



Publicly Available Specification(PAS); O-RAN R1 interface: Test Specification (O-RAN.WG2.TS.R1TS-R004-v03.00)

CAUTION

The present document has been submitted to ETSI as a PAS produced by O-RAN Alliance and approved by the ETSI Technical Committee Mobile Standards Group (MSG).

ETSI had been assigned all the relevant copyrights related to the document O-RAN.WG2.TS.R1TS-R004-v03.00 on an "as is basis". Consequently, to the fullest extent permitted by law, ETSI disclaims all warranties whether express, implied, statutory or otherwise including but not limited to merchantability, non-infringement of any intellectual property rights of third parties. No warranty is given about the accuracy and the completeness of the content of the present document.

Reference

DTS/MSG-001173

Keywords

interface, PAS

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 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Important notice

The present document can be downloaded from the
[ETSI Search & Browse Standards](#) application.

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 on [ETSI deliver](#) repository.

Users should be aware that the present document may be revised or have its status changed,
this information is available in the [Milestones listing](#).

If you find errors in the present document, please send your comments to
the relevant service listed under [Committee Support Staff](#).

If you find a security vulnerability in the present document, please report it through our
[Coordinated Vulnerability Disclosure \(CVD\)](#) program.

Notice of disclaimer & limitation of liability

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.

No recommendation as to products and services or vendors is made or should be implied.

No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.

In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and ETSI shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

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 2026.
All rights reserved.

Contents

Intellectual Property Rights	11
Foreword.....	11
Modal verbs terminology.....	11
Introduction	11
1 Scope	12
2 References	12
2.1 Normative references	12
2.2 Informative references.....	12
3 Definition of terms, symbols and abbreviations.....	12
3.1 Terms.....	12
3.2 Symbols.....	12
3.3 Abbreviations	13
4 Test methodology	13
4.1 General	13
4.2 Conformance testing of rApps.....	13
4.2.1 General.....	13
4.2.2 Test configuration.....	13
4.2.2.1 Overview	13
4.2.2.2 Device under test rApps	14
4.2.2.3 Test simulator.....	14
4.3 Conformance testing of SMO/Non-RT RIC framework	14
4.3.1 General.....	14
4.3.2 Test configuration.....	14
4.3.2.1 Overview	14
4.3.2.2 Device under test (SMO/Non-RT RIC Framework)	14
4.3.2.3 Test simulator.....	15
5 SME Service Registration service test cases	15
5.1 Conformance test cases for rApp	15
5.1.1 General.....	15
5.1.1.1 Device under test requirements.....	15
5.1.1.2 Test simulator capabilities.....	15
5.1.2 Register service API as API Consumer test scenario.....	15
5.1.2.1 Register service (positive case).....	15
5.1.2.1.1 Test description and applicability	15
5.1.2.1.2 Test entrance criteria	15
5.1.2.1.3 Test methodology	15
5.1.2.1.4 Expected result	16
5.1.2.2 Update registered service (positive case)	16
5.1.2.2.1 Test description and applicability	16
5.1.2.2.2 Test entrance criteria	16
5.1.2.2.3 Test methodology	16
5.1.2.2.4 Expected result	17
5.1.2.3 Deregister service (positive case).....	17
5.1.2.3.1 Test description and applicability	17
5.1.2.3.2 Test entrance criteria	17
5.1.2.3.3 Test methodology	17
5.1.2.3.4 Expected result	17
5.1.2.4 Query registered service (positive case).....	17
5.1.2.4.1 Test description and applicability	17
5.1.2.4.2 Test entrance criteria	18
5.1.2.4.3 Test methodology	18
5.1.2.4.4 Expected result	18
5.1.5.5 Partially updated registered service (positive case).....	18

5.1.5.5.1	Test description and applicability	18
5.1.5.5.2	Test entrance criteria	18
5.1.5.5.3	Test methodology	18
5.1.5.5.4	Expected result	19
5.2	Conformance test cases for SMO/Non-RT RIC framework.....	19
5.2.1	General.....	19
5.2.1.1	Device under test requirements.....	19
5.2.1.2	Test simulator capabilities.....	19
5.2.2	Register service API as API Producer test scenario.....	19
5.2.2.1	Register service (positive case).....	19
5.2.2.1.1	Test description and applicability	19
5.2.2.1.2	Test entrance criteria	20
5.2.2.1.3	Test methodology	20
5.2.2.1.4	Expected result	20
5.2.2.2	Update registered service (positive case)	20
5.2.2.2.1	Test description and applicability.....	20
5.2.2.2.2	Test entrance criteria	20
5.2.2.2.3	Test methodology	21
5.2.2.2.4	Expected result	21
5.2.2.3	Deregister service (positive case).....	21
5.2.2.3.1	Test description and applicability	21
5.2.2.3.2	Test entrance criteria	21
5.2.2.3.3	Test methodology	21
5.2.2.3.4	Expected result	22
5.2.2.4	Query registered service (positive case).....	22
5.2.2.4.1	Test description and applicability.....	22
5.2.2.4.2	Test entrance criteria	22
5.2.2.4.3	Test methodology	22
5.2.2.4.4	Expected result	23
5.2.5.5	Partially updated registered service (positive case).....	23
5.2.5.5.1	Test description and applicability.....	23
5.2.5.5.2	Test entrance criteria	23
5.2.5.5.3	Test methodology	23
5.2.5.5.4	Expected result	23
6	SME Service Discovery service test cases	24
6.1	Conformance test cases for rApp	24
6.1.1	General.....	24
6.1.1.1	Device under test requirements.....	24
6.1.1.2	Test simulator capabilities.....	24
6.1.2	Discover service API as API Consumer test scenario	24
6.1.2.1	Query service (positive case)	24
6.1.2.1.1	Test description and applicability	24
6.1.2.1.2	Test entrance criteria	24
6.1.2.1.3	Test methodology	24
6.1.2.1.4	Expected result	25
6.2	Conformance test cases for SMO/Non-RT RIC framework.....	25
6.2.1	General.....	25
6.2.1.1	Device under test requirements.....	25
6.2.1.2	Test simulator capabilities.....	25
6.2.2	Service discovery API as API Producer test scenario.....	25
6.2.2.1	Query service (positive case)	25
6.2.2.1.1	Test description and applicability.....	25
6.2.2.1.2	Test entrance criteria	25
6.2.2.1.3	Test methodology	25
6.2.2.1.4	Expected result	26
7	SME Service Subscription service test cases	26
7.1	Conformance test cases for rApp	26
7.1.1	General.....	26
7.1.1.1	Device under test requirements.....	26
7.1.1.2	Test simulator capabilities.....	26

7.1.2	Service events subscription API as API Consumer test scenario.....	27
7.1.2.1	Subscribe service events (positive case)	27
7.1.2.1.1	Test description and applicability	27
7.1.2.1.2	Test entrance criteria	27
7.1.2.1.3	Test methodology	27
7.1.2.1.4	Expected result	27
7.1.2.2	Unsubscribe service events (positive case)	27
7.1.2.2.1	Test description and applicability	27
7.1.2.2.2	Test entrance criteria	27
7.1.2.2.3	Test methodology	28
7.1.2.2.4	Expected result	28
7.2	Conformance test cases for SMO/Non-RT RIC framework.....	28
7.2.1	General.....	28
7.2.1.1	Device under test requirements.....	28
7.2.1.2	Test simulator capabilities.....	28
7.2.2	Service events subscription API as API Producer test scenario.....	29
7.2.2.1	Service events subscription (positive case)	29
7.2.2.1.1	Test description and applicability	29
7.2.2.1.2	Test entrance criteria	29
7.2.2.1.3	Test methodology	29
7.2.2.1.4	Expected result	29
7.2.2.2	Unsubscribe service events (positive case)	29
7.2.2.2.1	Test description and applicability	29
7.2.2.2.2	Test entrance criteria	30
7.2.2.2.3	Test methodology	30
7.2.2.2.4	Expected result	30
7.2.2.3	Notification service (positive case)	30
7.2.2.3.1	Test description and applicability	30
7.2.2.3.2	Test entrance criteria	30
7.2.2.3.3	Test methodology	31
7.2.2.3.4	Expected result	31
8	DME Data registration service test cases	31
8.1	Conformance test cases for rApp	31
8.1.1	General.....	31
8.1.1.1	Device under test requirements.....	31
8.1.1.2	Test simulator capabilities.....	31
8.1.2	Data registration API as API consumer test scenario	31
8.1.2.1	Register DME type (positive case).....	31
8.1.2.1.1	Test description and applicability	31
8.1.2.1.2	Test entrance criteria	32
8.1.2.1.3	Test methodology	32
8.1.2.1.4	Expected result	32
8.1.2.2	Deregister DME type (positive case)	32
8.1.2.2.1	Test description and applicability	32
8.1.2.2.2	Test entrance criteria	32
8.1.2.2.3	Test methodology	32
8.1.2.2.4	Expected result	33
8.1.2.3	Query DME type (positive case).....	33
8.1.2.3.1	Test description and applicability	33
8.1.2.3.2	Test entrance criteria	33
8.1.2.3.3	Test methodology	33
8.1.2.3.4	Expected result	33
8.1.2.4	Update DME type (positive case)	34
8.1.2.4.1	Test description and applicability	34
8.1.2.4.2	Test entrance criteria	34
8.1.2.4.3	Test methodology	34
8.1.2.4.4	Expected result	34
8.2	Conformance test cases for SMO/Non-RT RIC framework.....	34
8.2.1	General.....	34
8.2.1.1	Device under test requirements.....	34
8.2.1.2	Test simulator capabilities.....	35

8.2.2	Data registration API as API Producer test scenario	35
8.2.2.1	Register DME type (positive case).....	35
8.2.2.1.1	Test description and applicability	35
8.2.2.1.2	Test entrance criteria	35
8.2.2.1.3	Test methodology	35
8.2.2.1.4	Expected result	36
8.2.2.2	Deregister DME type (positive case)	36
8.2.2.2.1	Test description and applicability	36
8.2.2.2.2	Test entrance criteria	36
8.2.2.2.3	Test methodology	36
8.2.2.2.4	Expected result	36
8.2.2.3	Query DME type (positive case).....	36
8.2.2.3.1	Test description and applicability	36
8.2.2.3.2	Test entrance criteria	37
8.2.2.3.3	Test methodology	37
8.2.2.3.4	Expected result	37
8.2.2.4	Update DME type (positive case)	37
8.2.2.4.1	Test description and applicability	37
8.2.2.4.2	Test entrance criteria	37
8.2.2.4.3	Test methodology	38
8.2.2.4.4	Expected result	38
9	DME Data request service test cases.....	38
9.1	Conformance test cases for rApp	38
9.1.1	General.....	38
9.1.1.1	Device under test requirements.....	38
9.1.1.2	Test simulator capabilities.....	38
9.1.2	Data access API as API Consumer test scenario	39
9.1.2.1	Create data job (positive case)	39
9.1.2.1.1	Test description and applicability	39
9.1.2.1.2	Test entrance criteria	39
9.1.2.1.3	Test methodology	39
9.1.2.1.4	Expected result	39
9.1.2.2	Cancel data job (positive case).....	39
9.1.2.2.1	Test description and applicability	39
9.1.2.2.2	Test entrance criteria	39
9.1.2.2.3	Test methodology	40
9.1.2.2.4	Expected result	40
9.1.2.3	Query data job status (positive case).....	40
9.1.2.3.1	Test description and applicability	40
9.1.2.3.2	Test entrance criteria	40
9.1.2.3.3	Test methodology	40
9.1.2.3.4	Expected result	41
9.2	Conformance test cases for SMO/Non-RT RIC framework.....	41
9.2.1	General.....	41
9.2.1.1	Device under test requirements.....	41
9.2.1.2	Test simulator capabilities.....	41
9.2.2	Data access API as API Producer test scenario	41
9.2.2.1	Create data job (positive case)	41
9.2.2.1.1	Test description and applicability	41
9.2.2.1.2	Test entrance criteria	41
9.2.2.1.3	Test methodology	41
9.2.2.1.4	Expected result	42
9.2.2.2	Cancel data job (positive case).....	42
9.2.2.2.1	Test description and applicability	42
9.2.2.2.2	Test entrance criteria	42
9.2.2.2.3	Test methodology	42
9.2.2.2.4	Expected result	43
9.2.2.3	Query data job status (positive case).....	43
9.2.2.3.1	Test description and applicability	43
9.2.2.3.2	Test entrance criteria	43
9.2.2.3.3	Test methodology	43

9.2.2.3.4	Expected result	44
10	DME Data subscription service test cases	44
10.1	Conformance test cases for rApp	44
10.1.1	General.....	44
10.1.1.1	Device under test requirements	44
10.1.1.2	Test simulator capabilities.....	44
10.1.2	Data access API as API Consumer test scenario	44
10.1.2.1	Create data job (positive case)	44
10.1.2.1.1	Test description and applicability	44
10.1.2.1.2	Test entrance criteria	44
10.1.2.1.3	Test methodology	44
10.1.2.1.4	Expected result	45
10.1.2.2	Cancel data job (positive case).....	45
10.1.2.2.1	Test description and applicability	45
10.1.2.2.2	Test entrance criteria	45
10.1.2.2.3	Test methodology	45
10.1.2.2.4	Expected result	45
10.1.2.3	Query data job (positive case).....	46
10.1.2.3.1	Test description and applicability	46
10.1.2.3.2	Test entrance criteria	46
10.1.2.3.3	Test methodology	46
10.1.2.3.4	Expected result	46
10.1.2.4	Query data job status (positive case).....	46
10.1.2.4.1	Test description and applicability	46
10.1.2.4.2	Test entrance criteria	46
10.1.2.4.3	Test methodology	47
10.1.2.4.4	Expected result	47
10.1.2.5	Query data job identifiers (positive case).....	47
10.1.2.5.1	Test description and applicability	47
10.1.2.5.2	Test entrance criteria	47
10.1.2.5.3	Test methodology	47
10.1.2.5.4	Expected result	48
10.1.2.6	Update data job (positive case)	48
10.1.2.6.1	Test description and applicability	48
10.1.2.6.2	Test entrance criteria	48
10.1.2.6.3	Test methodology	48
10.1.2.6.4	Expected result	48
10.2	Conformance test cases for SMO/Non-RT RIC framework.....	49
10.2.1	General.....	49
10.2.1.1	Device under test requirements	49
10.2.1.2	Test simulator capabilities.....	49
10.2.2	Data access API as API Producer test scenario	49
10.2.2.1	Create data job (positive case)	49
10.2.2.1.1	Test description and applicability	49
10.2.2.1.2	Test entrance criteria	49
10.2.2.1.3	Test methodology	49
10.2.2.1.4	Expected result	50
10.2.2.2	Cancel data job (positive case).....	50
10.2.2.2.1	Test description and applicability	50
10.2.2.2.2	Test entrance criteria	50
10.2.2.2.3	Test methodology	50
10.2.2.2.4	Expected result	51
10.2.2.3	Query data job (positive case).....	51
10.2.2.3.1	Test description and applicability	51
10.2.2.3.2	Test entrance criteria	51
10.2.2.3.3	Test methodology	51
10.2.2.3.4	Expected result	51
10.2.2.4	Query data job status (positive case).....	52
10.2.2.4.1	Test description and applicability	52
10.2.2.4.2	Test entrance criteria	52
10.2.2.4.3	Test methodology	52

10.2.2.4.4	Expected result	52
10.2.2.5	Query data job identifiers (positive case)	52
10.2.2.5.1	Test description and applicability	52
10.2.2.5.2	Test entrance criteria	53
10.2.2.5.3	Test methodology	53
10.2.2.5.4	Expected result	53
10.2.2.6	Update data job (positive case)	53
10.2.2.6.1	Test description and applicability	53
10.2.2.6.2	Test entrance criteria	53
10.2.2.6.3	Test methodology	54
10.2.2.6.4	Expected result	54
10.2.2.7	Notify data availability (positive case)	54
10.2.2.7.1	Test description and applicability	54
10.2.2.7.2	Test entrance criteria	54
10.2.2.7.3	Test methodology	54
10.2.2.7.4	Expected result	55
11	A1 related services test cases	55
11.1	Conformance test cases for rApp	55
11.1.1	General	55
11.1.1.1	Device under test requirements	55
11.1.1.2	Test simulator capabilities	55
11.1.2	A1 policy management API as API Consumer test scenario	55
11.1.2.1	Query A1 policy type identifiers (positive case)	55
11.1.2.1.1	Test description and applicability	55
11.1.2.1.2	Test entrance criteria	55
11.1.2.1.3	Test methodology	56
11.1.2.1.4	Expected result	56
11.1.2.2	Query A1 policy type (positive case)	56
11.1.2.2.1	Test description and applicability	56
11.1.2.2.2	Test entrance criteria	56
11.1.2.2.3	Test methodology	56
11.1.2.2.4	Expected result	57
11.1.2.3	Query A1 policy identifiers (positive case)	57
11.1.2.3.1	Test description and applicability	57
11.1.2.3.2	Test entrance criteria	57
11.1.2.3.3	Test methodology	57
11.1.2.3.4	Expected result	57
11.1.2.4	Create A1 policy (positive case)	57
11.1.2.4.1	Test description and applicability	57
11.1.2.4.2	Test entrance criteria	57
11.1.2.4.3	Test methodology	58
11.1.2.4.4	Expected result	58
11.1.2.5	Query A1 policy (positive case)	58
11.1.2.5.1	Test description and applicability	58
11.1.2.5.2	Test entrance criteria	58
11.1.2.5.3	Test methodology	58
11.1.2.5.4	Expected result	59
11.1.2.6	Update A1 policy (positive case)	59
11.1.2.6.1	Test description and applicability	59
11.1.2.6.2	Test entrance criteria	59
11.1.2.6.3	Test methodology	59
11.1.2.6.4	Expected result	59
11.1.2.7	Delete A1 policy (positive case)	59
11.1.2.7.1	Test description and applicability	59
11.1.2.7.2	Test entrance criteria	60
11.1.2.7.3	Test methodology	60
11.1.2.7.4	Expected result	60
11.1.2.8	Query A1 policy status (positive case)	60
11.1.2.8.1	Test description and applicability	60
11.1.2.8.2	Test entrance criteria	60
11.1.2.8.3	Test methodology	60

11.1.2.8.4	Expected result	61
11.1.2.9	Subscribe A1 policy status (positive case)	61
11.1.2.9.1	Test description and applicability	61
11.1.2.9.2	Test entrance criteria	61
11.1.2.9.3	Test methodology	61
11.1.2.9.4	Expected result	61
11.1.2.10	Update A1 policy status subscription (positive case)	62
11.1.2.10.1	Test description and applicability	62
11.1.2.10.2	Test entrance criteria	62
11.1.2.10.3	Test methodology	62
11.1.2.10.4	Expected result	62
11.1.2.11	Query A1 policy status subscription (positive case)	62
11.1.2.11.1	Test description and applicability	62
11.1.2.11.2	Test entrance criteria	62
11.1.2.11.3	Test methodology	63
11.1.2.11.4	Expected result	63
11.1.2.12	Unsubscribe A1 policy status (positive case)	63
11.1.2.12.1	Test description and applicability	63
11.1.2.12.2	Test entrance criteria	63
11.1.2.12.3	Test methodology	63
11.1.2.12.4	Expected result	64
11.2	Conformance test cases for SMO/Non-RT RIC framework	64
11.2.1	General	64
11.2.1.1	Device under test requirements	64
11.2.1.2	Test simulator capabilities	64
11.2.2	A1 policy management API as API Producer test scenario	64
11.2.2.1	Query A1 policy type identifiers (positive case)	64
11.2.2.1.1	Test description and applicability	64
11.2.2.1.2	Test entrance criteria	64
11.2.2.1.3	Test methodology	64
11.2.2.1.4	Expected result	65
11.2.2.2	Query A1 policy type (positive case)	65
11.2.2.2.1	Test description and applicability	65
11.2.2.2.2	Test entrance criteria	65
11.2.2.2.3	Test methodology	65
11.2.2.2.4	Expected result	66
11.2.2.3	Query A1 policy identifiers (positive case)	66
11.2.2.3.1	Test description and applicability	66
11.2.2.3.2	Test entrance criteria	66
11.2.2.3.3	Test methodology	66
11.2.2.3.4	Expected result	67
11.2.2.4	Create A1 policy (positive case)	67
11.2.2.4.1	Test description and applicability	67
11.2.2.4.2	Test entrance criteria	67
11.2.2.4.3	Test methodology	67
11.2.2.4.4	Expected result	67
11.2.2.5	Query A1 policy (positive case)	68
11.2.2.5.1	Test description and applicability	68
11.2.2.5.2	Test entrance criteria	68
11.2.2.5.3	Test methodology	68
11.2.2.5.4	Expected result	68
11.2.2.6	Update A1 policy (positive case)	68
11.2.2.6.1	Test description and applicability	68
11.2.2.6.2	Test entrance criteria	69
11.2.2.6.3	Test methodology	69
11.2.2.6.4	Expected result	69
11.2.2.7	Delete A1 policy (positive case)	69
11.2.2.7.1	Test description and applicability	69
11.2.2.7.2	Test entrance criteria	69
11.2.2.7.3	Test methodology	70
11.2.2.7.4	Expected result	70
11.2.2.8	Query A1 policy status (positive case)	70

11.2.2.8.1	Test description and applicability	70
11.2.2.8.2	Test entrance criteria	70
11.2.2.8.3	Test methodology	70
11.2.2.8.4	Expected result	71
11.2.2.9	Subscribe A1 policy status (positive case)	71
11.2.2.9.1	Test description and applicability	71
11.2.2.9.2	Test entrance criteria	71
11.2.2.9.3	Test methodology	71
11.2.2.9.4	Expected result	72
11.2.2.10	Update A1 policy status subscription (positive case)	72
11.2.2.10.1	Test description and applicability	72
11.2.2.10.2	Test entrance criteria	72
11.2.2.10.3	Test methodology	72
11.2.2.10.4	Expected result	73
11.2.2.11	Query A1 policy status subscription (positive case)	73
11.2.2.11.1	Test description and applicability	73
11.2.2.11.2	Test entrance criteria	73
11.2.2.11.3	Test methodology	73
11.2.2.11.4	Expected result	74
11.2.2.12	Unsubscribe A1 policy status (positive case)	74
11.2.2.12.1	Test description and applicability	74
11.2.2.12.2	Test entrance criteria	74
11.2.2.12.3	Test methodology	74
11.2.2.12.4	Expected result	74
11.2.2.13	Notify A1 policy status changes (positive case)	75
11.2.2.13.1	Test description and applicability	75
11.2.2.13.2	Test entrance criteria	75
11.2.2.13.3	Test methodology	75
11.2.2.13.4	Expected result	75
Annex A (informative):	Change history	76
History		77

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are 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 IPR online database](#).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, 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.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™**, **LTE™** and **5G™** logo 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.

Foreword

This Technical Specification (TS) has been produced by O-RAN Alliance and approved by ETSI Technical Committee Mobile Standards Group (MSG).

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.

Introduction

The purpose of the present document is to specify conformance and interoperability test cases for the R1 interface. Test cases and methodology for the R1 Service Consumer and R1 Service Producer, are separately specified.

The present document contains R1 test cases for conformance testing and interoperability testing. For each specified test case, cover the entrance criteria, procedure and expected result. The test cases are defined based on the procedures, interface and functional requirements as specified in the R1 specifications series for which the test cases are used to validate conformance and interoperability.

1 Scope

The present document specifies test cases for conformance testing and interoperability testing of the rApps and R1 services over R1 interface.

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found in the [ETSI docbox](#).

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long-term validity.

The following referenced documents are necessary for the application of the present document.

- [1] [ETSI TS 104 228](#): "Publicly Available Specification (PAS); O-RAN R1 interface General Aspects and Principles (O-RAN.WG2.TS.R1GAP-R004-v11.00)".
- [2] [O-RAN.WG2.TS.R1UCR-R004](#): "R1 Use Cases and Requirements" ("R1UCR").
- [3] [O-RAN.WG2.TS.R1TP-R004](#): "Transport Protocols for R1 services" ("R1TP").
- [4] [O-RAN.WG2.TS.R1AP-R004](#): "Application Protocols for R1 services" ("R1AP").
- [5] [O-RAN.WG2.TS.Non-RT-RIC-ARCH-R004](#): "Non-RT RIC: Architecture" ("Non-RT RIC ARCH").

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long-term validity.

The following referenced documents may be useful in implementing an ETSI deliverable or add to the reader's understanding, but are not required for conformance to the present document.

Not applicable.

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in R1GAP [1], R1UCR [2] and R1TP [3] apply.

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in R1GAP [1], R1UCR [2] and R1TP [3] apply.

4 Test methodology

4.1 General

This clause describes the methodology for conformance and interoperability testing of rApps and SMO/Non-Real Time RIC framework over R1 interface.

For conformance tests, simulators are used for testing R1 procedures. These simulators will have capability of generating HTTP requests and responses. There will be flexibility in configuring URI, headers and body for these HTTP requests and responses to enable creation of various test cases.

For interoperability tests, devices under tests are rApps and SMO/Non-RT RIC framework that are defined in the Non-RT RIC architecture specification [5], these devices are brought to operation by connecting to appropriate real or simulated devices.

4.2 Conformance testing of rApps

4.2.1 General

For conformance testing of rApps, rApp is the device under test and SMO/Non-RT RIC framework is the test simulator.

The present document specifies conformance tests for API Consumer and API Producer functionality as specified in R1AP [4].

4.2.2 Test configuration

4.2.2.1 Overview

The test configuration for R1 conformance testing of rApp is illustrated in figure 4.2.2.1-1. For testing of rApp over R1 interface, the rApp is onboarded and instantiated in the cloud environment of the test simulator.

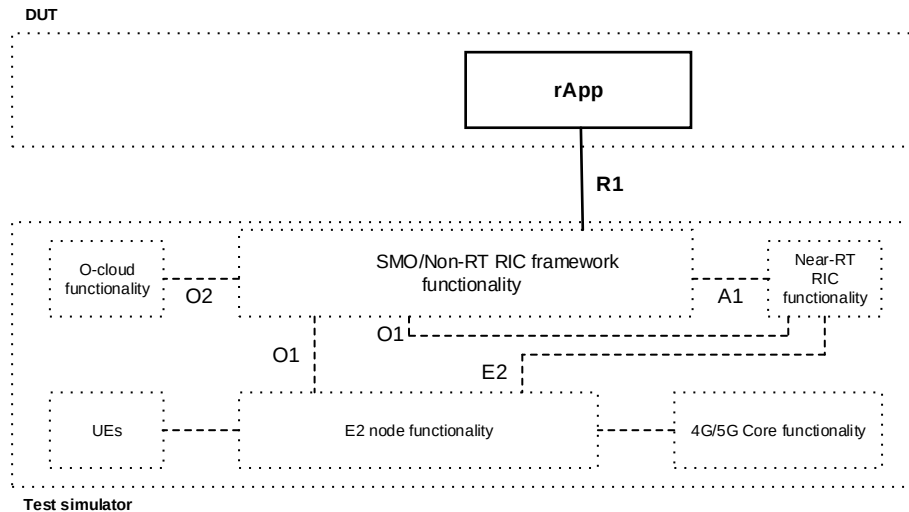


Figure 4.2.2.1-1: Illustration of R1 conformance testing of rApps

4.2.2.2 Device under test rApps

For enabling conformance testing, rApp has implemented API Consumer and/or API Producer functionality and the procedures as specified in R1AP [4] that are required to perform testing of the applicable R1 service test cases.

4.2.2.3 Test simulator

For enabling conformance testing, the test simulator has implemented API Producer and/or API Consumer functionality, have HTTP Client and HTTP Server capabilities, and have flexibility to generate, receive, and validate HTTP messages for all the R1 procedures. The test simulator logs all message content during the testing.

As illustrated in figure 4.2.2.1-1, test simulator has all the capabilities required to execute the conformance testing of an rApp. For example, the test simulator has the capabilities to simulate the functionality of A1, O1, and O2 in a cloud environment.

4.3 Conformance testing of SMO/Non-RT RIC framework

4.3.1 General

For conformance testing of SMO/Non-RT RIC framework, SMO/Non-RT RIC framework is the device under test and rApp is the test simulator.

The present document specifies conformance tests for API Consumer and API Producer functionality as specified in R1AP [4].

4.3.2 Test configuration

4.3.2.1 Overview

The test configuration for R1 conformance testing of SMO/Non-RT RIC framework is illustrated in figure 4.3.2.1-1.

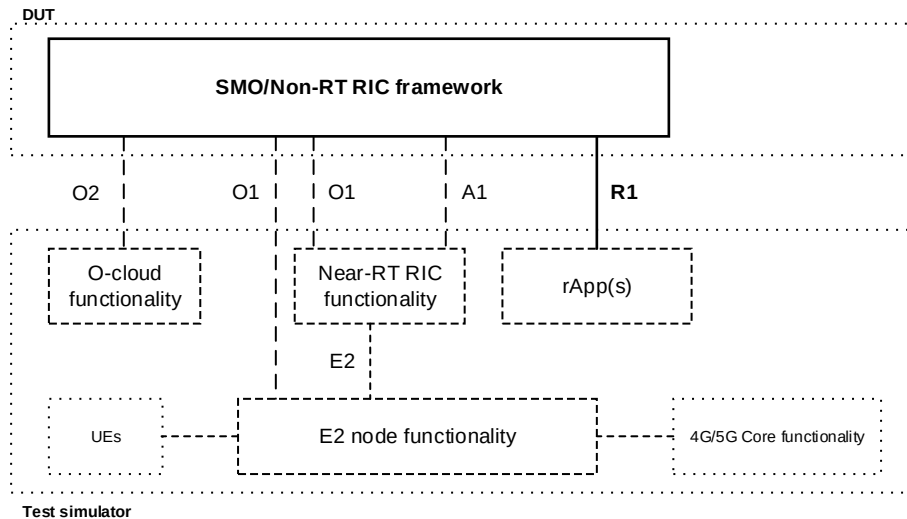


Figure 4.3.2.1-1: Illustration of R1 conformance testing of Non-RT RIC framework

4.3.2.2 Device under test (SMO/Non-RT RIC Framework)

For enabling conformance testing, the SMO/Non-RT RIC framework has implemented API Producer and/or API Consumer functionality, and the procedures specified in R1AP [4] that are required to perform testing of the applicable test cases.

4.3.2.3 Test simulator

For enabling conformance testing, the test simulator rApp(s) has both API Producer and/or API Consumer and HTTP Server capabilities and have flexibility to generate, receive and validate HTTP messages for all the R1 procedures.

As illustrated in figure 4.3.2.1-1, test simulator has all the capabilities required to execute the conformance testing of an SMO/Non-RT RIC framework. For example, the test simulator has the capabilities to simulate the functionality of A1, O1, and O2.

5 SME Service Registration service test cases

5.1 Conformance test cases for rApp

5.1.1 General

5.1.1.1 Device under test requirements

The rApp that acts as Device Under Test (DUT) in the test scenarios of this clause can be a function that is under development, or it can be a finalized commercial product. The requirements on the DUT for these tests are that it can handle the SME service registration service, and the purpose of the test scenarios is to validate that it confirms API Consumer functionality as specified in R1AP [4].

5.1.1.2 Test simulator capabilities

The test simulator has the capabilities as required for a SMO/Non-RT RIC framework. In addition, it has the following capabilities:

- Recording of received HTTP requests and responses and analysing them regarding conformance to the R1 service definitions.
- Controlled initiation of procedures with configurable URIs and payload formulated and modified.

Validating messages and issuing of verdicts related to the procedures in the test cases and thereby enabling determination of the DUT's conformance API Producer functionality as specified in R1AP [4].

5.1.2 Register service API as API Consumer test scenario

5.1.2.1 Register service (positive case)

5.1.2.1.1 Test description and applicability

The purpose of this test case is to test the Register service API as specified in R1AP [4], clause 6.1.4.1. The expected outcome is successful validation of the request from the DUT.

5.1.2.1.2 Test entrance criteria

The DUT has functionality to initiate the Register service procedure as defined in R1GAP [1].

NOTE: The DUT provides the service API description as defined in R1AP [4]. The service being registered can be a standard or non-standard R1 service.

5.1.2.1.3 Test methodology

5.1.2.1.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

5.1.2.1.3.2 Procedure

Step 1 The DUT as an API Consumer initiates a HTTP request to register a service that it produces.

Step 2 At the test simulator the contents of the received HTTP request are recorded.

Step 3 The test simulator does the following validation:

- a) The URI confirms to the format specified in R1AP [4], clause 6.1.5.2.
- b) The HTTP request is a POST operation.
- c) The HTTP request message content includes rApp identifier and the service API description conforms to the schema as specified in R1AP [4].

Step 4 The test simulator generates the service API identifier and constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 6.1.4.1.1.

5.1.2.1.4 Expected result

The test is considered passed if Step 3 validation has passed.

5.1.2.2 Update registered service (positive case)

5.1.2.2.1 Test description and applicability

The purpose of this test case is to test the Update registered service API as specified in R1AP [4], clause 6.1.4.2. The expected outcome is successful validation of the request from the DUT.

5.1.2.2.2 Test entrance criteria

- a) The DUT has functionality to initiate the Update registered service procedure as defined in R1GAP [1].
- b) A service API registration exists in test simulator and the serviceApiId and the apfId are known to DUT.

- c) The schema of the ServiceAPIDescription used for this test are available and used in DUT to formulate the Update registered service request, and in test simulator to validate the request.

NOTE: The DUT provides the ServiceApiDescription as defined in R1AP [4]. The service being updated can be a standard or non-standard R1 service.

5.1.2.2.3 Test methodology

5.1.2.2.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

5.1.2.2.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to update a registered service API identified by the apfId and the serviceApiId.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
- a) The URI confirms to the format specified in R1AP [4], clause 6.1.5.3.
 - b) The HTTP request is a PUT operation.
 - c) The apfId and the serviceApiId in the URI match the service API being updated.
 - d) The HTTP request message body contains the ServiceAPIDescription of the service API to be updated and conforms to the schema as specified in R1AP [4].
- Step 4 The test simulator updates the resource, and a representation of the updated resource shall be returned in the response body the appropriate HTTP response as specified in R1AP [4], clause 6.1.5.3.1.1.

5.1.2.2.4 Expected result

The test is considered passed if Step 3 validation has passed.

5.1.2.3 Deregister service (positive case)

5.1.2.3.1 Test description and applicability

The purpose of this test case is to test the Deregister service API as specified in R1AP [4], clause 6.1.4.3. The expected outcome is successful validation of the request from the DUT.

5.1.2.3.2 Test entrance criteria

- 1) The DUT has functionality to initiate the Deregister service procedure as defined in R1GAP [1].
- 2) A service API registration exists in test simulator and the serviceApiId and the apfId are known to DUT.

5.1.2.3.3 Test methodology

5.1.2.3.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

5.1.2.3.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to deregister a service API identified by the apfId and the serviceApiId.

- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
- a) The URI confirms to the format specified in R1AP [4], clause 6.1.5.3.
 - b) The HTTP request is a DELETE operation.
 - c) The serviceApiId and apfId in the URI match the service being deleted.
 - d) The message body is empty.
- Step 4 The test simulator generated the appropriate HTTP response as specified in R1AP [4], clause 6.1.5.3.1.2.

5.1.2.3.4 Expected result

The test is considered passed if Step 3 validation has passed.

5.1.2.4 Query registered service (positive case)

5.1.2.4.1 Test description and applicability

The purpose of this test case is to test the Query service APIs as specified in R1AP [4], clause 6.1.4.5. The expected outcome is successful validation of the request from the DUT.

5.1.2.4.2 Test entrance criteria

- 1) The DUT has functionality to initiate Query registered service procedure.
- 2) A set of service API registrations exist in the test simulator and the apfId is known to DUT.

5.1.2.4.3 Test methodology

5.1.2.4.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

5.1.2.4.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to query service APIs that are registered by the apfId.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
- a) The URI confirms to the format specified in R1AP [4], clause 6.1.5.3.
 - b) The HTTP request is a GET operation.
 - c) The apfId in the URI match the service APIs being queried.
 - d) The message body is empty.
- Step 4 The test simulator generated the appropriate HTTP response as specified in R1AP [4], clause 6.1.5.2.1.2.

5.1.2.4.4 Expected result

The test is considered passed if Step 3 validation has passed.

5.1.5.5 Partially updated registered service (positive case)

5.1.5.5.1 Test description and applicability

The purpose of this test case is to test the partially update registered service API as specified in R1AP [4], clause 6.1.4.4. The expected outcome is successful validation of the request from the DUT.

5.1.5.5.2 Test entrance criteria

- 1) The DUT has functionality to initiate the Partially update registered service procedure as defined in R1GAP [1].
- 2) A service API registration exists in test simulator and the serviceApiId and the apfId are known to DUT.
- 3) The schema of the ServiceAPIDescriptionPatch used for this test are available and used in DUT to formulate the Partially update registered service request, and in test simulator to validate the request.

NOTE: The DUT provides the ServiceApiDescriptionPatch as defined in R1AP [4]. The service being updated can be a standard or non-standard R1 service.

5.1.5.5.3 Test methodology

5.1.5.5.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

5.1.5.5.3.2 Procedure

- | | |
|--------|---|
| Step 1 | The DUT as an API Consumer initiates a HTTP request to partially update a registered service API identified by the apfId and the serviceApiId. |
| Step 2 | At the test simulator the contents of the received HTTP request are recorded. |
| Step 3 | <p>The test simulator does the following validation:</p> <ol style="list-style-type: none"> a) The URI confirms to the format specified in R1AP [4], clause 6.1.5.3. b) The HTTP request is a PATCH operation. c) The apfId and the serviceApiId in the URI match the service API being updated. d) The HTTP request message body contains the ServiceAPIDescriptionPatch of the service API to be updated and conforms to the schema as specified in R1AP [4]. |
| Step 4 | The test simulator updates the resource, and a representation of the partially updated resource shall be returned in the response body the appropriate HTTP response as specified in R1AP [4], clause 6.1.5.3.1.3. |

5.1.5.5.4 Expected result

The test is considered passed if Step 3 validation has passed.

5.2 Conformance test cases for SMO/Non-RT RIC framework

5.2.1 General

5.2.1.1 Device under test requirements

The SMO/Non-RT RIC framework that acts as Device Under Test (DUT) in the test scenarios of this clause can be a product that is under development, or it can be a finalized commercial product. The requirements on the DUT for these tests are that it can handle the SME service registration service, and the purpose of the test scenarios is to validate that it confirms API Producer functionality as specified in R1AP [4].

5.2.1.2 Test simulator capabilities

The test simulator has the capabilities as specified in clause 4.3.2. In addition, it has the following capabilities:

- 1) Recording of received HTTP requests and responses and analysing them regarding conformance to the R1 service definitions.
- 2) Controlled initiation of procedures with configurable URIs and payload formulated and modified.

Validating messages and issuing of verdicts related to the procedures in the test cases and thereby enabling determination of the DUT's conformance to API Producer functionality as specified in R1AP [4].

5.2.2 Register service API as API Producer test scenario

5.2.2.1 Register service (positive case)

5.2.2.1.1 Test description and applicability

The purpose of this test case is to test the Register service API as specified in R1AP [4], clause 6.1.4.1. The expected outcome is successful validation of the response from the DUT.

5.2.2.1.2 Test entrance criteria

The test simulator has functionality to initiate the Register service procedure as defined in R1GAP [1].

NOTE: The test simulator provides the service API description as defined in R1AP [4]. The service being registered can be a standard or non-standard R1 service.

5.2.2.1.3 Test methodology

5.2.2.1.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP responses from the DUT.

5.2.2.1.3.2 Procedure

- | | |
|--------|---|
| Step 1 | The test simulator as an API Consumer initiates a HTTP request to register a service that it produces. |
| Step 2 | The DUT receives the HTTP request. |
| Step 3 | The DUT does the following validation: <ol style="list-style-type: none"> a) The HTTP request is a POST request. b) The URI conforms to the format specified in R1AP [4], clause 6.1.5.2.1.1. c) The HTTP request message content includes the information as specified in R1AP [4]. |

- Step 4 The DUT generates the service API identifier and constructs the URI for the created resource.
- Step 5 The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 6.1.4.1.1.
- Step 6 At the test simulator the contents of the received HTTP response are recorded.
- Step 7 The test simulator does the following validation:
- a) The HTTP response message content includes the information as specified in R1AP [4], and the Location header will be present.
 - b) The URI confirms to the format specified in R1AP [4], clause 6.1.5.2.

5.2.2.1.4 Expected result

The test is considered passed if Step 7 validation has passed.

5.2.2.2 Update registered service (positive case)

5.2.2.2.1 Test description and applicability

The purpose of this test case is to test the Update registered service API as specified in R1AP [4], clause 6.1.4.2. The expected outcome is successful validation of the response from the DUT.

5.2.2.2.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the Update registered service procedure as defined in R1GAP [1].
- 2) A service API registration exists in DUT and the serviceApiId and the apfId are known to test simulator.
- 3) The schema of the ServiceAPIDescription used for this test are available and used in test simulator to formulate the Update registered service request, and in the test simulator to validate the request.

NOTE: The test simulator provides the ServiceApiDescription as defined in R1AP [4]. The service being updated can be a standard or non-standard R1 service.

5.2.2.2.3 Test methodology

5.2.2.2.3.1 Initial conditions

The DUT as API Producer is ready and available to receive HTTP requests from the test simulator.

5.2.2.2.3.2 Procedure

- Step 1 The test simulator as an API Consumer initiates a HTTP request to update a registered service API identified by the apfId and the serviceApiId.
- Step 2 The DUT receives the HTTP request.
- Step 3 The DUT does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 6.1.5.3.
 - b) The HTTP request is a PUT operation.
 - c) The apfId and the serviceApiId in the URI match the service API being updated.
 - d) The HTTP request message content includes the information as specified in R1AP [4].
- Step 4 The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 6.1.4.2.1.
- Step 5 At the test simulator the contents of the received HTTP response are recorded.

Step 6 The test simulator does the following validation:

- a) The HTTP response message content includes the information as specified in R1AP [4], and the Location header will be present.

5.2.2.2.4 Expected result

The test is considered passed if Step 6 validation has passed.

5.2.2.3 Deregister service (positive case)

5.2.2.3.1 Test description and applicability

The purpose of this test case is to test the Deregister service API as specified in R1AP [4], clause 6.1.4.3. The expected outcome is successful validation of the response from the DUT.

5.2.2.3.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the Deregister service procedure as defined in R1GAP [1].
- 2) A service API registration exists in DUT and the serviceApiId and the apfId are known to test simulator.

5.2.2.3.3 Test methodology

5.2.2.3.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP requests from the test simulator.

5.2.2.3.3.2 Procedure

Step 1 The test simulator as an API Consumer initiates a HTTP request to deregister a service API identified by the apfID and the serviceApiId.

Step 2 The DUT receives the HTTP request.

Step 3 The DUT does the following validation:

- a) The HTTP request is a DELETE request.
- b) The URI format as specified in R1AP [4], clause 6.1.5.3.1.2.
- c) The HTTP request content includes the information as specified in R1AP [4].

Step 4 The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 6.1.4.3.1.

Step 5 At the test simulator the contents of the received HTTP response are recorded.

Step 6 The test simulator does the following validation:

- a) The URI confirms to the format specified in R1AP [4], clause 6.1.5.3.
- b) The HTTP response message content includes the information as specified in R1AP [4].
- c) The message body is empty.

5.2.2.3.4 Expected result

The test is considered passed if Step 6 validation has passed.

5.2.2.4 Query registered service (positive case)

5.2.2.4.1 Test description and applicability

The purpose of this test case is to test the Query service APIs as specified in R1AP [4], clause 6.1.4.5. The expected outcome is successful validation of the response from the DUT.

5.2.2.4.2 Test entrance criteria

- 1) The test simulator has functionality to initiate Query registered service procedure.
- 2) A set of service API registrations exist in the DUT and the apfId is known to test simulator.

5.2.2.4.3 Test methodology

5.2.2.4.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP responses from the DUT.

5.2.2.4.3.2 Procedure

- | | |
|--------|---|
| Step 1 | The test simulator as an API Consumer initiates a HTTP request to query service APIs that are registered by the apfId. |
| Step 2 | The DUT receives the HTTP request. |
| Step 3 | The DUT does the following validation: <ol style="list-style-type: none"> a) The HTTP request is a GET request. b) The URI format as specified in R1AP [4], clause 6.1.5.2.1.2. c) The HTTP request message content includes the information as specified in R1AP [4]. |
| Step 4 | The DUT initiates a HTTP Response. |
| Step 5 | At the test simulator the contents of the received HTTP response are recorded. |
| Step 6 | The test simulator does the following validation: <ol style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 6.1.5.2. b) The HTTP response message content includes the information as specified in R1AP [4]. |

5.2.2.4.4 Expected result

The test is considered passed if Step 6 validation has passed.

5.2.5.5 Partially updated registered service (positive case)

5.2.5.5.1 Test description and applicability

The purpose of this test case is to test the partially update registered service API as specified in R1AP [4], clause 6.1.4.4. The expected outcome is successful validation of the response from the DUT.

5.2.5.5.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the Partially update registered service procedure as defined in R1GAP [1].
- 2) A service API registration exists in DUT and the serviceApiId and the apfId are known to test simulator.

- 3) The schema of the ServiceAPIDescriptionPatch used for this test are available and used in test simulator to formulate the Partially update registered service request, and in test simulator to validate the request.

NOTE: The test simulator provides the ServiceApiDescriptionPatch as defined in R1AP [4]. The service being updated can be a standard or non-standard R1 service.

5.2.5.5.3 Test methodology

5.2.5.5.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP responses from the DUT.

5.2.5.5.3.2 Procedure

- | | |
|--------|--|
| Step 1 | The test simulator as an API Consumer initiates a HTTP request to partially update a registered service API identified by the apfId and the serviceApiId. |
| Step 2 | The DUT receives the HTTP request. |
| Step 3 | <p>The DUT does the following validation:</p> <ul style="list-style-type: none"> a) The HTTP request is a PATCH request. b) The URI format as specified in R1AP [4], clause 6.1.5.3.1.3. c) The HTTP request message content includes the information as specified in R1AP [4]. |
| Step 4 | The DUT initiates a HTTP Response. |
| Step 5 | At the test simulator the contents of the received HTTP request are recorded. |
| Step 6 | <p>The test simulator does the following validation:</p> <ul style="list-style-type: none"> a) The HTTP response message content includes the information as specified in R1AP [4], and the Location header will be present. |

5.2.5.5.4 Expected result

The test is considered passed if Step 6 validation has passed.

6 SME Service Discovery service test cases

6.1 Conformance test cases for rApp

6.1.1 General

6.1.1.1 Device under test requirements

The rApp that acts as Device Under Test (DUT) in the test scenarios of this clause can be a function that is under development, or it can be a finalized commercial product. The requirements on the DUT for these tests are that it can handle the SME service discovery service, and the purpose of the test scenarios is to validate that it conforms to the Service discovery API specified in R1AP [4].

6.1.1.2 Test simulator capabilities

The test simulator has the capabilities as required for a SMO/Non-RT RIC framework. In addition, it has the following capabilities:

- Recording of received HTTP requests and responses and analysing them regarding conformance to the R1 service definitions.
- Controlled initiation of procedures with configurable URIs and payload formulated and modified.

Validating messages and issuing of verdicts related to the procedures in the test cases and thereby enabling determination of the DUT's conformance API Producer functionality as specified in R1AP [4].

6.1.2 Discover service API as API Consumer test scenario

6.1.2.1 Query service (positive case)

6.1.2.1.1 Test description and applicability

The purpose of this test case is to test the Query service APIs operation as specified in R1AP [4], clause 6.2.4.1. The expected outcome is successful validation of the request from the DUT.

6.1.2.1.2 Test entrance criteria

- 1) The DUT has functionality to initiate the Discover services procedure.
- 2) A set of service API registrations exist in the test simulator.

6.1.2.1.3 Test methodology

6.1.2.1.3.1 Initial conditions

- 1) The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

6.1.2.1.3.2 Procedure

- | | |
|--------|--|
| Step 1 | The DUT as an API Consumer initiates a HTTP request to query service APIs. |
| Step 2 | At the test simulator the contents of the received HTTP request are recorded. |
| Step 3 | The test simulator does the following validation: <ol style="list-style-type: none"> a) The URI confirms to the format specified in R1AP [4], clause 6.2.5.2. b) The HTTP request is a GET operation. c) The message body is empty. |
| Step 4 | The test simulator generated the appropriate HTTP response as specified in R1AP [4], clause 6.2.5.2.1.1. |

NOTE: Presence or validation of optional filter parameters is not used to determine validation on this test.

6.1.2.1.4 Expected result

The test is considered passed if Step 3 validation has passed.

6.2 Conformance test cases for SMO/Non-RT RIC framework

6.2.1 General

6.2.1.1 Device under test requirements

The SMO/Non-RT RIC framework that acts as Device Under Test (DUT) in the test scenarios of this clause can be a function that is under development, or it can be a finalized commercial product. The requirements on the DUT for these tests are that it can handle the SME service discovery service, and the purpose of the test scenarios is to validate that it conforms to the Service discovery API specified in R1AP [4], clause 6.2.

6.2.1.2 Test simulator capabilities

The test simulator has the capabilities as specified in clause 4.3.2. In addition, it has the following capabilities:

- 1) Recording of received HTTP requests and responses and analysing them regarding conformance to the R1 service definitions.
- 2) Controlled initiation of procedures with configurable URIs and payload formulated and modified.

Validating messages and issuing of verdicts related to the procedures in the test cases and thereby enabling determination of the DUT's conformance to API Producer functionality as specified in R1AP [4], clause 6.2.

6.2.2 Service discovery API as API Producer test scenario

6.2.2.1 Query service (positive case)

6.2.2.1.1 Test description and applicability

The purpose of this test case is to test the Query service APIs as specified in R1AP [4], clause 6.2.4.1. The expected outcome is successful validation of the response from the DUT.

6.2.2.1.2 Test entrance criteria

- 1) The DUT has functionality to initiate the Discover services procedure.
- 2) A set of service API registrations exist in the test simulator.

6.2.2.1.3 Test methodology

6.2.2.1.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP responses from the DUT.

6.2.2.1.3.2 Procedure

- | | |
|--------|---|
| Step 1 | The test simulator as an API Consumer initiates a HTTP request to query service APIs that are registered by the apfld. |
| Step 2 | The DUT receives the HTTP request. |
| Step 3 | The DUT does the following validation: <ol style="list-style-type: none"> a) The HTTP request is a GET request. b) The URI conforms to the format as specified in R1AP [4], clause 6.2.5.2. c) The HTTP request message content includes the information as specified in R1AP [4], clause 6.2.4.1. |

- Step 4 The DUT initiates a HTTP Response.
- Step 5 At the test simulator the contents of the received HTTP response are recorded.
- Step 6 The test simulator does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 6.2.5.2.
 - b) The HTTP response message content includes the information as specified in R1AP [4], clause 6.2.4.1.

6.2.2.1.4 Expected result

The test is considered passed if Step 6 validation has passed.

7 SME Service Subscription service test cases

7.1 Conformance test cases for rApp

7.1.1 General

7.1.1.1 Device under test requirements

The rApp that acts as DUT in the test scenarios of this clause can be a function that is under development, or it can be a finalized commercial product. The requirements on the DUT for these tests are that it can handle the SME service subscription service, and the purpose of the test scenarios is to validate that it conforms to the API Consumer functionality as specified in R1AP [4], clause 6.3.

7.1.1.2 Test simulator capabilities

The test simulator has the capabilities as required for a SMO/Non-RT RIC framework. In addition, it has the following capabilities:

- Recording of received HTTP requests and responses and analysing them regarding conformance to the R1 service definitions.
- Controlled initiation of procedures with configurable URIs and payload formulated and modified.

Validating messages and issuing of verdicts related to the procedures in the test cases and thereby enabling determination of the DUT's conformance to the API Consumer functionality as specified in R1AP [4], clause 6.3.

7.1.2 Service events subscription API as API Consumer test scenario

7.1.2.1 Subscribe service events (positive case)

7.1.2.1.1 Test description and applicability

The purpose of this test case is to test the Events subscription service API as specified in R1AP [4], clause 6.3.4.1. The expected outcome is successful validation of the request from the DUT.

7.1.2.1.2 Test entrance criteria

The DUT has functionality to initiate the Subscribe service availability procedure as defined in R1GAP [1], clause 5.1.4.

NOTE: The DUT provides the events subscription structure as defined in R1AP [4], clause 6.3.3. The service event being subscribed to can be a standard or non-standard R1 service event.

7.1.2.1.3 Test methodology

7.1.2.1.3.1 Initial conditions

- 1) The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.
- 2) The subscriberId is known to DUT.

7.1.2.1.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to subscribe to service event notifications.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
 - a) The URI conforms to the format specified in R1AP [4], clause 6.3.5.2.
 - b) The HTTP request is a POST operation.
 - c) The HTTP request message content includes the information as specified in R1AP [4], clause 6.3.4.1.1.
- Step 4 The test simulator generates the subscription identifier and constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 6.3.4.1.1.

7.1.2.1.4 Expected result

The test is considered passed if Step 3 validation has passed.

7.1.2.2 Unsubscribe service events (positive case)

7.1.2.2.1 Test description and applicability

The purpose of this test case is to test the unsubscribe service API as specified in R1AP [4], clause 6.3.4.2. The expected outcome is successful validation of the request from the DUT.

7.1.2.2.2 Test entrance criteria

- 1) The DUT has functionality to initiate the Unsubscribe service availability procedure as defined in R1GAP [1], clause 5.1.4.
- 2) A subscription exists and the subscriberId and subscriptionId are known to the DUT.

7.1.2.2.3 Test methodology

7.1.2.2.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

7.1.2.2.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to unsubscribe from service event notifications identified by the subscriberId and subscriptionId.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
 - a) The URI conforms to the format specified in R1AP [4], clause 6.3.5.3.
 - b) The HTTP request is a DELETE operation.

- c) The subscriberId and subscriptionId in the URI match the service events being deleted.
- d) The message body is empty.

Step 4 The test simulator generated the appropriate HTTP response as specified in R1AP [4], clause 6.3.5.3.1.1.

7.1.2.2.4 Expected result

The test is considered passed if Step 3 validation has passed.

7.2 Conformance test cases for SMO/Non-RT RIC framework

7.2.1 General

7.2.1.1 Device under test requirements

The SMO/Non-RT RIC framework that acts as DUT in the test scenarios of this clause can be a product that is under development, or it can be a finalized commercial product. The requirements on the DUT for these tests are that it can handle the SME service events subscription service, and the purpose of the test scenarios is to validate that it conforms to the API Producer functionality as specified in R1AP [4], clause 6.3.

7.2.1.2 Test simulator capabilities

The test simulator has the capabilities as specified in clause 4.3.2. In addition, it has the following capabilities:

- 1) Recording of received HTTP requests and responses and analysing them regarding conformance to the R1 service definitions.
- 2) Controlled initiation of procedures with configurable URIs and payload formulated and modified.

Validating messages and issuing of verdicts related to the procedures in the test cases and thereby enabling determination of the DUT's conformance to the API Producer functionality as specified in R1AP [4], clause 6.3.

7.2.2 Service events subscription API as API Producer test scenario

7.2.2.1 Service events subscription (positive case)

7.2.2.1.1 Test description and applicability

The purpose of this test case is to test the Events subscription service API as specified in R1AP [4], clause 6.3.4.1. The expected outcome is successful validation of the response from the DUT.

7.2.2.1.2 Test entrance criteria

The test simulator has functionality to initiate the Subscribe service availability procedure as defined in R1GAP [1], clause 5.1.4.

NOTE: The test simulator provides the events subscription structure as defined in R1AP [4], clause 6.3.3. The service being subscribed to can be a standard or non-standard R1 service.

7.2.2.1.3 Test methodology

7.2.2.1.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP responses from the DUT.

7.2.2.1.3.2 Procedure

- Step 1 The test simulator as an API Consumer initiates a HTTP request to subscribe to service event notifications.
- Step 2 The DUT receives the HTTP request.
- Step 3 The DUT does the following validation:
- a) The HTTP request is a POST request.
 - b) The URI conforms to the format specified in R1AP [4], clause 6.3.5.2.1.1.
 - c) The HTTP request message content includes the information as specified in R1AP [4], clause 6.3.4.1.1.
- Step 4 The DUT generates the subscription identifier and constructs the URI for the created resource.
- Step 5 The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 6.3.4.1.1.
- Step 6 At the test simulator the contents of the received HTTP response are recorded.
- Step 7 The test simulator does the following validation:
- a) The HTTP response message content includes the information as specified in R1AP [4], clause 6.3.4.1.1, and the Location header will be present.
 - b) The URI conforms to the format specified in R1AP [4], clause 6.3.5.2.

7.2.2.1.4 Expected result

The test is considered passed if Step 7 validation has passed.

7.2.2.2 Unsubscribe service events (positive case)

7.2.2.2.1 Test description and applicability

The purpose of this test case is to test the Unsubscribe service API as specified in R1AP [4], clause 6.3.4.2. The expected outcome is successful validation of the response from the DUT.

7.2.2.2.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the Unsubscribe service availability procedure as defined in R1GAP [1], clause 5.1.4.
- 2) A subscription exists and the subscriberId and subscriptionId are known to the test simulator.

7.2.2.2.3 Test methodology

7.2.2.2.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP requests from the DUT.

7.2.2.2.3.2 Procedure

- Step 1 The test simulator as an API Consumer initiates a HTTP request to unsubscribe from service event notifications identified by the subscriberId and subscriptionId.
- Step 2 The DUT receives the HTTP request.
- Step 3 The DUT does the following validation:
- a) The HTTP request is a DELETE request.

- b) The URI format as specified in R1AP [4], clause 6.3.5.3.1.1.
- c) The subscriberId and subscriptionId in the URI match the service events being deleted.
- d) The message body is empty.

Step 4 The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 6.3.4.2.1.

Step 5 At the test simulator the contents of the received HTTP response are recorded.

Step 6 The test simulator does the following validation:

- a) The return code is "204 No content".
- b) The response message body is empty.

7.2.2.2.4 Expected result

The test is considered passed if Step 6 validation has passed.

7.2.2.3 Notification service (positive case)

7.2.2.3.1 Test description and applicability

The purpose of this test case is to test the notification service API as specified in R1AP [4], clause 6.3.4.3. The expected outcome is successful validation of the event notification from the DUT.

7.2.2.3.2 Test entrance criteria

- 1) The DUT has functionality to initiate the Notify service availability changes procedure as defined in R1GAP [1], clause 5.1.4.
- 2) A subscription exists in the DUT.

7.2.2.3.3 Test methodology

7.2.2.3.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP requests from the DUT.

7.2.2.3.3.2 Procedure

- Step 1 The DUT as an API producer initiates the sending of a HTTP POST request.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
 - a) The HTTP request is a POST operation.
 - b) The callback URI matches the URI provided in the subscription.

7.2.2.3.4 Expected result

The test is considered passed if Step 3 validation has passed.

8 DME Data registration service test cases

8.1 Conformance test cases for rApp

8.1.1 General

8.1.1.1 Device under test requirements

The rApp that acts as DUT in the test scenarios of this clause can be a function that is under development, or it can be a finalized commercial product. The requirements on the DUT for these tests are that it can handle the Data registration service, and the purpose of the test scenarios is to validate that it conforms to the API Consumer functionality as specified in R1AP [4], clause 7.1.4.

8.1.1.2 Test simulator capabilities

The test simulator has the capabilities as required for a SMO/Non-RT RIC framework. In addition, it has the following capabilities:

- Recording of received HTTP requests and responses and analysing them regarding conformance to the R1 service definitions.
- Controlled initiation of procedures with configurable URIs and payload formulated and modified.

Validating messages and issuing of verdicts related to the procedures in the test cases and thereby enabling determination of the DUT's conformance to the API Consumer functionality as specified in R1AP [4], clause 7.1.4.

8.1.2 Data registration API as API consumer test scenario

8.1.2.1 Register DME type (positive case)

8.1.2.1.1 Test description and applicability

The purpose of this test case is to test the Register DME type in Data registration API as specified in R1AP [4], clause 7.1. The expected outcome is successful validation of the request from the DUT.

8.1.2.1.2 Test entrance criteria

The DUT has functionality to initiate the Register DME type procedure as defined in R1GAP [1].

NOTE: The DUT provides the service API description as defined in R1AP [4]. The service being registered can be a standard or non-standard R1 service.

8.1.2.1.3 Test methodology

8.1.2.1.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

8.1.2.1.3.2 Procedure

- | | |
|--------|--|
| Step 1 | The DUT as an API Consumer initiates a HTTP request to register a DmeTypeRelatedCapabilities. |
| Step 2 | At the test simulator the contents of the received HTTP request are recorded. |
| Step 3 | The test simulator does the following validation: <ul style="list-style-type: none"> a) The URI confirms to the format specified in R1AP [4], clause 7.1.5.2.2. |

- b) The HTTP request is a POST operation.
- c) The HTTP request message content includes DmeTypeRelatedCapabilities as specified in R1AP [4].

Step 4 The test simulator generates the registrationId and constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 7.1.5.2.3.1.

8.1.2.1.4 Expected result

The test is considered passed if Step 3 validation has passed.

8.1.2.2 Deregister DME type (positive case)

8.1.2.2.1 Test description and applicability

The purpose of this test case is to test the Deregister DME type in Data registration API as specified in R1AP [4], clause 7.1. The expected outcome is successful validation of the request from the DUT.

8.1.2.2.2 Test entrance criteria

- 1) The DUT has functionality to initiate the Deregister DME type procedure as defined in R1GAP [1], clause 5.2.2.2.
- 2) A DME type exists in test simulator and the registrationId is known to the DUT.

8.1.2.2.3 Test methodology

8.1.2.2.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

8.1.2.2.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to Deregister DME type.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
 - a) The URI conforms to the format specified in R1AP [4], clause 7.1.5.3.2.
 - b) The HTTP request is a DELETE operation.
 - c) The registrationId in the URI match the DME type being deregistered.
 - d) The HTTP request message content includes the information as specified in R1AP [4], clause 7.1.5.3.3.
- Step 4 The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 7.1.5.3.3.

8.1.2.2.4 Expected result

The test is considered passed if Step 3 validation has passed.

8.1.2.3 Query DME type (positive case)

8.1.2.3.1 Test description and applicability

The purpose of this test case is to test the Query DME type in Data registration API as specified in R1AP [4], clause 7.1.4.4. The expected outcome is successful validation of the request from the DUT.

8.1.2.3.2 Test entrance criteria

- 1) The DUT has functionality to initiate the query DME type registration procedure as defined in R1GAP [1], clause 5.2.2.2.
- 2) A DME type exists in test simulator and the registrationId is known to the DUT.

8.1.2.3.3 Test methodology

8.1.2.3.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

8.1.2.3.3.2 Procedure

- | | |
|--------|--|
| Step 1 | The DUT as an API Consumer initiates a HTTP request to query a DME type. |
| Step 2 | At the test simulator the contents of the received HTTP request are recorded. |
| Step 3 | <p>The test simulator does the following validation:</p> <ol style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 7.1.5.3.2. b) The HTTP request is a GET operation. c) The registrationId in the URI match the DME type being queried. d) The HTTP request message content includes the information as specified in R1AP [4], clause 7.1.5.3.3. |
| Step 4 | The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 7.1.5.3.3. |

8.1.2.3.4 Expected result

The test is considered passed if Step 3 validation has passed.

8.1.2.4 Update DME type (positive case)

8.1.2.4.1 Test description and applicability

The purpose of this test case is to test the Update DME type in Data registration API as specified in R1AP [4], clause 7.1.4.3. The expected outcome is successful validation of the request from the DUT.

8.1.2.4.2 Test entrance criteria

- 1) The DUT has functionality to initiate the Update DME type registration procedure as defined in R1GAP [1], clause 5.2.2.2.
- 2) A DME type exists in test simulator and the registrationId and DmeTypeRelatedCapabilities are known to the DUT.

8.1.2.4.3 Test methodology

8.1.2.4.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

8.1.2.4.3.2 Procedure

- | | |
|--------|---|
| Step 1 | The DUT as an API Consumer initiates a HTTP request to update a DME type identified by the registrationId and DmeTypeRelatedCapabilities. |
| Step 2 | At the test simulator the contents of the received HTTP request are recorded. |
| Step 3 | <p>The test simulator does the following validation:</p> <ul style="list-style-type: none"> a) The URI confirms to the format specified in R1AP [4], clause 7.1.5.3.2. b) The HTTP request is a PUT operation. c) The registrationId and DmeTypeRelatedCapabilities in the URI match the DME type being updated. d) The HTTP request message content includes the information as specified in R1AP [4], clause 7.1.5.3.3. |
| Step 4 | The test simulator updates the resource, and a representation of the updated resource shall be returned in the response body the appropriate HTTP response as specified in R1AP [4], clause 7.1.5.3.3. |

8.1.2.4.4 Expected result

The test is considered passed if Step 3 validation has passed.

8.2 Conformance test cases for SMO/Non-RT RIC framework

8.2.1 General

8.2.1.1 Device under test requirements

The SMO/Non-RT RIC framework that acts as DUT in the test scenarios of this clause can be a function that is under development, or it can be a finalized commercial product. The requirements on the DUT for these tests are that it can handle the Data registration service, and the purpose of the test scenarios is to validate that it conforms to the API Producer functionality as specified in R1AP [4], clause 7.1.4.

8.2.1.2 Test simulator capabilities

The test simulator has the capabilities as required for a rApp. In addition, it has the following capabilities:

- Recording of received HTTP requests and responses and analysing them regarding conformance to the R1 service definitions.
- Controlled initiation of procedures with configurable URIs and payload formulated and modified.

Validating messages and issuing of verdicts related to the procedures in the test cases and thereby enabling determination of the DUT's conformance to the API Producer functionality as specified in R1AP [4], clause 7.1.4.

8.2.2 Data registration API as API Producer test scenario

8.2.2.1 Register DME type (positive case)

8.2.2.1.1 Test description and applicability

The purpose of this test case is to test the Register DME type in Data registration API as specified in R1AP [4], clause 7.1. The expected outcome is successful validation of the request from the DUT.

8.2.2.1.2 Test entrance criteria

The test simulator has functionality to initiate the Register DME type procedure as defined in R1GAP [1].

8.2.2.1.3 Test methodology

8.2.2.1.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP responses from the DUT.

8.2.2.1.3.2 Procedure

- | | |
|--------|--|
| Step 1 | The test simulator as an API Consumer initiates a HTTP request to register a DmeTypeRelatedCapabilities. |
| Step 2 | The DUT receives the HTTP request. |
| Step 3 | The DUT does the following validation: <ul style="list-style-type: none"> a) The HTTP request is a POST request. b) The URI conforms to the format specified in R1AP [4], clause 7.1.5.2.2. c) The HTTP request message content includes the information as specified in R1AP [4], clause 7.1.5.2.3.1. |
| Step 4 | The DUT generates the data job and constructs the URI for the created resource. |
| Step 5 | The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 7.1.5.2.3.1. |
| Step 6 | At the test simulator the contents of the received HTTP response are recorded. |
| Step 7 | The test simulator does the following validation: <ul style="list-style-type: none"> a) The HTTP response message content includes the information as specified in R1AP [4], clause 7.1.5.2.3.1, and the Location header will be present. b) The URI conforms to the format specified in R1AP [4], clause 7.1.5.2.2. |

8.2.2.1.4 Expected result

The test is considered passed if Step 7 validation has passed.

8.2.2.2 Deregister DME type (positive case)

8.2.2.2.1 Test description and applicability

The purpose of this test case is to test the Deregister DME type in Data registration API as specified in R1AP [4]. The expected outcome is successful validation of the request from the DUT.

8.2.2.2.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the Deregister DME type procedure as defined in R1GAP [1], clause 5.2.2.2.
- 2) A DME type exists in test simulator and the registrationId is known to the DUT.

8.2.2.2.3 Test methodology

8.2.2.2.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP responses from the DUT.

8.2.2.2.3.2 Procedure

- | | |
|--------|---|
| Step 1 | The test simulator as an API Consumer initiates a HTTP request to Deregister DME type. |
| Step 2 | The DUT receives the HTTP request. |
| Step 3 | <p>The DUT does the following validation:</p> <ol style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 7.1.5.3.2. b) The HTTP request is a DELETE operation. c) The registrationId in the URI match the DME type being deregistered. d) The HTTP request message content includes the information as specified in R1AP [4], clause 7.1.5.3.3. |
| Step 4 | The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 7.1.5.3.3. |
| Step 5 | At the test simulator the contents of the received HTTP response are recorded. |
| Step 6 | <p>The test simulator does the following validation:</p> <ol style="list-style-type: none"> a) The URI confirms to the format specified in R1AP [4], clause 7.1.5.3.2. b) The HTTP response message content includes the information as specified in R1AP [4], clause 7.1.5.3.3. |

8.2.2.2.4 Expected result

The test is considered passed if Step 6 validation has passed.

8.2.2.3 Query DME type (positive case)

8.2.2.3.1 Test description and applicability

The purpose of this test case is to test the Query DME type in Data registration API as specified in R1AP [4], clause 7.1.4.4. The expected outcome is successful validation of the request from the DUT.

8.2.2.3.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the query DME type registration procedure as defined in R1Gap [1], clause 5.2.2.2.
- 2) A DME type exists in test simulator and the registrationId is known to the DUT.

8.2.2.3.3 Test methodology

8.2.2.3.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP responses from the DUT.

8.2.2.3.3.2 Procedure

- Step 1 The test simulator as an API Consumer initiates a HTTP request to query a DME type.
- Step 2 The DUT receives the HTTP request.
- Step 3 The DUT does the following validation:
 - a) The URI conforms to the format specified in R1AP [4], clause 7.1.5.3.2.
 - b) The HTTP request is a GET operation.
 - c) The registrationId in the URI match the data job status being queried.
 - d) The HTTP request message content includes the information as specified in R1AP [4], clause 7.1.5.3.3.
- Step 4 The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 7.1.5.3.3.
- Step 5 At the test simulator the contents of the received HTTP response are recorded.
- Step 6 The test simulator does the following validation:
 - a) The URI conforms to the format specified in R1AP [4], clause 7.1.5.3.2.
 - b) The HTTP response message content includes the information as specified in R1AP [4], clause 7.1.5.3.3.

8.2.2.3.4 Expected result

The test is considered passed if Step 6 validation has passed.

8.2.2.4 Update DME type (positive case)

8.2.2.4.1 Test description and applicability

The purpose of this test case is to test the Update DME type in Data registration API as specified in R1AP [4], clause 7.1.4.3. The expected outcome is successful validation of the request from the DUT.

8.2.2.4.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the Update DME type registration procedure as defined in R1GAP [1], clause 5.2.2.2
- 2) A DME type exists in test simulator and the registrationId and DmeTypeRelatedCapabilities are known to the DUT.

8.2.2.4.3 Test methodology

8.2.2.4.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP responses from the DUT.

8.2.2.4.3.2 Procedure

- Step 1 The test simulator as an API Consumer initiates a HTTP request to update a DME type identified by the registrationId and DmeTypeRelatedCapabilities.
- Step 2 The DUT receives the HTTP request.
- Step 3 The DUT does the following validation:
- a) The HTTP request is a PUT request.
 - b) The URI conforms to the format specified in R1AP [4], clause 7.1.5.3.2.
 - c) The HTTP request message content includes the information as specified in R1AP [4], clause 7.1.5.3.3.
- Step 4 The DUT generates the data job and constructs the URI for the created resource.
- Step 5 The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 7.1.5.3.3.
- Step 6 At the test simulator the contents of the received HTTP response are recorded.
- Step 7 The test simulator does the following validation:
- a) The HTTP response message content includes the information as specified in R1AP [4], clause 7.1.5.3.3.
 - b) The URI conforms to the format specified in R1AP [4], clause 7.1.5.3.2.

8.2.2.4.4 Expected result

The test is considered passed if Step 7 validation has passed.

9 DME Data request service test cases

9.1 Conformance test cases for rApp

9.1.1 General

9.1.1.1 Device under test requirements

The rApp that acts as DUT in the test scenarios of this clause can be a function that is under development, or it can be a finalized commercial product. The requirements on the DUT for these tests are that it can handle the DME Data request service, and the purpose of the test scenarios is to validate that it conforms to the API Consumer functionality as specified in R1AP [4], clause 7.3.4.

9.1.1.2 Test simulator capabilities

The test simulator has the capabilities as required for a SMO/Non-RT RIC framework. In addition, it has the following capabilities:

- Recording of received HTTP requests and responses and analysing them regarding conformance to the R1 service definitions.
- Controlled initiation of procedures with configurable URIs and payload formulated and modified.

Validating messages and issuing of verdicts related to the procedures in the test cases and thereby enabling determination of the DUT's conformance to the API Consumer functionality as specified in R1AP [4], clause 7.3.4.

9.1.2 Data access API as API Consumer test scenario

9.1.2.1 Create data job (positive case)

9.1.2.1.1 Test description and applicability

The purpose of this test case is to test the create data job API as specified in R1AP [4], clause 7.3.4.1. The expected outcome is successful validation of the request from the DUT.

9.1.2.1.2 Test entrance criteria

- 1) The DUT has functionality to initiate the request data procedure as defined in R1GAP [1], clause 5.2.4.
- 2) The DataJobInfo is known to the DUT.

9.1.2.1.3 Test methodology

9.1.2.1.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

9.1.2.1.3.2 Procedure

- | | |
|--------|---|
| Step 1 | The DUT as an API Consumer initiates a HTTP request to create a data job. |
| Step 2 | At the test simulator the contents of the received HTTP request are recorded. |
| Step 3 | <p>The test simulator does the following validation:</p> <ol style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 7.3.4.1. b) The HTTP request is a POST operation. c) The HTTP request message content includes the information as specified in R1AP [4], clause 7.3.4.1. |
| Step 4 | The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 7.3.4.1.1. |

9.1.2.1.4 Expected result

The test is considered passed if Step 3 validation has passed.

9.1.2.2 Cancel data job (positive case)

9.1.2.2.1 Test description and applicability

The purpose of this test case is to test the Cancel data job API as specified in R1AP [4], clause 7.3.4.2. The expected outcome is successful validation of the request from the DUT.

9.1.2.2.2 Test entrance criteria

- 1) The DUT has functionality to initiate the cancel data request procedure as defined in R1GAP [1], clause 5.2.4.
- 2) A data job exists in test simulator and the dataJobId is known to the DUT.

9.1.2.2.3 Test methodology

9.1.2.2.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

9.1.2.2.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to cancel a data job.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
 - a) The URI conforms to the format specified in R1AP [4], clause 7.3.5.3.
 - b) The HTTP request is a DELETE operation.
 - c) The dataJobID in the URI match the data job being deleted.
 - d) The HTTP request message content includes the information as specified in R1AP [4], clause 7.3.4.2.
- Step 4 The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 7.3.4.2.1.

9.1.2.2.4 Expected result

The test is considered passed if Step 3 validation has passed.

9.1.2.3 Query data job status (positive case)

9.1.2.3.1 Test description and applicability

The purpose of this test case is to test the query data job status API as specified in R1AP [4], clause 7.3.4.6. The expected outcome is successful validation of the request from the DUT.

9.1.2.3.2 Test entrance criteria

- 1) The DUT has functionality to initiate the query data request status procedure as defined in R1GAP [1], clause 5.2.4.
- 2) A data job exists in test simulator and the dataJobId is known to the DUT.

9.1.2.3.3 Test methodology

9.1.2.3.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

9.1.2.3.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to query a data job status.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
 - a) The URI conforms to the format specified in R1AP [4], clause 7.3.4.6.1.
 - b) The HTTP request is a GET operation.
 - c) The dataJobID in the URI match the data job being queried.

- d) The HTTP request message content includes the information as specified in R1AP [4], clause 7.3.4.6.1.

Step 4 The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 7.3.4.6.1.

9.1.2.3.4 Expected result

The test is considered passed if Step 3 validation has passed.

9.2 Conformance test cases for SMO/Non-RT RIC framework

9.2.1 General

9.2.1.1 Device under test requirements

The SMO/Non-RT RIC framework that acts as DUT in the test scenarios of this clause can be a function that is under development, or it can be a finalized commercial product. The requirements on the DUT for these tests are that it can handle the DME Data request service, and the purpose of the test scenarios is to validate that it conforms to the API Producer functionality as specified in R1AP [4], clause 7.3.4.

9.2.1.2 Test simulator capabilities

The test simulator has the capabilities as required for a rApp. In addition, it has the following capabilities:

- Recording of received HTTP requests and responses and analysing them regarding conformance to the R1 service definitions.
- Controlled initiation of procedures with configurable URIs and payload formulated and modified.

Validating messages and issuing of verdicts related to the procedures in the test cases and thereby enabling determination of the DUT's conformance to the API Producer functionality as specified in R1AP [4], clause 7.3.4.

9.2.2 Data access API as API Producer test scenario

9.2.2.1 Create data job (positive case)

9.2.2.1.1 Test description and applicability

The purpose of this test case is to test the create data job API as specified in R1AP [4], clause 7.3.4.1. The expected outcome is successful validation of the request from the DUT.

9.2.2.1.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the request data procedure as defined in R1GAP [1], clause 5.2.4.
- 2) The DataJobInfo is known to the DUT.

9.2.2.1.3 Test methodology

9.2.2.1.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP responses from the DUT.

9.2.2.1.3.2 Procedure

- Step 1 The test simulator as an API Consumer initiates a HTTP request to create a data job.

- Step 2 The DUT receives the HTTP request.
- Step 3 The DUT does the following validation:
- a) The HTTP request is a POST request.
 - b) The URI conforms to the format specified in R1AP [4], clause 7.3.5.2.3.1.
 - c) The HTTP request message content includes the information as specified in R1AP [4], clause 7.3.4.1.
- Step 4 The DUT generates the data job and constructs the URI for the created resource.
- Step 5 The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 7.3.4.1.
- Step 6 At the test simulator the contents of the received HTTP response are recorded.
- Step 7 The test simulator does the following validation:
- a) The HTTP response message content includes the information as specified in R1AP [4], clause 7.3.4.1, and the Location header will be present.
 - b) The URI conforms to the format specified in R1AP [4], clause 7.3.5.3.

9.2.2.1.4 Expected result

The test is considered passed if Step 7 validation has passed.

9.2.2.2 Cancel data job (positive case)

9.2.2.2.1 Test description and applicability

The purpose of this test case is to test the Cancel data job API as specified in R1AP [4], clause 7.3.4.2. The expected outcome is successful validation of the request from the DUT.

9.2.2.2.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the cancel data request procedure as defined in R1GAP [1], clause 5.2.4.
- 2) A data job exists in DUT and the dataJobId is known to the test simulator.

9.2.2.2.3 Test methodology

9.2.2.2.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP responses from the DUT.

9.2.2.2.3.2 Procedure

- Step 1 The test simulator as an API Consumer initiates a HTTP request to cancel a data job.
- Step 2 The DUT receives the HTTP request.
- Step 3 The DUT does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 7.3.5.3.3.1.
 - b) The HTTP request is a DELETE operation.
 - c) The dataJobID in the URI match the data job being deleted.
 - d) The HTTP request message content includes the information as specified in R1AP [4], clause 7.3.4.2.

- Step 4 The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 7.3.4.2.
- Step 5 At the test simulator the contents of the received HTTP response are recorded.
- Step 6 The test simulator does the following validation:
- a) The URI confirms to the format specified in R1AP [4] , clause 7.3.5.3.
 - b) The HTTP response message content includes the information as specified in R1AP [4], clause 7.3.4.2.

9.2.2.2.4 Expected result

The test is considered passed if Step 6 validation has passed.

9.2.2.3 Query data job status (positive case)

9.2.2.3.1 Test description and applicability

The purpose of this test case is to test the query data job status API as specified in R1AP [4], clause 7.3.4.6. The expected outcome is successful validation of the request from the DUT.

9.2.2.3.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the query data request status procedure as defined in R1GAP [1], clause 5.2.4.
- 2) A data job exists in the DUT and the dataJobId is known to the test simulator.

9.2.2.3.3 Test methodology

9.2.2.3.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP responses from the DUT.

9.2.2.3.3.2 Procedure

- Step 1 The test simulator as an API Consumer initiates a HTTP request to query a data job status.
- Step 2 The DUT receives the HTTP request.
- Step 3 The DUT does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 7.3.5.4.
 - b) The HTTP request is a GET operation.
 - c) The dataJobID in the URI match the data job status being queried.
 - d) The HTTP request message content includes the information as specified in R1AP [4], clause 7.3.4.6.1.
- Step 4 The DUT constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 7.3.4.6.1.
- Step 5 At the test simulator the contents of the received HTTP response are recorded.
- Step 6 The test simulator does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 7.3.5.4.
 - b) The HTTP response message content includes the information as specified in R1AP [4], clause 7.3.4.6.1.

9.2.2.3.4 Expected result

The test is considered passed if Step 6 validation has passed.

10 DME Data subscription service test cases

10.1 Conformance test cases for rApp

10.1.1 General

10.1.1.1 Device under test requirements

The rApp that acts as DUT in the test scenarios of this clause can be a function that is under development, or it can be a finalized commercial product. The requirements on the DUT for these tests are that it can handle the DME Data subscription service, and the purpose of the test scenarios is to validate that it conforms to the API Consumer functionality as specified in R1AP [4], clause 7.3.4.

10.1.1.2 Test simulator capabilities

The test simulator has the capabilities as required for a SMO/Non-RT RIC framework. In addition, it has the following capabilities:

- Recording of received HTTP requests and responses and analysing them regarding conformance to the R1 service definitions.
- Controlled initiation of procedures with configurable URIs and payload formulated and modified.

Validating messages and issuing of verdicts related to the procedures in the test cases and thereby enabling determination of the DUT's conformance to the API Consumer functionality as specified in R1AP [4], clause 7.3.4.

10.1.2 Data access API as API Consumer test scenario

10.1.2.1 Create data job (positive case)

10.1.2.1.1 Test description and applicability

The purpose of this test case is to test the create data job API as specified in R1AP [4], clause 7.3.4.1. The expected outcome is successful validation of the request from the DUT.

10.1.2.1.2 Test entrance criteria

- 1) The DUT has functionality to initiate the subscribe data procedure as defined in R1GAP [1], clause 5.2.5.
- 2) The DataJobInfo is known to the DUT.

10.1.2.1.3 Test methodology

10.1.2.1.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

10.1.2.1.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to create a data job.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.

- Step 3 The test simulator does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 7.3.4.1.
 - b) The HTTP request is a POST operation.
 - c) The HTTP request message content includes the information as specified in R1AP [4], clause 7.3.4.1.
- Step 4 The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 7.3.4.1.1.

10.1.2.1.4 Expected result

The test is considered passed if Step 3 validation has passed.

10.1.2.2 Cancel data job (positive case)

10.1.2.2.1 Test description and applicability

The purpose of this test case is to test the Cancel data job API as specified in R1AP [4], clause 7.3.4.2. The expected outcome is successful validation of the request from the DUT.

10.1.2.2.2 Test entrance criteria

- 1) The DUT has functionality to initiate the unsubscribe data procedure as defined in R1GAP [1], clause 5.2.5.
- 2) A data job exists in test simulator and the dataJobId is known to the DUT.

10.1.2.2.3 Test methodology

10.1.2.2.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

10.1.2.2.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to cancel a data job.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 7.3.5.3.
 - b) The HTTP request is a DELETE operation.
 - c) The dataJobID in the URI match the data job being deleted.
 - d) The HTTP request message content includes the information as specified in R1AP [4], clause 7.3.4.2.
- Step 4 The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 7.3.4.2.1.

10.1.2.2.4 Expected result

The test is considered passed if Step 3 validation has passed.

10.1.2.3 Query data job (positive case)

10.1.2.3.1 Test description and applicability

The purpose of this test case is to test the query data job API as specified in R1AP [4], clause 7.3.4.5. The expected outcome is successful validation of the request from the DUT.

10.1.2.3.2 Test entrance criteria

- 1) The DUT has functionality to initiate the query data subscription procedure as defined in R1GAP [1], clause 5.2.5.
- 2) A data job exists in test simulator and the dataJobId is known to the DUT.

10.1.2.3.3 Test methodology

10.1.2.3.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

10.1.2.3.3.2 Procedure

- | | |
|--------|---|
| Step 1 | The DUT as an API Consumer initiates a HTTP request to query a data job. |
| Step 2 | At the test simulator the contents of the received HTTP request are recorded. |
| Step 3 | <p>The test simulator does the following validation:</p> <ol style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 7.3.4.5.1. b) The HTTP request is a GET operation. c) The dataJobID in the URI match the data job being queried. d) The HTTP request message content includes the information as specified in R1AP [4], clause 7.3.4.5.1. |
| Step 4 | The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 7.3.4.6.1. |

10.1.2.3.4 Expected result

The test is considered passed if Step 3 validation has passed.

10.1.2.4 Query data job status (positive case)

10.1.2.4.1 Test description and applicability

The purpose of this test case is to test the query data job status API as specified in R1AP [4], clause 7.3.4.6. The expected outcome is successful validation of the request from the DUT.

10.1.2.4.2 Test entrance criteria

- 1) The DUT has functionality to initiate the query data subscription status procedure as defined in R1GAP [1], clause 5.2.5.
- 2) A data job exists in test simulator and the dataJobId is known to the DUT.

10.1.2.4.3 Test methodology

10.1.2.4.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

10.1.2.4.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to query a data job status.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 7.3.4.6.1.
 - b) The HTTP request is a GET operation.
 - c) The dataJobID in the URI match the data job being queried.
 - d) The HTTP request message content includes the information as specified in R1AP [4], clause 7.3.4.6.1.
- Step 4 The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 7.3.4.6.1.

10.1.2.4.4 Expected result

The test is considered passed if Step 3 validation has passed.

10.1.2.5 Query data job identifiers (positive case)

10.1.2.5.1 Test description and applicability

The purpose of this test case is to test the query data job identifiers API as specified in R1AP [4], clause 7.3.4.7. The expected outcome is successful validation of the request from the DUT.

10.1.2.5.2 Test entrance criteria

- 1) The DUT has functionality to initiate the query data subscription status procedure as defined in R1GAP [1], clause 5.2.5.
- 2) A set of data jobs exists in test simulator.

10.1.2.5.3 Test methodology

10.1.2.5.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

10.1.2.5.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to query data job identifiers.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 7.3.4.7.1.
 - b) The HTTP request is a GET operation.

- c) The HTTP request message content includes the information as specified in R1AP [4], clause 7.3.4.7.1.

Step 4 The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 7.3.4.7.1.

10.1.2.5.4 Expected result

The test is considered passed if Step 3 validation has passed.

10.1.2.6 Update data job (positive case)

10.1.2.6.1 Test description and applicability

The purpose of this test case is to test the Update data job API as specified in R1AP [4], clause 7.3.4.5. The expected outcome is successful validation of the request from the DUT.

10.1.2.6.2 Test entrance criteria

- 1) The DUT has functionality to initiate the Update data job procedure as defined in R1GAP [1], clause 5.2.5.
- 2) A data job exists in test simulator and the dataJobId and DataJobInfo are known to the DUT.

10.1.2.6.3 Test methodology

10.1.2.6.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

10.1.2.6.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to update a data job identified by the dataJobId and DataJobInfo.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
 - a) The URI confirms to the format specified in R1AP [4], clause 7.3.4.4.
 - b) The HTTP request is a PUT operation.
 - c) The dataJobId and DataJobInfo in the URI match the data job being updated.
 - d) The HTTP request message content includes the information as specified in R1AP [4], clause 7.3.4.4.1.
- Step 4 The test simulator updates the resource, and a representation of the updated resource shall be returned in the response body the appropriate HTTP response as specified in R1AP [4], clause 7.3.4.4.1.

10.1.2.6.4 Expected result

The test is considered passed if Step 3 validation has passed.

10.2 Conformance test cases for SMO/Non-RT RIC framework

10.2.1 General

10.2.1.1 Device under test requirements

The SMO/Non-RT RIC framework that acts as DUT in the test scenarios of this clause can be a function that is under development, or it can be a finalized commercial product. The requirements on the DUT for these tests are that it can handle the DME Data subscription service, and the purpose of the test scenarios is to validate that it conforms to the API Producer functionality as specified in R1AP [4], clause 7.3.4.

10.2.1.2 Test simulator capabilities

The test simulator has the capabilities as required for a rApp. In addition, it has the following capabilities:

- Recording of received HTTP requests and responses and analysing them regarding conformance to the R1 service definitions.
- Controlled initiation of procedures with configurable URIs and payload formulated and modified.

Validating messages and issuing of verdicts related to the procedures in the test cases and thereby enabling determination of the DUT's conformance to the API Producer functionality as specified in R1AP [4], clause 7.3.4.

10.2.2 Data access API as API Producer test scenario

10.2.2.1 Create data job (positive case)

10.2.2.1.1 Test description and applicability

The purpose of this test case is to test the create data job API as specified in R1AP [4], clause 7.3.4.1. The expected outcome is successful validation of the request from the DUT.

10.2.2.1.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the subscribe data procedure as defined in R1GAP [1], clause 5.2.5.
- 2) The DataJobInfo is known to the DUT.

10.2.2.1.3 Test methodology

10.2.2.1.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP responses from the DUT.

10.2.2.1.3.2 Procedure

- | | |
|--------|---|
| Step 1 | The test simulator as an API Consumer initiates a HTTP request to create a data job. |
| Step 2 | The DUT receives the HTTP request. |
| Step 3 | The DUT does the following validation: <ol style="list-style-type: none"> a) The HTTP request is a POST request. b) The URI conforms to the format specified in R1AP [4], clause 7.3.5.2.3.1. c) The HTTP request message content includes the information as specified in R1AP [4], clause 7.3.4.1. |

- Step 4 The DUT generates the data job and constructs the URI for the created resource.
- Step 5 The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 7.3.4.1.
- Step 6 At the test simulator the contents of the received HTTP response are recorded.
- Step 7 The test simulator does the following validation:
- a) The HTTP response message content includes the information as specified in R1AP [4], clause 7.3.4.1, and the Location header will be present.
 - b) The URI conforms to the format specified in R1AP [4], clause 7.3.5.3.

10.2.2.1.4 Expected result

The test is considered passed if Step 7 validation has passed.

10.2.2.2 Cancel data job (positive case)

10.2.2.2.1 Test description and applicability

The purpose of this test case is to test the Cancel data job API as specified in R1AP [4], clause 7.3.4.2. The expected outcome is successful validation of the request from the DUT.

10.2.2.2.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the unsubscribe data procedure as defined in R1GAP [1], clause 5.2.5.
- 2) A data job exists in DUT and the dataJobId is known to the test simulator.

10.2.2.2.3 Test methodology

10.2.2.2.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP responses from the DUT.

10.2.2.2.3.2 Procedure

- Step 1 The test simulator as an API Consumer initiates a HTTP request to cancel a data job.
- Step 2 The DUT receives the HTTP request.
- Step 3 The DUT does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 7.3.5.3.3.1.
 - b) The HTTP request is a DELETE operation.
 - c) The dataJobID in the URI match the data job being deleted.
 - d) The HTTP request message content includes the information as specified in R1AP [4], clause 7.3.4.2.
- Step 4 The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 7.3.4.2.
- Step 5 At the test simulator the contents of the received HTTP response are recorded.
- Step 6 The test simulator does the following validation:
- a) The URI confirms to the format specified in R1AP [4], clause 7.3.5.3.
 - b) The HTTP response message content includes the information as specified in R1AP [4], clause 7.3.4.2.

10.2.2.2.4 Expected result

The test is considered passed if Step 6 validation has passed.

10.2.2.3 Query data job (positive case)

10.2.2.3.1 Test description and applicability

The purpose of this test case is to test the query data job API as specified in R1AP [4], clause 7.3.4.5. The expected outcome is successful validation of the request from the DUT.

10.2.2.3.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the query data subscription procedure as defined in R1GAP [1], clause 5.2.5.
- 2) A data job exists in the DUT and the dataJobId is known to the test simulator.

10.2.2.3.3 Test methodology

10.2.2.3.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP requests from the DUT.

10.2.2.3.3.2 Procedure

- | | |
|--------|---|
| Step 1 | The test simulator as an API Consumer initiates a HTTP request to query a data job. |
| Step 2 | The DUT receives the HTTP request. |
| Step 3 | The DUT does the following validation: <ol style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 7.3.5.3. b) The HTTP request is a GET operation. c) The dataJobID in the URI match the data job being queried. d) The HTTP request message content includes the information as specified in R1AP [4], clause 7.3.4.5.1. |
| Step 4 | The DUT constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 7.3.4.5.1. |
| Step 5 | At the test simulator the contents of the received HTTP response are recorded. |
| Step 6 | The test simulator does the following validation: <ol style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 7.3.5.3. b) The HTTP response message content includes the information as specified in R1AP [4], clause 7.3.4.5.1. |

10.2.2.3.4 Expected result

The test is considered passed if Step 6 validation has passed.

10.2.2.4 Query data job status (positive case)

10.2.2.4.1 Test description and applicability

The purpose of this test case is to test the query data job status API as specified in R1AP [4], clause 7.3.4.6. The expected outcome is successful validation of the request from the DUT.

10.2.2.4.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the query data subscription status procedure as defined in R1GAP [1], clause 5.2.5.
- 2) A data job exists in the DUT and the dataJobId is known to the test simulator.

10.2.2.4.3 Test methodology

10.2.2.4.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP requests from the DUT.

10.2.2.4.3.2 Procedure

- | | |
|--------|--|
| Step 1 | The test simulator as an API Consumer initiates a HTTP request to query a data job status. |
| Step 2 | The DUT receives the HTTP request. |
| Step 3 | The DUT does the following validation: <ol style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 7.3.5.4. b) The HTTP request is a GET operation. c) The dataJobID in the URI match the data job status being queried. d) The HTTP request message content includes the information as specified in R1AP [4], clause 7.3.4.6.1. |
| Step 4 | The DUT constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 7.3.4.6.1. |
| Step 5 | At the test simulator the contents of the received HTTP response are recorded. |
| Step 6 | The test simulator does the following validation: <ol style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 7.3.5.4. b) The HTTP response message content includes the information as specified in R1AP [4], clause 7.3.4.6.1. |

10.2.2.4.4 Expected result

The test is considered passed if Step 6 validation has passed.

10.2.2.5 Query data job identifiers (positive case)

10.2.2.5.1 Test description and applicability

The purpose of this test case is to test the query data job identifiers API as specified in R1AP [4], clause 7.3.4.7. The expected outcome is successful validation of the request from the DUT.

10.2.2.5.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the query data subscription status procedure as defined in R1GAP [1], clause 5.2.5.
- 2) A set of data jobs exists in the DUT.

10.2.2.5.3 Test methodology

10.2.2.5.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP requests from the DUT.

10.2.2.5.3.2 Procedure

- | | |
|--------|---|
| Step 1 | The test simulator as an API Consumer initiates a HTTP request to query a set of data job identifiers. |
| Step 2 | The DUT receives the HTTP request. |
| Step 3 | <p>The DUT does the following validation:</p> <ol style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 7.3.5.2. b) The HTTP request is a GET operation. c) The HTTP request message content includes the information as specified in R1AP [4], clause 7.3.4.7.1. |
| Step 4 | The DUT constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 7.3.4.7.1. |
| Step 5 | At the test simulator the contents of the received HTTP response are recorded. |
| Step 6 | <p>The test simulator does the following validation:</p> <ol style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 7.3.5.2. b) The HTTP response message content includes the information as specified in R1AP [4], clause 7.3.4.7.1. |

10.2.2.5.4 Expected result

The test is considered passed if Step 6 validation has passed.

10.2.2.6 Update data job (positive case)

10.2.2.6.1 Test description and applicability

The purpose of this test case is to test the Update data job API as specified in R1AP [4], clause 7.3.4.4. The expected outcome is successful validation of the request from the DUT.

10.2.2.6.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the Update data job procedure as defined in R1GAP [1], clause 5.2.5.
- 2) A data job exists in the DUT and the dataJobId and DataJobInfo are known to the test simulator.

10.2.2.6.3 Test methodology

10.2.2.6.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

10.2.2.6.3.2 Procedure

- Step 1 The test simulator as an API Consumer initiates a HTTP request to update a data job identified by the dataJobId and DataJobInfo.
- Step 2 The DUT receives the HTTP request.
- Step 3 The DUT does the following validation:
 - a) The URI conforms to the format specified in R1AP [4], clause 7.3.5.3.
 - b) The HTTP request is a PUT operation.
 - c) The dataJobId and DataJobInfo in the URI match the data job being updated.
 - d) The HTTP request message content includes the information as specified in R1AP [4], clause 7.3.4.4.
- Step 4 The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 7.3.5.3.
- Step 5 At the test simulator the contents of the received HTTP response are recorded.
- Step 6 The test simulator does the following validation:
 - a) The URI conforms to the format specified in R1AP [4], clause 7.3.5.3.
 - b) The HTTP response message content includes the information as specified in R1AP [4], clause 7.3.4.4.

10.2.2.6.4 Expected result

The test is considered passed if Step 6 validation has passed.

10.2.2.7 Notify data availability (positive case)

10.2.2.7.1 Test description and applicability

The purpose of this test case is to test the Notify data availability API as specified in R1AP [4], clause 7.3.4.3. The expected outcome is successful validation of the request from the DUT.

10.2.2.7.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the Notify data availability procedure as defined in R1GAP [1], clause 5.2.5.
- 2) A data subscription exists in the DUT.

10.2.2.7.3 Test methodology

10.2.2.7.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP requests from the DUT.

10.2.2.7.3.2 Procedure

- Step 1 The DUT as an API producer initiates the sending of a HTTP POST request.

- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
- a) The HTTP request is a POST operation.
 - b) The callback URI matches the URI provided in the data subscription.

10.2.2.7.4 Expected result

The test is considered passed if Step 3 validation has passed.

11 A1 related services test cases

11.1 Conformance test cases for rApp

11.1.1 General

11.1.1.1 Device under test requirements

The rApp that acts as DUT in the test scenarios of this clause can be a function that is under development, or it can be a finalized commercial product. The requirements on the DUT for these tests are that it can handle the A1 policy management service, and the purpose of the test scenarios is to validate that it conforms to the API Consumer functionality as specified in R1AP [4], clause 9.1.4.

11.1.1.2 Test simulator capabilities

The test simulator has the capabilities as required for a SMO/Non-RT RIC framework. In addition, it has the following capabilities:

- Recording of received HTTP requests and responses and analysing them regarding conformance to the R1 service definitions.
- Controlled initiation of procedures with configurable URIs and payload formulated and modified.

Validating messages and issuing of verdicts related to the procedures in the test cases and thereby enabling determination of the DUT's conformance to the API Consumer functionality as specified in R1AP [4], clause 9.1.4.

11.1.2 A1 policy management API as API Consumer test scenario

11.1.2.1 Query A1 policy type identifiers (positive case)

11.1.2.1.1 Test description and applicability

The purpose of this test case is to test the query A1 policy type identifiers API as specified in R1AP [4], clause 9.1.4.1. The expected outcome is successful validation of the request from the DUT.

11.1.2.1.2 Test entrance criteria

- 1) The DUT has functionality to initiate the Query A1 policy type identifiers procedure as defined in R1GAP [1], clause 5.3.2.
- 2) A set of A1 policy types exists in test simulator.

11.1.2.1.3 Test methodology

11.1.2.1.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

11.1.2.1.3.2 Procedure

- | | |
|--------|---|
| Step 1 | The DUT as an API Consumer initiates a HTTP request to query A1 policy type identifiers. |
| Step 2 | At the test simulator the contents of the received HTTP request are recorded. |
| Step 3 | The test simulator does the following validation: <ul style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 9.1.4.1. b) The HTTP request is a GET operation. c) The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.1. |
| Step 4 | The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.5.2. |

11.1.2.1.4 Expected result

The test is considered passed if Step 3 validation has passed.

11.1.2.2 Query A1 policy type (positive case)

11.1.2.2.1 Test description and applicability

The purpose of this test case is to test the query A1 policy type API as specified in R1AP [4], clause 9.1.4.2. The expected outcome is successful validation of the request from the DUT.

11.1.2.2.2 Test entrance criteria

- 1) The DUT has functionality to initiate the query A1 policy type procedure as defined in R1GAP [1], clause 5.3.2.
- 2) An A1 policy type exists in test simulator and the policyTypeId is known to the DUT.

11.1.2.2.3 Test methodology

11.1.2.2.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

11.1.2.2.3.2 Procedure

- | | |
|--------|---|
| Step 1 | The DUT as an API Consumer initiates a HTTP request to query an A1 policy type. |
| Step 2 | At the test simulator the contents of the received HTTP request are recorded. |
| Step 3 | The test simulator does the following validation: <ul style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 9.1.4.2. b) The HTTP request is a GET operation. c) The policyTypeID in the URI match the A1 policy type being queried. The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.2. |

Step 4 The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.5.3.

11.1.2.2.4 Expected result

The test is considered passed if Step 3 validation has passed.

11.1.2.3 Query A1 policy identifiers (positive case)

11.1.2.3.1 Test description and applicability

The purpose of this test case is to test the query A1 policy identifiers API as specified in R1AP [4], clause 9.1.4.3. The expected outcome is successful validation of the request from the DUT.

11.1.2.3.2 Test entrance criteria

- 1) The DUT has functionality to initiate the query A1 policy identifiers procedure as defined in R1GAP [1], clause 5.3.2.
- 2) A set of A1 policies exists in test simulator.

11.1.2.3.3 Test methodology

11.1.2.3.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

11.1.2.3.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to query A1 policy identifiers.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
 - a) The URI conforms to the format specified in R1AP [4], clause 9.1.4.3.
 - b) The HTTP request is a GET operation.
 - c) The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.3.
- Step 4 The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.5.4.

11.1.2.3.4 Expected result

The test is considered passed if Step 3 validation has passed.

11.1.2.4 Create A1 policy (positive case)

11.1.2.4.1 Test description and applicability

The purpose of this test case is to test the create A1 policy API as specified in R1AP [4], clause 9.1.4.4. The expected outcome is successful validation of the request from the DUT.

11.1.2.4.2 Test entrance criteria

- 1) The DUT has functionality to initiate the create A1 policy procedure as defined in R1GAP [1], clause 5.3.2.
- 2) The PolicyObjectInformation is known to the DUT.

11.1.2.4.3 Test methodology

11.1.2.4.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

11.1.2.4.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to create an A1 policy.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
 - a) The URI conforms to the format specified in R1AP [4], clause 9.1.4.4.
 - b) The HTTP request is a POST operation.
 - c) The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.4.
- Step 4 The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.5.4.

11.1.2.4.4 Expected result

The test is considered passed if Step 3 validation has passed.

11.1.2.5 Query A1 policy (positive case)

11.1.2.5.1 Test description and applicability

The purpose of this test case is to test the query A1 policy API as specified in R1AP [4], clause 9.1.4.5. The expected outcome is successful validation of the request from the DUT.

11.1.2.5.2 Test entrance criteria

- 1) The DUT has functionality to initiate the query A1 policy procedure as defined in R1GAP [1], clause 5.3.2.
- 2) An A1 policy exists in test simulator and the policyId is known to the DUT.

11.1.2.5.3 Test methodology

11.1.2.5.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

11.1.2.5.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to query an A1 policy.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
 - a) The URI conforms to the format specified in R1AP [4], clause 9.1.4.5.
 - b) The HTTP request is a GET operation.
 - c) The policyId in the URI match the A1 policy being queried. The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.5.

Step 4 The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.5.5.

11.1.2.5.4 Expected result

The test is considered passed if Step 3 validation has passed.

11.1.2.6 Update A1 policy (positive case)

11.1.2.6.1 Test description and applicability

The purpose of this test case is to test the Update A1 policy API as specified in R1AP [4], clause 9.1.4.6. The expected outcome is successful validation of the request from the DUT.

11.1.2.6.2 Test entrance criteria

- 1) The DUT has functionality to initiate the Update A1 policy procedure as defined in R1GAP [1], clause 5.3.2.
- 2) An A1 policy exists in test simulator and the policyId and PolicyObject are known to the DUT.

11.1.2.6.3 Test methodology

11.1.2.6.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

11.1.2.6.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to update an A1 policy identified by the policyId and PolicyObject.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
 - a) The URI confirms to the format specified in R1AP [4], clause 9.1.4.6.
 - b) The HTTP request is a PUT operation.
 - c) The policyId and PolicyObject in the URI match the A1 policy being updated.
 - d) The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.6.
- Step 4 The test simulator updates the resource, and a representation of the updated resource shall be returned in the response body the appropriate HTTP response as specified in R1AP [4], clause 9.1.5.5.

11.1.2.6.4 Expected result

The test is considered passed if Step 3 validation has passed.

11.1.2.7 Delete A1 policy (positive case)

11.1.2.7.1 Test description and applicability

The purpose of this test case is to test the Delete A1 policy API as specified in R1AP [4], clause 9.1.4.6. The expected outcome is successful validation of the request from the DUT.

11.1.2.7.2 Test entrance criteria

- 1) The DUT has functionality to initiate the Delete A1 policy procedure as defined in R1GAP [1], clause 5.3.2.
- 2) An A1 policy exists in test simulator and the policyId is known to the DUT.

11.1.2.7.3 Test methodology

11.1.2.7.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

11.1.2.7.3.2 Procedure

- | | |
|--------|--|
| Step 1 | The DUT as an API Consumer initiates a HTTP request to delete an A1 policy. |
| Step 2 | At the test simulator the contents of the received HTTP request are recorded. |
| Step 3 | <p>The test simulator does the following validation:</p> <ol style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 9.1.4.7. b) The HTTP request is a DELETE operation. c) The policyId in the URI match the A1 policy being deleted. The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.7. |
| Step 4 | The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.5.5. |

11.1.2.7.4 Expected result

The test is considered passed if Step 3 validation has passed.

11.1.2.8 Query A1 policy status (positive case)

11.1.2.8.1 Test description and applicability

The purpose of this test case is to test the query A1 policy status API as specified in R1AP [4], clause 9.1.4.8. The expected outcome is successful validation of the request from the DUT.

11.1.2.8.2 Test entrance criteria

- 1) The DUT has functionality to initiate the query A1 policy status procedure as defined in R1GAP [1], clause 5.3.2.
- 2) An A1 policy exists in test simulator and the policyId is known to the DUT.

11.1.2.8.3 Test methodology

11.1.2.8.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

11.1.2.8.3.2 Procedure

- | | |
|--------|---|
| Step 1 | The DUT as an API Consumer initiates a HTTP request to query an A1 policy status. |
| Step 2 | At the test simulator the contents of the received HTTP request are recorded. |

- Step 3 The test simulator does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 9.1.4.8.
 - b) The HTTP request is a GET operation.
 - c) The policyId in the URI match the A1 policy status being queried.
 - d) The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.8.
- Step 4 The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.5.6.

11.1.2.8.4 Expected result

The test is considered passed if Step 3 validation has passed.

11.1.2.9 Subscribe A1 policy status (positive case)

11.1.2.9.1 Test description and applicability

The purpose of this test case is to test the subscribe A1 policy status API as specified in R1AP [4], clause 9.1.4.9. The expected outcome is successful validation of the request from the DUT.

11.1.2.9.2 Test entrance criteria

- 1) The DUT has functionality to initiate the subscribe A1 policy status procedure as defined in R1GAP [1], clause 5.3.2.
- 2) The PolicyStatusSubscription is known to the DUT.

11.1.2.9.3 Test methodology

11.1.2.9.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

11.1.2.9.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to create an A1 policy status subscription.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 9.1.4.9.
 - b) The HTTP request is a POST operation.
 - c) The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.9.
- Step 4 The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.5.7.

11.1.2.9.4 Expected result

The test is considered passed if Step 3 validation has passed.

11.1.2.10 Update A1 policy status subscription (positive case)

11.1.2.10.1 Test description and applicability

The purpose of this test case is to test the Update A1 policy status subscription API as specified in R1AP [4], clause 9.1.4.10. The expected outcome is successful validation of the request from the DUT.

11.1.2.10.2 Test entrance criteria

- 1) The DUT has functionality to initiate the Update A1 policy status subscription procedure as defined in R1GAP [1], clause 5.3.2.
- 2) An A1 policy status subscription exists in test simulator and the subscriptionId and PolicyStatusSubscription are known to the DUT.

11.1.2.10.3 Test methodology

11.1.2.10.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

11.1.2.10.3.2 Procedure

- | | |
|--------|--|
| Step 1 | The DUT as an API Consumer initiates a HTTP request to update an A1 policy status subscription identified by the subscriptionId and PolicyStatusSubscription. |
| Step 2 | At the test simulator the contents of the received HTTP request are recorded. |
| Step 3 | <p>The test simulator does the following validation:</p> <ol style="list-style-type: none"> a) The URI confirms to the format specified in R1AP [4], clause 9.1.4.10. b) The HTTP request is a PUT operation. c) The subscriptionId and PolicyStatusSubscription in the URI match the status subscription being updated. d) The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.10. |
| Step 4 | The test simulator updates the resource, and a representation of the updated resource shall be returned in the response body the appropriate HTTP response as specified in R1AP [4], clause 9.1.5.8. |

11.1.2.10.4 Expected result

The test is considered passed if Step 3 validation has passed.

11.1.2.11 Query A1 policy status subscription (positive case)

11.1.2.11.1 Test description and applicability

The purpose of this test case is to test the query A1 policy status subscription API as specified in R1AP [4], clause 9.1.4.11. The expected outcome is successful validation of the request from the DUT.

11.1.2.11.2 Test entrance criteria

- 1) The DUT has functionality to initiate the query A1 policy status subscription procedure as defined in R1GAP [1], clause 5.3.2.
- 2) An A1 policy status subscription exists in test simulator and the subscriptionId is known to the DUT.

11.1.2.11.3 Test methodology

11.1.2.11.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

11.1.2.11.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to query an A1 policy status subscription.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 9.1.4.11.
 - b) The HTTP request is a GET operation.
 - c) The subscriptionId in the URI match the A1 policy status subscription being queried.
 - d) The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.11.
- Step 4 The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.5.8.

11.1.2.11.4 Expected result

The test is considered passed if Step 3 validation has passed.

11.1.2.12 Unsubscribe A1 policy status (positive case)

11.1.2.12.1 Test description and applicability

The purpose of this test case is to test the Unsubscribe A1 policy status API as specified in R1AP [4], clause 9.1.4.12. The expected outcome is successful validation of the request from the DUT.

11.1.2.12.2 Test entrance criteria

- 1) The DUT has functionality to initiate the Unsubscribe A1 policy procedure as defined in R1GAP [1], clause 5.3.2.
- 2) An A1 policy status subscription exists in test simulator and the subscriptionId is known to the DUT.

11.1.2.12.3 Test methodology

11.1.2.12.3.1 Initial conditions

The test simulator as API Producer is ready and available to receive HTTP requests from the DUT.

11.1.2.12.3.2 Procedure

- Step 1 The DUT as an API Consumer initiates a HTTP request to delete an A1 policy status subscription.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 9.1.4.12.
 - b) The HTTP request is a DELETE operation.

- c) The subscriptionId in the URI match the A1 policy status subscription being deleted. The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.12.

Step 4 The test simulator constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.5.8.

11.1.2.12.4 Expected result

The test is considered passed if Step 3 validation has passed.

11.2 Conformance test cases for SMO/Non-RT RIC framework

11.2.1 General

11.2.1.1 Device under test requirements

The SMO/Non-RT RIC framework that acts as DUT in the test scenarios of this clause can be a function that is under development, or it can be a finalized commercial product. The requirements on the DUT for these tests are that it can handle the DME Data subscription service, and the purpose of the test scenarios is to validate that it conforms to the API Producer functionality as specified in R1AP [4], clause 9.1.4.

11.2.1.2 Test simulator capabilities

The test simulator has the capabilities as required for a rApp. In addition, it has the following capabilities:

- Recording of received HTTP requests and responses and analysing them regarding conformance to the R1 service definitions.
- Controlled initiation of procedures with configurable URIs and payload formulated and modified.

Validating messages and issuing of verdicts related to the procedures in the test cases and thereby enabling determination of the DUT's conformance to the API Producer functionality as specified in R1AP [4], clause 9.1.4.

11.2.2 A1 policy management API as API Producer test scenario

11.2.2.1 Query A1 policy type identifiers (positive case)

11.2.2.1.1 Test description and applicability

The purpose of this test case is to test the query A1 policy type identifiers API as specified in R1AP [4], clause 9.1.4.1. The expected outcome is successful validation of the request from the DUT.

11.2.2.1.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the Query A1 policy type identifiers procedure as defined in R1GAP [1], clause 5.3.2.
- 2) A set of A1 policy types exists in the DUT.

11.2.2.1.3 Test methodology

11.2.2.1.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP requests from the DUT.

11.2.2.1.3.2 Procedure

- Step 1 The test simulator as an API Consumer initiates a HTTP request to query a set of A1 policy type identifiers.
- Step 2 The DUT receives the HTTP request.
- Step 3 The DUT does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 9.1.4.1.
 - b) The HTTP request is a GET operation.
 - c) The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.1.
- Step 4 The DUT constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.4.1.
- Step 5 At the test simulator the contents of the received HTTP response are recorded.
- Step 6 The test simulator does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 9.1.5.2.
 - b) The HTTP response message content includes the information as specified in R1AP [4], clause 9.1.4.1.

11.2.2.1.4 Expected result

The test is considered passed if Step 6 validation has passed.

11.2.2.2 Query A1 policy type (positive case)

11.2.2.2.1 Test description and applicability

The purpose of this test case is to test the query A1 policy type API as specified in R1AP [4], clause 9.1.4.2. The expected outcome is successful validation of the request from the DUT.

11.2.2.2.2 Test entrance criteria

- 1) The DUT has functionality to initiate the query A1 policy type procedure as defined in R1GAP [1], clause 5.3.2.
- 2) An A1 policy type exists in DUT and the policyTypeId is known to the test simulator.

11.2.2.2.3 Test methodology

11.2.2.2.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP requests from the DUT.

11.2.2.2.3.2 Procedure

- Step 1 The test simulator as an API Consumer initiates a HTTP request to query an A1 policy type.
- Step 2 The DUT receives the HTTP request.
- Step 3 The DUT does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 9.1.5.3.
 - b) The HTTP request is a GET operation.

- c) The policyTypeId in the URI match the A1 policy type being queried. The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.2.

Step 4 The DUT constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.4.2.

Step 5 At the test simulator the contents of the received HTTP response are recorded.

Step 6 The test simulator does the following validation:

- a) The URI conforms to the format specified in R1AP [4], clause 9.1.5.3.
- b) The HTTP response message content includes the information as specified in R1AP [4], clause 9.1.4.2.

11.2.2.2.4 Expected result

The test is considered passed if Step 6 validation has passed.

11.2.2.3 Query A1 policy identifiers (positive case)

11.2.2.3.1 Test description and applicability

The purpose of this test case is to test the query A1 policy identifiers API as specified in R1AP [4], clause 9.1.4.3. The expected outcome is successful validation of the request from the DUT.

11.2.2.3.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the Query A1 policy identifiers procedure as defined in R1GAP [1], clause 5.3.2.
- 2) A set of A1 policies exists in the DUT.

11.2.2.3.3 Test methodology

11.2.2.3.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP requests from the DUT.

11.2.2.3.3.2 Procedure

Step 1 The test simulator as an API Consumer initiates a HTTP request to query a set of A1 policy identifiers.

Step 2 The DUT receives the HTTP request.

Step 3 The DUT does the following validation:

- a) The URI conforms to the format specified in R1AP [4], clause 9.1.5.4.
- b) The HTTP request is a GET operation.
- c) The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.3.

Step 4 The DUT constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.4.3.

Step 5 At the test simulator the contents of the received HTTP response are recorded.

Step 6 The test simulator does the following validation:

- a) The URI conforms to the format specified in R1AP [4], clause 9.1.5.4.

- b) The HTTP response message content includes the information as specified in R1AP [4], clause 9.1.4.3.

11.2.2.3.4 Expected result

The test is considered passed if Step 6 validation has passed.

11.2.2.4 Create A1 policy (positive case)

11.2.2.4.1 Test description and applicability

The purpose of this test case is to test the create A1 policy API as specified in R1AP [4], clause 9.1.4.4. The expected outcome is successful validation of the request from the DUT.

11.2.2.4.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the create A1 policy procedure as defined in R1GAP [1], clause 5.3.2.
- 2) The PolicyObjectInformation is known to the test simulator.

11.2.2.4.3 Test methodology

11.2.2.4.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP responses from the DUT.

11.2.2.4.3.2 Procedure

- | | |
|--------|--|
| Step 1 | The test simulator as an API Consumer initiates a HTTP request to create an A1 policy. |
| Step 2 | The DUT receives the HTTP request. |
| Step 3 | The DUT does the following validation: <ul style="list-style-type: none"> a) The HTTP request is a POST request. b) The URI conforms to the format specified in R1AP [4], clause 9.1.5.4. c) The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.4. |
| Step 4 | The DUT generates the A1 policy and constructs the URI for the created resource. |
| Step 5 | The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.4.4. |
| Step 6 | At the test simulator the contents of the received HTTP response are recorded. |
| Step 7 | The test simulator does the following validation: <ul style="list-style-type: none"> a) The HTTP response message content includes the information as specified in R1AP [4], clause 9.1.4.4, and the Location header will be present. b) The URI conforms to the format specified in R1AP [4], clause 9.1.5.4. |

11.2.2.4.4 Expected result

The test is considered passed if Step 7 validation has passed.

11.2.2.5 Query A1 policy (positive case)

11.2.2.5.1 Test description and applicability

The purpose of this test case is to test the query A1 policy API as specified in R1AP [4], clause 9.1.4.5. The expected outcome is successful validation of the request from the DUT.

11.2.2.5.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the query A1 policy procedure as defined in R1GAP [1], clause 5.3.2.
- 2) An A1 policy exists in DUT and the policyId is known to the test simulator.

11.2.2.5.3 Test methodology

11.2.2.5.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP requests from the DUT.

11.2.2.5.3.2 Procedure

- | | |
|--------|---|
| Step 1 | The test simulator as an API Consumer initiates a HTTP request to query an A1 policy. |
| Step 2 | The DUT receives the HTTP request. |
| Step 3 | The DUT does the following validation: <ol style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 9.1.5.5. b) The HTTP request is a GET operation. c) The policyId in the URI match the A1 policy being queried. d) The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.5. |
| Step 4 | The DUT constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.4.5. |
| Step 5 | At the test simulator the contents of the received HTTP response are recorded. |
| Step 6 | The test simulator does the following validation: <ol style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 9.1.5.5. b) The HTTP response message content includes the information as specified in R1AP [4], clause 9.1.4.5. |

11.2.2.5.4 Expected result

The test is considered passed if Step 6 validation has passed.

11.2.2.6 Update A1 policy (positive case)

11.2.2.6.1 Test description and applicability

The purpose of this test case is to test the Update A1 policy API as specified in R1AP [4], clause 9.1.4.6. The expected outcome is successful validation of the request from the DUT.

11.2.2.6.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the Update A1 policy procedure as defined in R1GAP [1], clause 5.3.2.
- 2) An A1 policy exists in the DUT and the policyId and PolicyObject are known to the test simulator.

11.2.2.6.3 Test methodology

11.2.2.6.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP requests from the DUT.

11.2.2.6.3.2 Procedure

- | | |
|--------|---|
| Step 1 | The test simulator as an API Consumer initiates a HTTP request to update an A1 policy identified by the policyId and PolicyObject. |
| Step 2 | The DUT receives the HTTP request. |
| Step 3 | <p>The DUT does the following validation:</p> <ol style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 9.1.5.5. b) The HTTP request is a PUT operation. c) The policyId and PolicyObject in the URI match the A1 policy being updated. d) The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.6. |
| Step 4 | The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.4.6. |
| Step 5 | At the test simulator the contents of the received HTTP response are recorded. |
| Step 6 | <p>The test simulator does the following validation:</p> <ol style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 9.1.5.5. b) The HTTP response message content includes the information as specified in R1AP [4], clause 9.1.4.6. |

11.2.2.6.4 Expected result

The test is considered passed if Step 6 validation has passed.

11.2.2.7 Delete A1 policy (positive case)

11.2.2.7.1 Test description and applicability

The purpose of this test case is to test the Delete A1 policy API as specified in R1AP [4], clause 9.1.4.6. The expected outcome is successful validation of the request from the DUT.

11.2.2.7.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the Delete A1 policy procedure as defined in R1GAP [1], clause 5.3.2.
- 2) An A1 policy exists in DUT and the policyId is known to the test simulator.

11.2.2.7.3 Test methodology

11.2.2.7.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP responses from the DUT.

11.2.2.7.3.2 Procedure

- Step 1 The test simulator as an API Consumer initiates a HTTP request to Delete an A1 policy.
- Step 2 The DUT receives the HTTP request.
- Step 3 The DUT does the following validation:
 - a) The URI conforms to the format specified in R1AP [4], clause 9.1.5.5.
 - b) The HTTP request is a DELETE operation.
 - c) The policyID in the URI match the A1 policy being deleted.
 - d) The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.7.
- Step 4 The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.4.7.
- Step 5 At the test simulator the contents of the received HTTP response are recorded.
- Step 6 The test simulator does the following validation:
 - a) The URI confirms to the format specified in R1AP [4], clause 9.1.5.5.
 - b) The HTTP response message content includes the information as specified in R1AP [4], clause 9.1.4.7.

11.2.2.7.4 Expected result

The test is considered passed if Step 6 validation has passed.

11.2.2.8 Query A1 policy status (positive case)

11.2.2.8.1 Test description and applicability

The purpose of this test case is to test the query A1 policy status API as specified in R1AP [4], clause 9.1.4.8. The expected outcome is successful validation of the request from the DUT.

11.2.2.8.2 Test entrance criteria

- 1) The DUT has functionality to initiate the query A1 policy procedure as defined in R1GAP [1], clause 5.3.2.
- 2) An A1 policy exists in DUT and the policyId is known to the test simulator.

11.2.2.8.3 Test methodology

11.2.2.8.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP requests from the DUT.

11.2.2.8.3.2 Procedure

- Step 1 The test simulator as an API Consumer initiates a HTTP request to query an A1 policy status.
- Step 2 The DUT receives the HTTP request.

- Step 3 The DUT does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 9.1.5.36.
 - b) The HTTP request is a GET operation.
 - c) The policyId in the URI match the A1 policy status being queried.
 - d) The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.8.
- Step 4 The DUT constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.4.8.
- Step 5 At the test simulator the contents of the received HTTP response are recorded.
- Step 6 The test simulator does the following validation:
- a) The URI conforms to the format specified in R1AP [4], clause 9.1.5.6.
 - b) The HTTP response message content includes the information as specified in R1AP [4], clause 9.1.4.8.

11.2.2.8.4 Expected result

The test is considered passed if Step 6 validation has passed.

11.2.2.9 Subscribe A1 policy status (positive case)

11.2.2.9.1 Test description and applicability

The purpose of this test case is to test the subscribe A1 policy status API as specified in R1AP [4], clause 9.1.4.9. The expected outcome is successful validation of the request from the DUT.

11.2.2.9.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the subscribe A1 policy status procedure as defined in R1GAP [1], clause 5.3.2.
- 2) The PolicyStatusSubscription is known to the test simulator.

11.2.2.9.3 Test methodology

11.2.2.9.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP requests from the DUT.

11.2.2.9.3.2 Procedure

- Step 1 The test simulator as an API Consumer initiates a HTTP request to create an A1 policy status subscription.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The DUT does the following validation:
- a) The HTTP request is a POST request.
 - b) The URI conforms to the format specified in R1AP [4], clause 9.1.5.7.
 - c) The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.9.

- Step 4 The DUT generates the A1 policy status subscription and constructs the URI for the created resource.
- Step 5 The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.4.9.
- Step 6 At the test simulator the contents of the received HTTP response are recorded.
- Step 7 The test simulator does the following validation:
- The HTTP response message content includes the information as specified in R1AP [4], clause 9.1.4.9, and the Location header will be present.
 - The URI conforms to the format specified in R1AP [4], clause 9.1.5.7.

11.2.2.9.4 Expected result

The test is considered passed if Step 7 validation has passed.

11.2.2.10 Update A1 policy status subscription (positive case)

11.2.2.10.1 Test description and applicability

The purpose of this test case is to test the Update A1 policy status subscription API as specified in R1AP [4], clause 9.1.4.10. The expected outcome is successful validation of the request from the DUT.

11.2.2.10.2 Test entrance criteria

- The test simulator has functionality to initiate the Update A1 policy status subscription procedure as defined in R1GAP [1], clause 5.3.2.
- An A1 policy status subscription exists in the DUT and the subscriptionId and PolicyStatusSubscription are known to the test simulator.

11.2.2.10.3 Test methodology

11.2.2.10.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP requests from the DUT.

11.2.2.10.3.2 Procedure

- Step 1 The test simulator as an API Consumer initiates a HTTP request to update an A1 policy status subscription identified by the subscriptionId and PolicyStatusSubscription.
- Step 2 The DUT receives the HTTP request.
- Step 3 The DUT does the following validation:
- The URI conforms to the format specified in R1AP [4], clause 9.1.5.8.
 - The HTTP request is a PUT operation.
 - The subscriptionId and PolicyStatusSubscription in the URI match the A1 policy status subscription being updated.
 - The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.10.
- Step 4 The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.4.10.
- Step 5 At the test simulator the contents of the received HTTP response are recorded.

Step 6 The test simulator does the following validation:

- a) The URI conforms to the format specified in R1AP [4], clause 9.1.5.8.
- b) The HTTP response message content includes the information as specified in R1AP [4], clause 9.1.4.10.

11.2.2.10.4 Expected result

The test is considered passed if Step 6 validation has passed.

11.2.2.11 Query A1 policy status subscription (positive case)

11.2.2.11.1 Test description and applicability

The purpose of this test case is to test the query A1 policy status subscription API as specified in R1AP [4], clause 9.1.4.11. The expected outcome is successful validation of the request from the DUT.

11.2.2.11.2 Test entrance criteria

- 1) The DUT has functionality to initiate the query A1 policy status subscription procedure as defined in R1GAP [1], clause 5.3.2.
- 2) An A1 policy status subscription exists in test simulator and the subscriptionId is known to the DUT.

11.2.2.11.3 Test methodology

11.2.2.11.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP requests from the DUT.

11.2.2.11.3.2 Procedure

- Step 1 The test simulator as an API Consumer initiates a HTTP request to query an A1 policy status subscription.
- Step 2 The DUT receives the HTTP request.
- Step 3 The DUT does the following validation:
 - a) The URI conforms to the format specified in R1AP [4], clause 9.1.5.8.
 - b) The HTTP request is a GET operation.
 - c) The subscriptionId in the URI match the A1 policy status subscription being queried.
 - d) The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.11.
- Step 4 The DUT constructs the URI for the created resource and sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.4.5.
- Step 5 At the test simulator the contents of the received HTTP response are recorded.
- Step 6 The test simulator does the following validation:
 - a) The URI conforms to the format specified in R1AP [4], clause 9.1.5.8.
 - b) The HTTP response message content includes the information as specified in R1AP [4], clause 9.1.4.11.

11.2.2.11.4 Expected result

The test is considered passed if Step 6 validation has passed.

11.2.2.12 Unsubscribe A1 policy status (positive case)

11.2.2.12.1 Test description and applicability

The purpose of this test case is to test the Unsubscribe A1 policy status API as specified in R1AP [4], clause 9.1.4.12. The expected outcome is successful validation of the request from the DUT.

11.2.2.12.2 Test entrance criteria

- 1) The DUT has functionality to initiate the Unsubscribe A1 policy procedure as defined in R1GAP [1], clause 5.3.2.
- 2) An A1 policy subscription exists in DUT and the subscriptionId is known to the test simulator.

11.2.2.12.3 Test methodology

11.2.2.12.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP responses from the DUT.

11.2.2.12.3.2 Procedure

- | | |
|--------|---|
| Step 1 | The test simulator as an API Consumer initiates a HTTP request to delete an A1 policy status subscription. |
| Step 2 | The DUT receives the HTTP request. |
| Step 3 | The DUT does the following validation: <ol style="list-style-type: none"> a) The URI conforms to the format specified in R1AP [4], clause 9.1.5.8. b) The HTTP request is a DELETE operation. c) The subscriptionId in the URI match the A1 policy status subscription being deleted. d) The HTTP request message content includes the information as specified in R1AP [4], clause 9.1.4.12. |
| Step 4 | The DUT sends the appropriate HTTP response as specified in R1AP [4], clause 9.1.4.12. |
| Step 5 | At the test simulator the contents of the received HTTP response are recorded. |
| Step 6 | The test simulator does the following validation: <ol style="list-style-type: none"> a) The URI confirms to the format specified in R1AP [4], clause 9.1.5.8. b) The HTTP response message content includes the information as specified in R1AP [4], clause 9.1.4.12. |

11.2.2.12.4 Expected result

The test is considered passed if Step 6 validation has passed.

11.2.2.13 Notify A1 policy status changes (positive case)

11.2.2.13.1 Test description and applicability

The purpose of this test case is to test the Notify A1 policy status changes API as specified in R1AP [4], clause 9.1.4.13. The expected outcome is successful validation of the request from the DUT.

11.2.2.13.2 Test entrance criteria

- 1) The test simulator has functionality to initiate the Notify A1 policy status changes procedure as defined in R1GAP [1], clause 5.3.2.
- 2) An A1 policy status exists in the DUT.

11.2.2.13.3 Test methodology

11.2.2.13.3.1 Initial conditions

The test simulator as API Consumer is ready and available to receive HTTP requests from the DUT.

11.2.2.13.3.2 Procedure

- Step 1 The DUT as an API producer initiates the sending of a HTTP POST request.
- Step 2 At the test simulator the contents of the received HTTP request are recorded.
- Step 3 The test simulator does the following validation:
 - a) The HTTP request is a POST operation.
 - b) The callback URI matches the URI provided in the A1 policy status subscription.

11.2.2.13.4 Expected result

The test is considered passed if Step 3 validation has passed.

Annex A (informative): Change history

Date	Version	Information about changes
2025.03.13	03.00	Published with Conformance test of rApp and Non-RT RIC as DUT for DME, A1 policy management and SME
2024.11.21	02.00	Published with Conformance test of Non-RT RIC as DUT for Service registration test case
2024.07.11	01.00	Published with conformance testing of rApps and SMO/Non-RT RIC Framework and with one Service registration test case

History

Version	Date	Status
V3.0.0	February 2026	Publication