

The Thread logo, featuring a stylized 'T' symbol followed by the word 'THREAD' in a bold, sans-serif font.The IoTer logo, consisting of a circular icon with a network-like pattern followed by the word 'ioter' in a lowercase, sans-serif font.The SmartThings logo, featuring the word 'SmartThings' in a sans-serif font.

Thread and Samsung-IoTer

Real Time Emulation of Smart Home

19-Dec-2024

Bahubali B Gumaji
Manjunath N Sataraddi

Contents

Thread – Why ?

Thread – What ?

Thread Networking Stack

Thread Group Intro

OpenThread

Samsung-IoTer

Need of Thread

- As per current IoT statistics, there are well over 18.8 billion connected IoT devices around the globe.
- It's expected there will be 40 billion IoT devices by 2030.

Connected IoT devices forecast 2024–2030

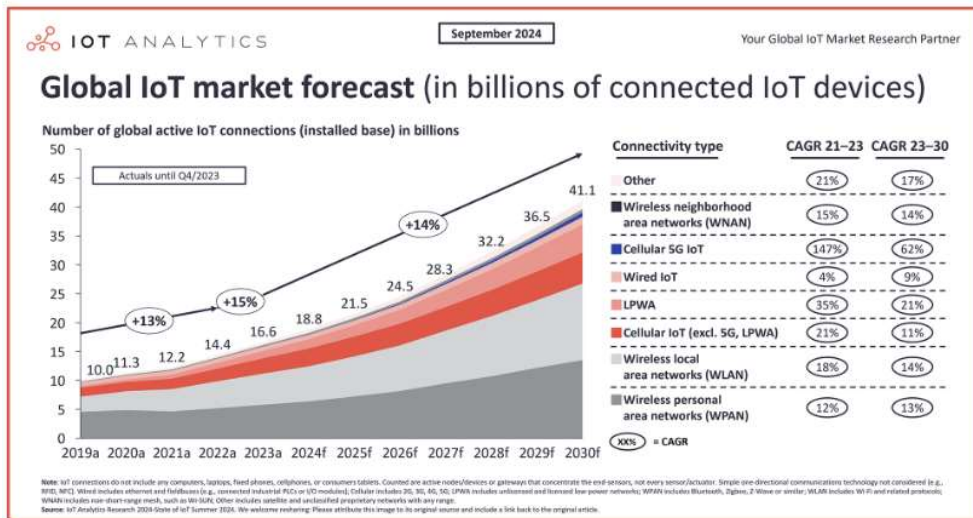
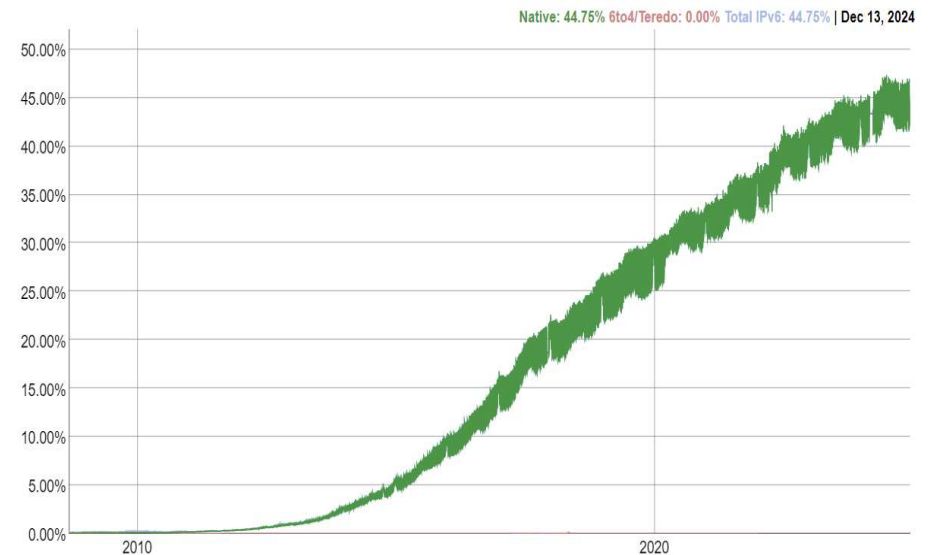


Image: Global IoT market forecast (in billions of connected IoT devices)

IPv6 Adoption



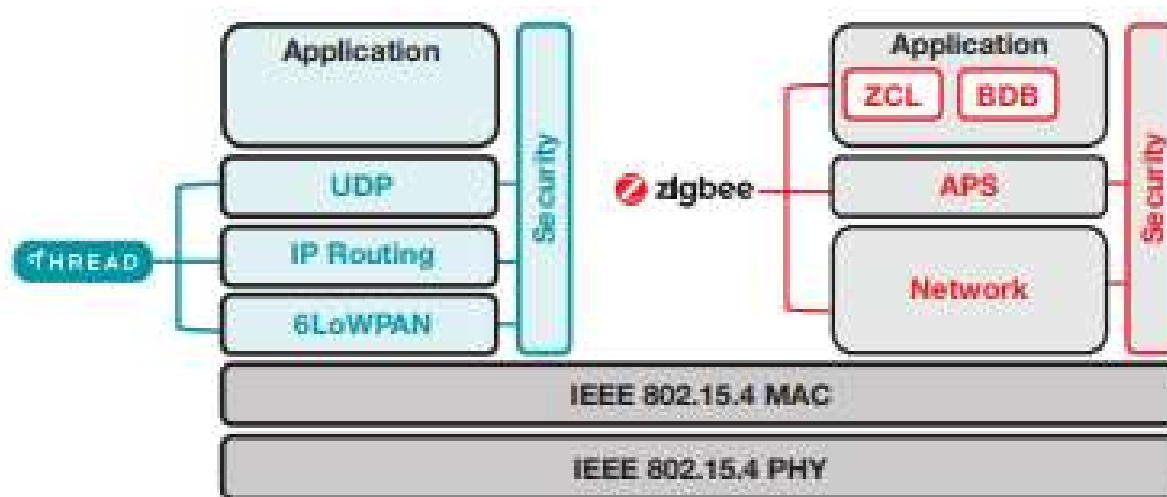
Reference Links:

- <https://iot-analytics.com/number-connected-iot-devices/>
- <https://www.google.com/intl/en/ipv6/statistics.html>

Need of Thread

2/2

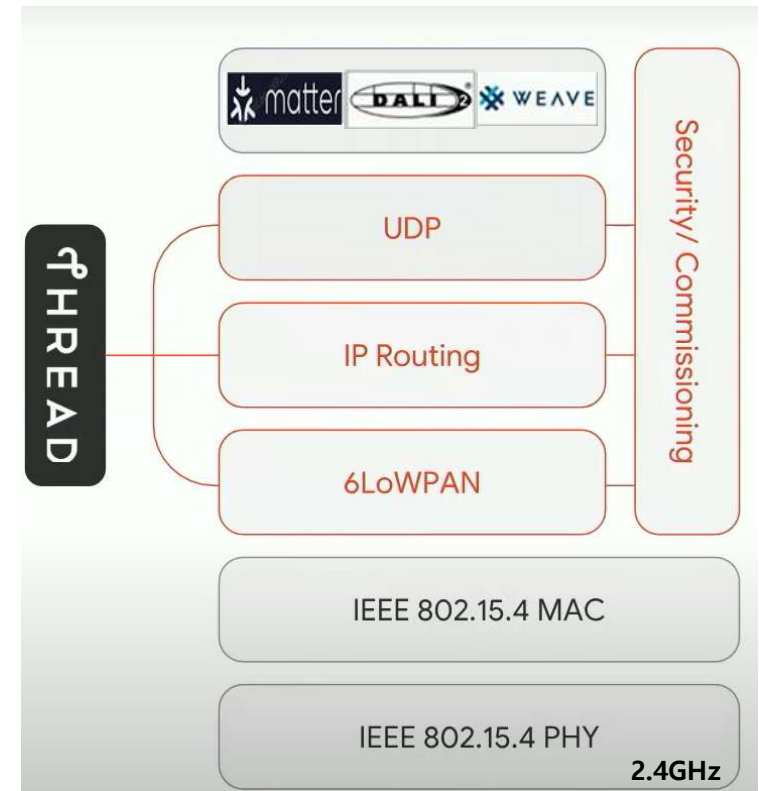
- Zigbee defines all layers in the OSI model
- Non IP mesh network
- Network header (and network addresses) must be adapted to IP
- Payload usually re-secured at IP Gateway and may require some adaptation for IP
- Proprietary translation, hampered end-to-end security



What is Thread?

An IP based, low power, secure mesh networking technology for IOT products.

- Built on proven technologies
 - Thread protocol is application layer agnostic and does not define an application layer
 - Ipv6 and 6LoWPAN – Thread is an Internet Protocol version 6 (IPv6) based mesh network, 6LoWPAN defines how to sent IPv6 packets over 802.15.4
 - IEEE 802.15.4 – the same radio used for Zigbee networking protocol



Thread | Mesh Network

Types:

FTD: Full Thread Device

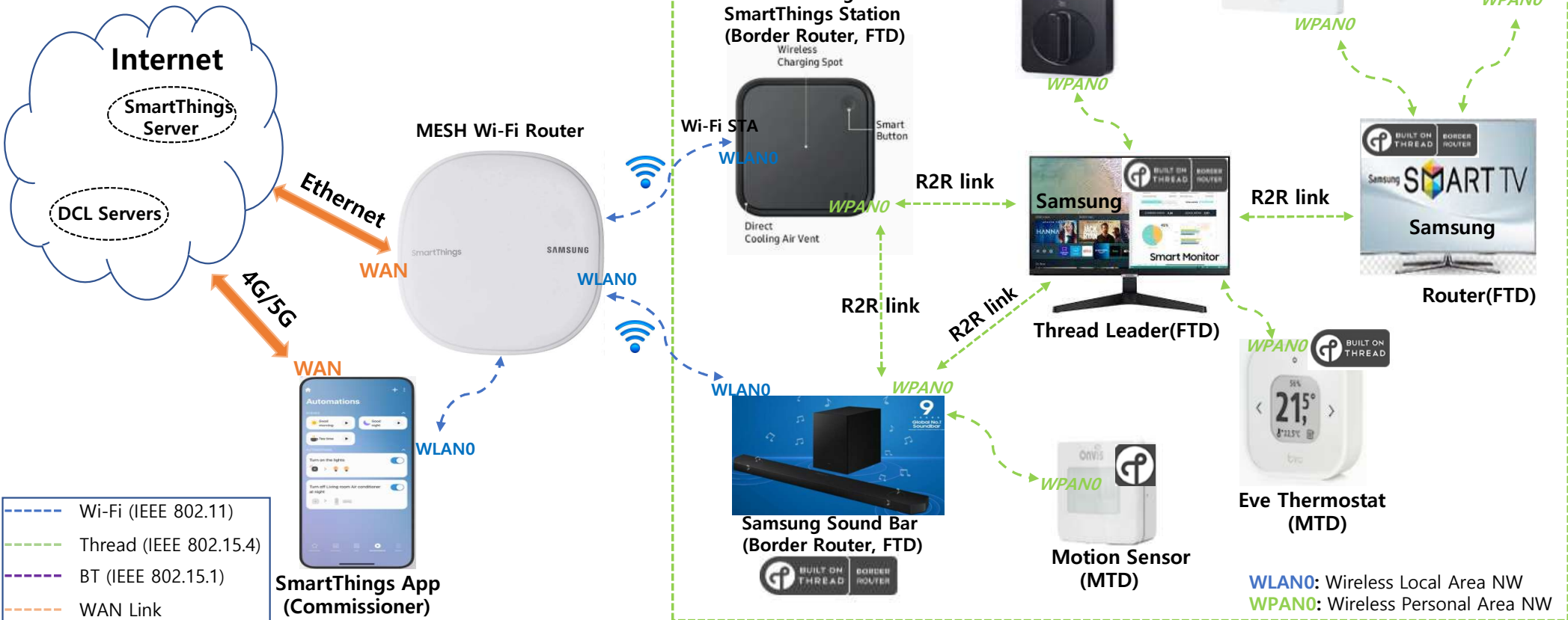
- Radio always ON
- Main Powered
- Router Role

MTD: Minimal Thread Device

- Radio ON(need based)
- Battery Powered
- IoT End Node

Roles:

- End Device
- Router
- Border Router
- Thread Leader



Thread Group

Mission

- Thread Group brings the Internet to IoT devices through its IP based, low power, secure mesh networking technology.

Vision

- To serve as the foundation of the Internet of Things by seamlessly connecting devices where we live and work.

- A Standardization body for creating, maintaining, and developing Thread protocol specification.
- Educate product developers and consumers on the key features and benefits of Thread.
- Ensure a great user experience through rigorous, meaningful product certifications

Sponsor members:

Apple

Google

LUTRON

NORDIC
SEMICONDUCTOR

NXP

OSRAM

Qualcomm

SIEMENS
Logically for Life

SILICON LABS

somfy.

Yale

SmartThings

OpenThread Project

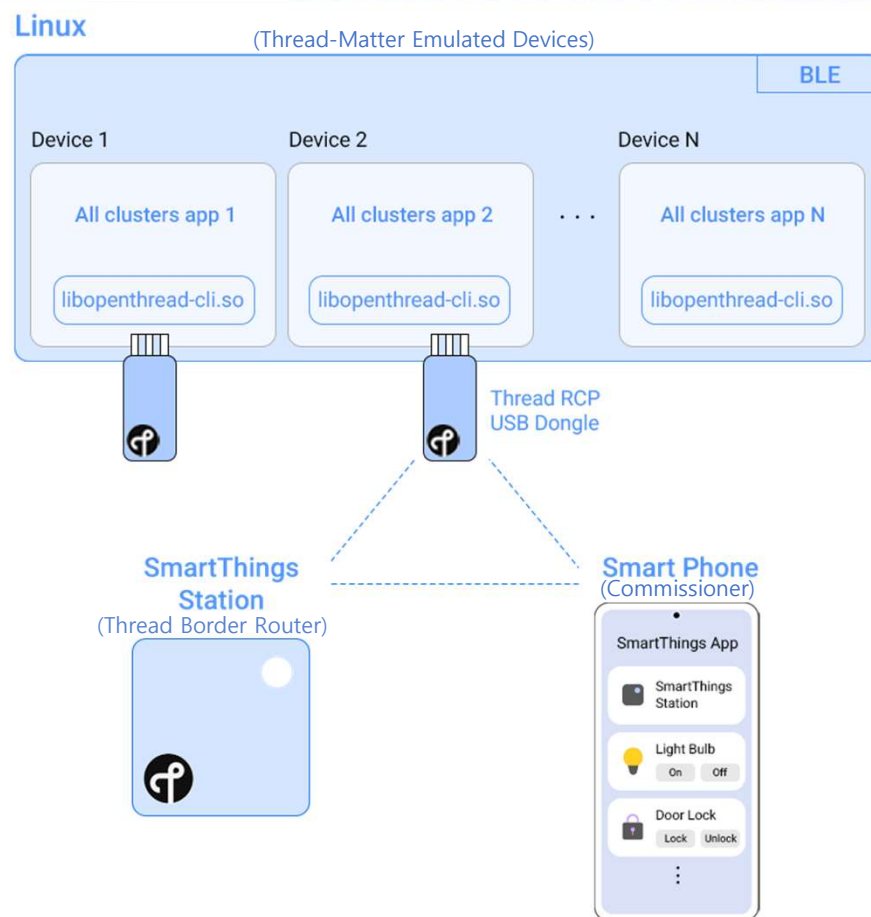
- OpenThread released by Google is an open source implementation of Thread networking protocol
- Key Components
 - Core Stack – Implementation of Thread specification
 - Border Router – Implementation of Thread Border Router
 - Commissioner – Implementation of Thread Commissioner
 - Network Simulator – Simulate large scale Thread networks
- Licensed under BSD-3. Hosted on Github - <https://github.com/openthread/>
- More info about OpenThread available in <https://openthread.io/>

The screenshot displays five GitHub repository cards for OpenThread components. Each card includes the repository name, a brief description, the programming language, and the number of stars and forks.

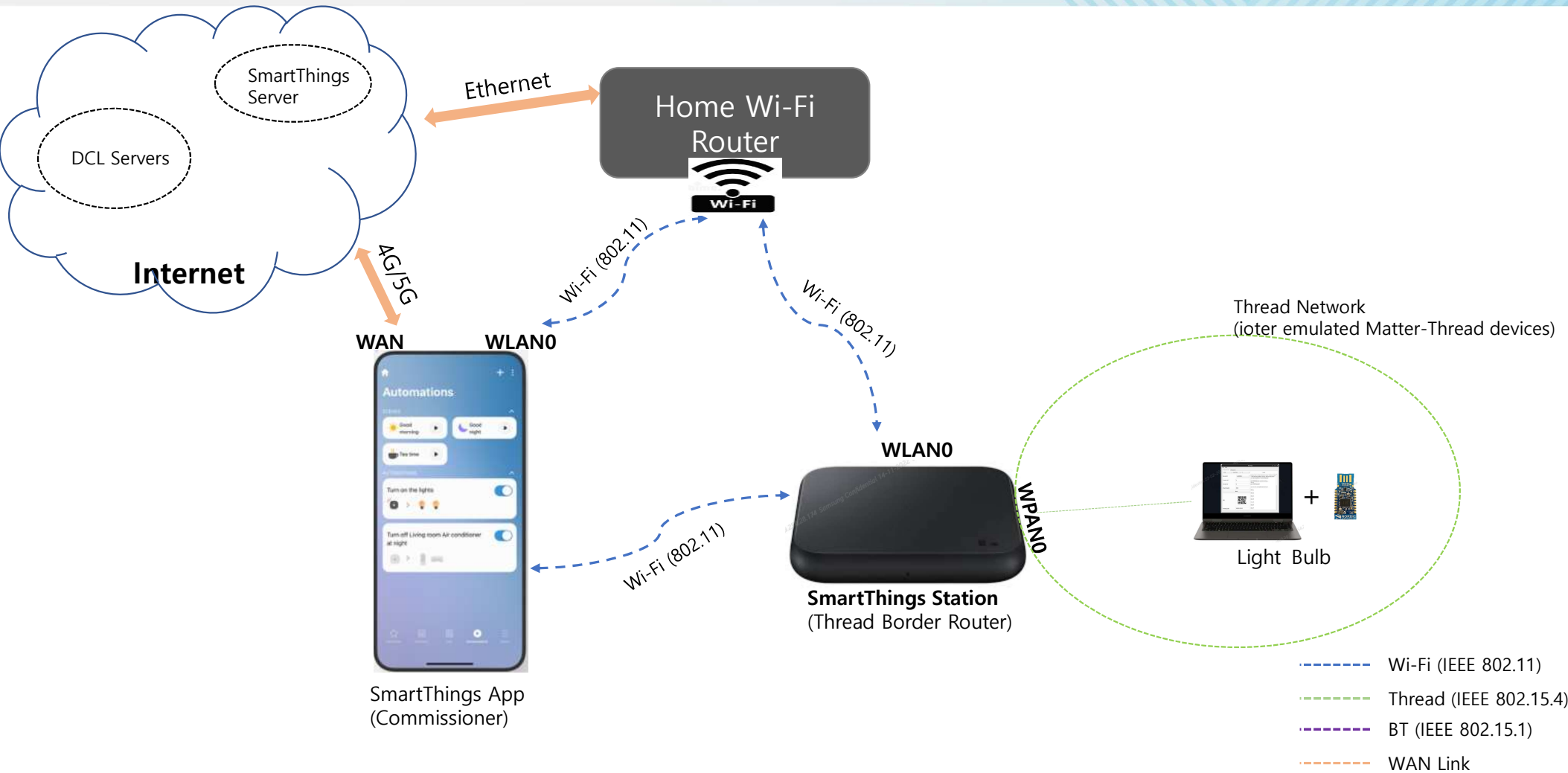
Repository Name	Description	Language	Stars	Forks
<code>openthread</code>	OpenThread released by Google is an open-source implementation of the Thread networking protocol	C++	2.5k	819
<code>ot-br-posix</code>	OpenThread Border Router, a Thread border router for POSIX-based platforms.	C++	182	121
<code>ot-commissioner</code>	OpenThread Commissioner, a Thread commissioner for joining new Thread devices and managing Thread networks.	C++	20	16
<code>ot-rtos</code>	OpenThread RTOS, an integration of OpenThread, LwIP, and FreeRTOS.	C	95	36
<code>ot-ns</code>	OpenThread Network Simulator, a discrete event simulator and visualization tool for Thread networks.	Go	14	9

How IoTer works?

- IoTer emulates all Matter supported devices with a Linux PC and Thread RCP dongle.
- IoTer runs the all-clusters-app of Matter on a Linux PC to emulate multiple instances of various Matter supported IoT nodes.
- Each of these IoT nodes uses the underlying Thread RCP-based USB dongle (radio) for data transmission.
- By using the SmartThings Station as a border router and the SmartThings Application along with emulated IoT nodes, we can configure a smart home.

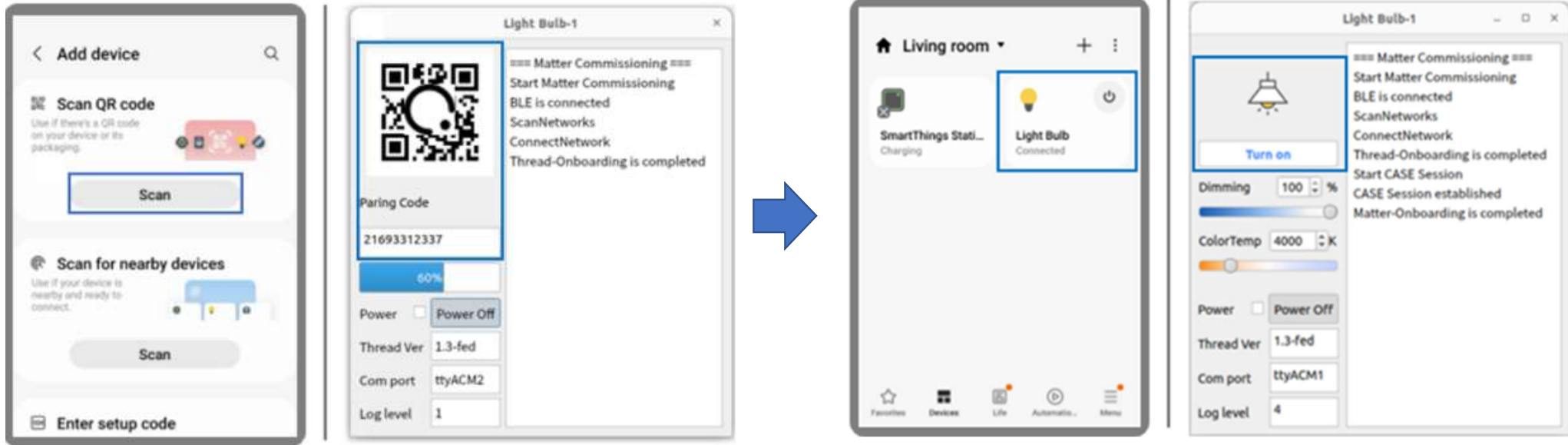


IoTer | Onboarding



IoT | Onboarding

- In the SmartThings App, click on the add device button in the upper right corner. Use Scan QR code or Scan for nearby devices to start onboarding
- After onboarding your virtual device, your device is now controllable from within the SmartThings app.

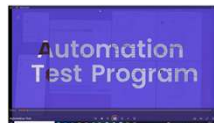


IoTer | Automation Testing

Use Automations to validate the connectivity and stability of various IoT device types.

The screenshot shows the IoTer Automation Testing interface. The main workspace contains a sequence of actions: 'Loop Start/End' (1), 'Device Command' (2) with a dropdown menu (4) and a value input (6), 'Sleep' (3) with an interval input (7) and a loop count input (8), and 'LoopEnd'. The 'Log Window' (13) displays the following logs: [2023-04-26-16:15]: Clean all, [2023-04-26-16:15]: File loaded : "/home/vedansh/ioter-ui-app/automation/output/test.xml" Successfully, [2023-04-26-16:15]: Automation Test Started, and [2023-04-26-16:15]: Automation Test stopped. The bottom control bar includes 'Clear All' (9), 'Run' (10), 'Clear Log' (11), and a progress bar (12) showing 100% completion.

- 1.Starts/ends the loop.
- 2.Add a command for the onboarded device.
- 3.Add sleep for a given interval as specified below.
- 4.Device type (light bulb, contact sensor, etc.).
- 5.Supported commands for the onboarded device.
- 6.Device command's value (for example, light bulb is On or Off).
- 7.Sleep interval, in seconds.
- 8.Loop count and loop interval, in seconds.
- 9.Clear all loops and commands.
- 10.Run the automation script.
- 11.Clear the log window.
- 12.Script completion progress bar.
- 13.The log window, showing activities including script loads, executions, saves, and number of successful/unsuccessful commands.



IoTer

Benefits:

- **Flexibility** - Multiple types of IoT devices can be implemented using a single RCP dongle.
- **Multi-Device Support** - Each RCP dongle supports a single device. ioter supports up to 10 RCP dongles at a time.
- **Low Cost** - Limited expenses for testing various IoT device types.
- **Time Saving** - Virtual devices on demand - no need to search for and procure multiple IoT device types.
- **Easy To Use** - Quickly configure and control various IoT device types.
- **Automated Testing** - Repeated testing through scripts can validate device stability and connection.

Supported IoT devices:

- Light Bulb
- Door Lock
- Contact Sensor
- Temperature Sensor
- Humidity Sensor
- Light Sensor
- Window Covering
- Occupancy Sensor
- OnOff Plugin





Thank You