Service Oriented Things:

Building Next Generation Services Using Internet of Things

- Kwei-Jay Lin
- University of California, Irvine
- August 2014

Intel-NTU Connected Context Computing Center

Influential New iOT Products

IoT is Hot!

Dropcam
INFLUENCE SCORE: 183

Nest and **Dropcam** confirmed the acquisition, but did not specify the price tag.
- Source: Sharenet 06/21/2014

Logitech Harmony Remote
INFLUENCE SCORE: 45

Logitech announces an update to its **Harmony Ultimate Remote**.
- Source: CNBC 06/24/2014

Philips Hue Smart Bulb
INFLUENCE SCORE: 19

Philips announced **Hue Tap** [kinetic energy] controller … tapping the button generates all the energy to carry out a command.
-Source: Techno Buffalo 03/29/2014

Honeywell Lyric Thermostat
INFLUENCE SCORE: 158

Honeywell has announced **'Lyric',** a new [smart] thermostat to compete with **Nest.**
- Source: iClarified 06/10/2014

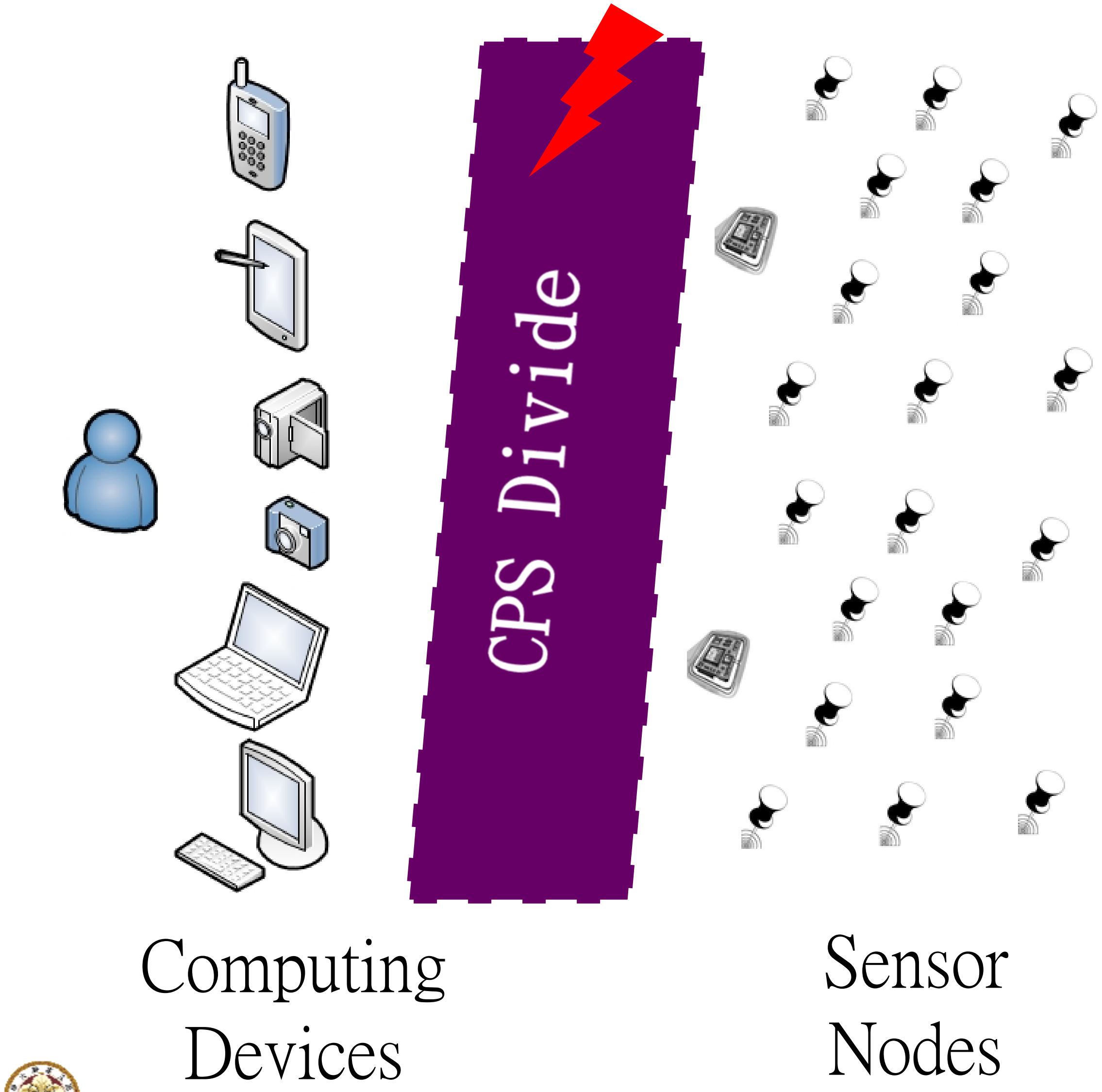
Kwikset Kevo E-Lock
INFLUENCE SCORE: 20

Kevo will start to work right away, as the included fob works out of the box. Kwikset does, however, recommend that you calibrate the fob and any smartphones in order to enable the sensors built into the device.
- Source: iLounge 06/10/2014

Belkin WeMo Home Controller
INFLUENCE SCORE: 18

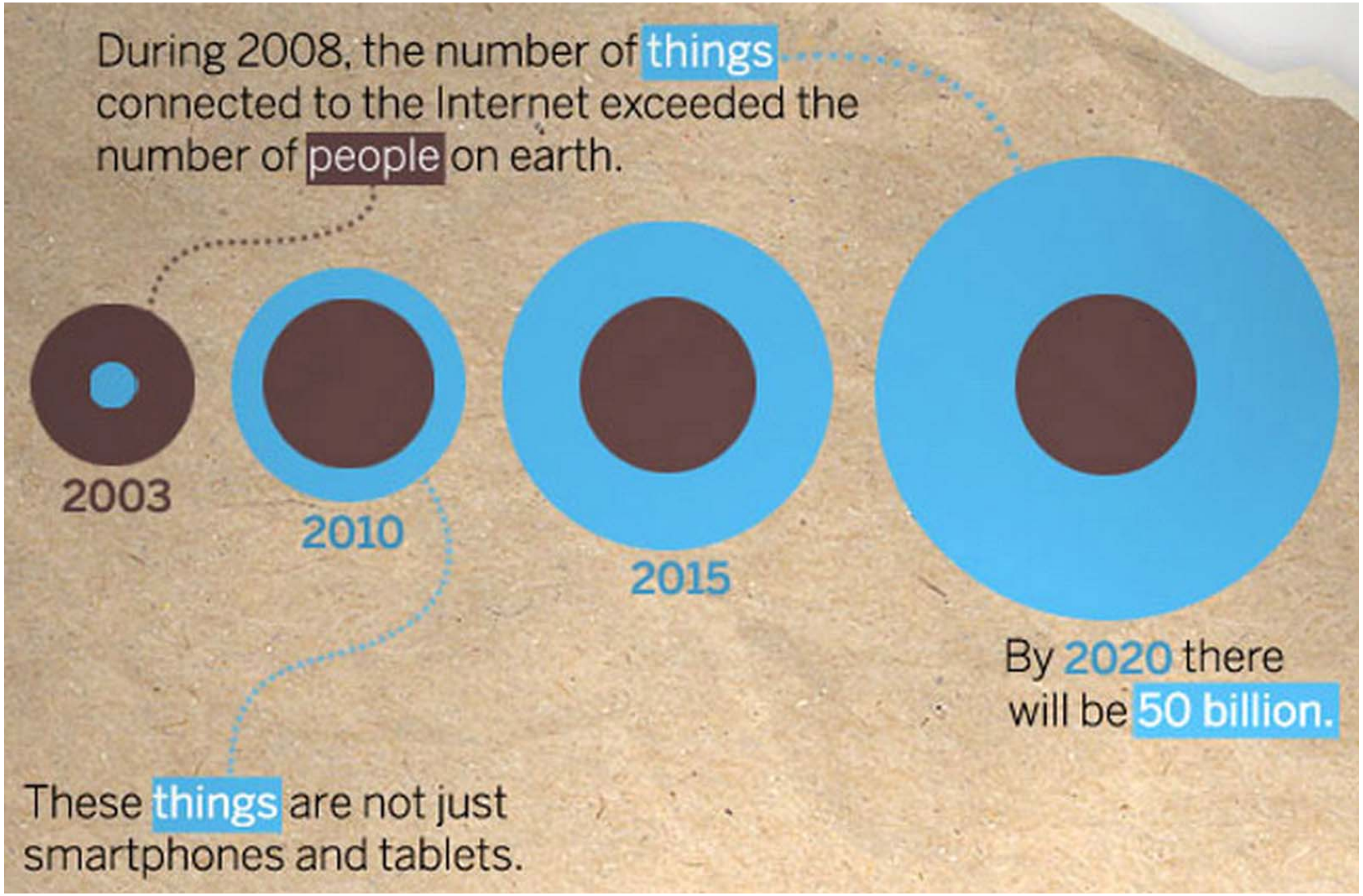
Belkin announced an updated version of its **WeMo** App for Android and iOS [with] new features for the WeMo Light Switch.
- Source: iClarified 04/17/2014

Why: Cost of IoT is Lower



- There used to be a cyber-physical divide between computing and sensing
- Diminishing IoT costs:
  - Integrating sensors into computing devices
  - Give them communications
  - Efficient remote configuration and management

Why: Market Size of IoT is HUGE



During 2008, the number of things connected to the Internet exceeded the number of people on earth.

These things are not just smartphones and tablets.

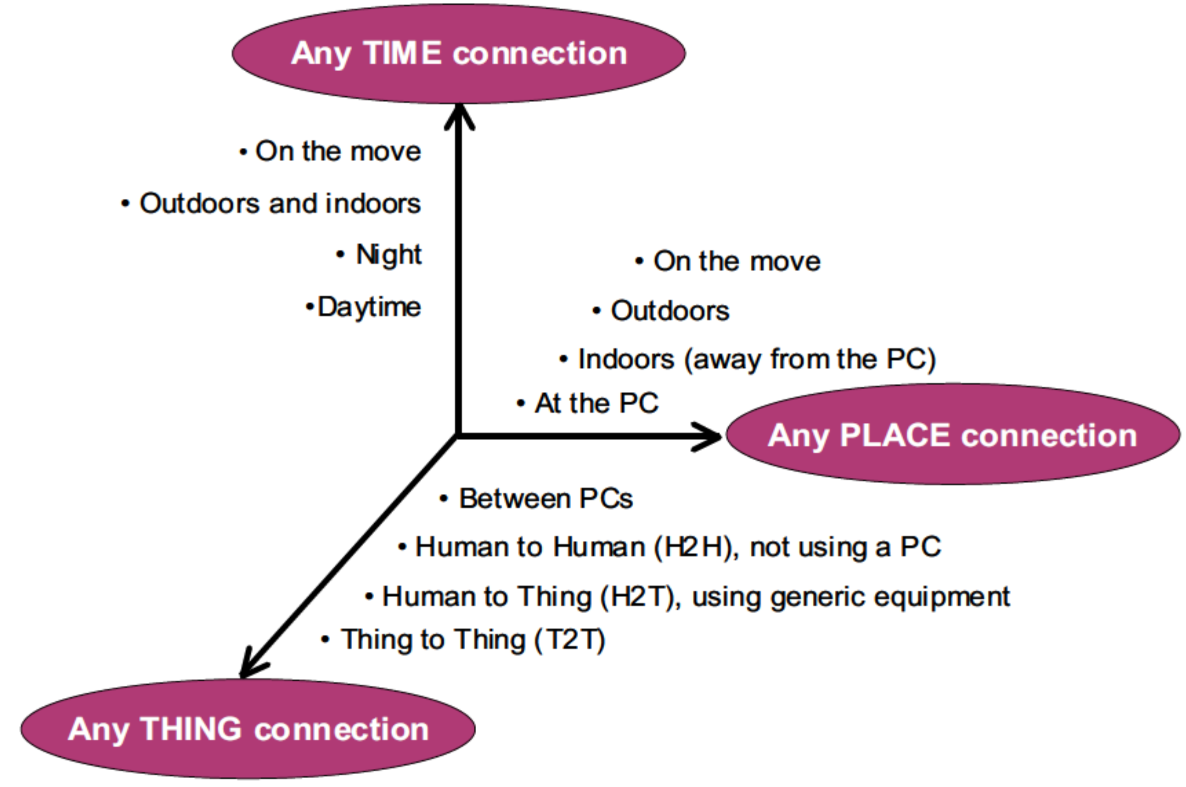
By 2020 there will be 50 billion.

There are 7.1 billion people in 2014 http://share.cisco.com/internet-of-things.html

Why: Convenience of IoT is Incredible

From any time, any place connectivity for anyone, we can have the connectivity for anything!

Figure 1 – A new dimension



Source: ITU adapted from Nomura Research Institute

IoT Challenges

- LA Times Story on CES, Jan 5, 2014

- ..., as the number of connected devices in people's lives proliferates, managing them will become increasingly taxing.
- Analysts point out that if the industry wants to keep the revolution going, it will have to … make it easier for all these devices to communicate with one another seamlessly and eventually work together with no human interaction.

http://www.latimes.com/business/la-fi-ces-internet-things-20140105,0,3796601.story

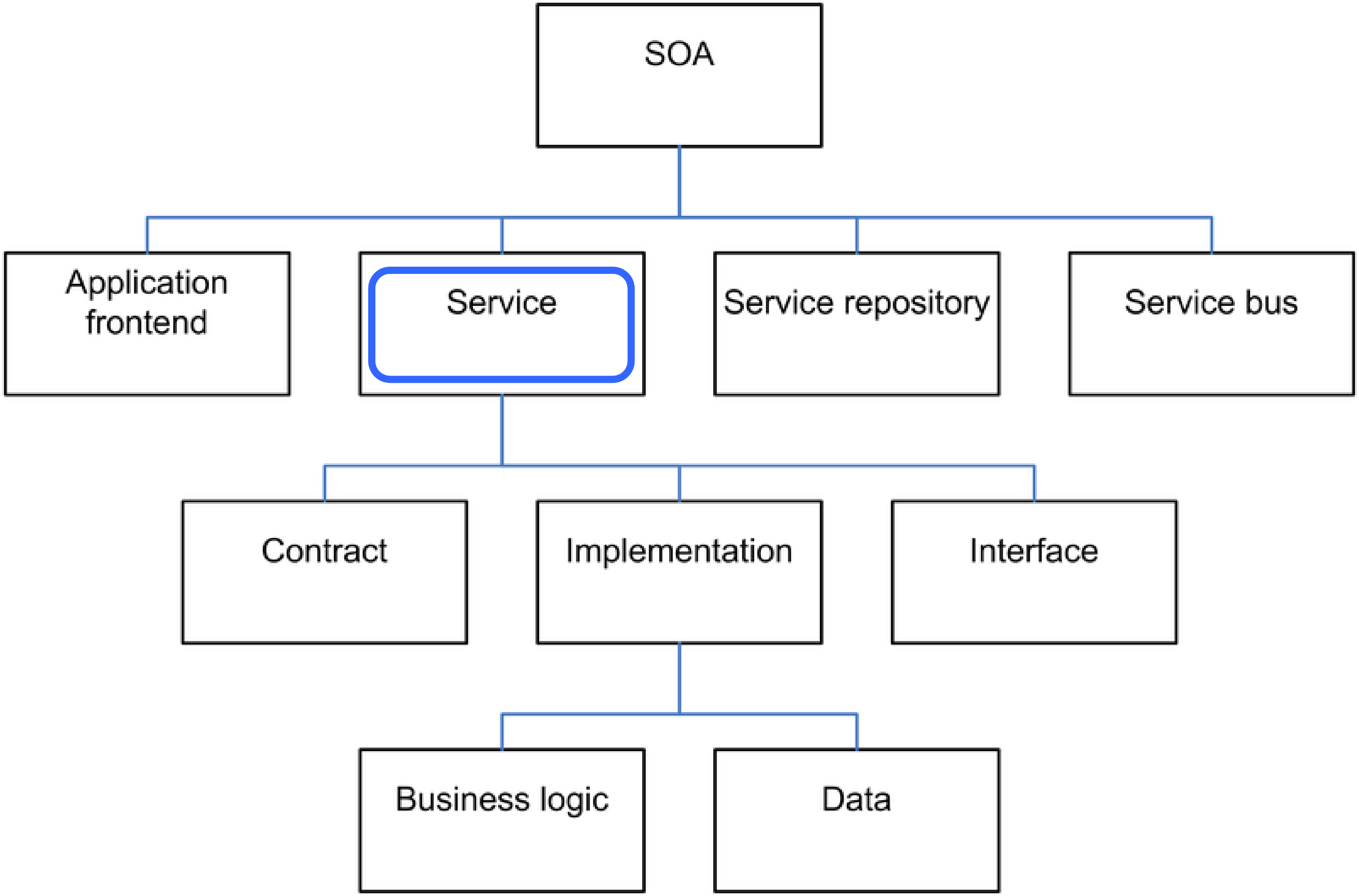
IoT Challenges (cont'd)

- LA Times Story on CES, Jan 5, 2014

- "We're entering the age of autonomy, we'll need these machines to make sense of all this information floating around us and make adjustments for us, whether it's just turning the lights on when we wake up or a cutting board that tells us how to adjust our caloric intake.
- It's too much information for people to manage on their own."

Shawn DuBravac, Chief economist and Director of research for the Consumer Electronics Assn. (CES)http://www.latimes.com/business/la-fi-ces-internet-things-20140105,0,3796601.story#ixzz2pXSiIaXm

Solution: Service-Oriented Architecture (SOA) for IoT

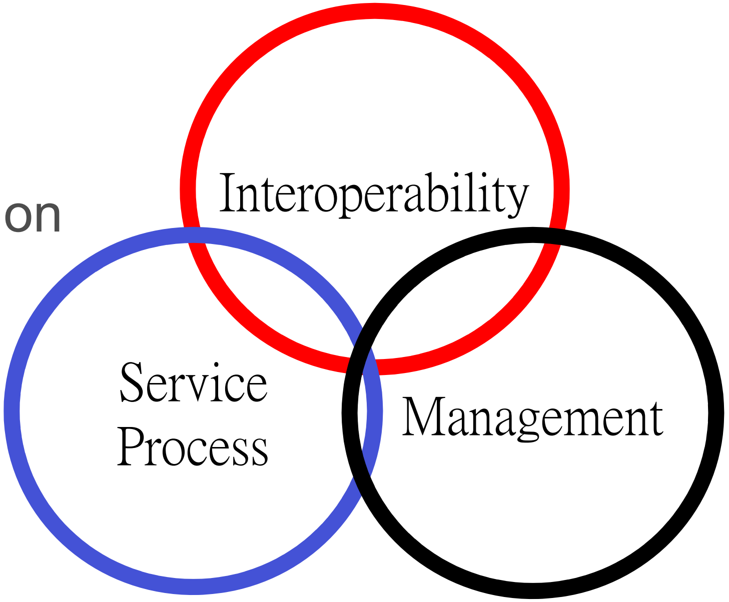


Source: Dirk Krafzig, Karl Banke, and Dirk Slama. *Enterprise SOA*. Prentice Hall, 2005

Benefits of SOA
(Service-Oriented Architecture)

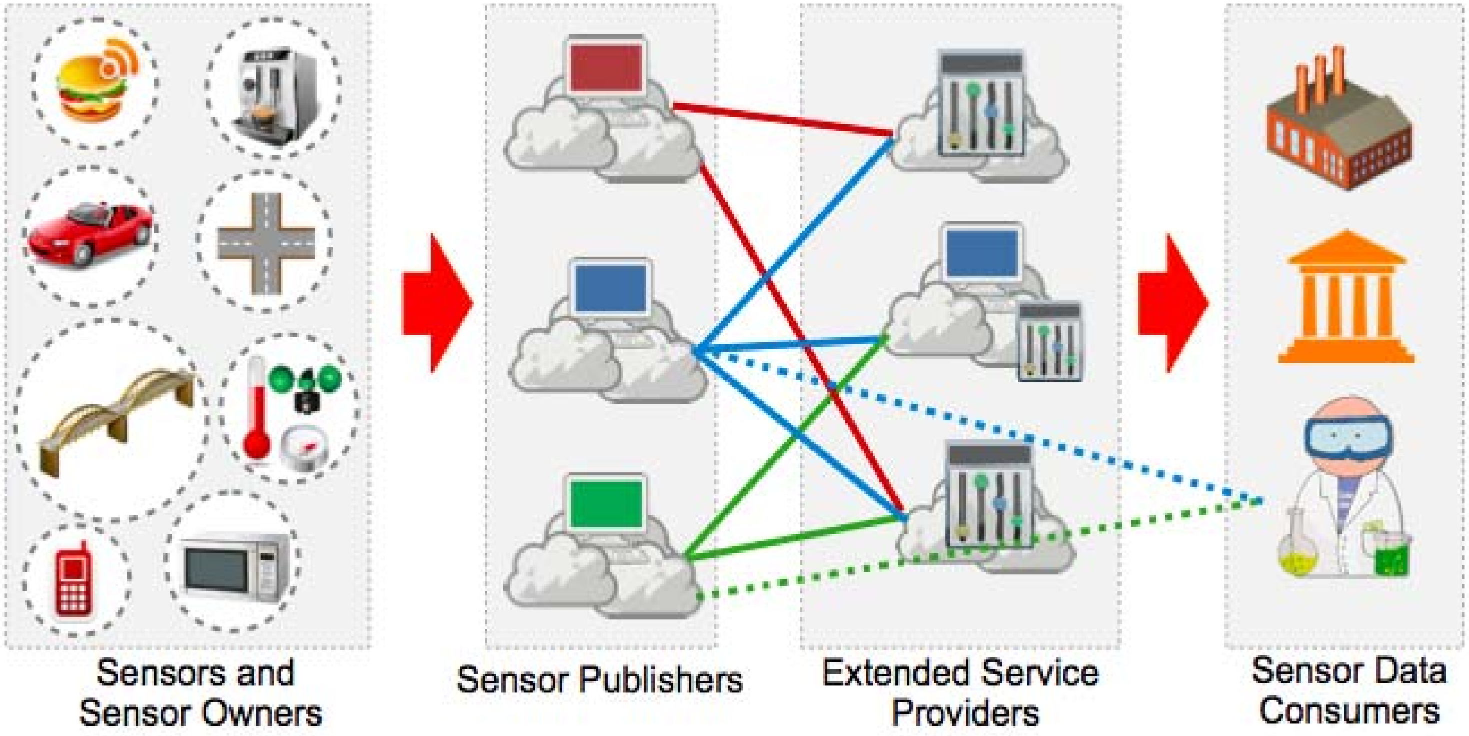
SOA provides:

- *Interoperability*: platform independent
  - Rapid application integration and discovery
- *Service Process*: dynamic process composition and integration
  - Automated and flexible service composition
- *Management*: vs. implementation
  - Allow people to concentrate on service management issues, not on service platform & technology issues



Service Models for SOT

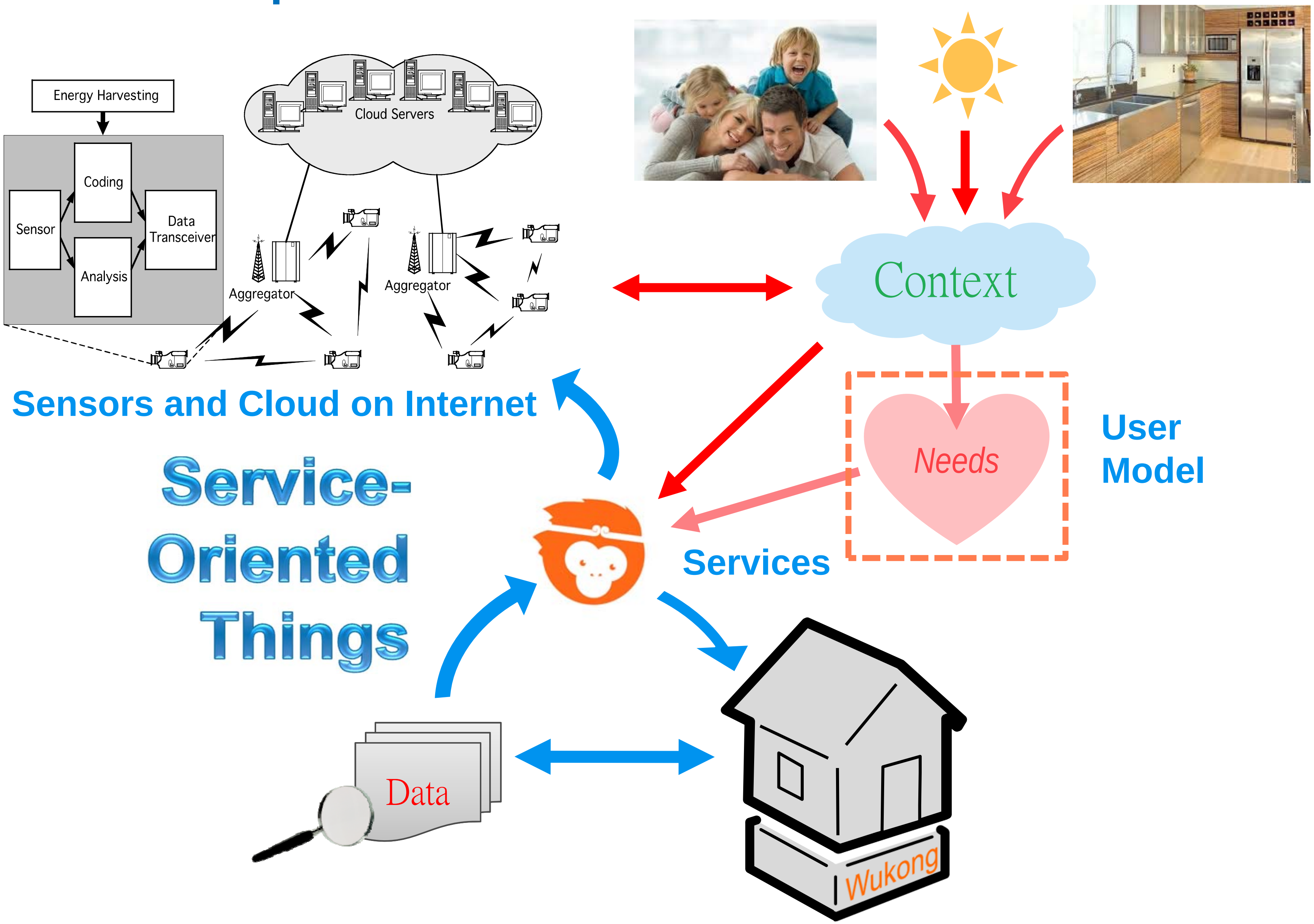
- Sensing as a service model
- Analytics as a service model
- Management as a service model



Adopting Service-Oriented Things (SOT)

- To compose *end-to-end* solutions for intelligent interaction and secure information sharing amongst a multitude of connected devices so that they can
  - efficiently sense data,
  - effectively communicate data,
  - collaboratively analyze context, and
  - intelligently serve users.

SOT Composition Vision



Sensors and Cloud on Internet

Service-Oriented Things

Context

Needs

User Model

Services

Data

Challenges of Building SOT

- Diverse Hardware Environment

  In future IoT systems, there will be many devices and platforms for hardware and software components.

- Evolving System Architecture

  IoT systems are deployed to serve for many years; the hardware and software components will evolve over time.

- Dynamic User Needs

  During the lifetime of IoT systems, users served by a system may change their needs, due to context, preference, lifestyle, etc.

- Service Composition Capability

  New services must be flexibly composed by the system components to add new capabilities integrating existing and new system components to create a new architecture.

The WuKong Project

The WuKong project is to build an *Intelligent Middleware* on platforms, so that IoT can *flexibly* and *dynamically*

- recognize sensor services and adapt to user policy;
- set up devices into service components;
- deploy context-dependent applications;
- adapt to the changing context and conditions.

WuKong also supports the new SOT paradigm

- *Given a user policy in a context, automatically provide THING selection and SERVICE deployment*

WuKong Service Definition

OOP	**Profile Framework (WKPF)**	**Identified by**
Class	WuClass	WuClass number (*global unique*)
Instance	WuObject	Port number (*node unique*)
Instance state variable	Property	Property number (*class unique*)

Definitions - WuClass

- Identified by a WuClass ID
- Defines a set of access functionalities
- Consists of a number of *properties* (state)
- We provide an initial set of WuClass definitions (similar to Zigbee)

WuClass Examples

- Temperature sensor
  - ✓ Refresh rate
  - ✓ Current Temperature
- Heater
  - ✓ On/Off
- Threshold Conditional
  - ✓ Operator
  - ✓ Value
  - ✓ Threshold
  - ✓ Output

Definitions - WuObject

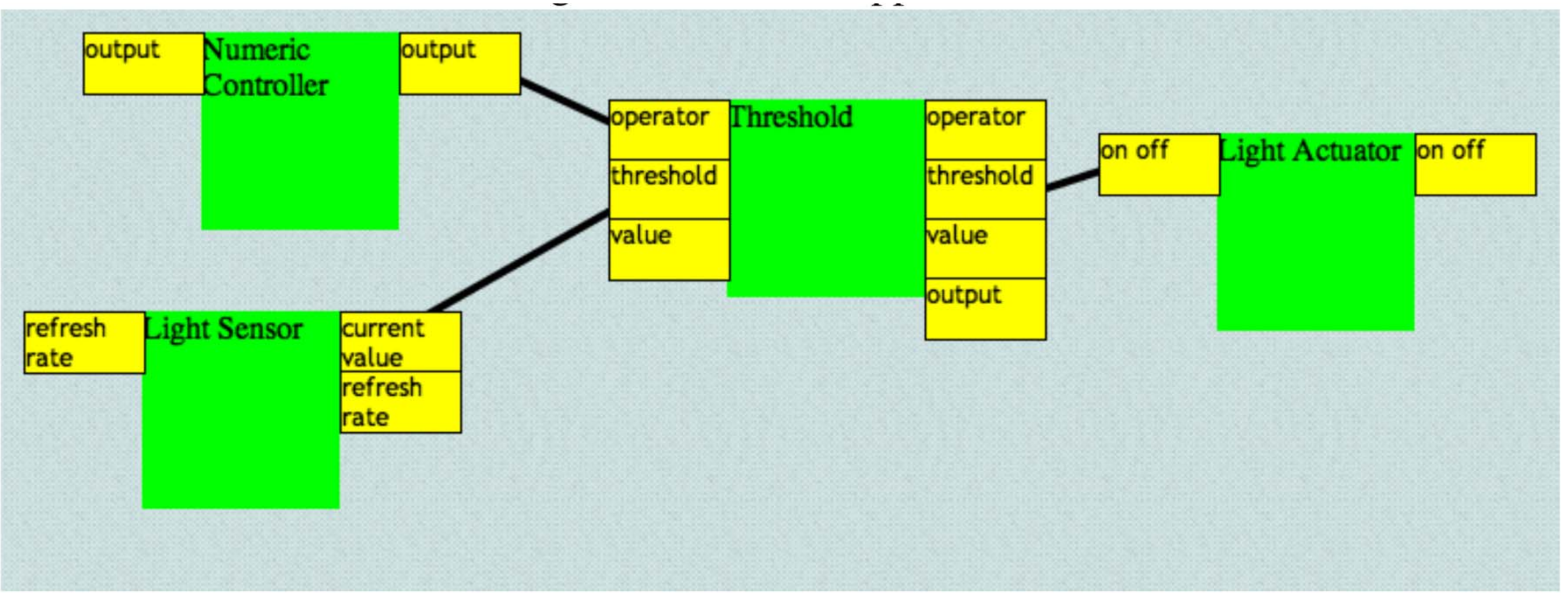
- Identified by a port number on a device (similar to IP ports)
- A node with two temperature sensors, would have two temperature sensor ports
- Some may be loaded by Master, some may be fixed (native)

Definition - Property

- Each property corresponds to a single value in an WuObject state
- Has a data type, and can be input or output
- For example, the Threshold WuClass has four properties
  - Operator: [LT|GT], input
  - Value: numeric, input
  - Threshold: numeric, input
  - Output: boolean, output
- Properties of the same type can be linked (value passing)
  - (Threshold.Output -> Heater.On/Off)

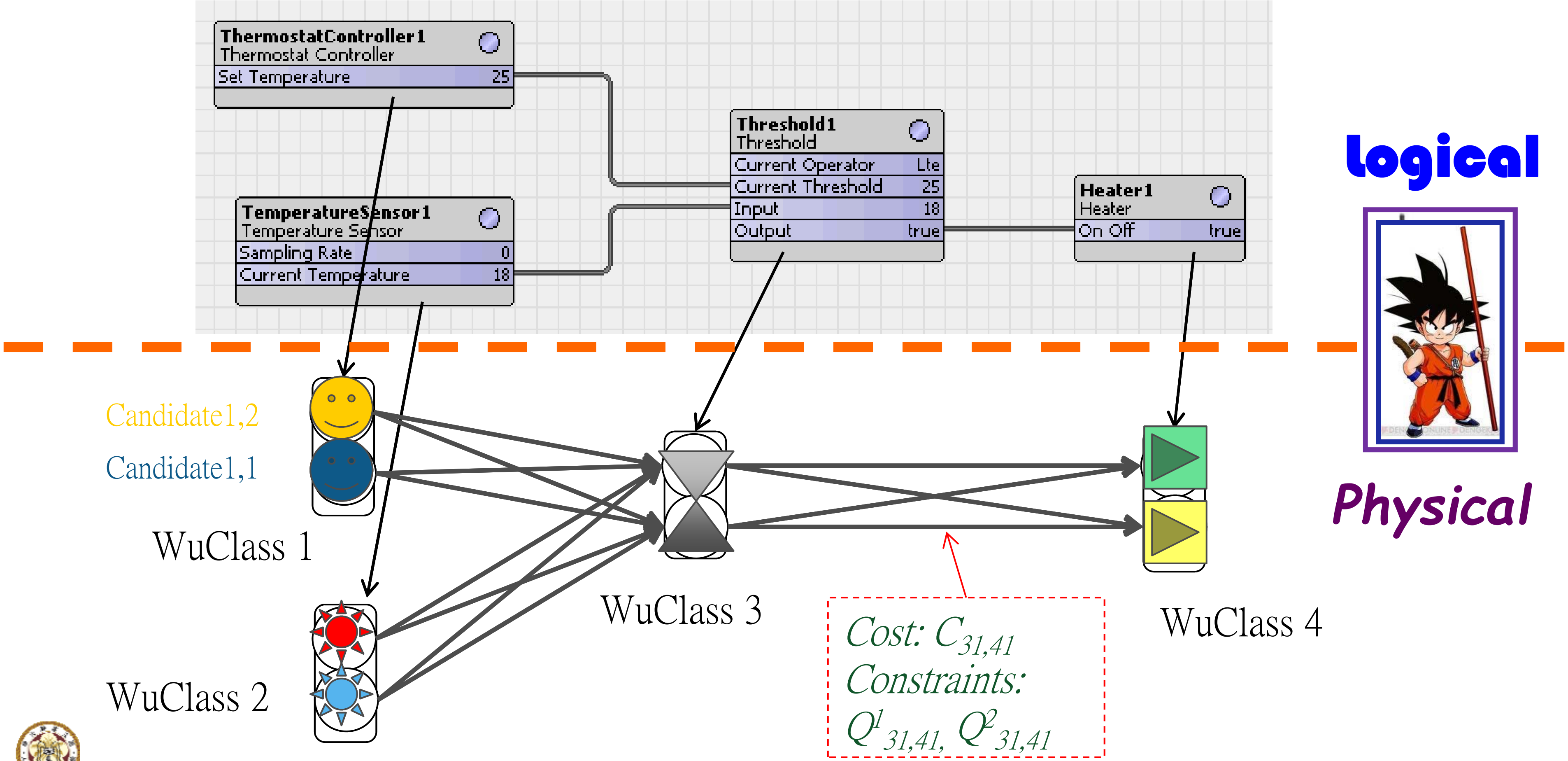
Process: Flow Based Program

A visual editor has been built for service process creation, for inspecting *WuKong Component Library*, and for specifying user policy.



FBP Mapping

Given logical flows that users defined, they must be mapped to some physical networks of sensor nodes for execution. We can use different mapping and selection.



ThermostatController1
Thermostat Controller
Set Temperature 25

TemperatureSensor1
Temperature Sensor
Sampling Rate 0
Current Temperature 18

Threshold1
Threshold
Current Operator Lte
Current Threshold 25
Input 18
Output true

Heater1
Heater
On Off true

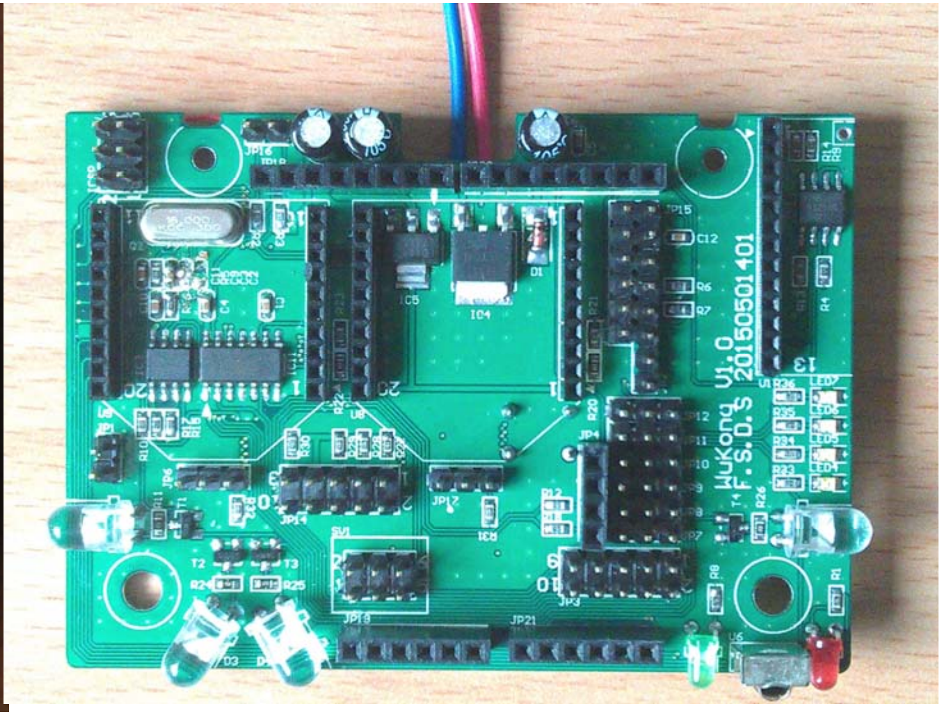
Cost: $C_{31,41}$
Constraints:
$Q^1_{31,41}$, $Q^2_{31,41}$

WuKong Profile Framework

- To configure an M2M network, Master needs to know what sensor resources are available on each sensor device.
- Master-Device protocol has three phases:
  1. Determine what devices are on the network (*discovery*)
  2. Determine what those devices can do (*identification*)
  3. Determine what those devices should do (*deployment*)
- After Master has "discovered" devices and queried them, each device reports its native profile that provides:
  - What resources are available on a device
  - How to access those resources

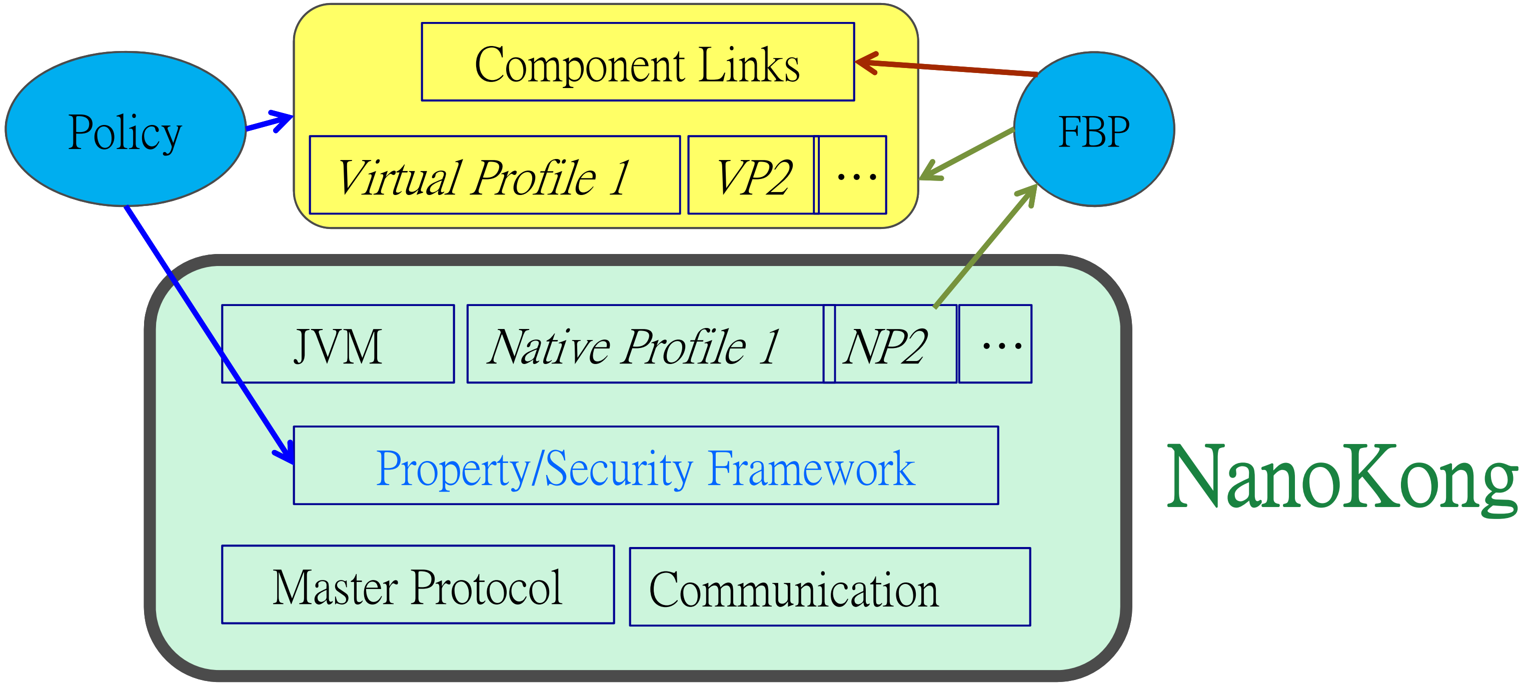
Hardware Used

- We have built our Arduino-based sensor platform, WuDevice, with ATMega 2560, 32KB EPROM, ZWave, WiFi, IR, 3 digital I/O and 3 analog I/O.



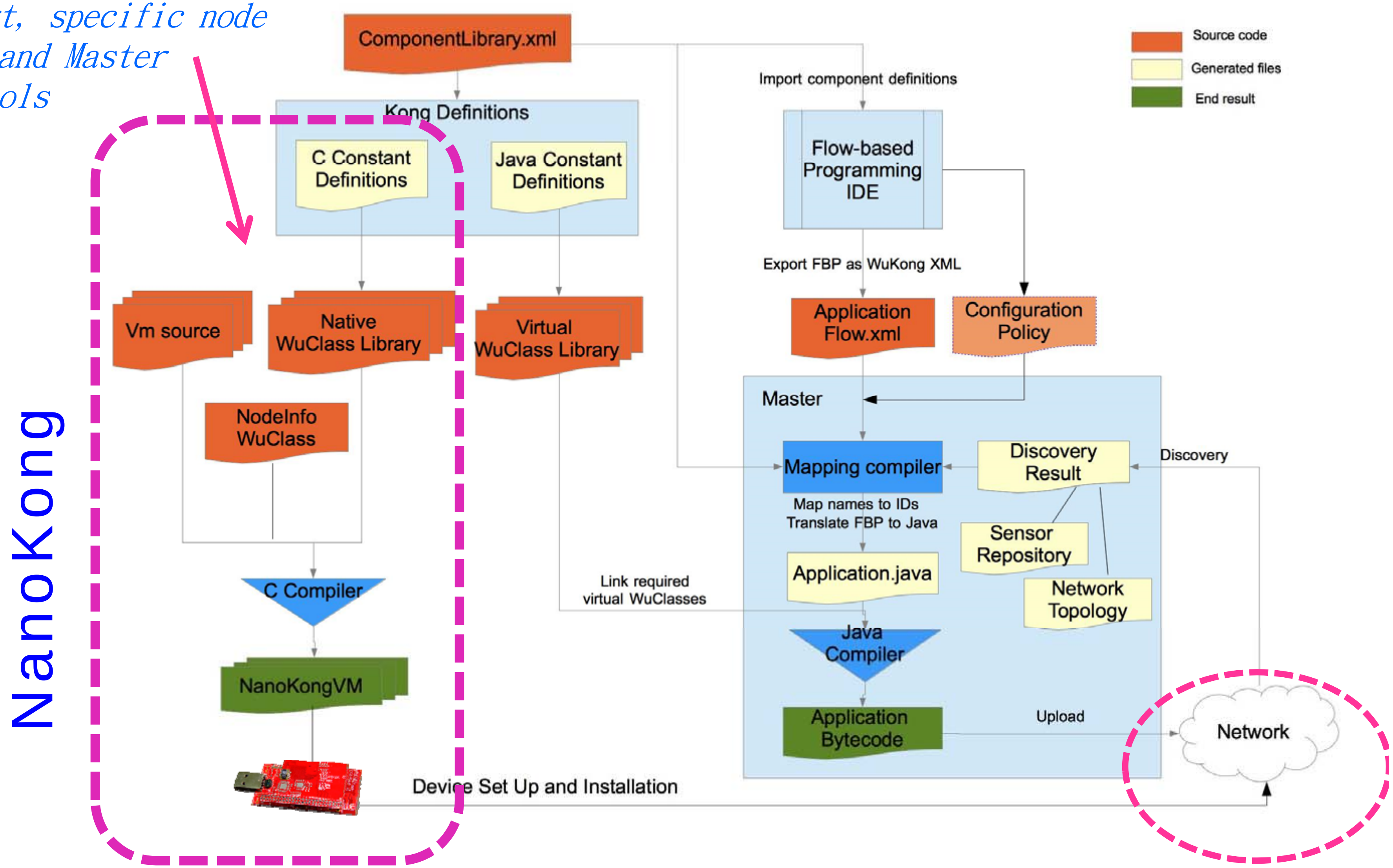
Building NanoKong Platform

- Every node is pre-loaded with sensor device drivers (native profiles), communication support, and JVM.
  - The pre-loaded code is called *NanoKong* (i.e. tiny WuKong)
- Security and other property framework can also be included.



Device Builder

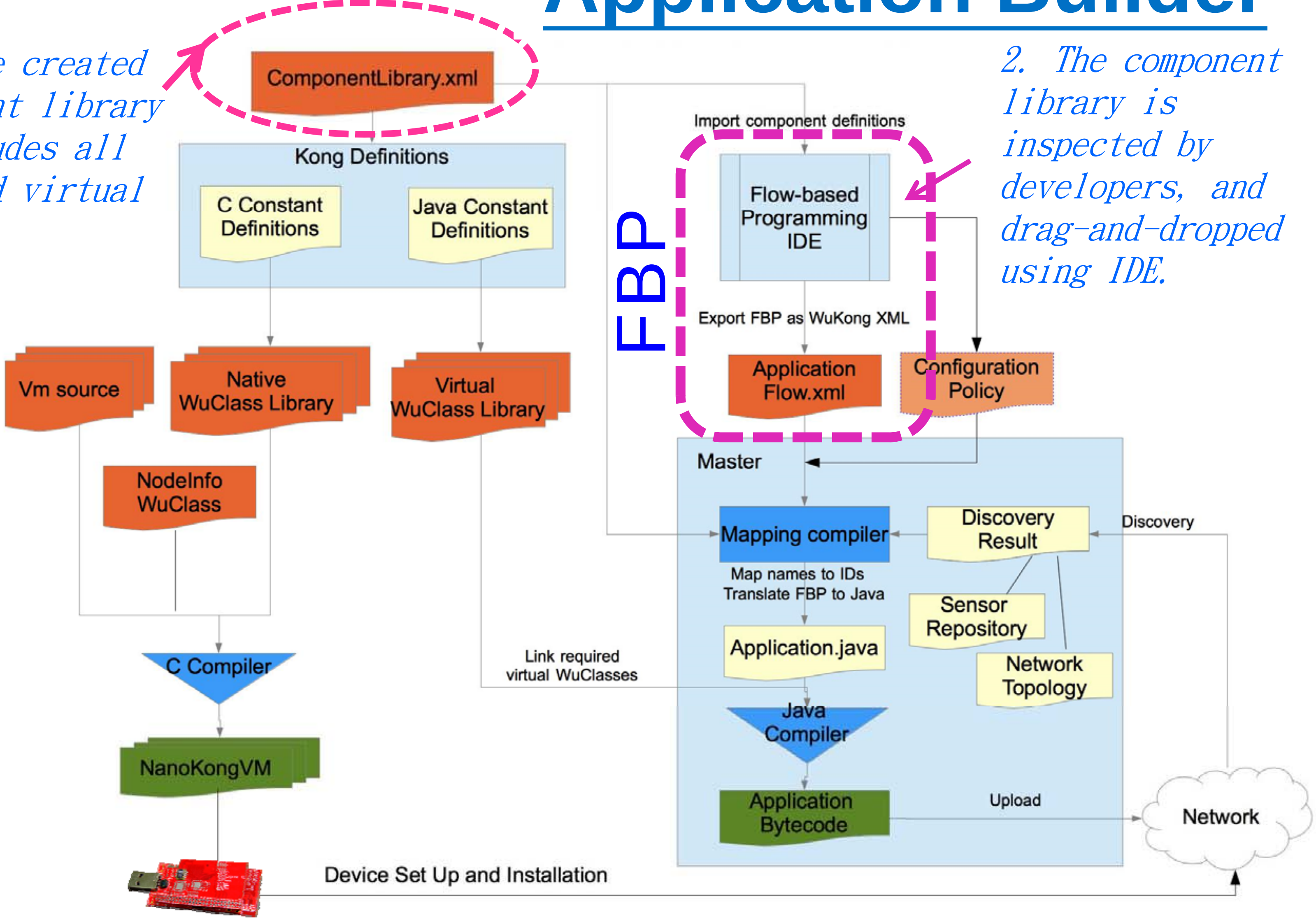
Each sensor device is loaded with native support, specific node info, and Master protocols



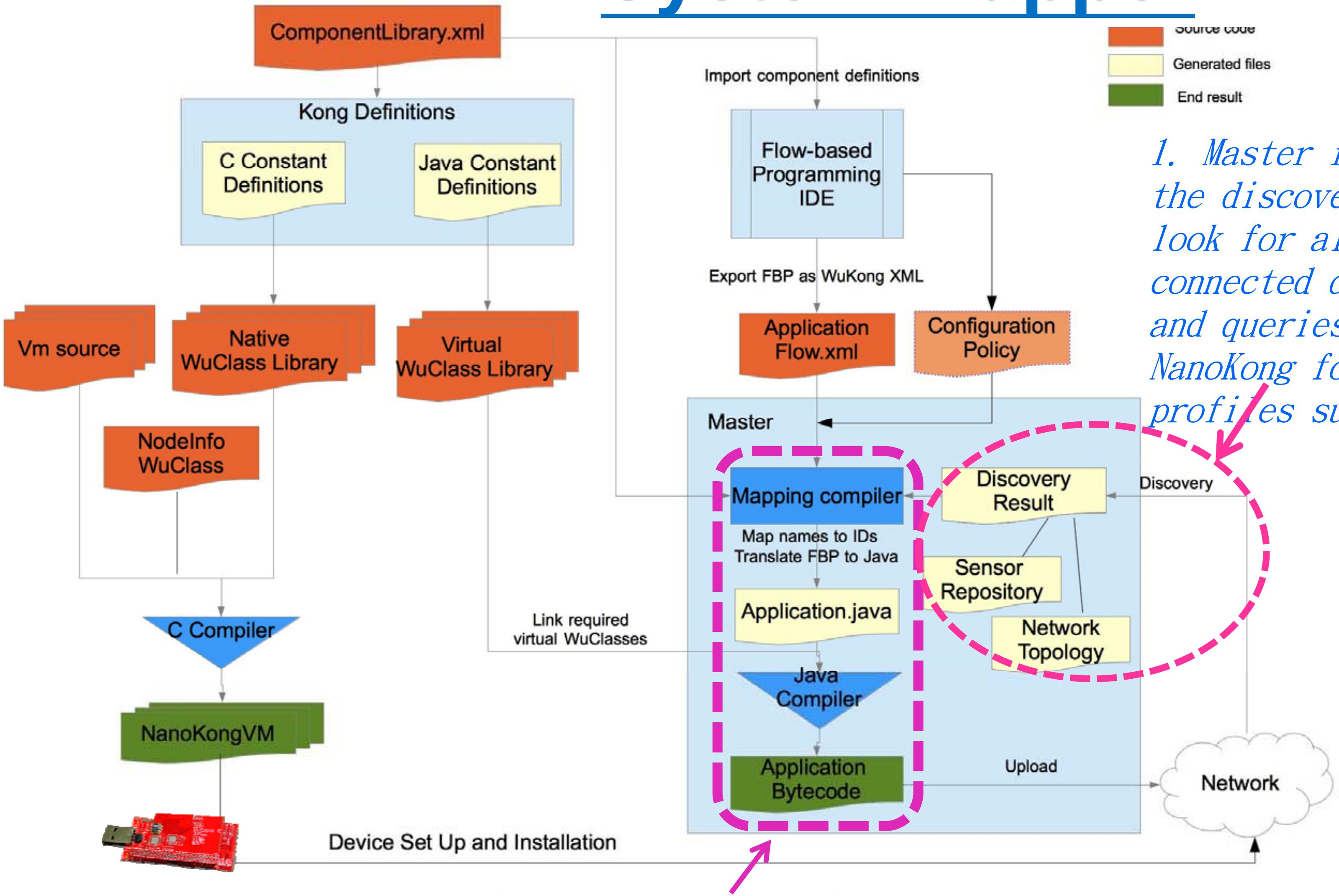
Application Builder

1. We have created a component library that includes all native and virtual profiles.

2. The component library is inspected by developers, and drag-and-dropped using IDE.



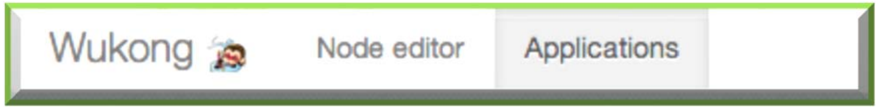
System Mapper

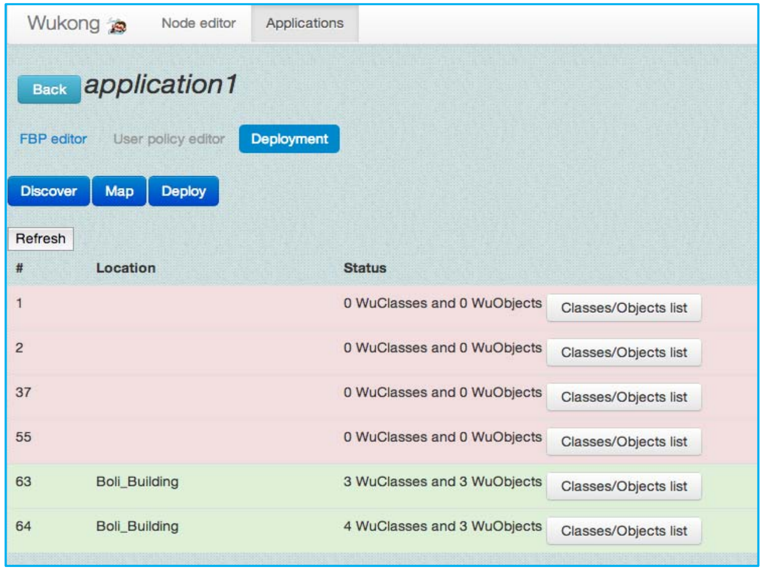


1. Master initiates the discovery to look for all connected devices and queries their NanoKong for the profiles supported.

2. Master uses mapping algorithms to selects different devices for a FBP, and upload the codes

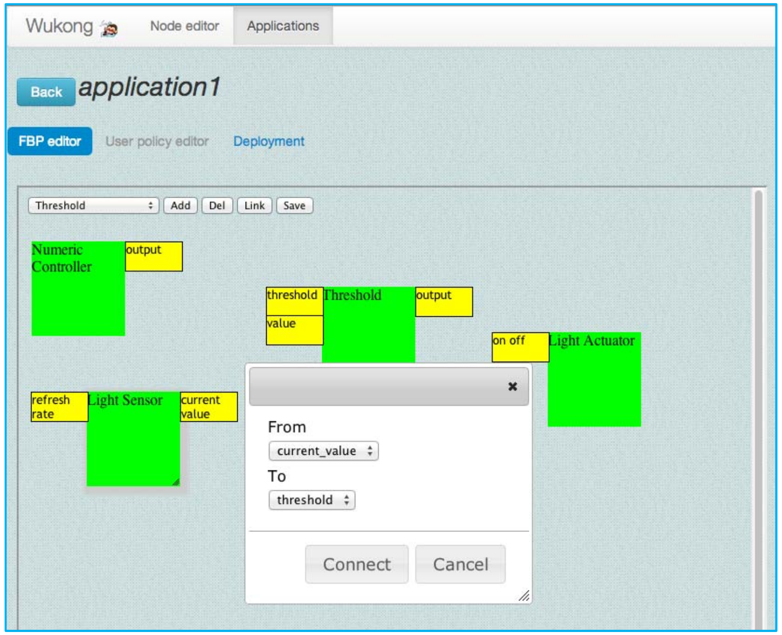
WuKong GUI





application1

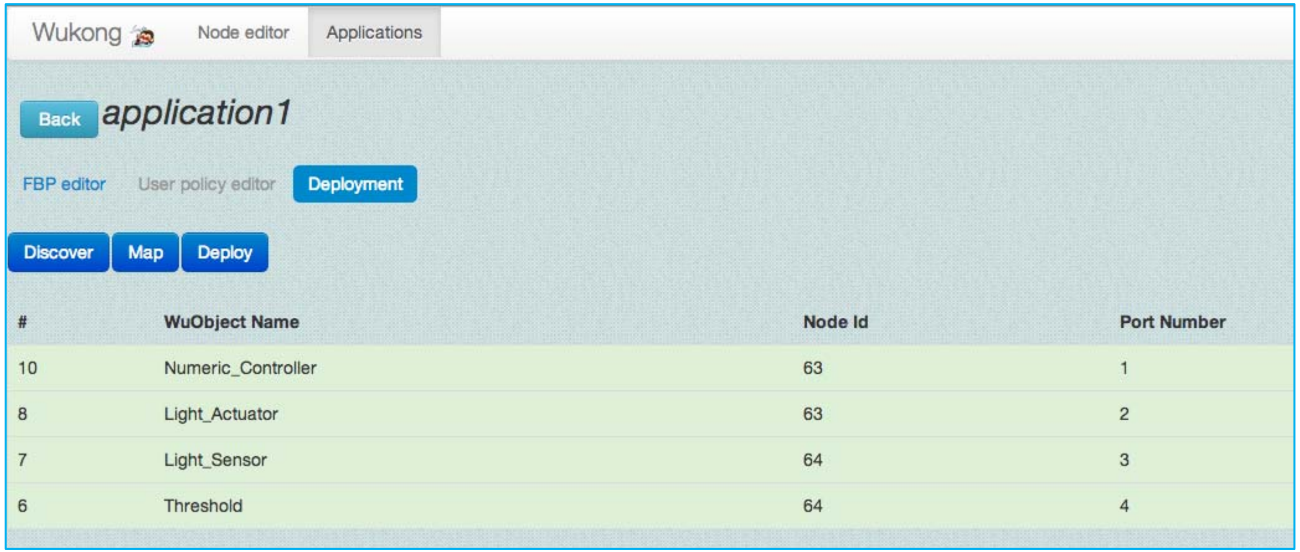
#	Location	Status	
1		0 WuClasses and 0 WuObjects	Classes/Objects list
2		0 WuClasses and 0 WuObjects	Classes/Objects list
37		0 WuClasses and 0 WuObjects	Classes/Objects list
55		0 WuClasses and 0 WuObjects	Classes/Objects list
63	Boli_Building	3 WuClasses and 3 WuObjects	Classes/Objects list
64	Boli_Building	4 WuClasses and 3 WuObjects	Classes/Objects list



application1

From: current_value

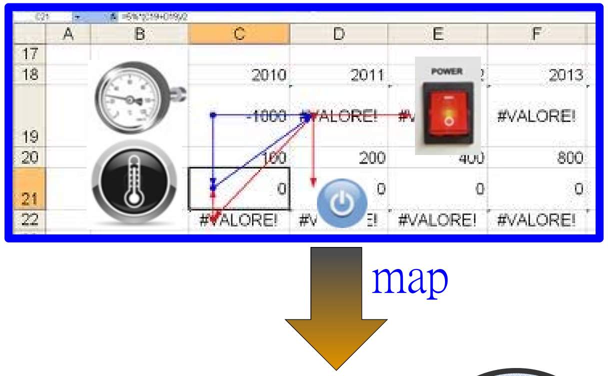
To: threshold



application1

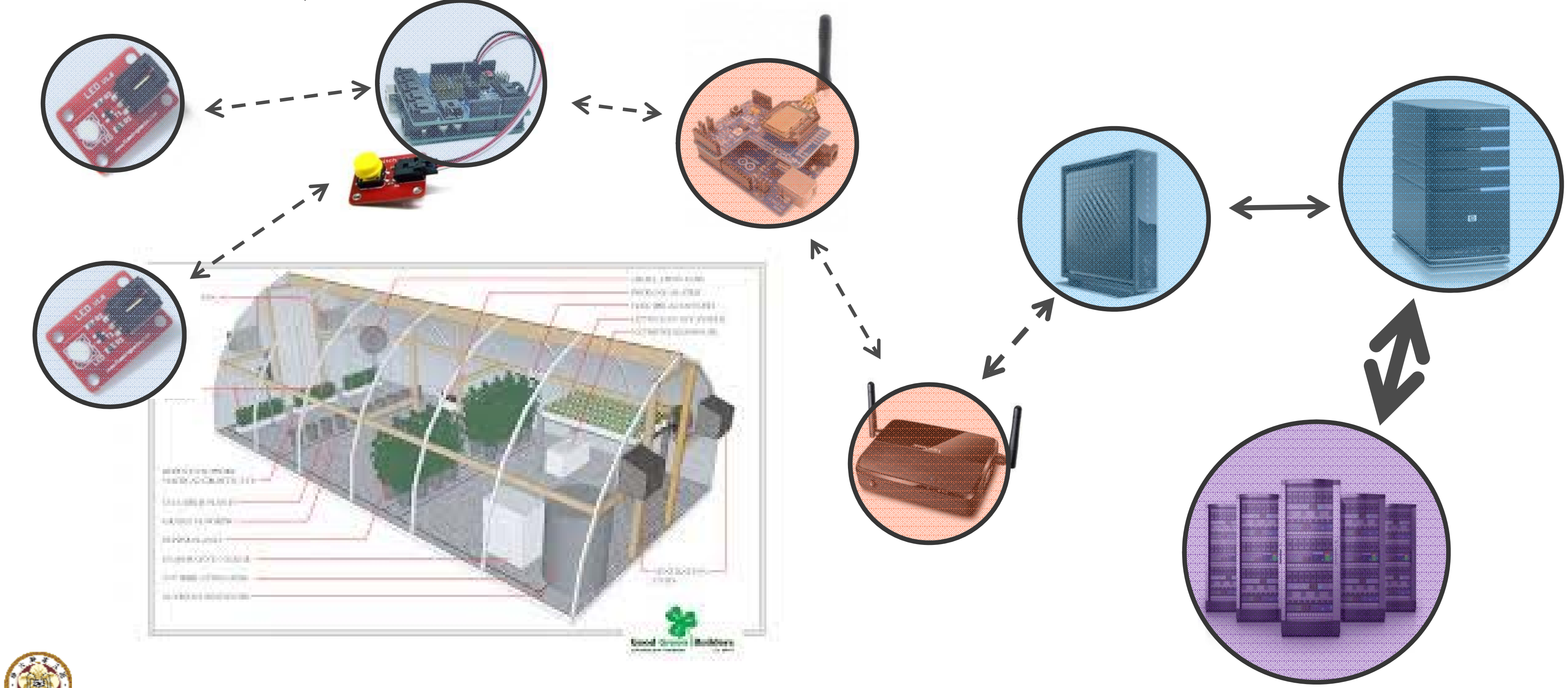
#	WuObject Name	Node Id	Port Number
10	Numeric_Controller	63	1
8	Light_Actuator	63	2
7	Light_Sensor	64	3
6	Threshold	64	4

WuKong Deployment



Given a virtualized-sensor FBP (flow-based program), WuKong will:

1. *Discover and identify the physical devices available*
2. *Map the virtual sensors to the devices*
3. *Deploy the software via network*
4. *Monitor data and make reconfiguration decisions*



Service Placement & Mapping Policy

Policy for IoT personalization, so that applications can be built, shared and used by many without re-programming.

Users can easily customize application operations by using simple policy language/interface.

- Location: Service placement decision must follow user's need for the context
- Communication: Service placement decision changes total communication behavior.
- Energy: Service placement decision changes the energy consumption on each device.

WuKong Meeting SOT Needs

- Diverse Hardware Environment
  - Profile has the knowledge of its capability and controllability
- Dynamic User Needs
  - User defines the service policy on the application to be deployed
- Evolving System Architecture
  - All hardware and software components are defined by their profiles, may evolve over time.
- Service Composition Capability
  - Services defined as FBP components can be dynamically composed and redeployed
- Multiple Objective Optimization
  - Mapper algorithms are designed for multiple objective optimization using user policy and device capabilities

Conclusion: Use SOA to Proliferate IoT

SOT makes it easier to build flexible and dynamic IoT that fits user’s needs.

1. Detect: (in real time) profile, platform, network capabilities
2. Define: (for user personalization) policy, FBP mapping, routing, location, energy QoS
3. Deploy: (with optimal setup) device and network selection, dynamic adoption, user experience

THANK
YOU

Classical Method of Building Sensor IoT

- tool chain: IDE, compiler, debugger
- microcontroller is programmed and executes the code
- radio chip is not programmed, but controlled by microcontroller, usually via SPI which sets/reads registers
- compiled code is loaded to the microcontroller using bootloader or JTAG
- protocol stack may be precompiled and available through API or available as library
- operating system (not needed for simple tasks)
- virtual machine (optional)

Wu-Kong：A Classical Chinese Epic Hero

- *Sun Wukong*, also known as the *Monkey King*, is a heroic character in the classical Chinese epic novel *Journey to the West*.
  - born from a stone, acquired super powers through many masters.
  - became a loyal guard for a monk travelling to the Western heaven.
- Wukong knows 72 **transformations**, into various animals and objects
- His hair has magical powers. Each can transform into a **clone** of himself, or various subjects, weapons, or animals.
- Wukong uses a **versatile** golden-banded staff, which could change its size, multiply itself, and fight according to the master.
- Wukong rides on a **cloud** that can travel at an extremely high speed, bringing him virtually anywhere at any time.