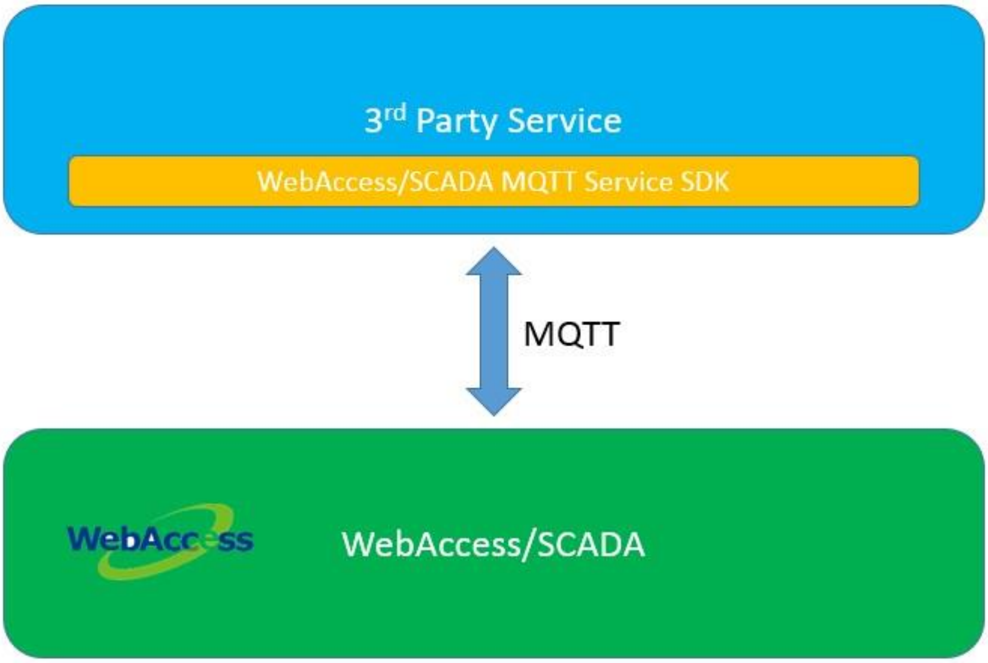
WebAccess/SCADA MQTT Service SDK

3rd party service receive data from WebAccess/SCADA

3rd party service send control message to WebAccess/SCADA



MQTT Service C# SDK

1. Source Code Reference

- Github Repo: MQTT-SDK-for-CSharp
- Service SDK: MQTT.Service.DotNet.SDK
- Sample Code: MQTT.Service.DotNet.SDK.Test

2. Environment

- IDE
  - Visual Studio 2013 or above
- Runtime
  - .Net Framework 4.5.2 or above

3. Instructions

3.1. Constructor(EdgeAgentOptions options)

New a EdgeAgent object.

```
EdgeAgentOptions options = new EdgeAgentOptions()
{
    AutoReconnect = true,
    ReconnectInterval = 1000,
    ScadaId = textBoxGroupId.Text,
    Heartbeat = 60000,    // default is 60 seconds,
    ConnectType = ConnectType.MQTT,
    UseSecure = checkBoxSSL.Checked
};
options.MQTT = new MQTTOptions()
{
    HostName = "127.0.0.1", // your broker location
    Port = 1883,
    Username = "admin",
    Password = "admin",
    ProtocolType = Protocol.TCP
};
_edgeAgent.Connect();
```

3.2. Event

EdgeAgent has three event for subscribing.

- Connected: When EdgeAgent is connected to cloud (WebAccess MQTT Broker).

- Disconnected: When EdgeAgent is disconnected to cloud.

- MessageReceived: When EdgeAgent received MQTT message from cloud. The message type as follows:

    - ReadValue: Change tag value from cloud.

    - ReadConfig: Change config from cloud.

```
_edgeAgent = new EdgeAgent(options);
_edgeAgent.Connected += _edgeAgent_Connected;
_edgeAgent.Disconnected += _edgeAgent_Disconnected;
_edgeAgent.MessageReceived += _edgeAgent_MessageReceived;
```

Create a JObject from an .NET object. It also implements IDictionary.

```
private void _edgeAgent_MessageReceived(object sender,
MessageReceivedEventArgs e)
{
    JObject jObj = JObject.Parse(e.Message.ToString());
}
```

3.3. Connect()

Connect to cloud (WebAccess MQTT Broker). When connect success, the connected event will be triggered.

```
edgeAgent.Connect();
```

3.4. Disconnect()

Disonnect to cloud (WebAccess MQTT Broker). When disconnect success, the disconnected event will be triggered.

```
edgeAgent.Disconnect();
```