

# ETSI TS 103 315 V1.1.1 (2015-01)



TECHNICAL SPECIFICATION

## **SmartM2M; Machine-to-Machine communications (M2M); Interoperability Test Specification for ETSI M2M Primitives**

*iTeh STANDARDS PREVIEW  
(standards.iteh.ai)  
Full standards list: <https://standards.iteh.ai/catalog/standards/sist/24e56bd0-a2a3-44fa-9143-f26250bde9d6/etsi-ts-103-315-v1.1.1-2015-01>*

---

**Reference**

DTS/SmartM2M-0028

---

**Keywords**

interoperability, M2M, 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 - NAF 742 C  
Association à but non lucratif enregistrée à la  
Sous-Préfecture de Grasse (06) N° 7803/88

---

**Important notice**

The present document can be downloaded from:

<http://www.etsi.org>

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 only prevailing document is the print of the Portable Document Format (PDF) version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at

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

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

[http://portal.etsi.org/chaicor/ETSI\\_support.asp](http://portal.etsi.org/chaicor/ETSI_support.asp)

---

**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.

© European Telecommunications Standards Institute 2015.

All rights reserved.

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

**GSM®** and the GSM logo are Trade Marks registered and owned by the GSM Association.

# Contents

Intellectual Property Rights .....	6
Foreword.....	6
Modal verbs terminology.....	6
1 Scope .....	7
2 References .....	7
2.1 Normative references .....	7
2.2 Informative references.....	7
3 Abbreviations .....	8
4 Conventions.....	8
4.1 The Test Description proforma .....	8
4.2 Test Description naming convention.....	8
5 Test Description Summary .....	9
5.1 ETSI M2M tests .....	9
6 Basic Configuration.....	12
6.1 Resources request samples .....	12
6.1.1 Application .....	12
6.1.2 Container .....	12
6.1.3 ContentInstance .....	12
6.1.4 Subscription .....	13
6.1.5 AccessRight .....	13
6.1.6 Group .....	14
6.2 Resource Representations examples .....	15
6.3 aPoc Re-Targeting Procedure.....	16
6.4 M2M Settings .....	17
7 Test configurations .....	17
7.1 dIa M2M DA-GSCL (M2M_CFG_01) .....	17
7.2 mId M2M (M2M_CFG_02) .....	17
7.3 mIa M2M (M2M_CFG_03) .....	18
7.4 Multi-Hop M2M DA to NSCL (M2M_CFG_04) .....	18
7.5 Multi-Hop M2M NA to GSCL (M2M_CFG_05) .....	18
7.6 End to End M2M DA-NA (M2M_CFG_06).....	19
7.7 dIa M2M GA-GSCL (M2M_CFG_07) .....	19
7.8 Multi-Hop M2M GA to NSCL (M2M_CFG_08) .....	19
7.9 dIa M2M DA-DSCL (M2M_CFG_09) .....	20
7.10 Multi-Hop M2M DA to NSCL (M2M_CFG_10) .....	20
7.11 End to End M2M GA-NA (M2M_CFG_11).....	20
8 Test Descriptions .....	21
8.1 Point to Point configuration testing .....	21
8.1.1 SCL Mutual Registration .....	21
8.1.2 SclBase Management.....	21
8.1.2.1 SclBase Retrieve .....	21
8.1.2.2 SclBase Update .....	22
8.1.3 Application Registration .....	22
8.1.3.1 Application Resource creation .....	22
8.1.3.2 Application Retrieve .....	23
8.1.3.3 Application Update .....	23
8.1.3.4 Application Delete .....	24
8.1.4 Container Management.....	24
8.1.4.1 Container Create .....	24
8.1.4.2 Container Retrieve .....	25
8.1.4.3 Container Update .....	25

8.1.4.4	Container Delete .....	26
8.1.5	ContentInstance Management.....	26
8.1.5.1	Content Instance Create .....	26
8.1.5.2	Content Instance Create using plain text.....	27
8.1.5.3	Content Instance Retrieve .....	28
8.1.5.4	ContentInstance Content retrieve (partial addressing) .....	28
8.1.5.5	Content Instance Delete .....	29
8.1.6	Discovery Management .....	29
8.1.6.1	Discovery of all resources .....	29
8.1.6.2	Discovery with Filter Criteria .....	30
8.1.6.3	Discovery with max size .....	30
8.1.6.4	Discovery with searchPrefix .....	31
8.1.6.5	Discovery with multiple parameters.....	31
8.1.7	Subscription Management .....	32
8.1.7.1	Subscription Create .....	32
8.1.7.2	Subscription Retrieve .....	32
8.1.7.3	Subscription Update .....	33
8.1.7.4	Subscription Delete .....	33
8.1.7.5	Subscription Notify (To server-capable application) .....	34
8.1.7.6	Subscription Notify (To application container).....	35
8.1.8	Notification Channel Management .....	36
8.1.8.1	NotificationChannel Create.....	36
8.1.8.2	NotificationChannel Retrieve.....	36
8.1.8.3	NotificationChannel Delete.....	37
8.1.8.4	Long polling based notifications .....	37
8.1.9	AccessRight Management .....	38
8.1.9.1	AccessRight Resource creation.....	38
8.1.9.2	AccessRight Retrieve .....	38
8.1.9.3	AccessRight Update .....	39
8.1.9.4	AccessRight Delete .....	39
8.1.9.5	Unauthorized operation (Empty Requesting Entity) .....	40
8.1.9.6	Unauthorized operation (Bad Requesting Entity) .....	40
8.1.9.7	Unauthorized operation (Insufficient Access Rights) .....	41
8.1.10	Group Management .....	41
8.1.10.1	Group Resource creation.....	41
8.1.10.2	Group Retrieve .....	42
8.1.10.3	Group Update .....	42
8.1.10.4	Group Delete.....	43
8.1.10.5	Group Operations (Create).....	43
8.1.10.6	Group Operations (Retrieve).....	44
8.2	Multi-Hop configuration testing .....	44
8.2.1	Container Management on remote SCL.....	44
8.2.1.1	Container Create .....	44
8.2.1.2	Container Retrieve .....	45
8.2.1.3	Container Update .....	46
8.2.1.4	Container Delete .....	46
8.2.2	ContentInstance Management on remote SCL .....	47
8.2.2.1	Content Instance Create .....	47
8.2.2.2	Content Instance Create using plain text.....	48
8.2.2.3	Content Instance Retrieve .....	49
8.2.2.4	ContentInstance Content retrieve (partial addressing) .....	50
8.2.2.5	Content Instance Delete .....	50
8.2.3	Discovery Management on remote SCL.....	51
8.2.3.1	Discovery of all resources .....	51
8.2.3.2	Discovery with Filter Criteria .....	52
8.2.3.3	Discovery with max size .....	52
8.2.3.4	Discovery with searchPrefix .....	53
8.2.3.5	Discovery with multiple parameters.....	54
8.2.4	Subscription Management on remote SCL .....	54
8.2.4.1	Subscription Create .....	54
8.2.4.2	Subscription Retrieve .....	55
8.2.4.3	Subscription Update .....	56

8.2.4.4	Subscription Delete .....	57
8.2.4.5	Subscription Notify (To server capable application).....	58
8.2.4.6	Subscription Notify (To application container).....	60
8.2.5	Notification Channel Management .....	62
8.2.5.1	NotificationChannel Create.....	62
8.2.5.2	NotificationChannel Retrieve.....	63
8.2.5.3	NotificationChannel Delete.....	63
8.2.5.4	Long polling based notifications .....	64
8.2.6	AccessRight Management on remote SCL .....	66
8.2.6.1	AccessRight Resource creation.....	66
8.2.6.2	AccessRight Retrieve .....	67
8.2.6.3	AccessRight Update .....	67
8.2.6.4	AccessRight Delete .....	68
8.2.6.5	Unauthorized operation (Empty Requesting Entity) .....	69
8.2.6.6	Unauthorized operation (Bad Requesting Entity) .....	69
8.2.6.7	Unauthorized operation (Insufficient Access Rights) .....	70
8.2.7	Group Management on remote SCL .....	71
8.2.7.1	Group Resource creation.....	71
8.2.7.2	Group Retrieve .....	72
8.2.7.3	Group Update .....	72
8.2.7.4	Group Delete .....	73
8.2.7.5	Group Operations .....	74
8.2.8	Announcement .....	76
8.2.8.1	Application registration with announcement.....	76
8.2.8.2	Application delete with announcement .....	77
8.2.8.3	Application De-Announce Ressources.....	78
8.3	End to End configuration testing .....	79
8.3.1	Resource management with re-targeting.....	79
History .....		83

iTeh STANDARD PREVIEW  
 (standards.iteh.ai)  
 Full standard:  
<https://standards.iteh.ai/catalog/standards/sist/24e55bda-a2a3-44fa-9143-f26250bdc9d6/etsi-ts-103-315-v1.1.1-2015-01>

---

## Intellectual Property Rights

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

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

---

## Foreword

This Technical Specification (TS) has been produced by ETSI Technical Committee Smart Machine-to-Machine communications (SmartM2M).

---

## Modal verbs terminology

In the present document "shall", "shall not", "should", "should not", "may", "may not", "need", "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.

---

# 1 Scope

The present document specifies Interoperability Test Descriptions (TDs) for the ETSI M2M Primitives as specified in ETSI TS 102 921 [1]. The purpose of the interoperability testing is to prove that end-to-end functionality between devices such as:

- dIa interface;
- mIa interface;
- Multi-Hop configuration (dIa + mId) and (mId+mIa);
- End to End configuration (dIa + mId + mIa);

and using HTTP as underlying application layer, is as required by the standard(s) on which those devices are based.

---

## 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 reference document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at <http://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] ETSI TS 102 921: "Machine-to-Machine communications (M2M); mIa, dIa and mId interfaces".
- [2] ETSI TS 102 690: "Machine-to-Machine communications (M2M); Functional architecture".

### 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 reference 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.

Not applicable.

## 3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

aPoC	The Application Point of Contract is a URI that identifies how requests are re-targeted
App	M2M Application (DA, GA or NA)
CFG	Configuration
DA	Device Application
Device' (D')	Hosts App that communicates to a GSCL using the dIa reference point.
dIa	device application Interface
DSCL	Device SCL
EUT	Equipment Under Test
GA	Gateway Application
GSCL	Gateway SCL
MH	Multi-Hop
mIa	M2M application Interface
mId	M2M device Interface
NA	Network Application
NSCL	Network SCL
SCL	Service Capability Layer
TD	Test Description
URI	Universal Resource Identifier
XML	eXtensible Markup Language

## 4 Conventions

### 4.1 The Test Description proforma

The test descriptions are provided in proforma tables. The following different types of test operator actions are considered during the test execution:

- A **stimulus** corresponds to an event that enforces an EUT to proceed with a specific protocol action, like sending a message for instance.
- A **verify** consists of verifying that the EUT behaves according to the expected behaviour (for instance the EUT behaviour shows that it receives the expected message).
- A **configure** corresponds to an action to modify the EUT configuration.
- A **check** ensures the receipt of protocol messages on reference points, with valid content. This "check" event type corresponds to the method called 'interoperability testing with conformance check'.

### 4.2 Test Description naming convention

Table 1: TD naming convention

TD/<root>/<gr>/<nn>		
<root> = root	M2M	
<gr> = group	P2P	Point to Point testing on dIa, mId or mIa
	MH	Multi-Hop: management of remote resources
	E2E	End to End
<nn> = sequential number		01 to 99



## 5 Test Description Summary

### 5.1 ETSI M2M tests

Table 2: ETSI M2M Tests

Nb	Procedure/Resource	TD ID	TD Description
1	SCL Mutual Registration	TD_M2M_P2P_01	GSCL registers to NSCL
2	SclBase Management	TD_M2M_P2P_02	App retrieves the sclBase resource via a sclBaseRetrieveRequest
3		TD_M2M_P2P_03	App updates the sclBase resource via an sclBaseUpdateRequest
4		Application	TD_M2M_P2P_04
5	TD_M2M_P2P_05		App retrieves application resource via an applicationRetrieveRequest
6	TD_M2M_P2P_06		App updates attribute in application resource via an applicationUpdateRequest
7	TD_M2M_P2P_07		App de-registers by deleting application resource via an applicationDeleteRequest
8	Container		TD_M2M_P2P_08
9		TD_M2M_P2P_09	App retrieves information of a container resource via a containerRetrieveRequest
10		TD_M2M_P2P_10	App updates attribute in application resource via an containerUpdateRequest
11		TD_M2M_P2P_11	App delete a specific container resource via a containerDeleteRequest
12		ContentInstance	TD_M2M_P2P_12
13	TD_M2M_P2P_13		App adds a contentInstance resource <contentInstance> to the contentInstances resource of a specific container in local SCL via a contentInstanceCreateRequest using plain text data
14	TD_M2M_P2P_14		App retrieves information of a contentInstance resource via a containerRetrieveRequest
15	TD_M2M_P2P_15		Handle contentInstanceRetrieveRequest with targetID using partial addressing (content attribute)
16	TD_M2M_P2P_16		App deletes contentInstance resource via a containerDeleteRequest
17	Discovery		TD_M2M_P2P_17
18		TD_M2M_P2P_18	App discovers resources residing in Local SCL using specific Filter criteria via DiscoveryRetrieveRequest
19		TD_M2M_P2P_19	App discovers resources residing in Local SCL with a limited max size via DiscoveryRetrieveRequest
20		TD_M2M_P2P_20	App discovers resources residing in Local SCL limited search prefix via DiscoveryRetrieveRequest
21		TD_M2M_P2P_21	App discovers resources residing in Local SCL using multiple parameters via DiscoveryRetrieveRequest
22	Subscription	TD_M2M_P2P_22	App creates a subscription to application resource via subscriptionCreateRequest
23		TD_M2M_P2P_23	App retrieves information about a subscription via subscriptionRetrieveRequest (GET), such as expirationTime, filterCriteria, contactURI, noRepresentation, timeoutReason, etc.
24		TD_M2M_P2P_24	App updates information about a subscription via subscriptionRetrieveRequest
25		TD_M2M_P2P_25	App cancels subscription via an subscriptionDeleteRequest
26		TD_M2M_P2P_26	SCL sends notification(s) via subscriptionNotifyRequest
27		TD_M2M_P2P_27	SCL sends notification(s) via subscriptionNotifyRequest (To application container)

Nb	Procedure/Resource	TD ID	TD Description
28	NotificationChannel	TD_M2M_P2P_28	App creates a notificationChannel to application resource via notificationChannelCreateRequest
29		TD_M2M_P2P_29	App retrieves information about a notificationChannel via notificationChannelRetrieveRequest
30		TD_M2M_P2P_30	App deletes a notificationChannel via a notificationChannelDeleteRequest
31		TD_M2M_P2P_31	NA/DA receives notifications via a notificationChannelRetrieveRequest (GET) using long polling
32	AccessRight	TD_M2M_P2P_32	App create an AccessRight resource via an accessRightCreateRequest
33		TD_M2M_P2P_33	App retrieves accessRight resource via an accessRightRetrieveRequest
34		TD_M2M_P2P_34	App updates attribute in accessRight resource via an accessRightUpdateRequest
35		TD_M2M_P2P_35	App de-registers by deleting accessRight resource via an accessRightDeleteRequest
36		TD_M2M_P2P_36	App retrieve request is rejected due to an Empty Requesting Entity
37		TD_M2M_P2P_37	App retrieve request is rejected due to bad request Entity
38		TD_M2M_P2P_38	App delete request is rejected due to Access Rights
39	Group	TD_M2M_P2P_39	App creates a group resource via a groupCreateRequest
40		TD_M2M_P2P_40	App retrieves group resource via a groupRetrieveRequest
41		TD_M2M_P2P_41	App updates attribute in group resource via an groupUpdateRequest
42		TD_M2M_P2P_42	App deletes group resource via an groupDeleteRequest
43		TD_M2M_P2P_43	App creates multiple contentInstance using group resource
44		TD_M2M_P2P_44	App retrieves multiple contentInstance using group resource
45	Container	TD_M2M_MH_01	App creates a container resource in remote SCL via a containerCreateRequest
46		TD_M2M_MH_02	App retrieves information of a container resource via a containerRetrieveRequest
47		TD_M2M_MH_03	App updates attribute in application resource via an containerUpdateRequest
48		TD_M2M_MH_04	App delete a specific container resource via a containerDeleteRequest
49	ContentInstance	TD_M2M_MH_05	App adds a contentInstance resource <contentInstance> to the contentInstances resource of a specific container in remote SCL via a contentInstanceCreateRequest
50		TD_M2M_MH_06	App adds a contentInstance resource <contentInstance> to the contentInstances resource of a specific container in remote SCL via a contentInstanceCreateRequest using Plain text
51		TD_M2M_MH_07	App retrieves remote contentInstance resource via a containerRetrieveRequest
52		TD_M2M_MH_08	Handle contentInstanceRetrieveRequest with targetID using partial addressing
53		TD_M2M_MH_09	App deletes remote contentInstance resource via a containerDeleteRequest
55	Discovery	TD_M2M_MH_10	App discovers resources residing in remote SCL via DiscoveryRetrieveRequest
56		TD_M2M_MH_11	App discovers resources residing in remote SCL using specific Filter criteria via DiscoveryRetrieveRequest
57		TD_M2M_MH_12	App discovers resources residing in remote SCL with a limited max size via DiscoveryRetrieveRequest
58		TD_M2M_MH_13	App discovers resources residing in remote SCL limited search prefix via DiscoveryRetrieveRequest
59		TD_M2M_MH_14	App discovers resources residing in remote SCL using multiple parameters via DiscoveryRetrieveRequest

Nb	Procedure/Resource	TD ID	TD Description
60	Subscription	TD_M2M_MH_15	App creates a subscription to application resource via subscriptionCreateRequest
61		TD_M2M_MH_16	App retrieves information about a subscription via subscriptionRetrieveRequest (GET), such as expirationTime, filterCriteria, contactURI, noRepresentation, timeoutReason, etc.
62		TD_M2M_MH_17	App updates information about a subscription via subscriptionRetrieveRequest
63		TD_M2M_MH_18	App cancels subscription via an subscriptionDeleteRequest
64		TD_M2M_MH_19	GSCL sends notification(s) via subscriptionNotifyRequest on NSCL
65		TD_M2M_MH_20	NSCL sends notification(s) via subscriptionNotifyRequest on GSCL
66		TD_M2M_MH_21	GSCL sends notification(s) via subscriptionNotifyRequest
67		TD_M2M_MH_22	NSCL sends notification(s) via subscriptionNotifyRequest
68	Notification Channel	TD_M2M_MH_23	App creates a notificationChannel to application resource on remote SCL via notificationChannelCreateRequest (POST)
69		TD_M2M_MH_24	App retrieves information about a notificationChannel on remote SCL via notificationChannelRetrieveRequest (GET)
70		TD_M2M_MH_25	App deletes a notificationChannel on remote SCL via a notificationChannelDeleteRequest
71		TD_M2M_MH_26	NA receives notifications via a notificationChannelRetrieveRequest (GET) using long polling
72		TD_M2M_MH_27	DA receives notifications via a notificationChannelRetrieveRequest (GET) using long polling
73	AccessRight	TD_M2M_MH_28	App creates an AccessRight resource via an accessRightCreateRequest
74		TD_M2M_MH_29	App retrieves accessRight resource via an accessRightRetrieveRequest
75		TD_M2M_MH_30	App updates attribute in accessRight resource via an accessRightUpdateRequest
76		TD_M2M_MH_31	App de-registers by deleting accessRight resource via an accessRightDeleteRequest
77		TD_M2M_MH_32	App delete request is rejected due to Access Rights (Empty Requesting Entity)
78		TD_M2M_MH_33	App retrieve request is rejected due to Access Rights (Bad Requesting Entity)
79		TD_M2M_MH_34	App delete request is rejected due to Access Rights on remote SCL (Insufficient Access Rights)
80	Group	TD_M2M_MH_35	App creates a group resource via a groupCreateRequest
81		TD_M2M_MH_36	App retrieves group resource via a groupRetrieveRequest
82		TD_M2M_MH_37	App updates attribute in group resource via an groupUpdateRequest
83		TD_M2M_MH_38	App delete group resource via an groupDeleteRequest
84		TD_M2M_MH_39	App creates multiple contentInstance using group resource
85		TD_M2M_MH_40	App retrieves multiple contentInstance using group resource
86		TD_M2M_MH_41	App registration with Announcement
87		TD_M2M_MH_42	App application delete with Announcement
88		TD_M2M_MH_43	App application De-Announce resource
89	Resource management with re-targeting	TD_M2M_E2E_01	DA adds a contentInstance resource <contentInstance> to the contentInstances resource of a specific container in remote NSCL via a contentInstanceCreateRequest
90		TD_M2M_E2E_02	NA adds a contentInstance resource <contentInstance> to the contentInstances resource of a specific container in remote GSCL via a contentInstanceCreateRequest
91		TD_M2M_E2E_03	NA retrieves a contentInstance resource <contentInstance> to the contentInstances resource of a specific container in remote GSCL via a contentInstanceRetrieveRequest
92		TD_M2M_E2E_04	NA deletes a contentInstance resource <contentInstance> to the contentInstances resource of a specific container in remote GSCL via a contentInstanceUpdateRequest

## 6 Basic Configuration

### 6.1 Resources request samples

#### 6.1.1 Application

```
POST 127.0.0.1:8080/context/nscl/applications
<tns:application xmlns:tns="http://uri.etsi.org/m2m" appId="appl" />
```

```
PUT 127.0.0.1:8080/context/nscl/applications/appl
<tns:application xmlns:tns="http://uri.etsi.org/m2m" >
  <tns:searchStrings>
    <tns:searchString>key</tns:searchString>
  </tns:searchStrings>
</tns:application>
```

```
GET 127.0.0.1:8080/context/nscl/applications/appl
DELETE 127.0.0.1:8080/context/nscl/applications/appl
```

#### 6.1.2 Container

```
POST 127.0.0.1:8080/context/nscl/applications/appl/containers
<tns:container xmlns:tns="http://uri.etsi.org/m2m" tns:id="cont1" />
```

```
PUT 127.0.0.1:8080/context/nscl/applications/appl/containers/cont1
<tns:container xmlns:tns="http://uri.etsi.org/m2m">
  <tns:searchStrings>
    <tns:searchString>key</tns:searchString>
  </tns:searchStrings>
</tns:container>
```

```
GET 127.0.0.1:8080/context/nscl/applications/appl
```

```
DELETE 127.0.0.1:8080/context/nscl/applications/appl
```

#### 6.1.3 ContentInstance

```
POST 127.0.0.1:8080/context/nscl/applications/appl/containers/cont1/contentInstances
<tns:contentInstance xmlns:tns="http://uri.etsi.org/m2m"
  xmlns:xmime="http://www.w3.org/2005/05/xmlmime" tns:id="cil" >
  <tns:content xmime:contentType="application/xml">
```

```
PG9iaj4KICA8aW50IG5hbWU9ImRhdGEiIHZhbD0iMjcilz4KICA8aW50IG5hbWU9InVuaXQiIHZhbD0iY2Vsc3VzIi8+Cjwv
b2JqPgoK
  </tns:content>
</tns:contentInstance>
```

```
POST(content) 127.0.0.1:8080/context/nscl/applications/appl/containers/cont1/contentInstances
<obj>
  <int name="data" val="27"/>
  <int name="unit" val="celsius"/>
</obj>
```

```
PUT 127.0.0.1:8080/context/nscl/applications/appl/containers/cont1
<tns:container xmlns:tns="http://uri.etsi.org/m2m">
  <tns:searchStrings>
    <tns:searchString>key</tns:searchString>
  </tns:searchStrings>
</tns:container>
```

```
GET 127.0.0.1:8080/context/nscl/applications/appl/containers/cont1/contentInstances/ci
```

```
GET (latest) 127.0.0.1:8080/context/nscl/applications/appl/containers/cont1/contentInstances/latest
```

```
GET (oldest) 127.0.0.1:8080/context/nscl/applications/appl/containers/cont1/contentInstances/oldest
```

```
GET (content)
127.0.0.1:8080/context/nscl/applications/appl/containers/cont1/contentInstances/ci/content
```

```
GET (latest content)
127.0.0.1:8080/context/nscl/applications/appl/containers/cont1/contentInstances/latest/content
```

```
GET (oldest content)
127.0.0.1:8080/context/nscl/applications/appl/containers/cont1/contentInstances/oldest/content
```

## 6.1.4 Subscription

```
POST 127.0.0.1:8080/context/nscl/applications/appl/containers/cont1/contentInstances/subscriptions
<tns:subscription xmlns:tns="http://uri.etsi.org/m2m" tns:id="sub1">
  <tns:contact>http://127.0.0.0:1400/monitor</tns:contact>
</tns:subscription>
```

```
GET
127.0.0.1:8080/context/nscl/applications/appl/containers/cont1/contentInstances/subscriptions/sub1
```

```
DELETE
127.0.0.1:8080/context/nscl/applications/appl/containers/cont1/contentInstances/subscriptions/sub1
```

## 6.1.5 AccessRight

```
POST 127.0.0.1:8080/context/nscl/accessRights
<tns:accessRight xmlns:tns="http://uri.etsi.org/m2m" tns:id="ar1">
  <tns:permissions>
    <tns:permission tns:id="AllRights">
      <tns:permissionFlags>
        <tns:flag>CREATE</tns:flag>
        <tns:flag>WRITE</tns:flag>
        <tns:flag>READ</tns:flag>
        <tns:flag>DELETE</tns:flag>
        <tns:flag>DISCOVER</tns:flag>
      </tns:permissionFlags>
      <tns:permissionHolders>
        <tns:holderRefs>
          <tns:holderRef>admin/admin</tns:holderRef>
        </tns:holderRefs>
      </tns:permissionHolders>
    </tns:permission>
    <tns:permission tns:id="RestrictedRights">
      <tns:permissionFlags>
        <tns:flag>READ</tns:flag>
        <tns:flag>DISCOVER</tns:flag>
      </tns:permissionFlags>
      <tns:permissionHolders>
        <tns:holderRefs>
          <tns:holderRef>guest/guest</tns:holderRef>
        </tns:holderRefs>
      </tns:permissionHolders>
    </tns:permission>
  </tns:permissions>
  <tns:selfPermissions>
    <tns:permission tns:id="SelfPermission">
      <tns:permissionFlags>
        <tns:flag>CREATE</tns:flag>
        <tns:flag>WRITE</tns:flag>
        <tns:flag>READ</tns:flag>
        <tns:flag>DELETE</tns:flag>
        <tns:flag>DISCOVER</tns:flag>
      </tns:permissionFlags>
      <tns:permissionHolders>
        <tns:holderRefs>
          <tns:holderRef>admin/admin</tns:holderRef>
        </tns:holderRefs>
      </tns:permissionHolders>
    </tns:permission>
  </tns:selfPermissions>
</tns:accessRight>
```

```
PUT 127.0.0.1:8080/context/nscl/accessRights/ar1
<tns:accessRight xmlns:tns="http://uri.etsi.org/m2m" >
  <tns:permissions>
    <tns:permission tns:id="AllRights">
      <tns:permissionFlags>
        <tns:flag>CREATE</tns:flag>
        <tns:flag>WRITE</tns:flag>
```