



jHome

Java EE Home Automation API

Having Fun with Home Automation and Java EE

Vinicius Senger @vsenger

Yara Senger @yarasenger



Yara and Vinicius Senger



Who we are?

Agenda



- Introduction
- Home Automation
- Arduino & open-source hardware
- jHome
 - Hardware and firmware
 - Protocol
 - jHome Server and Clients
- DEMO's

Before Starting...



- Thanks audience!
- Thanks Oracle and other sponsors!
- Special thanks for the "real heroes":
 - Mattias and Jfokus Team!



Introduction



- jHome = complete open-source platform for home automation
- Project history: coffee machine automation
- LIFE automation: what is important to YOU!
- Target community: do it yourself consumers
- Not a product competitor: z-wave, X10, etc.;
- Open-source hardware and software;
- BETA!

Fun with Java EE!



- There's life beyond database
- jHome is a funny way to learn Java EE 6:
 - Timer Service to schedule wall jackets
 - Injecting Coffee machine object into the servlet;
 - jQuery and REST to read sensors;
 - Using Webservice to control your house with other platforms;
 - Extreme feedback;

Not easy, VERY EASY!



- Children understandable:

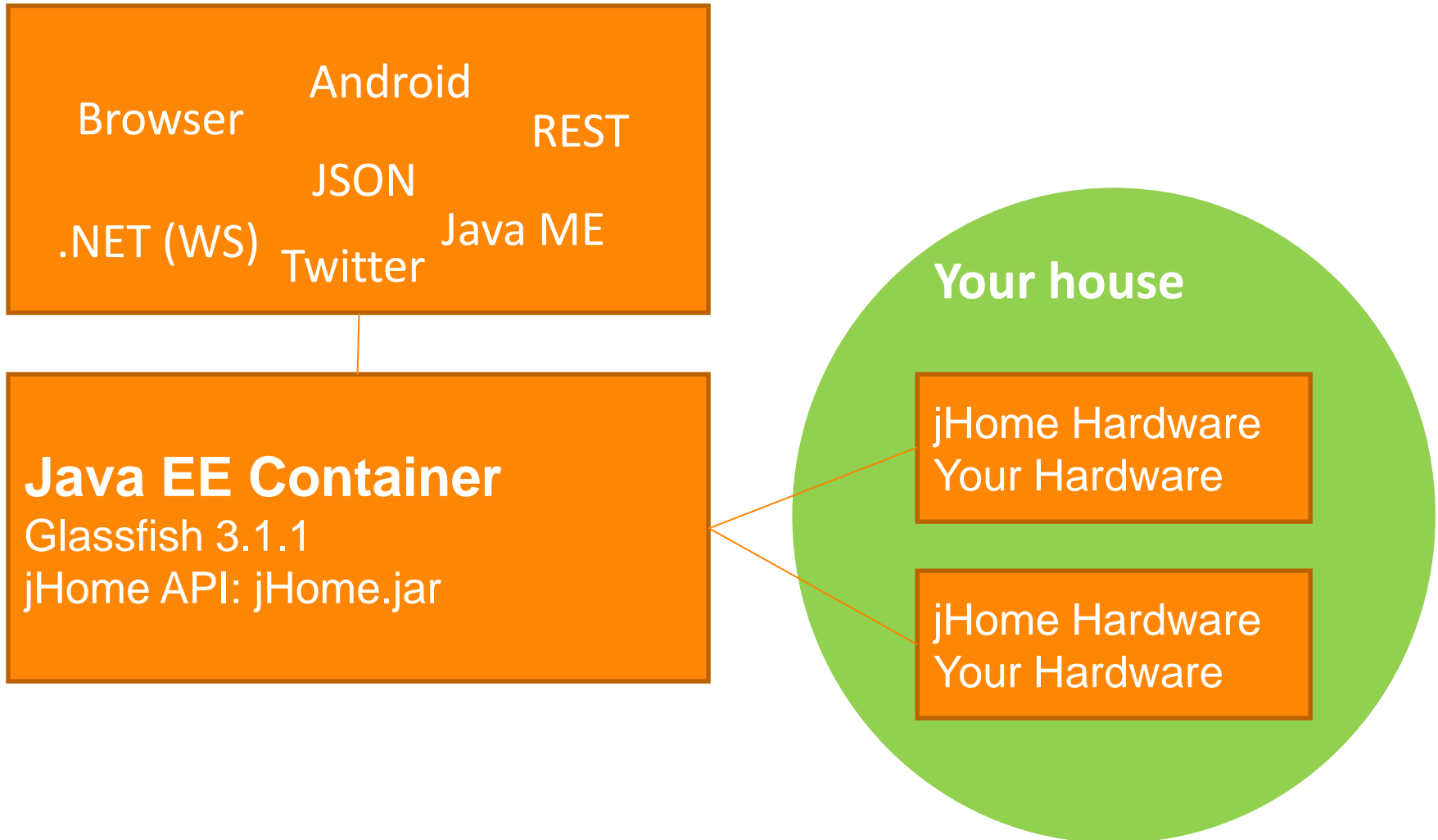


Arduino – Processing - Scratch Integration

Oracle Innovation Award 2011



Architecture Overview



Home Automation



- Market with many "standards":
 - Many offers with different protocols;
- Almost are proprietary solutions;
- All about controlling lamps, dimmer, monitoring and scheduling features;
- Hardware is always hided;
- z-wave and x-10 are booth good solutions:
 - z-wave: radio-frequency based;
 - x-10: signals through the power supply

jHome Vs. Market Solutions



- jHome can work with any protocol;
- jHome is more about specific automation:
 - I just want to automate my existent electric gate!
- jHome is a **COMPLETE** open-source platform:
 - Open-source firmware
 - Simple and easy to extend protocol
 - jHome Server monitors and provide services
 - Open-source hardware based on Arduino
 - Mobile Clients

Beyond home automation...

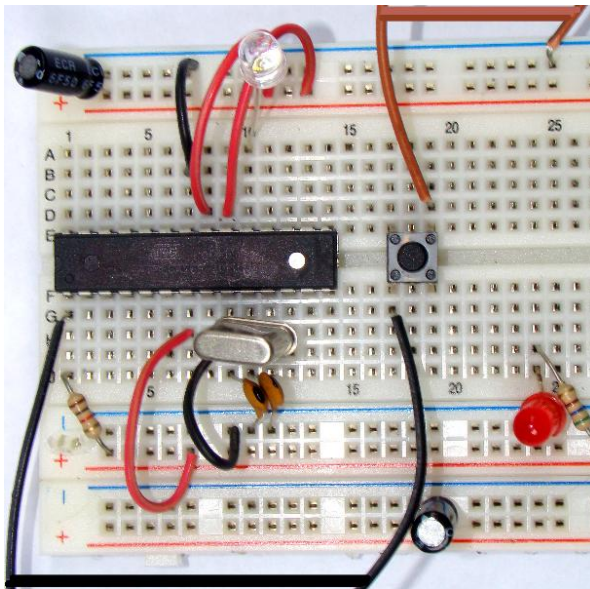


- Home-brewing automation
- Car / Boat automation
- Garden automation
- Data-center automation
- ...

Arduino



- Arduino is a electronic prototyping platform;
- Easy, cheap and BIG community;
- Is C based, but "almost" Java;

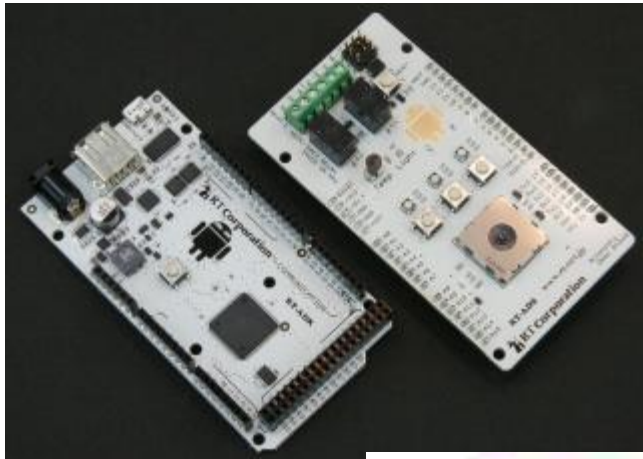


Arduino



- Bringing back the hardware control to the users;
- Is causing a new electronic fever (like 80's);
- No more assembly!
- More than 100 different types / branches of Arduino boards;

Arduino



What people are doing with Arduino?



- Robotics;
- Satellites;
- Home Automation;
- "Segway" open-source;
- Human instrumentation: sleep, alcohol, drugs, heart monitor etc.
- Custom cell phone;
- Inventions;

Some cool projects



- Boat control using phone accelerometer:
<http://vimeo.com/globalcode/automacaonauticaarduinoandroid>
- Futuristic dimmer:
<http://www.youtube.com/watch?v=wKBqFWvVEQI>

Take care with Arduino!



- I started with one board and now...

Take care with Arduino!



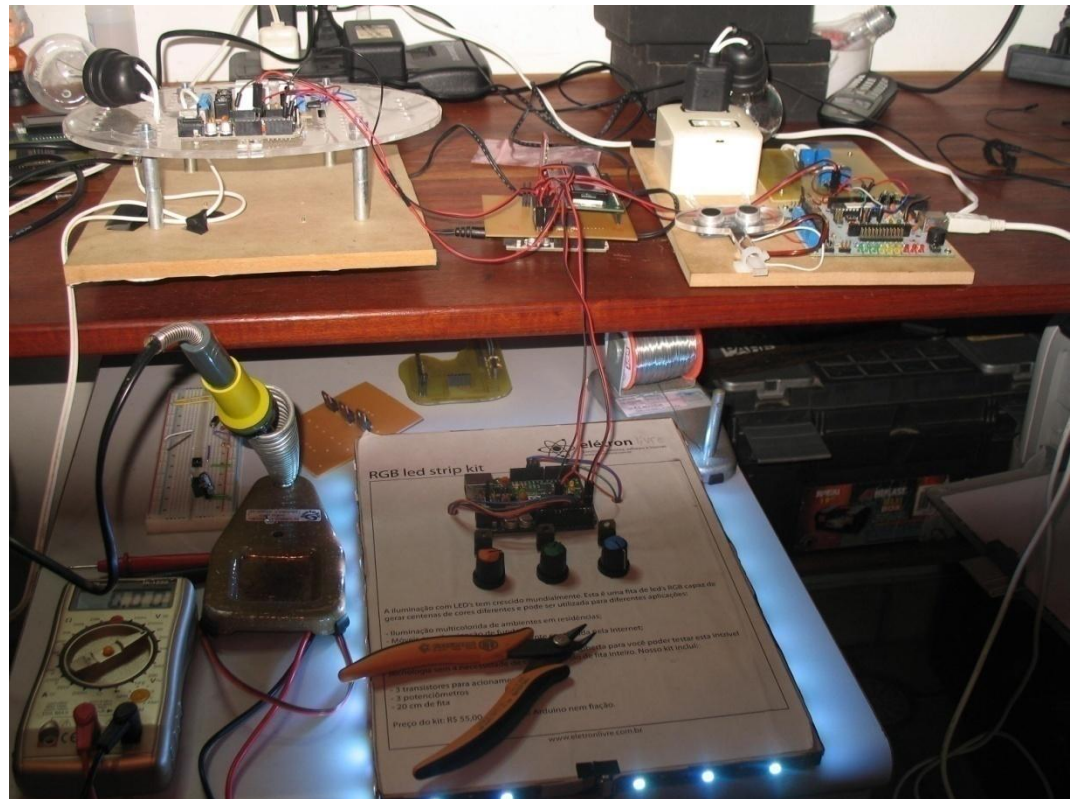
- I started with one board and now...
 - everything seems to be so useful..



Take care with Arduino!



- I started with one board and now...
 - no space anymore



Now serious, take care



- 230 volts + snow + barefoot = can kill;
- Don't work alone with more than 200 volts;
- Smoke test here is a serious test:
 - Work – ok, finish
 - Do not work, no smoke – you can fix
 - Do not work, smoke – you may fix it
 - Fire – forget about

jHome



- Five main components:
 1. jHome Device / Gadget – Hardware
 2. jHome Firmware – C Code
 3. jHome Protocol - TEXT
 4. jHome Server – Java EE
 5. jHome Clients – Java ME / JSF / jQuery / Android

jHome Device / Gadget



- Based on Arduino;
- Different sizes:
 - big, small and nano;
- D.I.Y. automation is not about having cheaper solutions:
 - Prototyping platforms are more expensive than final products!

Device Components



Actuator	Sensor	Communication
Lamps	Light	USB / Serial
Motor	Temperature	Bluetooth
LED's	Humidity	Zigbee
Fan	Accelerometer	Ethernet
Coffee machine	Presence	Wifi
	Reed	IR
	etc...	Radio frequency

Basic / Cheaper jHome Device



- Arduino – U\$ 35
- USB Communication - Free
- 2x Relay board – U\$ 15
- Light sensor – U\$ 2
- Temperature sensor – U\$ 5
- RGB LED + Power board – U\$ 30
- 12 volts power supply – U\$ 10
- *TOTAL COST: U\$ 97*

DEMO 1



- Controlling basic device using:
 - Command line
 - jHome Server with jQuery UI
 - Scheduling

DEMO 2



- Twitter Client:

@jhomeautomation turn on coffee @jfokus

@jhomeautomation turn off coffee @jfokus

@jhomeautomation red on @jfokus

@jhomeautomation red off @jfokus

@jhomeautomation green on @jfokus

@jhomeautomation green off @jfokus

@jhomeautomation blue on @jfokus

@jhomeautomation blue off @jfokus

Firmware / Arduino code



```
#include "Device.h"

Device homeDevice=Device("old-central");

void setup() {
    homeDevice.add("red",          PWM,    5);
    homeDevice.add("green",        PWM,    6);
    homeDevice.add("blue",         PWM,    9);
    homeDevice.add("relay1",        DIGITAL, 2);
    homeDevice.add("relay2",        DIGITAL, 4);
    homeDevice.add("light",         ANALOG, 3);
    homeDevice.add("temp",          ANALOG, 2);
    Serial.begin(115200);
}

void loop() {  homeDevice.loop(); }
```

Firmware / Arduino code



...

```
homeDevice.add("blue",          PWM,          9) ;
```

```
homeDevice.add("relay1",       DIGITAL,      2) ;
```

```
homeDevice.add("light",       ANALOG,       3) ;
```

...

Component name

Component type:

digital, analag, PWM, lib / custom component

Port number

jHome Protocol



- Devices are self described based on firmware declaration;
- Same protocol: Serial, HTTP, Bluetooth and Zigbee, etc.;
- Example:
 - relay1?1 – turn on relay 1
 - relay1?0 – turn off
 - red?128 – 50% red light
 - red?255 – 100% red light
 - light – read light sensor

jHome Protocol



relay1?1 – turn on relay 1

relay1?0 – turn off

red?128 – 50% red light

red?255 – 100% red light

light – read light sensor

```
homeDevice.add("red", PWM, 5);
homeDevice.add("green", PWM, 6);
homeDevice.add("blue", PWM, 9);
homeDevice.add("relay1", DIGITAL, 2);
homeDevice.add("relay2", DIGITAL, 4);
homeDevice.add("light", ANALOG, 3);
homeDevice.add("temp", ANALOG, 2);
Serial.begin(115200);
```

USB / Serial Device Support



```
#include "Device.h"
#include "etherShield.h"
#include "ETHER_28J60.h"
Device homeDevice=Device("old-central");

void setup() {
    homeDevice.add("red",          PWM,    5);
    ...component declaration...
    homeDevice.add("temp",        ANALOG,  2);
    Serial.begin(115200);
}

void loop() {  homeDevice.loop(); }
```


Ethernet Support



```
#include "Device.h"
#include "etherShield.h"
#include "ETHER_28J60.h"
Device homeDevice=Device("old-central");
Ethernet ethernet=Ethernet(&homeDevice);
void setup() {
    homeDevice.add("red",          PWM,      5);
    ...component declaration...
    homeDevice.add("temp",        ANALOG,    2);
    Serial.begin(115200);
    int ip[]={192,168,1,15};
    homeDevice.startNetwork(ip);
}
void loop() {  homeDevice.loop(); }
```

Communication Vs. Cost



Type	Recommend Usage	Cost
USB / Serial	Development time / Geeks House	U\$ 0
Bluetooth	Temp. communication / Develop.	U\$ 40
Ethernet	Real home / office implementation	U\$ 20 – U\$ 40
Zigbee	High reliability / REAL impl.	U\$ 25 p.p.
Wifi	Same as ethernet	U\$ 120
IR	Alternative communication	U\$ 5
Radio frequency	Simple wireless communication and as alternative	U\$ 15

DEMO 3



- Ethernet Gadget
 - Controlling via Browser
 - Discovery / Admin UI
 - Mobile App

jHome Server



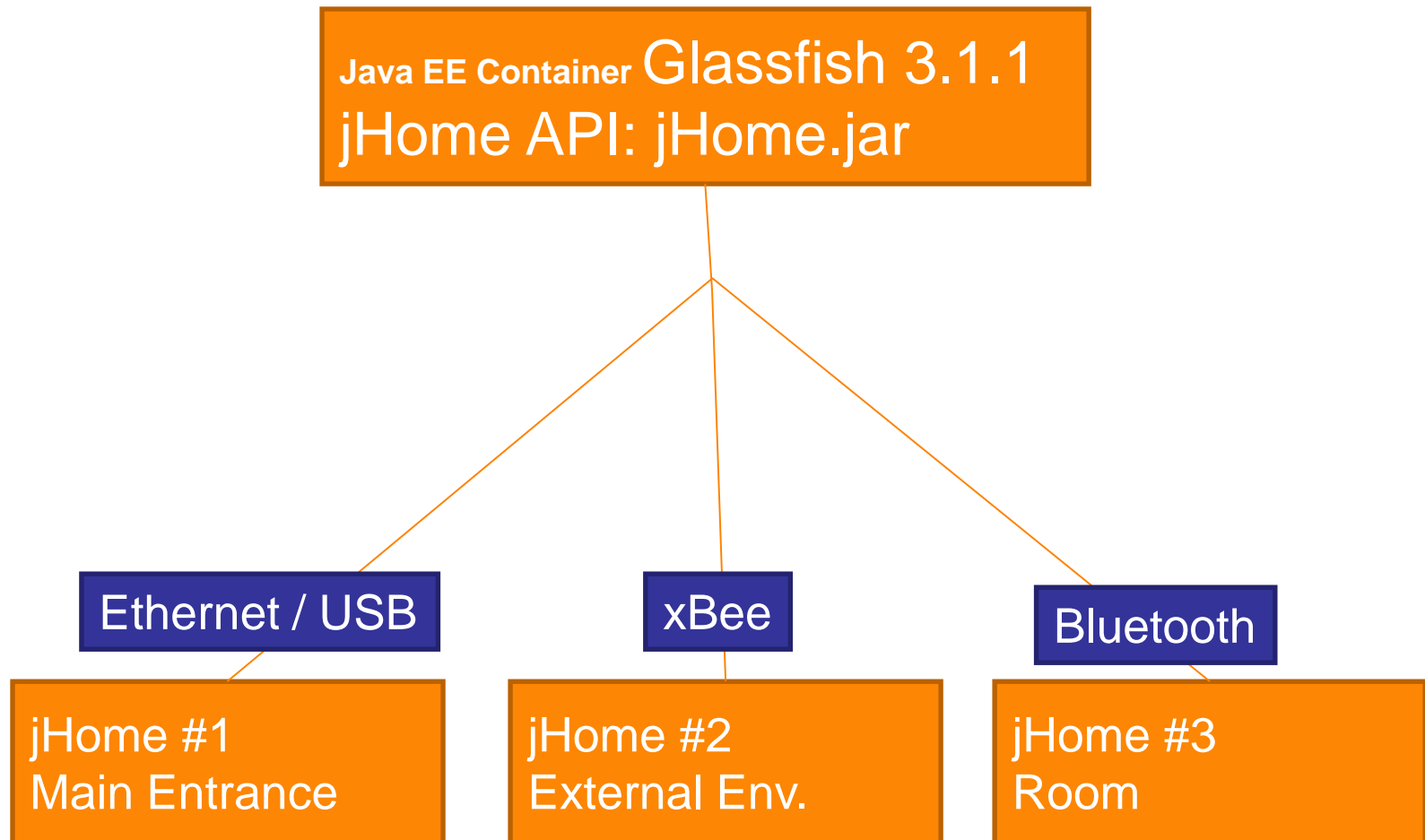
- Java EE 6 - Glassfish 3.1;
- Provides services to jHome gadgets:
 - Discovery
 - Monitoring
 - Scheduling
 - Persistence
 - Web Integration

Components

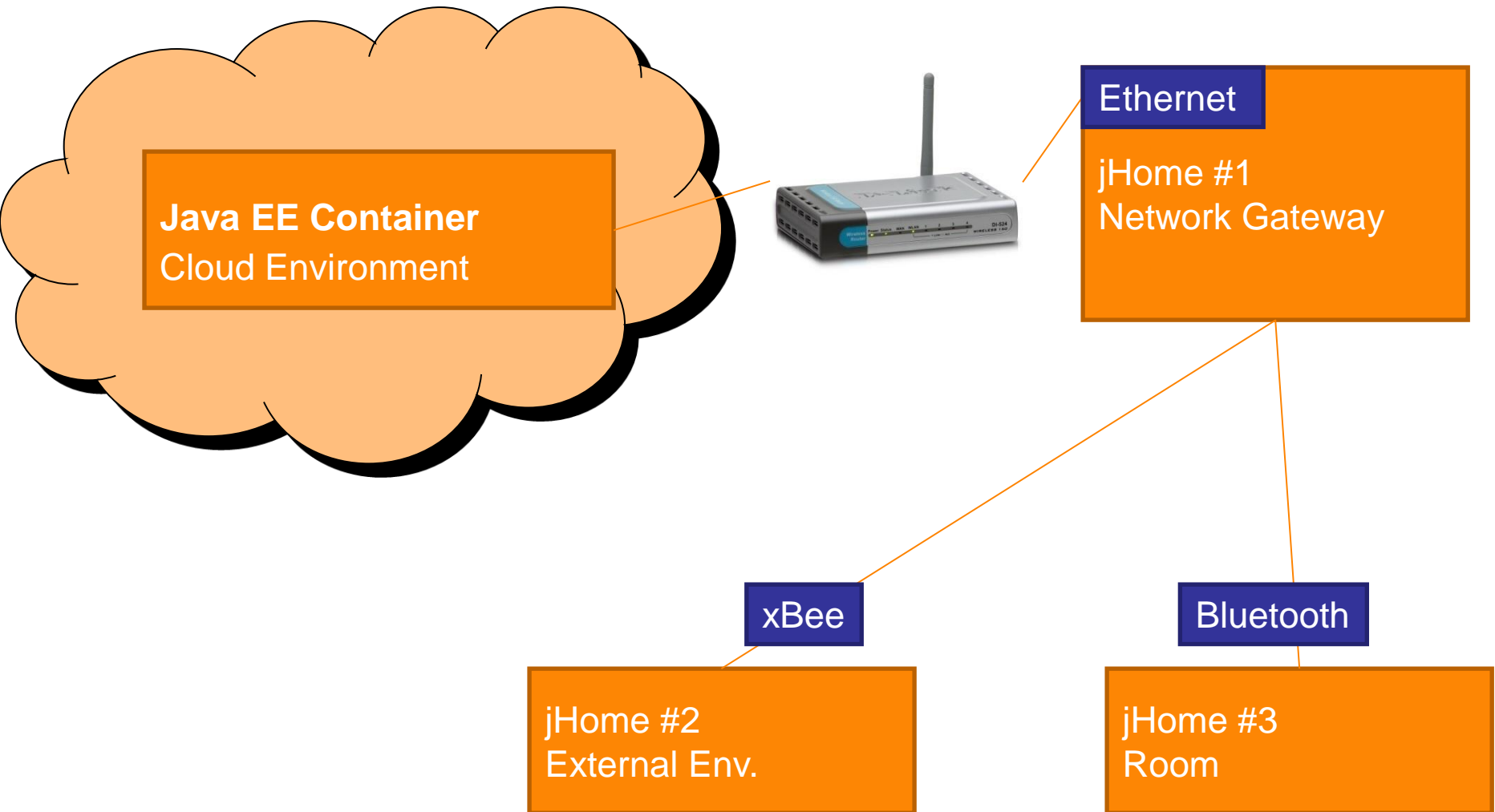


- jHome.jar
 - jHome interfaces
- jHomeCore.jar
 - core components and implementation
- jHomeWeb.war
 - web admin interface and jQuery UI
- jHomeTwitter.jar
 - twitter integration

jHome Server



jHome on cloud



Java EE 6 usage



Java EE Resource	Usage
EJB / Singleton	Device control with concurrency control
CDI	Device injection into components like servlets and other ejb's
Timer Service	Scheduling wall jackets / relay
JAX-RS	Provides sensor info using JSON
JSF	Web Admin Interface
Servlet	High-level web components

Sample code



```
@WebServlet(name = "Light", urlPatterns = {"/Light"})
public class Light extends HttpServlet {
    private boolean on;
    @EJB
    Relay light;
    protected void doGet(...) throws ... {
        if (on) {
            light.turnOff("relay1");
            on = false;
        } else {
            light.turnOn("relay1");
            on = true;
        }
    }
}
```

DEMO 4



- Heart-driven Bean

DEMO 5



- Sound Sensor with Fast Fourier Transforming and Processing

Coming soon...



- External DSL:
 - "schedule relay1 on 10:00 off 12:00"
 - "if temperature \geq 30 turn on fan"
- Multi-behavior devices
- Built-in IR support
- Services on cloud

More info



www.arduino.cc

jhome.globalcode.com.br

www.sparkfun.com

www.parallax.com

www.maker.com

www.dealextreme.com

jHome Source & Support



jHome Source Code

<https://globalcode.toolscloud.net/git/jHome.git/>
user jhome password jhome

jHome Discussion Group

<http://groups.google.com/group/jhome-globalcode>