

ETSI TS 129 122 V15.6.0 (2020-01)



Universal Mobile Telecommunications System (UMTS);

LTE;

5G;

**T8 reference point for Northbound APIs
(3GPP TS 29.122 version 15.6.0 Release 15)**



Reference

RTS/TSGC-0329122vf60

Keywords

5G,LTE,UMTS

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

The present document can be downloaded from:
<http://www.etsi.org/standards-search>

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at <https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx>

If you find errors in the present document, please send your comment to one of the following services:
<https://portal.etsi.org/People/CommiteeSupportStaff.aspx>

Copyright Notification

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI.

The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2020.

All rights reserved.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

oneM2M™ logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners.

GSM® and the GSM logo are trademarks registered and owned by the GSM Association.

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: "*Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards*", which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (<https://ipr.etsi.org/>).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

Legal Notice

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

The present document may refer to technical specifications or reports using their 3GPP identities. These shall be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between 3GPP and ETSI identities can be found under <http://webapp.etsi.org/key/queryform.asp>.

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the [ETSI Drafting Rules](#) (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Contents

Intellectual Property Rights	2
Legal Notice	2
Modal verbs terminology.....	2
Foreword.....	16
1 Scope	17
2 References	17
3 Definitions and abbreviations.....	19
3.1 Definitions	19
3.2 Abbreviations	19
4 T8 reference point	20
4.1 Overview	20
4.2 Reference model.....	20
4.3 Functional elements.....	21
4.3.1 SCEF.....	21
4.3.2 SCS/AS	21
4.4 Procedures over T8 reference point.....	21
4.4.1 Introduction.....	21
4.4.2 Monitoring Procedures	21
4.4.2.1 General	21
4.4.2.2 Monitoring Events Configuration	21
4.4.2.2.1 General	21
4.4.2.2.2 Monitoring Events Configuration via HSS.....	23
4.4.2.2.2.1 General.....	23
4.4.2.2.2.2 Configuration Request for an individual UE	23
4.4.2.2.2.3 Configuration Request for a group of UEs	23
4.4.2.2.3 Monitoring Events Configuration directly via MME/SGSN	24
4.4.2.2.4 Monitoring Events Configuration via PCRF	24
4.4.2.2.4.1 General.....	24
4.4.2.2.4.2 Configuration Request for an individual UE	25
4.4.2.2.4.3 Configuration Request for a group of UEs	25
4.4.2.3 Reporting of Monitoring Event Procedure	25
4.4.2.4 Network-initiated Explicit Monitoring Event Deletion Procedure.....	26
4.4.3 Procedures for resource management of Background Data Transfer.....	26
4.4.4 Procedures for changing the chargeable party at session set up or during the session.....	27
4.4.5 Procedures for Non-IP Data Delivery	28
4.4.5.1 General	28
4.4.5.2 NIDD Configuration	28
4.4.5.2.1 NIDD Configuration for a single UE.....	28
4.4.5.2.2 NIDD Configuration for a group of UEs	29
4.4.5.3 Mobile Terminated NIDD procedure	29
4.4.5.3.1 Mobile Terminated NIDD for a single UE	29
4.4.5.3.2 Mobile Terminated NIDD for a group of UEs	31
4.4.5.4 Mobile Originated NIDD procedure	31
4.4.5.5 NIDD Authorisation Update procedure	31
4.4.6 Procedures for Device Triggering.....	32
4.4.7 Procedures for Group Message Delivery	33
4.4.7.1 General	33
4.4.7.2 Group Message Delivery via MBMS	33
4.4.7.2.1 General	33
4.4.7.2.2 Group Message Delivery via MBMS by MB2	33
4.4.7.2.2.1 TMGI Allocation	33
4.4.7.2.2.2 TMGI Deallocation.....	34
4.4.7.2.2.3 Creation of group message delivery.....	34
4.4.7.2.2.4 Modification of previous submitted group message delivery	34

4.4.7.2.2.5	Cancellation of previous submitted group message delivery	35
4.4.7.2.3	Group message Delivery via MBMS by xMB	35
4.4.7.2.3.1	Service Creation	35
4.4.7.2.3.2	Service Deletion	35
4.4.7.2.3.3	Creation of group message delivery	35
4.4.7.2.3.4	Modification of previous submitted group message delivery	36
4.4.7.2.3.5	Cancellation of previous submitted group message delivery	36
4.4.8	Procedures for Reporting of Network Status	37
4.4.8.1	General	37
4.4.8.2	Network Status Reporting Subscription	37
4.4.8.3	Network Status Reporting Notification	38
4.4.9	Procedures for Communication Pattern Parameters Provisioning	38
4.4.10	Procedures for PFD Management	39
4.4.11	Procedures for Enhanced Coverage Restriction Control	40
4.4.12	Procedures for Network Parameter Configuration	41
4.4.12.1	General	41
4.4.12.2	Configuration Request for an individual UE	41
4.4.12.3	Configuration Request for a group of UEs	42
4.4.13	Procedures for setting up an AS session with required QoS	42
4.4.14	Procedures for MSISDN-less Mobile Originated SMS	43
4.4.14.1	General	43
4.4.14.2	Delivery of MSISDN-less MO SMS	43
5	T8 APIs	44
5.1	Introduction	44
5.2	Information applicable to several APIs	44
5.2.1	Data Types	44
5.2.1.1	Introduction	44
5.2.1.2	Referenced structured data types	44
5.2.1.2.1	Type: SponsorInformation	44
5.2.1.2.2	Type: UsageThreshold	44
5.2.1.2.3	Type: TimeWindow	44
5.2.1.2.4	Type: Acknowledgement	45
5.2.1.2.5	Type: NotificationData	45
5.2.1.2.6	Type: EventReport	45
5.2.1.2.7	Type: AccumulatedUsage	45
5.2.1.2.8	Type: FlowInfo	45
5.2.1.2.9	Type: TestNotification	46
5.2.1.2.10	Type: WebsocketNotifConfig	46
5.2.1.2.11	Type: LocationArea	46
5.2.1.2.12	Type: ProblemDetails	47
5.2.1.2.13	Type: InvalidParam	47
5.2.1.2.14	Type: PlmnId	47
5.2.1.2.15	Type: ConfigResult	48
5.2.1.2.16	Type: UsageThresholdRm	48
5.2.1.2.17	Type: LocationArea5G	48
5.2.1.3	Referenced Simple data types and enumerations	48
5.2.1.3.1	Introduction	48
5.2.1.3.2	Simple data types	48
5.2.1.3.3	Enumeration: Event	50
5.2.1.3.4	Enumeration: ResultReason	50
5.2.1.4	Conventions for documenting structured data types	50
5.2.2	Usage of HTTP	51
5.2.2.1	General	51
5.2.2.2	Usage of the HTTP PATCH method	51
5.2.3	Content type	51
5.2.4	URI structure	51
5.2.5	Notifications	52
5.2.5.1	General	52
5.2.5.2	Notification Delivery using a separate HTTP connection	52
5.2.5.3	Notification Test Event	52
5.2.5.4	Notification Delivery using Websocket	52

5.2.6	Error handling	54
5.2.7	Feature negotiation	57
5.2.8	HTTP custom headers	57
5.2.8.1	General	57
5.2.8.2	Reused HTTP custom headers	57
5.2.9	Conventions for Open API specification files	57
5.2.9.1	General	57
5.2.9.2	Formatting of OpenAPI files	57
5.2.9.3	Structured data types	58
5.2.9.4	Info	59
5.2.9.5	Servers	59
5.2.9.6	References to other 3GPP-defined Open API specification files	59
5.2.9.7	Server-initiated communication	60
5.2.9.8	Describing the body of HTTP PATCH requests	60
5.2.9.9	Error Responses	61
5.2.9.10	Enumerations	62
5.2.9.11	Read only attribute	62
5.2.9.12	externalDocs	63
5.3	MonitoringEvent API	63
5.3.1	Overview	63
5.3.2	Data model	63
5.3.2.1	Resource data types	63
5.3.2.1.1	Introduction	63
5.3.2.1.2	Type: MonitoringEventSubscription	63
5.3.2.2	Notification data types	67
5.3.2.2.1	Introduction	67
5.3.2.2.2	Type: MonitoringNotification	67
5.3.2.3	Referenced structured data types	68
5.3.2.3.1	Introduction	68
5.3.2.3.2	Type: MonitoringEventReport	68
5.3.2.3.3	Type: IdleStatusInfo	71
5.3.2.3.4	Type: UePerLocationReport	71
5.3.2.3.5	Type: LocationInfo	72
5.3.2.3.6	Type: FailureCause	72
5.3.2.4	Referenced simple data types and enumerations	73
5.3.2.4.1	Introduction	73
5.3.2.4.2	Simple data types	73
5.3.2.4.3	Enumeration: MonitoringType	73
5.3.2.4.4	Enumeration: ReachabilityType	74
5.3.2.4.5	Enumeration: LocationType	74
5.3.2.4.6	Enumeration: AssociationType	75
5.3.2.4.7	Enumeration: Accuracy	75
5.3.3	Resource structure	75
5.3.3.1	General	75
5.3.3.2	Resource: Monitoring Event Subscriptions	76
5.3.3.2.1	Introduction	76
5.3.3.2.2	Resource definition	76
5.3.3.2.3	Resource methods	76
5.3.3.2.3.1	GET	76
5.3.3.2.3.2	PUT	77
5.3.3.2.3.3	PATCH	77
5.3.3.2.3.4	POST	77
5.3.3.2.3.5	DELETE	78
5.3.3.3	Resource: Individual Monitoring Event Subscription	78
5.3.3.3.1	Introduction	78
5.3.3.3.2	Resource definition	79
5.3.3.3.3	Resource methods	79
5.3.3.3.3.1	GET	79
5.3.3.3.3.2	PUT	79
5.3.3.3.3.3	PATCH	80
5.3.3.3.3.4	POST	80
5.3.3.3.3.5	DELETE	80

5.3.3.4	Monitoring Notification	81
5.3.3.4.1	Introduction	81
5.3.3.4.2	Resource definition.....	81
5.3.3.4.3	Resource methods.....	81
5.3.3.4.3.1	Notification via POST.....	81
5.3.3.4.3.2	Notification via WebSocket.....	81
5.3.4	Used Features.....	81
5.3.5	Error handling.....	82
5.3.5.1	General	82
5.3.5.2	Protocol Errors	82
5.3.5.3	Application Errors.....	82
5.4	ResourceManagementOfBdt API.....	83
5.4.1	Overview	83
5.4.2	Data model.....	83
5.4.2.1	Resource data types.....	83
5.4.2.1.1	Introduction	83
5.4.2.1.2	Type: Bdt.....	83
5.4.2.1.3	Type: BdtPatch	84
5.4.2.2	Referenced structured data types.....	84
5.4.2.2.1	Introduction	84
5.4.2.2.2	Type: TransferPolicy	84
5.4.3	Resource structure.....	85
5.4.3.1	General	85
5.4.3.2	Resource: BDT Subscriptions	85
5.4.3.2.1	Introduction	85
5.4.3.2.2	Resource definition.....	85
5.4.3.2.3	Resource methods.....	86
5.4.3.2.3.1	GET.....	86
5.4.3.2.3.2	PUT.....	86
5.4.3.2.3.3	PATCH	86
5.4.3.2.3.4	POST.....	86
5.4.3.2.3.5	DELETE	87
5.4.3.3	Resource: Individual BDT Subscription	87
5.4.3.3.1	Introduction	87
5.4.3.3.2	Resource definition.....	87
5.4.3.3.3	Resource methods.....	87
5.4.3.3.3.1	GET.....	87
5.4.3.3.3.2	PUT.....	88
5.4.3.3.3.3	PATCH	88
5.4.3.3.3.4	POST.....	89
5.4.3.3.3.5	DELETE	89
5.4.4	Used Features.....	89
5.5	ChargeableParty API.....	89
5.5.1	Overview	89
5.5.2	Data model.....	90
5.5.2.1	Resource data types.....	90
5.5.2.1.1	Introduction	90
5.5.2.1.2	Type: ChargeableParty	90
5.5.2.1.3	Type: ChargeablePartyPatch	91
5.5.3	Resource structure.....	92
5.5.3.1	General	92
5.5.3.2	Resource: Chargeable Party Transactions	92
5.5.3.2.1	Introduction	92
5.5.3.2.2	Resource definition.....	92
5.5.3.2.3	Resource methods.....	93
5.5.3.2.3.1	GET.....	93
5.5.3.2.3.2	PUT.....	93
5.5.3.2.3.3	PATCH	93
5.5.3.2.3.4	POST.....	93
5.5.3.2.3.5	DELETE	94
5.5.3.3	Resource: Individual Chargeable Party Transaction	94
5.5.3.3.1	Introduction	94

5.5.3.3.2	Resource definition.....	94
5.5.3.3.3	Resource methods.....	94
5.5.3.3.3.1	GET.....	94
5.5.3.3.3.2	PUT.....	95
5.5.3.3.3.3	PATCH.....	95
5.5.3.3.3.4	POST.....	95
5.5.3.3.3.5	DELETE.....	95
5.5.3.4	Event Notification.....	96
5.5.3.4.1	Introduction.....	96
5.5.3.4.2	Resource definition.....	96
5.5.3.4.3	Resource methods.....	96
5.5.3.4.3.1	Notification via HTTP POST.....	96
5.5.3.4.3.2	Notification via WebSocket.....	96
5.5.4	Used Features.....	97
5.6	NIDD API.....	97
5.6.1	Overview.....	97
5.6.2	Data model.....	97
5.6.2.1	Resource data types.....	97
5.6.2.1.1	Introduction.....	97
5.6.2.1.2	Type: NiddConfiguration.....	97
5.6.2.1.3	Type: NiddDownlinkDataTransfer.....	100
5.6.2.1.4	Type: NiddUplinkDataNotification.....	103
5.6.2.1.5	Type: NiddDownlinkDataDeliveryStatusNotification.....	103
5.6.2.1.6	Type: NiddConfigurationStatusNotifiation.....	104
5.6.2.1.7	Type: NiddConfigurationPatch.....	104
5.6.2.1.8	Type: GmdNiddDownlinkDataDeliveryStatusNotification.....	105
5.6.2.2	Referenced structured data types.....	105
5.6.2.2.1	Introduction.....	105
5.6.2.2.2	Type: RdsPort.....	105
5.6.2.2.3	Type: GmdResult.....	106
5.6.2.2.4	Type: NiddDownlinkDataTransferFailure.....	106
5.6.2.3	Referenced simple data types and enumerations.....	106
5.6.2.3.1	Introduction.....	106
5.6.2.3.2	Simple data types.....	106
5.6.2.3.3	Enumeration: PdnEstablishmentOptions.....	107
5.6.2.3.4	Enumeration: DeliveryStatus.....	107
5.6.2.3.5	Enumeration: NiddStatus.....	107
5.6.2.3.6	Enumeration: PdnEstablishmentOptionsRm.....	108
5.6.3	Resource structure.....	108
5.6.3.1	General.....	108
5.6.3.2	Resource: NIDD Configurations.....	109
5.6.3.2.1	Introduction.....	109
5.6.3.2.2	Resource definition.....	109
5.6.3.2.3	Resource methods.....	110
5.6.3.2.3.1	GET.....	110
5.6.3.2.3.2	PUT.....	110
5.6.3.2.3.3	PATCH.....	110
5.6.3.2.3.4	POST.....	110
5.6.3.2.3.5	DELETE.....	111
5.6.3.3	Resource: Individual NIDD Configuration.....	111
5.6.3.3.1	Introduction.....	111
5.6.3.3.2	Resource definition.....	111
5.6.3.3.3	Resource methods.....	111
5.6.3.3.3.1	GET.....	111
5.6.3.3.3.2	PUT.....	112
5.6.3.3.3.3	PATCH.....	112
5.6.3.3.3.4	POST.....	112
5.6.3.3.3.5	DELETE.....	112
5.6.3.4	Resource: NIDD downlink data deliveries.....	113
5.6.3.4.1	Introduction.....	113
5.6.3.4.2	Resource definition.....	113
5.6.3.4.3	Resource methods.....	113

5.6.3.4.3.1	GET.....	113
5.6.3.4.3.2	PUT.....	114
5.6.3.4.3.3	PATCH.....	114
5.6.3.4.3.4	POST.....	114
5.6.3.4.3.5	DELETE.....	114
5.6.3.5	Resource: Individual NIDD downlink data delivery.....	115
5.6.3.5.1	Introduction.....	115
5.6.3.5.2	Resource definition.....	115
5.6.3.5.3	Resource methods.....	115
5.6.3.5.3.1	GET.....	115
5.6.3.5.3.2	PUT.....	115
5.6.3.5.3.3	PATCH.....	116
5.6.3.5.3.4	POST.....	116
5.6.3.5.3.5	DELETE.....	116
5.6.3.6	NIDD Configuration Update Notification.....	117
5.6.3.6.1	Introduction.....	117
5.6.3.6.2	Resource definition.....	117
5.6.3.6.3	Resource methods.....	117
5.6.3.6.3.1	Notification via HTTP POST.....	117
5.6.3.6.4	Notification via WebSocket.....	118
5.6.3.7	NIDD Downlink Data Delivery Status Notification.....	118
5.6.3.7.1	Introduction.....	118
5.6.3.7.2	Resource definition.....	118
5.6.3.7.3	Resource methods.....	118
5.6.3.7.3.1	Notification via HTTP POST.....	118
5.6.3.7.4	Notification via WebSocket.....	119
5.6.3.8	NIDD Uplink Data Notification.....	119
5.6.3.8.1	Introduction.....	119
5.6.3.8.2	Resource definition.....	119
5.6.3.8.3	Resource methods.....	119
5.6.3.8.3.1	Notification via HTTP POST.....	119
5.6.3.8.4	Notification via WebSocket.....	120
5.6.4	Used Features.....	120
5.6.5	Error handling.....	120
5.6.5.1	General.....	120
5.6.5.2	Protocol Errors.....	120
5.6.5.3	Application Errors.....	121
5.7	DeviceTriggering API.....	121
5.7.1	Overview.....	121
5.7.2	Data model.....	121
5.7.2.1	Resource data types.....	121
5.7.2.1.1	Introduction.....	121
5.7.2.1.2	Type: DeviceTriggering.....	121
5.7.2.1.3	Type: DeviceTriggeringDeliveryReportNotification.....	122
5.7.2.2	Referenced simple data types and enumerations.....	123
5.7.2.2.1	Introduction.....	123
5.7.2.2.2	Simple data types.....	123
5.7.2.2.3	Enumeration: DeliveryResult.....	123
5.7.2.2.4	Enumeration: Priority.....	124
5.7.3	Resource structure.....	124
5.7.3.1	General.....	124
5.7.3.2	Resource: Device Triggering Transactions.....	125
5.7.3.2.1	Introduction.....	125
5.7.3.2.2	Resource definition.....	125
5.7.3.2.3	Resource methods.....	125
5.7.3.2.3.1	GET.....	125
5.7.3.2.3.2	PUT.....	126
5.7.3.2.3.3	PATCH.....	126
5.7.3.2.3.4	POST.....	126
5.7.3.2.3.5	DELETE.....	126
5.7.3.3	Resource: Individual Device Triggering Transaction.....	127
5.7.3.3.1	Introduction.....	127

5.7.3.3.2	Resource definition.....	127
5.7.3.3.3	Resource methods.....	127
5.7.3.3.3.1	GET.....	127
5.7.3.3.3.2	PUT.....	127
5.7.3.3.3.3	PATCH.....	128
5.7.3.3.3.4	POST.....	128
5.7.3.3.3.5	DELETE.....	128
5.7.3.4	Device Triggering Delivery Report Notification.....	128
5.7.3.4.1	Introduction.....	128
5.7.3.4.2	Resource definition.....	128
5.7.3.4.3	Resource methods.....	129
5.7.3.4.3.1	Notification via HTTP POST.....	129
5.7.3.4.3.2	Notification via WebSocket.....	129
5.7.4	Used Features.....	129
5.8	GMD via MBMS related APIs.....	130
5.8.1	Overview.....	130
5.8.2	GMD via MBMS by MB2 API.....	130
5.8.2.1	Data model.....	130
5.8.2.1.1	Resource data types.....	130
5.8.2.1.1.1	Introduction.....	130
5.8.2.1.1.2	Type: TMGIAllocation.....	130
5.8.2.1.1.3	Type: GMDViaMBMSByMb2.....	131
5.8.2.1.1.4	Type: GMDByMb2Notification.....	132
5.8.2.1.1.5	Type: TMGIAllocationPatch.....	132
5.8.2.1.1.6	Type: GMDViaMBMSByMb2Patch.....	132
5.8.2.1.1.7	Type: MbmsLocArea.....	132
5.8.2.2	Resource structure.....	133
5.8.2.2.1	General.....	133
5.8.2.2.2	Resource: TMGI Allocation.....	134
5.8.2.2.2.1	Introduction.....	134
5.8.2.2.2.2	Resource definition.....	134
5.8.2.2.2.3	Resource methods.....	135
5.8.2.2.2.3.1	GET.....	135
5.8.2.2.2.3.2	PUT.....	135
5.8.2.2.2.3.3	PATCH.....	135
5.8.2.2.2.3.4	POST.....	135
5.8.2.2.2.3.5	DELETE.....	135
5.8.2.2.3	Resource: Individual TMGI Allocation.....	136
5.8.2.2.3.1	Introduction.....	136
5.8.2.2.3.2	Resource definition.....	136
5.8.2.2.3.3	Resource methods.....	136
5.8.2.2.3.3.1	GET.....	136
5.8.2.2.3.3.2	PUT.....	136
5.8.2.2.3.3.3	PATCH.....	137
5.8.2.2.3.3.4	POST.....	137
5.8.2.2.3.3.5	DELETE.....	137
5.8.2.2.4	Resource: GMD via MBMS by MB2.....	137
5.8.2.2.4.1	Introduction.....	137
5.8.2.2.4.2	Resource definition.....	137
5.8.2.2.4.3	Resource methods.....	138
5.8.2.2.4.3.1	GET.....	138
5.8.2.2.4.3.2	PUT.....	138
5.8.2.2.4.3.3	PATCH.....	138
5.8.2.2.4.3.4	POST.....	138
5.8.2.2.4.3.5	DELETE.....	139
5.8.2.2.5	Resource: Individual GMD via MBMS by MB2.....	139
5.8.2.2.5.1	Introduction.....	139
5.8.2.2.5.2	Resource definition.....	139
5.8.2.2.5.3	Resource methods.....	139
5.8.2.2.5.3.1	GET.....	139
5.8.2.2.5.3.2	PUT.....	140
5.8.2.2.5.3.3	PATCH.....	140

5.8.2.2.5.3.4	POST	141
5.8.2.2.5.3.5	DELETE.....	141
5.8.2.2.6	Resource: GMD via MBMS by MB2 Notification.....	141
5.8.2.2.6.1	Introduction.....	141
5.8.2.2.6.2	Resource definition	141
5.8.2.2.6.3	Resource methods	142
5.8.2.2.6.3.1	Notification via POST	142
5.8.2.2.6.3.2	Notification via WebSocket	142
5.8.2.3	Used Features	142
5.8.3	GMDviaMBMSbyxMB API.....	143
5.8.3.1	Data model	143
5.8.3.1.1	Resource data types	143
5.8.3.1.1.1	Introduction.....	143
5.8.3.1.1.2	Type: ServiceCreation	143
5.8.3.1.1.3	Type: GMDViaMBMSByxMB	144
5.8.3.1.1.4	Type: GMDByxMBNotification.....	144
5.8.3.1.1.5	Type: GMDViaMBMSByxMBPatch	145
5.8.3.1.1.6	Type: MbmsLocArea.....	145
5.8.3.1.2	Referenced simple data types and enumerations	145
5.8.3.1.2.1	Introduction.....	145
5.8.3.1.2.2	Simple data types	145
5.8.3.1.2.3	Enumeration: ServiceAnnouncementMode	146
5.8.3.2	Resource structure.....	146
5.8.3.2.1	General	146
5.8.3.2.2	Resource: xMB Services	147
5.8.3.2.2.1	Introduction.....	147
5.8.3.2.2.2	Resource definition	147
5.8.3.2.2.3	Resource methods	148
5.8.3.2.2.3.1	GET	148
5.8.3.2.2.3.2	PUT	148
5.8.3.2.2.3.3	PATCH.....	148
5.8.3.2.2.3.4	POST	148
5.8.3.2.2.3.5	DELETE.....	148
5.8.3.2.3	Resource: Individual xMB Service.....	149
5.8.3.2.3.1	Introduction.....	149
5.8.3.2.3.2	Resource definition	149
5.8.3.2.3.3	Resource methods	149
5.8.3.2.3.3.1	GET	149
5.8.3.2.3.3.2	PUT	149
5.8.3.2.3.3.3	PATCH.....	149
5.8.3.2.3.3.4	POST	149
5.8.3.2.3.3.5	DELETE.....	149
5.8.3.2.4	Resource: GMD via MBMS by xMB.....	150
5.8.3.2.4.1	Introduction.....	150
5.8.3.2.4.2	Resource definition	150
5.8.3.2.4.3	Resource methods	150
5.8.3.2.4.3.1	GET	150
5.8.3.2.4.3.2	PUT	151
5.8.3.2.4.3.3	PATCH.....	151
5.8.3.2.4.3.4	POST	151
5.8.3.2.4.3.5	DELETE.....	151
5.8.3.2.5	Resource: Individual GMD via MBMS by xMB.....	151
5.8.3.2.5.1	Introduction.....	151
5.8.3.2.5.2	Resource definition	151
5.8.3.2.5.3	Resource methods	152
5.8.3.2.5.3.1	GET	152
5.8.3.2.5.3.2	PUT	152
5.8.3.2.5.3.3	PATCH.....	152
5.8.3.2.5.3.4	POST	153
5.8.3.2.5.3.5	DELETE.....	153
5.8.3.2.6	Resource: GMD via MBMS by xMB Notification.....	153
5.8.3.2.6.1	Introduction.....	153

5.8.3.2.6.2	Resource definition	153
5.8.3.2.6.3	Resource methods	154
5.8.3.2.6.3.1	Notification via POST	154
5.8.3.2.6.3.2	Notification via WebSocket	154
5.8.3.3	Used Features	154
5.9	ReportingNetworkStatus API	155
5.9.1	Overview	155
5.9.2	Data model	155
5.9.2.1	Resource data types	155
5.9.2.1.1	Introduction	155
5.9.2.1.2	Type: NetworkStatusReportingSubscription	155
5.9.2.2	Notification data types	156
5.9.2.2.1	Introduction	156
5.9.2.2.2	Type: NetworkStatusReportingNotification	156
5.9.2.3	Referenced simple data types and enumerations	157
5.9.2.3.1	Introduction	157
5.9.2.3.2	Simple data types	157
5.9.2.3.3	Enumeration: CongestionType	157
5.9.3	Resource structure	157
5.9.3.1	General	157
5.9.3.2	Resource: Network Status Reporting Subscriptions	158
5.9.3.2.1	Introduction	158
5.9.3.2.2	Resource definition	158
5.9.3.2.3	Resource methods	158
5.9.3.2.3.1	GET	158
5.9.3.2.3.2	PUT	159
5.9.3.2.3.3	PATCH	159
5.9.3.2.3.4	POST	159
5.9.3.2.3.5	DELETE	159
5.9.3.3	Resource: Individual Network Status Reporting Subscription	160
5.9.3.3.1	Introduction	160
5.9.3.3.2	Resource definition	160
5.9.3.3.3	Resource methods	160
5.9.3.3.3.1	GET	160
5.9.3.3.3.2	PUT	160
5.9.3.3.3.3	PATCH	161
5.9.3.3.3.4	POST	161
5.9.3.3.3.5	DELETE	161
5.9.3.4	Network Status Reporting Notification	161
5.9.3.4.1	Introduction	161
5.9.3.4.2	Resource definition	161
5.9.3.4.3	Resource methods	162
5.9.3.4.3.1	Notification via POST	162
5.9.3.4.3.2	Notification via WebSocket	162
5.9.4	Used Features	162
5.9.5	Error handling	163
5.9.5.1	General	163
5.9.5.2	Protocol Errors	163
5.9.5.3	Application Errors	163
5.10	CpProvisioning API	163
5.10.1	Overview	163
5.10.2	Data model	163
5.10.2.1	Resource data types	163
5.10.2.1.1	Introduction	163
5.10.2.1.2	Type: CpInfo	164
5.10.2.2	Referenced structured data types	164
5.10.2.2.1	Introduction	164
5.10.2.2.2	Type: CpParameterSet	164
5.10.2.2.3	Type: ScheduledCommunicationTime	165
5.10.2.2.4	Type: CpReport	165
5.10.2.3	Referenced simple data types and enumerations	166
5.10.2.3.1	Introduction	166

5.10.2.3.2	Simple data types.....	166
5.10.2.3.3	Enumeration: CommunicationIndicator	166
5.10.2.3.4	Enumeration: StationaryIndication	166
5.10.2.3.5	Enumeration: CpFailureCode	166
5.10.3	Resource structure.....	167
5.10.3.1	General	167
5.10.3.2	Resource: CP Provisioning Subscriptions	167
5.10.3.2.1	Introduction	167
5.10.3.2.2	Resource definition.....	168
5.10.3.2.3	Resource methods.....	168
5.10.3.2.3.1	GET.....	168
5.10.3.2.3.2	PUT.....	168
5.10.3.2.3.3	PATCH	168
5.10.3.2.3.4	POST.....	168
5.10.3.2.3.5	DELETE	169
5.10.3.3	Resource: Individual CP Provisioning Subscription	169
5.10.3.3.1	Introduction	169
5.10.3.3.2	Resource definition.....	169
5.10.3.3.3	Resource methods.....	169
5.10.3.3.3.1	GET.....	169
5.10.3.3.3.2	PUT.....	170
5.10.3.3.3.3	PATCH	170
5.10.3.3.3.4	POST.....	170
5.10.3.3.3.5	DELETE	170
5.10.3.4	Resource: Individual CP Set Provisioning	171
5.10.3.4.1	Introduction	171
5.10.3.4.2	Resource definition.....	171
5.10.3.4.3	Resource methods.....	171
5.10.3.4.3.1	GET.....	171
5.10.3.4.3.2	PUT.....	172
5.10.3.4.3.3	PATCH	172
5.10.3.4.3.4	POST.....	172
5.10.3.4.3.5	DELETE	172
5.10.4	Used Features.....	173
5.11	PfdManagement API	173
5.11.1	Overview	173
5.11.2	Data model.....	173
5.11.2.1	Resource data types.....	173
5.11.2.1.1	Introduction	173
5.11.2.1.2	Type: PfdManagement	173
5.11.2.1.3	Type: PfdData.....	174
5.11.2.1.4	Type: Pfd	175
5.11.2.1.5	Type: PfdReport	175
5.11.2.2	Referenced simple data types and enumerations.....	175
5.11.2.2.1	Introduction	175
5.11.2.2.2	Simple data types.....	175
5.11.2.2.3	Enumeration: FailureCode.....	176
5.11.3	Resource structure.....	176
5.11.3.1	General	176
5.11.3.2	Resource: PFD Management Transactions	177
5.11.3.2.1	Introduction	177
5.11.3.2.2	Resource definition.....	177
5.11.3.2.3	Resource methods.....	177
5.11.3.2.3.1	GET.....	177
5.11.3.2.3.2	PUT.....	178
5.11.3.2.3.3	POST.....	178
5.11.3.2.3.4	PATCH	178
5.11.3.2.3.5	DELETE	178
5.11.3.3	Resource: Individual PFD Management Transaction.....	179
5.11.3.3.1	Introduction	179
5.11.3.3.2	Resource definition.....	179
5.11.3.3.3	Resource methods.....	179