



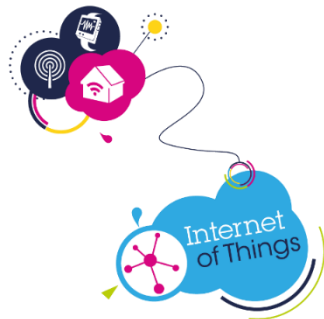
# BlueNRG-MESH

## Overview of the BlueNRG-MESH SDK for the ST Bluetooth low energy SOCs

September 2019

Francesco DODDO

Salvo BONINA



**ST Developers  
Conference**

September 12th, 2019  
Santa Clara Convention Center - Mission City Ballroom  
Santa Clara, CA



# BlueNRG-Mesh



**Bluetooth Mesh Basics**

**BlueNRG-Mesh SDK Solution**

**Getting started with BlueNRG-Mesh**



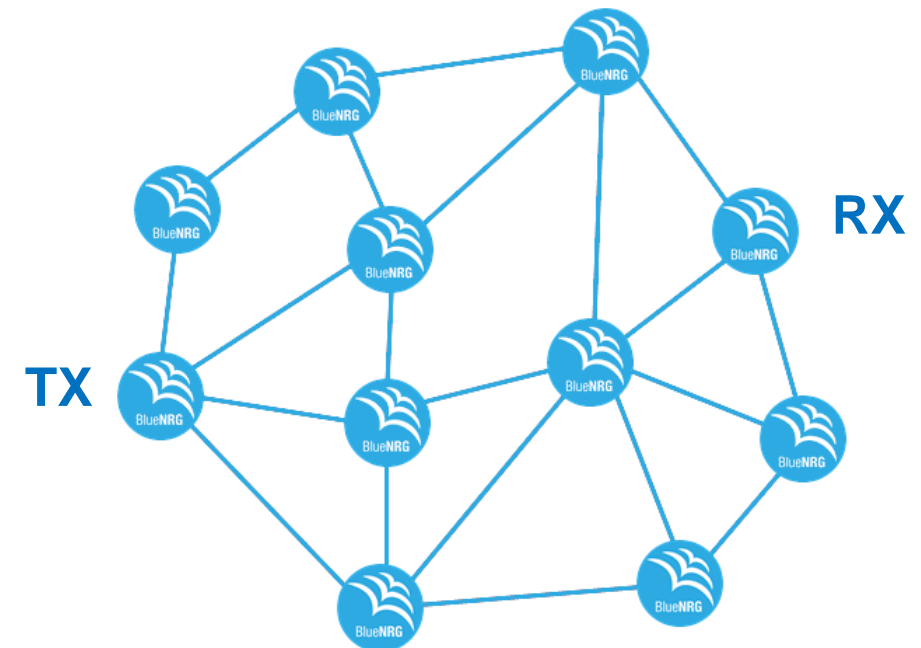
# What is the BLE Mesh?

3

**July 18, 2017 - Bluetooth SIG Announces Mesh Networking Capability**

Brings proven, **global interoperability** and the mature, **trusted ecosystem** of Bluetooth technology to industrial-grade device networks

- A standard network protocol to extend the communication range with a mesh network topology between several nodes over multiple hops when there is not direct reachability between two nodes.
- The BLE Mesh is standardized by the **Bluetooth Core** also known as **Special Interest Group (SIG)**.



# BLE MESH: Backed up by Industry Leaders



companies supporting the launch  
of Bluetooth mesh networking



Bluetooth  
5



# Extending Bluetooth Capabilities

5

The Bluetooth Mesh network topology optimizes the power consumption.

## PAIRING one-to-one



1 MASTER with  
Up to 8 SLAVES



## DATA TRANSFER

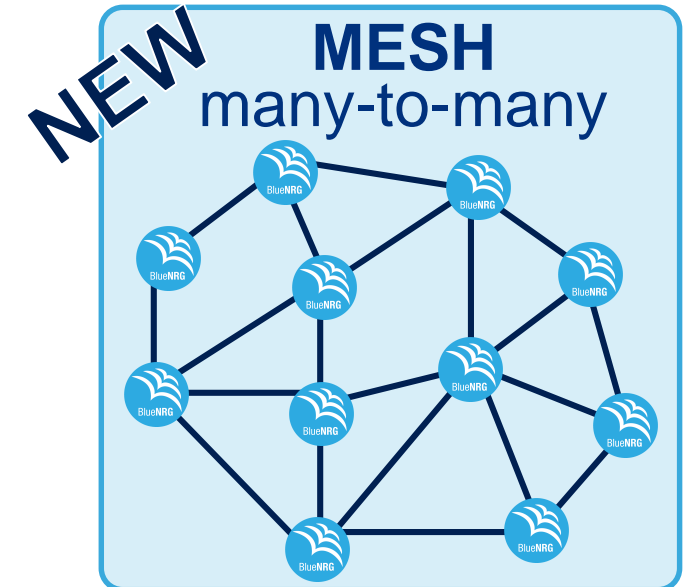
- Sports & fitness devices
- Health and wellness devices
- Peripherals and accessories

## BROADCASTING one-to-many



## LOCALIZED INFORMATION

- Point of interest beacons
- Item finding beacons
- Way finding beacons



## LARGE DEVICE NETWORKS

- Building automation
- Wireless sensor networks
- Asset tracking

Signal range depends on RF output power

Coverage extended by relaying  
messages over multiple hops



# Bluetooth® Mesh Applications

6

The Bluetooth SIG MESH extends the capabilities of Bluetooth Smart chips to answer **more and more complex applications.**

The protocol has been developed with the **Smart Lighting industry** in mind.

- Lighting
  - Interface to major light IOs (DALI, PWM, etc.)
- Smart Home and Building automation
  - Heater/Fan control
  - Temperature / Shutter control
- Smart Industry
  - M2M control
- Wireless sensor networks




Source : Bluetooth® SIG

  
Smart Home/Building



  
Smart Things



  
Smart Industry



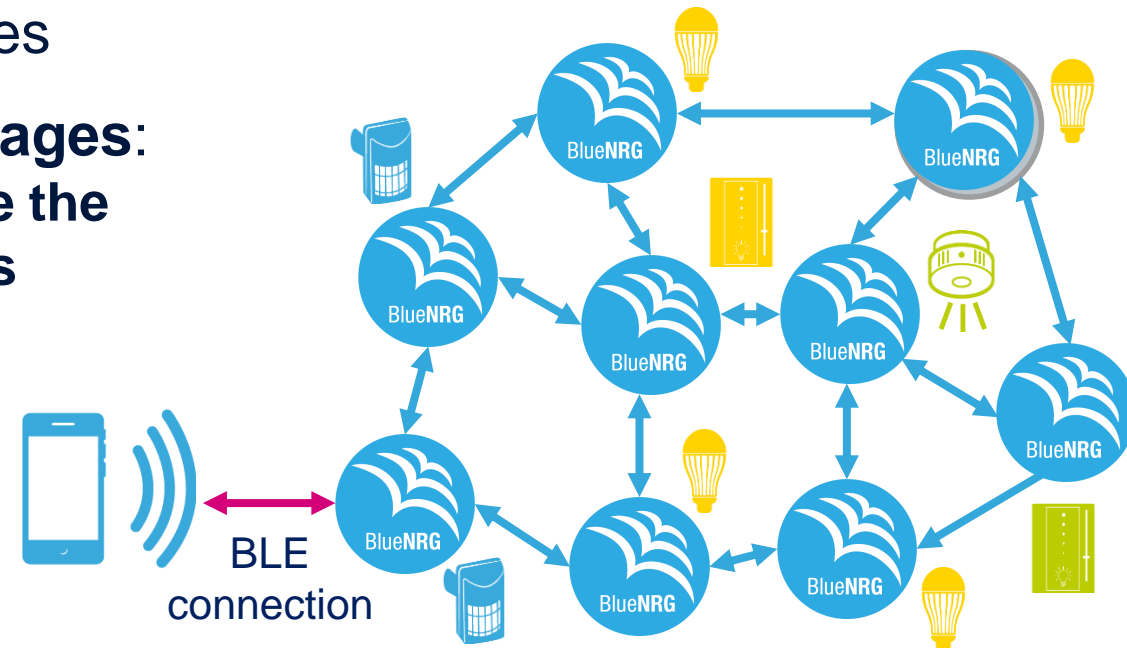
# Bluetooth® Mesh Topology

## Managed Flooding

7

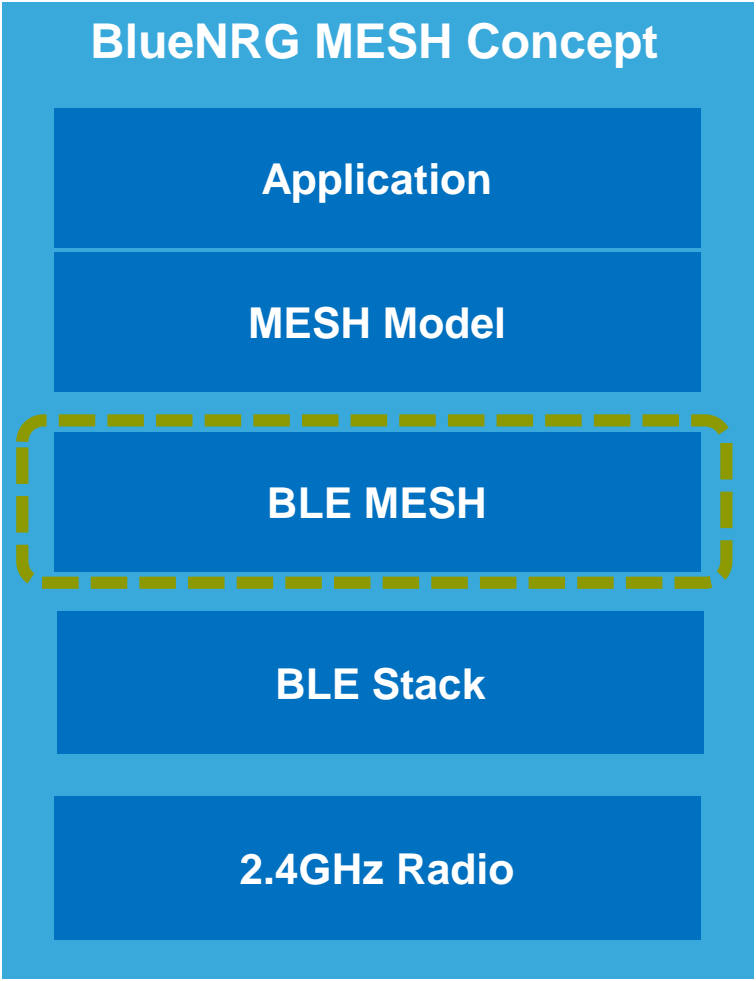
The Bluetooth Mesh network mechanism is based on a **managed flooding protocol**

- **Message relay** extends the communication range (Multi-hop data transmission)
- **No single point of failure**
  - does not require any centralized operation nor coordination
  - dynamic self healing
- **Direct communication** between contiguous nodes
- **Rules to restrict** devices from **re-relaying messages**:
  - Messages contain a **Sequence Number** to **optimize the network usage and protect against replay attacks**
    - Reject messages already received
  - **TTL** (Time To Live) method:
    - Limit the number of times a message is relayed





# Bluetooth MESH vs BLE Stack



- Bluetooth Mesh runs on top of the BLE Stack
- It is not linked to BLE 5.0
- It is backward compatible with BLE 4.x





# Security, Attacks and Protection

9

- The Mesh security uses a privacy mechanism called **obfuscation** through **AES** to encrypt the **header** information.
- **Messages** are **encrypted twice**. Once with an Application or device key and the second time with a Network key.
- Mesh security **protects the network against** third-party interference and monitoring: Replay/Eaves Dropping **attacks**.
- Nodes can be removed from the network securely, preventing trash-can attacks, by erasing the security keys stored in the flash memory.



# Security in the Bluetooth Mesh: Keys

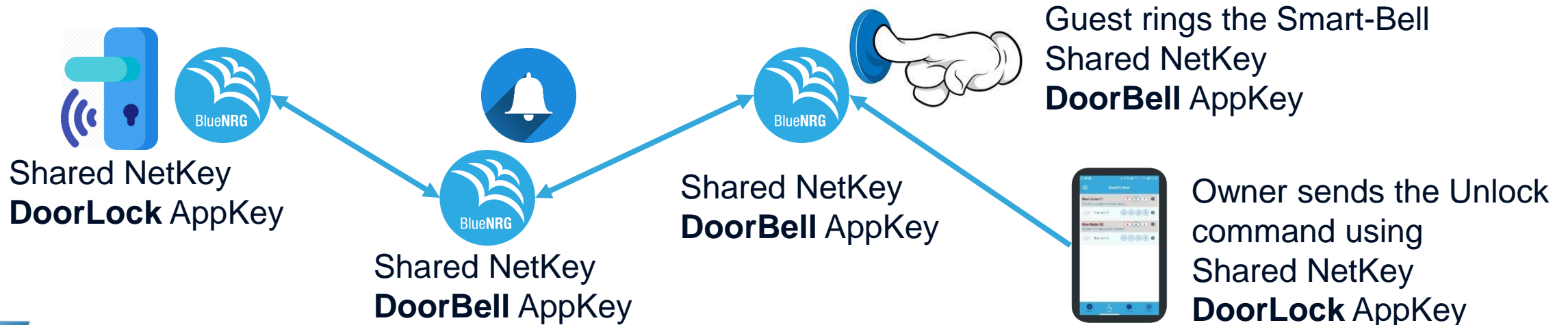
10

- **Device Key:** Never transmitted over air
  - **Unique key for each single device**, only known by the Provisioner (i.e. Smartphone) and the device, used for provisioning, configuration and key management.
- **Network Key:** Provided by Provisioner
  - Shared across all Nodes in the network, allows a node to decrypt message's header and to relay messages throughout a network
- **Application Key:** Provided by Provisioner
  - Different applications have a different “**Application key**” used to **encrypt/decrypt** messages payload (application data).

# Why Do You Need Both Network Key and Application Key?

11

- The NetKey provides security/authentication for all communication at network layer
- The AppKey protects the application data. Messages can be relayed by any nodes in the network without being able to read or change the application data.
- We can't have a compromised **Door Bell** allow anybody to unlock **Door Lock**

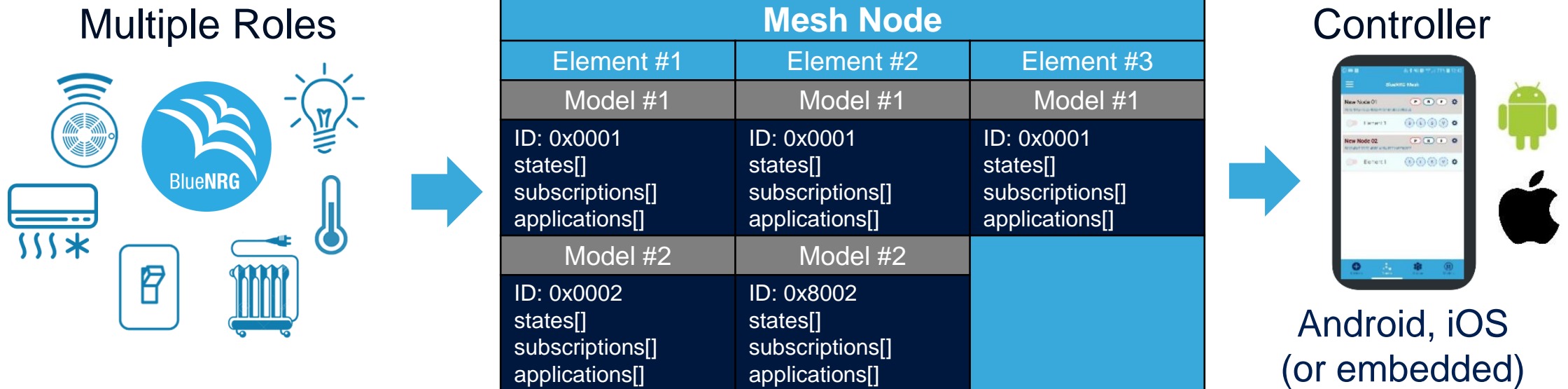


# Bluetooth® Mesh for Smarter Devices

12

How Node Features are Exposed – Two Entities: Elements and Models

Using Elements and Models helps **exchange messages** between a BLE device and a Smart Application (Controller) for simple **control and monitoring applications**



- **Elements:**

- define the **functionalities** of a single node, i.e. a light fixture may have two lamps, each of them is a **separate element** independently controlled. Each element contains models

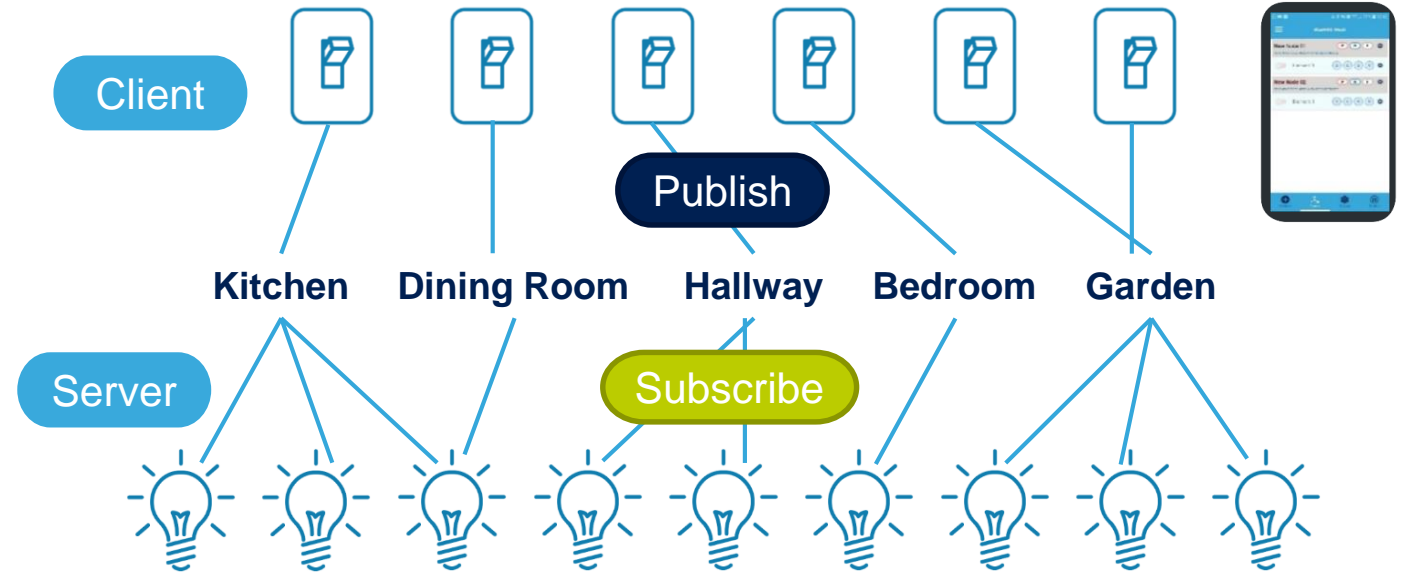
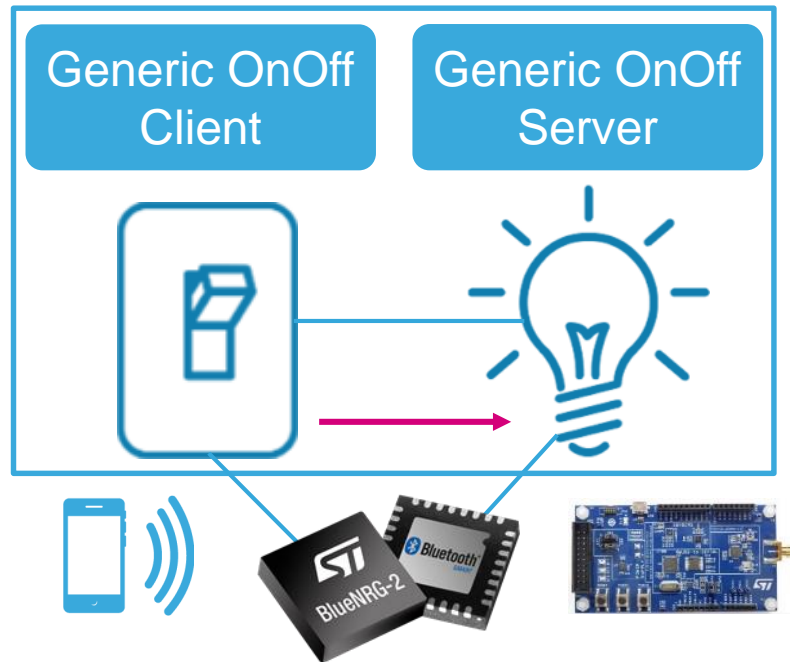
- **Models:**

- represent specific **Services** and define a set of **States and Messages** supported by the element; (i.e. a lamp supports the **Lighting** model for On/Off Control or Dimming)

# The Mesh Messaging Model

## Publish and Subscribe Paradigm

13



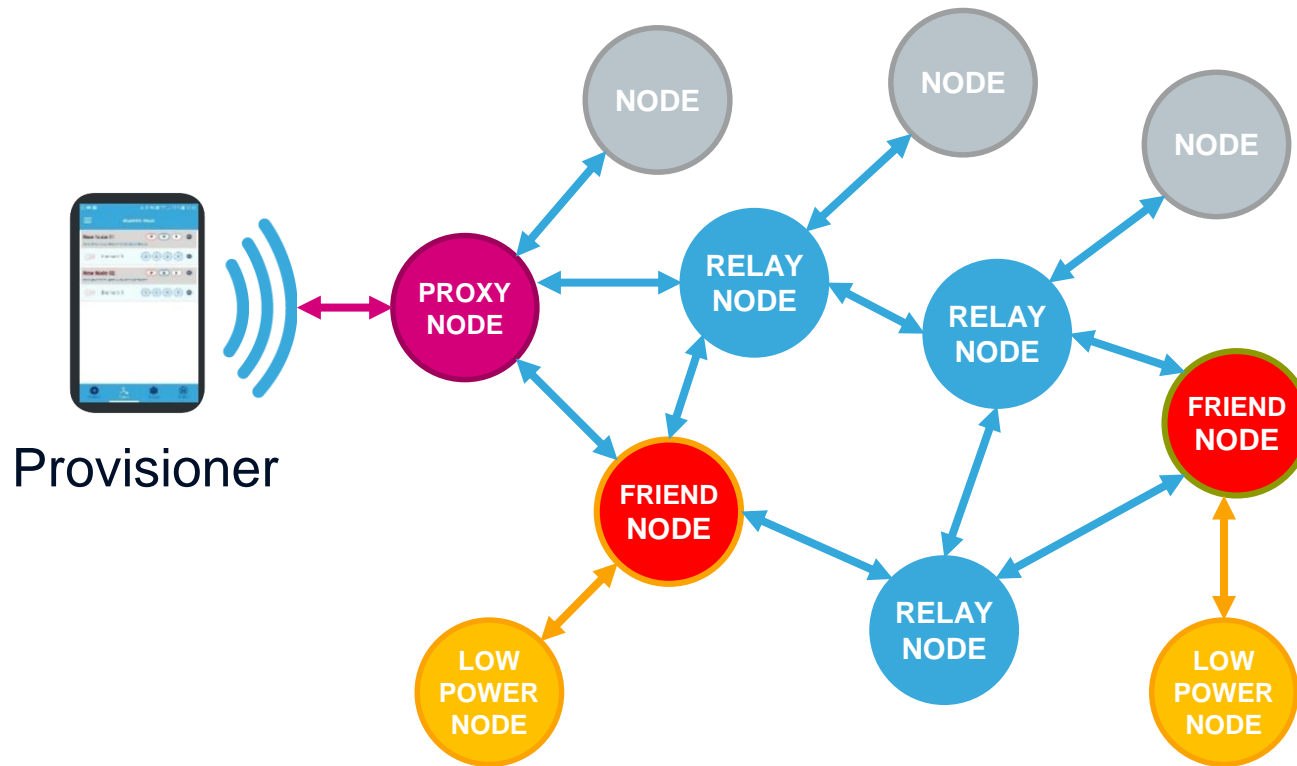
- Publish: send a message to unicast /virtual /group address
  - A **Client** device (e.g. Switch/Smartphone) can **publish messages** (e.g. ON/OFF control) and a **Server** device (e.g. a light bulb) can **subscribe** and be **notified** of new command arrival.
  - Also, a Client can **publish** a message to a **multicast address** (i.e. **Group**), and **multiple nodes** can **subscribe** to such specific address.

# Bluetooth® Mesh Network Topology

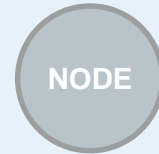
## Nodes Types

14

To stay efficient, the BLE Mesh take advantage of a **managed flooding network** mechanism. Compared to routed protocols, it is **much more simpler** to deploy.



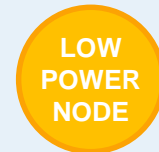
- Expose the interface for Smartphone/ Tablet to interact with a mesh network



- Simple leaf node whom you don't want to relay messages (Weak nodes or Resource constrained nodes)



- Able to retransmit received messages
- Enable multiple “hops” in the network



- Battery operated devices
- Primarily send but Rarely receive messages
- No need 100% duty cycle: mostly sleep



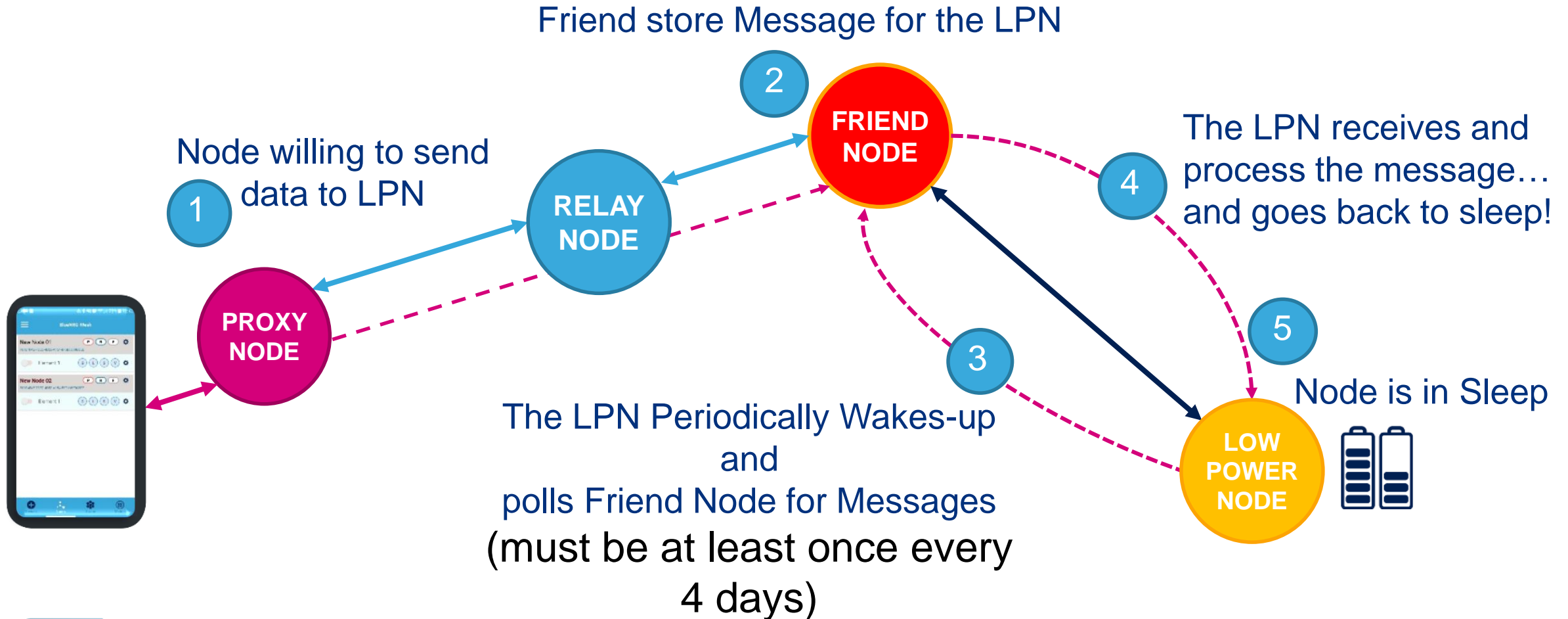
- Stores messages addressed to LPNs and delivers them whenever the LPN polls for “waiting messages”



# Bluetooth® Mesh Entities

## Low Power & Friendship Node

15



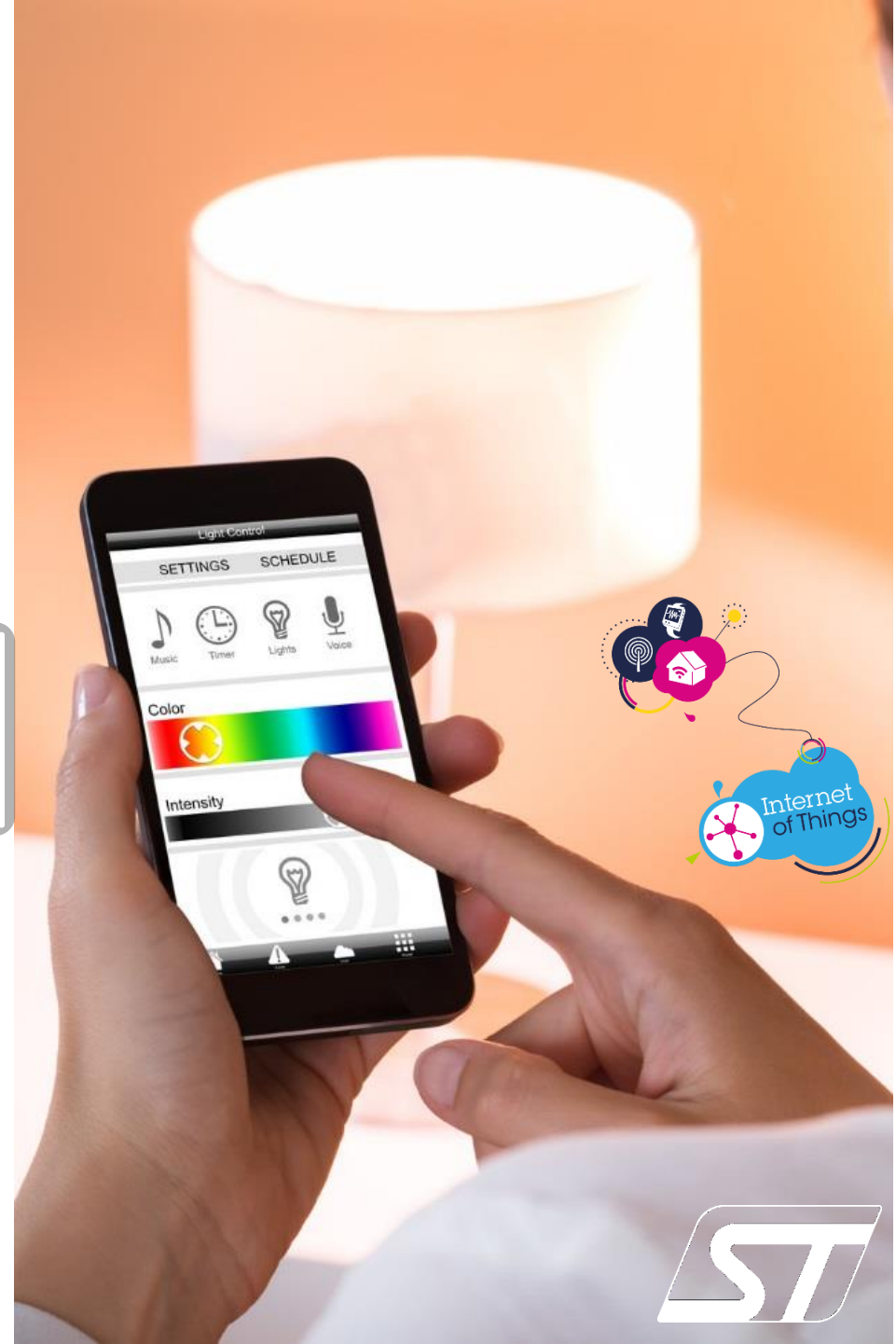
# BlueNRG-Mesh

Bluetooth Mesh Basics



BlueNRG-Mesh SDK Solution

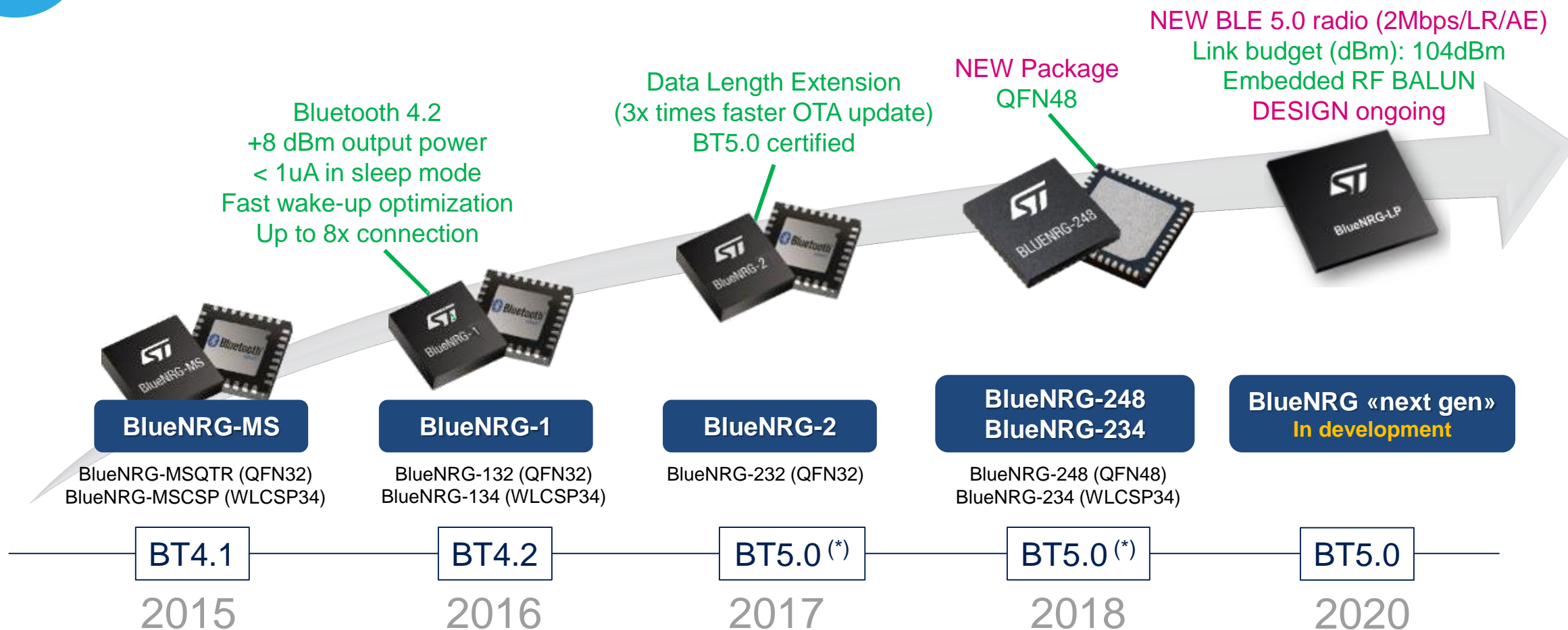
Getting started with BlueNRG-Mesh





# BlueNRG chipset evolution

17

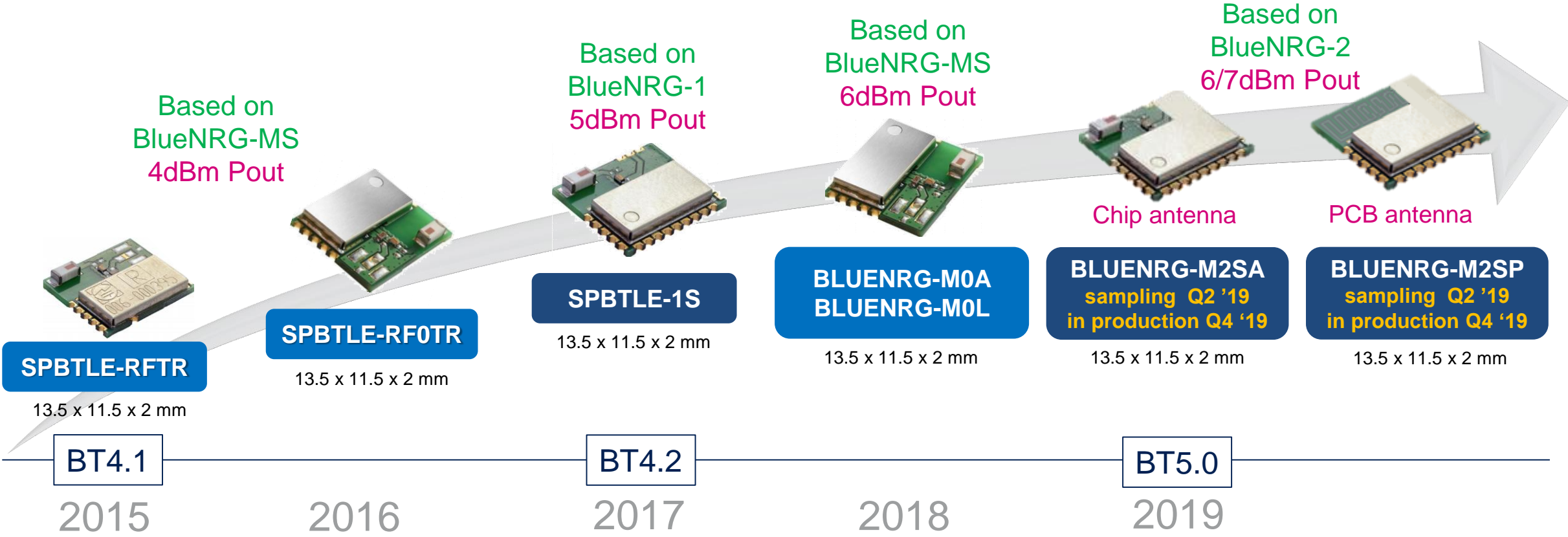


## Bluetooth LE product roadmap

(\*) Certified for BLE 5.0. It does not support other optional BLE 5.0 features like 2 Mbit/s, Long Range and Advertising Extensions.



# BlueNRG Modules Portfolio



## Bluetooth LE product roadmap



NETWORK PROCESSOR  
SoC PROCESSOR



- SDK
  - **Firmware, Android and iOS app sources**
  - [www.st.com/blemesh](http://www.st.com/blemesh)

## • ST BLE Mesh app on Store



- BlueNRG-mesh community forum
  - <http://community.st.com/blemesh>

# STSW-BNRG-Mesh

19

The screenshot shows the STSW-BNRG-Mesh product page. The header includes the ST logo and navigation links for Products, Applications, Tools & Software, and Login. The breadcrumb trail is: Embedded Software > Wireless Connectivity Software > STSW-BNRG-Mesh >. The main heading is "STSW-BNRG-Mesh" with a status indicator "ACTIVE". Below this, it says "Mesh over Bluetooth Low Energy". There are two buttons: "Get Software" and "Download databrief". The page has tabs for Overview, Tools & Software, and Resources. The Overview tab is selected, showing a description of BlueNRG-Mesh as a software solution for connecting multiple BLE devices in Mesh networks for IoT solutions. It also mentions that the solution is compatible with the ST BlueNRG product family range. At the bottom, there are three diagrams showing the software stacks for BlueNRG-Mesh SDK, Android SDK, and iOS SDK, detailing the layers from the Bluetooth chip up to the application level.

BlueNRG-Mesh is a software solution for connecting multiple BLE (Bluetooth low energy) devices in Mesh networks for Internet of Things (IoT) solutions. It enables true two-way communication between Bluetooth-enabled devices in powerful, secure, integrated and range-extending Mesh networks.

The solution is compatible with the ST BlueNRG product family range.

# Bluetooth SIG Certification

20

- Mandatory Features
  - **FW library compliant** to the **SIG specs v1.0**
  - Role: Node (Features: Proxy, Relay)
  - Bearer: Advertising, GATT
  - Provisioning: PB-GATT
  - Provisioning Protocol: Provisioning Server
  - Foundation Mesh Models: Configuration Server, Health Server
- Optional Features
  - **Friend Node**
  - **Low Power Node**
- ST Additional Features
  - **Models: Generic, Lighting, Sensor, Vendor, Time & Scene**
  - Light Intensity control (i.e. Dimming) using PWM control
  - **Provisioning information sharing** between mobile Phones/Tablets (Android and iOS) **via JSON file by eMail & Cloud Synchronization**



Bluetooth Qualification  
Certified  
**BlueNRG-Mesh**  
QDID: 116029





# BlueNRG-Mesh in Numbers

21

**Hop latency**  
30 ms max

**Power consumption**  
7 mA average  
(with no LPN/  
Friendship)

BLE Mesh 1.0 SIG  
certified

**Provisioning procedure**  
10-15 seconds average

**Mesh Library  
Memory footprint**  
55.2 KB Flash  
5.3 KB RAM

**Message Payload**  
8 bytes unsegmented  
64 bytes segmented



**Network size**


Max 32767 nodes for a single network  
Max network diameter of 126 hops

Compatible with both  
BLE 4.X or 5.0



# BlueNRG-Mesh Roadmap

22

STSW-BNRG-Mesh v1.05.000 X-CUBE-BLEMESH1 v1.0.0	STSW-BNRG-Mesh v1.06.000 X-CUBE-BLEMESH1 v1.1.0	STSW-BNRG-Mesh v1.07.000 X-CUBE-BLEMESH1 v1.2.0	STSW-BNRG-Mesh v1.08.000 X-CUBE-BLEMESH1 v1.3.0* FP-SNS-BLEMESH1 v1.1.0* * Partially supported	STSW-BNRG-Mesh v1.09.000 X-CUBE-BLEMESH1 v1.4.0 FP-SNS-BLEMESH1 v1.2.0
<ul style="list-style-type: none"><li>• Friendship &amp; Low Power</li><li>• <b>Health Model</b></li><li>• <b>Lighting Model</b></li><li>• Key-refresh</li><li>• IV Update procedure</li><li>• Database transfer (via Cloud)</li><li>• BT Mesh v1.0 Profile Certification QDID 116029</li><li>• Base for X-CUBE-BLEMESH1 v1.0.0</li></ul> 	<ul style="list-style-type: none"><li>• <b>Performance Optimization</b></li><li>• <b>Generic Model</b><ul style="list-style-type: none"><li>• Server: OnOff, Level</li><li>• Client: OnOff, Level</li></ul></li><li>• <b>Lighting Model</b><ul style="list-style-type: none"><li>• Server: Lighting HSL</li></ul></li><li>• Android &amp; iOS supporting HSL Model</li><li>• CID &amp; PID Configuration</li><li>• Static OOB Provisioning</li></ul>	<ul style="list-style-type: none"><li>• <b>Sensors Model</b></li><li>• <b>Lighting Model</b><ul style="list-style-type: none"><li>• Client</li></ul></li><li>• <b>Time and Scene Model</b><ul style="list-style-type: none"><li>• Empty wrappers</li></ul></li><li>• <b>PB-ADV – Provisioning over advertising</b></li></ul>	<ul style="list-style-type: none"><li>• <b>Multiple App Keys (up to 2 keys)</b></li><li>• <b>Model data/states saving into flash</b></li><li>• <b>Models Added: Generic PowerOnOff Server and Setup, Generic Default Transition Time, Light LC Server (Mode, OM, Light OnOff),</b></li><li>• <b>Light LC Controller state machine for Occupancy/Ambient Lux Level</b></li><li>• <b>Output/Input OOB, Public Key OOB Provisioning</b></li><li>• <b>Vendor Model moved to appl layer</b></li><li>• <b>FW example for STEVAL-BCN002V1B and STEVAL-BLUEPLUG1</b></li></ul>	<ul style="list-style-type: none"><li>• Multiple Network Keys</li><li>• <b>Embedded Provisioner</b></li><li>• <b>BT Mesh v1.0 Server Models Certification</b></li></ul>



August 2018

November 2018

March 2019

August 2019

November 2019

<http://www.st.com/blemesh>

<http://community.st.com/blemesh>

<https://itunes.apple.com/us/app/bluenrg-mesh/id1348645067>

<https://play.google.com/store/apps/details?id=com.st.bluenrgmesh>

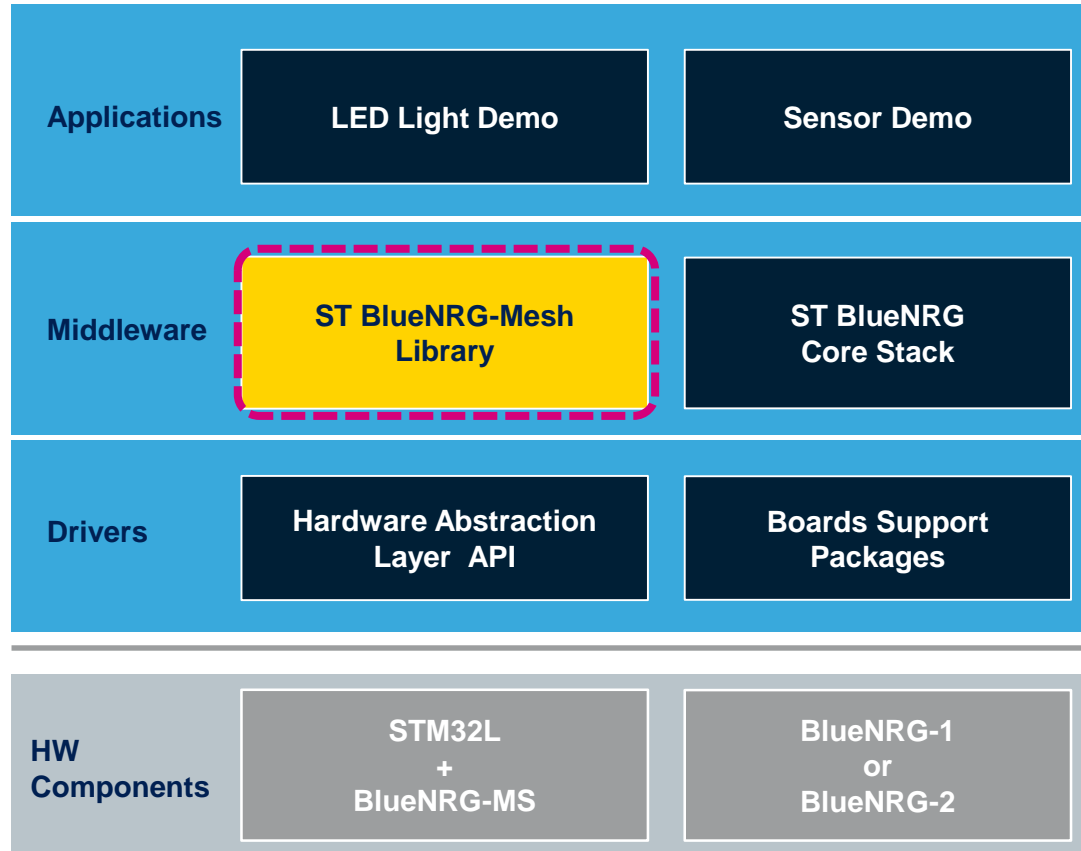


# BlueNRG-MESH SDK

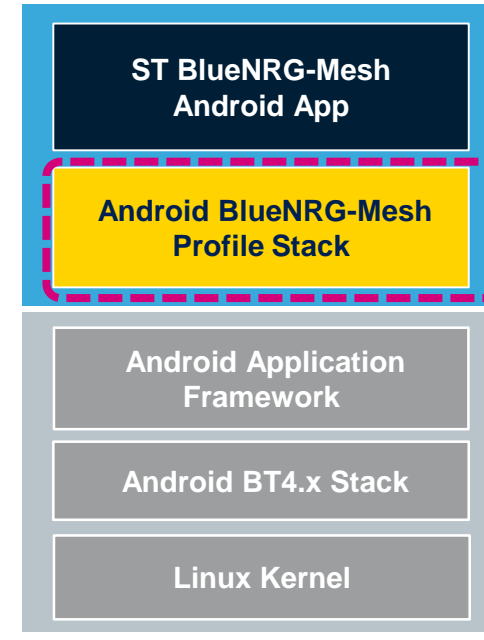
## for Embedded, Android and IOS

23

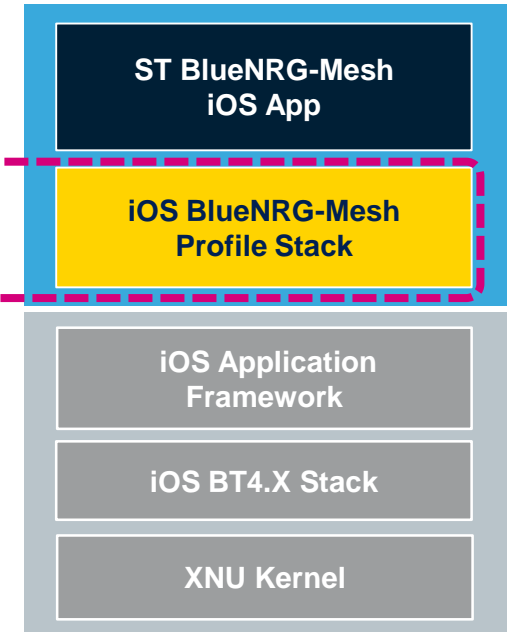
SDK SW Packages: STSW-BNRG-Mesh  
and X-CUBE-BLEMESH1



Android SDK



iOS SDK



# BlueNRG-Mesh

Bluetooth Mesh Basics

BlueNRG-Mesh SDK Solution

Getting started with BlueNRG-Mesh





# BlueNRG-MESH SDK

## Hardware Platform Support

25

### PLATFORMS



**STEVAL-IDB007V2**  
**BlueNRG-1**

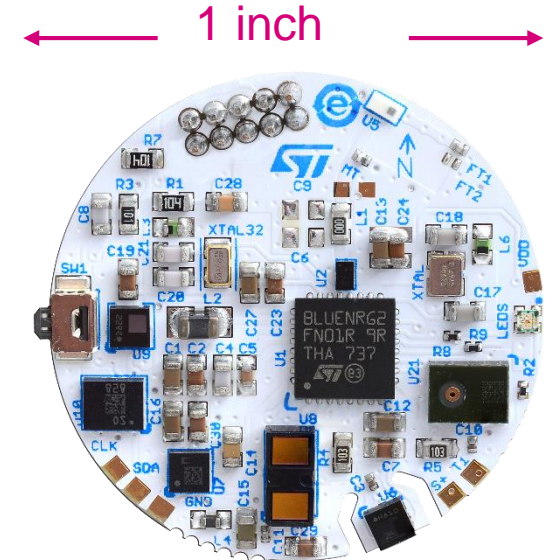
**STEVAL-IDB008V2**  
**STEVAL-IDB009V1**  
**BlueNRG-2**

**SW Package: STSW-BNRG-Mesh**



**Nucleo-L1/L4/F3/F4 + X-Nucleo-IDB05A1 + X-NUCLEO-IKS01A2**  
**BlueNRG-MS +**  
**STM32L1 or STM32L4 or STM32F3 or**  
**STM32F4**

**SW Packages: X-CUBE-BLEMESH1**  
**and FP-SNS-BLEMESH1**



**STEVAL-BCN002V1B**  
**aka «BlueTile»**  
**BlueNRG-2**  
**SW Package: STSW-BNRG-Mesh**

# BlueNRG-MESH SDK

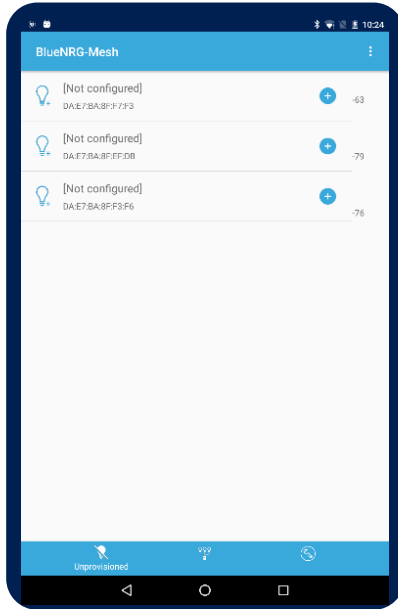
## Software Platform Support

26

### BlueNRG-Mesh App



**ST BLE Mesh app  
for Android and iOS**



### BlueNRG Platform SW

- **STSW-BNRG-Mesh**

- Mesh over Bluetooth Low Energy for **BlueNRG-1** and **BlueNRG-2**
- [https://www.st.com/content/st\\_com/en/products/embedded-software/wireless-connectivity-software/stsw-bnrg-mesh.html](https://www.st.com/content/st_com/en/products/embedded-software/wireless-connectivity-software/stsw-bnrg-mesh.html)

- **X-CUBE-BLEMESH1**

- Mesh over Bluetooth low energy software expansion for **STM32Cube**
- <https://www.st.com/en/embedded-software/x-cube-blemesh1.html>

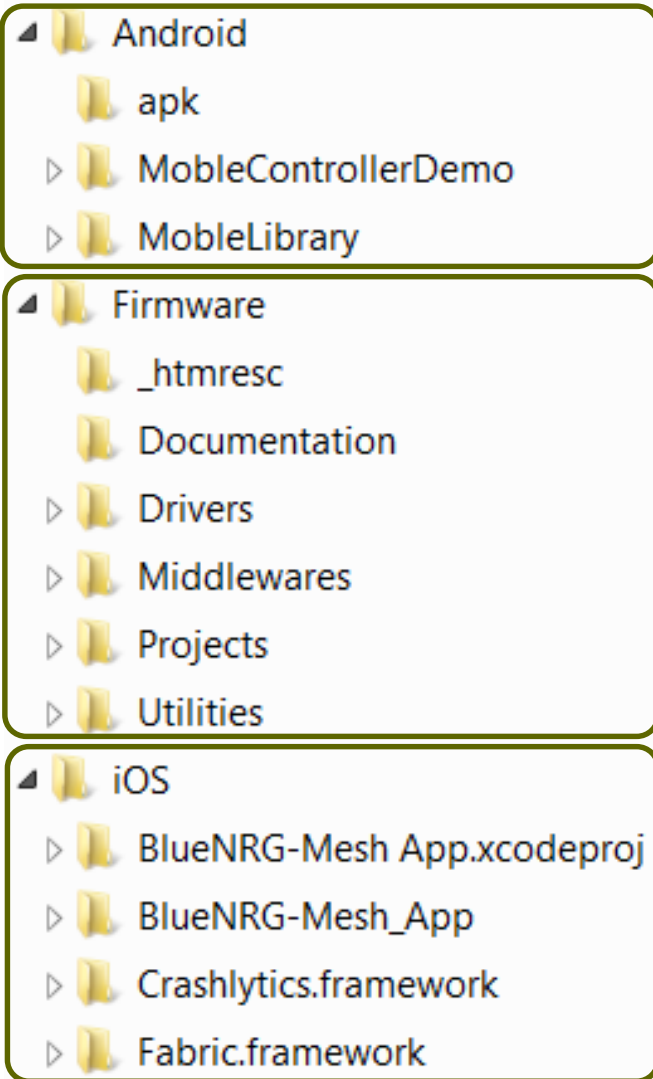
- **Application Example: FP-SNS-BLEMESH1**

- **STM32Cube** function pack for IoT node with BLE Mesh connectivity and **Sensor** model examples
- <https://www.st.com/en/embedded-software/fp-sns-blemesh1.html>



# Contents of STSW-BNRG-Mesh Package

27



- ST BLE Mesh Android SDK
  - **App Source Code for Android Studio**
  - Mesh Library
  - APK
- Firmware SDK
  - **Examples for BlueNRG-1, BlueNRG-2**
  - **Middleware** folder with **libraries** for **Mesh** and **BLE** communication
  - **Drivers** folder with **HAL** and **drivers** for supported **boards** and **components**
  - **Doxygen Documentation**
- ST BLE Mesh iOS SDK
  - **App Source Code for Xcode**
  - Mesh Library in Object code

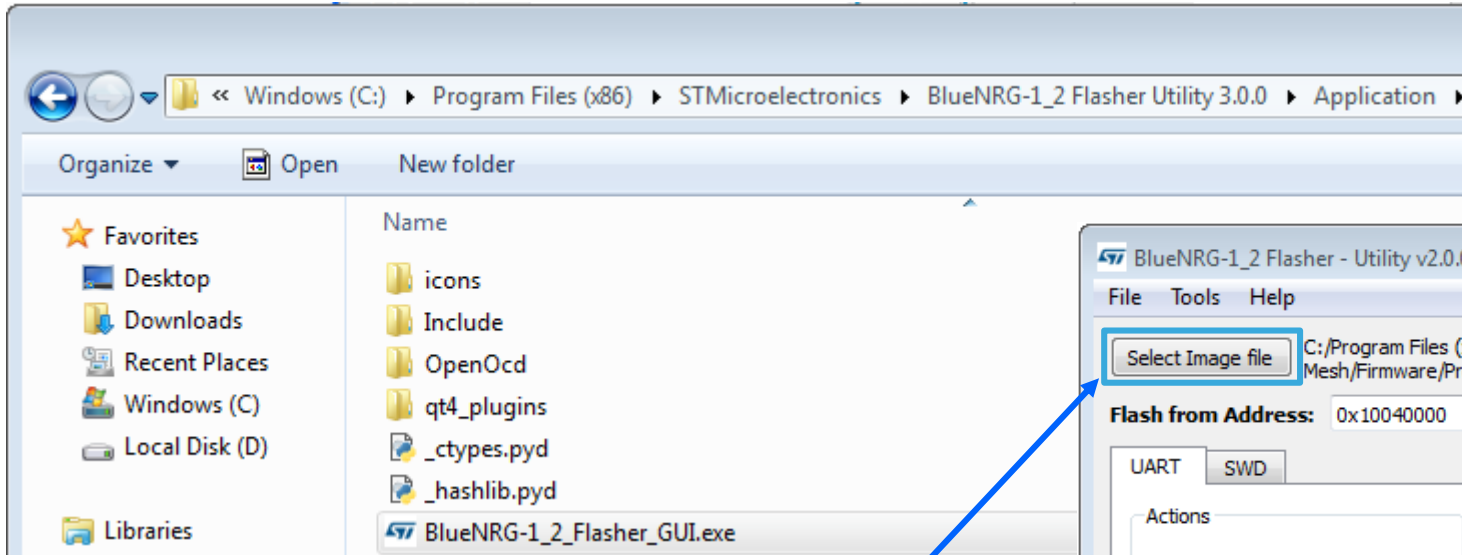


# STSW-BNRGFLASHER for BlueNRG-1/2

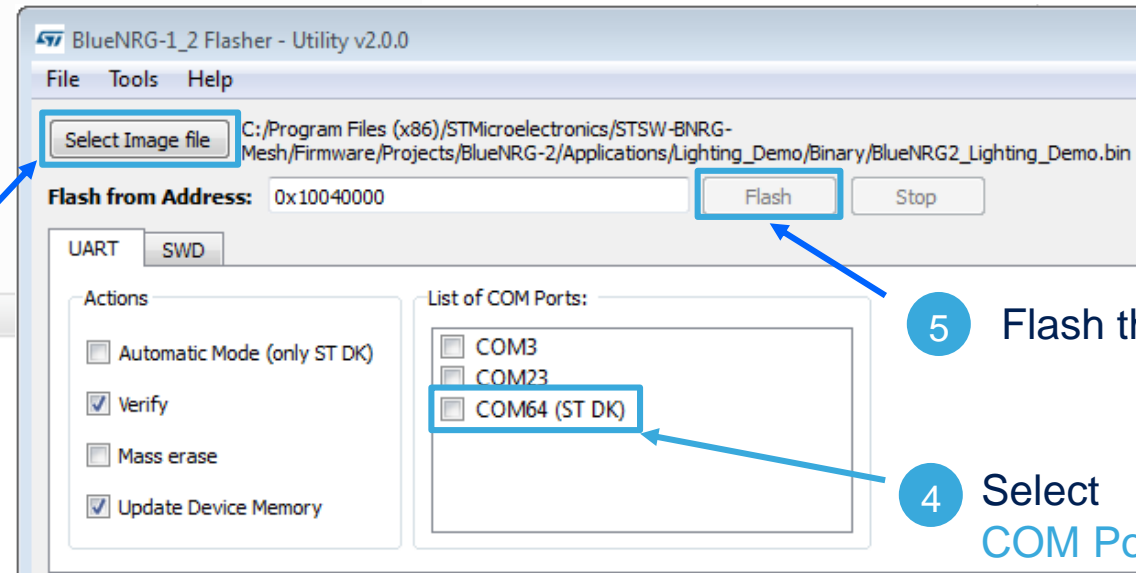
## Program in Just Few Seconds: Using Binaries

28

- 1 Open the Flasher Utility: Double click on **BlueNRG-1\_Flasher\_GUI.exe**



- 2 Click "Select Image file"



- 5 Flash the Board

- 4 Select COM Port

- 3 Go to `.\STSW-BNRG-Mesh\Firmware\Projects\BlueNRG-1\Applications\Lighting_Demo\Binary`  
`.\STSW-BNRG-Mesh\Firmware\Projects\BlueNRG-2\Applications\Lighting_Demo\Binary`  
And select the file `BlueNRG1_Lighting.bin` or `BlueNRG2_Lighting.bin`



# ST BLE Mesh: Provisioning of the Nodes

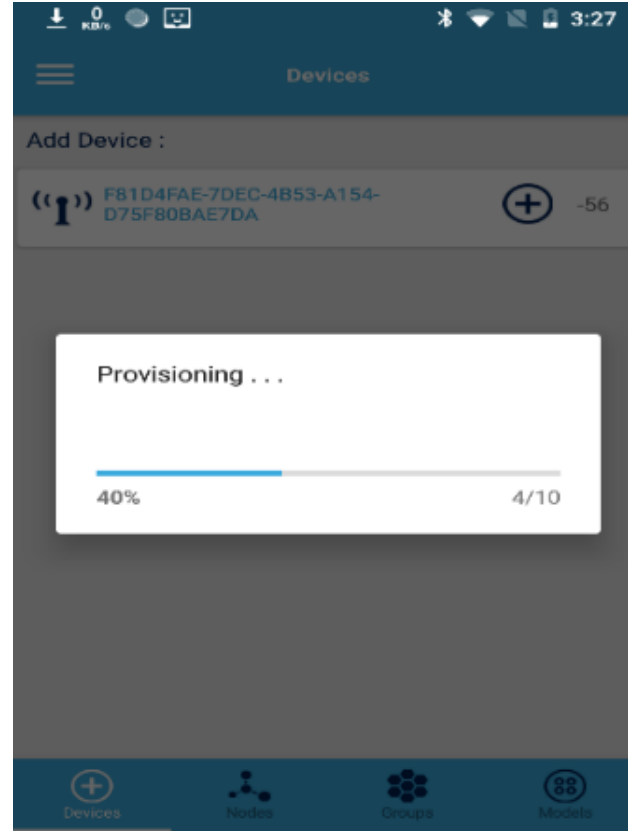
29

## Provisioning

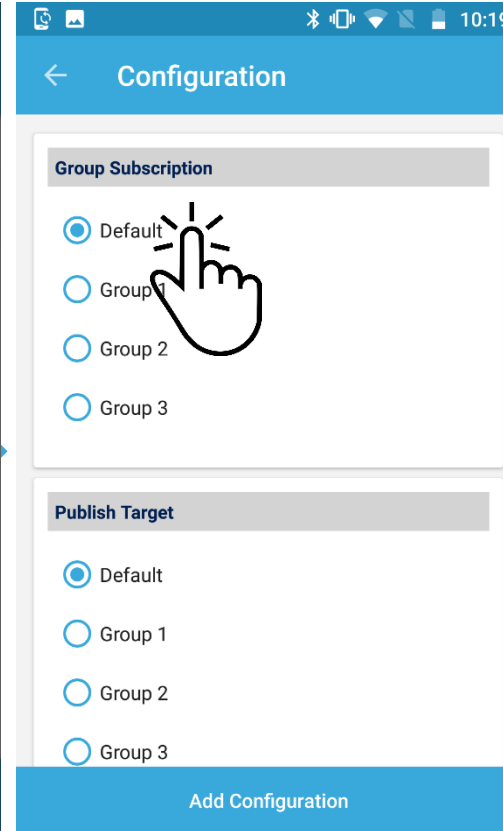


Before the provisioning the ones listed are called “**devices**”

## Progress bar

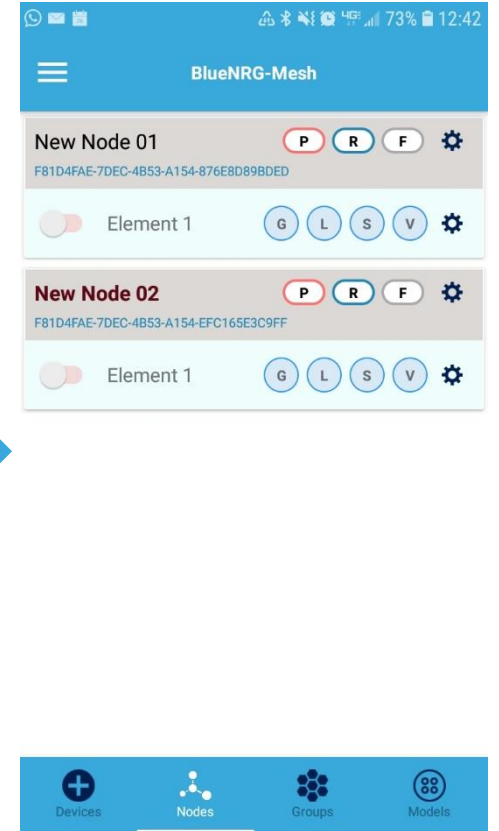


## Address Configuration



The Node can be added to a specific group of elements

## List of provisioned Nodes



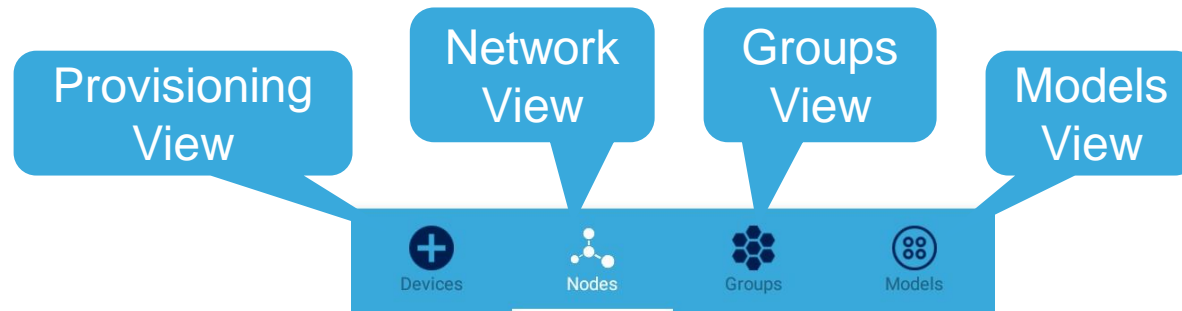


# ST BLE Mesh: Nodes List Tab Features

30

The screenshot shows the 'BlueNRG-Mesh' app interface. It lists two nodes: 'New Node 01' and 'New Node 02'. Each node has a set of role buttons (P, R, F) and a set of model buttons (G, L, S, V). Callouts highlight specific features:

- Element inside the node**: Points to the 'Element 1' toggle switch.
- Proxy, Relay and Friend roles are supported**: Points to the 'P', 'R', and 'F' role buttons.
- Proxy Node indicated with the name in bold**: Points to the 'New Node 02' title.
- Models Supported: Generic, Lighting, Sensor, Vendor**: Points to the 'G', 'L', 'S', and 'V' model buttons.

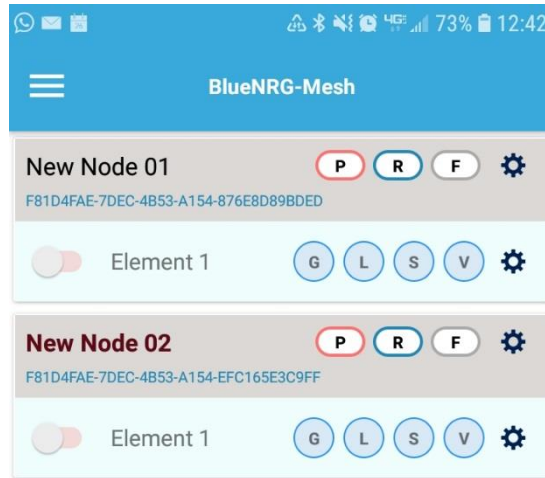




# ST BLE Mesh: Models Selection

31

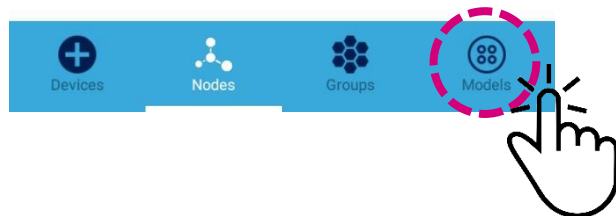
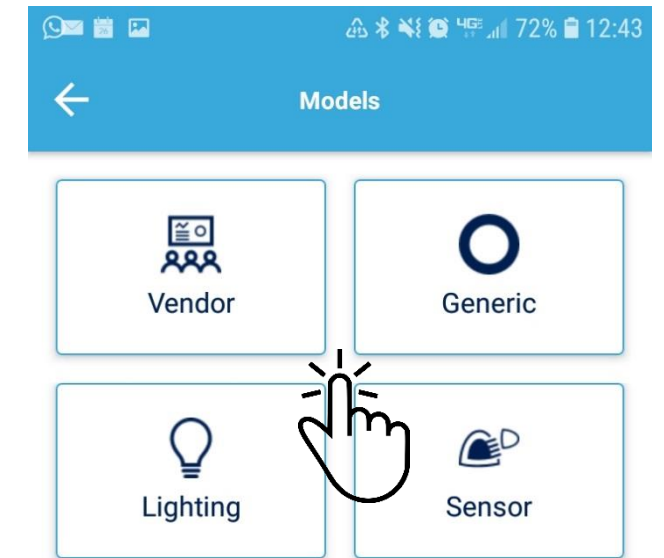
## Nodes List



## Models selection

- Different screen for each mesh model
- Send model specific commands
- Selected choice is saved

## Model Selection



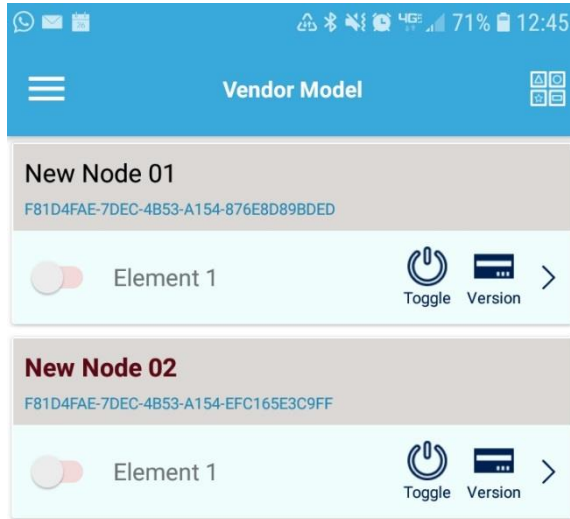


# ST BLE Mesh: Models Selection

32



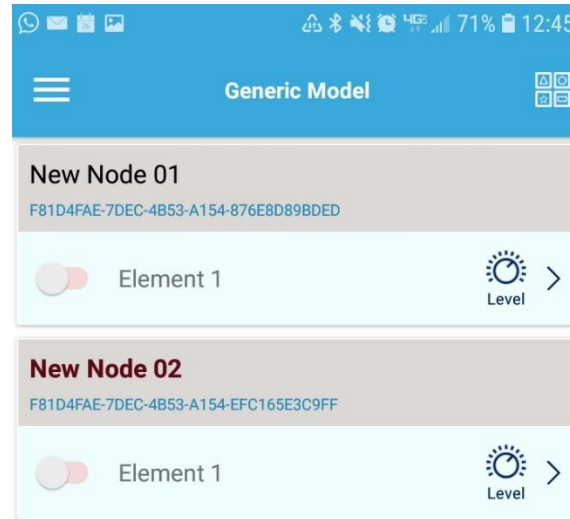
## Vendor Model



- Supported Commands
  - Vendor on/off
  - Version query
- \* **Vendor model allows for transmitting custom payload.**



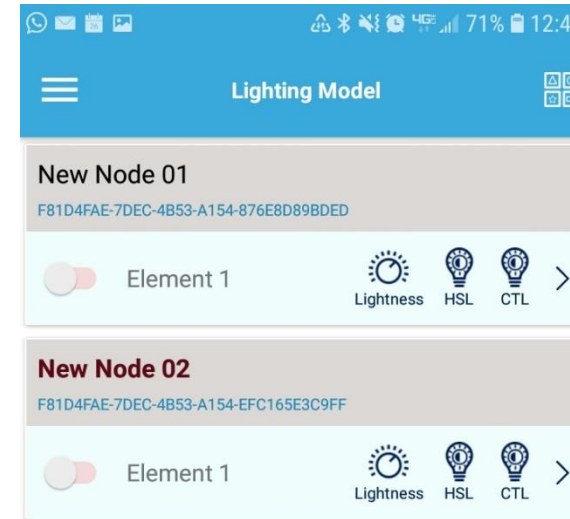
## Generic Model



- Supported Commands
  - **Generic on/off**: for device supporting On/Off feature
  - **Generic level**: Manage state of an element in a **16-bit signed integer**



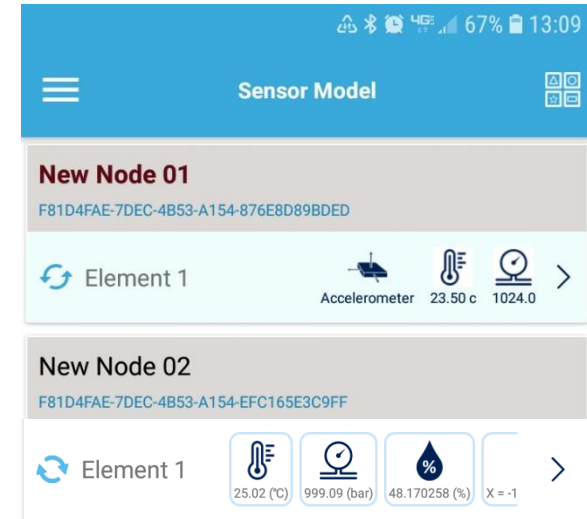
## Lighting Model



- Supported Commands
  - **Light lightness**
  - **Hue, Saturation, Lightness**
  - **CTL (Color temperature), Lightness**



## Sensor Model



- Supported Commands
  - **Sensor Get**: i.e. Acceleration, MAG & Gyro, Temperature, Humidity, Pressure
  - **Sensor Cadence Set**



Devices



Nodes



Groups



Models



Devices



Nodes



Groups



Models



Devices



Nodes



Groups



Models



Devices



Nodes



Groups



Models



# Network Configuration Database

33

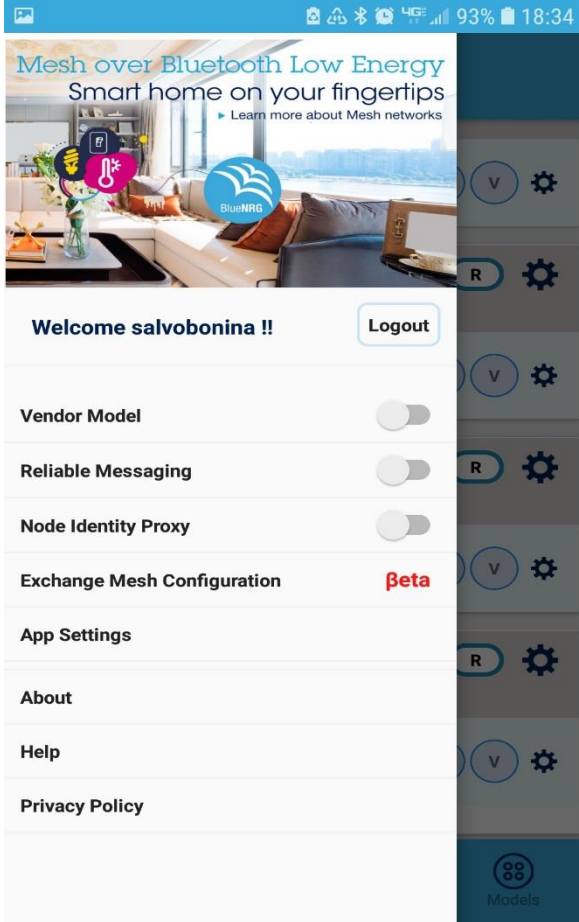
- **Stores security info & parameters** to operate in a mesh network
- Shared in **JSON** format.
- Allows another smartphone to manage and control the mesh network.
- Two ways to share
  - **Via email** – Instantly share network with another smart phone
  - **Via Cloud** – Our mesh cloud web app allows sharing between multiple phones.
- **Interoperable** on both iOS and Android **ST BLE Mesh Apps**



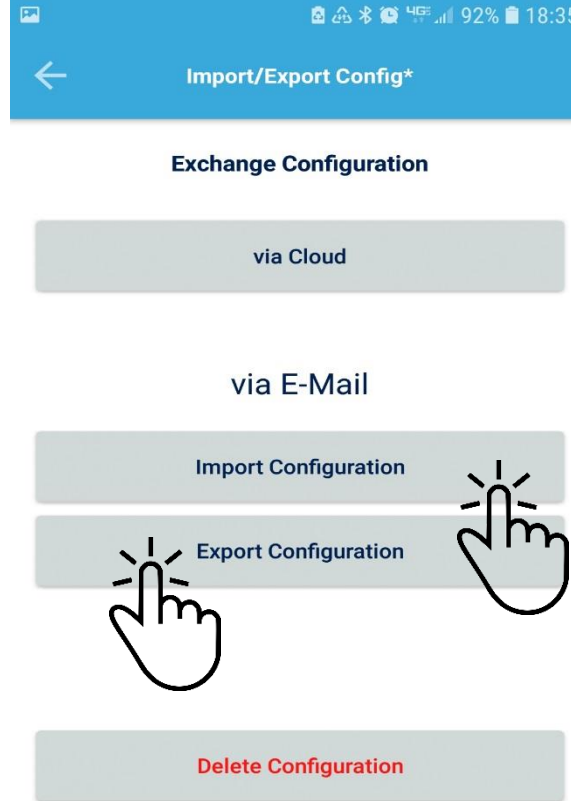
# Exchange Provisioning DB Configuration

34

## Settings



## Import/Export Configuration



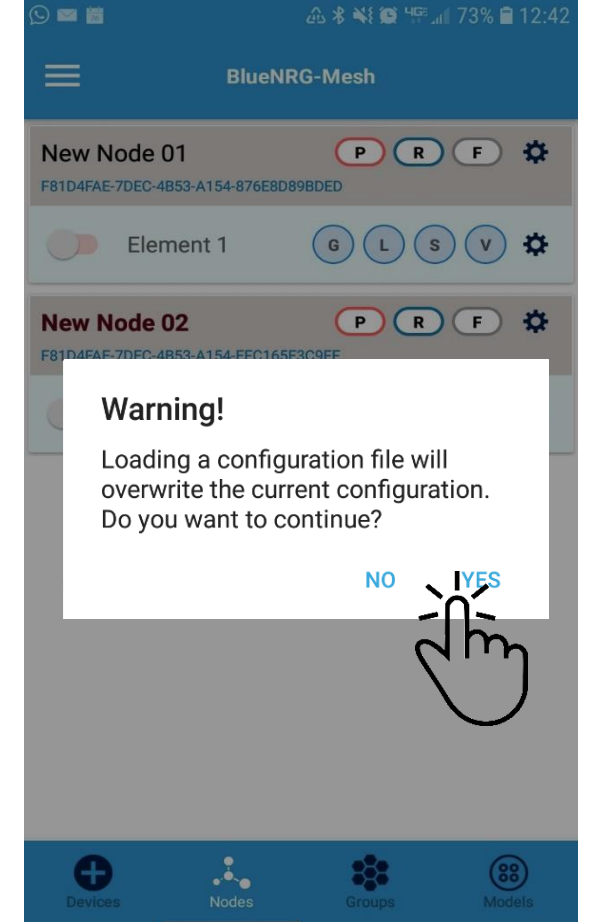
Next Panel  
➔

Import Configuration  
➔

**Export Configuration on one Smartphone.  
Import Configuration on another Smartphone.**

To import on iPhone/iPad, save the email attachment with "Save to Files" → On My iPhone/iPad → "BlueNRG\_Mesh"

## Replace current configuration

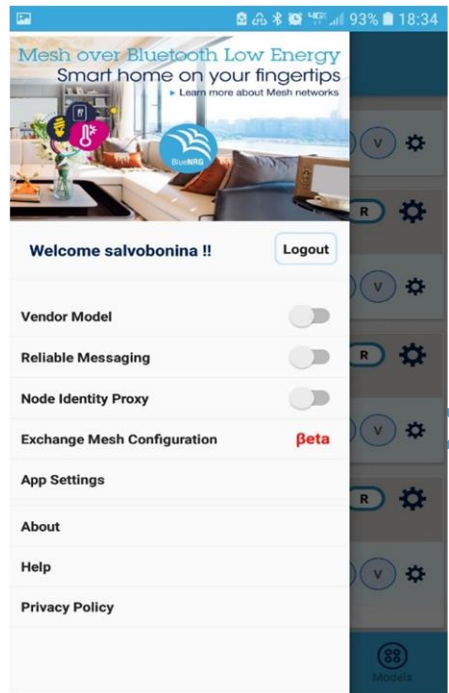




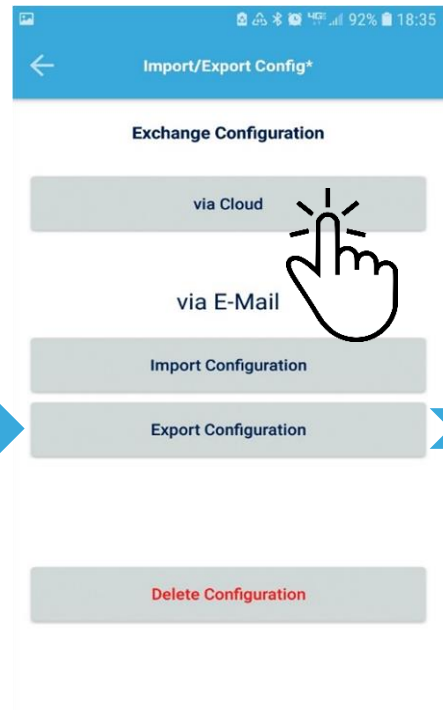
# Cloud Synchronization

35

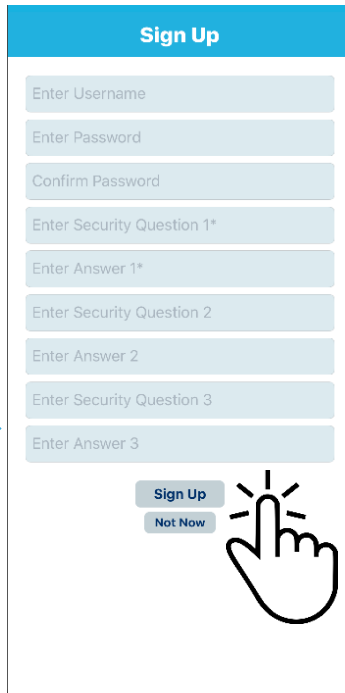
## Settings



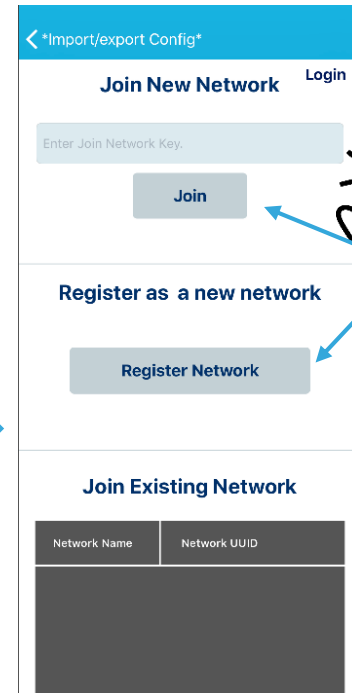
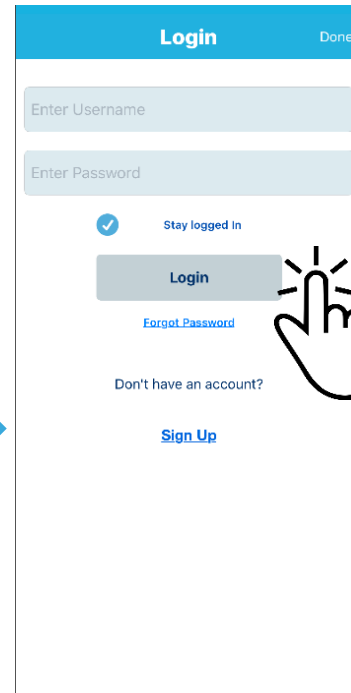
## via Cloud



## Sign Up



## Login



## Cloud Network scenarios

- Register a New Network
- (Re) Join an Existing Network (pre-registered by you)
- Join a New Network (by invitation)

# STSW-BNRRG-Mesh Documentation

36

## AN5285 - STSW-BNRRG-Mesh Friend and Low Power features (NEW!)

- How to enable low power operations, in agreement with Bluetooth Mesh Profile v1.0 and Friendship concept

## UM2290 - BlueNRG Mesh Android API guide for Mesh over Bluetooth low energy

## UM2417 - BlueNRG Mesh iOS API guide for Mesh over Bluetooth low energy



- Start developing Bluetooth Low Energy Mesh applications on Android devices
- Network creation
- List un-provisioned nodes
- Provision a new node
- Send unicast/broadcast commands
- Save/Restore network configuration
- Register/Un-register callbacks

## UM2180 - Getting started with the ST BLE Mesh Android application (\*)

## UM2361 - Getting started with the ST BLE Mesh iOS application (\*)

(\*) Differences due to available control and libraries to design the App Unified User Interface available since v1.05.000

- Provision, un-provision, and control nodes in Bluetooth Low Energy Mesh network

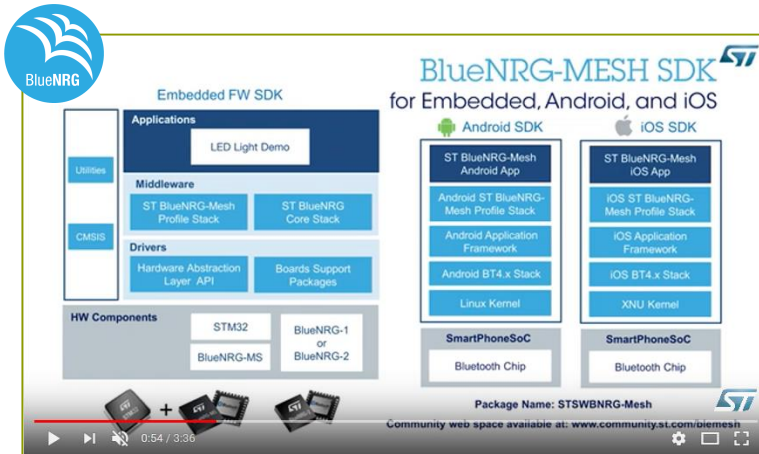
## UM2295 - Getting started with the ST BlueNRG-Mesh embedded firmware



- Start developing Bluetooth Low Energy Mesh applications on available platforms
- Core functionality for secure provisioning and communication
- Smart Light sample application

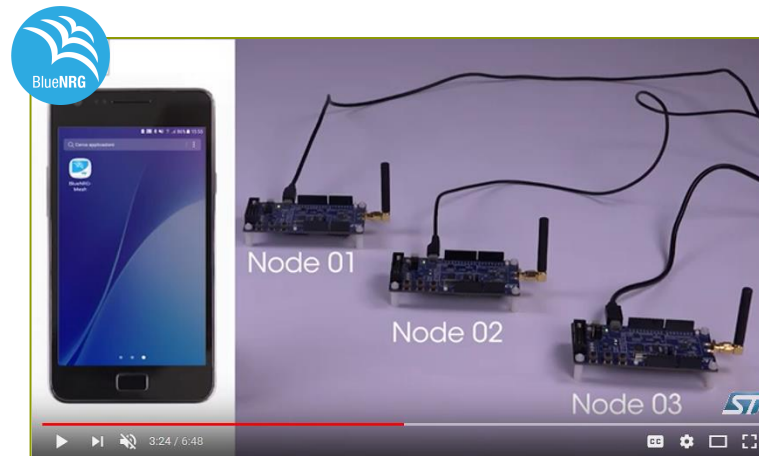
# BlueNRG-Mesh SDKs – Videos

37



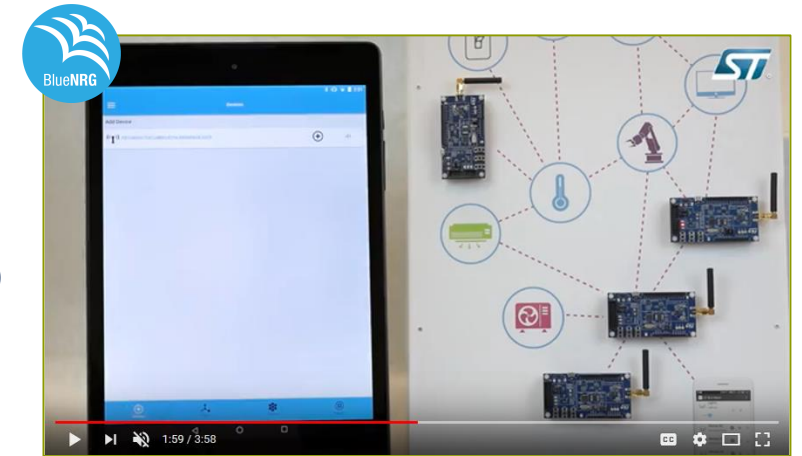
## Introduction to BlueNRG-MESH

<http://www.youtube.com/watch?v=NMeHUAiaPxl>



## Getting started with BlueNRG-Mesh

<http://www.youtube.com/watch?v=MV5M5AHMuU0>



## BlueNRG-MESH: Provisioning & Transfer

[www.youtube.com/watch?v=8ocXQmQa-dA](http://www.youtube.com/watch?v=8ocXQmQa-dA)



# BlueNRG-Mesh Additional Resources

38



- **Bluetooth SIG website:** Bluetooth Mesh deep dive from the official
  - [Bluetooth Mesh specification](#)
  - [Bluetooth Mesh overview](#)
  - [Bluetooth Mesh FAQ](#)
- **BlueNRG-Mesh SDK** available (firmware, mobile app for Android, iOS):
  - [www.st.com/blemesh](http://www.st.com/blemesh)
- **BlueNRG-mesh community forum**
  - <http://community.st.com/blemesh>
- **ST BLE Mesh Android application** available on **Google Play Store**:
  - <https://play.google.com/store/apps/details?id=com.st.bluenrgmesh&hl=en>
- **ST BLE Mesh iOS application** available on the **Apple iTunes App Store**:
  - <https://itunes.apple.com/us/app/bluenrg-mesh/id1348645067?mt=8>

# Conclusion: BlueNRG-Mesh

39

- A SIG certified stack running on ST's BlueNRG SoC series
- Multiple evaluation platforms: BlueNRG evalkit, Nucleo kit and BlueNRG-Tile
- Easy-to-use SDK, available for Embedded, Android and iOS
- Customers already shipping ST BlueNRG-2 with BlueNRG-Mesh in high volume!



life.augmented

