Motivation

SmartJS: a rich JavaScript-based self-adaptable runtime environment which features a universal programming API, a comprehensive monitoring framework and an ubiquitous communication substrate for engineering and developing dependable, scalable, adaptable large-scale IoT systems.

System Architecture

Physical and Logical Constraints

Physical Constraints - Device-Specific
- CPU (Workload units)
- RAM
- Bandwidth: incoming/outgoing, towards other nodes
- Latency (in relation to other nodes)

Logical Constraints - Component-Specific
- CPU (Workload units)
- RAM
- Bandwidth: incoming/outgoing, towards other components
- Latency (in relation to other components)

Writing a SmartJS Application

```javascript
// ...
// Connect
pubsub.connect(function() {
  // Repeat every second
  setInterval(function() {
    // Read temperature from GPIO pin
    var temperature = GPIO.readPin(12);
    // Publish temperature
    pubsub.publish("smartsensor/temperature", {
      id: "mySensorId",
      temperature: temperature
    });
  }, 1000);
});
// ...
// Connect
pubsub.connect(function() {
  // Subscribe to temperature messages
  pubsub.subscribe("smartsensor/temperature", function(id) {
    if (id > threshold) {
      pubsub.publish("smartsensor/actuation", {
        id: id,
        powerVariation: 5
      });
    } else if (id < threshold) {
      pubsub.publish("smartsensor/actuation", {
        id: id,
        powerVariation: -5
      });
    }
  });
});
// ...
// Connect
pubsub.connect(function() {
  // Subscribe to power topic to receive commands
  // from the manager
  pubsub.subscribe("smartactuator/actuation", function(id) {
    // Adjust the power by writing to GPIO pin
    GPIO.writePin(14, d.powerVariation);
  });
});
```

References