



Publicly Available Specification (PAS); A1 interface: Test Specification (O-RAN.WG2.A1TS-R003-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.A1TS-R003.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-001141

Keywords
conformance, interface, PAS, test, testing

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:
<https://www.etsi.org/standards-search>

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

Users of the present document should be aware that the document may be subject to revision or change of status.

Information on the current status of this and other ETSI documents is available at
<https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx>

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

If you find a security vulnerability in the present document, please report it through our
Coordinated Vulnerability Disclosure Program:
<https://www.etsi.org/standards/coordinated-vulnerability-disclosure>

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 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.

Contents

Intellectual Property Rights	11
Foreword.....	11
Modal verbs terminology.....	11
Introduction	12
1 Scope	13
2 References	13
2.1 Normative references	13
2.2 Informative references.....	13
3 Definition of terms, symbols and abbreviations.....	14
3.1 Terms.....	14
3.2 Symbols	14
3.3 Abbreviations	14
4 Test methodology	14
4.1 General	14
4.2 Conformance testing Non-Real Time RIC	14
4.2.1 General.....	14
4.2.2 Test configuration	14
4.2.2.0 Overview.....	14
4.2.2.1 Device under test (Non-Real Time RIC).....	15
4.2.2.2 Test simulator.....	15
4.3 Conformance testing Near-Real Time RIC	15
4.3.1 General.....	15
4.3.2 Test configuration	16
4.3.2.0 Overview.....	16
4.3.2.1 Device under test (Near-Real Time RIC).....	16
4.3.2.2 Test simulator.....	16
4.4 Interoperability testing between Non-Real Time RIC and Near-Real Time RIC.....	17
4.4.1 General.....	17
4.4.2 Test configuration	17
4.4.2.0 Overview.....	17
4.4.2.1 System under test (Non-Real Time RIC and Near-Real Time RIC)	17
4.4.2.2 Test tools and simulators (SMO framework, E2 nodes, UEs, etc.).....	18
4.4.2.2.1 SMO O1 functionality (Simulator or real).....	18
4.4.2.2.2 UEs and RAN (simulator or real)	18
4.4.2.2.3 4G/5G Core Network	18
4.4.2.2.4 Tap interface and protocol analyzer	18
5 Test cases for Non-Real Time RIC	18
5.1 General	18
5.1.1 Device under test requirements.....	18
5.1.2 Test simulator capabilities	18
5.2 Conformance test cases for A1-P Consumer	19
5.2.1 Query policy type test scenarios	19
5.2.1.1 Query all policy type identifiers (positive case).....	19
5.2.1.1.1 Test description and applicability	19
5.2.1.1.2 Test entrance criteria	19
5.2.1.1.3 Test methodology	19
5.2.1.2 Query single policy type (positive case)	19
5.2.1.2.1 Test description and applicability	19
5.2.1.2.2 Test entrance criteria	20
5.2.1.2.3 Test methodology	20
5.2.2 Create policy test scenarios.....	20
5.2.2.1 Create single policy (positive case).....	20
5.2.2.1.1 Test description and applicability	20

5.2.2.1.2	Test entrance criteria	20
5.2.2.1.3	Test methodology	21
5.2.3	Query policy test scenarios	21
5.2.3.1	Query all policy identifiers (positive case).....	21
5.2.3.1.1	Test description and applicability	21
5.2.3.1.2	Test entrance criteria	21
5.2.3.1.3	Test methodology	22
5.2.3.2	Query single policy (positive case).....	22
5.2.3.2.1	Test description and applicability	22
5.2.3.2.2	Test entrance criteria	22
5.2.3.2.3	Test methodology	22
5.2.3.3	Query policy status (positive case).....	23
5.2.3.3.1	Test description and applicability	23
5.2.3.3.2	Test entrance criteria	23
5.2.3.3.3	Test methodology	23
5.2.4	Update policy test scenarios.....	23
5.2.4.1	Update single policy (positive case).....	23
5.2.4.1.1	Test description and applicability	23
5.2.4.1.2	Test entrance criteria	24
5.2.4.1.3	Test methodology	24
5.2.5	Delete policy test scenarios.....	24
5.2.5.1	Delete single policy (positive case).....	24
5.2.5.1.1	Test description and applicability	24
5.2.5.1.2	Test entrance criteria	24
5.2.5.1.3	Test methodology	25
5.2.6	Feedback policy test scenarios.....	25
5.2.6.1	Policy feedback (positive case).....	25
5.2.6.1.1	Test description and applicability	25
5.2.6.1.2	Test entrance criteria	25
5.2.6.1.3	Test methodology	25
5.2.6.2	Policy feedback - schema validation failure (negative case).....	26
5.2.6.2.1	Test description and applicability	26
5.2.6.2.2	Test entrance criteria	26
5.2.6.2.3	Test methodology	26
5.2.6.3	Policy feedback -callback URI not supported (negative case).....	27
5.2.6.2.1	Test description and applicability	27
5.2.6.2.2	Test entrance criteria	27
5.2.6.2.3	Test methodology	27
5.3	Conformance test cases for A1-EI Producer	27
5.3.1	Query EI types test scenarios	27
5.3.1.1	Query EI type identifiers (positive case).....	27
5.3.1.1.1	Test description and applicability	27
5.3.1.1.2	Test entrance criteria	28
5.3.1.1.3	Test methodology	28
5.3.1.2	Query EI type (positive case).....	28
5.3.1.2.1	Test description and applicability	28
5.3.1.2.2	Test entrance criteria	28
5.3.1.2.3	Test methodology	29
5.3.1.3	Query EI type (negative case) - eiTypeId not supported.....	29
5.3.1.3.1	Test description and applicability	29
5.3.1.3.2	Test entrance criteria	29
5.3.1.3.3	Test methodology	29
5.3.2	Create EI job test scenarios.....	30
5.3.2.1	Create single EI job (positive case).....	30
5.3.2.1.1	Test description and applicability	30
5.3.2.1.2	Test entrance criteria	30
5.3.2.1.3	Test methodology	30
5.3.2.2	Create EI job (negative case) - eiTypeId not supported	31
5.3.2.2.1	Test description and applicability	31
5.3.2.2.2	Test entrance criteria	31
5.3.2.2.3	Test methodology	31
5.3.2.3	Create EI job (negative case) - schema validation failure	31

5.3.2.3.1	Test description and applicability	31
5.3.2.3.2	Test entrance criteria	31
5.3.2.3.3	Test methodology	31
5.3.3	Query EI jobs test scenarios.....	32
5.3.3.1	Query EI job identifiers for a single EI type (positive case)	32
5.3.3.1.1	Test description and applicability	32
5.3.3.1.2	Test entrance criteria	32
5.3.3.1.3	Test methodology	32
5.3.3.2	Query EI job identifiers for a single EI type (negative case) - eiTypeId not supported	33
5.3.3.2.1	Test description and applicability	33
5.3.3.2.2	Test entrance criteria	33
5.3.3.2.3	Test methodology	33
5.3.3.3	Query EI job identifiers for all EI types (positive case)	34
5.3.3.3.1	Test description and applicability	34
5.3.3.3.2	Test entrance criteria	34
5.3.3.3.3	Test methodology	34
5.3.3.4	Query EI job (positive case).....	34
5.3.3.4.1	Test description and applicability	34
5.3.3.4.2	Test entrance criteria	35
5.3.3.4.3	Test methodology	35
5.3.3.5	Query EI job (negative case) - EI job does not exist	35
5.3.3.5.1	Test description and applicability	35
5.3.3.5.2	Test entrance criteria	35
5.3.3.5.3	Test methodology	35
5.3.4	Update EI job test scenarios.....	36
5.3.4.1	Update EI job (positive case)	36
5.3.4.1.1	Test description and applicability	36
5.3.4.1.2	Test entrance criteria	36
5.3.4.1.3	Test methodology	36
5.3.4.2	Update EI job (negative case) - schema validation failure	37
5.3.4.2.1	Test description and applicability	37
5.3.4.2.2	Test entrance criteria	37
5.3.4.2.3	Test methodology	37
5.3.5	Delete EI job test scenarios.....	37
5.3.5.1	Delete EI job (positive case)	37
5.3.5.1.1	Test description and applicability	37
5.3.5.1.2	Test entrance criteria	37
5.3.5.1.3	Test methodology	38
5.3.5.2	Delete EI job (negative case) - EI job does not exist	38
5.3.5.2.1	Test description and applicability	38
5.3.5.2.2	Test entrance criteria	38
5.3.5.2.3	Test methodology	38
5.3.6	Status of EI jobs test scenarios.....	39
5.3.6.1	Query EI job status (positive case).....	39
5.3.6.1.1	Test description and applicability	39
5.3.6.1.2	Test entrance criteria	39
5.3.6.1.3	Test methodology	39
5.3.6.2	Query EI job status (negative case) EI job does not exist	39
5.3.6.2.1	Test description and applicability	39
5.3.6.2.2	Test entrance criteria	39
5.3.6.2.3	Test methodology	40
5.3.6.3	Notify EI job status (positive case)	40
5.3.6.1.1	Test description and applicability	40
5.3.6.1.2	Test entrance criteria	40
5.3.6.1.3	Test methodology	40
5.3.7	Deliver EI job result test scenarios	41
5.3.7.1	Deliver EI job result (positive case).....	41
5.3.7.1.1	Test description and applicability	41
5.3.7.1.2	Test entrance criteria	41
5.3.7.1.3	Test methodology	41
5.3.7.2	Deliver EI job result - no callback (negative case).....	42
5.3.7.2.1	Test description and applicability	42

5.3.7.2.2	Test entrance criteria	42
5.3.7.2.3	Test methodology	42
6	Test cases for Near-Real Time RIC	42
6.1	General	42
6.1.1	Device under test requirements.....	42
6.1.2	Test simulator capabilities	43
6.2	Conformance Test Cases for A1-P Producer.....	43
6.2.1	Query policy type test scenarios	43
6.2.1.1	Query all policy type identifiers (positive case).....	43
6.2.1.1.1	Test description and applicability	43
6.2.1.1.2	Test entrance criteria	43
6.2.1.1.3	Test methodology	43
6.2.1.2	Query single policy type (positive case)	44
6.2.1.2.1	Test description and applicability	44
6.2.1.2.2	Test entrance criteria	44
6.2.1.2.3	Test methodology	44
6.2.1.3	Query single policy type (negative case) - policyTypeId not supported	45
6.2.1.3.1	Test description and applicability	45
6.2.1.3.2	Test entrance criteria	45
6.2.1.3.3	Test methodology	45
6.2.2	Create policy test scenarios.....	45
6.2.2.1	Create single policy (positive case).....	45
6.2.2.1.1	Test description and applicability	45
6.2.2.1.2	Test entrance criteria	45
6.2.2.1.3	Test methodology	46
6.2.2.2	Create policy (negative case) - schema validation failure	46
6.2.2.2.1	Test description and applicability	46
6.2.2.2.2	Test entrance criteria	46
6.2.2.2.3	Test methodology	46
6.2.2.3	Create policy (negative case) - policyTypeId not supported	47
6.2.2.3.1	Test description and applicability	47
6.2.2.3.2	Test entrance criteria	47
6.2.2.3.3	Test methodology	47
6.2.3	Query policy test scenarios	47
6.2.3.1	Query all policy identifiers (positive case).....	47
6.2.3.1.1	Test description and applicability	47
6.2.3.1.2	Test entrance criteria	48
6.2.3.1.3	Test methodology	48
6.2.3.2	Query all policy identifiers (negative case) - policyTypeId not supported	48
6.2.3.2.1	Test description and applicability	48
6.2.3.2.2	Test entrance criteria	49
6.2.3.2.3	Test methodology	49
6.2.3.3	Query single policy (positive case)	49
6.2.3.3.1	Test description and applicability	49
6.2.3.3.2	Test entrance criteria	49
6.2.3.3.3	Test methodology	49
6.2.3.4	Query single policy (negative case) - policy does not exist	50
6.2.3.4.1	Test description and applicability	50
6.2.3.4.2	Test entrance criteria	50
6.2.3.4.3	Test methodology	50
6.2.3.5	Query policy status (positive case).....	50
6.2.3.5.1	Test description and applicability	50
6.2.3.5.2	Test entrance criteria	51
6.2.3.5.3	Test methodology	51
6.2.3.6	Query policy status (negative case) - policy does not exist.....	51
6.2.3.6.1	Test description and applicability	51
6.2.3.6.2	Test entrance criteria	51
6.2.3.6.3	Test methodology	51
6.2.4	Update policy test scenarios.....	52
6.2.4.1	Update single policy (positive case).....	52
6.2.4.1.1	Test description and applicability	52

6.2.4.1.2	Test entrance criteria	52
6.2.4.1.3	Test methodology	52
6.2.4.2	Update single policy (negative case) - schema validation failure	53
6.2.4.2.1	Test description and applicability	53
6.2.4.2.2	Test entrance criteria	53
6.2.4.2.3	Test methodology	53
6.2.5	Delete policy test scenarios.....	53
6.2.5.1	Delete single policy (positive case).....	53
6.2.5.1.1	Test description and applicability	53
6.2.5.1.2	Test entrance criteria	53
6.2.5.1.3	Test methodology	54
6.2.5.2	Delete single policy (negative case) - policy does not exist.....	54
6.2.5.2.1	Test description and applicability	54
6.2.5.2.2	Test entrance criteria	54
6.2.5.2.3	Test methodology	54
6.2.6	Feedback policy test scenarios.....	55
6.2.6.1	Feedback policy (positive case)	55
6.2.6.1.1	Test description and applicability	55
6.2.6.1.2	Test entrance criteria	55
6.2.6.1.3	Test methodology	55
6.3	Conformance test cases for A1-EI Consumer	56
6.3.1	Query EI types test scenarios	56
6.3.1.1	Query EI type identifiers (positive case)	56
6.3.1.1.1	Test description and applicability	56
6.3.1.1.2	Test entrance criteria	56
6.3.1.1.3	Test methodology	56
6.3.1.2	Query EI type (positive case)	56
6.3.1.2.1	Test description and applicability	56
6.3.1.2.2	Test entrance criteria	56
6.3.1.2.3	Test methodology	57
6.3.2	Create EI job test scenarios.....	57
6.3.2.1	Create single EI job (positive case).....	57
6.3.2.1.1	Test description and applicability	57
6.3.2.1.2	Test entrance criteria	57
6.3.2.1.3	Test methodology	57
6.3.3	Query EI jobs test scenarios.....	58
6.3.3.1	Query EI job identifiers for a single EI type (positive case)	58
6.3.3.1.1	Test description and applicability	58
6.3.3.1.2	Test entrance criteria	58
6.3.3.1.3	Test methodology	58
6.3.3.2	Query EI job identifiers for all EI types (positive case)	59
6.3.3.2.1	Test description and applicability	59
6.3.3.2.2	Test entrance criteria	59
6.3.3.2.3	Test methodology	59
6.3.3.3	Query EI job (positive case)	60
6.3.3.3.1	Test description and applicability	60
6.3.3.3.2	Test entrance criteria	60
6.3.3.3.3	Test methodology	60
6.3.4	Update EI job test scenarios.....	60
6.3.4.1	Update EI job (positive case)	60
6.3.4.1.1	Test description and applicability	60
6.3.4.1.2	Test entrance criteria	61
6.3.4.1.3	Test methodology	61
6.3.5	Delete EI job test scenarios.....	61
6.3.5.1	Delete EI job (positive case)	61
6.3.5.1.1	Test description and applicability	61
6.3.5.1.2	Test entrance criteria	61
6.3.5.1.3	Test methodology	62
6.3.6	Status of EI jobs test scenarios.....	62
6.3.6.1	Query EI job status (positive case).....	62
6.3.6.1.1	Test description and applicability	62
6.3.6.1.2	Test entrance criteria	62

6.3.6.1.3	Test methodology	62
6.3.6.2	Notify EI job status (positive case)	63
6.3.6.2.1	Test description and applicability	63
6.3.6.2.2	Test entrance criteria	63
6.3.6.2.3	Test methodology	63
6.3.6.3	Notify EI job status - schema validation failure (negative case)	64
6.3.6.3.1	Test description and applicability	64
6.3.6.3.2	Test entrance criteria	64
6.3.6.3.3	Test methodology	64
6.3.6.4	Notify EI job status - callback URI not supported (negative case)	64
6.3.6.4.1	Test description and applicability	64
6.3.6.4.2	Test entrance criteria	65
6.3.6.4.3	Test methodology	65
6.3.7	Deliver EI job result test scenarios	65
6.3.7.1	Deliver EI job result (positive case)	65
6.3.7.1.1	Test description and applicability	65
6.3.7.1.2	Test entrance criteria	65
6.3.7.1.3	Test methodology	65
6.3.7.2	Deliver EI job result - schema validation failure (negative case)	66
6.3.7.2.1	Test description and applicability	66
6.3.7.2.2	Test entrance criteria	66
6.3.7.2.3	Test methodology	66
6.3.7.3	Deliver EI job result - callback URI not supported (negative case)	67
6.3.7.2.1	Test description and applicability	67
6.3.7.2.2	Test entrance criteria	67
6.3.7.2.3	Test methodology	67
7	Test cases for interoperability between Non-Real Time RIC and Near-Real Time RIC	67
7.1	General	67
7.1.1	System under test requirements	67
7.1.2	Test tools and simulators capabilities	68
7.2	Interoperability test cases for A1-P	68
7.2.1	Query policy type test scenarios	68
7.2.1.1	Query all policy type identifiers (positive case)	68
7.2.1.1.1	Test description and applicability	68
7.2.1.1.2	Test entrance criteria	68
7.2.1.1.3	Test methodology	68
7.2.1.2	Query single policy type (positive)	69
7.2.1.2.1	Test description and applicability	69
7.2.1.2.2	Test entrance criteria	69
7.2.1.2.3	Test methodology	70
7.2.2	Create policy type test scenario	70
7.2.2.1	Create single policy (positive)	70
7.2.2.1.1	Test description and applicability	70
7.2.2.1.2	Test entrance criteria	70
7.2.2.1.3	Test methodology	71
7.2.3	Query policy test scenarios	71
7.2.3.1	Query all policy identifiers	71
7.2.3.1.1	Test description and applicability	71
7.2.3.1.2	Test entrance criteria	71
7.2.3.1.3	Test methodology	72
7.2.3.2	Query single policy	73
7.2.3.2.1	Test description and applicability	73
7.2.3.2.2	Test entrance criteria	73
7.2.3.2.3	Test methodology	73
7.2.3.3	Query policy status (positive case)	73
7.2.3.3.1	Test description and applicability	73
7.2.3.3.2	Test entrance criteria	74
7.2.3.3.3	Test methodology	74
7.2.4	Update policy test scenarios	74
7.2.4.1	Update single policy (positive case)	74
7.2.4.1.1	Test description and applicability	74

7.2.4.1.2	Test entrance criteria	75
7.2.4.1.3	Test methodology	75
7.2.5	Delete policy test scenarios.....	75
7.2.5.1	Delete single policy (positive case).....	75
7.2.5.1.1	Test description and applicability	75
7.2.5.1.2	Test entrance criteria	76
7.2.5.1.3	Test methodology	76
7.2.6	Feedback policy test scenarios.....	76
7.2.6.1	Feedback policy (positive case)	76
7.2.6.1.1	Test description and applicability	76
7.2.6.1.2	Test entrance criteria	76
7.2.6.1.3	Test methodology	77
7.3	Interoperability test cases for A1-EI.....	78
7.3.1	Query EI types test scenarios	78
7.3.1.1	Query EI type identifiers (positive case)	78
7.3.1.1.1	Test description and applicability	78
7.3.1.1.2	Test entrance criteria	78
7.3.1.1.3	Test methodology	78
7.3.1.2	Query EI type (positive case)	79
7.3.1.2.1	Test description and applicability	79
7.3.1.2.2	Test entrance criteria	79
7.3.1.2.3	Test methodology	79
7.3.2	Create EI job test scenario	80
7.3.2.1	Create single EI job (positive case).....	80
7.3.2.1.1	Test description and applicability	80
7.3.2.1.2	Test entrance criteria	80
7.3.2.1.3	Test methodology	80
7.3.3	Query EI jobs test scenarios.....	81
7.3.3.1	Query EI job identifiers for a single EI type (positive case)	81
7.3.3.1.1	Test description and applicability	81
7.3.3.1.2	Test entrance criteria	81
7.3.3.1.3	Test methodology	81
7.3.3.2	Query EI job identifiers for all EI types (positive case)	82
7.3.3.2.1	Test description and applicability	82
7.3.3.2.2	Test entrance criteria	82
7.3.3.2.3	Test methodology	82
7.3.3.3	Query EI job (positive case)	83
7.3.3.3.1	Test description and applicability	83
7.3.3.3.2	Test entrance criteria	83
7.3.3.3.3	Test methodology	83
7.3.4	Update EI job test scenarios.....	84
7.3.4.1	Update EI job (positive case)	84
7.3.4.1.1	Test description and applicability	84
7.3.4.1.2	Test entrance criteria	84
7.3.4.1.3	Test methodology	84
7.3.5	Delete EI job test scenarios.....	85
7.3.5.1	Delete EI job (positive case)	85
7.3.5.1.1	Test description and applicability	85
7.3.5.1.2	Test entrance criteria	85
7.3.5.1.3	Test methodology	85
7.3.6	Status of EI jobs test scenarios.....	86
7.3.6.1	Query EI job status (positive case)	86
7.3.6.1.1	Test description and applicability	86
7.3.6.1.2	Test entrance criteria	86
7.3.6.1.3	Test methodology	86
7.3.6.2	Notify EI job status (positive case)	87
7.3.6.2.1	Test description and applicability	87
7.3.6.2.2	Test entrance criteria	87
7.3.6.2.3	Test methodology	87
7.3.7	Deliver EI job result test scenarios	88
7.3.7.1	Deliver EI job result (positive case)	88
7.3.7.1.1	Test description and applicability	88

7.3.7.1.2	Test entrance criteria	88
7.3.7.1.3	Test methodology	88
Annex A (informative):	Change history	90
History		91

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 Web server (<https://ipr.etsi.org/>).

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™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM®** and the GSM logo are trademarks registered and owned by the GSM Association.

Foreword

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

The present document is part of a TS-family covering the A1 interface as identified below:

- "A1 interface: General Aspects and Principles";
- "A1 interface: Use Cases and Requirements";
- "A1 interface: Transport Protocol";
- "A1 interface: Application Protocol";
- "A1 interface: Type Definitions"; and
- "A1 interface: Test Specification".

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 test cases for the A1 interface. Test cases and methodology for A1-P Consumer, A1-P Producer, A1-EI Consumer, and A1-EI Producer are separately specified. The present document will also address A1 related test cases for Interoperability between Non-Real Time RIC and Near-Real Time RICs.

The present document contains normative requirements on A1 test cases for Conformance testing and Interoperability testing. For each specified test case, these requirements cover the entrance criteria, procedure and expected result. The test cases are defined based on the procedures specified for the A1 Application Protocol for which the test cases are used to validate conformance and interoperability. The present specification does not require that all test cases are supported by a DUT and a subset of the test cases may not be applicable in a specific testing. It is the responsibility of the tester to identify which test cases that are relevant based on an implementation statement related to a DUT.

The test cases require test equipment with specific capabilities needed to perform testing. The clauses on the test configurations in the present specification describes some of the main capabilities that test equipment will provide.

The test cases require certain functionality in the DUTs to perform testing. The present specification does not require any specific implementation of such functionality.

NOTE 1: When standardized APIs are available for rApps and xApps, it is foreseen that such apps can be part of the test equipment and used for testing of production grade DUTs.

Testing of the A1-P and A1-EI procedures requires that at least one policy type and one EI type is supported by both sides of the A1 interface during testing. The policy type(s) or EI type(s) to be used in a test case is agreed between the testing entities as described for the test entrance criteria of the test case.

NOTE 2: Testing of expected behaviour related to policy types and EI types is not in scope of the present specification, only the procedural aspects that requires presence of some policy type or EI type.

1 Scope

The present document specifies test cases for conformance testing and interoperability testing of the Non-RT RIC and the Near-RT RIC over the A1 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 at <https://docbox.etsi.org/Reference>.

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] Void.
- [2] Void.
- [3] [ETSI TS 103 986 \(V2.1.0\)](#): "Publicly Available Specification (PAS); A1 interface: Transport Protocol (O-RAN.WG2.A1TP-R003-v02.01)".
- NOTE: This reference is not mentioned in the current version of the present document but will be in future revisions.
- [4] [ETSI TS 103 987 \(V4.0.0\)](#): "Publicly Available Specification (PAS); A1 interface: Application Protocol (O-RAN.WG2.A1AP-R003-v04.00)".
- [5] Void.

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 are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

- [i.1] [ETSI TS 103 983 \(V3.1.0\)](#): "Publicly Available Specification (PAS); A1 interface: General Aspects and Principles (O-RAN.WG2.A1GAP-R003-v03.01)".
- [i.2] [ETSI TS 103 988 \(V5.0.0\)](#): "Publicly Available Specification (PAS); A1 interface: Type Definitions (O-RAN.WG2.A1TD-R003-v05.00)".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in A1GAP [i.1], A1AP [4] and A1TD [i.2] apply.

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in A1GAP [i.1], A1AP [4], A1TD [i.2], and the following apply:

DUT Device Under Test

4 Test methodology

4.1 General

This clause describes the methodology for conformance and interoperability testing of Non-Real Time RIC and Near-Real Time RIC over A1 interface.

For conformance tests, simulators are used for testing A1 procedures. These simulators will have capability of generating HTTP requests and responses for GET, PUT, POST and DELETE operations. 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 Non-Real Time RIC and Near-Real Time RIC, these devices are brought to operation by connecting to appropriate real or simulated devices.

4.2 Conformance testing Non-Real Time RIC

4.2.1 General

Non-Real Time RIC is the device under test, clause 5 of the present specification specifies conformance tests for A1-P Producer and A1-EI Consumer functionality as specified in document A1AP [4].

4.2.2 Test configuration

4.2.2.0 Overview

The test configuration for A1 conformance testing of Non-RT RIC is illustrated in figure 4.2.2.0-1.

Non-Real Time RIC (Device Under Test)

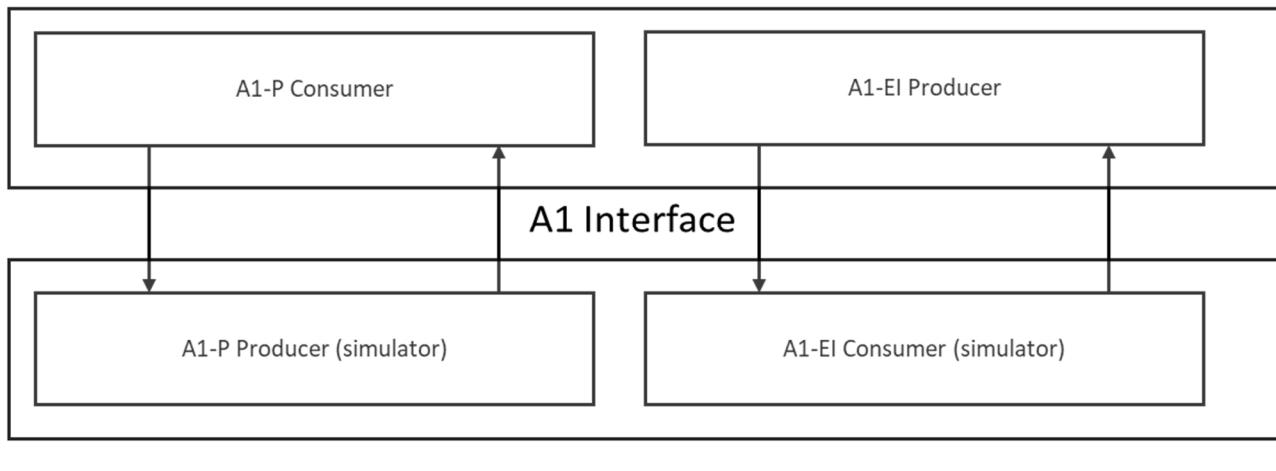


Figure 4.2.2.0-1: Illustration of Non-Real Time RIC A1 conformance test setup

4.2.2.1 Device under test (Non-Real Time RIC)

For enabling conformance testing, the Non-RT RIC has implemented A1-P Consumer and A1-EI Producer functionality and the procedures specified in A1AP [4] that are required to perform testing of the applicable test cases. It also supports one agreed policy type and one agreed EI type.

4.2.2.2 Test simulator

The test simulator has A1-P Producer and A1-EI Consumer that both have HTTP Client and HTTP Server capabilities and have flexibility to generate, receive and validate HTTP messages for all the A1 procedures. The test simulator logs all message content during the testing.

The A1-P Producer in the test simulator has all the capabilities needed to operate the A1-P test cases, including:

- Enable and disable policy types.
- Changing parameters in A1-P procedure response messages.
- Responding with success and failure messages.
- Ability to deliver policy status notifications by sending HTTP POST messages with JSON body created based on configured schemas.
- Validation of message contents based on configured schemas (headers, return codes, JSON body, etc.).

The A1-EI Consumer in the test simulator has all the capabilities needed to operate the A1-EI test cases, including:

- Send query messages for EI types.
- Create and delete EI jobs for available EI types.
- Request and receive EI job status notifications and EI job results.

4.3 Conformance testing Near-Real Time RIC

4.3.1 General

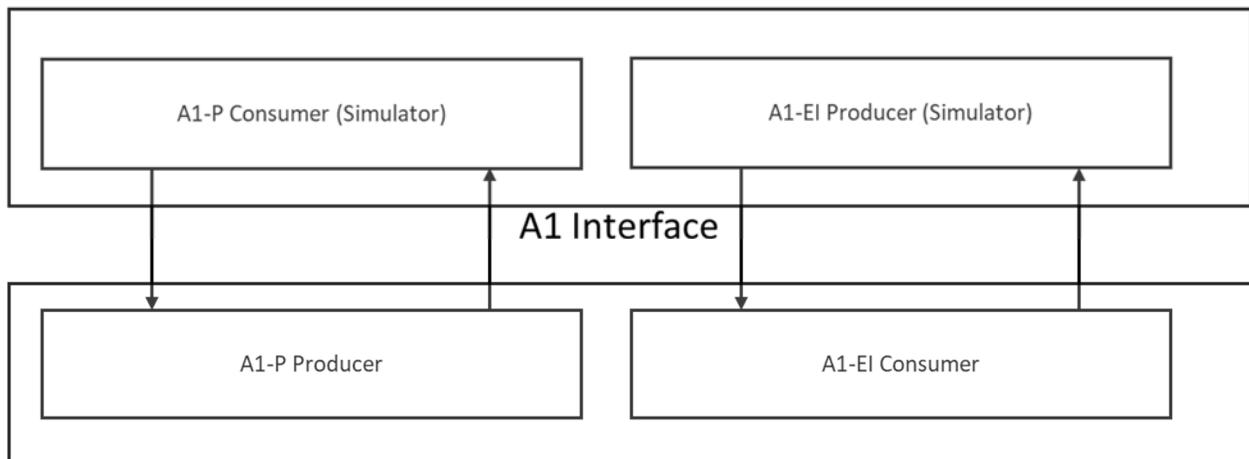
Near-Real Time RIC is the device under test, clause 6 of the present specification specifies conformance tests for A1-P Consumer and A1-EI Producer functionality as specified in document A1AP [4].

4.3.2 Test configuration

4.3.2.0 Overview

The test configuration for A1 conformance testing of Near-RT RIC is illustrated in figure 4.3.2.0-1.

Test Simulator



Device Under Test Near-Real Time RIC

Figure 4.3.2.0-1: Illustration of Near-Real Time RIC A1 conformance test setup

4.3.2.1 Device under test (Near-Real Time RIC)

For enabling conformance testing, the Near-RT RIC has implemented A1-P Producer and A1-EI Consumer functionality and the procedures specified in A1AP [4] that are required to perform testing of the applicable test cases. It also supports one agreed policy type and one agreed EI type.

4.3.2.2 Test simulator

The test simulator has A1-P Consumer and A1-EI Producer that both have HTTP Client and HTTP Server capabilities and have flexibility to generate, receive and validate HTTP messages for all the A1 procedures. The test simulator logs all message content during the testing.

The A1-P Consumer in the test simulator has all the capabilities needed to operate the A1-P test cases, including:

- Send query messages for policy types.
- Create and delete policies for available policy types.
- Request and receive policy feedback.

The A1-EI Producer in the test simulator has all the capabilities needed to operate the A1-EI test cases, including:

- Enable and disable EI job types.
- Changing parameters in A1-EI procedure response messages.
- Responding with success and failure messages.
- Ability to deliver EI job status notifications and EI job results by sending HTTP POST messages with JSON body created based on configured schemas.
- Validation of message contents based on configured schemas (headers, return codes, JSON body, etc.).

4.4 Interoperability testing between Non-Real Time RIC and Near-Real Time RIC

4.4.1 General

Both Non-Real Time RIC and Near-RT RIC are the devices under test or constitutes a system under test, they may be from different vendors or from the same vendor. Clause 7 of the present specification specifies interoperability test cases for A1-P and A1-EI functionality as specified in document A1AP [4].

Test methodology is conceived to be non-intrusive to ensure that production ready devices and software can be used in the test setup.

A protocol analyser connected to A1 interface use a tap to collects and analyses packets passively for test case validation.

In addition to the devices under test, some additional functionalities may be needed to bring the devices under test to operational state and to facilitate test case execution. These are optional depending on DUT implementation and can be provided by real devices or by simulators.

4.4.2 Test configuration

4.4.2.0 Overview

The test configuration for A1 interoperability testing between Non-RT RIC and Near-RT RIC is illustrated in figure 4.4.2.0-1.

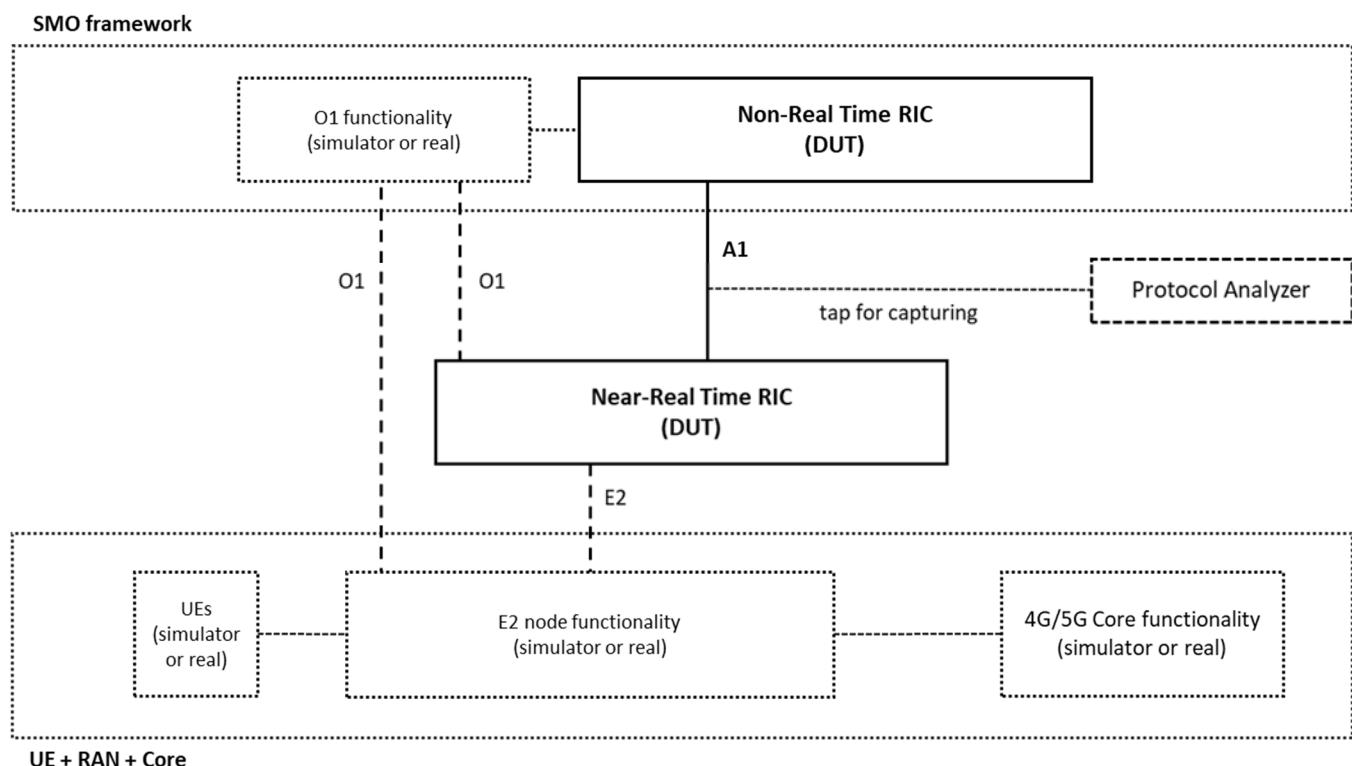


Figure 4.4.2.0-1: Illustration of Non-Real Time RIC and Near Real Time RIC interoperability test setup

4.4.2.1 System under test (Non-Real Time RIC and Near-Real Time RIC)

Non-Real Time RIC and Near-Real Time RIC will have a minimal set of matching functionalities to facilitate testing of A1-P and A1-EI procedures. As an entry criterion for interoperability testing, Non-Real Time RIC and Near Real Time RIC need to have at least one matching policy type and EI type.

4.4.2.2 Test tools and simulators (SMO framework, E2 nodes, UEs, etc.)

4.4.2.2.1 SMO O1 functionality (Simulator or real)

SMO O1 functionality may be needed if Non-Real Time RIC or Near-Real Time RIC relies on O1 functionality for provisioning. This interface is also needed if the Non-Real Time RIC collects performance metrics via O1 interface.

4.4.2.2.2 UEs and RAN (simulator or real)

For triggering A1-P and A1-E1 procedures across A1 interface, Near-Real Time RIC may need one or more E2 Node(s). These can be real or simulated nodes with supported E2 Service Model functionality as needed by Near-Real Time RIC involved in the tests.

A simulated or real UE call activity may be needed to invoke the appropriate Near-RT RIC functionality and/or O1 metrics.

4.4.2.2.3 4G/5G Core Network

Depending on the Non-RT RIC, Near-RT RIC and/or E2 Nodes used, a simulated or real 4G/5G Core Network may be needed for UE activity.

4.4.2.2.4 Tap interface and protocol analyzer

A tap interface and protocol analyzer can be used on A1 interface to passively capture and analyse packets for test case validation.

5 Test cases for Non-Real Time RIC

5.1 General

5.1.1 Device under test requirements

The Non-Real Time RIC 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 procedures defined for a Non-Real Time RIC and the purpose of the test scenarios is to validate that it conforms to the A1 service definitions in A1AP [4].

In addition to the basic conformance requirements, the following is required for a Non-Real Time RIC when it acts as DUT:

- It shall be possible for the DUT to initiate the A1-P procedures in a controlled manner for each test case.
- It shall be possible for the DUT to support A1 policy types and EI types to use in the test cases after agreement with the test simulator. It shall be possible to formulate A1 policies based on agreed policy types, and it shall be possible to validate PolicyStatusObject's and EiJobObject's based on the schemas of the agreed policy types and EI types.

NOTE: The present document does not require usage of any specific policy type or EI type since specific behaviour is not validated, only the application protocol aspects and procedures are validated for conformance testing of the A1 services.

5.1.2 Test simulator capabilities

The test simulator has the same basic capabilities as required for a Near-Real Time RIC. In addition, it has the following capabilities:

- Recording of received HTTP requests and responses and analysing them regarding conformance to the A1 service definitions.

- Controlled initiation of procedures with configurable URIs and payload formulated and modified based on the schemas of the agreed policy types and EI types.
- 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 A1 service definitions in A1AP [4].

5.2 Conformance test cases for A1-P Consumer

5.2.1 Query policy type test scenarios

5.2.1.1 Query all policy type identifiers (positive case)

5.2.1.1.1 Test description and applicability

The purpose of this test case is to test the query policy types functionality of A1-P Consumer as specified in A1AP [4], clause 5.2.3.2. The expected outcome is successful validation of the Query all policy type identifiers request.

This test case is conditionally mandatory if the DUT claims to support A1-P service and the Query all policy type identifiers procedure.

5.2.1.1.2 Test entrance criteria

- 1) The DUT has functionality to initiate A1-P Query all policy type identifiers procedure.
- 2) A set of policy types (at least two) are supported in the test simulator.

5.2.1.1.3 Test methodology

5.2.1.1.3.1 Initial conditions

- 1) The test simulator has A1-P Producer service ready and available to receive HTTP requests from the DUT.

5.2.1.1.3.2 Procedure

Step 1. Initiate appropriate action in the DUT to initiate Query all policy type identifiers 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 URI conforms to the format specified in A1AP [4], clause 6.2.3.
- b) The HTTP request is a GET operation.
- c) The HTTP request message body is empty.

Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.2.3.2.1.

5.2.1.1.3.3 Expected result

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

5.2.1.2 Query single policy type (positive case)

5.2.1.2.1 Test description and applicability

The purpose of this test case is to test the query policy type functionality of A1-P Consumer as specified in A1AP [4], clause 5.2.3.3. The expected outcome is successful validation of the Query single policy type request.

This test case is conditionally mandatory if the DUT claims to support A1-P service and the Query single policy type procedure.

5.2.1.2.2 Test entrance criteria

- 1) The DUT has the functionality to initiate A1-P Query single policy type procedure.
- 2) A known set of policy types are supported in the test simulator.
- 3) Policy type identifiers used for this test are known in advance and used in DUT to formulate the Query single policy type request, and in test Simulator for selecting the appropriate schemas for the policy type object.

5.2.1.2.3 Test methodology

5.2.1.2.3.1 Initial conditions

- 1) The test Simulator has A1-P Producer service ready and available to receive HTTP requests from the DUT.

5.2.1.2.3.2 Procedure

Step 1. Initiate appropriate action in the DUT to initiate Query single policy type request with a known policyTypeid.

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 A1AP [4], clause 6.2.3.
- b) The HTTP request is a GET operation.
- c) The policyTypeid in the URI match the policy type being queried.
- d) The message body is empty.

Step 4. The test Simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.2.3.3.1.

5.2.1.2.3.3 Expected result

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

5.2.2 Create policy test scenarios

5.2.2.1 Create single policy (positive case)

5.2.2.1.1 Test description and applicability

This purpose of this test case is to test the create policy functionality of A1-P Consumer as specified in A1AP [4], clause 5.2.4.3. The expected outcome is successful validation of the Create single policy request.

This test case is mandatory if the DUT claims to support A1-P service.

5.2.2.1.2 Test entrance criteria

- 1) The DUT has the functionality to initiate A1-P Create single policy procedure.
- 2) The DUT and the test simulator have agreed on a policy type to use for this test.
- 3) The policyTypeid and the JSON schemas of the policy type used for this test are available and used in DUT to formulate the create policy request, and in test simulator to validate the request.

5.2.2.1.3 Test methodology

5.2.2.1.3.1 Initial conditions

- 1) The test simulator has A1-P Producer service ready and available to receive HTTP requests from the DUT.
- 2) No policy exists in the test simulator for the agreed policy type with the same policyId that will be used by the DUT.

NOTE: The DUT is required to generate a unique policyId during the Create single policy procedure (see A1AP [4], clauses 5.2.2.2 and 5.2.3.2.2).

5.2.2.1.3.2 Procedure

Step 1. Initiate appropriate action in the DUT to initiate Create single policy request for the agreed 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:

- a) The URI conforms to the format specified in A1AP [4], clause 6.2.3.
- b) The HTTP request is a PUT operation.
- c) The policyTypeId is supported.
- d) A policy with the policyId does not exist.
- e) The HTTP request message body contains the PolicyObject of the policy to be created and the PolicyObject conforms to the schema.

Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.3.2.2.2.

5.2.2.1.3.3 Expected result

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

NOTE: Presence or validation of optional query parameter for notificationDestination is not used to determine validation on this test.

5.2.3 Query policy test scenarios

5.2.3.1 Query all policy identifiers (positive case)

5.2.3.1.1 Test description and applicability

The purpose of this test case is to test the query policies functionality of A1-P Consumer as specified in A1AP [4], clause 5.2.4.2. The expected outcome is successful validation of the Query all policy identifiers request.

This test case is conditionally mandatory if the DUT claims to support A1-P service and the Query all policy identifiers procedure.

5.2.3.1.2 Test entrance criteria

- 1) The DUT has functionality to initiate A1-P Query all policy identifiers procedure.
- 2) The DUT and the test simulator have agreed on a policy type to use for this test.
- 3) A set of policies (at least two) for the agreed policy type exist in the test simulator.

5.2.3.1.3 Test methodology

5.2.3.1.3.1 Initial conditions

- 1) The test simulator has A1-P Producer service ready and available to receive HTTP requests from the DUT.

5.2.3.1.3.2 Procedure

Step 1. Initiate appropriate action in the DUT to initiate Query all policy identifiers 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 URI conforms to the format specified in A1AP [4], clause 6.2.3.
- b) The HTTP request is a GET operation.
- c) The message body is empty.

Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.2.4.2.1.

5.2.3.1.3.3 Expected result

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

5.2.3.2 Query single policy (positive case)

5.2.3.2.1 Test description and applicability

The purpose of this test case is to test the query policy functionality of A1-P Consumer as specified in A1AP [4], clause 5.2.4.5. The expected outcome is successful validation of the Query single policy request.

This test case is conditionally mandatory if the DUT claims to support A1-P service and the Query single policy procedure.

5.2.3.2.2 Test entrance criteria

- 1) The DUT has functionality to initiate A1-P Query single policy procedure.
- 2) A policy exists in test simulator and the policyTypeId and the policyId is known to DUT.

5.2.3.2.3 Test methodology

5.2.3.2.3.1 Initial conditions

- 1) The test simulator has A1-P Producer service ready and available to receive HTTP requests from the DUT.

5.2.3.2.3.2 Procedure

Step 1. Initiate appropriate action in the DUT to initiate Query single policy request for the existing policy identified by policyTypeId and policyId.

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 A1AP [4], clause 6.2.3.
- b) The HTTP request is a GET operation.
- c) The policyTypeId and policyId in the URI match the policy being queried.

- d) The message body is empty.

Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.2.4.5.1.

5.2.3.2.3.3 Expected result

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

5.2.3.3 Query policy status (positive case)

5.2.3.3.1 Test description and applicability

The purpose of this test case is to test the query policy status functionality of A1-P Consumer as specified in A1AP [4], clause 5.2.4.7. The expected outcome is successful validation of the Query policy status request.

This test case is conditionally mandatory if the DUT claims to support A1-P service and the Query policy status procedure.

5.2.3.3.2 Test entrance criteria

- 1) The DUT has functionality to initiate A1-P Query policy status procedure.
- 2) A policy exists in test simulator and the policyTypeId and the policyId is known to DUT.

5.2.3.3.3 Test methodology

5.2.3.3.3.1 Initial conditions

- 1) The test simulator has A1-P Producer service ready and available to receive HTTP requests from the DUT.

5.2.3.3.3.2 Procedure

Step 1. Initiate appropriate action in the DUT to initiate Query policy status request for the existing policy identified by policyTypeId and policyId.

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 A1AP [4], clause 6.2.3.
- b) The HTTP request is a GET operation.
- c) The policyTypeId and policyId in the URI match the policy being queried.
- d) The message body is empty.

Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.2.4.7.1.

5.2.3.3.3.3 Expected result

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

5.2.4 Update policy test scenarios

5.2.4.1 Update single policy (positive case)

5.2.4.1.1 Test description and applicability

The purpose of this test case is to test the update policy functionality of A1-P Consumer as specified in A1AP [4], clause 5.2.4.4. The expected outcome is successful validation of Update single policy request.

This test case is conditionally mandatory if the DUT claims to support A1-P service and the Update single policy procedure.

5.2.4.1.2 Test entrance criteria

- 1) The DUT has functionality to initiate A1-P Update single policy procedure.
- 2) A policy exists in test simulator and the policyTypeId and the policyId is known to DUT.
- 3) The JSON schemas of the policy type used for this test are available and used in DUT to formulate the Update single policy request, and in test simulator to validate the request.

5.2.4.1.3 Test methodology

5.2.4.1.3.1 Initial conditions

- 1) The test simulator has A1-P Producer service ready and available to receive HTTP request from the DUT.

5.2.4.1.3.2 Procedure

Step 1. Initiate appropriate action in the DUT to initiate Update single policy request for the existing policy identified by policyTypeId and policyId.

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 A1AP [4], clause 6.2.3.
- b) The HTTP request is a PUT operation.
- c) The policyTypeId and policyId in the URI match the policy being updated.
- d) The HTTP request message body contains the PolicyObject of the policy to be updated and the PolicyObject conforms to the schema of the policy type.

Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.2.4.4.1.

5.2.4.1.3.3 Expected result

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

NOTE: Presence or validation of optional query parameter for notificationDestination is not used to determine validation on this test.

5.2.5 Delete policy test scenarios

5.2.5.1 Delete single policy (positive case)

5.2.5.1.1 Test description and applicability

This purpose of this test case is to test the delete policy functionality of A1-P Consumer as specified in A1AP [4], clause 5.2.4.6. The expected outcome is successful validation of the Delete single policy request.

This test case is mandatory if the DUT claims to support A1-P service.

5.2.5.1.2 Test entrance criteria

- 1) The DUT has the functionality to initiate A1-P Delete policy procedure.
- 2) A policy exists in test simulator and the policyTypeId and the policyId is known to DUT.

5.2.5.1.3 Test methodology

5.2.5.1.3.1 Initial conditions

- 1) The test simulator has A1-P Producer service ready and available to receive HTTP requests from the DUT.

5.2.5.1.3.2 Procedure

Step 1. Initiate appropriate action in the DUT to initiate Delete policy request for the existing policy identified by policyTypeid and policyId.

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 A1AP [4], clause 6.2.3.
- b) The HTTP request is a DELETE operation.
- c) The policyTypeid and policyId in the URI match the policy being deleted
- d) The message body is empty.

Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.2.4.6.1.

5.2.5.1.3.3 Expected result

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

5.2.6 Feedback policy test scenarios

5.2.6.1 Policy feedback (positive case)

5.2.6.1.1 Test description and applicability

This purpose of this test case is to test the policy feedback functionality of A1-P Consumer as specified in A1AP [4], clause 5.2.4.8. The expected outcome is successful policy status request and reception.

This test case is conditionally mandatory if the DUT claims to support A1-P service and supports the Policy feedback procedure.

5.2.6.1.2 Test entrance criteria

The test entrance criteria for Create single policy specified in clause 5.2.2.1.2 applies. And in addition:

- 1) The DUT has the functionality to request and receive policy status updates.

5.2.6.1.3 Test methodology

5.2.6.1.3.1 Initial conditions

The initial conditions for Create single policy specified in clause 5.2.2.1.3.1 applies. And in addition:

- 3) The test simulator supports generation of a PolicyStatusObject based on the policy status schema of the agreed policy type used for this test.
- 4) The DUT has HTTP server ready and available to receive HTTP requests from the test simulator.

NOTE: The DUT is expected to provide callback URI (notificationDestination) during the Create single policy procedure (see A1AP [4], clause 5.2.4.3) for which it can relate received policy status updates to the policyId that was generated when creating the policy for which policy status updates is requested.

5.2.6.1.3.2 Procedure

- Step 1. Initiate appropriate action in the DUT to initiate Create single policy request for the agreed policy type including the notificationDestination query parameter.
- 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)-e) As defined for Create single policy in clause 5.2.2.1.3.2.
 - f) The notificationDestination query parameter is included.
- Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.2.4.8.1.
- Step 5. The test simulator initiates the Policy feedback procedure and sends a HTTP POST request to the provided callback URI (notificationDestination) in the DUT with message body containing the PolicyStatusObject conforming to the schema of the used policy type.
- Step 6. At the test simulator the received HTTP response is recorded.

NOTE: For the time between Step 4 and Step 5, the test simulator need not simulate any change in policy status but can send status update directly with any content that conforms to the status schema of the used policy type.

5.2.6.1.3.3 Expected result

The test is considered passed if the following conditions are met:

- 1) Step 3 validation has passed
- 2) Check the HTTP response recorded in step 6 of procedure and that the return code is "204 No content".

5.2.6.2 Policy feedback - schema validation failure (negative case)

5.2.6.2.1 Test description and applicability

This purpose of this test case is to test the policy feedback functionality of A1-P Consumer as specified in A1AP [4], clause 5.2.4.3. The expected outcome is failure due to schema validation failure.

This test case is conditionally mandatory if the DUT claims to support A1-P service and supports the Policy feedback procedure.

5.2.6.2.2 Test entrance criteria

The test entrance criteria in clause 5.2.6.1.2 applies.

5.2.6.2.3 Test methodology

5.2.6.2.3.1 Initial conditions

The initial conditions in clause 5.2.6.1.3.1 applies.

5.2.6.2.3.2 Procedure

Steps 1-6 of clause 5.2.6.1.3.2 applies with the following modification:

- In Step 5: A spelling mistake should be introduced in the PolicyStatusObject for schema validation error in DUT.

5.2.6.2.3.3 Expected result

The test is considered passed if the following conditions are met:

- 1) Step 3 validation has passed.
- 2) Check the HTTP response recorded in step 6 of procedure and that the return code is "400 Bad request".

5.2.6.3 Policy feedback -callback URI not supported (negative case)

5.2.6.2.1 Test description and applicability

This purpose of this test case is to test the policy feedback functionality of A1-P Consumer as specified in A1AP [4], clause 5.2.4.3. The expected outcome is failure due to invalid callback URI.

This test case is conditionally mandatory if the DUT claims to support A1-P service and supports the Policy feedback procedure.

5.2.6.2.2 Test entrance criteria

The test entrance criteria in clause 5.2.6.1.2 applies.

5.2.6.2.3 Test methodology

5.2.6.2.3.1 Initial conditions

The initial conditions in clause 5.2.6.1.3.1 applies.

5.2.6.2.3.2 Procedure

Steps 1-6 of clause 5.2.6.1.3.2 applies with the following modification:

- In Step 5. A spelling mistake should be introduced in the used callback URI making it different from the provided notificationDestination and not identical to any callback URI supported in the DUT.

5.2.6.2.3.3 Expected result

The test is considered passed if the following conditions are met:

- 1) Step 3 validation has passed.
- 2) Check the HTTP response recorded in step 6 of procedure and that the return code is "400 Bad request".

5.3 Conformance test cases for A1-EI Producer

5.3.1 Query EI types test scenarios

5.3.1.1 Query EI type identifiers (positive case)

5.3.1.1.1 Test description and applicability

This purpose of this test case is to test query EI type identifiers functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.3.2. The expected outcome is successful retrieval of EI type identifiers.

This testcase is conditionally mandatory if the DUT claims to support A1-EI service and Query EI type identifiers procedure.

5.3.1.1.2 Test entrance criteria

- 1) The DUT supports the Query EI type identifiers procedure.
- 2) The test simulator has the functionality to initiate A1-EI Query EI type identifiers procedure.
- 3) The DUT supports a known set of EI types (two or more) used for this test.

5.3.1.1.3 Test methodology

5.3.1.1.3.1 Initial conditions

- 1) The DUT has A1-EI Producer service ready and available to receive HTTP requests from the test simulator.
- 2) The DUT supports the three different test configurations listed below to test complete functionality:
 - a) DUT has no EI type available.
 - b) DUT has a single EI type available.
 - c) DUT has two or more EI types available.

5.3.1.1.3.2 Procedure

Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.3.3 and empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

Step 3. Repeat Step 1 and Step 2 for the three test configurations.

5.3.1.1.3.3 Expected result

At the test simulator check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met for all three test configurations listed in clause 5.3.1.1.3.1:

- 1) The return code is "200 OK".
- 2) Response message body is validated depending on the test configuration used:
 - a) If the DUT has no EI type available, message body is an empty array.
 - b) If the DUT has single EI type available, message body contains an array with one eiTypeId.
 - c) If the DUT has two or more EI jobs available, message body contains an array of all the eiTypeIds available in DUT.

5.3.1.2 Query EI type (positive case)

5.3.1.2.1 Test description and applicability

The purpose of this test case is to test query EI type functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.3.3. The expected outcome is successful retrieval of EI type object.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and the Query EI type procedure.

5.3.1.2.2 Test entrance criteria

- 1) The DUT supports the Query EI type procedure.
- 2) The test simulator has the functionality to initiate A1-EI Query EI type procedure.

- 3) The DUT and the test simulator have agreed on an EI type to use for this test.
- 4) The EI type identifier and the JSON schemas of the EI type used for this test are available and used in test simulator to formulate the Query EI request, and in DUT for selecting the appropriate schemas for the EI type object.

5.3.1.2.3 Test methodology

5.3.1.2.3.1 Initial conditions

- 1) The DUT has A1-EI Producer service ready and available to receive HTTP requests from the test simulator.
- 2) The DUT has the EI type available.

5.3.1.2.3.2 Procedure

Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.3.3 containing the known eiTypeId and empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

5.3.1.2.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "200 OK".
- 2) Response message body content contains an EiTypeObject representing the read EI job.

5.3.1.3 Query EI type (negative case) - eiTypeId not supported

5.3.1.3.1 Test description and applicability

The purpose of this test case is to test query EI type functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.3.3. The expected outcome is failure due to eiTypeId not being supported.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and the Query EI type procedure.

5.3.1.3.2 Test entrance criteria

The test entrance criteria specified in clause 5.3.1.2.2 applies, except that the agreed eiTypeId is NOT available in DUT.

5.3.1.3.3 Test methodology

5.3.1.3.3.1 Initial conditions

The initial conditions specified in clause 5.3.1.2.3.1 applies.

5.3.1.3.3.2 Procedure

Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.3.3 containing the unsupported eiTypeId and with empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

5.3.1.3.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "404 Not Found".

NOTE: Presence or validation of optional ProblemDetails object in the response is not used to determine validation on this test.

5.3.2 Create EI job test scenarios

5.3.2.1 Create single EI job (positive case)

5.3.2.1.1 Test description and applicability

This purpose of this test case is to test the create EI job functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.4.3. The expected outcome is successful creation of the EI job.

This testcase is mandatory if the DUT claims to support A1-EI service.

5.3.2.1.2 Test entrance criteria

- 1) The DUT and the test simulator have agreed on an EI type to use for this test.
- 2) The eiTypeId and the JSON schemas of the EI type used for this test are available and used in test simulator to formulate the Create EI job request, and in DUT to validate and handle the request.
- 3) The DUT supports the A1-EI Create EI job policy procedure.
- 4) The test simulator has the functionality to initiate Create EI job procedure.

5.3.2.1.3 Test methodology

5.3.2.1.3.1 Initial conditions

- 1) The DUT has A1-EI Producer service ready and available to receive HTTP requests from the test simulator.
- 2) No EI job exists in the DUT for the agreed EI type with the same eiJobId that will be used by the test simulator.

5.3.2.1.3.2 Procedure

Step 1. Send an HTTP PUT request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.3.3 including a new eiJobId, message body containing the EiJobObject in JSON format with the agreed eiTypeId included and conforming to the schema of the EI type.

Step 2. At the test simulator the received HTTP response is recorded.

5.3.2.1.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "201 Created".
- 2) Response message body content contains an EiJobObject representing the created EI job.
- 3) The location header is present and carries the URI of the new EI job.

5.3.2.2 Create EI job (negative case) - eiTypeId not supported

5.3.2.2.1 Test description and applicability

The purpose of this test case is to test the create EI job functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.4.3. The expected outcome is failure due to eiTypeId not being supported.

This testcase is mandatory if the DUT claims to support A1-EI service

5.3.2.2.2 Test entrance criteria

The test entrance criteria specified in clause 5.3.2.1.2 applies except that the agreed eiTypeId is NOT available in DUT.

5.3.2.2.3 Test methodology

5.3.2.2.3.1 Initial conditions

The initial conditions specified in clause 5.3.2.1.3.1 applies.

5.3.2.2.3.2 Procedure

Step 1. Send an HTTP PUT request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.3.3, and a message body containing the EiJobObject in JSON format including the unsupported eiTypeId.

Step 2. At the test simulator the received HTTP response is recorded.

5.3.2.2.3.2 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "404 Not found".

NOTE: Presence or validation of optional ProblemDetails object in the response is not used to determine validation on this test.

5.3.2.3 Create EI job (negative case) - schema validation failure

5.3.2.3.1 Test description and applicability

The purpose of this test case is to test the create EI job functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.4.3. The expected outcome is failure due to schema validation failure.

This testcase is mandatory if the DUT claims to support A1-EI service.

5.3.2.3.2 Test entrance criteria

The test entrance criteria specified in clause 5.3.2.1.2 applies.

5.3.2.3.3 Test methodology

5.3.2.3.3.1 Initial conditions

The initial conditions specified in clause 5.3.2.1.3.1 applies.

5.3.2.3.3.2 Procedure

- Step 1. Send an HTTP PUT request from test simulator to DUT with the correct URI as specified in A1AP [4], clause 6.3.3 and a message body containing the EiJobObject in JSON format. A spelling mistake should be introduced in the EiJobObject for schema validation error in DUT.
- Step 2. At the test simulator the received HTTP response is recorded.

5.3.2.3.3.2 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "400 Bad Request".

NOTE: Presence or validation of optional ProblemDetails object in the response is not used to determine validation on this test.

5.3.3 Query EI jobs test scenarios

5.3.3.1 Query EI job identifiers for a single EI type (positive case)

5.3.3.1.1 Test description and applicability

This purpose of this test case is to test query EI job identifiers functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.4.2. The expected outcome is successful retrieval of all EI job identifiers of a given EI type.

This testcase is conditionally mandatory if the DUT claims to support A1-EI service and Query EI job identifiers procedure.

5.3.3.1.2 Test entrance criteria

- 1) The DUT supports the Query EI job identifiers procedure.
- 2) The test simulator has the functionality to initiate A1-EI Query EI job identifiers procedure.
- 3) The DUT and the test simulator have agreed on an EI type to use for this test.

5.3.3.1.3 Test methodology

5.3.3.1.3.1 Initial conditions

- 1) The DUT has A1-EI Producer service ready and available to receive HTTP requests from the test simulator.
- 2) The DUT supports the three different test configurations listed below to test complete functionality:
 - a) DUT has no EI job available for the agreed EI type
 - b) DUT has a single EI job available for the agreed EI type
 - c) DUT has two or more EI job available for the agreed EI type

5.3.3.1.3.2 Procedure

- Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.3.3 with the agreed eiTypeId as query parameter and with empty message body.
- Step 2. At the test simulator the received HTTP response is recorded.
- Step 3. Repeat Step 1 and Step 2 for the three test configurations.

5.3.3.1.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met for all three test configurations:

- 1) The return code is "200 OK".
- 2) Response message body is validated depending on the three test configurations used in clause 5.3.3.1.3.1:
 - a) If the DUT has no EI job available, message body is an empty array
 - b) If the DUT has single EI job available, message body contains an array with one eiJobId available with the given EI type.
 - c) If the DUT has two or more EI jobs available, message body contains an array of eiJobIds of all the available EI jobs of the given EI type.

5.3.3.2 Query EI job identifiers for a single EI type (negative case) - eiTypeId not supported

5.3.3.2.1 Test description and applicability

This purpose of this test case is to test query EI job identifiers functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.4.2. The expected outcome is failure due to eiTypeId not being supported.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and the Query EI job identifiers procedure.

5.3.3.2.2 Test entrance criteria

The test entrance criteria specified in clause 5.3.3.1.2 applies, except that the agreed eiTypeId is NOT available in DUT.

5.3.3.2.3 Test methodology

5.3.3.2.3.1 Initial conditions

- 1) The DUT has A1-EI Producer service ready and available to receive HTTP requests from the test simulator.

5.3.3.2.3.2 Procedure

Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.3.3 with the unsupported eiTypeId included as query parameter and with empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

5.3.3.2.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "400 Bad request".

NOTE: Presence or validation of optional ProblemDetails object in the response is not used to determine validation on this test.

5.3.3.3 Query EI job identifiers for all EI types (positive case)

5.3.3.3.1 Test description and applicability

This purpose of this test case is to test query EI job identifiers functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.4.2. The expected outcome is successful retrieval of all EI job identifiers of all EI types.

This testcase is conditional mandatory if the DUT claims to support A1-EI service and Query EI job identifiers procedure.

5.3.3.3.2 Test entrance criteria

- 1) The DUT supports the Query EI job identifiers procedure.
- 2) The test simulator has the functionality to initiate A1-EI Query EI job identifiers procedure.
- 3) The DUT supports a known set of EI types (two or more) used for this test.

5.3.3.3.3 Test methodology

5.3.3.3.3.1 Initial conditions

- 1) The DUT has A1-EI Producer service ready and available to receive HTTP requests from the test simulator.
- 2) The DUT supports the two different test configurations listed below to test complete functionality:
 - a) DUT has no EI job available for any of the EI types supported.
 - b) DUT has one or more EI jobs available for each EI type supported.

5.3.3.3.3.2 Procedure

Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.3.3 without eiTypeId included as query parameter and with empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

Step 3. Repeat Step 1 and Step 2 for the two test configurations.

5.3.3.3.3.3 Expected result

At the test simulator check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met for all two test configurations:

- 1) The return code is "200 OK".
- 2) Response message body is validated depending on the two test configurations used in clause 5.3.3.3.3.1:
 - a) If the DUT has no EI job available, message body is an empty array.
 - b) If the DUT has one or more EI jobs available, message body contains an array of eiJobIds of all the available EI jobs of all EI types.

5.3.3.4 Query EI job (positive case)

5.3.3.4.1 Test description and applicability

The purpose of this test case is to test query EI job functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.4.5. The expected outcome is successful retrieval of EI job for the given eiJobId.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and the Query EI job procedure.

5.3.3.4.2 Test entrance criteria

- 1) The DUT supports the Query EI job procedure.
- 2) The test simulator has the functionality to initiate A1-EI Query EI job procedure.
- 3) An EI job exists in the DUT and the eiJobId is known to the test simulator.

5.3.3.4.3 Test methodology

5.3.3.4.3.1 Initial conditions

- 1) The DUT has A1-EI Producer service ready and available to receive HTTP request from the test simulator.

5.3.3.4.3.2 Procedure

Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.3.3 containing the known eiJobId and with empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

5.3.3.4.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "200 OK".
- 2) Response message body content contains an EiJobObject representing the read EI job.

5.3.3.5 Query EI job (negative case) - EI job does not exist

5.3.3.5.1 Test description and applicability

The purpose of this test case is to test query EI job functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.4.5. The expected outcome is failure due to EI job being non-existent.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and the Query EI job procedure.

5.3.3.5.2 Test entrance criteria

The test entrance criteria specified in clause 5.3.3.4.2 applies, additionally an eiJobId to use for which no EI job exist in the DUT is known in advance.

5.3.3.5.3 Test methodology

5.3.3.5.3.1 Initial conditions

The initial conditions specified in clause 5.3.3.4.3.1 applies.

5.3.3.5.3.2 Procedure

Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.3.3 containing the eiJobId for which no EI job exist and with empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

5.3.3.5.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "404 Not Found".

NOTE: Presence or validation of optional ProblemDetails object in the response is not used to determine validation on this test.

5.3.4 Update EI job test scenarios

5.3.4.1 Update EI job (positive case)

5.3.4.1.1 Test description and applicability

The purpose of this test case is to test update EI job functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.4.4. The expected outcome is successful update of the EI job.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and Update EI job procedure.

5.3.4.1.2 Test entrance criteria

- 1) The DUT supports the Update EI job procedure.
- 2) The test simulator has the functionality to initiate Update EI job procedure.
- 3) An EI job exists in the DUT and the eiTypeID and eiJobID is known to the test simulator.
- 4) The EI type and the JSON schemas of the EI type used for this test are available and used in test simulator to formulate the Update EI job request, and in DUT to validate and handle the request.

5.3.4.1.3 Test methodology

5.3.4.1.3.1 Initial conditions

- 1) The DUT has A1-EI Producer service ready and available to receive HTTP request from the test simulator.

5.3.4.1.3.2 Procedure

Step 1. Send an HTTP PUT request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.3.3 and message body containing the EiJobObject in JSON format with the agreed eiTypeId included and confirming to the schema of the EI type.

Step 2. At the test simulator the received HTTP response is recorded.

5.3.4.1.3.2 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "200 OK".
- 2) Response message body content contains an EiJobObject representing the updated EI job sent in Step 1.

5.3.4.2 Update EI job (negative case) - schema validation failure

5.3.4.2.1 Test description and applicability

The purpose of this test case is to test update EI job functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.4.4. The expected outcome is failure to update the EI job due to schema validation failure.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and Update EI job procedure.

5.3.4.2.2 Test entrance criteria

The test entrance criteria specified in clause 5.3.4.1.2 applies.

5.3.4.2.3 Test methodology

5.3.4.2.3.1 Initial conditions

The initial conditions specified in clause 5.3.4.1.3.1 applies.

5.3.4.2.3.2 Procedure

Step 1. Send an HTTP PUT request from test simulator to DUT with the correct URI as specified in A1AP [4], clause 6.3.3 and a message body containing the EiJobObject in JSON format. A spelling mistake should be introduced in the EiJobObject for schema validation error in the DUT.

Step 2. At the test simulator the received HTTP response is recorded.

5.3.4.2.3.2 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "400 Bad Request".

NOTE: Presence or validation of optional ProblemDetails object in the response is not used to determine validation on this test.

5.3.5 Delete EI job test scenarios

5.3.5.1 Delete EI job (positive case)

5.3.5.1.1 Test description and applicability

The purpose of this test case is to test the delete EI job functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.4.6. The expected outcome is successful deletion of EI job.

This test case is mandatory if the DUT claims to support A1-EI service and the Delete EI job procedure.

5.3.5.1.2 Test entrance criteria

- 1) The DUT supports the Delete EI job procedure.
- 2) The test simulator has the functionality to initiate A1-EI Delete EI job procedure.
- 3) An EI job exists in DUT and the eiJobId is known to test simulator.

5.3.5.1.3 Test methodology

5.3.5.1.3.1 Initial conditions

- 1) The DUT has A1-EI Producer service ready and available to receive HTTP request from the test simulator.

5.3.5.1.3.2 Procedure

Step 1. Send an HTTP DELETE request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.3.3 containing the known eiJobId and with empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

5.3.5.1.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "204 No Content"
- 2) The message body is empty.

5.3.5.2 Delete EI job (negative case) - EI job does not exist

5.3.5.2.1 Test description and applicability

The purpose of this test case is to test the delete EI job functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.4.6. The expected outcome is failure due to EI job being non-existent.

This test case is mandatory if the DUT claims to support A1-EI service and the Delete EI job procedure.

5.3.5.2.2 Test entrance criteria

The test entrance criteria specified in clause 5.3.5.1.2 applies, additionally the eiJobId to use for which no EI job exists in the DUT is known in advance.

5.3.5.2.3 Test methodology

5.3.5.2.3.1 Initial conditions

The initial conditions specified in clause 5.3.5.1.3.1 applies.

5.3.5.2.3.2 Procedure

Step 1. Send an HTTP DELETE request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.3.3 containing the unsupported eiJobId and with empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

5.3.5.2.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "404 Not Found".

NOTE: Presence or validation of optional ProblemDetails object in the response is not used to determine validation on this test.

5.3.6 Status of EI jobs test scenarios

5.3.6.1 Query EI job status (positive case)

5.3.6.1.1 Test description and applicability

The purpose of this test case is to test the query EI job status functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.4.7. The expected outcome is successful retrieval of EI job status object.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and the Query EI job status procedure.

5.3.6.1.2 Test entrance criteria

- 1) The DUT supports the Query EI job status procedure.
- 2) The test simulator has the functionality to initiate A1-EI Query EI job status procedure.
- 3) An EI job exists in the DUT and test simulator is aware of the eiJobId.

5.3.6.1.3 Test methodology

5.3.6.1.3.1 Initial conditions

- 1) The DUT has A1-EI Producer service ready and available to receive HTTP requests from the test simulator.

5.3.6.1.3.2 Procedure

Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.3.3 containing the known eiJobId and with empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

5.3.6.1.3.2 Expected result

Check the HTTP response recorded in step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "200 OK".
- 2) Response message body content contains an EiJobStatusObject representing the status of the EI job.

5.3.6.2 Query EI job status (negative case) EI job does not exist

5.3.6.2.1 Test description and applicability

The purpose of this test case is to test the query EI job status functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.4.7. The expected outcome is failure due to EI job being non-existent.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and the Query EI job status procedure.

5.3.6.2.2 Test entrance criteria

The test entrance criteria specified in clause 5.3.6.1.2 applies, additionally an eiJobId to use for which no EI job exist in the DUT is known in advance.

5.3.6.2.3 Test methodology

5.3.6.2.3.1 Initial conditions

The initial conditions specified in clause 5.3.6.1.3.1 applies.

5.3.6.2.3.2 Procedure

Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.3.3 containing the eiJobId for which no EI job exist and with empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

5.3.6.2.3.2 Expected result

Check the HTTP response recorded in step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "404 Not Found".

NOTE: Presence or validation of optional ProblemDetails object in the response is not used to determine validation on this test.

5.3.6.3 Notify EI job status (positive case)

5.3.6.1.1 Test description and applicability

This purpose of this test case is to test the notify EI job status functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.4.8. The expected outcome is successful request for EI job status.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and supports the Notify EI job status procedure.

5.3.6.1.2 Test entrance criteria

The test entrance criteria for Create EI job specified in clause 5.3.2.1.2 applies.

5.3.6.1.3 Test methodology

5.3.6.1.3.1 Initial conditions

The initial conditions for Create EI job specified in clause 5.3.2.1.3.1 applies.

5.3.6.1.3.2 Procedure

Step 1. Same as for Create EI job, see clause 5.3.2.1.3.2, including the jobStatusNotificationUri parameter in the EiJobObject.

Step 2. At the test simulator the received HTTP response is recorded.

Step 3. Same as Step 1 but with another jobStatusNotificationUri parameter included.

Step 4. At the test simulator the received HTTP response is recorded.

Step 5. Same as Step 1 but without a jobStatusNotificationUri parameter included.

Step 6. At the test simulator the received HTTP response is recorded.

NOTE: Steps 3-4 and 5-6 correspond to the procedure for Update EI job but it is not required that the EiJobObject in the message body is modified in the test case for Notify EI job status except for the jobStatusNotificationUri.

5.3.6.1.3.3 Expected result

Check the HTTP response recorded in Step 2, 4 and 6 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "201 Created" in Step 2.
- 2) The return code is "200 OK" in Step 4.
- 3) The return code is v200 OK" in Step 6.

5.3.7 Deliver EI job result test scenarios

5.3.7.1 Deliver EI job result (positive case)

5.3.7.1.1 Test description and applicability

This purpose of this test case is to test the deliver EI job result functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.5.2. The expected outcome is successful request for EI job results.

This test case is mandatory if the DUT claims to support A1-EI service and supports the Deliver EI job result procedure.

5.3.7.1.2 Test entrance criteria

The test entrance criteria for Create EI job specified in clause 5.3.2.1.2 applies.

5.3.7.1.3 Test methodology

5.3.7.1.3.1 Initial conditions

The initial conditions for Create EI job specified in clause 5.3.2.1.3.1 applies.

5.3.7.1.3.2 Procedure

- Step 1. Same as for Create EI job, see clause 5.3.2.1.3.2, including the jobResultUri parameter in the EiJobObject.
- Step 2. At the test simulator the received HTTP response is recorded.
- Step 3. Same as Step 1 but with another jobResultUri parameter included.
- Step 4. At the test simulator the received HTTP response is recorded.
- Step 5. Same as Step 1 but without a jobResultUri parameter included.
- Step 6. At the test simulator the received HTTP response is recorded.

NOTE: Steps 3-4 correspond to the procedure for Update EI job, but it is not required that the EiJobObject in the message body is modified in the test case for Deliver EI job status except for the jobResultUri.

5.3.7.1.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "201 Created" in Step 2.
- 2) The return code is "200 OK" in Step 4.

5.3.7.2 Deliver EI job result - no callback (negative case)

5.3.7.2.1 Test description and applicability

This purpose of this test case is to test the deliver EI job result functionality of A1-EI Producer as specified in A1AP [4], clause 5.3.5.2. The expected outcome is successful request for EI job results.

This test case is mandatory if the DUT claims to support A1-EI service and supports the Deliver EI job result procedure.

5.3.7.2.2 Test entrance criteria

The test entrance criteria specified in clause 5.3.7.1.2 applies.

5.3.7.2.3 Test methodology

5.3.7.2.3.1 Initial conditions

The initial conditions specified in clause 5.3.7.1.3.1 applies.

5.3.7.2.3.2 Procedure

Step 1. Same as for Create EI job, see clause 5.3.2.1.3.2 but without a jobResultUri parameter included.

Step 2. At the test simulator the received HTTP response is recorded.

5.3.7.2.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "400 Bad request" in Step 2.

6 Test cases for Near-Real Time RIC

6.1 General

6.1.1 Device under test requirements

The Near-Real Time RIC 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 procedures defined for a Near-Real Time RIC and the purpose of the test scenarios is to validate that it conforms to the A1 service definitions in A1AP [4].

In addition to the basic conformance requirements, the following is required for a Near-Real Time RIC when it acts as DUT:

- It shall be possible for the DUT to initiate the A1-EI procedures in a controlled manner for each test case.
- It shall be possible for the DUT to support A1 policy types or EI types to use in the test cases after agreement with the test simulator. It shall be possible to validate PolicyObject's and EiJobStatusObject's based on agreed policy types and EI types, and it shall be possible to formulate EI jobs based on the schemas of the agreed EI types.

NOTE: The present document does not require usage of any specific policy type or EI type since specific behaviour is not validated, only the application protocol aspects and procedures are validated for conformance testing of the A1 services.

6.1.2 Test simulator capabilities

The test simulator has the same basic capabilities as required for a Non-Real Time RIC. In addition, it has the following capabilities:

- Recording of received HTTP requests and responses and analysing them regarding compliance to the A1 service definitions.
- Controlled initiation of procedures with configurable URIs and payload formulated and modified based on the schemas of the agreed policy types and EI types.
- 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 A1 service definitions in A1AP [4].

6.2 Conformance Test Cases for A1-P Producer

6.2.1 Query policy type test scenarios

6.2.1.1 Query all policy type identifiers (positive case)

6.2.1.1.1 Test description and applicability

The purpose of this test case is to test the query policy types functionality of A1-P Producer as specified in A1AP [4], clause 5.2.3.2. The expected outcome is successful retrieval of policy type identifiers.

This test case is conditionally mandatory if the DUT claims to support A1-P service and the Query all policy type identifiers procedure.

6.2.1.1.2 Test entrance criteria

- 1) The DUT supports the Query all policy type identifiers procedure.
- 2) The test simulator has the functionality to initiate A1-P Query all policy type identifiers procedure.
- 3) A set of policy types (at least two) are supported in the DUT.

6.2.1.1.3 Test methodology

6.2.1.1.3.1 Initial conditions

- 1) The DUT has A1-P Producer service ready and available to receive HTTP requests from the test simulator.
- 2) The DUT supports the three different test configurations listed below to test complete functionality:
 - a) DUT has no policy types available.
 - b) DUT has a single policy type available.
 - c) DUT has two or more policy types available.

6.2.1.1.3.2 Procedure

Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.2.3 and message body is empty.

Step 2. At the test simulator the received HTTP response is recorded.

Step 3. Repeat Step 1 and Step 2 for the three test configurations.

6.2.1.1.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met for all three test configurations listed in clause 6.2.1.1.3.1:

- 1) The return code is "200 OK".
- 2) Response message body is validated depending on the three test configurations used:
 - a) If the DUT has no policy types available, message body is an empty array.
 - b) If the DUT has single policy type available, message body contains an array with one policyTypeid.
 - c) If the DUT has two or more policy types available, message body contains an array with all the policyTypeid's available in DUT.

6.2.1.2 Query single policy type (positive case)

6.2.1.2.1 Test description and applicability

This purpose of this test case is to test the query policy type functionality of A1-P Producer as specified in A1AP [4], clause 5.2.3.3. The expected outcome is successful retrieval of the policy type object.

This test case is mandatory if the DUT claims to support A1-P service.

6.2.1.2.2 Test entrance criteria

- 1) The DUT supports the Query single policy type procedure.
- 2) The test simulator has the functionality to initiate A1-P Query single policy type procedure.
- 3) A known set of policy types are supported in the DUT.
- 4) The policy type identifier used for this test is known in advance and used in test simulator to formulate the Query single policy type request, and in the DUT for selecting the appropriate schemas for the policy type object.

6.2.1.2.3 Test methodology

6.2.1.2.3.1 Initial conditions

- 1) The DUT has A1-P Producer service ready and available to receive HTTP requests from the test simulator.
- 2) The DUT has the policy types available.

6.2.1.2.3.2 Procedure

Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.2.3, including a known policyTypeid, and empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

6.2.1.2.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "200 OK".
- 2) Response message body content contains a PolicyTypeObject representing the read policy type.

6.2.1.3 Query single policy type (negative case) - policyTypeId not supported

6.2.1.3.1 Test description and applicability

This purpose of this test case is to test the query policy type functionality of A1-P Producer as specified in A1AP [4], clause 5.2.3.3. The expected outcome is failure due to policyTypeId not being supported.

This test case is mandatory if the DUT claims to support A1-P service.

6.2.1.3.2 Test entrance criteria

The test entrance criteria specified in clause 6.2.1.2.2 applies, except that the agreed policyTypeId is NOT available in DUT.

6.2.1.3.3 Test methodology

6.2.1.3.3.1 Initial conditions

The initial conditions specified in clause 6.2.1.2.3.1 applies.

6.2.1.3.3.2 Procedure

Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.2.3 containing the unsupported policyTypeId and with empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

6.2.1.3.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "404 Not Found".

NOTE: Presence or validation of optional ProblemDetails object in the response is not used to determine validation on this test.

6.2.2 Create policy test scenarios

6.2.2.1 Create single policy (positive case)

6.2.2.1.1 Test description and applicability

This purpose of this test case is to test the create policy functionality of A1-P Producer as specified in A1AP [4], clause 5.2.4.3. The expected outcome is successful creation of the policy.

This test case is mandatory if the DUT claims to support A1-P service.

6.2.2.1.2 Test entrance criteria

- 1) The DUT supports the A1-P Create single policy procedure.
- 2) The test simulator has the functionality to initiate A1-P Create single policy procedure.
- 3) DUT and the test simulator have agreed on a policy type to use for this test.
- 4) The policyTypeId and the JSON schemas of the policy type used for this test are available and used in test simulator to formulate the Create single policy request, and in DUT to validate and handle the request.

6.2.2.1.3 Test methodology

6.2.2.1.3.1 Initial conditions

- 1) The DUT has A1-P Producer service ready and available to receive HTTP requests from the test simulator.
- 2) No policy exists in the DUT for the agreed policy type with the same policyId that will be used by the test simulator.

6.2.2.1.3.2 Procedure

Step 1. Send an HTTP PUT request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.2.3 including policyTypeId and a new policyId, and message body containing the PolicyObject in JSON format conforming to the schema of the used policy type.

Step 2. At the test simulator the received HTTP response is recorded.

6.2.2.1.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "201 Created".
- 2) Response message body content contains a PolicyObject representing the created policy.
- 3) The location header is present and carries the URI of the new policy.

6.2.2.2 Create policy (negative case) - schema validation failure

6.2.2.2.1 Test description and applicability

This purpose of this test case is to test the create policy functionality of A1-P Producer as specified in A1AP [4], clause 5.2.4.3. The expected outcome is failure due to schema validation failure.

This test case is mandatory if the DUT claims to support A1-P service.

6.2.2.2.2 Test entrance criteria

The test entrance criteria specified in clause 6.2.6.1.2 applies.

6.2.2.2.3 Test methodology

6.2.2.2.3.1 Initial conditions

The initial conditions specified in clause 6.2.6.1.3.1 applies.

6.2.2.2.3.2 Procedure

Step 1. Send an HTTP PUT request from test simulator to DUT with the correct URI as specified in A1AP [4], clause 6.2.3 and a message body containing the PolicyObject in JSON format. A spelling mistake should be introduced in the PolicyObject for schema validation error in DUT.

Step 2. At the test simulator the received HTTP response is recorded.

6.2.2.2.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "400 Bad Request".

NOTE: Presence or validation of optional ProblemDetails object in the response is not used to determine validation on this test.

6.2.2.3 Create policy (negative case) - policyTypeId not supported

6.2.2.3.1 Test description and applicability

This purpose of this test case is to test the create policy functionality of A1-P Producer as specified in A1AP [4], clause 5.2.4.3. The expected outcome is failure due to policyTypeId not being supported.

This test case is mandatory if the DUT claims to support A1-P service.

6.2.2.3.2 Test entrance criteria

The test entrance criteria specified in clause 6.2.6.1.2 applies except that the agreed policyTypeId is NOT available in DUT.

6.2.2.3.3 Test methodology

6.2.2.3.3.1 Initial conditions

The initial conditions specified in clause 6.2.6.1.3.1 applies.

6.2.2.3.3.2 Procedure

Step 1. Send an HTTP PUT request from test simulator to DUT with the incorrect URI, i.e. containing the not supported policyTypeId, and a message body containing the PolicyObject in JSON format.

Step 2. At the test simulator the received HTTP response is recorded.

6.2.2.3.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "404 Not Found".

NOTE: Presence or validation of optional ProblemDetails object in the response is not used to determine validation on this test.

6.2.3 Query policy test scenarios

6.2.3.1 Query all policy identifiers (positive case)

6.2.3.1.1 Test description and applicability

The purpose of this test case is to test the query policies functionality of A1-P Producer as specified in A1AP [4], clause 5.2.4.2. The expected outcome is successful retrieval of policy identifiers.

This test case is conditionally mandatory if the DUT claims to support A1-P service and the Query all policy identifiers procedure.

6.2.3.1.2 Test entrance criteria

- 1) The DUT supports Query all policy identifiers procedure.
- 2) The test simulator has the functionality to initiate A1-P Query all policy identifiers procedure.
- 3) The DUT and the test simulator have agreed on a policy type to use for this test.

6.2.3.1.3 Test methodology

6.2.3.1.3.1 Initial conditions

- 1) The DUT has A1-P Producer service ready and available to receive HTTP requests from the test simulator.
- 2) The DUT supports the three different test configurations listed below to test complete functionality:
 - a) DUT has no policies available for the agreed policy type.
 - b) DUT has a single policy available for the agreed policy type.
 - c) DUT has two or more policies available for the agreed policy type.

6.2.3.1.3.2 Procedure

- Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.2.3 and empty message body.
- Step 2. At the test simulator the received HTTP response is recorded.
- Step 3. Repeat Step 1 and Step 2 for the three test configurations.

6.2.3.1.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met for all three test configurations specified in clause 6.2.3.1.3.1:

- 1) The return code is "200 OK".
- 2) Response message body is validated depending on the three test configurations used:
 - a) If the DUT has no policies available, message body is an empty array.
 - b) If the DUT has single policy available, message body contains an array with one policyId available with the given policy type.
 - c) If the DUT has two or more policies available, message body contains an array of policyIds of all the available policies of the given policy type.

6.2.3.2 Query all policy identifiers (negative case) - policyTypeId not supported

6.2.3.2.1 Test description and applicability

The purpose of this test case is to test the query policies functionality of A1-P Producer as specified in A1AP [4], clause 5.2.4.2. The expected outcome is failure due to policyTypeId not being supported.

This test case is conditionally mandatory if the DUT claims to support A1-P service and the Query all policy identifiers procedure.

6.2.3.2.2 Test entrance criteria

The test entrance criteria specified in clause 6.2.3.1.2 applies, except that the agreed policyTypeId is NOT available in DUT.

6.2.3.2.3 Test methodology

6.2.3.2.3.1 Initial conditions

- 1) The DUT has A1-P Producer service ready and available to receive HTTP requests from the test simulator.

6.2.3.2.3.2 Procedure

Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.2.3 and empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

6.2.3.2.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The appropriate error code "404 Not Found" is returned.

NOTE: Presence or validation of optional ProblemDetails object in the response is not used to determine validation on this test.

6.2.3.3 Query single policy (positive case)

6.2.3.3.1 Test description and applicability

The purpose of this test case is to test the query policy functionality of A1-P Producer as specified in A1AP [4], clause 5.2.4.5. The expected outcome is successful retrieval of policy object.

This test case is conditionally mandatory if the DUT claims to support A1-P service and the Query single policy procedure.

6.2.3.3.2 Test entrance criteria

- 1) A policy exists in the DUT and test simulator is aware of the policyTypeId and policyId.
- 2) The DUT supports the Query single policy procedure.
- 3) The test simulator has the functionality to initiate A1-P Query single policy procedure.

6.2.3.3.3 Test methodology

6.2.3.3.3.1 Initial conditions

- 1) The DUT supports the Query single policy procedure.
- 2) The test simulator has the functionality to initiate A1-P Query single policy procedure.

6.2.3.3.3.2 Procedure

Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.2.3 and empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

6.2.3.3.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "200 OK".
- 2) Response message body content contains a PolicyObject representing the read policy.

6.2.3.4 Query single policy (negative case) - policy does not exist

6.2.3.4.1 Test description and applicability

The purpose of this test case is to test the query policy functionality of A1-P Producer as specified in A1AP [4], clause 5.2.4.5. The expected outcome is failure due to policy being non-existent.

This test case is conditionally mandatory if the DUT claims to support A1-P service and the Query single policy procedure.

6.2.3.4.2 Test entrance criteria

- 1) The DUT supports Query single policy procedure.
- 2) The test simulator has the functionality to initiate A1-P Query single policy procedure.
- 3) test simulator is aware of the policyTypeId and policyId to use for which no policy exists in the DUT.

6.2.3.4.3 Test methodology

6.2.3.4.3.1 Initial conditions

- 1) The DUT has A1-P Producer service ready and available to receive HTTP requests from the test simulator.

6.2.3.4.3.2 Procedure

Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.2.3 and empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

6.2.3.4.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "404 Not Found".

NOTE: Presence or validation of optional ProblemDetails object in the response is not used to determine validation on this test.

6.2.3.5 Query policy status (positive case)

6.2.3.5.1 Test description and applicability

The purpose of this test case is to test the query policy status functionality of A1-P Producer as specified in A1AP [4], clause 5.2.4.7. The expected outcome is successful retrieval of policy status object.

This test case is conditionally mandatory if the DUT claims to support A1-P service and the Query policy status procedure.

6.2.3.5.2 Test entrance criteria

- 1) The DUT supports the Query policy status procedure.
- 2) The test simulator has the functionality to initiate A1-P Query policy status procedure.
- 3) A policy exists in the DUT and the policyTypeId and policyId is known to the test simulator.

6.2.3.5.3 Test methodology

6.2.3.5.3.1 Initial conditions

- 1) The DUT has A1-P Producer service ready and available to receive HTTP requests from the test simulator.

6.2.3.5.3.2 Procedure

Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.2.3 and empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

6.2.3.5.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "200 OK".
- 2) Response message body content contains a PolicyStatusObject representing the status of the policy.

6.2.3.6 Query policy status (negative case) - policy does not exist

6.2.3.6.1 Test description and applicability

The purpose of this test case is to test the query policy status functionality of A1-P Producer as specified in A1AP [4], clause 5.2.4.7. The expected outcome is failure due to policy being non-existent.

This test case is conditionally mandatory if the DUT claims to support A1-P service and the Query policy status procedure.

6.2.3.6.2 Test entrance criteria

- 1) The DUT supports Query policy status procedure.
- 2) The test simulator has the functionality to initiate A1-P Query policy status procedure.
- 3) Test simulator is aware of the policyTypeId and policyId to use for which no policy exists in the DUT.

6.2.3.6.3 Test methodology

6.2.3.6.3.1 Initial conditions

- 1) The DUT has A1-P Producer service ready and available to receive HTTP requests from the test simulator.
- 2) No policy exists for the policyTypeId and policyId that will be used by the test simulator.

6.2.3.6.3.2 Procedure

Step 1. Send an HTTP GET request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.2.3 and empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

6.2.3.6.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "404 Not Found".

NOTE: Presence or validation of optional ProblemDetails object in the response is not used to determine validation on this test.

6.2.4 Update policy test scenarios

6.2.4.1 Update single policy (positive case)

6.2.4.1.1 Test description and applicability

The purpose of this test case is to test the update policy functionality of A1-P Producer as specified in A1AP [4], clause 5.2.4.4. The expected outcome is successful update of the policy.

This test case is conditionally mandatory if the DUT claims to support A1-P service and the Update single policy procedure.

6.2.4.1.2 Test entrance criteria

- 1) The DUT supports the Update single policy procedure.
- 2) The test simulator has the functionality to initiate A1-P Update single policy procedure.
- 3) A policy exists in the DUT and the policyTypeId and policyId is known to the test simulator.
- 4) Policy type identifiers used for this test are known in advance and used in DUT to formulate the Update single policy request, in test simulator for selecting the appropriate schemas for the policy type object.

6.2.4.1.3 Test methodology

6.2.4.1.3.1 Initial conditions

- 1) The DUT has A1-P Producer service ready and available to receive HTTP request from the test simulator.

6.2.4.1.3.2 Procedure

Step 1. Send an HTTP PUT request from test simulator to DUT with the correct URI as specified in A1AP [4], clause 6.2.3 and message body containing the PolicyObject in JSON format.

Step 2. At the test simulator the received HTTP response is recorded.

6.2.4.1.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "200 OK".
- 2) Response message body content is verified to match the PolicyObject sent in Step 1.

6.2.4.2 Update single policy (negative case) - schema validation failure

6.2.4.2.1 Test description and applicability

The purpose of this test case is to test the update policy functionality of A1-P Producer as specified in A1AP [4], clause 5.2.4.4. The expected outcome is failure due to schema validation failure.

This test case is conditionally mandatory if the DUT claims to support A1-P service and the Update single policy procedure.

6.2.4.2.2 Test entrance criteria

The test entrance criteria specified in clause 6.2.6.1.2 applies.

6.2.4.2.3 Test methodology

6.2.4.2.3.1 Initial conditions

The initial conditions specified in clause 6.2.4.1.3.1 applies.

6.2.4.2.3.2 Procedure

Step 1. Send an HTTP PUT request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.2.3 and a message body containing the PolicyObject in JSON format. A spelling mistake should be introduced in the PolicyObject for schema validation error in DUT

Step 2. At the test simulator the received HTTP response is recorded.

6.2.4.2.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "400 Bad Request".

NOTE: Presence or validation of optional ProblemDetails object in the response is not used to determine validation on this test.

6.2.5 Delete policy test scenarios

6.2.5.1 Delete single policy (positive case)

6.2.5.1.1 Test description and applicability

The purpose of this test case is to test the delete policy functionality of A1-P Producer as specified in A1AP [4], clause 5.2.4.6. The expected outcome is successful deletion of the policy

This test case is mandatory if the DUT claims to support A1-P service.

6.2.5.1.2 Test entrance criteria

- 1) The DUT supports the Delete policy procedure
- 2) The test simulator has the functionality to initiate Delete policy procedure.
- 3) A policy exists in the DUT and the policyTypeId and policyId is known to the test simulator.

6.2.5.1.3 Test methodology

6.2.5.1.3.1 Initial conditions

- 1) The DUT has A1-P Producer service ready and available to receive HTTP requests from the test simulator.

6.2.5.1.3.2 Procedure

Step 1. Send an HTTP DELETE request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.2.3.

Step 2. At the test simulator the received HTTP response is recorded.

6.2.5.1.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "204 No content".
- 2) Response message body is empty.

6.2.5.2 Delete single policy (negative case) - policy does not exist

6.2.5.2.1 Test description and applicability

This purpose of this test case is to test the delete policy functionality of A1-P Producer as specified in A1AP [4], clause 5.2.4.6. The expected outcome is failure due to policy being non-existent.

This test case is mandatory if the DUT claims to support A1-P service.

6.2.5.2.2 Test entrance criteria

- 1) The DUT supports the Delete single procedure.
- 2) The test simulator has the functionality to initiate Delete single policy procedure.
- 3) Test simulator is aware of the policyTypeId and policyId to use for which no policy exists in the DUT.

6.2.5.2.3 Test methodology

6.2.5.2.3.1 Initial conditions

- 1) DUT has A1-P Producer service ready and available to receive HTTP requests from the test simulator.
- 2) No policy exists in DUT for the policyTypeId and policyId that will be used by the test simulator.

6.2.5.2.3.2 Procedure

Step 1. Send an HTTP DELETE request from test simulator to DUT with the correct URI format as specified in A1AP [4], clause 6.2.3 and empty message body.

Step 2. At the test simulator the received HTTP response is recorded.

6.2.5.2.3.3 Expected result

Check the HTTP response recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "404 Not Found".

NOTE: Presence or validation of optional ProblemDetails object in the response is not used to determine validation on this test.

6.2.6 Feedback policy test scenarios

6.2.6.1 Feedback policy (positive case)

6.2.6.1.1 Test description and applicability

This purpose of this test case is to test the feedback policy functionality of A1-P Producer as specified in A1AP [4], clause 5.2.4.8. The expected outcome is successful request for policy status.

This test case is conditionally mandatory if the DUT claims to support A1-P service and supports the Policy feedback procedure.

6.2.6.1.2 Test entrance criteria

The test entrance criteria for Create single policy specified in clause 6.2.2.1.2 applies.

6.2.6.1.3 Test methodology

6.2.6.1.3.1 Initial conditions

The initial conditions for Create single policy specified in clause 6.2.2.1.3.1 applies.

6.2.6.1.3.2 Procedure

- Step 1. Same as for Create single policy, see clause 6.2.2.1.3.2, including the notificationDestination query parameter.
- Step 2. At the test simulator the received HTTP response is recorded.
- Step 3. Same as Step 1 but with another notificationDestination query parameter included.
- Step 4. At the test simulator the received HTTP response is recorded.
- Step 5. Same as Step 1 but without a notificationDestination query parameter included.
- Step 6. At the test simulator the received HTTP response is recorded.

NOTE: Steps 3-4 and 5-6 correspond to the procedure for Update single policy but is not required that the PolicyObject in the message body is modified in the test case for Feedback policy.

6.2.6.1.3.3 Expected result

Check the HTTP response recorded in Step 2, 4 and 6 of procedure.

The test is considered passed if the following conditions are met:

- 1) The return code is "201 Created" in Step 2.
- 2) The return code is "200 OK" in Step 4.
- 3) The return code is "200 OK" in Step 6.

6.3 Conformance test cases for A1-EI Consumer

6.3.1 Query EI types test scenarios

6.3.1.1 Query EI type identifiers (positive case)

6.3.1.1.1 Test description and applicability

The purpose of this test case is to test query EI type identifiers functionality of A1-EI Consumer as specified in A1AP [4], clause 5.3.3.2. The expected outcome is successful validation of the Query EI type identifiers request.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and the Query EI type identifiers procedure.

6.3.1.1.2 Test entrance criteria

- 1) The DUT has functionality to initiate A1-EI Query EI type identifiers procedure.
- 2) The test simulator supports the Query EI type identifiers procedure.

6.3.1.1.3 Test methodology

6.3.1.1.3.1 Initial conditions

- 1) The test simulator has A1-EI Producer service ready and available to receive HTTP requests from the DUT.

6.3.1.1.3.2 Procedure

Step 1. Initiate appropriate action in the DUT to initiate Query EI type identifiers 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 URI conforms to the format specified in A1AP [4], clause 6.3.3.
- b) The HTTP request is a GET operation.
- c) The message body is empty.

Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.3.3.2.1.

6.3.1.1.3.3 Expected result

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

6.3.1.2 Query EI type (positive case)

6.3.1.2.1 Test description and applicability

The purpose of this test case is to test query EI type functionality of A1-EI Consumer as specified in A1AP [4], clause 5.3.3.3. The expected outcome is successful validation of the Query EI type request.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and the Query EI type procedure.

6.3.1.2.2 Test entrance criteria

- 1) The DUT has the functionality to initiate A1-P Query EI type procedure.
- 2) A known set of EI types are supported in the test simulator.

- 3) The EI type identifier used for this test is known in advance and available in DUT to send the Query EI type request for the corresponding EI type and used by the test simulator for selecting the appropriate schemas for the EI job object.

6.3.1.2.3 Test methodology

6.3.1.2.3.1 Initial conditions

- 1) The test simulator has A1-EI Producer service ready and available to receive HTTP requests from the DUT.
- 2) The test simulator has EI types available.

6.3.1.2.3.2 Procedure

Step 1. Initiate appropriate action in the DUT to initiate Query EI type request with a known eiTypeId.

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 GET operation.
- b) The URI conforms to the format specified in A1AP [4], clause 6.3.3.
- c) The HTTP request message body is empty.
- d) The eiTypeId in the URI match the EI type being queried.

Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.3.3.3.1.

6.3.1.2.3.3 Expected result

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

6.3.2 Create EI job test scenarios

6.3.2.1 Create single EI job (positive case)

6.3.2.1.1 Test description and applicability

This purpose of this test case is to test the create EI job functionality of A1-EI Consumer as specified in A1AP [4], clause 5.3.4.3. The expected outcome is successful validation of the EI job creation request.

This testcase is mandatory if the DUT claims to support A1-EI service.

6.3.2.1.2 Test entrance criteria

- 1) The DUT and the test simulator have agreed on an EI type to use for this test.
- 2) The eiTypeId and the JSON schemas of the EI type used for this test are available and used in DUT to formulate the Create EI job request, and in test simulator to validate the request.
- 3) The DUT has the functionality to initiate A1-EI Create EI job procedure.

6.3.2.1.3 Test methodology

6.3.2.1.3.1 Initial conditions

- 1) The test simulator has A1-EI Producer service ready and available to receive HTTP requests from the DUT.

- 2) No EI job exists in the test simulator for the agreed EI type with the same eiJobId that will be used by the DUT.

NOTE: The DUT is required to generate a unique eiJobId during the Create single EI job procedure (see A1AP [4], clause 5.3.4.3).

6.3.2.1.3.2 Procedure

- Step 1. Initiate appropriate action in the DUT to initiate Create EI job request for the agreed EI 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 A1AP [4], clause 6.3.3.
 - b) The HTTP request is a PUT operation.
 - c) The eiTypeId is supported.
 - d) An EI job with the eiJobId does not exist.
 - e) The HTTP request message body contains the EiJobObject of the EI job to be created and the EiJobObject conforms to the schema.
- Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.3.4.3.1.

6.3.2.1.3.3 Expected result

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

NOTE: Presence or validation of optional parameter for jobStatusNotificationUri is not used to determine validation on this test.

6.3.3 Query EI jobs test scenarios

6.3.3.1 Query EI job identifiers for a single EI type (positive case)

6.3.3.1.1 Test description and applicability

The purpose of this test case is to test query EI job identifiers functionality of A1-EI Consumer as specified in A1AP [4], clause 5.3.4.2. The expected outcome is successful validation of the Query EI job identifiers request querying for a single EI type.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and the Query EI job identifiers procedure.

6.3.3.1.2 Test entrance criteria

- 1) The DUT has functionality to initiate A1-EI Query EI job identifiers procedure.
- 2) An EI type is agreed to be used in this test and the eiTypeId is known in advance.
- 3) A set of EI jobs (at least two) exist in the test simulator for the agreed EI type.

6.3.3.1.3 Test methodology

6.3.3.1.3.1 Initial conditions

- 1) The test simulator has A1-EI Producer service ready and available to receive HTTP requests from the DUT.
- 2) The test simulator has EI jobs available for the agreed EI type.

6.3.3.1.3.2 Procedure

- Step 1. Initiate appropriate action in the DUT to initiate Query EI job identifiers request for the agreed EI 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 HTTP request is a GET operation.
 - b) The URI conforms to the format specified in A1AP [4], clause 6.3.3.
 - c) The message body is empty.
 - d) A query parameter with an eiTypeId is included.

Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.3.4.2.1.

6.3.3.1.3.3 Expected result

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

6.3.3.2 Query EI job identifiers for all EI types (positive case)

6.3.3.2.1 Test description and applicability

The purpose of this test case is to test query EI job identifiers functionality of A1-EI Consumer as specified in A1AP [4], clause 5.3.4.2. The expected outcome is successful validation of the Query EI job identifiers request querying for all EI types.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and the Query EI job identifiers procedure.

6.3.3.2.2 Test entrance criteria

- 1) The DUT has functionality to initiate A1-EI Query EI job identifiers procedure.
- 2) The test simulator supports at least one EI type.

6.3.3.2.3 Test methodology

6.3.3.2.3.1 Initial conditions

- 1) The test simulator has A1-EI Producer service ready and available to receive HTTP requests from the DUT.
- 2) One or more EI jobs exist in the test simulator for each EI type supported.

6.3.3.2.3.2 Procedure

Step 1. Initiate appropriate action in the DUT to initiate Query EI job identifiers request for all EI types.

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 A1AP [4], clause 6.3.3.
- b) The HTTP request is a GET operation.
- c) No query parameter with an eiTypeId is included.
- d) The message body is empty.

Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.3.4.2.1.

6.3.3.2.3.3 Expected result

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

6.3.3.3 Query EI job (positive case)

6.3.3.3.1 Test description and applicability

The purpose of this test case is to test query EI job functionality of A1-EI Consumer as specified in A1AP [4], clause 5.3.4.5. The expected outcome is successful validation of the Query EI job request.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and the Query EI job procedure.

6.3.3.3.2 Test entrance criteria

- 1) The DUT has functionality to initiate A1-EI Query EI job procedure.
- 2) An EI job exists in test simulator and DUT is aware of the eiJobId.

6.3.3.3.3 Test methodology

6.3.3.3.3.1 Initial conditions

- 1) The test simulator has A1-EI Producer service ready and available to receive HTTP request from the DUT.

6.3.3.3.3.2 Procedure

Step 1. Initiate appropriate action in the DUT to initiate Query EI job request for the existing EI job identified by the eiJobId.

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 A1AP [4], clause 6.3.3.
- b) The HTTP request is a GET operation.
- c) The eiJobId in the URI match the EI job being queried.
- d) The HTTP request message body is empty.

Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.3.4.5.1.

6.3.3.3.3.3 Expected result

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

6.3.4 Update EI job test scenarios

6.3.4.1 Update EI job (positive case)

6.3.4.1.1 Test description and applicability

The purpose of this test case is to test update EI job functionality of A1-EI Consumer as specified in A1AP [4], clause 5.3.4.4. The expected outcome is successful validation of the Update EI job request.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and the Update EI job procedure.

6.3.4.1.2 Test entrance criteria

- 1) The DUT has functionality to initiate A1-EI Update Job procedure.
- 2) An EI job exists in test simulator and DUT is aware of the eiTypeId and eiJobId.
- 3) The eiTypeId and the JSON schemas of the EI job type used for this test are available in DUT to formulate the Update EI job request and used by the test simulator for validation purpose.

6.3.4.1.3 Test methodology

6.3.4.1.3.1 Initial conditions

- 1) The test simulator has A1-EI Producer service ready and available to receive HTTP request from the DUT.

6.3.4.1.3.2 Procedure

Step 1. Initiate appropriate action in the DUT to initiate Update EI job request for the existing EI job identified by the eiJobId.

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 A1AP [4], clause 6.3.3.
- b) The HTTP request is a PUT operation.
- c) The eiJobId in the URI match the EI job to be updated.
- d) The HTTP request message body contains the EiJobObject of the EI job being updated with the correct eiTypeId included and the EiJobObject conforms to the schema of the EI type.

Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.3.4.4.1.

6.3.4.1.3.3 Expected result

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

NOTE: Presence or validation of optional query parameter for jobStatusNotificationUri is not used to determine validation on this test

6.3.5 Delete EI job test scenarios

6.3.5.1 Delete EI job (positive case)

6.3.5.1.1 Test description and applicability

The purpose of this test case is to test the delete EI job functionality of A1-EI Consumer as specified in A1AP [4], clause 5.3.4.6. The expected outcome is successful validation of the Delete EI job request.

This test case is mandatory if the DUT claims to support A1-EI service.

6.3.5.1.2 Test entrance criteria

- 1) The DUT has functionality to initiate A1-EI Delete EI job procedure.
- 2) An EI job exists in test simulator and the eiJobId is known to DUT.

6.3.5.1.3 Test methodology

6.3.5.1.3.1 Initial conditions

- 1) The test simulator has A1-EI Producer service ready and available to receive HTTP request from the DUT.

6.3.5.1.3.2 Procedure

Step 1. Initiate appropriate action in the DUT to initiate Delete EI job request for the existing EI job identified by the eiJobId.

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 A1AP [4], clause 6.3.3.
- b) The HTTP request is a DELETE operation.
- c) The eiJobId in the URI match the EI job being deleted.
- d) The HTTP request message body is empty.

Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.3.4.6.1.

6.3.5.1.3.3 Expected result

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

6.3.6 Status of EI jobs test scenarios

6.3.6.1 Query EI job status (positive case)

6.3.6.1.1 Test description and applicability

The purpose of this test case is to test the query EI job status functionality of A1-EI Consumer as specified in A1AP [4], clause 5.3.4.7. The expected outcome is successful validation of the Query EI job status request.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and the Query EI job status procedure.

6.3.6.1.2 Test entrance criteria

- 1) The DUT has functionality to initiate A1-EI Query EI job status procedure.
- 2) An EI job exists in test simulator and DUT is aware of the eiJobId.

6.3.6.1.3 Test methodology

6.3.6.1.3.1 Initial conditions

- 1) The test simulator has A1-EI Producer service ready and available to receive HTTP request from the DUT.

6.3.6.1.3.2 Procedure

Step 1. Initiate appropriate action in the DUT to initiate Query EI job status request for the existing EI job identified by the eiJobId.

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 A1AP [4], clause 6.3.3.
- b) The HTTP request is a GET operation.
- c) The eiJobId in the URI match the EI job being queried.
- d) The HTTP request message body is empty.

If the validation is successful, an HTTP response "200 OK" is sent with the message body containing EiJobStatusObject for the given eiJobId.

Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.3.4.7.1.

6.3.6.1.3.3 Expected result

The test is considered passed if Step 3 validation has passed and HTTP response "200 OK" is sent.

6.3.6.2 Notify EI job status (positive case)

6.3.6.2.1 Test description and applicability

This purpose of this test case is to test the notify EI job status functionality of A1-EI Consumer as specified in A1AP [4], clause 5.3.4.8. The expected outcome is successful EI job status notification request and reception.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and supports the Notify EI job procedure.

6.3.6.2.2 Test entrance criteria

The test entrance criteria for Create EI job specified in clause 6.3.2.1.2 applies. And in addition:

- 1) The DUT has the functionality to request and receive EI job status updates.

6.3.6.2.3 Test methodology

6.3.6.2.3.1 Initial conditions

The initial conditions for Create EI job specified in clause 6.3.2.1.3.1 applies. And in addition:

- 1) The test simulator supports generation of an EiJobStatusObject based on the EI job status schema of the agreed EI type used for this test.
- 2) The DUT has HTTP server ready and available to receive HTTP requests from the test simulator.

NOTE: The DUT is expected to provide callback URI (jobStatusNotificationUri) during the Create EI job procedure (see A1AP [4], clause 5.3.4.43) for which it can relate received EI job status updates to the eiJobId that was generated when creating the EI job for which EI job status updates is requested.

6.3.6.2.3.2 Procedure

Step 1. Initiate appropriate action in the DUT to initiate Create EI job request for the agreed EI job type including the jobStatusNotificationUri parameter in the EiJobObject.

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)-e) As defined for Create EI job in clause 6.3.2.1.3.2.
- f) The jobStatusNotificationUri parameter is included in the EiJobObject.

Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.3.4.3.1.

Step 5. The test simulator initiates the Notify EI job status procedure and sends a HTTP POST request to the provided callback URI (jobStatusNotificationUri) in the DUT with message body containing the EiJobStatusObject conforming to the schema of the used EI type.

Step 6. At the test simulator the received HTTP response is recorded.

NOTE: For the time between Step 4 and Step 5, the test simulator need not simulate any change in job status but can send status update directly with any content that conforms to the status schema of the used EI type.

6.3.6.2.3.3 Expected result

The test is considered passed if the following conditions are met:

- 1) Step 3 validation has passed
- 2) Check the HTTP response recorded in step 6 of procedure and that the return code is "204 No content".

6.3.6.3 Notify EI job status - schema validation failure (negative case)

6.3.6.3.1 Test description and applicability

This purpose of this test case is to test the notify EI job status functionality of A1-EI Consumer as specified in A1AP [4], clause 5.3.4.8. The expected outcome is failure due to schema validation failure.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and supports the Notify EI job status procedure.

6.3.6.3.2 Test entrance criteria

The test entrance criteria in clause 6.3.6.2.2 applies.

6.3.6.3.3 Test methodology

6.3.6.3.3.1 Initial conditions

The initial conditions in clause 6.3.6.2.3.1 applies.

6.3.6.3.3.2 Procedure

Steps 1-6 of clause 6.3.6.2.3.2 applies with the following modification:

- In Step 5: A spelling mistake should be introduced in the EiJobStatusObject for schema validation error in DUT.

6.3.6.3.3.3 Expected result

The test is considered passed if the following conditions are met:

- 1) Step 3 validation has passed.
- 2) Check the HTTP response recorded in step 6 of procedure and that the return code is "400 Bad request".

6.3.6.4 Notify EI job status - callback URI not supported (negative case)

6.3.6.4.1 Test description and applicability

This purpose of this test case is to test the notify EI job status functionality of A1-EI Consumer as specified in A1AP [4], clause 5.3.4.8. The expected outcome is failure due to invalid callback URI.

This test case is conditionally mandatory if the DUT claims to support A1-EI service and supports the Notify EI job status procedure.

6.3.6.4.2 Test entrance criteria

The test entrance criteria in clause 6.3.6.2.2 applies.

6.3.6.4.3 Test methodology

6.3.6.4.3.1 Initial conditions

The initial conditions in clause 6.3.6.2.3.1 applies.

6.3.6.4.3.2 Procedure

Steps 1-6 of clause 6.3.6.2.3.2 applies with the following modification:

- In Step 5. A spelling mistake should be introduced in the used callback URI making it different from the provided jobStatusNotificationUri and not identical to any callback URI supported in the DUT.

6.3.6.4.3.3 Expected result

The test is considered passed if the following conditions are met:

- 1) Step 3 validation has passed.
- 2) Check the HTTP response recorded in step 6 of procedure and that the return code is "400 Bad request".

6.3.7 Deliver EI job result test scenarios

6.3.7.1 Deliver EI job result (positive case)

6.3.7.1.1 Test description and applicability

This purpose of this test case is to test the deliver EI job result functionality of A1-EI Consumer as specified in A1AP [4], clause 5.3.5.2. The expected outcome is successful EI job result request and reception.

This test case is mandatory if the DUT claims to support A1-EI service.

6.3.7.1.2 Test entrance criteria

The test entrance criteria for Create EI job specified in clause 6.3.2.1.2 applies.

NOTE: No specific functionality is needed in DUT to request EI job results as it is covered by Create EI job procedure.

6.3.7.1.3 Test methodology

6.3.7.1.3.1 Initial conditions

The initial conditions for Create EI job specified in clause 6.3.2.1.3.1 applies. And in addition:

- 1) The test simulator supports generation of an EiJobResultObject based on the EI job result schema of the agreed EI type used for this test.
- 2) The DUT has HTTP server ready and available to receive HTTP requests from the test simulator.

NOTE: The DUT is expected to provide callback URI (jobResultUri) during the Create EI job procedure (see A1AP [4], clause 5.3.4.3) for which it can relate received EI job results to the eiJobId that was generated when creating the EI job for which EI job result is delivered.

6.3.7.1.3.2 Procedure

- Step 1. Initiate appropriate action in the DUT to initiate Create EI job request for the agreed EI job type including the jobResultUri parameter in the EiJobObject.
- 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)-e) As defined for Create EI job in clause 6.3.2.1.3.2.
 - f) The jobResultUri parameter is included in the EiJobObject.
- Step 4. The test simulator generates the appropriate HTTP response as specified in A1AP [4], clause 5.3.4.3.1.
- Step 5. The test simulator initiates the Deliver EI job result procedure and sends a HTTP POST request to the provided callback URI (jobResultUri) in the DUT with message body containing the EiJobResultObject conforming to the schema of the used EI type.
- Step 6. At the test simulator the received HTTP response is recorded.

NOTE: For the time between Step 4 and Step 5, the test simulator need not follow the timing attributes in the EiJobObject if such are defined for the used EI type but can send the job delivery directly with any content that conforms to the result schema of the used EI type.

6.3.7.1.3.3 Expected result

The test is considered passed if the following conditions are met:

- 1) Step 3 validation has passed.
- 2) Check the HTTP response recorded in step 6 of procedure and that the return code is "204 No content".

6.3.7.2 Deliver EI job result - schema validation failure (negative case)

6.3.7.2.1 Test description and applicability

This purpose of this test case is to test the deliver EI job result functionality of A1-EI Consumer as specified in A1AP [4], clause 5.3.5.2. The expected outcome is failure due to schema validation failure.

This test case is mandatory if the DUT claims to support A1-EI service

6.3.7.2.2 Test entrance criteria

The test entrance criteria in clause 6.3.7.1.2 applies.

6.3.7.2.3 Test methodology

6.3.7.2.3.1 Initial conditions

The initial conditions in clause 6.3.7.1.3.1 applies.

6.3.7.2.3.2 Procedure

Steps 1-6 of clause 6.3.7.1.3.2 applies with the following modification:

- In Step 5: A spelling mistake should be introduced in the EiJobResultObject for schema validation error in DUT.

6.3.7.2.3.3 Expected result

The test is considered passed if the following conditions are met:

- 1) Step 3 validation has passed.
- 2) Check the HTTP response recorded in step 6 of procedure and that the return code is "400 Bad request".

6.3.7.3 Deliver EI job result - callback URI not supported (negative case)

6.3.7.2.1 Test description and applicability

This purpose of this test case is to test the deliver EI job result functionality of A1-EI Consumer as specified in A1AP [4], clause 5.3.5.2. The expected outcome is failure due to invalid callback URI.

This test case is mandatory if the DUT claims to support A1-EI service.

6.3.7.2.2 Test entrance criteria

The test entrance criteria in clause 6.3.7.1.2 applies.

6.3.7.2.3 Test methodology

6.3.7.2.3.1 Initial conditions

The initial conditions in clause 6.3.7.1.3.1 applies.

6.3.7.2.3.2 Procedure

Steps 1-6 of clause 6.3.7.1.3.2 applies with the following modification:

- In Step 5. A spelling mistake should be introduced in the used callback URI making it different from the provided jobResultUri and not identical to any callback URI supported in the DUT.

6.3.7.2.3.3 Expected result

The test is considered passed if the following conditions are met:

- 1) Step 3 validation has passed.
- 2) Check the HTTP response recorded in step 6 of procedure and that the return code is "400 Bad request".

7 Test cases for interoperability between Non-Real Time RIC and Near-Real Time RIC

7.1 General

7.1.1 System under test requirements

The system under test is a Non-RT RIC and a Near-RT RIC that acts as DUTs referred to Non-RTR-DUT and Near-RTR-DUT in the test scenarios of this clause. The requirements on the DUTs for these tests are that they can handle the procedures defined for a Non-RT RIC and Near-RT RIC and the purpose of the test scenarios is to validate that Non-RT RIC and Near-RT RIC are interoperable. The Non-RT RIC and the Near-RT RIC can be functions that are under development, or they can be finalized commercial products, that have passed the conformance test scenarios defined in clauses 5 and 6.

In addition to the basic conformance requirements, and the DUT requirements in clause 5 and 6, the following is required for Non-RT RIC and Near-RT RIC when they act as DUTs in this test system:

- It shall be possible for the DUTs to initiate the A1 procedures in a controlled manner and to generate relevant response for each test case. For this some additional functionality is required, this can be implemented as installable test applications or in other vendor specific way.
- To enable validation of A1 procedure messages captured between the DUTs an ability to disable encryption is required.
- A1 policy types and EI types used in test scenarios are agreed and configured in the DUTs in advance.

NOTE: The present document does not require usage of any specific policy type or EI type since specific behaviour is not validated, only the application protocol aspects and procedures are validated for interoperability testing of the A1 services.

7.1.2 Test tools and simulators capabilities

A TAP interface and protocol analyzer is required for recording and validating A1 procedure message sequences and contents and issuing of verdicts related to the procedures in the test cases and thereby enabling determination of the DUTs level of interoperability.

SMO framework with O1 functionality may be required to bring the DUTs into operational state.

Depending on how the functionality in the DUTs required for the test scenarios is implemented, there may be a need to connect Near-RTR-DUT to E2 Nodes connected to RUs, UEs and Core Network. These can be replaced by appropriate simulators.

NOTE: In the future, when then open APIs for rApps and xApps are fully specified, test apps can be provided as part of the test tools and used to trigger the request messages and generate the content of the response messages for the test scenarios.

7.2 Interoperability test cases for A1-P

7.2.1 Query policy type test scenarios

7.2.1.1 Query all policy type identifiers (positive case)

7.2.1.1.1 Test description and applicability

The purpose of this test case is to test the query policy types functionality of A1-P Consumer and A1-P Producer as specified in A1AP [4], clause 5.2.3.2. The expected outcome is successful Query all policy type identifiers operation.

This test case is conditionally mandatory if the DUTs claim to support A1-P service and the Query all policy type identifiers procedure.

7.2.1.1.2 Test entrance criteria

- 1) The Non-RTR-DUT has functionality to initiate A1-P Query all policy type identifiers procedure.
- 2) A set of policy types (at least two) are supported in the Near-RTR-DUT.
- 3) The supported policyTypeIds are known to the "protocol analyzer".

7.2.1.1.3 Test methodology

7.2.1.1.3.1 Initial conditions

- 1) The Near-RTR-DUT has A1-P Producer service ready and available to receive HTTP requests from the Non-RTR-DUT.

- 2) The Near-RTR-DUT supports the three different test configurations listed below to test complete functionality:
 - a) Near-RTR-DUT has no policy types available.
 - b) Near-RTR-DUT has a single policy type available.
 - c) Near-RTR-DUT has two or more policy types available.

7.2.1.1.3.2 Procedure

- Step 0. Enable protocol analyzer to capture and decode the A1 signalling between the DUTs via the tap/TAP interface.
- Step 1. Initiate appropriate action in the Non-RTR-DUT to initiate Query all policy type identifiers request.
- Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.
- Step 3. Repeat Step 1 and Step 2 for the three test configurations.

7.2.1.1.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met for all three test configurations listed in clause 7.2.1.1.3.1.

For the request message:

- a) The URI conforms to the format specified in A1AP [4], clause 6.2.3.
- b) The HTTP request is a GET operation.
- c) The HTTP request message body is empty.

For the response message:

- 1) The return code is "200 OK".
- 2) Response message body is validated depending on the three test configurations used:
 - a) If the Near-RTR-DUT has no policy types available, message body is an empty array.
 - b) If the Near-RTR-DUT has single policy type available, message body contains an array with one policyType Id.
 - c) If the Near-RTR-DUT has two or more policy types available, message body contains an array with all the policyType Id's available in Near-RTR-DUT.

7.2.1.2 Query single policy type (positive)

7.2.1.2.1 Test description and applicability

The purpose of this test case is to test the query policy types functionality of A1-P Consumer and A1-P Producer as specified in A1AP [4], clause 5.2.3.3. The expected outcome is successful Query single policy type operation.

This test case is conditionally mandatory if the DUTs claim to support A1-P service and the Query single policy type procedure.

7.2.1.2.2 Test entrance criteria

- 1) The Non-RTR-DUT has functionality to initiate A1-P Query single policy procedure.
- 2) A known set of policy types are supported in the Near-RTR-DUT.
- 3) The supported policyType Id's are known to the protocol analyzer.

7.2.1.2.3 Test methodology

7.2.1.2.3.1 Initial conditions

- 1) The Near-RTR-DUT has A1-P Producer service ready and available to receive HTTP requests from the Non-RTR-DUT.

7.2.1.2.3.2 Procedure

- Step 0. Enable protocol analyzer to capture and decode the A1 signalling between the DUTs via the tap interface.
- Step 1. Initiate appropriate action in the Non-RTR-DUT to initiate Query single policy request with a known policyTypeId.
- Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.

7.2.1.2.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met.

For the request message:

- a) The URI conforms to the format specified in A1AP [4], clause 6.2.3.
- b) The HTTP request is a GET operation.
- c) The policyTypeId in the URI match the policy type being queried.
- d) The HTTP request message body is empty.

For the response message:

- 1) The return code is "200 OK".
- 2) Response message body content contains a PolicyTypeObject representing the read policy type.

7.2.2 Create policy type test scenario

7.2.2.1 Create single policy (positive)

7.2.2.1.1 Test description and applicability

This purpose of this test case is to test the create policy functionality of A1-P Consumer and A1-P Producer as specified in A1AP [4], clause 5.2.4.3. The expected outcome is successful creation of the policy.

This test case is mandatory if the DUTs claim to support A1-P service.

7.2.2.1.2 Test entrance criteria

- 1) The Non-RTR-DUT has functionality to initiate A1-P Create single policy procedure.
- 2) The DUTs have agreed on a policy type to use for this test.
- 3) The policyTypeId and the JSON schemas of the policy type used for this test are available and used in Non-RTR-DUT to formulate the Create single policy request, and in Near-RTR-DUT to validate and handle the request.
- 4) The agreed policyTypeId is known to the protocol analyzer.

7.2.2.1.3 Test methodology

7.2.2.1.3.1 Initial conditions

- 1) The Near-RTR-DUT has A1-P Producer service ready and available to receive HTTP requests from the Non-RTR-DUT.
- 2) No policy exists in the Near-RTR-DUT for the policy type with the same policyId that will be used by the Non-RTR-DUT.

7.2.2.1.3.2 Procedure

- Step 0. Enable protocol analyzer to capture and decode the A1 signalling between the DUTs via the tap interface.
- Step 1. Initiate appropriate action in the Non-RTR-DUT to initiate Create single policy request for the agreed policy type.
- Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.

7.2.2.1.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

For the request message:

- a) The URI conforms to the format specified in A1AP [4], clause 6.2.3.
- b) The HTTP request is a PUT operation.
- c) The policyTypeId is as agreed to be used.
- d) The HTTP request message body contains the PolicyObject of the policy to be created and the PolicyObject conforms to the schema.

NOTE: Presence or validation of optional query parameter for notificationDestination is not used to determine validation on this test.

For the response message:

- 1) The return code is "201 Created".
- 2) Response message body content contains a PolicyObject representing the created policy.
- 3) The location header is present and carries the URI of the new policy.

7.2.3 Query policy test scenarios

7.2.3.1 Query all policy identifiers

7.2.3.1.1 Test description and applicability

The purpose of this test case is to test the query all policies functionality of A1-P Consumer and A1-P Producer as specified in A1AP [4], clause 5.2.4.2. The expected outcome is successful retrieval of policy identifiers.

This test case is conditionally mandatory if the DUTs claim to support A1-P service and the Query all policy identifiers procedure.

7.2.3.1.2 Test entrance criteria

- 1) The Non-RTR-DUT has functionality to initiate A1-P Query all policy identifiers procedure.

- 2) The DUTs have agreed on a policy type to use for this test.
- 3) A set of policies (at least two) for the agreed policy type exist in the Near-RTR-DUT.
- 4) The agreed policyTypeId is known to the protocol analyzer.

7.2.3.1.3 Test methodology

7.2.3.1.3.1 Initial conditions

- 1) The Near-RTR-DUT has A1-P Producer service ready and available to receive HTTP requests from the Non-RTR-DUT.
- 2) The Near-RTR-DUT supports three different test configurations listed below to test complete functionality:
 - a) Near-RTR-DUT has no policies available for the agreed policy type.
 - b) Near-RTR-DUT has a single policy available for the agreed policy type.
 - c) Near-RTR-DUT has two or more policies available for the agreed policy type.

7.2.3.1.3.2 Procedure

- Step 0. Enable protocol analyzer to capture and decode the A1 signalling between the DUTs via the tap interface.
- Step 1. Initiate appropriate action in the Non-RTR-DUT to initiate Query all policy identifiers request.
- Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.
- Step 3. Repeat Step 1 and Step 2 for the three test configurations.

7.2.3.1.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met for all three test configurations listed in clause 7.2.3.1.3.1.

For the request messages:

- a) The URI conforms to the format specified in A1AP [4], clause 6.2.3.
- b) The HTTP request is a GET operation.
- c) The message body is empty.

For the response messages:

- 1) The return code is "200 OK".
- 2) Response message body is validated depending on the three test configurations used:
 - a) If the Near-RTR-DUT has no policies available, message body is an empty array.
 - b) If the Near-RTR-DUT has single policy available, message body contains an array with one policyId available with the given policy type.
 - c) If the Near-RTR-DUT has two or more policies available, message body contains an array of policyIds of all the available policies of the given policy type.

7.2.3.2 Query single policy

7.2.3.2.1 Test description and applicability

The purpose of this test case is to test the query single policy functionality of A1-P Consumer and A1-P Producer as specified in A1AP [4], clause 5.2.4.5. The expected outcome is successful retrieval of policy object.

This test case is conditionally mandatory if the DUTs claim to support A1-P service and the Query single policy procedure.

7.2.3.2.2 Test entrance criteria

- 1) The Non-RTR-DUT has functionality to initiate A1-P Query single policy procedure.
- 2) A policy exists in the Near-RTR-DUT and the policyTypeId and policyId are known to Non-RTR-DUT.
- 3) The policyTypeId and policyId are known to the protocol analyzer.

7.2.3.2.3 Test methodology

7.2.3.2.3.1 Initial conditions

- 1) The Near-RTR-DUT has A1-P Producer service ready and available to receive HTTP requests from the Non-RTR-DUT.

7.2.3.2.3.2 Procedure

- Step 0. Enable protocol analyzer to capture and decode the A1 signalling between the DUTs via the tap interface.
- Step 1. Initiate appropriate action in the Non-RTR-DUT to initiate Query single policy request.
- Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.

7.2.3.2.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met.

For the request message:

- a) The URI conforms to the format specified in A1AP [4], clause 6.2.3.
- b) The HTTP request is a GET operation.
- c) The policyTypeId and policyId in the URI match the policy being queried.
- d) The message body is empty.

For the response messages:

- 1) The return code is "200 OK".
- 2) Response message body content contains a PolicyObject representing the read policy.

7.2.3.3 Query policy status (positive case)

7.2.3.3.1 Test description and applicability

The purpose of this test case is to test the query policy status functionality of A1-P Consumer and A1-P Producer as specified in A1AP [4], clause 5.2.4.7. The expected outcome is successful retrieval of policy status object.

This test case is conditionally mandatory if the DUTs claim to support A1-P service and the Query policy status procedure.

7.2.3.3.2 Test entrance criteria

- 1) The Non-RTR-DUT has functionality to initiate A1-P Query policy status procedure.
- 2) A policy exists in the Near-RTR-DUT and the policyTypeId and policyId are known to the Non-RTR-DUT.
- 3) The policyTypeId and policyId are known to the protocol analyzer.

7.2.3.3.3 Test methodology

7.2.3.3.3.1 Initial conditions

- 1) The Near-RTR-DUT has A1-P Producer service ready and available to receive HTTP requests from the Non-RTR-DUT.

7.2.3.3.3.2 Procedure

- Step 0. Enable protocol analyzer to capture and decode the A1 signalling between the DUTs via the tap interface.
- Step 1. Initiate appropriate action in the Non-RTR-DUT to initiate Query policy status request.
- Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.

7.2.3.3.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met.

For the request message:

- a) The URI conforms to the format specified in A1AP [4], clause 6.2.3.
- b) The HTTP request is a GET operation.
- c) The policyTypeId and policyId in the URI match the policy being queried.
- d) The message body is empty.

For the response messages:

- 1) The return code is "200 OK".
- 2) Response message body content contains a PolicyStatusObject representing the status of the policy.

7.2.4 Update policy test scenarios

7.2.4.1 Update single policy (positive case)

7.2.4.1.1 Test description and applicability

The purpose of this test case is to test the update policy functionality of A1-P Consumer and A1-P Producer as specified in A1AP [4], clause 5.2.4.4. The expected outcome is successful update of the policy.

This test case is conditionally mandatory if the DUTs claim to support A1-P service and the Update single policy procedure.

7.2.4.1.2 Test entrance criteria

- 1) The Non-RTR-DUT has functionality to initiate A1-P Update single policy procedure.
- 2) A policy exists in the Near-RTR-DUT and the policyTypeId and policyId are known to the Non-RTR-DUT.
- 3) The policyTypeId and the JSON schemas of the policy type used for this test are available and used in Non-RTR-DUT to formulate the Update single policy request, and in Near-RTR-DUT to validate and handle the request.
- 4) The policyTypeId and the policyId are known to the protocol analyzer.

7.2.4.1.3 Test methodology

7.2.4.1.3.1 Initial conditions

- 1) The Near-RTR-DUT has A1-P Producer service ready and available to receive HTTP requests from the Non-RTR-DUT.

7.2.4.1.3.2 Procedure

Step 0. Enable protocol analyzer to capture and decode the A1 signalling between the DUTs via the tap interface.

Step 1. Initiate appropriate action in the Non-RTR-DUT to initiate Update single policy request.

Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.

7.2.4.1.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met.

For the request messages:

- a) The URI conforms to the format specified in A1AP [4], clause 6.2.3.
- b) The HTTP request is a PUT operation.
- c) The policyTypeId and policyId in the URI match the policy being updated.
- d) The HTTP request message body contains the PolicyObject of the policy to be updated and the PolicyObject conforms to the schema of the policy type.

For the response messages:

- 1) The return code is "200 OK".
- 2) Response message body content is verified to match the PolicyObject sent in Step 1.

7.2.5 Delete policy test scenarios

7.2.5.1 Delete single policy (positive case)

7.2.5.1.1 Test description and applicability

The purpose of this test case is to test the delete policy functionality of A1-P Consumer and A1-P Producer as specified in A1AP [4], clause 5.2.4.6. The expected outcome is successful deletion of the policy.

This test case is mandatory if the DUTs claim to support A1-P service.

7.2.5.1.2 Test entrance criteria

- 1) The Non-RTR-DUT has functionality to initiate A1-P Delete single policy procedure.
- 2) A policy exists in the Near-RTR-DUT and the policyTypeId and policyId are known to the Non-RTR-DUT.
- 3) The policyTypeId and the policyId are known to the protocol analyzer.

7.2.5.1.3 Test methodology

7.2.5.1.3.1 Initial conditions

- 1) The Near-RTR-DUT has A1-P Producer service ready and available to receive HTTP requests from the Non-RTR-DUT.

7.2.5.1.3.2 Procedure

Step 0. Enable protocol analyzer to capture and decode the A1 signalling between the DUTs via the tap interface.

Step 1. Initiate appropriate action in the Non-RTR-DUT to initiate Delete single policy request.

Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.

7.2.5.1.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met.

For the request messages:

- a) The URI conforms to the format specified in A1AP [4], clause 6.2.3.
- b) The HTTP request is a DELETE operation.
- c) The policyTypeId and policyId in the URI match the policy being deleted.
- d) The message body is empty.

For the response messages:

- 1) The return code is "204 No content".
- 2) Response message body is empty.

7.2.6 Feedback policy test scenarios

7.2.6.1 Feedback policy (positive case)

7.2.6.1.1 Test description and applicability

The purpose of this test case is to test the policy feedback functionality of A1-P Consumer and A1-P Producer as specified in A1AP [4], clause 5.2.4.8. The expected outcome is successful policy status request and reception.

This test case is optional if the DUTs claim to support A1-P service and supports the Policy feedback procedure.

7.2.6.1.2 Test entrance criteria

The test entrance criteria for Create single policy specified in clause 7.2.2.1.2 applies. And in addition:

- 1) Non-RTR-DUT has the functionality to request and receive policy status updates.
- 2) Near-RTR-DUT has the functionality to provide policy status updates.

- 3) Non-RTR-DUT and Near-RTR-DUT have agreed on a how to requests and trigger the Feedback policy notification based on supported policy types and functionality in the Near-RTR-DUT.

7.2.6.1.3 Test methodology

7.2.6.1.3.1 Initial conditions

The initial conditions for Create single policy specified in clause 7.2.2.1.3.1 applies. And in addition:

- 1) The Near-RTR-DUT supports generation of a PolicyStatusObject based on the policy status schema of the agreed policy type used for this test.
- 2) The Non-RTR-DUT has HTTP server ready and available to receive HTTP requests from the Near-RTR-DUT for policy status updates.

NOTE: The Non-RTR-DUT is expected to provide callback URI (notificationDestination) during the Create single policy procedure (see A1AP [4], clause 5.2.4.3) for which it can relate received policy status updates to the policyId that was generated when creating the policy for which policy status updates is requested.

7.2.6.1.3.2 Procedure

- Step 1. Same as for Create single policy, see clause 7.2.2.1.3.2, including the notificationDestination query parameter.
- Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.
- Step 3. Initiate appropriate action in the Near-RTR-DUT to initiate Feedback policy request for the created policy.
- Step 4. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.

7.2.6.1.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met for the Create policy operation.

For the request message:

- a)-e) As defined for Create single policy in clause 7.2.2.1.3.3.
- f) The notificationDestination query parameter is included.

For the response message:

- 1) The return code is "201 Created".

Check the HTTP messages recorded in Step 4 of procedure.

The test is considered passed if the following conditions are met for the Feedback policy operation.

For the request message:

- a) The URI is the notificationDestination provided in the Create policy operation.
- b) The HTTP request is a POST operation.
- c) The HTTP request message body contains the PolicyStatusObject of the policy for which status is updated and the PolicyStatusObject conforms to the status schema.

For the response message:

- 1) The return code is "204 No content".

7.3 Interoperability test cases for A1-EI

7.3.1 Query EI types test scenarios

7.3.1.1 Query EI type identifiers (positive case)

7.3.1.1.1 Test description and applicability

The purpose of this test case is to test query EI type identifiers functionality of A1-EI Producer and A1-EI Consumer as specified in A1AP [4], clause 5.3.3.2. The expected outcome is successful retrieval of EI type identifiers.

This testcase is conditionally mandatory if the DUTs claim to support A1-EI service and Query EI type identifiers procedure.

7.3.1.1.2 Test entrance criteria

- 1) The Near-RTR-DUT has functionality to initiate A1-EI Query EI type identifiers procedure.
- 2) The Non-RTR-DUT supports a known set of EI types (two or more) used for this test.
- 3) The supported eiTypeId's are known to the protocol analyzer

7.3.1.1.3 Test methodology

7.3.1.1.3.1 Initial conditions

- 1) The Non-RTR-DUT has A1-EI Producer service ready and available to receive HTTP requests from the Near-RTR-DUT.
- 2) The Non-RTR-DUT supports the three different test configurations listed below to test complete functionality:
 - a) Non-RTR-DUT has no EI type available.
 - b) Non-RTR-DUT has a single EI type available.
 - c) Non-RTR-DUT has two or more EI types available.

7.3.1.1.3.2 Procedure

- Step 0. Enable protocol analyzer to capture and decode the A1 signalling between the DUTs via the tap interface.
- Step 1. Initiate appropriate action in the Near-RTR-DUT to initiate Query EI type identifiers request.
- Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.
- Step 3. Repeat Step 1 and Step 2 for the three test configurations.

7.3.1.1.3.3 Expected result

The test is considered passed if the following conditions are met for all three test configurations listed in clause 7.3.1.1.3.1.

For the request message:

- a) The URI conforms to the format specified in A1AP [4], clause 6.2.3.
- b) The HTTP request is a GET operation.
- c) The HTTP request message body is empty.

For the response message:

- 1) The return code is "200 OK".
- 2) Response message body is validated depending on the test configuration used:
 - a) If the Non-RTR-DUT has no EI type available, message body is an empty array.
 - b) If the Non-RTR-DUT has single EI type available, message body contains an array with one eiTypeId.
 - c) If the Non-RTR-DUT has two or more EI jobs available, message body contains an array of all the eiTypeIds available in DUT.

7.3.1.2 Query EI type (positive case)

7.3.1.2.1 Test description and applicability

The purpose of this test case is to test query EI type functionality of A1-EI Producer and A1-EI Consumer as specified in A1AP [4], clause 5.3.3.3. The expected outcome is successful retrieval of EI type object.

This test case is conditionally mandatory if the DUTs claim to support A1-EI service and the Query EI type procedure.

7.3.1.2.2 Test entrance criteria

- 1) The Near-RTR-DUT has the functionality to initiate A1-EI Query EI type procedure.
- 2) A known set of EI types are supported in the Non-RTR-DUT.
- 3) The supported eiTypeId's are known to the protocol analyzer.

7.3.1.2.3 Test methodology

7.3.1.2.3.1 Initial conditions

- 1) The Non-RTR-DUT has A1-EI Producer service ready and available to receive HTTP requests from the Near-RTR-DUT.

7.3.1.2.3.2 Procedure

- Step 0. Enable protocol analyzer to capture and decode the A1 signalling between the DUTs via the tap interface.
- Step 1. Initiate appropriate action in the Near-RTR-DUT to initiate Query EI type request with a known eiTypeId.
- Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.

7.3.1.2.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met.

For the request message:

- a) The URI conforms to the format specified in A1AP [4], clause 6.3.3.
- b) The HTTP request is a GET operation.
- c) The eTypeId in the URI match the EI type being queried.
- d) The HTTP request message body is empty.

For the response message:

- 1) The return code is "200 OK".

- 2) Response message body content contains an EiTypeObject representing the read EI job.

7.3.2 Create EI job test scenario

7.3.2.1 Create single EI job (positive case)

7.3.2.1.1 Test description and applicability

This purpose of this test case is to test the create EI job functionality of A1-EI Producer and A1-EI Consumer as specified in A1AP [4], clause 5.3.4.3. The expected outcome is successful creation of the EI job.

This testcase is mandatory if the DUTs claim to support A1-EI service.

7.3.2.1.2 Test entrance criteria

- 1) The Near-RTR-DUT has functionality to initiate A1-EI Create single EI job procedure.
- 2) The DUTs have agreed on an EI type to use for this test.
- 3) The eiTypeId and the JSON schemas of the EI type used for this test are available and used in Near-RTR-DUT to formulate the Create EI job request, and in Non-RTR-DUT to validate and handle the request.
- 4) The agreed eiTypeId is known to the protocol analyzer.

7.3.2.1.3 Test methodology

7.3.2.1.3.1 Initial conditions

- 1) The Non-RTR-DUT has A1-EI Producer service ready and available to receive HTTP requests from the Near-RTR-DUT.
- 2) No EI job exists in the Non-RTR-DUT for the EI job type with the same eiJobId that will be used by the Near-RTR-DUT.

7.3.2.1.3.2 Procedure

Step 0. Enable protocol analyzer to capture and decode the A1 signalling between the DUTs via the tap interface.

Step 1. Initiate appropriate action in the Near-RTR-DUT to initiate Create EI job request for the agreed EI job type.

Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.

7.3.2.1.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met.

For the request message:

- a) The URI conforms to the format specified in A1AP [4], clause 6.3.3.
- b) The HTTP request is a PUT operation.
- c) The eiTypeId is as agreed to be used.
- d) The HTTP request message body contains the EiJobObject of the EI job to be created and the EiJobObject conforms to the schema.

For the response message:

- 1) The return code is "201 Created".

- 2) Response message body content contains an EiJobObject representing the created EI job.
- 3) The location header is present and carries the URI of the new EI job.

7.3.3 Query EI jobs test scenarios

7.3.3.1 Query EI job identifiers for a single EI type (positive case)

7.3.3.1.1 Test description and applicability

This purpose of this test case is to test query EI job identifiers functionality of A1-EI Producer and A1-EI Consumer as specified in A1AP [4], clause 5.3.4.2. The expected outcome is successful retrieval of all EI job identifiers of a given EI type.

This testcase is conditionally mandatory if the DUTs claim to support A1-EI service and Query EI job identifiers procedure.

7.3.3.1.2 Test entrance criteria

- 1) The Near-RTR-DUT has functionality to initiate A1-EI Query EI job identifiers procedure.
- 2) The DUTs have agreed on an EI type to use for this test.
- 3) A set of EI jobs (at least two) for the agreed EI type exist in the Non-RT-DUT.
- 4) The agreed eiTypeId is known to the protocol analyzer.

7.3.3.1.3 Test methodology

7.3.3.1.3.1 Initial conditions

- 1) The Non-RTR-DUT has A1-EI Producer service ready and available to receive HTTP requests from the Near-RTR-DUT.
- 2) The Non-RTR-DUT supports three different test configurations listed below to test complete functionality:
 - a) Non-RTR-DUT has no EI jobs available for the agreed EI type.
 - b) Non-RTR-DUT has an EI job available for the agreed EI type.
 - c) Non-RTR-DUT has two or more EI jobs available for the agreed EI type.

7.3.3.1.3.2 Procedure

- Step 0. Enable protocol analyzer to capture and decode the A1 signalling between the DUTs via the tap interface.
- Step 1. Initiate appropriate action in the Near-RTR-DUT to initiate Query all EI job identifiers.
- Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.
- Step 3. Repeat Step 1 and Step 2 for the three test configurations.

7.3.3.1.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met for all three test configurations listed in clause 7.3.3.1.3.1.

For the request messages:

- a) The URI conforms to the format specified in A1AP [4], clause 6.3.3 with the agreed eiTypeId as query parameter.

- b) The HTTP request is a GET operation.
- c) The message body is empty.
- d) A query parameter with an eiTypeId is included.

For the response messages:

- 1) The return code is "200 OK".
- 2) Response message body is validated depending on the three test configurations used:
 - a) If the Non-RTR-DUT has no EI job available, message body is an empty array.
 - b) If the Non-RTR-DUT has single EI job available, message body contains an array with one eiJobId available with the given EI type.
 - c) If the Non-RTR-DUT has two or more EI jobs available, message body contains an array of eiJobIds of all the available EI jobs of the given EI type.

7.3.3.2 Query EI job identifiers for all EI types (positive case)

7.3.3.2.1 Test description and applicability

This purpose of this test case is to test query EI job identifiers functionality of A1-EI Producer and A1-EI Consumer as specified in A1AP [4], clause 5.3.4.2. The expected outcome is successful retrieval of all EI job identifiers of all EI types.

This testcase is conditional mandatory if the DUTs claim to support A1-EI service and Query EI job identifiers procedure.

7.3.3.2.2 Test entrance criteria

- 1) The Near-RTR-DUT has functionality to initiate A1-EI Query EI job identifiers procedure.
- 2) The DUTs supports a known set of EI types (two or more) used for this test.
- 3) The agreed eiTypeIds are known to the protocol analyzer.

7.3.3.2.3 Test methodology

7.3.3.2.3.1 Initial conditions

- 1) The Non-RTR-DUT has A1-EI Producer service ready and available to receive HTTP requests from the Near-RTR-DUT.
- 2) The Non-RTR-DUT supports the two different test configurations listed below to test complete functionality:
 - a) Non-RTR-DUT has no EI job available for any of the EI types supported.
 - b) Non-RTR-DUT has one or more EI jobs available for each EI type supported.

7.3.3.2.3.2 Procedure

- Step 0. Enable protocol analyzer to capture and decode the A1 signalling between the DUTs via the tap interface.
- Step 1. Initiate appropriate action in the Near-RTR-DUT to initiate Query EI job identifiers request.
- Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.

7.3.3.2.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met for all two test configurations listed in clause 7.3.3.2.3.1.

For the request messages:

- a) The URI conforms to the format specified in A1AP [4], clause 6.3.3 without eiTypeId included as query parameter.
- b) The HTTP request is a GET operation.
- c) No query parameter with an eiTypeId is included.
- d) The message body is empty.

For the response messages:

- 1) The return code is "200 OK".
- 2) Response message body is validated depending on the two test configurations used:
 - a) If the Non-RTR-DUT has no EI job available, message body is an empty array.
 - b) If the Non-RTR-DUT has one or more EI jobs available, message body contains an array of eiJobIds of all the available EI jobs of all EI types.

7.3.3.3 Query EI job (positive case)

7.3.3.3.1 Test description and applicability

The purpose of this test case is to test query EI job functionality of A1-EI Producer and A1-EI Consumer as specified in A1AP [4], clause 5.3.4.5. The expected outcome is successful retrieval of EI job for the given eiJobId.

This test case is conditionally mandatory if the DUTs claim to support A1-EI service and the Query EI job procedure.

7.3.3.3.2 Test entrance criteria

- 1) The Near-RTR-DUT has functionality to initiate A1-EI Query EI job procedure.
- 2) An EI job exists in the Non-RTR-DUT and the eiJobId is known to the Near-RTR-DUT.
- 3) The eiJobId is known to the protocol analyzer.

7.3.3.3.3 Test methodology

7.3.3.3.3.1 Initial conditions

- 1) The Non-RTR-DUT has A1-EI Producer service ready and available to receive HTTP requests from the Near-RTR-DUT.

7.3.3.3.3.2 Procedure

- Step 0. Enable protocol analyzer to capture and decode the A1 signalling between the DUTs via the tap interface.
- Step 1. Initiate appropriate action in the Near-RTR-DUT to initiate Query EI job request.
- Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.

7.3.3.3.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met:

- a) The URI conforms to the format specified in A1AP [4], clause 6.3.3.

- b) The HTTP request is a GET operation.
- c) The eiJobId in the URI match the EI job being queried.
- d) The HTTP request message body is empty.

For the response messages:

- 1) The return code is "200 OK".
- 2) Response message body content contains an EiJobObject representing the EI job.

7.3.4 Update EI job test scenarios

7.3.4.1 Update EI job (positive case)

7.3.4.1.1 Test description and applicability

The purpose of this test case is to test update EI job functionality of A1-EI Producer and A1-EI Consumer as specified in A1AP [4], clause 5.3.4.4. The expected outcome is successful update of the EI job.

This test case is conditionally mandatory if the DUTs claim to support A1-EI service and Update EI job procedure.

7.3.4.1.2 Test entrance criteria

- 1) The Non-RTR-DUT has functionality to initiate A1-EI Update EI job procedure.
- 2) A EI job exists in the Non-RTR-DUT and the eiTypeID and eiJobID are known to the Near-RTR-DUT.
- 3) The eiTypeID and the JSON schemas of the EI type used for this test are available and used in Near-RTR-DUT to formulate the Update EI job request, and in Non-RTR-DUT to validate and handle the request.
- 4) The eiTypeID and the eiTypeID are known to the protocol analyzer.

7.3.4.1.3 Test methodology

7.3.4.1.3.1 Initial conditions

- 1) The Non-RTR-DUT has A1-EI Producer service ready and available to receive HTTP requests from the Near-RTR-DUT.

7.3.4.1.3.2 Procedure

- Step 0. Enable protocol analyzer to capture and decode the A1 signalling between the DUTs via the tap interface.
- Step 1. Initiate appropriate action in the Near-RTR-DUT to initiate Update EI job request.
- Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.

7.3.4.1.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met.

For the request messages:

- a) The URI conforms to the format specified in A1AP [4], clause 6.3.3.
- b) The HTTP request is a PUT operation.
- c) The eiJobId in the URI match the EI job to be updated.

- d) The HTTP request message body contains the EiJobObject of the EI job being updated and the EiJobObject conforms to the schema of the EI type.

For the response messages:

- 1) The return code is "200 OK".
- 2) Response message body content is verified to match the EiJobObject sent in Step 1.

7.3.5 Delete EI job test scenarios

7.3.5.1 Delete EI job (positive case)

7.3.5.1.1 Test description and applicability

The purpose of this test case is to test the delete EI job functionality of A1-EI Producer and A1-EI Consumer as specified in A1AP [4], clause 5.3.4.6. The expected outcome is successful validation of the Delete EI job request

This test case is mandatory if the DUTs claim to support A1-EI service.

7.3.5.1.2 Test entrance criteria

- 1) The Non-RTR-DUT has functionality to initiate A1-EI Delete EI job procedure.
- 2) A EI job exists in the Non-RTR-DUT and the eiJobId is known to the Near-RTR-DUT.
- 3) The eiJobId is known to the protocol analyzer.

7.3.5.1.3 Test methodology

7.3.5.1.3.1 Initial conditions

- 1) The Non-RTR-DUT has A1-EI Producer service ready and available to receive HTTP requests from the Near-RTR-DUT.

7.3.5.1.3.2 Procedure

- Step 0. Enable protocol analyzer to capture and decode the A1 signalling between the DUTs via the tap interface.
- Step 1. Initiate appropriate action in the Near-RTR-DUT to initiate Delete EI job request.
- Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.

7.3.5.1.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met.

For the request messages:

- a) The URI conforms to the format specified in A1AP [4], clause 6.3.3.
- b) The HTTP request is a DELETE operation.
- c) The eiJobId in the URI match the EI job being deleted.
- d) The message body is empty.

For the response messages:

- 1) The return code is "204 No content".

- 2) Response message body is empty.

7.3.6 Status of EI jobs test scenarios

7.3.6.1 Query EI job status (positive case)

7.3.6.1.1 Test description and applicability

The purpose of this test case is to test the query EI job status functionality of A1-EI Producer and A1-EI Consumer as specified in A1AP [4], clause 5.3.4.7. The expected outcome is successful retrieval of EI job status object.

This test case is conditionally mandatory if the DUTs claim to support A1-EI service and the Query EI job status procedure.

7.3.6.1.2 Test entrance criteria

- 1) The Near-RTR-DUT has functionality to initiate Query EI job status procedure.
- 2) An EI job exists in the Non-RTR-DUT and the eiJobId is known to the Near-RTR-DUT.
- 3) The agreed eiTypeId is known to the protocol analyzer.

7.3.6.1.3 Test methodology

7.3.6.1.3.1 Initial conditions

- 1) The Non-RTR-DUT has A1-EI Producer service ready and available to receive HTTP requests from the Near-RTR-DUT.

7.3.6.1.3.2 Procedure

Step 0. Enable protocol analyzer to capture and decode the A1 signalling between the DUTs via the tap interface.

Step 1. Initiate appropriate action in the Near-RTR-DUT to initiate Query EI job status request.

Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.

7.3.6.1.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met.

For the request messages:

- a) The URI conforms to the format specified in A1AP [4], clause 6.3.3.
- b) The HTTP request is a GET operation.
- c) The eiJobId in the URI match the EI job being queried.
- d) The HTTP request message body is empty.

For the response messages:

- 1) The return code is "200 OK".
- 2) Response message body content contains an EiJobStatusObject representing the status of the EI job.

7.3.6.2 Notify EI job status (positive case)

7.3.6.2.1 Test description and applicability

This purpose of this test case is to test the notify EI job status functionality of A1-EI Producer and AI-EI Consumer as specified in A1AP [4], clause 5.3.4.8. The expected outcome is successful request and delivery of EI job status notification.

This test case is optional if the DUTs claim to support A1-EI service and supports the Notify EI job status procedure.

7.3.6.2.2 Test entrance criteria

The test entrance criteria for Create EI job specified in clause 7.3.2.1.2 applies. And in addition:

- 1) Near-RTR-DUT has the functionality to request and receive EI job status notification.
- 2) Non-RTR-DUT has the functionality to provide EI job status notification.
- 3) Non-RTR-DUT and Near-RTR-DUT have agreed on a how to request and trigger the EI job status notification based on supported EI types and functionality in the Non-RTR-DUT.

7.3.6.2.3 Test methodology

7.3.6.2.3.1 Initial conditions

The initial conditions for Create EI job specified in clause 7.3.2.1.3.1 applies. And in addition:

- 1) The Near-RTR-DUT supports generation of a EiJobStatusObject based on the EI job status schema of the agreed EI type used for this test.
- 2) The Near-RTR-DUT has HTTP server ready and available to receive HTTP requests from the Non-RTR-DUT for EI job status notifications.

NOTE: The Near-RTR-DUT is expected to provide callback URI (jobStatusNotificationUri) during the Create EI job procedure (see A1AP [4], clause 5.3.4.3) for which it can relate received EI job status updates to the eiJobId that was generated when creating the EI job for which EI job status updates is requested.

7.3.6.2.3.2 Procedure

Step 1. Same as for Create EI job, see clause 7.3.2.1.3.2, including the jobStatusNotificationUri parameter in EiJobObject.

Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.

Step 3. Initiate appropriate action in the Non-RTR-DUT to initiate EI job status notification for the created EI job.

Step 4. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.

7.3.6.2.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met for the Create EI job operation.

For the request message:

- a)-d) As defined for Create EI job in clause 7.3.2.1.3.3.
- e) The jobStatusNotificationUri parameter is included in the EiJobObject.

For the response message:

- 1)-3) As defined for Create EI job in clause 7.3.2.1.3.3.

Check the HTTP messages recorded in Step 4 of procedure.

The test is considered passed if the following conditions are met for the EI job status notification operation.

For the request message:

- a) The URI is the jobStatusNotificationUri provided in the Create EI job operation.
- b) The HTTP request is a POST operation.
- c) The HTTP request message body contains the EiJobStatusObject of the EI job for which status is updated and the EiJobStatusObject conforms to the EI type status schema.

For the response message:

- 1) The return code is "204 No content".

7.3.7 Deliver EI job result test scenarios

7.3.7.1 Deliver EI job result (positive case)

7.3.7.1.1 Test description and applicability

This purpose of this test case is to test the deliver EI job result functionality of A1-EI Producer and A1-EI Consumer as specified in A1AP [4], clause 5.3.5.2. The expected outcome is successful EI job result request and delivery of EI job results.

This test case is optional if the DUTs claim to support A1-EI service.

7.3.7.1.2 Test entrance criteria

The test entrance criteria for Create EI job specified in clause 7.3.2.1.2 applies. And in addition:

- 1) Non-RTR-DUT and Near-RTR-DUT have agreed on a how to trigger the EI job results delivery based on supported EI types and functionality in the Non-RTR-DUT.

7.3.7.1.3 Test methodology

7.3.7.1.3.1 Initial conditions

The initial conditions for Create EI job specified in clause 7.3.2.1.3.1 applies. And in addition:

- 1) The Near-RTR-DUT has HTTP server ready and available to receive HTTP requests from the Non-RTR-DUT for EI job result delivery.

NOTE: The Near-RTR-DUT is expected to provide callback URI (jobResultUri) during the Create EI job procedure (see A1AP [4], clause 5.3.4.3) for which it can relate received EI job results to the eiJobId that was generated when creating the EI job for which EI job result is delivered.

7.3.7.1.3.2 Procedure

- Step 1. Same as for Create EI job, see clause 7.3.2.1.3.2, including the jobResultUri parameter in the EiJobObject.
- Step 2. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.
- Step 3. Initiate appropriate action in the Non-RTR-DUT to initiate EI job result delivery for the created EI job.
- Step 4. At the protocol analyzer the contents of the HTTP request and the HTTP response are recorded.

7.3.7.1.3.3 Expected result

Check the HTTP messages recorded in Step 2 of procedure.

The test is considered passed if the following conditions are met for the Create EI job operation.

For the request message:

- a)-d) As defined for Create EI job in clause 7.3.2.1.3.3.

For the response message

- 1)-3) As defined for Create EI job in clause 7.3.2.1.3.3.

Check the HTTP messages recorded in Step 4 of procedure.

The test is considered passed if the following conditions are met for the EI job result notification operation.

For the request message:

- a) The URI is the jobResultUri provided in the Create EI job operation.
- b) The HTTP request is a POST operation.
- c) The HTTP request message body contains the EiJobResultObject of the EI job for which result is delivered and the EiJobResultObject conforms to the EI type result schema.

For the response message:

- 1) The return code is "204 No content".

Annex A (informative): Change history

Date	Version	Information about changes
2021.11.24	01.00	A1-P conformance test cases for Non-RT RIC and Near-RT RIC
2022.04.01	02.00	Introducing A1-EI conformance test cases for Non-RT RIC and Near-RT RIC
2022.11.17	03.00	Introducing A1-P and A1-EI interoperability test cases for Non-RT RIC and Near-RT RIC. Aligning to O-RAN drafting rules.

History

Document history		
V3.0.0	January 2024	Publication