

Web Console for Multiple IoT Gateways

Huaqiao Zhang (huaqiaoz@vmware.com)

Preface

When users started using EdgeX Foundry project, they could quickly run the service framework according to the official documents of EdgeX Foundry. But because of the lack of an intuitive Web Console, they may not know how to proceed furthermore. This is exactly the problem I encountered when I started to play with EdgeX Foundry. My tactic was to use the HTTP client tool and call its Restful APIs to become familiar with the features. But those CRUD operations on devices are not very simple and the process is also not very smooth.

At that time, I had an idea to create a Web Console where users only need to operate in the browser instead of manually type in a lot of commands with parameters and assemble complex JSON data, which will provide much better experiences to users. Later, I learned that EdgeX Foundry community enlisted the Web Console for multiple gateway management in the roadmap, so I developed this prototype and plan to contribute it to EdgeX Foundry community.

In the following sections, I will introduce the Web Console prototype completely, including development motivation, design thoughts, and detailed technology implementation.

Why we need the Web Console

When a new user wants to add a new device to a gateway, if no Web Console, he has to put some time and effort of learning the Restfull API of EdgeX Foundry, and he needs to confirm whether the relative data exist for DeviceService, DeviceProfile, DeviceAddress, etc. If not, he has to create that, then gets the ID or Name of that feature. Finally, he assembles complex JSON data and upload it. As another example, sending commands to a device could be even more complicated. All these could be hard for a new user or an on-site debugging engineer. If there is one Web Console, it will be easier.

How to manage multiple gateway instances

When an enterprise uses EdgeX Foundry, multiple gateways may be deployed onsite. In most cases, each gateway has an internal IP address in the LAN rather than an Internet address. So, how to manage these gateways via a web console? There may be two approaches:

- A Web Console is deployed to each gateway, users need to remember the address of each gateway to operate, and have to maintain multiple web consoles. Each gateway has to cost some resources to run its own web console.
- Multiple gateways share one Web Console, and there is a method to switch among all the gateways. All operation requests will be proxied to the selected gateway. In this way, users only need to remember one address and maintain one Web Console.

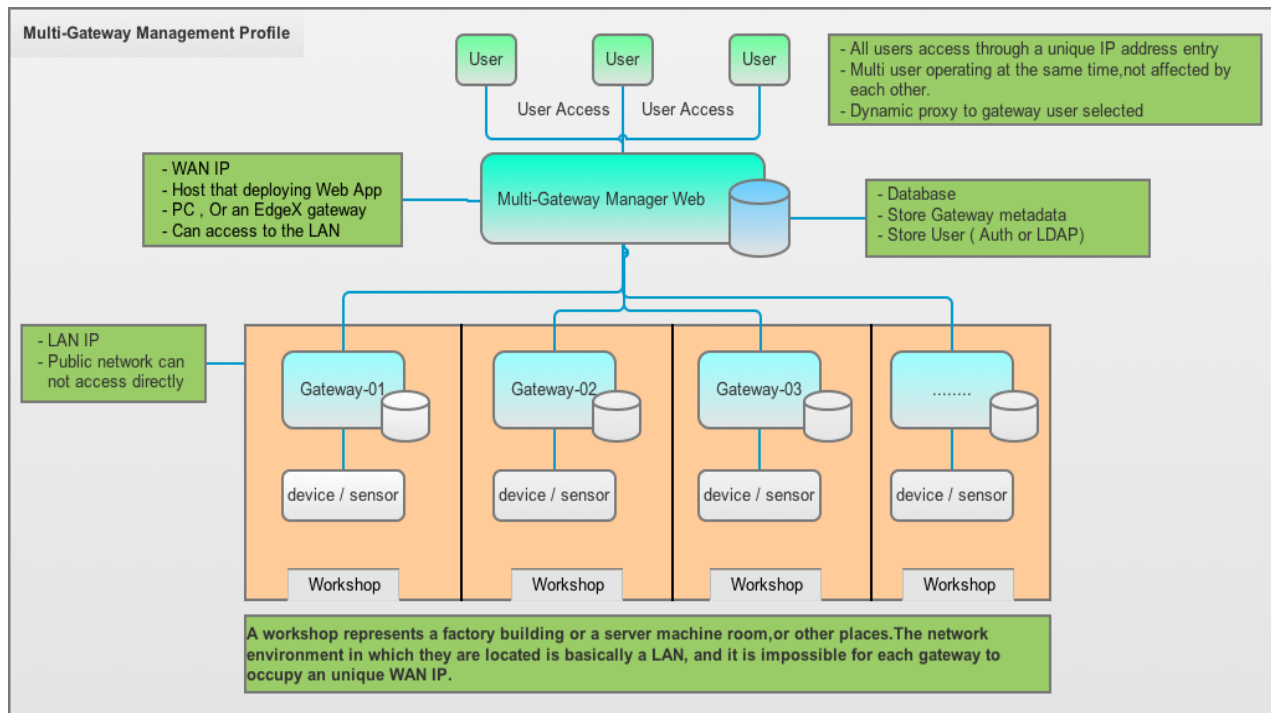
Comparing the two options above, I prefer the later one. The limitation is that all gateways must be accessible to the host where Web Console is deployed. But in a company's intranet, this should not be something particularly difficult.

Problems solved and basic implementation

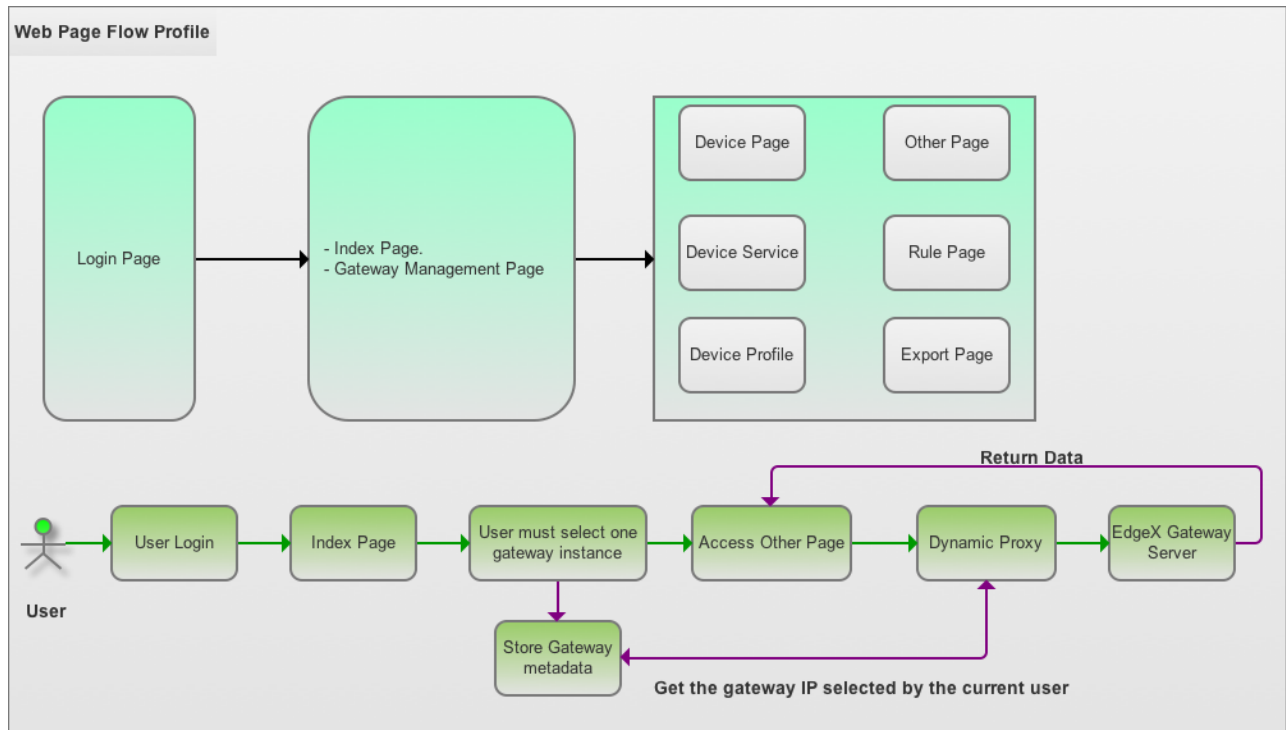
The assumptions and expectations of multi gateway sharing Web Console are:

- Gateways can be anywhere, but for an enterprise, they may all be in the intranet.
- The host, which the Web Console is deployed on, can be one of the gateway or a PC that can access all gateways directly. So, the console should be very light weighted.
- All operation requests should be dynamically proxied to the gateway selected by the user.
- Multiple users could operate different gateways at the same time without affecting one another.

Based on these requirements, the fundamental architecture is shown below:



The basic user's operation flow is shown as below:



- After login, user will be navigated to the management page of the default gateway. If there are no gateways at all or no gateway selected, most menu items will not be permitted to operate on.
- When the user selects or creates one gateway, the metadata of gateway are stored into the local database in the web console.
- Once one given gateway is activated, all operations will be proxied to the target gateway, then the data will be returned to Web Console.
- Multi-user's operations on different gateways will not affected one another.

Some necessary operations are illustrated below.

EdgeX Foundry Console 0.2.0

#	ID	Gateway Name	Description	Address	Created Time
1	402881e4	test-gateway-01	test-gateway-01	10.112.122.28	2018-03-27 16:25:53

Gateway Management

EdgeX Foundry Console 0.2.0

Gateway

Devices

Device

Device Service

Device Profile

Multimedia

Export

Rule Engine

BackSubmit

New Device General Info

Nametest-device-01Descriptiontest device-01

AdminStateUNLOCKEDOperatingStateENABLED

New Device Related Profile

DeviceProfileTest.Device.MQTT.Profile-01

No desired ? Click upload.Upload

New Device Addressable

Nametest-device-addrMethodPOST

ProtocolHTTPAddress100.100.100.100

Port8888Path/api/v1/callback

Publisherhuaiqiao_zhangUserhuaiqiao_zhang

Password1234Topictest-device-01

New Device Related Service

DeviceServiceedgex-device-mqtt

Adding a Device

EdgeX Foundry Console 0.2.0

Gateway

Devices

Device

Device Service

Device Profile

Multimedia

Export

Rule Engine

+ DeviceRefreshJSON show

#	ID	Service Name	Label	Operate State
1	Sab3528b	edgex-device-mqtt	MQTT	
2	Sab35234	edgex-device-virtual	virtual	
3	Sab35180	edgex-support-scheduler	No Labels	

```
{  "id": "Sab3528be4b0c393a5f0c06d",  "created": 1521701515456,  "modified": 1521701515456,  "origin": 1521701515222,  "description": null,  "name": "edgex-device-mqtt",  "lastConnected": 0,  "lastReported": 0,  "operatingState": "enabled",  "labels": [    "MQTT"  ],  "addressable": {    "id": "5ab3528be4b0c393a5f0c06c",    "created": 1521701515180,    "modified": 1521701515180,    "origin": 1521701514482,    "name": "edgex-device-mqtt",    "method": "POST",    "protocol": "HTTP",    "address": "edgex-device-mqtt",    "port": 49982,    "path": "/api/v1/callback",    "publisher": null,    "user": null,    "password": null,    "topic": null,    "url": "HTTP://edgex-device-mqtt:4",    "baseUrl": "HTTP://edgex-device-mq"},    "adminState": "unlocked"  }  }
```

Device Service Management

<https://github.com/badboy-huaqiao/simple-local-gateway-console> or <https://github.com/badboy-huaqiao/edgex-foundry-web-console>.