IETF 88 - Vancouver

Chairs:
Pascal Thubert
Thomas Watteyne
Mailing list:
6tisch@ietf.org
Jabber:
6tisch@jabber.ietf.org
Etherpad for minutes:
http://etherpad.tools.ietf.org:9000/p/6tisch

IPv6 over the TSCH mode of IEEE 802.15.4e

Note Well

This summary is only meant to point you in the right direction, and doesn't have all the nuances. The IETF's IPR Policy is set forth in BCP 79; please read it carefully.

The brief summary:

- ❖ **By participating with the IETF, you agree to follow IETF processes.**
- ❖ **If you are aware that a contribution of yours (something you write, say, or discuss in any IETF context) is covered by patents or patent applications, you must disclose that fact.**
- ❖ **You understand that meetings might be recorded, broadcast, photographed, and publicly archived.**

For further information, talk to a chair, ask an Area Director, or review the following:

BCP 9 (on the Internet Standards Process)
BCP 25 (on the Working Group processes)
BCP 78 (on the IETF Trust)
BCP 79 (on Intellectual Property Rights in the IETF)

Reminder:

Minutes are taken *

This meeting is recorded **

Presence is logged ***

* Scribe; please contribute online to the minutes at http://etherpad.tools.ietf.org:9000/p/6tisch?useMonospaceFont=true

** Recordings and Minutes are public and may be subject to discovery in the event of litigation.

*** Please make sure you sign the blue sheets

Administrivia

- Blue Sheets
- Scribes (Thanks!)
  - Xavi Vilajosana
  - Dominique Barthel
- Jabber (Thanks!)
  - Michael Richardson

Objectives

- First WG meeting
- Sort through the drafts, suggest reshuffling to prepare for WG docs
- Obtain a general direction in which the WG wants to go with respect to any further standardization
- Elect initial set of WG docs, if any

Agenda

Intro and Status	[10min]
– Agenda Bashing	
– draft-palattella-6tisch-terminology	
– 6TiSCH charter recap	
Architecture	[15min]
– <draft-watteyne-6tisch-tsch-00>	(Thomas Watteyne)
– <draft-thubert-6tisch-architecture-01>	(Pascal Thubert)
Information and Data Models	[25min]
– <draft-wang-6tisch-6top-00>	(Xavi Vilajosana)
– <draft-sudhaakar-6tisch-coap-00>	(Pouria Zand)
– Coverage gap analysis vs. charter	(Dominique Barthel)
Minimal RPL support	[10min]
– <draft-vilajosana-6tisch-minimal-00>	(Xavi Vilajosana)
Call for draft adoption	[10min] (Chairs)
Unchartered drafts if time permits	[15min]
– <draft-ohba-6tisch-security-00>	(Yoshihiro Ohba)
– Overview discussion on slot allocation principles	
– <draft-piro-6tisch-security-issues-00>	
AOB	[5min]

draft-palattella-6tisch-terminology-00

Maria Rita Palattella (Ed.)
Pascal Thubert
Thomas Watteyne
Qin Wang

draft-palattella-6tisch-terminology-00

- Status:
  - New revision of draft-palattella-**6tsch**-terminology-**01**
  - Latest version published 10/11/2013 http://tools.ietf.org/html/draft-palattella-6tisch-terminology-00
  - Running version at https://bitbucket.org/6tisch/draft-palattella-6tisch-terminology
- Changes since IETF87
  - Additional terms (see next slides)

Goals

1. Provide **additional terminology** elements to cover terms new to the context of TSCH wireless networks and other deterministic networks.
2. **Avoid colliding definitions** across standards and standardization bodies. It does not reuse terms from IEEE802.15.4e std. (e.g., link, and path)
3. Extends **ROLL terminology** [I-D.ietf-roll-terminology], and refers to terms from **RFC3444**, **RFC6550** and **RFC6552**.

Defined Terms

- 6TiSCH
- 6F
- 6top
- 6top Data Convey Model
- ASN
- Blacklisting
- BBR
- Bundle
- Cell
- ChannelOffset
- Communication Paradigm
- Dedicated Cell
- Distributed cell reservation
- Distributed track reservation
- EB
- FF
- GMPLS
- Hard Cell
- Hopping Sequence
- IE
- I-MUX module
- Interaction Model
- KMP
- LBR
- Link
- Logical Cell
- MAC
- MUX module
- NEAR
- NME
- PANA
- PCE
- PCE cell reservation
- PCE track reservation
- QoS
- SA
- Shared Cell
- SlotOffset
- Slotframe
- Soft Cell
- TF
- Timeslot
- Time Source Neighbor
- Track
- Track ID
- TSCH
- TSCH Schedule

Charter Recap

Description of Working Group

The Working Group will focus on enabling **IPv6** over the **TSCH mode of the IEEE802.15.4e standard**. The extent of the problem space for the WG is **one or more LLNs**, eventually federated through a common backbone link via one or more LLN Border Routers (**LBRs**).

The WG will rely on, and if necessary extend, existing mechanisms for authenticating LBRs. Initially, the WG will **limit its scope to distributed routing over a static schedule**. In that case, a node's schedule can be either preconfigured, or learnt by a node when joining the network, but it remains unchanged after the node has joined a network.

The Routing Protocol for LLNs (**RPL**) is used on the resulting network. The WG will interface with other appropriate groups in the IETF Internet, Operations and Management, Routing and Security areas.

Work Item 1

Produce "**6TiSCH architecture**" to describe the design of 6TiSCH networks. This document will highlight the different architectural blocks and signalling flows, including the operation of the network in the presence of **multiple LBRs**. Initially, the document will focus on **distributed routing operation over a static TSCH schedule**.

Work Item 2

Produce an **Information Model** containing the management requirements of a 6TiSCH node. This includes describing how an entity can manage the TSCH schedule on a 6TiSCH node, and query timeslot information from that node. A data model mapping for an existing protocol (such as Concise Binary Object Representation (**CBOR**) over the Constrained Application Protocol (**CoAP**)) will be provided.

Work Item 3

Produce "**Minimal 6TiSCH Configuration**" defining how to build a 6TiSCH network using the Routing Protocol for LLNs (**RPL**) and a **static TSCH schedule**. It is expected that RPL and the Objective Function 0 (**OF0**) will be reused as-is.

The work will include a **best practice** configuration for RPL and OF0 operation over the **static schedule**. Based on that experience the group may produce a requirements draft for OF0 extensions, to be studied in ROLL.

draft-watteyne-6tisch-tsch-00

Thomas Watteyne (Ed.)
Maria Rita Palattella
Luigi Alfredo Grieco

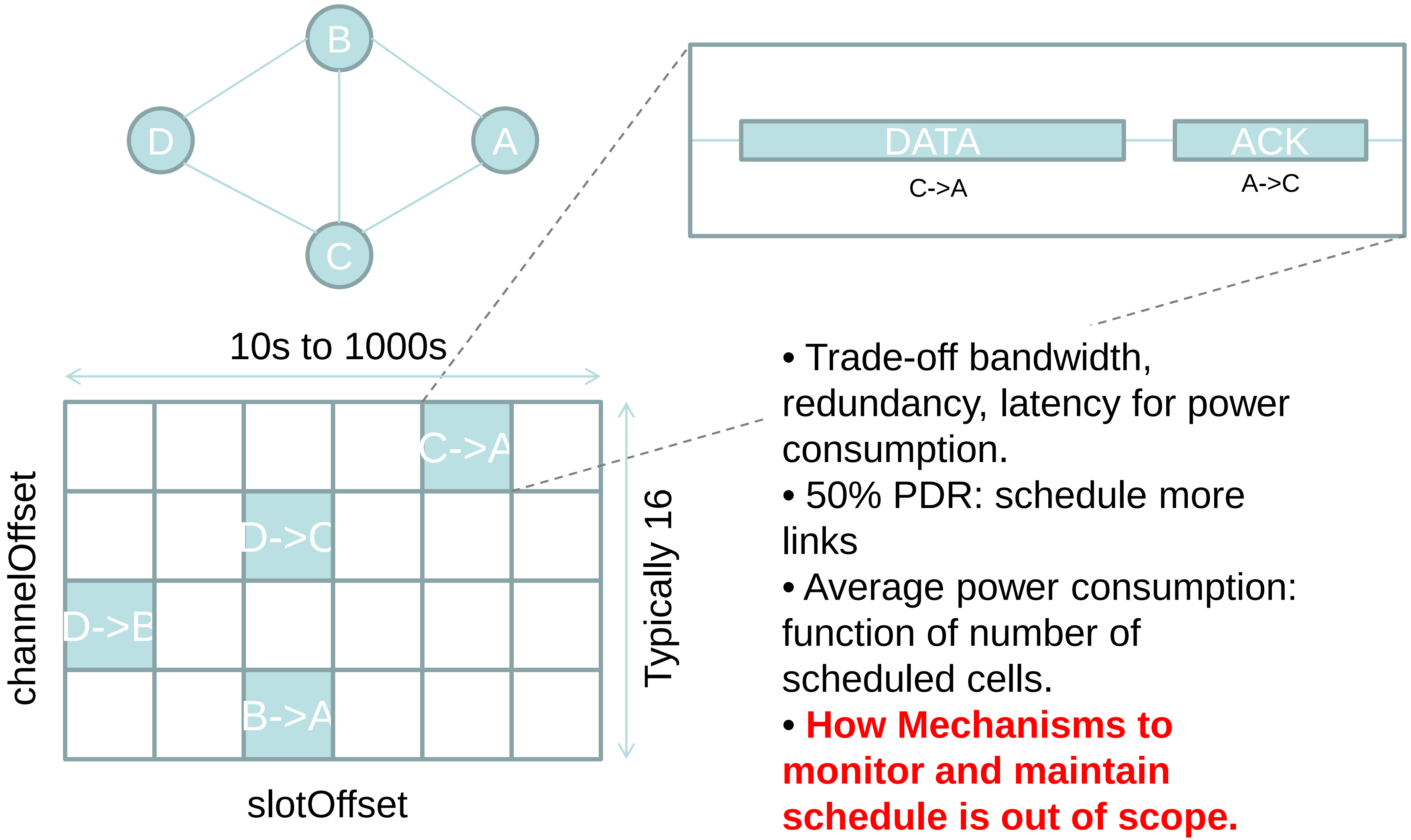
draft-watteyne-6tisch-tsch-00

- Status:
  - New revision of draft-watteyne-**6tsch**-tsch-lln-context-**02**
  - Latest version published 10/20/2013 http://tools.ietf.org/html/draft-watteyne-6tisch-tsch-00
  - Running version at https://bitbucket.org/6tisch/draft-watteyne-6tisch-tsch
- Changes since IETF87
  - minor rewording
  - typos

Goals

1. Provide overview of IEEE802.15.4e TSCH
2. List problems and goals:
   - What does TSCH not do?
   - What should 6TiSCH provide?

Timeslotted Channel Hopping



- Trade-off bandwidth, redundancy, latency for power consumption.
- 50% PDR: schedule more links
- Average power consumption: function of number of scheduled cells.
- **How Mechanisms to monitor and maintain schedule is out of scope.**

draft-thubert-6tisch-architecture-01

Pascal Thubert (Ed.)
Thomas Watteyne
Robert Assimiti

draft-thubert-6tisch-architecture-01

- Status
  - New revision of draft-thubert-**6tsch**-architecture-**02**
  - Latest version published 10/21/2013 http://tools.ietf.org/html/draft-thubert-6tisch-architecture-00
  - Running version at https://bitbucket.org/6tisch/draft-thubert-6tisch-architecture
- Changes since IETF87
  - Reorganization (see next slide)
- Open Questions
  - Participation to information model and paradigms

draft-wang-6tisch-6top-00

Qin Wang (Ed.)

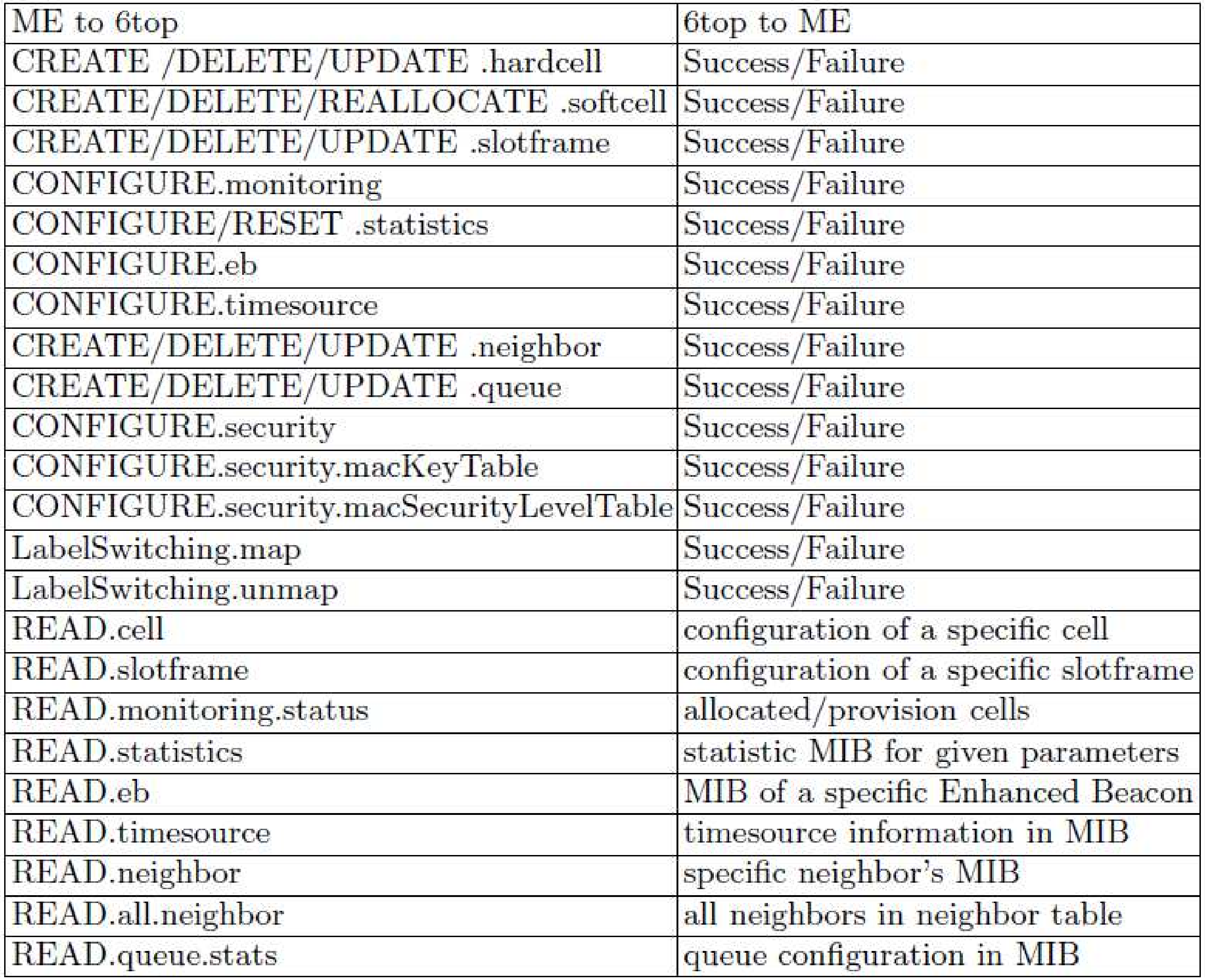
Xavi Vilajosana

Thomas Watteyne

draft-wang-6tisch-6top-00

- Status:
  - New revision of draft-wang-**6tsch**-6top-00
  - Latest version published 10/20/2013 http://tools.ietf.org/html/draft-wang-6tisch-6top-00
  - Running version at https://bitbucket.org/6tisch/draft-wang-6tisch-6top
- Changes since IETF87
  - Make terminology consistent
  - Remove behavior table for each command
  - Make security related commands refer to IEEE802.15.4-2011
  - Refine descriptions
  - Typos

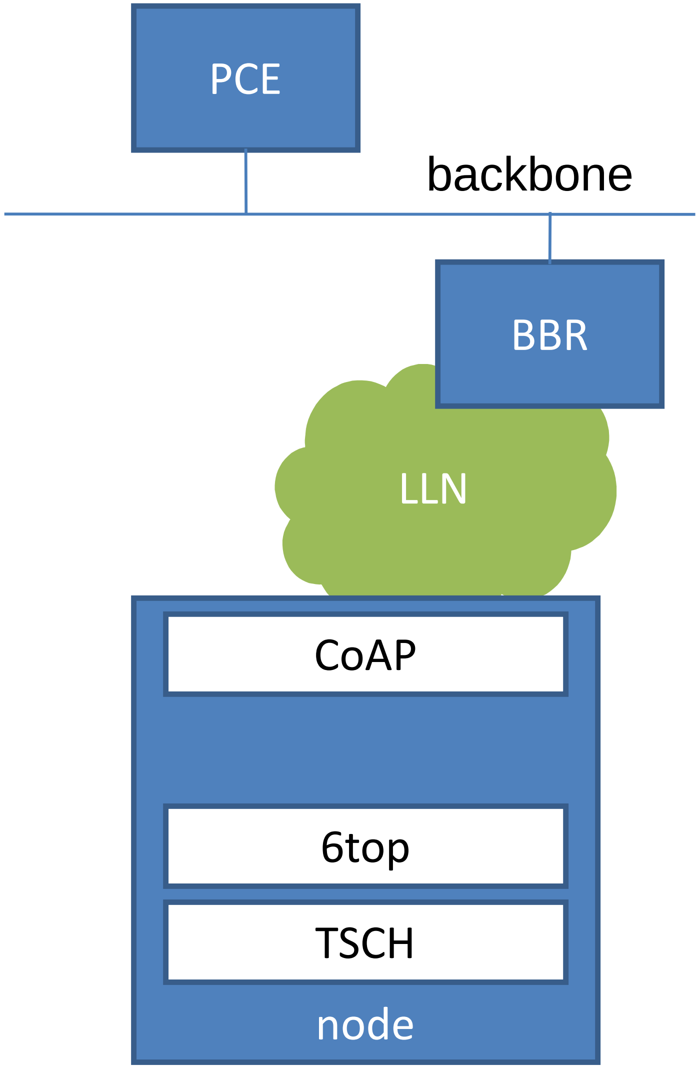
Commands



ME to 6top	6top to ME
CREATE /DELETE/UPDATE .hardcell	Success/Failure
CREATE/DELETE/REALLOCATE .softcell	Success/Failure
CREATE/DELETE/UPDATE .slotframe	Success/Failure
CONFIGURE.monitoring	Success/Failure
CONFIGURE/RESET .statistics	Success/Failure
CONFIGURE.eb	Success/Failure
CONFIGURE.timesource	Success/Failure
CREATE/DELETE/UPDATE .neighbor	Success/Failure
CREATE/DELETE/UPDATE .queue	Success/Failure
CONFIGURE.security	Success/Failure
CONFIGURE.security.macKeyTable	Success/Failure
CONFIGURE.security.macSecurityLevelTable	Success/Failure
LabelSwitching.map	Success/Failure
LabelSwitching.unmap	Success/Failure
READ.cell	configuration of a specific cell
READ.slotframe	configuration of a specific slotframe
READ.monitoring.status	allocated/provision cells
READ.statistics	statistic MIB for given parameters
READ.eb	MIB of a specific Enhanced Beacon
READ.timesource	timesource information in MIB
READ.neighbor	specific neighbor's MIB
READ.all.neighbor	all neighbors in neighbor table
READ.queue.stats	queue configuration in MIB

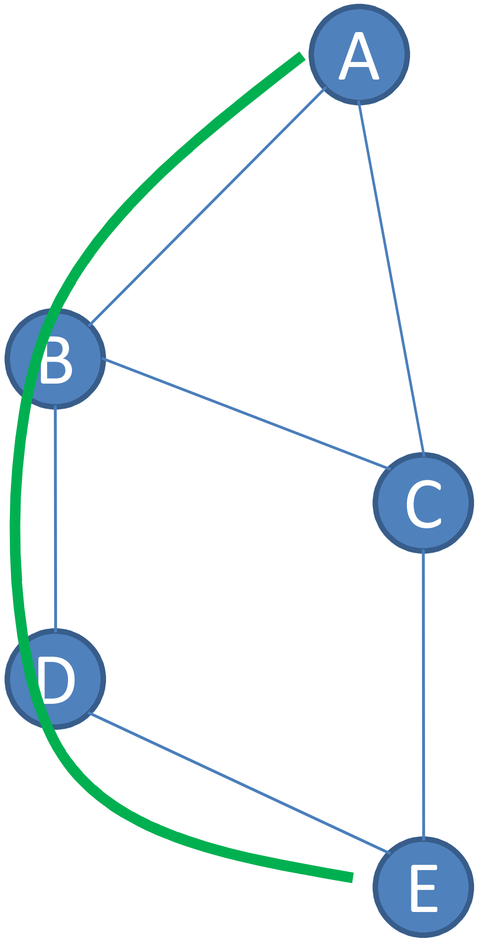
Using 6top with a PCE

- PCE has full knowledge of topology and traffic requirements
- PCE computes schedule
- Communicates with nodes to configure their schedule
- PCE-node protocol
  - e.g. **CoAP**
- PCE typically schedules **hard cells**
- **Charter Scope: define operational API an 6top mechanisms**



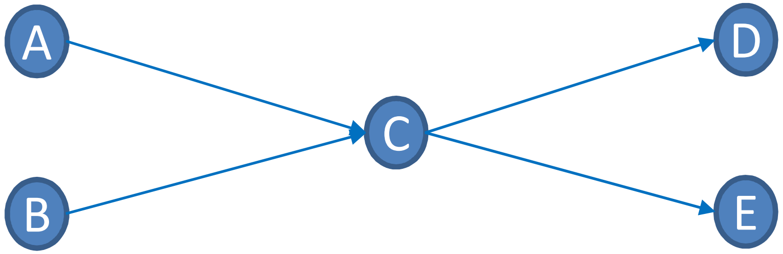
6top with distributed scheduling

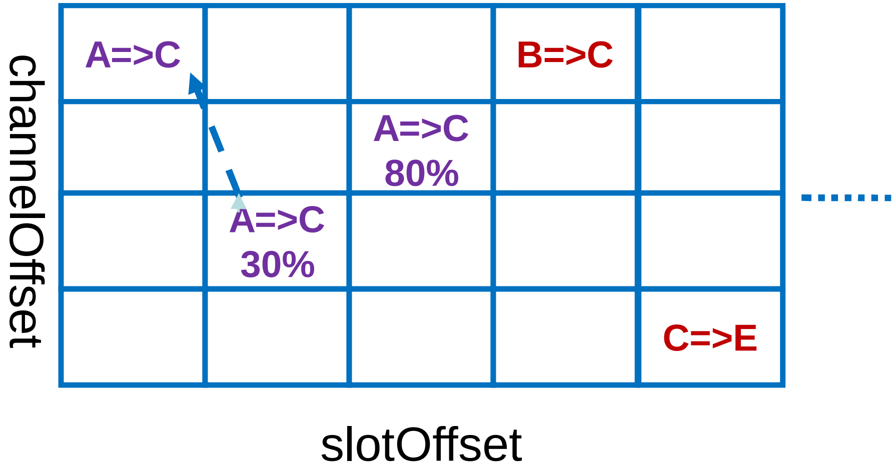
- Distributed scheduling can use RPL routes
- Neighbor schedule bandwidth with each other, rather than explicit cells
  - Soft cells
- 6top monitoring process monitors performance of cells and reschedules the ones that perform bad.
- **Charter Scope: define operational API an 6top mechanisms**



6top Monitoring Process

- Configure 6tus statistics collection to gather e.g. PDR
- Compare PDR for equivalent cells
- If a cell performs bad, reallocate in the schedule
- Reallocation involves negotiation with the neighbor

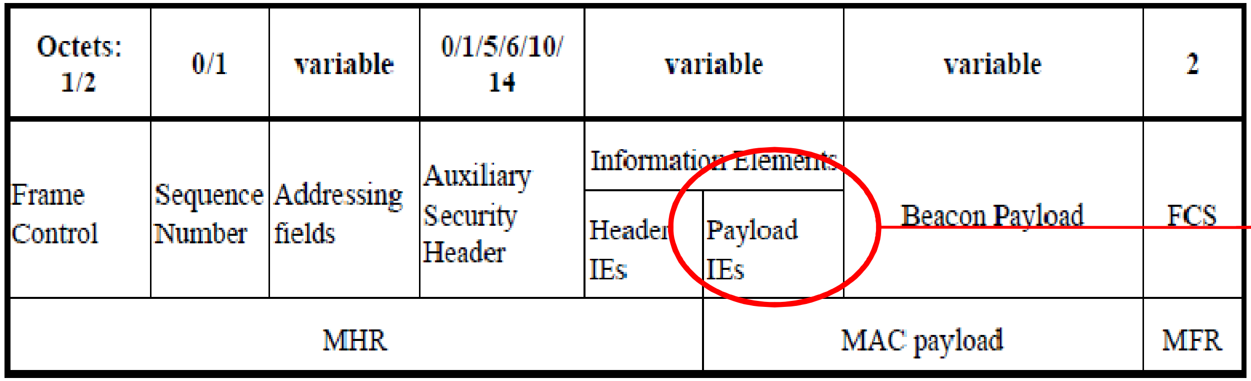




A=>C			B=>C	
		A=>C 80%		
	A=>C 30%			
				C=>E

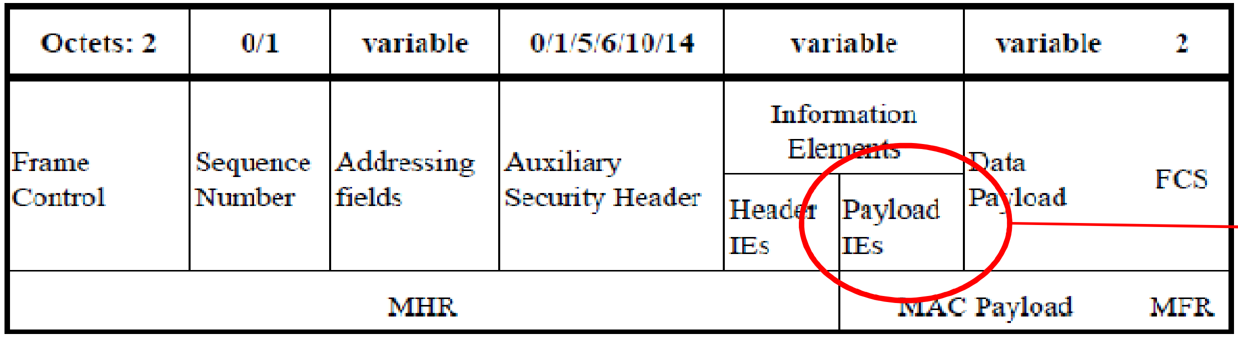
Packets

- IEEE802.15.4e Frame Formats



Octets: 1/2	0/1	variable	0/1/5/6/10/14	variable		variable	2
				Information Elements			
Frame Control	Sequence Number	Addressing fields	Auxiliary Security Header	Header IEs	Payload IEs	Beacon Payload	FCS
MHR					MAC payload		MFR

Figure 40a—Enhanced Beacon frame format



Octets: 2	0/1	variable	0/1/5/6/10/14	variable		variable	2
				Information Elements			
Frame Control	Sequence Number	Addressing fields	Auxiliary Security Header	Header IEs	Payload IEs	Data Payload	FCS
MHR					MAC Payload		MFR

Figure 46—Data frame format

- 6top IE-based Packet formats

TSCH Enhanced Beacon

-Synchronization IE
-Timeslot Template IE
-Channel Hopping IE
-Slotframe and Link IE

Reserve Hard/Soft Cell Request
Reserve Soft Cell Response
Remove Hard/Soft Cell Request

-Opcode IE
-Bandwidth IE
-Track ID IE
-Generic Schedule IE

Issues to be addressed (1/3)

- **How to express Soft/Hard cell of 6top?** In the current 6top draft, we extend 15.4e attribute LinkOptions, adding bit(4) to indicate soft/hard cell. It may result a unclean interface with 15.4e.

- **Rename “Data Convey Model” with something else?**

- **Contents of 6top draft**: Should we divide the current 6top draft into several drafts to make each one more focus?

Issues to be addressed (2/3)

Regarding to commands

■**How to express management interface of 6top with upper layer?** In current draft, we use a set of commands to express the interface of 6top with upper layer. Some of command are associated with message exchange, e.g. CREATE.softcell; but many of them are only associated with the operations on local MIB, e.g. CONFIGURE.eb, READ.statistics. Should we separate the two parts, express them as primitives and SET/GET MIB, respectively? E.g. **using communication primitives and Yang model, respectively?**

■**Description of commands behavior needs to be improved**, especially on clarifying the behavior inside 6top and the behavior of IEEE802.15.4e caused by 6top.

■**Add "lifetime" parameter to CREATE.hardcell and CREATE.softcell command**? From "OTF <->" thread. 6top will remove the hard/soft cells at the end of their lifetime.

Issues to be addressed (3/3)

Regarding to 6top ⇔ 6top, and more

- **Add 6top-level ACK in response to a Delete soft cell and Delete hard cell ?** Maybe the requesting node could indicate it expects an ACK as part of the Opcode IE.

- **Should Deleting hard cells command trigger a Hard Cell Remove Request?** For example when we delete hard cells because the neighbor has disappeared.

- **How to make 6top extensible with profiles?** In the 6top draft, we leave some attributes/functions open to upper layer or application, e.g. “The exact metrics for statistics are out of the scope of this document”, “The policy to select cells corresponding to a Delete soft cell command is out of scope of this document.”. Profile is a way to implement the flexibility and extendibility. We need to define how to make a profile in the next step.

draft-sudhaakar-6tisch-coap-00

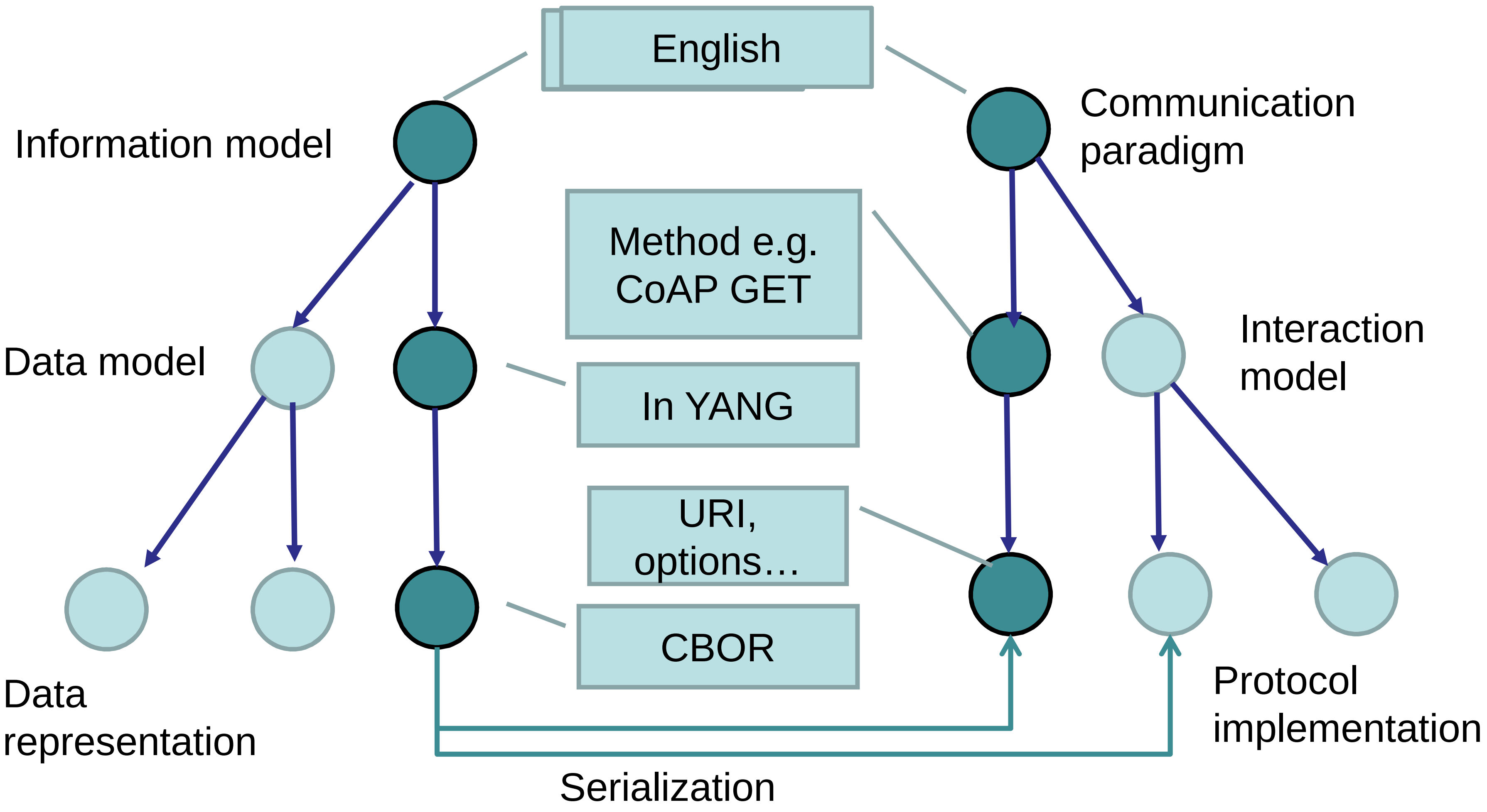
Raghuram S Sudhaakar (Ed.)

Pouria Zand

draft-sudhaakar-6tisch-coap-00

- Status:
  - New draft.
  - Latest version published 10/2/2013 http://tools.ietf.org/html/draft-sudhaakar-6tisch-coap-00
  - Running version at https://bitbucket.org/6tisch/draft-sudhaakar-6tisch-coap
- Changes since IETF87
  - New draft

Subject of the work



English

Information model

Communication paradigm

Method e.g. CoAP GET

Data model

In YANG

Interaction model

URI, options…

CBOR

Data representation

Protocol implementation

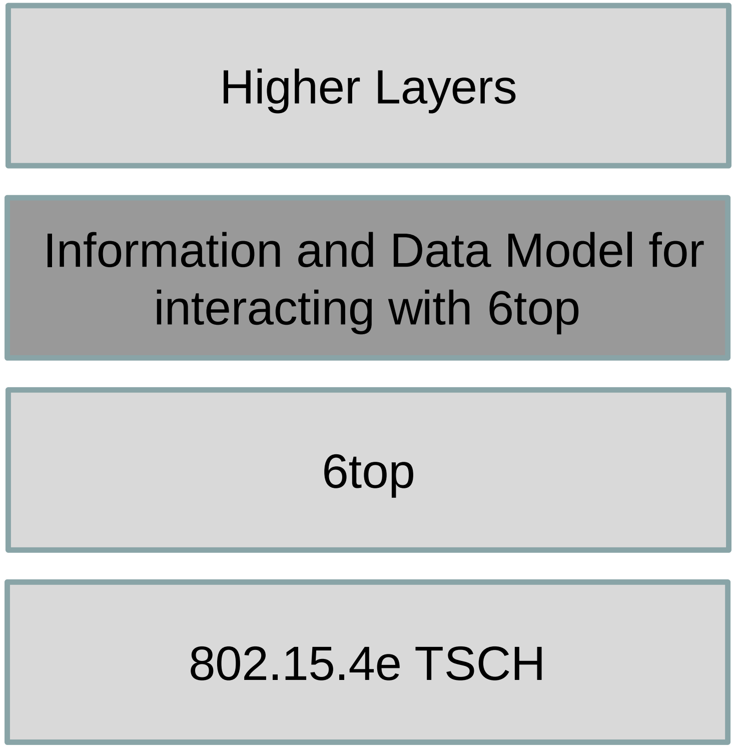
Serialization

Outline

- Data Model definition for CoAP
  - Naming Convention for URI schemes
  - Convention for accessing URIs
  - 6TiSCH Resources
    - Management Resources
    - Informational Resources
    - Message Formats
    - Extensible Resources
  - Example

Introduction

- Logical positioning of layers



Introduction

- 6top defines a set of commands to monitor and manage the TSCH schedule
- We need to define how to interact with 6top, control and modify schedules, monitor parameters etc.
- We need to define a generic data model between monitoring and management entities and the 6top layer
- We need to define a mapping to the 6top commands.
- We also presents a particular implementation of the model based on CoAP and CBOR

Naming Convention for URI schemes

- All URIs naming 6top resources MUST use the 'coap' **scheme**
- The **authority** MUST have the username '6top' and the IP address of 6top node
- The root **path** MUST always start with '6t‘
- Each component of the path SHOULD be of minimum possible length while being self descriptive.

Convention for accessing URIs

Mapping between CoAP methods and 6top commands

CoAP method	6top command	Description
GET	READ	Retrieves 6top resources
POST	CREATE / UPDATE	Creates/Updates a new entry
DELETE	DELETE	Deletes an entry
POST	CONFIGURE	Configures a setting

6TiSCH Resources

- Management Resources

6top management resources and the related URI paths

Name	Accessibility 6top Commands	URI path
Neighbor Table	CREATE/READ/DELETE/UPDATE	6t/Neighbor
Slotframe Table	CREATE/READ/DELETE/UPDATE	6t/slotframe
Cell Table	CREATE/READ/DELETE/UPDATE	6t/Cell
Time Source	CREATE/READ/DELETE/UPDATE	6t/TimeSource
Bundle Table	CREATE/READ/DELETE/UPDATE	6t/Bundle
Track Table	CREATE/READ/DELETE/UPDATE	6t/Track
EB Table	CREATE/READ/DELETE/UPDATE	6t/EB

6TiSCH Resources

- Management Resources

An example about how Neighbor table attributes can be addressed

Field name	URI path
Neighbor Short Addr	6t/Neighbor/ShortAddr
numTx	6t/Neighbor/numTx
numTxAck	6t/Neighbor/numTxAck
numRx	6t/Neighbor/numRx
Neighbor Long Addr	6t/Neighbor/LongAddr
ASN	6t/Neighbor/ASN
RPL rank	6t/Neighbor/RPLrank
Etc …	

6TiSCH Resources

- Informational Resources

6top informational resources and the related URI paths

Name	Accessibility 6top Commands	URI path
Queue	READ/CONFIGURE	6t/Queue
Queue stats	READ/CONFIGURE	6t/QueueStats
Monitoring status	READ/CONFIGURE	6t/MonitoringStatus
Statistics metrics	READ/CONFIGURE	6t/StatisticsMetrics

Message Formats

- GET

Header	GET
Uri-Path	/6t/Neighbor
Options	Accept: application/cbor Uri-Query: ABNF(ShortAddr==0x1234)

- POST

Header	POST
Uri-Path	/6t/Neighbor
Payload	CBOR({ShortAddr: 0x1234})

- DELETE

Header	DELETE
Uri-Path	/6t/Neighbor
Options	Uri-Query: ABNF(ShortAddr==0x1234)

Open Issue -

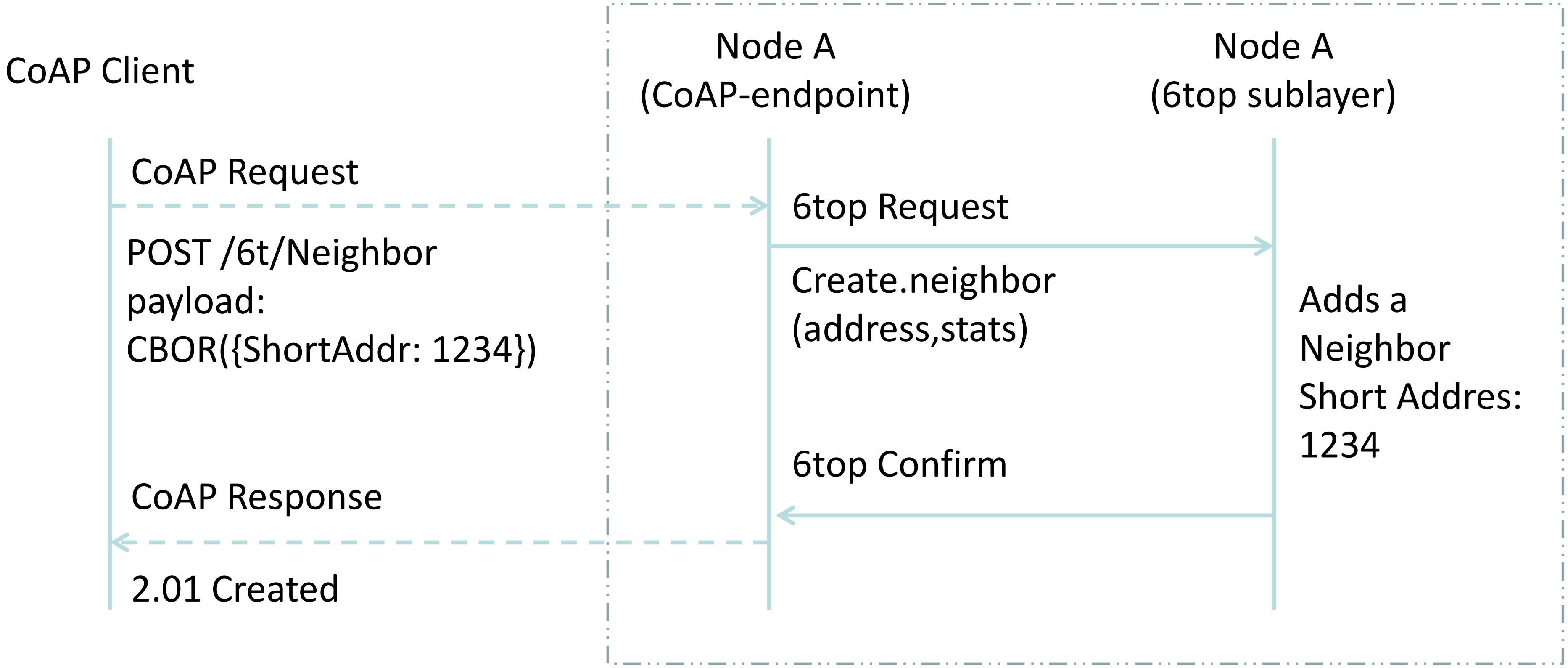
- ABNF to be used as a description method for queries.
- Can be used to define collections of resource
- In the observe model it will be used to define patterns to be monitored

A sample of mapping between CoAP methods and 6top commands

CoAP method	6top command	6top behavior	CoAP Response
POST /6t/Neighbor CBOR({ShortAddr: 1234})	Create.neighbor (address,stats)	Adds a neighbor	2.01 Created
GET /6t/Neighbor	Read.all. neighbor()	Reads all neighbors	2.05 Content CBOR(Neighbor table)
GET /6t/Neighbor Uri-Query – ShortAddr == 0x1234	Read.neighbor (address)	Reads neighbor information	2.05 Content CBOR(Neighbor table)
POST /6t/Neighbor CBOR({ShortAddr: 1234})	Update.neighbor (address,stats)	Updates an entry	2.04 Changed
DELETE /6t/Neighbor Uri-Query – ShortAddr == 0x1234	Delete.neighbor (address)	Removes a neighbor	2.02 Deleted

Example (1/3)

- Request-Response



CoAP Client

Node A
(CoAP-endpoint)

Node A
(6top sublayer)

CoAP Request

POST /6t/Neighbor
payload:
CBOR({ShortAddr: 1234})

6top Request

Create.neighbor
(address,stats)

Adds a
Neighbor
Short Addres:
1234

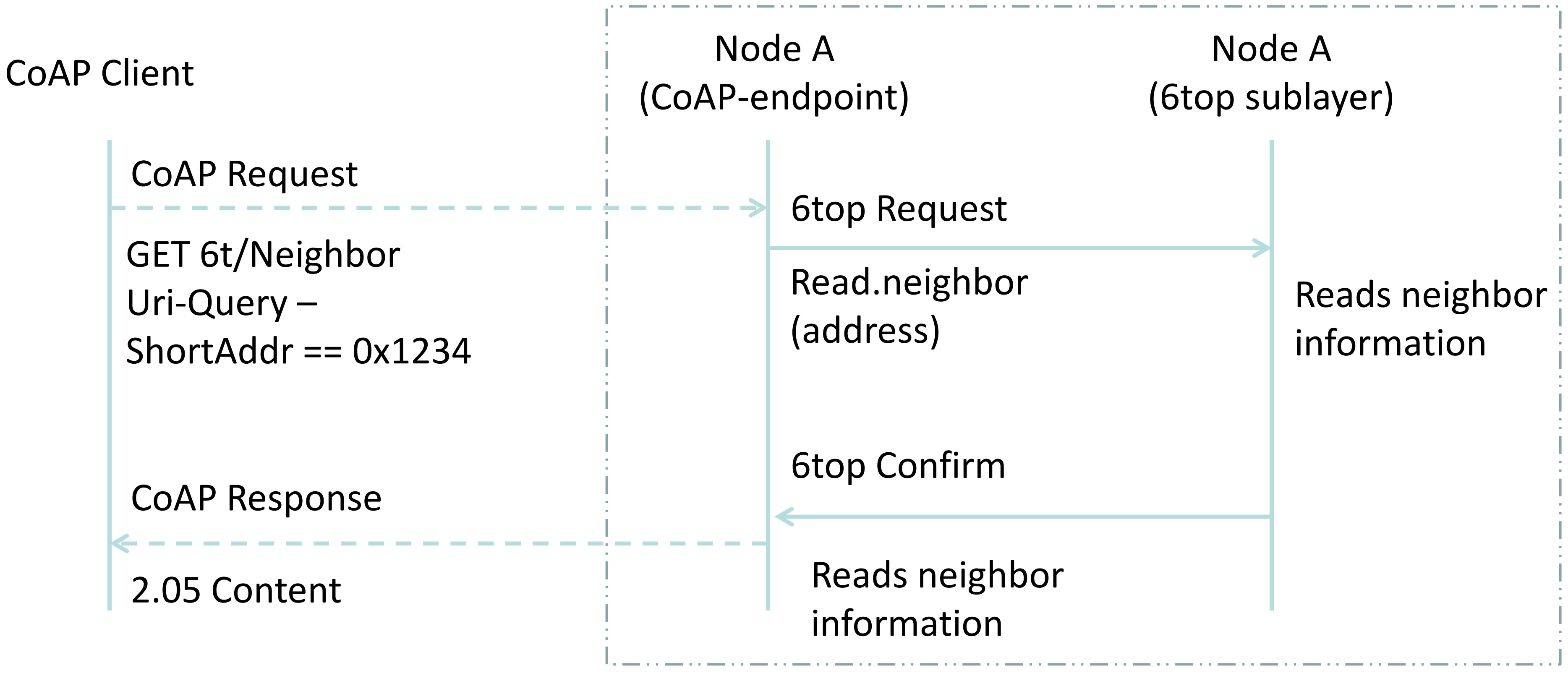
6top Confirm

CoAP Response

2.01 Created

Example (2/3)

- Request-Response



CoAP Client

Node A (CoAP-endpoint)

Node A (6top sublayer)

CoAP Request

GET 6t/Neighbor
Uri-Query –
ShortAddr == 0x1234

6top Request

Read.neighbor
(address)

Reads neighbor information

6top Confirm

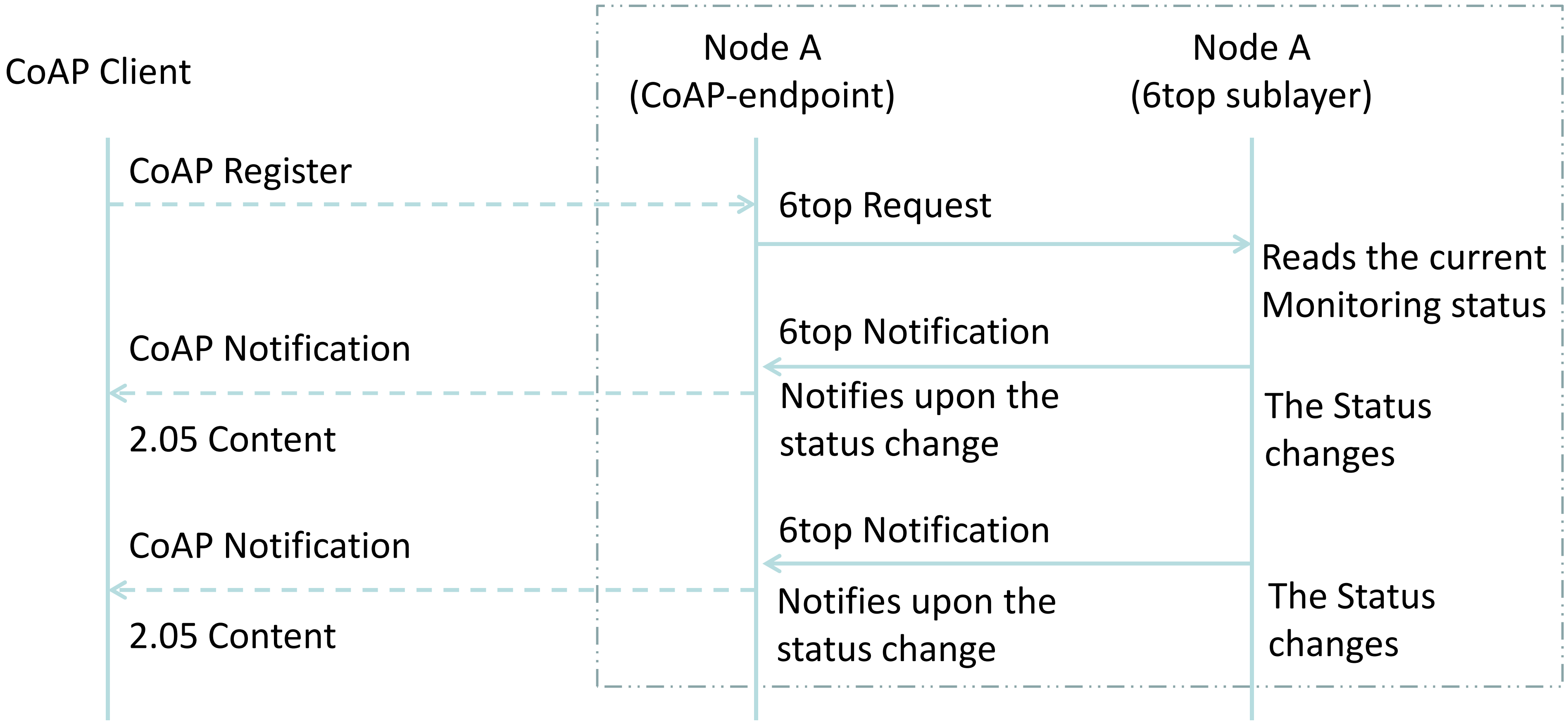
CoAP Response

2.05 Content

Reads neighbor information

Example (3/3)

- Publish-Subscribe



CoAP Client

Node A
(CoAP-endpoint)

Node A
(6top sublayer)

CoAP Register

6top Request

Reads the current
Monitoring status

6top Notification

CoAP Notification

Notifies upon the
status change

2.05 Content

The Status
changes

6top Notification

CoAP Notification

Notifies upon the
status change

2.05 Content

The Status
changes

Generic Data Model

- Need to define a data model that can be used across other protocols e.g. a track reservation protocol

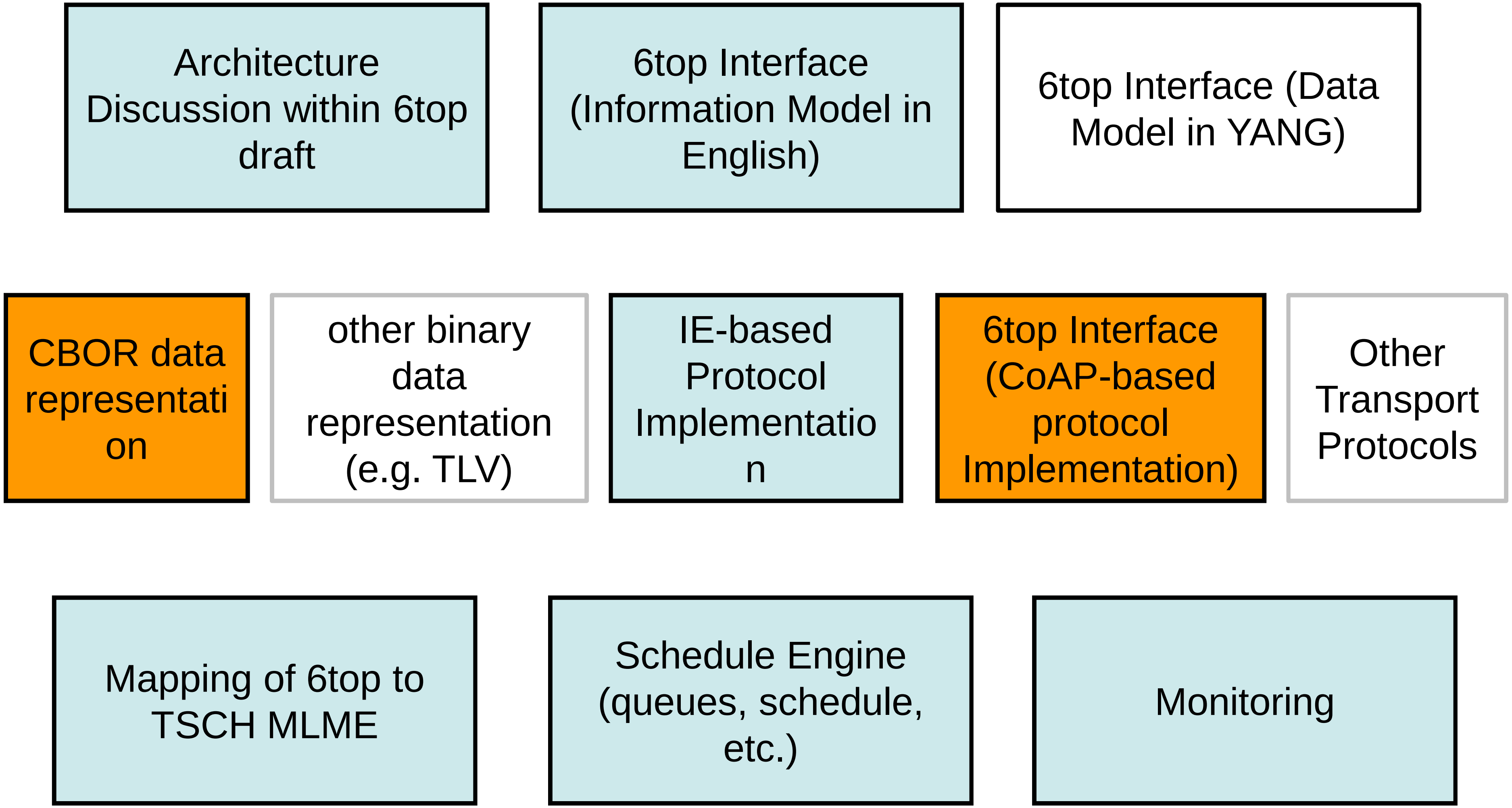
- YANG to be used as a data model?

Coverage gap analysis vs. charter

Dominique Barthel

- 🟦 draft-wang-6tisch-6top-00
- 🟨 draft-sudhaakar-6tisch-coap-00
- 🟩 draft-thubert-6tisch-architecture-01

Drafts Organization



Architecture Discussion within 6top draft

6top Interface (Information Model in English)

6top Interface (Data Model in YANG)

CBOR data representation

other binary data representation (e.g. TLV)

IE-based Protocol Implementation

6top Interface (CoAP-based protocol Implementation)

Other Transport Protocols

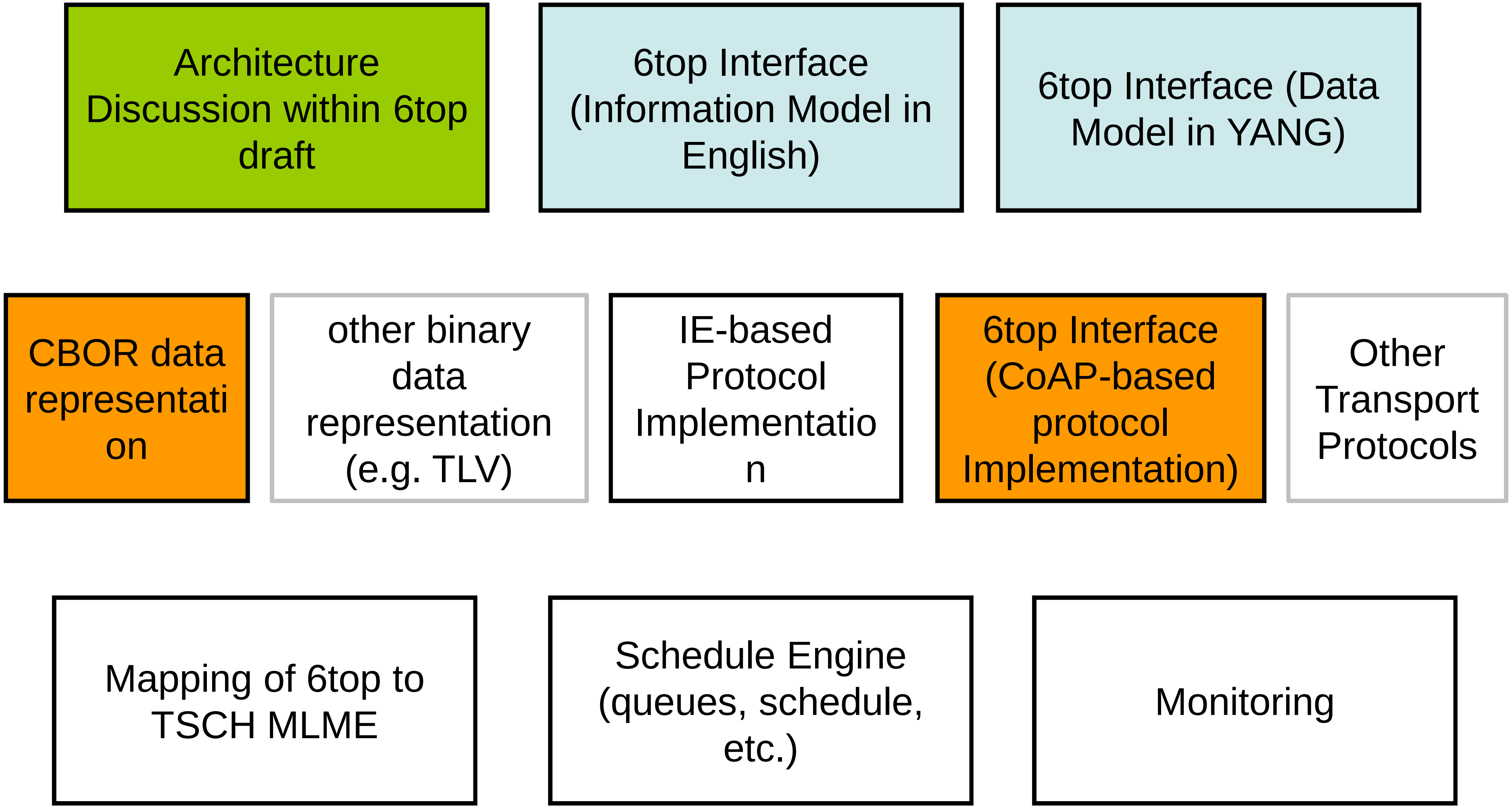
Mapping of 6top to TSCH MLME

Schedule Engine (queues, schedule, etc.)

Monitoring

Drafts Organization

- 🟦 draft-wang-6tisch-6top-00
- 🟨 draft-sudhaakar-6tisch-coap-00
- 🟩 draft-thubert-6tisch-architecture-01


Architecture Discussion within 6top draft
6top Interface (Information Model in English)
6top Interface (Data Model in YANG)
CBOR data representation
other binary data representation (e.g. TLV)
IE-based Protocol Implementation
6top Interface (CoAP-based protocol Implementation)
Other Transport Protocols
Mapping of 6top to TSCH MLME
Schedule Engine (queues, schedule, etc.)
Monitoring

draft-vilajosana-6tisch-minimal-00

Xavi Vilajosana (Ed.)

Kris Pister

draft-vilajosana-6tisch-minimal-00

- Status:
  - New revision of draft-vilajosana-**6tsch**-basic-**01**
  - Latest version published 10/09/2013 http://tools.ietf.org/html/draft-vilajosana-6tisch-minimal-00
  - Running version at https://bitbucket.org/6tisch/draft-vilajosana-6tisch-minimal

- Changes since IETF87
  - New draft

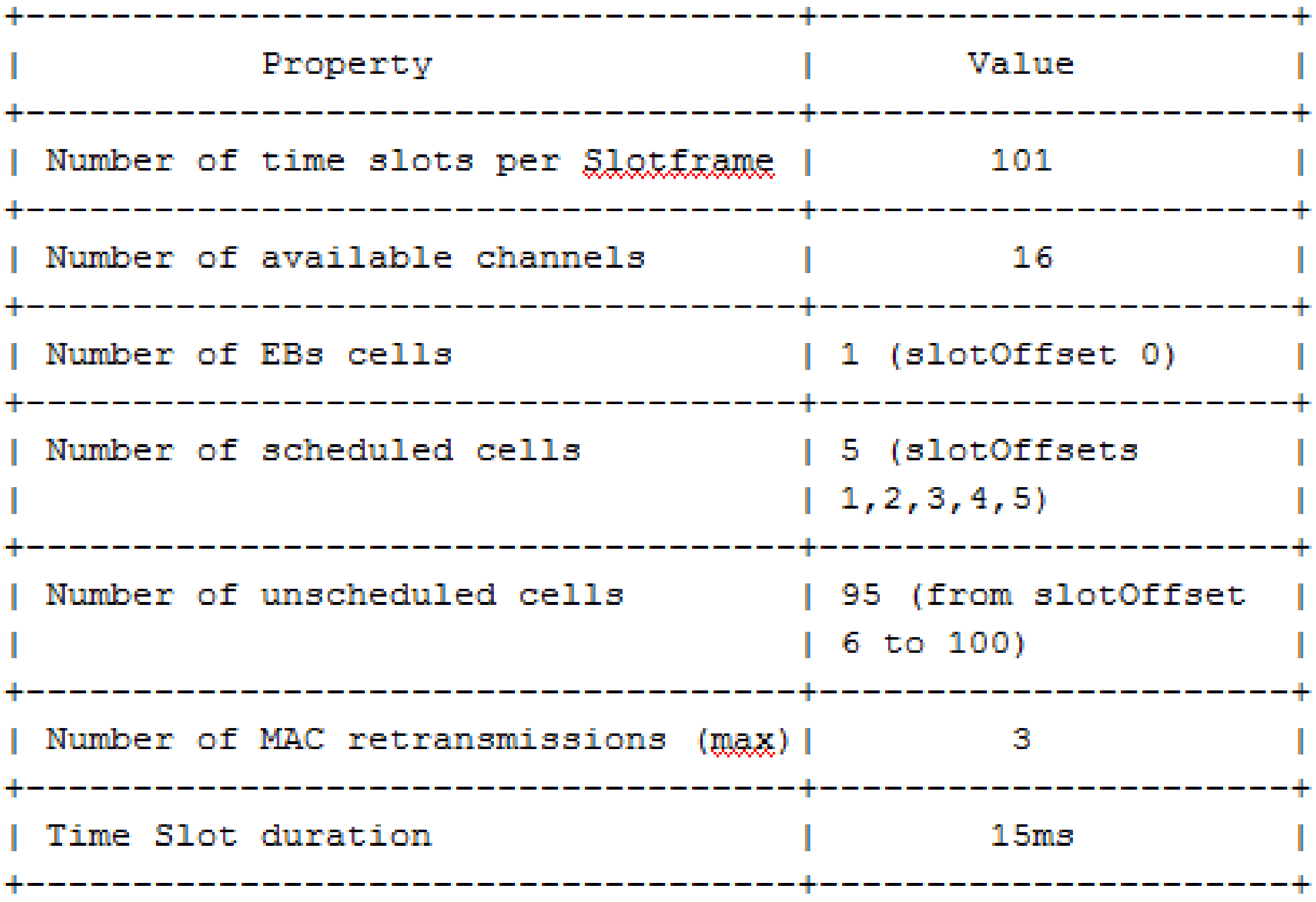
Charter

“Produce -Minimal 6TiSCH Configuration- defining how to build a 6TiSCH network using the Routing Protocol for LLNs (RPL) and a static TSCH schedule. It is expected that RPL and the Objective Function 0 (OF0) will be reused as-is. The work will include a best practice configuration for RPL and OF0 operation over the static schedule … .”

Objectives

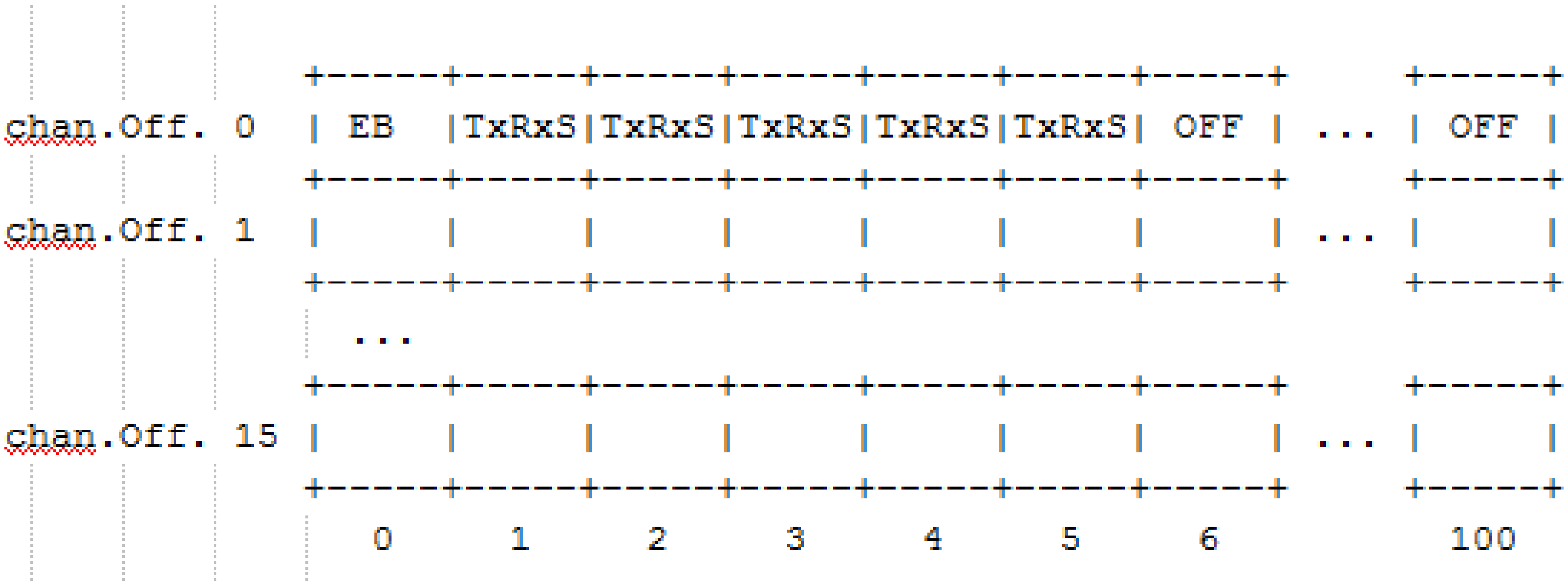
- Define the “default” slotframe configuration
- Define minimal information carried by the EB
- Define time source neighbor selection mechanism
- Define minimal set of information to track neighbours
- Specify how to use OF0 (RFC6552) on TSCH

Slotframe parameters



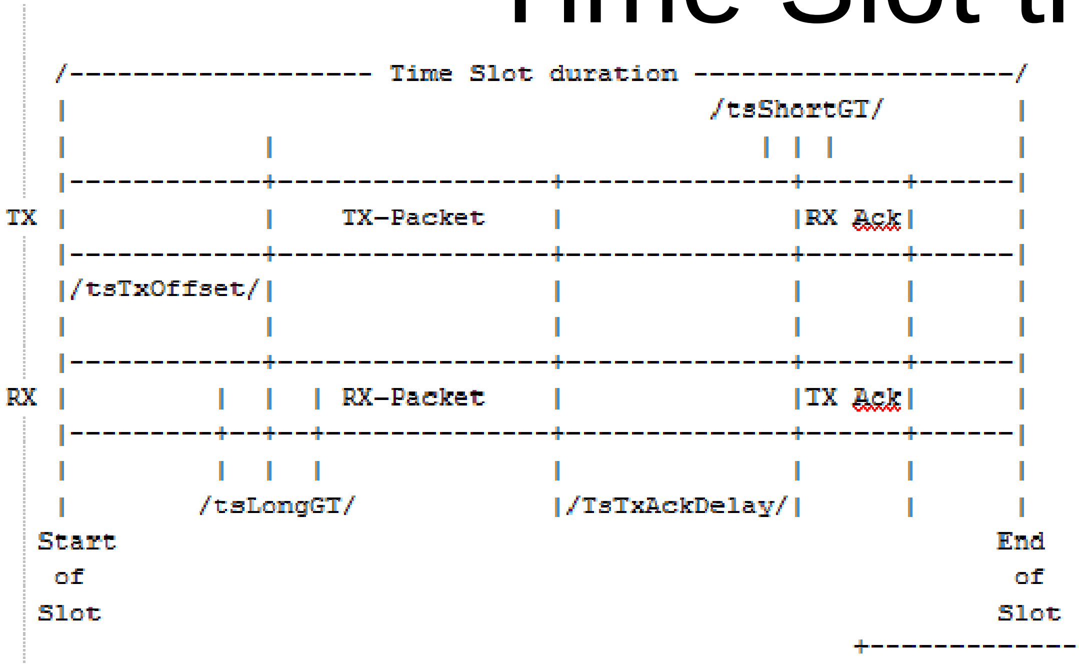
Property	Value
Number of time slots per Slotframe	101
Number of available channels	16
Number of EBs cells	1 (slotOffset 0)
Number of scheduled cells	5 (slotOffsets 1,2,3,4,5)
Number of unscheduled cells	95 (from slotOffset 6 to 100)
Number of MAC retransmissions (max)	3
Time Slot duration	15ms

Slotframe representation

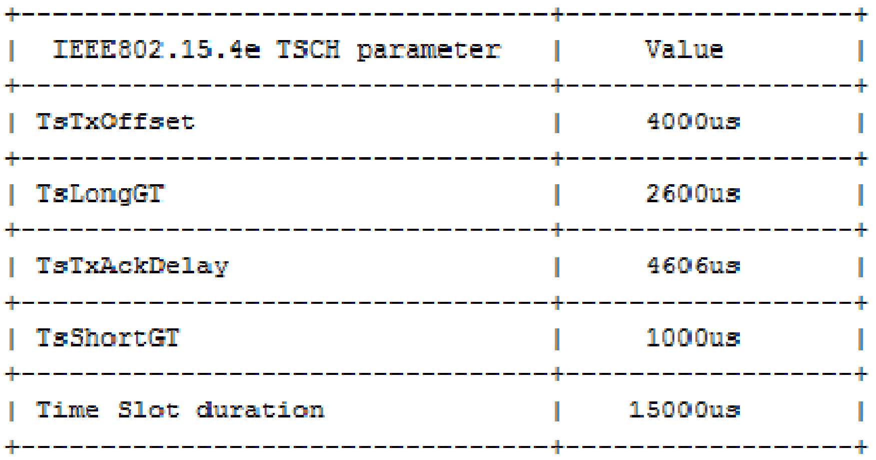


```
                 +-----+-----+-----+-----+-----+-----+-----+     +-----+
chan.Off. 0      | EB  |TxRxS|TxRxS|TxRxS|TxRxS|TxRxS| OFF | ... | OFF |
                 +-----+-----+-----+-----+-----+-----+-----+     +-----+
chan.Off. 1      |     |     |     |     |     |     |     | ... |     |
                 +-----+-----+-----+-----+-----+-----+-----+     +-----+
                    ...
                 +-----+-----+-----+-----+-----+-----+-----+     +-----+
chan.Off. 15     |     |     |     |     |     |     |     | ... |     |
                 +-----+-----+-----+-----+-----+-----+-----+     +-----+
                    0     1     2     3     4     5     6          100
```

Time Slot timing



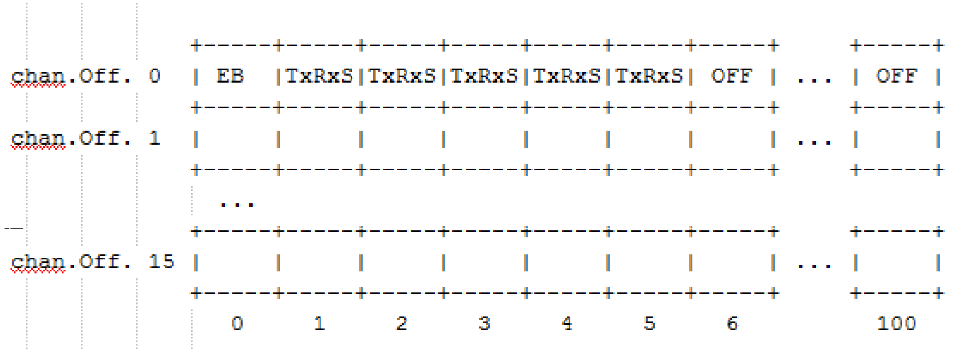
```
   /------------------- Time Slot duration --------------------/
   |                                        /tsShortGT/        |
   |            |                              | | |           |
   |------------+-----------------+--------------+------+------|
TX |            |    TX-Packet    |              |RX Ack|      |
   |------------+-----------------+--------------+------+------|
   |/tsTxOffset/|                 |              |      |      |
   |            |                 |              |      |      |
   |------------+-----------------+--------------+------+------|
RX |         |  |  | RX-Packet    |              |TX Ack|      |
   |---------+--+--+--------------+--------------+------+------|
   |         |  |  |              |              |      |      |
   |        /tsLongGT/            |/TsTxAckDelay/|      |      |
  Start                                                       End
   of                                                          of
  Slot                                                        Slot
```



IEEE802.15.4e TSCH parameter	Value
TsTxOffset	4000us
TsLongGT	2600us
TsTxAckDelay	4606us
TsShortGT	1000us
Time Slot duration	15000us

EB Content

- Sync IE : ASN + Join Priority
- SlotFrame and Link IE
  - # Slotframes (b16-b23) = 0x01
  - Slotframe ID (b24-b31) = 0x01
  - Size Slotframe (b32-b47) = 0x6
  - # Links (b48-b55) = 0x06
  - For each link in the minimal schedule:
    - Channel Offset (2B) = 0x00
    - Slot Number (2B) = from (0x00 to 0x05)
    - LinkOption (1B) = as described in Section 2.2



```
                +-----+-----+-----+-----+-----+-----+-----+     +-----+
chan.Off. 0     | EB  |TxRxS|TxRxS|TxRxS|TxRxS|TxRxS| OFF | ... | OFF |
                +-----+-----+-----+-----+-----+-----+-----+     +-----+
chan.Off. 1     |     |     |     |     |     |     |     | ... |     |
                +-----+-----+-----+-----+-----+-----+-----+     +-----+
                  ...
                +-----+-----+-----+-----+-----+-----+-----+     +-----+
chan.Off. 15    |     |     |     |     |     |     |     | ... |     |
                +-----+-----+-----+-----+-----+-----+-----+     +-----+
                   0     1     2     3     4     5     6          100
```

Time Source Neighbour Selection

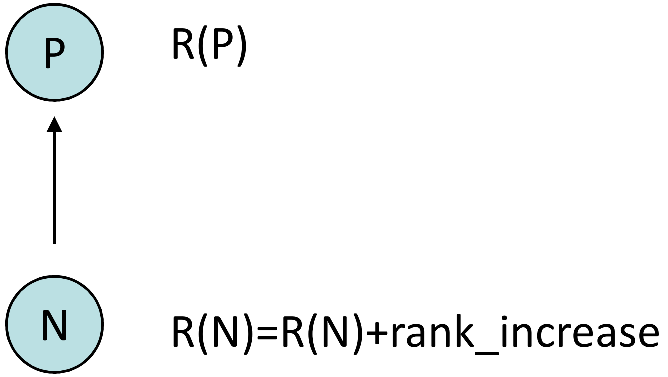
- Each node selects at least one time parent amongst its known neighbours
- EB[Join Priority]==DAGRank(rank)
  - Cannot EB until rank is set (DIO - DIS/DIO)
- TimeParent is min(JoinPriority)
- Backup TimeParent is also selected as 2nd best Join Priority
- Optional:
  - Stability counter
  - In case of equal Join Priority use RSSI or other metric to disambiguate

Neighbours

- **Neighbour Table: for each neighbor**
- Neighbour statistics:
  - number of transmitted packets to that neighbour
  - number of transmitted packets that have been acknowledged by that neighbour
  - number of received packets from that neighbour
  - neighbour address
- Timestamp (ASN) when that neighbour was heard for the last time.
- RPL rank of that neighbour.
- A flag which indicates whether this neighbour is a time source neighbour.
- Connectivity statistics (RSSI, etc)

RPL on Minimal

- RFC6552 “Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL)”
- Definitions
  - – Rf: rank_factor
  - – Sp: step_of_rank
  - – Sr: stretch_of_rank



rank_increase = (Rf*Sp + Sr) * MinHopRankIncrease

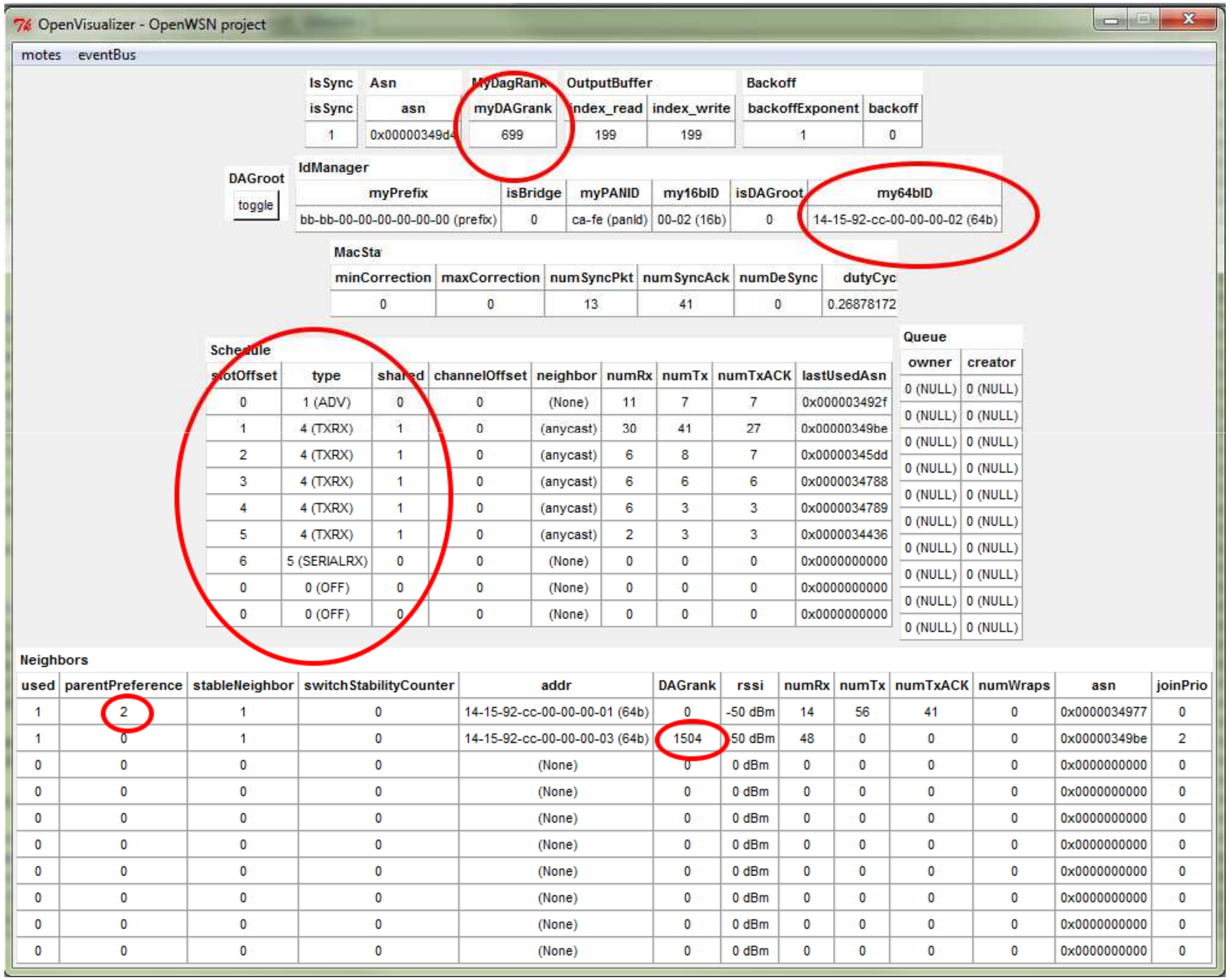
RPL Configuration

- Non Storing Mode (MUST) – Storing Mode (MAY)
- Trickle Timer:
  - As defined in [RFC6550] (Section 8.3.1) and [RFC6206]
- Hysteresis
  - Use a boundary value (PARENT_SWITCH_THRESHOLD) to avoid constant changes of parent when ranks are compared. [RFC6719]

Open Issues

- ASN vs Timestamp
- 2*ETX is not defined in OF0
- Can Minimal Conf. Cells be modified?
- How many neighbors are needed before selecting time source neighbor and joining?

OpenWSN.org



IsSync	Asn	MyDagRank	OutputBuffer		Backoff	
isSync	asn	myDAGrank	index_read	index_write	backoffExponent	backoff
1	0x00000349d4	699	199	199	1	0

IdManager

myPrefix	isBridge	myPANID	my16bID	isDAGroot	my64bID
bb-bb-00-00-00-00-00-00 (prefix)	0	ca-fe (panId)	00-02 (16b)	0	14-15-92-cc-00-00-00-02 (64b)

MacSta

minCorrection	maxCorrection	numSyncPkt	numSyncAck	numDeSync	dutyCyc
0	0	13	41	0	0.26878172

Schedule

slotOffset	type	shared	channelOffset	neighbor	numRx	numTx	numTxACK	lastUsedAsn
0	1 (ADV)	0	0	(None)	11	7	7	0x000003492f
1	4 (TXRX)	1	0	(anycast)	30	41	27	0x00000349be
2	4 (TXRX)	1	0	(anycast)	6	8	7	0x00000345dd
3	4 (TXRX)	1	0	(anycast)	6	6	6	0x0000034788
4	4 (TXRX)	1	0	(anycast)	6	3	3	0x0000034789
5	4 (TXRX)	1	0	(anycast)	2	3	3	0x0000034436
6	5 (SERIALRX)	0	0	(None)	0	0	0	0x0000000000
0	0 (OFF)	0	0	(None)	0	0	0	0x0000000000
0	0 (OFF)	0	0	(None)	0	0	0	0x0000000000

Queue

owner	creator
0 (NULL)	0 (NULL)
0 (NULL)	0 (NULL)
0 (NULL)	0 (NULL)
0 (NULL)	0 (NULL)
0 (NULL)	0 (NULL)
0 (NULL)	0 (NULL)
0 (NULL)	0 (NULL)
0 (NULL)	0 (NULL)
0 (NULL)	0 (NULL)
0 (NULL)	0 (NULL)

Neighbors

used	parentPreference	stableNeighbor	switchStabilityCounter	addr	DAGrank	rssi	numRx	numTx	numTxACK	numWraps	asn	joinPrio
1	2	1	0	14-15-92-cc-00-00-00-01 (64b)	0	-50 dBm	14	56	41	0	0x0000034977	0
1	0	1	0	14-15-92-cc-00-00-00-03 (64b)	1504	-50 dBm	48	0	0	0	0x00000349be	2
0	0	0	0	(None)	0	0 dBm	0	0	0	0	0x0000000000	0
0	0	0	0	(None)	0	0 dBm	0	0	0	0	0x0000000000	0
0	0	0	0	(None)	0	0 dBm	0	0	0	0	0x0000000000	0
0	0	0	0	(None)	0	0 dBm	0	0	0	0	0x0000000000	0
0	0	0	0	(None)	0	0 dBm	0	0	0	0	0x0000000000	0
0	0	0	0	(None)	0	0 dBm	0	0	0	0	0x0000000000	0
0	0	0	0	(None)	0	0 dBm	0	0	0	0	0x0000000000	0
0	0	0	0	(None)	0	0 dBm	0	0	0	0	0x0000000000	0

Call for draft adoption

Agenda

Intro and Status	[10min]
– Agenda Bashing	
– draft-palattella-6tisch-terminology	
– 6TiSCH charter recap	
Architecture	[15min]
– <draft-watteyne-6tisch-tsch-00>	(Thomas Watteyne)
– <draft-thubert-6tisch-architecture-01>	(Pascal Thubert)
Information and Data Models	[25min]
– <draft-wang-6tisch-6top-00>	(Xavi Vilajosana)
– <draft-sudhaakar-6tisch-coap-00>	(Pouria Zand)
– Coverage gap analysis vs. charter	(Dominique Barthel)
Minimal RPL support	[10min]
– <draft-vilajosana-6tisch-minimal-00>	(Xavi Vilajosana)
Call for draft adoption	[10min] (Chairs)
Unchartered drafts if time permits	[15min]
– <draft-ohba-6tisch-security-00>	(Yoshihiro Ohba)
– Overview discussion on slot allocation principles	
– <draft-piro-6tisch-security-issues-00>	
AOB	[5min]

draft-ohba-6tisch-security-00

Stephen Chasko
Subir Das
Rafa Marin-Lopez
Yoshihiro Ohba (Ed.)
Pascal Thubert
Alper Yegin

draft-ohba-6tisch-security-00

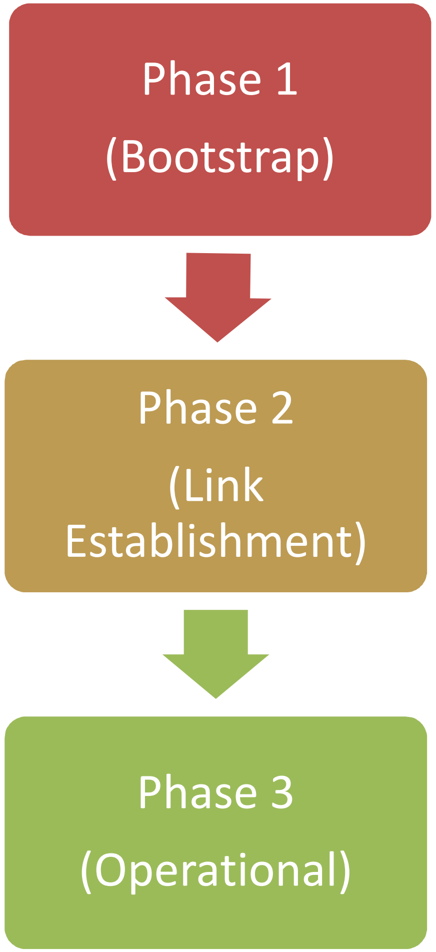
- Status:
  - New revision of draft-ohba-**6tsch**-security-**01**
  - Latest version published 10/21/2013 http://tools.ietf.org/html/draft-ohba-6tisch-security-00
  - Running version at https://bitbucket.org/6tisch/draft-ohba-6tisch-security
- Changes since IETF87
  - See next slides.

Background and Objectives

- 6TiSCH operates over IEEE 802.15.4e TSCH MAC
  - 5-octet global counter called ASN (Absolute Slot Number) is used as part of CCM* nonce
  - A pre-installed common network key has been used in existing TSCH deployment, which is undesirable from security point of view
- Objectives of this work:
  - To provide a secure and scalable key management framework that allows dynamic CCM* key establishment and update for 6TiSCH networks
  - To identify requirements on key management protocols that realizes the framework
- Non-objective of this work
  - Defining a key management protocol

Key Management Framework

Phase 0 (Commissioning) : Phase 1 KMP credentials are pre-installed to a mesh device in a physically secure and managed environment before the device is placed where it is expected to operate



Phase 1 (Bootstrap)

- A mesh device (re)install credentials used for subsequent phases
- Communication with an authentication server takes place via a neighboring node acting as a relay
- There may be no link layer security with the neighboring node

Phase 2 (Link Establishment)

- **Two-party** authentication and key establishment takes place between the mesh device and its neighbor for establishing link layer keys

Phase 3 (Operational)

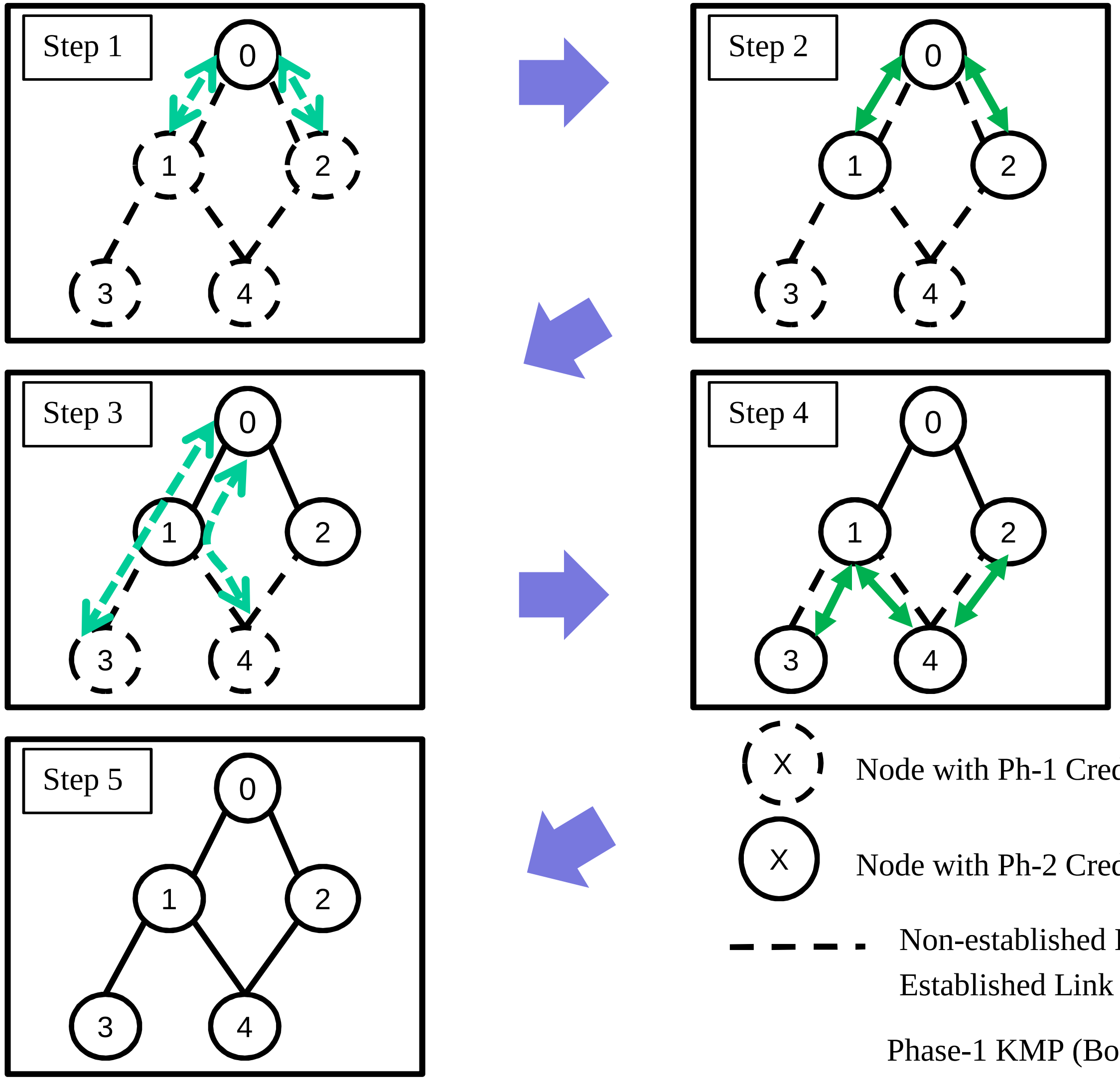
- The mesh device is able to run various protocols over IP
- Additional authentication and key establishment may take place for those protocols

L2 keys established in Phase-2

- Notation
  - K_ij : L2 key for unicast L2 frames transmitted from node i to node j
  - K_i: L2 key for broadcast L2 frames transmitted by node i

- L2 key modes
  - Mode 1: broadcast & unicast keys are the same
    - K_ij=K_i for all j
    - Variations:
      - **Per-network key**: K_i,j = K (for all I, j)
      - **Per-neighbor key**: K_i,j = K_i (for all j)
  - Mode 2: broadcast & unicast keys are different with bidirectional unicast keys
    - K_ij=K_ji, K_i!=K_j for all i,j (i!=j)
  - Mode 3: broadcast & unicast keys are different with unidirectional unicast keys
    - K_ij!=K_ji, K_i!=K_j for all i,j (i!=j)

Example Sequence



Node with Ph-1 Credentials only

Node with Ph-2 Credentials

Non-established Link

Established Link (over which Ph-3 KMP can be run)

Phase-1 KMP (Bootstrapping KMP)

Phase-2 KMP (Link Establishment KMP)

KMP: Key Management Protocol

Phase-1 KMP Requirements

R1-1: Phase-1 KMP MUST support mutual authentication

R1-2: Phase-1 KMP MUST support stateless authentication relay operation

R1-3: Phase-1 KMP MUST support secure credential distribution.

KMP: Key Management Protocol

Phase-2 KMP Requirements

R2-1: Phase-2 KMP Nodes MUST mutually authenticate each other before establishing a link and forming a mesh network. No authentication server is involved in the Phase-2 authentication.

R2-2: Phase-2 KMP authentication credentials MAY be pre-provisioned or MAY be obtained via Phase-1 KMP.

R2-3: Phase-2 KMP authentication credentials MUST have a lifetime.

R2-4: Phase-2 KMP MUST support certificates for scalable operation.

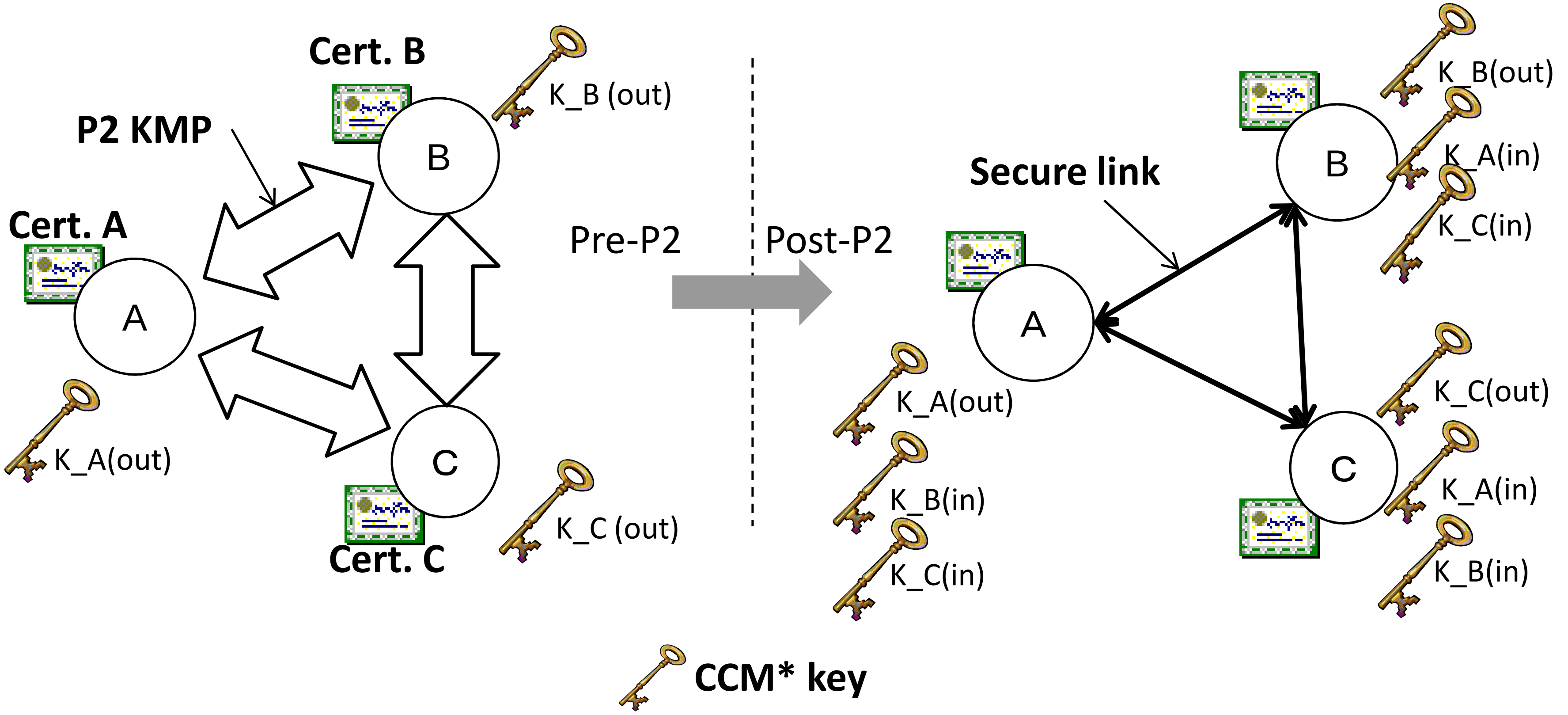
R2-5: Phase-2 KMP message exchanges MUST be integrity and replay protected after successful authentication.

R2-6: Phase-2 KMP MUST have the capability to establish security association and unicast session keys after successful authentication to protect unicast MAC frames between nodes.

R2-7: Phase-2 KMP MUST have the capability to establish security association and broadcast session keys after successful authentication to protect broadcast MAC frames between nodes.

R2-8: Phase-2 KMP MUST support confidentiality to distribute the broadcast session keys securely.

Example Phase-2 KMP Operation (per-neighbor key model)



On-The-Fly Scheduling *

Diego Dujovne
Luigi Alfredo Grieco
Maria Rita Palattella
Nicola Accettura

* on-going work. No draft published yet.

Problem:
timeSlot Distributed Allocation is complex

Is there an the optimal distribution of role between centralized planning and distributed performances

???

Allocator

- Centralized Allocation
  - – Limits the scalability
  - – Indirect sensitivity to overlap
- Fully distributed (Node picks its xmit slots)
  - – Sensitive to collision on rcv side
- Fully distributed (Node picks its rcv slots)
  - – Sender may impact terminal 2 hops away
- Parent-Base allocation
  - – Reduces the amount of allocators by an order
  - – Allows for bulk allocation and redistribution

Problem:
non-deterministic (VBR) traffic yields poor use of reserved resources

How do we optimize the usage of timeSlots for general traffic

???

Possible Answers

- Traditional answers
  - – Statistical multiplexing (FR, ATM …)
  - – Best Effort and then QoS
  - – Random Access with Long preamble
- TSCH additional answers
  - – On-The-Fly time slot allocation (! Overhead)
  - – Next-Slot-Used indication (! Slot consumption)
  - – Multiply allocated time slots (!risk of collision)

draft-piro-6tisch-security-issues-00

Giuseppe Piro
Gennaro Boggia
Luigi Alfredo Grieco

draft-piro-6tisch-security-issues-00

- Status:
  - New draft
  - Latest version published 10/18/2013 http://tools.ietf.org/html/draft-piro-6tisch-security-issues-00
  - Running version at https://bitbucket.org/6tisch/draft-piro-6tisch-security-issues

- Changes since IETF87
  - New draft

Overview

- **Main goals**:
  - define a standard compliant framework offering security services at the MAC layer
  - 5 different secured network configurations
  - 3 consecutive phases to build a secured domain
  - lightweight scheme to negotiate link keys
  - how to configuring MAC security attributes

- **Position wrt 6tisch WG and draft-ohba-6tisch-security-00**
  - security services for upper layers (i.e., 6top)
  - fully compatible wrt 6tisch networks (i.e., by extending the “domain” concept to multihop environments)
  - fully compatible with IEEE 802.15.4(e)
  - in line wrt Yoshihiro’s work (piro- and ohba- drafts focus on different aspects)

- **Future goals**:
  - demonstrate properties of the proposed solution through additional references
  - understand how the idea can be exploited/extended by/to upper layers

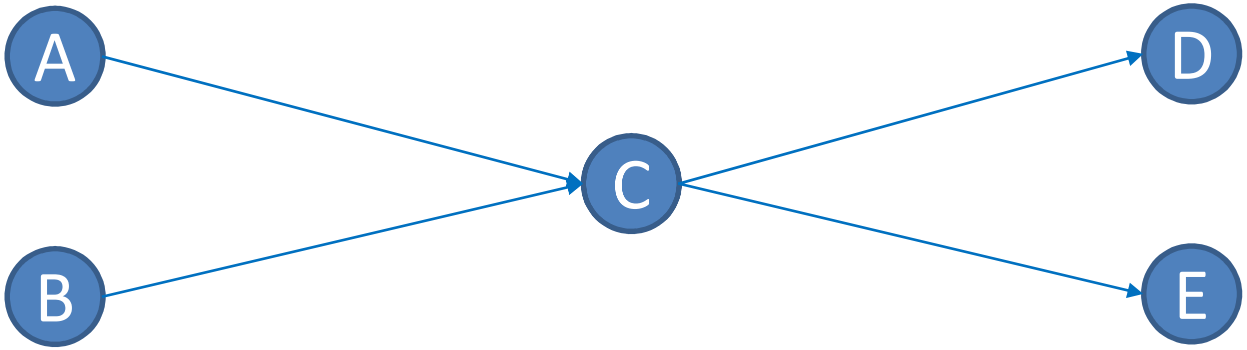
Any Other Business?

Thank you!

Hard Cells vs. Soft Cells

- Mandatory flag for each scheduled cell
- Hard Cell
  - Can not be moved by 6top
  - When cell is scheduled at exact slotOffset/channelOffset (e.g. PCE)
- Soft Cell
  - Performance is monitored by the 6top monitoring process and moved if needed
  - When bandwidth is scheduled rather than exact cell (e.g. distributed scheduling

Cells and Tracks



Track1: based on hard cells **Track2: based on soft cells**

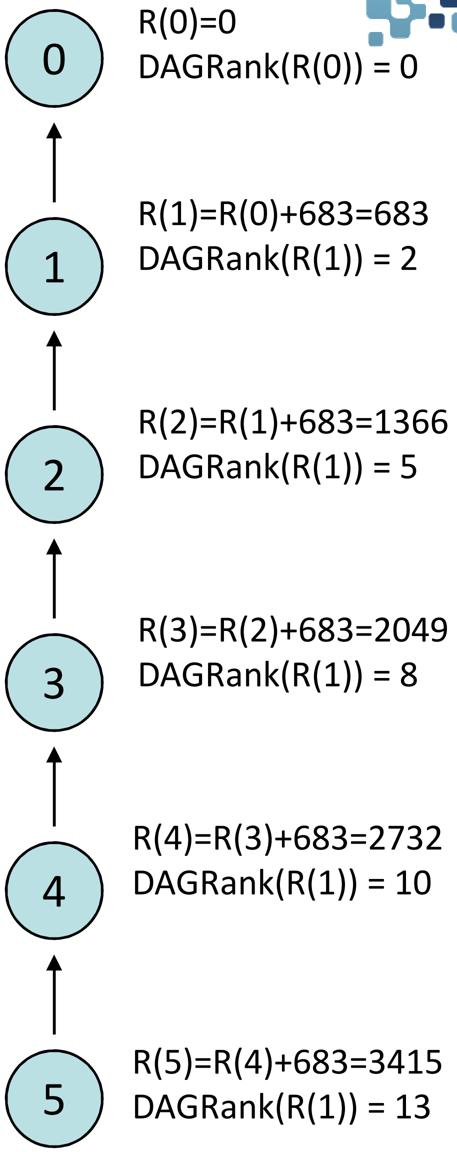
channelOffset

			B=>C						
		A=>C			C=>D		B=>C		
	A=>C								C=>E
				C=>E					

slotFrame

Example

- Given:
  - Rf = 1
  - Sp = 2* ETX
  - Sr = 0
  - minHopRankIncrease = 256 (default in RPL)
  - ETX=(xmit/ack)
  - r(n) = r(p) + rank_increase
  - rank_increase= (Rf*Sp + Sr) * minHopRankIncrease
  - rank_increase=(512*xmit/ack)
- Example:
  - 5-hop network
  - r(0)=0
  - xmit=100 ack=75 for all links



R(0)=0
DAGRank(R(0)) = 0

R(1)=R(0)+683=683
DAGRank(R(1)) = 2

R(2)=R(1)+683=1366
DAGRank(R(1)) = 5

R(3)=R(2)+683=2049
DAGRank(R(1)) = 8

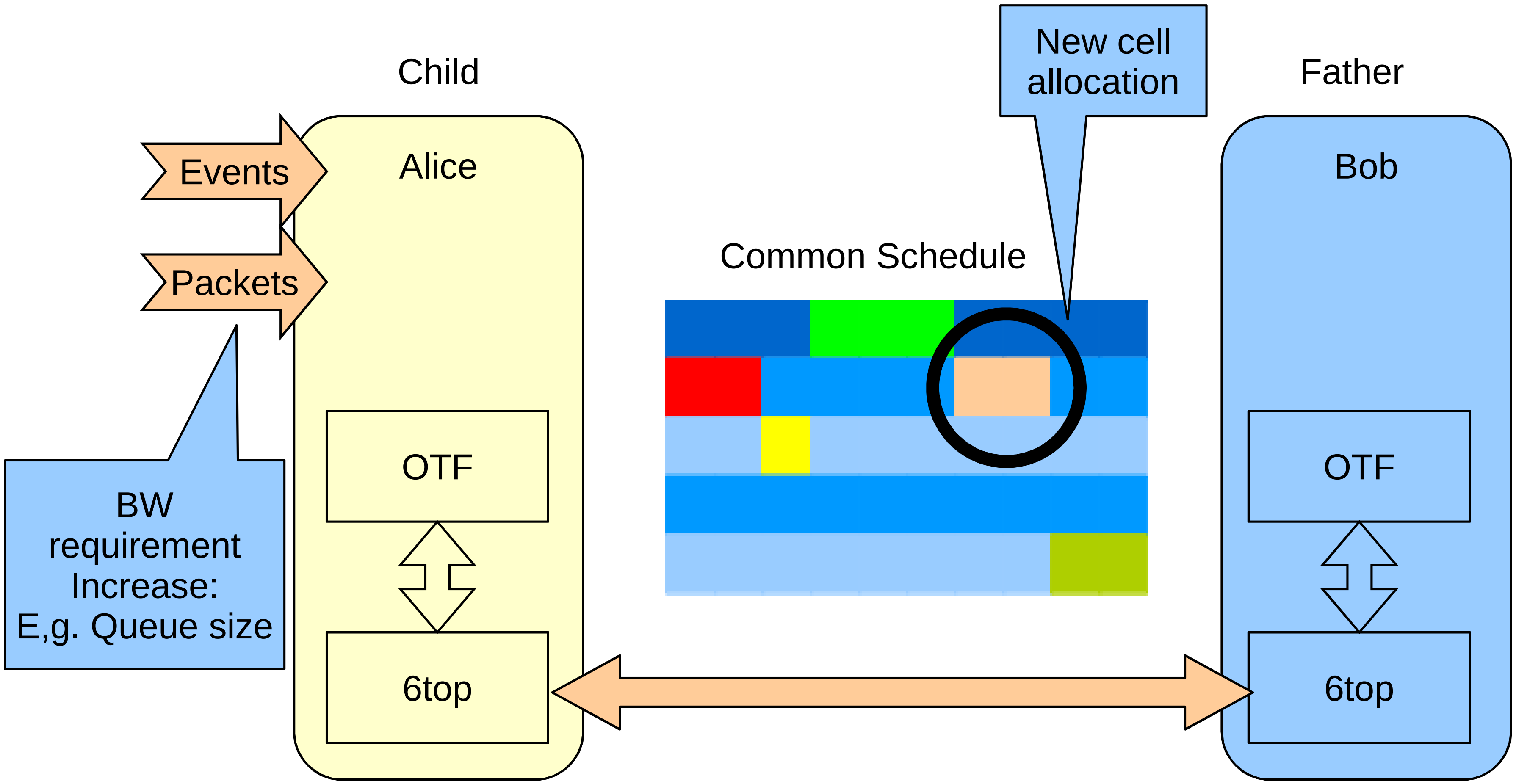
R(4)=R(3)+683=2732
DAGRank(R(1)) = 10

R(5)=R(4)+683=3415
DAGRank(R(1)) = 13

Objective

- Dynamic adaptation of the number of allocated bandwidth or cells between neighbor nodes without the intervention of a centralized entity.
- Defines a framework and a set of methods
- OTF Module includes:
  - A profile to extract statistics from the 6top layer (open)
  - One or more scheduling algorithms (open)
  - An allocation policy for bandwidth, cells or bundles

On-The-Fly scheduling / mechanism



On-The-Fly scheduling / index

- 1. Introduction
- 2. Allocation policy
- 2.1. Allocation methods
- 3. Input parameters: statistics and instant values
- 4. Bundle usage management in OTF
- 4.1. Cell Reservation/Deletion
- 4.2. Bundle Size Increase/Decrease
- 5. Schedule storage on OTF
- 6. Acknowledgements
- 7. References
- 7.1. Informative References
- 7.2. External Informative References

Proposed secured network configurations

- **unsecured** → security services are not supported at all
- **partially secured** → integrity check for all the messages
- **fully secured** → confidentiality and integrity check for all the messages
- **hybrid** → broadcast messages are transmitted in clear. There could be some devices that protect their point-to-point communication
- **flexible** → the network is fully secured from the beginning. It skips to the hybrid configuration in the case there is a device that does not support security capabilities

3 phases for configuring a secured domain

- **Setting-up phase**
  - install **MasterKey M_k** and other security parameters (i.e., minimum secured levels, secured network configuration, and a table of prime numbers for the DH algorithm)
  - handled by the manufacturer or network administrator

- **Bootstrap phase**
  - compute **DefaultKey D_k** (used to protect broadcast messages) and update security attributes at the MAC layer
  - implemented when the device joins to the network
  - different procedures for FFD and RFD in both beacon-enabled and not-beacon-enabled configurations

- **Key negotiation phase**
  - compute **LinkKeys L_k** (used to protect unicast messages exchanged between two nodes) through a DH-based scheme and update of security attributes at the MAC layer
  - definition of a new MAC command frame
  - key negotiation mechanism composed by 4 messages exchanged between nodes (see Fig. 7 of the draft)