UNIT III

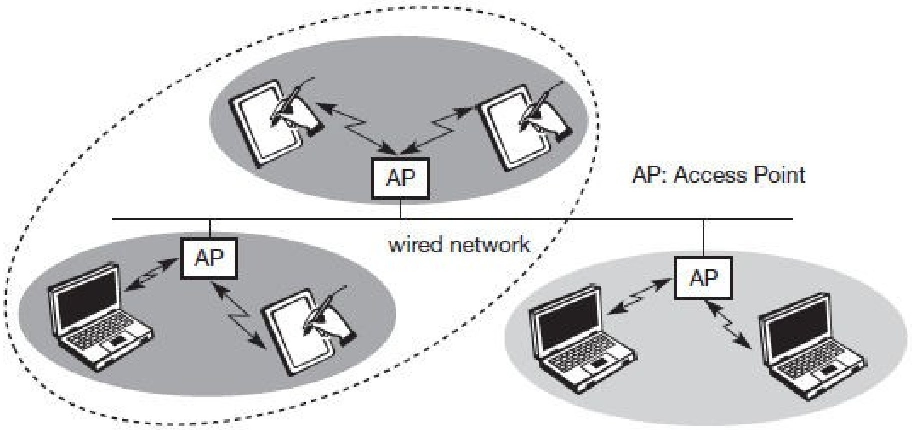
Some of the fundamental differences between wired networks & ad-hoc networks are:

 Asymmetric links: - Routing information collected for one direction is of no use for the other direction. Many routing algorithms for wired networks rely on a symmetric scenario.

 Redundant links: - In wired networks, some redundancy is present to survive link failures and this redundancy is controlled by a network administrator. In ad-hoc networks, nobody controls redundancy resulting in many redundant links up to the extreme of a complete meshed topology.

 Interference: - In wired networks, links exist only where a wire exists, and connections are planned by network administrators. But, in ad-hoc networks links come and go depending on transmission characteristics, one transmission might interfere with another and nodes might overhear the transmission of other nodes.

Figure 7.1
Example of three infrastructure-based wireless networks



 Dynamic topology: - The mobile nodes might move in an arbitrary manner or medium characteristics might change. This result in frequent changes in topology, so snapshots are valid only for a very short period of time. So, in ad-hoc networks, routing tables must somehow reflect these frequent changes in topology and routing algorithms have to be adopted.

BLUETOOTH

"Bluetooth" was the nickname of Harald Blåtland II, king of Denmark from 940 to 981, who united all of Denmark and part of Norway under his rule. **Bluetooth** is a proprietary open wireless technology standard for exchanging data over short distances (using short wavelength radio transmissions in the ISM band from 2400-2480 MHz) from fixed and mobile devices, creating personal area networks (PANs) with high levels of security. The Bluetooth technology aims at so-called **ad-hoc piconets**, which are local area networks with a very limited coverage and without the need for an infrastructure.

Bluetooth Features

- Bluetooth is wireless and automatic. You don't have to keep track of cables, connectors, and connections, and you don't need to do anything special to initiate communications. Devices find each other automatically and start conversing without user input, expect where authentication is required; for example, users must log in to use their email accounts.
- Bluetooth is inexpensive. Market analysts peg the cost to incorporate Bluetooth technology into a PDA, cell phone, or other product at a minimum cost.
- The ISM band that Bluetooth uses is regulated, but unlicensed. Governments have converged on a single standard, so it's possible to use the same devices virtually wherever you travel, and you don't need to obtain legal permission in advance to begin using the technology.
- Bluetooth handles both data and voice. Its ability to handle both kinds of transmissions simultaneously makes possible such innovations as a mobile hands-free headset for voice with applications that print to fax, and that synchronize the address books on your PDA, your laptop, and your cell phone.
- Signals are omni-directional and can pass through walls and briefcases. Communicating devices don't need to be aligned and don't need an unobstructed line of sight like infrared.
- Bluetooth uses frequency hopping. Its spread spectrum approach greatly reduces the risk that communications will be intercepted.

Bluetooth Applications

- File transfer.
- Ad-hoc networking: Communicating devices can spontaneously form a community of networks that persists only as long as it's needed
- Device synchronization: Seamless connectivity among PDAs, computers, and mobile phones allows applications to update information on multiple devices automatically when data on any one device changes.
- Peripheral connectivity.
- Car kits: Hands-free packages enable users to access phones and other devices without taking their hands off the steering wheel
- Mobile payments: Your Bluetooth-enabled phone can communicate with a Bluetooth-enabled vending machine to buy a can of Diet Pepsi, and put the charge on your phone bill.

The 802.11b protocol is designed to connect relatively large devices with lots of power and speed, such as desktops and laptops, where devices communicate at up to 11 Mbit/sec, at greater distances (up to 300 feet, or 100 meters). By contrast, Bluetooth is designed to connect small devices like

PDAs, mobile phones, and peripherals at slower speeds (1 Mbit/sec), within a shorter range (30 feet, or 10 meters), which reduces power requirements. Another major difference is that 802.11b wasn't designed for voice communications, while any Bluetooth connection can support both data and voice communications.

User scenarios

Many different user scenarios can be imagined for wireless piconets or WPANs:

Connection of peripheral devices: Today, most devices are connected to a desktop computer via wires (e.g., keyboard, mouse, joystick, headset, speakers). This type of connection has several disadvantages: each device has its own type of cable, different plugs are needed, wires block office space. In a wireless network, no wires are needed for data transmission. However, batteries now have to replace the power supply, as the wires not only transfer data but also supply the peripheral devices with power.

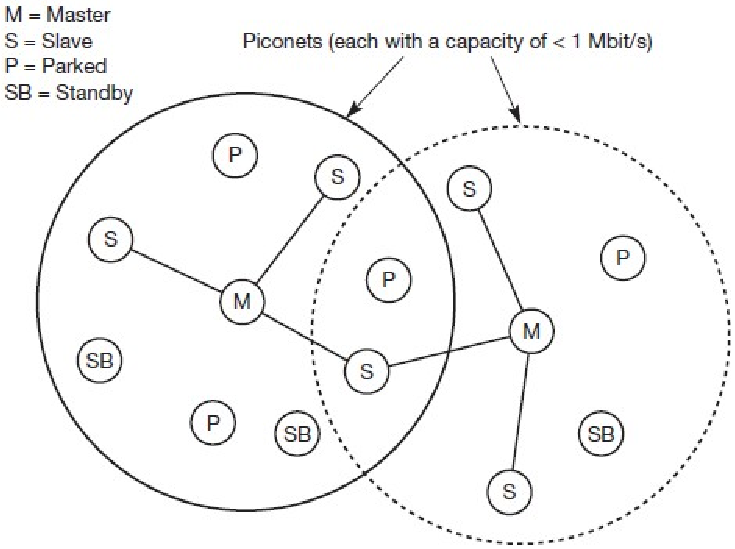
Support of ad-hoc networking: Imagine several people coming together, discussing issues, exchanging data (schedules, sales figures etc.). For instance, students might join a lecture, with the teacher distributing data to their personal digital assistants (PDAs). Wireless networks can support this type of interaction; small devices might not have WLAN adapters following the IEEE 802.11 standard, but cheaper Bluetooth chips built in.

Bridging of networks: Using wireless piconets, a mobile phone can be connected to a PDA or laptop in a simple way. Mobile phones will not have full WLAN adapters built in, but could have a Bluetooth chip. The mobile phone can then act as a bridge between the local piconet and, e.g., the global GSM network.

Networking in Bluetooth

Bluetooth operates on 79 channels in the 2.4 GHz band with 1 MHz carrier spacing. Each device performs frequency hopping with 1,600 hops/s in a pseudo random fashion. A piconet is a collection of Bluetooth devices which are synchronized to the same hopping sequence. One device in the piconet can act as **master** (M), all other devices connected to the master must act as **slaves** (S). The master determines the hopping pattern in the piconet and the slaves have to synchronize to this pattern. Each piconet has a unique hopping pattern. If a device wants to participate it has to synchronize to this. A typical piconet is shown below:

Figure 7.43
Bluetooth scatternet



Parked devices (P) can not actively participate in the piconet (i.e., they do not have a connection), but are known and can be reactivated within some milliseconds. Devices in stand-by (SB) do not participate in the piconet. Each piconet has exactly one master and up to seven simultaneous slaves. More than 200 devices can be parked. The first step in forming a piconet involves a master sending its clock and device ID. All the Bluetooth devices have the same capability to become a master or a slave and two or three devices are sufficient to form a piconet. The unit establishing the piconet automatically becomes the master, all other devices will be slaves. The hopping pattern is determined by the device ID, a 48-bit worldwide unique identifier.

The phase in the hopping pattern is determined by the master's clock. After adjusting the internal clock according to the master a device may participate in the piconet. All active devices are assigned a 3-bit **active member address** (AMA). All parked devices use an 8-bit **parked member address** (PMA). Devices in stand-by do not need an address.

A device in one piconet can communicate to another device in another piconet, forming a **scatternet**. A master in one piconet may be a slave in another piconet. Both piconets use a different hopping sequence, always determined by the master of the piconet. Bluetooth applies **FH-CDMA** for separation of piconets. A collision occurs if two or more piconets use the same carrier frequency at the same time. This will probably happen as the hopping sequences are not coordinated. If a device wants to participate in more than one piconet, it has to synchronize to the hopping sequence of the piconet it wants to take part in. If a device acts as slave in one piconet, it simply starts to synchronize with the hopping sequence of the piconet it wants to join. After synchronization, it acts as a slave in this piconet and no longer participates in its former piconet. To enable synchronization, a slave has to know the identity of the master that determines the hopping sequence of a piconet.

Before leaving one piconet, a slave informs the current master that it will be unavailable for a certain amount of time. The remaining devices in the piconet continue to communicate as usual.

Bluetooth Protocol Stack

The Bluetooth protocol stack can be divided into a **core specification**, which describes the protocols from physical layer to the data link control together with management functions, and **profile specifications** describing many protocols and functions needed to adapt the wireless Bluetooth technology to legacy and new applications.

A high-level view of the architecture is shown. The responsibilities of the layers in this stack are as follows:

□ *The radio layer* is the physical wireless connection. To avoid interference with other devices that communicate in the ISM band, the modulation is based on fast frequency hopping. Bluetooth divides the 2.4 GHz frequency band into 79 channels 1 MHz apart (from 2.402 to 2.480 GHz), and uses this spread spectrum to hop from one channel to another, up to 1600 times a second. The standard wavelength range is 10 cm to 10 m, and can be extended to 100 m by increasing transmission power.

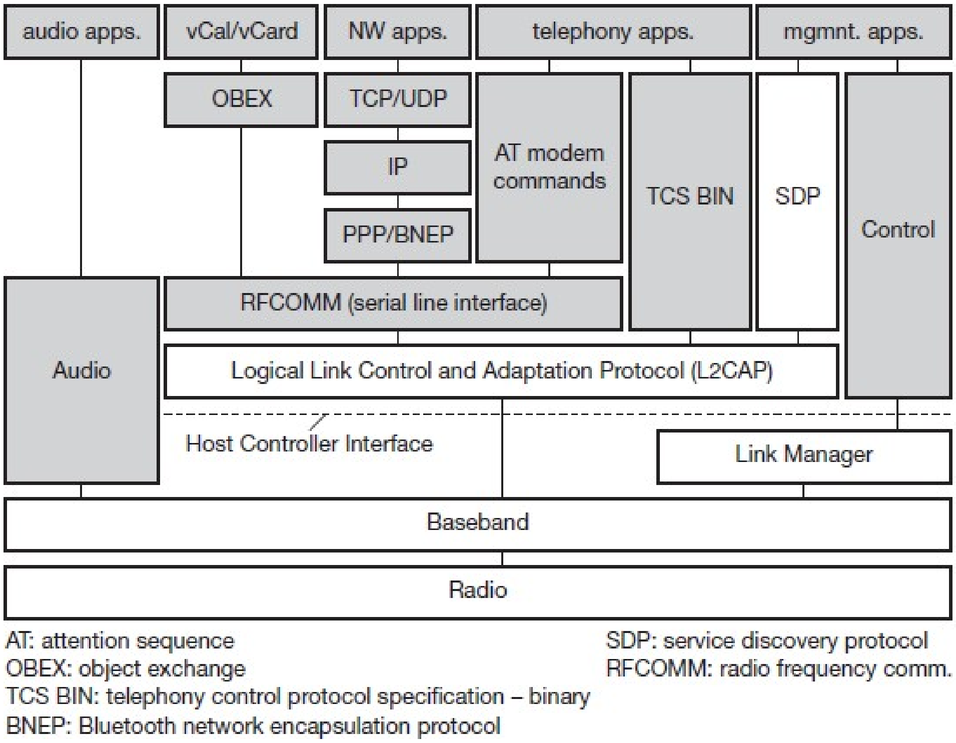

audio apps.
vCal/vCard
NW apps.
telephony apps.
mgmnt. apps.
OBEX
TCP/UDP
IP
PPP/BNEP
AT modem commands
TCS BIN
SDP
Control
Audio
RFCOMM (serial line interface)
Logical Link Control and Adaptation Protocol (L2CAP)
Host Controller Interface
Link Manager
Baseband
Radio
AT: attention sequence
OBEX: object exchange
TCS BIN: telephony control protocol specification – binary
BNEP: Bluetooth network encapsulation protocol
SDP: service discovery protocol
RFCOMM: radio frequency comm.

Figure 7.44
Bluetooth protocol stack

Bluetooth Protocol Stack

□ *The baseband layer* is responsible for controlling and sending data packets over the radio link. It provides transmission channels for both data and voice. The baseband layer maintains Synchronous Connection-Oriented (SCO) links for voice and Asynchronous Connectionless (ACL) links for data. SCO packets are never retransmitted but ACL packets are, to ensure data integrity.

SCO links are point-to-point symmetric connections, where time slots are reserved to guarantee timely transmission. A slave device is allowed to respond during the time slot immediately following an SCO transmission from the master. A master can support up to three SCO links to a single slave or to multiple slaves, and a single slave can support up to two SCO links to different slaves. Data transmissions on ACL links, on the other hand, are established on a per-slot basis (using slots not reserved for SCO links). ACL links support point-to-multipoint transmissions. After an ACL transmission from the master, only a slave addressed specifically may respond during the next time slot; if no device is addressed, the message is treated as a broadcast.

- *The Link Manager Protocol (LMP)* uses the links set up by the baseband to establish connections and manage piconets. Responsibilities of the LMP also include authentication and security services, and monitoring of service quality.

- *The Host Controller Interface (HCI)* is the dividing line between software and hardware. The L2CAP and layers above it are currently implemented in software, and the LMP and lower layers are in hardware. The HCI is the driver interface for the physical bus that connects these two components. The HCI may not be required. The L2CAP may be accessed directly by the application, or through certain support protocols provided to ease the burden on application programmers.

- *The Logical Link Control and Adaptation Protocol (L2CAP)* receives application data and adapts it to the Bluetooth format. Quality of Service (QoS) parameters are exchanged at this layer.

Link Manager Protocol

The link manager protocol (LMP) manages various aspects of the radio link between a master and a slave and the current parameter setting of the devices. LMP enhances baseband functionality, but higher layers can still directly access the baseband. The following groups of functions are covered by the LMP:

- **Authentication, pairing, and encryption**: Although basic authentication is handled in the baseband, LMP has to control the exchange of random numbers and signed responses. LMP is not directly involved in the encryption process, but sets the encryption mode (no encryption, point-to-point, or broadcast), key size, and random speed.

- **Synchronization:** Precise synchronization is of major importance within a Bluetooth network. The clock offset is updated each time a packet is received from the master.

- **Capability negotiation:** Not only the version of the LMP can be exchanged but also information about the supported features. Not all Bluetooth devices will support all features that are described in the standard, so devices have to agree the usage of, e.g., multi-slot packets, encryption, SCO links, voice encoding, park/sniff/hold mode, HV2/HV3 packets etc.

- **Quality of service negotiation:**

Different parameters control the QoS of a Bluetooth device at these lower layers. The poll interval, i.e., the maximum time between transmissions from a master to a particular slave, controls the latency and transfer capacity. A master can also limit the number of slots available for slaves' answers to increase its own bandwidth.

□ **Power control:** A Bluetooth device can measure the received signal strength. Depending on this signal level the device can direct the sender of the measured signal to increase or decrease its transmit power.

□ **Link supervision:** LMP has to control the activity of a link, it may set up new SCO links, or it may declare the failure of a link.

□ **State and transmission mode change:** Devices might switch the master/slave role, detach themselves from a connection, or change the operating mode

L2CAP

The logical link control and adaptation protocol (L2CAP) is a data link control protocol on top of the baseband layer offering logical channels between Bluetooth devices with QoS properties. L2CAP is available for ACLs only.

L2CAP provides three different types of logical channels that are transported via the ACL between master and slave:

□ *Connectionless*: These unidirectional channels are typically used for broadcasts from a master to its slave(s).

□ *Connection-oriented*: Each channel of this type is bi-directional and supports QoS flow specifications for each direction. These flow specs follow RFC 1363 and define average/peak data rate, maximum burst size, latency, and jitter.

□ *Signaling:* This third type of logical channel is used to exchanging signaling messages between L2CAP entities.

Each channel can be identified by its **channel identifier (CID)**. Signaling channels always use a CID value of 1, a CID value of 2 is reserved for connectionless channels. For connection-oriented channels a unique CID (>= 64) is dynamically assigned at each end of the channel to identify the connection.

The following figure shows the three packet types belonging to the three logical channel types.

The **length** field indicates the length of the payload (plus PSM for connectionless PDUs). The **CID** has the multiplexing/demultiplexing function. For connectionless PDUs a **protocol/service multiplexor (PSM)** field is needed to identify the higher layer recipient for the payload. For connection-oriented PDUs the CID already fulfills this function. Several PSM values have been defined, e.g., 1 (SDP), 3 (RFCOMM), 5 (TCS-BIN). Values above 4096 can be assigned dynamically. The payload of the signaling PDU contains one or more **commands**. Each command has its own **code** (e.g., for command reject, connection request, disconnection response etc.) and an **ID** that matches a request with its reply. The **length** field indicates the length of the **data** field for this command.

Besides protocol multiplexing, flow specification, and group management, the L2CAP layer also provides segmentation and reassembly functions. Depending on the baseband capabilities, large packets have to be chopped into smaller segments.

Security

The main security features offered by Bluetooth include a challenge response routine for authentication, a stream cipher for encryption, and a session key generation. Each connection may require a one-way, two-way, or no authentication using the challenge-response routine. The security algorithms use the public identity of a device, a secret private user key, and an internally generated random key as input parameters. For each transaction, a new random number is generated on the Bluetooth chip. Key management is left to higher layer software. The following figure shows several steps in the security architecture of Bluetooth.

The first step, called **pairing**, is necessary if two Bluetooth devices have never met before. To set up trust between the two devices a user can enter a secret PIN into both devices. This PIN can have a length of up to 16 byte. Based on the PIN, the device address, and random numbers, several keys can be computed which can be used as link key for **authentication**. The authentication is a challenge-response process based on the link key, a random number generated by a verifier (the device that requests authentication), and the device address of the claimat (the device that is authenticated

Based on the link key, and again a random number an encryption key is generated during the **encryption** stage of the security architecture. This key has a maximum size of 128 bits and can be individually generated for each transmission. Based on the encryption key, the device address and the current clock a payload key is generated for ciphering user data. The payload key is a stream of pseudo-random bits. The **ciphering** process is a simple XOR of the user data and the payload key.

All Bluetooth-enabled devices must implement the Generic Access Profile, which contains all the Bluetooth protocols and possible devices. This profile defines a security model that includes three security modes:

□ *Mode 1* is an insecure mode of operation. No security procedures are initiated.

□ *Mode 2* is known as *service-level enforced security*. When devices operate in this mode, no security procedures are initiated before the channel is established. This mode enables applications to have different access policies and run them in parallel.

□ *Mode 3* is known as *link-level enforced security*. In this mode, security procedures are initiated before link setup is complete.

Though Bluetooth offers a better security than WER in 802.11, it has several limitations. The PIN's are often fixed and some keys are permanently stored on the devices. The quality of the random number generators has not been specified.

SDP

To find new services available in the radio proximity, Bluetooth defined the **service discovery protocol (SDP)**. SDP defines only the discovery of services, not their usage. Discovered services

can be cached and gradual discovery is possible. All the information an SDP server has about a service is contained in a **service record**. This consists of a list of service attributes and is identified by a 32-bit service record handle.
A service attribute consists of an attribute ID and an attribute value. The 16-bit attribute ID distinguishes each service attribute from other service attributes within a service record. The attribute ID also identifies the semantics of the associated attribute value. The attribute value can be an integer, a UUID (universally unique identifier), a string, a Boolean, a URL (uniform resource locator) etc.

HiperLAN (High Performance Radio LAN) is a Wireless LAN standard It is a European alternative for the IEEE 802.11standards (the IEEE is an international organization). It is defined by the European Telecommunications Standards Institute(ETSI). In ETSI the standards are defined by the BRAN project (Broadband Radio Access Networks). The HiperLAN standard family has four different versions.

Planning for the first version of the standard, called HiperLAN/1, started 1991, when planning of 802.11 was already going on. The goal of the HiperLAN was the high data rate, higher than 802.11. The standard was approved in 1996. The functional specification is EN300652, the rest is in ETS300836.

The standard covers the Physical layer and the Media Access Control part of the Data link layer like 802.11. There is a new sublayer called Channel Access and Control sublayer (CAC). This sublayer deals with the access requests to the channels. The accomplishing of the request is dependent on the usage of the channel and the priority of the request.

CAC layer provides hierarchical independence with Elimination-Yield Non-Preemptive Multiple Access mechanism (EY-NPMA). EY-NPMA codes priority choices and other functions into one variable length radio pulse preceding the packet data. EY-NPMA enables the network to function with few collisions even though there would be a large number of users.Multimedia applications work in HiperLAN because of EY-NPMA priority mechanism. MAC layer defines protocols for routing, security and power saving and provides naturally data transfer to the upper layers.

On the physical layer FSK and GMSK modulations are used in HiperLAN/1.

HiperLAN features:

- range 50 m
- slow mobility (1.4 m/s)
- supports asynchronous and synchronous traffic
- Bit rate - 23.2 Mbit/s
- Description- Wireless Ethernet
- Frequency range- 5 GHz

HiperLAN does not conflict with microwave and other kitchen appliances, which are on 2.4 GHz. An innovative feature of HIPERLAN 1, which may other wireless networks do not offer, is its ability to forward data packets using several relays. Relays can extend the communication on the MAC layer beyond the radio range. For power conservation, a node may set up a specific wake up pattern. This pattern determines at what time the node is ready to receive, so that at other times, the node can turn off its receiver and save energy. These nodes are called p-savers and need so called p-supporters that contain information about wake up patterns of all the p-savers they are responsible for. A p-supporter only forwards data to a p-saver at the moment p-saver is awake. This action also requires buffering mechanisms for packets on p-supporting forwaders.

HiperLAN/2

HiperLAN/2 functional specification was accomplished February 2000. Version 2 is designed as a fast wireless connection for many kinds of networks. Those are UMTS back bone network, ATM and IP networks. Also it works as a network at home like HiperLAN/1. HiperLAN/2 uses the 5 GHz band and up to 54 Mbit/s data rate.

The physical layer of HiperLAN/2 is very similar to IEEE 802.11a wireless local area networks. However, the media access control (the multiple access protocol) is Dynamic TDMA in HiperLAN/2, while CSMA/CA is used in 802.11a/n.

Basic services in HiperLAN/2 are data, sound, and video transmission. The emphasis is in the quality of these services (QoS)

The standard covers Physical, Data Link Control and Convergence layers. Convergence layer takes care of service dependent functionality between DLC and Network layer (OSI 3). Convergence sublayers can be used also on the physical layer to connect IP, ATM or UMTS networks. This feature makes HiperLAN/2 suitable for the wireless connection of various networks.

On the physical layer BPSK, QPSK, 16QAM or 64QAM modulations are used.

HiperLAN/2 offers security measures. The data are secured with DES or Triple DES algorithms. The wireless access pointand the wireless terminal can authenticate each other.

The present document, the term "HIPERLAN" is used to refer to HIPERLAN, Type 1.

A HIPERLAN is a Radio Local Area Network (RLAN) in which all nodes communicate using a single shared communication channel. A HIPERLAN has the following properties:

- it provides a service that is compatible with the ISO MAC service definition in ISO/IEC 15 802-1

- its operations are compatible with the ISO MAC bridges specification in ISO/IEC 10 038 for interconnection with other LANs;

- it may be deployed in a pre-arranged or an ad-hoc fashion;

- it supports node mobility;

- it may have a coverage beyond the radio range limitation of a single node;

- it supports both asynchronous and time-bounded communication by means of a

Channel Access Mechanism (CAM) with priorities providing hierarchical independence of performance;

- its nodes may attempt to conserve power in communication by arranging when they need to be active for reception.

The HIPERLAN MAC service:

- is based on, and therefore is compatible with, the ISO MAC service definition;

- defines the communication service over a single HIPERLAN;

- allows the timing requirements of the MSDU transfer to be specified; and

- allows exploration of available HIPERLANs for dynamic HIPERLAN access.

The HIPERLAN CAC service:

- defines the communication service over a single shared communication channel;

- allows the channel access priority requirements of the HCSDU transfer to be specified; and

- frees the HCS-user from the concerns of the characteristics peculiar to any particular communication channel.

The HIPERLAN MAC protocol:

- provides the HIPERLAN MAC service;

- specifies the behaviour of a HM-entity in a given HIPERLAN;

- is compatible with the ISO MAC bridges specification in ISO/IEC 10 038 [8]; and

- uses the HIPERLAN CAC service.

The HIPERLAN CAC protocol:

- provides the HIPERLAN CAC service;

- specifies, for a particular set of one or more shared radio channels, the appropriate hierarchically independent

channel access mechanism used by a HC-entity in a given HIPERLAN; and

- uses the transmission and reception facilities specified by the HIPERLAN physical layer.

HIPERLAN addressing

The HIPERLAN addressing requirements are elaborated in the following subclauses.

MAC Service Access Point (MSAP) addressing

In order to be compatible with the ISO MAC service definition, the HIPERLAN MAC service uses the 48-bit LAN

MAC address for MSAP identification.

A HIPERLAN MAC entity (HM-entity) shall be attached to a single MSAP, through which the HM-entity provides the HIPERLAN MAC service to a single HMS-user; and it shall be attached toa single HIPERLAN CAC Service Access Point (HCSAP), through which the HM-entity uses the HIPERLAN CAC service provided by the HIPERLAN CAC service provider (HCS-provider).

An individual 48-bit LAN MAC address is used, as an individual-MSAP-address, to identify a single MSAP and its attached HMS-user and HM-entity. On the other hand, a group 48-bit LAN MAC address is used, as a group-MSAP-address, to identify a group of MSAPs and their attached HMS-users. Individual-MSAP-address and group-MSAP-address assignment is outside the scope of the present document and is governed by other relevant LAN standards. HCSAP addressing

A HIPERLAN CAC entity (HC-entity) shall be attached to a single HCSAP, through which the HC-entity provides the HIPERLAN CAC service to a single HCS-user. As a result, a HC-entity is attached to a single HM-entity. For practical reasons, a HM-entity's attached HCSAP shall also be identified by the same individual 48-bit LAN MAC address assigned to its attached MSAP. Therefore, an individual 48-bit LAN MAC address is inherited, as an individual-HCSAP-address, toidentify a single HCSAP and its attached HCS-user and HC-entity. A group 48-bit LAN MAC address is then used, as a group-HCSAP-address, to identify a group of HCSAPs and their attached HCS-users. The group-HCSAP-address assignment from the entire group 48-bit LAN MAC addressspace is independent of the group-MSAP-address assignment.