6TiSCH
Internet-Draft
Intended status: Standards Track
Expires: December 18, 2014

P. Thubert, Ed.
Cisco
T. Watteyne
Linear Technology
RA. Assimiti
Centero
June 16, 2014

An Architecture for IPv6 over the TSCH mode of IEEE 802.15.4e
draft-ietf-6tisch-architecture-02

Abstract

This document presents an architecture for an IPv6 Multi-Link subnet that is composed of a high speed powered backbone and a number of IEEE802.15.4e TSCH wireless networks attached and synchronized by Backbone Routers. The TSCH schedule can be static or dynamic. 6TiSCH defines mechanisms to establish and maintain the routing and scheduling operations in a centralized, distributed, or mixed fashion. Backbone Routers perform proxy Neighbor Discovery operations over the backbone on behalf of the wireless devices, so they can share a same subnet and appear to be connected to the same backbone as classical devices

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

Table of Contents

1. Introduction

The emergence of radio technology enabled a large variety of new types of devices to be interconnected, at a very low marginal cost compared to wire, at any range from Near Field to interplanetary distances, and in circumstances where wiring would be less than practical, for instance rotating devices.

At the same time, a new breed of Time Sensitive Networks is being developed to enable traffic that is highly sensitive to jitter and quite sensitive to latency. Such traffic is not limited to voice and video, but also includes command and control operations such as found in industrial automation or in-vehicle sensors and actuators.

At IEEE802.1, the "Audio/Video Task Group", was renamed TSN for Time Sensitive Networking to address Deterministic Ethernet. The IEEE802.15.4 Medium access Control (MAC) has evolved with IEEE802.15.4e that provides in particular the timeSlotted Channel Hopping (TSCH) mode for industrial-type applications.

Though at a different time scale, both standards provide Deterministic capabilities to the point that a packet that pertains to a certain flow crosses the network from node to node following a very precise schedule, as a train that leaves intermediate stations at precise times along its path. With TSCH, time is formatted into timeSlots, and an individual cell is allocated to unicast or broadcast communication at the MAC level. The time slotted operation reduces collisions, saves energy, and enables to more closely engineer the network for deterministic properties. The channel hopping aspect is a simple and efficient technique to combat multipath fading and external interference (for example by WiFi emitters).

This document presents an architecture for an IPv6 Multi-Link subnet that is composed of a high speed powered backbone and a number of IEEE802.15.4e TSCH wireless networks attached and synchronized by backbone routers. Route Computation may be achieved in a centralized fashion by a Path Computation Element (PCE), in a distributed fashion using the Routing Protocol for Low Power and Lossy Networks (RPL), or in a mixed mode. The Backbone Routers perform proxy IPv6 neighbor Discovery (ND) operations over the backbone on behalf of the wireless devices, so they can share a same IPv6 subnet and appear to be connected to the same backbone as classical devices. timeSlots and other device resources are managed by an abstract Network Management

Entity (NME) that may cooperate with the PCE in order to minimize the interaction with and the load on the constrained device.

2. Terminology

Readers are expected to be familiar with all the terms and concepts that are discussed in "neighbor Discovery for IP version 6" [RFC4861], "IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals" [RFC4919], neighbor Discovery Optimization for Low-power and Lossy Networks [RFC6775] and "Multi-link Subnet Support in IPv6" [I-D.ietf-ipv6-multilink-subnets].

Readers may benefit from reading the "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks" [RFC6550] specification; "Multi-Link Subnet Issues" [RFC4903]; "Mobility Support in IPv6" [RFC6275]; "neighbor Discovery Proxies (ND Proxy)" [RFC4389]; "IPv6 Stateless Address Autoconfiguration" [RFC4862]; "FCFS SAVI: First-Come, First-Served Source Address Validation Improvement for Locally Assigned IPv6 Addresses" [RFC6620]; and "Optimistic Duplicate Address Detection" [RFC4429] prior to this specification for a clear understanding of the art in ND-proxying and binding.

The draft uses terminology defined or referenced in [I-D.ietf-6tisch-terminology], [I-D.chakrabarti-nordmark-6man-efficient-nd], [I-D.ietf-roll-rpl-industrial-applicability], [RFC5191] and [RFC4080].

The draft also conforms to the terms and models described in [RFC3444] and [RFC5889] and uses the vocabulary and the concepts defined in [RFC4291] for the IPv6 Architecture.

3. Applications and Goals

The architecture derives from existing industrial standards for Process Control by its focus on Deterministic Networking, in particular with the use of the IEEE802.15.4e TSCH MAC [IEEE802154e] and the centralized PCE. This approach leverages the TSCH MAC benefits for high reliability against interference, low-power consumption on deterministic traffic, and its Traffic Engineering capabilities. Deterministic Networking applies in particular to open and closed control loops, as well as supervisory control flows and management.

An incremental set of industrial requirements are addressed with the addition of an autonomic and distributed routing operation based on RPL. These use cases include plant setup and decommissioning, as

well as monitoring of lots of lesser importance measurements such as corrosion and events. RPL also enables mobile use cases such as mobile workers and cranes.

A Backbone Router is included in order to scale the factory plant subnet to address large deployments, with proxy ND and time synchronization over a high speed backbone.

The architecture also applies to building automation that leverage RPL's storing mode to address multipath over a large number of hops, in-vehicle command and control that can be as demanding as industrial applications, commercial automation and asset Tracking with mobile scenarios, home automation and domotics which become more reliable and thus provide a better user experience, and resource management (energy, water, etc.).

4. Overview and Scope

The scope of the present work is a subnet that, in its basic configuration, is made of a IEEE802.15.4e timeSlotted Channel Hopping (TSCH) [I-D.ietf-6tisch-tsch] MAC Low Power Lossy Network (LLN).

```
            ---+-------- ............ ------------
               |      External Network       |
               |                          +-----+
            +-----+                       | NME |
            |     | LLN Border            |     |
            |     | router                +-----+
            +-----+
          o    o   o
   o     o   o     o
      o   o LLN   o    o     o
         o   o   o       o
                 o
```

Figure 1: Basic Configuration

The LLN devices communicate over IPv6 [RFC2460] using the 6LoWPAN Header Compression (6LoWPAN HC) [RFC6282]. From the perspective of Layer 3, a single LLN interface (typically an IEEE802.15.4-compliant radio) may be seen as a collection of Links with different capabilities for unicast or multicast services. An IPv6 subnet spans over multiple links, effectively forming a Multi-Link subnet. Within that subnet, neighbor Devices are discovered with 6LoWPAN neighbor Discovery (6LoWPAN ND) [RFC6775]. RPL [RFC6550] enables routing within the LLN, typically within the Multi-Link subnet in the so called Route Over fashion.

RPL forms Destination Oriented Directed Acyclic Graphs (DODAGs) within Instances of the protocol, each Instance being associated with an Objective Function (OF) to form a routing topology. A particular LLN device, the LLN Border Router (LBR), acts as RPL root, 6LoWPAN HC terminator, and LLN Border Router (LBR) to the outside. The LBR is usually powered. More on RPL Instances can be found in RPL [RFC6550], sections "3.1.2. RPL Identifiers" and "3.1.3. Instances, DODAGs, and DODAG Versions".

An extended configuration of the subnet comprises multiple LLNs. The LLNs are interconnected and synchronized over a backbone, that can be wired or wireless. The backbone can be a classical IPv6 network, with neighbor Discovery operating as defined in [RFC4861] and [RFC4862]. The backbone can also support Efficiency-aware IPv6 neighbor Discovery Optimizations [I-D.chakrabarti-nordmark-6man-efficient-nd] in mixed mode as described in [I-D.thubert-6lowpan-backbone-router].

Security is often handled at layer 2 and Layer 4. Authentication during the join process can be handled by the Protocol for Carrying Authentication for Network access (PANA) [RFC5191].

The LLN devices are time-synchronized at the MAC level. The LBR that serves as time source is a RPL parent in a particular RPL Instance that serves for time synchronization; this way, the time synchronization starts at the RPL root and follows the RPL DODAGs with no timing loop.

In the extended configuration, the functionality of the LBR is enhanced to that of Backbone Router (BBR). A BBR is an LBR, but also an Energy Aware Default Router (NEAR) as defined in [I-D.chakrabarti-nordmark-6man-efficient-nd]. The BBR performs ND proxy operations between the registered devices and the classical ND devices that are located over the backbone. 6TiSCH BBRs synchronize with one another over the backbone, so as to ensure that the multiple LLNs that form the IPv6 subnet stay tightly synchronized. If the Backbone is Deterministic (such as defined by the Time Sensitive Networking WG at IEEE), then the Backbone Router ensures that the end-to-end deterministic behavior is maintained between the LLN and the backbone.

```
                  ---+-------- ............ ------------
                     |      External Network       |
                     |                          +-----+
                     |             +-----+      | NME |
                  +-----+          |  +-----+   |     |
                  |     | Router   |  | PCE |   +-----+
                  |     |          +--|     |
                  +-----+             +-----+
                     |                   |
                     | Subnet Backbone   |
               +--------------------+------------------+
               |                    |                  |
            +-----+             +-----+             +-----+
            |     | Backbone    |     | Backbone    |     | Backbone
       o    |     | router      |     | router      |     | router
            +-----+             +-----+             +-----+
       o                  o                   o                 o   o
           o    o   o         o   o  o   o         o  o   o    o
      o             o        o  LLN      o      o         o      o
         o   o    o      o      o o     o  o   o    o    o     o
```

                     Figure 2: Extended Configuration

The main architectural blocks are arranged as follows:

```
      +-----+-----+-----+-----+-------+-----+
      |PCEP |   CoAP    |PANA |6LoWPAN| RPL |
      | PCE |DTLS |     |     |   ND  |     |
      +-----+-----+-----+-----+-------+-----+-----+
      | TCP |       UDP       |    ICMP     |RSVP |
      +-----+-----+-----+-----+-------+-----+-----+
      |                 IPv6                      |
      +-------------------------------------------+
      |               6LoWPAN HC                  |
      +-------------------------------------------+
      |                   6top                    |
      +-------------------------------------------+
      |             IEEE802.15.4e   TSCH          |
      +-------------------------------------------+
```

                          Figure 3: 6TiSCH stack

RPL is the routing protocol of choice for LLNs. (TBD RPL) whether there is a need to define a 6TiSCH OF.

(tbd NME) COMAN is working on network Management for LLN. They are considering the Open Mobile Alliance (OMA) Lightweight M2M (LWM2M)

Object system. This standard includes DTLS, CoAP (core plus Block and Observe patterns), SenML and CoAP Resource Directory.

(tbd PCE) need to work with PCE WG to define flows to PCE, and define how to accommodate PCE routes and reservation. Will probably look a lot like GMPLS.

(tbd PANA) There is a debate whether PANA (layer 3) or IEEE802.1x (layer 2) should be used in the join process. There is also a debate whether the node should be able to send any unprotected packet on the medium. Regardless, the security model must ensure that, prior to a join process, packets from a untrusted device must be controlled in volume and in reachability.

(tbd Backbone Router) need to work with 6MAN to define ND proxy. Also need BBR sync sync between deterministic Ethernet and 6TiSCH LLNs.

IEEE802.1TSN: external, maintain consistency. See also AVnu.

IEEE802.15.4: external, (tbd need updates?).

ISA100.20 Common Network Management: external, maintain consistency.

The 6TiSCH Operation sublayer (6top) [I-D.wang-6tisch-6top-sublayer] is an Logical Link Control (LLC) or a portion thereof that provides the abstraction of an IP link over a TSCH MAC.

5. Communication Paradigms and Interaction Models

[I-D.ietf-6tisch-terminology] defines the terms of Communication Paradigms and Interaction Models, which can be placed in parallel to the Information Models and Data Models that are defined in [RFC3444].

A Communication Paradigms would be an abstract view of a protocol exchange, and would come with an Information Model for the information that is being exchanged. In contrast, an Interaction Models would be more refined and could point on standard operation such as a Representational state transfer (REST) "GET" operation and would match a Data Model for the data that is provided over the protocol exchange.

[I-D.ietf-roll-rpl-industrial-applicability] section 2.1.3. and next discusses appplication-layer paradigms, such as Source-sink (SS) that is a Multipeer to Multipeer (MP2MP) model that is primarily used for alarms and alerts, Publish-subscribe (PS, or pub/sub) that is typically used for sensor data, as well as Peer-to-peer (P2P) and Peer-to-multipeer (P2MP) communications. Additional considerations

on Duocast and its N-cast generalization are also provided. Those paradigms are frequently used in industrial automation, which is a major use case for IEEE802.15.4e TSCH wireless networks with [ISA100.11a] and [WirelessHART], that provides a wireless access to [HART] applications and devices.

This specification focuses on Communication Paradigms and Interaction Models for packet forwarding and TSCH resources (cells) management. L ink-layer and Network-layer Packet forwarding interactions are discussed in Section 6, whereas Link-layer (one-hop), Network-layer (multithop along a track), and Application-layer (remote control) management mechanisms for the TSCH schedule are discussed in Section 8.

6. Forwarding Models

6TiSCH supports three different forwarding model, G-MPLS Track Forwarding (TF), 6LoWPAN Fragment Forwarding (FF) and IPv6 Forwarding (6F).

6.1. Track Forwarding

Track Forwarding is the simplest and fastest. A bundle of cells set to receive (RX-cells) is uniquely paired to a bundle of cells that are set to transmit (TX-cells), representing a layer-2 forwarding state that can be used regardless of the network layer protocol. This model can effectively be seen as a Generalized Multi-protocol Label Switching (G-MPLS) operation in that the information used to switch a frame is not an explicit label, but rather related to other properties of the way the packet was received, a particular cell in the case of 6TiSCH. As a result, as long as the TSCH MAC (and Layer 2 security) accepts a frame, that frame can be switched regardless of the protocol, whether this is an IPv6 packet, a 6LoWPAN fragment, or a frame from an alternate protocol such as WirelessHART or ISA100.11a.

A Track is defined end-to-end as a succession of paired bundles. A cell in such a bundle belongs to at most one Track but it may be reused opportunistically on a per-hop basis for routed packets. For a given iteration of the device schedule, the effective channel of the cell is obtained by adding a pseudo-random number to the channelOffset of the cell, which results in a rotation of the frequency that used for transmission.

A data frame that is forwarded along a Track has a destination MAC address set to broadcast or a multicast address depending on MAC support. This way, the MAC layer in the intermediate nodes accepts the incoming frame and 6top switches it without incurring a change in

the MAC header. In the case of IEEE802.15.4e, this means effectively broadcast, so that along the Track the short address for the destination is set to 0xFFFF.

Conversely, a frame that is received along a Track with a destination MAC address set to this node is extracted from the Track stream and delivered to the upper layer. A frame with an unrecognized MAC address is dropped at the MAC layer and thus is not received at the 6top sublayer.

There are 2 modes for a Track, transport mode and tunnel mode.

6.1.1. Transport Mode

In transport mode, the Protocol Data Unit (PDU) is associated with flow-dependant meta-data that refers uniquely to the Track, so the 6top sublayer can place the frame in the appropriate cell without ambiguity. In the case of IPv6 traffic, this flow identification is transported in the Flow Label of the IPv6 header. Associated with the source IPv6 address, the Flow Label forms a globally unique identifier for that particular Track that is validated at egress before restoring the destination MAC address (DMAC) and punting to the upper layer.

```
                       |                                    ^
   +--------------+    |                                    |
   |     IPv6     |    |                                    |
   +--------------+    |                                    |
   |  6LoWPAN HC  |    |                                    |
   +--------------+  ingress                              egress
   |     6top     |   sets     +----+          +----+     restores
   +--------------+  dmac to   |    |          |    |     dmac to
   |   TSCH MAC   |   brdcst   |    |          |    |      self
   +--------------+    |       |    |          |    |       |
   |   LLN PHY    |    +-------+    +--...-----+    +-------+
   +--------------+
```

Track Forwarding, Transport Mode

6.1.2. Tunnel Mode

In tunnel mode, the frames originate from an arbitrary protocol over a compatible MAC that may or may not be synchronized with the 6TiSCH network. An example of this would be a router with a dual radio that is capable of receiving and sending WirelessHART or ISA100.11a frames with the second radio, by presenting itself as an access Point or a Backbone Router, respectively.

In that mode, some entity (e.g. PCE) can coordinate with a WirelessHART Network Manager or an ISA100.11a System Manager to specify the flows that are to be transported transparently over the Track.

```
   +--------------+
   |     IPv6     |
   +--------------+
   |  6LoWPAN HC  |
   +--------------+             set           restore
   |     6top     |            +dmac+          +dmac+
   +--------------+            |    |          |    |
   |   TSCH MAC   |            |    |          |    |
   +--------------+            |    |          |    |
   |   LLN PHY    |    +-------+    +--...-----+    +-------+
   +--------------+    |   ingress                 egress   |
                       |                                    |
   +--------------+    |                                    |
   |   LLN PHY    |    |                                    |
   +--------------+    |                                    |
   |   TSCH MAC   |    |                                    |
   +--------------+    |                                    |
   |ISA100/WiHART |    |                                    v
   +--------------+
```

Figure 4: Track Forwarding, Tunnel Mode

In that case, the flow information that identifies the Track at the ingress 6TiSCH router is derived from the RX-cell. The dmac is set to this node but the flow information indicates that the frame must be tunnelled over a particular Track so the frame is not passed to the upper layer. Instead, the dmac is forced to broadcast and the frame is passed to the 6top sublayer for switching.

At the egress 6TiSCH router, the reverse operation occurs. Based on metadata associated to the Track, the frame is passed to the appropriate link layer with the destination MAC restored.

6.1.3. Tunnel Metadata

Metadata coming with the Track configuration is expected to provide the destination MAC address of the egress endpoint as well as the tunnel mode and specific data depending on the mode, for instance a service access point for frame delivery at egress. If the tunnel egress point does not have a MAC address that matches the configuration, the Track installation fails.

In transport mode, if the final layer-3 destination is the tunnel termination, then it is possible that the IPv6 address of the destination is compressed at the 6LoWPAN sublayer based on the MAC address. It is thus mandatory at the ingress point to validate that the MAC address that was used at the 6LoWPAN sublayer for compression matches that of the tunnel egress point. For that reason, the node that injects a packet on a Track checks that the destination is effectively that of the tunnel egress point before it overwrites it to broadcast. The 6top sublayer at the tunnel egress point reverts that operation to the MAC address obtained from the tunnel metadata.

6.2. Fragment Forwarding

Considering that 6LoWPAN packets can be as large as 1280 bytes (the IPv6 MTU), and that the non-storing mode of RPL implies Source Routing that requires space for routing headers, and that a IEEE802.15.4 frame with security may carry in the order of 80 bytes of effective payload, an IPv6 packet might be fragmented into more than 16 fragments at the 6LoWPAN sublayer.

This level of fragmentation is much higher than that traditionally experienced over the Internet with IPv4 fragments, where fragmentation is already known as harmful.

In the case to a multihop route within a 6TiSCH network, Hop-by-Hop recomposition occurs at each hop in order to reform the packet and route it. This creates additional latency and forces intermediate nodes to store a portion of a packet for an undetermined time, thus impacting critical resources such as memory and battery.

[I-D.thubert-roll-forwarding-frags] describes a mechanism whereby the datagram tag in the 6LoWPAN Fragment is used as a label for switching at the 6LoWPAN sublayer. The draft allows for a degree of flow control base on an Explicit Congestion Notification, as well as end-to-end individual fragment recovery.

```
                       |                                    ^
   +--------------+    |                                    |
   |     IPv6     |    |       +----+          +----+       |
   +--------------+    |       |    |          |    |       |
   |  6LoWPAN HC  |    |       learn           learn        |
   +--------------+    |       |    |          |    |       |
   |     6top     |    |       |    |          |    |       |
   +--------------+    |       |    |          |    |       |
   |   TSCH MAC   |    |       |    |          |    |       |
   +--------------+    |       |    |          |    |       |
   |   LLN PHY    |    +-------+    +--...-----+    +-------+
   +--------------+
```

Figure 5: Forwarding First Fragment

In that model, the first fragment is routed based on the IPv6 header that is present in that fragment. The 6LoWPAN sublayer learns the next hop selection, generates a new datagram tag for transmission to the next hop, and stores that information indexed by the incoming MAC address and datagram tag. The next fragments are then switched based on that stored state.

```
                       |                                    ^
   +--------------+    |                                    |
   |     IPv6     |    |                                    |
   +--------------+    |                                    |
   |  6LoWPAN HC  |    |       replay          replay       |
   +--------------+    |       |    |          |    |       |
   |     6top     |    |       |    |          |    |       |
   +--------------+    |       |    |          |    |       |
   |   TSCH MAC   |    |       |    |          |    |       |
   +--------------+    |       |    |          |    |       |
   |   LLN PHY    |    +-------+    +--...-----+    +-------+
   +--------------+
```

Figure 6: Forwarding Next Fragment

A bitmap and an ECN echo in the end-to-end acknowledgement enable the source to resend the missing fragments selectively. The first fragment may be resent to carve a new path in case of a path failure. The ECN echo set indicates that the number of outstanding fragments should be reduced.

6.3. IPv6 Forwarding

As the packets are routed at layer 3, traditional QoS and RED operations are expected to prioritize flows with differentiated services. A new class of service for Deterministic Forwarding is

being defined to that effect in
[I-D.svshah-tsvwg-lln-diffserv-recommendations].

```
                       |                                    ^
   +--------------+    |                                    |
   |     IPv6     |    |       +-QoS+          +-QoS+       |
   +--------------+    |       |    |          |    |       |
   |  6LoWPAN HC  |    |       |    |          |    |       |
   +--------------+    |       |    |          |    |       |
   |     6top     |    |       |    |          |    |       |
   +--------------+    |       |    |          |    |       |
   |   TSCH MAC   |    |       |    |          |    |       |
   +--------------+    |       |    |          |    |       |
   |   LLN PHY    |    +-------+    +--...-----+    +-------+
   +--------------+
```

Figure 7: IP Forwarding

7. TSCH and 6top

7.1. 6top

6top is a logical link control sitting between the IP layer and the TSCH MAC layer, which provides the link abstraction that is required for IP operations. The 6top operations are specified in [I-D.wang-6tisch-6top-sublayer]. In particular, 6top provides a management interface that enables an external management entity to schedule cells and slotFrames, and allows the addition of complementary functionality, for instance to support a dynamic schedule management based on observed resource usage as discussed in section Section 8.2. The 6top data model and management interfaces are further discussed in Section 8.3.

If the scheduling entity explicitly specifies the slotOffset/ channelOffset of the cells to be added/deleted, those cells are marked as "hard". 6top cannot move hard cells in the TSCH schedule. Hard cells are for example used by a central PCE.

6top contains a monitoring process which monitors the performance of cells, and can move a cell in the TSCH schedule when it performs bad. This is only applicable to cells which are marked as "soft". To reserve a soft cell, the higher layer does not indicate the exact slotOffset/channelOffset of the cell to add, but rather the resulting bandwidth and QoS requirements. When the monitoring process triggers a cell reallocation, the two neighbor motes communicating over this cell negotiate its new position in the TSCH schedule.

7.2. 6top and RPL Objective Function operations

An implementation of a RPL [RFC6550] Objective Function (OF), such as the RPL Objective Function Zero (OF0) [RFC6552] that is used in the Minimal 6TiSCH Configuration [I-D.ietf-6tisch-minimal] to support RPL over a static schedule, may leverage, for its internal computation, the information maintained by 6top.

In particular, 6top creates and maintains an abstract neighbor table. A neighbor table entry contains a set of statistics with respect to that specific neighbor including the time when the last packet has been received from that neighbor, a set of cell quality metrics (RSSI, LQI), the number of packets sent to the neighbor or the number of packets received from it. This information can be obtained through 6top management APIs as detailed in the 6top sublayer specification [I-D.wang-6tisch-6top-sublayer] and used to compute a Rank Increment that will determine the selection of the preferred parent.

6top provides statistics about the underlying layer so the OF can be tuned to the nature of the TSCH MAC layer. 6top also enables the RPL OF to influence the MAC behaviour, for instance by configuring the periodicity of IEEE802.15.4e Extended Beacons (EB's). By augmenting the EB periodicity, it is possible to change the network dynamics so as to improve the support of devices that may change their point of attachment in the 6TiSCH network.

Some RPL control messages, such as the DODAG Information Object (DIO) are ICMPv6 messages that are broadcast to all neighbor nodes. With 6TiSCH, the broadcast channel requirement is addressed by 6top by configuring TSCH to provide a broadcast channel, as opposed to, for instance, piggybacking the DIO messages in Enhance Beacons.

In the TSCH schedule, each cell has the IEEE802.15.4e LinkType attribute. Setting the LinkType to ADVERTISING indicates that the cell MAY be used to send an Enhanced Beacon. When a node forms its Enhanced Beacon, the cell, with LinkType=ADVERTISING, SHOULD be included in the FrameAndLinkIE, and its LinkOption field SHOULD be set to the combination of "Receive" and "Timekeeping". The receiver of the Enhanced Beacon MAY be listening at the cell to get the Enhanced Beacon ([IEEE802154e]). 6top takes this way to establish broadcast channel, which not only allows TSCH to broadcast Enhanced Beacons, but also allows an upper layer like RPL.

To broadcast ICMPv6 control messages used by RPL such as DIO or DAO, 6top uses the payload of a Data frames. The message is inserted into the queue associated with the cells which LinkType is set to ADVERTISING. Then, taking advantage of the broadcast cell feature

established with FrameAndLinkIE (as described above), the RPL control message can be received by neighbors, which enables the maintenance of RPL DODAGs.

A LinkOption combining "Receive" and "Timekeeping" bits indicates to the receivers of the Enhanced Beacon that the cell MUST be used as a broadcast cell. The frequency of sending Enhanced Beacons or other broadcast messages by the upper layer is determined by the timers associated with the messages. For example, the transmission of Enhance Beacons is triggered by a timer in 6top; transmission of a DIO message is triggered by the trickle timer of RPL.

7.3. Network Synchronization

Nodes in a TSCH network must be time synchronized. A node keeps synchronized to its time source neighbor through a combination of frame-based and acknowledgement-based synchronization. In order to maximize battery life and network throughput, it is advisable that RPL ICMP discovery and maintenance traffic (governed by the trickle timer) be somehow coordinated with the transmission of time synchronization packets (especially with enhanced beacons). This could be achieved through an interaction of the 6top sublayer and the RPL objective Function, or could be controlled by a management entity.

Time distribution requires a loop-less structure. Nodes taken in a synchronization loop will rapidly desynchronize from the network and become isolated. It is expected that a RPL DAG with a dedicated global Instance is deployed for the purpose of time synchronization. That Instance is referred to as the Time Synchronization Global Instance (TSGI). The TSGI can be operated in either of the 3 modes that are detailed in RPL [RFC6550] section "3.1.3. Instances, DODAGs, and DODAG Versions". Multiple uncoordinated DODAGs with independent roots may be used if all the roots share a common time source such as the Global Positioning System (GPS). In the absence of a common time source, the TSGI should form a single DODAG with a virtual root. A backbone network is then used to synchronize and coordinate RPL operations between the backbone routers that act as sinks for the LLN.

A node that has not joined the TSGI advertises a MAC level Join Priority of 0xFF to notify its neighbors that is is not capable of serving as time parent. A node that has joined the TSGI advertises a MAC level Join Priority set to its DAGRank() in that Instance, where DAGRank() is the operation specified in [RFC6550], section "3.5.1. Rank Comparison".

A root is configured or obtains by some external means the knowledge of the RPLInstanceID for the TSGI. The root advertises its DagRank in the TSGI, that MUST be less than 0xFF, as its Join Priority (JP) in its IEEE802.15.4e Extended Beacons (EB). We'll note that the JP is now specified between 0 and 0x3F leaving 2 bits in the octet unused in the IEEE802.15.4e specification. After consultation with IEEE authors, it was asserted that 6TiSCH can make a full use of the octet to carry an integer value up to 0xFF.

A node that reads a Join Priority of less than 0xFF should join the neighbor with the lesser Join Priority and use it as time parent. If the node is configured to serve as time parent, then the node should join the TSGI, obtain a Rank in that Instance and start advertising its own DagRank in the TSGI as its Join Priority in its EBs.

7.4. SlotFrames and Priorities

6TiSCH enables in essence the capability to use IPv6 over a MAC layer that enables to schedule some of the transmissions. In order to ensure that the medium if free of contending packets when time arrives for a scheduled transmission, a window of time is defined around the scheduled transmission time where the medium must be free of contending energy.

One simple way to obtain such a window is to format time and frequencies in cells of transmission of equal duration. This is the method that is adopted in IEEE802.15.4e TSCH as well as the Long Term Evolution (LTE) of cellular networks.

In order to describe that formatting of time and frequencies, the 6TiSCH architecture defines a global concept that is called a Channel Distribution and Usage (CDU) matrix; a CDU matrix is a matrix of cells with an height equal to the number of available channels (indexed by ChannelOffsets), a timeSlot duration (10-15 milliseconds are typical in 802.15.4e TSCH) and a width (in timeSlots) that is the period of the network scheduling operation (indexed by slotOffsets) for that CDU matrix.

A CDU matrix iterates over and over with a pseudo-random rotation from an epoch time. In a given network, there might be multiple CDU matrices that operate with different width, so they have different durations and represent different periodic operations. It is RECOMMENDED that all CDU matrices in a 6TiSCH domain operate with the same cell duration and are aligned, so as to optimize the chances of interferences from slotted-aloha operations. The knowledge of the CDU matrices is shared between all the nodes and used in particular to define slotFrames.

A slotFrame is a MAC-level abstraction that is common to all nodes and contains a series of timeSlots of equal length and precedence. It is characterized by a slotFrame_ID, and a slotFrame_size. A slotFrame aligns to a CDU matrix for its parameters, such as number and duration of timeSlots.

Multiple slotFrames can coexist in a node schedule, i.e., a node can have multiple activities scheduled in different slotFrames, based on the precedence of the 6TiSCH topologies. The slotFrames may be aligned to different CDU matrices and thus have different width. There is typically one slotFrame for scheduled traffic that has the highest precedence and one or more slotFrame(s) for RPL traffic. The timeSlots in the slotFrame are indexed by the SlotOffset; the first cell is at SlotOffset 0.

A 6TISCH Instance is associated to one slotFrame. A slotFrame may be shared by multiple Instances of equal relative precedence. Within an Instance, 6top uses priority queues to manage concurrent data flows of different priorities within an Instance and between Instances of a same precedence, associated to a given IPv6 link and a given bundle of TX-cells. When a packet is received from an higher layer for transmission, 6top inserts that packet in the outgoing queue which matches the packet best (DSCP can therefore be used). At each scheduled transmit slot, 6top looks for the frame in all the outgoing queues that best matches the cells. If a frame is found, it is given to the TSCH MAC for transmission.

7.5. Distributing the reservation of cells

6TiSCH expects a high degree of scalability together with a distributed routing functionality based on RPL. To achieve this goal, the spectrum must be allocated in a way that allows for spatial reuse between zones that will not interfere with one another. In a large and spatially distributed network, a 6TiSCH node is often in a good position to determine usage of spectrum in its vicinity.

Use cases for distributed routing are often associated with a statistical distribution of best-effort traffic with variable needs for bandwidth on each individual link. With 6TiSCH, the link abstraction is implemented as a bundle of cells; the size of a bundle is optimal when both the energy wasted idle listening and the packet drops due to congestion loss are minimized. This can be maintained if the number of cells in a bundle is adapted dynamically, and with enough reactivity, to match the variations of best-effort traffic. In turn, the agility to fulfil the needs for additional cells improves when the number of interactions with other devices and the protocol latencies are minimized.

6TiSCH limits that interaction to RPL parents that will only negotiate with other RPL parents, and performs that negotiation by groups of cells as opposed to individual cells. The 6TiSCH architecture allows RPL parents to adjust dynamically, and independently from the PCE, the amount of bandwidth that is used to communicate between themselves and their children, in both directions; to that effect, an allocation mechanism enables a RPL parent to obtain the exclusive use of a portion of a CDU matrix within its interference domain.

The 6TiSCH architecture introduces the concept of chunks [I-D.ietf-6tisch-terminology]) to operate such spectrum distribution for a whole group of cells at a time. The CDU matrix is formatted into a set of chunks, each of them identified uniquely by a chunk-ID. The knowledge of this formatting is shared between all the nodes in a 6TiSCH network. 6TiSCH also defines the process of chunk ownership appropriation whereby a RPL parent discovers a chunk that is not used in its interference domain (e.g lack of energy detected in reference cells in that chunk); then claims the chunk, and then defends it in case another RPL parent would attempt to appropriate it while it is in use. The chunks is the basic unit of ownership that is used in that process.

```
             +-----+-----+-----+-----+-----+-----+-----+     +-----+
chan.Off. 0  |chnkA|chnkP|chnk7|chnkO|chnk2|chnkK|chnk1| ... |chnkZ|
             +-----+-----+-----+-----+-----+-----+-----+     +-----+
chan.Off. 1  |chnkB|chnkQ|chnkA|chnkP|chnk3|chnkL|chnk2| ... |chnk1|
             +-----+-----+-----+-----+-----+-----+-----+     +-----+
               ...
             +-----+-----+-----+-----+-----+-----+-----+     +-----+
chan.Off. 15 |chnkO|chnk6|chnkN|chnk1|chnkJ|chnkZ|chnkI| ... |chnkG|
             +-----+-----+-----+-----+-----+-----+-----+     +-----+
                0     1     2     3     4     5     6          M
```

Figure 8: CDU matrix Partitioning in Chunks

As a result of the process of chunk ownership appropriation, the RPL parent has exclusive authority to decide which cell in the appropriated chunk can be used by which node in its interference domain. In other words, it is implicitly delegated the right to manage the portion of the CDU matrix that is represented by the chunk. The RPL parent may thus orchestrate which transmissions occur in any of the cells in the chunk, by allocating cells from the chunk to any form of communication (unicast, multicast) in any direction between itself and its children. Initially, those cells are added to the heap of free cells, then dynamically placed into existing

bundles, in new bundles, or allocated opportunistically for one transmission.

The appropriation of a chunk can also be requested explicitly by the PCE to any node. In that case, the node still may need to perform the appropriation process to validate that no other node has claimed that chunk already. After a successful appropriation, the PCE owns the cells in that chunk, and may use them as hard cells to set up tracks.

8. Schedule Management Mechanisms

6TiSCH uses 4 paradigms to manage the TSCH schedule of the LLN nodes: Static Scheduling, neighbor-to-neighbor Scheduling, remote monitoring and scheduling management, and Hop-by-hop scheduling. Multiple mechanisms are defined that implement the associated Interaction Models, and can be combined and used in the same LLN. Which mechanism(s) to use depends on application requirements.

8.1. Minimal Static Scheduling

In the simplest instantiation of a 6TiSCH network, a common fixed schedule may be shared by all nodes in the network. Cells are shared, and nodes contend for slot access in a slotted aloha manner.

A static TSCH schedule can be used to bootstrap a network, as an initial phase during implementation, or as a fall-back mechanism in case of network malfunction. This scheduled can be preconfigured or learnt by a node when joining the network. Regardless, the schedule remains unchanged after the node has joined a network. The Routing Protocol for LLNs (RPL) is used on the resulting network. This "minimal" scheduling mechanism that implements this paradigm is detailed in [I-D.ietf-6tisch-minimal].

8.2. Neighbor-to-neighbor Scheduling

In the simplest instantiation of a 6TiSCH network described in Section 8.1, nodes may expect a packet at any cell in the schedule and will waste energy idle listening. In a more complex instantiation of a 6TiSCH network, a matching portion of the schedule is established between peers to reflect the observed amount of transmissions between those nodes. The aggregation of the cells between a node and a peer forms a bundle that the 6top layer uses to implement the abstraction of a link for IP. The bandwidth on that link is proportional to the number of cells in the bundle.

If the size of a bundle is configured to fit an average amount of bandwidth, peak emissions will be destroyed. If the size is

configured to allow for peak emissions, energy is be wasted idle listening.

In the most efficient instantiation of a 6TiSCH network, the size of the bundles that implement the links may be changed dynamically in order to adapt to the need of end-to-end flows routed by RPL. An optional On-The-Fly (OTF) component may be used to monitor bandwidth usage and perform requests for dynamic allocation by the 6top sublayer. The OTF component is not part of the 6top sublayer. It may be collocated on the same device or may be partially or fully offloaded to an external system.

The 6top sublayer [I-D.wang-6tisch-6top-sublayer] defines a protocol for neighbor nodes to reserve soft cells to one another. Because this reservation is done without global knowledge of the schedule of nodes in the LLN, scheduling collisions are possible. 6top defines a monitoring process which continuously tracks the packet delivery ratio of soft cells. It uses these statistics to trigger the relocation of a soft cell in the schedule, using a negotiation protocol between the neighbors nodes communicating over that cell.

Monitoring and relocation is done in the 6top layer. For the upper layer, the connection between two neighbor nodes appears as an number of cells. Depending on traffic requirements, the upper layer can request 6top to add or delete a number of cells scheduled to a particular neighbor, without being responsible for choosing the exact slotOffset/channelOffset of those cells.

8.3. Remote Monitoring and Schedule Management

The 6top interface document [I-D.ietf-6tisch-6top-interface] specifies the generic data model that can be used to monitor and manage resources at the 6top sublayer. Abstract methods are suggested for use by a management entity in the device. The data model also enables remote control operations on the 6top sublayer.

Being able to interact with the 6top sublayer of a node multiple hops away can be used for monitoring, scheduling, or a combination of both. The architecture supports variations on the deployment model, and focuses on the flows rather than whether there is a proxy or a translational operation on the way.

[I-D.ietf-6tisch-coap] defines an mapping of 6top's set of commands described in [I-D.ietf-6tisch-6top-interface] to CoAP resources. This allows an entity to interact with the 6top layer of a node that is multiple hops away in a RESTful fashion.

[I-D.ietf-6tisch-coap] defines a basic set CoAP resources and associated RESTful access methods (GET/PUT/POST/DELETE). The payload (body) of the CoAP messages is encoded using the CBOR format. The draft also defines the concept of "profiles" to allow for future or specific extensions, as well as a mechanism for a CoAP client to discover the profiles installed on a node.

The entity issuing the CoAP requests can be a central scheduling entity (e.g. a PCE), a node multiple hops away with the authority to modify the TSCH schedule (e.g. the head of a local cluster), or a external device monitoring the overall state of the network (e.g. NME). The architecture allows for different types of interactions between this CoAP client and a node in the network:

8.4. Hop-by-hop Scheduling

A node can reserve a track to a destination node multiple hops away by installing soft cells at each intermediate node. This forms a track of soft cells. It is the responsibility of the 6top sublayer of each node on the track to monitor these soft cells and trigger relocation when needed.

This hop-by-hop reservation mechanism is similar to [RFC2119] and [RFC5974]. The protocol for a node to trigger hop-by-hop scheduling is not yet defined.

9. Centralized vs. Distributed Routing

6TiSCH supports a mixed model of centralized routes and distributed routes. Centralized routes can for example computed by a entity such as a PCE. Distributed routes are computed by RPL.

Both methods may inject routes in the Routing Tables of the 6TiSCH routers. In either case, each route is associated with a 6TiSCH topology that can be a RPL Instance topology or a track. The 6TiSCH topology is indexed by a Instance ID, in a format that reuses the RPLInstanceID as defined in RPL [RFC6550].

Both RPL and PCE rely on shared sources such as policies to define Global and Local RPLInstanceIDs that can be used by either method. It is possible for centralized and distributed routing to share a same topology. Generally they will operate in different slotFrames, and centralized routes will be used for scheduled traffic and will have precedence over distributed routes in case of conflict between the slotFrames.

9.1. Packet Marking and Handling

All packets inside a 6TiSCH domain MUST carry the Instance ID that identifies the 6TiSCH topology that is to be used for routing and forwarding that packet. The location of that information MUST be the same for all packets forwarded inside the domain.

For packets that are routed by RPL [RFC6550], that information is the RPLInstanceID that is carried as part of the RPL Packet Information, which is defined in section 11.2 "Loop Avoidance and Detection".

At the time of this writing, there are 2 methods to transport the RPL Packet Information in an IPv6 packet, either in a IPv6 Hop-By-Hop Header, or encoded in a compressed fashion in the IPv6 Flow Label.

The former method places a RPL option [RFC6553] in the IPv6 Hop-By-Hop Header. It MUST be used if at least one RPL Instance uses a MinHopRankIncrease that is less than DEFAULT_MIN_HOP_RANK_INCREASE (defined to 256 in [RFC6550]), which bars the capability to compress the SenderRank in the RPL Packet Information to a single octet. If that is not the case, it is RECOMMENDED to use the latter method of encoding the RPL Packet Information in the Flow Label, which is specified in [I-D.thubert-6man-flow-label-for-rpl].

Either way, the method and format used for encoding the RPLInstanceID is generalized to all 6TiSCH topological Instances, which include both RPL Instances and Tracks.

10. IANA Considerations

This specification does not require IANA action.

11. Security Considerations

This specification is not found to introduce new security threat.

12. Contributors

The editors and authors wish to recognize the contribution of

Xavier Vilajosana who lead the design of the minimal support with RPL and contributed deeply to the 6top design.

Qin Wang who lead the design of the 6top sublayer and contributed related text that was moved and/or adapted in this document.

13. Acknowledgements

This specification is the result interactions in particular during the 6TiSCH (bi)Weekly call. The authors wish to thank: Alaeddine Weslati, Alfredo Grieco, Bert Greevenbosch, Cedric Adjih, Diego Dujovne, Dominique Barthel, Elvis Vogli, Geraldine Texier, Giuseppe Piro, Guillaume Gaillard, Herman Storey, Ines Robles, Jonathan Simon, Kazushi Muraoka, Ken Bannister, Kuor Hsin Chang, Laurent Toutain, Maik Seewald, Maria Rita Palattella, Michael Behringer, Michael Richardson, Nancy Cam Winget, Nicola Accettura, Nicolas Montavont, Oleg Hahm, Pat Kinney, Patrick Wetterwald, Paul Duffy, Peter van der Stock, Pieter de Mil, Pouria Zand, Rouhollah Nabati, Rafa Marin-Lopez, Raghuram Sudhaakar, Rene Struik, Sedat Gormus, Shitanshu Shah, Steve Simlo, Subir Das, Tengfei Chang, Tina Tsou, Tom Phinney, Xavier Lagrange and Yoshihiro Ohba for their various participation.

14. References

14.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC3444] Pras, A. and J. Schoenwaelder, "On the Difference between Information Models and Data Models", RFC 3444, January 2003.

[RFC4080] Hancock, R., Karagiannis, G., Loughney, J., and S. Van den Bosch, "Next Steps in Signaling (NSIS): Framework", RFC 4080, June 2005.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.

[RFC4389] Thaler, D., Talwar, M., and C. Patel, "Neighbor Discovery Proxies (ND Proxy)", RFC 4389, April 2006.

[RFC4429] Moore, N., "Optimistic Duplicate Address Detection (DAD) for IPv6", RFC 4429, April 2006.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.

[RFC4903] Thaler, D., "Multi-Link Subnet Issues", RFC 4903, June 2007.

[RFC4919] Kushalnagar, N., Montenegro, G., and C. Schumacher, "IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals", RFC 4919, August 2007.

[RFC5191] Forsberg, D., Ohba, Y., Patil, B., Tschofenig, H., and A. Yegin, "Protocol for Carrying Authentication for Network Access (PANA)", RFC 5191, May 2008.

[RFC5889] Baccelli, E. and M. Townsley, "IP Addressing Model in Ad Hoc Networks", RFC 5889, September 2010.

[RFC5974] Manner, J., Karagiannis, G., and A. McDonald, "NSIS Signaling Layer Protocol (NSLP) for Quality-of-Service Signaling", RFC 5974, October 2010.

[RFC6275] Perkins, C., Johnson, D., and J. Arkko, "Mobility Support in IPv6", RFC 6275, July 2011.

[RFC6282] Hui, J. and P. Thubert, "Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks", RFC 6282, September 2011.

[RFC6550] Winter, T., Thubert, P., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, JP., and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", RFC 6550, March 2012.

[RFC6552] Thubert, P., "Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL)", RFC 6552, March 2012.

[RFC6553] Hui, J. and JP. Vasseur, "The Routing Protocol for Low-Power and Lossy Networks (RPL) Option for Carrying RPL Information in Data-Plane Datagrams", RFC 6553, March 2012.

[RFC6620] Nordmark, E., Bagnulo, M., and E. Levy-Abegnoli, "FCFS SAVI: First-Come, First-Served Source Address Validation Improvement for Locally Assigned IPv6 Addresses", RFC 6620, May 2012.

[RFC6775] Shelby, Z., Chakrabarti, S., Nordmark, E., and C. Bormann, "Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)", RFC 6775, November 2012.

14.2. Informative References

[I-D.chakrabarti-nordmark-6man-efficient-nd]
Chakrabarti, S., Nordmark, E., Thubert, P., and M. Wasserman, "IPv6 Neighbor Discovery Optimizations for Wired and Wireless Networks", draft-chakrabarti-nordmark-6man-efficient-nd-05 (work in progress), February 2014.

[I-D.ietf-6tisch-6top-interface]
Wang, Q., Vilajosana, X., and T. Watteyne, "6TiSCH Operation Sublayer (6top) Interface", draft-ietf-6tisch-6top-interface-00 (work in progress), March 2014.

[I-D.ietf-6tisch-coap]
Sudhaakar, R. and P. Zand, "6TiSCH Resource Management and Interaction using CoAP", draft-ietf-6tisch-coap-00 (work in progress), May 2014.

[I-D.ietf-6tisch-minimal]
Vilajosana, X. and K. Pister, "Minimal 6TiSCH Configuration", draft-ietf-6tisch-minimal-00 (work in progress), November 2013.

[I-D.ietf-6tisch-terminology]
Palattella, M., Thubert, P., Watteyne, T., and Q. Wang, "Terminology in IPv6 over the TSCH mode of IEEE 802.15.4e", draft-ietf-6tisch-terminology-01 (work in progress), February 2014.

[I-D.ietf-6tisch-tsch]
Watteyne, T., Palattella, M., and L. Grieco, "Using IEEE802.15.4e TSCH in an LLN context: Overview, Problem Statement and Goals", draft-ietf-6tisch-tsch-00 (work in progress), November 2013.

[I-D.ietf-ipv6-multilink-subnets]
Thaler, D. and C. Huitema, "Multi-link Subnet Support in IPv6", draft-ietf-ipv6-multilink-subnets-00 (work in progress), July 2002.

[I-D.ietf-roll-rpl-industrial-applicability]
Phinney, T., Thubert, P., and R. Assimiti, "RPL applicability in industrial networks", draft-ietf-roll-rpl-industrial-applicability-02 (work in progress), October 2013.

[I-D.svshah-tsvwg-lln-diffserv-recommendations]
Shah, S. and P. Thubert, "Differentiated Service Class Recommendations for LLN Traffic", draft-svshah-tsvwg-lln-diffserv-recommendations-02 (work in progress), February 2014.

[I-D.thubert-6lowpan-backbone-router]
Thubert, P., "6LoWPAN Backbone Router", draft-thubert-6lowpan-backbone-router-03 (work in progress), February 2013.

[I-D.thubert-6man-flow-label-for-rpl]
Thubert, P., "The IPv6 Flow Label within a RPL domain", draft-thubert-6man-flow-label-for-rpl-03 (work in progress), May 2014.

[I-D.thubert-roll-forwarding-frags]
Thubert, P. and J. Hui, "LLN Fragment Forwarding and Recovery", draft-thubert-roll-forwarding-frags-02 (work in progress), September 2013.

[I-D.wang-6tisch-6top-sublayer]
Wang, Q., Vilajosana, X., and T. Watteyne, "6TiSCH Operation Sublayer (6top)", draft-wang-6tisch-6top-sublayer-00 (work in progress), February 2014.

14.3. External Informative References

[HART] www.hartcomm.org, "Highway Addressable Remote Transducer, a group of specifications for industrial process and control devices administered by the HART Foundation", .

[IEEE802.1TSNTG]
IEEE Standards Association, "IEEE 802.1 Time-Sensitive Networks Task Group", March 2013, <http://www.ieee802.org/1/pages/avbridges.html>.

[IEEE802154e]
IEEE standard for Information Technology, "IEEE std. 802.15.4e, Part. 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendament 1: MAC sublayer", April 2012.

[ISA100.11a]
ISA/ANSI, "Wireless Systems for Industrial Automation: Process Control and Related Applications - ISA100.11a-2011 - IEC 62734", 2011, <http://www.isa.org/Community/SP100WirelessSystemsforAutomation>.

[WirelessHART]
www.hartcomm.org, "Industrial Communication Networks - Wireless Communication Network and Communication Profiles - WirelessHART - IEC 62591", 2010.

Authors' Addresses

Pascal Thubert (editor)
Cisco Systems, Inc
Building D
45 Allee des Ormes - BP1200
MOUGINS - Sophia Antipolis 06254
FRANCE

Phone: +33 497 23 26 34
Email: pthubert@cisco.com

Thomas Watteyne
Linear Technology, Dust Networks Product Group
30695 Huntwood Avenue
Hayward, CA 94544
USA

Phone: +1 (510) 400-2978
Email: twatteyne@linear.com

Robert Assimiti
Centero
961 Indian Hills Parkway
Marietta, GA 30068
USA

Phone: +1 404 461 9614
Email: robert.assimiti@centerotech.com