Using WebIOPi from the browser is easy, but using it from an application is now easy as well. Just like the Javascript library, you can use the Python and Java Client libraries to make anything-to-Pi or Pi-to-Pi communication. Python client library provides both HTTP and CoAP WebIOPi Clients, and also a Mixed one, which uses CoAP with a HTTP fallback.

```python
from webiopi.clients import *
from time import sleep

# Create a WebIOPi client
client = PiHttpClient("192.168.1.234")
#client = PiMixedClient("192.168.1.234")
#client = PiCoapClient("192.168.1.234")
#client = PiMulticastClient()

client.setCredentials("webiopi", "raspberry")

# RPi native GPIO
gpio = NativeGPIO(client)
gpio.setFunction(25, "out")
state = True

# DAC named "dac1"
dac = DAC(client, "dac1")

# ADC named "adc1"
adc = ADC(client, "adc1")
value = 0.0

# Temperature sensor named "temp0"
temp = Temperature(client, "temp0")

while True:
    # toggle digital state
    state = not state
    gpio.digitalWrite(25, state)

    # increase analog value
    value += 0.01
    if value > 1.0:
        value = 0.0
    dac.writeFloat(0, value)

    # DAC output 0 is wired to ADC input 1
    val = adc.readFloat(1)
    print("Analog = %.2f" % val)

    # Retrieve temperature
    t = temp.getCelsius()
    print("Temperature = %.2f Celsius" % t)

    sleep(1)
```