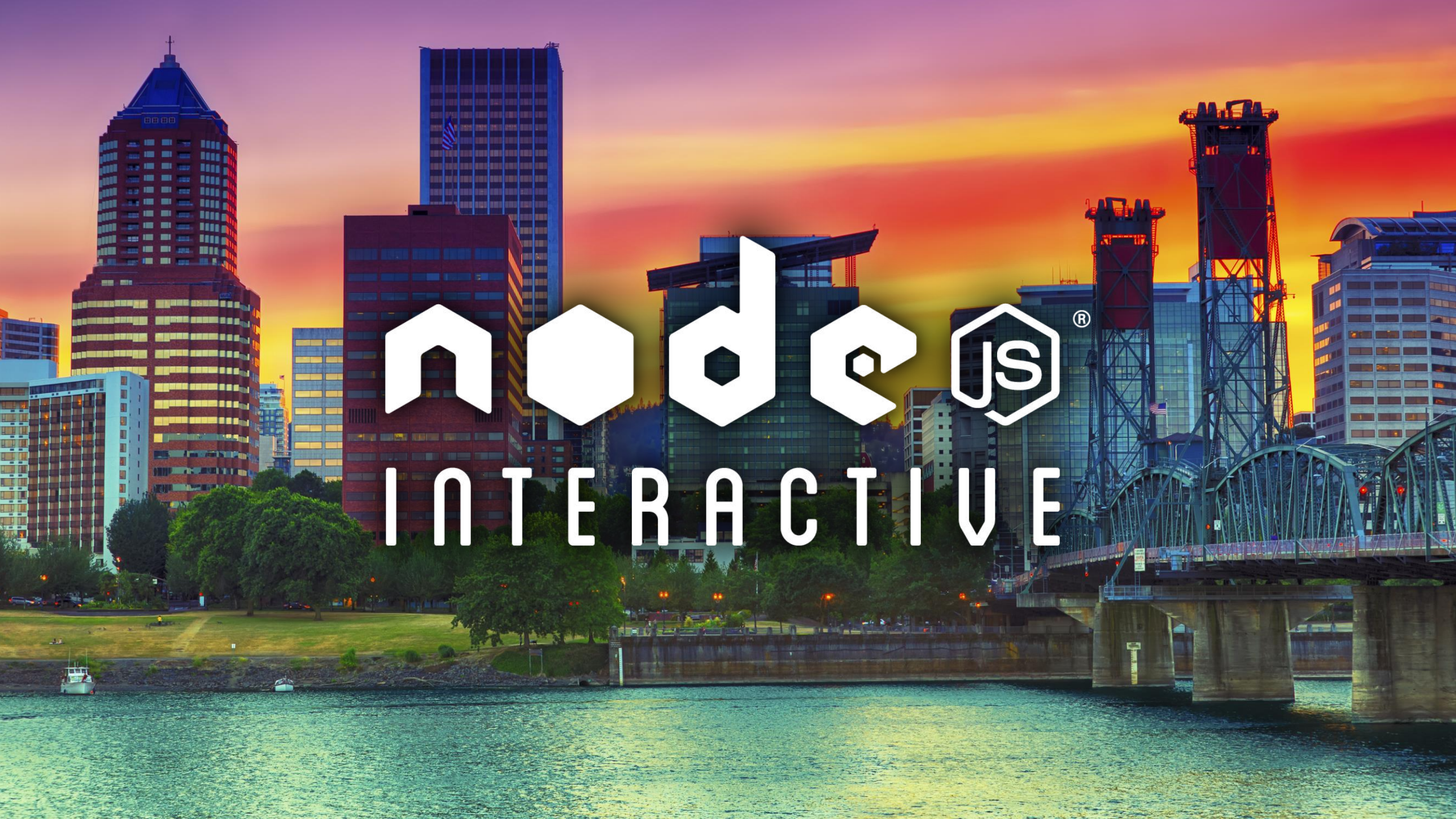

node JS®
INTERACTIVE

IoT Services Orchestration Layer

– help you create IoT apps in minutes

Jonathan Ding, Intel Corporation
jonathan.ding@intel.com

Problem Statement

big data
storage &
analytics &
intelligence

Employing Cloud

Internet of Things

Services

Many
Distributed
Dynamic
Heterogeneous

Connecting the dots → IoT Solutions

Complexity expected … …

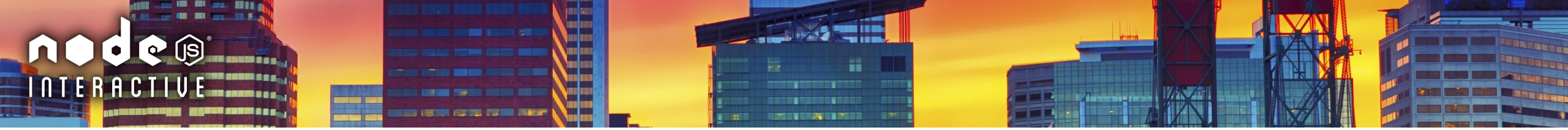
So, possible to make it easier & faster to
Make, Deploy and Manage ???

Intel® IoT Services Orchestration Layer

sensors | actuators | devices

Intel®
iot-devkit

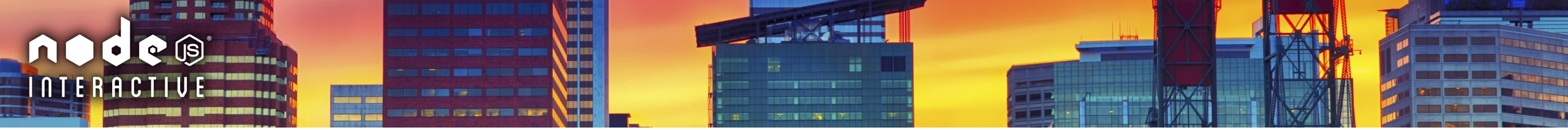
Enabling a “Thing”



Short Term Goal → Long Term Objective

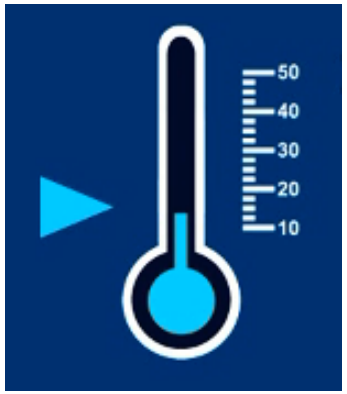
Deliver a robust, extensible, high quality Solution for creating IoT Apps in minutes

IoT App Store - apps developed on one environment but deployable for many other environments similar but not exactly the same

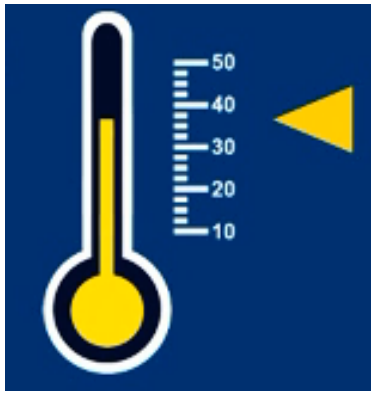


Demo

VIDEO



IF **Temperature @ A** **>**



Temperature @ B

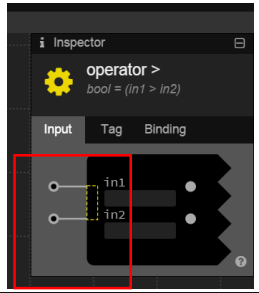
THEN 

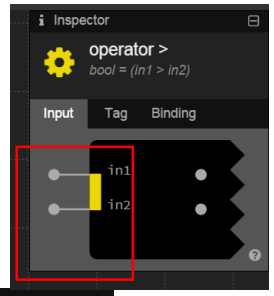
Turn on Fan

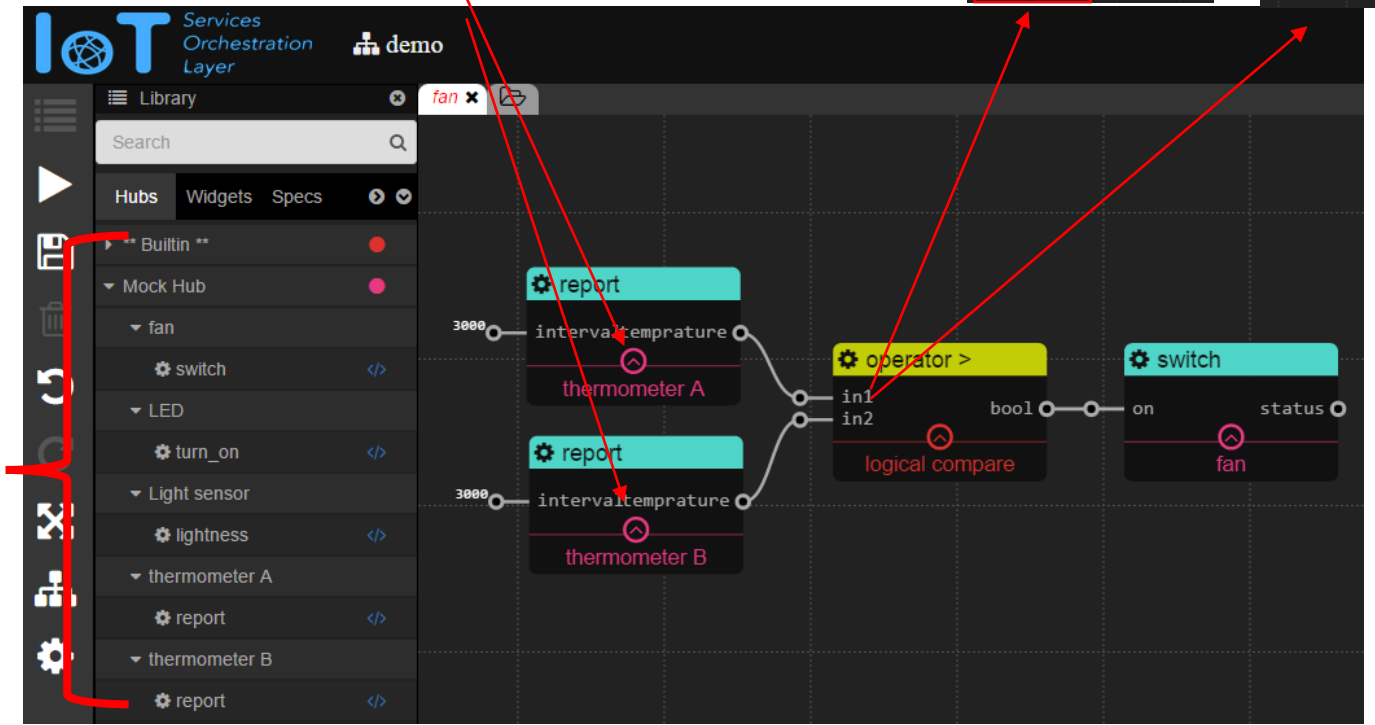
ELSE **Turn off Fan**

Demo – Create Workflow

Binding: Dynamically map to real service provider
e.g. a fixed Device ID, or a Query, or a function







Pattern to Trigger Operation

Configure to various data pattern

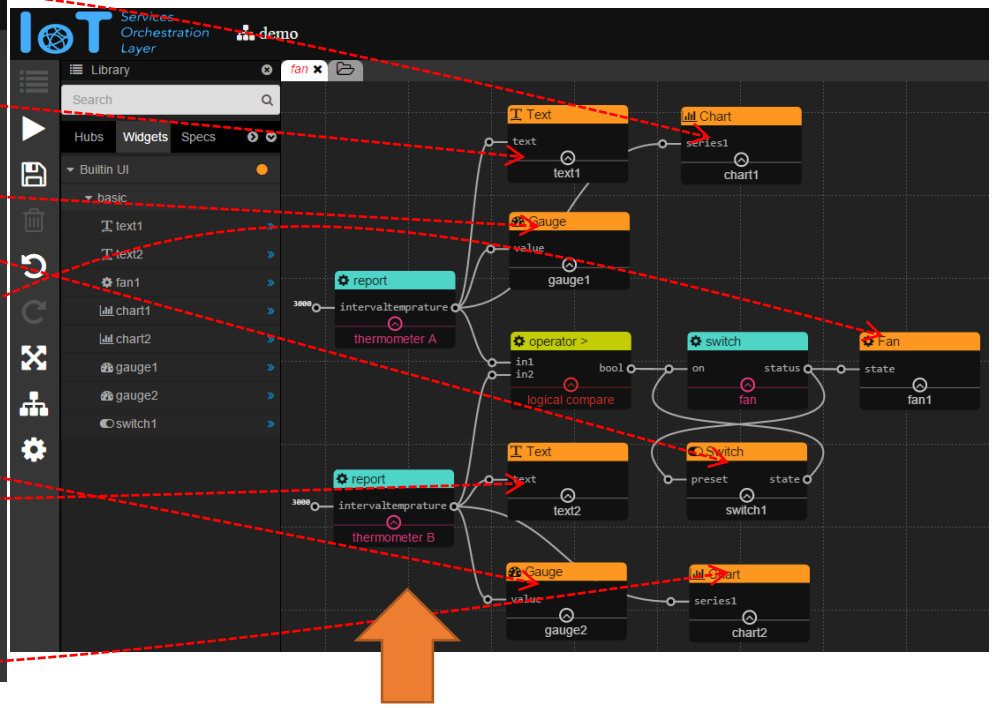
AND – operate when **both** ports have data
OR – operate when **either** port has data

Services

standard or **customized**

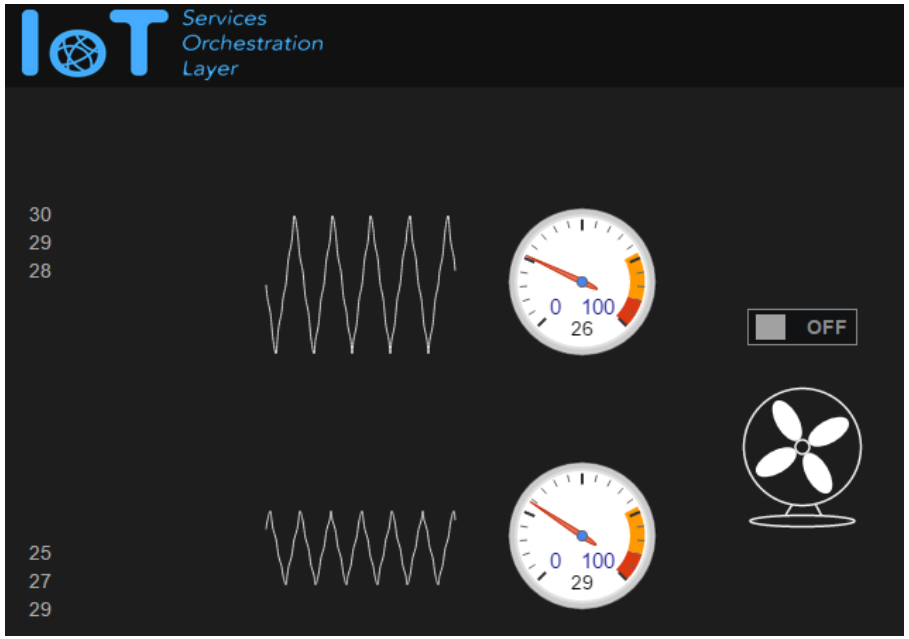
Demo – Add UI into Workflow

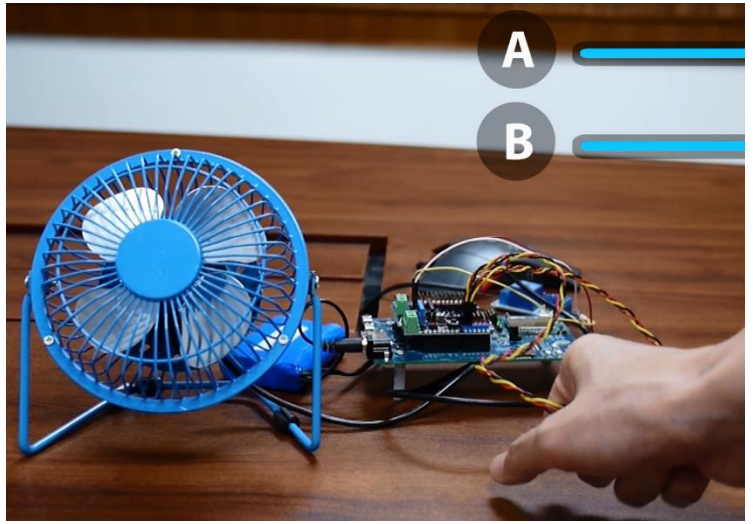
Design what End User would see



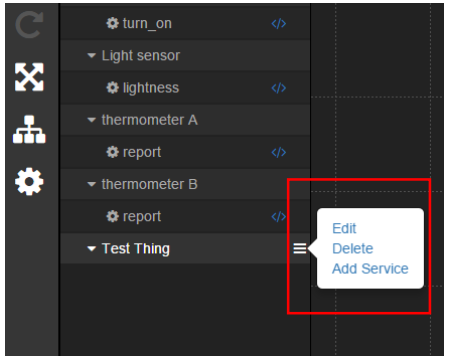
HTML5 UI widgets behave like a virtual IoT Thing

Demo – End Users would see

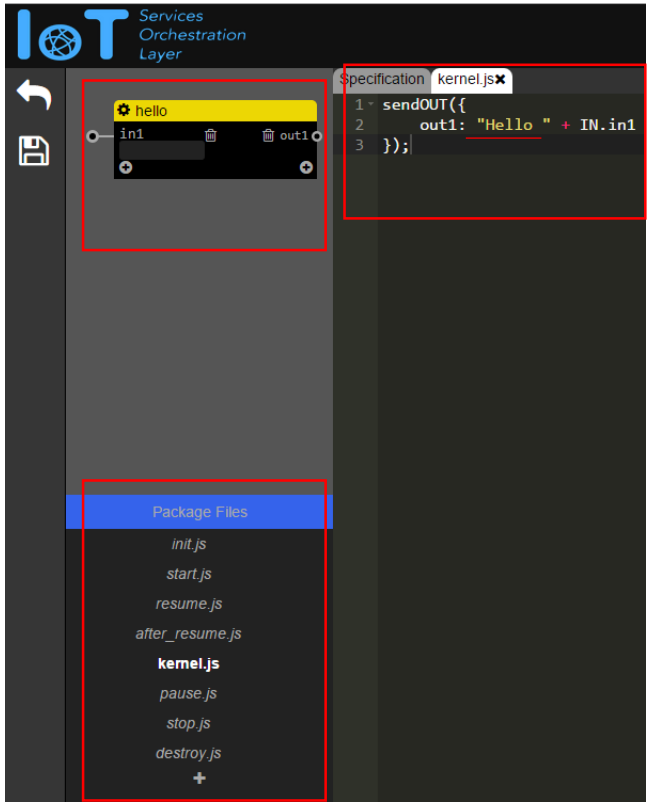


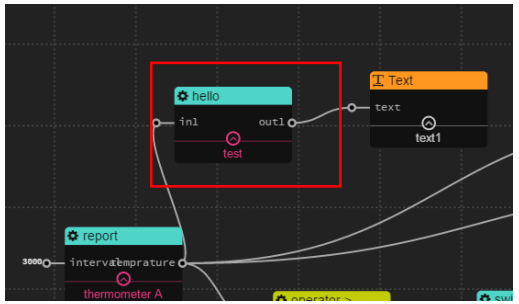


Demo – Create New Customized Service

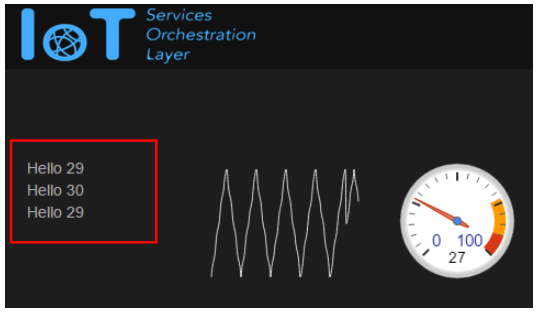


CRUD thing / service





Use that in application



JS editor to add logic for each stage of the service lifecycle

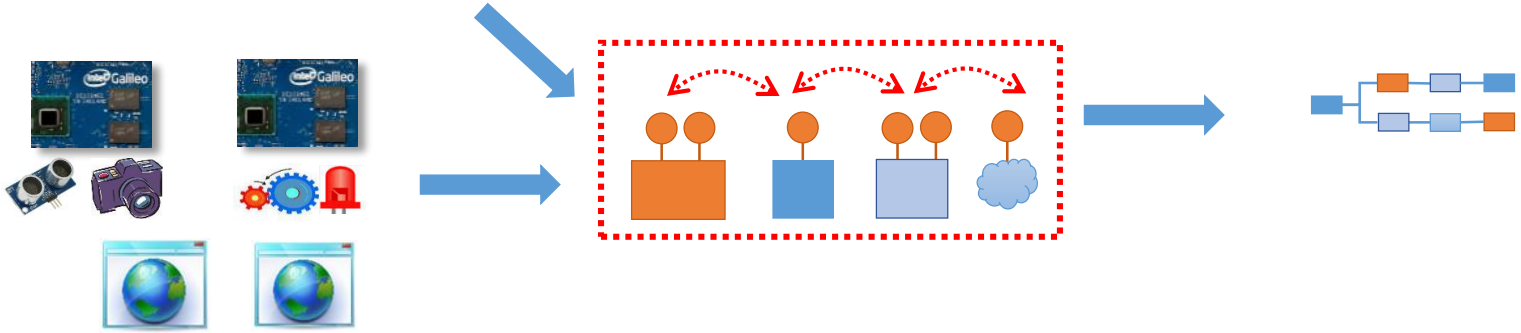
Intel® IoT Services Orchestration Layer

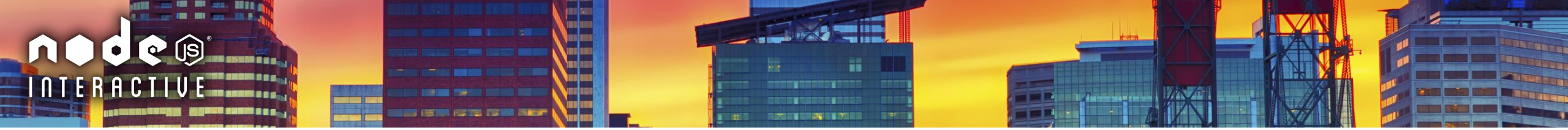
node.js ≈≈ Distributed Middleware + HTML Web IDE

Transparent Remote Communication

Software Defined IoT Devices
(Device/Cloud/UIWidget → API/Service/Data)

Graphical Programming

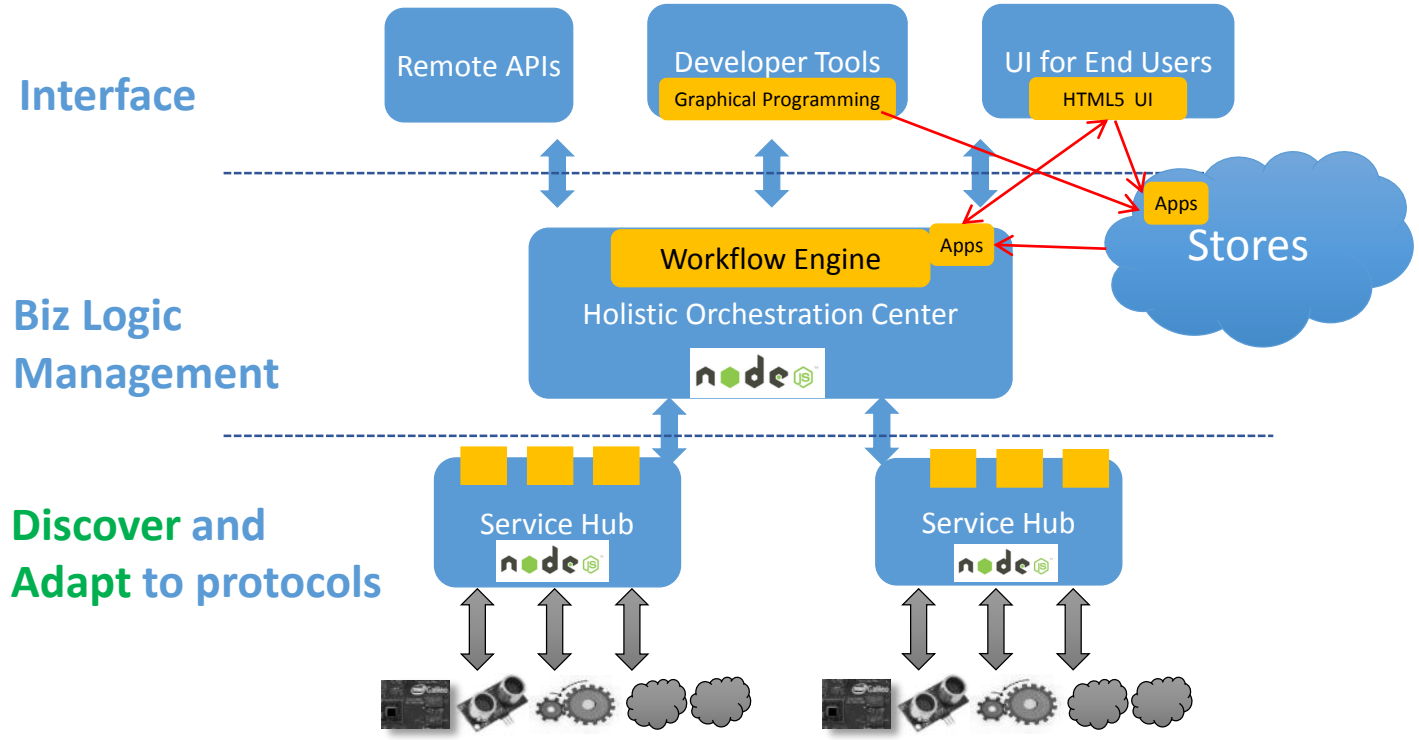




Inspirations and Differentiations

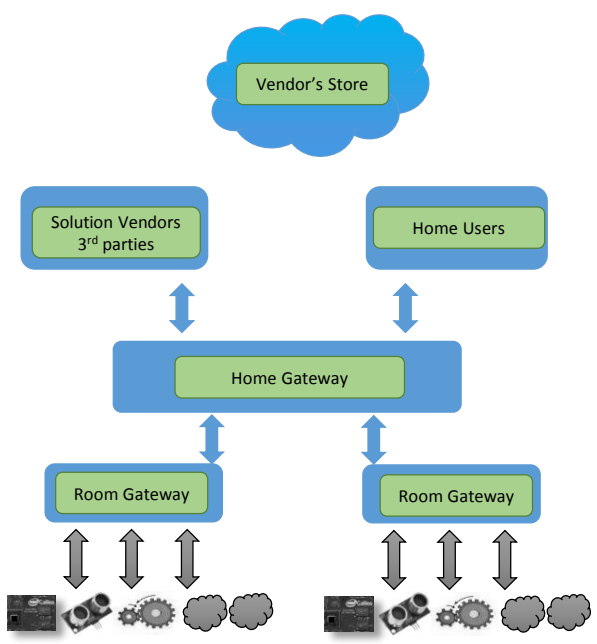
- FBP (Flow Based Programming) variants: Node-Red, Noflo, …
- Unique features we offered – for IoT:
  - Holistic View of Entire IoT System
    - Remote communication are transparent to developers
  - UI as virtual IoT Things
    - Consistent development experience for logic and UI
  - IoT Oriented Workflow Engine
    - Dynamic Binding
    - Configurable Data Pattern for Multiple inports and outports
    - Traces
  - More in development

Architecture

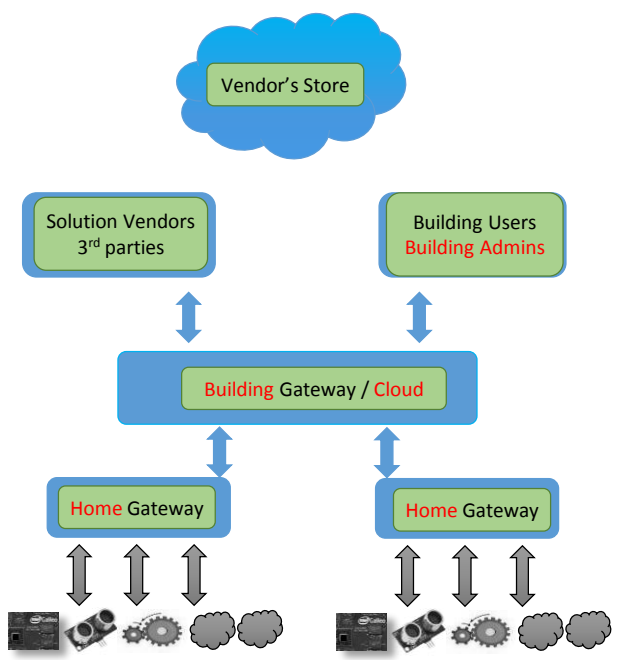


Colorful Items are part of our IoT Services Orchestration Layer

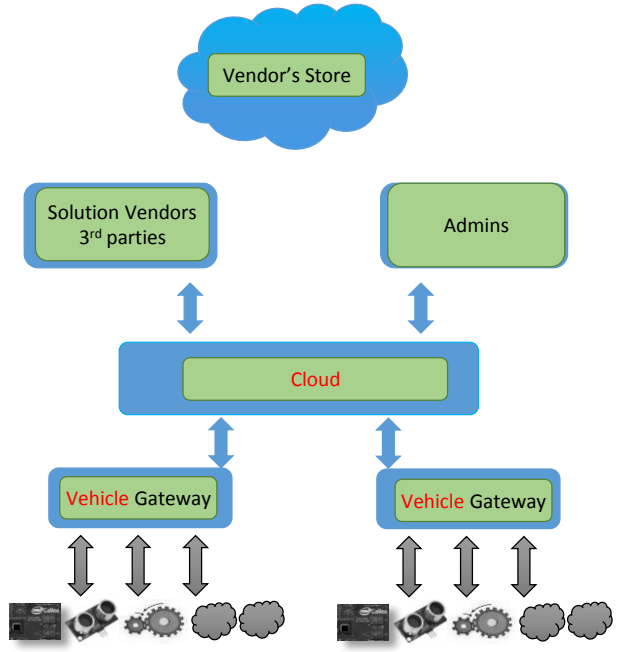
Flexible Deployment Options



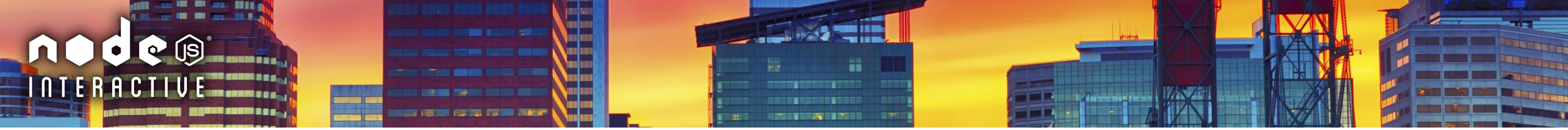
Smart Home



Smart Building



Smart Transportation



Open Source Resources

- **v0.9 open source, BSD license**
  - https://github.com/01org/intel-iot-services-orchestration-layer
- **v1.0 release by end of 2015, with more**
  - Documentation & Samples
  - Built-in Services & UI Widgets
  - Protocols supported
- **Future releases**
  - Enhancement for various verticals
- **Contact**
  - jonathan.ding@intel.com

Contributions from Community are Highly Appreciated