3-2: Mapping of IS operation output parameters to SS equivalents (WebSocket Close frame received)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS operation parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| status | Opcode | -- | For a successful operation, the Opcode is 0x8, and for an unsuccessful operation, the Opcode has a value other than 0x8 (see clause 5 of IETF RFC 6455 [40]). | M |

##### 12.5.1.1.4 Operation "reportStreamData"

The IS operation parameters are mapped to SS equivalents according to the tables 12.5.1.1.4-1 and 12.5.1.1.4-2.

Table 12.5.1.1.4-1: Mapping of IS operation input parameters to SS equivalents (WebSocket Data frame sent with Opcode of 0x2)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS operation parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| connectionId | n/a | -- | n/a | n/a |
| -- | Opcode (see clause 5 of IETF RFC 6455 [40]) | -- | Constant value: 0x2 ("binary") | M |
| streamingData | Payload data | Streaming Trace Payload  or  streaming performance data payload  or  streaming analytics payload  or  proprietary data payload | See clause 5 of TS 32.423 [39] for detailed definition of the Streaming Trace Payload format and Annex G of TS 28.550 [40] for detailed definition of the streaming performance data payload format. | M |

The protocol stack with Streaming Trace Payloads formatted as per clause 5 of TS 32.423 [39] carried by WebSocket binary data frames (see clause 5.6 of IETF RFC 6455 [40]) is illustrated on Figure 12.5.1.1.4-1.

The protocol stack with streaming performance data payloads formatted as per Annex G of TS 28.550 [42] carried by WebSocket binary data frames (see clause 5.6 of IETF RFC 6455 [40]) is illustrated on Figure 12.5.1.1.4-2.

Table 12.5.1.1.4-2: Mapping of IS operation output parameters to SS equivalents

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS operation parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| status | n/a | -- See Note 1. | n/a | n/a |
| NOTE 1: The delivery of WebSocket Data frame is taken care of by the underlying TCP (see IETF RFC 793 [41]) which provides reliable data transmission and ensures the data delivery. There is no mechanism at WebSocket protocol level to report the delivery status for WebSocket Data frame. | | | | |

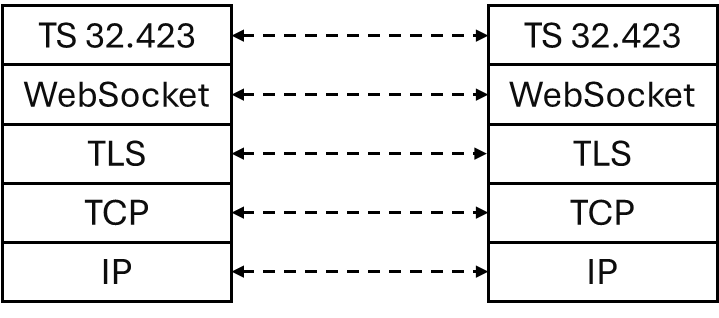


Figure 12.5.1.1.4-1: Protocol stack for streaming trace data reporting

A black background with white squares

Description automatically generated

Figure 12.5.1.1.4-2: Protocol stack for streaming performance data reporting

##### 12.5.1.1.5 Operation "addStream"

The IS operation parameters are mapped to SS equivalents according to the tables 12.5.1.1.5-1 and 12.5.1.1.5-2.

Table 12.5.1.1.5-1: Mapping of IS operation input parameters to SS equivalents (HTTP POST)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS operation parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| connectionId | Headers | Request-URI | String | n/a |
| streamInfoList | request body | streamInfoList | array(streamInfo-Type) | M |

Table 12.5.1.1.5-2: Mapping of IS operation output parameters to SS equivalents (HTTP POST)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS operation parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| streamInfoList | response body | streamInfoList | array(streamInfo-Type) | M |
| status | response status codes  response body | n/a  error | n/a  error-ResponseType | M |

##### 12.5.1.1.6 Operation "deleteStream"

The IS operation parameters are mapped to SS equivalents according to the tables 12.5.1.1.6-1 and 12.5.1.1.6-2.

Table 12.5.1.1.6-1: Mapping of IS operation input parameters to SS equivalents (HTTP DELETE)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS operation parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| connectionId | headers | Request-URI | String | n/a |
| streamIdList | path,  query | /connections/{connectionId}/streams,  streamIdList | array(streamId-Type) | M |

Table 12.5.1.1.6-2: Mapping of IS operation output parameters to SS equivalents (HTTP DELETE)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS operation parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| status | response status codes  response body | n/a  error | n/a  error-ResponseType | M |

##### 12.5.1.1.7 Operation "getConnectionInfo"

The IS operation parameters are mapped to SS equivalents according to the tables 12.5.1.1.7-1 and 12.5.1.1.7-2.

Table 12.5.1.1.7-1: Mapping of IS operation input parameters to SS equivalents (HTTP GET)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS operation parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| connectionId | headers | Request-URI | String | n/a |
| connectionIdList | path,  query | /connections,  /connections/{connectionId} | array(uri-Type) | M |

Table 12.5.1.1.7-2: Mapping of IS operation output parameters to SS equivalents (HTTP GET)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS operation parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| connectionInfoList | response body | connectionInfoList | array(uri-Type, streamReporter-Type, streamIdList-Type) | M |
| status | response status codes  response body | n/a  error | n/a  error-ResponseType | M |

##### 12.5.1.1.8 Operation "getStreamInfo"

The IS operation parameters are mapped to SS equivalents according to the tables 12.5.1.1.8-1 and 12.5.1.1.8-2.

Table 12.5.1.1.8-1: Mapping of IS operation input parameters to SS equivalents (HTTP GET)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS operation parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| connectionId | headers | Request-URI | String | n/a |
| streamIdList | path,  query | /connections/{connectionId}/streams,  streamIdList | array(streamId-Type) | M |

Table 12.5.1.1.8-2: Mapping of IS operation output parameters to SS equivalents (HTTP GET)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS operation parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| streamInfoSumList | response body | streamInfoSumList | array(streamInfo-Type, streamReporters-Type) | M |
| status | response status codes  response body | n/a  error | n/a  error-ResponseType | M |

#### 12.5.1.2 Mapping of notifications

Not applicable (no notifications defined in IS).

#### 12.5.1.3 Resources

##### 12.5.1.3.1 Resources structure

Figure 12.5.1.3.1-1 shows the resource structure of the Streaming data reporting service.



Figure 12.5.1.3.1-1: Resource URI structure of the Streaming data reporting service

Table 12.5.1.3.1-1 provides an overview of the resources and applicable HTTP methods.

|  |  |  |  |
| --- | --- | --- | --- |
| Resource name | Resource URI | HTTP method | Description |
| connections | …/connections | POST | Inform consumer about reporting streams to be carried by the new connection and receive a new connection id. |
| GET | Obtain information about connections |
| connection | …/connections/{connectionId} | GET (Upgrade) | Establish WebSocket for a given connection |
| GET | Obtain information about connection |
| WebSocket 0x2 | Send a unit of streaming data |
| WebSocket 0x8 | Terminate a WebSocket connection |
| streams | …/connections/{connectionId}/streams | POST | Inform consumer about new reporting streams on an existing connection. |
| DELETE | Remove reporting streams from an existing connection |
| GET | Obtain information about streams |
| stream | …/connections/{connectionId}/streams/{streamId} | GET | Obtain information about stream |

##### 12.5.1.3.2 Resources definitions

12.5.1.3.2.1 Resource "…/connections"

12.5.1.3.2.1.1 Description

This resource represents a collection of connections and can be used to establish new connections or to obtain information about existing connections.

12.5.1.3.2.1.2 URI

The resource URI is: {MnSRroot}/StreamingDataReportingMnS/{MnSVersion}/connections

This resource shall support the resource URI variables defined in the table 12.5.1.3.2.1.2-1.

Table 12.5.1.3.2.1.2-1: URI variables

|  |  |
| --- | --- |
| Name | Definition |
| MnSRoot | See clause 4.4.3 of TS 32.158 [15] |
| MnSVersion | See clause 4.4.3 of TS 32.158 [15] |

12.5.1.3.2.1.3 HTTP methods

12.5.1.3.2.1.3.1 HTTP POST

This method shall support the URI query parameters specified in the following table.

**Table 12.5.1.3.2.1.3.1-1: URI query parameters supported by the POST method on this resource**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data type** | **Description** | **S** |
| none supported |  |  |  |

This method shall support the request data structures, the response data structures and response codes specified in the following table.

**Table 12.5.1.3.2.1.3.1-2: Data structures supported by the POST request body on this resource**

|  |  |  |
| --- | --- | --- |
| **Data type** | **Description** | **S** |
| producerId | String representing the DN of the streaming data reporting MnS producer. | M |
| array(streamInfo-Type) | List of meta-data about each reporting stream. Where each reporting stream is represented by a streamInfo. |  |

**Table 12.5.1.3.2.1.3.1-3: Data structures supported by the POST Response Body on this resource**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data type** | **Response codes** | **Description** | **S** |
| error-ResponseType | 4xx/5xx | Returned in case of an error | M |
| uri-Type | 201 Posted | Connection identifier assigned by the MnS consumer | M |

12.5.1.3.2.1.3.2 HTTP GET

This method shall support the URI query parameters specified in the following table.

**Table 12.5.1.3.2.1.3.2-1: URI query parameters supported by the GET method on this resource**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data type** | **Description** | **S** |
| connectionIdList | array(uri-Type) | The list of connectionId for which the connection information is to be returned. | O |

This method shall support the request data structures, the response data structures and response codes specified in the following table.

**Table 12.5.1.3.2.1.3.2-2: Data structures supported by the GET request body on this resource**

|  |  |  |
| --- | --- | --- |
| **Data type** | **Description** | **S** |
| n/a | n/a | n/a |

**Table 12.5.1.3.2.1.3.2-3: Data structures supported by the GET Response Body on this resource**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data type** | **Response codes** | **Description** | **S** |
| error-ResponseType | 4xx/5xx | Returned in case of an error | M |
| array(uri-Type, streamReporter-Type, streamIdList-Type) | 200 OK | In case of success the representation of the retrieved information is returned. | M |
| 202 Partially retrieved | In case of partial success the representation of the retrieved information is returned. | M |

12.5.1.3.2.2 Resource "…/connections/{connectionId}"

12.5.1.3.2.2.1 Description

This resource represents an individual connection and can be used for an "upgrade" to WebSocket as part of the connection establishment, or to obtain information about an existing connection, or to terminate an existing connection, or to send a unit of streaming data.

12.5.1.3.2.2.2 URI

The resource URI is: {MnSRoot}/StreamingDataReportingMnS/{MnSVersion}/connections/{connectionId}

This resource shall support the resource URI variables defined in the table 12.5.1.3.2.2.2-1.

Table 12.5.1.3.2.2.2-1: URI variables

|  |  |
| --- | --- |
| Name | Definition |
| MnSRoot | See clause 4.4.3 of TS 32.158 [15] |
| MnSVersion | See clause 4.4.3 of TS 32.158 [15] |
| connectionId | Represents identifier of an individual connection assigned by the MnS consumer during connection establishment |

12.5.1.3.2.2.3 HTTP methods

12.5.1.3.2.2.3.1 HTTP GET (Upgrade)

This method shall support the URI header parameters specified in the following table.

**Table 12.5.1.3.2.2.3.2-1: Header parameters supported by the GET request on this resource**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data type** | **Description** | **S** |
| connectionId | uri-Type | To indicate the ID (URI) of the connection being upgraded to WebSocket | M |
| Upgrade | Upgrade-HeaderType | To indicate the HTTP GET operation is to upgrade the connection to WebSocket protocol | M |
| Connection | Connection-HeaderType | To indicate the HTTP GET operation is to upgrade the connection to another protocol | M |
| Sec-WebSocket-Key | Sec-WebSocket-Key-HeaderType | The Sec-WebSocket-Key needed for establishing the WebSocket connection. | M |
| Sec-WebSocket-Version | Sec-WebSocket-Version-HeaderType | The Sec-WebSocket-Version needed for establishing the WebSocket connection. | M |

This method shall support the URI query parameters specified in the following table.

**Table 12.5.1.3.2.2.3.2-2: URI query parameters supported by the GET method on this resource**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data type** | **Description** | **S** |
| none supported |  |  |  |

This method shall support the request data structures, the response data structures and response codes specified in the following table.

**Table 12.5.1.3.2.2.3.2-3: Data structures supported by the GET request body on this resource**

|  |  |  |
| --- | --- | --- |
| **Data type** | **Description** | **S** |
| n/a | n/a | n/a |

**Table 12.5.1.3.2.2.3.2-4: Header parameters supported by the GET response on this resource**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data type** | **Description** | **S** |
| Upgrade | Upgrade-HeaderType | To indicate the HTTP GET operation is to upgrade the connection to WebSocket protocol | M |
| Connection | Connection-HeaderType | To indicate the HTTP GET operation is to upgrade the connection to another protocol | M |
| Sec-WebSocket-Accept | Sec-WebSocket-Accept-HeaderType | The Sec-WebSocket-Accept responded when establishing the WebSocket connection. | M |

**Table 12.5.1.3.2.2.3.2-5: Data structures supported by the GET response body on this resource**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data type** | **Response**  **codes** | **Description** | **S** |
| n/a | 101 Switching Protocols | The status code indicating the connection has been successfully upgraded to WebSocket. | M |
| error-ResponseType | 4xx/5xx | Returned in case of an error | M |

12.5.1.3.2.2.3.2 HTTP GET

This method shall support the URI query parameters specified in the following table.

**Table 12.5.1.3.2.1.3.2-1: URI query parameters supported by the GET method on this resource**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data type** | **Description** | **S** |
| none supported |  |  |  |

This method shall support the request data structures, the response data structures and response codes specified in the following table.

**Table 12.5.1.3.2.1.3.2-2: Data structures supported by the GET request body on this resource**

|  |  |  |
| --- | --- | --- |
| **Data type** | **Description** | **S** |
| n/a | n/a | n/a |

**Table 12.5.1.3.2.1.3.2-3: Data structures supported by the GET Response Body on this resource**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data type** | **Response codes** | **Description** | **S** |
| error-ResponseType | 4xx/5xx | Returned in case of an error | M |
| uri-Type | 200 OK | In case of success the representation of the connectionId is returned. | M |
| streamReporter-Type | 200 OK | In case of success the representation of the streamReporter is returned. | M |
| streamIdList-Type | 200 OK | In case of success the representation of the streamIdList is returned. | M |

12.5.1.3.2.3 Resource "…/connections/{connectionId}/streams"

12.5.1.3.2.3.1 Description

This resource represents a collection of reporting streams on a particular connection and can be used to add a new reporting stream to an existing connection, or to remove a reporting stream from an existing connection, or to obtain information about reporting streams.

12.5.1.3.2.3.2 URI

The resource URI is: {MnSRoot}/StreamingDataReportingMnS/{MnSVersion}/connections/{connectionId}/streams

This resource shall support the resource URI variables defined in the table 12.5.1.3.2.3.2-1.

Table 12.5.1.3.2.3.2-1: URI variables

|  |  |
| --- | --- |
| Name | Definition |
| MnSRoot | See clause 4.4.3 of TS 32.158 [15] |
| MnSVersion | See clause 4.4.3 of TS 32.158 [15] |
| connectionId | See table 12.5.1.3.2.2.2-1 |

12.5.1.3.2.3.3 HTTP methods

12.5.1.3.2.3.3.1 HTTP POST

This method shall support the URI query parameters specified in the following table.

**Table 12.5.1.3.2.3.3.1-1: URI query parameters supported by the POST method on this resource**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data type** | **Description** | **S** |
| none supported |  |  |  |

This method shall support the request data structures, the response data structures and response codes specified in the following table.

**Table 12.5.1.3.2.3.3.1-2: Data structures supported by the POST request body on this resource**

|  |  |  |
| --- | --- | --- |
| **Data type** | **Description** | **S** |
| array(streamInfo-Type) | The resource representation of the set of information about streams to be posted. | M |

**Table 12.5.1.3.2.3.3.1-3: Data structures supported by the POST Response Body on this resource**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data type** | **Response codes** | **Description** | **S** |
| array(streamInfo-Type) | 201 Posted | In case of success the representation of the posted information about streams is returned. | M |
| 202 Partially posted | In case of partial success the representation of the posted information about streams is returned. | M |
| error-ResponseType | 4xx/5xx | Returned in case of an error | M |

12.5.1.3.2.3.3.2 HTTP DELETE

This method shall support the URI query parameters specified in the following table.

Table 12.5.1.3.2.3.3.2-1: URI query parameters supported by the DELETE method on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data type** | **Description** | **S** |
| streamIdList | array(streamId-Type) | The list of streamId for the stream(s) to be deleted. | M |

This method shall support the request data structures, the response data structures and response codes specified in the following table.

Table 12.5.1.3.2.3.3.2: Data structures supported by the DELETE request body on this resource

|  |  |  |
| --- | --- | --- |
| **Data type** | **Description** | **S** |
| n/a | n/a | n/a |

Table 12.5.1.3.2.3.3.2-3: Data structures supported by the DELETE Response Body on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| **Data type** | **Response codes** | **Description** | **S** |
| n/a | 204 No Content | In case of success no message body is returned | M |
| error-ResponseType | 4xx/5xx | Returned in case of an error | M |

12.5.1.3.2.3.3.3 HTTP GET

This method shall support the URI query parameters specified in the following table.

Table 12.5.1.3.2.3.3.3-1: URI query parameters supported by the GET method on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data type** | **Description** | **S** |
| streamIdList | array(streamId-Type) | The list of streamId for which the stream information are to be returned. | O |

This method shall support the request data structures, the response data structures and response codes specified in the following table.

Table 12.5.1.3.2.3.3.3-2: Data structures supported by the GET request body on this resource

|  |  |  |
| --- | --- | --- |
| **Data type** | **Description** | **S** |
| n/a | n/a | n/a |

Table 12.5.1.3.2.3.3.3-3: Data structures supported by the GET Response Body on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| **Data type** | **Response codes** | **Description** | **S** |
| array(streamInfo-Type, streamReporters-Type) | 200 OK | In case of success the representation of the retrieved stream information is returned. | M |
| 202 Partially retrieved | In case of partial success the representation of the retrieved stream information is returned. | M |
| error-ResponseType | 4xx/5xx | Returned in case of an error | M |

12.5.1.3.2.4 Resource "…/connections/{connectionId}/streams/{streamId}"

12.5.1.3.2.4.1 Description

This resource represents an individual reporting stream on an existing connection and can be used to obtain information about reporting stream.

12.5.1.3.2.4.2 URI

The resource URI is: {MnSRoot}/StreamingDataReportingMnS/{MnSVersion}/connections/{connectionId}/streams/{streamId}

This resource shall support the resource URI variables defined in the table 12.5.1.3.2.4.2-1.

Table 12.5.1.3.2.4.2-1: URI variables

|  |  |
| --- | --- |
| Name | Definition |
| MnSRoot | See clause 4.4.3 of TS 32.158 [15] |
| MnSVersion | See clause 4.4.3 of TS 32.158 [15] |
| connectionId | See table 12.5.1.3.2.2.2-1 |
| streamId | Represents identifier of an individual stream. For Streaming Trace reporting, the Trace Reference (see clause 5.6 of TS 32.422 [38]) is used as stream identifier |

12.5.1.3.2.4.3 HTTP methods

12.5.1.3.2.4.3.1 HTTP GET

This method shall support the URI query parameters specified in the following table.

**Table 12.5.1.3.2.4.3.1-1: URI query parameters supported by the GET method on this resource**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data type** | **Description** | **S** |
| none supported |  |  |  |

This method shall support the request data structures, the response data structures and response codes specified in the following table.

**Table 12.5.1.3.2.4.3.1-2: Data structures supported by the GET request body on this resource**

|  |  |  |
| --- | --- | --- |
| **Data type** | **Description** | **S** |
| n/a | n/a | n/a |

**Table 12.5.1.3.2.4.3.1-3: Data structures supported by the GET Response Body on this resource**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data type** | **Response codes** | **Description** | **S** |
| streamInfo-Type | 200 OK | In case of success the representation of the retrieved stream information is returned. | M |
| streamReporters-Type | 200 OK | In case of success the representation of the retrieved stream reporters information is returned. | M |
| error-ResponseType | 4xx/5xx | Returned in case of an error | M |

#### 12.5.1.4 Data type definitions

##### 12.5.1.4.1 General

Table 12.5.1.4.1-1: Data types defined

|  |  |  |
| --- | --- | --- |
| **Data type** | **Reference** | **Description** |
| **General types** | | |
| uri-Type | 12.5.1.4.3 | Used to represent a URI |
| **Types used in paths** | | |
| connectionId-Type | 12.5.1.4.3 | Used to indicate the connection as a context of the operation |
| streamId-Type | 12.5.1.4.3 | Used to indicate the stream as a context of the operation |
| **Types used in headers** | | |
| websocketHeaderConnection-Type | 12.5.1.4.3 | Header value for the upgrade request and response |
| websocketHeaderUpgrade-Type | 12.5.1.4.3 | Header value for the upgrade to WebSocket request and response |
| websocketHeader-Sec-WebSocket-Accept-Type | 12.5.1.4.3 | Header value for secure WebSocket response. Carries hash. |
| websocketHeader-Sec-WebSocket-Extensions-Type | 12.5.1.4.3 | Header value for secure WebSocket request. Carries protocol extensions. |
| websocketHeader-Sec-WebSocket-Key-Type | 12.5.1.4.3 | Header value for secure WebSocket request. Provides information to the server which is needed in order to confirm that the client is entitled to request an upgrade to WebSocket. |
| websocketHeader-Sec-WebSocket-Protocol-Type | 12.5.1.4.3 | Header value for secure WebSocket request. Carries a comma-separated list of subprotocol names, in the order of preference. |
| websocketHeader-Sec-WebSocket-Version-Type | 12.5.1.4.3 | Header value for secure WebSocket request and response. Carries the WebSocket protocol version to be used. |
| **Types used in query parts** | | |
| connectionId-Type | 12.5.1.4.3 | Used to indicate the connection as a context of the operation |
| streamId-Type | 12.5.1.4.3 | Used to indicate the stream as a context of the operation |
| **Types used in request bodies** | | |
| connectionRequest-Type | 12.5.1.4.2.2 | Used to carry the meta-data during connection establishment |
| streamInfo-Type | 12.5.1.4.2.5 | Reporting stream meta-data. |
| **Types used in response bodies** | | |
| failedConnectionResponse-Type | 12.5.1.4.2.4 | Used to carry the details of a failed connection establishment |
| connectionInfo-Type | 12.5.1.4.2.1 | Used to carry connection meta-data |
| errorResponse-Type | 12.5.1.4.2.3 | Used to carry the details of an error |
| streamInfo-Type | 12.5.1.4.2.5 | Used to carry the stream meta-data |
| streamInfoWithReporters-Type | 12.5.1.4.2.6 | Used to carry the augmented stream meta-data |
| **Types used for resources** | | |
| uri-Type | 12.5.1.4.3 | Used to represent resource URI |
| **Types referenced by the definitions above** | | |
| systemDN-Type | 12.5.1.4.3 | Used to represent DN of the reporting entity |
| traceJob-Type | Generic NRM | Used to represent Trace configuration |
| producerId-Type | 12.5.1.4.3 | Used to identify the reporting entity |
| streamType-Type | 12.5.1.4.3 | Used to identify the type of a reporting stream |
| serializationFormat-Type | 12.5.1.4.3 | Used to identify serialization method |
| measObjDn-Type | 12.5.1.4.3 | Used to represent DN of the measured object instance |
| measTypes-Type | 12.5.1.4.3 | Used to represent an ordered list of measurement types or KPI |
| analyticsInfo-Type | 12.5.1.4.3 | Used to represents information about streamed analytics |
| vsDataContainer-Type | Generic NRM | Used to represent details about proprietary data |

Table 12.5.1.4.1-2: Data types imported

|  |  |  |
| --- | --- | --- |
| **Data type** | **Reference** | **Description** |
| traceJob-Type | Generic NRM | Attributes container of the TraceJob IOC (see TS 28.622 [11]). |
| vsDataContainer-Type | Generic NRM | Vendor specific data container (see TS 28.622 [11]). |

##### 12.5.1.4.2 Query, message body and resource data types

12.5.1.4.2.1 Type connectionInfo-Type

Table 12.5.1.4.2.1-1: Definition of type connectionInfo-Type

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Description** | **S** |
| connection | connectionId-Type | Connection identifier | M |
| producer | producerId-Type | Producer identifier | M |
| streams | array(streamId-Type) | List of stream identifiers | M |

12.5.1.4.2.2 Type connectionRequest-Type

Table 12.5.1.4.2.2-1: Definition of type connectionRequest-Type

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Description** | **S** |
| producer | producerId-Type | Producer identifier | M |
| streams | array(streamInfo-Type) | List of stream meta-data | M |

12.5.1.4.2.3 Type errorResponse-Type

Table 12.5.1.4.2.3-1: Definition of type errorResponse-Type

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Description** | **S** |
| error | object | Key indicating the response body containing an error | M |
| > errorInfo | string | Attribute allowing to convey error information in string format | M |

12.5.1.4.2.4 Type failedConnectionResponse-Type

Table 12.5.1.4.2.4-1: Definition of type failedConnectionResponse-Type

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Description** | **S** |
| error | object | Key indicating the response body containing an error | M |
| > streamId | array(streamId-Type) | Attribute conveying the list of "problematic" stream IDs | M |
| > errorReason | string | Attribute allowing to convey error information in string format |  |

12.5.1.4.2.5 Type streamInfo-Type

Table 12.5.1.4.2.5-1: Definition of type streamInfo-Type

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Description** | **S** |
| streamId | streamId-Type | Stream identifier | M |
| streamType | streamType-Type | Enumerated stream type | M |
| serializationFormat | serializationFormat-Type | Enumerated serialization method | M |
| measObjDn | measObjDn-Type | DN of the measured object instance. Used for streaming performance data only. | CM |
| measTypes | measTypes-Type | Ordered list of measurement types or KPI. Used for streaming performance data only. | CM |
| analyticsInfo | analyticsInfo-Type | Information about streamed analytics. Used for streaming analytics only. | CM |
| vsDataContainer | vsDataContainer-Type | Details about proprietary data. Mandatory for proprietary data streaming only. | CM |
| traceInfo | traceJob-Type | Trace configuration. Used for streaming trace data reporting streams only. | CM |

Table 12.5.1.4.2.5-2: Attribute constraints

|  |  |
| --- | --- |
| Name | Definition |
| measObjDn (support qualifier) | Attribute shall be present for streaming performance data only. |
| measTypes (support qualifier) | Attribute shall be present for streaming performance data only. |
| analyticsInfo (support qualifier) | Attribute shall be present for streaming analytics only. |
| vsDataContainer (support qualifier) | Attribute shall be present for proprietary data streaming. |
| traceInfo (support qualifier) | Attribute shall be present for streaming trace data only. |

12.5.1.4.2.6 Type streamInfoWithReporters-Type

Table 12.5.1.4.2.6-1: Definition of type streamInfoWithReporters-Type

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Description** | **S** |
| streamInfo | streamInfo-Type | Stream meta-data | M |
| reporters | producerId-Type | List of entities reporting streaming data | M |

##### 12.5.1.4.3 Simple data types and enumerations

12.5.1.4.3.1 General

This subclause defines simple data types and enumerations that are used by the data structures defined in the previous subclauses.

12.5.1.4.3.2 Simple data types

Table 12.5.1.4.3.2-1: Simple data types

|  |  |  |
| --- | --- | --- |
| Type name | Type definition | Description |
| analyticsInfo-Type | string | Information about streamed analytics. |
| measObjDn-Type | DN | See TS 32.300 [25] |
| measTypes-Type | string | See TS 28.550 [42] |
| websocketHeaderConnection-Type | Constant string "Upgrade" | Header value for the upgrade request and response. |
| websocketHeaderUpgrade-Type | Constant string "websocket" | Header value for the upgrade to WebSocket request and response. |
| websocketHeader-Sec-WebSocket-Accept-Type | string | Header value for secure WebSocket response. Carries hash. |
| websocketHeader-Sec-WebSocket-Extensions-Type | string | Header value for secure WebSocket request. Carries protocol extensions. |
| websocketHeader-Sec-WebSocket-Key-Type | string | Header value for secure WebSocket request. Provides information to the server which is needed in order to confirm that the client is entitled to request an upgrade to WebSocket. |
| websocketHeader-Sec-WebSocket-Protocol-Type | string | Header value for secure WebSocket request. Carries a comma-separated list of subprotocol names, in the order of preference. |
| websocketHeader-Sec-WebSocket-Version-Type | string | Header value for secure WebSocket request and response. Carries the WebSocket protocol version to be used. |
| connectionId-Type | uri-Type | Used to indicate the connection as a context of the operation |
| producerId-Type | systemDN-Type | Used to identify the reporting entity |
| serializationFormat-Type | enum | Enumerated serialization method with values: "GPB", "ASN1" |
| streamId-Type | Trace Reference | See TS 32.422 [38] |
| streamType-Type | enum | Enumerated stream type with values: "TRACE", "PERFORMANCE", "ANALYTICS", "PROPRIETARY" |
| systemDN-Type | DN | See TS 32.300 [25] |
| uri-Type | string | Used to represent resource URI |

## 12.6 File data reporting service

### 12.6.1 RESTful HTTP-based solution set

#### 12.6.1.1 Mapping of operations

##### 12.6.1.1.1 Introduction

The IS operations are mapped to SS equivalents according to table 12.6.1.1.1-1.

Table 12.6.1.1.1-1: Mapping of IS operations to SS equivalents

|  |  |  |  |
| --- | --- | --- | --- |
| IS operation | HTTP Method | Resource URI | S |
| listAvailableFiles | GET | /files | M |
| subscribe | POST | /subscriptions | M |
| unsubscribe | DELETE | /subscriptions/{subscriptionId} | M |

##### 12.6.1.1.2 Operation listAvailableFiles

The IS operation parameters are mapped to SS equivalents according to table 12.6.1.1.2-1 and table 12.6.1.1.2-2.

Table 12.6.1.1.2-1: Mapping of IS operation input parameters to SS equivalents (HTTP GET)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| fileDataType | query | fileDataType | FileDataType | M |
| beginTime | query | beginTime | DateTime | M |
| endTime | query | endTime | DateTime | M |

Table 12.6.1.1.2-2: Mapping of IS operation output parameters to SS equivalents (HTTP GET)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| fileInfoList | response body | n/a | array(FileInfo) | M |
| status | response status codes | n/a | n/a | M |
| response body | error | ErrorResponse | O |

The message flow is as follows:

- 1. The MnS consumer sends a HTTP GET request to the MnS producer.

- The URI identifies the "…/files" collection resource.

- The query part may contain filter parameters. Absence of the query component means all available files shall be returned.

- The request message body shall be empty.

2. The MnS producer sends a HTTP GET response to the MnS consumer.

- On success "200 OK" shall be returned. The response message body shall carry the information of available files. The response format is defined by " array(FileInfo) ".

- On failure, an appropriate error code shall be returned. The response message body may provide additional error information..

##### 12.6.1.1.3 Operation subscribe

See clause 12.2.1.1.8.

##### 12.6.1.1.4 Operation unsubscribe

See clause 12.2.1.1.9.

#### 12.6.1.2 Mapping of notifications

##### 12.6.1.2.1 Introduction

The IS notifications are mapped to SS equivalents according to table 12.6.1.2.1-1.

**Table 12.6.1.2.1-1: Mapping of IS notifications to SS equivalents**

|  |  |  |  |
| --- | --- | --- | --- |
| **IS notification** | **HTTP Method** | **Resource URI** | **S** |
| notifyFileReady | POST | {notificationTarget} | M |
| notifyFilePreparationError | POST | {notificationTarget} | M |

##### 12.6.1.2.2 Notification notifyFileReady

The IS notification parameters are mapped to SS equivalents according to table 12.6.1.2.2-1.

Table 12.6.1.2.2-1: Mapping of IS notification input parameters to SS equivalents (HTTP POST)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| objectClass | request body | href | Uri | M |
| objectInstance |
| notificationId | request body | notificationId | NotificationId | M |
| notificationType | request body | notificationType | NotificationType | M |
| eventTime | request body | eventTime | DateTime | M |
| systemDN | request body | systemDN | SystemDN | M |
| fileInfoList | request body | fileInfoList | array(FileInfo) | M |
| additionalText | request body | additionalText | string | O |

##### 12.6.1.2.3 Notification notifyFilePreparationError

The IS notification parameters are mapped to SS equivalents according to table 12.6.1.2.3-1.

Table 12.6.1.2.3-1: Mapping of IS notification input parameters to SS equivalents (HTTP POST)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **IS parameter name** | **SS parameter location** | **SS parameter name** | **SS parameter type** | **S** |
| objectClass | request body | href | Uri | M |
| objectInstance |
| notificationId | request body | notificationId | NotificationId | M |
| notificationType | request body | notificationType | NotificationType | M |
| eventTime | request body | eventTime | DateTime | M |
| systemDN | request body | systemDN | SystemDN | M |
| fileInfoList | request body | fileInfoList | array(FileInfo) | M |
| reason | request body | reason | string | O |
| additionalText | request body | additionalText | string | O |

#### 12.6.1.3 Resources

##### 12.6.1.3.1 Resource structure

###### 12.6.1.3.1.1 Resource structure on the MnS producer

Figure 12.6.1.3.1.1-1 shows the resource structure of the File Data Reporting MnS on the MnS producer.



Figure 12.6.1.3.1.1-1: Resource URI structure of the File Data Reporting MnS on the MnS producer

Table 12.2.1.3.1.1-1 provides an overview of the resources and applicable HTTP methods.

Table 12.2.1.3.1.1-1: Resources and methods overview

|  |  |  |  |
| --- | --- | --- | --- |
| Resource name | Resource URI | HTTP method | Description |
| Files | …/files | GET | Retrieve the information of the available files |
| Subscriptions | …/subscriptions | POST | Create a subscription |
| Subscription | …/subscriptions/{subscriptionId} | DELETE | Delete a single subscription |
| Notification Target | {notificationTarget} | POST | Send a notification to the notification target |

###### 12.6.1.3.1.2 Resource structure on the MnS consumer

Figure 12.6.1.3.1.2-1 shows the resource structure of the File Data Reporting MnS on the MnS consumer.



Figure 12.6.1.3.1.2-1: Resource URI structure of the File Data Reporting MnS on the MnS consumer

Table 12.6.1.3.1.2-1 provides an overview of the resources and applicable HTTP methods.

Table 12.6.1.3.1.2-1: Resources and methods overview

|  |  |  |  |
| --- | --- | --- | --- |
| Resource name | Resource URI | HTTP method | Description |
| Notification Target | {notificationTarget} | POST | Send a notification to the notification target |

##### 12.6.1.3.2 Resource definitions

12.6.1.3.2.1 Resource "…/files"

12.6.1.3.2.1.1 Description

This resource represents the information about a collection of available files.

12.6.1.3.2.1.2 URI

Resource URI = {MnSRoot}/FileDataReportingMnS/{MnSVersion}/files

The resource URI variables are defined in table 12.6.1.3.2.1.1-1.

Table 12.6.1.3.2.1.1-1: URI variables

|  |  |
| --- | --- |
| **Name** | **Definition** |
| MnSRoot | See clause 4.4.3 of TS 32.158 [15] |
| MnSVersion | See clause 4.4.3 of TS 32.158 [15] |

12.6.1.3.2.1.3 HTTP methods

12.6.1.3.2.1.3.1 HTTP GET

This method shall support the URI query parameters specified in the following table.

**Table 12.6.1.3.2.1.3.1-1: URI query parameters supported by the GET method on this resource**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data type** | **Description** | **S** |
| fileDataType | FileDataType | Selects files based on the file data type. | M |
| beginTime | DateTime | Selects files based on the earliest time they became available | M |
| endTime | DateTime | Selects files based on the latest time they became available | M |

This method shall support the request data structures, the response data structures and response codes specified in the following tables.

**Table 12.6.1.3.2.1.3.1-2: Data structures supported by the GET request body on this resource**

|  |  |  |
| --- | --- | --- |
| **Data type** | **Description** | **S** |
| n/a | n/a | n/a |

**Table 12.6.1.3.2.1.3.1-3: Data structures supported by the GET response body on this resource**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data type** | **Response codes** | **Description** | **S** |
| array(FileInfo) | 200 OK | Information about the files identified in the request | M |
| ErrorResponse | 4xx/5xx | Returned in case of an error | M |

12.6.1.3.2.2 Resource "…/subscriptions"

12.6.1.3.2.2.1 Description

This resource is a container resource for individual subscriptions.

12.6.1.3.2.2.2 URI

Resource URI: {MnSRoot}/FileDataReportingMnS/{MnSVersion}/subscriptions

The resource URI variables are defined in table 12.6.1.3.3.2.2.2-1:

Table 12.6.1.3.3.2.2.2-1: URI variables

|  |  |
| --- | --- |
| **Name** | **Definition** |
| MnSRoot | See clause 4.4.3 of TS 32.158 [15] |
| MnSVersion | See clause 4.4.3 of TS 32.158 [15] |

12.6.1.3.2.2.3 HTTP methods

12.6.1.3.2.2.3.1 POST

This method shall support the URI query parameters specified in table 12.6.1.3.2.2.3.1-1.

Table 12.6.1.3.2.2.3.1-1: URI query parameters supported by the POST method on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data type | Description | S |
| n/a | n/a | n/a | n/a |

This method shall support the request data structures specified in table 12.6.1.3.2.2.3.1-2 and the response data structures and response codes specified in table 12.6.1.3.2.2.3.1-3.

Table 12.6.1.3.2.2.3.1-2: Data structures supported by the POST Request Body on this resource

|  |  |  |
| --- | --- | --- |
| Data type | Description | S |
| Subscription | Details of the subscription to be created | M |

Table 12.6.1.3.2.2.3.1-3: Data structures supported by the POST Response Body on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| Data type | Response codes | Description | S |
| Subscription | 201 Created | In case of success the representation of the created subscription is returned. | M |
| ErrorResponse | 4xx/5xx | In case of failure the error object is returned. | M |

12.6.1.3.2.2.3.2 Void

12.6.1.3.2.3 Resource ".../subscriptions/{subscriptionId}"

12.6.1.3.2.3.1 Description

This resource represents a subscription.

12.6.1.3.2.3.2 URI

Resource URI: {MnSRoot}/FileDataReportingMnS/{MnSVersion}/subscriptions/{subscriptionId}

The resource URI variables are defined in table 12.6.1.3.2.3.2-1.

Table 12.6.1.3.2.3.2-1: URI variables

|  |  |
| --- | --- |
| Name | Definition |
| MnSRoot | See clause 4.4.3 of TS 32.158 [15] |
| MnSVersion | See clause 4.4.3 of TS 32.158 [15] |
| subscriptionId | Subscription identifier |

12.6.1.3.2.3.3 HTTP methods

12.6.1.3.2.3.3.1 DELETE

This method shall support the URI query parameters specified in table 12.6.1.3.2.3.3-1.

Table 12.6.1.3.2.3.3-1: URI query parameters supported by the DELETE method on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data type | Description | S |
| n/a | n/a | n/a | n/a |

This method shall support the request data structures specified in table 12.6.1.3.2.3.3-2 and the response data structures and response codes specified in table 12.6.1.3.2.3.3-3.

Table 12.6.1.3.2.3.3-2: Data structures supported by the DELETE Request Body on this resource

|  |  |  |
| --- | --- | --- |
| Data type | Description | S |
| n/a | n/a | n/a |

Table 12.6.1.3.2.3.3-3: Data structures supported by the DELETE Response Body on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| Data type | Response codes | Description | S |
| n/a | 204 No Content | In case of success no message body is returned | M |
| ErrorResponse | 4xx/5xx | In case of failure the error object is returned. | M |

12.6.1.3.2.4 Resource "/notificationTarget"

12.6.1.3.2.4.1 Description

This resource represents a notification target on the MnS consumer.

12.6.1.3.2.4.2 URI

Resource URI: {notificationTarget}

The resource URI variables are defined in table 12.6.1.3.2.4.2-1.

Table 12.6.1.3.2.4.2-1: URI variables

|  |  |
| --- | --- |
| Name | Definition |
| notificationTarget | URI of the notification target on the MnS consumer, contained in the notification subscription |

12.6.1.3.2.4.3 HTTP methods

12.6.1.3.2.4.3.1 POST

This method shall support the URI query parameters specified in table 12.6.1.3.2.4.3.1-1.

Table 12.6.1.3.2.4.3.1-1: URI query parameters supported by the POST method on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data type | Description | S |
| n/a | n/a | n/a | n/a |

This method shall support the request data structures specified in table 12.6.1.3.2.4.3.1-2 and the response data structures and response codes specified in table 12.6.1.3.2.4.3.1-3.

Table 12.6.1.3.2.4.3.1-2: Data structures supported by the POST Request Body on this resource

|  |  |  |
| --- | --- | --- |
| Data type | Description | S |
| NotifyFileReady | Type in case a notifyFileReady notification is sent | M |
| NotifyFilePreparationError | Type in case a notifyFilePreparationError notification is sent | M |

Table 12.6.1.3.2.4.3.1-3: Data structures supported by the POST Response Body on this resource

|  |  |  |  |
| --- | --- | --- | --- |
| Data type | Response codes | Description | S |
| n/a | 204 No Content | In case of success no message body is returned | M |
| ErrorResponse | 4xx/5xx | In case of failure the error object is returned. | M |

#### 12.6.1.4 Data type definitions

##### 12.6.1.4.1 General

Table 12.6.1.4.1-1: Data types defined in this specification

|  |  |  |
| --- | --- | --- |
| **Data type** | **Reference** | **Description** |
| FileInfo | 12.6.1.4.2.1 | Information describing a file |
| NotifyFileReady | 12.6.1.4.2.2 | Used in the request body of HTTP POST for the notification type notifyFileReady |
| NotifyFilePreparationError | 12.6.1.4.2.3 | Used in the request body of HTTP POST for the notification type notifyFilePreparationError |
| FileDataType | 12.6.1.4.6.3 | File data types |
| FileNotificationTypes | 12.6.1.4.6.4 | File notification types |

Table 12.6.1.4.1-2: Data types imported

|  |  |  |
| --- | --- | --- |
| **Data type** | **Reference** | **Description** |
| DateTime | TS 28.623 [44] | Date and time |
| Float | TS 28.623 [44] | Float type |
| Uri | TS 28.623 [44] | URI type |
| SystemDN | TS 28.623 [44] | systemDN type |
| NotificationId | TS 28.623 [44] | Notification identifier as defined in ITU-T Rec. X. 733 [4] |
| NotificationHeader | TS 28.623 [44] | Notification header |
| ErrorResponse | TS 28.623 [44] | Used in the response body of multiple HTTP methods in case of error |
| Subscription | 12.2.1.4.1a.8 | Subscription resource |

##### 12.6.1.4.2 Structured data types

12.6.1.4.2.1 Type FileInfo

Table 12.6.1.4.2.1-1: Definition of FileInfo

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Description** | **S** |
| fileLocation | Uri | Location of the file | M |
| fileCompression | string | Name of the compression algorithm used for compressing the file | M |
| fileSize | integer | Size of the file, unit is byte | M |
| fileDataType | FileDataType | Type of management data stored in the file | M |
| fileFormat | string | Encoding technique used for encoding the file. Its value should indicate the version of the file format specification plus to indicate if "ASN1" or "XML-schema" is used | M |
| fileReadyTime | DateTime | Date and time when the file was last closed and made available in the MnS producer. The file content will not be changed any more. | M |
| fileExpirationTime | DateTime | Date and time after which the file may be deleted | M |
| jobId | string | Job identifier of the "PerfMetricJob" or "TraceJob" that produced the file | CM |

12.6.1.4.2.2 Type NotifyFileReady

**Table 12.6.1.4.2.2-1: Definition of type NotifyFileReady**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Description** | **S** |
| href | Uri | URI of the object representing the process, managed element or management node, which made the file available | M |
| notificationId | NotificationId | Notification identifier as defined in ITU-T Rec. X. 733 [4] | M |
| notificationType | NotificationType | Notification type (notifyFileReady, etc.) | M |
| eventTime | DateTime | Event occurrence time (e.g., the file ready time) | M |
| systemDN | SystemDN | DN of the MnS Agent emitting the notification | M |
| fileInfoList | array(FileInfo) | Information describing the available files | M |
| additionalText | string | Allows a free form text description to be reported as defined in ITU-T Rec. X. 733 [4] | O |

12.6.1.4.2.3 Type NotifyFilePreparationError

**Table 12.6.1.4.2.3-1: Definition of type NotifyFilePreparationError**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute name** | **Data type** | **Description** | **S** |
| href | Uri | URI of the object representing the process, managed element or management node, where the file preparation error occured | M |
| notificationId | NotificationId | Notification identifier as defined in ITU-T Rec. X. 733 [4] | M |
| notificationType | NotificationType | Notification type (notifyFileReady, etc.) | M |
| eventTime | DateTime | Event occurrence time (e.g., the file ready time) | M |
| systemDN | SystemDN | DN of the MnS Agent emitting the notification | M |
| fileInfoList | array(FileInfo) | Information about the files with a preparation error. | M |
| reason | string | Reason for the file preparation error | O |
| additionalText | string | Allows a free form text description to be reported as defined in ITU-T Rec. X. 733 [4] | O |

##### 12.6.1.4.3 Void

##### 12.6.1.4.4 Void

##### 12.6.1.4.5 Void

##### 12.6.1.4.6 Simple data types and enumerations

12.6.1.4.6.1 General

This clause defines simple data types and enumerations that are used by the data structures defined in the previous clauses.

12.6.1.4.6.2 Simple data types

Table 12.6.1.4.6.2-1: Simple data types

|  |  |  |
| --- | --- | --- |
| Type name | Type definition | Description |
| n/a | n/a | n/a |

12.6.1.4.6.3 Enumeration FileDataType

Table 12.6.1.4.6.3-1: Enumeration FileDataType

|  |  |
| --- | --- |
| Enumeration value | Description |
| PERFORMANCE | Performance data file (measurements and KPIs) |
| TRACE | Trace data file |
| ANALYTICS | Analytics data file |
| PROPRIETARY | Proprietary data file |

12.6.1.4.6.4 Enumeration FileNotificationTypes

Table 12.6.1.4.6.4-1: Enumeration FileNotificationTypes

|  |  |
| --- | --- |
| Enumeration value | Description |
| notifyFileReady | Notification type is notifyFileReady |
| notifyFilePreparationError | Notification type is notifyFilePreparationError |

Annex A (normative):  
OpenAPI specification

# A.0 Introduction

This clause describes the capabilities of the service in the structure of the OpenAPI Specification Version 3.0.1 [A9]. The OpenAPI definitions are provided in YAML or JSON format.

The OpenAPI/YAML definitions are specified in 3GPP Forge, refer to clause 4.3 of TS 28.623 [44] for the Forge location. An example of Forge location is: "https://forge.3gpp.org/rep/sa5/MnS/-/tree/Tag\_Rel18\_SA104/".

Directory: OpenAPI

File: TS28532\_ProvMnS.yaml

File: TS28532\_PerfMnS.yam

File: TS28532\_HeartbeatNtf.yaml

File: TS28532\_StreamingDataMnS.yaml

File: TS28532\_FileDataReportingMnS.yaml

# A.1 Provisioning management service

## A.1.0 Introduction

Clause A.1.1 contains the OpenAPI definition of the provisioning MnS which includes the provisioning MnS operations and the provisioning MnS notifications.

Clause A.1.2 provides indications regarding the content of the generic provisioning MnS notifications when the consumer of these notifications supports the ONAP VES API. This content is sent as payload of VES events (see Annex B).

## A.1.1 OpenAPI document "TS28532\_ProvMnS.yaml"

Note that clause A.0 includes the location of TS28532\_ProvMnS.yaml.

## A.1.2 Integration with ONAP VES

Detailed guidelines for integration of provisioning MnS notifications with ONAP VES are provided in Annex B.

# A.2 Void

# A.3 Void

# A.4 Generic performance assurance management service

## A.4.1 Void

## A.4.2 OpenAPI document "TS28532\_PerfMnS.yaml"

Note that clause A.0 includes the location of TS28532\_PerfMnS.yaml.

## A.4.3 Integration with ONAP VES

Detailed guidelines for integration of performance assurance MnS notifications with ONAP VES are provided in Annex B.

# A.5 Heartbeat

### A.5.0 Introduction

Clause A.5.1 contains the OpenAPI definition of the heartbeat management capability.

Clause A.5.2 provides indications regarding the content of the heartbeat management capability notifications when the consumer of these notifications supports the ONAP VES API. This content is sent as payload of VES events (see Annex B).

## A.5.1 OpenAPI document "TS28532\_HeartbeatNtf.yaml"

Note that clause A.0 includes the location of TS28532\_HeartbeatNtf.yaml.

## A.5.2 Integration with ONAP VES

NOTE: Void.

Detailed guidelines for integration of heartbeat notifications with ONAP VES are provided in Annex B.

# A.6 Streaming data reporting management service

## A.6.1 Introduction

Clause A.6.2 contains the OpenAPI specification of the Streaming data reporting MnS.

## A.6.2 OpenAPI document "TS28532\_StreamingDataMnS.yaml"

Note that clause A.0 includes the location of TS28532\_StreamingDataMnS.yaml.

# A.7 File data reporting management service

## A.7.1 Introduction

Clause A.7.2 contains the OpenAPI definition of the File Data Reporting MnS.

Clause A.7.3 provides indications regarding the content of the File Data Reporting MnS notifications when the consumer of these notifications supports the ONAP VES API. This content is sent as payload of VES events (see Annex B).

## A.7.2 OpenAPI document "TS28532\_FileDataReportingMnS.yaml"

Note that clause A.0 includes the location of TS28532\_FileDataReportingMnS.yaml.

## A.7.3 Integration with ONAP VES

Detailed guidelines for integration of file data reporting MnS notifications with ONAP VES are provided in Annex B.

Annex B (Informative):  
Guidelines for the integration of 3GPP MnS notifications with ONAP VES

In case the consumer of the 3GPP MnS notifications specified in the present document is an ONAP VES collector, the following guidelines are for the developer of the corresponding notification producer:

- The produced notification conforms to ONAP-defined VES specification;

- The VES Common Event Header fields are populated by the producer is as follows:

- The domain "stndDefined" is used,

- The "stndDefinedNamespace" field value is the concatenation of "3GPP-" and the name of the 3GPP MnS which the 3GPP IS notification is part of. Based on the MnS names defined in the present version of this document, VES name space values corresponding to 3GPP MnS could be:

- "3GPP-Provisioning",

- "3GPP-FaultSupervision",

- "3GPP-PerformanceAssurance",

- "3GPP-Heartbeat",

- "3GPP-DataStreamingReporting",

- "3GPP-DataFileReporting".

- How the other fields of the Common Event Header are populated is not in the scope of the present document;

- The payload part of the VES event specification conforms to the OpenAPI definitions of clause A.1.1 (for provisioning MnS notifications), A.2.1 (for the fault supervision MnS notifications), A4.2 (for the performance assurance MnS notifications), A.5.1 (for the heartbeat notifications) and A.7.2 (for the file data reporting MnS notifications) of the present document. The OpenAPI definitions of Annex A in the present document may also be found on 3GPP FORGE (see [53]).

A picture containing text, screenshot, line, font

Description automatically generated

Figure B-1: 3GPP MnS notifications consumed by ONAP VES Collector(s)

Annex C (informative):  
Change history

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Change history** | | | | | | | |
| **Date** | **Meeting** | **TDoc** | **CR** | **Rev** | **Cat** | **Subject/Comment** | **New version** |
| 2018-09 | SA#81 |  |  |  |  | Upgrade to change control version | 15.0.0 |
| 2018-09 | SA#81 |  |  |  |  | EditHelp editorial fix | 15.0.1 |
| 2018-12 | SA#82 | SP-181042 | 0002 | 1 | F | Correction of references | 15.1.0 |
| 2018-12 | SA#82 | SP-181042 | 0003 | 1 | F | Align with 3GPP draft rules of the usage of must | 15.1.0 |
| 2018-12 | SA#82 | SP-181042 | 0004 | 1 | F | Correction of the numbering and title of figures and tables | 15.1.0 |
| 2018-12 | SA#82 | SP-181042 | 0005 | 1 | F | Remove unnecessary Editor’s Note and figure | 15.1.0 |
| 2018-12 | SA#82 | SP-181045 | 0006 | 1 | F | Update Resource URI of alarmCount | 15.1.0 |
| 2018-12 | SA#82 | SP-181045 | 0009 | 1 | F | Change the name of IRPAgent and IRPManager | 15.1.0 |
| 2018-12 | SA#82 | SP-181045 | 0010 | 1 | F | Remove unnecessary import table and state diagram | 15.1.0 |
| 2018-12 | SA#82 | SP-181045 | 0012 | - | F | Correct the subscription resource related errors | 15.1.0 |
| 2018-12 | SA#82 | SP-181043 | 0018 | - | F | Add notifyNewSecurityAlarm to notification type | 15.1.0 |
| 2018-12 | SA#82 | SP-181045 | 0020 | 1 | F | Change alarmIRP to FaultSupervision MnS producer | 15.1.0 |
| 2018-12 | SA#82 | SP-181042 | 0021 | 1 | F | Add stage 2 definition for provisioning management service related notifications | 15.1.0 |
| 2018-12 | SA#82 | SP-181042 | 0022 | 1 | F | Correct stage 3 description of the Provisioning Management Service | 15.1.0 |
| 2018-12 | SA#82 | SP-181045 | 0025 | - | F | Correct erroneous reference to notification header | 15.1.0 |
| 2019-03 | SA#83 | SP-190120 | 0029 | 1 | F | Correction of references | 15.2.0 |
| 2019-06 | SA#84 | SP-190372 | 0031 | 2 | B | Add RESTful HTTP-based solution set of fault supervision for integration with ONAP VES | 16.0.0 |
| 2019-06 | SA#84 | SP-190371 | 0038 | 1 | B | Add performance threshold crossing notification | 16.0.0 |
| 2019-09 | SA#85 | SP-190742 | 0038A |  |  | Global reorganization, correcting operation names, notification parameter and wrong references | 16.1.0 |
| 2019-12 | SA#86 | SP-191178 | 0055 | 1 | B | RESTful CM notifications for integration with ONAP VES | 16.2.0 |
| 2019-12 | SA#86 | SP-191219 | 0059 | 1 | A | Corrections to provisioning MnS notification definitions (Stage 2) | 16.2.0 |
| 2019-12 | SA#86 | SP-191219 | 0061 | 2 | A | Correct fault supervision management service | 16.2.0 |
| 2019-12 | SA#86 | SP-191159 | 0069 | 2 | C | Make scoping and filtering optional in the ProvMnS | 16.2.0 |
| 2019-12 | SA#86 | SP-191159 | 0071 | 2 | F | Correct and update the RESTful HTTP-based solution set of provisioning | 16.2.0 |
| 2019-12 | SA#86 | SP-191178 | 0073 | 2 | B | Introduce Heartbeat | 16.2.0 |
| 2019-12 | SA#86 | SP-191173 | 0075 | 1 | A | Correct event time defn | 16.2.0 |
| 2019-12 | SA#86 | SP-191166 | 0076 | 1 | B | Add notifyEvent | 16.2.0 |
| 2019-12 | SA#86 | SP-191159 | 0081 | 1 | F | Correct schema to reflect location in the specifications | 16.2.0 |
| 2019-12 | SA#86 | SP-191159 | 0082 | - | F | Correct XML Schema for consistency and clarity | 16.2.0 |
| 2020-03 | SA#87E | SP-200174 | 0089 | - | A | Add missing definition for matching-criteria-attributes | 16.3.0 |
| 2020-03 | SA#87E | SP-200166 | 0092 | 1 | F | Clarify capability of ack alarms and filter constraint | 16.3.0 |
| 2020-03 | SA#87E | SP-200176 | 0094 | 1 | F | Correction of MnS Stage 3 solution sets for integration with ONAP VES | 16.3.0 |
| 2020-03 | SA#87E | SP-200166 | 0096 | - | F | Rapporteur clean up | 16.3.0 |
| 2020-03 | SA#87E | SP-200169 | 0098 | 1 | B | YANG\_Netconf Operations | 16.3.0 |
| 2020-03 | SA#87E | SP-200166 | 0101 | 1 | F | Clarify and add numerous issues in the REST SS of the ProvMnS | 16.3.0 |
| 2020-03 | SA#87E | SP-200166 | 0103 | 2 | F | Correct OpenAPI definition of the ProvMnS | 16.3.0 |
| 2020-03 | SA#87E | SP-200174 | 0104 | - | A | Correct ackState attribute name | 16.3.0 |
| 2020-03 | SA#87E | SP-200169 | 0105 | - | F | Correct Heartbeat | 16.3.0 |
| 2020-06 | SA#88-e | SP-200484 | 0100 | 2 | B | Add summary CM notification to the ProvMnS | 16.4.0 |
| 2020-06 | SA#88-e | SP-200484 | 0102 | 1 | F | Remove subscribe and unsubscribe operation from ProvMnS | 16.4.0 |
| 2020-06 | SA#88-e | SP-200484 | 0107 | 1 | F | Void meaningless clauses 12.1.2.2.1.2 and 12.2.2.2.1.2 | 16.4.0 |
| 2020-06 | SA#88-e | SP-200484 | 0111 | - | F | Add missing callbacks for notifications to ProvMnS | 16.4.0 |
| 2020-06 | SA#88-e | SP-200484 | 0113 | - | F | Remove attribute referenceObjectInstance which is not supported by solution set | 16.4.0 |
| 2020-06 | SA#88-e | SP-200485 | 0114 | 2 | F | Update URI for generic fault supervision management service | 16.4.0 |
| 2020-06 | SA#88-e | SP-200485 | 0115 | 2 | F | Update URI for performance data file reporting management service | 16.4.0 |
| 2020-06 | SA#88-e | SP-200484 | 0116 | - | F | Remove data object from response types in the ProvMnS | 16.4.0 |
| 2020-06 | SA#88-e | SP-200483 | 0117 | 3 | B | Add streaming trace data reporting service stage 2 definition | 16.4.0 |
| 2020-06 | SA#88-e | SP-200483 | 0118 | 2 | B | Add streaming data reporting service stage 3 mapping of operations | 16.4.0 |
| 2020-06 | SA#88-e | SP-200483 | 0119 | 2 | B |  | 16.4.0 |
| 2020-06 | SA#88-e | SP-200483 | 0120 | 2 | B |  | 16.4.0 |
| 2020-06 | SA#88-e | SP-200483 | 0121 | 2 | B | Add streaming data reporting service stage 3 OpenAPI definition | 16.4.0 |
| 2020-06 | SA#88-e | SP-200499 | 0123 | - | A | Move XML file format from stage2 to stage3 | 16.4.0 |
| 2020-06 | SA#88-e | SP-200485 | 0126 | 1 | C | Update Fault Supervision MnS (stage 2) | 16.4.0 |
| 2020-06 | SA#88-e | SP-200485 | 0127 | 1 | C | Update Fault Supervision MnS (REST SS) | 16.4.0 |
| 2020-06 | SA#88-e | SP-200485 | 0128 | 1 | C | Update Fault Supervision MnS (OpenAPI definitions) | 16.4.0 |
| 2020-06 | SA#88-e | SP-200500 | 0133 | - | F | Correction of ONAP references | 16.4.0 |
| 2020-06 | SA#88-e | SP-200611 | 0134 | 1 | F | Convert JSON schema to YAML file for performance threshold monitoring service | 16.4.0 |
| 2020-09 | SA#89e | SP-200738 | 0135 | - | F | Change stage2 definition for performance data file report MnS to generic file data report MnS | 16.5.0 |
| 2020-09 | SA#89e | SP-200738 | 0136 | - | F | Change RESTFUL definition for performance data file report MnS to generic file data report MnS | 16.5.0 |
| 2020-09 | SA#89e | SP-200724 | 0137 | - | F | Change openAPI definition for performance data file report MnS to generic file data report MnS | 16.5.0 |
| 2020-09 | SA#89e | SP-200737 | 0138 | 1 | F | Clarification on Annex A.1, A.2 and A.5 | 16.5.0 |
| 2020-09 | SA#89e | SP-200723 | 0139 | - | F | Update URI for streamingDataReportingMnS to aligh with URI structure defined in 32.158 | 16.5.0 |
| 2020-09 | SA#89e | SP-200736 | 0141 | 1 | A | Correct the description for generic provisioning MnS | 16.5.0 |
| 2020-09 | SA#89e | SP-200724 | 0143 | - | F | Correct various smaller errors (e.g. validation errors) in faultMnS.yaml (OpenAPI definitions) | 16.5.0 |
| 2020-09 | SA#89e | SP-200724 | 0144 | - | F | Correct definition of ThresholdLevelInd (REST SS) | 16.5.0 |
| 2020-09 | SA#89e | SP-200737 | 0147 | - | F | Remove unintended normative statement from informative clause | 16.5.0 |
| 2020-09 | SA#89e |  |  |  |  | Correction of clause numbering | 16.5.1 |
| 2020-11 |  |  |  |  |  | Cleanup of custom XML, watermarks, hidden text, etc.. no technical changes | 16.5.2 |
| 2020-12 | SA#90e | SP-201050 | 0148 | 1 | F | Correction on generic file data report MnS | 16.6.0 |
| 2020-12 | SA#90e | SP-201088 | 0149 | 2 | F | Update generic streaming MnS | 16.6.0 |
| 2020-12 | SA#90e | SP-201050 | 0150 | 1 | F | Correct CR implementation errors (Fault MnS) | 16.6.0 |
| 2020-12 | SA#90e | SP-201050 | 0152 | 1 | F | Correct ThresholdLevelInd (REST SS, OpenAPI definition) | 16.6.0 |
| 2020-12 | SA#90e | SP-201054 | 0153 | - | F | Correct notifyThresholdCrossing (stage 2) | 16.6.0 |
| 2020-12 | SA#90e | SP-201050 | 0154 | 1 | F | Correct notifyThresholdCrossing (REST SS, OpenAPI definition) | 16.6.0 |
| 2020-12 | SA#90e | SP-201050 | 0155 | 1 | F | Correct notifyHeartbeat (stage 2, REST SS, OpenAPI definition) | 16.6.0 |
| 2020-12 | SA#90e | SP-201050 | 0156 | - | F | Correct small errors in faultMnS.yaml (OpenAPI definition) | 16.6.0 |
| 2020-12 | SA#90e | SP-201050 | 0157 | 1 | F | Correct notifyChangedAlarmGeneral (stage 2) | 16.6.0 |
| 2020-12 | SA#90e | SP-201050 | 0158 | - | F | Correct notifyChangedAlarmGeneral (REST SS, OpenAPI definitions) | 16.6.0 |
| 2020-12 | SA#90e | SP-201055 | 0160 | 1 | F | Fix inconsistencies in guidelines for integration with ONAP VES | 16.6.0 |
| 2020-12 | SA#90e | SP-201088 | 0161 | - | F | Correct small errors in the Fault MnS (REST SS) | 16.6.0 |
| 2020-12 | SA#90e | SP-201088 | 0162 | - | F | Align ProvMnS data type names to UpperCamel (REST SS, OpenAPI definition) | 16.6.0 |
| 2021-03 | SA#91e | SP-210150 | 0163 | 2 | F |  | 16.7.0 |
| 2021-03 | SA#91e | SP-210150 | 0164 | 2 | F | Correct definitions for the File MnS (REST SS) | 16.7.0 |
| 2021-03 | SA#91e | SP-210150 | 0165 | 2 | F | Correct definitions for the File MnS (OpenAPI definitions) | 16.7.0 |
| 2021-03 | SA#91e | SP-210150 | 0166 | 1 | F | Correct support qualifiers of the notifyThresholdCrossing parameters (stage 2) | 16.7.0 |
| 2021-03 | SA#91e | SP-210146 | 0167 | - | F | Fix compilation errors | 16.7.0 |
| 2021-03 | SA#91e | SP-210146 | 0168 | 1 | F | Correct the misalignment information for stage2 Fault Supervision MnS | 16.7.0 |
| 2021-03 | SA#91e | SP-210146 | 0170 | 1 | F | Correct some minor errors in the Fault MnS definition (REST SS) | 16.7.0 |
| 2021-03 | SA#91e | SP-210146 | 0171 | - | F | Correct some minor errors in the Prov MnS definition (REST SS) | 16.7.0 |
| 2021-04 | SA#91e |  |  |  |  | Editorial cleanup with the help of the Rapporteur | 16.7.1 |
| 2021-06 | SA#92e | SP-210406 | 0173 | 1 | F |  | 16.8.0 |
| 2021-06 | SA#92e | SP-210406 | 0174 | 1 | F | Correct definitions for file management (stage 2, REST SS, OpenAPI definition) | 16.8.0 |
| 2021-06 | SA#92e | SP-210416 | 0175 | - | F | Align different (abbreviated) names for support qualifier to S | 16.8.0 |
| 2021-06 | SA#92e | SP-210406 | 0176 | 1 | F | Update clause 11.2.2 Managed information for fault supervision management service | 16.8.0 |
| 2021-06 | SA#92e |  |  |  |  | Editorial fix: format of tables | 16.8.1 |
| 2021-09 | SA#93e | SP-210885 | 0178 | 1 | F | Remove last occurrences of “-Type” in data type names | 16.9.0 |
| 2021-09 | SA#93e | SP-210885 | 0179 | 1 | F | Correct definition of the timeTick parameter in the File MnS | 16.9.0 |
| 2021-09 | SA#93e | SP-210885 | 0180 | 1 | F | Alignment the description for streaming data reporting MnS producer | 16.9.0 |
| 2021-09 | SA#93e | SP-210885 | 0185 | - | F | Add missing reference for TS 32.404 and RFC 6901 | 16.9.0 |
| 2021-12 | SA#94e | SP-211454 | 0187 | 1 | F | Align the description for generic provisioning MnS | 16.10.0 |
| 2021-12 | SA#94e | SP-211454 | 0188 | - | F | Fix the incorrect reference of Generic fault supervision management service to TS 32.158 | 16.10.0 |
| 2021-12 | SA#94e | SP-211454 | 0189 | - | F | Fix the incorrect reference of File data reporting service to TS 32.158 | 16.10.0 |
| 2021-12 | SA#94e | SP-211454 | 0190 | 1 | F | Fix the URI description for streaming data report MnS | 16.10.0 |
| 2021-12 | SA#94e | SP-211454 | 0193 | 1 | F | Correct spelling of notifyAlarmListRebuilt | 16.10.0 |
| 2022-03 | SA#95e | SP-220183 | 0196 | 1 | B | Add jobId to FileInfo | 17.0.0 |
| 2022-06 | SA#96 | SP-220497 | 0200 | - | A | Correct REST SS of deleteMOI | 17.1.0 |
| 2022-06 | SA#96 | SP-220497 | 0201 | - | F | Align allowed file transfer protocols in stage 2 with stage 1 requirements | 17.1.0 |
| 2022-06 | SA#96 | SP-200502 | 0202 | - | B | Update proMnS yaml file to include the resources-intentNrm | 17.1.0 |
| 2022-06 | SA#96 | SP-220497 | 0205 | - | A | OpenAPI file name and dependence change- part1 | 17.1.0 |
| 2022-06 | SA#96 | SP-220497 | 0206 | - | A | OpenAPI file name and dependence change- part2 | 17.1.0 |
| 2022-06 | SA#96 | SP-220497 | 0208 | 1 | A | Correct definition of Resource | 17.1.0 |
| 2022-06 | SA#96 | SP-220564 | 0209 | 1 | F | Correct notifyMOIChanges (stage 2) | 17.1.0 |
| 2022-06 | SA#96 | SP-220564 | 0210 | 1 | F | Correct notifyMOIChanges (REST SS) | 17.1.0 |
| 2022-06 | SA#96 | SP-220564 | 0211 | 1 | F | Correct notifyMOIChanges (OpenAPI definitions) | 17.1.0 |
| 2022-06 | SA#96 | SP-220564 | 0213 | 1 | B | Data change notifications YANG-in-Rest format | 17.1.0 |
| 2022-06 | SA#96 | SP-220497 | 0216 | - | A | Fix FileDataType definition in OpenAPI | 17.1.0 |
| 2022-06 | SA#96 |  |  |  |  | CR implementation corrections | 17.1.1 |
| 2022-09 | SA#97e | SP-220849 | 0219 | - | F | Updating Hysteresis from M to O in notifyThresholdCrossing | 17.2.0 |
| 2022-09 | SA#97e | SP-220858 | 0221 | - | A | Update provMnS yaml to include resources-coslaNrm | 17.2.0 |
| 2022-09 | SA#97e | SP-220851 | 0222 | - | F | Update provMnS yaml to include MDA NRM related resources | 17.2.0 |
| 2022-09 | SA#97e | SP-220859 | 0223 | - | F | Correct notifyMOIChanges handling for YANG leaf-lists | 17.2.0 |
| 2022-09 | SA#97e |  |  |  |  | Annex A.1.1 aligned with FORGE content | 17.2.1 |
| 2022-12 | SA#98e | SP-221169 | 0227 | 1 | A | Correct OpenAPI definition of HTTP DELETE | 17.3.0 |
| 2022-12 | SA#98e | SP-221169 | 0229 | 1 | A | Correct type of observedValue attribute | 17.3.0 |
| 2022-12 | SA#98e | SP-221169 | 0231 | 1 | A | Correct definition of the HTTP GET response | 17.3.0 |
| 2022-12 | SA#98e | SP-221169 | 0233 | 2 | A | Add missing definition of the JSON Patch document | 17.3.0 |
| 2022-12 | SA#98e | SP-221169 | 0235 | - | A | Remove duplicated message flows (REST SS of ProvMnS) | 17.3.0 |
| 2022-12 | SA#98e | SP-221169 | 0237 | 2 | A | Add introduction clause to the Prov MnS definition | 17.3.0 |
| 2022-12 | SA#98e | SP-221167 | 0238 | 1 | F | Add missing insert attribute to the data type MoiChange | 17.3.0 |
| 2022-12 | SA#98e | SP-221167 | 0239 | - | F | Clarify allowed values for href parameter in notifyMOIChanges (NETCONF/YANG) | 17.3.0 |
| 2023-03 | SA#99 | SP-230199 | 0241 | - | A | Align media type names with TS 32.158 | 17.4.0 |
| 2023-03 | SA#99 | SP-230199 | 0243 | 1 | A | Add examples for notifyMOICreation, notifyMOIDeletion and notifyAttributeValueChanges | 17.4.0 |
| 2023-03 | SA#99 | SP-230196 | 0244 | 1 | F | Clarify definitions related to attributes | 17.4.0 |
| 2023-03 | SA#99 | SP-230200 | 0245 | - | A | Updates for generic management services | 17.4.0 |
| 2023-06 | SA#100 | SP-230648 | 0249 | 1 | A | Netconf with-defaults | 17.5.0 |
| 2023-06 | SA#100 | SP-230648 | 0253 | - | A | Add missing definition of the JSON Patch document | 17.5.0 |
| 2023-06 | SA#100 | SP-230681 | 0255 | 1 | A | Correction the Information Type for objectClass and objectInstance | 17.5.0 |
| 2023-06 | SA#100 | SP-230649 | 0256 | - | F | Resources-edgeNrm is missing in resource schema | 17.5.0 |
| 2023-06 | SA#100 | SP-230648 | 0258 | - | A | Correct media types used with HTTP Patch | 17.5.0 |
| 2023-06 | SA#100 | SP-230648 | 0260 | - | A | Clarification on notification target | 17.5.0 |
| 2023-06 | SA#100 | SP-230647 | 0263 | - | A | Correction of RFC references, and alarm information | 17.5.0 |
| 2023-06 | SA#100 |  |  |  |  | Adding code files to the zip | 17.5.1 |
| 2023-07 | SA#100 |  |  |  |  | Fixing header and footer | 17.5.2 |
| 2023-09 | SA#101 | SP-230944 | 0265 | - | F | Make probableCause mandatory in notifyChangedAlarmGeneral | 17.6.0 |
| 2023-09 | SA#101 | SP-230940 | 0267 | 1 | A | Correction to ProvMnS stage3 issue concerning parameter attributes | 17.6.0 |
| 2023-09 | SA#101 | SP-230940 | 0270 | - | A | Clarify complete attribute values must be included in notifyMOIAttributeValueChanges | 17.6.0 |
| 2023-09 | SA#101 | SP-230940 | 0272 | - | A | Clarify usage of the attributes container in notifyMOIChanges | 17.6.0 |
| 2023-09 | SA#101 | SP-230940 | 0280 | 1 | A | Correction of reference to Forge OpenAPI definition | 17.6.0 |
| 2023-09 | SA#101 | SP-230940 | 0282 | - | A | Clarify description of generic provisioning service | 17.6.0 |
| 2023-09 | SA#101 | SP-230960 | 0268 | - | C |  | 18.0.0 |
| 2023-12 | SA#102 | SP-231487 | 0285 | - | A | Correction to eventTime description for NotifyMoiDeletion & NotifyMoiAttributeValueChanges | 18.1.0 |
| 2023-12 | SA#102 | SP-231472 | 0293 | 1 | C | Update definition of createMOI (stage 2) | 18.1.0 |
| 2023-12 | SA#102 | SP-231472 | 0294 | 1 | C | Update definition of modifyMOIAttributes | 18.1.0 |
| 2023-12 | SA#102 | SP-231472 | 0295 | 1 | B | Add definition of changeMOIs (stage 2) | 18.1.0 |
| 2023-12 | SA#102 | SP-231494 | 0296 | - | D | Editorial Correction – Not implemented due to clash with 0294 | 18.1.0 |
| 2023-12 | SA#102 | SP-231487 | 0297 | 1 | A | Clarify MnS capability definitions | 18.1.0 |
| 2023-12 | SA#102 | SP-231485 | 0299 | - | B | Add resources-RanScNrm as Resource for provisioning MnS | 18.1.0 |
| 2023-12 | SA#102 | SP-231490 | 0300 | - | A | Clarify streaming data reporting service definitions | 18.1.0 |
| 2023-12 | SA#102 | SP-231458 | 0301 | - | F | Update the reference for TraceJob in StreamingDataReport MnS | 18.1.0 |
| 2024-03 | SA#103 | SP-240185 | 0304 |  | A |  | 18.2.0 |
| 2024-03 | SA#103 | SP-240168 | 0305 |  | B |  | 18.2.0 |
| 2024-03 | SA#103 | SP-240168 | 0306 | 1 | B |  | 18.2.0 |
| 2024-03 | SA#103 | SP-240168 | 0307 | 1 | B | Add dataNodeSelector to getMOIAttributes (NETCONF) | 18.2.0 |
| 2024-03 | SA#103 | SP-240168 | 0308 | 1 | B | Add stage 3 definition of changeMOIs (REST SS) | 18.2.0 |
| 2024-03 | SA#103 | SP-240168 | 0309 | 1 | C | Remove-Update FM related parts | 18.2.0 |
| 2024-03 | SA#103 | SP-240168 | 0310 |  | B | Add new HTTP error response format (OpenAPI) | 18.2.0 |
| 2024-03 | SA#103 | SP-240168 | 0311 |  | B | Add dataNodeSelector to getMOIAttributes (OpenAPI) | 18.2.0 |
| 2024-03 | SA#103 | SP-240395 | 0312 | 1 | F | Add resource-nrm for control NRM in provisioning MnS | 18.2.0 |
| 2024-03 | SA#103 | SP-240174 | 0313 | 1 | B | resources-msacNrm is missing in resource schema | 18.2.0 |
| 2024-03 | SA#103 | SP-240168 | 0314 | 1 | F | Clarify for each CM notification type the allowed targets of notification subscriptions | 18.2.0 |
| 2024-03 | SA#103 | SP-240185 | 0319 |  | A | Correction of attribute description | 18.2.0 |
| 2024-06 | SA#104 | SP-240820 | 0323 |  | F | Rel-18 CR TS 28.532 add missing resource-NRM for fault mangement in provisioning MnS | 18.3.0 |
| 2024-06 | SA#104 | SP-240803 | 0325 |  | A | Rel-18 CR 28.532 Fix inconsistent streaming data reporting service input parameter | 18.3.0 |
| 2024-06 | SA#104 | SP-240820 | 0328 |  | F | TS28.532 Rel18 corrections to remove reference to TS28532\_FaultMnS.yaml | 18.3.0 |
| 2024-06 | SA#104 | SP-240820 | 0329 | 1 | F | Rel-18 CR TS 28.532 Correct Missing reference to TS 28.811 | 18.3.0 |
| 2024-06 | SA#104 | SP-240808 | 0330 | 1 | F | TS28.532 Rel18 Moving normative stage3 to Forge | 18.3.0 |
| 2024-06 | SA#104 | SP-240820 | 0331 |  | F | Rel-18 CR 28.532 Add reference to the new Fault Management specification  NOT IMPLEMENTED AS NOT COMPLIANT WITH DRAFTING TULES | 18.3.0 |
| 2024-06 | SA#104 | SP-240803 | 0333 |  | A | Rel-18 CR 28.532 Correct notifyMOIChanges YANG mapping | 18.3.0 |
| 2024-09 | SA#105 | SP-241171 | 0339 | 1 | A | Rel-18 CR TS 28.532 Correcting the TLS component in the protocol stack diagram | 18.4.0 |