UNIT 7

MOBILE DEVICES, SERVER AND MANAGEMENT, WIRELESS LAN, MOBILE INTERNET CONNECTIVITY AND PERSONAL AREA NETWORK

UNIT – 7: SYLLABUS

MOBILE DEVICES, SERVER AND MANAGEMENT, WIRELESS LAN, MOBILE INTERNET CONNECTIVITY AND PERSONAL AREA NETWORK:

Mobile agent, Application Server, Gateways, Portals, Service Discovery, Device Management, Mobile File Systems.

Wireless LAN (Wi-Fi) Architecture and Protocol Layers, WAP 1.1 and WAP 2.0 Architectures, Bluetooth – enabled Devices Network, ZigBee. **8 Hours**

7.1 Mobile agent

A mobile agent consists of software and data, which can move from one computing system to another autonomously and functions for a device or system the host. A mobile agent can also be described as an autonomous software which runs on a host with some data and dynamically moves to another host as and when required. Mobile-agent-based architecture provides for mobility of codes and data.

The various characteristics of a mobile agent are:

- Mobility of code and data from one computing system (host) to another
- Ability to learn in order to adapt code and data to a host computing system
- Ability to clone, extend, or dispose itself after its role is over
- Compatibility to the hosts
- Ability to continuously and autonomously process requests and send responses and alerts (an alert is an unsolicited message, record, or information.)

Some of the advantages of a mobile agent are:

- Asynchronous running of codes on diversified heterogeneous hosts
- Reduced computational and data requirements on devices with limited resources
- Tolerance to connection failures
- Only the agent source (e.g., device middleware, which sends the agent) needs to be modified in order to redefine the functions expected from the agent.

Below fig shows a mobile-agent-based architecture, in which the agent moves at instants T1, T2, and T3 to process a request, get mail, and get records from database respectively. When a mobile agent moves at instant T1, T2 or T3, it saves its own state at the host and transmits this saved state to the next host in order to resume execution of the codes starting from the saved state.

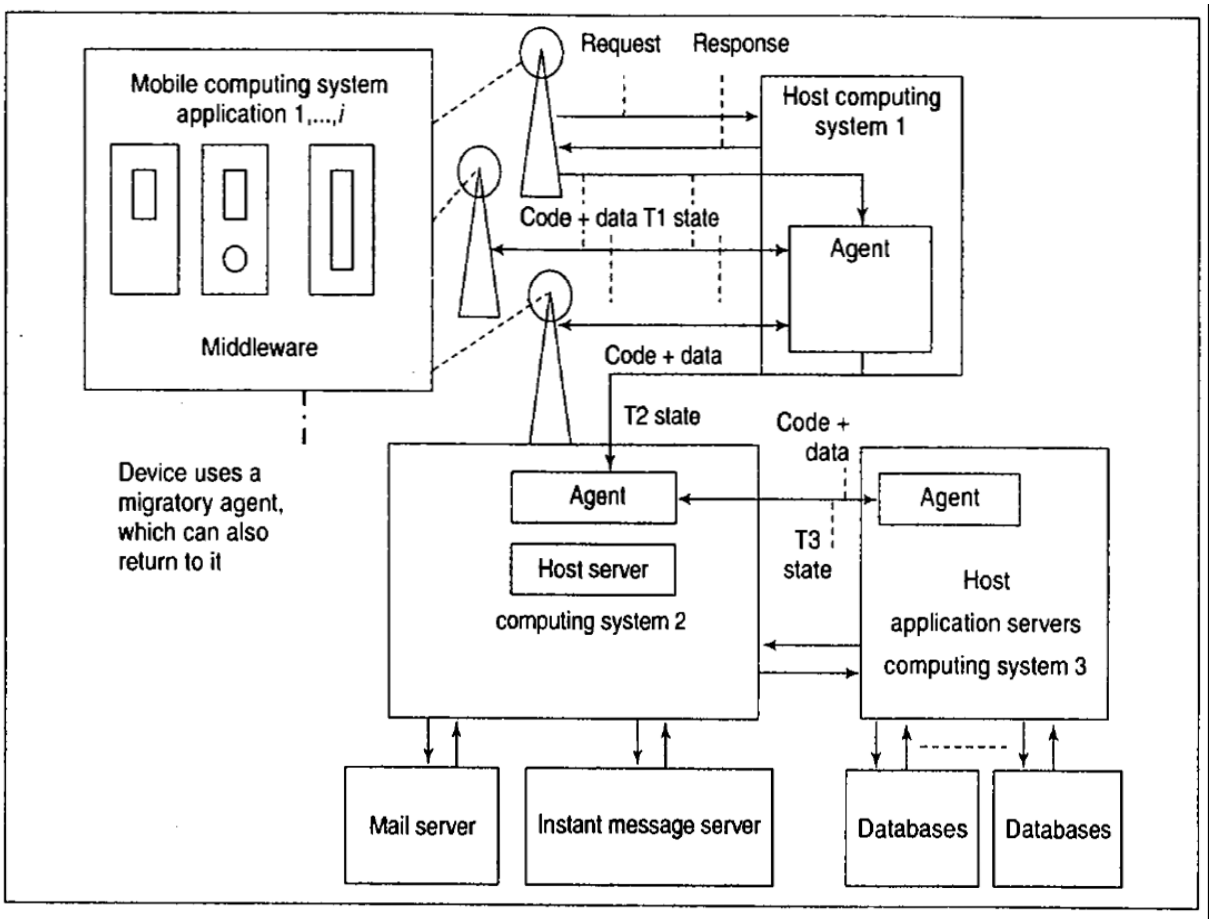

Fig. 10.2 Mobile-agent-based architecture

A mobile agent is a powerful tool for distributed applications and retrieval of remote host information. Advantages when deploying an agent in a computing system are as follows:

- ✓ It provides application visualisation environment (i.e., the system hardware runs at a host OS and host system hardware of the mobile agent of that system) and allows automatic thread migration (ATM), the threads running on the host being independent of the OS.
- ✓ An agent can send the request to a computing system as well as generate responses for request from the system.
- ✓ The connection protocol and the connecting network between the host and source are immaterial.
- ✓ There is no need of a centralized or an application-specific server.

Issues in use of the agent are as follows:

- ✓ An agent may have strong or weak mobility.
- ✓ An agent possesses migration latencies (waiting period in migrating from one host to another) and collaboration latency (waiting period in start of collaboration between the application server and the service-requesting system).
- ✓ There can be security-specific issues related to the agent.
- ✓ There can be environment- and platform-specific difficulties in implementing adaptability and compatibility at diversified hosts.

7.2 Application Server

- Application Server is a software, which is executed on a server and serves the application-level logic of the business functions.
- Application-level logic means the logic commands or instructions which an application server uses for sending and receiving the logic results from a computing system.
- The term 'business functions' indicates the logical way in which transactions are carried out between server at one end and application at the other.
- Application server gets requests from all the collaborating or independent mobile devices of an enterprise or from a distributed mobile computing system. It processes the requests and generates responses.

Below fig shows an application-server-based N-tier architecture (N>=3)

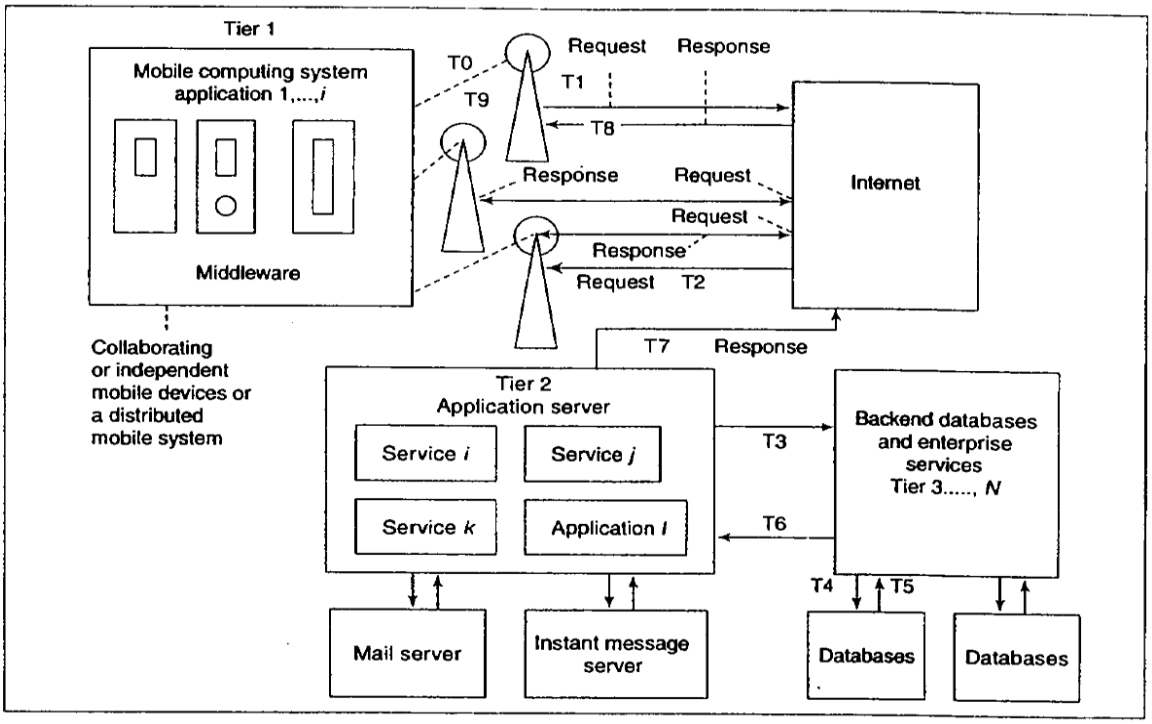

Fig. 10.3 Application-server-based *N*-tier architecture (*N* = 3) with requests processed at the application server using backend databases and system

Assume that there are j clients which can request to the server, A client 1,….., or j sends the request from the collaborating or independent mobile devices of an enterprise or from a distributed mobile computing system. The fig shows the stages the data transmission requests at instants from T0 to T4. Requests are processed through tier 1 –N. Responses are sent from the backend system to tier 1 at instant from T5 to T9.

Consider the application server at Tier 2. It provides the following services:

- Service *i*: application logic processing at the server
- Service *j*: presentation services for device responses and decoding the device requests (e.g., presentation service of a middleware application server for universal device access)
- Service *k*: transaction services with support to pervasive computing model of mobile applications
- Application *l*: system integration service for backend services and database at tiers 3, …, *N*.

Consider the examples of web database and enterprise application servers at tiers 3, ..., *N*:

1. IBM DB2 database server—IBM DB2 is an RDBMS (Relational Database Management System) data server from IBM and its versions run on handheld devices and in enterprises the application logic processing is at the server.
2. Oracle 9i database server—RDBMS Oracle9i server has a large number of features and it supports XML documents and has an option for cluster database.

3. Enterprise Server – A server that connects to enterprise centralised data server in an enterprise. The centralised server holds most of the information about the enterprise and it responds to the requests that are only from authenticated and authorised clients.

Some example for application servers are as follows:

- Web Generic application servers for Java-based web applications (Microsoft, Sun, and Netscape) with additional support for wireless network and mobile devices
- IBM WebSphere application server with specialized mobile web computing application server (it supports J2EE web applications and XML databases)
- IBM Domino application server for workgroups, email applications, and support for handheld and Windows CE devices
- Microsoft Mobile Information Server (e.g., for messenger and email)
- Oracle 9i application server for database services with mobile support
- Puma and Synchrologic iMobile Suite for data-synchronization services
- Nokia WAP (Wireless Application Protocol) Server for wireless Internet WAP applications
- Funambol (earlier known as Sync4J) (Section 9.4) has provisions for mobile application server for PIM (personal information manager, for example, for the push email, address book, and calendar), open source DB Connector, and data synchronization services

➢ **Sun Java System Web Server 6**

- It is meant for large business applications and is compatible with a number of operating systems. It allows deployment of CGI, PHP, ColdFusion, Servlets, JSPs and ASPs.
- The server provides application services and runtime environment. Application services include email service, security, file system, and session management. An access manager provides a secure access to web-based resources.
- The server has a content engine which is a software layer for publishing the Web application response. Content engine is a program which uses the content sources at a server and drives them to the destination.
- The engine has three components – search engine, content management engine and HTTP engine.
- The content engine is a layer for web content creation, addition to existing contents, and storage of content page using the HTML, JSPs and ASPs. It maintains integrity of the client contents with the server.

➢ **IBM WebSphere MQe**

- IBM WebSphere application server runs on a mobile device. Application accelerator accelerates the running of an application so that results are obtained faster. MQe (Messaging and Queuing Everyplace) refers to an added set of features.

- MQe assures secure messaging, decouples from the application, and uses a queue manager. MQe also has IBM WebSphere Voice Server, Text-to-speech (TTS) software and IBM WebSphere RFID Premises Server.
- MQe development kit is used for writing applications for messaging and queuing for mobile devices.
- WebSphere MQe development kit writes applications based on Java and C.
- There are portal, transcoding, and personalization servers.
- Deployment and integration are the two other functions of the WebSphere.

Below figure gives an overview of IBM WebSphere application server for web applications on mobile devices with the feature of MQe.

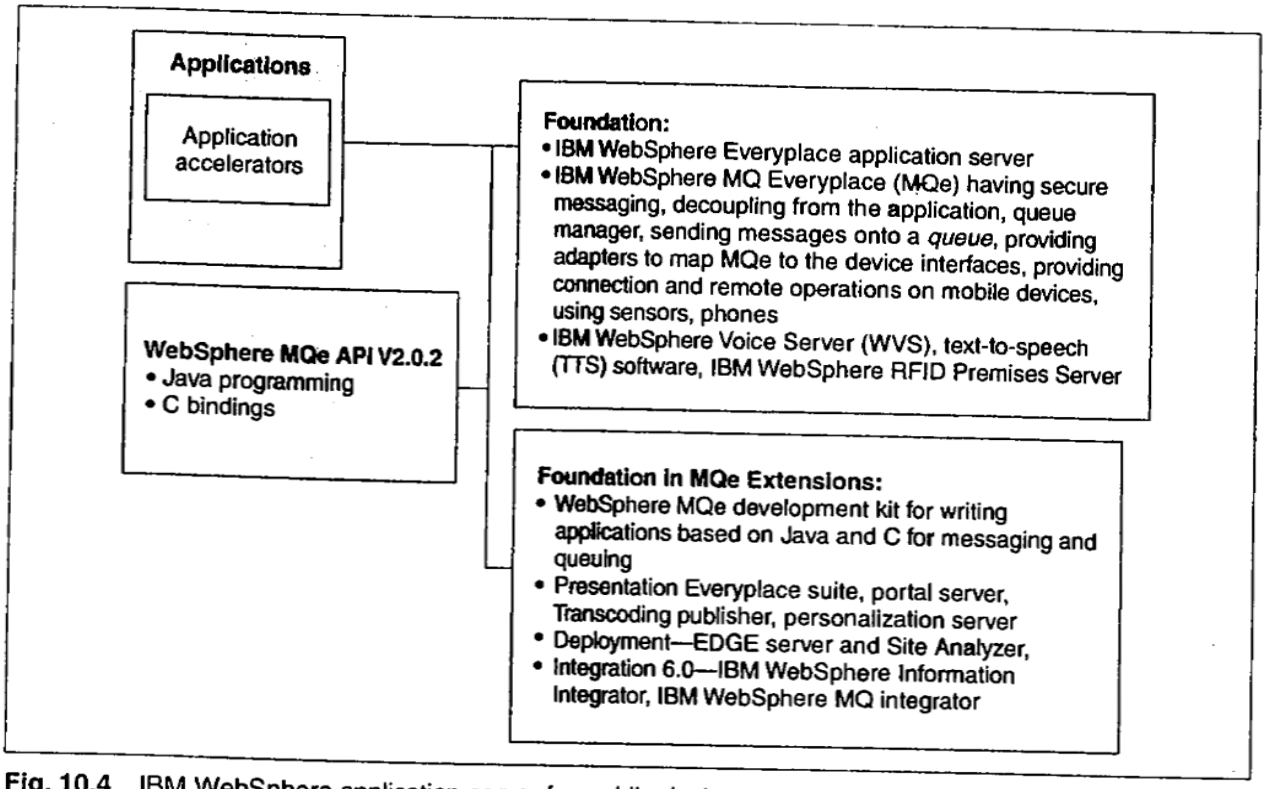

Fig. 10.4 IBM WebSphere application server for mobile devices

- The IBM WebSphere Everyplace has three basic services -- *configuring services, directory services* and *networking protocols*. A basic service of this software is configuring of the server and operations. Directory services deploys JNDS (Java naming and directory service) and LDAP (lightweight directory access protocol). IBM WebSphere Everyplace also has an HTTP server which functions as website server. Other services of IBM WebSphere Everyplace are as follows:
  - ✓ Device management and management of subscriptions for the applications.
  - ✓ Provides a synchronization engine which synchronises the data available with a device having the records at the server.
  - ✓ Provides for device caching.
  - ✓ Enables load balancing.

Below fig gives an overview of IBM WebSphere Everyplace services and functions.

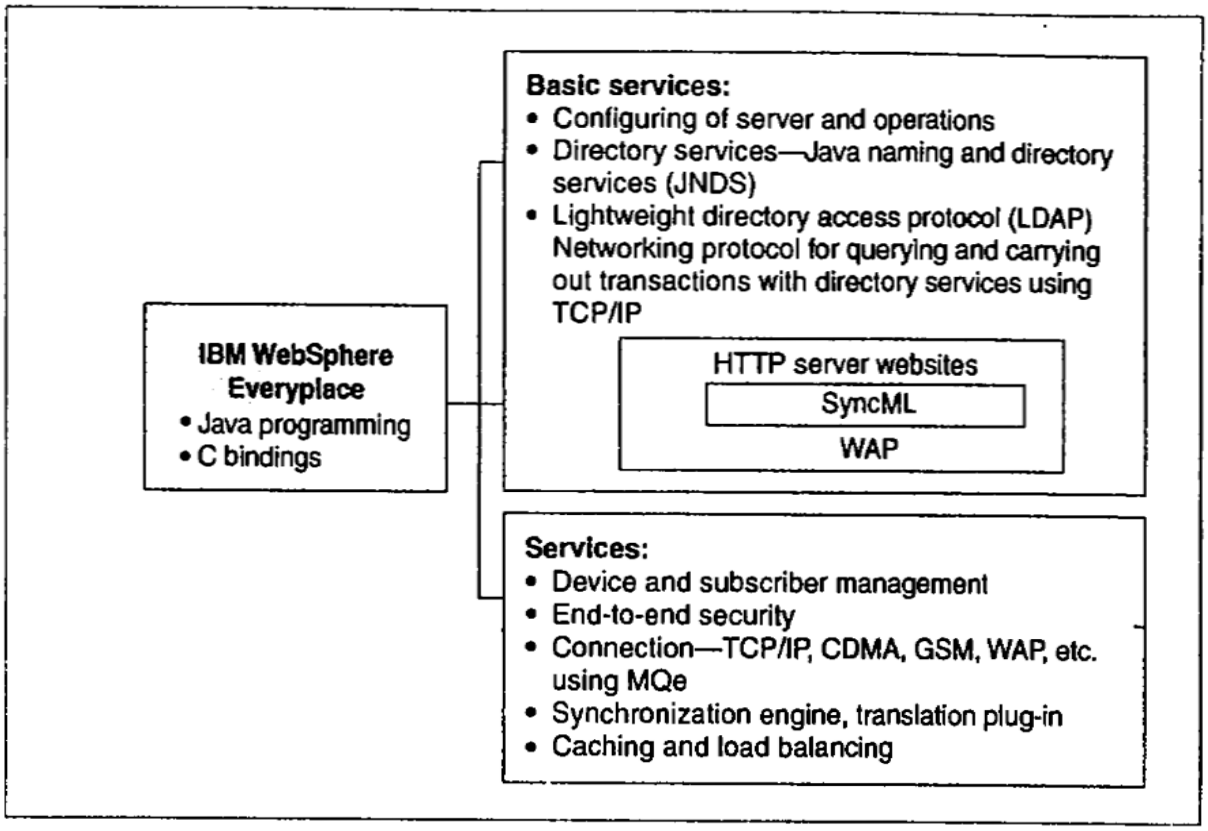

Fig. 10.5 Services and functions of IBM WebSphere Everyplace

- **Oracle Application Server**
  - Oracle is the most popular RDBMS. Application server (AS) enables web-based transactions with the databases. Oracle Application server 10g is the latest version where g stands for grid Computing feature.
  - A brief description of Oracle Application Server 9i (9iAS) are as follows:
    - ✓ 9iAS is used at Windows as well as UNIX platforms. It includes the Oracle XML parser and OracleJSP.
    - ✓ 9iAS has two components – HTTP server and Z9iAS portal with OC4J (Oracle AS containers for J2EE 1.4 specification EJBs).
    - ✓ The HTTP server responds to the SQL from the clients. A client sends requests and gets responses through an in between web cache called 9iAS Enterprise Web Cache.
    - ✓ The HTTP server interfaces to 9iAS portal. The portal has a parallel page engine (PPE) which enables the retrieving of pages in parallel at the portal.
    - ✓ 9iAS has OC4J J2EE EJBs in a container.
    - ✓ 9iAS integrates business logic, management, security, e-business and portal functions.

7.3 Gateways

Gateways are nothing but the one which connects the two networks, each using different protocols in its network layers. The three different types of gateways are:

- **Protocol Conversion Gateway**
  - Connection gateway is a software which connects two application layers using two different protocols. It connects the application server and client application.

Below fig shows the WAP or IBM secureWay Wireless Gateway with 3 tier architecture.

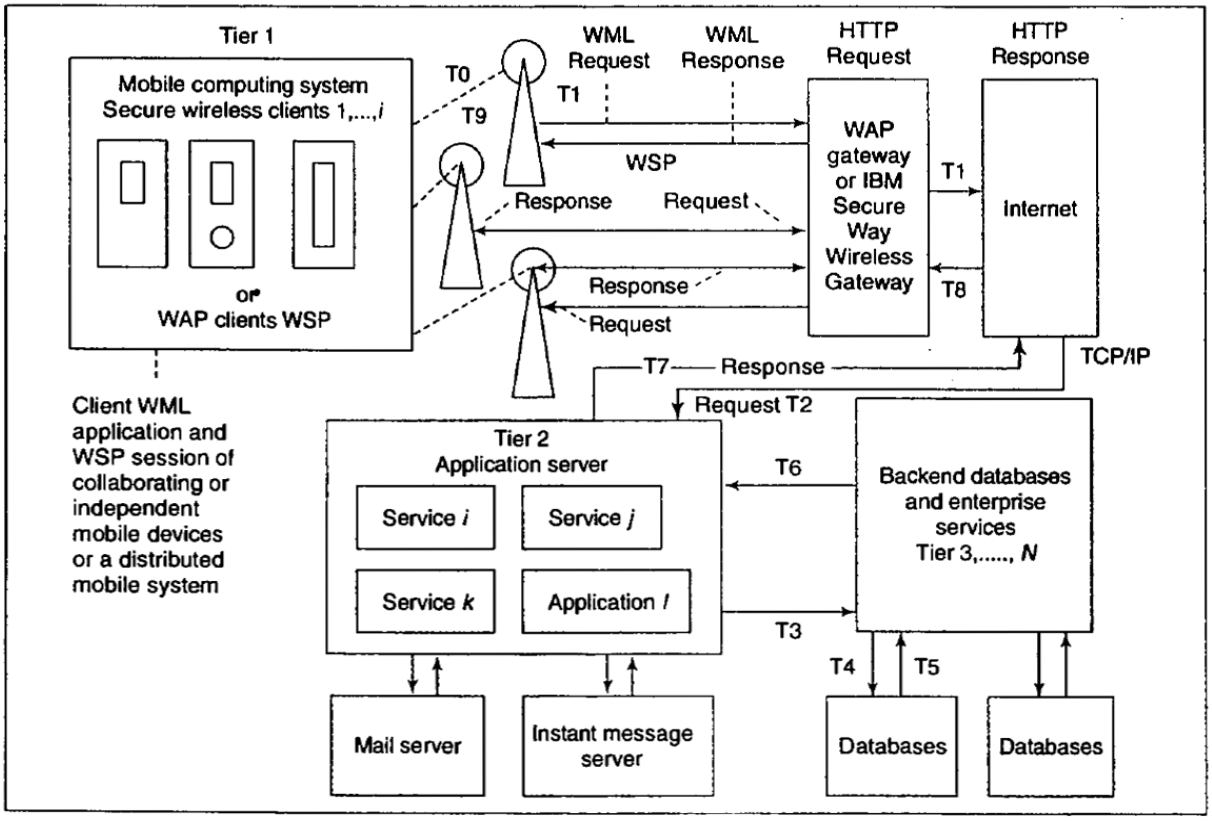

Fig. 10.6 WAP or IBM SecureWay Wireless Gateway

- Tier 1 (client) consists of a client using WML (wireless markup language). The client sends the application service request and obtains the responses. A WSP (Wireless session protocol) establishes the session between the application server (AS) at the Internet and the clients. A client sends requests for the service required by collaborating or independent mobile devices or a mobile computing system.
- Tier 2 (Application server) connects to tier 1 through Internet. The Internet uses TCP/IP while the wireless devices use WAP or the other protocols. When the protocol used is WAP, a gateway which connects tier 1 to Internet which has the AS is called WAP gateway. IBM SecureWay Wireless gateway is a WAP gateway which connects these two tiers.

- When Funambol is used for developing the mobile computing applications, the connector objects provide the gateways. The gateways connect to the file systems, databases, email systems and applications. It is used for the connections and transcoding between the tiers.

➢ **Transcoding Gateway or Proxy**

- A gateway is called transcoding gateway when it adapts its responses to the content format of the client device and its requests to the format required by the application running on the application server.
- Transcoding proxy has conversion, computational, and analyzing capabilities, while gateway has conversion and computational capabilities only. A proxy can execute itself on the client system or application server.
- Transcoding applications involve formats, data, and code conversion from one end to another when the multimedia data is transferred from a server to the mobile TV, Internet TV, WAP phone and Smartphone as the client devices.
- Transcoding involves filtering, compression, decompression, scaling, mapping and colour adaptation.
- Transcoding facilitates TTS (text-to-speech) and STT(speech-to-text) conversion.
- It is also helps in converting XML style sheets to WML or HTML.

➢ **Residential Gateway**

- Cable TV, setup box, personal computer, laptop, digital camera, iPod, home theatre, iPhone or mobile phone and Bluetooth enabled devices forms the home network.
- Outside world typically consists of Internet, enterprise server, backend servers, and application servers. The outside network has high speed Internet.
- A residential gateway is a wireless device which provides connectivity through service provider. The network also connects to application server and backend server through the Internet.
- Residential gateway is a gateway between a network of home devices and the outside world. Below fig shows the residential gateway architecture.

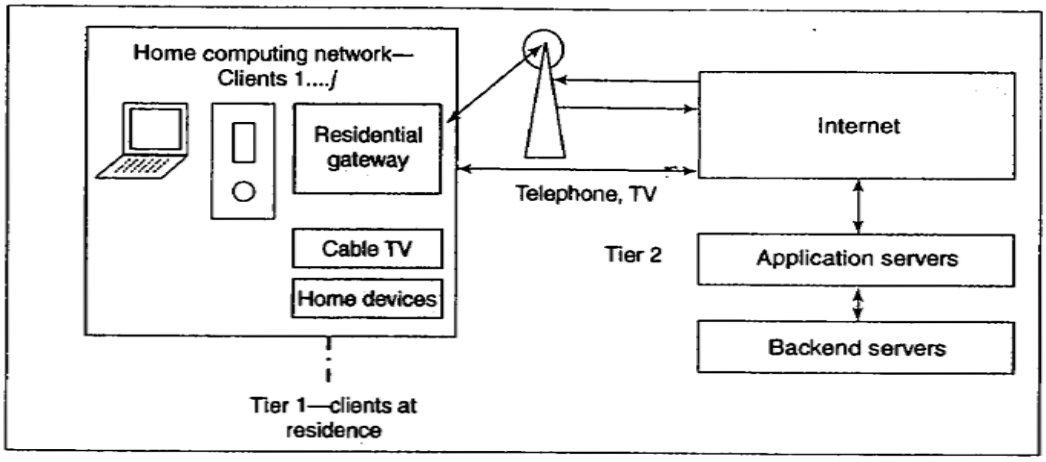

Fig. 10.7 Residential gateway architecture

7.4 Portals

- A web Portal is a website that provides application-specific programs and personalized capabilities like business transactions to client devices or computing systems.
- The portal has portlets (application-specific small portals). Example: mail(POP3) portlet.
- The portal provides services available from a number of websites. It enables them using diversified middleware. It provides services to PCs as well as mobile devices.
- IBM WebSphere portal is one portal which provides a software architecture for the application and business transactions. It has scalability of any size organisation. This software has an extension for voice-based applications.

Below figure shows the portal-based client-server architecture. A portal includes an authentication server, a content aggregator, and APIs for the services.

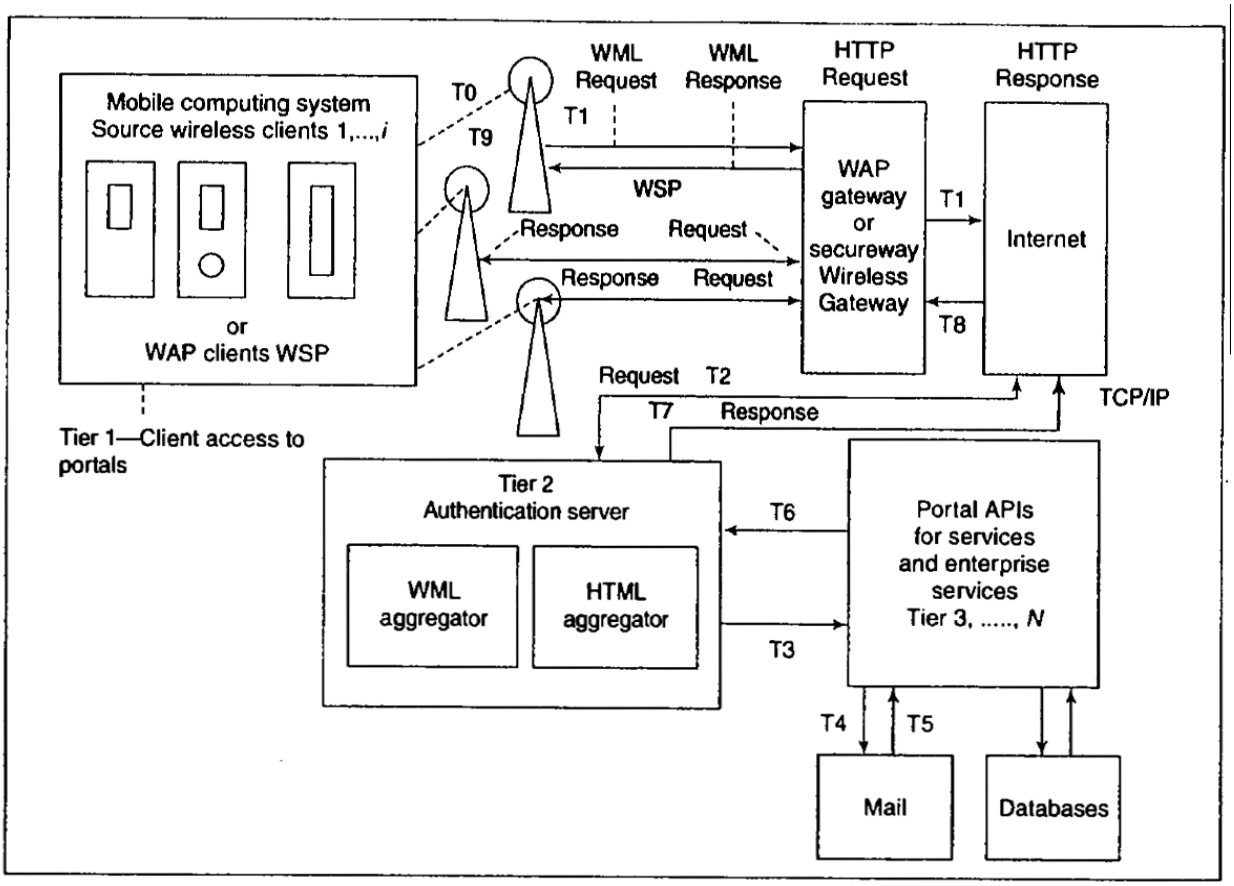

Fig. 10.8 Portal based client–server architecture for mobile clients

7.5 Service Discovery

Service discovery is an adaptable middleware in a device that dynamically discovers services by carrying out the following steps in chronological order:

1. Letting nearby service network recognize that device.
2. Letting the nearby network know of device services.
3. Searching and discovering new services at the network.
4. Interacting with nearby network using discovered services.

- *Self-administration* software means software for starting the operating system, allocating network and system access addresses, initiating accesses, establishing and terminating the connections, and making secure connections.
- *Self-configuration* means establishing and modifying the route information for the connections by a system on its own.
- *Self-healing* network means that the network can establish an alternative route when a connection or enroute node breaks.
- Service discovery middleware has features of self-healing and self-configuring.
- *Bluetooth, Jini, SLP (service location protocol),* and *UPnP (Universal Plug and Play)* have functions of service discovery.

➢ **Jini**

- ✓ Jini enables the programming for the distributed computing system environment.
- ✓ It not only provisions for service discovery but also for the lookup for the databases, RMIs, and the joining of APIs and programs of a device with those of the other devices discovered using the lookups.
- ✓ Jini includes JavaSpaces Technology, which is simple and powerful high-level tool. Its purpose is to share network-based object spaces.
- ✓ Jini includes extensible remote invocation (Jini ERI). It enables programming and dynamic computing in a device and provides a platform to create adaptable, scalable, evolvable, and flexible network-centric services.

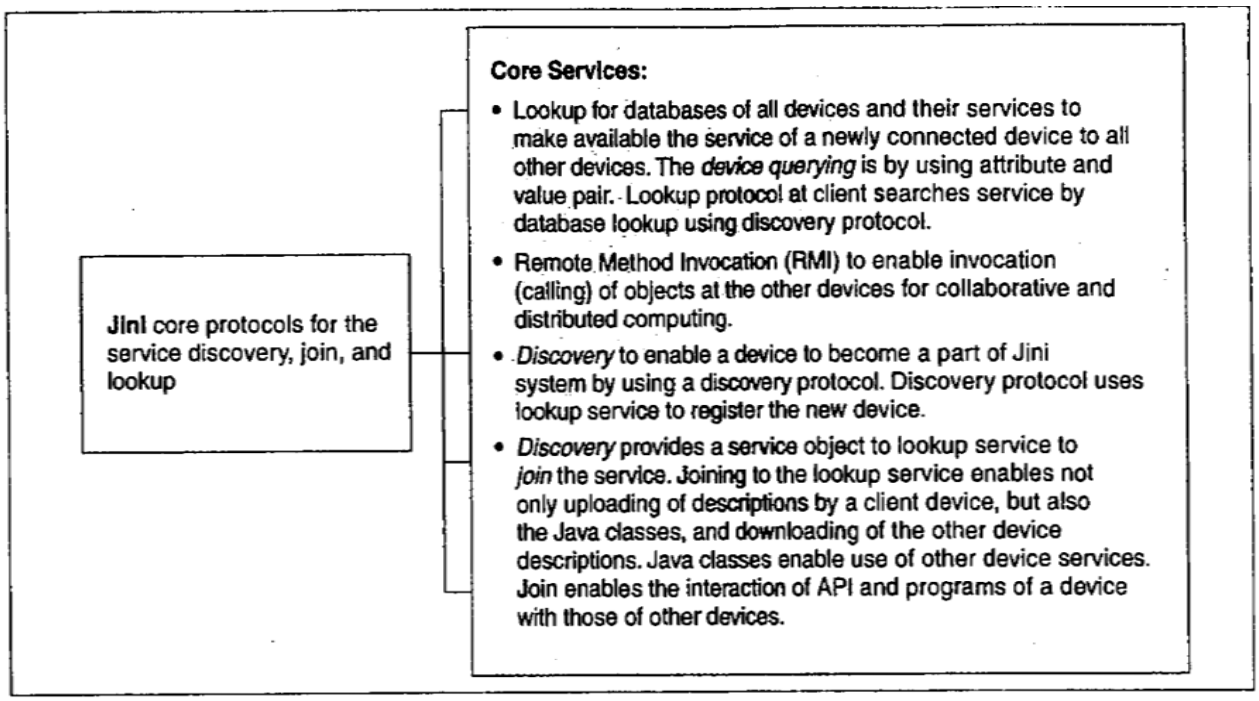

Fig. 10.9 Jini core protocols for the service discovery, join, and lookup

- **Service Location Protocol (SLP)**
  - ✓ A client device using SLP dynamically discovers a service. SLP stack consists of SLP header and SLP service URL. This URL enables other devices to use the service of the client device and provides the location of the device in the network.
  - ✓ SLP is an alternative to lookup service database in Jini or a service catalogue. SLP differs from Jini as it just provides the lookup for the URLs and description of other devices during discovery phase and it does not provide for database lookup,join, and spaces as Jini.

Unicasting, Multicasting and Advertisement:

- ✓ A unicast using TCP/IP or any other SDP means that the service discovery is directly by connecting to the lookup service.
- ✓ A multicast using a protocol means that the service discovery is by broadcast of description of the service to a group of devices.
- ✓ An advertisement of service or description enables a device to need not for discovering a service.

- **UPnP**
  - ✓ Universal Plug and Play (UPnP) is a Microsoft solution for service discovery, service, and device descriptions, use of control points, event notifications etc.
  - ✓ UPnP provides for control points. A control point is a registry for any device establishing connection with the network or disconnecting from the network.
  - ✓ UPnP includes description. UPnP clients are called control points. Control point automatically discovers UPnP server on the network.
  - ✓ Each description has a URL which control provides to other devices requiring the service.

✓ UPnP devices do the multicasting of messages and services. Discovery by new device is from the messages and after discovery for its services at other devices.

Below fig shows the UPnP core protocols for the service discovery, description, control and eventing.

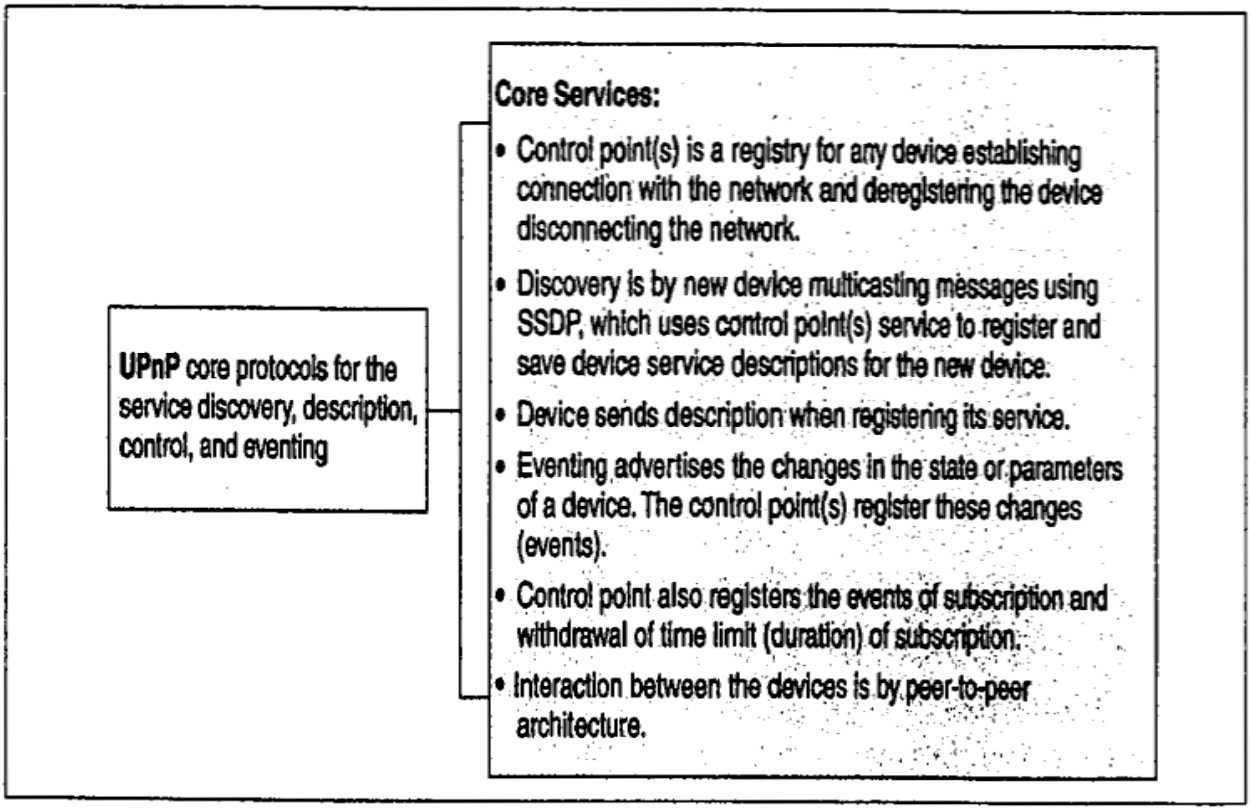

Fig. 10.10 UPnP core protocols for the service discovery, description, control, and eventing

7.6 Device Management

- Device management means *configuring at initialisation, monitoring current configuration, processing maintenance requests* and *taking care of location and handover* of each device.
- Device is self-administered. It boots up, starts the operating system, initiates accesses, establishes and terminates the connections, and makes secure connections.
- Devices has the feature of self-healing and self-configuring network.

➢ **Device Support Infrastructure**

✓ Tivoli DSI (Device support Infrastructure) is an IBM software. It is used for ATMs, handheld devices, set-up boxes and cable modems..

✓ A device Gateway has a device management agent to connect devices at one end with the gateway at the other end. The Gateway includes Tivoli management Gateway. It connects to device management server of the service provider.

- ✓ Device manager at the device management server assigns a unique ID to a device and a local ID to the device which is supported by the support infrastructure.
- ✓ Device unique ID remains fixed and is assigned once. Local ID can be reassigned when the device moves from one personal area network to another.

Below fig shows the Tivoli DSI architecture.

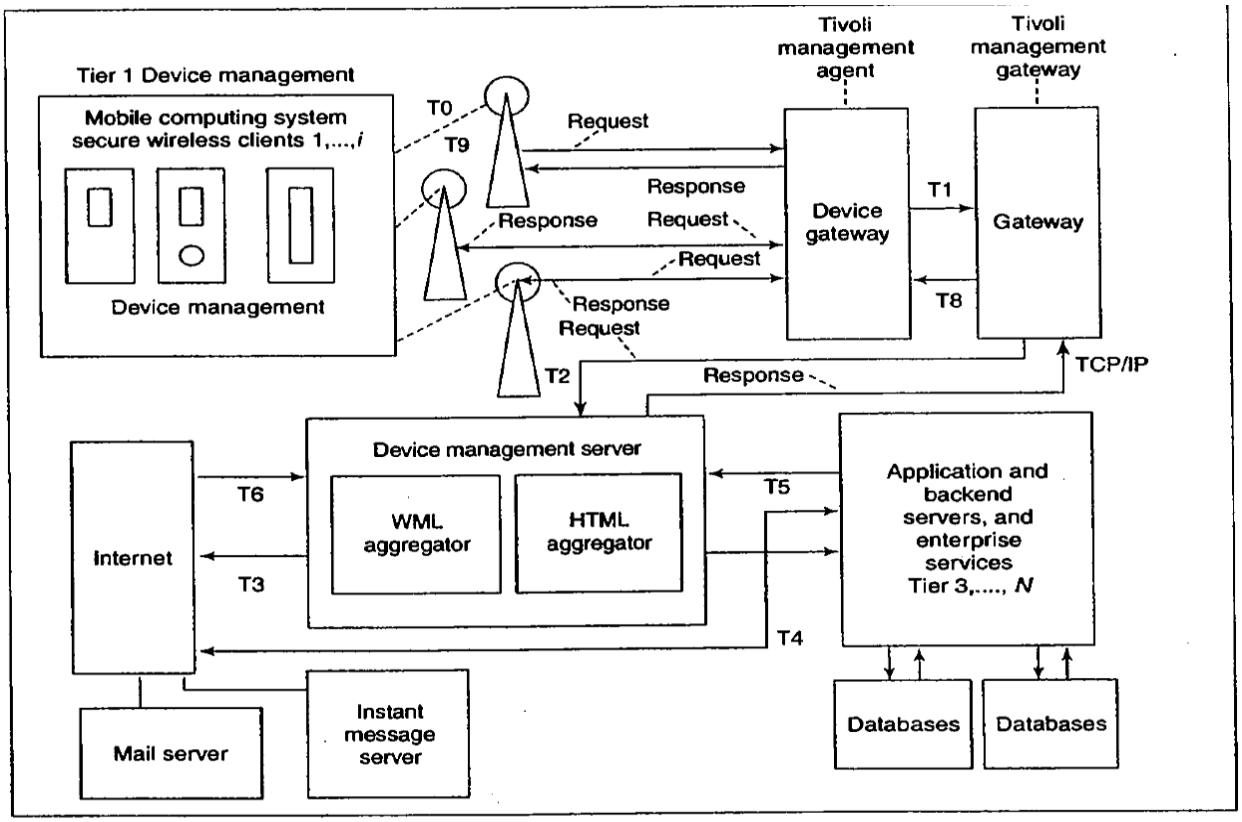

Fig. 10.11 IBM Tivoli device support infrastructure architecture

- **User, Device, and Network Profiles**
  - ✓ Profiles provide a specification for the use of software such as Device manager or Device management server. Device management requires profiles for the user, device and network.
  - ✓ Mobile Information device profile (MIDP) provides a specification for the mobile devices.
  - ✓ User profile is used as follows – Device management is facilitated by system to access user profile which consists of password and ID. A user can add PIM data, security credentials to the profile. Device profile includes unique ID, local ID, individual preferences and available resources. Network profile specifies the current location address of the device and networked devices and the description of the network services.

- **Directory Service**
  - ✓ Directory service means a service protocol which specifies and provisions for the set of operations with the given objects or entries in a directory.
  - ✓ Directory is an efficient way of storing and accessing data, It has a tree-like structure with entries at the tree-leaves and nodes representing the printers, documents, persons, organizational units etc. The tree has a root with number of nodes. A root object is a parent object which has the number of child objects.
  - ✓ An object is accessed by its name and attributes.
  - ✓ LDAP is an open source networking protocol for accessing, modifying and querying TCP/IP directory services.

- **OMA DM (Open Mobile Alliance Device Management)**
  - ✓ OMA DM objects are most used standard in mobile device computing system. DM defines a description framework and has hierarchical structure in which there is a management object tree.
  - ✓ OMA DM has one way synchronization.
  - ✓ DM is based on SyncML Data Synchronisation specifications. Below fig shows the SyncML DM stack.

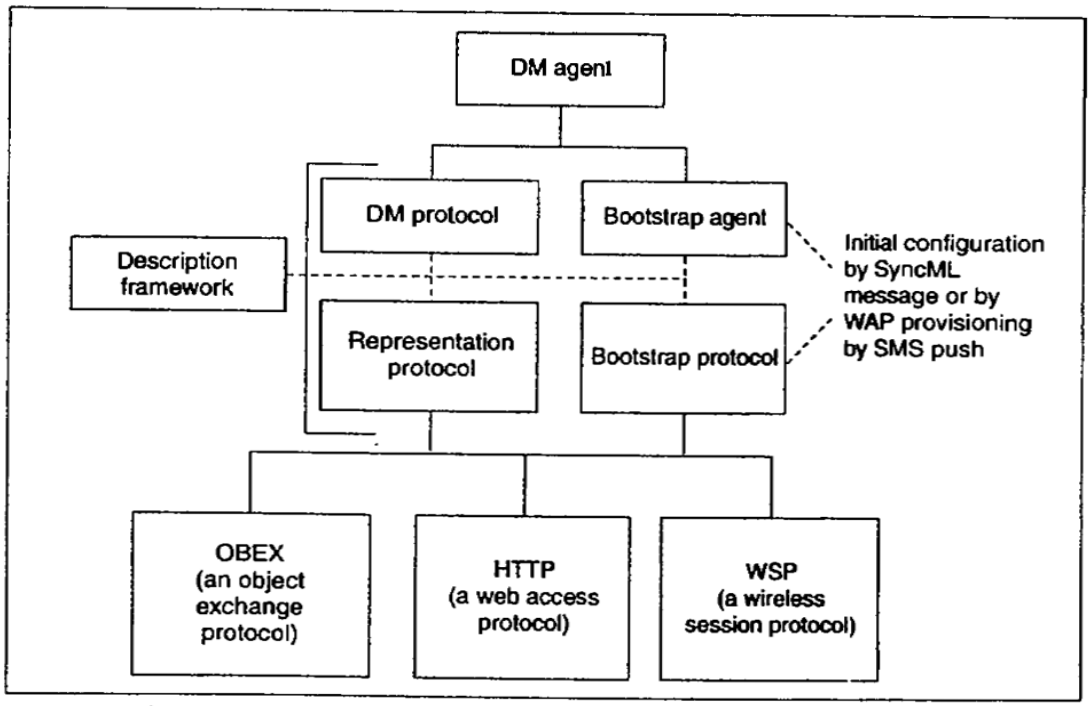

Fig. 10.12 SyncML DM stack

  - ✓ Another example of parent node of a management object for data synchronisation is DSAcc (Data Synchronisation account). Its functions and specifications are as mentioned below:

1. *Addr* (address) which has one occurrence, must exist, and be configured. The node Addr is accessible by Add and Get functions.
2. Nodes *AddrType* (address type), *PortNbr* (Port number), *Name*, *DB* (database), *clientUserName*, *clientPW*, *ToNapID* (default physical reference by relative URI for connectivity information stored elsewhere in the management tree), and *ServerID* (Data synchronization server ID or DSS ID) each having zero or one occurrence.
3. DB node is a named parent (root) of the database objects. For example, *ContactsDB* is a parent of database for the contacts. CTType is object at the leaf, which is used to define the supported media contents by the database. LDBURI (local database relative URI) and RDBURI (remote database relative URI) are the leaves of a DB root object.

7.7 Mobile File Systems

- Mobile file system is defined as the method of organising and storing files on a mobile devices. The files may be distributed. Some features of a file system are:
  - Hierarchical organization, storage, modification, navigation, access, and retrieval of files
  - Easy search and access of files
  - Maintenance of physical location of files on a storage device (e.g., memory stick, or hard disk)
  - The system functions use the data from the file server software in response to client requests
  - Technology similar to that of databases

- A file system for smart cards consists of a master file which is a root directory. It stores all file headers. A header contains the description about a file.

A file system for a mobile computing system must have following features:

- ✓ Scalability
- ✓ Support for defined semantics for sharing of files even in case of network failure.

- ✓ Support for disconnected operations and provision for reintegration of data from disconnected clients or server.
- ✓ High performance through client-side persistent caching.
- ✓ Provision for replication at server.
- ✓ Security, access control, authentication and encryption.
- ✓ Continuous operation even in case of partial failure of network connectivity.
- ✓ Network which adapts to the bandwidth available at a given instant.

- CODA is a distributed file system which possesses all the features of a file system. Files on CODA server are organised by server partitioning which contain files grouped into volumes. The three states of connection to the clients to which files are distributed to server are – (i) disconnection (ii) weak connection (iii) strong connection.
  - ✓ *Integration* takes place in case of disconnection. It means of merging of objects received from a connection at different instants.
  - ✓ *Trickle reintegration* takes place in case of weak connection. It means adding of objects received at smaller bandwidth.
  - ✓ *Hoarding* takes place in case of strong connection. It means collecting objects received at large bandwidth.

Some deficiencies of CODA are as follows:

1. It assumes that mostly one user writes the data into a file.
2. Each change is not tracked.
3. It detects at an instant the conflicts only when a different user is changing a file. This facilitates reintegration of the file data from users.
4. There is inconsistency if a file is being read at the same time when reintegration is taking place.
5. Hoarding is not permitted in case of a weak connection.

7.8 Wireless LAN (Wi-Fi) Architecture and Protocol Layers

- Wireless LAN—WLAN (IEEE 802.11a, 802.11b and 802.11g) architecture has two service sets—basic service set (BSS) and Independent basic service set (IBSS).
- BSS devices in each set interconnect to the access-point using 802.11 and form a single station STA_A of WLAN using the same frequencies for radio and the station interconnects to other stations through access-points.
- IBSS has devices which network with each other using 802.11 protocol. These devices either communicate directly with one another or communicate among themselves after forming an ad-hoc network. They form a set of stations (STA_B, STA_C,….) in a WLAN. Each station uses same frequency band for radio coverage. IBSS does not connect to an access-point also. A station has many devices which can interact through peer-to-peer communication.

Below fig shows the two service sets of WLAN architecture.

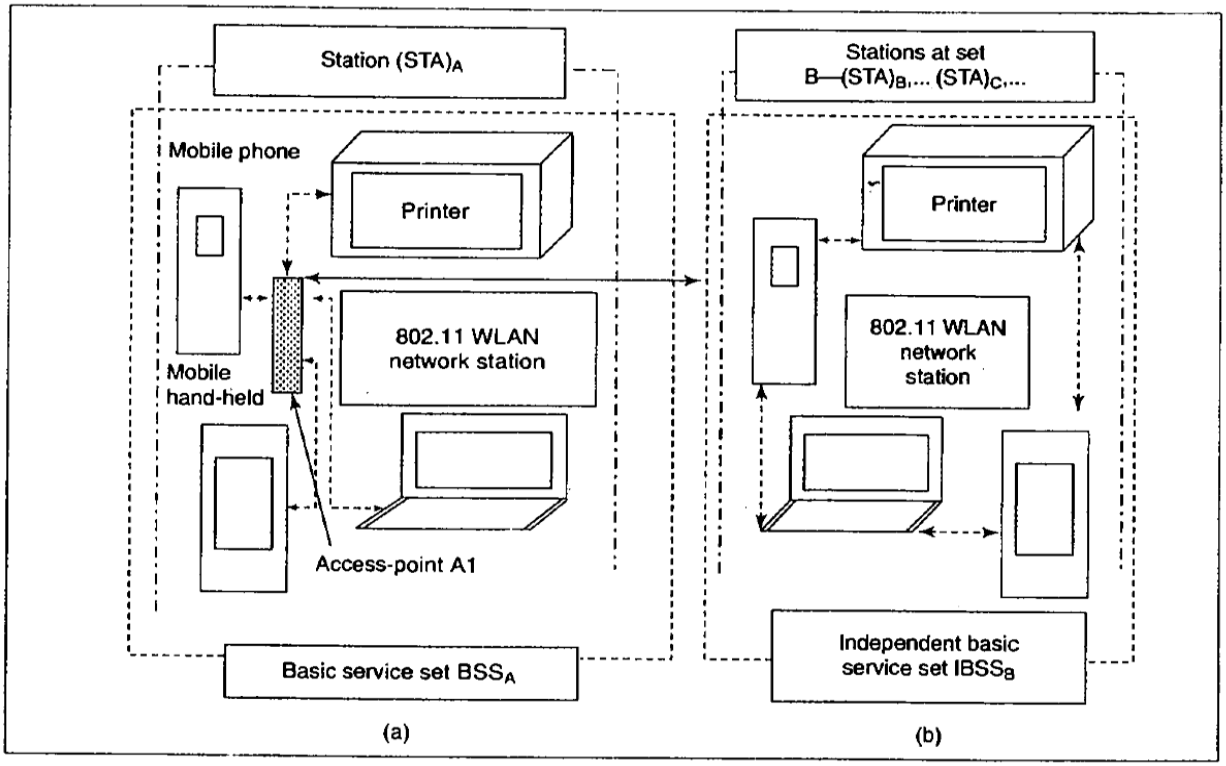

Fig. 12.1 (a) Basic service set (BSS), which also has an access point for connectivity to a distribution system (b) Independent basic service set (IBSS), which has no access point for connectivity to the distribution system and which may have multiple stations, which also cannot communicate among themselves

The following examples highlight a standard basic feature of 802.11 that it supports both access-point-based fixed infrastructure WLAN network using BSSs and adhoc peer-to-peer data routing network using IBSS stations.

- ✓ Stations in a given IBSS
- ✓ WLAN network having stations at the BSS in an ESS.
- ✓ Roaming in a WLAN network.

Below fig shows the 802.11 LAN access points networked together using extended service set (ESS), which functions as a distribution system

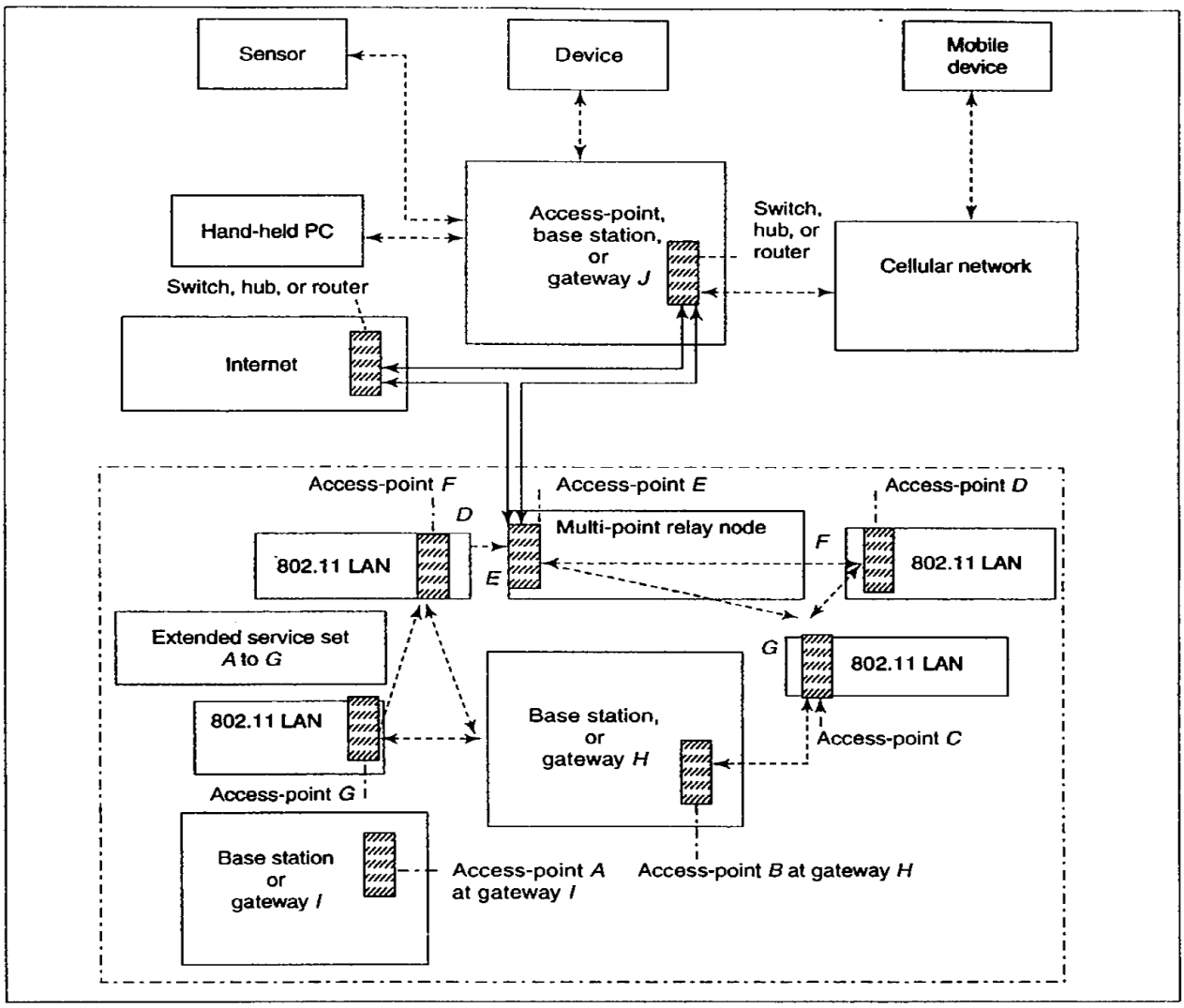

Fig. 12.2 802.11 LAN access points networked together using extended service set (ESS), which functions as a distribution system

7.8.1 Protocol Layers

IEEE 802.x is a set of protocols defined for networking the computers. x=1 gives specifications for bridging of LLC (Logic Link control) and MAC (medium access control) sub-layers and for management of layers 1 and 2. x=2 gives specifications for LLC sub-layer 2. x=1 and 2 specifications are common to all standards in 802.x for x=3 and above. x=3 gives specifications for MAC sub-layer of layer 2 and physical layer for wired LAN, called Ethernet. Upper layers are common in protocols 802.x.x=10 gives security specifications for layers 2 and above and is common in protocols 802.1y.

Below fig shows the function of physical layer and MAC sub-layer of layer2 and also shows the basic protocol layers of the transmitter and receiver in IEEE 802.11.

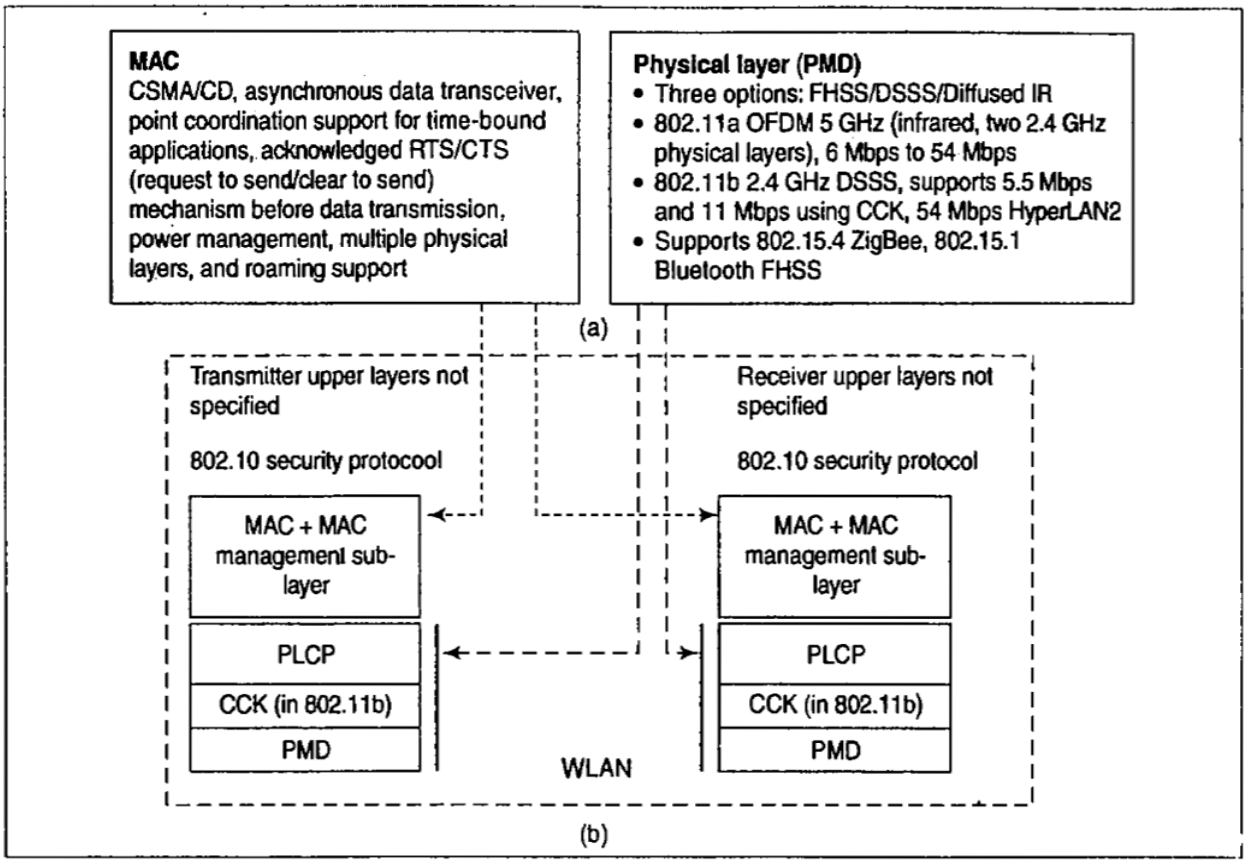

Fig. 12.3 (a) Functions of MAC and physical layer (b) Basic protocols layers in 802.11 in receiver and transmitter

Physical layer has two sub-layers in 802.11a – PMD (Physical medium dependent) sub-layer and PLCP (physical layer convergence protocol) sub-layer. There is an additional sub-layer in 802.11b – CCK (complementary code keying). PMD, PLCP, MAC and MAC management sub-layers are explained in the following subsections.

- **PMD Sub-layer**
  - PMD protocol specifies the modulation and coding methods. The three different methods are briefly described below:
    - ✓ FHSS – Radiated at 10mW, 100mW and 1W as per country-specific restrictions. Modulation is 1Mbps Gaussian BPSK or 2Mbps Gaussian QPSK.
    - ✓ DSSS – Using 11-bit Barker code radiated at 10mW, 100mW and 1W as per country-specific restrictions and 1Mbps or 2Mbps data rates. Modulation is DQPSK.
    - ✓ PPM (Pulse Position Modulation)—a modulation method. 16 PPM is used for 1 Mbps and 4 –PPM for 2 Mbps data rate.
- **PLCP Sub-layer**
  - PLCP specifies sensing of the carrier at the receiver and packet formation at the transmitter. PLCP sub-layer protocol prescribes the standard procedure for convergence of PMD to MAC at receiver and from MAC to PMD at transmitter.

The header and payload in each frame are defined for different cases as follows:

(a) PLCP for FHSS:

1. First 80 bits (010101...) are for synchronization.
2. Next 16 bits (0000 1100 1011 1101) are for SFD (start frame delimiter) for synchronization between the frames.
3. Next 12 bits specify the data length in bytes. This includes the error check (CRC) bits in the payload (PLCP-PDU-LW (protocol data unit with length in words) of 1 byte each).
4. Next 4 bits are for PSiF (PLCP signalling field) (0000 for 500 kbps, 0010 for 2Mbps, and .... the maximum is 8.5Mbps).
5. Next 16 bits are for checksum, called HEC (header error check).
6. Remaining bits in the frame are for the payload.

(b) PLCP for DSSS:

1. First 128 bits are for synchronization, gain set, energy detection, and frequency offsets.
2. Next 16 bits (1111 0011 0100 0010) are for SFD (start frame delimiter) for synchronization between the frames.
3. Next 8 bits are for PSiF (0000 1010 for 1 Mbps DBPSK and 0001 0100 for 2 Mbps DQPSK, and so on for higher data rates).
4. Next 8 bits (0000 0000) are for PSF (PLCP service field) for 802.11 standard frame.
5. Next 16 bits are for PLF (PLCP length field) to specify the data length of MPDU (maximum protocol data units) in μs.
6. Next 16 bits are for checksum of PSiF, PSF, and PLF. The sum is the HEC.
7. Remaining bits in the frame are for the payload.

(c) PLCP for diffused IR:

1. 57–73 bits (010101...) are for synchronization and frequency offsets.
2. Next 4 bits (1001) are for SFD for synchronization between the frames.
3. Next 3 bits for PSiF (000 for 1 Mbps 16-PPM and 001 for 2 Mbps 4-PPM).
4. Next 32 bits are for PSF for DC level adjustment in the frame.
5. Next 16 bits are for PLF to specify data length of MPDU in μs.
6. Next 16 bits are for checksum of PLF, PSF, and PSiF. The sum is the FCS (frame checksum).
7. Remaining bits in the frame are for the payload (MPDU less than 2500 bytes).

➢ **MAC and MAC Management Sub-layers**

- A MAC sub-layer specifies CSMA/CD, RTS/CTS, and point coordination function (PCF) mechanisms. Another sub-layer specifies MAC management.
- MAC layer for medium access control has the following features:
  - CSMA/CD
  - Point coordination support for time-bound applications
  - Acknowledged RTS/CTS (request to send/clear to send) mechanism before the data transmission
  - Power management
  - Multiple physical layers for same MAC support
  - Mobile node roaming within an ESS by registration and node association, dissociation, and re-association on moving to another BSS.
- MAC frame format of 802.11 in each frame is as follows:
  1. 16 bits for frame control with specifications as follows:
     (a) 2-bit field for protocol version.
     (b) 2-bit field for frame type—(i) 00 means that the frame is for management (registration, handover, power management) (802.11i security using AES and DES), (ii) 01 means that the frame is for control, and (iii) 10 means that the frame is for data.
     (c) 4-bit field for subtype (for the management frame, it is 1100_b for CTS, 0000 for request for association, and 1011_b for RTS) and subtype = 0000 when a user data frame is transmitted.
     (d) 2-bit field to specify one of the four possible addresses when transmission is between mobile stations and access-point (within BSS nodes and devices) or between the access-points over a data source (DS) (using the ESS).
     (e) 1-bit field which is 1 when more data fragment(s) is to follow for a present service data units frame of the MAC protocol (before transmission, a service data units frame may be fragmented and therefore another fragment may be required to follow the present one. This field is set to 1 when another fragment is to follow the present one and to 0 when none is to follow.)
     (f) Four 1-bit fields for retry, power management, more data expected from sender, and WEP (wired equivalent privacy in 802.11b).
     (g) 1-bit field which is 1 for *order* (means receiver must strictly process as per the order in which bits are received).

per the order in which bits are received).

2. Next 16 bits for duration/ID field (msb 0 means duration in μs for frame and 1 means ID) for SFD for synchronization between the frames (Also used as NAV (net allocation vector)).
3. Total 26 × 8 bits for three 48-bit address fields to specify addresses, 16-bit sequence control field, and fourth 48-bit address field. (Sequence number in control field is necessary due to separate duplicate frame or acknowledgements.)
4. Data bits (<2312 bytes) in a frame are for the payload (MPDU less than 2500 bytes).
5. 32 bits CRC as in Ethernet or other 802.x data frames.

Functions of MAC management sub-layer are as follows:

- Roaming management which is done using scanning of the nodes (devices) moving into a new area. The scanning is done by the access-points and detects the new devices or the loss of devices in the area. The access-point registers or deregisters the devices after the scanning. New device registration provisions for device association at new access-point when it roams into the new area from another access-point area.
- Internal receiver clocks are synchronized, which is necessary.
- Generation of beacon signals is also part of management functions. (Beacon signals are used for helping or warning others by indicating the presence of a device or an access-point. A BSS periodically sends beacon signals, which contain—(i) time stamp for synchronizing node clock and (ii) power management and roaming data.)
- Transmitter switches to power-save mode after each successful data transmission for power management periodically activating the sleep mode. Buffering by a receiver and start of processing after enough data is received in buffer also saves power.

7.9 WAP 1.1 and WAP 2.0 Architectures

WAP offers open development platform for integrated services for voice, data, Internet, picture, music attachment to mail, gaming applications, control services etc. WAP 2.0 is wireless protocol for synchronization of WAP client computers and WAP or HTTP server. A WAP gateway connects WAP client to HTTP servers. WAP 2.0 has three important features over WAP 1.1—

- SyncML synchronization.
- WAP push service.
- MMS service.

Below fig shows the architecture for communication between a WAP and HTTP server.

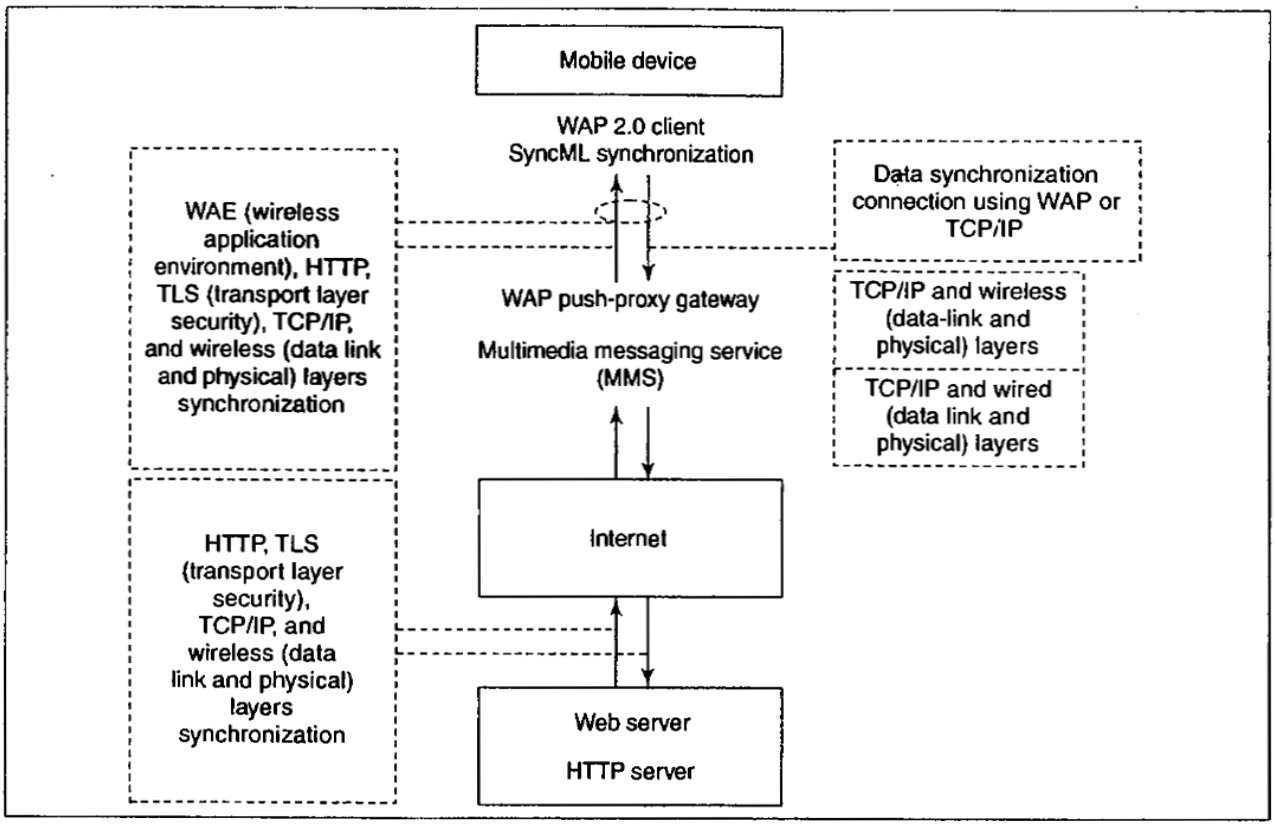

Fig. 12.4 WAP 2.0 client, gateway, and web HTTP server architecture

7.9.1 Layers of WAP

The Wireless Application Environment (WAE) in WAP 1.1 consists of the following components:

- WML (Wireless markup Language)
- WMLScript
- WBXML (WAP Binary XML)
- WTA (wireless telephony application)
- Data format [vCard 2.1,vCalendar 1.0, address book, pictures (jpg, gif,..)..]

WAP gateway provides *access to the wireless and wired networks* and also builds *caches* due to frequent disconnections in the wireless environment. It ensures security in wireless and wired networks.

➢ **WAP Gateway**

- ✓ A WAP gateway is required for *protocol conversion* between two ends – mobile client device and HTTP server.
- ✓ Gateway converts WAE 1.1, WSP, WTP, WTLS and UDP layers encoded data packets into HTTP, TLS, and UDP layers encoded data packets when the device transmits data to server. It does decoding when vice versa is done.
- ✓ WAP 1.1 gateway also has a *WML encoder-decoder* so that the application written in WML gets converted to HTML when WAE application is sending requests to HTTP server.

✓ The gateway performs *iWMLScript compilation* into CGI script which runs at the HTTP server to get HTML response which is sent to client application. Refer the below figure.

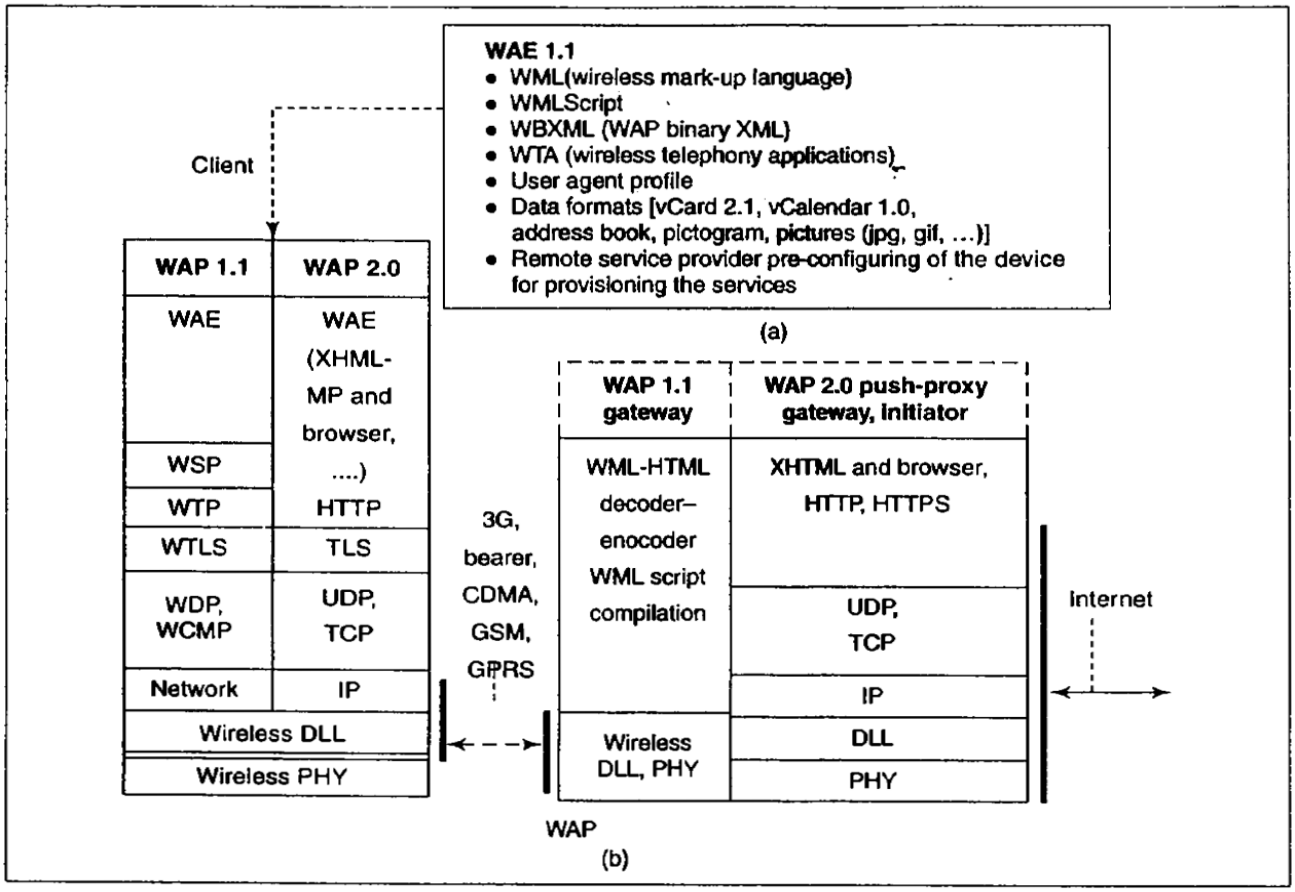

Fig. 12.5 (a) Functions of WAE (b) Protocols layers WAP 1.1/WAP 2.0 transmitter and gateway/proxy, respectively

➢ **WAP Push-Proxy Gateway**

- ✓ WAP 2.0 uses XHTMLMP (Extensible Hypertext Markup Language Mobile Profile) in place of WML, Therefore encoding and compilation is not required and only a WAP 2.0 proxy suffices.
- ✓ HTTP web servers function in pull mode .i.e., HTTP client application sends a request to the server and the HTTP server sends the response.
- ✓ A WAP 2.0 proxy is required for pull mode.
- ✓ A push-proxy gateway is used to exchange data packets between a mobile device through wired Internet and Web servers.
- ✓ The role of WAP 2.0 gateway is restricted to provisioning for push and pull mode services from the servers.

7.9.2 Physical and networking Layers

- ✓ A datagram gives independent information and is stateless. The data of a datagram is sent by a connectionless protocol. WDP (wireless datagram protocol) is one such protocol to send connectionless information as datagram.

- ✓ A WDP in WAP suite is for datagram service. It can be used for multicasting a datagram on the network.
- ✓ WCMP (wireless control message protocol) is another connectionless protocol, which is a part of WAP protocol suite. WCMP employs a datagram with a WCMP header when sending the messages for querying to find information, reporting errors, making route address advertisement for a router seeking messages.
- ✓ Transmitted datagram has a header and then user data which is received from the upper layers at the device. The header consists of a source port, a destination port, source address, destination address, length of data and checksum bytes for the header. Below fig shows the transfer of a WDP datagram.

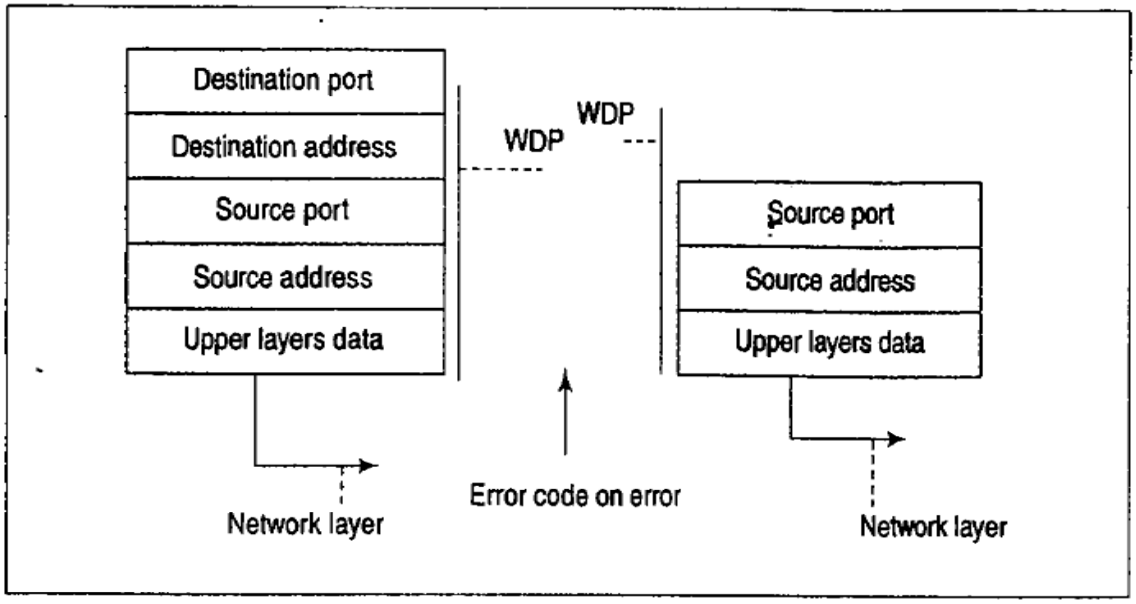

Fig. 12.6 WDP protocol header over the upper layer data and return of error code as per the error

7.9.3 Wireless Transport Layer security

- ✓ When data transaction occurs between client device and gateway, wireless transport layer security (WTLS) assures integrity and privacy in transactions and device authentication.
- ✓ WTLS maps SSL(secure socket layer) in HTTPS. WTLS supports TCP, WDP and WCMP.
- ✓ A secure session is established before data from upper layer that are above WTLS layer is transmitted through gateway or proxy to the other end peer and received through gateway or proxy to the upper layers.

✓ WTLS specifies the following sequence of peer-to-peer message exchanges for establishment of the secure session:

1. Source device messages to *create* a secure channel as follows—(i) source address and port, (ii) destination address and port, (iii) RSA or ECC (a proposed suite of algorithms for key exchange), (iv) IDEA or DES (a proposed suite of algorithms for ciphering the data), and (v) compression method for data compression.
2. Other end messages for secure channel exchange for confirmation of *create* process as follows—(i) sequence number mode, (ii) how many times key is refreshed and exchanged again, (iii) identification of session after establishment of the session, (iv) RSA or ECC (a chosen suite of algorithms for key exchange), (v) IDEA or DES (a chosen suite of algorithms for ciphering the data), and (vi) chosen compression method for data compression.
3. On request from the other end, source device messages for secure channel public key authentication by a client certificate.
4. Source device messages to commit request (Section 7.4 explained meaning of committed transaction).
5. Other end peer messages for commit confirmation request.

After the above exchanges, the user data exchanges start as shown in Fig. 12.7.

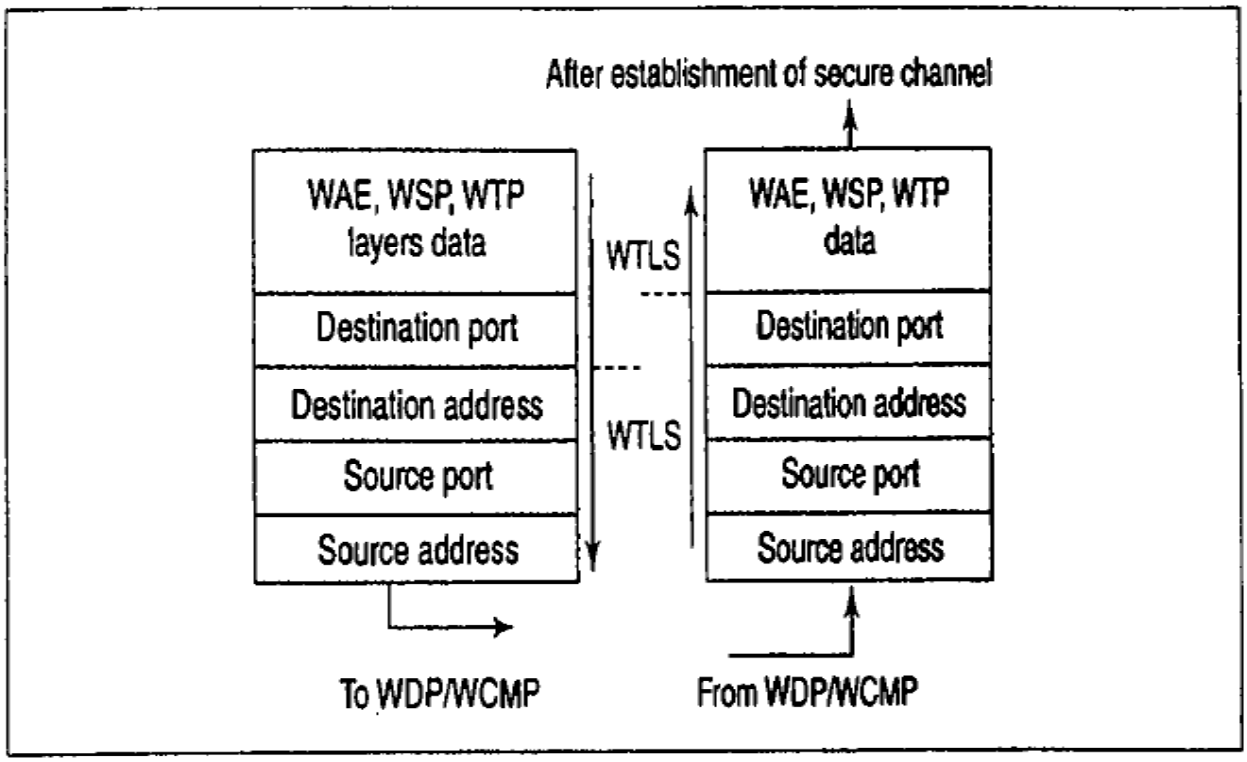

Fig. 12.7 WTLS protocol header over the upper layer data in requests and WTLS protocol header from the lower layers in responses

7.9.4 Wireless Transaction Protocol

- ✓ Wireless transaction protocol (WTP) transmits data to the WTLS in case of secure transactions and directly to WDP or WCMP.
- ✓ WTP supports joining of the messages and enables asynchronous transactions.
- ✓ WTP supports abortion of the transactions and provides the information about the success or failure of a transaction to the sender.
- ✓ WTP is a interface to ensure reliability of transactions. There are three WTP service classes—0,1, and 2.

1. Class 0—In this class, a source sends the messages with no response from the other end.
2. Class 1—This class is for reliable data transfer which takes place in the following manner—Source first invokes a transaction along with the request. Device then obtains the confirmation of invocation. This is followed by the transaction for the resulting response. The device sends the acknowledgement. The transaction removes duplicate data, provides retransmission as well as a transaction identifier. It provides push services in which there is no acknowledgement of data by user, except that there is confirmation of invocation.
3. Class 2—This class is also for reliable data transfer occurring as follows—Source first invokes a transaction along with the request. Device then obtains the acknowledgement of data (through gateway or proxy) from user. This is followed by a transaction for the resulting response. The device sends the acknowledgement. The transaction removes duplicate data, provides retransmission as well as a transaction identifier. It provides acknowledgement of two types—user acknowledgement and automatic acknowledgement.

Below fig shows the WTP headers when sending WTP invocation and request for results, confirmation of WTP invocation.

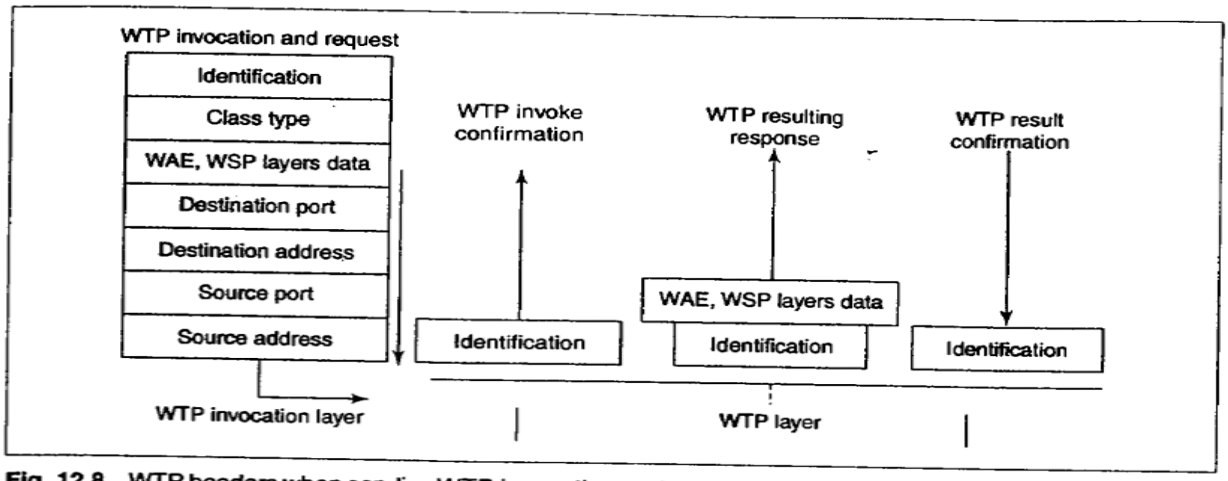

Fig. 12.8 WTP headers when sending WTP invocation and request for results, confirmation of WTP invocation, WTP resulting response and WTP result confirmation

7.9.5 Wireless Session Protocol

- ✓ Wireless session protocol (WSP) transmits data to WTP in case of thin client transactions or directly to WDP or WCMP.
- ✓ WSP manages session as follows:
  - o A session is first called.
  - o An established session can be suspended and then resumed from the point at which it was suspended.
  - o A session can be terminated.

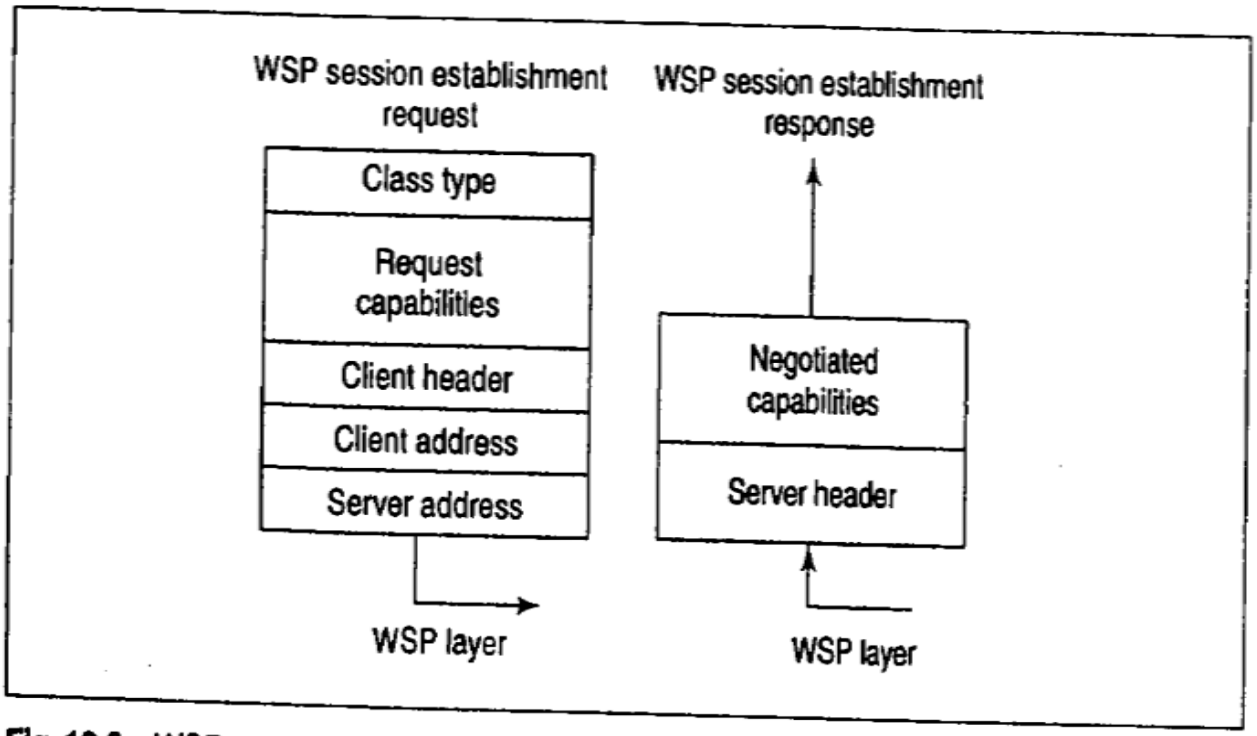

Fig. 12.9 WSP protocol session connection establishment and resulting response headers

✓ There are three WSP service classes—0,1 and 2.

1. Class 0—This class is for a source sending the unconfirmed push. It supports session suspension, resumption, and management. The messages sent from the source do not get any response from the other end.
2. Class 1—for a source sending the confirmed push.
3. Class 2—for a source supporting session invocation, suspension, and resumption.

Below figure shows the WSP headers.

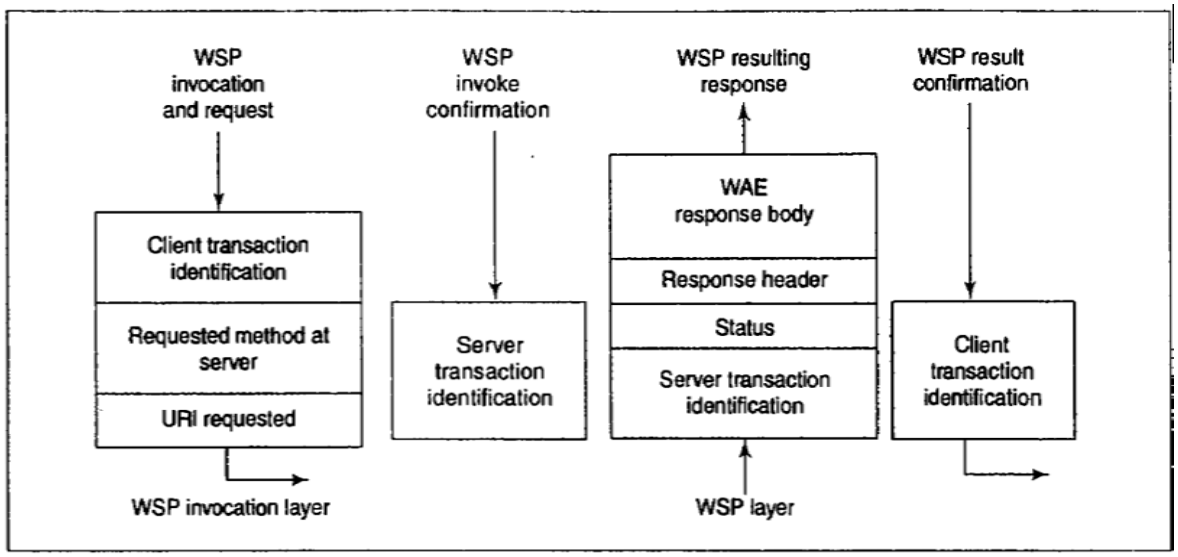

Fig. 12.10 WSP headers when sending WSP invocation and request for results, confirmation of WSP invocation, WSP resulting response and WSP result confirmation

✓ WSP exchanges take place for WSP in the following sequence:

- A method is invoked and server is requested for the results.
- The method runs at the server and generates a response.
- The server pushes the response. This is shown in below figure.

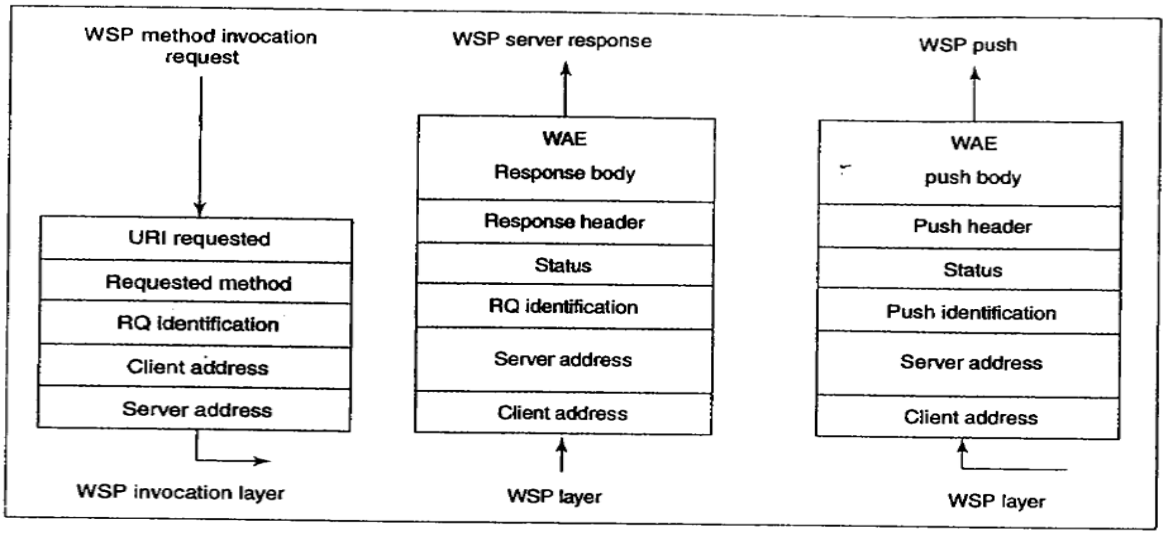

Fig. 12.11 WSP headers when connectionless WSP exchanges take place for WSP invocation and request for results, method resulting response and server pushes, respectively

7.9.6 Wireless Application Environment (WAE)

- ✓ WAP 1.1 has two sets of software – WAE user agent and WAE services.
- ✓ WAE consists of WML, WBXML, WTA, WMLScript, and data formats.

- ➢ **WML (Wireless markup language)**
  - ✓ WML is used to create cards for mobile applications. Two versions are WML 2.x and WML 1.x.
  - ✓ WML 2.x includes XHTML-MP which includes XHTML. WML 1.x does not include XHTML.
  - ✓ A card represents an interaction with the user and the deck contains the cards. A WML parser parses the tags, attributes, and underlying text within the tags present within the deck or card. A WML card has following features:
    - Provides the *content.*
    - Supports variety of formatting commands as well as layout commands. The commands are defined by tags and attributes.
    - Provides user interface for mobile devices with constraints.
    - Organizes similar to deck and cards. A WML deck file is saved in a file with extension *wml*. Each file contains one deck.
  - ✓ A WML card is first validated against its declared document type using WML 1.3 DTD before parsing. Parsed data, information, and contents are given as input to a Java program for the application or server which runs methods at the browsers or server.

The following explains how an application runs at a mobile device in WAE. A WML card containing a client-request is transmitted and is decoded at the gateway. After decoding, the gateway communicates the request to the server in HTML format. The server generates an HTML response by employing the method(s) which run at the server. The response is encoded in WML form at the gateway. The gateway transmits the WML response to mobile-device client which runs an application after parsing the WML. The application uses the methods which run at the client.

WML has tags <b>, <u>, <i>, <big>, <small>, , and for bold, underline, italic, big, small, strong, and emphasis rendering, respectively, for a text-display. HTML anchor tag <a with attribute href = ...> is used for linking and navigation in WML also. <timer> is another tag. The actions on events are by the tags <ontimer>, <oneventbackward>, <oneventforward>, and <oneventpick>. The renderings after the actions on the events occur as per the interior paragraph entities.

- **WMLScript**
  - ✓A WMLScript is a script language in which each line is loaded in computer and is executed at run time only. There is no pre-compilation.
  - ✓WMLScript is a client-side scripting.
  - ✓WMLScript can embed the marups in WML.
  - ✓The execution of WMLScript is fast because it is compiled at the gateway and the byte code is sent to client.
  - ✓It also used for generating error messages. Example for WMLScript:

Sample Code 12.3 This code shows how can the division $z = x \div y$ can be carried out using WMLScript.

```
extern function divide (varCompute x, y)
{
    var z = x/y;
    WMLBrowser.setvar (varCompute, z)
    WMLBrowser.refresh ( );
}
```

Standard library functions: WMLScript standard libraries have WMLScript functions which cannot be easily implemented sometimes. A brief description of different libraries and the corresponding functions is as follows:

- *WMLBrowser* library has the functions to control the WML browser or to get information from the browser.
- *WMLDialogs* library has the functions which display the input boxes to users. It also provides for alert and confirmation messages.
- WMLLang library has the core WML functions. For example, converting a data type integer to string character.
- *WMLString* library has the functions that help in concatenation, truncation, picking of select portions, and manipulation or finding the length of the strings. An example is the *find()* function. It helps in knowing whether a sub-string is a part of a string. If yes, then the function returns the index of the first character of the match in the string, otherwise it returns –1. String.find ("09229122230", "30") returns 9 which is the index of first character of the match in the string. String.find ("09229122230", "39") returns-1 since there is no match between sub-string characters and the string. Another example is var *strlen* = String.length ("WELCOME TO ABC MOBILE). It returns 21 because number of string characters are 21 (Space is also a character.).
- WMLURL has functions for using relative URLs or absolute URLs for finding the port number or for testing whether a URL is valid or not. [For example, http://www.microsoft.com/msoffice/winword/ is a relative URL.

Full form http://www.microsoft.com/msoffice/winword/newfile.doc in which the file name newfile.doc is also mentioned in the end after the winword/ is called absolute URL.]

- *WMLFloat* library has the functions that help in performing floating-point arithmetic operations in case a specific WAP device supports floating-point operations, conversions, and calculations.

➢ **WBXML**

WAP 1.1 provides for communication of client with gateway or proxy using WBXML. XML and WML page document are not compact. WBXML is a specification in binary representation so that XML or XML-based language can be transmitted in compact format. A binary number can represent a tag in place of characters. Another binary number can represent an attribute in place of characters. For example, attribute ID needs two characters. It is represented by a single byte. Attribute title needs five characters. It is also represented by a single byte.

➢ **WTA**

- ✓ WTAI(WTA Interface) defines features like call set up, call accept, call forwarding, caller line ID, call hold, call waiting, conferencing, ring tone, speed dial, MMS, SMS up to 160 characters, emergency number etc.
- ✓ It provides the interfaces for the features using WML browser.
- ✓ A WTA URI can be wtai://wap.mcard: followed by a telephone number. This is identical to port number specification provided in the URL.
- ✓ WTA provides security interface.
- ✓ WTA server can push the WMLScript or deck contents.
- ✓ A WTA event handler can handle events.

➢ **User Agent Profile**

User agent is software used by the user to give input using VUI (voice user interface) and GUI (graphic user interface) and to interact with mini browser (browser with limited screen size). It executes the WMLScript at the client and displays the results. User agent displays the WML decks received as response from the server. User Agent Profile provides small screen device characteristics, font, and display capabilities. The profile also enhances the input capabilities, for example, the use of T9 keypad, stylus, and touch screen is enabled.

- **Data Formats**
  - ✓ vCard 2.1 is the format for visiting card. vCalendar is the format for calendar.
  - ✓ Mobile devices provides pictogram which is a small picture of very low resolution that cannot be split and can be placed along with the text. Pictogram is used for displaying logo.

- **Remote service provider pre-configuring of the device for provisioning of services**
  - ✓ WAP 2.0 provides for SyncML which is used for data synchronisation between the server and mobile devices.
  - ✓ WAP 1.1 also provides the functions and WML methods at the server for pre-configuration of the device from the server.

- **WAP Push**

WAP provides Push OTA (over the air) which is a simple protocol sub-layer in WSP. It provides authentication of the push initiator (server) and also helps in selection of the pushed contents. The protocol handles push-session-request, connect, suspend, resume, and disconnect functions. It also handles push, server-confirmed push, abort, and unit-push functions.

7.10 Bluetooth-enabled Devices Network

Bluetooth devices can form a network known as piconet with the devices within a distance of about 10m. Various piconets form an ad-hoc network called scatternet within 100m through a Bluetooth-enabled bridging device. Below table shows the basic features of Bluetooth.

Table 12.1 Basic features of Bluetooth

Property	Description
Frequency band	2.4 GHz with Bluetooth radio characteristics given in Table 12.2
Bluetooth protocol layers	(a) Layer 1—baseband and radio (b) Data link layer—layer 2—RF-communication, L2CAP (logical link control and adaptation protocol) and LM (link manager) or host-controller interface, layer 2—L2CAP, or layer 2—audio (c) Layer 6—session—object exchange (OBEX) (d) Layer 7—security and application software layers are as specified by Bluetooth Sync, vCal (for Calendar), vCard [for contact (visiting card)] or Object Push (PIM) or Binary File Transfer or audio applications.
Sessions, object exchange, and other protocols	SyncML client, SyncML engine, and OBEX (Fig. 12.13(a))
Network characteristics	(a) Connection-oriented communication (b) Master–slave communication within same piconet (Section 12.4.2) (c) Negligible interference between piconets as each uses distinct channel-frequency hopping sequences (d) Ad-hoc network peer-to-peer communication between two devices on two different piconets in a scatternet
Bluetooth features	• Used for low power short range transmission • Employed for wireless short range exchanges in mobile environment within 10 m network in master–slave mode and within 100 m in scatternet. Bluetooth radiations between piconets are omni-directional. • Network connection latency—3 s • Bit rate—less than 1 Mbps • Protocol stack—larger than IrDA or Bluetooth • Code size—2% to 50% more compared to that for a Zigbee device • Bluetooth radio—FHSS

(*Contd*)

(*Contd*)

Application example	• To connect mobile device to hands-free head phone for hands-free talking [wireless connectivity between the headset, ear-buds, and the mobile handset] • To connect PC to joystick, keyboard, and mouse, data exchange between the computer and printer, or object exchanges between computer and mobile phone handset • Bluetooth-enabled digital camera placed near a Bluetooth-enabled PC for downloading the pictures or video clips onto the PC • A Bluetooth-enabled PC downloading MP3 files from CD player or broadband Internet • An iPod is placed near a PC. The PC transfers the media files to the iPod so that the user can listen to selected music when mobile.

➢ **WPAN Synchronization**

- Bluetooth protocol is used for wireless personal area synchronisation among mobile devices and Bluetooth-enabled PCs.
- Bluetooth protocol is connection-oriented protocol using Bluetooth object exchange OBEX. An object can be a file, address book, or presentation with a specification for a method which runs a specific task.
- Synchronisation can be through SyncML codes. SyncML uses the message CONNECT, followed by PUT or GET, and then ABORT, either automatically at periodic intervals, or initiated by device or PC.
- Bluetooth synchronizes PIM data. Refer the below fig.

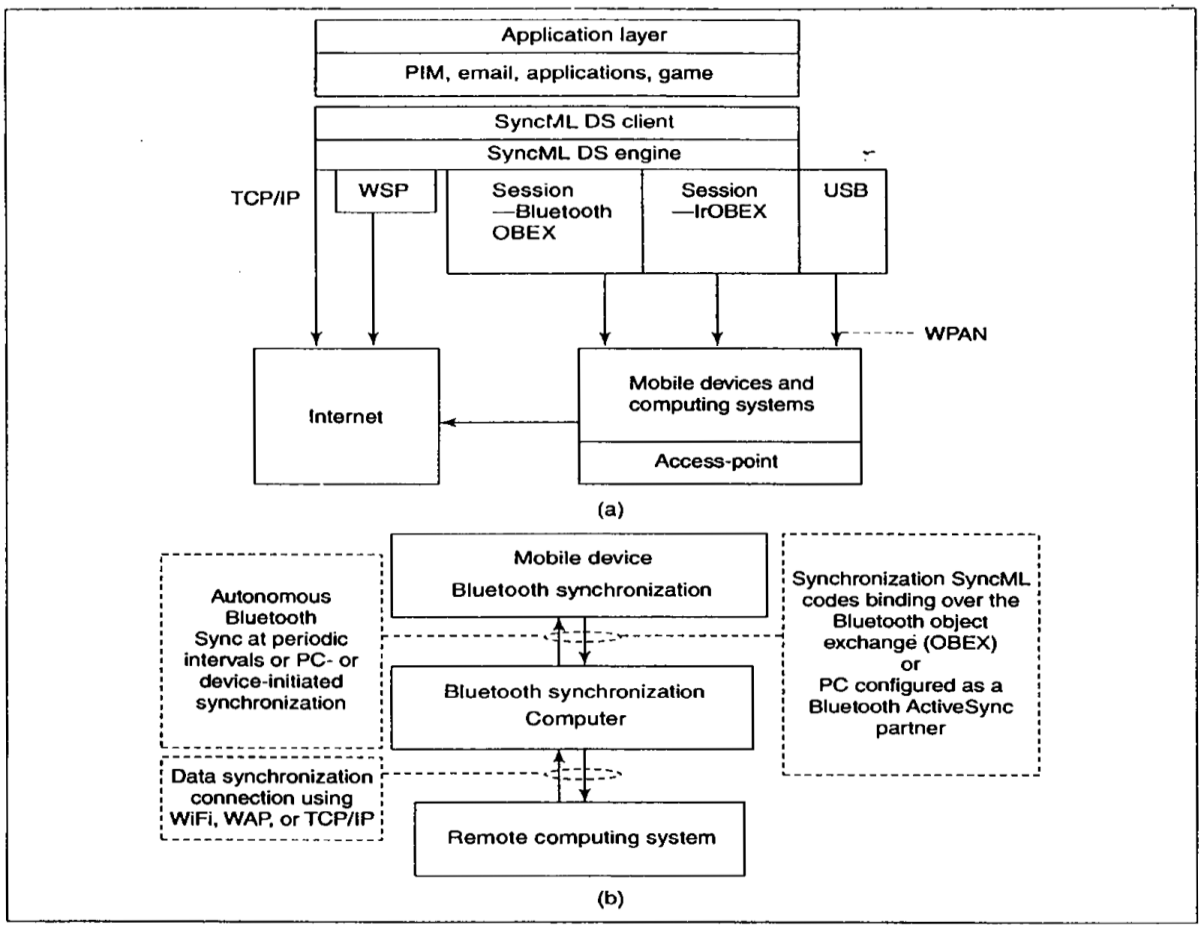

Fig. 12.13 (a) Data synchronization (Section 9.1) between the mobile devices and computing systems in a WPAN takes place and how remote computing systems connect to each other through Internet (b) Synchronization between the Bluetooth devices within short distance (1 m to 100 m range as per the radio) and remote computing systems

- **Bluetooth Networks**
  - The device first establishes the piconet becomes master and others which discover the master becomes the slaves in the piconet. Slave means that the clock of the master functions as reference for synchronisation.
  - A master synchronizes all active devices and there are identical hopping sequences of their frequencies for each device *radio.*
  - The devices in piconet can be present in the following states:
    - ✓ Standby state – When a device is in this state, it is actually waiting to discover the master and thus the piconet. The device is yet to be assigned an address in the piconet.
    - ✓ Active state – An active state can be one of the three modes – (a) *inquiring* (carrying out discovery broadcasting in all neighbourhood and listening to the response) (b) *paging* (sending a page specifying the relationship with master after discovering the channel), and (c) connected and performing data transactions. The master device here assigns the 3-bit address called AMA (active member address).

- ✓ Park state—A device in this state has already discovered the piconet but is not communicating at present and is held in power-saving mode. Such a device is assigned a 3-bit address called PMA (parked member address).
- ✓ Hold state—In this state, device retains the AMA but suspends asynchronous connectionless link (ACL). It maintains synchronous connection oriented (SCO) link and reduces power dissipation for communication.
- ✓ Sniff state—In this state, the device retains the AMA, operates at high power level, and sniffs the data of communicating piconet at large programmable intervals as compared to active state short intervals. Sniffing means listening to the existing Bluetooth device in the vicinity.

Below fig shows the Bluetooth networks.

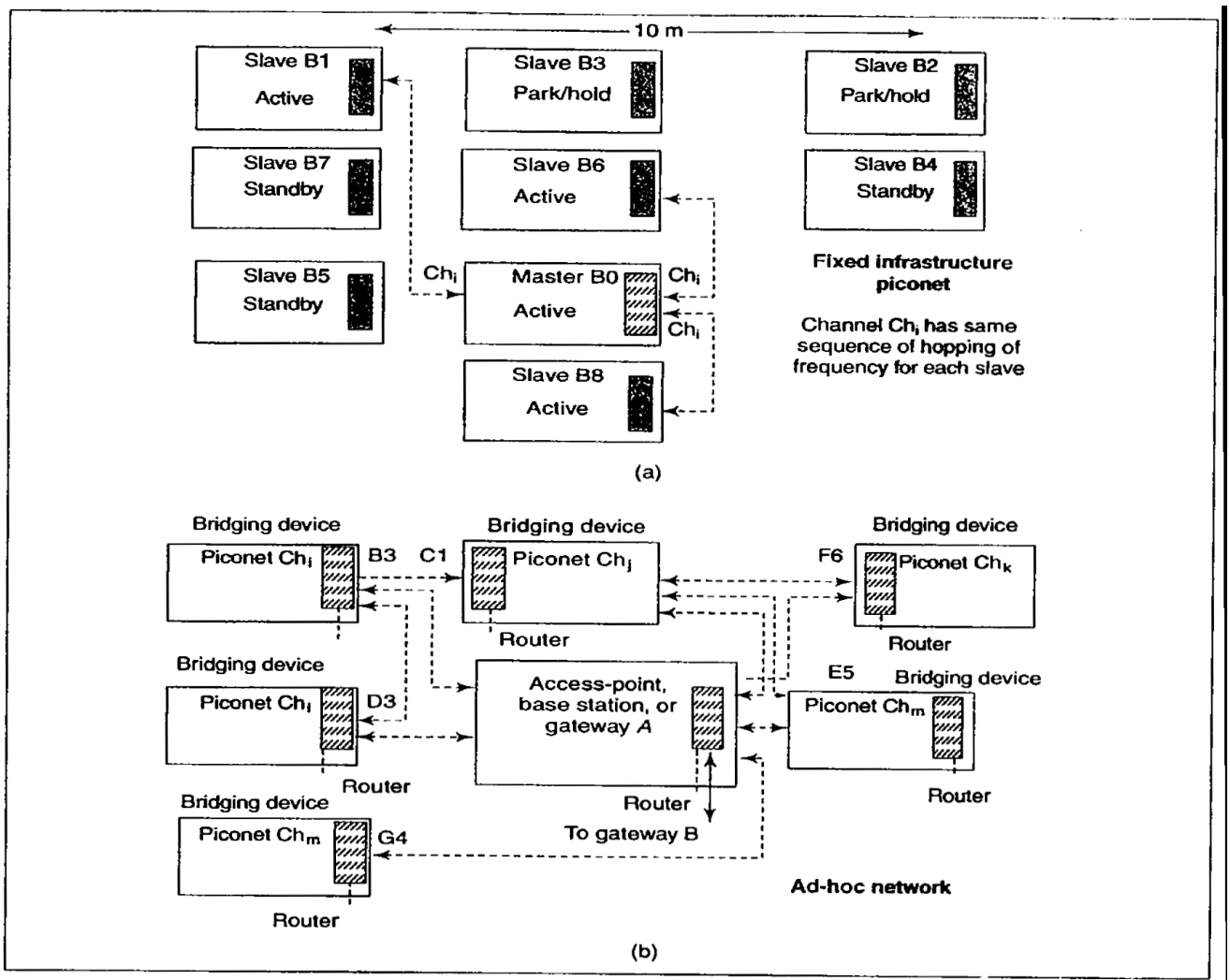

Fig. 12.14 (a) Piconet of three active Bluetooth devices and a master, each synchronized and using same hopping sequence of their frequencies forming a fixed infrastructure network architecture. Two are in park/hold state and three are in standby mode
(b) Bluetooth enabled devices ad-hoc network architecture forming a scatternet through a bridging devices of the piconets

7.11 ZigBee

ZigBee is a suite of high-level communication protocols.ZigBee devices conform to the IEEE 802.15.4-2003 Wireless Personal Area Network (WPAN) standards for operations at low data rates and low power dissipation. ZigBee devices form a personal area (home) network of embedded sensors, industrial controllers, or medical data systems. It consists of three type of ZigBee devices which are as follows:

- ✓ ZigBee coordinator—root node at each ZigBee network tree. It can connect to other networks and has full network information along with a store of the security keys for the ZigBee network nodes.
- ✓ ZigBee router node—responsible for transfer of packets from the neighbouring source to nearby node in the path to destination.
- ✓ ZigBee end-device—receives packet from a nearby node in the path from a source.

A ZigBee network can be of two types:

- ✓ *Peer-to-peer:* Here each node has a single path to the neighbouring node only.
- ✓ *Mesh:* Here each node has a path to every node.

Consider the below fig.

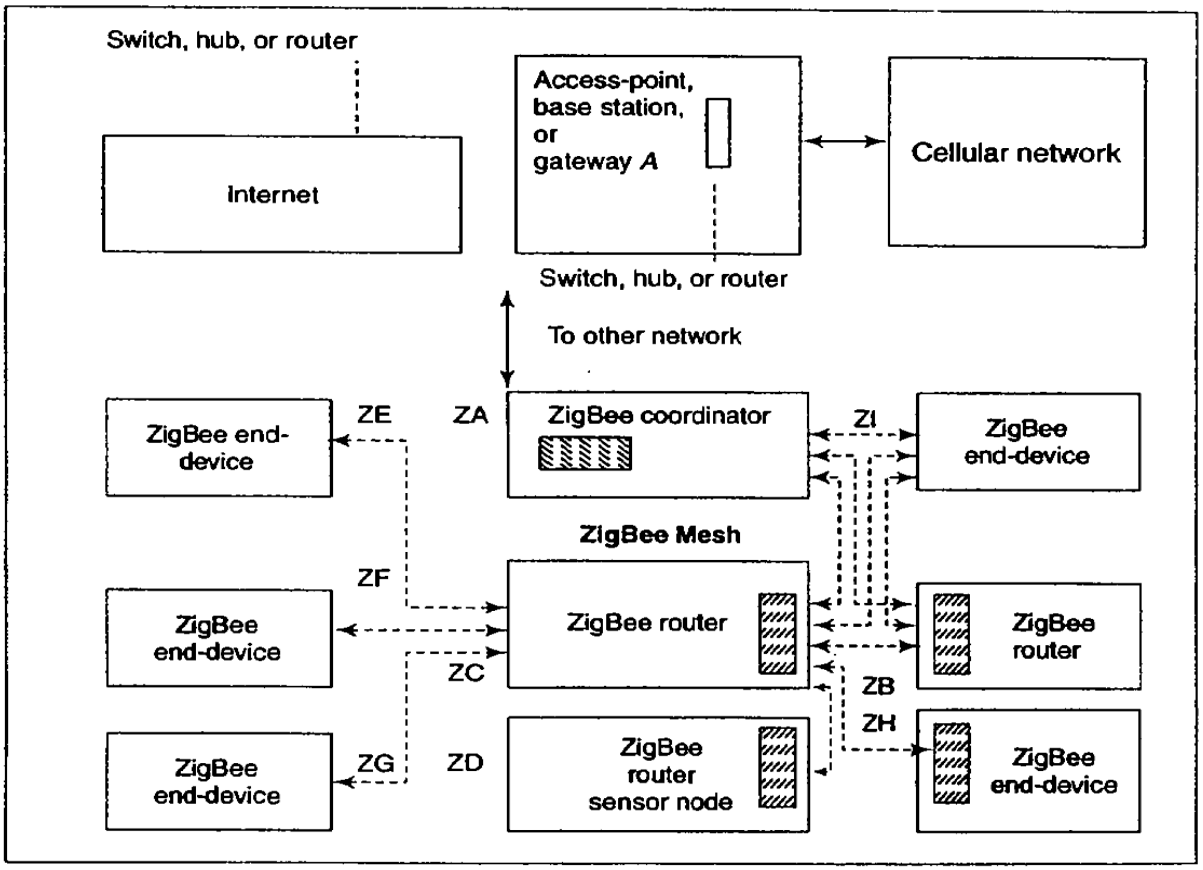

Fig. 12.17 ZigBee sensors, end-devices, and ZigBee router-devices networks

Basic features of ZigBee are as listed below:

Property	*Description*
Radio frequency bands and modulation methods	ISM bands—2.4 GHz orthogonal QPSK , 915 MHz (USA) BPSK, and 868 MHz (USA) BPSK
ZigBee device channels	For 2.4 GHz, there are 16 ZigBee channels. Each channel has freuency band $(2400 + 5\times n) \pm 1.5$ MHz, where $n = 1, 2, ..., 15$, or 16
ZigBee data transfer rates	2.4 GHz at 250 kbits/s per channel, 915 MHz bands at 40 kbit/s per channel, and 868 MHz bands at 40 kbit/s per channel.
Radio interface	DSSS (Section 4.3.1)
ZigBee protocol layers	Physical and a DLL (data link layer) part, called MAC (media access control)
Device types	Coordinator, router, and end-device types
Routing protocol	AODV (Section 11.2.4.2)
Protocol layers	• Physical layer as provided in IEEE 802.15 • MAC layer as provided in IEEE 802.15 • Security and application software layers as specified by the ZigBee Alliance
Network characteristics	Self-organization, peer-to-peer, and mesh networks
Dissimilarity with Bluetooth	• Bluetooth used for wireless short range exchanges in mobile environment and ZigBee for big scale mesh-network-based automation and remote control • Network connection latency—3 s for Bluetooth and 20 ms for ZigBee • Bit rate—1 Mbps for Bluetooth and 250 kbps for ZigBee • Protocol stack—250 kB for Bluetooth and 28 kB for ZigBee • Code size—50% down to 2% as compared to a Bluetooth device • FHSS used for Bluetooth and DSSS for ZigBee

(*Contd*)

(*Contd*)

Similarity with Bluetooth	• Both conform to IEEE 802.15 set of standards • Use of spread spectrum modulation results in spectrum efficiency in both • Use of 2.4 GHz (in USA) in both • Used for low power short range transmission • Both have small form factors radiation pattern
Application examples	• A ZigBee-enabled electric meter communicates electricity consumption data to the mobile meter reader • A ZigBee-enabled home security system alerts the mobile user of any security breach at home