|  |
| --- |
|  |

|  |  |
| --- | --- |
| CHANGE REQUEST | |
| Meeting ID:\* | PRO 30.1 |
| Source:\* | Poornima, C-DOT, [poornima@cdot.in](mailto:poornima@cdot.in)  Giribabu Naik Moode, C-DOT, [moode@cdot.in](mailto:moode@cdot.in) |
| Date:\* | 2017-08-04 |
| Reason for Change/s:\* | See the introduction |
| CR against: Release\* | Release 2 |
| CR against: WI\* | Active <Work Item number>  MNT maintenance / < Work Item number(optional)>  Is this a mirror CR? Yes  No  mirror CR number: <MAS-2017-0204>(Note to Rapporteur - use latest agreed revision)  STE Small Technical Enhancements / < Work Item number (optional)>  Only ONE of the above shall be ticked |
| CR against: TS/TR\* | TS-0006 Version 2.0.1 |
| Clauses \* | Chapter 5, clause 8.1 |
| Type of change: \* | Editorial change  Bug Fix or Correction  Change to existing feature or functionality  New feature or functionality  Only ONE of the above shall be ticked |
| Impacted other TS/TR(s) | <TS/TR number>, <Version Number>, and <Description on which aspect should be reflected in this TS/TR> |
| Post Freeze checking:\* | This CR contains only essential changes and corrections? YES  NO  This CR may break backwards compatibility with the last approved version of the TS? YES  NO |
| Template Version: January 2017 (Do not modify) | |

**oneM2M Notice**

The document to which this cover statement is attached is submitted to oneM2M. Participation in, or attendance at, any activity of oneM2M, constitutes acceptance of and agreement to be bound by terms of the Working Procedures and the Partnership Agreement, including the Intellectual Property Rights (IPR) Principles Governing oneM2M Work found in Annex 1 of the Partnership Agreement.

GUIDELINES for Change Requests:

Provide an informative introduction containing the problem(s) being solved, and a summary list of proposals.

Each CR should contain changes related to only one particular issue/problem.

In case of a correction, and the change apply to previous releases, a separate “mirror CR” should be posted at the same time of this CR

Mirror CR: applies only when the text, including clause numbering are exactly the same.

Companion CR: applies when the change means the same but the baselines differ in some way (e.g. clause number).

Follow the principle of completeness, where all changes related to the issue or problem within a deliverable are simultaneously proposed to be made E.g. A change impacting 5 tables should not only include a proposal to change only 3 tables. Includes any changes to references, definitions, and acronyms in the same deliverable.

Follow the drafting rules.

All pictures must be editable.

Check spelling and grammar to the extent practicable.

Use Change bars for modifications.

The change should include the current and surrounding clauses to clearly show where a change is located and to provide technical context of the proposed change. Additions of complete clauses need not show surrounding clauses as long as the proposed clause number clearly shows where the new clause is proposed to be located.

Multiple changes in a single CR shall be clearly separated by horizontal lines with embedded text such as, start of change 1, end of change 1, start of new clause, end of new clause.

When subsequent changes are made to content of a CR, then the accepted version should not show changes over changes. The accepted version of the CR should only show changes relative to the baseline approved text.

## Introduction

The CR proposes Response Status Codes in TS-0006 to map with Response Status Codes in TS-0004

### -----------------------Start of change 1-------------------------------------------

# 5 Mapping of basic data types

TR-106 [5] specifies the object structure supported by TR-069 enabled devices and specifies the structural requirements for the data hierarchy. This clause includes the mapping attribute data types to TR-181 [6] parameters which follows the conventions of section 3 of TR-106 [5] and data types described in Table 4 of TR-106 [5].

Table 5-1: Data Type Mapping

| **oneM2M Data Types** | **Mapping to data types in TR-106** | **Conversion Notes** |
| --- | --- | --- |
| xs:boolean | boolean |  |
| xs:string | string | Mapping is constrained to the size of the string |
| xs:unsignedInt | unsignedInt |  |
| xs:unsignedLong | unsignedLong |  |
| xs:integer | long | Mapping is constrained to the size of the long data type. |
| Xs:positiveInteger | unsignedLong | Mapping is constrained to a lower limit of 1 and the size of the unsignedLong data type. |
| Xs:nonNegativeInteger | unsignedLong | Mapping is constrained the size of the unsignedLong data type. |
| Comma separated Lists | Comma separated Lists | Data structure is represented by comma separated list as described in section 3.2.3 of TR-106 [5]. |

In some instances the conversion of the contents between data types will cause an error to occur (e.g. xs:integer to long). When an error occurs in the conversion of a data type, the 4000 (BAD\_REQUEST) response status code shall be given.

### -----------------------End of change 1-------------------------------------------

### -----------------------Start of change 2-------------------------------------------

# 8 Mapping of procedures for management

## 8.0 Introduction

This clause contains all information on how to map management resource primitives from TS-0004 [2] to the Remote Procedure Calls (RPCs) in TR-069 [4].

## 8.1 Resource Type <mgmtObj> primitive mappings

## 8.1.0 Introduction

This clause contains all information on how to map Resource Type <mgmtObj> primitives from TS-0004 [2] to the Remote Procedure Calls (RPCs) in TR-069 [4].

## 8.1.1 Alias-Based Addressing Mechanism

In order to utilize the Alias-Based Addressing Mechanism, the mechanism has to be supported by the ACS and CPE in order to map the M2M Service Layer identifier for the Resource instance to the CPE object instance. If the Alias‑Based Addressing Mechanism feature is not supported by either the ACS or CPE, the CSE has to retain the mapping of the these M2M Resource instance identifiers.

### 8.1.2 Create primitive mapping

#### 8.1.2.0 Introduction

The Create Request and Response primitives shall map to the AddObject RPC. The AddObject RPC is defined in TR-069 [4] as a synchronous RPC and returns a successful response or one of the following fault codes in Table 8.1.2-1.

Table 8.1.2-1: AddObject Fault Code Mapping

| Fault code | Description | Response Status Code |
| --- | --- | --- |
| 9001 | Request denied (no reason specified) | 4000 (BAD\_REQUEST) |
| 9002 | Internal error | 4000 (BAD\_REQUEST) |
| 9003 | Invalid arguments | 4000 (BAD\_REQUEST) |
| 9004 | Resources exceeded (when used in association with SetParameterValues, this cannot be used to indicate Parameters in error) | 4000 (BAD\_REQUEST) |
| 9005 | Invalid Parameter name (associated with Set/GetParameterValues, GetParameterNames, Set/GetParameterAttributes, AddObject, and DeleteObject) | 5001 (NOT\_IMPLEMENTED) |

#### 8.1.2.1 M2M Service Layer Resource Instance Identifier mapping

When the Resource is a multi-instance Resource, the AddObject RPC should utilize the Alias-Based Addressing Mechanism as defined in Section 3.6.1 of TR-069 [4] in order to use the Resource instance value of the URI.

### 8.1.3 Delete primitive mapping

#### 8.1.3.1 Delete primitive mapping for deletion of Object Instances

The Delete Request and Response primitives that results in the deletion of a Resource shall map to the DeleteObject RPC. The DeleteObject RPC is defined in TR-069 [4] as a synchronous RPC and returns a successful response or one of the following fault codes in Table 8.1.3.1-1.

Table 8.1.3.1-1: DeleteObject Fault Code Mapping

| Fault code | Description | Response Status Code |
| --- | --- | --- |
| 9001 | Request denied (no reason specified) | 4000 (BAD\_REQUEST) |
| 9002 | Internal error | 4000 (BAD\_REQUEST) |
| 9003 | Invalid arguments | 4000 (BAD\_REQUEST) |
| 9005 | Invalid Parameter name (associated with Set/GetParameterValues, GetParameterNames, Set/GetParameterAttributes, AddObject, and DeleteObject) | 5001 (NOT\_IMPLEMENTED) |

#### 8.1.3.2 Delete primitive mapping for software un-install operation

The Delete Request and Response primitives that results in a software un-install operation (e.g. Resource [software]) shall use the ChangeDUState mechanism defined in TR-069 [4]. The ChangeDUState mechanism is an asynchronous command that consists of the synchronous ChangeDUState RPC for the un-installation request and the asynchronous ChangeDUStateComplete RPC. The ChangeDUState RPC returns a successful response or one of the following fault codes in Table 8.1.3.2-1. A successful response means that the CPE has accepted the ChangeDUState RPC.

Table 8.1.3.2-1: ChangeDUState Fault Code Mapping

| Fault code | Description | Response Status Code |
| --- | --- | --- |
| 9000 | Method not supported | 4000 (BAD\_REQUEST) |
| 9001 | Request denied (no reason specified) | 4000 (BAD\_REQUEST) |
| 9002 | Internal error | 4000 (BAD\_REQUEST) |
| 9004 | Resources exceeded (when used in association with SetParameterValues, this cannot be used to indicate Parameters in error) | 4000 (BAD\_REQUEST) |

Once the CPE has attempted to change the state of the deployment unit, the CPE reports the result of the state change operation using the ChangeDUStateComplete RPC. The ChangeDUStateComplete RPC indicates a successful operation or one of the following fault codes in Table 8.1.3.2-2.

Table 8.1.3.2-2: ChangeDUStateComplete Fault Code Mapping

| Fault code | Description | Response Status Code |
| --- | --- | --- |
| 9001 | Request denied (no reason specified) | 4000 (BAD\_REQUEST) |
| 9003 | Invalid arguments | 4000 (BAD\_REQUEST) |
| 9012 | File transfer server authentication failure (associated with Upload, Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9013 | Unsupported protocol for file transfer (associated with Upload, Download, ScheduleDownload, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9015 | File transfer failure: unable to contact file server (associated with Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9016 | File transfer failure: unable to access file (associated with Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9017 | File transfer failure: unable to complete download (associated with Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9018 | File transfer failure: file corrupted or otherwise unusable (associated with Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9022 | Invalid UUID Format (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Install, Update, and Uninstall) | 4000 (BAD\_REQUEST) |
| 9023 | Unknown Execution Environment (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Install only) | 4000 (BAD\_REQUEST) |
| 9024 | Disabled Execution Environment (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Install, Update, and Uninstall) | 4000 (BAD\_REQUEST) |
| 9025 | Deployment Unit to Execution Environment Mismatch (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Install and Update) | 4000 (BAD\_REQUEST) |
| 9026 | Duplicate Deployment Unit (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Install only) | 4000 (BAD\_REQUEST) |
| 9027 | System Resources Exceeded (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Install and Update) | 4000 (BAD\_REQUEST) |
| 9028 | Unknown Deployment Unit (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Update and Uninstall) | 4000 (BAD\_REQUEST) |
| 9029 | Invalid Deployment Unit State (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Install, Update and Uninstall) | 4000 (BAD\_REQUEST) |
| 9030 | Invalid Deployment Unit Update - Downgrade not permitted (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Update only) | 4000 (BAD\_REQUEST) |
| 9031 | Invalid Deployment Unit Update - Version not specified (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Update only) | 4000 (BAD\_REQUEST) |
| 9032 | Invalid Deployment Unit Update - Version already exists (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Update only) | 4000 (BAD\_REQUEST) |

### 8.1.4 Update primitive mapping

#### 8.1.4.1 Update primitive mapping for Parameter modifications

The Update Request and Response primitives that modifies the value of Resource attributes shall map to the SetParameterValues RPC. The SetParametersValue RPC is defined in TR-069 [4] as a synchronous RPC and returns a successful response or one of the following fault codes in Table 8.1.4.1-1.

Table 8.1.4.1-1: SetParameterValues Fault Code Mapping

| Fault code | Description | Response Status Code |
| --- | --- | --- |
| 9001 | Request denied (no reason specified) | 4000 (BAD\_REQUEST) |
| 9002 | Internal error | 4000 (BAD\_REQUEST) |
| 9003 | Invalid arguments | 4000 (BAD\_REQUEST) |
| 9004 | Resources exceeded (when used in association with SetParameterValues, this cannot be used to indicate Parameters in error) | 4000 (BAD\_REQUEST) |
| 9005 | Invalid Parameter name (associated with Set/GetParameterValues, GetParameterNames, Set/GetParameterAttributes, AddObject, and DeleteObject) | 5001 (NOT\_IMPLEMENTED) |
| 9006 | Invalid Parameter type (associated with SetParameterValues) | 4000 (BAD\_REQUEST) |
| 9007 | Invalid Parameter value (associated with SetParameterValues) | 4000 (BAD\_REQUEST) |
| 9008 | Attempt to set a non-writable Parameter (associated with SetParameterValues) | 4000 (BAD\_REQUEST) |

#### 8.1.4.2 Update primitive mapping for upload file transfer operations

The Update Request and Response primitives that results in an upload file transfer operation (e.g. logStop attribute of the Resource [eventLog]) shall use the Upload mechanism defined in TR-069 [4]. The Upload mechanism is an asynchronous command that consists of the synchronous Upload RPC for the Upload and the asynchronous TransferComplete RPC. The Upload RPC returns a successful response or one of the following fault codes in Table 8.1.4.2-1. A successful response means that the CPE has accepted the Upload RPC.

Table 8.1.4.2-1: Upload Fault Code Mapping

| Fault code | Description | Response Status Code |
| --- | --- | --- |
| 9000 | Method not supported | 4000 (BAD\_REQUEST) |
| 9001 | Request denied (no reason specified) | 4000 (BAD\_REQUEST) |
| 9002 | Internal error | 4000 (BAD\_REQUEST) |
| 9003 | Invalid arguments | 4000 (BAD\_REQUEST) |
| 9004 | Resources exceeded (when used in association with SetParameterValues, this cannot be used to indicate Parameters in error) | 4000 (BAD\_REQUEST) |
| 9011 | Upload failure (associated with Upload, TransferComplete or Autonomous­Trans­ferComplete methods). | 4000 (BAD\_REQUEST) |
| 9012 | File transfer server authentication failure (associated with Upload, Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9013 | Unsupported protocol for file transfer (associated with Upload, Download, ScheduleDownload, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |

Once the CPE has attempted to upload the file, the CPE reports the result of the Upload operation using the TransferComplete RPC. The TransferComplete RPC indicates a successful operation or one of the following fault codes in Table 8.1.4.2-2.

Table 8.1.4.2-2: TransferComplete Fault Code Mapping

| Fault code | Description | Response Status Code |
| --- | --- | --- |
| 9001 | Request denied (no reason specified) | 4000 (BAD\_REQUEST) |
| 9002 | Internal error | 4000 (BAD\_REQUEST) |
| 9010 | File transfer failure (associated with Download, ScheduleDownload, TransferComplete or AutonomousTransferComplete methods). | 4000 (BAD\_REQUEST) |
| 9011 | Upload failure (associated with Upload, TransferComplete or Autonomous­Trans­ferComplete methods). | 4000 (BAD\_REQUEST) |
| 9012 | File transfer server authentication failure (associated with Upload, Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9014 | File transfer failure: unable to join multicast group (associated with Download, TransferComplete or AutonomousTransferComplete methods). | 4000 (BAD\_REQUEST) |
| 9015 | File transfer failure: unable to contact file server (associated with Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9016 | File transfer failure: unable to access file (associated with Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9017 | File transfer failure: unable to complete download (associated with Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9018 | File transfer failure: file corrupted or otherwise unusable (associated with Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9019 | File transfer failure: file authentication failure (associated with Download, TransferComplete or AutonomousTransferComplete methods). | 4000 (BAD\_REQUEST) |
| 9020 | File transfer failure: unable to complete download within specified time windows (associated with TransferComplete method). | 4000 (BAD\_REQUEST) |

#### 8.1.4.3 Update primitive mapping for download file transfer operations

The Update Request and Response primitives that results in a download file transfer operation (e.g. update attribute of Resource [firmware]) shall use the Download mechanism defined in TR-069 [4]. The Download mechanism is an asynchronous command that consists of the synchronous Download RPC for the Download and the asynchronous TransferComplete RPC. The Download RPC returns a successful response or one of the following fault codes in Table 8.1.4.3-1. A successful response means that the CPE has accepted the Download RPC.

Table 8.1.4.3-1: Download Fault Code Mapping

| Fault code | Description | Response Status Code |
| --- | --- | --- |
| 9000 | Method not supported | 4000 (BAD\_REQUEST) |
| 9001 | Request denied (no reason specified) | 4000 (BAD\_REQUEST) |
| 9002 | Internal error | 4000 (BAD\_REQUEST) |
| 9003 | Invalid arguments | 4000 (BAD\_REQUEST) |
| 9004 | Resources exceeded (when used in association with SetParameterValues, this cannot be used to indicate Parameters in error) | 4000 (BAD\_REQUEST) |
| 9010 | File transfer failure (associated with Download, ScheduleDownload, TransferComplete or AutonomousTransferComplete methods). | 4000 (BAD\_REQUEST) |
| 9012 | File transfer server authentication failure (associated with Upload, Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9013 | Unsupported protocol for file transfer (associated with Upload, Download, ScheduleDownload, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |

Once the CPE has attempted to download the file, the CPE reports the result of the download operation using the TransferComplete RPC. The TransferComplete RPC indicates a successful operation or one of the following fault codes in Table 8.1.4.3-2.

Table 8.1.4.3-2: TransferComplete Fault Code Mapping

| **Fault code** | **Description** | **Response Status Code** |
| --- | --- | --- |
| 9001 | Request denied (no reason specified) | 4000 (BAD\_REQUEST) |
| 9002 | Internal error | 4000 (BAD\_REQUEST) |
| 9010 | File transfer failure (associated with Download, ScheduleDownload, TransferComplete or AutonomousTransferComplete methods). | 4000 (BAD\_REQUEST) |
| 9011 | Upload failure (associated with Upload, TransferComplete or Autonomous­Trans­ferComplete methods). | 4000 (BAD\_REQUEST) |
| 9012 | File transfer server authentication failure (associated with Upload, Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9014 | File transfer failure: unable to join multicast group (associated with Download, TransferComplete or AutonomousTransferComplete methods). | 4000 (BAD\_REQUEST) |
| 9015 | File transfer failure: unable to contact file server (associated with Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9016 | File transfer failure: unable to access file (associated with Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9017 | File transfer failure: unable to complete download (associated with Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9018 | File transfer failure: file corrupted or otherwise unusable (associated with Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9019 | File transfer failure: file authentication failure (associated with Download, TransferComplete or AutonomousTransferComplete methods). | 4000 (BAD\_REQUEST) |
| 9020 | File transfer failure: unable to complete download within specified time windows (associated with TransferComplete method). | 4000 (BAD\_REQUEST) |

#### 8.1.4.4 Update primitive mapping for reboot operation

The Update Request and Response primitives that results in a reboot operation (e.g. reboot attribute of Resource [reboot]) shall use the Reboot RPC defined in TR-069 [4]. The Reboot RPC is asynchronous command. The Reboot RPC returns a successful response or one of the following fault codes in Table 8.1.4.4-1.

Table 8.1.4.4-1: Reboot Fault Code Mapping

| Fault code | Description | Response Status Code |
| --- | --- | --- |
| 9001 | Request denied (no reason specified) | 4000 (BAD\_REQUEST) |
| 9002 | Internal error | 4000 (BAD\_REQUEST) |
| 9003 | Invalid arguments | 4000 (BAD\_REQUEST) |

#### 8.1.4.5 Update primitive mapping for factory reset operation

The Update Request and Response primitives that results in a factory reset operation (e.g. factoryReset attribute of Resource [ reboot]) shall use the FactoryReset RPC defined in TR-069 [4]. The FactoryReset RPC is an asynchronous command. The FactoryReset RPC returns a successful response or one of the following fault codes in Table 8.1.4.5-1.

Table 8.1.4.5-1: FactoryReset Fault Code Mapping

| Fault code | Description | Response Status Code |
| --- | --- | --- |
| 9000 | Method not supported | 4000 (BAD\_REQUEST) |
| 9001 | Request denied (no reason specified) | 4000 (BAD\_REQUEST) |
| 9002 | Internal error | 4000 (BAD\_REQUEST) |
| 9003 | Invalid arguments | 4000 (BAD\_REQUEST) |

#### 8.1.4.6 Update primitive mapping for software install operation

The Update Request and Response primitives that results in a software installation operation (e.g. install attribute of Resource [software]) shall use the ChangeDUState mechanism defined in TR-069 [4]. The ChangeDUState mechanism is an asynchronous command that consists of the synchronous ChangeDUState RPC for the download and the asynchronous ChangeDUStateComplete RPC. The ChangeDUState RPC returns a successful response or one of the following fault codes in Table 8.1.4.6-1. A successful response means that the CPE has accepted the ChangeDUState RPC.

Table 8.1.4.6-1: ChangeDUState Fault Code Mapping

| Fault code | Description | Response Status Code |
| --- | --- | --- |
| 9000 | Method not supported | 4000 (BAD\_REQUEST) |
| 9001 | Request denied (no reason specified) | 4000 (BAD\_REQUEST) |
| 9002 | Internal error | 4000 (BAD\_REQUEST) |
| 9004 | Resources exceeded (when used in association with SetParameterValues, this cannot be used to indicate Parameters in error) | 4000 (BAD\_REQUEST) |

Once the CPE has attempted to change the state of the deployment unit, the CPE reports the result of the state change operation using the ChangeDUStateComplete RPC. The ChangeDUStateComplete RPC indicates a successful operation or one of the following fault codes in Table 8.1.4.6-2.

Table 8.1.4.6-2: ChangeDUStateComplete Fault Code Mapping

| Fault code | Description | Response Status Code |
| --- | --- | --- |
| 9001 | Request denied (no reason specified) | 4000 (BAD\_REQUEST) |
| 9003 | Invalid arguments | 4000 (BAD\_REQUEST) |
| 9012 | File transfer server authentication failure (associated with Upload, Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9013 | Unsupported protocol for file transfer (associated with Upload, Download, ScheduleDownload, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9015 | File transfer failure: unable to contact file server (associated with Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9016 | File transfer failure: unable to access file (associated with Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9017 | File transfer failure: unable to complete download (associated with Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9018 | File transfer failure: file corrupted or otherwise unusable (associated with Download, TransferComplete, AutonomousTransferComplete, DUStateChangeComplete, or AutonomousDUStateChangeComplete methods). | 4000 (BAD\_REQUEST) |
| 9022 | Invalid UUID Format (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Install, Update, and Uninstall) | 4000 (BAD\_REQUEST) |
| 9023 | Unknown Execution Environment (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Install only) | 4000 (BAD\_REQUEST) |
| 9024 | Disabled Execution Environment (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Install, Update, and Uninstall) | 4000 (BAD\_REQUEST) |
| 9025 | Deployment Unit to Execution Environment Mismatch (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Install and Update) | 4000 (BAD\_REQUEST) |
| 9026 | Duplicate Deployment Unit (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Install only) | 4000 (BAD\_REQUEST) |
| 9027 | System Resources Exceeded (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Install and Update) | 4000 (BAD\_REQUEST) |
| 9028 | Unknown Deployment Unit (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Update and Uninstall) | 4000 (BAD\_REQUEST) |
| 9029 | Invalid Deployment Unit State (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Install, Update and Uninstall) | 4000 (BAD\_REQUEST) |
| 9030 | Invalid Deployment Unit Update - Downgrade not permitted (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Update only) | 4000 (BAD\_REQUEST) |
| 9031 | Invalid Deployment Unit Update - Version not specified (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Update only) | 4000 (BAD\_REQUEST) |
| 9032 | Invalid Deployment Unit Update - Version already exists (associated with DUStateChangeComplete or AutonomousDUStateChangeComplete methods: Update only) | 4000 (BAD\_REQUEST) |

### 8.1.5 Retrieve primitive mapping

The Retrieve Request and Response primitives shall map to the GetParameterValues RPC. The GetParametersValue RPC is defined in TR-069 [4] as a synchronous RPC and returns a successful response or one of the following fault codes in Table 8.1.5-1.

Table 8.1.5-1: GetParameterValues Fault Code Mapping

| Fault code | Description | Response Status Code |
| --- | --- | --- |
| 9001 | Request denied (no reason specified) | 4000 (BAD\_REQUEST) |
| 9002 | Internal error | 4000 (BAD\_REQUEST) |
| 9003 | Invalid arguments | 4000 (BAD\_REQUEST) |
| 9004 | Resources exceeded (when used in association with SetParameterValues, this cannot used to indicate Parameters in error) | 4000 (BAD\_REQUEST) |
| 9005 | Invalid Parameter name (associated with Set/GetParameterValues, GetParameterNames, Set/GetParameterAttributes, AddObject, and DeleteObject) | 4000 (BAD\_REQUEST) |

### 8.1.6 Notify primitive mapping

#### 8.1.6.0 Introduction

The NotifyRequest and Response primitives permit notifications to AE or CSEs that have subscribed to a Resource.

While TR-069 [4] has the capability to notify the subscribed ACS when an object's parameter has been modified, TR‑069 [4] does not have the capability for an ACS to be notified if any parameter within the object has been modified unless the ACS individually subscribes to all the parameters of the object.

As such the procedure for mapping the Notify Request and Response primitives for TR-069 [4] is not possible unless the CSE subscribes to receive notification to all the parameters of an Object that are mapped to the Resource's attributes.

NOTE: In many implementations, subscribing to all the parameters of an Object that are mapped to the Resource can cause performance issues in the CPE as well as the CSE. As such using the attribute based subscription capabilities of TR-069 [4] for subscription of Resources should be avoided when possible.

#### 8.1.6.1 Procedure for subscribed Resource attributes.

When a <subscription> Resource for a <mgmtObj> Resource is Created, Deleted or Updated the CSE shall map to the SetParameterAttributes RPC in the following manner:

* TR-069 [4] provides the capability to subscribe to changes of a specific attribute through the use of the SetParameterAttributes RPC using the "Active" value for the Notification parameter.
* TR-069 [4] provides the capability to un-subscribe to changes of a specific attribute through the use of the SetParameterAttributes RPC using the "None" value for the Notification parameter.

The SetParametersAttributes RPC is defined in TR-069 [4] as a synchronous RPC and returns a successful response or one of the following fault codes in Table 8.1.6.1-1.

Table 8.1.6.1-1: SetParameterAttributes Fault Code Mapping

| Fault code | Description | Response Status Code |
| --- | --- | --- |
| 9000 | Method not supported | 4000 (BAD\_REQUEST) |
| 9001 | Request denied (no reason specified) | 4000 (BAD\_REQUEST) |
| 9002 | Internal error | 4000 (BAD\_REQUEST) |
| 9003 | Invalid arguments | 4000 (BAD\_REQUEST) |
| 9004 | Resources exceeded (when used in association with SetParameterValues, this cannot be used to indicate Parameters in error) | 4000 (BAD\_REQUEST) |
| 9010 | File transfer failure (associated with Download, ScheduleDownload, TransferComplete or AutonomousTransferComplete methods). | 4000 (BAD\_REQUEST) |

#### 8.1.6.2 Notification primitive mapping

Notify Request and Response primitives shall map to the TR-069 notification mechanism. CPEs produce notifications for subscribed attributes using the TR-069 Inform method, the Inform method has an argument Event that has as one of the EventCodes with the value "4 VALUE CHANGE" indicating that a subscribed parameter's value has changed. The parameter(s) that have changed are included ParameterList argument of the Inform method.

The ParameterList argument is list of name-value pairs; the name is parameter name and shall be mapped to the objectPath attribute of the Resource while the value is the most recent value of the parameter.

NOTE: TR-069 CPEs do not report value changes of parameters that were modified by the ACS.

### -----------------------End of change 2-------------------------------------------

CHECK LIST

* Does this Change Request include an informative introduction containing the problem(s) being solved, and a summary list of proposals.?
* Does this CR contain changes related to only one particular issue/problem?
* Have any mirror CRs been posted?
* Does this Change Request make **all** the changes necessary to address the issue or problem? E.g. A change impacting 5 tables should not include a proposal to change only 3 tables?Does this Change Request follow the drafting rules?
* Are all pictures editable?
* Have you checked the spelling and grammar?
* Have you used change bars for all modifications?
* Does the change include the current and surrounding clauses to clearly show where a change is located and to provide technical context of the proposed change? (Additions of complete clauses need not show surrounding clauses as long as the proposed clause number clearly shows where the new clause is proposed to be located.)
* Are multiple changes in this CR clearly separated by horizontal lines with embedded text such as, start of change 1, end of change 1, start of new clause, end of new clause.?