ETSI TR 103 777 V1.1.1 (2024-10)

Digital Enhanced Cordless Telecommunications (DECT);
DECT-2020 New Radio (NR) interface;
Study of additional functionality for the support of new applications in further releases

Reference
DTR/DECT-00367

Keywords
DECT-2020, future releases, new functionality

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM**® and the GSM logo are trademarks registered and owned by the GSM Association.

Foreword

This Technical Report (TR) has been produced by ETSI Technical Committee Digital Enhanced Cordless Telecommunications (DECT).

Modal verbs terminology

In the present document "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

1 Scope

The present document is a study on DECT-2020 NR technical requirements and additional functionality for the support of new applications in further releases of the specification.

2 References

2.1 Normative references

Normative references are not applicable in the present document.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long-term validity.

The following referenced documents are not necessary for the application of the present document, but they assist the user with regard to a particular subject area.

[i.1] ETSI TS103 636-4: "DECT-2020 New Radio (NR); Part 4: MAC layer; Release 2".

[i.2] DECT(22)000084: "DECT-2020 NR Release - 2 features", Wirepas Oy.

[i.3] ETSI TS 103 874-1: "DECT-2020 New Radio (NR); Access Profile; Part 1: Overview".

[i.4] ETSI TS 103 636-5: "DECT-2020 New Radio (NR); Part 5: DLC and Convergence layers; Release 2".

[i.5] DECT(22)095028: "Add paging support option to association request, Nordic Semiconductor ASA".

[i.6] DECT(22)095027: "Add lifetime to association request", Nordic Semiconductor ASA.

[i.7] DECT(22)000260: "Access without security credentials (rel.2)", Wirepas Oy.

[i.8] DECT(22)000256: "Downlink routing enhancement", Wirepas Oy.

[i.9] DECT(22)000273: "Downlink routing enhancement", Nordic Semiconductor ASA.

[i.10] DECT(23)000038: "DECT-2020 Configuration data distribution data structure", Wirepas Oy.

[i.11] DECT(23)000047: "DECT-2020 IPv6 address configuration", Wirepas Oy.

[i.12] DECT(23)000120: "Rel 2 Selective DL source routing", Wirepas Oy.

[i.13] DECT(23)000121r2: "DLC Header modification", Wirepas Oy.

[i.14] ETSI TS 103 636-1: "DECT-2020 New Radio (NR); Part 1: Overview; Release 2".

[i.15] IETF RFC 4921 (Version 2.5.1): "IP Version 6 Addressing Architecture".

[i.16] IETF RFC 3587: "IPv6 Global Unicast Address Format".

[i.17] IETF RFC 4193: "Unique Local IPv6 Unicast Addresses".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI TS 103 636-1 [i.14] apply.

NOTE: A term defined in the present document takes precedence over the definition of the same term, if any, in ETSI TS 103 636-1 [i.14].

3.2 Symbols

For the purposes of the present document, the symbols given in ETSI TS 103 636-1 [i.14] apply.

NOTE: A symbol defined in the present document takes precedence over the definition of the same symbol, if any, in ETSI TS 103 636-1 [i.14].

3.3 Abbreviations

For the purposes of the present document, the abbreviations given ETSI TS 103 636-1 [i.14] and the following apply:

NOTE: An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in ETSI TS 103 636-1 [i.14].

IE Information Element
TLV Type Length Value coding

4 Overview

4.1 Introduction

The present document is used to develop new features of the DECT-2020 NR releases. It provides the technical information on each proposed new feature and impact analysis to the specifications in clause 5. Clause 6 documents an initial plan how to divide the new features between the (coming) releases.

4.2 Uplink data transmission without association

This is a mechanism is intended for the very-low power devices that produce uplink data only. Examples of such applications are asset tracking and water metering. In asset tracking, battery size can be very limited due to cost and size requirements, whereas in water meters long operating times are needed and mains power is not allowed due to regulation.

4.3 DECT-2020 Configuration data distribution

This functionality is intended for distributing configuration/management data in (large) mesh networks. Such data is typically the same for several or all radio devices in a network.

4.4 Access without security credentials to enable RD authentication and key distribution

This feature is about enabling network operation with an RD that does not have valid MAC security credentials. Typical use case is adding a new RD to a network.

4.5 Paging support option to association request

This feature allows communicating if paging is supported or not by the RD that is associating.

4.6 Lifetime to association request

This feature is about communicating lifetime of association between an RD_{FT} and RD associated with it.

4.7 Opportunistic improvement of downlink routing efficiency

This feature improves the downlink routing efficiency under certain scenarios by reducing the required number of transmissions to reach the destination RD.

4.8 Source routing for downlink efficiency

This feature targets to improve downlink routing using source routing principle, registration of nodes and error reporting. The feature can be used as an alternative or potentially in co-operation with the feature of clauses 4.7 and 5.7 opportunistic improvement of downlink routing efficiency.

4.9 Selective downlink source routing

This feature provides means to limit the number of registrations and the downlink packet routing overhead related with the downlink source routing.

4.10 IPv6 Address configuration

This feature targets to autonomous IPv6 unicast address generation by the (mesh) network nodes to avoid address generation/validation related signalling node-by-node, using the prefix information disseminated by the Sink to its cluster tree. Sink and node addresses are used in the address generation to enable directing the downlink traffic to the correct Sink.

5 DECT-2020 NR improvement features

5.1 Introduction

Each new feature is shortly introduced, its impact to the DECT-2020 NR is discussed and the specifications required to be modified are identified. The references to the original contribution and other related material of the feature are given.

5.2 Uplink data transmission without association

5.2.1 Introduction

The current, release 1, model of the MAC association process is the basic assumption that RD is constantly associated with the other RD so that route between RD and backend is constantly maintained and data may flow in either direction. This is independent of whether the network operates in mesh or star topology.

When associated, RD performs reception of all scheduled resources, and if those are not allocated, the RD performs reception of cluster beacon messages and performs potential uplink transmissions on Random Access resources. The reception of the beacons will introduce the minimum activity for the receiver depending on the beacon period which can be for e.g. 4 seconds. The benefit of this is that the RD can be addressed in the downlink at least within the beacon period and RD can operate also in FT mode to provide connectivity to other devices.

The drawback of this is that in certain applications where device power consumption is the most critical and data is uplink only, monitoring beacons on every beacon period is introducing unnecessary activity, reducing the maximum battery lifetime.

Naturally, an RD may perform association before uplink data transmission and then release the association, without establishing any other association. This option is illustrated in Figure 5.2-1, where RD would remain out of the connectivity, i.e. practically power off from the rest of the system's point of view.

If the amount of data to be transferred between association establishment and association release is very small, e.g. under 100 bytes, it is apparent that signalling needed to establish association and releasing it is a significant overhead and thus consumes a significant amount of energy from the RDs battery. Additionally, if the number of devices in a certain location is very high, e.g. in the case of asset tracking, the needed signalling can consume the non-negligible amount of radio resources and resources of RD FT.

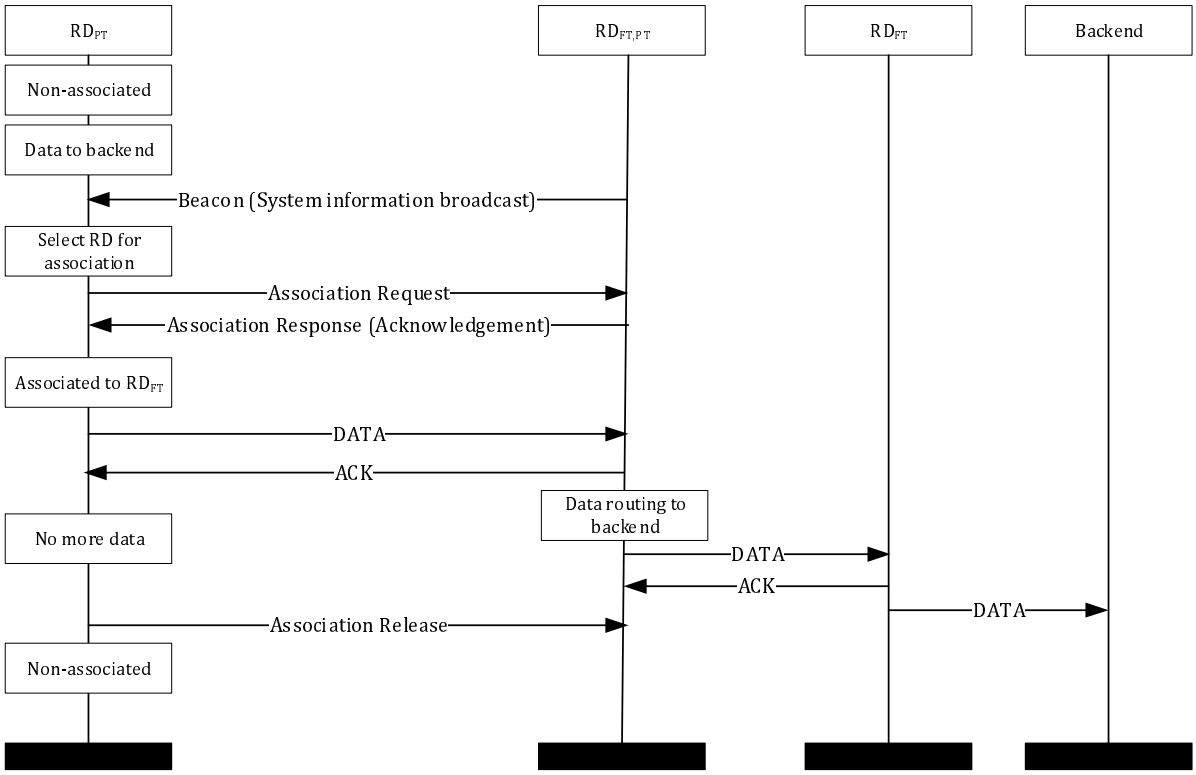


Figure 5.2-1: Data transfer for with association and association release

5.2.2 Impact analysis

To optimize data transfer presented in Figure 5.2-1, for very efficient energy operation, the DECT-2020 NR specification should support data transfer without association. Figure 5.2-2 presents the signalling chart of transmitting uplink data without association.

The RD process for sending data without association is the following:

1) Search network and cluster beacon to detect at least one RD operating in FT mode.

2) If multiple RDs in FT mode are found select RD for uplink data as selecting RD in for association as defined in clause 5.1.4 in ETSI TS103 636-4 [i.1].

3) Apply configuration received from Network/Cluster beacon message including RACH resources.

4) Send data from application specific Endpoint with target address in routing IE set to "any sink" or RD ID that is the final target of the data to valid RACH resources. Uses Unicast header option in MAC header that contains both 32-bit receiver and transmitter address for user plane data.

5) When RD in FT-mode receives the data, it can recognize from short RD ID at PHY header that it does not have association with the sending RD. FT receives full 32-bit RD ID of the transmitter from MAC Unicast header.

6) RD in FT-mode sends Acknowledgement to the data transmission. RD may include some application or any other DECT-2020 NR protocol configuration data to the response.

7) After reception of the ACK and possible protocol actions, the RD goes to sleep if no more data is available for sending.

8) RD in FT mode makes routing decision based on routing IE as defined today and initiates data transfer towards backend or target RD.

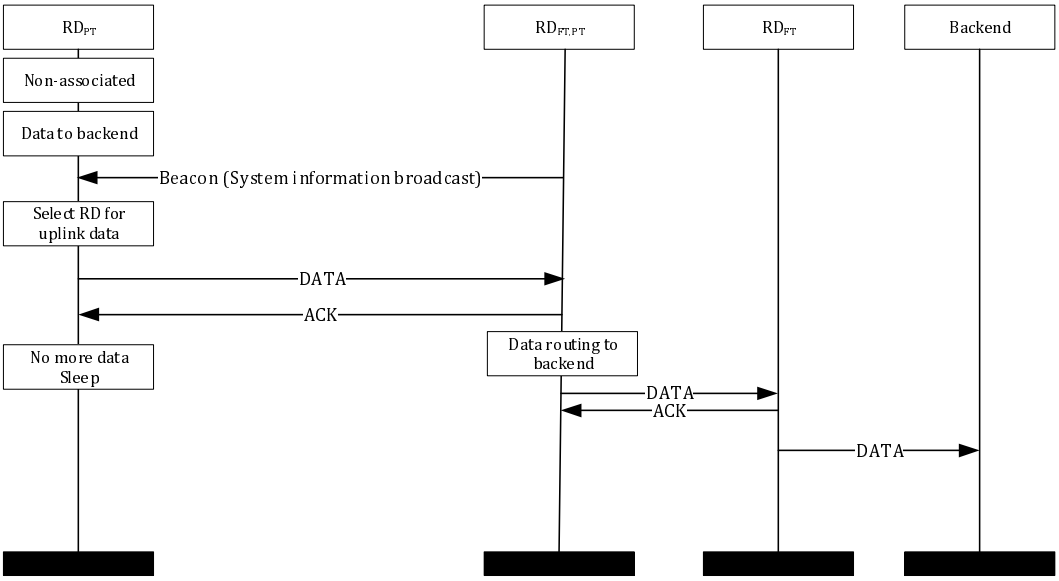


Figure 5.2-2: Data transfer for without association

It should be noted that data transfer without association has the following pre-requirements and limitations:

a) RD authentication and MAC layer security keys as well as CVG end to end security keys needs to be obtained in advance.

b) Downlink data transfer is very limited. RD operating in FT mode may provide application or DECT-2020 NR configuration as part of ACK response message, that may initiate normal association. Otherwise, device cannot be addressed in DL.

c) Due to limitations of DL traffic End to end ARQ on CVG layer cannot be used.

5.2.3 Conclusion

When analysing needed modifications, one can note that above process is not necessarily disallowed but description of the process does not exist.

Needed modifications are expected to be following:

- ETSI TS 103 636-1 [i.3]: generic description of this transmission process.
- ETSI TS 103 636-4 [i.1]:
  - Clause 5.1.4 Selecting RD for association: add text that process is also valid for selecting RD for transmitting data without association.
  - Clause 6.3.3.3 Unicast Header, add clarification to the text that this MAC header option should be used.

For the original contribution, see [i.2].

5.3 DECT-2020 Configuration data distribution

5.3.1 Introduction

The basic reason for this proposal is that obtaining configuration/management data always individually from backend by each RD is not efficient in large mesh networks. The configuration data is typically the same for several or all radio devices in a local network, so any RD could share the data with its neighbours. Capability for RD_{PT} to request configuration data from any RD_{FT} locally would be both faster and less resource consuming than acquiring the data individually from backed. For example, those RDs that join the network as they are switched on or when moving from elsewhere, would be served by their direct neighbours.

Current DECT-2020 NR MAC specification already contains "application sequence number" in the Route Info IE. DECT-2020 NR MAC defines:

> "*The application sequence number provides identification sequence number for network level application data that needs to be distributed in the DL direction to all members of the mesh network. The application sequence number is used by the RD associated to its next hop to identify whether the application data has changed compared its current application data. If sequence number is increased the RD requests the application data, from its next hop, i.e. RD to which it is associated.*"

Thus, the application sequence number is a version field for higher layer information, e.g. configuration/management data. The configuration/management data messaging structure is proposed to gain interoperability.

5.3.2 Impact analysis

The architectural overview of the configuration data distribution is described in Figure 5.3-1.

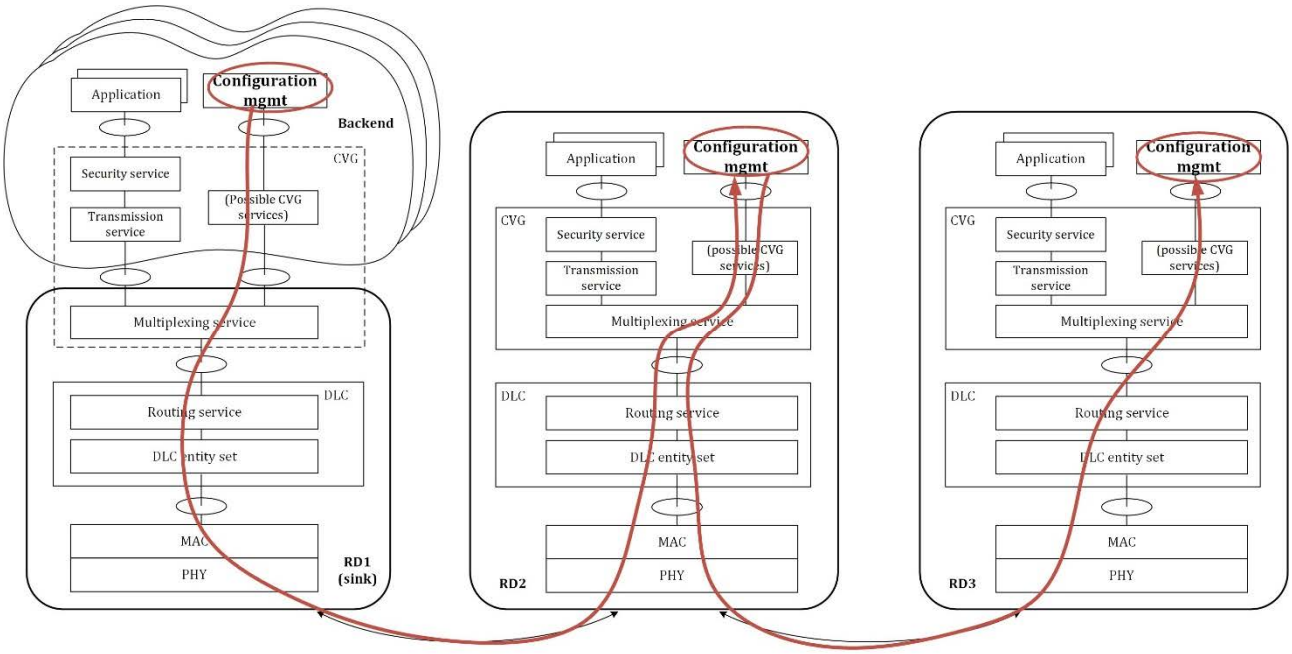


Figure 5.3-1: Architecture

There is a (implementation specific) configuration management entity in possible backend and RDs. The configuration data is sent from backend for distribution in a local radio network through the Sink (RD1). The distribution covers all devices in the radio network - including those that join later.

Current MAC [i.1] and DLC Convergence layer [i.4] protocols are sufficient to carry the higher layer data. Thus, there is no need to specify additional messages, but the configuration data looks like normal data for DECT-2020 layers 1 to 4. An example message sequence chart in Figure 5.3-2 describes how the configuration/management data distribution protocol according to the proposal in general works.

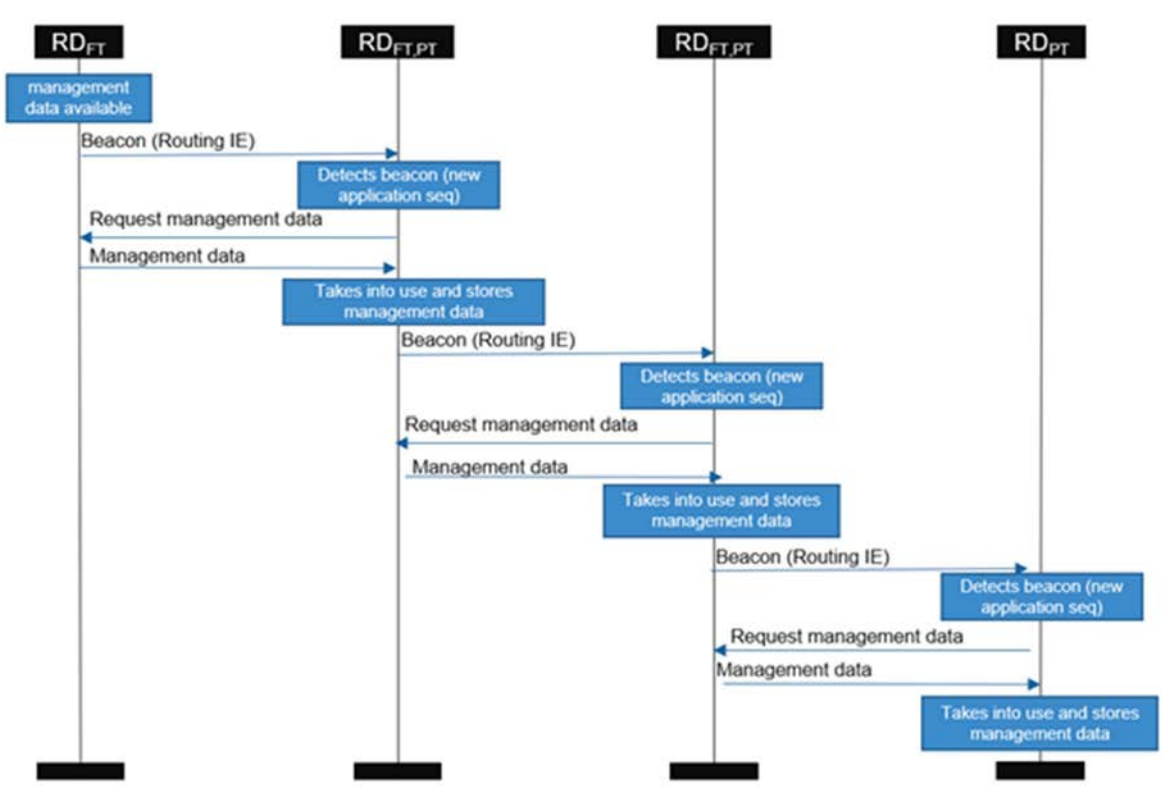


Figure 5.3-2: Message sequence chart for configuration/management data distribution

Each request and management data delivery are considered as higher layer data for DECT-2020 radio, so they are not identified by lower layers of DECT-2020. A specific Endpoint multiplexing value is reserved for the purpose of multiplexing these messages to a proper entity.

The configuration/management messages are identified with EP value 0x8001 and the message is carried as CVG payload. The configuration message contains a header octet (Figure 5.3-3) and further contents depending on the *Configuration message type* field.

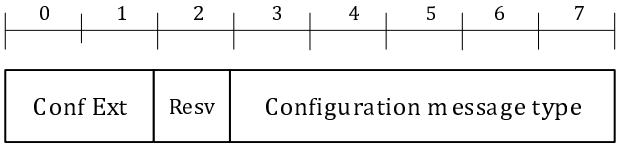


Figure 5.3-3:Header octet for Configuration messages

Two *Configuration message types* are identified: *Configuration data request* and *Configuration data response*.

Configuration data request contains only the Header octet (Figure 5.3-3).

Configuration data response is sent as a response to *Configuration data request*. The structure of the *Configuration data response* is depicted in Figure 5.3-4. The Header octet is as in Figure 5.3-3. *Nr_of_TLVs* field defines how many TLVs are included. The *Type* of the TLV is applied with an Endpoint value, which can be e.g. company specific, allowing the *EP specific structure* (Value) field to contain a company specific data structure. While there can be multiple of TLVs, there can also be multiple of different companies and/or applications involved with the same configuration data distribution.

Header (1 octet)	Nr of TLVs (1 octet)	Type - EP (2 octets)	Length (2 octets)	EP specific structure (N octets)

Figure 5.3-4: Data structure for configuration data

5.3.3 Conclusion

Needed modifications are expected to be following:

- ETSI TS 103 636-1 [i.3]:
  - Add a brief description of this feature.
- ETSI TS 103 636-5 [i.4]:
  - Reserve Endpoint mux value (0x8001) in clause A.1.
  - Define the Configuration data structure in clause A.1.

For the original contributions, see [i.2] and [i.10].

5.4 Access without security credentials to enable RD authentication and key distribution

5.4.1 Introduction

In normal operation, DECT-2020 network has MAC security activated. Also, network and cluster beacons are ciphered, and integrity protected. This feature enables an authentication and/or provisioning process between RD and the backend.

The new feature to provide access for RDs without security credentials should:

- be enabled when needed by the network owner;
- be enabled only at a certain location of the network;
- support different authentication protocols.

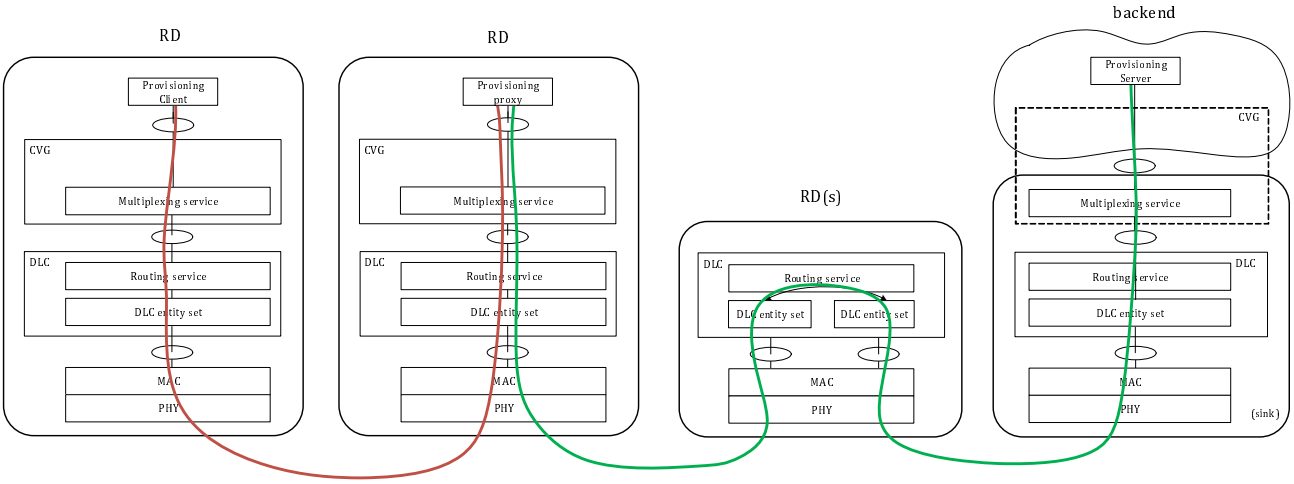
The technical proposal is based on sending a new type of beacon message, Joining beacon, by dedicated RD_{FT} or $RD_{FT,PT}$, named here as Provisioning proxy RD(s). The RDs looking for the access without credentials, are named Provisioning client RDs. The communication without network credentials is limited to happen between the Provisioning client RD and the Provisioning proxy RD. The Provisioning proxy RD forwards the communication between the Provisioning client RD and the Provisioning Server, which may locate e.g. in an external network. Once the authentication and/or provisioning protocols are completed and the Provisioning client RD has received the network credentials, it can join the network in the same way as RDs with credentials normally do.

5.4.2 Impact analysis

System Architecture

Figure 5.4-1 illustrates the communication path and the functional blocks of DECT-2020 NR or above participating in the feature. Provisioning entities controlling the communication are involved at three parts of the system:

- The Provisioning client RD's management entity controls selecting the network, unsecure association with the Provisioning proxy RD and the communication to complete the authentication and credentials procurement.
- The Provisioning proxy RD's management is responsible for configuring Joining beacons and other needed messaging for the unsecure association. It also forwards the messages between the unsecured and secured networks, and if needed, filters the communication.
- The Provisioning server at the backend controls the authentication and/or provisioning according to the selected protocol and if the Provisioning client RD is approved, sends the network security credentials.



NOTE: The red line refers to the communication path without link security, and the green to the path with secured links.

Figure 5.4-1: Joining signalling flow

Joining Beacon

Any RD_{FT} or $RD_{FT,PT}$ could be configured to transmit new type of beacon message (new MAC IE type), Joining Beacon. Enabling/disabling per RD gives control for the security credential delivery in terms of time and location. The Joining beacon is sent unencrypted.

The Joining beacon transmission period, frequency channel and possibly additional data can be selected by the system operator, depending on the use case requirements. These may also be defined in application specific access profiles of DECT-2020 NR.

Joining Beacon is proposed to use the Beacon Header, clause 6.3.3.2 in ETSI TS103 636-4 [i.1] like the other beacon messages, see Figure 5.4-2. It is up to the system operator if the Network ID is the same as used in the secured network operation. The same applies with the Transmitter Address, which in Joining Beacons could be generated e.g. a random Transmitter Address for the open communication.

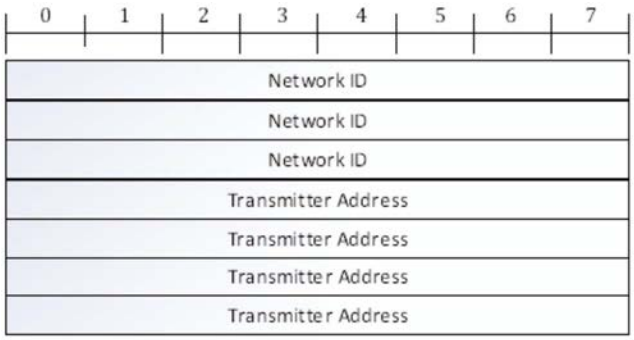


Figure 5.4-2: Beacon Header as in MAC spec

Joining beacon needs to be allocated a MAC IE Type, which is then used in the MAC Mux header to identify the Joining beacon. The MAC Mux header type with variable length is required (see the usage of multiple network channels and context identifier below). The applicable MAC Mux header is defined in the clause 6.3.4 MAC multiplexing header [i.1] and shown below in Figure 5.4-3.

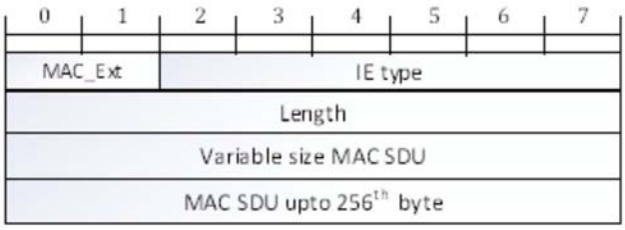


Figure 5.4-3: MAC Mux header with one octet length indicator as in MAC spec

The Joining Beacon IE proposal is described in Figure 5.4-4. The field "Network beacon channels" defines how many channels (1 to 4) are listed in the next octets. Each channel requires two octets (13 bits + padding) as in the current MAC specification. The field "Number of EPs" defines how many EndPoint mux values are included. The EP mux values are used to help the Provisioning client RD to decide whether the network is useful for it to join or not. The EP mux values may be used to identify the available authentication protocol(s), and also other context information, like available services.

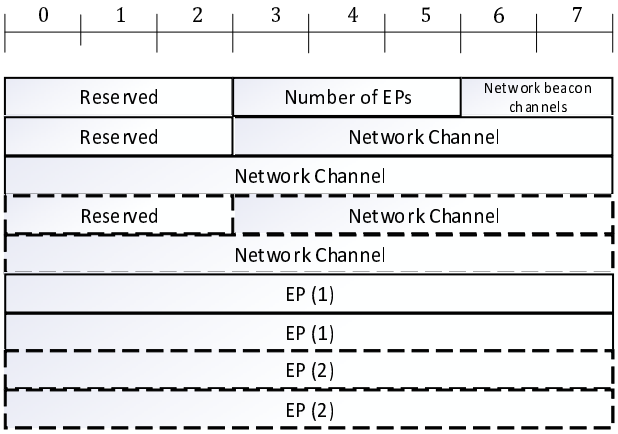


Figure 5.4-4: Joining Beacon IE

The Joining beacon may also be appended with high layer data by including an IE carrying higher layer signalling.

Protocol for delivering the security credentials

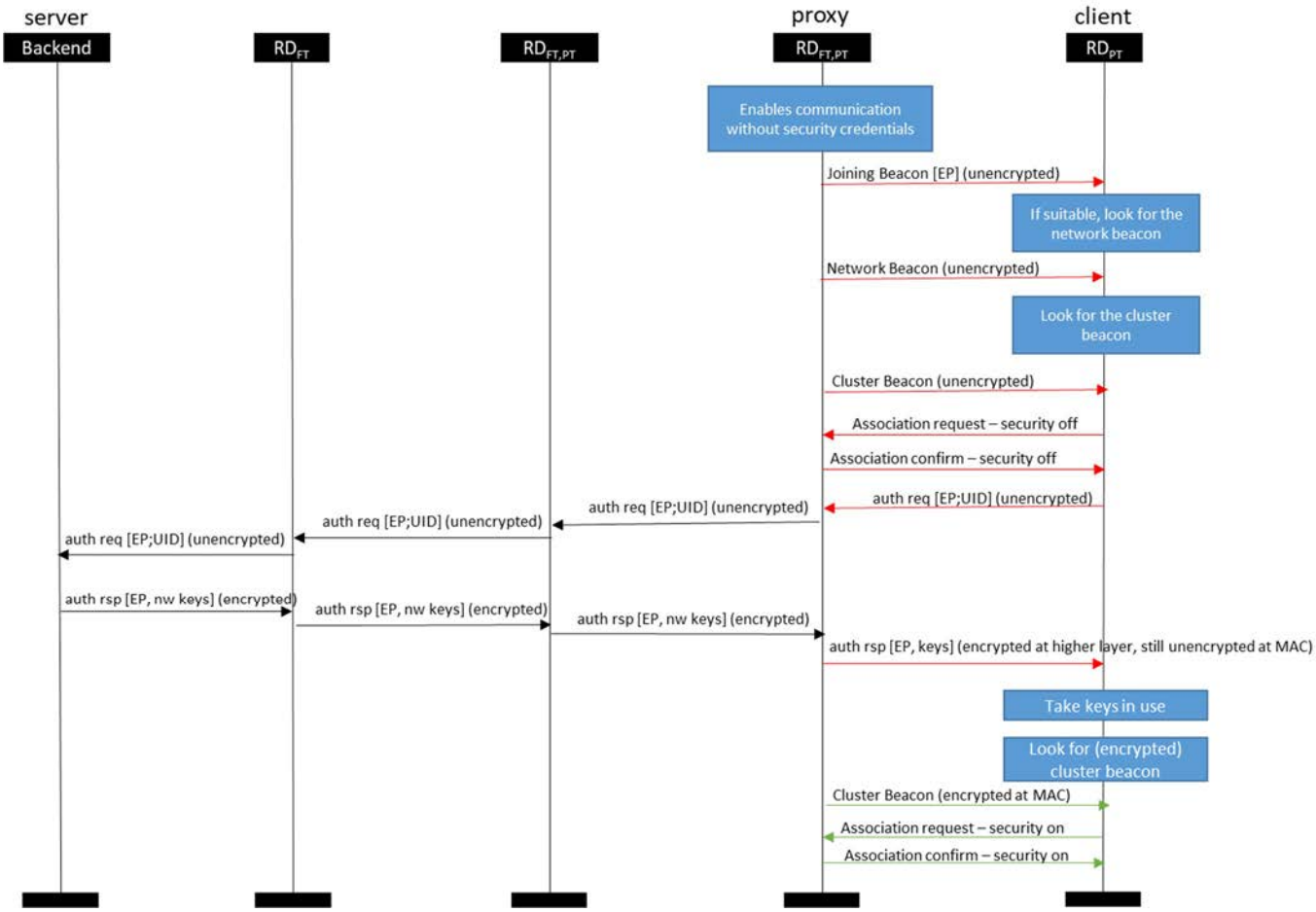
The intention of the proposal is that different kind of higher layer authentication protocols can be used by applying the Joining beacon and the existing standard DECT-2020 NR messages. The standard needs to define the protocol from the Joining beaconing until the unsecure association. The message exchange within the unsecure communication depends on the higher layer provisioning/authentication protocol. The proposal is that:

- In addition to the Joining beacons, the Provisioning proxy RD sends also unencrypted Network beacons on the announced network channel(s) and unencrypted Cluster Beacons on the announced Cluster channel.
- The unencrypted Network beacons instruct the frequency channel and timing of the Cluster beacons.
- The Cluster beacon allocates RACH to enable the Provisioning client RD to send Association request (on the cluster channel).
- The Provisioning proxy RD responds with Association response message.
- The Provisioning client RD and the proxy RD communicate using the given EP until the unsecure communication is closed.

An example message sequence chart to apply the protocol messages is shown in Figure 5.4-5. This example uses pre-shared keys as a basis:

- The Provisioning proxy RD sends Joining beacon messages containing information on the network channel(s) and an EP mux value that refers to a provisioning protocol using pre-shared keys.
- The Provisioning client RD selects the network based on the given EP value in the received Joining beacon.

- The Provisioning proxy RD sends unencrypted network beacons and cluster beacons (with RACH allocation) to allow communication with Provisioning client RDs.
- The Provisioning client RD scans for the network beacons to get instructions to find the cluster beacons.
- The Provisioning client RD receives a cluster beacon.
- The Provisioning client RD sends unsecured Association Request to the Provisioning proxy RD.
- The Provisioning proxy RD responds with unsecured Association Response.
- Upon receiving approval for the unsecured association, the Provisioning client RD sends the authentication request using an applicable EndPoint value and RD's Unique Identifier (UID).
- The Provisioning proxy RD forwards the joining request through the secured DECT-2020 network to the Provisioning server a backend, using backend ("anysink") address and the same EP as the client.
- The Provisioning server identifies the Provisioning client RD based on the request and decides to approve the joining.
- The Provisioning server encrypts the network keys, using the pre-shared keys and sends them back to the Provisioning proxy through the DECT-2020 network, using the same EP.
- The Provisioning proxy RD routes the network keys over the unencrypted link (credentials still ciphered with the pre-shared keys) to the Provisioning client RD, using the same EP.
- The Provisioning client RD deciphers the received network keys using its pre-shared keys.
- The Provisioning client RD joins the network using the security credentials like any RD having the credentials.



NOTE: The red lines refer to communication unencrypted by MAC, the green encrypted by MAC.

Figure 5.4-5: Message sequence chart for delivering the security credentials

Why not just using unencrypted Network Beacon, instead of a new Joining Beacon?

This is an alternative, but has the following pros:

- The Joining beacons may be dedicated on specific frequency channel, thus reducing RD search time to identify possible networks.
- The Joining beacon transmission period can be designed separately of network/cluster beacons transmission periods, again reducing RD search time.
- The short Joining Beacon may be sent relatively often without significant energy consumption of the transmitting device or load to the spectrum.
- Joining beacon transmission power can be adjusted so that it is only detected in a preferred location, i.e. coverage of the beacon can be smaller than network and cluster beacon.

5.4.3 Conclusion

Needed modifications are expected to be the following:

- ETSI TS103 636-4 [i.1]:
  - Define signalling framework:
    - Joining Beacon.
    - Unencrypted RD_{PT} and RD_{FT} communication sequence.

- ETSI TS 103 636-5 [i.4]:
  - Assign EP mux field value(s) to identify authentication protocol signalling.

For the original contributions of this feature, see [i.2] and [i.7].

5.5 Paging support option to the association request

5.5.1 Introduction

Currently, paging is supported by default for every device and there is no option to disable it. This feature would introduce an option to control the paging.

5.5.2 Impact analysis

Each RD in the DECT-2020 network is expected to listen to cluster beacons of their parent RD_{FT} . Downlink data for a member is indicated in the cluster beacon. Therefore, the paging speed of an RD is assumed to follow the cluster beacon period length, multiplied by the number of hops from the sink to the RD.

The proposal contains a bit included in the Association Request Message IE, to indicate whether the associating RD does or does not support paging. In case the RD supports paging, the operation does not change from the above-described operation of specification release 1 [i.1]. If the RD indicates that paging is not supported, DL data transmission cannot be initiated by using paging, rather DL data transmission may only be initiated to response to uplink activity.

5.5.3 Conclusion

Needed modifications are expected to be the following:

- ETSI TS 103 636-1 [i.3]:
  - Add a brief description of this feature.
- ETSI TS 103 636-4 [i.1]:
  - Clause 6.4.2.4 Association request message included with the paging support indication.

For the CR of this feature, see [i.5].

5.6 Lifetime to the association request

5.6.1 Introduction

Currently, association requests or association response message doesn't contain Association lifetime field, so FT assumes that the association is valid until reception of the Association Release message or determining DL communication failure as not receiving a response (HARQ feedback or other UL transmission) to the DL transmission from the PT side. Currently, it is left for FT implementation to determine the criteria when releasing association due to DL communication failure.

The association lifetime field would define, how long FT side should keep the association valid even though there has not been any uplink activity from the PT. Any successful transmission in either direction would reset the association lifetime timer to the original value.

In high mobility cases, the transmission of the association release may fail, resulting that FT may maintain association information for unnecessary time. In case PT has history knowledge of the mobility and durations of association it could indicate the most optimum lifetime value, providing better means for FT to clear old inactive associations in such scenarios.

Additionally, when combined with clause 5.5 Paging support option to association request, an associated but inactive PT may maintain association without the following paging from cluster beacons, and thus skip beacon reception, until the lifetime is expiring.

5.6.2 Impact analysis

Association request message IE is proposed to extend with a Lifetime parameter to tell how long the association is valid for a device in case of inactivity. Lifetime is updated after every successful Transmission or reception from the device. There could be a default value for the Lifetime parameter.

5.6.3 Conclusion

Needed modifications are expected to be following:

- ETSI TS 103 636-1 [i.3]:
  - Add a brief description of this feature.
- ETSI TS 103 636-4 [i.1]:
  - Clause 6.4.2.4 Association request message included with parameters to describe lifetime request.

For the original contribution (CR) of this feature, including two options on using 6 or 8 bits for the Lifetime, see [i.6].

5.7 Opportunistic improvement of downlink routing efficiency

5.7.1 Introduction

Mesh network topologies can support high device densities and autonomous routing to provide the ability to quickly adapt to changes in the device environment, traffic load and propagation conditions.

In a cluster tree topology, there is a clear parent-child relationship at every link where each device individually decides its next hop, i.e. parent, to the sink. Once a device is connected to an RD FT mode, it can send data in the uplink to the internet through an FT node with a connection to the backend. Likewise, the backend can send data in the downlink to the device. An RD can send data in uplink following the most optimal path just by forwarding the packet to the RD_{FT} which is associated to. However, in the downlink direction, there is no information about the direct route to reach the RD.

The current solution in release 1 is the restrictive flooding in the network, i.e. an RD_{FT} forwards the packet to every RD_{FT} having associated members, that are associated with it. This solution has inefficiency since the packet is transmitted to branches of the network tree where the destination device is not present; furthermore, it is more inefficient the larger the network is, since the number of unnecessary transmissions increases. On the other hand, restricted flooding ensures that the downlink packet will always reach the destination node independently of the uplink activity of the nodes or their mobility. Thus, this is the only possible solution for a random DL packet towards an RD whose location is unknown.

Most applications require very little downlink traffic to devices, so the inefficiency of flooding routing in downlink is not an obstacle for many use cases. However, there are some situations where downlink traffic requirements increase. One potential use case is when end-to-end reliability is ensured by using end-to-end ARQ, this mechanism implies that downlink traffic is generated in response to uplink traffic. It is important to reduce the number of transmissions required by end-to-end ARQ, so the network capacity is not severely impacted by this feature.

5.7.2 Impact analysis

This proposal makes opportunistic use of uplink traffic to make routing decisions in the downlink. When an RD_{FT} receives an uplink packet, it can learn from which RD_{FT} the packet is coming and create a relation between the routing source address and MAC source address, so when this RD_{FT} receives a downlink packet to that routing destination address, the RD_{FT} can forward the packet only to the RD_{FT} where the uplink packet came from. Figure 5.7-1 illustrates how an RD_{FT} learns downlink routes based on uplink traffic.

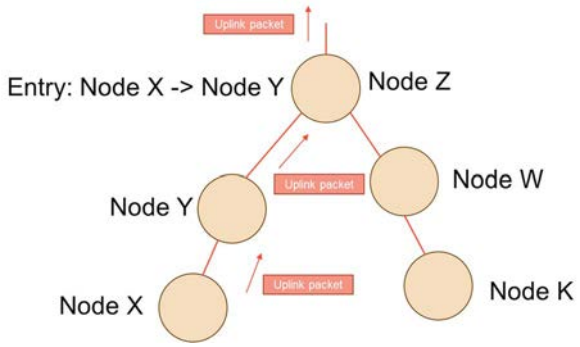


Figure 5.7-1: RD_{FT} learning downlink routes from uplink traffic

Every time an RD_{FT} applies this routing decision, the cluster tree is purged, and this leads to exponential reductions in the number of transmissions the closer this RD_{FT} is to the sink.

The main challenge of this proposal is the mobility scenarios and how long an entry is valid. Since RDs can reassociate to a different RD_{FT} if the channel or traffic load conditions change, purging the cluster tree with obsolete routing information would mean losing the packet. Therefore, these uplink and downlink routing entries are time-sensitive and best suited for bidirectional traffic, i.e. uplink and downlink traffic simultaneously from/to the same device.

RD_{FT} nodes should prioritize keeping track of uplink packets that expect a prompt downlink response, for two main reasons. Firstly, the destination node may change its route to the sink over time, so the less time that passes between uplink and downlink packets, the less likely a route change is to occur. Secondly, the amount of addresses a device can store is limited. The need for immediate responses can be implicitly or explicitly stated.

- **Implicitly:** The RD_{FT} can inspect the upper layer content to know if the packet requires a downlink response. For example, when the packet contains a CVG ARQ Poll message. However, this would lead requirement that routing devices analyse CVG protocol headers.
- **Explicitly:** A flag in the routing header of the DLC layer can indicate that the application layer expects a downlink response for this packet. This information could be set by the original sender of the packet.

It is noted that:

- The proposed method can enable efficient downlink routing for bidirectional traffic without compromising the massive scalability of the mesh topology and without increasing the overall energy consumption of the network.
- The proposed method allows a fully backward compatible way of improving downlink routing efficiency, as every RD in the network does not need to support this method. Devices do not have to have an equal amount of memory or even be aware of whether this feature is used by some RD of the network.
- There is no need to define any extra signaling to maintain route tables, including how often such messages are sent, how extensive signaling can be avoided or what are procedures if routing table is not valid.

5.7.3 Conclusion

The needed modifications are expected to be the following:

- ETSI TS 103 636-1 [i.3]:
  - Add a brief description of this feature.
- ETSI TS 103 636-5 [i.4]:
  - Clause 5.2.8.2 Packet Routing to backend (uplink): Add text for which cases the RD_{FT} might keep routing information from the uplink packet.
  - Clause 5.2.8.3 Packet Routing from backend (downlink): Add text about how to use the routing information in case it is available at the RD_{FT}.

For the original contribution (CR) of this feature, see [i.8].

5.8 Source routing for downlink efficiency

5.8.1 Introduction

Source routing is based on recording the cluster tree topology at the root (sink) and it avoids unnecessary transmissions of the downlink packets by appending the packets with the known route to the destination RD. This feature is described as an alternative or addition to the feature in clause 5.7 Opportunistic improvement of downlink routing efficiency.

After association, RD registers its parent RD to the Sink by sending a DLC Route register control message. The registration is done using either long or both long and short RD IDs. The Sink collects the registering messages, and based on this information, builds and updates its cluster tree's topology.

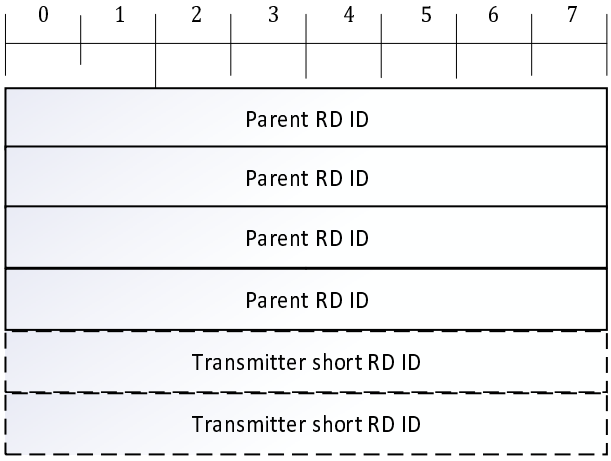


Figure 5.8-1: DLC Route register control message

When a downlink message arrives to the Sink, it appends the step-by-step route to the downlink message. The routing header is proposed to be modified as in Figure 5.8-2 to carry the route information.

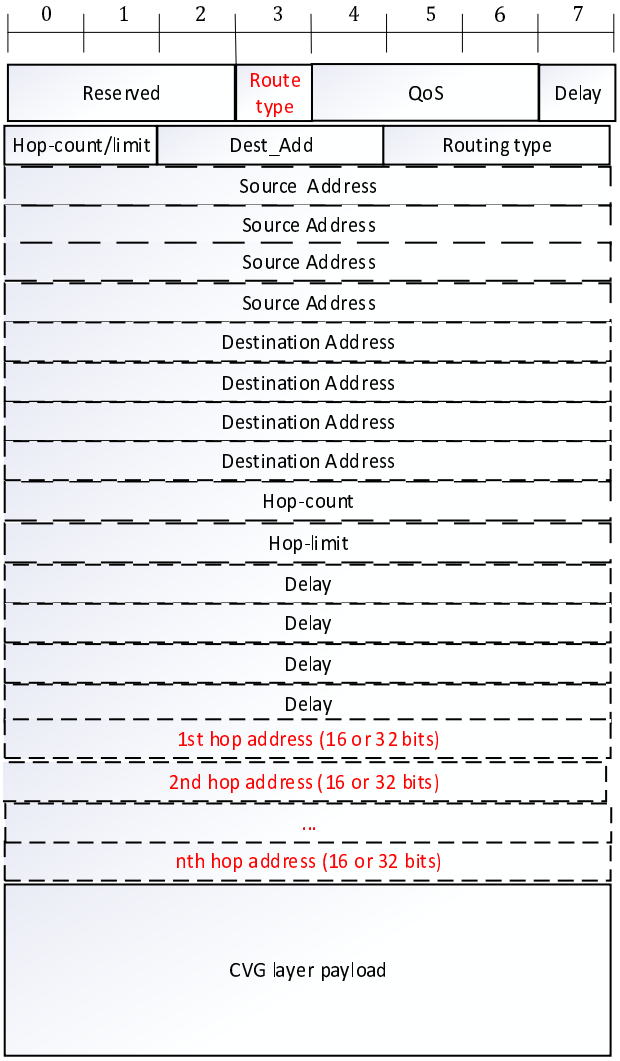


Figure 5.8-2: Routing header with appended route addresses

Upon receiving a downlink packet with the routing header, each RD will check if destination address is its own, if not then if the hop-count equals with the hop-limit:

- If the hop-count is less than the hop-limit, the RD forwards the packet to the next hop address appended in the packet.
- If the hop-count equals with the hop-limit, the RD forwards the packet to the destination address, which expected to be an associated member of the RD.

In case the RD does not have an association with the next hop or destination address RD, the RD sends DLC route error message to Sink. The DLC route error message is proposed as in Figure 5.8-3.

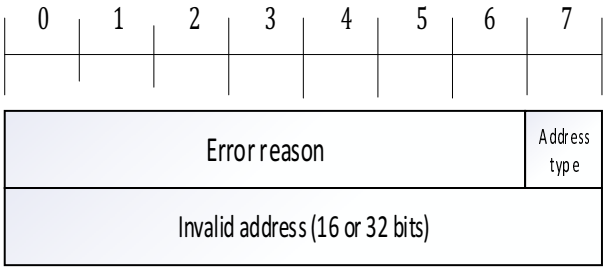


Figure 5.8-3: DLC Route error control message

Table 5.8-1: Route error field coding

Parameter	Description
Error reason	Route lost, Route released, reserved
Address type	0: 16 bit address 1: 32 bit address
Invalid address	Address which was invalid in specified route. The originator can then inspect the route by checking parent child relationship. If the originator cannot link parent and child anymore, it should not modify the route as it might have updated already

5.8.2 Impact analysis

This feature makes it possible to avoid flooding in sending downlink packets, at least a significant reduction of redundant packet transmissions caused be the flooding. The downside is the traffic caused by the registration messages - any change in the topology causes need to update the registrations for the Sink. Thus, the use of the source routing needs to be considered based on the characteristics of the network and additional registration load should not prevent devices to perform re-associations to optimize their next hop and uplink path towards Sink. Furthermore, in case that 16-bit address would be used in source routing, the effects needs to be understood in larger scale networks as source address is no longer globally unique rather unique between two pairs associated to each other.

5.8.3 Conclusion

The needed modifications are expected to be the following:

- ETSI TS 103 636-1 [i.3]:
  - Add a brief description of this feature.
- ETSI TS 103 636-5 **[**i.4**]:**
  - Modify DLC routing header to enable indicating the presence of the route addresses and to append the needed number of route addresses.
  - Define DLC route register message and its usage.
  - Define DLC route error control message and its usage.

For the original contribution (CR) of this feature, see [i.8].

5.9 Selective downlink source routing

5.9.1 Introduction

Source routing was proposed in DECT-2020 NR to minimize the need of flooding in downlink messaging [i.9]. However, two downsides of the proposal were seen:

1) high number of registration messages in mobile and/or large networks; and
2) growth packet overhead/length when the route is many hops.

Thirdly, the original source routing proposal did not discuss how the RDs know if the source routing is on or off in the network, whether there is a possibility to control source routing functionality by some means and whether source routing can cooperate with opportunistic routing table generation.

This proposal targets for mitigating the number of registrations and limiting the packet length overhead by delimiting the cluster tree into parts wherein RDs need or don't need to send registration upon association:

- Sink can define which branches of the cluster tree the DL source routing is applied, while others are flooded.
- Sink can define how many hops are included in the DL source routing, while the remaining hops are flooded.

- The "opportunistic DL routing enhancement" [i.8] can be used in the same network, and its useful in the parts of the network wherein the source routing does not reach to.

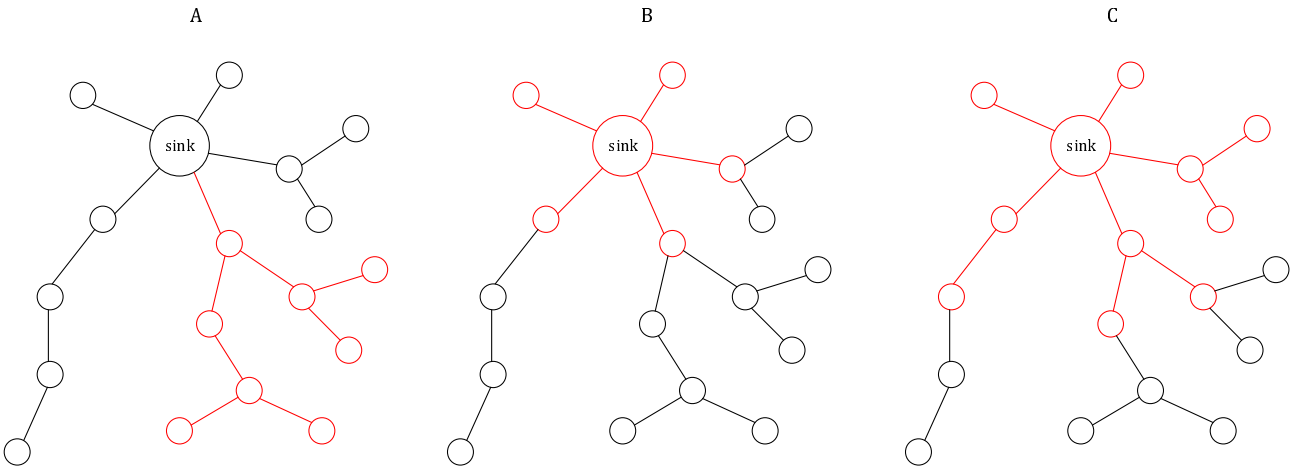


Figure 5.9-1: Examples of selected source routing

Three examples of selecting source routing to serve only part of the network are depicted in Figure 5.9-1. In case A, source routing is limited to one "main branch" only, while in case B source routing is applied to one hop and in case C to two hops.

The technical solution is based on sharing a Source Routing ID, together with the Sink ID. When an RD associates, it checks if the Source Routing ID and the Sink ID from the association target RD are same or different as from the previous association. If either of them or both changes, the associating RD sends registration message to the Sink. If neither of them changes, no registration message is needed. If node changes to a branch that does not use source routing, from a branch where source routing is enabled, the node has to inform the Sink that the previous registration is invalid.

The related protocols are:

- Dissemination of source routing information.
- Registration.
- Route error reporting.
- The actual DL source routing/flooding.

Protocol: Dissemination of Source Routing Information

Sink: In typical case sink includes Source Routing Information (SR IE) in cluster beacon message. This makes source routing configuration be distributed to all associated members. The basic content of Source Routing IE is:

- Source Routing ID (SR ID), containing 32-bit Long RD ID.
- Hop-limit, indicate how many hops from the Sink the source routing covers.
- Hop-count, how many hops from the Sink the current RD exposing the SR information is.
- Validity time of the registrations.

Alternatively, by including SR IE unicast at associations, a Sink can choose which branches of the cluster tree it disseminates the source routing information. Cluster tree may grow and change over the time, so the Sink may want to make changes as well.

EXAMPLE: If Sink shares SR IE only in associations and an RDPT associates to it, the Sink will not send SR IE. However, later when the associated RD changes its role into RDPT-FT, and reassociates, the Sink sends an SR IE and the RD registrates.

Routing RD (RDFT-PT): Each RDFT-PT in the network monitors the SR IE sent by its parent (either Sink or another RDFT-PT), and sets the SR IE in its own beacon. The parameters are set as the following:

- Source Routing ID:
  - If the hop-count =< hop-limit: RD sets its own Long RD ID as SR ID.
  - If the hop-count > hop-limit: RD copies the SR ID from the parent RD as SR ID.
- Hop-limit:
  - Copy from the parent RD.
- Hop-count:
  - Increment by one compared to parent RD.
- Non-Routing RD (RDPT):
  - Leaf-nodes do not disseminate SR information.

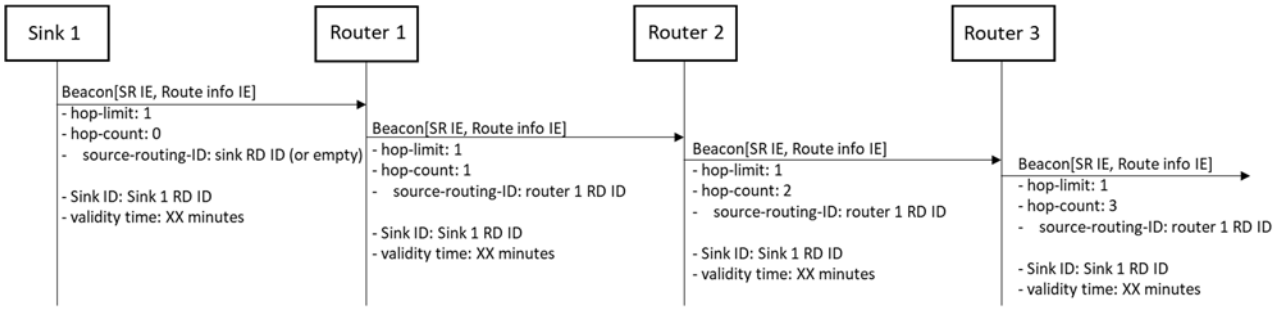


Figure 5.9-2: An example of the dissemination protocol of 1-hop covering source routing

Figure 5.9-2 shows an example of SR information dissemination. In this example, the hop-limit is set to 1, which means that SR ID is changed only by routing RD 1 (in this branch).

Protocol: Registration

Sink collects registrations but does not register itself.

Routing and Non-routing RDs:

- When an RD associates, it compares the given SR IE to that of the previous association:
  - If the Sink ID and Source Routing ID have not changed -> Do not send Route Register message.
  - If the Sink ID or Source Routing ID or both have changed -> Send Route Register message.

In case the validity time expires, the RD has to send Route Register message to sink.

Routing RDs forward the Route Register messages towards the Sink as normal uplink messages.

In registration message the RD sends the source routing ID to the sink.

Protocol: Error Reporting

Sink collects Route Error reports and is updates the source routing accordingly.

Routing RD generates Route Error report in case the next DL hop address is set in the DL message, but it does not exist (anymore). Routing RD forwards received Route Error messages to Sink as normal uplink messages.

Non-Routing RD does not report Route errors.

PDUs

The related messages, information elements and headers are:

- Source Routing IE.
- Route Register control message.
- Route Error control message.
- Routing header.

Source Routing IE

A Source Routing Information Element (SR IE) is defined to contain all the needed information to inform about the local usage of source routing. The fields included:

- Source Routing ID (SR ID):
  - 32-bit Long RD ID of the next source routing RD.

  (Sink ID is included in Route Info IE, so no need to repeat it.)
- Hop-limit:
  - 4 bits to indicate how many hops from the Sink the source routing covers.
- Hop-count:
  - 4 bits to indicate how many hops from the Sink the current RD exposing the SR information is. The value is incremented by one at each hop.
- Validity time:
  - This parameter can be applied to define an interval a re-registration.

Route Register control message

In the Route Register Message defined by Nordic [i.9] it was proposed to signal parent RD ID. However, it is proposed to use Source Routing ID, i.e. the message payload contain the Source Routing ID (no parent RD ID needed). The message is sent to Sink address. Sink address and original sender address (both Long RD ID) are carried in the routing header.

Route Error control message

The Route Error control message defined by Nordic [i.9]. However, the Address type -bit is proposed to be removed and the invalid address to use Long RD ID.

DLC Extension Header

The Routing Header changes proposed by Nordic [i.9] are suggested to be replaced the technical proposal of the DLC Extension Header are described in paper DECT(23)000121r2 [i.13].

5.9.2 Impact analysis

The impact analysis discusses about the needed protocols and protocol data units required by the feature.

By above definition the system can control how many hops the source routing is operating. Any association change below source routing does not introduce any registration message limiting the number of registrations. Further, this avoids large extension of the routing PDU header with next hop addresses and if hop limit is set to 1 as in Figure 5.9-2 there is no need to add any next hop address into routing PDU.

It should be noted that in case as in the example topology in Figure 5.9-3, where the number of associated routers to sink and the next hop routers is 10 routers, which each have 10 routers (tree of 1 + 10 + 100) and the target device is found from any RD after the second hop, the source routing is already limiting flooding by 98 %. However, the exact reduction depends on the topology of the network. Further, as the number of RDs and both uplink and downlink packets are going significantly down, the opportunistic routing can operate efficiently with smaller memory consumption.

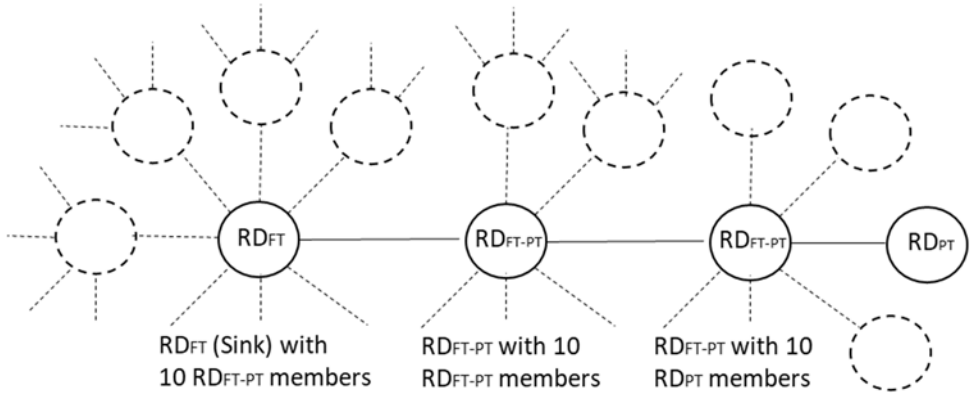


Figure 5.9-3: An example topology, where the Sink has 10 routing members, all of which has 10 routing members as well

5.9.3 Conclusion

- ETSI TS 103 636-1 [i.3]:
  - Add a brief description of this feature.
- ETSI TS 103 636-5 [i.1]:
  - Define the SR IE.
- ETSI TS 103 636-5 [i.4]:
  - Modify DLC headers to carry the next hop addresses, e.g. as defined in [i.13].
  - Define DLC route register message and its usage.
  - Define DLC route error control message and its usage.

For the original contribution of this feature, see [i.12].

5.10 IPv6 Address configuration

5.10.1 Introduction

IPv6 address configuration targets to provide autonomous IPv6 address generation by the nodes of DECT-2020 (mesh) network. This is gained by disseminating the prefix information from the Sink to every node in the network below the Sink. The dissemination is proposed to be done using the DECT-2020 Configuration data distribution mechanism described in clause 5.3. Each node generates the IPv6 addresses, using the fixed or disseminated prefix information, sink address and node's own address.

Address generation

IPv6 address consists of 128 bits. Number of different address formats have been defined by IETF. In DECT-2020 mesh networking, the proposal is that each node creates a Link local unicast address and a "Non-link local unicast address" as explained below.

The standard *Link local unicast address* is composed of the Fixed network prefix (FE80:/64), zeroes and a Link local IID (Figure 5.10-1). The first 32 bits of the Link local IID can be set to 0's, the last 32 are set the Long RD ID of the node. This is compliant with IETF RFC 4921 [i.15].

10 b	54 bits	64 bits
prefix	zeroes	Link local IID

Figure 5.10-1: Link local unicast address

Other than link local addresses "*Non-link local unicast address*" (Figure 5.10-2) can be applied with the IPv6 Global Unicast Addressing (GUA, IETF RFC 3587 [i.16]) or Unique Local IPv6 Addressing (ULA, IETF RFC 4193 [i.17]) standards. These standards define specific network prefixes that are applied in the first 64 bits.

64 bits	64 bits
Network prefix (GUA/ULA)	Non-link local IID

Figure 5.10-2: Non-link local unicast address

The DECT-2020 (mesh) networking specific proposal is that the first 32 bits of the *Non-link local IID* field are set to into the current Sink's Long RD ID and the last 32 bits into node's own Long RD ID. As an exception in the format, the seventh bit of the Sink Long RD ID is always set to zero (i.e. local), as this bit is the universal/local bit of the MAC address definition. In general, the Long RD ID of the nodes (and the Sink) can be based e.g. on EUI-64.

32 bits	32 bits
Sink Long RD ID	Node Long RD ID

Figure 5.10-3: Non-link local IID

Dissemination

The dissemination of the network prefix for the *Non-link local unicast address* is done using the *Configuration data* distribution mechanism (see clause 5.3). Such network prefixes are identified with an Endpoint value (value 0x8004 proposed). The *Configuration data response* containing a network prefix is depicted in Figure 5.10-4.

Header (1 octet)	Nr of TLVs (1 octet)	Type – EP 0x8004 (2 octets)	Length (2 octets)	Network prefix (GUA/ULA) (8 octets)

Figure 5.10-4: Configuration data response containing Non-link local IID prefix

5.10.2 Impact analysis

The dissemination of the prefix information using the Configuration data distribution mechanism and skipping DHCP direct server communication by the nodes are designed to avoid unnecessary traffic in DECT-2020 (mesh) networks.

The nodes can generate the IPv6 addresses themselves, using the information disseminated by each Sink. Including Sink's address into the *Non-link local unicast address* also enables the system to route downlink IPv6 packets for the node to the correct Sink. Of course, this necessitates the change of the IPv6 address every time the node changes from a Sink to another Sink.

5.10.3 Conclusion

The needed modifications are expected to be the following:

- ETSI TS 103 636-1 [i.3]:
  - Add a brief description of this feature.
- ETSI TS 103 636-5 [i.4]:
  - Reserve Endpoint mux value (0x8004) in Annex A for the purpose of identifying the network prefix of the Non-link local unicast address.

For the original contribution of this feature, see [i.11].

6 DECT-2020 NR features for release 2

Table 6-1

Feature	Rel.2	Note
Uplink data transmission without association	X	
DECT-2020 Configuration data distribution	X	
Access without security credentials to enable RD authentication and key distribution	X	
Paging support option to the association request	X	
Lifetime to the association request	X	
Opportunistic improvement of downlink routing efficiency	X	
Source routing for downlink efficiency	X	
Selective source routing	X	
IPv6 Address configuration	X	

History

Document history		
V1.1.1	October 2024	Publication