



VAN

FP6/2004/IST/NMP/2 - 016696 VAN

Virtual Automation Networks

Work Package 3

Wireless in Industries

Task 3.2

Specification of Wireless Communications for
Automation

Deliverable D03.2-1

Specification for wireless in industrial
environment and industrial embedded
devices

Document type	: Report
Document version	: Draft
Document Preparation Date	: 04.04.2007
Classification	: Public
Contract Start Date	: 01.04.2006
Duration	: 28.02.2007



Project funded by the European Community
under the "Information Society Technology"
Programme (2002-2006)

Rev.	Content	Resp. Partner	Date
1.0	Final Draft	ifak (8)	04.04.2007

Everybody please state revision index and short description of what has been done + partners involved and date.

Final approval	Name	Partner
Review Task Level	Lutz Rauchhaupt	ifak (8)
Review WP Level	Christoph Weiler	Siemens (1)
Review Board Level	Christian Schwab	Siemens (1)

Executive summary

This document is deliverable D03.2-1 “Specification for wireless in industrial environment and industrial embedded devices” of the VAN project and it reports the outcome of the work within work package 3, task 3.2 “Specification of Wireless Communications for Automation”. The duration of the task is from month 8 to month 23.

In the previous deliverable of WP 3, D03.1-1 “Status and Analysis”, several wireless technologies and standards have been analysed and four wireless technologies (Bluetooth, Ultra-Wideband, Wireless LAN and IEEE 802.15.4/ZigBee) have been selected to be considered in the following work of WP 3.

This document deals with the relevant aspects for the integration of wireless networks into the open platform and system architecture of VAN. Therefore it contains the contributions of the selected wireless technologies and standards to the VAN Application Service Elements - ASEs.

The document is structured into the following main chapters:

- CH1: Scope
- CH2: Wireless Technologies in VAN Architecture
- CH3: Wireless Contribution to VAN Device Descriptions
- CH4: Wireless Contribution to VAN ASEs
- CH5: Bluetooth Contribution to VAN ASEs
- CH6: Ultra-Wideband Contribution to VAN ASEs
- CH7: Wireless LAN Contribution to VAN ASEs
- CH8: IEEE 802.15.4 and ZigBee Contribution to VAN ASEs
- CH9: Conclusions

Based on the specifications of this document, a detailed design specification and a specification to test wireless solutions in industrial environments will be worked out in deliverable D03.4.

Since Task 3.2 is not completed with the deadline of this deliverable, the document is still subject to revision.

The content of this document is a compilation of contributions from BUT (Brno University of Technology), Phoenix Contact, Schneider Electric, Siemens, and ifak Magdeburg. The task leader was ifak Magdeburg. The document structure and compilation was also performed by ifak Magdeburg.

Contents

1	Scope	9
2	Wireless Technologies in VAN Architecture.....	10
2.1	Preface	10
2.2	Bluetooth.....	10
2.2.1	Application Fields.....	10
2.2.2	Topologies and Device Types	10
2.3	Ultra-Wideband.....	11
2.3.1	Application Fields.....	11
2.3.2	Topologies and Device Types	12
2.4	Wireless LAN	13
2.4.1	Application Fields.....	13
2.4.2	Topologies and Device Types	14
2.5	IEEE 802.15.4 and ZigBee	14
2.5.1	Application Fields.....	14
2.5.2	Topologies and Device Types	15
3	Wireless Device Description Contents	16
3.1	Preface	16
3.2	Applicable Also for Other VAN Technologies.....	16
3.3	Common to All Wireless Technologies.....	17
3.3.1	General Description	17
3.3.2	Physical Radio Description	17
3.3.3	Media Access Description	18
3.4	Specific for IEEE 802.15.4 and ZigBee	19
3.4.1	Parameters	19
3.4.2	Node Descriptor	19
3.4.3	Node Power Descriptor.....	21
3.4.4	Simple Descriptor.....	21
3.4.5	Complex descriptor.....	23
4	Wireless Contributions to VAN ASEs	24
4.1	Preface	24
4.2	Applicable Also for Other VAN Technologies.....	24
4.3	Wireless Device Configuration Class	24
4.3.1	Object Overview.....	24
4.3.2	Formal model	25
4.3.3	Attribute description	25
4.4	Wireless Security Configuration Class	26
4.4.1	Object Overview.....	26
4.4.2	Formal model	27
4.4.3	Attribute description	27
4.5	Wireless Diagnosis Class	27
4.5.1	Object Overview.....	27
4.5.2	Formal model	28
4.5.3	Attribute description	28
5	Bluetooth Contributions to VAN ASEs	29
5.1	Relevant Architecture Elements	29
5.2	Bluetooth Configuration Class	29
5.2.1	Object Overview.....	29
5.2.2	Refinement of Inherited Attributes	29
5.2.3	Formal Model.....	29

5.2.4	Attribute Description.....	30
5.3	Bluetooth Security Configuration Class.....	33
5.3.1	Object Overview.....	33
5.3.2	Refinement of Inherited Attributes.....	33
5.3.3	Formal Model.....	34
5.3.4	Attribute Description.....	34
5.4	Bluetooth Diagnosis Class.....	35
5.4.1	Object Overview.....	35
5.4.2	Refinement of Inherited Attributes.....	35
5.4.3	Formal Model.....	35
5.4.4	Attribute Description.....	35
6	Ultra-Wideband Contributions to VAN ASEs.....	37
6.1	Relevant Architecture Elements.....	37
6.2	UWB Device Configuration Class.....	37
6.2.1	Object Overview.....	37
6.2.2	Refinement of Inherited Attributes.....	37
6.2.3	Formal Model.....	38
6.2.4	Attribute Description.....	38
6.3	UWB Security Configuration Class.....	40
6.3.1	Object Overview.....	40
6.3.2	Refinement of Inherited Attributes.....	40
6.3.3	Formal Model.....	41
6.3.4	Attribute Description.....	41
6.4	UWB Diagnosis Class.....	44
6.4.1	Object Overview.....	44
6.4.2	Refinement of Inherited Attributes.....	45
6.4.3	Formal Model.....	45
6.4.4	Attribute Description.....	45
6.5	VAN Heterogeneous Network Technologies Adaptation Layer.....	46
7	Wireless LAN Contributions to VAN ASEs.....	48
7.1	Relevant Architecture Elements.....	48
7.2	WLAN Device Configuration Class.....	48
7.2.1	Object Overview.....	48
7.2.2	Refinement of Inherited Attributes.....	48
7.2.3	Formal Model.....	48
7.2.4	Attribute Description.....	49
7.3	WLAN Security Configuration Class.....	49
7.3.1	Object Overview.....	49
7.3.2	Refinement of Inherited Attributes.....	50
7.3.3	Formal Model.....	50
7.3.4	Attribute Description.....	50
8	IEEE 802.15.4 and ZigBee Contributions to VAN ASEs.....	52
8.1	Relevant Architecture Elements.....	52
8.2	ZigBee Device Configuration Class.....	52
8.2.1	Object Overview.....	52
8.2.2	Refinement of Inherited Attributes.....	52
8.2.3	Formal Model.....	54
8.2.4	Attribute Description.....	55
8.3	ZigBee Security Configuration Class.....	59
8.3.1	Object Overview.....	59
8.3.2	Refinement of Inherited Attributes.....	60
8.3.3	Formal Model.....	60
8.3.4	Attribute Description.....	61

- 8.4 ZigBee Diagnosis Class 64
 - 8.4.1 Object Overview..... 64
 - 8.4.2 Refinement of Inherited Attributes 65
 - 8.4.3 Formal Model 65
 - 8.4.4 Attribute Description..... 65
- 9 Conclusions..... 68**

List of Figures

Figure 2-1: Possible device types for Bluetooth devices in a VAN domain.....	11
Figure 2-2: Device topology for UWB devices in the VAN domain	13
Figure 2-3 VAN topology with IEEE802.11 and ZigBee devices.....	15
Figure 4-1: Classes contributed by Task3.2 to the Device Config ASE	25
Figure 4-2: Classes contributed by Task3.2 to the Security Config ASE	27
Figure 4-3: Classes contributed by Task3.2 to the Diagnosis ASE.....	28
Figure 5-1: Deduction of the BLUETOOTH DEVICE CONFIG class structure	29
Figure 5-2: Deduction of the BLUETOOTH SECURITY CONFIG class structure	33
Figure 5-3: Deduction of the BLUETOOTH DIAGNOSIS CONFIG class structure	35
Figure 6-1: Deduction of the UWB DEVICE CONFIG class structure.....	37
Figure 6-2: Deduction of the UWB SECURITY CONFIG class structure	40
Figure 6-3: Deduction of the UWB DIAGNOSIS class structure	45
Figure 6-1: Deduction of the WLAN DEVICE CONFIG class structure.....	48
Figure 6-2: Deduction of the WLAN SECURITY CONFIG class structure.....	50
Figure 8-1: Deduction of the ZIGBEE DEVICE CONFIG class structure.....	52
Figure 8-2: MAC superframe structure with GTS slots	54
Figure 8-3: Deduction of the ZIGBEE SECURITY CONFIG class structure	60
Figure 8-4: Deduction of the ZIGBEE DIAGNOSIS class structure	65

List of Tables

Table 3-1:	Logical Device Type	19
Table 3-2:	Frequency Band Coding	20
Table 3-3:	Capability Flag Coding.....	20
Table 3-4:	Supported Power Sources.....	21
Table 3-5:	Application Device Version	22
Table 3-6:	Application Flag Coding.....	22
Table 5-1:	1-bit field of the attribute PhyChannelsUsed	30
Table 5-2:	Inquiry Mode parameter description	31
Table 5-3:	Link_Policy_Settings.....	32
Table 5-4:	Bluetooth security modes	34
Table 8-1:	Number of channels per band	53
Table 8-2:	Transmit power levels.....	53
Table 8-3:	MAC beacon payload format	55
Table 8-4:	CCA mode descriptions	56
Table 8-5:	Security Modes	61
Table 8-6:	Access Control List Entry Descriptor Set.....	61
Table 8-7:	Security Suite Identifiers	62
Table 8-8:	Security levels available to the MAC, NWK and APS layers	63
Table 8-9:	Network Security Material Descriptors.....	63
Table 8-10:	Key-pair descriptor elements	64
Table 8-11:	Capability Information bit-fields.....	66
Table 8-12:	NWK Address Map	66
Table 8-13:	Current Power Mode Codes	67
Table 8-14:	Current Power Source Modes	67
Table 8-15:	Current Power Source Level Modes.....	67

1 Scope

This deliverable deals with wireless technologies and standards which were selected as promising technologies to introduce mobility and flexibility into industrial automation applications. These are Bluetooth, Ultra-Wideband, Wireless LAN and IEEE 802.15.4/ZigBee. Provider based wireless technologies such as GSM, UMTS or WiMAX have not been taken into account in this work package. These wireless technologies are in the scope of work package 7.

The fields of application are briefly described first, since each of the selected wireless technologies addresses a different application field. Afterwards for each technology, the topologies are introduced which are reasonable for the given application field. According to the open platform and system architecture defined in [D02.2-1], the VAN device types are indicated within the given topologies. The specification of the relevant ASEs for these VAN device types is the main subject of this document.

There are a number of attributes which are static and describe the characteristic of a wireless device. These attributes are described first. Thereby, it is distinguished between attributes:

1. which are supposed to be general

It is assumed that the meaning of these attributes is specified in WP2. Here these attributes are just listed or necessary additions are defined.

2. which are general for wireless solutions

These attributes are defined only once and are valid for all technologies. However, in detail there may be additions necessary for one or the other technology. These additions are mentioned in a technology specific sub-section.

3. which are specific for a technology

These attributes are explained in the technology specific sub-section.

Chapter 4 deals with the general contribution to the VAN ASEs from the general point of view of wireless communication. As in the case of the static attributes, some of the ASE attributes are expected to be defined by WP2. In these cases possible additions are proposed.

Chapter 5 to chapter 8 contain the ASE descriptions for the selected wireless technologies. Based on the global VAN device architecture, the relevant function building blocks are described in detail for the device types defined in chapter 0.

The definitions are meant to give other work packages the required information to work out VAN related strategies. Thus, the attributes of the Device Configuration ASEs are necessary to specify a unique VAN engineering strategy or to consider wireless networks in the plug and play concept. The attributes of the Security ASEs are important to develop a general VAN security approach to overcome isolate security solutions which do not fit to each other.

Consequently, this document is based on a close co-operation with work package 2 and provides important information to work packages 2, 6, and 8. In work package 3 the definitions are specified in more detail and are thus the basis for the wireless prototype implementations.

2 Wireless Technologies in VAN Architecture

2.1 Preface

The requirement analysis for wireless communication systems in industrial automation applications (see [D03.1-1]) has shown that the demands differ from application field to application field. That is why several wireless technologies and standards will be needed to meet the requirements of these different application fields.

With reference to [D03.1-1] this chapter shortly describes which application fields are covered best by the chosen wireless technologies. Afterwards the relevant wireless topologies are discussed as well as the position of the wireless network in the context of the overall VAN system architecture. This includes the description of device types defined in [D02.2-1].

2.2 Bluetooth

2.2.1 Application Fields

Bluetooth has its advantages in applications when it comes to bandwidth needs in the order of up to 300 kbps and only a limited number of wireless devices per radio cell. Especially the adaptive frequency hopping together with the transmit power regulation of each individual Bluetooth link makes this technology attractive in applications which need a high density of wireless systems operating independently in the same local area without additional commissioning effort for a detailed frequency planning and device configuration.

Examples for such applications are small to medium standard machines where a wireless communication is needed to a few rotating or linearly moving parts with a limited number of IOs of the machine. An example is given in [D03.1-1] for a bottle filling plant which could alternatively be operated by a Bluetooth point to point connection. The latter has the advantage to better reserve scarce bandwidth for other wireless communication by simultaneously guaranteeing a deterministic behaviour of the communication in the order of 8 to 10 ms.

Other examples exploit the possibility of Bluetooth to operate several communication profiles. Especially devices needing serial communication like some fieldbusses or devices with a serial configuration interface which is still very common in automation systems can be accessed wirelessly. Bluetooth not only provides a transparent channel for Ethernet communication but also provides services for standard serial communication. This is helpful in cases of diagnostics, maintenance and configuration as it is described in [D03.1-1].

A third type of applications for Bluetooth arise from the fact that machine builders need a mobile and temporary connection to their machines during commissioning or maintenance and are simply not allowed to use e.g. the WLAN network of their end customers due to their IT security policies.

In applications where only simple digital or analog input or output channels are needed Bluetooth can be used to only transmit the "raw" digital or analog IO data instead of transmitting a full Ethernet based communication protocol. This makes the wireless transmission very efficient and leads to the fastest possible reaction times. Considering VAN in such a scenario the IO device with Bluetooth interface acts as a VAN-VD whereas the Bluetooth basestation acts as VAN-PD according to [D02.2-1] providing the full VAN functionality for the connected VAN domain.

2.2.2 Topologies and Device Types

The use of Bluetooth within a VAN Domain can lead to different network topologies which are summarized in the following figure.

Due to these different topologies there are different VAN device types involved in Bluetooth communication.

VAN-AP:

A VAN-AP with integrated Bluetooth interfaces provides access of automation devices to the VAN domain.

VAN-MG:

A VAN-MG with integrated Bluetooth interface provides a media gateway between the wired physical layer and the wireless physical layer of a VAN Domain. In order to provide VAN configuration services it has to have a VAN-FS built in but does not contain any automation functions nor application processes.

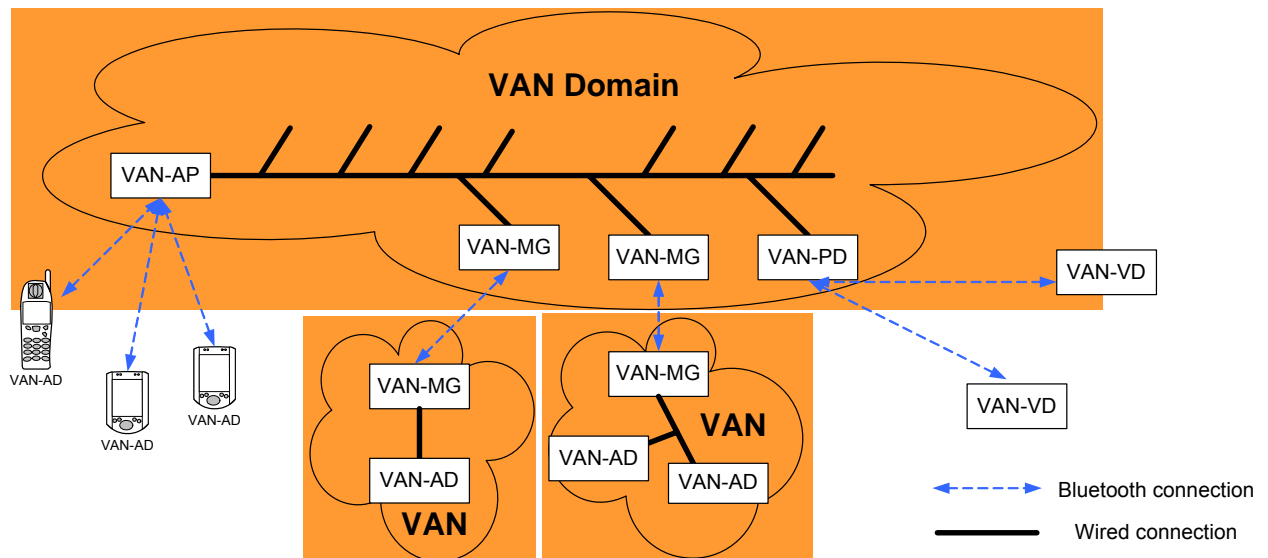


Figure 2-1: Possible device types for Bluetooth devices in a VAN domain

VAN-AD:

Of course a VAN-AD can have a Bluetooth interface and can communicate via a VAN-AP within the VAN Domain.

VAN-PD:

As Bluetooth provides different application profiles and not only transparent Ethernet tunnelling, Bluetooth can also be used for direct IO communication with simple IO devices. To integrate such devices into a VAN Domain, a VAN proxy device (VAN-PD) is necessary.

VAN-VD:

In a Bluetooth proxy scenario the mentioned simple IO devices, which might not be VAN aware, have to be integrated into a VAN Domain via a virtual device (VAN-VD).

2.3 Ultra-Wideband

2.3.1 Application Fields

UWB has many advantages that make it suitable for indoor wireless networks in general and, more specifically, for consumer electronics applications. Basic physics gives UWB an inherent ability to maintain high speed through walls and in cluttered high-multipath environments. UWB is a technology that provides potentially unlicensed operation, simplicity, very low transmit power, multipath and interference immunity, and the capability to deliver data rates in excess of several hundred Mbps all the while consuming very little battery power and relatively small amounts of silicon area, translating to low cost.

UWB systems operate coherently across a wide range of frequency spectrum relative to the centre frequency. The wide relative bandwidth is of key importance because it governs how immune the radio is to multipath interference while simultaneously penetrating walls or other objects. Thanks to its low spectral density, unlicensed UWB radio emissions do not add up to cause harmful interference to other radio systems operating in dedicated bands. In fact, normal propagation attenuation causes the signals to dissipate faster than they can add up. Furthermore, the power spectrum can be adjusted to reduce levels even lower in sensitive bands, such as global-positioning system or personal communications services receivers. That means UWB devices can be co-located with GPS and PCS equipment. Moreover, UWB's low power, wide spectrum and coded waveform make it difficult for eavesdroppers to detect. A further security benefit lies in the ability of an UWB device to detect the range of another UWB device based on round-trip delay information. The high precision of this range information allows a device to reject communications

with another device unless it is at or within an authorized range. Thus UWB receivers can use range information to reduce their level of interference even further and optimize network configuration and traffic flow.

An UWB signal can be typified by a series of low-power derivative of Gaussian pulses. Each pulse is extremely short in duration (10 to 1,000 picoseconds), typically much shorter than the interval corresponding to a single bit. Because of the short duration of the pulses, the frequency spectrum of an UWB signal can be very wide, overlaying the bands used by existing narrowband systems. As a result of UWB's distribution of energy, the spectral density is extremely low. An UWB radio is designed to transmit less than 75 nanowatts of power per megahertz of frequency bandwidth, which is equivalent to an aggregated power of 0.26 milliwatts, in contrast to 30 to 100 mW for 802.11b WLAN radios and 1 mW to 1 W for Bluetooth radios.

Typical applications of the UWB technology are listed below.

- Short-range applications replacing WLAN/Bluetooth
- High-speed audio/video applications
 - High speed digital video transfer from a digicam to a TV screen
 - High definition (HD) MPEG2 between video players/gateways and multiple HD displays
 - Home theatre audio distribution
 - PC to LCD projector
 - Interactive video gaming
- For drive systems: to replace trailing cables and slip rings even for fast rotating machine parts
- Motion control applications in manufacturing
- Location-aware communications applications
 - Inventory control and asset management
 - Tracking mobile objects
 - In sensor networks to monitor industrial automation and control
 - To organize and perform tactical manoeuvres in mobile nodes
 - Heating, ventilation and air conditioning (HVAC) control for home and office environments
 - To augment the GPS indoors to enable location-assisted routing of network data
- High speed data transfer using Wireless USB
 - In devices such as digital cameras, MP3 players, storage devices, scanners and printers
 - Hubs and dongles

2.3.2 Topologies and Device Types

The device types and the topologies depend, mainly, on the underlying application areas. For instance, sensor networks and location-based applications, where UWB might find application from the VAN perspective, are based on the peer-to-peer ad-hoc network topology. A specific example could be a low-rate precise location application in industrial and building automation based on the UWB Impulse Radio (IEEE 802.15.4a) standard. UWB is especially optimized for such low data rate (≤ 1 Mbps) applications, since it offers advantages such as low power consumption, longer range (100+ metres), higher accuracy (sub-centimetre range), higher aggregate throughput (many 1000s of nodes), resistance against multipath, and interference and cost effectiveness. These features make a good case for UWB in comparison to ZigBee-which is too complex, and less reliable, WiFi-which is more expensive and Bluetooth-which is limited by the number of nodes. The figure given below illustrates the possible use of UWB within the VAN domain in terms of its topology and device types.

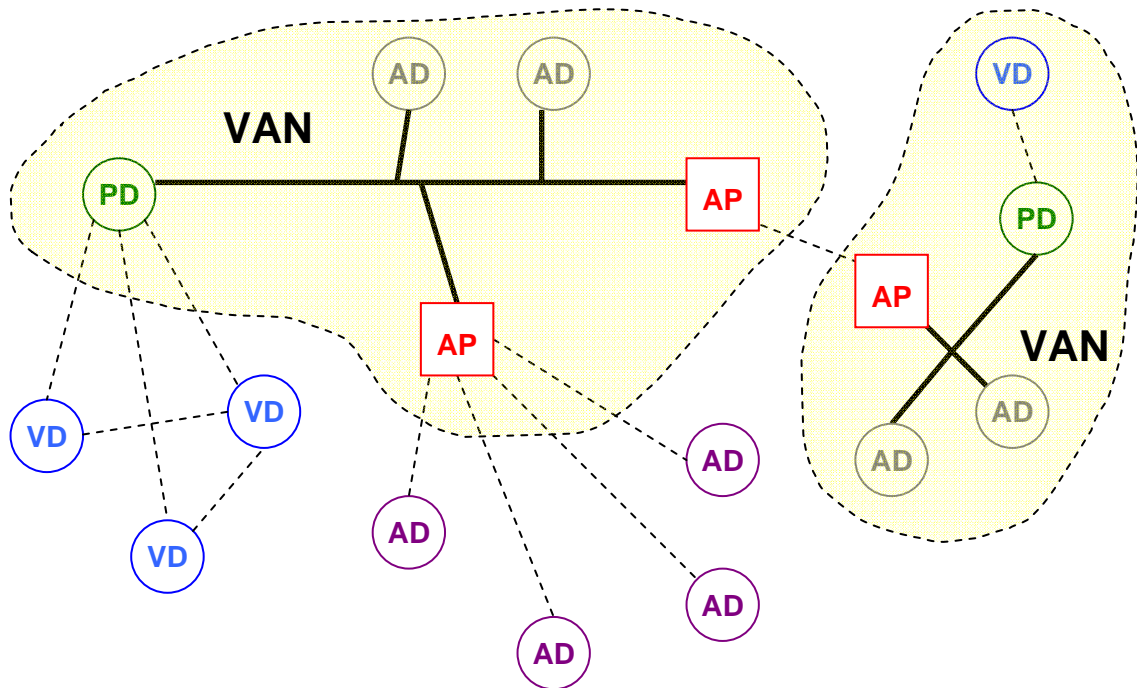


Figure 2-2: Device topology for UWB devices in the VAN domain

2.4 Wireless LAN

2.4.1 Application Fields

The standardisation body of WLAN is IEEE 802.11, the neighbour of 802.3, where the wired Ethernet has its location. Therefore, WLAN is considered to be the “wireless Ethernet”. With the success of Ethernet in automation WLAN can be seen as the perfect extension for wire line automation applications. In general, WLAN offers a high data rate (up to 54 Mbit/s). This makes it applicable where large process files have to be transmitted, but also applications in automation with short data frames but high real time requirement to establish the communication between PLCs and field devices. WLAN helps to fulfil the demand for reliability as it offers redundancy mechanisms and package retries. In addition, WLAN can be operated in the popular ISM band at 2,4 GHz, but has the possibility to use the 5 GHz-band, what is less crowded. Combined with powerful security mechanisms, this radio technology can be used in applications where failsafe data (e.g. emergency stop) need to be transmitted.

Beyond data services, WLAN can be used to integrate voice into the communication stream. Voice-over-IP (VoIP) can be extended into the WLAN world and offers on-site service personal a flexible and direct communication with the control room. Data services are perfectly completed by voice to give maintenance personal a wide overview e.g. of the status of a damaged machine.

Overview of WLAN applications in automation:

- Wireless HMI (human machine interface)
WLAN is common in mobile devices like laptops and PDAs, transmission of large process images, off-the-shelf components
- Wireless Control, M2M (machine-to-machine)
PLCs are more and more connected to Ethernet control networks, WLAN interfaces easily to Ethernet, cost reduced gateway and interfaces
- Wireless failsafe
One communication channel for standard data and failsafe data, simplified interfaces and gateways
- Mono track rail, AGV (automated guided vehicles), train
Contact less communication to avoid wear and maintenance costs of slippery rings, high reliability

- Large plant floors (up to kilometres), temporary installations
Remote production sites can be simply integrated into the plant network, fast ramp up of communication network at quick changing plant layouts
- Voice, VoIP (voice-over-IP)
Integrated communication to substitute walkie-talkies with proprietary radio technologies, seamless roaming between office and shop floor
- Cranes, Wind mills
Simple connection to turning and large unit
- Mobile service and diagnosis
Right information on-site to reduce costs for maintenance and service

2.4.2 Topologies and Device Types

WLAN allows several different topologies. The topology mainly depends on the application and their requirements. Similar to the other short range radio techniques, topologies like point-to-point, point-to-multipoint, backbone and meshes are supported.

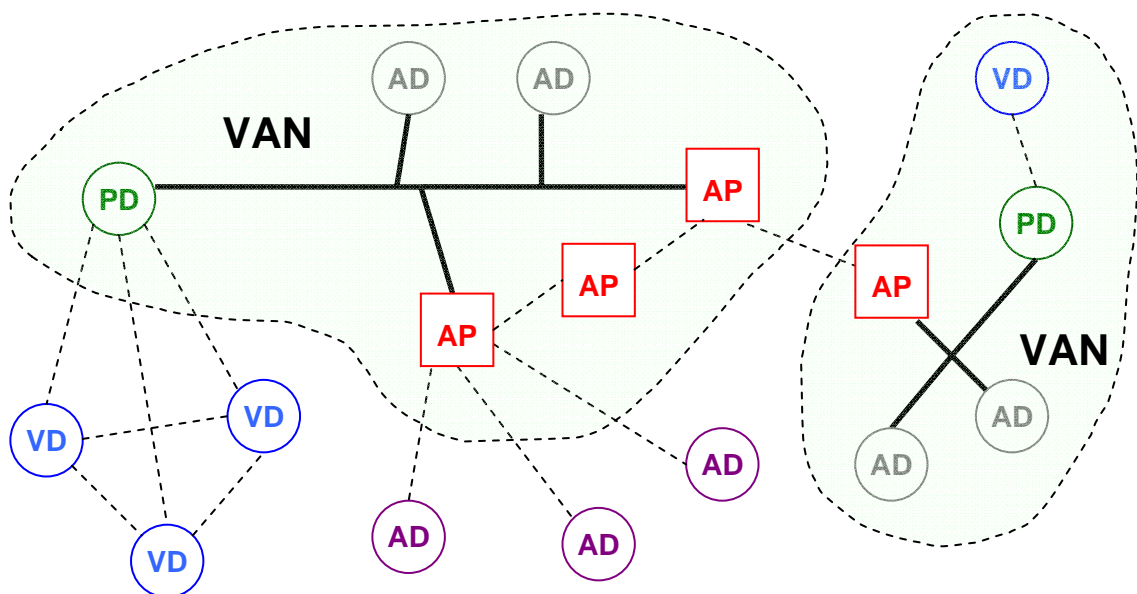


Figure 2-3: Tentative device topology for WLAN devices in the VAN domain

2.5 IEEE 802.15.4 and ZigBee

2.5.1 Application Fields

Wireless sensor network (WSN) technologies such as IEEE 802.15.4-based devices and ZigBee enables the use of multiple, very low-powered nodes to cover wide areas of interest using low data rates. WSNs are typical used to overcome physical and economical constraints for traditional wired sensor solutions and consequently could find applications in some of the following fields of industrial automation:

- Plant and process monitoring via sensor reading (indoors and outdoors)
- Non-critical closed loop applications (e.g. simple switching applications)
- Wireless network extensions for existing wired field devices
- Location awareness applications (e.g. inventory tracking and asset management)
- Plant building automation and management (e.g. lighting, HVAC, security)

From the above list we can see that application fields for IEEE 802.15.4-based and ZigBee devices are not meant to compete with, but instead to complement, existing wireless solutions for industry, such as WLAN and Bluetooth. Due to the relatively low data rates (250 kbps max) of ZigBee / IEEE 802.15.4-based systems, it is not envisaged that WSNs will be used to provide wireless bridging

capabilities for traffic intensive data networks (e.g. multimedia data). Instead, applications areas where IEEE 802.15.4-based and ZigBee devices will be prevalent could be for applications requiring battery-operated devices, which could operate in the order of months or years.

The main characteristics of WSNs compared to the other wireless technologies mentioned in this document are:

- large number of nodes in one network (> 100)
- network layer which supports not only star and tree topologies but also meshed networks
- low transmission rate
- low power and cost end devices

Although it is too early to say which WSN technology will be the dominant one for industrial automation, it is a reasonable assumption that it will be based (or adapted) to some degree on the IEEE 802.15.4 standard. Additionally, several non-standard WSN solutions will exist as well, since standards development is a relatively slow process. The VAN approach takes on a long-term view and will follow the standards development closely in order to accommodate new developments within its framework if necessary. The work presented on IEEE 802.15.4 and ZigBee technology in this deliverable is used to form foundational concepts for WSNs in VAN and will be modified to include any necessary changes in future updates.

2.5.2 Topologies and Device Types

Most of the deployed wireless sensor network items will be used in simple field devices in high quantities. Due to the small device footprint and reduced onboard resources (e.g. memory and computation power) of the field devices for IEEE 802.15.4 and ZigBee applications, the most likely scenario is that a VAN proxy device (VAN-PD) will be used to integrate IEEE 802.15.4 and ZigBee devices into the VAN Domain. The resulting connected devices to the VAN-PD will therefore be VAN virtual devices (VAN-VD).

Three basic topologies (star, cluster-tree and mesh) exist for 802.15.4 (only star and cluster-tree) and ZigBee devices. This is illustrated within the VAN context below:

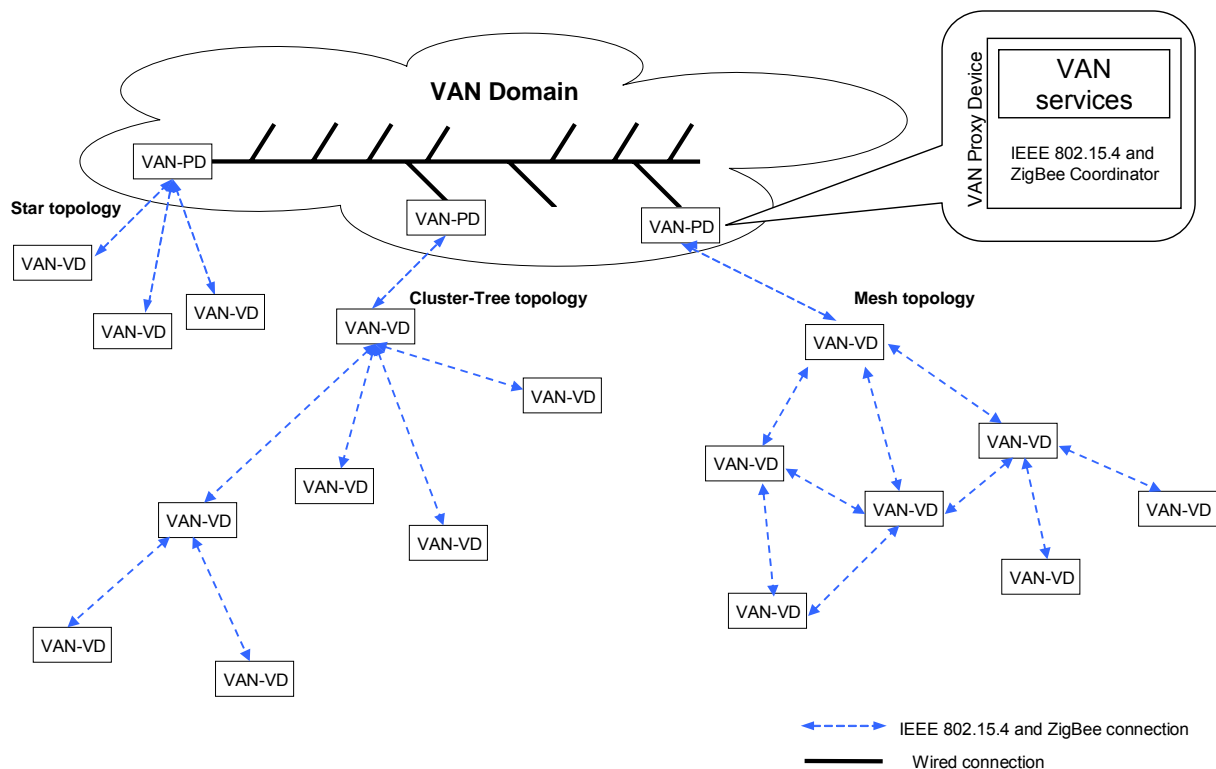


Figure 2-3 VAN topology with IEEE802.11 and ZigBee devices

3 Wireless Device Description Contents

3.1 Preface

In order to automatically support engineering processes in industrial automation formal descriptions of the device's characteristics are necessary. You can distinguish between static information such as allowed temperature range of the device and dynamic parameters which can be configured or which provide status information during operation. However, depending on the implementation of a wireless device some characteristics may be static or dynamic. One example is the antenna. You may have the option to use different antennas depending on the application's requirements nevertheless the antenna could also build e.g. as a PCB antenna.

The characteristics can be device related such as the antenna or system related such as the frequency channel.

This chapter lists and explains the possible static information which you will find today in the data sheets of the wireless devices.

3.2 Applicable Also for Other VAN Technologies

In this section characteristics are introduced which are common for all wireless devices of all technologies or even for all VAN devices.

Device Information

For each wireless device information should be provided such as product name, product type, serial number, hardware version, software version, production date etc.

The information is common to all VAN devices and has to be in line with the definition in deliverable D2.2-2.

Vendor Information

For each wireless device vendor information should be provided such as vendor name, contact data, vendor identification number etc.

The information is common to all VAN devices and has to be in line with the definition in deliverable D2.2-2.

Maximum Payload

This attribute specifies the maximum number of bytes which can be carried by the communication technology.

The information is common to all VAN devices and has to be in line with the definition in deliverable D2.2-2.

Power Supply Related Attributes

This attribute is used to indicate how the device is supplied with power and which power saving modes are implemented.

- Available Power Sources: e.g. mains or autarchic via solar panels
- Available Power Saving Modes: e.g. listen only or inactive

The information could be common to other devices as well (e.g. switches) and should be defined in deliverable D2.2-2.

Available Security Services

Security functionality can be very extensive, so the related attributes are described in an extra ASE. However, in the device description can be listed which security mechanism are available or which security goals are taken into account.

- Available Security Modes: e.g. encryption or authentication mechanism
- Considered Security Objectives: these are availability, confidentiality, integrity, authentication, non-repudiation, auditability

The information is common to all VAN devices and has to be in line with the definition in deliverable D6.3-1.

3.3 Common to All Wireless Technologies

3.3.1 General Description

Radio Technology or Standard

A number of characteristics are already predefined by radio technologies or standards implemented in the wireless devices. So this parameter can give implicitly information e.g. modulation and coding for IEEE802.11b, data rate for Bluetooth or band width for ZigBee.

Topologies Supported

Depending on the technology or the implementation different topologies are supported a system. These are:

- point-to-point,
- star,
- tree, and
- mesh.

Network Device Type

Depending on the technology or the implementation a device can play different rolls within a network. It can be an end device, a router device, an access point or a client. The network device type should not be confused with the VAN Device Type. An end device can be a VAN Automation Device or a VAN Virtual Device.

Quality Of Service

The quality of service of a link can be described with following characteristic parameters:

- Transmission Delay
- Response Time
- Update Time
- Packet Loss Rate

Since, these characteristic parameters have to be measured under defined test conditions for the exact definition is referred to the test specification which will be worked out in relation with deliverable D3.4-1.

3.3.2 Physical Radio Description

The physical layer radio characteristics are necessary to estimate the quality of service of a connection under defined environmental definitions. It can be described by following attributes:

Frequency Bands

Different frequency bands can be used for wireless communication. Examples are the 2,4 GHz band and the 5 GHz band.

Centre Frequencies

This attribute describes which centre frequencies can be used within the frequency band.

Band Width

This attribute describes the band width in the frequency band used during the transmission. It is fixed for a selected technology.

Channel Separation

This attribute describes the distance between two centre frequencies. It is fixed for a selected technology.

Modulation

This attribute describes the used modulation scheme. It is fixed for a selected technology or standard.

Coding

This attribute describes the used coding for a symbol during the radio transmission. It is fixed for a selected technology or standard.

Frequency Channels

For some technologies frequency channels are defined which can be used instead of Centre Frequency and Band Width to address a certain frequency area within a frequency band. This attribute describes the channels which can be used by a device.

Supported Data Rates

This attributes describes the data rate which is used at the air interface.

Maximum Transmission Power (EIRP)

This attribute depends on the transmit power which can be configured in the device, the power loss of the circuitry and the antenna gain.

Receiver Sensitivity

This attribute depends on the technology and on the implementation.

Antenna Type

The antenna plays an important roll for the quality of the wireless communication. Examples for antenna types are:

- external,
- ceramic,
- PCB.

Antenna Characteristic

The antenna characteristic is given by diagrams.

Antenna Gain

Since real antennas are not ideal omni-directional antennas here the directivity is described.

Antenna Architecture

Besides single antennas also divers antennas or antenna areas (MIMO) are possible.

3.3.3 Media Access Description

Information about the media access control should be provided to be able to estimate the real time capabilities of the wireless network. Depending on the wireless technology or standard there MAC functionality can be configured or not. Following Media Access Related Attributes are relevant:

Media Access Mode

Examples for media access modes are time division media access (TDMA) and carrier sense multiple access collision avoidance (CSMA/CA).

Global Time Period

A global time frame is often defined in order to be able to control the media access. This attribute contains a time value which defines the duration of that time frame.

Isochronous Time Period

This attribute contains the time value which defines a contention free period within the global time frame.

Asynchronous Time Period

This attribute contains the time value which defines a contention period within the global time frame. Collisions in that period of time are likely.

Number Retries

This attribute is used to define how often transmissions are retried in case of recognized errors.

3.4 Specific for IEEE 802.15.4 and ZigBee

The ZigBee specification considers the capabilities of a device by different parameters and descriptors. In this section these are explained. For attributes which are already mentioned in section 3.2 the special coding for ZigBee is given here.

3.4.1 Parameters

FrequencyChannels

Unsigned32

The attribute FrequencyChannels is represented in IEEE 802.15.4 by the attribute PhyChannelsSupported. According to the IEEE 802.15.4 standard, all devices should be capable of performing passive and orphan scans across a specific list of channels. A FFD should additionally be able to perform energy detect (ED) and active scans. A list of channels chosen from the channels specified by PhyChannelsSupported will be issued by the next higher layer when submitting a scan request.

The availability status for each of the 27 channels (see table 6-2) is indicated by this attribute and is dependent on the PHY hardware used as well as local regulations. The user may specify to use a preset (reduced) number of channels (in the case of the 2450MHz and 915MHz band) within a particular PHY. However, to ensure maximum interoperability for this device with other IEEE 802.15.4 (ZigBee) networks, it is advisable not to restrict this channel range, particularly for devices other than the IEEE 802.15.4 (ZigBee) coordinator.

The PhyChannelSupported attribute describes the current status of the channels (1 = available, 0 = unavailable). A total of 27 channels are available across 3 frequency bands: sixteen in 2450 MHz band, 10 in the 915 MHz band and 1 in the 868 MHz band as shown in table 6-2.

3.4.2 Node Descriptor

This structure contains information about the capabilities of a ZigBee node. There is only one per node.

Logical Type

Unsigned8

This attribute contains the logical device type of a ZigBee node. This could be either:

Table 3-1: Logical Device Type

Device description	Logical type value (binary)
ZigBee coordinator	000
ZigBee router	001
ZigBee end device	010

This attribute is the ZigBee equivalent of the Network Device Type defined in section 3.3. Concerning the decision made in this work package there are only following relations to VAN device types possible:

- ZigBee coordinator - VAN proxy device
- ZigBee router - VAN virtual device
- ZigBee end device - VAN virtual device

APS Flags

The APS attribute specifies the application support sub-layer capabilities of a node. For the first version of ZigBee this attribute is not supported but later version might include this field.

Frequency Band

BitString8

This attribute specifies the supported frequency bands of the IEEE 802.15.4 radio. A logical 1 indicates that the frequency band is supported.

Table 3-2: Frequency Band Coding

Frequency Band supported	Frequency band bit field number
868 – 868.6 MHz	0
Reserved	1
902 – 928 MHz	2
2400 – 2483.5 MHz	3

This attribute is the ZigBee equivalent of the Physical Radio Description attribute "Frequency Band" defined in section 3.3.

MAC Capabilities Flags field BitString8

This attribute specifies the node capabilities as required by the IEEE 802.15.4. MAC sub-layer.

Table 3-3: Capability Flag Coding

Field name	Bit field number	Description
Alternate PAN Coordinator	0	Set to 1 is node is capable of becoming a PAN coordinator and 0 if not.
Device Type	1	1 – node is a full function device (FFD) 2 – node is a reduce function device (RFD)
Power Source	2	1 – mains powered 0 – other source of power
Receiver On When Idle	3	1 – device does not switch off receiver to conserve power 0 – device switches off receiver to conserve power
Reserved for this version	4-5	
Security Capability	6	1 – device is able to send and receive frames secured using IEEE 802.15.4. security suite specification 0 – device is unable to perform the above.
Reserved	7	

This Device Type is the IEEE 802.15.4 equivalent of the Network Device Type defined in section 3.3. Concerning the decision made in this work package there are only following relations to VAN device types possible:

- Full-Function Device (FFD) - VAN proxy device
- Full-Function Device (FFD) - VAN virtual device
- Reduced-Function Device (FFD) - VAN virtual device

The Power Source field is the IEEE 802.15.4 equivalent of the Power Supply Related Attribute "Available Power Sources" defined in section 3.2.

The Security Capability field is the IEEE 802.15.4 equivalent of the Available Security Services attribute "Available Security Modes" defined in section 3.2.

Manufacturer Code Unsigned16

The Manufacturer Code field of the node descriptor is sixteen bits in length and specifies a manufacturer code that is allocated by the ZigBee Alliance, relating the manufacturer of the device. This attribute is a ZigBee related part of the Vendor Information defined in section 3.2.

Maximum Buffer Size Unsigned8

The Maximum Buffer Size field of the node descriptor is eight bits in length, with a valid range of 0x00-0x7f, and specifies the maximum size, in octets, of the application support sub-layer data unit (ASDU) for this node. This is the maximum size of data or commands passed to or from the application by the application support sub-layer, before any fragmentation or re-assembly (fragmentation is not currently supported). This field can be used as a high level indication for network management.

This attribute is the ZigBee equivalent of the Maximum Payload attribute defined in section 3.2.

Maximum Transfer Size Unsigned16

The Maximum Transfer Size field of the node descriptor is sixteen bits in length, with a valid range of 0x0000-0x7fff, and specifies the maximum size, in octets, that can be transferred to or from this node in one single message transfer. This value can exceed the value of the node maximum buffer size field. However, this field is currently not supported and shall be set to zero.

3.4.3 Node Power Descriptor

The Node Power Descriptor gives a indication of the power status of the node.

Current Power Mode Unsigned8

Not relevant in this section, see section **Fehler! Verweisquelle konnte nicht gefunden werden..**

Available Power Sources field BitString8

The available power sources field of the node power descriptor specifies the power sources available on this node. For each power source supported on this node, the corresponding bit of the available power sources field shall be set to 1. All other bits shall be set to 0.

Table 3-4: Supported Power Sources

Supported Power Source	Bit field number
Constant (mains) power	0
Rechargeable battery	1
Disposable battery	2
Reserved	3

The Supported Power Source field is the ZigBee equivalent of the Power Supply Related Attribute "Available Power Sources" defined in section 3.2.

Current Power Source field BitString8

Not relevant in this section, see section 8.4.

Current Power Source level field Unsigned8

Not relevant in this section, see section 8.4.

3.4.4 Simple Descriptor

The Simple Descriptor contains information specific to each application endpoint contained in a node. An endpoint is an application related entity within a node. The Simple Descriptor is mandatory for each endpoint present in the node.

Presuming that application profiles are available which can be mapped to the ZigBee Simple Descriptor all information is available to discover the device type concerning it's possible roll in the automation process.

The fields of the simple descriptor are described below.

Endpoint Unsigned8

The Endpoint field of the simple descriptor is eight bits in length and specifies the endpoint within the node to which this description refers. Applications shall only use endpoints 1-240.

Application Profile Identifier Unsigned16

The Application Profile Identifier field of the simple descriptor is sixteen bits in length and specifies the profile that is supported on this endpoint. Profile Identifiers shall be obtained from the ZigBee Alliance.

This attribute is a ZigBee related part of the Device Information defined in section 3.2.

Application Device Identifier Unsigned16

The Application Device Identifier field of the simple descriptor is sixteen bits in length and specifies the device description supported on this endpoint. Device description identifiers shall be obtained from the ZigBee Alliance.

This attribute is a ZigBee related part of the Device Information defined in section 3.2.

Application Device Version Unsigned8

The Application Device Version field of the simple descriptor specifies the version of the device description supported on this endpoint. The Application Device Version field shall be set to one of the values listed below:

Table 3-5: Application Device Version

Application device version value $b_3b_2b_1b_0$	Description
0000	Version 1.0
0001–1111	Reserved

This attribute is a ZigBee related part of the Device Information defined in section 3.2.

Application Flags Unsigned8

The Application Flags field of the simple descriptor specifies application specific flags. For each feature supported by the application on this endpoint, the corresponding bit of the application flags field, as listed in the table below, shall be set to 1. All other bits shall be set to 0.

Table 3-6: Application Flag Coding

Application flags field bit number	Supported feature
0	Complex descriptor available
1	User descriptor available
2-3	Reserved

Application Input Cluster Count field Unsigned8

The Application Input Cluster Count field of the simple descriptor specifies the number of input clusters, supported on this endpoint, that will appear in the Application Input Cluster List field. If the value of this field is zero, the application input cluster list field shall not be included.

An input cluster is an application input variable of an endpoint.

Application Input Cluster List Unsigned8 * (application input cluster count)

The Application Input Cluster List of the simple descriptor is $8 \cdot i$ bits in length, where i is the value of the Application Input Cluster Count field, and specifies the list of input clusters supported on this endpoint, used during the binding procedure.

The Application Input Cluster List field shall be included only if the value of the application input cluster count field is greater than zero.

Application Output Cluster Count field Unsigned8

The Application Output Cluster Count field of the simple descriptor is eight bits in length and specifies the number of output clusters, supported on this endpoint, that will appear in the Application Output Cluster List field. If the value of this field is zero, the Application Output Cluster List field shall not be included.

Application Output Cluster List Unsigned8 * (application output cluster count)

The Application Output Cluster List of the simple descriptor is 8*o bits in length, where o is the value of the Application Output Cluster Count field, and specifies the list of output clusters supported on this endpoint, used during the binding procedure. The Application Output Cluster List field shall be included only if the value of the Application Output Cluster Count field is greater than zero.

3.4.5 Complex descriptor

The complex descriptor contains extended information for each of the device descriptions contained in this node. The use of the complex descriptor is optional.

Language And Character Set field Unsigned32

The Language And Character Set field is three octets in length and specifies the language and character set used by the character strings in the complex descriptor.

Manufacturer Name field Character string

The Manufacturer Name field has a variable length and contains a character string representing the name of the manufacturer of the device.

This attribute is a ZigBee related part of the Vendor Information defined in section 3.2.

Model Name field Character string

The Model Name field has a variable length and contains a character string representing the name of the manufacturers model of the device.

This attribute is a ZigBee related part of the Device Information defined in section 3.2.

Serial Number field Character string

The Serial Number field has a variable length and contains a character string representing the manufacturers serial number of the device.

This attribute is a ZigBee related part of the Device Information defined in section 3.2.

Device URL field Character string

The Device URL field has a variable length and contains a character string representing the URL through which more information relating to the device can be obtained.

This attribute is a ZigBee related part of the Device Information defined in section 3.2.

Icon field Not defined

The Icon field has a variable length and contains the data for an icon that can represent the device on a computer, gateway or PDA. The format of the icon data is not specified in this document.

This attribute is a ZigBee related part of the Device Information defined in section 3.2.

Icon URL field Character string

The Icon URL field has a variable length and contains a character string representing the URL through which the icon for the device can be obtained.

This attribute is a ZigBee related part of the Device Information defined in section 3.2.

4 Wireless Contributions to VAN ASEs

4.1 Preface

There are a number of attributes which are common for all wireless technologies. These attributes are described here taking into account that some of them should also be relevant for other components of VAN. For more flexible implementations of wireless devices an overlapping is possible to the already describe static parameters. In these cases a reference is made to the related attributes. The description of attributes follows the object oriented open platform and system architecture defined in [D02.2-1].

4.2 Applicable Also for Other VAN Technologies

The definitions of this chapter is also applicable for other communication technologies used in VAN. Therefore, it is expected that these attributes are specified in WP2. In some cases wireless related additions are made.

Network Type

This work package adds four new transmission technologies to the VAN concept which are based on Bluetooth, Ultra-Wideband, Wireless LAN and IEEE 802.15.4/ZigBee. Therefore, the following new network types have to be considered by the overall device and system architecture in WP2:

- Bluetooth
- UWB
- WLAN Ad-hoc
- WLAN Infrastructure
- IEEE 802.15.4
- ZigBee

In addition codes for special UWB implementations and IEEE 802.15.4 versions should be reserved.

Address Information

The identified wireless VAN device types have to be addressed in accordance to the definition of WP2. If necessary address translations will be defined in Task 3.4.

Supported Internet Protocols

Since wireless devices may implement different kind of IP services a related attribute is necessary. This should be in line with definitions for other VAN devices made in WP2.

4.3 Wireless Device Configuration Class

4.3.1 Object Overview

The figure below shows the classes contributed within this deliverable to the Device Config ASE. There is the abstract class WIRELESS DEVICE CONFIG that defines attributes that are contained in all further derived classes. For each considered wireless technology a single specific class is defined that specifies the specific structure and attributes needed to use the respective technology within VAN. These single classes are: BLUETOOTH DEVICE CONFIG, UWB DEVICE CONFIG, WLAN DEVICE CONFIG and ZIGBEE DEVICE CONFIG.

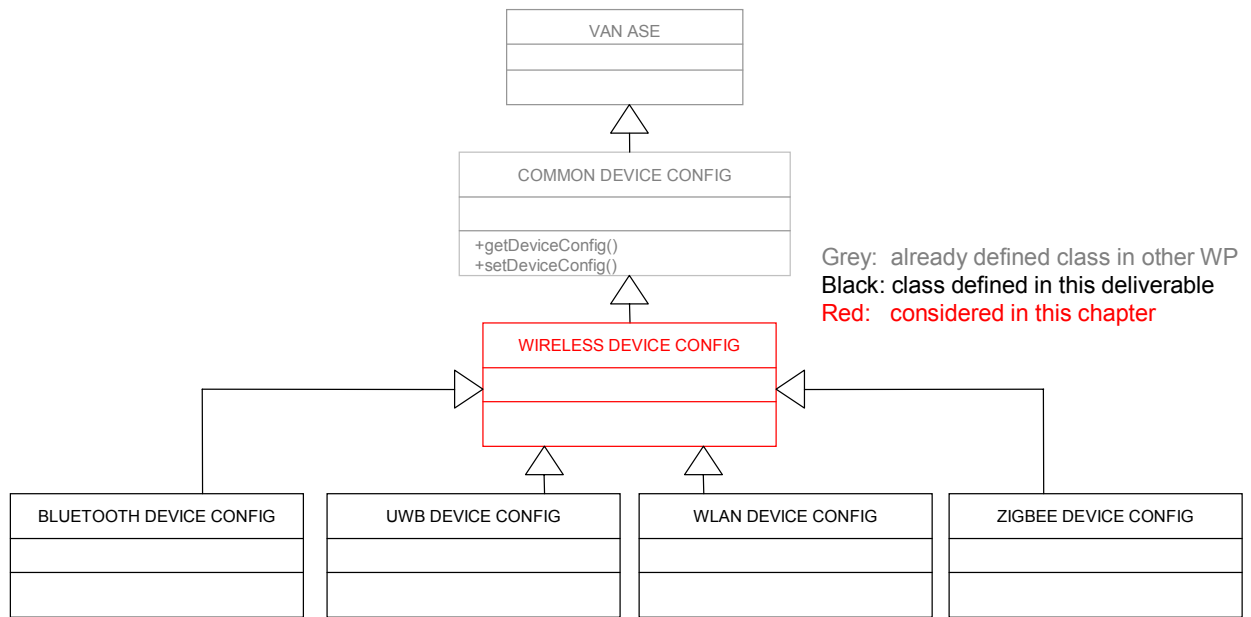


Figure 4-1: Classes contributed by Task3.2 to the Device Config ASE

4.3.2 Formal model

VAN ASE: DEVICE CONFIG ASE

CLASS: WIRELESS DEVICE CONFIG

CLASS ID:

PARENT CLASS: COMMON DEVICE CONFIG

ATTRIBUTES:

- 1 (o) Attribute: PromiscuousMode
- 2 (m) Attribute: CurrentPowerSource
- 3 (m) Attribute: PhysicalRadio
 - 3.1 (o) Attribute: CurrentFrequencyBand
 - 3.2 (o) Attribute: CurrentCentreFrequency
 - 3.3 (o) Attribute: CurrentFrequencyChannel
 - 3.4 (m) Attribute: TransmitPower
- 4 (o) Attribute: MediaAccess
 - 4.1 (o) Attribute: GlobalTimePeriod
 - 4.2 (o) Attribute: IsochronousTimePeriod
 - 4.3 (o) Attribute: AsynchronousTimePeriod
 - 4.4 (o) Attribute: NumberOfRetries

4.3.3 Attribute description

4.3.3.1 General Attributes

PromiscuousMode Boolean

This attribute indicates whether the node is in a receive all (promiscuous) mode. A value of TRUE indicates that the node accepts all frames received from the PHY. This could be viewed as a type of packet-sniffing feature. Additional attributes such as a specific frequency channel may be defined in the technology specific section. Otherwise the current configuration is valid.

CurrentPowerSource BitString8

This attribute specifies the current power source being utilized by the node. Following values are available:

- Constant (mains) power
- Rechargeable battery
- Disposable battery
- Reserved

4.3.3.2 Physical Radio Related Attributes

CurrentFrequencyBand

Different frequency bands can be used for wireless communication. Examples are the 2,4 GHz band and the 5 GHz band. Depending on the implementation this attribute can be configured.

CurrentCentreFrequency

This attribute can be used to describe a certain channel in the frequency band for UWB, WLAN or IEEE 802.15.4/ZigBee. Depending on the implementation this attribute can be configured.

CurrentFrequencyChannel

For some technologies frequency channels are defined which can be used instead of Centre Frequency and Band Width to address a certain frequency area within a frequency band. Depending on the implementation this attribute can be configured.

TransmitPower

This attribute is used to set the radio frequency (RF) transmit power output level on a device. To get the equivalent isotropic radiated power (EIRP) the power loss in the circuitry as well as the antenna gain has to be taken into account. This attribute may be fixed for a device and is then part of the wireless device description.

4.3.3.3 Media Access Related Attributes

GlobalTimePeriod

A global time frame is often defined in order to be able to control the media access. This attribute contains a time value which defines the duration of that time frame.

IsochronousTimePeriod

This attribute contains the time value which defines a contention free period within the global time frame.

AsynchronousTimePeriod

This attribute contains the time value which defines a contention period within the global time frame. Collisions in that period of time are likely.

NumberOfRetries

This attribute is used to define how often transmissions are retried in case of recognized errors.

4.4 Wireless Security Configuration Class

4.4.1 Object Overview

The figure below shows the classes contributed within this deliverable to the Security Config ASE. There is the abstract class WIRELESS SECURITY CONFIG that defines attributes that are contained in all further derived classes. For each considered wireless technology a single specific class is defined that specifies the specific structure and attributes needed to use the respective technology within VAN. These single classes are: BLUETOOTH SECURITY CONFIG, UWB SECURITY CONFIG, WLAN SECURITY CONFIG and ZIGBEE SECURITY CONFIG.

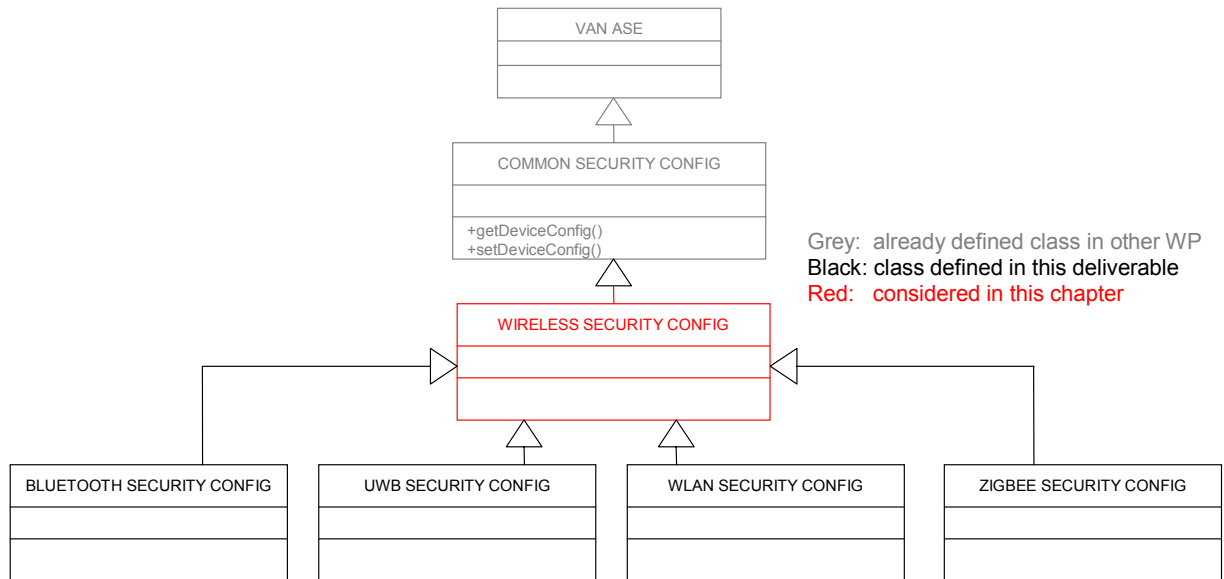


Figure 4-2: Classes contributed by Task3.2 to the Security Config ASE

4.4.2 Formal model

VAN ASE: SECURITY CONFIG ASE

CLASS: WIRELESS SECURITY CONFIG

CLASS ID:

PARENT CLASS: COMMON SECURITY CONFIG

ATTRIBUTES:

- 1 (m) Attribute: SecurityActivated
- 2 (m) Attribute: CurrentSecurityMode

4.4.3 Attribute description

SecurityActivated Boolean

This attribute indicates whether a security services is active or not.

CurrentSecurityMode

This attribute specifies which security services are activated.

4.5 Wireless Diagnosis Class

4.5.1 Object Overview

The figure below shows the classes contributed within this deliverable to the DIAGNOSIS ASE. There is the abstract class WIRELESS DIAGNOSIS that defines attributes that are contained in all further derived classes. For each considered wireless technology a single specific class is defined that specifies the specific structure and attributes needed to use the respective technology within VAN. These single classes are: BLUETOOTH DIAGNOSIS, UWB DIAGNOSIS, WLAN DIAGNOSIS and ZIGBEE DIAGNOSIS.

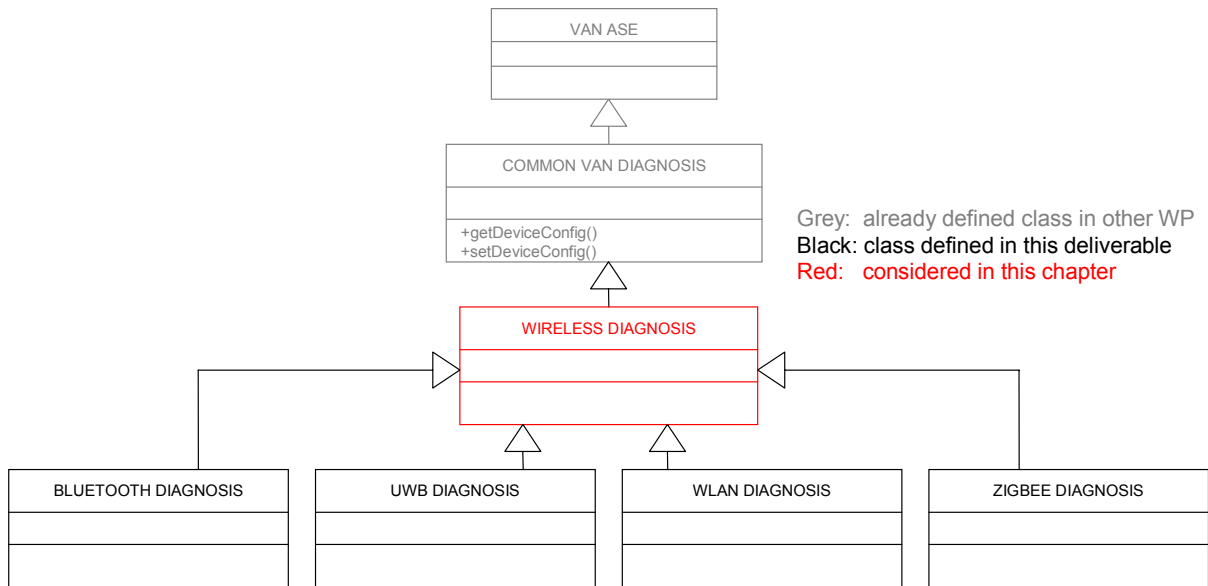


Figure 4-3: Classes contributed by Task3.2 to the Diagnosis ASE

4.5.2 Formal model

VAN ASE: DIAGNOSIS ASE

CLASS: WIRELESS DIAGNOSIS

CLASS ID:

PARENT CLASS: COMMON VAN DIAGNOSIS

ATTRIBUTES:

- 1 (m) Attribute: SupplyPowerLevel
- 2 (o) Attribute: StatusInformation
- 3 (m) Attribute: ErrorIndication
- 4 (m) Attribute: ReceiverSignalStrength

4.5.3 Attribute description

SupplyPowerLevel

This attribute indicates the still available power for battery powered devices.

StatusInformation

This attribute provides information about the operational status of the device. Possible values are:

- Initialising
- Inactive (configuration possible)
- Operational
- Error

ErrorIndication

This attribute provides information about the current error status.

ReceiverSignalStrength

This attribute indicates the quality of the radio link for the given device.

5 Bluetooth Contributions to VAN ASEs

5.1 Relevant Architecture Elements

This chapter describes elements of the VAN device architecture which are specific for the Bluetooth wireless technology for instance the VAN Proxy application process. It is mentioned which part of the wireless VAN device architecture deals with which special requirement of industrial application environments.

5.2 Bluetooth Configuration Class

5.2.1 Object Overview

The BLUETOOTH DEVICE CONFIG class specifies the attributes that are used to configure the Bluetooth devices via its Management Information Base. Furthermore in case of a VAN-Bluetooth-Proxy, the proxy contains one BLUETOOTH DEVICE CONFIG instance for each Bluetooth node in the network.

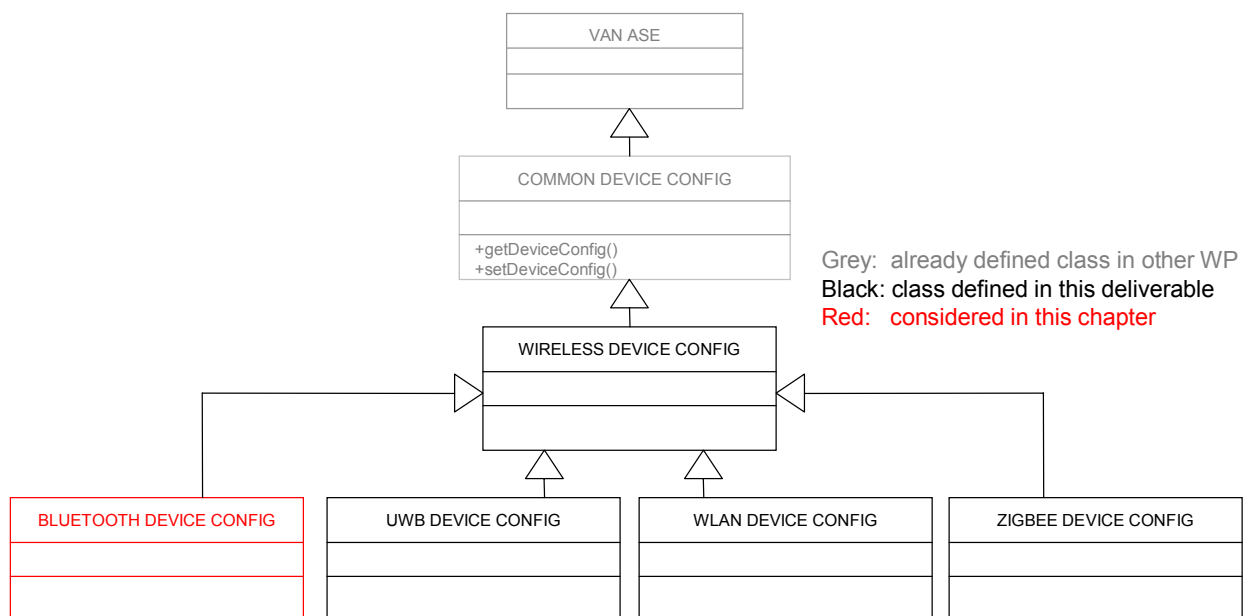


Figure 5-1: Deduction of the BLUETOOTH DEVICE CONFIG class structure

5.2.2 Refinement of Inherited Attributes

object-reference

String

This attribute is inherited from the VAN ASE. The content of the string for this class is "Bluetooth".

5.2.3 Formal Model

VAN ASE: DEVICE CONFIG ASE

CLASS: BLUETOOTH DEVICE CONFIG

CLASS ID:

PARENT CLASS: WIRELESS DEVICE CONFIG

ATTRIBUTES:

- 1 (m) Attribute: TransmitPower
- 2 (m) Attribute: PhyChannelsUsed
3. (m) Attribute: Inquiry_Scan_Interval
4. (o) Attribute: Inquiry_Scan_Window
5. (o) Attribute: Inquiry_Mode

- 6. (o) Attribute: Page_Timeout
- 7. (m) Attribute: Page_Scan_Interval
- 8. (o) Attribute: Page_Scan_Window
- 9. (m) Attribute: Link_Policy_Settings
- 10. (o) Attribute: Flush_Timeout
- 11. (o) Attribute: Num_Broadcast_Retransmissions
- 12. (m) Attribute: Link_Supervision_Timeout
- 13. (m) Attribute: Class_of_Device
- 14. (m) Attribute: BTDeviceName
- 15. (m) Attribute: BTRole (MASTER, SLAVE, AUTO)
- 16. (m) Attribute: List-of-BTRemotePeers
- 16.1(m) Attribute: BTRemotePeer

5.2.4 Attribute Description

5.2.4.1 Attributes related to the Physical Layer

Transmit Power integer8

This attribute is used to set the RF transmit power output level on a Bluetooth / 802.15.1 device. In order to keep the value for the transmit power independent of the used chipset, this attribute uses the power level in dBm as integer8 value.

PhyChannelsUsed Bitstring [80]

According to IEEE802.15.1 Bluetooth uses 79 different channels in the 2,4 GHz ISM Band. Each channel has a width of 1 MHz. Devices using AFH (adaptive frequency hopping) are allowed to use less than 79 channels. Due to regulatory limitations the minimum number of channels used by a frequency hopping system is 20. Measurements have shown, that the AFH algorithm is not always good enough to avoid all interferences with wireless systems using static frequencies like WLAN (IEEE 802.11) or ZigBee (IEEE 802.15.4). Therefore it is useful in terms of a good frequency planning to be able to exclude certain static channels from the hopping sequence by manual configuration.

This attributes indicates which channels are used for the Bluetooth communication. Each of the possible 79 Bluetooth channels is represented by a single Bit. The definition here is the same as for the so called AFH_Channel_Map according to IEEE 802.15.1.

This attribute contains 79 one-bit fields. The n^{th} (numbering from 0) such field (in the range 0 to 78) contains the value for channel n . Bit 79 is reserved. That means it is set to 0 when the attribute is read and is ignored when a value is written to it. The 1-bit field is interpreted as follows:

Table 5-1: 1-bit field of the attribute PhyChannelsUsed

Bit	Meaning
0	Channel n is unused
1	Channel n is used

5.2.4.2 Attributes related to the MAC Layer

Inquiry_Scan_Interval unsigned16

There are two different phases involved in the connection establishment of Bluetooth devices. The first one is the inquiry phase. During this phase a device doing an inquiry can find out which other devices in its own vicinity are available and which services are offered by these devices. As Bluetooth devices do frequency hopping in their normal communication they have to listen in certain intervals on certain inquiry scan channels in order to be able to receive inquiry messages from other Bluetooth devices.

The Inquiry_Scan_Interval configuration parameter defines the amount of time between consecutive inquiry scans. This is defined as the time interval from when the Bluetooth device started its last inquiry scan until it begins the next inquiry scan.

Values can range from 0x0012 to 0x1000 and only even values are valid. As the value gives the interval in terms of Bluetooth time slots the value has to be multiplied by 0.625 ms to have the scan interval in ms. The default value is 2.56 sec.

Inquiry_Scan_Window unsigned16

The Inquiry_Scan_Window configuration parameter defines the amount of time for the duration of the inquiry scan. The Inquiry_Scan_Window can only be less than or equal to the Inquiry_Scan_Interval.

Possible values range from 0x0011 to 0x1000. The value is given in terms of Bluetooth time slots. To get the window size in ms, the value of this attribute has to be multiplied by 0.625 ms.

Inquiry_Mode unsigned8

This parameter defines whether inquiry returns Inquiry Result events in the standard format or with RSSI (received signal strength indication) values.

Table 5-2: Inquiry Mode parameter description

Value	Parameter Description
0x00	Standard Inquiry Result format
0x01	Inquiry Result format with RSSI
0x02-0xff	Reserved

Page_Timeout unsigned16

The real establishment of a Bluetooth physical link is done during the page process. Again a Bluetooth device which accepts new links has to listen in certain page scan channels whereas a device trying to establish a link has to transmit on these channels.

The Page_Timeout configuration parameter defines the maximum time the local Link Manager will wait for a baseband page response from the remote device at a locally initiated connection attempt. If this time expires and the remote device has not responded to the page at baseband level, the connection attempt will be considered to have failed.

Possible values range from 0x001 to 0xffff. The value is given in number of Bluetooth timeslots.

Page_Scan_Interval unsigned16

The Page_Scan_Interval configuration parameter defines the amount of time between consecutive page scans. This time interval is defined from when the device started its last page scan until it begins the next page scan.

Possible values range from 0x0012 to 0x1000. The value is given in number of Bluetooth timeslots.

Page_Scan_Window unsigned16

The Page_Scan_Window configuration parameter defines the amount of time for the duration of the page scan. The Page_Scan_Window can only be less than or equal to the Page_Scan_Interval.

Possible values range from 0x0011 to 0x1000 and are given in number of Bluetooth timeslots.

Link_Policy_Settings unsigned16

The Link_Policy_Settings parameter determines the behaviour of the local Link Manager when it receives a request from a remote device or it determines itself to change the master-slave role or to enter park state, hold, or sniff mode. The local Link Manager will automatically accept or reject such a request from the remote device, and may even autonomously request itself, depending on the value of the Link_Policy_Settings parameter. When the value of the link_Policy_Settings parameter is changed for a certain connection, the new value will only be used for requests from a remote device or from the local Link Manager itself made after this command has been completed. By enabling each mode individually, the Host can choose any combination needed to support various modes of operation. Multiple Link Manager policies may be specified for the Link_Policy_Settings parameter by performing a bitwise OR operation of the different activity types.

Table 5-3: Link_Policy_Settings

Value	Parameter Description
0x0000	Disable all LM modes
0x0001	Enable role switch
0x0002	Enable hold mode
0x0004	Enable sniff mode
0x0008	Enable park state
0x0010 – 0x8000	Reserved for future use

Flush_Timeout unsigned16

The Flush_Timeout configuration parameter is used for ACL connections only. With the help of the Flush_Timeout retransmissions on baseband level can be limited. This parameter allows packets to be automatically flushed without the host device issuing a specific command. This is important especially for isochronous data, such as audio or industrial realtime communication. When the a packet that is currently being transmitted is automatically “flushed”, the Failed_Contact_Counter is incremented by one.

The values range from 0x001 to 0x07ff and are given in units of Bluetooth timeslots. The timeout value in ms can be calculated by multiplying the Flush_Timeout value by 0.625 ms.

Num_Broadcast_Retransmissions unsigned8

Broadcast packets are not acknowledged and are unreliable. The number of broadcast retransmissions parameter, N, is used to increase the reliability of a broadcast message by retransmitting the broadcast message multiple times. This parameter defines the number of times the device will retransmit a broadcast data packet. This sets the value N_{BC} (number of transmissions per broadcast) in the baseband to one greater than the Num_Broadcast_Retransmissions value. This parameter should be adjusted as the link quality measurement changes.

Value range: 0x00 – 0xfe

Link_Supervision_Timeout unsigned16

The Link_Supervision_timeout parameter is used by the master or slave Bluetooth device to monitor link loss. If, for any reason, no Baseband packets are received from a link for a duration longer than the Link_Supervision_Timeout, the connection is disconnected.

Note: Setting the Link_Supervision_Timeout to No Link_Supervision_Timeout (0x0000) will disable the Link_Supervision_Timeout check for the specified link.

Value range: 0x0000 (no link supervision), 0x0001-0xffff

Default value: 0x7d00

The value is given in units of Bluetooth timeslots. The default value equals therefore 20 sec.

Class_of_Device

Class_of_Device is a parameter received during the device discovery procedure, indicating the type of device and which types of service that are supported.

The information within the Class_of_Device parameter should be referred to as “Bluetooth Device Class” and “Bluetooth Service Type”. The terms for the defined Bluetooth Device Types and Bluetooth Service Types are defined in https://www.bluetooth.org/foundry/assignnumb/document/assigned_numbers.

5.2.4.3 Attributes related to the Configuration of Connections

BTDeviceName OctetString

Bluetooth connections are established with the help of the so called Bluetooth Address (BD_ADDR). This address is comparable to the 48bit MAC address known from Ethernet systems.

In order to make connections more user-friendly, Bluetooth specifies a user-friendly name for connectable devices. This name can be used for the establishment of a connection instead of the BD_ADDR.

List-of-BTRemotePeers

This attribute list contains all remote peers to which the device tries to establish a connection

BTRemotePeer OctetString

This attribute contains the BTDeviceName or the BD_ADDR of a remote peer to which the device tries to establish a connection.

BTRole

The BTRole is used to configure the device whether it should be the MASTER of the Bluetooth network or whether it should be SLAVE in a piconet of another master. A MASTER can have up to 7 active links to SLAVES. On the other hand a SLAVE can only have one link to one MASTER.

Due to that fact, Bluetooth devices with VAN-PD functionality can only be operated as MASTER. Devices with VAN-AD functionality can only be operated as SLAVE. Bluetooth devices working as VAN-APs can be configured to have the MASTER of the SLAVE role. Using the nomenclature of WLAN, the VAN-AP configured as MASTER has the access point functionality whereas the VAN-AP configured as SLAVE takes the part of the station (STA).

It is also possible to configure this parameter to AUTO. That means, that the device accepts a role switch initiated by another connected device.

5.3 Bluetooth Security Configuration Class

5.3.1 Object Overview

This chapter describes the security measures and parameters, which are specified by Bluetooth. The aim is to provide the information to work package 6 to be considered in the overall security concept and to work package 8.

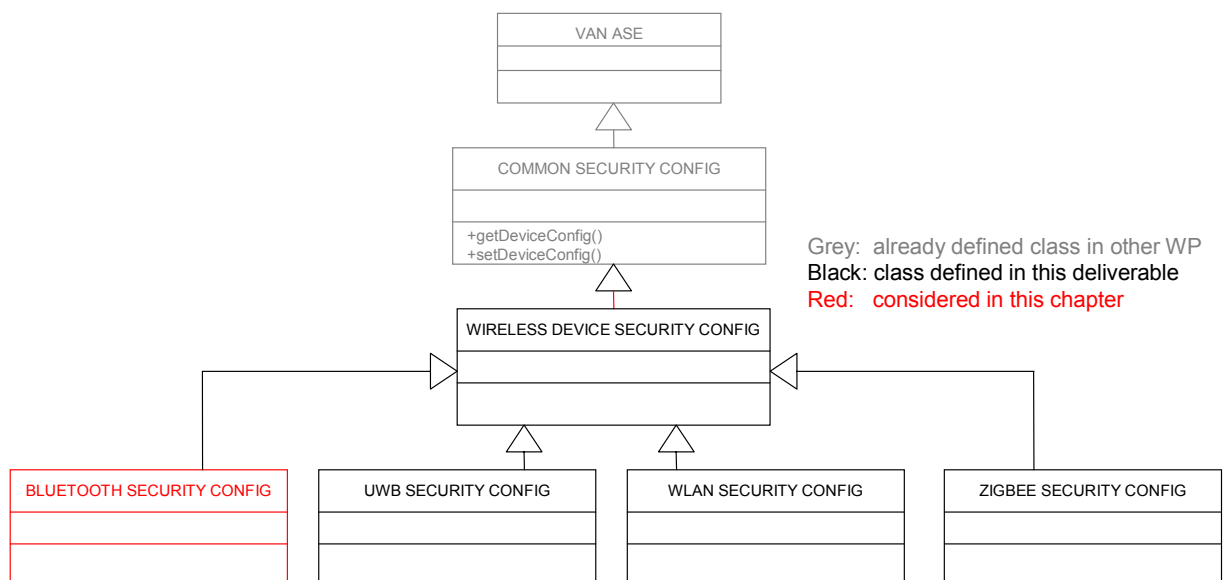


Figure 5-2: Deduction of the BLUETOOTH SECURITY CONFIG class structure

5.3.2 Refinement of Inherited Attributes

object-reference String

This attribute is inherited from the VAN ASE. The content of the string for this class is "Bluetooth".

5.3.3 Formal Model

VAN ASE: SECURITY CONFIG ASE

CLASS: BLUETOOTH SECURITY CONFIG

CLASS ID:

PARENT CLASS: WIRELESS DEVICE SECURITY CONFIG

ATTRIBUTES:

- 1 (m) Attribute: BTSecurityMode
- 2 (m) Attribute: BTPIN
- 3 (o) Attribute: BTconnectable
- 4 (o) Attribute: BTdiscoverable

5.3.4 Attribute Description

BTSecurityMode unsigned8

With this attribute it can be configured which security mode is used by the Bluetooth device. There are three different security modes specified in IEEE 802.15.1:

Table 5-4: Bluetooth security modes

Security Mode	Description
1	Nonsecure: no authentication and no encryption is used
2	Service-level enforced security: A device in security mode 2 does not initiate any security procedure before a channel establishment request has been received or a channel establishment procedure has been initiated by itself. Whether a security procedure is initiated depends on the security requirements of the requested channel or service.
3	Link level enforced security: When a device is in security mode 3, it shall initiate security procedures before the link setup is completed. A device in security mode 3 may reject connections from other devices not fulfilling the own security requirements.

Values other than 1, 2 or 3 are not valid.

BTPIN OctetString[16]

According to IEEE 802.15.1 authentication and encryption is established with the help of a PIN (personal identification number) shared by all devices which have to connect to each other. A PIN is necessary for the establishment of a connection, when the devices use security mode 2 or 3. The maximum length of the PIN is 16 characters.

BTconnectable boolean

This attribute describes whether the Bluetooth device is connectable. A connectable device listens periodically on its page scan physical channel and will respond to a page (i.e. connection establishment) on that channel. This implies that a device which is not connectable cannot be connected by any means from another Bluetooth device.

BTdiscoverable boolean

This attribute provides the possibility to configure a Bluetooth device to be undiscoverable. A device being undiscoverable does not listen to its so called inquiry scan physical channel and will therefore not respond to any inquiries on that channel. Such a device cannot be found by normal scanning for available Bluetooth devices.

On the other hand, a device being discoverable is normally also connectable, because it shows its presence and available services to other devices.

5.4 Bluetooth Diagnosis Class

5.4.1 Object Overview

This section describes the information base attributes of Bluetooth that can be used for monitoring and diagnostic purposes. These are parameters whose values can change during runtime and access to it could provide useful data for diagnosis.

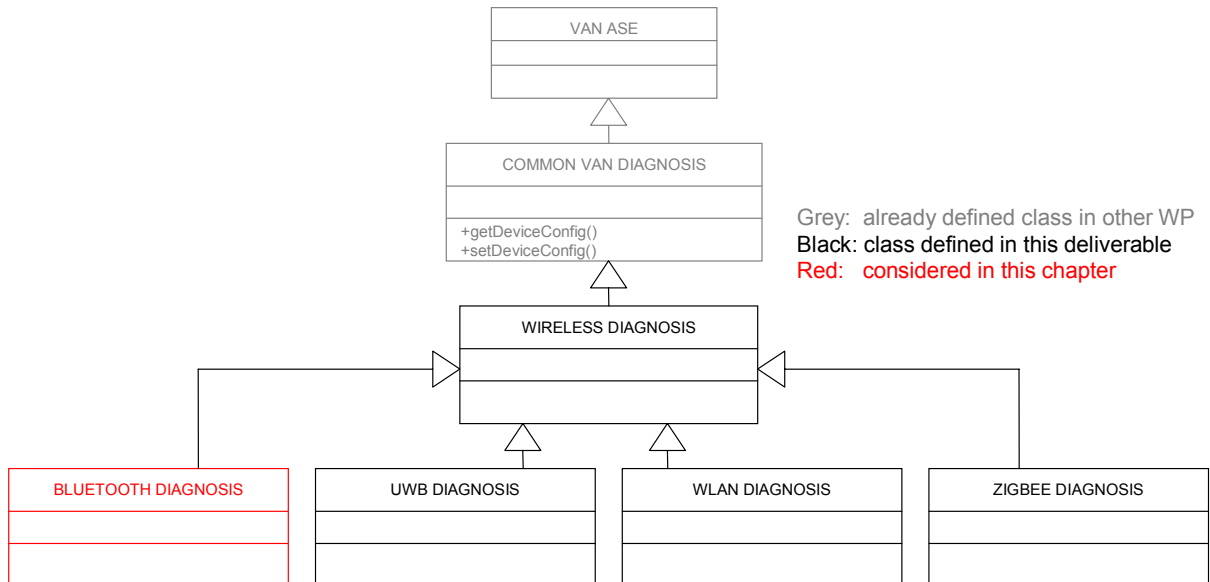


Figure 5-3: Deduction of the BLUETOOTH DIAGNOSIS CONFIG class structure

5.4.2 Refinement of Inherited Attributes

object-reference

String

This attribute is inherited from the VAN ASE. The content of the string in this class is "Bluetooth".

5.4.3 Formal Model

VAN ASE: DIAGNOSIS ASE

CLASS: BLUETOOTH DIAGNOSIS

CLASS ID:

PARENT CLASS: WIRELESS DIAGNOSIS

ATTRIBUTES:

- 1 (m) Attribute: List of BTConnectedPeers
- 1.1 (m) Attribute: ConnectedBD_ADDR
- 1.2 (m) Attribute: ConnectedBTDeviceName
- 1.3 (m) Attribute: LinkQuality
- 1.4 (o) Attribute: RSSI
- 1.5 (o) Attribute: Failed_Contact_Counter

5.4.4 Attribute Description

List of BTConnectedPeers

This attribute list contains all remote peers to which the device has an active connection / link.

ConnectedBD_ADDR

OctetString[6]

This attribute contains the BD_ADDR (Bluetooth Address / MAC Address) of a connected remote peer.

ConnectedBTDeviceName OctetString

This attribute contains the Device Name of a connected remote peer

LinkQuality unsigned8

This attribute contains the Link Quality of a Bluetooth link to a connected remote peer. The Link Quality is a measure for the bit error rate (BER) of the connection.

Values are given in % and range from 0 % to 100 % (no bit errors).

RSSI integer

The RSSI (received signal strength indication) value provides the power with which a connected remote peer is received.

Values are given in dBm.

Failed_Contact_Counter unsigned16

The Failed_Contact_counter records the number of consecutive incidents in which either the slave or master didn't respond after the flush timeout had expired, and the L2CAP packet that was currently being transmitted was automatically "flushed". When this occurs, the Failed_Contact_Counter is incremented by 1. The Failed_Contact_Counter for a connection is reset to zero on the following conditions:

1. When a new connection is established
2. When the counter is reset manually.

6 Ultra-Wideband Contributions to VAN ASEs

Currently, the features of ultra wideband technology depicted here are based on the ECMA-368 standard for PHY and MAC layers of high-rate UWB [ECMA368], though the other relevant specifications might need to be considered depending on the VAN-specific applications of the technology that would be chosen ultimately. The IEEE 802.15.4a standard that uses an UWB-based physical layer specification for short-range wireless networks with a precision ranging capability is yet to be ratified.

6.1 Relevant Architecture Elements

Currently, the features of ultra wideband technology depicted here are based on the ECMA-368 standard for PHY and MAC layers of high-rate UWB [ECMA368], though the other relevant specifications might need to be looked into depending on the VAN-specific applications of the technology that would be ultimately chosen. The IEEE 802.15.4a standard that uses an UWB-based physical layer specification for short-range wireless networks with a precision ranging capability is one such candidate but is yet to be ratified.

6.2 UWB Device Configuration Class

6.2.1 Object Overview

The UWB DEVICE CONFIG class specifies the attributes that are used to configure the UWB devices via its Management Information Base.

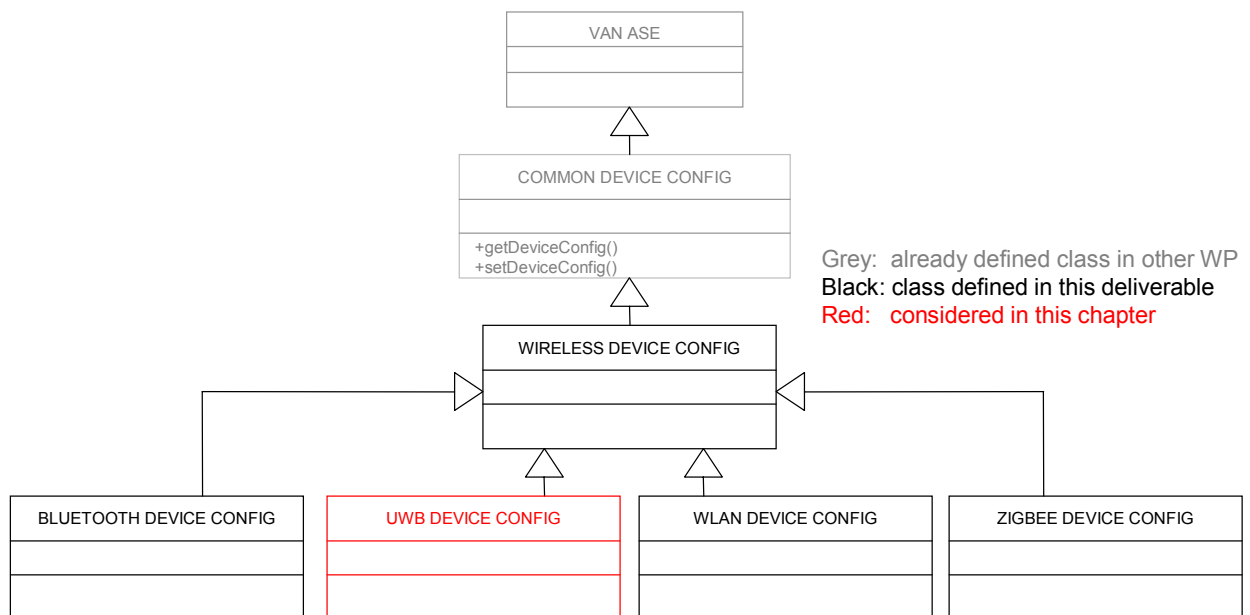


Figure 6-1: Deduction of the UWB DEVICE CONFIG class structure

6.2.2 Refinement of Inherited Attributes

object-reference

String

This attribute is inherited from the VAN ASE. The content of the string in this class is "UWB".

6.2.3 Formal Model

VAN ASE: DEVICE CONFIG ASE

CLASS: UWB DEVICE CONFIG

CLASS ID:

PARENT CLASS: COMMON DEVICE CONFIG

ATTRIBUTES:

- 1 (m) Attribute: FrequencyRange
- 2 (m) Attribute: BandGrpAllocation
- 3 (m) Attribute: TxPowerControl
- 4 (m) Attribute: TxConstellationError
- 5 (m) Attribute: RxSensitivity
- 6 (m) Attribute: LinkQualityIndicator

6.2.4 Attribute Description

FrequencyRange Unsigned8

High-rate (HR) UWB PHY operates in the 3.1-10.6 GHz frequency band.

BandGrpAllocation BitString [8]

Table 6-1: Allocation of HR UWB band groups

Band Group	Band_ID	Lower Freq (MHz)	Center Freq (MHz)	Upper Freq (MHz)
1	1	3168	3432	3696
	2	3696	3960	4224
	3	4224	4488	4752
2	4	4752	5016	5280
	5	5280	5544	5808
	6	5808	6072	6336
3	7	6336	6600	6864
	8	6864	7128	7392
	9	7392	7656	7920
4	10	7920	8184	8448
	11	8448	8712	8976
	12	8976	9240	9504
5	13	9504	9768	10032
	14	10032	10296	10560

TxPowerControl BitString [8]

When the device is using time-frequency interleaving (TFI), the monotonic dynamic range for the attenuation of the transmit power is 0-12 dB, with a step size granularity of 2 dB. On the other hand, when the device is using fixed-frequency interleaving (FFI), the monotonic dynamic range for the attenuation of the transmit power is 0-8 dB, with a step size granularity of 2 dB.

Table 6-2: HR UWB transmit power levels and attenuation

TXPWR_LEVEL	TX Power Attenuation for TFI Modes in dB	TX Power Attenuation for FFI Modes in dB
0	0	0
1	2	2
2	4	4
3	6	6
4	8	8
5	10	Reserved
6	12	Reserved
7	Reserved	Reserved

TxConstellationError

Unsigned8

The relative constellation error values are a function of the transmit power attenuation. The relative constellation RMS error, averaged over all data and pilot subcarriers of the OFDM symbols and over all of the frames, should not exceed the values given in Table 6-3.

Table 6-3: Permissible transmitter constellation errors in UWB

Data Rate	Relative Constellation RMS Error		
	No TX Attenuation	TX Attenuation of 2, 4, 6 dB (both TFI and FFI)	TX Attenuation of 8, 10, 12 dB (both TFI and FFI)
53.3 Mbps, 80 Mbps, 106.7 Mbps, 160 Mbps, 200 Mbps	-17.0 dB	-15.5 dB	-14.5 dB
320 Mbps, 400 Mbps, 480 Mbps	-19.5 dB	-18.0 dB	-17.0 dB

RxSensitivity

Unsigned8

Table 6-4: Minimum receiver sensitivities for band group 1 in UWB

Data Rate (Mbps)	Min. Rx Sensitivity (dBm)
53.3	-80.8
80	-78.9
106.7	-77.8
160	-75.9
200	-74.5
320	-72.8
400	-71.5
480	-70.4

LinkQualityIndicator

Unsigned8

All the HR UWB devices have to be capable of estimating values in the range from -6 dB to +12 dB. Estimating values above +12 dB is optional.

6.3 UWB Security Configuration Class

6.3.1 Object Overview

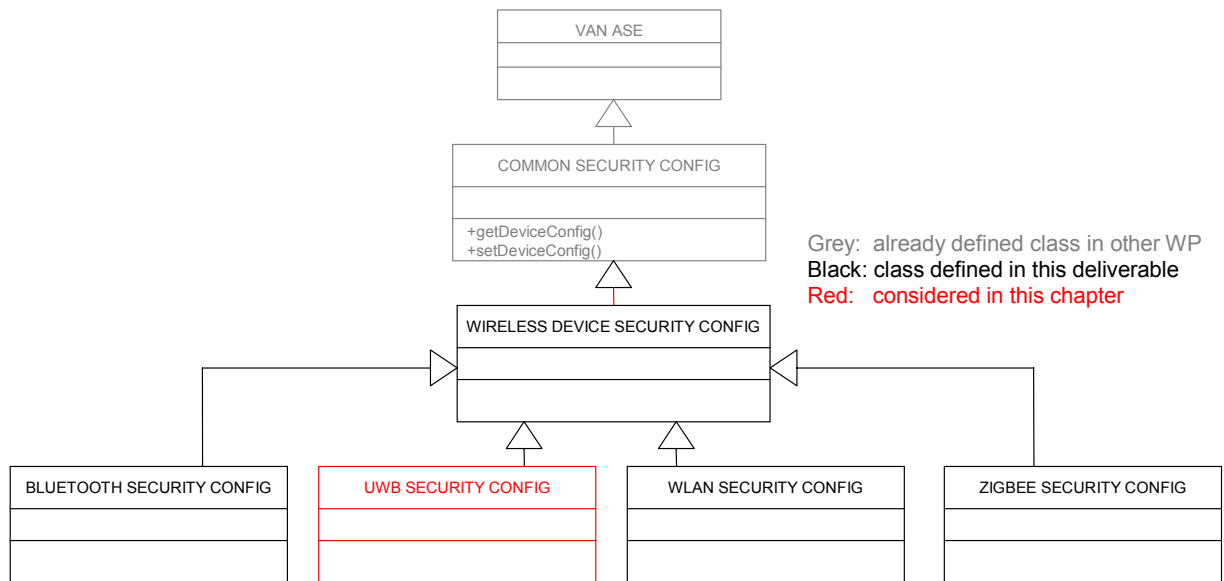


Figure 6-2: Deduction of the UWB SECURITY CONFIG class structure

The salient features of security mechanisms in UWB are as given below:

- Two levels of security: no security and strong security protection
- Strong protection includes: data encryption, message integrity and replay attack protection
- Three security modes:
 - Mode 0: a device can communicate without any security protection
 - Mode 1: a device can use both secure and non-secure frames for data exchange
 - Mode 2: a device is restricted to use security features in transceiving certain frames
- A 4-way handshake procedure is used between two devices to establish PTKs (pair-wise temporal keys) and consequently a secure relationship. A shared master key is used for this purpose.
- GTKs (group temporal keys) are used within a secure relationship for protecting multicast and broadcast frames.
- AES-128 (advanced encryption standard) encryption algorithm with CCM (cipher block chaining message authentication code) cryptography is used to provide payload encryption and MIC (message integrity code) generation. AES is specified in FIPS PUB 197. AES-128 defines a symmetric block cipher that processes 128-bit data blocks using 128-bit cipher keys. CCM, counter with CBC-MAC, is specified in RFC 3610. CCM employs counter mode for encryption and cipher block chaining for authentication. AES-128 CCM combines AES-128 with CCM to encrypt and authenticate messages.
- A PRF (pseudo-random function) based on the MIC generation by CCM using AES-128 can be made available to entities outside the MAC sub-layer for random number generation.

6.3.2 Refinement of Inherited Attributes

object-reference

String

This attribute is inherited from the VAN ASE. The content of the string in this class is "UWB".

6.3.3 Formal Model

VAN ASE: SECURITY CONFIG ASE

CLASS: UWB SECURITY CONFIG

CLASS ID:

PARENT CLASS: WIRELESS DEVICE SECURITY CONFIG

ATTRIBUTES:

1. (m) Attribute: SecurityLevel
2. (m) Attribute: SecurityMode
3. (m) Attribute: 4WayHandshakeMsg
4. (m) Attribute: GTKexchange
5. (m) Attribute: PRF

6.3.4 Attribute Description

SecurityLevel Boolean

Two levels of security are provided in UWB: no security and strong security protection. Security protection includes data encryption, message integrity, and replay attack protection. Secure frames are used to provide security protection to data and aggregated data frames as well as selected control and command frames.

Table 6-5: Security levels in UWB

	SecurityLevel	
Value	0	1
Meaning	No security	Strong security

SecurityMode Binary

Security modes control the level of security required for a device while it is communicating with other devices. Three security modes are provided. Mode 0 allows a device to communicate without any security protection. Mode 1 allows a device to use both secure and non-secure frames for data exchange. Mode 2 restricts a device to use security facilities in transmitting and receiving certain frames. A device announces its selected security mode in the Beacon Parameters field in its beacons.

Table 6-6: UWB security modes

SecurityMode	0	1	2
Value*	00	01	10

* This value corresponds to bits b7-b6 of the Device Control field of Beacon Parameters field of UWB MAC's beacon frame payload (Ref. section 16.3 of [ECMA368]).

4WayHandshakeMsg String

The 4-way handshake mechanism enables two devices to use a shared master key to authenticate the identity of each other and to establish a new PTK for protecting certain frames exchanged between the two devices. A successful 4-way handshake enables devices to establish a secure relationship with each other. To perform a 4-way handshake, the two devices assume the roles of "initiator" and "responder", respectively. A 4-way handshake consists of four messages, called message 1, message 2, message 3, and message 4, which are sent back and forth between the two devices. The device sending message 1 becomes the initiator, with the other device becoming the responder. The handshake procedure is illustrated in the following table.

Table 6-7: Handshake messages in UWB

Message	Sender	Description
4-way handshake message 1	Initiator	<ul style="list-style-type: none"> * specify MKID (master key identifier) * propose a TKID (temporal key identifier) * include I-Nonce (a unique 128-bit cryptographic random number)
4-way handshake message 2	Responder	<ul style="list-style-type: none"> * generate R-Nonce (a new 128-bit cryptographic random number) * derive PTK and KCK (key confirmation key) * indicate if TKID is unique or not in Status Code
4-way handshake message 3	Initiator	<ul style="list-style-type: none"> * derive PTK and KCK * include the same I-Nonce as in message 1 * recalculate PTK MIC for received message using KCK and check if responder holds correct master key * check Status Code returned in received message
4-way handshake message 4	Responder	<ul style="list-style-type: none"> * verify PTK MIC for message 3 using KCK and check if initiator holds correct master key * include the same R-Nonce as in message 2 * install PTK using MLME-KEY-UPDATE primitives

GTKexchange

Unsigned128

Upon successful completion of a 4-way handshake and installation of the resulting PTK, the initiator and responder each shall use GTK command frames (with Message Number set to 1) to distribute their respective GTKs for broadcast traffic to each other. Each may also use a GTK command to distribute a GTK for protecting certain multicast traffic to an intended recipient with which it holds a valid PTK.

The following steps define the GTK exchange procedure:

- On reception of a valid GTK command frame marked as Message Number 1, a device shall verify that the GTKID is a unique TKID. The device shall then respond with a GTK command frame with Message Number set to 2 and Status Code set to the appropriate value.
- A recipient may request a GTK for certain multicast traffic in the form of a GTK command (with Message Number set to 0) from the source device if it holds a valid PTK with the source.
- On reception of a valid GTK command marked as Message Number 0, the multicast source device shall respond with a GTK command marked as Message Number 1, which may or may not contain the requested GTK. The requesting device, upon receiving this GTK command and verifying the uniqueness of the proposed TKID, shall further return a GTK command with Message Number set to 2 and Status Code set to the appropriate value.
- A source device distributing a GTK shall check the Status Code indicated in the returned GTK command (Message Number set to 2). If the Status Code indicates a conflict of the proposed TKID at the recipient device, the source device shall propose a new TKID and redistribute the GTK to the recipient. After receiving a returned GTK command from the recipient with the Status Code indicating a normal status, the source device shall use the new TKID to redistribute the GTK to each of the devices to which it has previously distributed the GTK and with which it maintains a secure relationship.
- A device installs a newly distributed or received GTK using the MLME-KEY-UPDATE primitives. A GTK shall be a 128-bit cryptographic-grade random number. A fresh GTK shall be generated when the distributing device establishes a new group relationship.

PRF

Unsigned256

There are three variants of the pseudo-random function:

- PRF-64 that outputs 64 bits
- PRF-128 that outputs 128 bits
- PRF-256 that outputs 256 bits

CCM-MAC-FUNCTION($K, N, A, B, Blen$)

begin

Form authentication block B_0 from flags = 0x59, N , and $l(m) = 0$

Form authentication block B_1 from $l(a) = 14 + Blen$ and A

Form additional authentication blocks from B

(with last block zero padded as needed)

Form encryption block A_0 from flags = 0x01, N , and Counter_0 = 0

$R \leftarrow \text{MIC}(K, B_0, B_1, \dots, A_0)$

return R

PRF($K, N, A, B, Blen, Len$)

for $i \leftarrow 1$ **to** $(Len + 63)/64$ **do**

$R \leftarrow R \parallel \text{CCM-MAC-FUNCTION}(K, N, A, B, Blen)$

$N \leftarrow N + 1$

return $L(R, 0, Len) = Len$ most-significant bits of R

where, K : 128-bit symmetric key

N : 13-octet nonce value

A : 14-octet ASCII text label for each different use of PRF

B : input data stream

$Blen$: length of input data stream

\parallel : concatenation

The blocks are each 16 octets long and are defined as inputs to the AES-128 CCM for MIC generation.

PRF-64($K, N, A, B, Blen$) = PRF($K, N, A, B, Blen, 64$)

PRF-128($K, N, A, B, Blen$) = PRF($K, N, A, B, Blen, 128$)

PRF-256($K, N, A, B, Blen$) = PRF($K, N, A, B, Blen, 256$)

PTK and KCK generation

PRF-256 is employed in the derivation of PTK and KCK (associated with a 4-way handshake) as described below:

KeyStream \leftarrow PRF-256($K, N, A, B, Blen$)

where, K : PMK

N : B12-11= InitiatorDevAddr, B10-9= ResponderDevAddr, B8-6 = PTKID, B5-0 = zero

A : "Pair-wise keys"

B : I-Nonce \parallel R-Nonce

$Blen$: 32

This key stream is then split to form the desired PTK and KCK. The least-significant 16 octets (0-15) of KeyStream form the KCK while the most-significant 16 octets (16-31) form the PTK.

The PTK and KCK generation parameters are given in the following table.

Table 6-8: PTK and KCK generation parameters in UWB

Name	Size (octets)	Description
InitiatorDevAddr	2	DevAddr of device with role of initiator
ResponderDevAddr	2	DevAddr of device with role of responder
I-Nonce	16	Random number selected by initiator (in message 1)
R-Nonce	16	Random number selected by responder (in message 2)
PTKID	3	Negotiated TKID value for the PTK to be derived (in message 1)
PMK	16	A pre-shared pair-wise master key identified by the MKID (in message 1)

PTK MIC generation

PRF-64 is used to provide the PTK MIC calculation. The 4-way handshake uses an "out-of-band MIC" calculation for the PTK MIC field in handshake messages 2-4.

$PTK\ MIC \leftarrow PRF-64(K, N, A, B, Blen)$

where, K: KCK

N: B12-11 = InitiatorDevAddr, B10-9 = ResponderDevAddr, B8-6 = PTKID, B5-0 = zero

A: "out-of-bandMIC"

B: Fields from Message Number to I-Nonce/R-Nonce contained in the PTK command

Blen: Length in octets of B = 48

Random number generation

The 4-way handshake requires each party to supply a 128-bit random number, which can be generated using the seed and PRF-128 as given below.

GenerateRandomNonce

begin

N = DevAddr || DevAddr || zero

Collect randomness samples

result = PRF-128(Global Seed, N, "Random Numbers", <randomness samples>, length of samples)

return result

GlobalSeed can be derived using random samples and PRF-128 as follows:

LoopCounter = 0

Nonce = 0

while LoopCounter < 32 begin

result = PRF-128(0, Nonce, "InitRandomSeed", DevAddr || Time || result || LoopCounter, dataLen)

Nonce ← Nonce + 1

result ← result || <randomness samples>

end

GlobalSeed=PRF-128(0, Nonce,"InitRandomSeed", DevAddr||Time||result||LoopCounter, dataLen)

6.4 UWB Diagnosis Class

6.4.1 Object Overview

This sub-section defines the UWB MAC sub-Layer Management Entity (MLME) SAP interface attributes that are provided for the Device Management Entity (DME) to monitor events related to the link status of UWB devices.

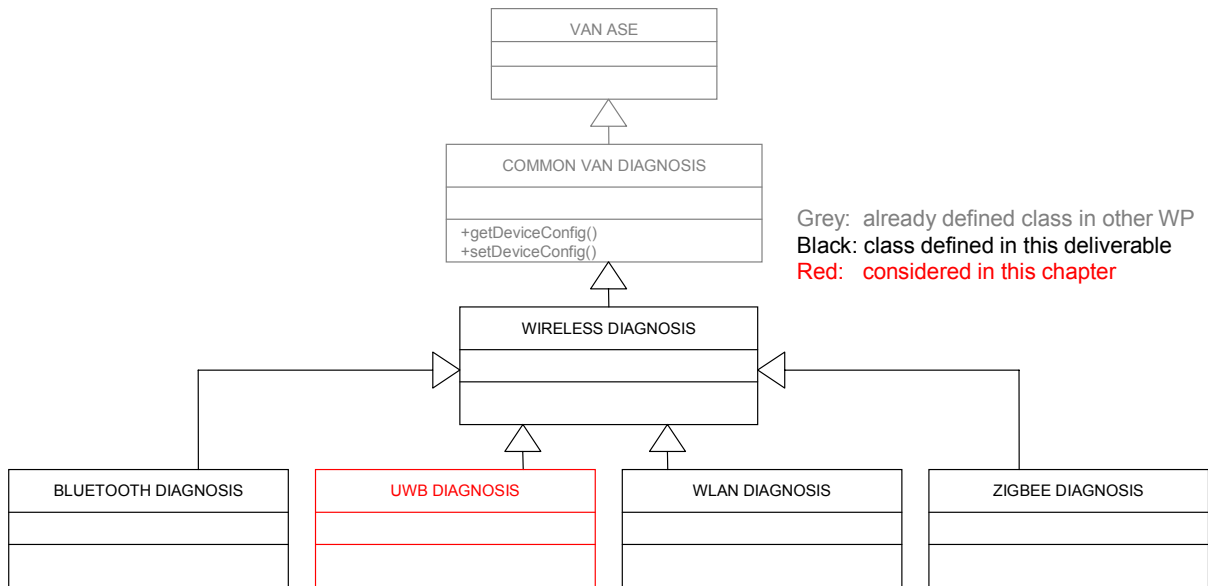


Figure 6-3: Deduction of the UWB DIAGNOSIS class structure

6.4.2 Refinement of Inherited Attributes

object-reference

String

This attribute is inherited from the VAN ASE. The content of the string in this class is "UWB".

6.4.3 Formal Model

VAN ASE: DIAGNOSIS ASE

CLASS: UWB DIAGNOSIS

CLASS ID:

PARENT CLASS: WIRELESS DIAGNOSIS

ATTRIBUTES:

- 1 (m) Attribute: MonitorState
- 2 (m) Attribute: Beacon
- 3 (m) Attribute: LinkEventType
- 4 (m) Attribute: ReceiveErrorInfo

6.4.4 Attribute Description

MonitorState

Enumeration

This attribute specifies whether link event observation is active for the specified link. Valid values are: DISABLED, ENABLED.

Beacon

Enumeration

Whether the received frame was a beacon is indicated by this attribute. Valid values are: FALSE, TRUE.

LinkEventType

Enumeration

This attribute indicates the type of link event that occurred on the link being monitored. Valid values are: RECEIVE_SUCCESS, RECEIVE_ERROR, TRANSMIT_SUCCESS, TRANSMIT_ERROR.

ReceiveErrorInfo

Enumeration

This attribute provides additional information for an RECEIVE_ERROR type of link event. Valid values are: PAYLOAD_ERROR, UNSUPPORTED_RATE_ERROR, GENERAL_ERROR.

6.5 VAN Heterogeneous Network Technologies Adaptation Layer

This section contains a brief description of a UWB MAC frame format defined by the WiMedia Alliance. The UWB MAC frame consists of a fixed-length MAC Header (Table 6-9) and an optional variable-length MAC Frame Body. The frame body contains (secured) payload data and frame check sequence (4 octets).

Table 6-9: UWB MAC Header

Octets	2	2	2	2	2
Meaning	Frame control	Source address	Destination address	Sequence control	Access information

The MAC header consists of the following fields:

- Frame control octets are defined in Table 6-11.
- Source address specifies address of the source device.
- The Destination address is an address of the intended recipient(s) of the frame. It can be a single device for a unicast frame, a group of devices for a multicast frame or all devices for a broadcast frame (Table 6.10)
- The Sequence Number field value is used for duplicate detection for frames. A device shall assign each transmitted frame a sequence number from a counter modulo 2 048.
- Access information contains information about frame fragmentation and estimated time of a frame transmission.

Table 6-10: Device addressing in UWB

Type	Range
Private	0x0000 - 0x00FF
Generated	0x0100 - 0xFEFF
Multicast	0xFF00 - 0xFFFE
Broadcast	0xFFFF

Table 6-11: Frame control in UWB

Bits	15-14	13	12-9	8-6	5-4	3	2-0
Meaning	Reserved	Retry	Frame Subtype / Delivery ID	Frame Type	ACK Policy	Secure	Protocol Version

Frame control field contains the following fields:

- Protocol Version – in current version is zero.
- Secure – if an encryption is applied, this bit is set to one.
- ACK Policy – this field is set to the type of acknowledgement requested by the transmitter (see Table 6-12).
- Frame Type – value specifies one of the following frame types: beacon frame, control frame, command frame, data frame and aggregated data frame.
- The Frame Subtype / Delivery ID field is used to assist a receiver in the proper processing of received frames. For data frames, it contains stream index or user priority.
- Retry – This bit is set to one in any data, aggregated data, or command frame, in other cases is reserved.
- Reserved.

Table 6-12: ACK policy in UWB

Value	ACK policy type	Description
0	No-ACK	The recipient(s) do not acknowledge the transmission, and the sender treats the transmission as successful without regard for the actual result
1	Imm-ACK	The addressed recipient returns an Imm-ACK frame after correct reception. It provides an acknowledgement process in which each frame is individually acknowledged following the reception of the frame
2*	B-ACK	The addressed recipient keeps track of the frames received with this policy until requested to respond with a B-ACK frame, lets the source send multiple frames without intervening ACK frames. The acknowledgements of the individual frames are grouped into a single response frame that is sent when requested by the source device.
3*	B-ACK Request	The B-ACK mechanism allows a source device to transmit multiple frames and to receive a single acknowledgement frame from the recipient indicating which frames were received and which need to be retransmitted

* The B-ACK mechanism allows a source device to transmit multiple frames and to receive a single acknowledgement frame from the recipient indicating which frames were received and which need to be retransmitted. A source device initiates the use of the B-ACK mechanism with a recipient device for frames either from the same stream or of the same user priority. If the recipient device accepts use of the B-ACK mechanism, it indicates the maximum number and size of the frames it can buffer. The source device transmits a sequence of frames to the recipient, each from the same stream or of the same user priority, limited by the announced buffer size and maximum number of frames. The initial frames in the sequence are all transmitted with ACK Policy set to B-ACK. The final frame in the sequence is transmitted with ACK Policy set to B-ACK Request. On receipt of such a frame, the recipient device returns a B-ACK frame giving feedback on the frames received and indicating the buffer space available for the next B-ACK sequence. A source device may invoke multiple instances of the B-ACK mechanism with the same recipient device, each for a different stream or user priority. A source device may also invoke the B-ACK mechanism with multiple recipient devices.

7 Wireless LAN Contributions to VAN ASEs

7.1 Relevant Architecture Elements

This chapter describes elements of the VAN device architecture which are specific for the different wireless technologies for instance the VAN Proxy application process. Furthermore, it contributes to common elements such as the VAN Heterogeneous Network Technologies Adaptation Layer. It is mentioned which part of the wireless VAN device architecture deals with which special requirement of industrial application environments.

7.2 WLAN Device Configuration Class

7.2.1 Object Overview

The VAN device configuration object specifies the attributes that can be used to configure IEEE 802.11 devices or networks. In case of a VAN-Proxy-Approach, the proxy contains one Management Information Base (MIB) for each node in the network. Additional management applications on coordinator and routers or devices are necessary to exchange the attribute values between all relevant nodes.

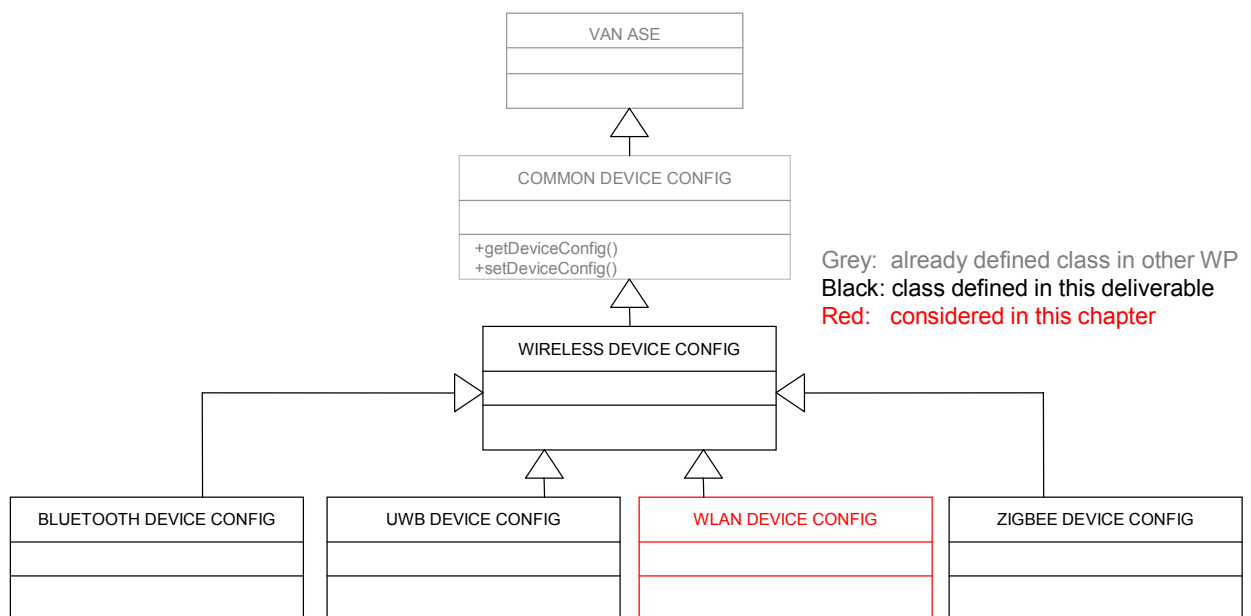


Figure 6-1: Deduction of the WLAN DEVICE CONFIG class structure

7.2.2 Refinement of Inherited Attributes

object-reference String

This attribute is inherited from the VAN ASE. The content of the string in this class is "WLAN".

7.2.3 Formal Model

VAN ASE: DEVICE CONFIG ASE

CLASS: WLAN DEVICE CONFIG

CLASS ID:

PARENT CLASS: WIRELESS DEVICE CONFIG

ATTRIBUTES:

- 1 (m) Attribute: TransmitPower
- 2 (m) Attribute: PhyCurrentChannel
3. (m) Attribute: RadioMode

4. (m) Attribute: BeaconFreq
6. (m) Attribute: SSID
7. (m) Attribute: DataRate
8. (m) Attribute: PhyChannelsSupported
9. (m) Attribute: ipAddress
10. (m) Attribute: ipMask
11. (m) Attribute: CountryCode

7.2.4 Attribute Description

Subchapter 5.2 provides a subset of the WLAN attributes. These basic WLAN Attributes can be controlled by a VAN engineering.

TransmitPower integer8

The transmit power can be reduced from a max value to 1/16 of this value. This allows the user to reduce the size of the WLAN cell.

PhyCurrentChannel unsigned8

The radio channel currently used. This could be changed during runtime.

RadioMode Character

One of the basic 802.11 modes, i.e. a,b,g,h

BeaconFreq unsigned16

Frequency of beacon send. A common default value is 10/s

SSID String

Name of the wireless network.

DataRate unsigned8

Allowed data rates depend on the selected RadioMode and can vary between 1 MBit/s and 54 MBit/s

PhyChannelsSupported unsigned32

All supported channels, also depending on the selected RadioMode.

ipAddress unsigned32

IP-Address of the WLAN device.

ipMask unsigned32

IP-Mask of the WLAN device

CountryCode String

Channels and max transmit power depend on the national legislation. Therefore a country code must be set in each WLAN device to assure the correct behaviour.

7.3 WLAN Security Configuration Class

7.3.1 Object Overview

This chapter describes the security measures and parameters, which are specified by the selected wireless technologies. The aim is to provide these information to work package 6 to be considered in the overall security concept and to work package 8.

The WLAN security is defined by IEEE802.11i. This standard describes several options to encrypt data and to authenticate devices on a network. VAN supports a common subset of these options. The parameters needed therefore are defined in subsection 5.6.2.

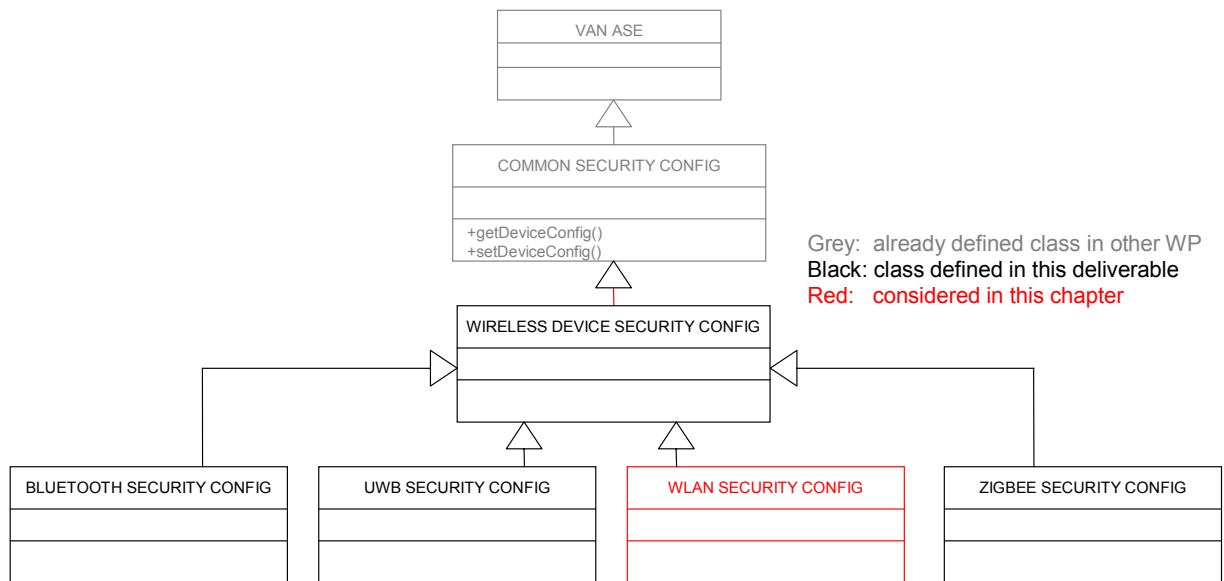


Figure 6-2: Deduction of the WLAN SECURITY CONFIG class structure

7.3.2 Refinement of Inherited Attributes

object-reference

String

This attribute is inherited from the VAN ASE. The content of the string in this class is "WLAN".

7.3.3 Formal Model

VAN ASE: SECURITY CONFIG ASE

CLASS: WLAN SECURITY CONFIG

CLASS ID:

PARENT CLASS: WIRELESS DEVICE SECURITY CONFIG

ATTRIBUTES:

- 1 (m) Attribute: WLANSecurityMode
- 2 (m) Attribute: WLANEncryptionMode
- 3 (m) Attribute: WLANAuthenticationMode
- 4 (m) Attribute: WLANBasicKeys
- 5 (m) Attribute: WLANAccessRights
- 6 (m) Attribute: WLANServiceAccess

7.3.4 Attribute Description

WLANSecurityMode

String

A general security mode must be defined for a WLAN link. The values for this mode depend on the implementation of the WLAN devices. VAN could define several security mode levels, e.g. low, medium, high and map certain values for the encryption and authentication to these levels.

WLANEncryptionMode

unsigned8

Basically three different encryption approaches are available for the WLAN devices. The weakest is WEP. A bit stronger is TKIP. The best option is AES.

WLANAuthenticationMode

unsigned8

Several authentication solutions are defined in the IEEE 802.11i standard. Beside pre-shared key (PSK) several EAP options based on RADIUS are defined.

WLANBasicKeys unsigned32

A set of basic keys is required on each WLAN device. The size of this set and the key length is defined by the selected security mode.

WLANAccessRights unsigned8

The access to a WLAN device could be defined by the access rights. Depending on the access rights a user could access to a WLAN device, could access to diagnosis information or even could reconfigure the device.

WLANServiceAccess unsigned8

A WLAN device could, again depending on the implementation, support several services for monitoring, diagnosis and configuration. Examples are http, ftp, SNMP or a CLI. Such services could be enabled/disabled according to a security policy.

8 IEEE 802.15.4 and ZigBee Contributions to VAN ASEs

8.1 Relevant Architecture Elements

This chapter specifies contributions to VAN ASEs from the point of view of the wireless standards IEEE 802.15.4 and ZigBee. Based on these standards attributes are assigned to the associated ASEs and described in detail so that they can be used by other work packages to work out overall VAN strategies. IEEE 802.15.4 specifies physical layer and MAC layer for wireless sensor networks. Based on this standard several proprietary higher layer implementations are known. However, today ZigBee is the only specification which has been worked out by an international non-profit organisation, the ZigBee Alliance. The ZigBee specification considers also industrial automation applications. That is, even if ZigBee should not be used in the context of industrial automation other standards have to specify the same attributes and a similar device and system behaviour. That is why the ZigBee specification is used in this document as an example for higher layer specifications on top of IEEE 802.15.4.

With respect to the application field and topologies mentioned in section 2.5 the VAN device types VAN Proxy and VAN Virtual Device are considered in this document. A VAN ZigBee Proxy Device contains one set of ASEs for each VAN Virtual Device. ZigBee management services are used in order to transfer the attribute data from the VAN Virtual Devices to the VAN ZigBee Proxy Device.

8.2 ZigBee Device Configuration Class

8.2.1 Object Overview

The ZIGBEE DEVICE CONFIG ASE object specifies the attributes that can be used to configure IEEE 802.15.4 or ZigBee devices or networks. It is derived from the WIRELESS DEVICES CONFIG as shown in Figure 8-1.

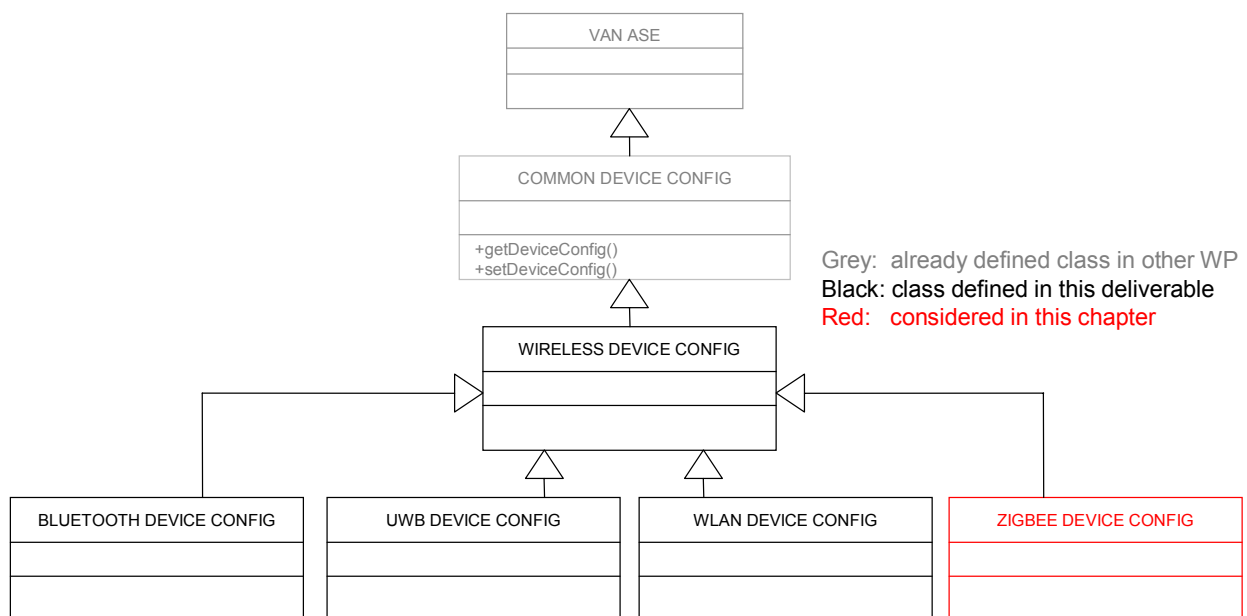


Figure 8-1: Deduction of the ZIGBEE DEVICE CONFIG class structure

8.2.2 Refinement of Inherited Attributes

object-reference

String

This attribute is inherited from the VAN ASE. The content of the string in this class is "ZigBee".

PromiscuousMode

Boolean

This attribute is inherited from class WIRELESS DEVICE CONFIG. The attribute `macPromiscuousMode` defined in IEEE 802.15.4 is equivalent to `PromiscuousMode`. It indicates whether the MAC sublayer is in a receive all (promiscuous) mode. A value of TRUE indicates that the MAC sublayer accepts all frames received from the PHY. This could be viewed as a type of packet-sniffing feature.

CurrentFrequencyChannel

This attribute is inherited from `PhysicalRadio-structure` of the class WIRELESS DEVICE CONFIG. The attribute `PhyCurrentChannel` (Unsigned8) defined in IEEE 802.15.4 is related to `CurrentFrequencyChannel`. It indicates the RF channel to use for all transmissions and receptions. The exact channel number allowed is restricted to the range determined by which physical layer is used (see Table 6-2). The channel number used for the network is normally selected by a higher layer based on an algorithm which uses the results returned by the energy detect scan, but this can be manually selected and changed as well. However, the ability to change the channel number is restricted to the PAN coordinator only.

The channel value selected should correspond to within the allowed channel number range as presented in Table 6-2. Since the number of useable channels is dependant on the PHY hardware platform used, the vendor datasheet for the device should be consulted to determine the exact input value range allowed to select the desired RF channel.

Table 8-1: Number of channels per band

Frequency	Region	Channels	Input Range
2450MHz	Worldwide	16	12 – 27
915MHz	USA	10	2 – 11
868MHz	Europe	1	1

TransmitPower

Unsigned16

This attribute is inherited from class WIRELESS DEVICE CONFIG. In case of IEEE 802.15.4 and ZigBee devices several values to specify the transmit power are possible. This value will be entered directly because of the different vendor implementations for setting the RF transmit power. Therefore, the vendor datasheet for the PHY hardware platform used should be consulted to determine the exact input value required to achieve the desired RF output power level. The table below is an example of Chipcon's CC2420 transceivers implementation for the transmit output power parameter. The power amplifier level (`PA_level`) is the decimal input value required in the specific register (in the CC2420 case this is the 16 bit `TXCTRL` register) in order to obtain the corresponding output RF power level (in dBm).

Table 8-2: Transmit power levels

PA_level	TXCTRL register value	Output (dBm)
3	0xA0E3	-25
7	0xA0E7	-15
11	0xA0EB	-10
15	0xA0EF	-7
19	0xA0F3	-5
23	0xA0F7	-3
27	0xA0FB	-1
31	0xA0FF	0

GlobalTimePeriod

This attribute is inherited from MediaAccess-Structure of the class WIRELESS DEVICE CONFIG. The attribute macBeaconOrder (Unsigned8) defined in IEEE 802.15.4 is related to GlobalTimePeriod.

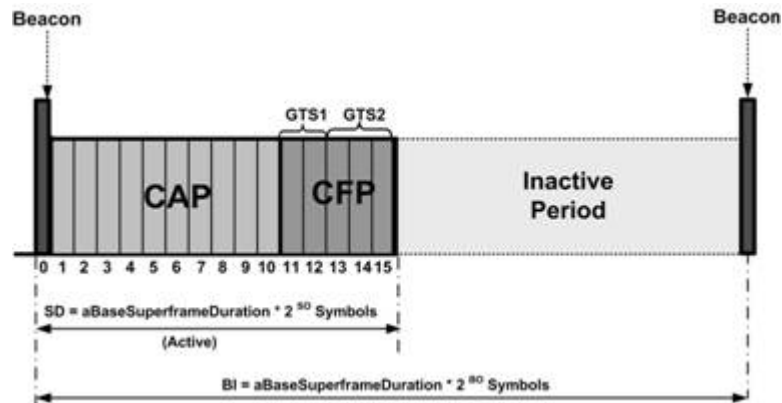


Figure 8-2: MAC superframe structure with GTS slots

The value of the macBeaconOrder attribute is an indication of how often a PAN coordinator will transmit a beacon. The beacons are used to synchronise devices, to identify the PAN, as well as to describe the superframe structure.

The allowable range of this attribute is from 0 to 15. For non-beacon networks, this value will be set to 15. In this case, the network will use unslotted CSMA/CA to gain access to the medium. For beacon-enabled networks, where the use of a superframe structure bounded by two periodic signaling beacon frames is used (see figure 6-4 above), this value can vary from 0 to 14. A superframe structure contains 16 equal time slots, during which other nodes belonging to this PAN are allowed to transmit. The transmission scheme used in this mode is slotted CSMA/CA during the contention access period (CAP). In this mode, it is also possible to allocate guaranteed time slots (GTS) which are contention free and used for time sensitive application requirements.

The detailed structure of the MAC superframe is specified by the macBeaconOrder attribute, as well as the macSuperframeOrder.

The Beacon Interval (BI) is the time between two consecutive beacons which includes an active period (the superframe) and possibly an inactive period (during which time the nodes may sleep).

The Beacon Interval (BI) and macBeaconOrder (BO) are related such that the $BI = 960 * 2^{BO}$ where BO can be in the range from and including 0 to 15. A value for BO of 0 will result in a beacon interval of 15.36ms and a BO value of 14 will result in a beacon interval of 251 seconds. Recall that a value of 15 for BO will disable the transmission of beacons.

The GlobalTimePeriod is equivalent to the Beacon Interval.

8.2.3 Formal Model

VAN ASE: DEVICE CONFIG ASE

CLASS: ZIGBEE DEVICE CONFIG

CLASS ID:

PARENT CLASS: WIRELESS DEVICE CONFIG

ATTRIBUTES:

1. (m) Attribute: macBeaconPayload
2. (m) Attribute: macSuperframeOrder
3. (m) Attribute: macAssociationPermit
4. (m) Attribute: macBSN
5. (m) Attribute: macDSN
6. (m) Attribute: PhyCCAMode
7. (m) Attribute: macPANId
8. (m) Attribute: macCoordExtendedAddress
9. (m) Attribute: macCoordShortAddress

10. (m) Attribute: macShortAddress
11. (m) Attribute: macAutoRequest
12. (m) Attribute: macTransactionPersistenceTime
13. (m) Attribute: macGTSPermit
14. (m) Attribute: macMaxCSMABackoffs
15. (m) Attribute: macMinBE
16. (m) Attribute: nwkSequenceNumber
17. (m) Attribute: nwkPassiveAckTimeout
18. (m) Attribute: nwkMaxBroadcastRetries
19. (m) Attribute: nwkRouteTable
20. (m) Attribute: nwkMaxChildren
21. (m) Attribute: nwkMaxDepth
22. (m) Attribute: nwkMaxRouters
23. (m) Attribute: nwkUseTreeAddrAlloc
24. (m) Attribute: nwkUseTreeRouting
25. (m) Attribute: nwkNextAddress
26. (m) Attribute: nwkAddressIncrement
27. (m) Attribute: nwkTransactionPersistenceTime
28. (m) Attribute: macAckWaitDuration
29. (m) Attribute: nwkNetworkBroadcastDeliveryTime
30. (m) Attribute: macBeaconPayloadLength
31. (m) Attribute: macBattLifeExt
32. (m) Attribute: macBattLifeExtPeriods
33. (m) Attribute: macRxOnWhenIdle
34. (m) Attribute: nwkSymLink
35. (m) Attribute: nwkReportConstantCost
36. (m) Attribute: nwkRouteDiscoveryRetriesPermitted
37. (m) Attribute: UserDescriptor

8.2.4 Attribute Description

IEEE 802.15.4 standard as well as the ZigBee specification defines attributes which can be changed during operation, which can be set-up within the commissioning phase or which are fixed depending on the type of network. The latter e.g. depend on the frequency band used and therefore on the hardware.

Even if the values of attributes can only be changed in the programming phase and are therefore mostly fixed in the program code they are listed here to provide a complete picture of the possible devices and network constellations.

macBeaconPayload

STRUCTURE

The current contents of the beacon payload are contained in this attribute. The NWK layer of the ZigBee coordinator shall update the beacon payload immediately following network formation. All other ZigBee devices shall update it immediately after the association is completed and anytime the network configuration changes. Another attribute (macBeaconPayloadLength) specifies the length of the beacon payload. The beacon payload is written into the macBeaconPayload attribute when the length is non-zero. The formatting of the byte sequence representing the beacon payload is shown in Table 6-3 (taken from IEEE 802.15.4. specification)

Table 8-3: MAC beacon payload format

Bits: 0-7	8-11	12-15	16-17	18	19-22	23	24-27
Protocol ID	Stack profile	nwkcProtocolVersion	Reserved	Router capacity	Device depth	End device capacity	Tx offset (optional)

macSuperframeOrder Unsigned8

This attribute specifies the length of the active portion of the superframe, including the beacon frame. The lengths of the Beacon Interval and the Superframe Duration (SD) are determined by two parameters BO (Beacon Order) and SO (Superframe Order) respectively. The SD, which determines the length of the active period, is defined as follows: $SD = 960 * 2^{SO}$ where SO can be in the range from and including 0 up to and including either BO or 14.

macAssociationPermit Boolean

The enabling or disabling of this attribute is an indication of whether a PAN coordinator is currently allowing association of new devices to the network. For the case where devices have previously associated with the network and wish to rejoin, the PAN coordinator will remove all its previous device-specific information and the usual association procedure will take place. A Boolean value of TRUE indicates that association is permitted and that the device is allowed to join the network. The default for this attribute however, is FALSE. The attribute is optional for reduced function devices since it applies to the coordinator.

macBSN Unsigned8

This is the beacon sequence number added to the end of the transmitted beacon frame.

macDSN Unsigned8

This is the sequence number added to each transmitted data or MAC command frame.

PhyCCAMode Unsigned8

The PhyCCAMode attribute is used to indicate the operation mode for clear channel assessment (CCA) in the device. This can be one of three options according to Table 6-5. The clear channel assessment signal is usually based on the received signal strength indicator (RSSI) value and a programmable threshold. The CCA function is used to implement the CSMA-CA functionality required in a IEEE 802.15.4 (ZigBee) radio. Not all vendor implementations support all three CCA modes and it is advisable to consult the appropriate vendor datasheet in order to determine the available mode options.

Table 8-4: CCA mode descriptions

CCA Mode	Description
Mode 1: Energy above threshold	The CCA will report a busy medium if any energy about the ED (energy detection) threshold is detected
Mode 2: Carrier sense only	A busy medium will be reported by the CCA if a signal with the modulation and spreading characteristic of the IEEE 802.15.4 is detected that may be above or below the ED threshold.
Mode 3: Carrier sense with energy above threshold	This mode is a combination of mode 1 and 2 and the CCA will report busy medium only upon detection of a signal with the modulation and spreading characteristic of the IEEE 802.15.4 and if energy about the ED threshold is detected

macPANId Unsigned8

The 16 bit identifier of the PAN on which the device is operating. If this value is 0xffff, the device is not associated

macCoordExtendedAddress Unsigned64 (IEEE address)

This is the 64bit address of the PAN coordinator with which the device is currently associated.

macCoordShortAddress Unsigned16

This attribute indicates the 16bit short address that is assigned to the PAN coordinator. A value of 0xfffe indicates that the coordinator is only using its 64 bit extended address. On the other hand, a value of 0xffff indicates that this value is unknown.

macShortAddress Unsigned16

This is the 16 bit address that the device uses to communicate in the PAN. If the device is a PAN coordinator, this value shall be chosen before a PAN is started. Otherwise, the address is allocated by a coordinator during association. A value of 0xffff indicates that the device has associated but has not been allocated an address. A value of 0xffff indicates that the device does not have a short address.

macAutoRequest Boolean

This attribute is an indication of whether a ZigBee device automatically sends a data request command if its address is listed in the beacon frame. A value of TRUE (default) indicates that the data request command is automatically sent.

macTransactionPersistenceTime Unsigned16

The maximum time (in superframe periods) that a transaction is stored by a PAN coordinator. This value is indicated in its beacon.

macGTSPermit Boolean

This is an indication of whether the PAN coordinator is to accept GTS (Guaranteed Time Slot) requests or not. A value of TRUE indicated that the PAN coordinator is accepting GTS requests.

macMaxCSMABackoffs Unsigned8

This is the maximum number of backoffs that the CSMA-CA algorithm will attempt before declaring a channel access failure. The number of backoffs range from 0 to 5 and the default value is 4.

macMinBE Unsigned8

This attribute indicates the minimum value of the backoff exponent in the CSMA-CA algorithm. If this value is set to zero then collision avoidance is disabled in the first execution of the algorithm. The value ranges from 0 to 3 (default).

nwkSequenceNumber Unsigned8

A sequence number used to identify outgoing frames. The NWK layer on every device shall maintain a sequence number that is initialized with a random value. The sequence number shall be incremented by one, each time the NWK layer constructs a new NWK frame, either as a result of a request from the next higher layer to transmit a new NWK data frame or when it needs to construct a new NWK layer command frame. After being incremented the value of the sequence number shall be inserted into the sequence number field of the frame's NWK header.

nwkPassiveAckTimeout Unsigned8

The passive acknowledgment timeout attribute is the maximum time duration (in seconds) allowed for the parent and all child devices to retransmit a broadcast message. The default value is 3 seconds (0x03).

nwkMaxBroadcastRetries Unsigned8

When the transmission of a broadcast message fails, this value specifies the maximum number of retries allowed subsequent to this failure. The value ranges from 0 to 5 with the default being a maximum of 3 retries.

nwkRouteTable STRUCTURE

The current set of routing table entries in the device. Three main fields included in this table are:

- Destination address: the 16bit network address of this route
- Status: Current status of the route which can be either active, discovery_underway, discovery_failed or inactive.
- Next-hop address: The 16-bit network address of the next hop on the way to the destination.

nwkMaxChildren Unsigned8

This value is an indication of the maximum number of children a device is allowed to have on its current network. A large value of this attribute will translate to more resources (i.e. available address space and consequently more device memory) being used. To ensure interoperability between different ZigBee devices from different vendors, this is one of the values which is preset to a specified value for a particular ZigBee profile (e.g. home automations profile has a maximum of 20 children allowed). This value can range from 1 up to and including 32 allowable children.

nwkMaxDepth Unsigned8

This attribute specifies the maximum allowable depth a device may have. For star network topologies this is always one but for tree and mesh networks this value specifies the maximum allowable levels in a particular tree. This value ranges from 2 up till and including 7. This means that a device may have an allowable depth of up to 7 devices. To ensure interoperability between different ZigBee devices from different vendors, this is one of the values which is preset to a specified value for a particular ZigBee profile (e.g. home automations profile allows a maximum depth setting of 5).

nwkMaxRouters Unsigned8

This value indicates the maximum number of routers any one device is allowed to have as children. This value is determined by the ZigBee coordinator for all devices in the network. Therefore, this value may be used to limit the amount of routers that a ZigBee coordinator or router will allow as children. Its value is assigned by the ZigBee coordinator at start-up time and is distributed to all other devices on the network. This value can range from 1 to 32 routers. For the home controls profile, the value is set to 6.

nwkUseTreeAddrAlloc Boolean

A flag that determines whether the NWK layer should use the default distributed address allocation scheme (TRUE) or allow the next higher layer to define a block of addresses for the NWK layer to allocate to its children (FALSE)

nwkUseTreeRouting Boolean

This flag determines whether the NWK layer should assume the ability to use hierarchical routing (TRUE) or not (FALSE).

nwkNextAddress Unsigned16

This is the next network address that will be given to a device requesting association. This value shall be incremented by the amount specified by the nwkAddressIncrement attribute for every time an address is assigned.

nwkAddressIncrement Unsigned16

The amount by which nwkNextAddress (see previous) is incremented each time an address is assigned.

nwkTransactionPersistenceTime Unsigned16

The maximum time (measured in superframe periods) that a transaction is stored by a coordinator and is indicated in its beacon. This attribute reflects the value of the lower layer MAC PIB attribute macTransactionPersistenceTime. Any changes made by the higher layer will be reflected in the MAC PIB attribute value as well.

macAckWaitDuration Unsigned8

This attribute is an indication of the number of symbols that an IEEE 802.15.4 (ZigBee) device should wait for an acknowledgment frame to arrive after it has transmitted a data frame. However, this value is dependent on the currently selected logical channel. Therefore, for ZigBee devices operating in the 2.4GHz range, this value is equal to 54 symbols. For all other frequencies (868MHz and 915MHz) this value is 120 symbols.

nwkNetworkBroadcastDeliveryTime Unsigned8

The time duration (in seconds) that a broadcast message requires in order to encompass the entire network.

macBeaconPayloadLength Unsigned8

This is an indicator of the length of the current beacon payload. The maximum value can be 960 symbols (a symbol corresponds to 4 bits).

macBattLifeExt Boolean

This is an indication of whether battery life extension on the device is enabled. This mode is enabled by reducing the coordinator receiver operation time during the contention access period of a beacon-enabled network. This attribute is also applicable for non-beacon networks. A value of TRUE indicates that it is enabled. The default value is FALSE. This value has an effect on the number of backoff periods (units of time) a device will wait before starting to access a channel since it is related to the CSMA-CA algorithm.

macBattLifeExtPeriods Unsigned8

This value indicates the number of backoff periods during which the receiver is enabled following a beacon in battery life extension mode (see previous). However, this value is dependent on the currently selected logical channel. Therefore, for ZigBee devices operating in the 2.4GHz range, this value is equal to 6 periods (default). For all other frequencies (868MHz and 915MHz) this value is 8 periods.

macRxOnWhenIdle Boolean

This attribute is an indication as to whether the MAC sublayer is to enable its receiver during idle periods. A value of TRUE will cause the receiver to enable its receiver during idle periods and although this might improve the timing performance it will sacrifice the energy supply. The default is FALSE.

nwkSymLink Boolean

This attribute is an indication of the route symmetry setting and says whether the routes are considered to be comprised of symmetric links (TRUE) or FALSE if the forward and return route is different or non symmetric. The default is FALSE and in this option only the forward route is stored during route discovery.

nwkReportConstantCost Unsigned8

If this value is set to zero, the NWK layer shall calculate link cost from all neighbour nodes using the LQI values reported by the MAC layer. Otherwise it shall report a constant value. This value can be either 0x00 (default) or 0x01.

nwkRouteDiscoveryRetriesPermitted Unsigned8

This is the maximum number of retries (from 0 to and including 3) allowed after an unsuccessful route request. The default is three.

UserDescriptor Unsigned16

The User Descriptor contains information that allows the user to identify the device using a user-friendly character string, such as "Tank 5 level" or "Pressure valve 3". The use of the user descriptor is optional. This descriptor contains a single field, which uses the character string data type and has a maximum value of 16 characters.

8.3 ZigBee Security Configuration Class

8.3.1 Object Overview

This chapter describes the security measures and parameters, which are specified by IEEE 802.15.4 and ZigBee. The aim is to provide this information to work package 6 to be considered in the overall security concept as well as to work package 8. The attributes listed are those required to manage the security for different levels within the IEEE 802.15.4 (ZigBee) network.

The ZIGBEE SECURITY CONFIG ASE is derived from the WIRELESS DEVICE SECURITY CONFIG ASE as shown in Figure 8-3.

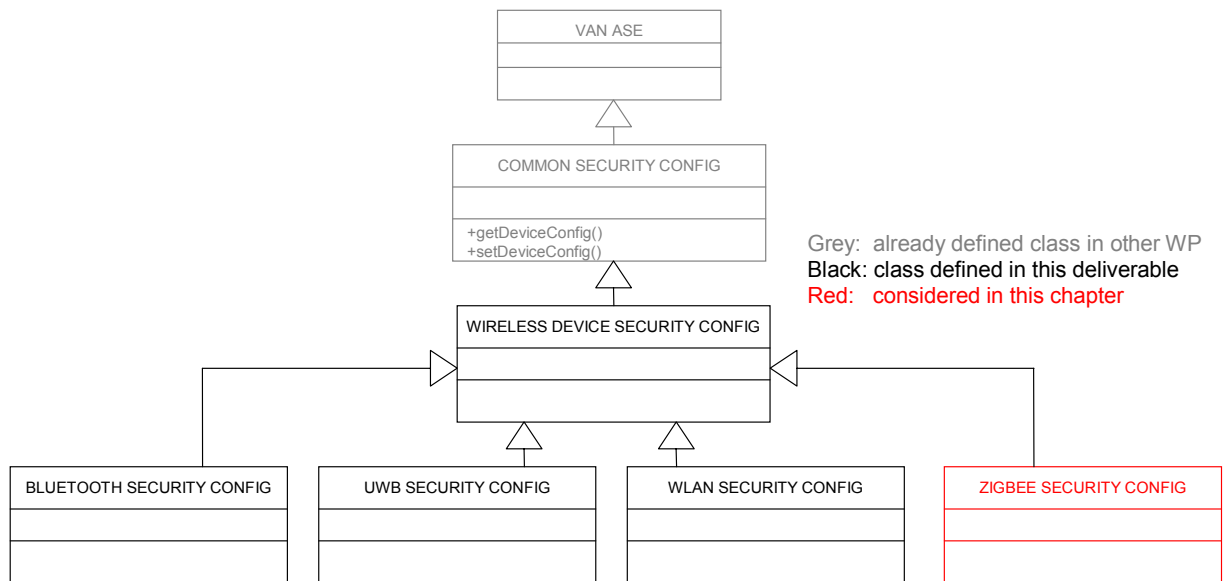


Figure 8-3: Deduction of the ZIGBEE SECURITY CONFIG class structure

8.3.2 Refinement of Inherited Attributes

object-reference

String

This attribute is inherited from the VAN ASE. The content of the string in this class is "ZigBee".

8.3.3 Formal Model

VAN ASE: SECURITY CONFIG ASE

CLASS: ZIGBEE SECURITY CONFIG

CLASS ID:

PARENT CLASS: WIRELESS DEVICE SECURITY CONFIG

ATTRIBUTES:

1. (m) Attribute: macDefaultSecurity
2. (m) Attribute: macSecurityMode
3. (m) Attribute: macACLEntryDescriptorSet
4. (m) Attribute: macACLEntryDescriptorSetSize
5. (m) Attribute: macDefaultSecurityMaterialLength
6. (m) Attribute: macDefaultSecurityMaterial
7. (m) Attribute: macDefaultSecuritySuite
8. (m) Attribute: nwkSecurityLevel
9. (m) Attribute: nwkSecureAllFrames
10. (m) Attribute: nwkSecurityMaterialSet
11. (m) Attribute: nwkActiveKeySeqNumber
12. (m) Attribute: nwkAllFresh
13. (m) Attribute: apsDeviceKeyPairSet
14. (m) Attribute: apsSecurityTime-OutPeriod
15. (m) Attribute: apsTrustCenterAddress

8.3.4 Attribute Description

8.3.4.1 MAC Layer

macDefaultSecurity Boolean

This attribute indicates (default: FALSE) whether a device is able to transmit or receive secure frames from devices not explicitly listed in the access control list (ACL).

macSecurityMode Unsigned8

The MAC layer allows different security services, which is indicated by one of three modes as outlined in the table below.

Table 8-5: Security Modes

Security Mode	Description
0x00 = Unsecured mode	No security services are provided in this mode
0x01 = ACL mode	Limited security services is provided based on the origin of the frame and may cause higher layer to reject specific frames
0x02 = Secured mode	May provide any number of the following 4 security services: Access control Data encryption Frame integrity Sequential freshness

macACLEntryDescriptorSet STRUCTURE

For the macACLEntryDescriptorSet attribute (from the MAC PIB) the upper layer shall set the symmetric key, and outgoing frame counter equal to the corresponding elements of the Network key-pair descriptor in the apsDeviceKeyPairSet of the AIB. The optional external frame counter shall be set to the incoming frame counter. The key sequence counter shall be set to 0x00, and the optional external key sequence counter shall not be used.

For MAC security in ZigBee, the security material from the attribute macACLEntryDescriptorSet shall be used. The values within this attribute are indicated in the table below.

This security attribute is used to Access Control List (ACL) – ACL are used so that only predefined nodes can join the network.

Table 8-6: Access Control List Entry Descriptor Set

Name	Type	Range	Description	Default
ACLExtendedAddress	IEEE address	Any valid 64 bit device address	The 64 bit IEEE extended address of the device in this ACL entry.	Device specific
ACLShortAddress	Integer	0x0000–0xffff	The 16 bit short address of the device in this ACL entry. A value of 0xfffe indicates that the device is using only its 64 bit extended address. A value of 0xffff indicates that this value is unknown.	0xffff
ACLPANId	Integer	0x0000–0xffff	The 16 bit PAN identifier of the device in this ACL entry.	Device specific
ACLSecurityMaterial-Length	Integer	0–26	The number of octets contained in ACLSecurityMaterial.	21
ACLSecurityMaterial	Octet string	Variable	The specific keying material to be used to protect frames between the MAC sublayer and the device indicated by the associated ACLExtendedAddress.	Empty string
ACLSecuritySuite	Integer	0x00–0x07	The unique identifier of the security suite to	0x00

Name	Type	Range	Description	Default
			be used to protect communications between the MAC sublayer and the device indicated by the associated ACLExtendedAddress.	
macACLEntryDescriptorSetSize				Unsigned8
This value is an indication of the total number of entries in the above ACL descriptor set.				
macDefaultSecurityMaterialLength				Unsigned8
This is an indication of the number of octets that the ACL entry descriptor <i>ACLSecurityMaterial</i> contains.				
macDefaultSecurityMaterial				String
This is specific material used to protect frames between MAC sublayer and devices not in the ACL. For the <i>macDefaultSecurityMaterial</i> attribute from the MAC PIB, the upper layer shall set the symmetric key, outgoing frame counter, and optional external key sequence counter equal to the corresponding elements of the network security material descriptor in the <i>nwkSecurityMaterialSet</i> of the NIB referenced by the <i>nwkActiveKeySeqNumber</i> attribute of the NIB. The optional external frame counter shall not be used and the optional external key sequence counter shall correspond to the sequence number of the Network key.				
macDefaultSecuritySuite				Unsigned8
This is the unique identifier of the security suite to be used to protect communications between the MAC and devices not in the ACL as specified in the following table.				

Table 8-7: Security Suite Identifiers

Identifier	Security suite name	Security services			Sequential freshness (optional)
		Access control	Data encryption	Frame integrity	
0x00	None				
0x01	AES-CTR	X	X		X
0x02	AES-CCM-128	X	X	X	X
0x03	AES-CCM-64	X	X	X	X
0x04	AES-CCM-32	X	X	X	X
0x05	AES-CBC-MAC-128	X		X	
0x06	AES-CBC-MAC-64	X		X	
0x07	AES-CBC-MAC-32	X		X	

8.3.4.2 Network Layer

nwkSecurityLevel	Unsigned8
-------------------------	-----------

This attribute is an indication for the security level of the incoming and outgoing network frames. The security level identifier gives an indication of how an outgoing or incoming frame will be secured according to the table below. It indicates:

- Whether the payload is encrypted or not
- The extent of data authenticity over the frame (reflected by the message integrity code (MIC)). The length of the MIC may be 0, 32, 64 or 128 bits.

The security properties of the security levels are listed in the table below.

Table 8-8: Security levels available to the MAC, NWK and APS layers

Security level identifier	Security Level Sub-Field (Table 80)	Security Attributes	Data Encryption	Frame Integrity (length M of MIC, in number of octets)
0x00	'000'	None	OFF	NO (M = 0)
0x01	'001'	MIC-32	OFF	YES (M=4)
0x02	'010'	MIC-64	OFF	YES (M=8)
0x03	'011'	MIC-128	OFF	YES (M=16)
0x04	'100'	ENC	ON	NO (M = 0)
0x05	'101'	ENC-MIC-32	ON	YES (M=4)
0x06	'110'	ENC-MIC-64	ON	YES (M=8)
0x07	'111'	ENC-MIC-128	ON	YES (M=16)

nwkSecureAllFrames Boolean

Enabling or disabling this attribute will indicate if security shall be applied to incoming and outgoing NWK frames. If set to TRUE, security processing shall be applied to all incoming and outgoing frames. The exception is data frames destined for the current device that have the security sub-field of the frame control field set to FALSE (0). If this attribute has a value of 0x01 the NWK layer shall not relay frames that have the security sub-field of the frame control field disabled.

nwkSecurityMaterialSet STRUCTURE

This is the set of network security material descriptors, which are capable of maintaining an active and alternate Network key.

Table 8-9: Network Security Material Descriptors

Name	Type	Range	Description	Default
KeySeqNumber	Octet	0x00-0xFF	A sequence number assigned to a Network key by the trust center and used to distinguish Network keys for purposes of key updates, and incoming frame security operations.	00
OutgoingFrame-Counter	Ordered set of 4 octets	0x00000000-0xFFFFFFFF	Outgoing frame counter used for outgoing frames.	0x00000000
IncomingFrame-CounterSet	Set of incoming frame counter descriptor values. See Table 142.	Variable	Set of incoming frame counter values and corresponding device addresses.	Null set
Key	Ordered set of 16 octets	-	The actual value of the key.	-

nwkActiveKeySeqNumber Unsigned8

The sequence number of the active Network key in nwkSecurityMaterialSet.

nwkAllFresh Boolean

The status of this attribute indicates whether incoming NWK frames must be all checked for freshness when the memory for incoming frame counts is exceeded.

8.3.4.3 APS Layer

apsDeviceKeyPairSet STRUCTURE

This attribute is an indication of the set of key-pair descriptors containing master and link key pairs shared with other devices.

Table 8-10: Key-pair descriptor elements

Name	Type	Range	Description	Default
DeviceAddress	Device address	Any valid 64-bit address	Identifies the address of the entity with which this key-pair is shared.	-
MasterKey	Set of 16 octets	-	The actual value of the master key.	-
LinkKey	Set of 16 octets	-	The actual value of the link key.	-
OutgoingFrame-Counter	Set of 4 octets	0x00000000-0xFFFFFFFF	Unique identifier of the key originating with the device indicated by <i>KeySrcAddress</i> .	0x00000000
IncomingFrame-Counter	Set of 4 octets	0x00000000-0xFFFFFFFF	Incoming frame counter value corresponding to <i>DeviceAddress</i> .	0x00000000

apsSecurityTime-OutPeriod Integer

The period of time (in milliseconds) a device will wait for an expected security protocol frame.

apsTrustCenterAddress Device address

Identifies the address of the device's trust centre.

8.4 ZigBee Diagnosis Class

8.4.1 Object Overview

This section describes the information base attributes of ZigBee that can be used for monitoring and diagnostic purposes. These are parameters whose values can change during runtime and access to it could provide useful data for diagnosis.

The ZIGBEE DIAGNOSIS ASE is derived from the WIRELESS DIAGNOSIS ASE as shown in Figure 8-4.

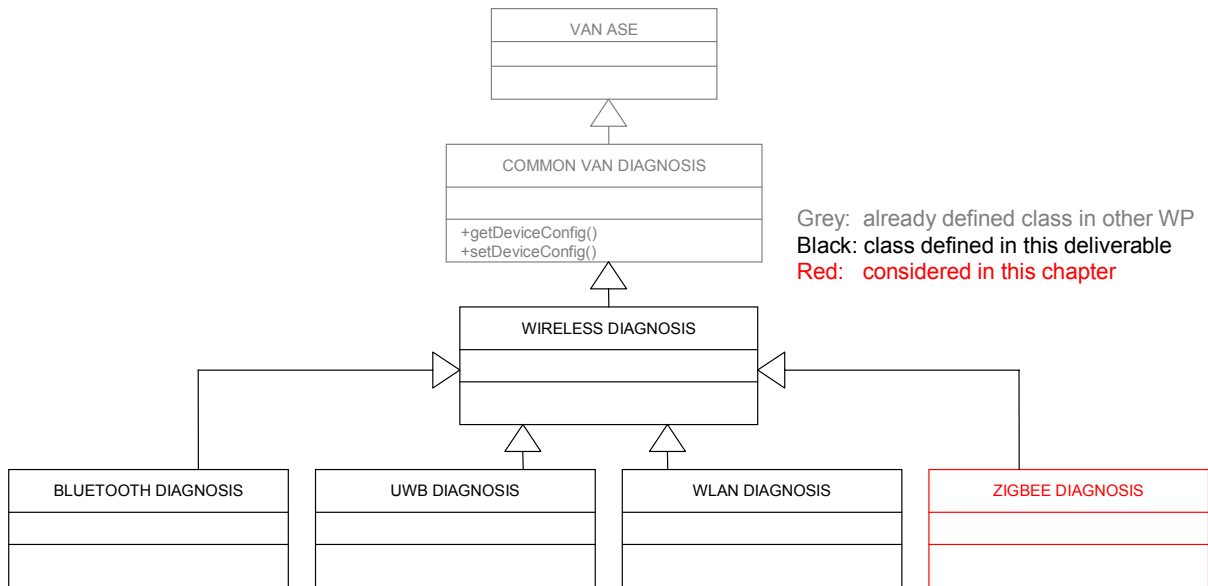


Figure 8-4: Deduction of the ZIGBEE DIAGNOSIS class structure

8.4.2 Refinement of Inherited Attributes

object-reference

String

This attribute is inherited from the VAN ASE. The content of the string in this class is "ZigBee".

8.4.3 Formal Model

VAN ASE: DIAGNOSIS ASE

CLASS: ZIGBEE DIAGNOSIS

CLASS ID:

PARENT CLASS: WIRELESS DIAGNOSIS

ATTRIBUTES:

- 1 (m) Attribute: macBeaconTxTime
- 2 (m) Attribute: nwkCapabilityInformation
- 3 (m) Attribute: nwkAvailableAddresses
- 4 (m) Attribute: nwkNeighborTable
- 5 (m) Attribute: apsAddressMap
- 6 (m) Attribute: apsBindingTable
- 7 (m) Attribute: CurrentPowerMode
- 8 (m) Attribute: CurrentPowerSource
- 9 (m) Attribute: CurrentPowerSourceLevel

8.4.4 Attribute Description

macBeaconTxTime

Unsigned32

This attribute is an indication of when the device transmitted its last beacon frame. It is indicated in symbol periods and measured at the same symbol boundary point for every transmitted beacon frame. This is a high precision value and a 32bit data type is therefore used.

nwkCapabilityInformation

Bit vector

This field shall contain the device capability information established at network joining time. The table below illustrates what the bits in this field indicate (taken from IEEE 802.15.4 specification):

Table 8-11: Capability Information bit-fields

Bit	Name	Description
0	Alternate PAN coordinator	This field will always have a value of 0 in implementations of this specification.
1	Device type	This field will have a value of 1 if the joining device is a ZigBee router and the JoinAsRouter parameter has a value of TRUE. It will have a value of 0 if the device is a ZigBee end device or else a router-capable device that is joining as an end device.
2	Power source	This field shall be set to the value of lowest-order bit of the PowerSource parameter passed to the NLME-JOINrequest primitive. The values are: 0x01 = Mains-powered device. 0x00 = other power source.
3	Receiver on when idle	This field shall be set to the value of the lowest-order bit of the RxOnWhenIdle parameter passed to the NLME-JOIN.request primitive. 0x01 = The receiver is enabled when the device is idle. 0x00 = The receiver may be disabled when the device is idle.
4 – 5	Reserved	This field will always have a value of 0 in implementations of this specification.
6	Security capability	This field shall be set to the value of lowest-order bit of the MACSecurity parameter passed to the NLME-JOINrequest primitive. The values are: 0x01 = MAC security enabled. 0x00 = MAC security disabled.
7	Allocate address	This field will always have a value of 1 in implementations of this specification, indicating that the joining device must be issued a 16-bit short address.

nwkAvailableAddresses

Unsigned16

The size of the remaining block of addresses that must be assigned. This value will be decremented by 1 every time an address is assigned. When this attribute has a value of 0, no more associations may be accepted.

nwkNeighborTable

STRUCTURE

The neighbour table is used for keeping track of devices neighbours in the network. There are different options for populating a neighbour table entry and this is left up to the implementer to decide on a suitable value. The value of nwkNeighborTable indicates the current set of neighbor table entries in the device. The following information is contained within the table:

- PAN id: The 16bit PAN identifier of the neighbouring device
- Extended address: Unique 64bit IEEE address present if the neighbour is a child or parent of the device
- Network address: The 16bit network address of the neighbouring device
- Device Type: This can be a coordinator, router or end device
- Relationship: The relationship between the neighbour and the current device

More information may additionally be included and depends on the actually vendors implementation of the ZigBee protocol stack.

apsAddressMap

STRUCTURE

The current set of 64 bit IEEE to 16 bit NWK address maps (see Table below). The exact value of apscMaxAddrMap-Entries is implementation specific.

Table 8-12: NWK Address Map

Entry Number	64bit IEEE address	16bit Network address
0x00 - apscMaxAddrMap-Entries	0x00000000 – 0xffffffff	0x0000 – 0xffff

apsBindingTable

STRUCTURE

The current set of binding table entries in the device. The binding table length is implementation specific.

CurrentPowerMode Unsigned8

The field CurrentPowerMode of the node power descriptor specifies the current sleep/power-saving mode of the node. The current power mode attribute can be one of the values.

Table 8-13: Current Power Mode Codes

Description	Current power mode value
Receiver synchronized with the receiver on when idle sub-field of the node descriptor.	0000
Receiver comes on periodically as defined by the node power descriptor.	0001
Receiver comes on when stimulated, e.g. by a user pressing a button.	0010

CurrentPowerSource BitString8

The field CurrentPowerSource of the node power descriptor specifies the current power source being utilized by the node. For the current power source selected, the corresponding bit of the current power source field shall be set to 1. All other bits shall be set to 0.

Table 8-14: Current Power Source Modes

Current power source	Bit field number
Constant (mains) power	0
Rechargeable battery	1
Disposable battery	2
Reserved	3

CurrentPowerSourceLevel Unsigned8

The field CurrentPowerSourceLevel of the node power descriptor specifies the level of charge of the power source. The current power source level field shall be set to one of the values listed below:

Table 8-15: Current Power Source Level Modes

Charge level	Binary value
Critical	0000
33%	0100
66%	1000
100%	1100
All other values	Reserved

9 Conclusions

Wireless communication is without a doubt an important part of future heterogeneous automation networks. It ensures the implementation of mobile, movable or flexible industrial automation applications. Therefore, it is essential to include these technologies into the overall approach of the Virtual Automation Network (VAN).

None of the discussed technologies has been developed for automation applications, however the IEEE 802.15.4/ZigBee specification has also industrial measurement and control in its scope. However, at least with Bluetooth and Wireless LAN industrial products are already on the market. This document provides the necessary descriptions in order to integrate products based on these technologies into the general configuration, plug and play, security and maintenance approaches of VAN. ZigBee is the only available higher layer specification for networks based on IEEE 802.15.4 today and was thus chosen to show how sensor network related attributes can be taken into account within the VAN concept. Furthermore, this document makes an effort in integrating UWB into the VAN architecture in the form of contributions of the pertinent ultra wideband specifications. into various VAN ASEs. These ASEs are mainly, device configuration, security configuration, diagnosis features, and relevant MAC parameters for the adaptation layer, which are specified with the aim of facilitating subsequent implementation of UWB design and test specifications in the chosen industrial environments.

This document is not only necessary for the future work in WP3. Other work packages which are not familiar with wireless technologies can also find here the required information to develop general VAN strategies. Examples are WP2 for plug and play, WP6 for security and WP8 for engineering.

The specification ensures that solutions of the chosen wireless technologies can be part of a Virtual Automation Network. This means these technologies can be planned, configured, commissioned, diagnosed and maintained by VAN methods in the same way as industrial Ethernet solutions or public networks which are also part of VAN.

Glossary

AES	Advanced Encryption Standard
ASE	Application Service Element
CCM	Cipher block Chaining Message authentication code
DME	Device Management Entity
ECMA	European Computer Manufacturers Association
FFI	Fixed-Frequency Interleaving
GPS	Global Positioning System
GTK	Group Temporal Key
HVAC	Heating Ventilation and Air Conditioning
IEEE	Institute of Electrical and Electronics Engineers
LCD	Liquid Crystal Display
MAC	Medium Access Control
MIB	Management Information Base
MIC	Message Integrity Code
MLME	MAC sub-Layer Management Entity
MPEG	Moving Picture Experts Group
MTU	Maximum Transmission Unit
OFDM	Orthogonal Frequency Division Multiplexing
PAN	Personal Area Network
PCS	Personal Communications Service
PHY	PHYsical layer
PRF	Pseudo Random Function
PTK	Pair-wise Temporal Key
TFI	Time-Frequency Interleaving
USB	Universal Serial Bus
UWB	Ultra Wide Band
VAN	Virtual Automation Network

References

- [D02.2-1] VAN Task Group 2.2, "Topology Architecture for the VAN virtual Automation Domain.", Report of Task 2.2 "Specification of the Open Platform for Automation Infrastructure" of Work Package 2 "Open Platform and System Architecture", 27.02.2006
- [D03.1-1] VAN Task Group 3.1, "Mapping of the state-of-the-art of utilization of wireless communication technologies in industries; Wireless closed loop control of dynamic systems; Report of status of GSM, GPRS and UMTS mobile technologies infrastructures and it's suitability for automation/industry purposes.", Report of Task 3.1 "Status and Analysis" of Work Package 3 "Wireless in Industries", 03.03.2006
- [ECMA368] High Rate Ultra Wideband PHY and MAC Standard, December 2005.