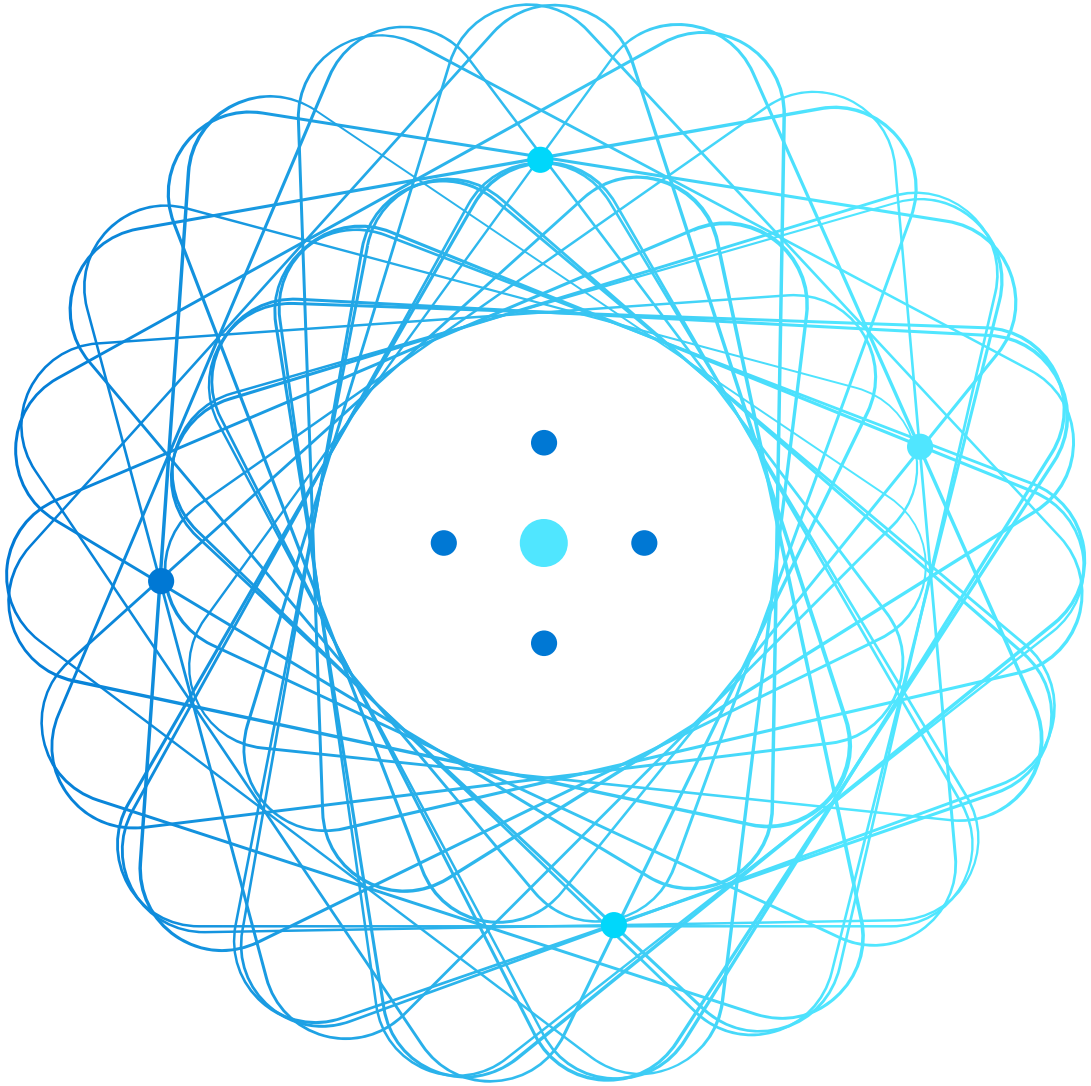
Microsoft Azure

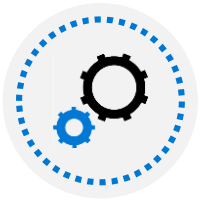
AZ-220T01
Module 8: Device management



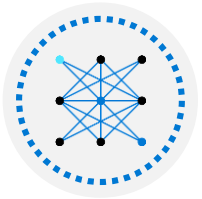
Lesson 1: Learning objectives

Module 8 – Learning objectives

Describe the most common device management patterns and configuration best practices



Describe when and how to use device twins and direct methods to implement device management



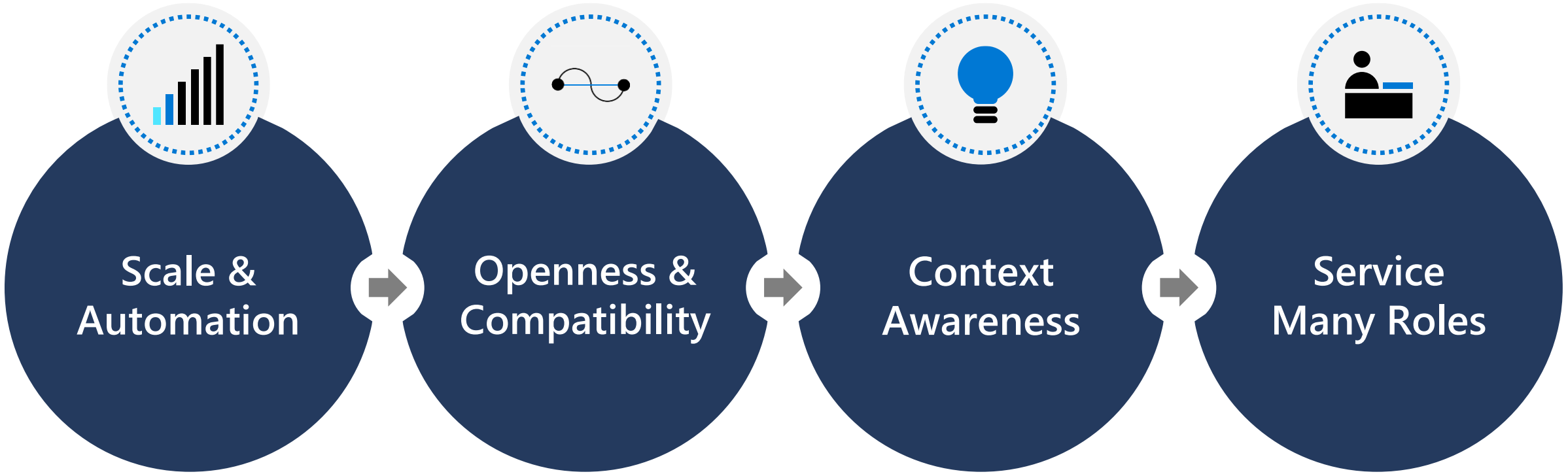
Implement device management for various patterns using device twins and direct methods

Implement device management at scale using automatic device management and jobs

Lesson 2: Introduction to IoT device management

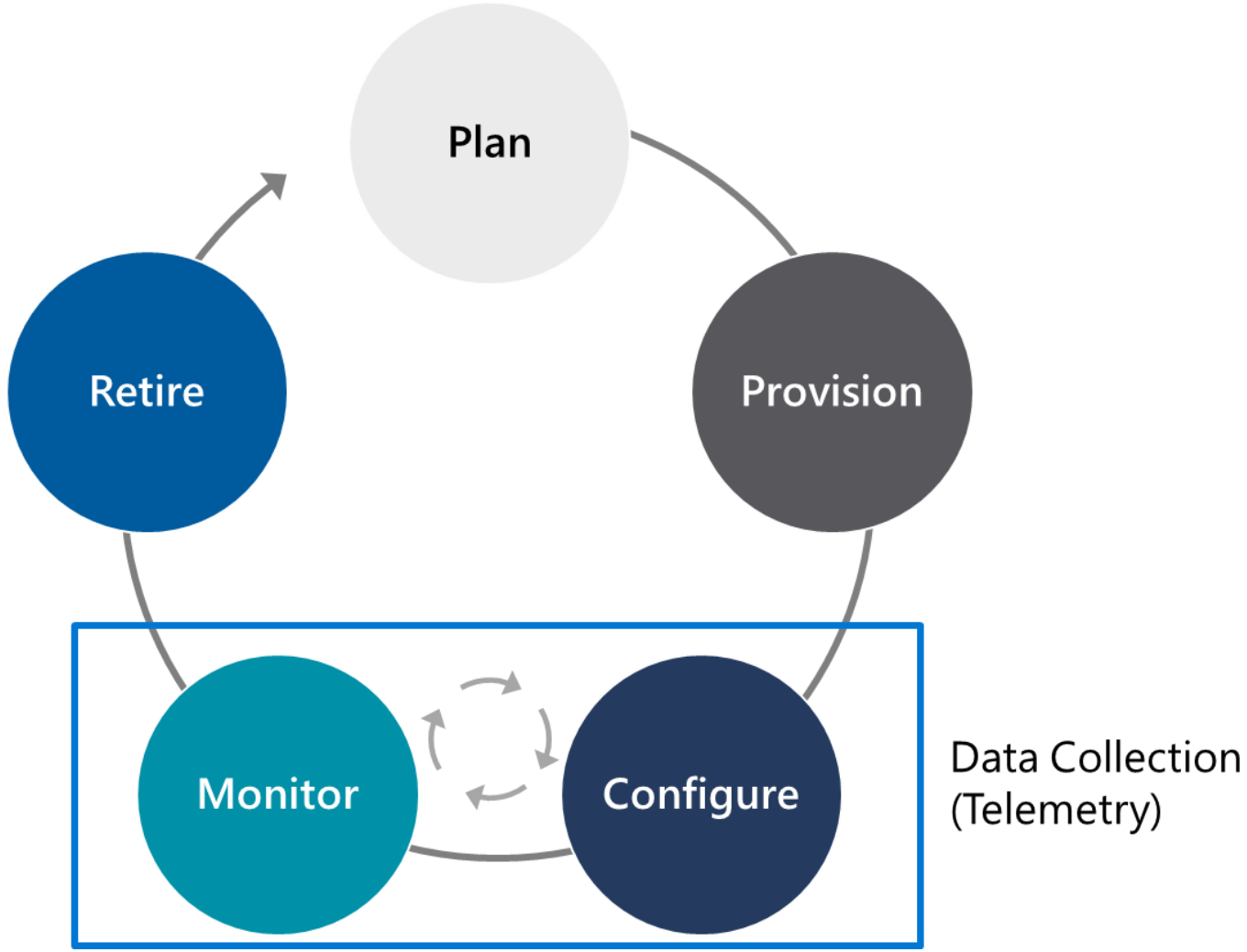
What is device management?

Device management principles:



What is device management?

Device Lifecycle



Device management patterns

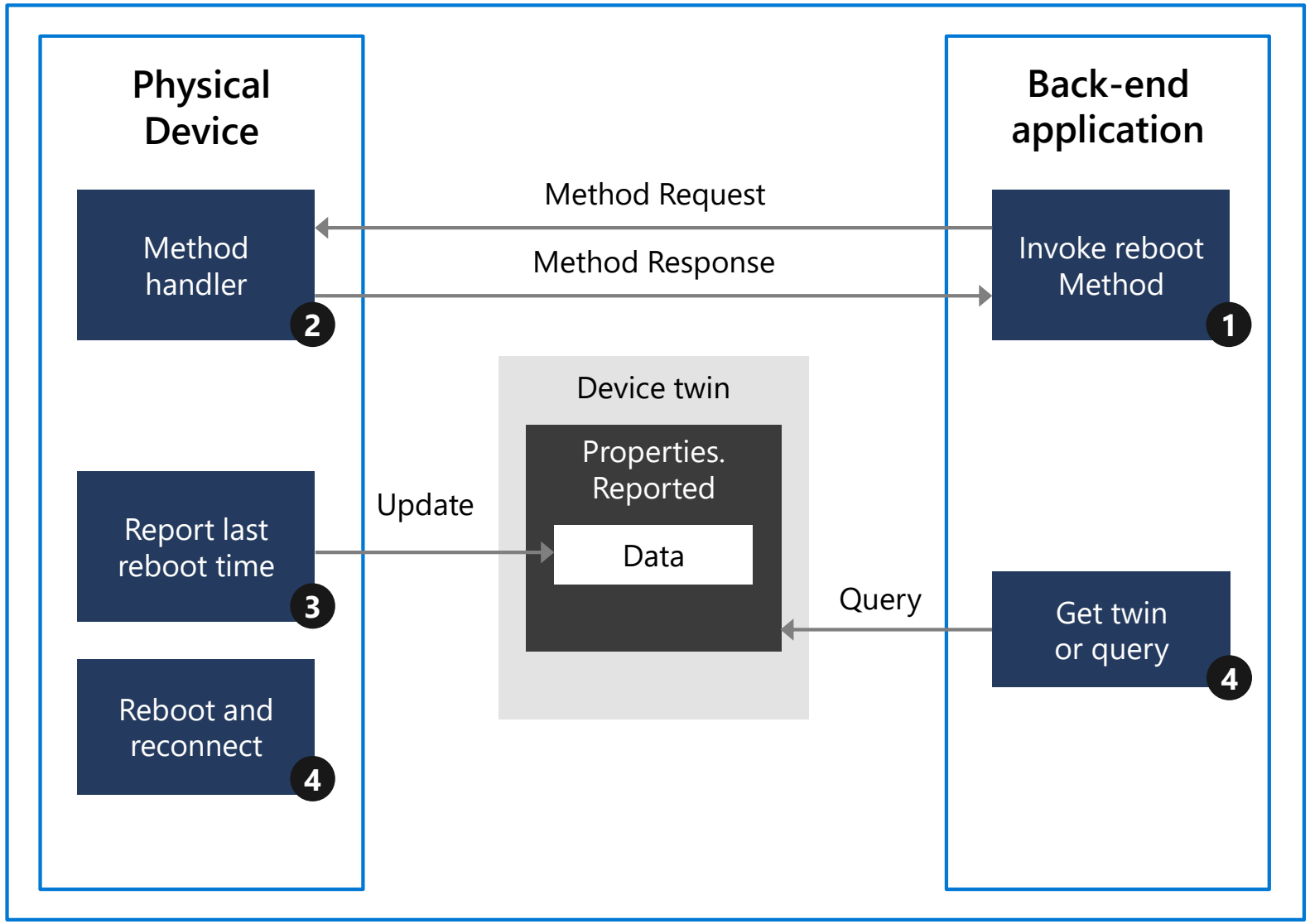
Reboot

Factory reset

Configuration

Firmware update

Reporting progress and status



Device management patterns

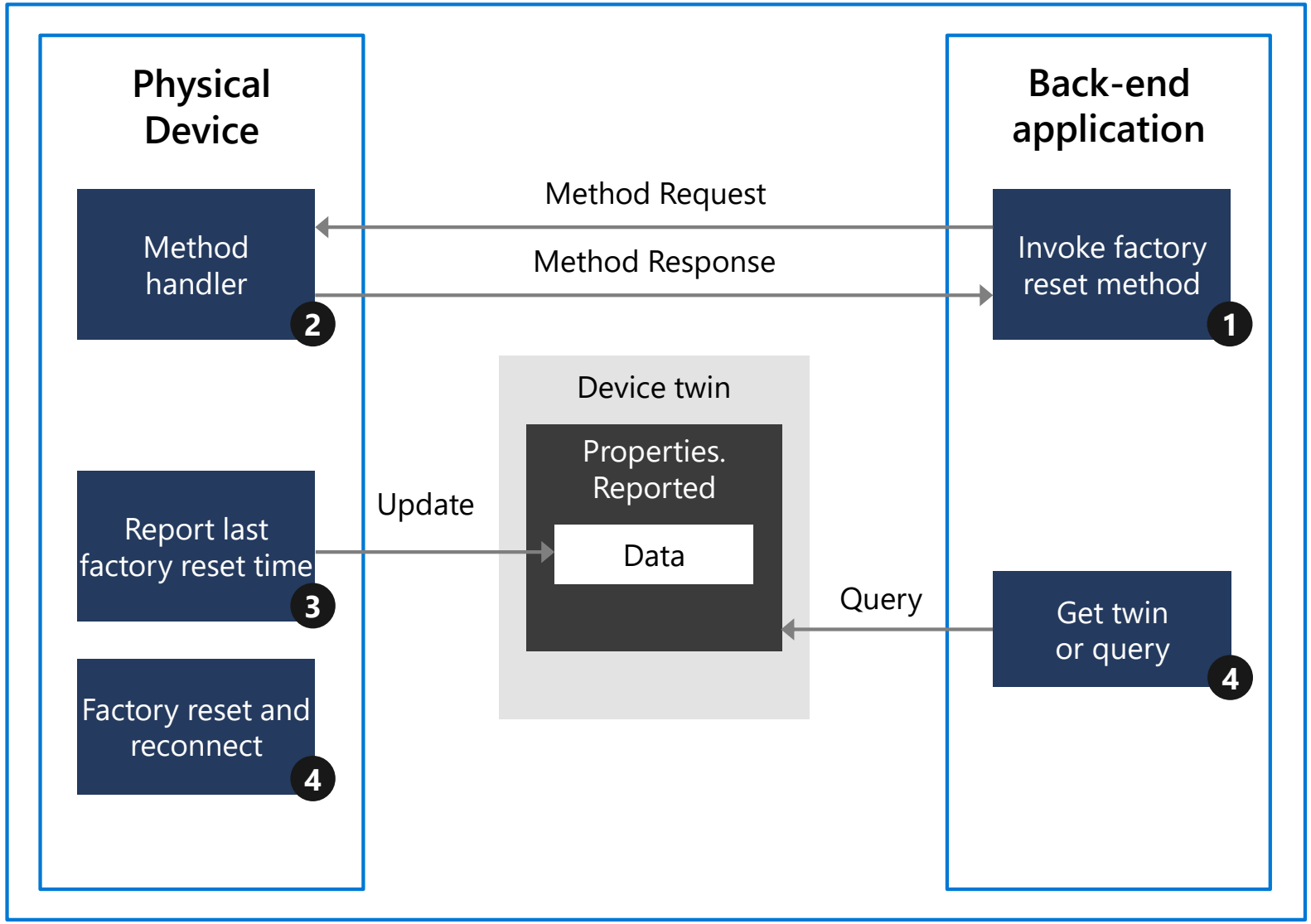
Reboot

Factory reset

Configuration

Firmware update

Reporting progress and status



Device management patterns

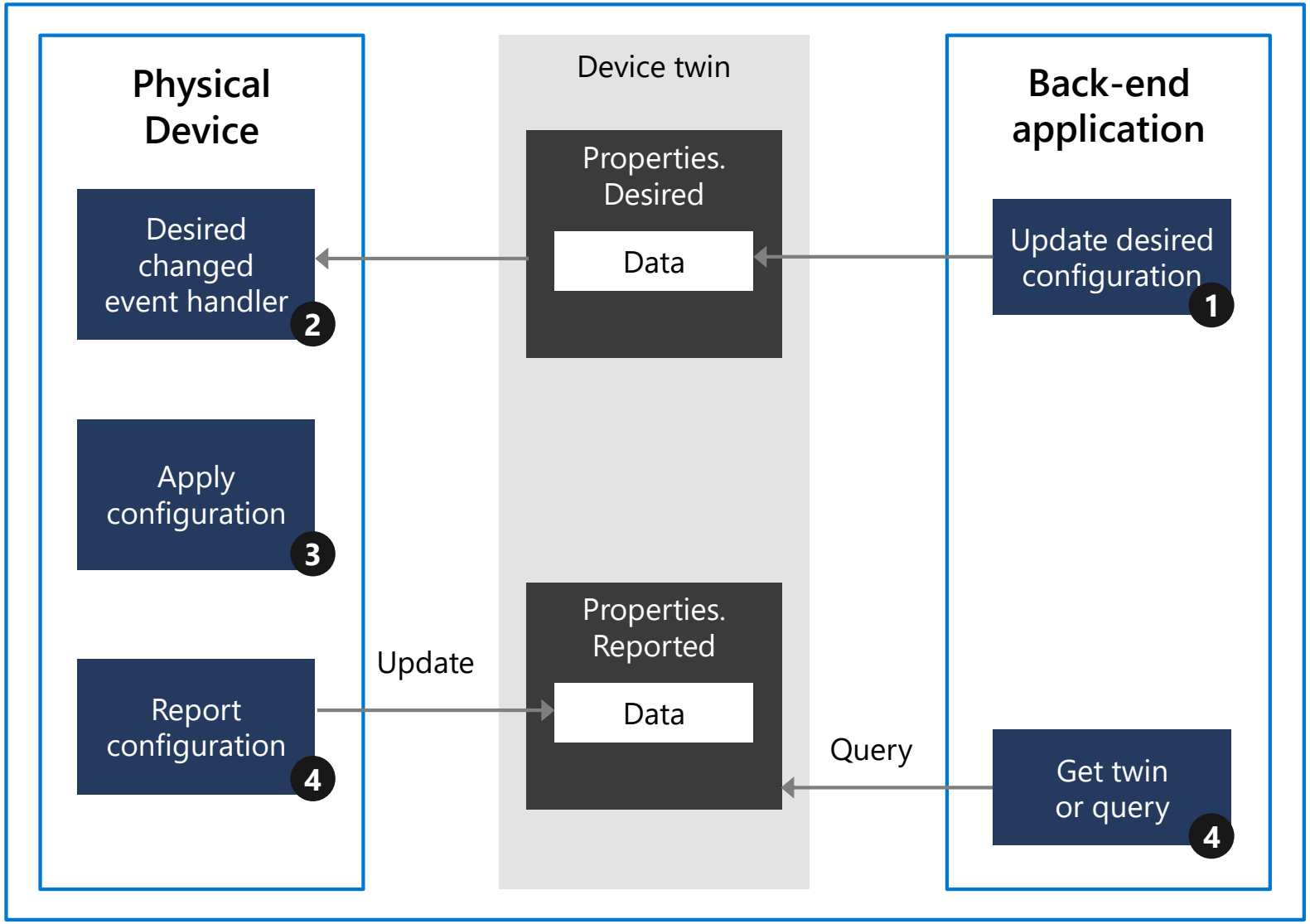
Reboot

Factory reset

Configuration

Firmware update

Reporting progress and status



Device management patterns

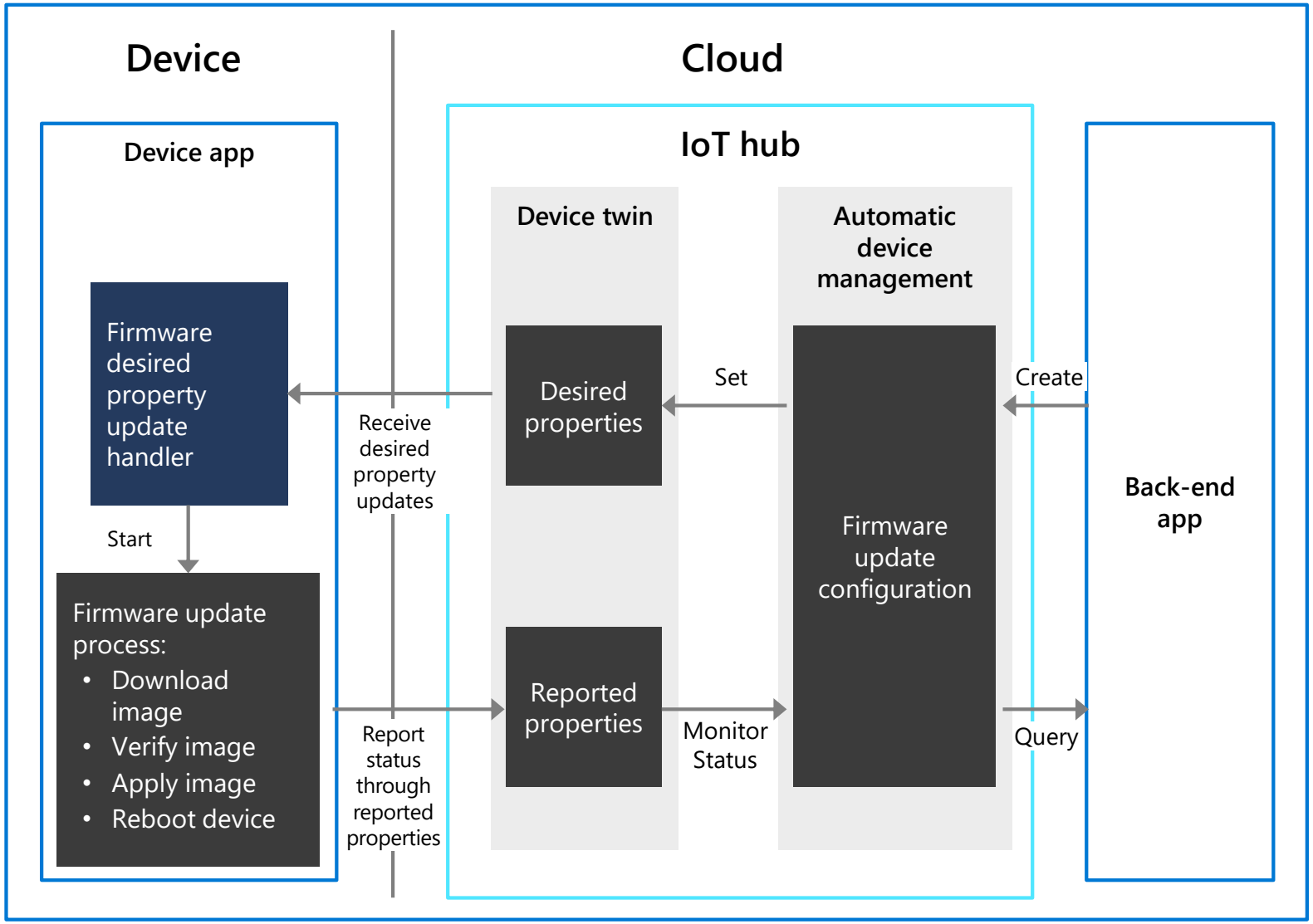
Reboot

Factory reset

Configuration

Firmware update

Reporting progress and status



Device management patterns

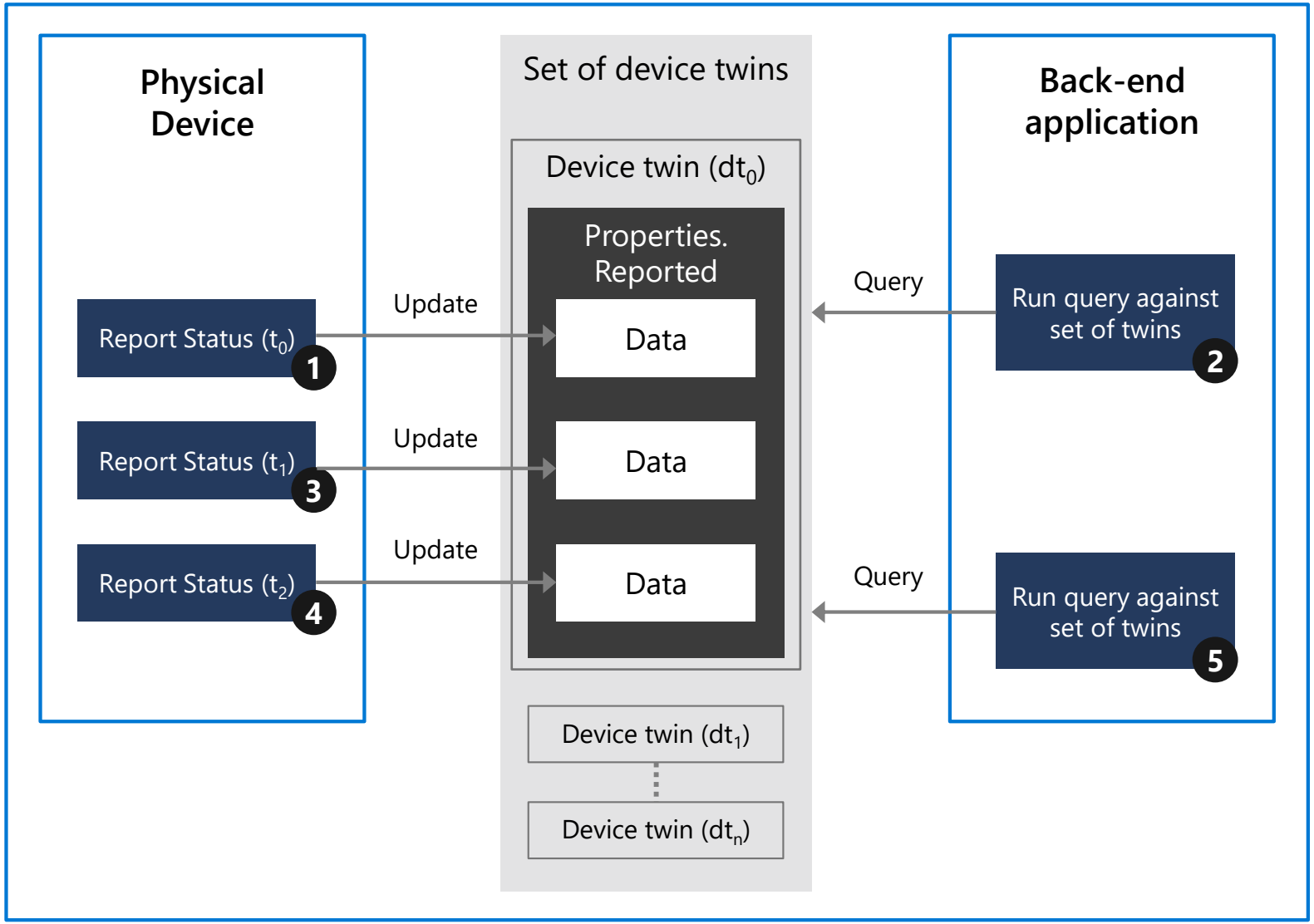
Reboot

Factory reset

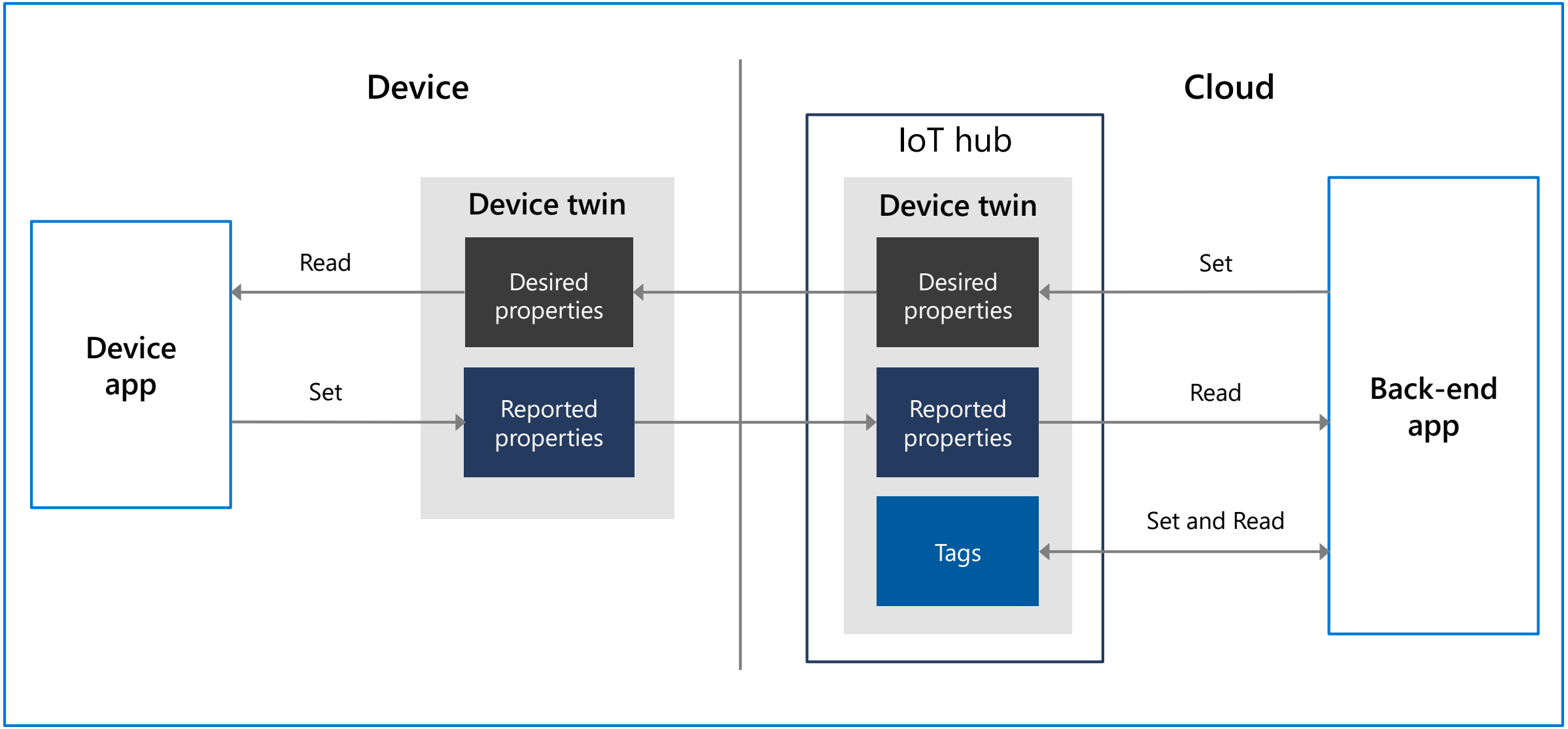
Configuration

Firmware update

Reporting progress and status



Device configuration with device twins



Device configuration with device twins

Device twin properties

Back-end operations

Device operations

Device twin metadata

Optimistic concurrency

Device reconnection flow

Device configuration with device twins

Device twin properties

Back-end operations

Device operations

Device twin metadata

Optimistic concurrency

Device reconnection flow

Device configuration with device twins

Device twin properties

Back-end operations

Device operations

Device twin metadata

Optimistic concurrency

Device reconnection flow

Device configuration with device twins

Device twin properties

Back-end operations

Device operations

Device twin metadata

Optimistic concurrency

Device reconnection flow

Device configuration with device twins

Device twin properties

Back-end operations

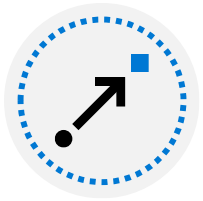
Device operations

Device twin metadata

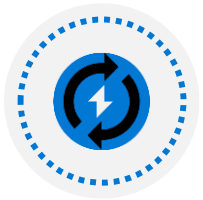
Optimistic concurrency

Device reconnection flow

Direct methods: Introduction



Direct methods **– requests from the cloud to a device, executing code directly on the target**

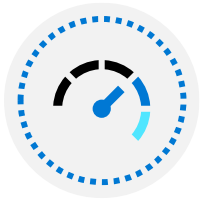


Features:

Each call targets a single device or module instance

Can be used by anyone with appropriate IoT Hub permissions

Follow a request-response pattern for immediate feedback



Lifecycle:

Called by a back-end application through an HTTPS URL pattern on the IoT Hub

Translated to MQTT or AMQP on the device side

Reply received from the device sent directly back to the back-end application

Direct methods: Sample back-end service call

```
curl -X POST \
  https://iothubname.azure-devices.net/twins/myfirstdevice/methods?api-version=2018-06-30 \
  -H 'Authorization: SharedAccessSignature sr=iothubname.azure-
devices.net&sig=x&se=x&skn=iothubowner’ \
  -H 'Content-Type: application/json’ \
  -d ‘{
    "methodName": "reboot",
    "responseTimeoutInSeconds": 200,
    "payload": {
    "input1": "someInput",
    "input2": "anotherInput"
    }
  }'
```

Direct methods: Return value to the back-end

HTTP Status Code

Standard HTTP headers:

Etag

Request ID

Content type

Content encoding

JSON Body

```
{ “status” : 201, “payload” : { … } }
```

Direct methods: Device-side view

Handle a direct method on a device: MQTT

```
IoT Hub posts to MQTT topic: $iothub/methods/POST/{method name}/?$rid={request id}

Device posts response to $iothub/methods/res/{status}/?$rid={request id}
```

Handle a direct method on a device: AMQP

```
Device creates a receive link for IoT Hub to use against the IoT Hub at
amqps://{hostname}:5671/devices/{deviceId}/methods/deviceBound

Device creates a send link against IoT Hub at the same endpoint
```

Comparing device management approaches

	Direct Method Call	Device Twins	Cloud-to-Device Messages
Scenario	Requires immediate confirmation	Long-running desired state configuration	One-way notifications
Data flow	Two-way with immediate response	One-way. The device app receives a notification with the property change	One-way. The device app receives the message
Durability	Disconnected devices are not contacted. The solution back end is notified that the device is not connected	Property values are preserved in the device twin. Device will read it at next reconnection. Property values are retrievable with the IoT Hub query language	Messages can be retained by IoT Hub for up to 48 hours
Targets	Single device using deviceId, or multiple devices using jobs.	Single device using deviceId, or multiple devices using jobs	Single device by deviceId
Size	Payload maximum is 128 KB	Desired properties maximum is 32 KB	Up to 64 KB messages
Frequency	High	Medium	Low
Protocol	MQTT or AMQP	MQTT or AMQP	MQTT, AMQP, HTTPS

Lesson 3: Manage IoT and IoT Edge devices

Device management tools

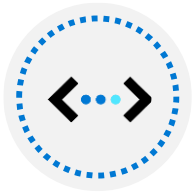
IoT Hub (Azure portal)

Azure CLI

Visual Studio Code

SDKs

Device management using the IoT extension for Azure CLI

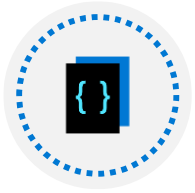


Direct Methods

Device twin desired properties

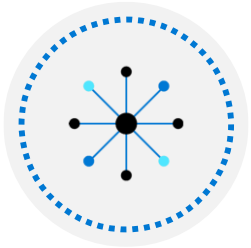
Device twin reported properties

Device twin tags

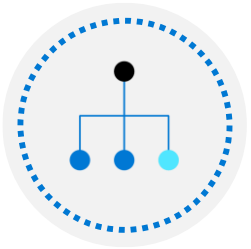


Device twin queries

Device management using the Azure IoT tools for VS Code

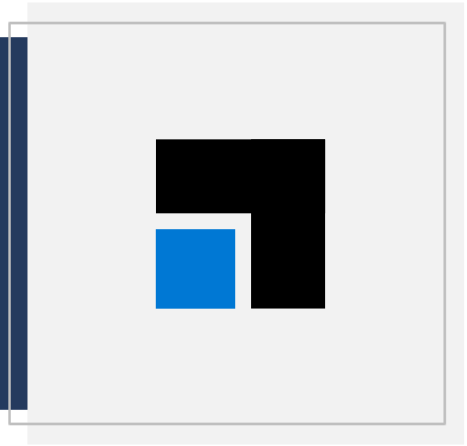


Access Your IoT Hub and Devices

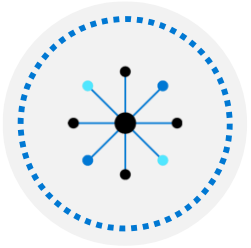


Access Device Management Commands

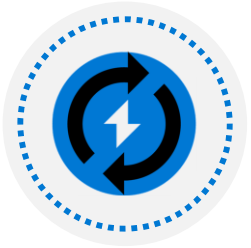
Lesson 4: Device management at scale



Schedule jobs on multiple devices: Concepts



IoT Hub *jobs* – allow calling direct methods or setting device twin properties across a large number of devices



Job lifecycle:

Call service URL to create the job

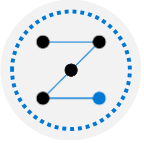
Call different URL to query on the status of the job

Schedule jobs: Direct methods example

```
PUT /jobs/v2/<jobId>?api-version=2018-06-30
Authorization: <config.sharedAcessSignature>
Content-Type: application/json;ccharset=utf-8
    {
        “jobId”: “<jobId>”,
        “type”: “scheduleDeviceMethod”,
        “cloudToDeviceMethod”:
            {
                “methodName”: “<methodName>”,
                “payload”: <payload>,
                “respondTimeoutInSeconds”: methosTimeoutSeconds
            },
        “queryCondition”: “<queryOrDevice>”, //query condition
        “startTime”: <jobStartTime>, //as an Iso-8601 date string
        “maxecutionTimeInSeconds”: <maxecutionTimeInSecond>
    }
```

Automatic device management: Introduction

Automatic device management – bulk assignment of device twin data (and thus device configuration)



Requires the Standard tier

Reports the status of the deployment including possible custom metrics by queries against device twin reported properties

Can be done with the Portal, the CLI, or the IoT Hub Service SDK

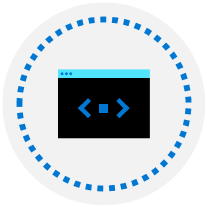
Identify target devices using a tag query

Runs soon after creation, then every five minutes afterwards to handle changes in the target device list

Ordered priority of configurations

Automatic device management: Modifications

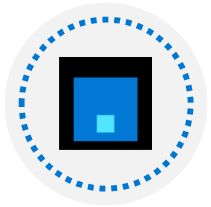
Device comes into the target list: Appropriate highest-priority configuration is applied



Device leaves the target list:

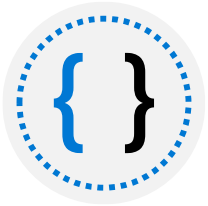
If a lower priority match exists: Configuration values are removed and next appropriate priority configuration is applied

If a lower priority match does not exist: Configuration values are removed, no other changes

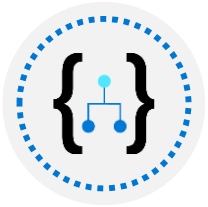


Configuration is deleted: No longer applies, but values are not removed

Device configuration best practices



Implement device twins

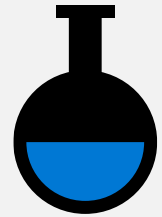


Organize devices using device twin tags

Implement automatic device configurations

Use the Device Provisioning Service

Lesson 5: Module labs



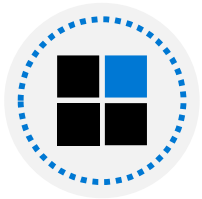
Module 8 labs

Lab 15: Remotely monitor and control devices with Azure IoT hub

You will create a back-end service app to listen for the telemetry

You will implement a direct method, to communicate settings to the IoT device

You will implement device twins functionality, to manage IoT device properties



Lab 16: Automate IoT device management with Azure IoT hub

You will write code for a simulated device that will implement a firmware update

You will test the firmware update process on a single device using Azure IoT Hub automatic device management

Lesson 6: Module 8 review questions

Module review: Question 8.1

What are the two primary technologies that can be used to implement IoT device management?

Answer A:
Machine Learning and Artificial Intelligence.

Answer B:
Direct Methods and Device Twins.

Answer C:
Azure Functions and Logic Apps.

Module review: Question 8.2

Which of the following choices is an example of an on-device operation that could occur during the device management process?

Answer A:
Observe desired properties.

Answer B:
Receive twin notifications.

Answer C:
Replace desired properties.

Module review: Question 8.3

Direct methods are recommended for which of the following device management patterns?

Answer A:
Reboot and Factory Reset.

Answer B:
Configuration and Firmware Update.

Answer C:
Reporting Progress and Status.

Module review: Question 8.4

Which of the following tasks can be accomplished using the Azure portal (IoT Hub resource), Azure CLI extension for IoT, and the Azure IoT Hub Device SDK?

Answer A:
Automatic device management.

Answer B:
Update a device twin property.

Answer C:
Update a device twin desired property.

Module review: Question 8.5

When using Azure CLI for device management, which of the following command parameters would a developer use to instruct a device to take an action?

Answer A:
device-twin update

Answer B:
device-twin show

Answer C:
invoke-device-method

Module review: Question 8.6

When using IoT Hub automatic device management, which approach to device management is implemented?

Answer A:
Either device twins or direct methods.

Answer B:
Only direct methods.

Answer C:
Only device twins.

Module review: Question 8.7

When using IoT Hub jobs for device management, which approach to device management can be implemented?

Answer A:
Either device twins or direct methods.

Answer B:
Only direct methods.

Answer C:
Only device twins.

Module review: Question 8.8

When implementing device management using device twins, which two actions are performed on the device-side?

Answer A:
Handles desired property changed notifications and updates reported properties.

Answer B:
Handles reported property changed notifications and updates reported properties.

Answer C:
Updates desired and reported properties.

Module review: Question 8.9

Which of the following statements about automatic device management is correct?

Answer A:
Automatic device management requires the Standard tier of the IoT Hub service.

Answer B:
Automatic device management uses a JSON document called a Manifest.

Answer C:
Automatic device management works with any tier of the IoT Hub service.