Home (https://www.brainkart.com) | | **Mobile and Pervasive Computing (https://www.brainkart.com/subject/Mobile-and-Pervasive-Computing_162/)** | Wireless session protocol (WSP)

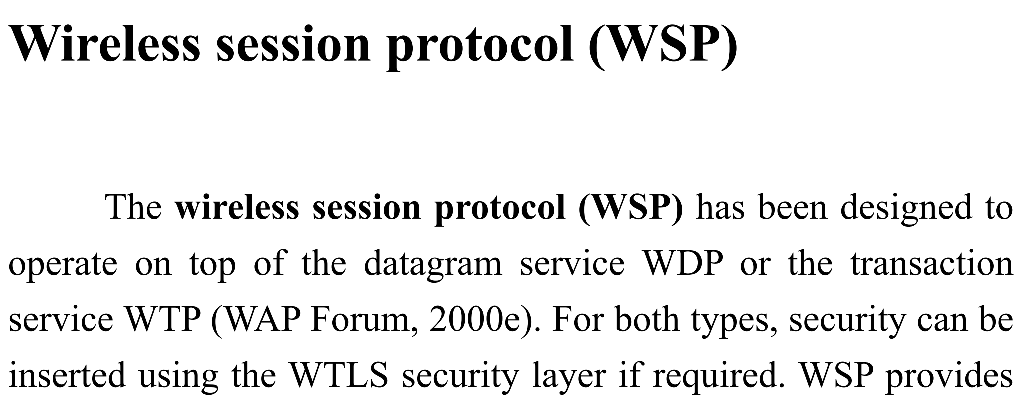
Prev Page (https://www.brainkart.com/article/Wireless-transaction-protocol-(WTP)_9951/)

Next Page (https://www.brainkart.com/article/Wireless-Application-Environment_9953/)

Chapter: **Mobile Networks : Transport and Application Layers**

Wireless session protocol (WSP)

The wireless session protocol (WSP) has been designed to operate on top of the datagram service WDP or the transaction service WTP (WAP Forum, 2000e).



Wireless session protocol (WSP)

The **wireless session protocol (WSP)** has been designed to operate on top of the datagram service WDP or the transaction service WTP (WAP Forum, 2000e). For both types, security can be inserted using the WTLS security layer if required. WSP provides a shared state between a client and a server to optimize content transfer. HTTP, a protocol WSP tries to replace within the wireless domain, is stateless, which already causes many problems in fixed networks. Many web content providers therefore use cookies to store some state on a client machine, which is not an elegant solution. State is needed in web browsing, for example, to resume browsing in exactly the same context in which browsing has been suspended. This is an important feature for clients and servers. Client users can continue to work where they left the browser or when the network was interrupted, or users can get their customized environment every time they start the browser. Content providers can customize their pages to clients' needs and do not have to retransmit the same pages over and over again. WSP offers the following general features needed for content exchange between cooperating clients and servers:

- **Session management:** WSP introduces sessions that can be **established** from a client to a server and may be long lived. Sessions can also be **released** in an orderly manner. The capabilities of **suspending** and **resuming** a session are important to mobile applications. Assume a mobile device is being switched off – it would be useful for a user to be able to continue operation at exactly the point where the device was switched off. Session lifetime is independent of transport connection lifetime or continuous operation of a bearer network.

- **Capability negotiation:** Clients and servers can agree upon a common level of protocol functionality during session establishment. Example parameters to negotiate are maximum client SDU size, maximum outstanding requests, protocol options, and server SDU size.

c) **Content encoding:** WSP also defines the efficient binary encoding for the content it transfers. WSP offers content typing and composite objects, as explained for web browsing. While WSP is a general-purpose session protocol, WAP has specified the **wireless session protocol/browsing (WSP/B)** which comprises protocols and services most suited for browsing-type applications. In addition to the general

features of WSP, WSP/B offers the following features adapted to web browsing:

d) **HTTP/1.1 functionality:** WSP/B supports the functions HTTP/1.1 offers, such as extensible request/reply methods, composite objects, and content type negotiation. WSP/B is a binary form of HTTP/1.1. HTTP/1.1 content headers are used to define content type, character set encoding, languages etc., but binary encodings are defined for well-known headers to reduce protocol overheads.

e) **Exchange of session headers:** Client and server can exchange request/reply headers that remain constant over the lifetime of the session. These headers may include content types, character sets, languages, device capabilities, and other static parameters. WSP/B will not interpret header information

but passes all headers directly to service users.

f) **Push and pull data transfer:** Pulling data from a server is the traditional mechanism of the web. This is also supported by WSP/B using the request/response mechanism from HTTP/1.1. Additionally, WSP/B supports three push mechanisms for data transfer: a confirmed data push within an existing session context, a non-confirmed data push within an existing session context, and a non-confirmed data push without an existing session context.

g) **Asynchronous requests:** Optionally, WSP/B supports a client that can send multiple requests to a server simultaneously. This improves efficiency for the requests and replies can now be coalesced into fewer messages. Latency is also improved, as each result can be sent to the client as soon as it is available.

As already mentioned, WSP/B can run over the transaction service WTP or the datagram service WDP. The following shows several protocol sequences typical for session management, method invocation, and push services.

⏪ Prev Page (https://www.brainkart.com/article/Wireless-transaction-protocol-(WTP)_9951/)

Next Page ⏩ (https://www.brainkart.com/article/Wireless-Application-Environment_9953/)

⏪ Prev Page (https://www.brainkart.com/article/Wireless-transaction-protocol-(WTP)_9951/)

Next Page ⏩ (https://www.brainkart.com/article/Wireless-Application-Environment_9953/)

Related Topics

Mobile and Pervasive Computing (https://www.brainkart.com/subject/Mobile-and-Pervasive-Computing_162/)

Mobile Networks : Transport and Application Layers

Mobile TCP and its Advantages (https://www.brainkart.com/article/Mobile-TCP-and-its-Advantages_9946/)

Wireless Application Protocol (https://www.brainkart.com/article/Wireless-Application-Protocol_9947/)

WAP-Architecture (https://www.brainkart.com/article/WAP-Architecture_9948/)

Wireless Datagram Protocol (WDP) (https://www.brainkart.com/article/Wireless-Datagram-Protocol-(WDP)_9949/)

Wireless Transport Layer Security (WTLS) (https://www.brainkart.com/article/Wireless-Transport-Layer-Security-(WTLS)_9950/)

Wireless transaction protocol (WTP) (https://www.brainkart.com/article/Wireless-transaction-protocol-(WTP)_9951/)

Wireless session protocol (WSP) (https://www.brainkart.com/article/Wireless-session-protocol-(WSP)_9952/)

Wireless Application Environment (https://www.brainkart.com/article/Wireless-Application-Environment_9953/)

WTA Architecture (https://www.brainkart.com/article/WTA-Architecture_9954/)

Wireless markup language (WML) (https://www.brainkart.com/article/Wireless-markup-language-(WML)_9955/)

WMLScript (https://www.brainkart.com/article/WMLScript_9956/)