



# Cloud Classifier

FlexContainer & FlexContainerInstance를 사용한 구름 분류



# Table of Contents

---

I . Outline

II . Architecture

III . Using flow

IV . Resource

V . demo

# CHAPTER

1

Outline

# 1. Problem



## ● 해결하고자 하는 문제점

- ✓ 자동화하기 힘든 기상 요소는 관측을 통해 해결해야 하는데 관측이 힘든 지역에 인간 관찰자를 둘 수 없으니, TinyML을 사용하여 구름 및 구름의 유형 관측을 자동화할 수 있음
- ✓ TinyML은 평균 1milliwatt 이하의 저에너지 시스템에서 구현되어 임베디드 장치에서 실행하는 머신러닝으로 이번 문제점을 해결하기에 용이함
- ✓ 해당 프로젝트를 통해 심해와 같이 인간 관찰자를 쉽게 배치할 수 없는 지역에도 TinyML을 사용하여 기계 관찰자를 배치할 수 있을 것을 기대함

## 2. Solution



### 구체적인 해결 방안

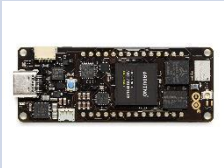

- ✓ 저전력 카메라와 소형 컨트롤러 보드에서 동작 가능한 TinyML 기반의 구름 모양 분류 알고리즘을 사용한 솔루션
- ✓ 한 시간마다 하늘의 사진을 찍어 TinyML 알고리즘에게 전송하면, 구름의 유형을 추측하기 위해 이를 사용하고, 분류를 시행한 후 추측 결과와 이미지를 원격 서버로 전송함
- ✓ 소형 컨트롤러 보드는 다른 이미지와 정보들을 구름 이미지를 추측하기 위한 데이터 셋으로 기록함  
(추가로 이 데이터 셋을 사용하여 TinyML의 정확도를 강화하는데 사용될 수 있음)
- ✓ 구름의 유형 추측, 추측 정보 전달, 데이터셋 생성을 수행한 이후 소형 컨트롤러 보드는 배터리를 아끼기 위해 저전력 모드 상태가 됨(한 시간 뒤에 다시 소형 컨트롤러 보드는 위와 동일하게 동작)

# 3. Hardware



## Required

### ✔ 필요 하드웨어

필요 하드웨어	사진	개수
Arduino Portenta H7		1
Arduino Portenta Vision Shield		1
Type C to Type C Cable		1
tripod		1

## 4. Software



- Required

✔ 필요 소프트웨어



Arduino Portenta H7에 python3  
코드를 이식하기 위한 소프트웨어



OpenMV에 사용될 Arduino  
Portenta H7 Upload Driver를 설  
치하기 위한 소프트웨어

CHAPTER

2

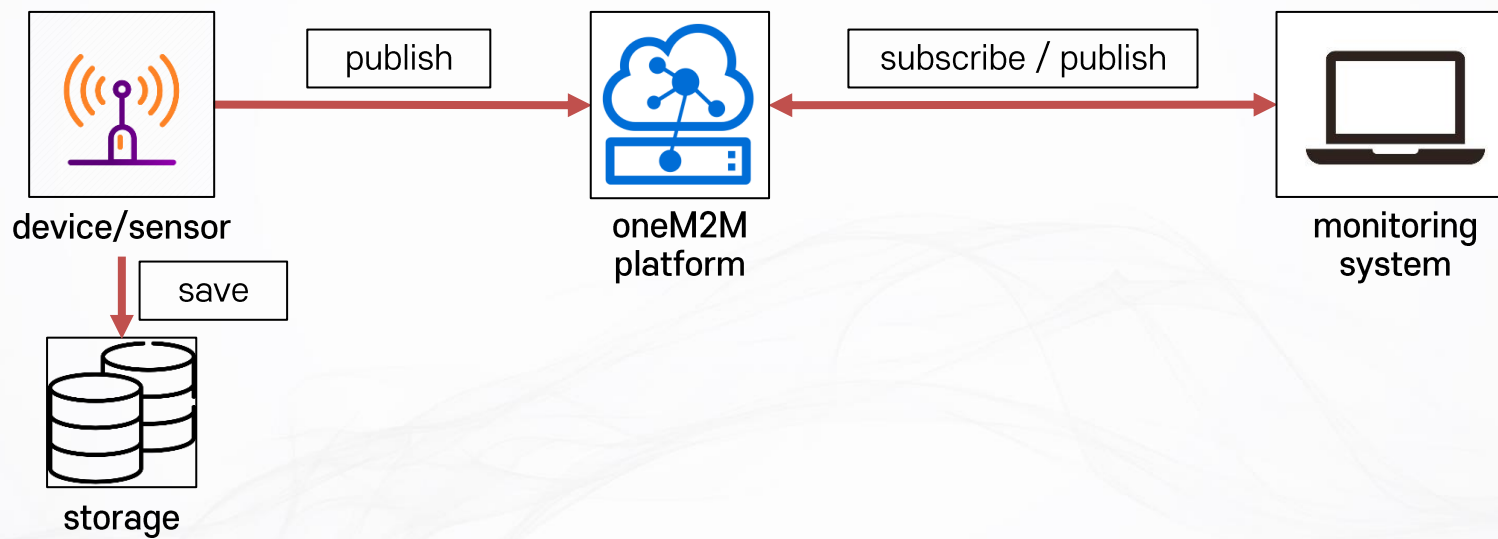
**Architecture**



# 1. Architecture



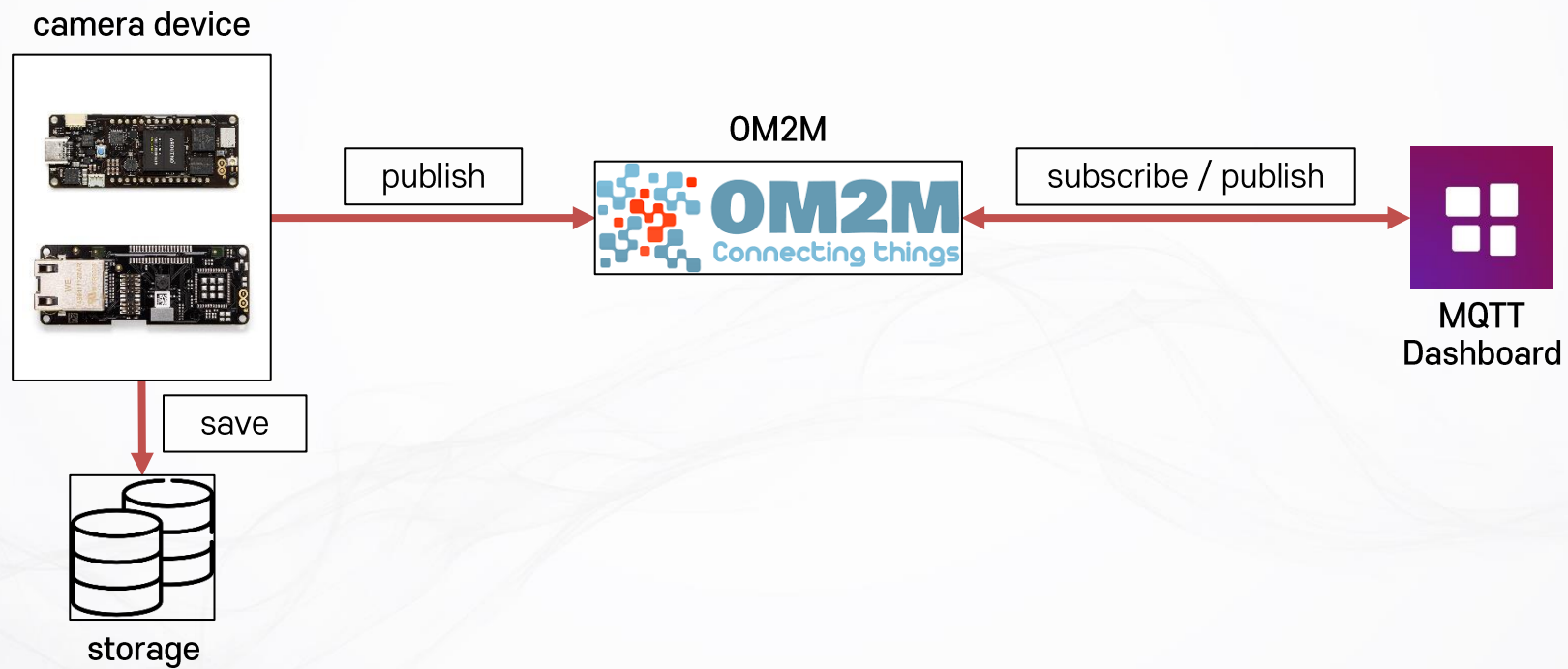
- logical 구조



# 1. Architecture



- physical 구조



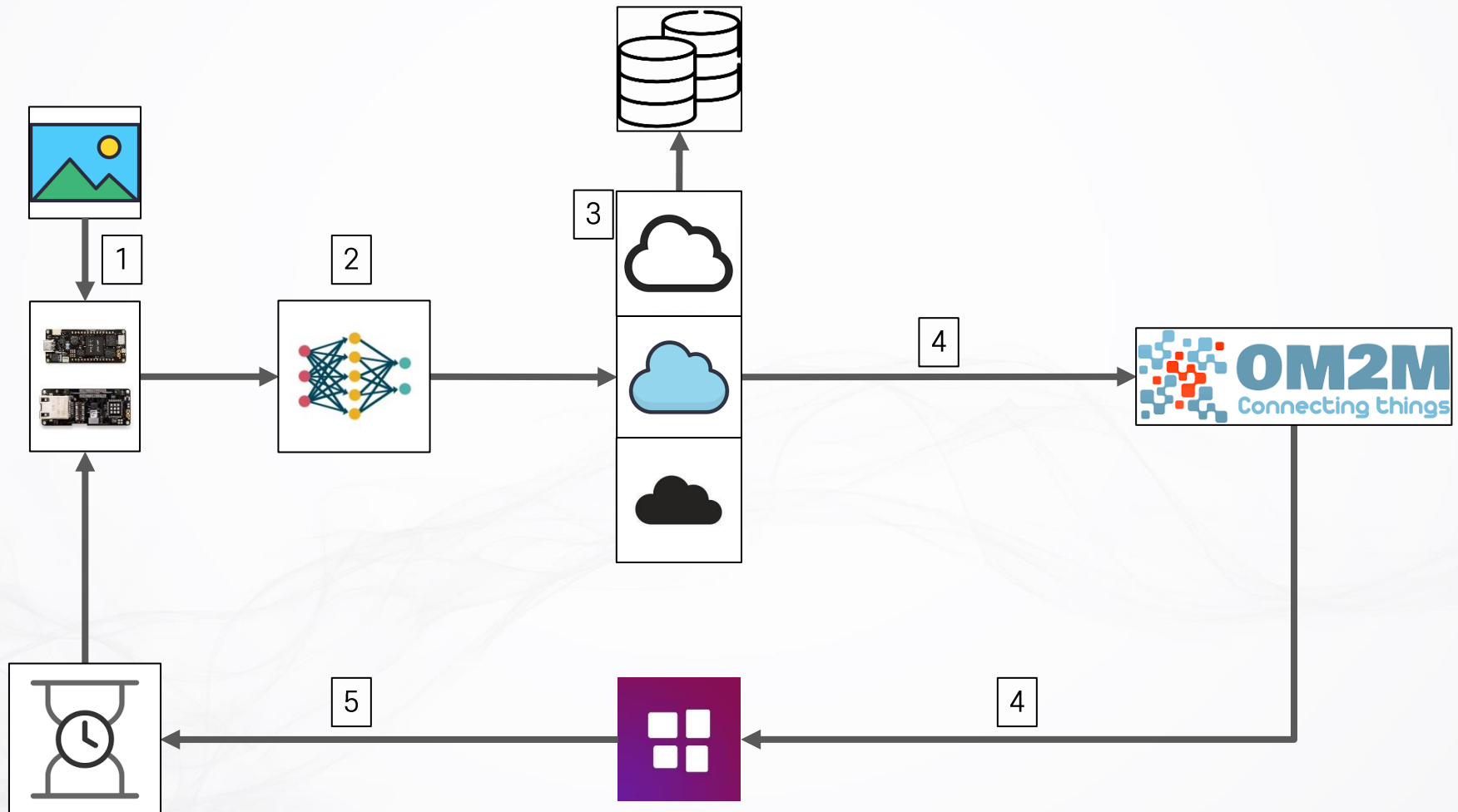
CHAPTER

3

Using Flow

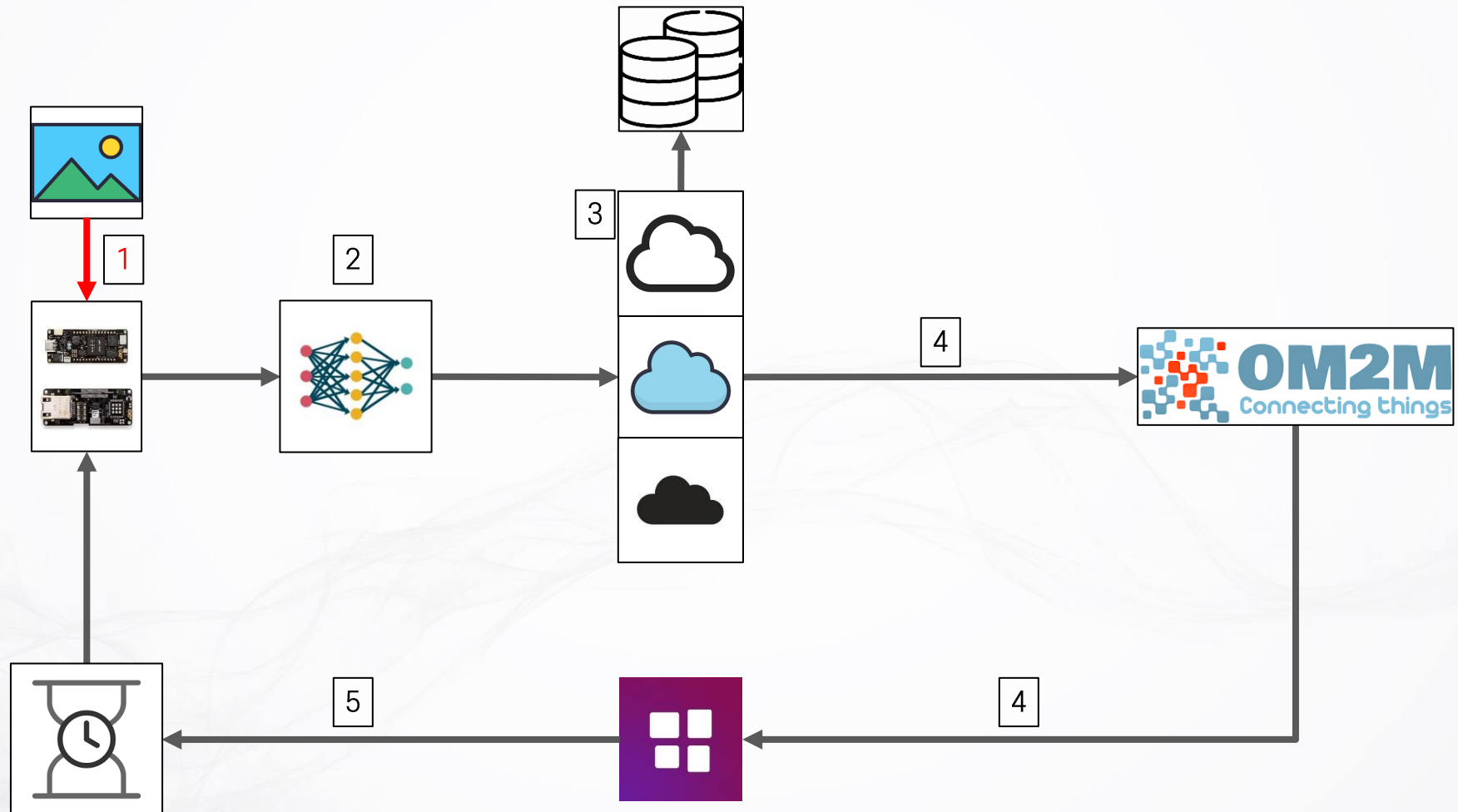
# 1. Entire View

- Using Flow의 전체적인 모습



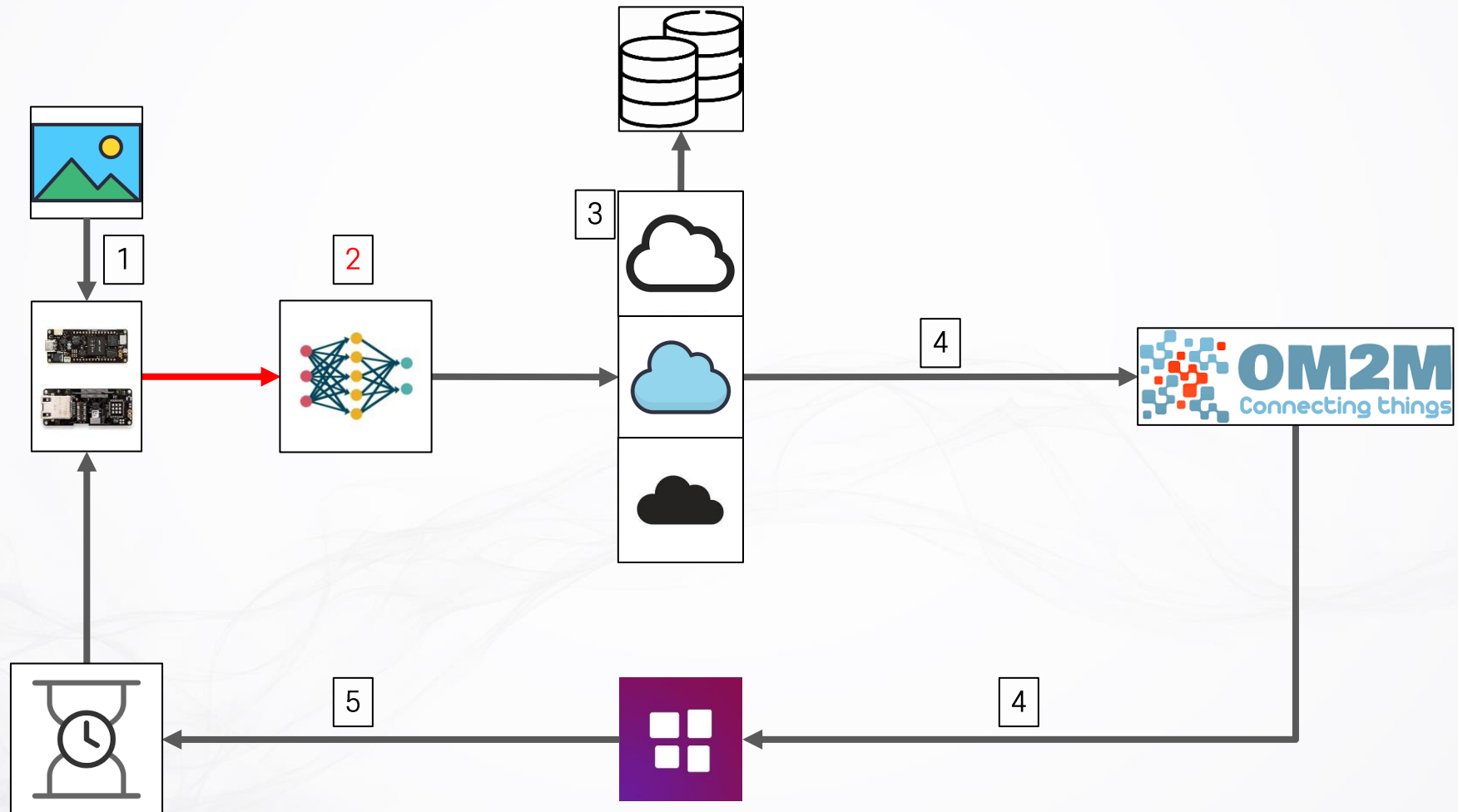
## 2. Steps

### S#1 > Portenta H7 takes an image using the vision shield



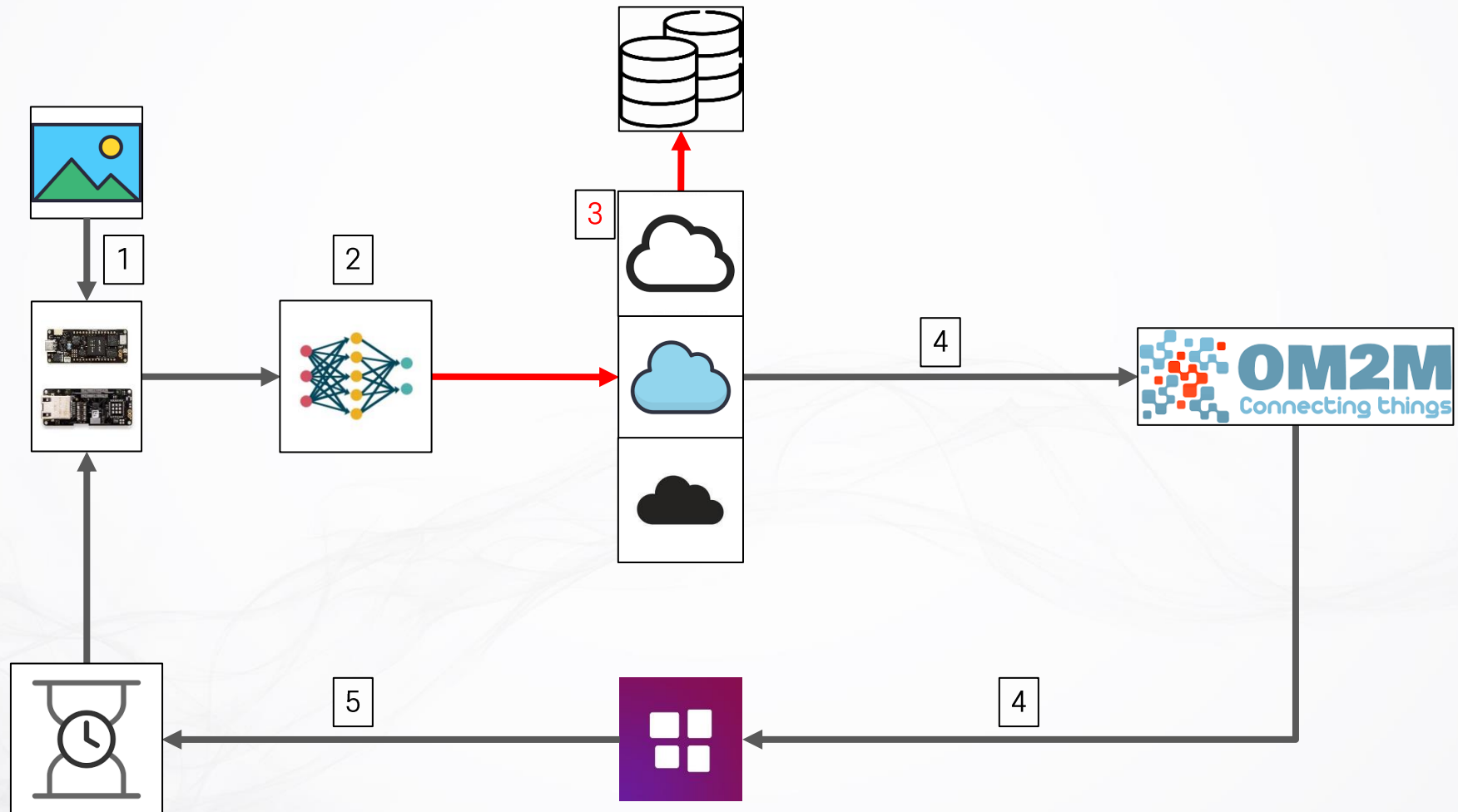
## 2. Steps

S#2 ➤ Image is fed to a TinyML model, which predicts the cloud type



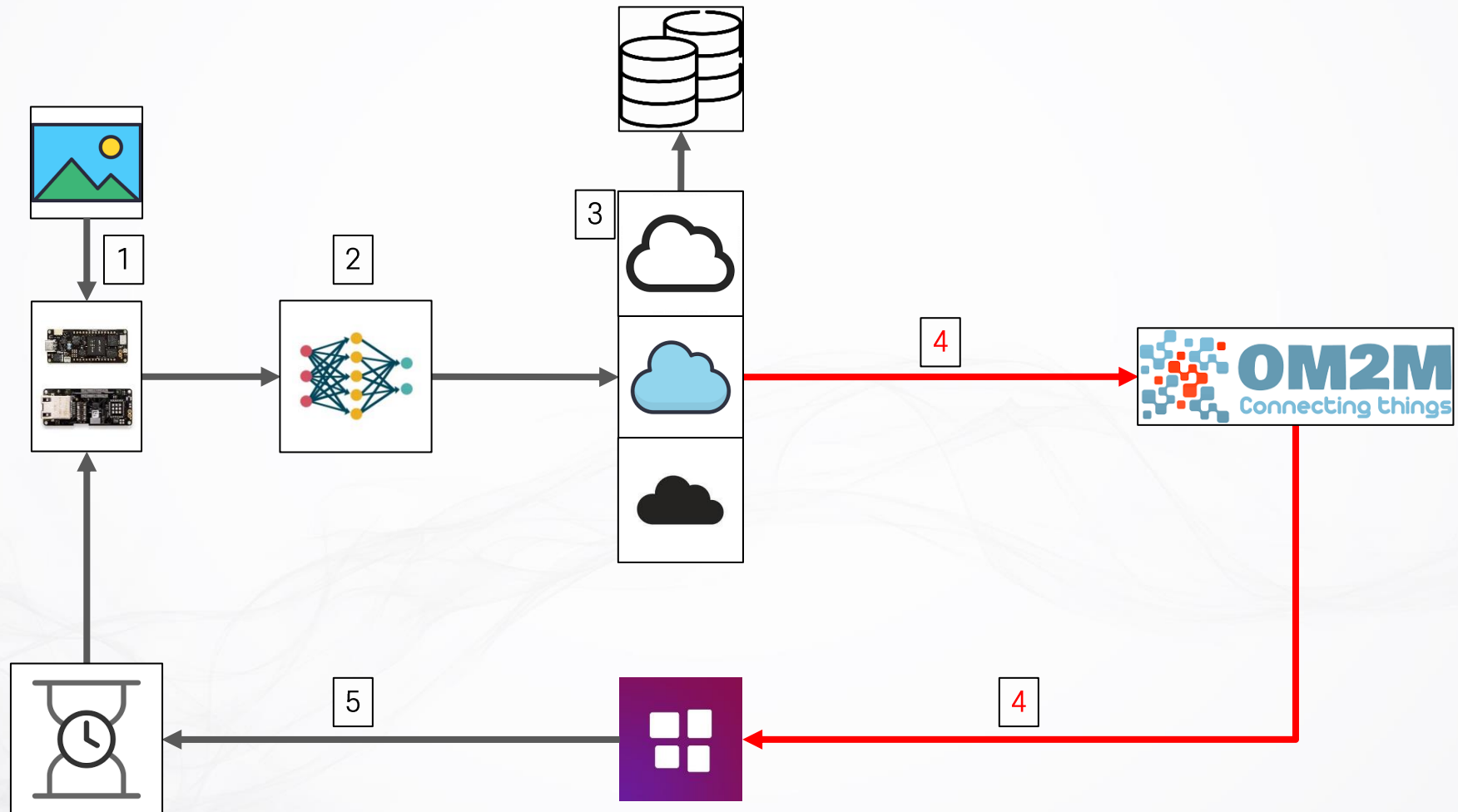
## 2. Steps

S#3 ➤ Informations (cloud image/type and other) are recorded to prepare the dataset



## 2. Steps

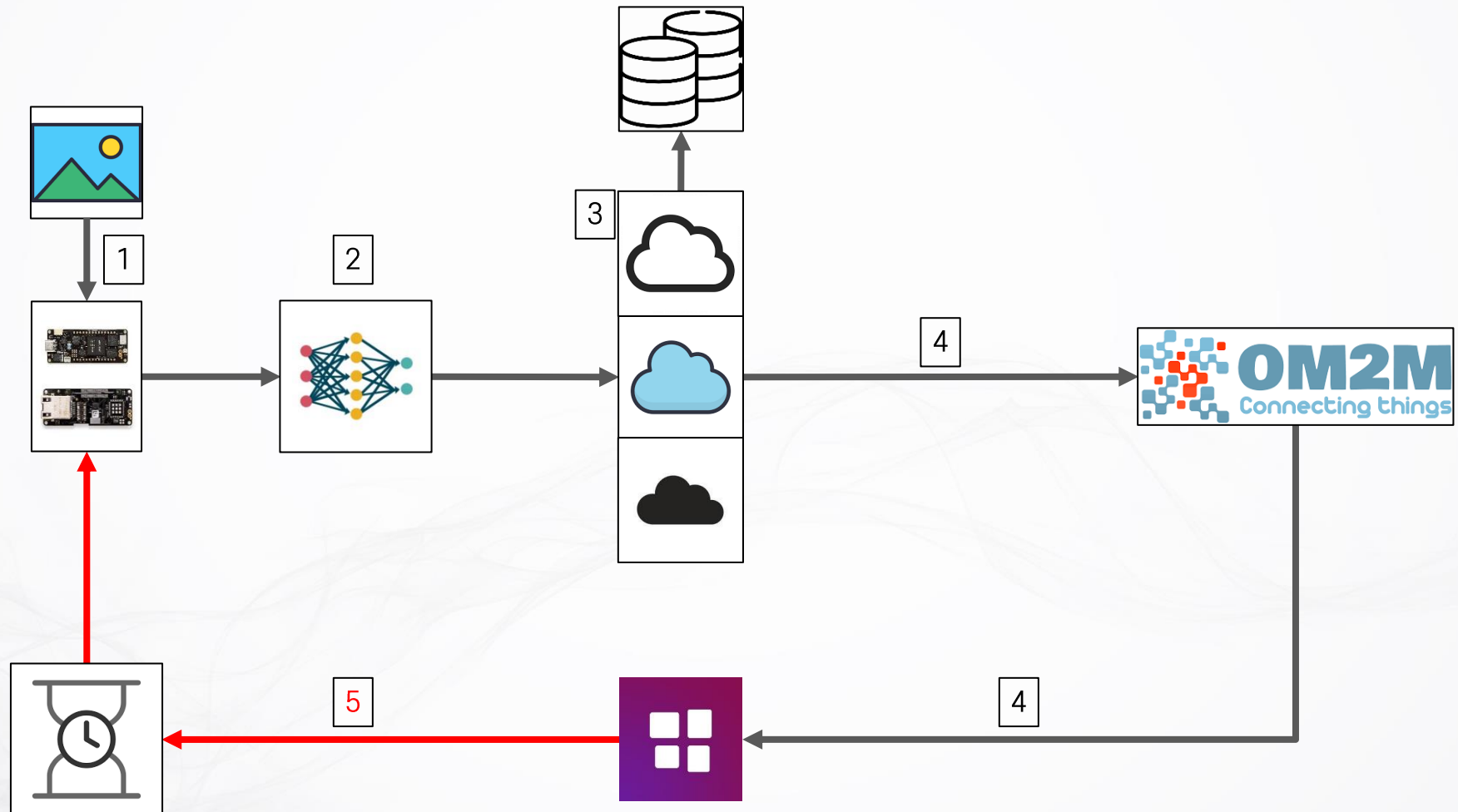
S#4 ➤ Prediction and other information are published via the MQTT protocol





## 2. Steps

S#5 ➤ System then goes into a deep sleep and waits up for performing again



CHAPTER

4

Resource

# 1. BinaryObject

## ● BinaryObject를 선택한 이유

✔ 아래와 같은 후보 군들이 있었지만, 아래 조건에 의해 BinaryObjectFlexContainer를 선택하게 됨

### <후보군>

BatteryFlexContainer.java  
 BinaryObjectFlexContainer.java  
 BinarySwitchFlexContainer.java  
 DeviceAirConditionerFlexContainer.java  
 DeviceCameraFlexContainer.java  
 DeviceHumidifierFlexContainer.java  
 DeviceStorageBatteryFlexContainer.java  
 DeviceSwitchButtonFlexContainer.java  
 DeviceSwitchFlexContainer.java  
 DeviceWeatherStationFlexContainer.java  
 LiquidLevelFlexContainer.java  
 LiquidRemainingFlexContainer.java  
 NumberValueFlexContainer.java  
 PowerSaveFlexContainer.java  
 PushButtonFlexContainer.java  
 RelativeHumidityFlexContainer.java  
 SmokeSensorFlexContainer.java

### <조건>

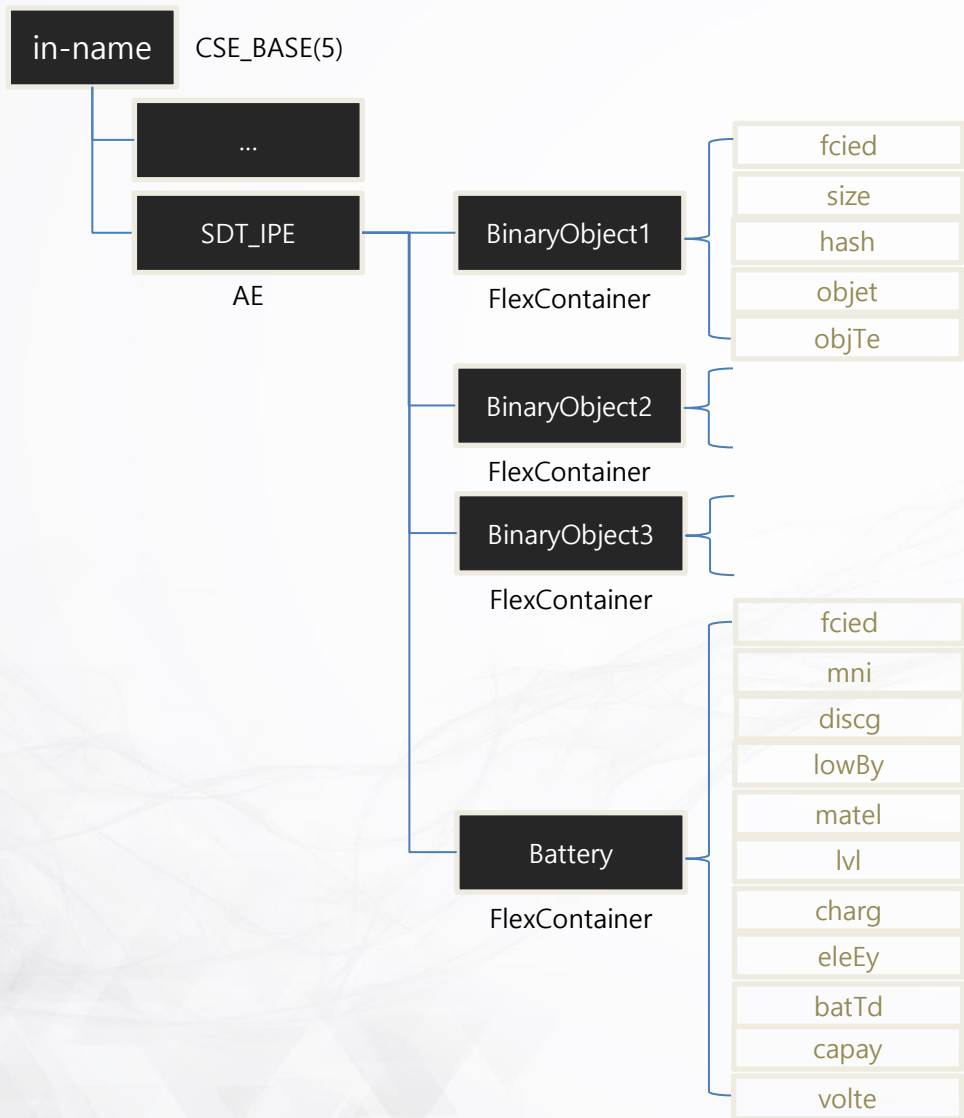
1. 사진은 정수형태로 OM2M으로 전달됨
2. Battery 관련 데이터를 송신하므로 BatteryFlexContainer에 해당 데이터를 저장
3. 사진이 정수형태로 송신되기 때문에 사진의 byte 크기와 관련된 속성이 들어있는 SDT가 용이함
4. OM2M은 모든 데이터를 string으로 저장하기 때문에 DataType를 표시할 속성이 들어있는 SDT가 용이함
5. 데이터의 무결성을 위한 hash 값 저장과 관련된 속성이 들어있는 SDT가 용이함

size	4800
hash	1183803893312091282863447370533851381357341200167
objct	15090506091489611768202868130078909359757604320527
objTe	byte

BinaryFlexContainer의 mandatory custom attribute

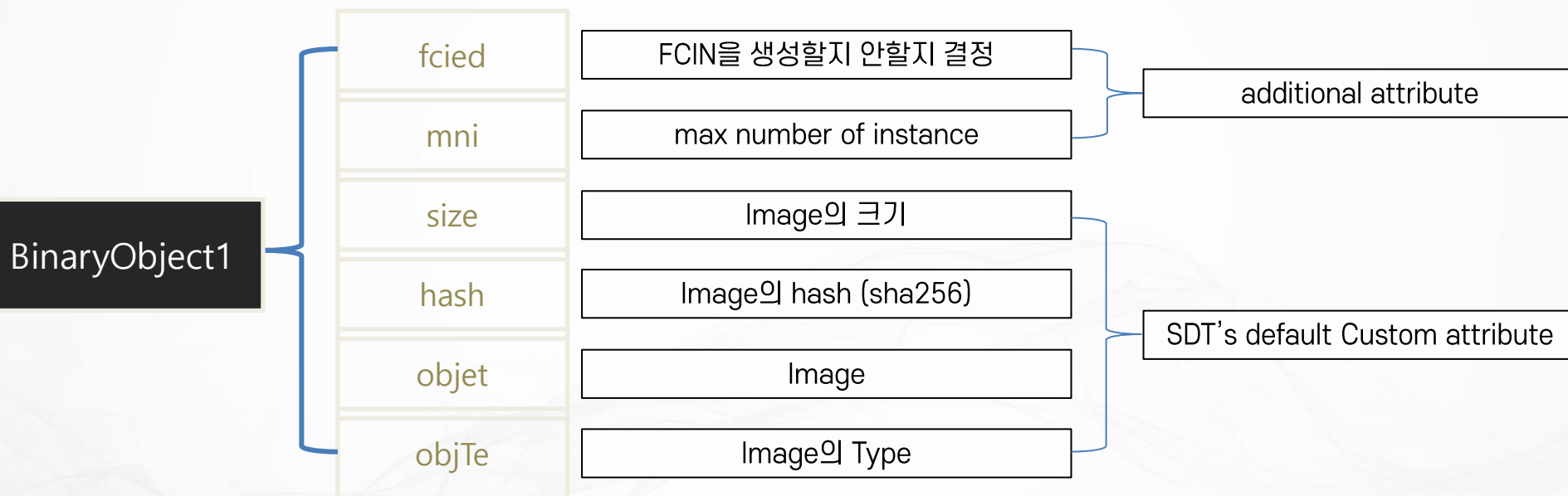
## 2. Resource View

### Resource 모습 및 설명



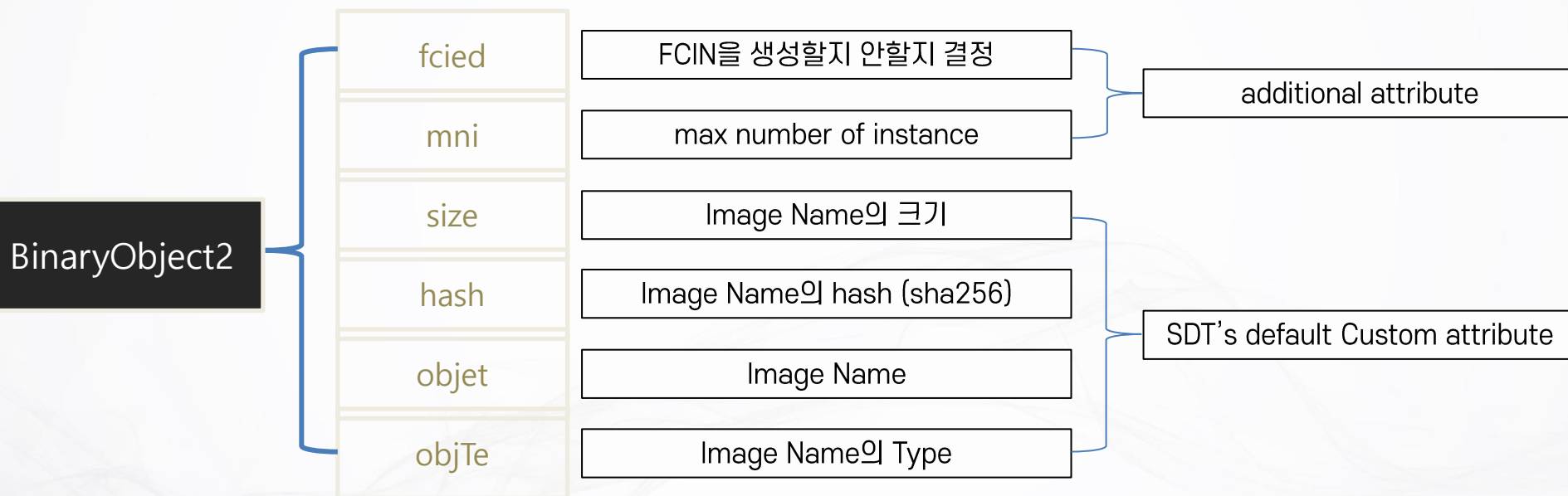
## 3. Resource View

- Resource 모습 및 설명



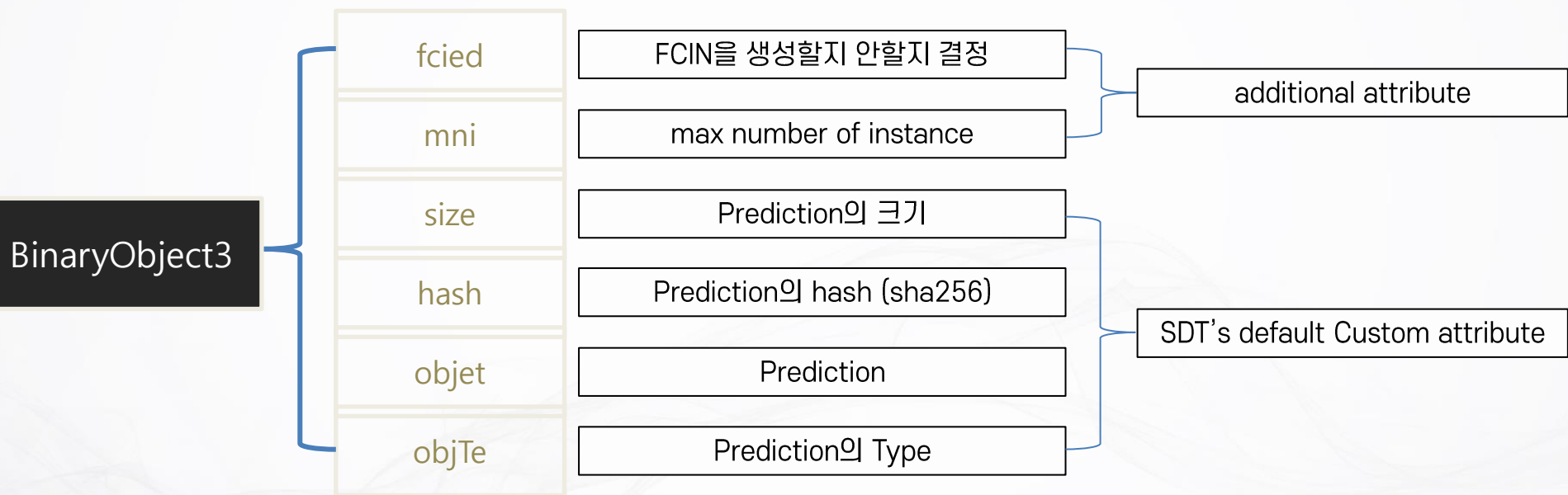
## 3. Resource View

- Resource 모습 및 설명



## 3. Resource View

- Resource 모습 및 설명



### 3. Resource View

- Resource 모습 및 설명

#### Battery

fcied

FCIN을 생성할지 안할지 결정

mni

max number of instance

discg

Battery Discharging

lowBy

lowBattery

matel

material

lvl

level

charg

charging

eleEy

electricEnergy

batTd

batteryThreshold

capay

capacity

volte

voltage

additional attribute

SDT's default Custom attribute



CHAPTER

5

demo

# 1. demo


## ● resource view로 본 모습

Logout

OM2M CSE Resource Tree

<http://20.214.226.173:8080/~in-cse/fcnt-631939576>

- in-name
  - acp\_admin
  - SDT\_Home\_Monitoring\_Application\_ACP
  - ACP\_Device\_Admin\_1669897511603
  - SDT\_Home\_Monitoring\_Application
  - SDT\_IPE
    - BinaryObject
      - fcin\_43876404
      - fcin\_436877205
      - fcin\_693188265
      - fcin\_904404808
      - fcin\_494754405
      - fcin\_428166808
      - fcin\_229323995
      - fcin\_429035033
      - fcin\_719370069
      - fcin\_34026798



Attribute	
rn	BinaryObject
ty	28
ri	/in-cse/fcnt-631939576
pi	/in-cse/CAE834866151
ct	20221201T122622
lt	20221201T124411
acpi	<div> <input type="text" value="/in-cse/acp-551069814"/> </div>
et	20231201T122622
st	17
cnd	org.onem2m.home.moduleclass.binaryObject
cni	10
cbs	116399
mni	10
size	4800
hash	1183803893312091282863447370533851381357341200167
objct	15090506091489611768202868130078909359757604320527
objTe	byte
predic	Clear_Sky
imgNa	I20221201124408_Clear_Sky
batLe	83



THANK  
YOU!