W3C Web of Things Interest Group

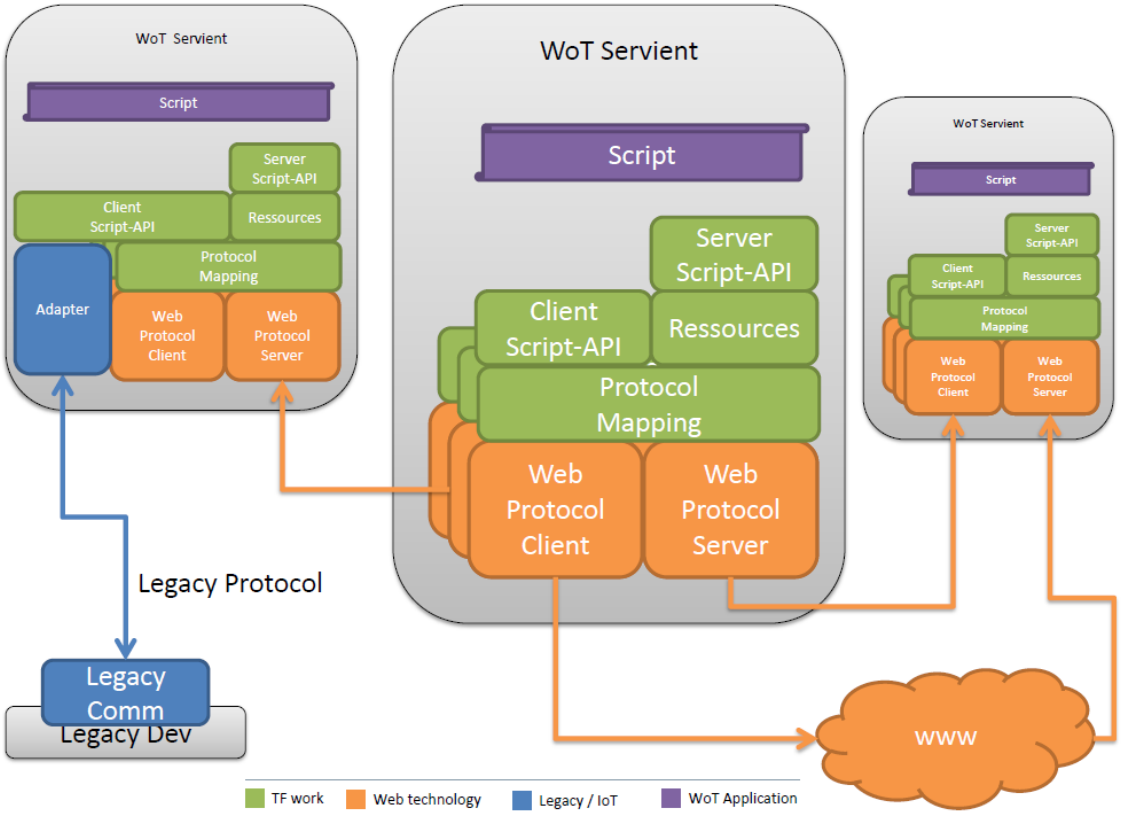
# Web of Things architecture

27th January, 2016

Panasonic, Fujitsu

# Abstract architecture on TPAC 2015

- Abstract model noted at TPAC 2015 Face-to-Face



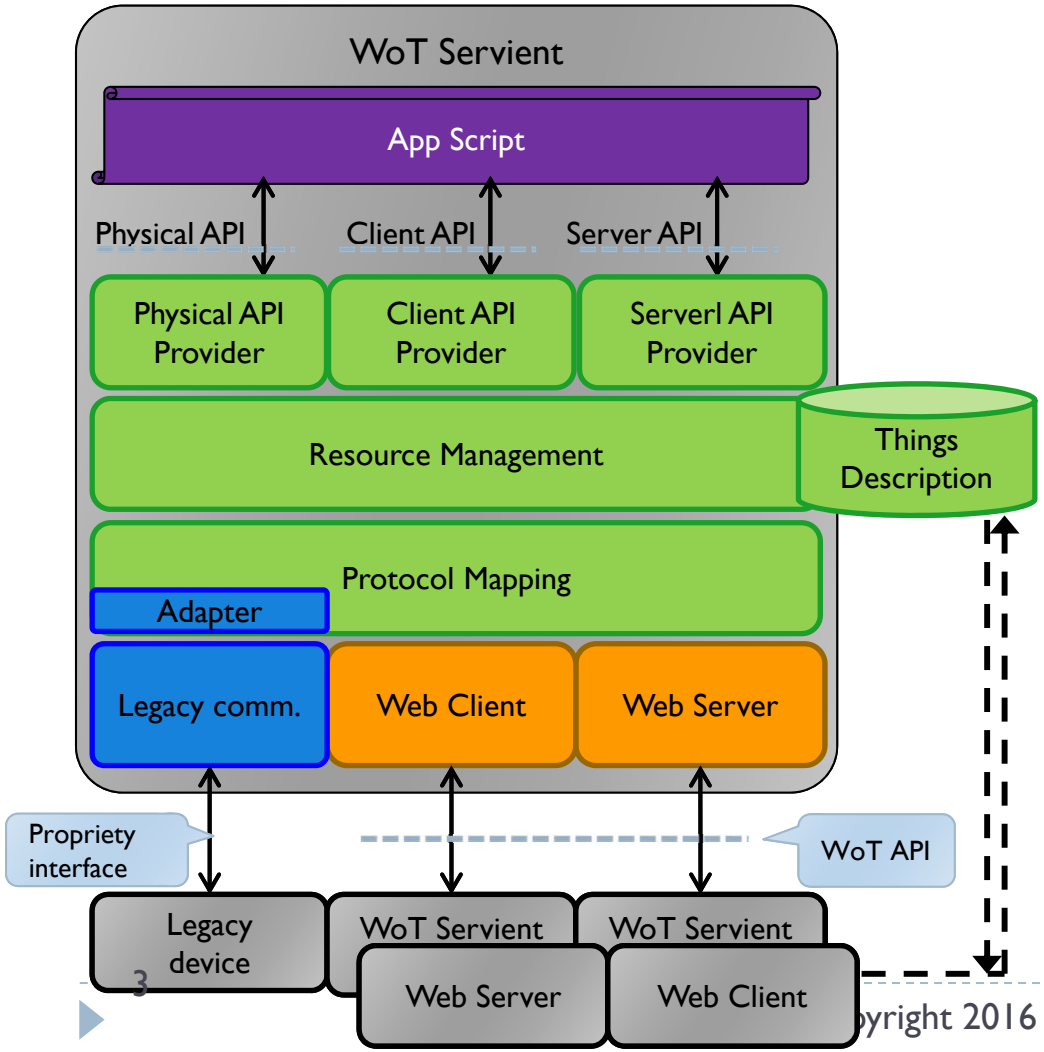

Features of WoT Servient

- Can serve either (or both) as a server and a client, using bi-directional communication
- Can connect legacy devices
- Device abstraction through the concept of Resources and Protocol mapping
- Script API provides access to devices

This is an attempt to re-formulate the model using the four implementation models introduced by Panasonic

# Modified diagram of the basic WoT servient architecture

- Functional architecture of WoT Servient




**Script:**
Provides access to and control of internal data of devices connected to WoT servient according to WoT API interaction

**Things Description**
Declares providing API name, parameter type and so on. External client refers this description to call WoT APIs

**API Provider**
Interprets scripts, retrieves information of devices connected to WoT servient in conjunction with Resource Management.

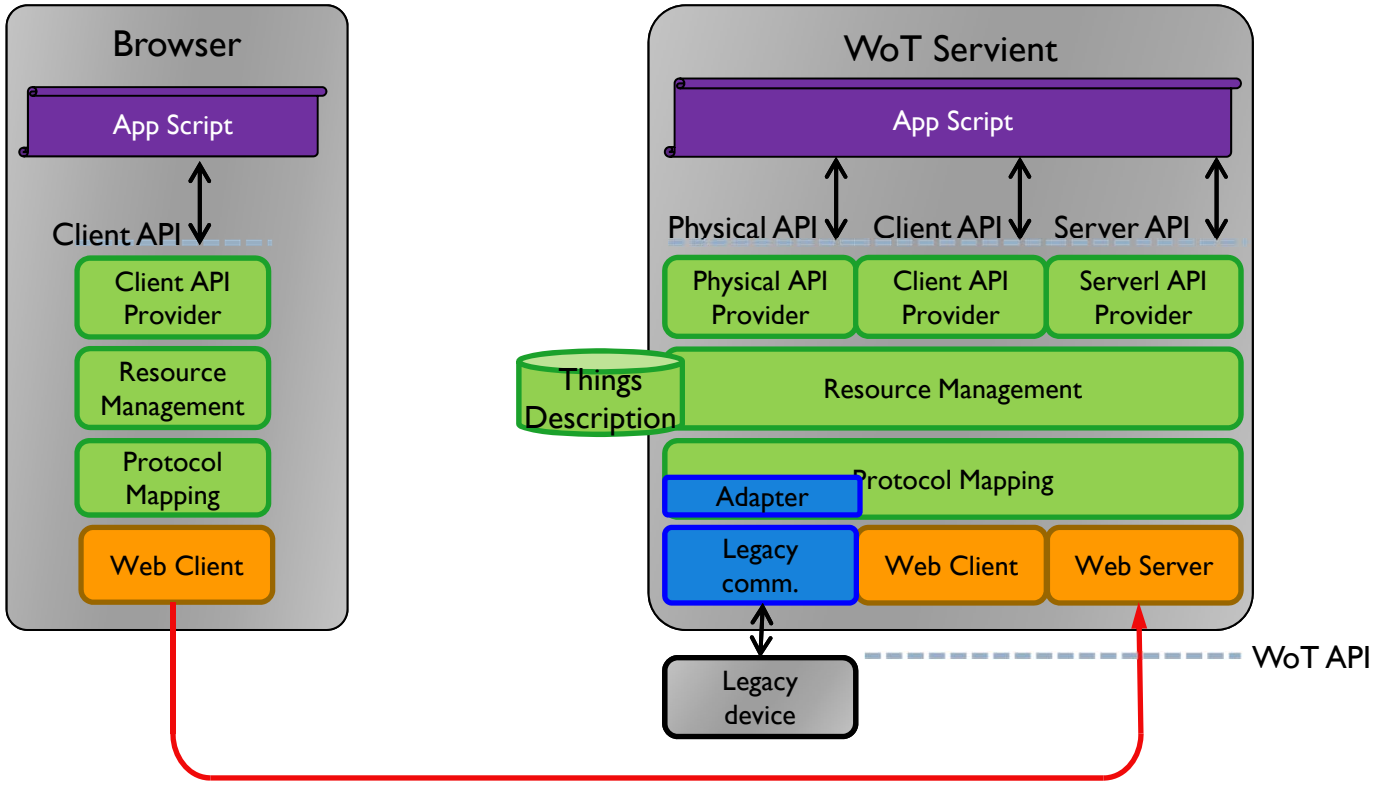
**Resource Management:**
Manages status of connected devices, request ques and so on.

**Protocol Mapping:**
Converts interactions with devices using information in TD in accordance with lower layer protocols. For legacy devices, adapters available for those devices convert the protocol.

**Communication protocol:**
Protocol used in actual communication with objects. In the case of physical devices, every protocol that is necessary to communicate with those devices is installed.

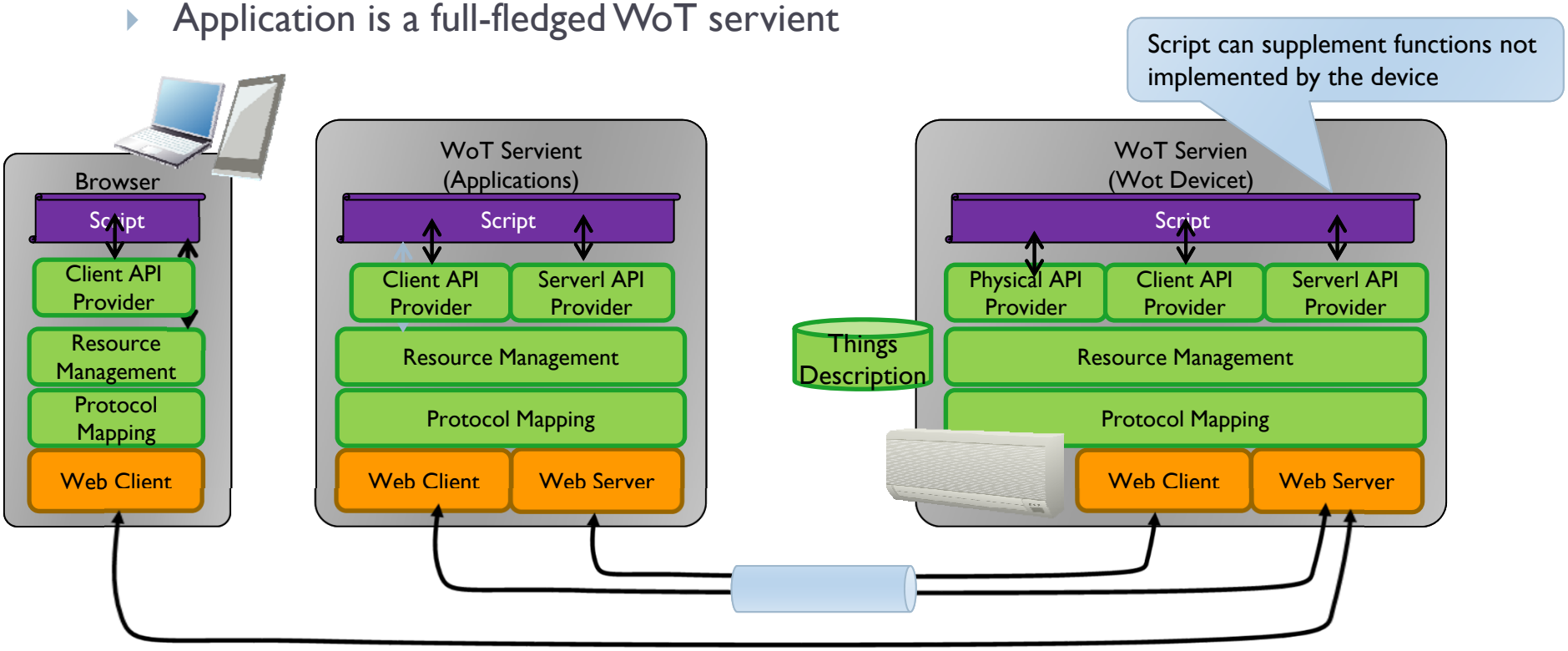
# Simple example for WoT

- Controlling WoT devices from Browser
  - Browser access to information of and control devices by using scripts to access devices through WoT Servient. Device semantics (TD) are referred by browser script



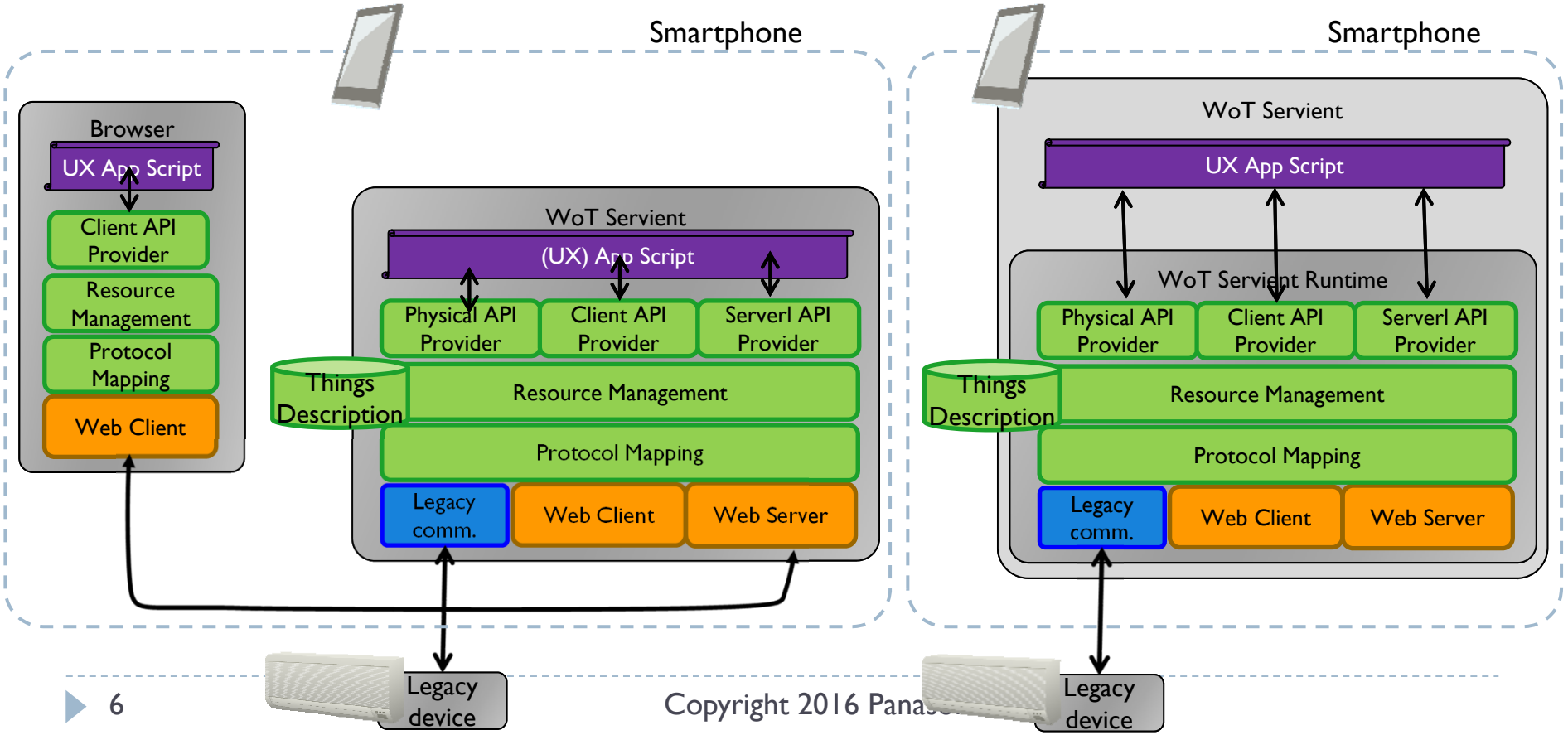
# (A) WoT servient on device (WoT device)

- Device is a WoT servient, directly interprets WoT API
  - Browser is a WoT servient that serves only as Web client
  - Application is a full-fledged WoT servient



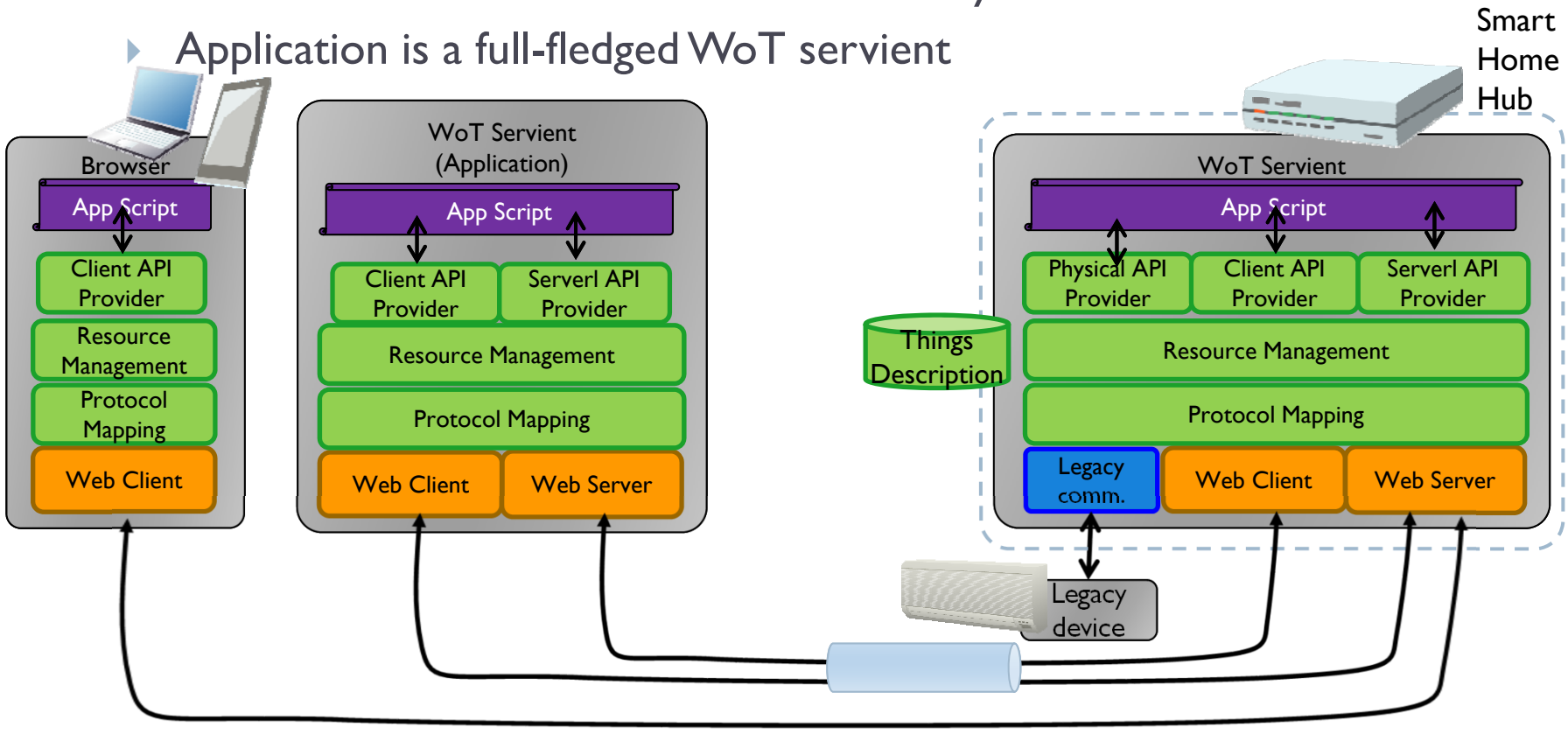
# (B) WoT servient on Smartphone

- WoT servient on Smartphone
  - From the relationship of Wot Servient and UX App script point of view, there are 2 types
  - One is browser calling inside WoT Servient model, the other is WoT Servient itself providing UX
  - Nevertheless the difference of view, WoT servient architecture is logically the same in both models.



# (C) WoT Servient on Smart Home Hub

- WoT servient as a specialized device named "Hub", connecting devices and applications.
  - Browser is a WoT servient that serves only as Web client
  - Application is a full-fledged WoT servient



# (D) WoT Servient on Cloud Server

- WoT servient installed both in cloud and LAN, connecting cloud application and devices in LAN each other
  - Gateway (local) WoT servient holds latest information about connected devices
  - Platform(cloud) WoT servient cashes copy of Gateway's Resource information, routing browser/application's access to devices