Markdown Input

```
---
title: Model.Patch (V2)
lang: en-us
description: Overview of how the Device API functions to support PATCH operations
---

# Model.Patch Request

## Overview

The Ability Platform supports sending info model patches using the `model.patch`
API endpoint. The `model.patch` action message is sent to the platform by a
directly connected device (i.e. Edge) via the IoTHub. All C2D messages (acks and
notifications) sent in response to a `model.patch` action are sent in Device API
v2 format.

The `model.patch` message is Idempotent. This means that sending several
identical requests will result in the model being updated successfully in the
Info Model Service:

- The first partial update of an existing object model (determined via objectId + model) will be successful.

- Subsequent partial updates of this object model will be rejected due to an
  object model version mismatch.

## Prerequisites

- The requested info model **must** exist.
```

Message Format

```json
{
  "properties": {
    "iothub-connection-device-id": "<authenticated device id>",
    "msgType": "action",
    "action": "model.patch",
    "version": 2,
    "correlationId": "<correlation id>",
    "objectId": "<GUID>",
    "model": "<model definition>",
    "ack": "<ack>",
    "target": "<connected device path>",
    "timeout": "<timeout>"
  },
  "body": "<body>"
}
```

Message Properties

Added by the IoT Hub

Property	Mandatory	Data Type	Description
iothub-connection-device-id	Mandatory	string	Device ID of the directly connected device.

Common for all v2 Requests

|Property|Mandatory|Description|

```
|---|---|---|
|msgType|Mandatory|Must be equal to 'action'|
|action|Mandatory|Must be equal to 'model.patch'|
|version|Mandatory|Must be equal to 2|
|correlationId|Optional|Correlation ID. Should be included to track correlation of platform components. If not p
resent, the DCS will generate a new one.|
|ack|Optional|Initializes to "none" value if the value is missing. Indicates an additional acknowledgement messa
ge to be sent to the device.For more details see [Acknowledgement Handling](../ack-handling.md).|
|target|Optional|If not provided, defaults to empty string. Represents the path to a connected device that wishe
s to receive the acknowledgement (if requested).|
|timeout|Optional|If not provided, defaults to a 300 integer. Represents the parameter that will be used to indi
cate the maximum number of seconds available for processing the D2C message until the timeout occurs. This exclu
des sending back response C2D messages.|

### Specific to a 'model.patch' Request

|Property|Mandatory|Description|
|---|---|---|
objectId|Mandatory|Source object identifier. Must be in GUID format.
|model|Optional|Source model definition identifier (modelId). Case sensitive. If not provided, defaults to "abb.
ability.device".

### Message Body

The body must consist of a well-formed JSON compliant with the request payload
format of a `PATCH /objects/{objectId}/models/{modelId}` operation.

Property|Mandatory|Data Type|Description
|---|---|---|---|
|version|Mandatory|integer|Must be equal to or higher than the version of the model existing in the IM database.
|
```

```
## Validation Rules

- The DCS will validate all message properties as described above.

- The DCS will validate only the body property **version**.

## Functionality

### Description

- The object model is partially updated in the info model service.

- In case of version gaps, the DCS will perform a patch operation multiple
  times.

- After all patches are completed, the patched model version in the IM is equal
  to the request message version + 1.

- A model notification C2D message is not sent back.

:::tip NOTE
Any platform events generated by the info model service related to this request are ignored by the DCS and are n
ot propagated.
:::

- If requested, an acknowledgement C2D message is sent back to the directly
  connected device where it can also be propagated further if needed (based on
  target). For more details see [Acknowledgement Handling](../ack-handling.md).

  - The target is reused from the requested message.

### Format of a Returned Acknowledgment C2D Message
```

Format of a Successful Acknowledgment

```json
{
    "properties": {
      "msgType": "ack",
      "timestamp": "YYYY-MM-DDTHH:mm:ss.sssZ",
      "action": "model.patch",
      "correlationId": "<correlation id>",
      "version": 2,
      "target": "<target matches the target in the corresponding action>"
    },
    "body": {
       "success": true,
       "code": "ok",
       "details": "''",
       "objectId": "<objectId>",
       "model": "<model>",
       "version": "<version>"
    }
}
```

Description of body properties

Property	Mandatory	Data Type	Description
objectId	Mandatory	JSON object	ObjectId of patched model.
model	Mandatory	JSON object	Model identifier of patched model.
version	Mandatory	JSON object	Version of patched model.

Format of a Failing Acknowledgment

```json
{
    "properties": {
      "msgType": "ack",
      "action": "model.patch",
      "timestamp": "YYYY-MM-DDTHH:mm:ss.sssZ",
      "correlationId": "<correlation id>",
      "version": 2,
      "target": "<target matches the target in the corresponding action>"
    },
    "body": {
      "success": false,
      "code": "<error code>",
      "details": "<error detail>"
    }
}
```

Error Handling

When an error occurs, error information is logged inside the Ability Platform (in Application Insights). If a response can be delivered to a device, an acknowledgment will be sent, if requested. For more details see [Acknowledgement Handling](../ack-handling.md).

Markdown Preview

Model.Patch Request

Overview

The Ability Platform supports sending info model patches using the `model.patch` API endpoint. The `model.patch` action message is sent to the platform by a directly connected device (i.e. Edge) via the IoTHub. All C2D messages (acks and notifications) sent in response to a `model.patch` action are sent in Device API v2 format.

The `model.patch` message is Idempotent. This means that sending several identical requests will result in the model being updated successfully in the Info Model Service:

- The first partial update of an existing object model (determined via objectId + model) will be successful.
- Subsequent partial updates of this object model will be rejected due to an object model version mismatch.

Prerequisites

- The requested info model **must** exist.

Message Format

```
{
  "properties": {
    "iothub-connection-device-id": "<authenticated device id>",
    "msgType": "action",
    "action": "model.patch",
    "version": 2,
    "correlationId": "<correlation id>",
    "objectId": "<GUID>",
```

```
    "model": "<model definition>",
    "ack": "<ack>",
    "target": "<connected device path>",
    "timeout": "<timeout>"
  },
  "body": "<body>"
}
```

Message Properties

Added by the IoT Hub

Property	Mandatory	Data Type	Description
iothub-connection-device-id	Mandatory	string	Device ID of the directly connected device.

Common for all v2 Requests

Property	Mandatory	Description
msgType	Mandatory	Must be equal to 'action'
action	Mandatory	Must be equal to 'model.patch'
version	Mandatory	Must be equal to 2
correlationId	Optional	Correlation ID. Should be included to track correlation of platform components. If not present, the DCS will generate a new one.

Property	Mandatory	Description
ack	Optional	Initializes to "none" value if the value is missing. Indicates an additional acknowledgement message to be sent to the device.For more details see Acknowledgement Handling.
target	Optional	If not provided, defaults to empty string. Represents the path to a connected device that wishes to receive the acknowledgement (if requested).
timeout	Optional	If not provided, defaults to a 300 integer. Represents the parameter that will be used to indicate the maximum number of seconds available for processing the D2C message until the timeout occurs. This excludes sending back response C2D messages.

Specific to a 'model.patch' Request

Property	Mandatory	Description
objectId	Mandatory	Source object identifier. Must be in GUID format.
model	Optional	Source model definition identifier (modelId). Case sensitive. If not provided, defaults to "abb.ability.device".

Message Body

The body must consist of a well-formed JSON compliant with the request payload format of a `PATCH /objects/{objectId}/models/{modelId}` operation.

Property	Mandatory	Data Type	Description

Property	Mandatory	Data Type	Description
version	Mandatory	integer	Must be equal to or higher than the version of the model existing in the IM database.

Validation Rules

- The DCS will validate all message properties as described above.
- The DCS will validate only the body property **version**.

Functionality

Description

- The object model is partially updated in the info model service.
- In case of version gaps, the DCS will perform a patch operation multiple times.
- After all patches are completed, the patched model version in the IM is equal to the request message version + 1.
- A model notification C2D message is not sent back.

:::tip NOTE Any platform events generated by the info model service related to this request are ignored by the DCS and are not propagated.
:::

- If requested, an acknowledgement C2D message is sent back to the directly connected device where it can also be propagated further if needed (based on target). For more details see Acknowledgement Handling.
    - The target is reused from the requested message.

Format of a Returned Acknowledgment C2D Message

Format of a Successful Acknowledgment

```
{
    "properties": {
```

```
    "msgType": "ack",
    "timestamp": "YYYY-MM-DDTHH:mm:ss.sssZ",
    "action": "model.patch",
    "correlationId": "<correlation id>",
    "version": 2,
    "target": "<target matches the target in the corresponding action>"
  },
  "body": {
     "success": true,
     "code": "ok",
     "details": "''",
     "objectId": "<objectId>",
     "model": "<model>",
     "version": "<version>"
  }
}
```

Description of body properties

Property	Mandatory	Data Type	Description
objectId	Mandatory	JSON object	ObjectId of patched model.
model	Mandatory	JSON object	Model identifier of patched model.
version	Mandatory	JSON object	Version of patched model.

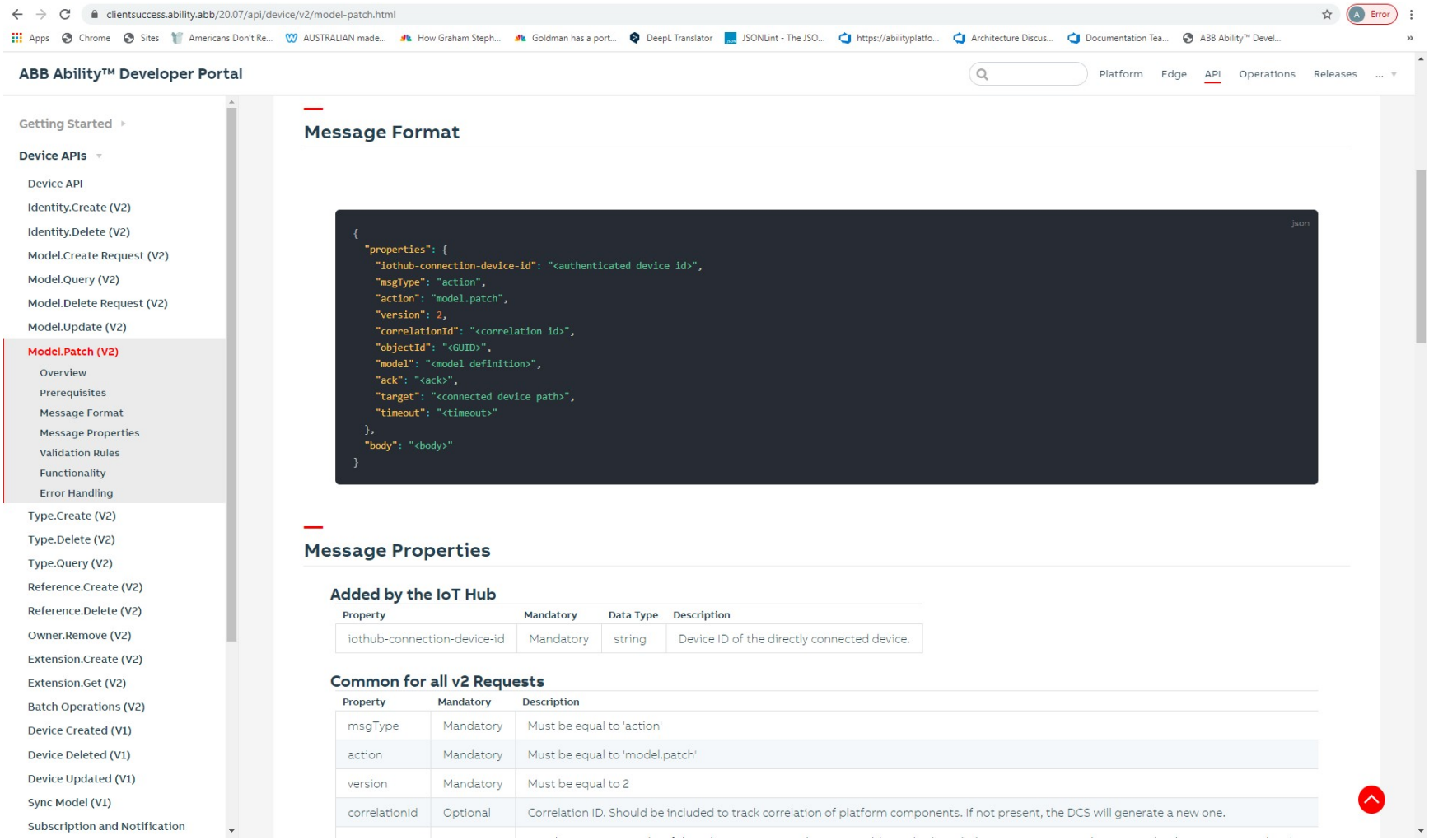
Format of a Failing Acknowledgment

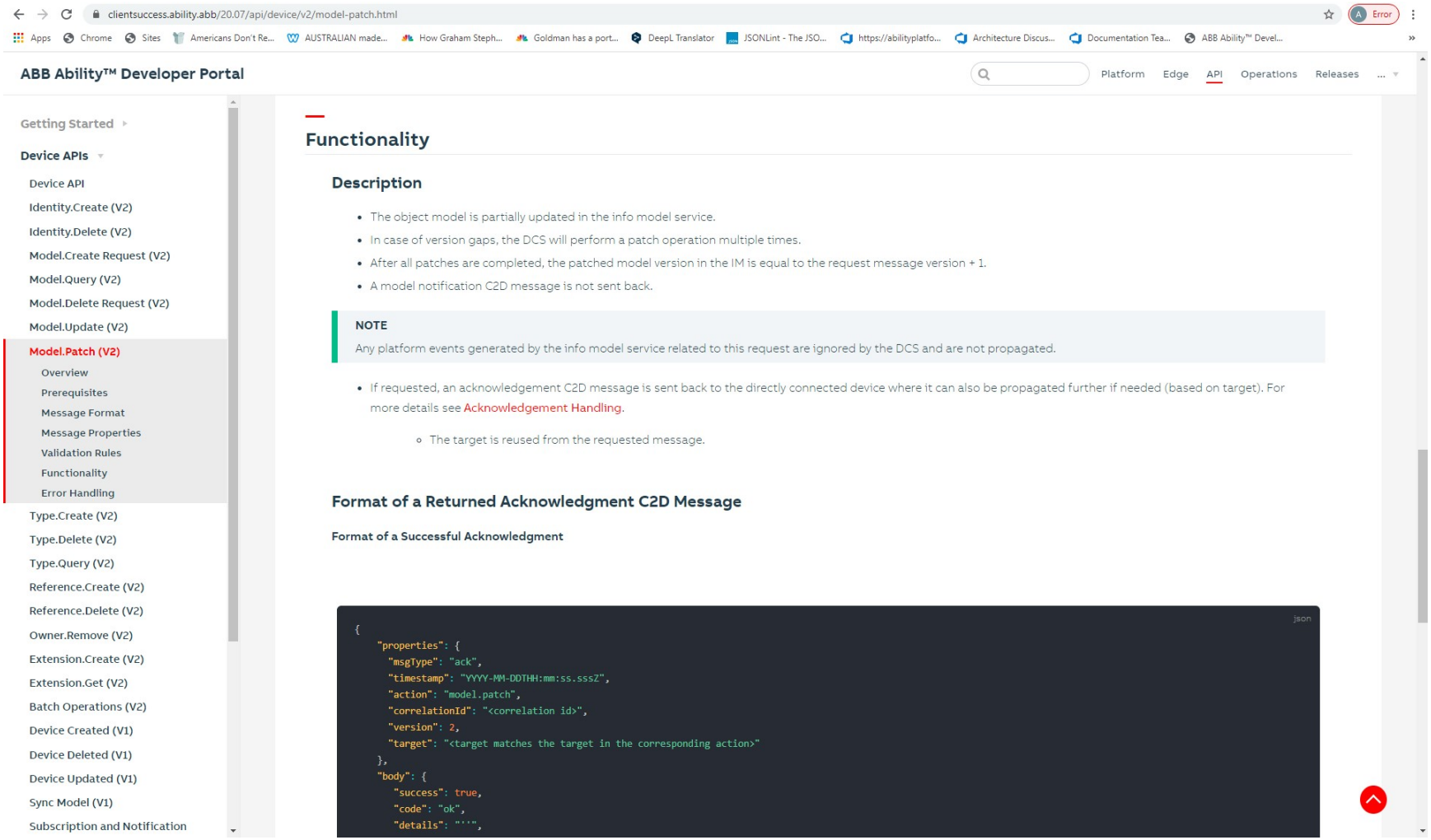
```
{
    "properties": {
      "msgType": "ack",
      "action": "model.patch",
      "timestamp": "YYYY-MM-DDTHH:mm:ss.sssZ",
      "correlationId": "<correlation id>",
      "version": 2,
      "target": "<target matches the target in the corresponding action>"
    },
    "body": {
      "success": false,
      "code": "<error code>",
      "details": "<error detail>"
    }
}
```

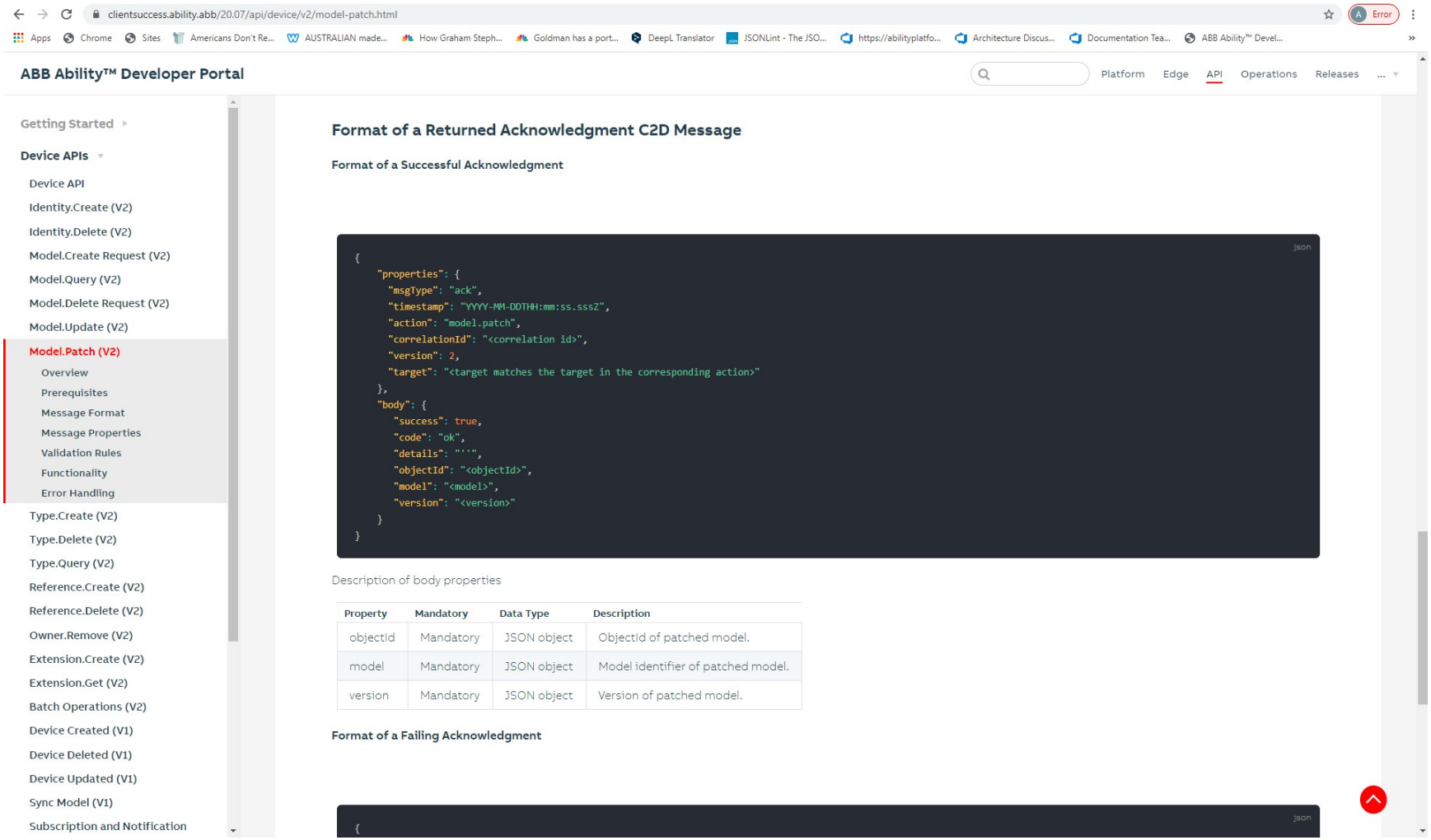
Error Handling

When an error occurs, error information is logged inside the Ability Platform (in Application Insights). If a response can be delivered to a device, an acknowledgment will be sent, if requested. For more details see Acknowledgement Handling.

Markdown User Interface




clientsuccess.ability.abb/20.07/api/device/v2/model-patch.html
ABB Ability™ Developer Portal
Platform
Edge
API
Operations
Releases
Getting Started
Device APIs
Device API
Identity.Create (V2)
Identity.Delete (V2)
Model.Create Request (V2)
Model.Query (V2)
Model.Delete Request (V2)
Model.Update (V2)
Model.Patch (V2)
Overview
Prerequisites
Message Format
Message Properties
Validation Rules
Functionality
Error Handling
Type.Create (V2)
Type.Delete (V2)
Type.Query (V2)
Reference.Create (V2)
Reference.Delete (V2)
Owner.Remove (V2)
Extension.Create (V2)
Extension.Get (V2)
Batch Operations (V2)
Device Created (V1)
Device Deleted (V1)
Device Updated (V1)
Sync Model (V1)
Subscription and Notification
Functionality
Description
The object model is partially updated in the info model service.
In case of version gaps, the DCS will perform a patch operation multiple times.
After all patches are completed, the patched model version in the IM is equal to the request message version + 1.
A model notification C2D message is not sent back.
NOTE
Any platform events generated by the info model service related to this request are ignored by the DCS and are not propagated.
If requested, an acknowledgement C2D message is sent back to the directly connected device where it can also be propagated further if needed (based on target). For more details see Acknowledgement Handling.
The target is reused from the requested message.
Format of a Returned Acknowledgment C2D Message
Format of a Successful Acknowledgment
json
{
"properties": {
"msgType": "ack",
"timestamp": "YYYY-MM-DDTHH:mm:ss.sssZ",
"action": "model.patch",
"correlationId": "<correlation id>",
"version": 2,
"target": "<target matches the target in the corresponding action>"
},
"body": {
"success": true,
"code": "ok",
"details": "''",


clientsuccess.ability.abb/20.07/api/device/v2/model-patch.html
Error
Apps
Chrome
Sites
Americans Don't Re...
AUSTRALIAN made...
How Graham Steph...
Goldman has a port...
DeepL Translator
JSONLint - The JSO...
https://abilityplatfo...
Architecture Discus...
Documentation Tea...
ABB Ability™ Devel...
ABB Ability™ Developer Portal
Platform
Edge
API
Operations
Releases
Getting Started
Device APIs
Device API
Identity.Create (V2)
Identity.Delete (V2)
Model.Create Request (V2)
Model.Query (V2)
Model.Delete Request (V2)
Model.Update (V2)
Model.Patch (V2)
Overview
Prerequisites
Message Format
Message Properties
Validation Rules
Functionality
Error Handling
Type.Create (V2)
Type.Delete (V2)
Type.Query (V2)
Reference.Create (V2)
Reference.Delete (V2)
Owner.Remove (V2)
Extension.Create (V2)
Extension.Get (V2)
Batch Operations (V2)
Device Created (V1)
Device Deleted (V1)
Device Updated (V1)
Sync Model (V1)
Subscription and Notification
Format of a Returned Acknowledgment C2D Message
Format of a Successful Acknowledgment
json
{
"properties": {
"msgType": "ack",
"timestamp": "YYYY-MM-DDTHH:mm:ss.sssZ",
"action": "model.patch",
"correlationId": "<correlation id>",
"version": 2,
"target": "<target matches the target in the corresponding action>"
},
"body": {
"success": true,
"code": "ok",
"details": "''",
"objectId": "<objectId>",
"model": "<model>",
"version": "<version>"
}
}
Description of body properties
Property
Mandatory
Data Type
Description
objectId
Mandatory
JSON object
ObjectId of patched model.
model
Mandatory
JSON object
Model identifier of patched model.
version
Mandatory
JSON object
Version of patched model.
Format of a Failing Acknowledgment
json
{