

**Open Service Access (OSA);
Application Programming Interface (API);
Part 3: Framework
(Parlay 5)**



iTeh STANDARD PREVIEW
(standards.iteh.ai)

Full standard:
<https://standards.iteh.ai/catalog/standards/sist/229e1f97-1c83-44c5-b1a2-98a2b70fc86a/etsi-es-203-915-3-v1.3.1-2008-04>



Reference

RES/TISPAN-01055-03-OSA

Keywords

API, IDL, OSA, UML

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

Individual copies of the present document can be downloaded from:

<http://www.etsi.org>

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

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

<http://portal.etsi.org/tb/status/status.asp>

If you find errors in the present document, please send your comment to one of the following services:

http://portal.etsi.org/chaicor/ETSI_support.asp

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.

© European Telecommunications Standards Institute 2008.

© The Parlay Group 2008.

All rights reserved.

DECTTM, PLUGTESTSTM, UMTSTM, TIPHONTM, the TIPHON logo and the ETSI logo are Trade Marks of ETSI registered for the benefit of its Members.

3GPPTM is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

Contents

Intellectual Property Rights	14
Foreword.....	14
1 Scope	15
2 References	15
3 Definitions and abbreviations.....	15
3.1 Definitions	15
3.2 Abbreviations	15
4 Overview of the Framework.....	16
5 The Base Interface Specification.....	17
5.1 Interface Specification Format	17
5.1.1 Interface Class	17
5.1.2 Method descriptions.....	18
5.1.3 Parameter descriptions.....	18
5.1.4 State Model.....	18
5.2 Base Interface.....	18
5.2.1 Interface Class IpInterface	18
5.3 Service Interfaces	18
5.3.1 Overview	18
5.4 Generic Service Interface	18
5.4.1 Interface Class IpService	18
5.4.1.1 Method setCallback()	19
5.4.1.2 Method setCallbackWithSessionID().....	19
6 Framework Access Session API.....	20
6.1 Sequence Diagrams	20
6.1.1 Trust and Security Management Sequence Diagrams	20
6.1.1.1 Initial Access.....	20
6.1.1.2 Framework Terminates Access	21
6.1.1.3 Application Terminates Access.....	22
6.1.1.4 Non-API level Authentication.....	23
6.1.1.5 API Level Authentication	24
6.2 Class Diagrams.....	26
6.3 Interface Classes.....	27
6.3.1 Trust and Security Management Interface Classes	27
6.3.1.1 Interface Class IpClientAPILevelAuthentication.....	27
6.3.1.1.1 Method <<deprecated>> authenticate().....	28
6.3.1.1.2 Method abortAuthentication()	28
6.3.1.1.3 Method authenticationSucceeded()	29
6.3.1.1.4 Method challenge().....	29
6.3.1.2 Interface Class IpClientAccess.....	30
6.3.1.2.1 Method terminateAccess().....	30
6.3.1.3 Interface Class IpInitial	31
6.3.1.3.1 Method <<deprecated>> initiateAuthentication()	31
6.3.1.3.2 Method initiateAuthenticationWithVersion().....	32
6.3.1.4 Interface Class IpAuthentication	33
6.3.1.4.1 Method requestAccess()	34
6.3.1.5 Interface Class IpAPILevelAuthentication	34
6.3.1.5.1 Method <<deprecated>> selectEncryptionMethod().....	35
6.3.1.5.2 Method <<deprecated>> authenticate().....	36
6.3.1.5.3 Method abortAuthentication()	36
6.3.1.5.4 Method authenticationSucceeded()	36
6.3.1.5.5 Method selectAuthenticationMechanism().....	37
6.3.1.5.6 Method challenge().....	37
6.3.1.6 Interface Class IpAccess	38

6.3.1.6.1	Method obtainInterface()	39
6.3.1.6.2	Method obtainInterfaceWithCallback()	39
6.3.1.6.3	Method <<deprecated>> endAccess()	40
6.3.1.6.4	Method listInterfaces()	40
6.3.1.6.5	Method <<deprecated>> releaseInterface()	40
6.3.1.6.6	Method selectSigningAlgorithm()	41
6.3.1.6.7	Method terminateAccess()	41
6.3.1.6.8	Method relinquishInterface()	42
6.4	State Transition Diagrams	42
6.4.1	Trust and Security Management State Transition Diagrams	43
6.4.1.1	State Transition Diagrams for IpInitial	43
6.4.1.2	State Transition Diagrams for IpAPILevelAuthentication	44
6.4.1.2.1	Idle State	44
6.4.1.2.2	Authenticating Framework State	45
6.4.1.2.3	Framework Authenticated State	45
6.4.1.2.4	Authenticating Client State	45
6.4.1.2.5	Client Authenticated State	45
6.4.1.2.6	Idle State	46
6.4.1.2.7	Authenticating Framework State	47
6.4.1.2.8	Framework Authenticated State	47
6.4.1.2.9	Authenticating Client State	47
6.4.1.2.10	Client Authenticated State	47
6.4.1.2.11	Idle State	48
6.4.1.2.12	Authenticating Framework State	48
6.4.1.2.13	Framework Authenticated State	49
6.4.1.2.14	Authenticating Client State	49
6.4.1.2.15	Client Authenticated State	49
6.4.1.2.16	Idle State	50
6.4.1.2.17	Authenticating Framework State	50
6.4.1.2.18	Framework Authenticated State	51
6.4.1.2.19	Authenticating Client State	51
6.4.1.2.20	Client Authenticated State	51
6.4.1.3	State Transition Diagrams for IpAccess	52
6.4.1.3.1	Active State	52
7	Framework-to-Application API	53
7.1	Sequence Diagrams	53
7.1.1	Event Notification Sequence Diagrams	53
7.1.1.1	Enable Event Notification	53
7.1.2	Integrity Management Sequence Diagrams	54
7.1.2.1	Load Management: Suspend/resume notification from application	54
7.1.2.2	Load Management: Framework queries load statistics	54
7.1.2.3	Load Management: Framework callback registration and Application load control	55
7.1.2.4	Load Management: Application reports current load condition	56
7.1.2.5	Load Management: Application queries load statistics	57
7.1.2.6	Load Management: Application callback registration and load control	58
7.1.2.7	Heartbeat Management: Start/perform/end heartbeat supervision of the application	59
7.1.2.8	Fault Management: Framework detects a Service failure	60
7.1.2.9	Fault Management: Application requests a Framework activity test	62
7.1.3	Service Agreement Management Sequence Diagrams	62
7.1.3.1	Service Selection	62
7.1.4	Service Discovery Sequence Diagrams	64
7.1.4.1	Service Discovery	64
7.2	Class Diagrams	67
7.3	Interface Classes	69
7.3.1	Service Discovery Interface Classes	69
7.3.1.1	Interface Class IpServiceDiscovery	69
7.3.1.1.1	Method listServiceTypes()	70
7.3.1.1.3	Method discoverService()	71
7.3.1.1.4	Method listSubscribedServices()	72
7.3.2	Service Agreement Management Interface Classes	72
7.3.2.1	Interface Class IpAppServiceAgreementManagement	72

7.3.2.1.1	Method signServiceAgreement().....	72
7.3.2.1.2	Method terminateServiceAgreement()	73
7.3.2.2	Interface Class IpServiceAgreementManagement	74
7.3.2.2.1	Method signServiceAgreement().....	74
7.3.2.2.2	Method terminateServiceAgreement()	75
7.3.2.2.3	Method selectService().....	76
7.3.2.2.4	Method initiateSignServiceAgreement()	76
7.3.3	Integrity Management Interface Classes	77
7.3.3.1	Interface Class IpAppFaultManager	77
7.3.3.1.1	Method activityTestRes()	78
7.3.3.1.2	Method appActivityTestReq()	78
7.3.3.1.3	Method <<deprecated>> fwFaultReportInd()	78
7.3.3.1.4	Method <<deprecated>> fwFaultRecoveryInd()	78
7.3.3.1.5	Method <<deprecated>> svcUnavailableInd()	78
7.3.3.1.6	Method <<deprecated>> genFaultStatsRecordRes().....	79
7.3.3.1.7	Method <<deprecated>> fwUnavailableInd()	79
7.3.3.1.8	Method activityTestErr()	79
7.3.3.1.9	Method <<deprecated>> genFaultStatsRecordErr().....	79
7.3.3.1.10	Method appUnavailableInd().....	80
7.3.3.1.11	Method <<deprecated>> genFaultStatsRecordReq()	80
7.3.3.1.12	Method svcAvailStatusInd()	80
7.3.3.1.13	Method <<new>> generateFaultStatisticsRecordRes()	80
7.3.3.1.14	Method <<new>> generateFaultStatisticsRecordErr().....	81
7.3.3.1.15	Method <<new>> generateFaultStatisticsRecordReq().....	81
7.3.3.1.16	Method <<new>> fwAvailStatusInd()	81
7.3.3.2	Interface Class IpFaultManager	82
7.3.3.2.1	Method activityTestReq().....	82
7.3.3.2.2	Method appActivityTestRes()	83
7.3.3.2.3	Method svcUnavailableInd().....	83
7.3.3.2.4	Method <<deprecated>> genFaultStatsRecordReq()	84
7.3.3.2.5	Method appActivityTestErr()	84
7.3.3.2.6	Method <<deprecated>> appUnavailableInd().....	84
7.3.3.2.7	Method <<deprecated>> genFaultStatsRecordRes().....	85
7.3.3.2.8	Method <<deprecated>> genFaultStatsRecordErr().....	85
7.3.3.2.9	Method appAvailStatusInd()	85
7.3.3.2.10	Method <<new>> generateFaultStatisticsRecordReq().....	86
7.3.3.2.11	Method <<new>> generateFaultStatisticsRecordRes()	86
7.3.3.2.12	Method <<new>> generateFaultStatisticsRecordErr().....	86
7.3.3.3	Interface Class IpAppHeartBeatMgmt.....	87
7.3.3.3.1	Method enableAppHeartBeat().....	87
7.3.3.3.2	Method disableAppHeartBeat().....	87
7.3.3.3.3	Method changeInterval()	87
7.3.3.4	Interface Class IpAppHeartBeat.....	88
7.3.3.4.1	Method pulse()	88
7.3.3.5	Interface Class IpHeartBeatMgmt.....	88
7.3.3.5.1	Method enableHeartBeat()	88
7.3.3.5.2	Method disableHeartBeat().....	89
7.3.3.5.3	Method changeInterval()	89
7.3.3.6	Interface Class IpHeartBeat	89
7.3.3.6.1	Method pulse()	90
7.3.3.7	Interface Class IpAppLoadManager	90
7.3.3.7.1	Method <<deprecated>> queryAppLoadReq()	90
7.3.3.7.2	Method <<deprecated>> queryLoadRes().....	91
7.3.3.7.3	Method <<deprecated>> queryLoadErr()	91
7.3.3.7.4	Method loadLevelNotification().....	91
7.3.3.7.5	Method resumeNotification()	91
7.3.3.7.6	Method suspendNotification()	92
7.3.3.7.7	Method createLoadLevelNotification()	92
7.3.3.7.8	Method destroyLoadLevelNotification()	92
7.3.3.7.9	Method <<new>> queryAppLoadStatsReq()	92
7.3.3.7.10	Method <<new>> queryLoadStatsRes()	92
7.3.3.7.11	Method <<new>> queryLoadStatsErr()	92

7.3.3.8	Interface Class IpLoadManager	93
7.3.3.8.1	Method reportLoad()	94
7.3.3.8.2	Method <<deprecated>> queryLoadReq()	94
7.3.3.8.3	Method <<deprecated>> queryAppLoadRes()	95
7.3.3.8.4	Method <<deprecated>> queryAppLoadErr()	95
7.3.3.8.5	Method createLoadLevelNotification()	96
7.3.3.8.6	Method destroyLoadLevelNotification()	96
7.3.3.8.7	Method resumeNotification()	96
7.3.3.8.8	Method suspendNotification()	97
7.3.3.8.9	Method <<new>> queryLoadStatsReq()	97
7.3.3.8.10	Method <<new>> queryAppLoadStatsRes()	97
7.3.3.8.11	Method <<new>> queryAppLoadStatsErr()	98
7.3.3.9	Interface Class IpOAM	98
7.3.3.9.1	Method systemDateTimeQuery()	98
7.3.3.10	Interface Class IpAppOAM	99
7.3.3.10.1	Method systemDateTimeQuery()	99
7.3.4	Event Notification Interface Classes	99
7.3.4.1	Interface Class IpAppEventNotification	99
7.3.4.1.1	Method reportNotification()	100
7.3.4.1.2	Method notificationTerminated()	100
7.3.4.2	Interface Class IpEventNotification	100
7.3.4.2.1	Method createNotification()	101
7.3.4.2.2	Method destroyNotification()	101
7.4	State Transition Diagrams	101
7.4.1	Service Discovery State Transition Diagrams	102
7.4.1.1	State Transition Diagrams for IpServiceDiscovery	102
7.4.1.1.1	Active State	102
7.4.2	Service Agreement Management State Transition Diagrams	102
7.4.3	Integrity Management State Transition Diagrams	103
7.4.3.1	State Transition Diagrams for IpLoadManager	103
7.4.3.1.1	Idle State	103
7.4.3.1.2	Notification Suspended State	103
7.4.3.1.3	Active State	104
7.4.3.2	State Transition Diagrams for LoadManagerInternal	104
7.4.3.2.1	Normal load State	104
7.4.3.2.2	Application Overload State	104
7.4.3.2.3	Internal overload State	105
7.4.3.2.4	Internal and Application Overload State	105
7.4.3.3	State Transition Diagrams for IpOAM	105
7.4.3.3.1	Active State	105
7.4.3.4	State Transition Diagrams for IpFaultManager	106
7.4.3.4.1	Framework Active State	106
7.4.3.4.2	Framework Faulty State	106
7.4.3.4.3	Framework Activity Test State	106
7.4.3.4.4	Service Activity Test State	106
7.4.4	Event Notification State Transition Diagrams	107
7.4.4.1	State Transition Diagrams for IpEventNotification	107
8	Framework-to-Enterprise Operator API	107
8.1	Sequence Diagrams	111
8.1.1	Event Notification Sequence Diagrams	111
8.1.2	Service Subscription Sequence Diagrams	111
8.1.2.1	Service Discovery and Subscription Scenario	111
8.1.2.2	Enterprise Operator and Client Application Subscription Management Sequence Diagram	113
8.2	Class Diagrams	115
8.3	Interface Classes	117
8.3.1	Event Notification Interface Classes	117
8.3.1.1	Interface Class IpClientEventNotification	117
8.3.1.1.1	Method reportNotification()	118
8.3.1.1.2	Method notificationTerminated()	118
8.3.1.2	Interface Class IpEventNotification	118
8.3.1.2.1	Method createNotification()	119

8.3.1.2.2	Method destroyNotification()	119
8.3.2	Service Subscription Interface Classes	119
8.3.2.1	Interface Class IpClientAppManagement	119
8.3.2.1.1	Method createClientApp()	120
8.3.2.1.2	Method modifyClientApp()	120
8.3.2.1.3	Method deleteClientApp()	121
8.3.2.1.4	Method createSAG()	121
8.3.2.1.5	Method modifySAG()	121
8.3.2.1.6	Method deleteSAG()	122
8.3.2.1.7	Method addSAGMembers()	122
8.3.2.1.8	Method removeSAGMembers()	122
8.3.2.1.9	Method requestConflictInfo()	123
8.3.2.2	Interface Class IpClientAppInfoQuery	123
8.3.2.2.1	Method describeClientApp()	124
8.3.2.2.2	Method listClientApps()	124
8.3.2.2.3	Method describeSAG()	125
8.3.2.2.4	Method listSAGs()	125
8.3.2.2.5	Method listSAGMembers()	125
8.3.2.2.6	Method listClientAppMembership()	126
8.3.2.3	Interface Class IpServiceProfileManagement	126
8.3.2.3.1	Method createServiceProfile()	126
8.3.2.3.2	Method modifyServiceProfile()	127
8.3.2.3.3	Method deleteServiceProfile()	127
8.3.2.3.4	Method assign()	127
8.3.2.3.5	Method deassign()	128
8.3.2.3.6	Method requestConflictInfo()	128
8.3.2.4	Interface Class IpServiceProfileInfoQuery	129
8.3.2.4.1	Method listServiceProfiles()	129
8.3.2.4.2	Method describeServiceProfile()	130
8.3.2.4.3	Method listAssignedMembers()	130
8.3.2.5	Interface Class IpServiceContractManagement	130
8.3.2.5.1	Method createServiceContract()	131
8.3.2.5.2	Method modifyServiceContract()	131
8.3.2.5.3	Method deleteServiceContract()	132
8.3.2.6	Interface Class IpServiceContractInfoQuery	132
8.3.2.6.1	Method describeServiceContract()	132
8.3.2.6.2	Method listServiceContracts()	133
8.3.2.6.3	Method listServiceProfiles()	133
8.3.2.7	Interface Class IpEntOpAccountManagement	134
8.3.2.7.1	Method modifyEntOpAccount()	134
8.3.2.7.2	Method deleteEntOpAccount()	134
8.3.2.8	Interface Class IpEntOpAccountInfoQuery	134
8.3.2.8.1	Method describeEntOpAccount()	135
8.4	State Transition Diagrams	135
8.4.1	Event Notification State Transition Diagrams	135
8.4.2	Service Subscription State Transition Diagrams	135
9	Framework-to-Service API	136
9.1	Sequence Diagrams	136
9.1.1	Service Discovery Sequence Diagrams	136
9.1.2	Service Registration Sequence Diagrams	136
9.1.2.1	New SCF Sub Type Registration	136
9.1.2.2	New SCF Registration	136
9.1.3	Service Instance Lifecycle Manager Sequence Diagrams	138
9.1.3.1	Sign Service Agreement	138
9.1.4	Integrity Management Sequence Diagrams	139
9.1.4.1	Load Management: Service callback registration and load control	139
9.1.4.2	Load Management: Framework callback registration and service load control	140
9.1.4.3	Load Management: Client and Service Load Balancing	141
9.1.4.4	Heartbeat Management: Start/perform/end heartbeat supervision of the service	142
9.1.4.5	Fault Management: Service requests Framework activity test	143
9.1.4.6	Fault Management: Service requests Application activity test	144

9.1.4.7	Fault Management: Application requests Service activity test	145
9.1.4.8	Fault Management: Application detects service is unavailable.....	146
9.1.5	Event Notification Sequence Diagrams	146
9.2	Class Diagrams.....	147
9.3	Interface Classes.....	149
9.3.1	Service Registration Interface Classes.....	149
9.3.1.1	Interface Class IpFwServiceRegistration	149
9.3.1.1.1	Method registerService()	150
9.3.1.1.2	Method announceServiceAvailability().....	151
9.3.1.1.3	Method unregisterService()	151
9.3.1.1.4	Method describeService().....	152
9.3.1.1.5	Method unannounceService().....	152
9.3.1.1.6	Method <<new>> registerServiceSubType()	153
9.3.2	Service Instance Lifecycle Manager Interface Classes	154
9.3.2.1	Interface Class IpServiceInstanceLifecycleManager	154
9.3.2.1.1	Method createServiceManager()	154
9.3.2.1.2	Method destroyServiceManager()	155
9.3.3	Service Discovery Interface Classes	155
9.3.3.1	Interface Class IpFwServiceDiscovery	155
9.3.3.1.1	Method listServiceTypes()	156
9.3.3.1.2	Method describeServiceType().....	156
9.3.3.1.3	Method discoverService().....	157
9.3.3.1.4	Method listRegisteredServices().....	157
9.3.4	Integrity Management Interface Classes	158
9.3.4.1	Interface Class IpFwFaultManager	158
9.3.4.1.1	Method activityTestReq()	159
9.3.4.1.2	Method svcActivityTestRes().....	160
9.3.4.1.3	Method appUnavailableInd().....	160
9.3.4.1.4	Method <<deprecated>> genFaultStatsRecordReq()	160
9.3.4.1.5	Method <<deprecated>> svcUnavailableInd().....	161
9.3.4.1.6	Method svcActivityTestErr().....	161
9.3.4.1.7	Method <<deprecated>> genFaultStatsRecordRes().....	161
9.3.4.1.8	Method <<deprecated>> genFaultStatsRecordErr().....	162
9.3.4.1.9	Method <<deprecated>> generateFaultStatsRecordRes()	162
9.3.4.1.10	Method <<deprecated>> generateFaultStatsRecordErr()	162
9.3.4.1.11	Method svcAvailStatusInd().....	163
9.3.4.1.12	Method <<new>> generateFaultStatisticsRecordReq().....	163
9.3.4.1.13	Method <<new>> generateFaultStatisticsRecordRes()	163
9.3.4.1.14	Method <<new>> generateFaultStatisticsRecordErr()	164
9.3.4.2	Interface Class IpSvcFaultManager	164
9.3.4.2.1	Method activityTestRes()	165
9.3.4.2.2	Method svcActivityTestReq()	166
9.3.4.2.3	Method <<deprecated>> fwFaultReportInd()	166
9.3.4.2.4	Method <<deprecated>> fwFaultRecoveryInd()	166
9.3.4.2.5	Method <<deprecated>> fwUnavailableInd()	166
9.3.4.2.6	Method svcUnavailableInd()	167
9.3.4.2.7	Method <<deprecated>> appUnavailableInd().....	167
9.3.4.2.8	Method <<deprecated>> genFaultStatsRecordRes().....	167
9.3.4.2.9	Method activityTestErr()	168
9.3.4.2.10	Method <<deprecated>> genFaultStatsRecordErr().....	168
9.3.4.2.11	Method <<deprecated>> genFaultStatsRecordReq()	168
9.3.4.2.12	Method <<deprecated>> generateFaultStatsRecordReq().....	169
9.3.4.2.13	Method appAvailStatusInd()	169
9.3.4.2.14	Method <<new>> generateFaultStatisticsRecordRes()	169
9.3.4.2.15	Method <<new>> generateFaultStatisticsRecordErr()	170
9.3.4.2.16	Method <<new>> generateFaultStatisticsRecordReq().....	170
9.3.4.2.17	Method <<new>> fwAvailStatusInd()	171
9.3.4.3	Interface Class IpFwHeartBeatMgmt.....	171
9.3.4.3.1	Method enableHeartBeat()	171
9.3.4.3.2	Method disableHeartBeat().....	172
9.3.4.3.3	Method changeInterval()	172
9.3.4.4	Interface Class IpFwHeartBeat	172

9.3.4.4.1	Method pulse()	172
9.3.4.5	Interface Class IpSvcHeartBeatMgmt	173
9.3.4.5.1	Method enableSvcHeartBeat()	173
9.3.4.5.2	Method disableSvcHeartBeat()	173
9.3.4.5.3	Method changeInterval()	174
9.3.4.6	Interface Class IpSvcHeartBeat	174
9.3.4.6.1	Method pulse()	174
9.3.4.7	Interface Class IpFwLoadManager	175
9.3.4.7.1	Method reportLoad()	175
9.3.4.7.2	Method <<deprecated>> queryLoadReq()	176
9.3.4.7.3	Method <<deprecated>> querySvcLoadRes()	176
9.3.4.7.4	Method <<deprecated>> querySvcLoadErr()	176
9.3.4.7.5	Method createLoadLevelNotification()	177
9.3.4.7.6	Method destroyLoadLevelNotification()	177
9.3.4.7.7	Method suspendNotification()	177
9.3.4.7.8	Method resumeNotification()	178
9.3.4.7.9	Method <<new>> queryLoadStatsReq()	178
9.3.4.7.10	Method <<new>> querySvcLoadStatsRes()	178
9.3.4.7.11	Method <<new>> querySvcLoadStatsErr()	179
9.3.4.8	Interface Class IpSvcLoadManager	179
9.3.4.8.1	Method <<deprecated>> querySvcLoadReq()	180
9.3.4.8.2	Method <<deprecated>> queryLoadRes()	180
9.3.4.8.3	Method <<deprecated>> queryLoadErr()	181
9.3.4.8.4	Method loadLevelNotification()	181
9.3.4.8.5	Method suspendNotification()	181
9.3.4.8.6	Method resumeNotification()	181
9.3.4.8.7	Method createLoadLevelNotification()	182
9.3.4.8.8	Method destroyLoadLevelNotification()	182
9.3.4.8.9	Method <<new>> querySvcLoadStatsReq()	182
9.3.4.8.10	Method <<new>> queryLoadStatsRes()	183
9.3.4.8.11	Method <<new>> queryLoadStatsErr()	183
9.3.4.9	Interface Class IpFwOAM	183
9.3.4.9.1	Method systemDateTimeQuery()	184
9.3.4.10	Interface Class IpSvcOAM	184
9.3.4.10.1	Method systemDateTimeQuery()	184
9.3.5	Event Notification Interface Classes	185
9.3.5.1	Interface Class IpFwEventNotification	185
9.3.5.1.1	Method createNotification()	185
9.3.5.1.2	Method destroyNotification()	185
9.3.5.2	Interface Class IpSvcEventNotification	186
9.3.5.2.1	Method reportNotification()	186
9.3.5.2.2	Method notificationTerminated()	186
9.4	State Transition Diagrams	187
9.4.1	Service Registration State Transition Diagrams	188
9.4.1.1	State Transition Diagrams for IpFwServiceRegistration	188
9.4.1.1.1	SCF Registered State	188
9.4.1.1.2	SCF Announced State	188
9.4.2	Service Instance Lifecycle Manager State Transition Diagrams	189
9.4.3	Service Discovery State Transition Diagrams	189
9.4.4	Integrity Management State Transition Diagrams	189
9.4.4.1	State Transition Diagrams for IpFwLoadManager	189
9.4.4.1.1	Idle State	189
9.4.4.1.2	Notification Suspended State	189
9.4.4.1.3	Active State	190
9.4.4.2	State Transition Diagrams for IpFwFaultManager	190
9.4.4.2.1	Framework Active State	190
9.4.4.2.2	Framework Activity Test State	190
9.4.4.2.3	Application Activity Test State	190
9.4.4.2.4	Framework Faulty State	190
9.4.5	Event Notification State Transition Diagrams	191
10	Service Properties	191

10.1	Service Super and Sub Types	191
10.2	Service Property Types	191
10.3	General Service Properties	193
10.3.1	Service Name	194
10.3.2	Service Version	194
10.3.3	Service ID	194
10.3.4	Service Description	194
10.3.5	Product Name	194
10.3.6	Product Version	194
10.3.7	<<deprecated>> Supported Interfaces	195
10.3.8	Operation Set	195
10.3.9	Compatible Service	195
10.3.10	Backward Compatibility Level	196
10.3.11	Migration Required	196
10.3.12	Data Migrated	197
10.3.13	Migration Date And Time	198
10.3.14	Support for Regular Expressions in Address Range	198
11	Data Definitions	198
11.1	Common Framework Data Definitions	199
11.1.1	TpClientAppID	199
11.1.2	TpClientAppIDList	199
11.1.3	TpDomainID	199
11.1.4	TpDomainIDType	199
11.1.5	TpEntOpID	199
11.1.6	TpPropertyName	199
11.1.7	TpPropertyValue	200
11.1.8	TpProperty	200
11.1.9	TpPropertyList	200
11.1.10	TpEntOpIDList	200
11.1.11	TpFwID	200
11.1.12	TpService	200
11.1.13	TpServiceList	200
11.1.14	TpServiceDescription	200
11.1.15	TpServiceID	200
11.1.16	TpServiceIDList	201
11.1.17	TpServiceInstanceID	201
11.1.18	TpServiceTypeProperty	201
11.1.19	TpServiceTypePropertyList	201
11.1.20	TpServiceTypePropertyMode	201
11.1.21	TpServicePropertyTypeName	201
11.1.22	TpServicePropertyName	201
11.1.23	TpServicePropertyNameList	201
11.1.24	TpServicePropertyValue	201
11.1.25	TpServicePropertyValueList	202
11.1.26	TpServiceProperty	202
11.1.27	TpServicePropertyList	202
11.1.28	TpServiceSupplierID	202
11.1.29	TpServiceTypeDescription	202
11.1.30	TpServiceTypeName	203
11.1.31	TpServiceTypeNameList	203
11.1.32	TpSubjectType	203
11.1.33	TpServiceTypePropertyValue	204
11.1.34	TpServiceTypePropertyValueList	204
11.2	Event Notification Data Definitions	204
11.2.1	TpFwEventName	204
11.2.2	TpFwEventCriteria	205
11.2.3	TpFwEventInfo	205
11.2.4	TpFwMigrationServiceAvailableInfo	205
11.2.5	TpMigrationAdditionalInfo	206
11.2.6	TpMigrationAdditionalInfoType	207
11.2.7	TpMigrationAdditionalInfoSet	207

11.2.8	TpFwAgreementInfo	207
11.3	Trust and Security Management Data Definitions	207
11.3.1	TpAccessType	207
11.3.2	TpAuthType	207
11.3.3	TpEncryptionCapability	208
11.3.4	TpEncryptionCapabilityList	208
11.3.5	TpEndAccessProperties	208
11.3.6	TpAuthDomain	208
11.3.7	TpInterfaceName	209
11.3.8	TpInterfaceNameList	209
11.3.9	TpServiceToken	209
11.3.10	TpSignatureAndServiceMgr	209
11.3.11	TpSigningAlgorithm	210
11.3.12	TpSigningAlgorithmCapabilityList	210
11.3.13	TpAuthMechanism	210
11.3.14	TpAuthMechanismList	210
11.4	Integrity Management Data Definitions	211
11.4.1	TpActivityTestRes	211
11.4.2	TpFaultStatsRecord	211
11.4.3	TpFaultStats	211
11.4.4	TpFaultStatisticsError	211
11.4.5	TpFaultStatsSet	211
11.4.6	TpActivityTestID	211
11.4.7	TpInterfaceFault	212
11.4.8	TpSvcUnavailReason	212
11.4.9	TpFwUnavailReason	212
11.4.10	TpLoadLevel	212
11.4.11	TpLoadThreshold	212
11.4.12	TpLoadInitVal	213
11.4.13	TpLoadPolicy	213
11.4.14	TpLoadStatistic	213
11.4.15	TpLoadStatisticList	213
11.4.16	TpLoadStatisticData	213
11.4.17	TpLoadStatisticEntityID	213
11.4.18	TpLoadStatisticEntityType	214
11.4.19	TpLoadStatisticInfo	214
11.4.20	TpLoadStatisticInfoType	214
11.4.21	TpLoadStatisticError	214
11.4.22	TpSvcAvailStatusReason	214
11.4.23	TpAppAvailStatusReason	215
11.4.24	TpLoadTestID	215
11.4.25	TpFaultStatsErrorList	216
11.4.26	TpFaultReqID	216
11.4.27	TpFwAvailStatusReason	216
11.5	Service Subscription Data Definitions	216
11.5.1	TpPropertyName	216
11.5.2	TpPropertyValue	216
11.5.3	TpProperty	216
11.5.4	TpPropertyList	216
11.5.5	TpEntOpProperties	217
11.5.6	TpEntOp	217
11.5.7	TpServiceContractID	217
11.5.8	TpServiceContractIDList	217
11.5.9	TpPersonName	217
11.5.10	TpPostalAddress	217
11.5.11	TpTelephoneNumber	217
11.5.12	TpEmail	217
11.5.13	TpHomePage	217
11.5.14	TpPersonProperties	217
11.5.15	TpPerson	218
11.5.16	TpServiceStartDate	218
11.5.17	TpServiceEndDate	218

11.5.18	TpServiceRequestor.....	218
11.5.19	TpBillingContact	218
11.5.20	TpServiceSubscriptionProperties.....	218
11.5.21	TpServiceContract	218
11.5.22	TpServiceContractDescription.....	219
11.5.23	TpClientAppProperties	219
11.5.24	TpClientAppDescription.....	219
11.5.25	TpSagID.....	219
11.5.26	TpSagIDList	219
11.5.27	TpSagDescription	219
11.5.28	TpSag.....	220
11.5.29	TpServiceProfileID	220
11.5.30	TpServiceProfileIDList.....	220
11.5.31	TpServiceProfile	220
11.5.32	TpServiceProfileDescription.....	220
11.5.33	TpSagProfilePair.....	221
11.5.34	TpAddSagMembersConflict	221
11.5.35	TpAddSagMembersConflictList.....	221
11.5.36	TpAssignSagToServiceProfileConflict.....	222
11.5.37	TpAssignSagToServiceProfileConflictList	222
12	Exception Classes.....	222
Annex A (normative):	OMG IDL Description of Framework	224
Annex B (informative):	W3C WSDL Description of Framework.....	225
Annex C (informative):	Java™ API Description of the Framework.....	226
Annex D (informative):	Contents of 3GPP OSA R6 Framework.....	227
Annex E (informative):	Description of the Framework for 3GPP2 cdma2000 networks.....	228
E.1	General Exceptions.....	228
E.2	Specific Exceptions	228
E.2.1	Clause 1: Scope	228
E.2.2	Clause 2: References	228
E.2.3	Clause 3: Definitions and abbreviations.....	228
E.2.4	Clause 4: Overview of the Framework.....	228
E.2.5	Clause 5: The Base Interface Specification.....	228
E.2.6	Clause 6: Framework Access Session API.....	228
E.2.7	Clause 7 Framework-to-Application Sequence Diagrams.....	228
E.2.8	Clause 9: Framework-to-Service API.....	229
E.2.9	Clause 10: Service Properties.....	229
E.2.10	Clause 11: Data Definitions.....	229
E.2.11	Clause 12: Exception Classes.....	229
E.2.12	Annex A (normative): OMG IDL Description of the Framework.....	229
E.2.13	Annex B (informative): W3C WSDL Description of the Framework.....	229
E.2.14	Annex C (informative): Java™ API Description of the Framework.....	229
Annex F (informative):	Record of changes	230
F.1	Interfaces	230
F.1.1	New	230
F.1.2	Deprecated.....	230
F.1.3	Removed.....	230
F.2	Methods	231
F.2.1	New	231
F.2.2	Deprecated.....	231
F.2.3	Modified	231
F.2.4	Removed.....	231
F.3	Data Definitions	232

F.3.1	New	232
F.3.2	Modified	233
F.3.3	Removed.....	233
F.4	Service Properties.....	234
F.4.1	New	234
F.4.2	Deprecated.....	234
F.4.3	Modified	234
F.4.4	Removed.....	234
F.5	Exceptions	235
F.5.1	New	235
F.5.2	Modified	235
F.5.3	Removed.....	235
F.6	Others	235
History	236

iTeh STANDARD PREVIEW
 (standards.iteh.ai)

Full standard:
<https://standards.iteh.ai/catalog/standards/sist/229e1f97-1e83-44c5-b1a2-98a2b70fc86a/etsi-es-203-915-3-v1.3.1-2008-04>