

# Bluetooth Mesh under the Microscope: How much ICN is Inside?

Hauke Petersen  
Freie Universität Berlin  
hauke.petersen@fu-berlin.de

Peter Kietzmann  
HAW Hamburg  
peter.kietzmann@haw-hamburg.de

Cenk Gündoğan  
HAW Hamburg  
cenk.guendogan@haw-hamburg.de

Thomas C. Schmidt  
HAW Hamburg  
t.schmidt@haw-hamburg.de

Matthias Wählisch  
Freie Universität Berlin  
m.waehlich@fu-berlin.de

## ABSTRACT

Bluetooth (BT) mesh is a new mode of BT operation for low-energy devices that offers group-based publish-subscribe as a network service with additional caching capabilities. These features resemble concepts of information-centric networking (ICN), and the analogy to ICN has been repeatedly drawn in the BT community. In this paper, we compare BT mesh with ICN both conceptually and in real-world experiments. We contrast both architectures and their design decisions in detail. Experiments are performed on an IoT testbed using NDN/CCNx and BT mesh on constrained RIOT nodes. Our findings indicate significant differences both in concepts and in real-world performance. Supported by new insights, we identify synergies and sketch a design of a BT-ICN that benefits from both worlds.

## CCS CONCEPTS

• **Networks** → **Network layer protocols**; *Naming and addressing*; *Network experimentation*;

## KEYWORDS

IoT, ICN, Bluetooth, Constrained devices

### ACM Reference Format:

Hauke Petersen, Peter Kietzmann, Cenk Gündoğan, Thomas C. Schmidt, and Matthias Wählisch. 2019. Bluetooth Mesh under the Microscope: How much ICN is Inside?. In *ICN '19: Conference on Information-Centric Networking, September 24–26, 2019, Macao, China*. ACM, New York, NY, USA, 7 pages. <https://doi.org/10.1145/3357150.3357398>

## 1 INTRODUCTION

*The Bluetooth mesh standard adopted last year is a wireless technology based on the ICN principles. [...] It implements all of the major paradigms of information-centric networking in order to enable simplicity, scalability and reliability [...]. — [32]*

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

*ICN '19, September 24–26, 2019, Macao, China*  
© 2019 Association for Computing Machinery.  
ACM ISBN 978-1-4503-6970-1/19/09...\$15.00  
<https://doi.org/10.1145/3357150.3357398>

The above claim triggered the investigations of the present paper. We are curious how information-centric networking (ICN) features re-appear in Bluetooth (BT) mesh, a new flavor of Bluetooth Low Energy (BLE) that was designed for intelligent home installations such as smart lighting, and how they compare to the ICN instances NDN/CCNx [21, 39].

In recent years, BLE has seen a remarkable adoption for the sake of introducing cheap and popular personal area networking (PAN) to the constrained embedded world. BLE promises easy and robust networking between sensors or actuators and mobile devices such as smartphones, tablets, notebooks. BLE offers two modes of operation: a connection-oriented point-to-point mode and a connection-less advertising mode. BT mesh is a new flavor of BLE communication, based on the latter. It supports multi-hop networking, network layer caching, and a group-based publish-subscribe, which resembles ICN features—but is BT mesh really built according to ICN principles?

We investigate this question from two perspectives. First, we explore and discuss BT networking functions in comparison with ICN principles and their concrete implementation in NDN/CCNx. Second, we empirically contrast the technologies by performing common ICN and BLE experiments on RIOT [4, 5] in a testbed deploying CCN-lite and the NimBLE stack.

Our findings from basic experiments measuring standard metrics indicate that in simple scenarios BT mesh performs comparably to NDN at the price of consuming significantly higher network resources. In more complex multi-hop settings, NDN easily outperforms BT mesh due to its more efficient packet transport and more widely applicable in-network caching. These insights obtained from conceptual review and experiments lead us to outline a future approach of how BT and ICN technologies could converge to a truly information-centric low-power wireless PAN (LoWPAN) system.

The remainder of this paper is structured as follows. We first dive into the BT mesh technology in Section 2, followed by a detailed theoretical comparison of ICN and NDN concepts in Section 3. Experimental evaluations are discussed in Section 4. Section 5 presents our technological vision on future ICN networking based on Bluetooth. Related work on BT and NDN performance is supplied in Section 6.

## 2 BLUETOOTH MESH

BLE [16] focuses on robust wireless communication in the open 2.4 GHz band and is widely used in IoT scenarios due to its omnipresence on mobiles. BT mesh [17] is standardized on top of

BLE, offering management of multi-hop topologies and enabling many-to-one and many-to-many group communication.

The BT mesh standard specifies a complete networking system, starting from the mapping to BLE as transport over network configurations including functional roles of participants and the provisioning of security credentials, up to application layer formats. In the following, we focus on aspects of BT mesh that are most relevant when comparing with ICN. For this, we do not further cover everything related to node provisioning or the proxy role.

## 2.1 Basic communication primitives

**Lower layers.** A Bluetooth mesh node always connects to three broadcast channels for advertisement. Data is transmitted consecutively to all of these channels in a defined advertisement interval. Receivers constantly scan for packets on all three channels alternately, leading to a 100 % duty-cycle. This sacrifices energy savings attained by BLE.

**Network layer.** The network layer defines four types of addresses: unassigned, unicast, group, and virtual. Group addresses are fixed and map to different roles of a node in the mesh network, *i.e.*, *all-relays* and *all nodes*. Virtual addresses act as configurable multicast group addresses, which are mainly set by application models during network provisioning time, but it is also possible to define virtual addresses at runtime using distributed control methods.

**Application layer.** Addressing data for applications is organized by a *model layer* that specifies different application scenarios and their interfaces, which includes a set of valid messages and states. The underlying *access layer* defines the format of application data.

**Multi-hop forwarding.** To span multi-hop topologies, BT mesh nodes use a *relay feature* which floods packets to omit routing complexity. This mechanism is known as *managed flooding* since it includes methods for avoiding duplicates and loops. Its two main concepts are a hop limit counter per packet as well as a network message cache.

The *network message cache* includes a list of messages recently seen by a node. If a message is already known, it is ignored and not forwarded. This cache is mandatory for all nodes to prevent redundant forwarding and thereby to reduce network load on the mesh. The Bluetooth mesh standard does not require to cache the whole packet payload and leaves this open to the specific implementation. Hence, content caching is not originally supported. We will explore the differences to caching in ICN in the following sections.

## 2.2 Data networking

Data exchange is organized using a publish-subscribe approach. Data is announced via a publish broadcast message to any of the addresses mentioned above. A publish can either be an unsolicited message initiated by any node in the network or it is sent as a reply to a previously received message. Replies are sent back to the BT source address of the requester. It should be noted that these unicast messages are mapped to broadcast advertisements on the lower layer. Reception of messages is done by subscribing to addresses. *Models* specify the basic functionality of mesh nodes. Thereby, a node can act as client, server, or control node that implements both functionalities and adds the intermediate control logic.

Like the entire BT family, BT mesh is designed for local, isolated networks as there is no way to transmit packets via transparent gateways to the Internet. Wide area communication can only be reached with the help of application layer gateways that translate the BT mesh communication Models into corresponding application protocols.

## 2.3 Support for constrained environments

Constrained nodes usually run on batteries, which quickly drain by the continuous channel scanning of Bluetooth mesh. To foster sleep, Bluetooth mesh introduces so-called *Friend nodes* that act as proxies for the constrained devices. Any constrained node is assigned to exactly one friend, but a friend may cover multiple constrained nodes.

After establishing this relation, the more powerful node maintains a message queue for each partner. Whenever the low-power node wakes up, it requests cached packets from its friend node.

## 2.4 Security

All communication is secured by authenticating and encrypting every packet using pre-shared, symmetric keys. In a packet, all data above the network layer is encrypted using an *application key*, which is shared among nodes belonging to a defined application domain (*i.e.*, all lights and switches on a building floor). The complete packet, including the already encrypted application payload, is further encrypted using a *network key*. Network keys are shared among all nodes in a subnet, allowing them to authenticate and forward packets without disclosing any application data.

All keys are initially assigned during node provisioning and can be updated during runtime using specified update procedures defined by the BT mesh standard.

To prevent a simple passive eavesdropper from tracking nodes, BT mesh applies obfuscation techniques to key packet identifiers such as source addresses and sequence numbers. In addition, packets are equipped with monotonic sequence numbers to protect against replay attacks, as well as an *IV Index*. This is a shared number between network members, which acts as entropy source for nonce generation and can be updated in case of state compromise. Updating the index triggers a predefined procedure of distributed control.

# 3 BLUETOOTH MESH VS ICN

## 3.1 Stack overview

BT mesh specifies a full vertical stack, where the network layer is only one part. In contrast, ICN defines a network architecture relying on supporting protocols on the layers above and below for successful deployment. Figure 1 compares overall views of the BT mesh design with an ICN stack as we propose for constrained IoT networks.

Both stacks can utilize the same link technology, but BT mesh only defines bearers that are tied to BLE radios where ICN is not restricted to those. For multi-hop communication BT mesh is further restricted to the broadcast-based advertising bearer. The ICN stack may use a face to link mapping selecting the outgoing unicast channel (*i.e.*, Layer-2 unicast destination address) according to the network-layer routing or forwarding decisions.

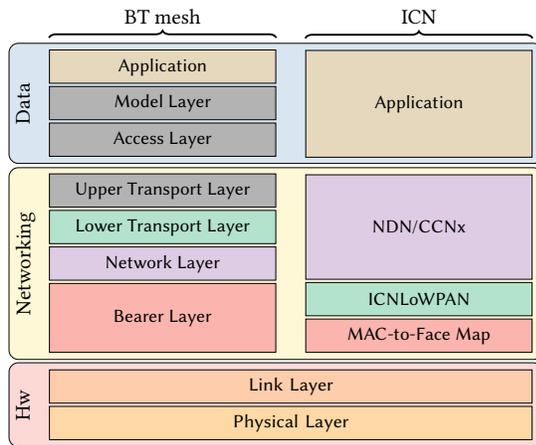


Figure 1: Bluetooth mesh stack (left) vs our ICN stack (right).

On top, the ICNLoWPAN [20] convergence layer implements segmentation and compression to adapt and reduce data to be sent over low-power links. In contrast, BT mesh does not compress any headers and implements segmentation on top of the network layer in the transport layers, since its link properties are predefined and no flexible adaptation is needed. Unlike NDN, networking in BT mesh defines a stacked transport layer that covers segmentation, reassembly, but also control traffic.

The ties to BLE and the vertical integration make BT mesh a locally confined, monolithic technology, whereas ICN is open w.r.t. various link layers, applications, and wide-area deployment.

### 3.2 Networking

**Routing and forwarding.** BT mesh only supports a single network interface. Furthermore, it does not support multiple faces over this interface as the used link layer does not support destination addresses. Multi-hop and multi-path topologies are built without routing by simply forwarding all packets over the broadcast channel of this interface. Although flooding follows a managed approach, it leads to substantial resource consumption for all nodes in reach. In contrast, NDN guides Interest packets according to a Forwarding Information Base (FIB), which can be populated by a variety of routing protocols. In addition, multiple requests (solicited publishes in BT) to the same resource can be aggregated in the Pending Interest Table (PIT) and drastically reduce the number of packet traversals in a multi-hop environment [18, 29]. On the other hand, BT mesh natively supports group requests, as all responses are returned to the initiator. Data aggregation in ICN is not yet visible as a standard service. A proposal for multi-source data retrieval in ICN has been published in [3].

**Reliable transport.** BT mesh supports transmission of unacknowledged and acknowledged messages, where the latter requires each subscriber to send a response. The message type in use is defined by the application *model* and can be adjusted during runtime via control models. Publish nodes are allowed to retransmit messages until a new message of the same model is published, which stops ongoing retransmissions. In contrast, NDN implements a best-effort

hop traversal by retransmitting pending Interests on time triggers until data returns, a NACK is received, or PIT entries time out.

The BT mesh standard advises against the use of acknowledged messages in multicast scenarios, as the number of expected responses is unknown. This burdens application logic with implementing acknowledgements and timeout. The flooding nature of BT mesh plus its retransmission capabilities can lead to high resource demands. The effects and a collection of counter-measures were presented in [13, 14, 33].

**Medium Access Control (MAC) layer adaptation.** Although BLE defines reliable point-to-point links that involve time division and frequency hopping, BT mesh simply broadcasts on the advertising channels, which discards the original reliability features of BLE. NDN, on the other hand, does not natively define a mapping of faces to the MAC layer. In typical deployments however, faces are often mapped directly to the broadcast address similarly, or to the unicast MAC address. While broadcasting enables path diversity with no additional routing overhead, it increases resource consumption, whereas a mapping of faces to unicast traffic saves resources but comes to the cost of neighbor management. These effects have been analyzed in [22].

**Application layer adaptation.** In comparison to BT mesh that specifies a data *model* for different device types, such as light sensors and dimmers, the main requirement to forwarding data to the application layer in NDN is a matching PIT entry. Respectively, there is no strict requirement on the format of names. This shifts complexity to the namespace but on the other hand, flexibly widens the field of possible applications. Introducing a new traffic type in BT mesh requires implementation of a new model. By restricting applications to models, BT mesh reduces its areas of use.

Regarding its **overall networking capabilities**, BT mesh starts from utmost flexibility on the lower layers by broadcasting. Up the stack it narrows down by restricting networking functions and—even further—the application interfaces. NDN follows an opposite approach. It reduces lower layers to pipes behind faces that were initially understood to interface point-to-point links, but remains open to various routing and forwarding strategies and leaves applications with a simple, yet fully flexible name-based network access.

### 3.3 Names and publish-subscribe

BT mesh uses (group) addresses to label data streams. NDN acts upon hierarchical tokenized names assigned to data entities. The semantic representation in NDN gives several degrees of freedom in terms of network configuration and application design whereas the addressing scheme in BT mesh is mostly static and normally configured by manual provisioning of a node before deployment. In both cases, however, data is addressed and not an endpoint, which makes BT mesh content-centric like NDN.

Publish-subscribe services require a rendezvous function to bring together publishers with subscribers, which are decoupled otherwise. The common way to implement rendezvous is a well-known broker that mediates between peers. Some ICN flavors such as PSIRP/Pursuit and related [10, 23] natively follow such a publish-subscribe paradigm [38] by foreseeing a broker function as part of the network service architecture. NDN and CCNx are based on

a request-response mechanism without a push option for data. In contrast, publish is commonly performed as a push operation and subscription as a state that awaits it, which led to workarounds and discussions in previous work [2, 11, 19]. Nonetheless, support publish-subscribe operations can be realized without protocol modifications using NDN primitives [19, 25, 27, 31]. Here, rendezvous is provided by names or prefixes, which are matched by some content repository or cache that is well known to the routing system.

In BT mesh all content is distributed via publish-subscribe. The approach in BT mesh does not employ a broker, but advertises content items to group addresses. Group advertisements are filtered by the upper layers so that only subscribers of a specific group receive them. Conversely, publishing in BT mesh can be a solicited operation, which then resembles the request-response scheme of ICN. If a publish hits a subscriber, it can trigger a publish being sent back to the initiator. This does not prevent broadcasting of publish requests on the link layer, though.

By relying on broadcasts, BT mesh remains independent of any infrastructure entity such as a service broker. This feature, however, is achieved by significantly straining the network. The latter imposes severe scaling limits, as we will demonstrate in our experiments.

### 3.4 Low power assistance and caching

The low-power friendship relation is the feature of BT to integrate very constrained, battery driven nodes. As a default, this role is not defined in ICN, but there exists work on investigating this topic [37]. Caching principles are present in both stacks.

In BT mesh, friend nodes allocate caches for each constrained node that has established a relation before going to sleep. When a constrained node awakes, it notifies its friend which then sends all cached packets. On relay nodes, different caches are responsible for identifying previously seen packets. They act as a forwarding filters for managed flooding and do not serve data—in contrast to the ICN approach of distributed in-network caching.

It is noteworthy that BT mesh nodes can always only have one cache available at its friend. This cache drops entries after their first retrieval, which counters a re-use of cached data. In contrast, NDN can cache any data packet at any node for access throughout the network and the cache replacement strategy is open to implementers and deployment.

## 4 EVALUATIONS

### 4.1 Experiment Setup

**Hardware Platform.** We deploy our experiments on common low-end IoT hardware based on ARM Cortex-M providing 64 kB RAM and 512 kB flash. For the BT mesh measurements, we use nRF52dk development kits from Nordic Semiconductor featuring BLE-compliant Cortex-M4F SoC (nRF52832) running at 64 MHz. Our NDN measurements run on iotlab-m3 nodes featuring a Cortex-M3 SoC (STM32F1) at 72 MHz. They carry an 802.15.4 radio (Atmel AT86RF231) that provides basic MAC features such as address filtering in hardware. All nodes are part of the Saclay site of the FIT IoT-LAB testbed.

We use different hardware platforms to approach real deployment as close as possible. The differences between both CPUs (instruction set, core clock frequency) are neglectable and do not have a noticeable impact on our measurements. The radios on the NDN and BT mesh platforms, however, differ in some aspects. The Nordic radio in BT mesh mode uses the default BLE bit rate of 1 Mbit/s and does not support carrier sense multiple access (CSMA) and automatic repeat request (ARQ). On the other hand, the Atmel radio runs at 250 kbit/s and features CSMA and ARQ.

To check for consistency, we performed all measurements related to NDN on both platforms. The traffic load results are equal. The content arrival time measurements show similar success rates but vary in the time to completion, which roughly corresponds to the differences in radio bit rates.

We argue that the CSMA and ARQ support of the iotlab-m3 platform comes much closer to a real-world production deployment than the proprietary mode of the Nordic radio. For this reason, all NDN measurements presented in the following have been done on the iotlab-m3 boards.

**Software Setup.** Our experiments are based on the RIOT [4] operating system version 2019.04. To analyse BT mesh, we use the NimBLE stack.<sup>1</sup> Our NDN measurements are based on CCN-lite [36]. Both network stacks are available as third-party packages in RIOT. In the Saclay testbed all nodes are in wireless range of each other. To employ multi-hop topologies, we apply MAC address filtering on the nodes. Although this enforces a virtual topology on the network layer, it does not prevent two nodes, that topologically do not see each other, to cause interference on the physical layer.

**Configuration.** We align parameters between experiments where possible to follow a consistent setup. On the link layer, we used the default parameters given by NimBLE and adapted CCN-lite to apply comparable behavior. BT mesh always transmits every packet 5 times per channel (5 advertising events) with an interval of 20 ms. For IEEE 802.15.4 we enable link layer acknowledgments with a maximum of 4 retransmissions. Additionally, we run CCN-lite with 4 network layer Interest retransmissions, a retry interval of 1 second, and a timeout value of 10 seconds. The content store (CS) is limited to 30 entries.

**Scenarios.** We deploy (i) 10 nodes in a full mesh to analyze single-hop scenarios, and (ii) the same nodes along a line to evaluate multi-hop use cases. The ICN stack configures FIBs with faces that map to unicast MAC addresses of their neighbors. The BT mesh stack uses a group address on the network layer but will broadcast on the link layer. For each topology, we configure two different traffic patterns:

**Many to one** One node acts as data sink (consumer) and all other nodes are data sources (producers). In ICN, the consumer requests data from all producers iteratively, which reflects common ICN scenarios. In BT mesh, every producer node publishes data to a group address to which the sink is subscribed. In both scenarios, each producer publishes 100 content items with a publishing rate of one item every  $5 \text{ s} \pm 2.5 \text{ s}$  of jitter. This publishing rate is limited by the publisher node's system resources.

<sup>1</sup><https://github.com/apache/mynewt-nimble>

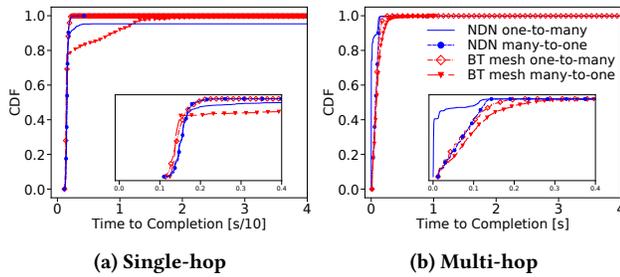


Figure 2: Time to content arrival.

**One to many** One node acts as producer and all other nodes are data consumers. In ICN, all consumers request data ‘simultaneously’. In BT mesh, the producer publishes data to a group address. In both cases, 100 content items are published by a producer with a publishing rate of one item every  $1\text{ s} \pm 0.5\text{ s}$  of jitter.

## 4.2 Results

The average content delivery rate for each measurement setup is 100%. Network performance, however, is more subtle.

**Content Arrival Time (Single-hop).** We first measure the time to content arrival for each data packet sent from a publisher to a consumer. Figure 2 shows the cumulative frequency of all events for all configurations.

In the single-hop scenario (cf., Figure 2a), NDN and BT mesh exhibit almost the same behavior for the many-to-one and one-to-many scenario respectively. Nearly 80% of the contents arrive during the first 15 ms. Even though NDN sends two packets per content (Interest and data) differences compared to BT mesh are negligible. The data delivery time is only limited by the local link layer transmission delay because no competing packets occur.

In the one-to-many case of NDN, the results change slightly. Most noticeable is that the content arrival time exceeds 20 ms for 5% of the published content. In fact, very few content items need up to 1 s (not shown). In this scenario multiple consumers request content independently and in parallel, which results in some packet loss of Interest messages that are (successfully) retransmitted after 1 s.

In the many-to-one scenario of BT mesh,  $\approx 80\%$  of the content items arrive also within the first 15 ms. The remaining items need up to 120 ms to be successfully delivered. The reason for this relies in the broadcast nature of BT mesh and the simple reliability scheme which requires uninformed retransmissions. In our case every node sends each content item five times with a delay of 20 ms, explaining the linear slope converging around 120ms. Having retransmissions without packet loss does not pay off but leads to increased collisions, even in small networks.

**Content Arrival Time (Multi-hop).** Multi-hop communication delays content delivery by one order of magnitude, see Figure 2b. The only remarkable exception of a significant performance decrease is NDN in the one-to-many scenario. Here, 75% of the content is delivered almost immediately ( $\leq 1\text{ ms}$ ) because of in-network caching and properties of the topology. A line topology enforces

that a content item quickly passes all (consumer) nodes, which will cache the content. When consumers that did not request content before, send their Interest, they benefit from their local cache. Then, the data delivery time is mainly limited by local system resources (RAM access *etc.*).

In the many-to-one and one-to-many use case of NDN and BT mesh respectively, data delivery does not significantly diverge, for similar reasons as in the single-hop scenario. It is worth noting that caching in NDN becomes only effective in case of retransmissions, because a single consumer requests content in this scenario. Surprisingly, in the many-to-one use case, BT mesh performs similar to the previous configuration, with less divergence compared to single-hop. We attribute these results to data submission on different channels. Providing non-overlapping channels between two peers reduces collisions in particular in multi-hop scenarios.

**Traffic Load.** We now analyze the traffic load per node, differentiating the original network traffic and retransmissions in place (cf., Figure 3). NDN clearly outperforms BT mesh. In contrast to BT mesh, NDN reduces the overall traffic load due to improved error handling, *i.e.*, packets on the network layer are retransmitted only when needed, and caching. Caching effects are nicely visible in the line topology (cf., Figure 3b), and comply with our discussions about the content delivery time.

BT mesh experiences many more packets per node. Any new content item is replicated a priori, leading to at least 1500 sent packets in our topology. Furthermore, content is relayed by each node that receives it, leading to additional 1500 sent packets by each relay, and increased receive overhead. Note that each BT mesh node is only listening to 1 of the 3 advertising channels at any time.

In the single producer scenarios (Fig 3a and 3b), the amount of transmitted packets is close to equal for all BT mesh nodes, as all packets are evenly relayed throughout the network, independent of its topology. For the single producer, multi-hop setup, NDN demonstrates the advantage of the in-network caching by a reduced network load at the producer node.

Looking at the multi producer scenarios (Fig. 3c and 3d), the effect of BT mesh’s static link layer retransmissions is even more distinctive. The number of transmitted and received network packets is again roughly equal for the single- and multi-hop setups, but their absolute numbers are multiplied due to the increase in produced content items. In contrast, NDN shows comparable behavior to the single producer scenarios, as here only the source and destination of interest and content packets is turned around.

In general, the static link layer retransmission of BT mesh amplifies network load in all of the measured scenarios.

## 5 BT-ICN: ICN PERSPECTIVE FOR BLE

In the following, we discuss two perspectives for joining the best properties out of both worlds: (i) building BT mesh-like properties by using BLE as a reliable transport for ICN, and (ii) implement BT mesh’s friend relationship applying ICN principles.

**ICN-over-BLE.** BLE’s connection-oriented mode provides a resilient link based on time sliced channel hopping (TSCH). Running ICN over such BLE links, while tightly coupling ICN routing to BLE connection management, promises to be efficient and reliable in terms of network resource usage. Such a setup can potentially

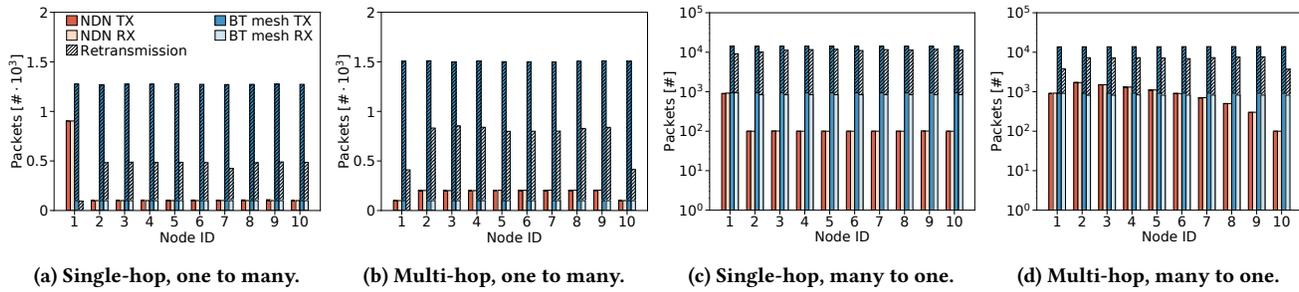


Figure 3: Traffic load for BT mesh and NDN.

deliver similar service properties as BT mesh, while taking full advantage of both, caching and routing capabilities of ICN, as well as the reliable BLE link layer.

**Friend role using ICN principles.** Following, we propose an implementation of the low-power friend feature in ICN to demonstrate the similarity of the friend feature and the principles of NDN.

A friend node that requests content from its sleepy constrained node may establish PIT state on both participants with a *long-lived Interest* at a point in time, with a PIT timeout longer than the node’s sleep cycle. The node may reply to the Interest whenever it wakes up. Alternatively, a method like HoPP [19] may be used to publish a content to the friend node in a timely manner.

If a constrained node wants to request content, it may simply state an Interest to its friend. If the content is not replied shortly, the low-power node may decide to repeat the Interest after some time, with a high probability that the request is now satisfied from the friend’s content cache.

In BT mesh, dedicated caches are maintained for each low-power node. This is simplified by our approach, as the friend nodes need simply to deploy vanilla ICN caching.

## 6 RELATED WORK

The use of BLE and BT mesh in low-power environments has been analyzed from various perspectives: (i) IETF standard solutions for IPv6 over connection oriented BLE networks have been released in [28], extended, and analyzed in [24, 34]. There are ongoing efforts to standardize mesh topologies for IPv6 traffic over BLE [15]. (ii) Initial performance analysis of plain BT mesh networks were presented in [7, 12] and a collection of academic solutions deal with improvements on the routing mechanism, which are summarized in [13]. (iii) Few work deals with Bluetooth in the context of ICNs. The authors of [35] introduce a Bluetooth convergence layer for NetInf [1] and provide an implementation for Android devices. [26] bases on the this implementation and focuses an opportunistic networking platform for content dissemination. In [9], the NDN approach is applied for BT mesh networks to aggregate distributed databases.

Experimental evaluations of ICN in the constrained IoT employing the large FIT IoT testbed have started in 2014 [6] and since then established a tradition of reality-checks in this rather unpredictable domain.

## 7 CONCLUSIONS AND OUTLOOK

Starting from the common claim that Bluetooth mesh implements ICN principles, we conducted a detailed comparative analysis. Our theoretical and experimental study, however, revealed that there is no evidence for this statement. BT mesh performs flooding without content caching, leading to degraded network performance, while ICN principles still need to be added in future work.

The holistic nature of BT mesh, providing a full vertical stack and covering all aspects of a nodes lifetime, makes it a production-ready solution. While there exists a large number of candidate solutions that can be combined with ICN-based networking to achieve a similar set of functionality, some parts are not covered by proven go-to solutions (*i.e.*, provisioning and managing of security artifacts, managing of large-scale namespaces).

From a network perspective, BT mesh proved suitable for strictly local appliances (*i.e.*, smart lighting), provided the installation size does not grow too large and not too many neighbors use this technology simultaneously. This is a significant limitation at the core of this technology. ICN on the other hand is designed specifically to cope with large-scale deployments and we showed that NDN can utilize network resources more efficiently than BT mesh.

## A Note on Reproducibility

We fully support reproducible research [8, 30] and perform all our experiments using open source software and an open access testbed. Code and documentation is available on Github at <https://github.com/5G-I3/ACM-ICN-2019-BLE>.

## Acknowledgments

We would like to thank the anonymous reviewers and our shepherd Alex Afanasyev for their valuable feedback. This work was supported in parts by the German Federal Ministry of Education and Research (BMBF) within the 5G project *I3* and the VIP+ project *RAPstore*.

## REFERENCES

- [1] B. Ahlgren et al. 2010. *Second NetInf Architecture Description*. Tech.report D-6.2 v2.0. 4Ward EU FP7 Project.
- [2] M. Amadeo, C. Campolo, A. Iera, and A. Molinaro. 2014. Named data networking for IoT: An architectural perspective. In *2014 European Conference on Networks and Communications (EuCNC)*. IEEE, Piscataway, NJ, USA, 1–5.
- [3] Marica Amadeo, Claudia Campolo, and Antonella Molinaro. 2014. Multi-source Data Retrieval in IoT via Named Data Networking. In *Proc. of ACM ICN*. ACM, New York, NY, USA, 67–76.

- [4] Emmanuel Baccelli, Cenk Gündogan, Oliver Hahm, Peter Kietzmann, Martine Lenders, Hauke Petersen, Kaspar Schleiser, Thomas C. Schmidt, and Matthias Wählisch. 2018. RIOT: an Open Source Operating System for Low-end Embedded Devices in the IoT. *IEEE Internet of Things Journal* 5, 6 (December 2018), 4428–4440. <http://dx.doi.org/10.1109/JIOT.2018.2815038>
- [5] Emmanuel Baccelli, Oliver Hahm, Mesut Günes, Matthias Wählisch, and Thomas C. Schmidt. 2013. RIOT OS: Towards an OS for the Internet of Things. In *Proc. of the 32nd IEEE INFOCOM. Poster*. IEEE Press, Piscataway, NJ, USA, 79–80.
- [6] Emmanuel Baccelli, Christian Mehlis, Oliver Hahm, Thomas C. Schmidt, and Matthias Wählisch. 2014. Information Centric Networking in the IoT: Experiments with NDN in the Wild. In *Proc. of 1st ACM Conf. on Information-Centric Networking (ICN-2014)*. ACM, New York, 77–86. <http://dx.doi.org/10.1145/2660129.2660144>
- [7] Mathias Baert, Jen Rossey, Adnan Shahid, and Jeroen Hoebeke. 2018. The Bluetooth Mesh Standard: An Overview and Experimental Evaluation. *Sensors* 18, 8 (2018), 2409.
- [8] Vaibhav Bajpai, Anna Brunstrom, Anja Feldmann, Wolfgang Kellerer, Aiko Pras, Henning Schulzrinne, Georgios Smaragdakis, Matthias Wählisch, and Klaus Wehrle. 2019. The Dagstuhl Beginners Guide to Reproducibility for Experimental Networking Research. *ACM SIGCOMM Computer Communication Review* 49, 1 (January 2019), 24–30. Editorial note.
- [9] András Balogh, Sándor Imre, Károly Lendvai, and Sándor Szabó. 2015. Service Mediation in multihop Bluetooth Low Energy networks based on NDN approach. In *SoftCOM*. IEEE, 285–289.
- [10] Antonio Carzaniga, Michele Papalini, and Alexander L. Wolf. 2011. Content-based Publish/Subscribe Networking and Information-centric Networking. In *Proc. of the ACM SIGCOMM WS on Information-centric Networking (ICN '11)*. ACM, New York, NY, USA, 56–61.
- [11] Jiachen Chen, Mayutan Arumathurai, Lei Jiao, Xiaoming Fu, and Kadangode Ramakrishnan. 2011. COPSS: An Efficient Content Oriented Publish/Subscribe System. In *ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS'11)*. IEEE Computer Society, Los Alamitos, CA, USA, 99–110.
- [12] Seyed Mahdi Darroudi, Raúl Caldera-Sánchez, and Carles Gomez. 2019. Bluetooth Mesh Energy Consumption: A Model. *Sensors* 19, 5 (2019), 1238.
- [13] Seyed Mahdi Darroudi and Carles Gomez. 2017. Bluetooth Low Energy Mesh Networks: A Survey. *Sensors* 17, 7 (2017), 1467.
- [14] Asmir Gogic, Aljo Mujcic, Sandra Ibric, and Nermin Suljanovic. 2016. Performance Analysis of Bluetooth Low Energy Mesh Routing Algorithm in Case of Disaster Prediction. *Journal of Information, Control and Computer Sciences* 10, 6 (2016), 1075 – 1081.
- [15] Carles Gomez, Seyed Darroudi, Teemu Savolainen, and Michael Spoerk. 2019. *IPv6 Mesh over BLUETOOTH(R) Low Energy using IPSF*. Internet-Draft – work in progress 05. IETF.
- [16] Bluetooth Special Interest Group. 2019. *Bluetooth Core Specification Version 5.1*. Technical Report. <https://www.bluetooth.com/specifications/bluetooth-core-specification>
- [17] Bluetooth Special Interest Group. 2019. *Bluetooth Mesh Profile Specification Version 1.0.1*. Technical Report. <https://www.bluetooth.com/specifications/mesh-specifications/>
- [18] Cenk Gündogan, Peter Kietzmann, Martine Lenders, Hauke Petersen, Thomas C. Schmidt, and Matthias Wählisch. 2018. NDN, CoAP, and MQTT: A Comparative Measurement Study in the IoT. In *Proc. of 5th ACM Conference on Information-Centric Networking (ICN)*. ACM, New York, NY, USA, 159–171. <https://conferences.sigcomm.org/acm-icn/2018/proceedings/icn18-final46.pdf>
- [19] Cenk Gündogan, Peter Kietzmann, Thomas C. Schmidt, and Matthias Wählisch. 2018. HoPP: Robust and Resilient Publish-Subscribe for an Information-Centric Internet of Things. In *Proc. of the 43rd IEEE Conference on Local Computer Networks (LCN)*. IEEE Press, Piscataway, NJ, USA, 331–334. <http://doi.org/10.1109/LCN.2018.8638030>
- [20] Cenk Gündogan, Peter Kietzmann, Thomas C. Schmidt, and Matthias Wählisch. 2019. ICNLoWPAN – Named-Data Networking in Low Power IoT Networks. In *Proc. of 18th IFIP Networking Conference*. IEEE Press, 1–9. <http://dx.doi.org/10.23919/IFIPNetworking.2019.8816850>
- [21] Van Jacobson, Diana K. Smetters, James D. Thornton, and Michael F. Plass. 2009. Networking Named Content. In *5th Int. Conf. on emerging Networking Experiments and Technologies (ACM CoNEXT'09)*. ACM, New York, NY, USA, 1–12.
- [22] Peter Kietzmann, Cenk Gündogan, Thomas C. Schmidt, Oliver Hahm, and Matthias Wählisch. 2017. The Need for a Name to MAC Address Mapping in NDN: Towards Quantifying the Resource Gain. In *Proc. of 4th ACM Conference on Information-Centric Networking (ICN)*. ACM, New York, NY, USA, 36–42.
- [23] Dmitrij Lagutin, Kari Visala, and Sasu Tarkoma. 2010. Publish/Subscribe for Internet: PSIRP Perspective. *Future Internet Assembly* 84 (2010), 75–84.
- [24] Taeseop Lee, Myung-Sup Lee, Hyung-Sin Kim, and Saewoong Bahk. 2016. A Synergistic Architecture for RPL over BLE. In *SECON*. IEEE, 1–9.
- [25] Tianxiang Li, Wentao Shang, Alex Afanasyev, Lan Wang, and Lixia Zhang. 2018. A Brief Introduction to NDN Dataset Synchronization (NDN Sync). In *MILCOM 2018 – 2018 IEEE Military Communications Conference (MILCOM)*, 612–618.
- [26] Pourya Moradina and Imal Sakhi. 2014. *Mobile Opportunistic Services for Experience Sharing : Via a NetInf Android Application*. Master's thesis. KTH, Radio Systems Laboratory (RS Lab).
- [27] Kathleen Nichols. 2019. Lessons Learned Building a Secure, Network Measurement Framework using Basic NDN. In *Proc. of ACM ICN*. ACM, New York, NY, USA.
- [28] J. Nieminen, T. Savolainen, M. Isomaki, B. Patil, Z. Shelby, and C. Gomez. 2015. *IPv6 over BLUETOOTH(R) Low Energy*. RFC 7668. IETF.
- [29] Giuseppe Rossini and D. Rossi. 2012. A dive into the caching performance of Content Centric Networking. In *IEEE 17th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD'12)*. Barcelona, Spain, 105–109.
- [30] Quirin Scheitle, Matthias Wählisch, Oliver Gasser, Thomas C. Schmidt, and Georg Carle. 2017. Towards an Ecosystem for Reproducible Research in Computer Networking. In *Proc. of ACM SIGCOMM Reproducibility Workshop*. ACM, New York, NY, USA, 5–8.
- [31] Wentao Shang, Minsheng Zhang, Alexander Afanasyev, Jeff Burke, Lan Wang, and Lixia Zhang. 2018. *Publish-Subscribe Communication in Building Management Systems over Named Data Networking*. Technical Report NDN-0066. NDN-Project. <https://named-data.net/wp-content/uploads/2018/10/ndn-tr-0055-1-ndn-ps.pdf>
- [32] SILVAIR. 2018. Information-centric networking – a revolutionary approach to wireless lighting control. <http://silvair.com/resources/information-centric-networking/>
- [33] Hyun soo Kim, JungYub Lee, and Ju Wook Jang. 2015. BLEmesh: A Wireless Mesh Network Protocol for Bluetooth Low Energy Devices. In *International Conference on Future Internet of Things and Cloud*. 558–563.
- [34] Michael Spörk, Carlo Alberto Boano, Marco Zimmerling, and Kay Römer. 2017. BLEach: Exploiting the Full Potential of IPv6 over BLE in Constrained Embedded IoT Devices. In *SenSys*. ACM, 2:1–2:14.
- [35] Linus Sunde. 2013. *NetInf Node for Bluetooth Enabled Android Devices*. Master's thesis. Uppsala University, Department of Information Technology.
- [36] Christian Tschudin, Christopher Scherb, et al. 2018. CCN Lite: Lightweight implementation of the Content Centric Networking protocol. <http://ccn-lite.net>
- [37] X. Xu, H. Zhang, T. Li, and L. Zhang. 2018. Achieving Resilient Data Availability in Wireless Sensor Networks. In *2018 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 1–6.
- [38] George Xylomenos, Christopher N. Ververidis, Vasilios A. Siris, Nikos Fotiou, Christos Tsilopoulos, Xenofon Vasilakos, Konstantinos V. Katsaros, and George C. Polyzos. 2014. A Survey of Information-Centric Networking Research. *IEEE Communications Surveys and Tutorials* 16, 2 (2014), 1024–1049.
- [39] Lixia Zhang, Alexander Afanasyev, Jeffrey Burke, Van Jacobson, kc claffy, Patrick Crowley, Christos Papadopoulos, Lan Wang, and Beichuan Zhang. 2014. Named Data Networking. *SIGCOMM Comput. Commun. Rev.* 44, 3 (2014), 66–73.