

# Machine Intelligence based IoT Management System

Presented by:-  
Abin Mathew Abraham  
(150948015)

Guided by:-  
Sucharitha Shetty  
Swarup Kumar Mohalik



**MANIPAL INSTITUTE  
OF TECHNOLOGY**  
**MANIPAL**

*A Constituent Institution of Manipal University*

# Contents

- Introduction
- Objectives
- Background
- Overview of Solution
- Implementation Details
- Use Cases
- Result Analysis
- Summary
- Future Work
- Future Scope
- References

# Introduction

- Future is IoT
- Data -> Information -> Context Awareness -> Decisions
- Scaling
- Device Management
- Autonomous and Adaptive
- Areas:-
  - Network Protocols
  - AI Planning
  - Software Development

# Objectives

- Prototype of an Autonomous and Adaptive IoT Management System
- Salient Features:-
  - Efficient Communication with Constrained Devices
  - Use of a Standard Industry Protocol for Device Management
  - Autonomous and Adaptive Application

# Background

- CoAP
  - Embedded web transfer protocol
  - Asynchronous transaction model
  - UDP binding, reliability, multicast
  - GET, POST, PUT, DELETE methods
  - URI support
  - 4 byte header
  - DTLS
  - Built in discovery
  - Optional Observation and Block Transfer
  - Subset of MIME types and response codes

# Background (contd.)

- LwM2M
  - OMA Alliance
  - Independent of type/category of device
  - Independent of network
  - Object Models (supports IPSO object models and Custom models)
  - Interfaces – Bootstrapping, Registration, Management, Reporting
  - Object Access – Read, Write, Execute (easy translation from HTTP)
- AI Planning
  - To achieve pre-stated objectives
  - Formulate a problem as search problem and search for goals
  - Planners (FF, Metric FF, Optic, PANDA service)
  - PDDL

# Background (contd.)

- Leshan
  - Open Source
  - Eclipse IoT Project
  - Java libs for developing LwM2M Server and Client
  - CoAP implementation - Californium
  - DTLS implementation - Scandium
- openHAB 2.0
  - Open Home Automation Bus
  - Runs on JVM
  - Integration platform for devices/technologies -> one solution
  - Automation Rules

# Overview of Solution

- Part 1 - System
  - Server for handling clients
  - Constrained Environment - CoAP
  - Device Management - LwM2M
  - REST APIs
  - HTTP to CoAP translations
- Part 2 - Application
  - Decoupled from the System
  - Pull states from the server using REST APIs
  - Policy incorporation
  - LTL integration
  - Generate PDDL files from states, policies and LTL formulas
  - Execute plan through REST APIs



# Overview of Solution (contd.)

- Part 3 - Planning
  - Decoupled from Application
  - Planner Application/Service
  - Takes PDDL domain and problem files as inputs
  - Generates a plan if possible
  - BFS, A\*, EHC and their combinations
  - Sends the plan back to the Application

# Overview of Solution (contd.)

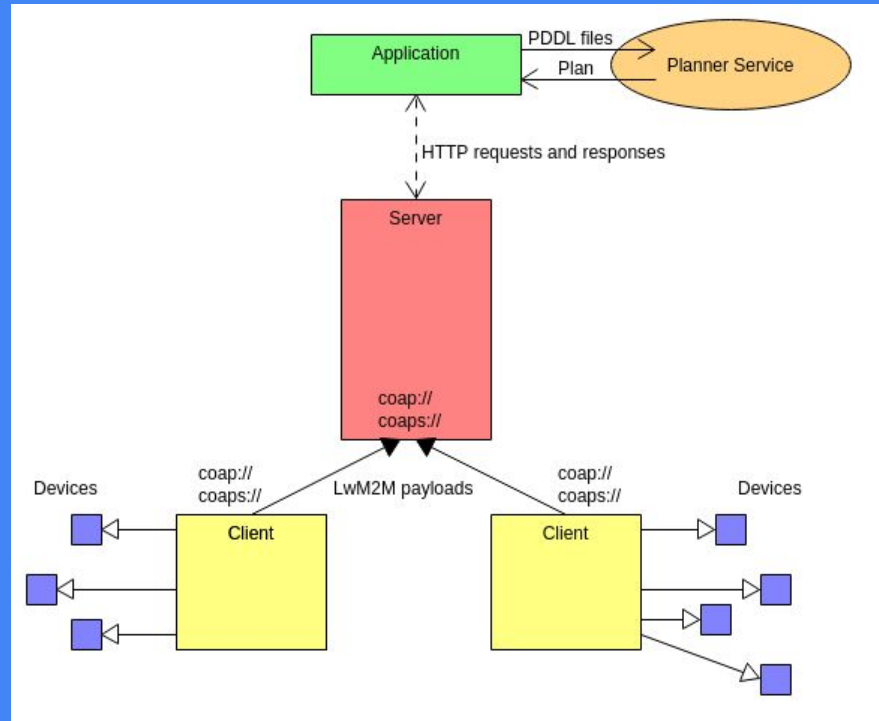


Figure 1 - Proposed System Architecture

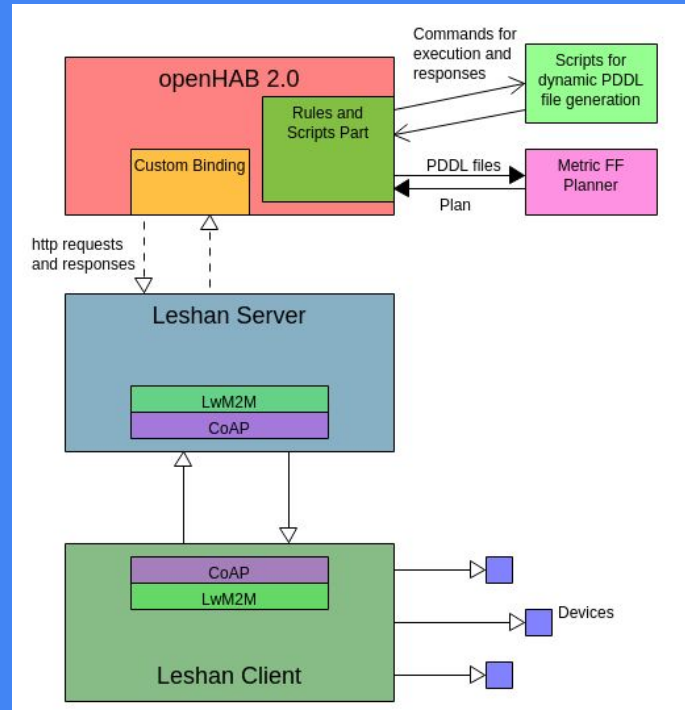
# Implementation Details

- Leshan - Server and Device Clients
  - LwM2M over CoAP
  - Clients are specifically developed for the devices - JAVA
  - Server has GUI
- openHAB 2.0 - Application
  - Custom Binding for Leshan Server
    - Fetches states through Server's REST APIs
    - Clients can be controlled through GUI
  - Rules and Scripts - For automation
    - Generates the PDDL problem file - external Python script
    - Calls the planner service - http call from shell/python script
    - Executes Plan - through REST APIs of Server

# Implementation Details (contd.)

- PANDA - Planner Service
  - Ericsson's internal tool
- Metric FF, FF, Optic - Planner application
  - Open source

# Implementation Details (contd.)



# Use Cases

- Building Use Case
  - Purpose: Create the autonomous and adaptive system
  - Fire Sensor and Sprinkler in every room
  - A critical device like building server will be there in one room
  - Room Client - Leshan Client
    - Fire Sensor - Generic Sensor Object
    - Sprinkler - Actuation Object
  - Building Server Client - Leshan Client
    - Power up/Shutdown - Actuation Object
    - Location - Set Point Object
  - Clients <--> Server communication - LwM2M over CoAP
  - openHAB 2.0 frequently pulls the device states

# Use Cases (contd.)

- Building Use Case (contd.)
  - Expected System Behaviour
    - Detection, Planning, Execution
    - If fire and no building server in the room - turn on sprinkler
    - If fire and building server is there in the room - turn off the server first and then turn on sprinkler

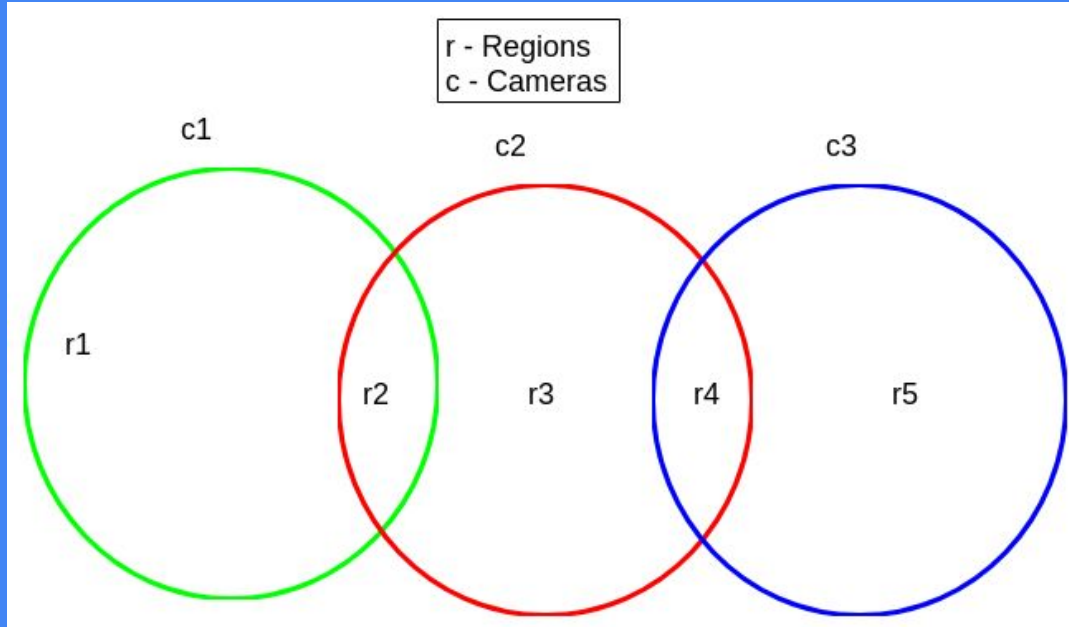
# Use Cases (contd.)

- Camera use cases
  - Purpose: Policy incorporation, LTL formula incorporation
  - Cameras and Regions covered
  - Critical Regions
  - Critical Regions must not be unmonitored
  - Cameras must be upgraded
  - Upgradation can only happen when camera is switched off
  - Switching off cameras can result in unmonitored critical regions
  - Goal: Sequence of steps which doesn't allow unmonitored critical regions



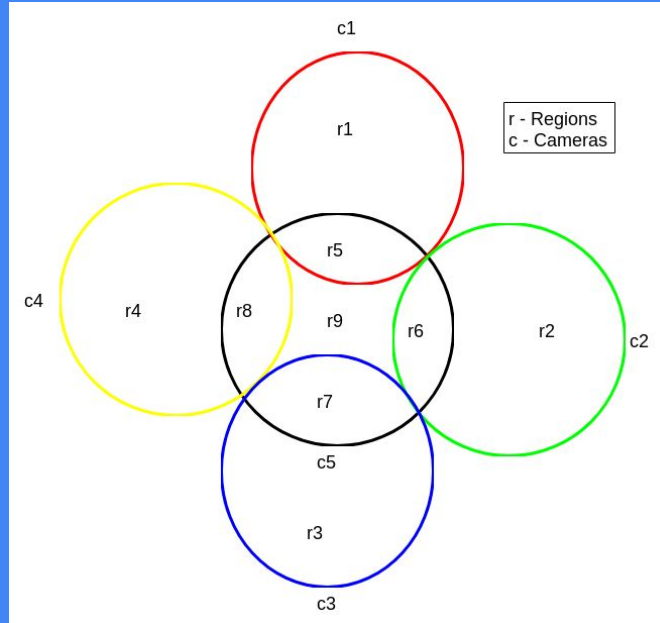
# Use Cases (contd.)

- Camera use cases (contd.)
  - 3 Camera use case
    - Approaching the problem in simple way



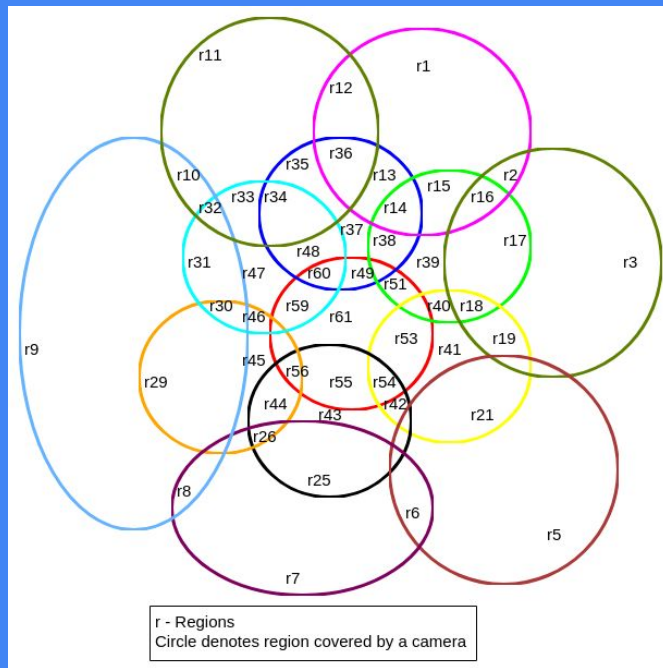
# Use Cases (contd.)

- Camera use cases (contd.)
  - 5 Camera use case
    - Adopting the simple approach to complex case



# Use Cases (contd.)

- Camera use cases (contd.)
  - 13 Camera use case
    - Scaling (61 regions, 24 critical regions), LTL



# Use Cases (contd.)

- Camera use cases (contd.)
  - Every policy change shouldn't be a remodelling task
  - Policy from Policy file → Domain PDDL file
  - Dummy action
    - Force the action after every real action
  - LTL - always, eventually, next time, until and release
    - LTL formula → Büchi Automaton [3]
      - Online software was used

# Result Analysis

- Building use case
  - Leshan Server start

LeshanServerProject [Java Application] /usr/lib/jvm/oracle\_jdk8/bin/java (25-May-2017, 12:34:51 PM)

```
May 25, 2017 12:34:51 PM org.eclipse.californium.core.network.config.NetworkConfig load
INFO: loading properties from file /home/abin/leshan_project_workspace/leshan-server-project/Californium.properties
May 25, 2017 12:34:51 PM org.eclipse.californium.core.CoapServer start
INFO: Starting server
May 25, 2017 12:34:51 PM org.eclipse.californium.core.network.CoapEndpoint start
INFO: Starting endpoint at coap://0.0.0.0:5683
May 25, 2017 12:34:51 PM org.eclipse.californium.core.network.CoapEndpoint start
INFO: Started endpoint at coap://0.0.0.0:5683
May 25, 2017 12:34:51 PM org.eclipse.californium.core.network.CoapEndpoint start
INFO: Starting endpoint at coaps://0.0.0.0:5684
May 25, 2017 12:34:51 PM org.eclipse.californium.scandium.DTLSConnector start
INFO: DTLS connector listening on [0.0.0.0/0.0.0.0:5684] with MTU [1,280] using (inbound) datagram buffer size [16,474 bytes]
May 25, 2017 12:34:51 PM org.eclipse.californium.core.network.CoapEndpoint start
INFO: Started endpoint at coaps://0.0.0.0:5684
2017-05-25 12:34:51,554 INFO LeshanServer - LWM2M server started at coap://0.0.0.0/0.0.0.0:5683, coaps://0.0.0.0/0.0.0.0:5684.
2017-05-25 12:34:51,611 INFO LeshanServerDemo - Web server started at http://127.0.1.1:45456/.
```

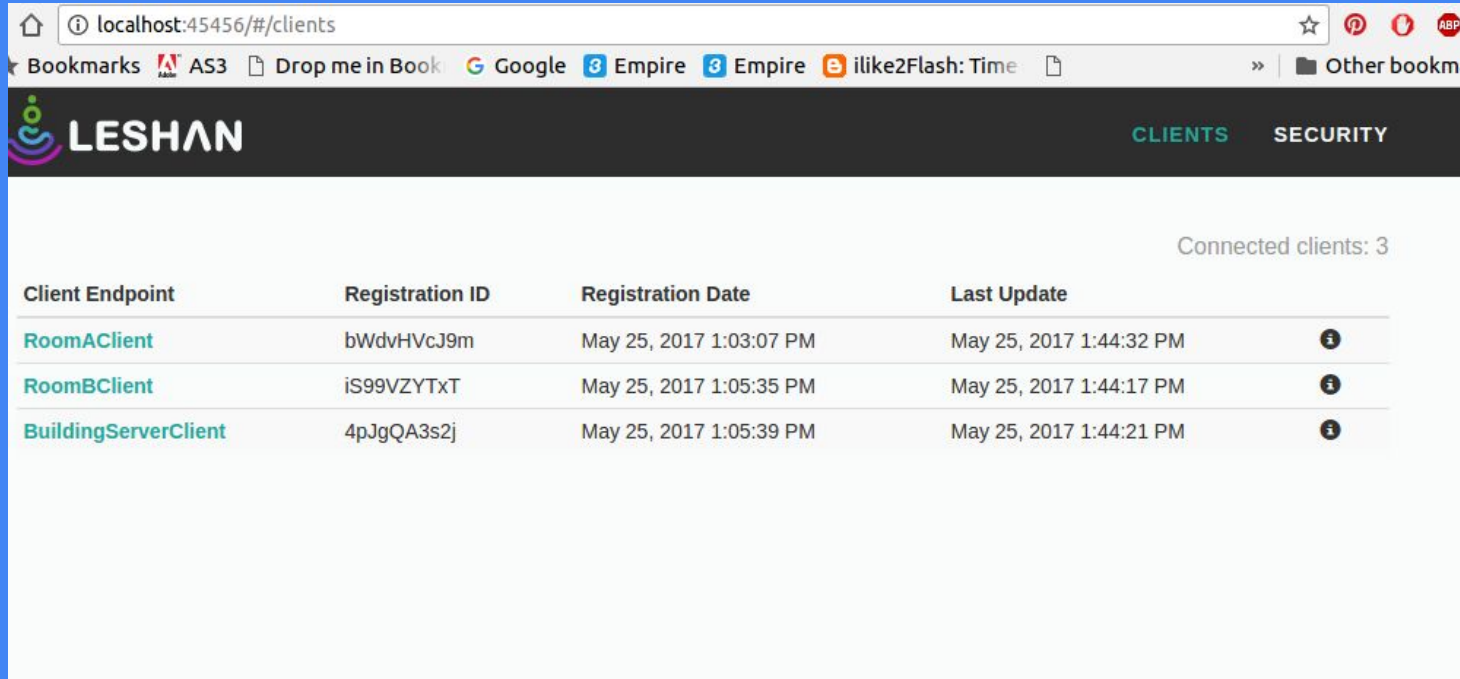
# Result Analysis (contd.)

- Building use case (contd.)
  - Leshan Clients start

```
BuildingServerLeshanClient [Java Application] /usr/lib/jvm/oracle_jdk8/bin/java (25-May-2017, 1:05:38 PM)
May 25, 2017 1:05:39 PM org.eclipse.californium.core.network.config.NetworkConfig load
INFO: loading properties from file /home/abin/leshan_project_workspace/building_server_leshan-client/Californium.properties
2017-05-25 13:05:39,112 INFO LeshanClient - Starting Leshan client ...
May 25, 2017 1:05:39 PM org.eclipse.californium.core.CoapServer start
INFO: Starting server
May 25, 2017 1:05:39 PM org.eclipse.californium.core.network.CoapEndpoint start
INFO: Starting endpoint at coaps://0.0.0.0:0
May 25, 2017 1:05:39 PM org.eclipse.californium.scandium.DTLSConnector start
INFO: DTLS connector listening on [0.0.0.0/0.0.0.0:49472] with MTU [1,280] using (inbound) datagram buffer size [16,474 bytes]
May 25, 2017 1:05:39 PM org.eclipse.californium.core.network.CoapEndpoint start
INFO: Started endpoint at coaps://0.0.0.0:49472
May 25, 2017 1:05:39 PM org.eclipse.californium.core.network.CoapEndpoint start
INFO: Starting endpoint at coap://0.0.0.0:0
May 25, 2017 1:05:39 PM org.eclipse.californium.core.network.CoapEndpoint start
INFO: Started endpoint at coap://0.0.0.0:52840
2017-05-25 13:05:39,126 INFO LeshanClient - Leshan client started [endpoint:BuildingServerClient].
2017-05-25 13:05:39,129 INFO RegistrationEngine - Trying to register to coap://localhost:5683 ...
2017-05-25 13:05:39,142 INFO RegistrationEngine - Next registration update in 27.0s...
2017-05-25 13:05:39,143 INFO RegistrationEngine - Registered with location '/rd/4pJgQA3s2j'.
```

# Result Analysis (contd.)

- Building use case (contd.)
  - Leshan Server GUI after client registrations



The screenshot shows a web browser window with the address bar displaying 'localhost:45456/#/clients'. The browser's bookmark bar includes 'AS3', 'Drop me in Book', 'Google', 'Empire', and 'ilike2Flash: Time'. The Leshan Server GUI has a dark header with the 'LESHAN' logo and navigation tabs for 'CLIENTS' and 'SECURITY'. Below the header, it indicates 'Connected clients: 3'. A table lists the following clients:

Client Endpoint	Registration ID	Registration Date	Last Update
RoomAClient	bWdvHVcJ9m	May 25, 2017 1:03:07 PM	May 25, 2017 1:44:32 PM
RoomBClient	iS99VZYTtT	May 25, 2017 1:05:35 PM	May 25, 2017 1:44:17 PM
BuildingServerClient	4pJgQA3s2j	May 25, 2017 1:05:39 PM	May 25, 2017 1:44:21 PM

# Result Analysis (contd.)

- Building use case (contd.)
  - Client Controls in GUI

The screenshot shows a web browser window with the address bar displaying 'localhost:45456/#/clients/BuildingServerClient'. The browser's bookmark bar includes 'AS3', 'Drop me in Book', 'Google', 'Empire', and 'iLike2Flash: Time'. The main content area is divided into three sections: a top table of system parameters, an 'Actuation' section for a server power actuator, and a 'Set Point' section for a server location.

Parameter	Value	Control
Lifetime	/1/0/1	Observe ▶ Read Write
Default Minimum Period	/1/0/2	Observe ▶ Read Write
Default Maximum Period	/1/0/3	Observe ▶ Read Write
Disable	/1/0/4	Exec ⚙
Disable Timeout	/1/0/5	Observe ▶ Read Write
Notification Storing When Disabled or Offline	/1/0/6	Observe ▶ Read Write
Binding	/1/0/7	Observe ▶ Read Write
Registration Update Trigger	/1/0/8	Exec ⚙

**Actuation** /3306

Create New Instance

Instance	Value	Control
Instance 0	/3306/0	Observe ▶ Read Write Delete
Application Type	/3306/0/5750	Observe ▶ Read Write
On/Off	/3306/0/5850	Observe ▶ Read Write
Dimmer	/3306/0/5851	Observe ▶ Read Write
On Time	/3306/0/5852	Observe ▶ Read Write
Multi-state Output	/3306/0/5853	Observe ▶ Read Write

**Server Power Actuator**  
true

**Set Point** /3308

Create New Instance

Instance	Value	Control
Instance 0	/3308/0	Observe ▶ Read Write Delete
Sensor Units	/3308/0/5701	Observe ▶ Read
Colour	/3308/0/5706	Observe ▶ Read Write
Application Type	/3308/0/5750	Observe ▶ Read Write
Set Point Value	/3308/0/5900	Observe ▶ Read Write

**Server Location**  
1



# Result Analysis (contd.)

- Building use case (contd.)
  - GET request on Application Type resource

## ▼ Response Headers [view parsed](#)

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 86
Server: Jetty(9.1.4.v20140401)
```

## ▼ Request Headers [view parsed](#)

```
GET /api/clients/RoomAClient/3306/0/5750?format=JSON HTTP/1.1
Host: localhost:45456
Connection: keep-alive
Accept: application/json, text/plain, */*
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/58.0.3029.110 Safari/537.36
Referer: http://localhost:45456/
Accept-Encoding: gzip, deflate, sdch, br
Accept-Language: en-US,en;q=0.8
```

## ▼ Query String Parameters [view parsed](#)

```
format=JSON
```

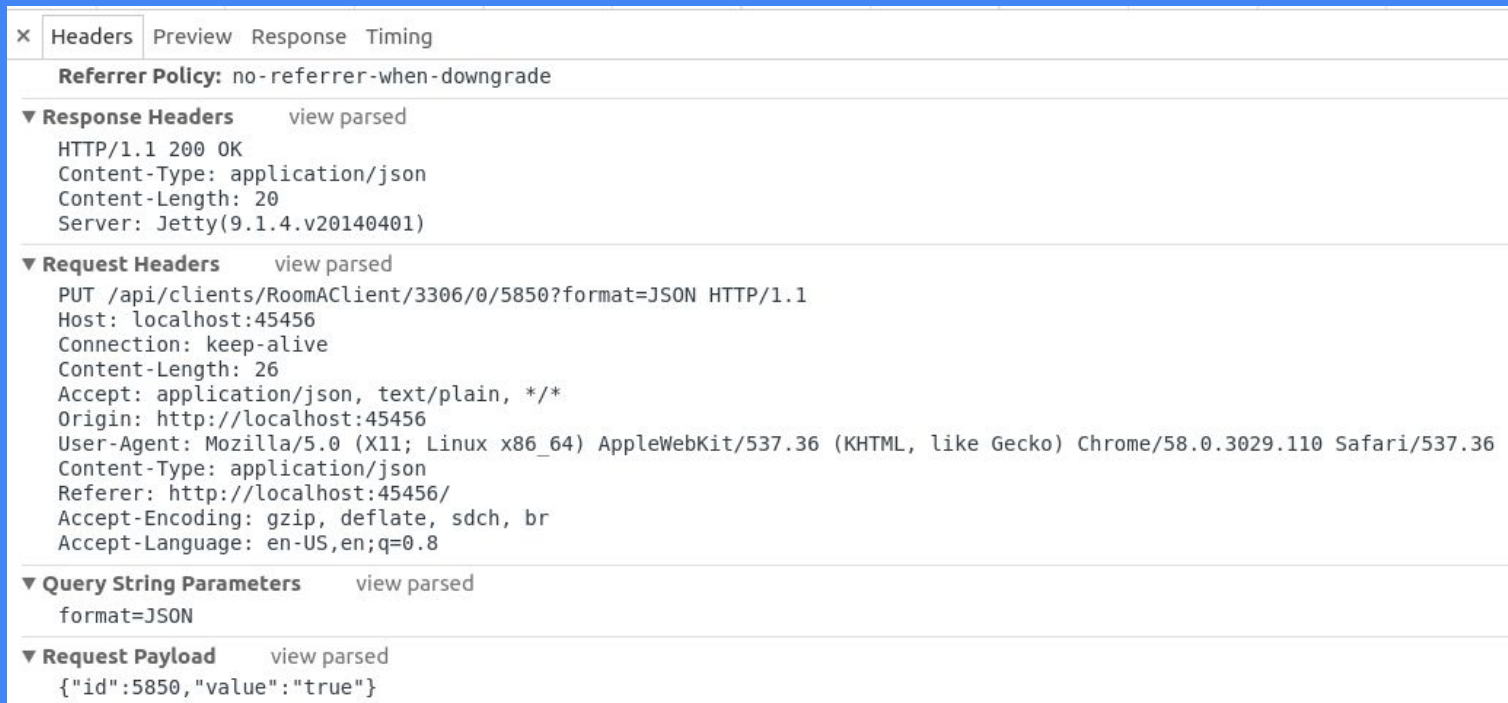
# Result Analysis (contd.)

- Building use case (contd.)
  - GET request's response

X Headers		Preview	Response	Timing
1	<pre>{<u>"status"</u>:<u>"CONTENT"</u>,<u>"content"</u>:{"id":5750,"value":"Room A Sprinkler control Actuator"}}<u>}</u></pre>			

# Result Analysis (contd.)

- Building use case (contd.)
  - PUT request to On/Off resource



The screenshot displays the 'Headers' tab of a web browser's developer tools. It shows the details of an HTTP PUT request and its response. The 'Response' section indicates a successful status (200 OK) with a JSON content type. The 'Request' section shows the full request, including headers like 'Host', 'User-Agent', and 'Referer', and a query string 'format=JSON'. The 'Request Payload' section shows the JSON body: `{\"id\":5850,\"value\":\"true\"}`.

Section	Details
Referrer Policy	no-referrer-when-downgrade
Response Headers	<p>HTTP/1.1 200 OK</p> <p>Content-Type: application/json</p> <p>Content-Length: 20</p> <p>Server: Jetty(9.1.4.v20140401)</p>
Request Headers	<p>PUT /api/clients/RoomAClient/3306/0/5850?format=JSON HTTP/1.1</p> <p>Host: localhost:45456</p> <p>Connection: keep-alive</p> <p>Content-Length: 26</p> <p>Accept: application/json, text/plain, */*</p> <p>Origin: http://localhost:45456</p> <p>User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/58.0.3029.110 Safari/537.36</p> <p>Content-Type: application/json</p> <p>Referer: http://localhost:45456/</p> <p>Accept-Encoding: gzip, deflate, sdch, br</p> <p>Accept-Language: en-US,en;q=0.8</p>
Query String Parameters	<p>format=JSON</p>
Request Payload	<pre>{\"id\":5850,\"value\":\"true\"}</pre>

# Result Analysis (contd.)

- Building use case (contd.)
  - PUT request's response on success

X	Headers	Preview	Response	Timing
1			<code>{"status": "CHANGED"}</code>	

# Result Analysis (contd.)

- Building use case (contd.)
  - REST API call for registry details

```
localhost:45456/api/clients?format=JSON
App3 Apps  ★ Bookmarks  AS3  Drop me in Book  Google  3 Empire  3 Empire  Other bookmarks

[{"endpoint": "RoomAClient", "registrationId": "ZX5dW0acYG", "registrationDate": "2017-05-28T02:02:09+05:30", "lastUpdate": "2017-05-28T02:02:09+05:30", "address": "127.0.0.1:38139", "lwM2mVersion": "1.0", "lifetime": 30, "bindingMode": "U", "rootPath": "/", "objectLinks": [{"url": "/", "attributes": {"rt": "oma.lwm2m"}}, {"url": "/1/0", "attributes": {}}, {"url": "/3300/0", "attributes": {}}, {"url": "/3306/0", "attributes": {}}], "secure": false, "additionalRegistrationAttributes": {}}, {"endpoint": "RoomBClient", "registrationId": "mrKwWslTxX", "registrationDate": "2017-05-28T02:02:12+05:30", "lastUpdate": "2017-05-28T02:02:12+05:30", "address": "127.0.0.1:44398", "lwM2mVersion": "1.0", "lifetime": 30, "bindingMode": "U", "rootPath": "/", "objectLinks": [{"url": "/", "attributes": {"rt": "oma.lwm2m"}}, {"url": "/1/0", "attributes": {}}, {"url": "/3300/0", "attributes": {}}, {"url": "/3306/0", "attributes": {}}], "secure": false, "additionalRegistrationAttributes": {}}, {"endpoint": "BuildingServerClient", "registrationId": "R1TB0K9YAv", "registrationDate": "2017-05-28T02:02:04+05:30", "lastUpdate": "2017-05-28T02:02:04+05:30", "address": "127.0.0.1:52218", "lwM2mVersion": "1.0", "lifetime": 30, "bindingMode": "U", "rootPath": "/", "objectLinks": [{"url": "/", "attributes": {"rt": "oma.lwm2m"}}, {"url": "/1/0", "attributes": {}}, {"url": "/3306/0", "attributes": {}}, {"url": "/3308/0", "attributes": {}}], "secure": false, "additionalRegistrationAttributes": {}}]
```

# Result Analysis (contd.)

- Building use case (contd.)
  - Parsed response for previously mentioned call

```
[
  {
    "endpoint": "RoomAClient",
    "registrationId": "AVbdiXTNfP",
    "registrationDate": "2017-05-26T13:25:36+05:30",
    "lastUpdate": "2017-05-26T13:27:51+05:30",
    "address": "127.0.0.1:53840",
    "lwM2mVersion": "1.0",
    "lifetime": 30,
    "bindingMode": "U",
    "rootPath": "/",
    "objectLinks": [4],
    "secure": false,
    "additionalRegistrationAttributes": {}
  },
  {},
  {}
]
```

# Result Analysis (contd.)

- Building use case (contd.)
  - Expanding object links in previous response

```
"objectLinks": [
  {
    "url": "/1/0",
    "attributes": {
    }
  },
  {
    "url": "/3300/0",
    "attributes": {
    }
  },
  {
    "url": "/3306/0",
    "attributes": {
    }
  }
],
```

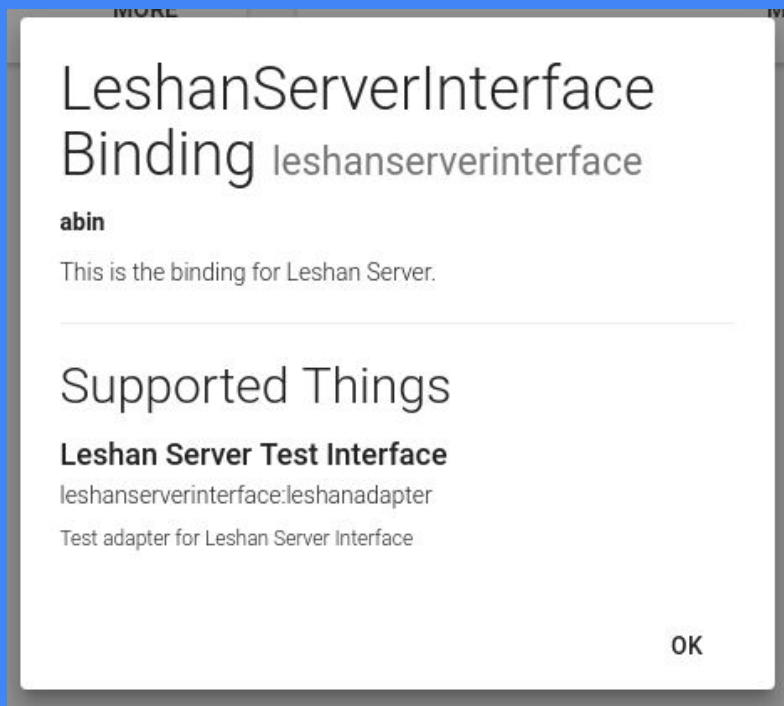
# Result Analysis (contd.)

- Building use case (contd.)
  - Custom binding to communicate with Leshan Server
    - Hash table of client handlers (key - registration id)
    - Hash table of object handlers (key - channel id)
    - Commands are handled in a thread
    - From the thread corresponding API calls are made



# Result Analysis (contd.)

- Building use case (contd.)
  - Custom binding's description



# Result Analysis (contd.)

- Building use case (contd.)
  - Thing creation

### Configure Leshan Server Test Interface

Name	Leshan Server Test Interface		
Thing ID	ProjectLeshanServer		
Location	Bangalore		








































#### Configuration Parameters

Configure parameters for the thing.

IP Address	127.0.0.1	Port Number	45456
The IP address of the Toy Server.		Port to which the connection is to be made.	









# Result Analysis (contd.)

- Building use case (contd.)
  - GUI after Refresh button click

 Refresh 	 Client Count <b>3 Clients</b>
 RoomAClient:IPSOGenericSensor:0:Read Application Type 	 RoomAClient:IPSOGenericSensor:0:Application Type
 RoomAClient:IPSOGenericSensor:0:Read Value 	 RoomAClient:IPSOGenericSensor:0:Value
 RoomAClient:IPSOActuation:0:Read Application Type 	 RoomAClient:IPSOActuation:0:Application Type
 RoomAClient:IPSOActuation:0:Read On/Off State 	 RoomAClient:IPSOActuation:0:On/Off State
 RoomBClient:IPSOGenericSensor:0:Read Application Type 	 RoomBClient:IPSOGenericSensor:0:Application Type
 RoomBClient:IPSOGenericSensor:0:Read Value 	 RoomBClient:IPSOGenericSensor:0:Value
 RoomBClient:IPSOActuation:0:Read Application Type 	 RoomBClient:IPSOActuation:0:Application Type
 RoomBClient:IPSOActuation:0:Read On/Off State 	 RoomBClient:IPSOActuation:0:On/Off State
 BuildingServerClient:IPSOActuation:0:Read Application Type 	 BuildingServerClient:IPSOActuation:0:Application Type
 BuildingServerClient:IPSOActuation:0:Read On/Off State 	 BuildingServerClient:IPSOActuation:0:On/Off State
 BuildingServerClient:IPSOSetPoint:0:Read Application Type 	 BuildingServerClient:IPSOSetPoint:0:Application Type
 BuildingServerClient:IPSOSetPoint:0:Read Value 	 BuildingServerClient:IPSOSetPoint:0:Value

# Result Analysis (contd.)

- Building use case (contd.)
  - Response for button clicks

	RoomAClient:IPSOGenericSensor:0:Read Application Type	<input type="checkbox"/>		RoomAClient:IPSOGenericSensor:0:Applicatio... <b>Room A Fire Sen...</b>
	RoomAClient:IPSOGenericSensor:0:Read Value	<input type="checkbox"/>		RoomAClient:IPSOGenericSensor:0:Value <b>OFF</b>
	RoomAClient:IPSOActuation:0:Read Application Type	<input type="checkbox"/>		RoomAClient:IPSOActuation:0:Applic... <b>Room A Sprinkler control ...</b>
	RoomAClient:IPSOActuation:0:Read On/Off State	<input type="checkbox"/>		RoomAClient:IPSOActuation:0:On/Off State <b>OFF</b>

# Result Analysis (contd.)

- Building use case (contd.)
  - Cron job

```
rule "frequent state fetch"
when
    Time cron "0/10 * * * * ?"
then
    sendCommand("leshanserverinterface_leshanadapter_ProjectLeshanServer_BuildingServerClient_IPSOSetPoint_0_readValue","ON");
    sendCommand("leshanserverinterface_leshanadapter_ProjectLeshanServer_BuildingServerClient_IPSOActuation_0_readOnOffState","ON");

    sendCommand("leshanserverinterface_leshanadapter_ProjectLeshanServer_RoomAClient_IPSOActuation_0_readOnOffState","ON");
    sendCommand("leshanserverinterface_leshanadapter_ProjectLeshanServer_RoomAClient_IPSOGenericSensor_0_readValue","ON");

    sendCommand("leshanserverinterface_leshanadapter_ProjectLeshanServer_RoomBClient_IPSOActuation_0_readOnOffState","ON");
    sendCommand("leshanserverinterface_leshanadapter_ProjectLeshanServer_RoomBClient_IPSOGenericSensor_0_readValue","ON");
end
```

# Result Analysis (contd.)

- Building use case (contd.)
  - Rules

```
rule "Fire in Room A detected"
when
    Item leshanserverinterface_leshanadapter_ProjectLeshanServer_RoomAClient_IPSOGenericSensor_0_readValueResponse changed from OFF to ON or
    Item leshanserverinterface_leshanadapter_ProjectLeshanServer_RoomAClient_IPSOGenericSensor_0_readValueResponse changed from NULL to ON
then
    callScript("switchonsprinklerA")
end

rule "Fire in Room B detected"
when
    Item leshanserverinterface_leshanadapter_ProjectLeshanServer_RoomBClient_IPSOGenericSensor_0_readValueResponse changed from OFF to ON or
    Item leshanserverinterface_leshanadapter_ProjectLeshanServer_RoomBClient_IPSOGenericSensor_0_readValueResponse changed from NULL to ON
then
    callScript("switchonsprinklerB")
end
```

# Result Analysis (contd.)

- Building use case (contd.)
  - Shell script execution for PDDL generation and planning

```
var String args = "roomA:roomB roomB "  
if(leshanserverinterface leshanadapter ProjectLeshanServer BuildingServerClient IPS0SetPoint 0 readValueResponse.state.toString.equals("RoomA")){  
    args += "roomA"  
}  
else if(leshanserverinterface leshanadapter ProjectLeshanServer BuildingServerClient IPS0SetPoint 0 readValueResponse.state.toString.equals("RoomB")){  
    args += "roomB"  
}  
var String response = executeCommandLine("./getplan.sh "+args,5000)
```

# Result Analysis (contd.)

- Building use case (contd.)
  - Plan generated when building server is in Room B and fire is in Room A

```
ff: found legal plan as follows  
step    0: SWITCH_ON_SPRINKLER ROOMA
```

- Plan generated when both building server and fire is in Room B

```
ff: found legal plan as follows  
step    0: SWITCH_OFF_SERVER ROOMB  
        1: SWITCH_ON_SPRINKLER ROOMB
```



# Result Analysis (contd.)

- Building use case (contd.)
  - Plan generated when building server is in Room A and fire is in Room B

```
ff: found legal plan as follows  
step    0: SWITCH_ON_SPRINKLER ROOMB
```

- Plan generated when both building server and fire is in Room A

```
ff: found legal plan as follows  
step    0: SWITCH_OFF_SERVER ROOMA  
        1: SWITCH_ON_SPRINKLER ROOMA
```








































# Result Analysis (contd.)

- Building use case (contd.)
  - Parsing and Plan Execution

```
1 - 0
while(i < len){
    cmd = commands.get(i)
    param = params.get(i)
    if(cmd.equals("switch_off_server")){
        item = "leshanserverinterface_leshanadapter_ProjectLeshanServer_BuildingServerClient_IPSOActuation_0_readOnOffStateResponse"
        sendCommand(item,"OFF")
    }else if(cmd.equals("switch_on_sprinkler") && param.equals("roomA")){
        item = "leshanserverinterface_leshanadapter_ProjectLeshanServer_RoomAClient_IPSOActuation_0_readOnOffStateResponse"
        sendCommand(item,"ON")
    }else if(cmd.equals("switch_on_sprinkler") && param.equals("roomB")){
        item = "leshanserverinterface_leshanadapter_ProjectLeshanServer_RoomBClient_IPSOActuation_0_readOnOffStateResponse"
        sendCommand(item,"ON")
    }
    i++
}
```

# Result Analysis (contd.)

- Building use case (contd.)
  - States after fire sensor is triggered in room A

 Refresh 	 Client Count 3 Clients
 RoomAClient:IPSOGenericSensor:0:Read Application Type 	 RoomAClient:IPSOGenericSensor:0:Applicatio... Room A Fire Sen...
 RoomAClient:IPSOGenericSensor:0:Read Value 	 RoomAClient:IPSOGenericSensor:0:Value ON
 RoomAClient:IPSOActuation:0:Read Application Type 	 RoomAClient:IPSOActuation:0:Applic... Room A Sprinkler control ...
 RoomAClient:IPSOActuation:0:Read On/Off State 	 RoomAClient:IPSOActuation:0:On/Off State CHANGED
 RoomBClient:IPSOGenericSensor:0:Read Application Type 	 RoomBClient:IPSOGenericSensor:0:Applicatio... Room B Fire Sen...
 RoomBClient:IPSOGenericSensor:0:Read Value 	 RoomBClient:IPSOGenericSensor:0:Value OFF
 RoomBClient:IPSOActuation:0:Read Application Type 	 RoomBClient:IPSOActuation:0:Applic... Room B Sprinkler control ...
 RoomBClient:IPSOActuation:0:Read On/Off State 	 RoomBClient:IPSOActuation:0:On/Off State OFF
 BuildingServerClient:IPSOActuation:0:Read Application Type 	 BuildingServerClient:IPSOActuation:0:Applica... Server Power Act...
 BuildingServerClient:IPSOActuation:0:Read On/Off State 	 BuildingServerClient:IPSOActuation:0:On/Off State CHANGED
 BuildingServerClient:IPSOSetPoint:0:Read Application Type 	 BuildingServerClient:IPSOSetPoint:0:Application T... Server Locati...
 BuildingServerClient:IPSOSetPoint:0:Read Value 	 BuildingServerClient:IPSOSetPoint:0:Value RoomA

# Result Analysis (contd.)

- Building use case (contd.)
  - openHAB logs after execution

```
13:15:50.168 [INFO ] [smarthome.event.ItemCommandEvent    ] - Item 'leshanserver  
interface_leshanadapter_ProjectLeshanServer_BuildingServerClient_IPSOActuation_0  
_readOnOffStateResponse' received command OFF  
13:15:50.168 [INFO ] [marthome.event.ItemStateChangedEvent] - leshanserverinterf  
ace_leshanadapter_ProjectLeshanServer_BuildingServerClient_IPSOActuation_0_readO  
nOffStateResponse changed from ON to OFF  
13:15:50.169 [INFO ] [smarthome.event.ItemCommandEvent    ] - Item 'leshanserver  
interface_leshanadapter_ProjectLeshanServer_RoomAClient_IPSOActuation_0_readOnOf  
fStateResponse' received command ON  
13:15:50.172 [INFO ] [marthome.event.ItemStateChangedEvent] - leshanserverinterf  
ace_leshanadapter_ProjectLeshanServer_RoomAClient_IPSOActuation_0_readOnOffState  
Response changed from OFF to ON
```

- Autonomous and Adaptive

# Result Analysis (contd.)

- Camera use case
  - 'Safety check' action
  - 'Check' predicate
  - 'Safe' predicate

```
(:action SAFETY-CHECK
:parameters ()
:precondition (check)
:effect
  (and (not(check))
    (when
      (forall (?r - region) (imply
        (critical ?r)
        (exists (?c) (iscovering ?c ?r))))
      (safe)
    )
    (when
      (exists (?r - region)
        (and (critical ?r)
          (forall (?c - cam) (not (iscovering ?c ?r)))))
      (not (safe))
    )
  )
)
```

# Result Analysis (contd.)

- Camera use case (contd.)
  - Plan generated for 3 camera use case
  - SAFETY-CHECKs will be removed

```
ff: found legal plan as follows
step    0: SAFETY-CHECK
         1: UPGRADE C2
         2: SAFETY-CHECK
         3: TURN-ON C2
         4: SAFETY-CHECK
         5: TURN-OFF C1
         6: SAFETY-CHECK
         7: UPGRADE C1
         8: SAFETY-CHECK
         9: TURN-ON C1
        10: SAFETY-CHECK
        11: TURN-OFF C3
        12: SAFETY-CHECK
        13: UPGRADE C3
        14: SAFETY-CHECK
        15: TURN-ON C3
        16: SAFETY-CHECK
```

# Result Analysis (contd.)

- Camera use case (contd.)
  - Plan generated for 5 camera use case

```
ff: found legal plan as follows
step  0: SAFETY-CHECK
      1: TURN-OFF C1
      2: SAFETY-CHECK
      3: UPGRADE C1
      4: SAFETY-CHECK
      5: TURN-ON C1
      6: SAFETY-CHECK
      7: TURN-OFF C2
      8: SAFETY-CHECK
      9: UPGRADE C2
     10: SAFETY-CHECK
     11: TURN-ON C2
     12: SAFETY-CHECK
     13: TURN-OFF C3
     14: SAFETY-CHECK
     15: UPGRADE C3
     16: SAFETY-CHECK
     17: TURN-ON C3
     18: SAFETY-CHECK
     19: TURN-OFF C4
     20: SAFETY-CHECK
     21: UPGRADE C4
     22: SAFETY-CHECK
     23: TURN-ON C4
     24: SAFETY-CHECK
     25: TURN-OFF C5
     26: SAFETY-CHECK
     27: UPGRADE C5
     28: SAFETY-CHECK
     29: TURN-ON C5
     30: SAFETY-CHECK
```

# Result Analysis (contd.)

- Camera use case (contd.)
  - Optimized 'Safety Check' action

```
(:action SAFETY-CHECK
:parameters ()
:precondition (check)
:effect
  (and (not(check))
    (when
      (exists (?r - region)
        (and (critical ?r)
          (forall (?c - cam) (not (iscovering ?c ?r))))
      (not (safe))
    )
  )
)
```



# Result Analysis (contd.)

- Camera use case (contd.)
  - Sample Policy File

```
"policy_condition": "(forall (?r - region) (imply (critical ?r) (exists (?c) (iscovering ?c ?r))))",  
"policy_effect": "(safe)"
```

- 'Safety check' action in meta domain file

```
(:action SAFETY-CHECK  
  :parameters ()  
  :precondition (check)  
  :effect  
    (and (not(check))  
          (when <policyTag> <policyEffectTag>  
            (when <NegPolicyTag> <NegPolicyEffectTag>  
              )  
          )  
    )
```

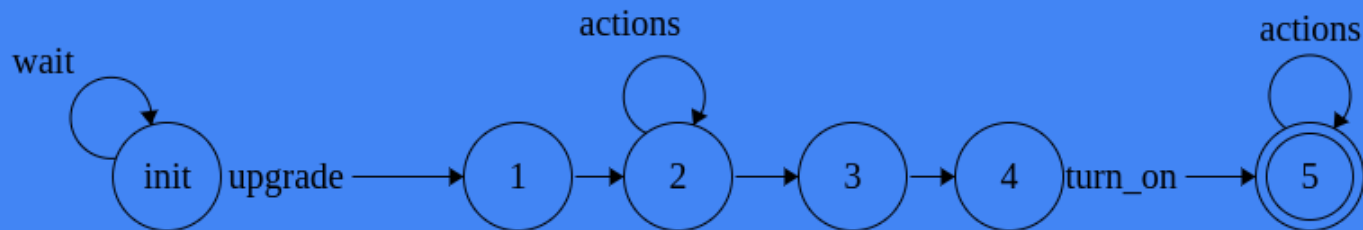
# Result Analysis (contd.)

- Camera use case (contd.)
  - 'Safety Check' action in generated domain file

```
(:action SAFETY-CHECK
  :parameters ()
  :precondition (check)
  :effect
    (and (not(check))
          (when (forall (?r - region) (imply (critical ?r) (exists (?c) (iscovering ?c ?r)))) (safe))
          (when (not (forall (?r - region) (imply (critical ?r) (exists (?c) (iscovering ?c ?r))))) (not (safe))))
    )
)
```

# Result Analysis (contd.)

- Camera use case - LTL
  - Camera cannot turn on right after upgradation
  - Has to wait at least 3 steps
  - LTL formula:  $\text{wait} \text{ U } (\text{upgrade} \ \& \ \text{XF}(\text{XXX} \ \text{turn\_on}))$



# Result Analysis (contd.)

- Camera use case - LTL (contd.)
  - Implementation of the Büchi Automaton

```
;; IMPLEMENTATION OF BUCHI AUTOMATON
(when
  (exists (?c - cam) (and (>= (current_state ?c) 1) (<= (current_state ?c) 3) (ison ?c)))
  (not (safe))
)
)

(forall (?c - cam)
  (when (and (= (current_state ?c) 0) (upgraded ?c) (not (ison ?c))))
    (increase (current_state ?c) 1)
  )
)

(forall (?c - cam)
  (when (and (>= (current_state ?c) 1) (<= (current_state ?c) 3))
    (increase (current_state ?c) 1)
  )
)
;;BUCHI AUTOMATON END
```

# Result Analysis (contd.)

- Camera use case - LTL (contd.)
  - Meaningless plan generated

```
78: UPGRADE C13
79: SAFETY-CHECK
80: TURN-OFF C6
81: SAFETY-CHECK
82: TURN-ON C6
83: SAFETY-CHECK
84: TURN-OFF C6
85: SAFETY-CHECK
86: TURN-ON C13
87: SAFETY-CHECK
88: TURN-ON C6
89: SAFETY-CHECK
```

# Result Analysis (contd.)

- Camera use case - LTL (contd.)
  - Pause action
    - To counter meaningless plans
    - Less costly plans

```
(:action PAUSE
  :parameters ()
  :precondition (and (not(check)) (safe))
  :effect
    (and (check))
)
```

# Result Analysis (contd.)

- Camera use case - LTL (contd.)
  - Plan generated after including LTL formula and Pause action

0: TURN-OFF C1	22: TURN-OFF C5	44: <b>TURN-ON C7</b>	66: <b>UPGRADE C12</b>
1: SAFETY-CHECK	23: SAFETY-CHECK	45: SAFETY-CHECK	67: SAFETY-CHECK
2: <b>UPGRADE C1</b>	24: <b>UPGRADE C5</b>	46: TURN-OFF C9	68: <b>TURN-ON C11</b>
3: SAFETY-CHECK	25: SAFETY-CHECK	47: SAFETY-CHECK	69: SAFETY-CHECK
4: TURN-OFF C2	26: <b>TURN-ON C4</b>	48: <b>UPGRADE C9</b>	70: PAUSE
5: SAFETY-CHECK	27: SAFETY-CHECK	49: SAFETY-CHECK	71: SAFETY-CHECK
6: <b>UPGRADE C2</b>	28: TURN-OFF C6	50: <b>TURN-ON C8</b>	72: PAUSE
7: SAFETY-CHECK	29: SAFETY-CHECK	51: SAFETY-CHECK	73: SAFETY-CHECK
8: TURN-OFF C3	30: <b>UPGRADE C6</b>	52: TURN-OFF C10	74: <b>TURN-ON C12</b>
9: SAFETY-CHECK	31: SAFETY-CHECK	53: SAFETY-CHECK	75: SAFETY-CHECK
10: <b>UPGRADE C3</b>	32: <b>TURN-ON C5</b>	54: <b>UPGRADE C10</b>	76: TURN-OFF C13
11: SAFETY-CHECK	33: SAFETY-CHECK	55: SAFETY-CHECK	77: SAFETY-CHECK
12: <b>TURN-ON C1</b>	34: TURN-OFF C7	56: <b>TURN-ON C9</b>	78: <b>UPGRADE C13</b>
13: SAFETY-CHECK	35: SAFETY-CHECK	57: SAFETY-CHECK	79: SAFETY-CHECK
14: <b>TURN-ON C2</b>	36: <b>UPGRADE C7</b>	58: TURN-OFF C11	80: PAUSE
15: SAFETY-CHECK	37: SAFETY-CHECK	59: SAFETY-CHECK	81: SAFETY-CHECK
16: TURN-OFF C4	38: <b>TURN-ON C6</b>	60: <b>UPGRADE C11</b>	82: PAUSE
17: SAFETY-CHECK	39: SAFETY-CHECK	61: SAFETY-CHECK	83: SAFETY-CHECK
18: <b>UPGRADE C4</b>	40: TURN-OFF C8	62: <b>TURN-ON C10</b>	84: PAUSE
19: SAFETY-CHECK	41: SAFETY-CHECK	63: SAFETY-CHECK	85: SAFETY-CHECK
20: <b>TURN-ON C3</b>	42: <b>UPGRADE C8</b>	64: TURN-OFF C12	86: <b>TURN-ON C13</b>
21: SAFETY-CHECK	43: SAFETY-CHECK	65: SAFETY-CHECK	87: SAFETY-CHECK

# Summary

- System
  - Constrained Devices
  - Device Management
- Autonomous and Adaptive
  - AI Planning
  - Application which takes decisions and executes them
- Policy Incorporation
- Linear Temporal Logic



# Future Work

- Test Scalability
- Test complex use cases
- Replanning
  - Fault Injection System
- Linear Temporal Logic
  - Doesn't fully capture the notion of time
  - Incorporating LTL like the policies
- EARS

# Future Scope

- LwM2M and CoAP
  - Industrial Standard
  - Will be widely accepted
- Long Term Projects
- Less Maintenance
- Will be able to handle unexpected scenarios
- Cost Effective

# References

1. Semantic Interoperability, Release 2.0, AIOTI WG03 – IoT Standardisation, 2015
2. Swarup Mohalik, Mahesh Babu Jayaraman, Badrinath Ramamurthy and Aneta Vulgarakis, “SOA-PE : A Service-oriented Architecture for Planning and Execution in Cyber-physical Systems”, in IEEE-IC -SSS 2015
3. Fabio Patrizi, Nir Lipoveztky, Giuseppe De Giacomo and Hector Geffner, “Computing Infinite Plans for LTL Goals Using a Classical Planner”
4. B.Nebel, “ The FF Planning System: Fast Plan Generation Through Heuristic Search ”, in Journal of Artificial Intelligence Research, Volume 14, 2001, Pages 253 - 302
5. Dan Klein and Pieter Abbeel. University of Berkeley. Artificial Intelligence [Online]. Available: <https://www.edx.org/course/artificial-intelligence-uc-berkeleyx>

# References (contd.)

6. Dr. Gerhard Wickler and Prof. Austin Tate. University of Edinburgh. Artificial Intelligence Planning [Online]. Available:  
<https://www.youtube.com/playlist?list=PLwJ2VKmefmxpUJEGB1ff6yUZ5Zd7Gegn2>
7. Leshan. [Online]. Available: <https://github.com/eclipse/leshan>
8. openHAB documents. [Online]. Available: <http://docs.openhab.org/>
9. Eclipse Smarthome.[Online].Available: <http://www.eclipse.org/smarthome/>
10. John Terzakis. Intel Corporation. EARS: The Easy Approach to Requirements Syntax.[Online]. Available:  
[https://www.iaria.org/conferences2013/filesICCGI13/ICCGI\\_2013\\_Tutorial\\_Terzakis.pdf](https://www.iaria.org/conferences2013/filesICCGI13/ICCGI_2013_Tutorial_Terzakis.pdf)

Questions...  
Suggestions...

[https://github.com/abinmathewabraham/machine\\_intelligence\\_based\\_iot\\_management\\_system](https://github.com/abinmathewabraham/machine_intelligence_based_iot_management_system)



Thank You