Unit 7

Application Layer

WAP Model- Mobile Location based services -WAP Gateway – WAP protocols – WAP user agent profile, Caching model-wireless bearers for WAP - WML – WMLScripts – WTA.

WAP Model:

- The fig shows the example of WAP network architecture.
- Here, a WAP handset communicates with the origin server through mobile network.
- The origin server is a standard Hypertext Transfer Protocol(HTTP)/Web server, which can be developed by using tools such as Perl and Cold Fusion.
- On the origin server, given resource resides in or is to be created.
- The content received by the WAP handset are encoded in the compact binary format of wireless Mark-up Language (WML) so that they can be efficiently delivered in the bandwidth-limited mobile network.
- The WAP Gateway, located between the internet and the mobile network receives the WAP request from the handset, decodes the request from binary format into the text format, and forwards it to the origin server.
- The server origin parses the received WAP request and determines what to retrieve.
- If the URL specifies a static file, the origin server retrieves the file and returns the file to the WAP gateway via HTTP.
- If the URL specifies a CGI application, the origin server launches the application.
- If the origin server provides the WML contents, it returns the WML deck with the retrieved result to the WAP Gateway.
- The origin server may only provide HTML content and thus cannot understand WML, in which case, the HTTP response of the origin server is first send to an HTML filter.
- This HTML filter is typically built into the WAP Gateway, which translates the HTML text into WML format.
- The WAP Gateway verifies the HTTP header and the WML content sent from the origin server, and encodes them into WML bytecodes.
- The result is then sent to the WAP handset. (write about WML and WML script)

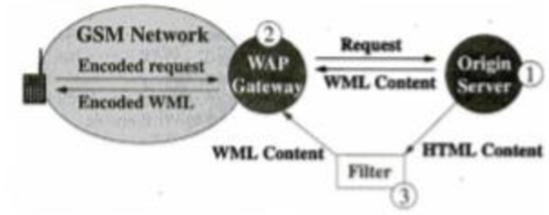


Figure 19.1 A WAP network configuration.

Mobile Location based services

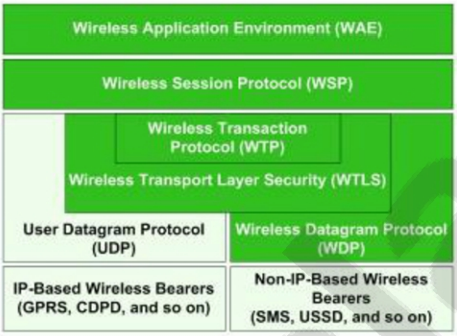
WAP Gateway:

- The WAP Gateway utilizes Web proxy technology to provide efficient wireless access to the Internet.
- A proxy plays the role of both server and client making request on behalf of the client.
- Because the WAP handset cannot directly communicate with the origin server, the WAP Gateway serves as a proxy to handle the requests from the WAP handset and passes the request to the origin servers.
- On the internet side, the WAP Gateway translates requests from the WAP protocol stack to the Internet protocol stack.
- On the wireless network side, the encoder/decoder in the WAP Gateway performs WML text and bytecode conversion to reduce the information transmitted over the wireless networks.
- WAP Gateway typically supports the DNS service to resolve the domain names used in the URLs.
- It also provides quick response to the WAP handset by aggregating data from the different origin servers, and by caching frequently used information.
- Though the WAP specifications do not specify mechanisms for charging or subscription management, the WAP architecture suggest that appropriate charging information can be collected in the WAP gateway, where the WAP security protocol can be used to authenticate the subscriber.
- WAP Gateway may use the distillation technique to perform the on-demand transformation, which effectively reduces the wireless traffic.
- Distillation is highly lossy. Real-time, datatype-specific compression that preserves most of semantic content of the document.
- It scales down a color images by reducing the number of color and thus the size of the representation or reduces he frame size, frame rate and resolution of video to create the reduced quality representation.
- The WAP Gateway is a middleware product available in the market.

- The Motorola WAP Gateway and the Ericsson WAP Gateway are based on the Windows NT platform.

WAP protocols:

- WAP specifications define a set of lightweight protocols, designed to operate over a variety of wireless bearer services.
- This services are IP-based or non-IP-based.


Wireless Application Environment (WAE)
Wireless Session Protocol (WSP)
Wireless Transaction Protocol (WTP)
Wireless Transport Layer Security (WTLS)
User Datagram Protocol (UDP)
Wireless Datagram Protocol (WDP)
IP-Based Wireless Bearers (GPRS, CDPD, and so on)
Non-IP-Based Wireless Bearers (SMS, USSD, and so on)

The WAP protocol stack is shown in figure:

1. **Wireless Datagram Protocol(WDP):**

- The WDP defines the WAP transport layer.
- WDP offers a consistent service to the upper layer protocols of WAP.
- With WDP, the higher WAP layers can function independently of the underlying wireless network.
- Because different functions are provided by the bearers, different bearer adaptations are required so that WDP services can operate over that bearer to maintain same WDP services and service primitives to the higher WAP layers.

2. **Wireless Transport Layer Security(WTLS):**

- WTLS defines the WAP security layer which optimizes the industry standards transport layer security(TLS)protocol so that it can be used in narrowband communication channels.
- WTLS supports unchanged and uncorrupted data integrity delivery with encryption, and performs authentication and denial-of-service protection.
- The WTLS features can be enabled or disabled by the applications.

3. **Wireless Transaction Protocol(WTP):**

- WTP defines the WAP transaction layer.
- WTP provides functions similar to TCP, except that WTP has reduces the amount of information needed for each transaction.
- In other words, the WTP is lighter than TCP which saves processing and memory costs in a WAP handset.
- WTP is message-oriented.
- The basic unit of interchange is an entire message instead of a stream of bytes.
- WTP supports three types of transaction: unreliable one-way requests, reliable one-way requests, and reliable two-way request-reply transactions.
- A WTP user can confirm each received message to enhance reliability.

4. **Wireless Session Protocol(WSP):**

- WSP defines the WAP session layer, which is optimized for low-bandwidth bearer networks with relatively long latency.
- WSP supports content exchange for client/server applications by establishing a session from client to server and releasing that session in orderly manner.
- The life cycle of the WSP session is not tied to the underlying transport.
- When the session is idle, WSP may suspend the session to release network resources and to save power consumption of the WAP handset.
- Two session service types are defined in WSP: a connection-oriented service that operates above WTP and a connectionless service that operates above WDP or WTP.

5. **Wireless Application Environment(WAE):**

- WAE defines the WAP application layer, which provides and environment for mobile operators and content providers to efficiently build applications on top of different wireless platforms.
- WAE defines a set of content formats, including images, phone book records etc.
- It also defines a micro browser for WML, WMLScripts, and wireless telephony applications.
- WAE also supports User Agent Profile(UAProf) and push technologies.

WAP user agent profile:

- Existing mark up language contents are designed for PCs with large displays and large memory capacities.

- Under the existing Internet technologies, WAP handsets may not be able to store and display the received contents.
- To resolve this issue, WAP specifies the User Agent Profile (UAProf) also known as Capability and Preference Information (CPI) that allows content generation to be tailored based on the WAP handsets capabilities.
- The CPI consists of information gathered from the device hardware, active user agent software, and user preferences, which include:
    - Hardware characteristics, such as screen size, color capabilities, image capabilities and manufacturer.
    - Software characteristics including operating systems vendor and version, support for MExE, and a list of audio and video encoders.
    - Application/user preferences, such as browser manufacturer and version, mark up languages and version supported and scripting languages supported.
    - WAP characteristics, including WMLScript libraries, WAP version and WML deck size.
    - Network characteristics such as device location and bearer characteristics.
- CPI is likely to be preinstalled directly on the device.
- This information is initially conveyed when a WSP session is established with the WAP Gateway.
- The WAP handset then assumes that the WAP Gateway calls the CPI and will apply it to all requests initiated during the lifetime of the WSP session.

Caching model:

- The WAP caching model tailors the HTTP caching model to support WAP handset with limited functions.
- For cached resources that will not change during user retrievals, the resources can be efficiently accessed by the WAP handset without revalidation.
- A time-sensitive cached resource is set to must-revalidate.
- If this cached resources are stale when the user tries to go back in the history, the user agent revalidation this cached source.

- In general, navigation and processing within a single cached resource does not require revalidation, except for the first fetch.
- The HTTP caching model is sensitive to time synchronization.
- Since WAP follows this model, a reliable time-of-day clock should be maintained in the WAP gateway.
- If a WAP user agent does not have access to a time-of-day clock, it should exchange the time-of-day request and response message with the WAP Gateway and synchronize with the clock value returned from the WAP Gateway.
- Another important issue for caching is security.
- The private information in the user agent cache is protected from unintended or malicious access WAP Gateways implementing a caching function must obey all security related consideration defined in HTTP.

Wireless bearers for WAP:
WML:

- The wireless mark-up language (WML) is based on the standard HTML known from the www and on HDML.
- WML is specified as an XML document type.
- When designing WML, several constraints of wireless handheld devices had to be taken into account.
- WML follows a deck and card metaphor.
- A WML document is made up of multiple cards. Cards can be grouped together into a deck.
- A WML deck is similar to an HTML page, in that it is identified by a URL and is the unit of content transmission.
- A user navigates with the WML browser through a series of WML cards, reviews the contents, enters requested data, makes choices etc.
- The WML browser fetches decks as required from origin servers.
- Either these decks can be static files on the server or they can be dynamically generated.
- It is important to note that WML does not specify how the implementation of a WML browser has to interact with a user.
- Instead, WML describes the intent of interaction in an abstract manner.
- The user agent on a handheld device has to decide how to best present all elements of a card.
- This presentation depends much on the capabilities of the device.
- WML includes several basic features:

- ○ ***Text and images:*** WML gives, as do other mark-up languages, hints how text and images can be presented to a user. WML only provides a set of mark-up elements, such as emphasis elements (bold, italic, etc.) for text, or tab columns for tabbing alignment.
- ○ ***User interaction:*** WML supports different elements for user input. Examples are: text entry controls for text or password entry, option selections or controls for task invocation. Again, the user agent is free to choose how these inputs are implemented. They could be bound to, e.g., physical keys, soft keys, or voice input.
- ○ ***Navigation:*** As with HTML browsers, WML offers a history mechanism with navigation through the browsing history, hyperlinks and other inter-card navigation elements.
- ○ ***Context management:*** WML allows for saving the state between different decks without server interaction, i.e., variable state can last longer than a single deck, and so state can be shared across different decks. Cards can have parameters defined by using this state without access to the server over the narrow-band wireless channel.

WML Scripts:

- WMLScript complements to WML and provides a general scripting capability in the WAP architecture.
- While all WML content is static (after loading on the client), WMLScript offers several capabilities not supported by WML:
    - ○ ***Validity check of user input:*** before user input is sent to a server, WMLScript can check the validity and save bandwidth and latency in case of an error. Otherwise, the server has to perform all the checks, which always includes at least one round-trip if problems occur.
    - ○ ***Access to device facilities:*** WMLScript offers functions to access hardware components and software functions of the device. On a phone a user could, e.g., make a phone call, access the address book, or send a message via the message service of the mobile phone.
    - ○ ***Local user interaction:*** Without introducing round-trip delays, WMLScript can directly and locally interact with a user, show messages or prompt for input. Only, for example, the result of several interactions could be transmitted to a server.
    - ○ ***Extensions to the device software:*** With the help of WMLScript a device can be configured and new functionality can be added

even after deployment. Users can download new software from vendors and, thus, upgrade their device easily.

- WMLScript is based on JavaScript (Flanagan, 1997), but adapted to the wireless environment.
- This includes a small memory footprint of the simple WMLScript bytecode interpreter and an efficient over the-air transport via a space efficient bytecode.
- A WMLScript compiler is used to generate this bytecode.
- This compiler may be located in a gateway or the origin servers store pre-compiled WMLScript bytecode.
- WMLScript is event-based, i.e., a script may be invoked in response to certain user or environment events.
- WMLScript also has full access to the state model of WML, i.e., WMLScript can set and read WML variables.
- WMLScript provides many features known from standard programming languages such as functions, expressions, or while, if, for, return etc. statements.
- The language is weakly-typed, i.e., any variable can contain any type (such as integer, float, string, Boolean) – no explicit typing is necessary.
- WMLScript provides an automatic conversion between different types if possible. Parameters are always passed by value to functions.

WTA:

- Browsing the web using the WML browser is only one application for a handheld vice user.
- Say a user still wants to make phone calls and access all the features of the mobile phone network as with a traditional mobile phone.
- This is where the wireless telephony application (WTA), the WTA user agent and the wireless telephony application interface.
- WTA is a collection of telephony specific extensions for call and feature control mechanisms, merging data networks and voice networks.
- The WTA framework integrates advanced telephony services using a consistent user interface (e.g., the WML browser) and allows network operators to increase accessibility for various special services in their network.
- A network operator can reach more end-devices using WTA because this is integrated in the wireless application environment (WAE) which handles device-specific characteristics and environments.

- WTA should enable third-party developers as well as network operators to create network-independent content that accesses the basic features of the bearer network.
- However, most of the WTA functionality is reserved for the network operators for security and stability reasons.
- Figure 10.30 gives an overview of the WTA logical architecture.
- The components shown are not all mandatory in this architecture; however, firewalls or other origin servers may be useful.
- A minimal configuration could be a single server from the network operator serving all clients.
- The client is connected via a mobile network with a WTA server, other telephone networks (e.g., fixed PSTN), and a WAP gateway.
- A WML user agent running on the client or on other user agents is not shown here.
- The client may have voice and data connections over the mobile network. Other origin servers within the trusted domain may be connected via the WAP gateway.
- A firewall is useful to connect third-party origin servers outside the trusted domain.

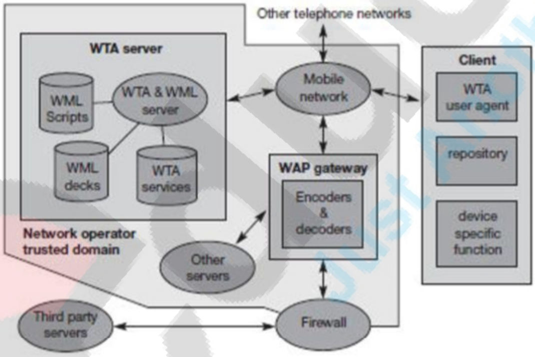


Figure 10.30
WTA logical architecture

WTA extends the basic WAE application model in several ways:

- ***Content push:*** A WTA origin server can push content, i.e., WML decks or WMLScript, to the client. A push can take place without prior client request (see sections 10.3.10 and 10.3.11). The content can enable, e.g., the client to handle new network events that were unknown before. An example is given in Figure 10.31.
- ***Access to telephony functions:*** The wireless telephony application interface (WTAI, WAP Forum, 2000m) provides many functions to

handle telephony events (call accept, call setup, change of phone book entries etc.).

- ***Repository for event handlers:*** The repository represents a persistent storage on the client for content required to offer WTA services. Content are either channels or resources. Examples for resources are WML decks, WMLScript objects, or WBMP pictures.
- ***Security model:*** Mandatory for WTA is a security model as many frauds happen with wrong phone numbers or faked services. WTA allows the client to only connect to trustworthy gateways, which then have to check if the servers providing content are authorized to send this content to the client.