WV-044 Client-Server Protocol Transport Bindings

Approved Version 1.2 – 25 Jan 2005

Open Mobile Alliance

OMA-IMPS-WV-CSP-Transport-V1_2-20050125-A

Continues the Technical Activities
Originated in the Wireless Village Initiative

Contents

1. Scope

The Wireless Village Instant Messaging and Presence Service (IMPS) includes four primary features:

- Presence
- Instant Messaging
- Groups
- Shared Content

Presence is the key enabling technology for IMPS. It includes client device availability (my phone is on/off, in a call), user status (available, unavailable, in a meeting), location, client device capabilities (voice, text, GPRS, multimedia) and searchable personal statuses such as mood (happy, angry) and hobbies (football, fishing, computing, dancing). Since presence information is personal, it is only made available according to the user's wishes - access control features put the control of the user presence information in the users' hands.

Instant Messaging (IM) is a familiar concept in both the mobile and desktop worlds. Desktop IM clients, two-way SMS and two-way paging are all forms of Instant Messaging. Wireless Village IM will enable interoperable mobile IM in concert with other innovative features to provide an enhanced user experience.

Groups or chat are a fun and familiar concept on the Internet. Both operators and end-users are able to create and manage groups. Users can invite their friends and family to chat in group discussions. Operators can build common interest groups where end-users can meet each other online.

Shared Content allows users and operators to setup their own storage area where they can post pictures, music and other multimedia content while enabling the sharing with other individuals and groups in an IM or chat session.

These features, taken in part or as a whole, provide the basis for innovative new services that build upon a common interoperable framework.

2. References

2.1 Normative References

[CSP SCR] "WV-048 Client-Server Protocol Static Conformance Requirement Version 1.2". Open Mobile Alliance.
URL: http://www.openmobilealliance.org

[IOPPROC] "OMA Interoperability Policy and Process", Version 1.1, Open Mobile Alliance™, OMA-IOP-Process-V1_1, URL:http://www.openmobilealliance.org

[JSR120] "Wireless Messaging API for Java 2 Micro Edition". Java Community Process. August 2002. URL: http://jcp.org/aboutJava/communityprocess/final/jsr120/index.html

[RFC1928] SOCKS Protocol Version 5. M. Leech, M. Ganis, Y. Lee, R. Kuris, D. Koblas, L. Jones. RFC 1928. March 1996. URL: http://www.ietf.org/rfc/rfc1928.txt

[RFC2119] "Key words for use in RFCs to Indicate Requirement Levels". S. Bradner. March 1997. URL: http://www.ietf.org/rfc/rfc2119.txt

[RFC2234] "Augmented BNF for Syntax Specifications: ABNF". D. Crocker, Ed., P. Overell. November 1997. URL: http://www.ietf.org/rfc/rfc2234.txt

[RFC2616] "Hypertext Transfer Protocol – HTTP/1.1". Fielding R.; Gettys J.; Mogul J.; Frystyk H.; Masinter L.; Leach P.; Berners-Lee T., June 1999. URL: http://www.ietf.org/rfc/rfc2616.txt

[SSP SCR] "WV-055 SSP – Server-Server Protocol Static Conformance Requirement Version 1.2". Open Mobile Alliance.
URL: http://www.openmobilealliance.org

[TCPTunnel] Luotonen, A., "Tunneling TCP based protocols through Web proxy servers" URL: http://www.web-cache.com/Writings/Internet-Drafts/draft-luotonen-web-proxy-tunneling-01.txt

[TIAEIA-637] "ANSI/TIA/EIA-637-B: Short Message Service for Wideband Spread Spectrum Systems", 2002. URL: http://global.ihs.com/search_res.cfm?RID=TIA&INPUT_DOC_NUMBER=TIA%2FEIA%2D637

[TS 23.038] "Alphabets and Language-specific Information (Release 5), 3GPP TS 23.038 v5.0.0", 3rd Generation Partnership Project, March 2002. URL: ftp://ftp.3gpp.org/Specs/archive/23_series/23.038/23038-500.zip

[TS 23.040] "Technical Realization of the Short Message Service (Release 5), 3GPP TS 23.040 v5.4.0", 3rd Generation Partnership Project, June 2002. URL: ftp://ftp.3gpp.org/Specs/archive/23_series/23.040/23040-540.zip

[TS 24.011] "Point-to-Point (PP) Short Message Service (SMS) Support on Mobile Radio Interface (Release 5), 3GPP TS 24.011 v5.1.0", 3rd Generation Partnership Project, December 2002. URL: ftp://ftp.3gpp.org/Specs/archive/24_series/24.011/24011-510.zip

[WAPWDP] "Wireless Datagram Protocol, version 14-Jun-2001", WAP Forum, June 2001. URL: http://www.openmobilealliance.org

[WAPWSP] "Wireless Session Protocol Specification, Version 05-July-2001", WAP Forum, July 2001. URL: http://www.openmobilealliance.org

[WAPPush] http://www.openmobilealliance.org

2.2 Informative References

[Arch] "WV-040 System Architecture Model Version 1.2". Open Mobile Alliance.
URL: http://www.openmobilealliance.org

[FeaFun] "WV-041 Features and Functions Version 1.2". Open Mobile Alliance.
URL: http://www.openmobilealliance.org

[CSP] "WV-042 Client-Server Protocol Session and Transactions Version 1.2". Open Mobile Alliance.
URL: http://www.openmobilealliance.org

[CSP DTD] "WV-043 Client-Server Protocol DTD and Examples Version 1.2". Open Mobile Alliance.
URL: http://www.openmobilealliance.org

[CSP Trans] "WV-044 Client-Server Protocol Transport Bindings Version 1.2". Open Mobile Alliance.
URL: http://www.openmobilealliance.org

[CSP DataType] "WV-045 Client-Server Protocol Data Types Version 1.2". Open Mobile Alliance.
URL: http://www.openmobilealliance.org

[CSP SMS] "WV-046 Client-Server Protocol SMS Binding Version 1.2". Open Mobile Alliance.
URL: http://www.openmobilealliance.org

[CSP WBXML] "WV-047 Client-Server Protocol Binary Definition and Examples Version 1.2". Open Mobile Alliance.
URL: http://www.openmobilealliance.org

[CSP SCR] "WV-048 Client-Server Protocol Static Conformance Requirement Version 1.2". Open Mobile Alliance.
URL: http://www.openmobilealliance.org

[PA] "WV-049 Presence Attributes Version 1.2". Open Mobile Alliance.
URL: http://www.openmobilealliance.org

[PA DTD] "WV-050 Presence Attribute DTD and Examples Version 1.2". Open Mobile Alliance.
URL: http://www.openmobilealliance.org

[CLP] "WV-051 Command Line Protocol Version 1.2". Open Mobile Alliance.
URL: http://www.openmobilealliance.org

[SSP] "WV-052 SSP - Server-Server Protocol Semantics Document Version 1.2". Open Mobile Alliance.
URL: http://www.openmobilealliance.org

[SSP Syntax] "WV-053 Server-Server Protocol XML Syntax Document Version 1.2". Open Mobile Alliance.
URL: http://www.openmobilealliance.org

[SSP Trans] "WV-054 SSP - Transport Binding Version 1.2". Open Mobile Alliance.
URL: http://www.openmobilealliance.org

[SSP SCR] "WV-055 SSP – Server-Server Protocol Static Conformance Requirement Version 1.2". Open Mobile Alliance.
URL: http://www.openmobilealliance.org

[WAPARCH] “WAP Architecture, Version 12-July-2001”. Open Mobile Alliance™. WAP-210-WAPArch.
URL: http://www.openmobilealliance.org

3. Terminology and Conventions

3.1 Conventions

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

All sections and appendixes, except “Scope” and “Introduction”, are normative, unless they are explicitly indicated to be informative.

3.2 Definitions

None.

3.3 Abbreviations

CIR	Communications Initiation Request
CSP	Client-Server Protocol
HTTP	Hypertext Transfer Protocol
SMS	Short Message Service
WAP	Wireless Application Protocol
WSP	Wireless Session Protocol
WV	Wireless Village

4. Introduction

This document describes the binding of the session and transactions to different transports. This document describes four bindings: WSP [WAPWSP], HTTP [RFC2616], HTTPS and SMS [TS 23.040]. In addition, it defines the use of a Communications Initiation Request (CIR) used to initiate communication process between server and clients.

A WV client and server *MUST* support at least one transport binding, either WSP or HTTP or HTTPS or SMS. The server support for WSP can be implemented by using a WAP Gateway in front of the server and then use HTTP communication between the WAP Gateway and the WV server. The CIR channel is mandatory for all data channel transport bindings except the SMS binding.

5. Logical Model of Communications

Logically the WV transport binding is divided into two channels: a mandatory *data channel* in which all the exchange of CSP primitives is done and a conditional *CIR channel* used to activate the data channel whenever the data channel is not established, or the communication is halted in the data channel and needs to be reactivated. Both channels are depicted on Figure 1.

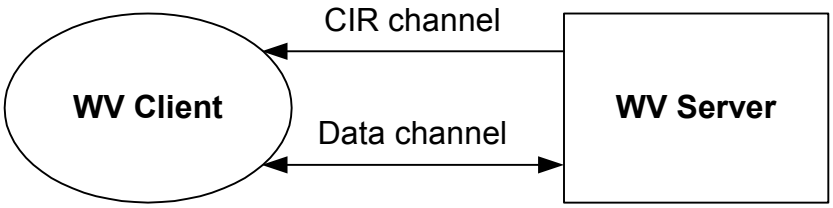


Figure 1. Logical Model of Communications

The need and use of a CIR channel depends on the protocol and bearer used in data channel. The protocol bindings in data channel are WSP, HTTP, HTTPS and SMS. In case of WSP, HTTP and HTTPS, the communication is asymmetric, i.e., it always originates from WV client to the server. Thus, the client can always start a transaction from the client to the server. If the WV server needs to start a transaction, there are two alternatives:

- The server inserts the transaction request into a response message for a pending transaction from the client to the server
- The server sends a communication initiation request message through the CIR channel to the client in order to request an immediate CSP PollingRequest message from the client to the server on the data channel. The transaction request is then inserted into the response part of the poll request.

In addition to the use described above, the CIR channel is also used to establish the data channel when the channel is not available. For instance, if a TCP/IP connection for the data channel has been disconnected, or the PDP context in 2.5G or 3G mobile networks is not allocated, the CIR channel is used to reestablish the channel connection to the server.

In the SMS technology, both the client and server can originate transactions and the data channel is always available. Thus, separate CIR channel is not needed.

6. WV Session and Channel MANAGEMENT

The WV session and transaction models are independent of the WV transport binding and the underlying bearer protocols. The WV session does not require persistent underlying bearer for the data channel. The TCP/IP connection or WSP session MAY be disconnected during the session for performance reasons or it MAY be lost for some other reason. If disconnected, the client reestablishes the connection when it needs to send a request or when it receives the CIR.

Servers MAY support the usage of different bearers for the data channel within a single session due to the independence of the session and transaction models from the WV transport binding. Clients MUST however not depend on this behaviour since it MAY NOT be supported by the server or disabled by the operator due to security reasons. Supporting multiple bearers is beneficial for mobile clients that MAY loose connectivity over a preferred bearer while maintaining connectivity over alternative bearers.

The CIR channel is either connectionless, or connection-oriented, or based on polling. If the channel is connection-oriented, the connection needs to be persistent (for the duration of a session). However, if the server discovers that the CIR channel over standalone TCP/IP is disconnected, the server MAY notify the client that the CIR channel is disconnected by setting the CIR flag 'F' in an http POST response (see 7.2 for the details of WSP/HTTP encapsulation of CSP transactions).

The relation of the channel connections and the WV session is illustrated in Figure 2.

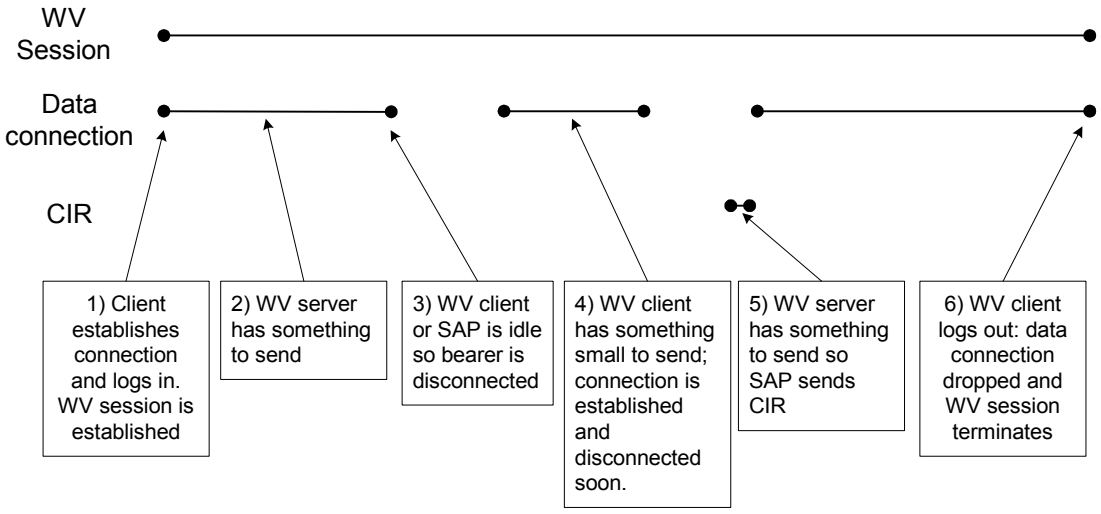


Figure 2. The relation of the WV session and bearer.

7. Transport Binding for WSP/HTTP/HTTPS Data Channels

7.1 Overview

The CSP transport binding alternatives for data channel are HTTP 1.1, HTTPS, WSP 1.2 or WSP 2.0. In HTTP and HTTPS binding, the bearer protocol in data channel is TCP/IP. For WSP bindings, the bearer protocol alternatives are described in WAP specifications. There is no requirement for persistent bearer connection.

The bearer connection for the data channel is always set up from the WV client to the WV SAP.

7.2 WSP/HTTP Encapsulation of CSP Transactions

The WSP and HTTP(S) are both asymmetric, client-server protocols, in which requests always originate from the client and responses from the server. In WV transactions, however, there is a need for symmetric transactions: the requests MAY originate from client or server.

The encapsulation of symmetric CSP transactions to asymmetric WSP/HTTP(S) methods is based on the use of WSP/HTTP(S) POST method only. The WSP/HTTP(S) POST-request, POST-response and the CSP transactions are completely separated. Each WSP/HTTP(S) POST-request MAY contain at least one CSP transaction request or CSP transaction response message. Similarly, each WSP/HTTP(S) POST-reply MAY contain at least one CSP transaction request or CSP transaction response message.

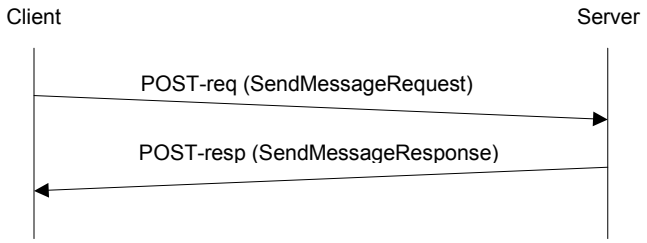
If the WV client supports more than one CSP request or response within single HTTP(S) POST request or response, it is indicated during the client capabilities negotiation procedure.

For each transaction at POST-reply, the WV server indicates whether (for server reasons) the next POST request is needed. If it is needed, but the WV client has nothing to send, the client SHALL WSP/HTTP(S) POST request with CSP PollingRequest primitive as content. Similarly, if server has no WV transaction request or reply to be sent to the WV client, the WSP/HTTP(S) POST response SHOULD contain no content and the 200 OK response code.

This communication continues until neither WV client nor the server has CSP primitives to send. In such a case, the communication grinds to a halt. If, at this point, the WV client has something to send, it simply issues a WSP/HTTP(S) POST with the CSP transaction request.

If server needs to send any data (CSP request or response) to the particular client, first the server has to send a Communications Initiation Request to that client which is a signal to the WV client to initiate WSP/HTTP(S) POST with CSP PollRequest primitive as content.

Examples of the mapping of the message flow for the CSP SendMessageRequest and MessageNotification transactions are depicted on Figure 3.



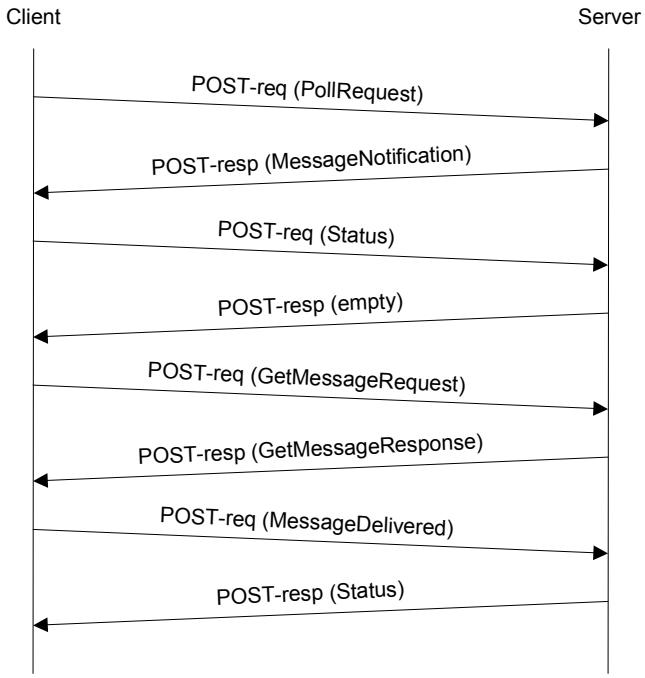


Figure 3. Examples of mapping of WV CSP transactions to WSP/HTTP(S)

7.3 Access Point Definition

For WAP/WSP bindings, the access point towards WV SAP requires the WAP access point and the URL of the WV SAP.

For HTTP(S) binding, the access point towards WV SAP requires the general ISP access point for TCP/IP and the URL of the WV SAP.

The URL used in WAP/WSP and HTTP(S) bindings MAY contain a non-empty path and a non-standard port number.

7.4 Redirection

The WV client that uses HTTP binding MUST understand standard HTTP redirection codes [RFC2616] and associated information headers. HTTP redirection mechanism allows a WV server to redirect clients to other servers or other parts of the same server based on load or session related information.

HTTP Redirect [RFC2616] indicates that such redirect code as 301 (Permanent), 302 (Found) and 307 (Temporary) MUST NOT automatically redirect without being confirmed by the user. It is recognized that such interactions in the UI will give rise to a poor user experience in a real product. HTTP Redirect in WV HTTP Binding SHOULD be supported in the following ways:

- The client MAY automatically perform the redirect action without further user confirmation.
- Only two types of redirect are allowed in the client server communication:
  - Permanent Redirection (301): The server address is redirected forever. The client MAY save the redirected server address to flash and reuse it in the future transactions and sessions if the client supports persistent storage.
  - Temporary Session Redirection (307): The server address is redirected for this session. The client uses the redirected server address only in the context of current IMPS session. The purpose of this code would be so that the server MAY change the redirect address for different sessions, however, the client is not being redirected on every transaction. The client MAY save the redirected server address to flash and reuse it in the future transactions during this session if the client supports persistent storage.

- The caching headers, which are specified in the HTTP [RFC2616] when redirecting, SHALL be ignored by the client. The server SHOULD not enclose those caching headers when redirecting.

7.5 HTTP Headers

All headers MUST conform to [RFC2616].

7.6 Error Handling

Support for the standard HTTP error responses is mandatory for both client and server implementations.

When a client or server implementation receives an error response in a POST-reply, it SHALL:

- assume that all of the transaction requests in the corresponding POST-request are invalid; and
- make no assumptions about the HTTP headers, content type and content.

When a client or server implementation generates an HTTP error response, it MAY set the HTTP headers, content type and content to any appropriate value.

A client or server implementation that detects an error in the received XML data in a POST-request SHALL reply with an HTTP error response with code 400, BAD_REQUEST.

8. Transport Binding for CIR Channel

8.1 Transport Alternatives and Message Format

The CIR channel is a push-type channel that can be implemented as connectionless or connection-oriented channel. The purpose of the CIR channel is to carry communication initiation requests from the server to the client only. It does not carry any CSP primitives.

For the CIR channel, the following bindings are defined:

- WAP 1.2 or WAP 2.0 push using WSP unit push message and SMS as a bearer
- WAP 1.2 or WAP 2.0 push using WSP unit push message and UDP/IP as a bearer
- Standalone SMS binding
- Standalone UDP/IP binding
- Standalone TCP/IP binding
- Standalone HTTP binding

The WV client MAY support one or more CIR channel bindings that are indicated in the client capability negotiation defined in CSP.

If CIR channel is connection-oriented, the connection of CIR channel and data channel are independent of each other.

In general, the bindings of CIR channel and data channel are independent. However, if the binding for data channel is WSP, the CIR channel binding MUST use WAP 1.2/2.0 CIR bindings.

The communication initiation request message is textual message in the following format:

- WVCI <CSP-version> <Session-cookie>
- Where:
  - CSP-version is the version number of the WV specification. Major version and minor version numbers are separated by the dot (“.”).
  - *Session-cookie* is the client-defined session cookie generated at every client login.

The encoding is UTF-8 Unicode Basic Latin (US-ASCII) by default, unless it is specified otherwise in the specific binding section(s).

The HTTP binding CIR Channel does not use a textual CIR message. Instead, the HTTP reply code is used to trigger the communication on the data channel.

8.1.1 WAP Push Binding

To be able initiate a WAP Push request, the WV server MUST be provisioned with an address of WAP PPG and support WAP Push Access Protocol. The WV server uses the push submission operation to send CIR to the terminal. Each push message SHOULD contain one CIR. Content type of the content entity of a PAP request is “application/vnd.wv.csp.cir”.

The use of WAP Push does not require that the WV client has active PDP context. The Push Proxy Gateway MAY use a SMS bearer to send the initiation request or, if a PDP context is already active and the IP address is known, it MAY push the message over TCP or UDP.

The WV client in a mobile handset MAY provide its mobile number in the CSP protocol login transaction (as a part of Client ID). If the mobile number is not present, the WV SAP MUST be able to obtain the mobile number if it is required.

8.1.2 Standalone UDP/IP Binding

In the case of a standalone UDP/IP binding, the WV server sends the client the CIR messages enclosed in UDP datagrams. Each UDP datagram contains exactly one CIR message. To use this binding, WV client MUST be able to receive UDP datagrams directly from the WV server.

The WV client MAY accept the CIR request either to default UDP port defined in this document or to provide the UDP port in the capability negotiation phase of client login.

Due to the small size of the CIR message, it is guaranteed that the UDP will not be fragmented or rejected because of size.

8.1.3 Standalone TCP/IP Binding

TCP/IP binding uses a persistent connection from the WV client to server to provide a low-latency always-on CIR channel.

The WV client is responsible for setting up the TCP/IP connection and maintaining its persistency.

The WV client opens the CIR TCP/IP connection to the server right after a successful login procedure including client capability and service negotiation. The IP address and port for the CIR channel are provided by the server in the capability negotiation. The IP address and port are valid throughout the session.

As soon as a connection opens, the client MUST send the authentication message “HELO” with Session ID as a parameter. This allows the WV server to associate the new TCP/IP connection with one of the existing sessions. If the WV server does not receive a “HELO” message in 10 seconds after a new connection has been opened from the client or the received Session ID is unknown, the server MUST terminate the connection. The WV server replies to the client’s “HELO” message with an “OK” message. The client is not allowed to open more than one connection to the WV server.

In some cases a TCP/IP connection MAY be closed by the intermediate network entities, or a connection MAY be broken due to network problems. To prevent this from happening or to be able to recover, the WV client SHOULD periodically send “PING” messages over an opened connection to determine if it is still available. The server MUST respond to these messages with the “OK” message. If client doesn’t receive an “OK” message or detects that the connection is broken, it MUST open a new TCP/IP connection and send the “HELO” message again.

When a server has any data (CSP request or response) that needs to be sent to the client, it sends a CIR message over the TCP/IP connection associated with this client.

All client and server originated messages MUST be terminated with a <CR><LF> (carriage return, line feed) sequence.

The encoding is UTF-8 Unicode Basic Latin (US-ASCII)

The connection establishment for a standalone TCP/IP binding for CIR channel MAY not work directly when the WV client is behind a firewall or proxy. The technology alternatives to facilitate the connection initiation and management are:

HTTP Tunnelling [TCPTunnel]

SOCK4

SOCK5

An example of data traffic on the TCP/IP-based CIR channel (“C→S” indicates client originated messages, “S→C” indicates server originated messages) is:

<client opened TCP/IP connection to the server>

C→S: HELO abcd123

S→C: OK

S→C: WVCI 1.2 cookie123

C→S: PING

S→C: OK

<client closes TCP/IP connection>

8.1.4 Standalone SMS Binding for CIR

The standalone SMS binding for the CIR channel uses either GSM short message or CDMA IS-637 short message technology to facilitate the CIR channel. The WV client and the short message service center MUST support both mobile-originated (MO) and mobile-terminated (MT) short messages.

The standalone SMS binding for the CIR channel supports both GSM SMS [TS 23.040] and CDMA SMS IS-637 [TIAEIA-637]. Each SMS message SHALL contain exactly one CIR message. The encoding of SMS messages for CIR SHALL be the GSM 7-bit default alphabet defined in [TS 23.038].

After a successful complete login procedure that includes client capability and service negotiation, the WV client MUST send an SMS-MO to the server comprising the message “HELO” with the Session-ID as a parameter, as defined in section 8.1.3. The SMS-MO carrying the "HELO" message allows the server to discover the client's MSISDN. The WV client SHALL not periodically send a “PING” message over the CIR channel. The encoding of the “HELO” message SHALL be the GSM 7-bit default alphabet [TS 23.038].

The WV client SHALL accept the CIR request through the standalone SMS CIR port defined in section 10.5. However, the CDMA IS-637 SMS does not include a port number or any other field for differentiation between recipient applications. For this purpose, the WAP WDP for IS-637 SMS, which is defined in section 6.5 of WDP specification [WAPWDP], MUST be used.

In order for the SMS-based CIR channel to work properly through a J2ME platform, the guidelines defined in J2ME Wireless Messaging API [JSR120] MUST be followed.

8.1.5 Standalone HTTP Binding

The HTTP binding is used by clients that cannot establish any other CIR channel. The client periodically polls on the CIR channel for a CIR trigger. When a CIR trigger is received, the client performs a CSP PollingRequest transaction to enable the server-initiated transaction in the same way as for the other CIR channels.

Polling is very resource consuming when it comes to bandwidth and server load. To minimize this overhead, the periodic polling is only done for CIR with a minimal HTTP GET on a non-persistent HTTP connection.

The URL to be used for the CIR poll is provided by the server in the capability negotiation. The format of the address is such as the server can identify the session but for security reasons the actual Session ID SHOULD never be revealed in the HTTP binding. Example of poll URL:

```
MyServiceProvider.com/poll?pc=1234567
```

The ‘pc=123456’ is a poll cookie that the server generates and internally uses to map to the real session.

The URL is valid throughout the session and the client closes the HTTP connection after each poll. The HTTP binding always uses HTTP even if the data channel uses HTTPS.

The WV client is responsible for setting up the HTTP connection and to do the polling. The minimum time between two polls is given by the server during client capability negotiation. The WV client starts to poll after successfully logs in and has completed appropriate client capability and service negotiation. .

It is RECOMMENDED that clients implement an adaptive polling policy. Normally, when polling requests return a CIR trigger to do a CSP PollingRequest Transaction, the client needs to initiate the next polling request after the minimum interval. However, in the case of empty responses, it SHOULD gradually increase the polling intervals (up to 10 seconds or more). This significantly decreases the server load and reduces unnecessary network traffic.

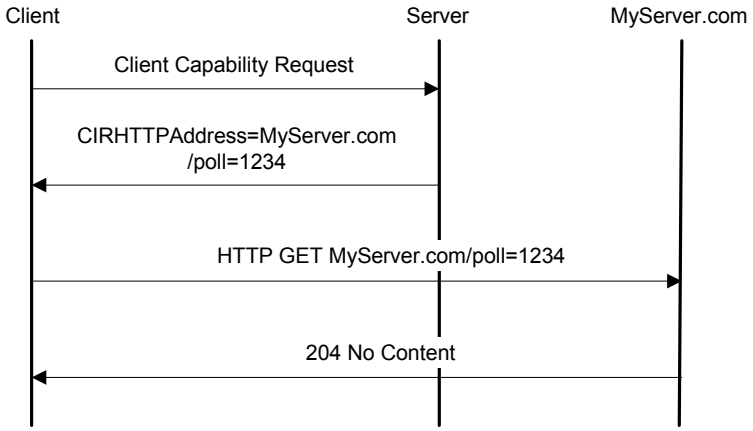


Figure 4**. HTTP binding for CIR channel – the server does not have any queued requests or responses.**

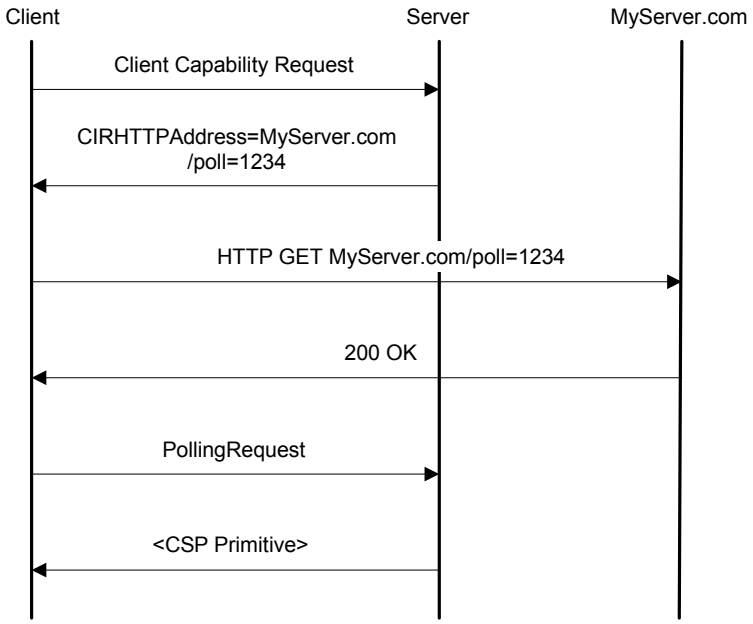


Figure 5**. HTTP binding for CIR channel – the server has some queued requests or responses.**

The Client uses the HTTP GET method to poll on the HTTP binding.

If CSP messages are queued on the server, the server replies with a '200 OK'. The reply substitutes for the textual CIR message that is used in other CIR channel bindings. The client then needs to perform a CSP PollRequest Transaction to retrieve the server initiated transaction.

If there are no queued CSP messages on the server, the server replies with '204 No Content'.

9. SMS TRANSPORT for Mobile Clients

The SMS transport bindings use the GSM short message technology to facilitate the WV transactions. In the transport binding, the WV client and the short message service center MUST support both mobile-originating (MO) short messages as well as mobile-terminating (MT) short messages. Due to the symmetric nature of SMS transport, the CIR channel is not needed.

The message encoding for the SMS binding is based on the SMS binding document.

9.1 Overview

In the SMS transport binding, the WV client communicates with the WV SAP through a SMSC. The CSP transactions and session document [CSP] as well as and the relevant SMS binding document [CSP SMS] describe the SMS application level communication.

The SMSC MUST be able to route the messages from the WV client to the WV SAP. For this purpose, the short message is sent to a recipient that identifies the WV SAP as a special, IN-type number. The SMSC MUST have the capability to route messages using this special number to the WV SAP. When the WV SAP sends a message to the WV client, the SMSC is able to deliver the message directly when the recipient is identified with a mobile number. The architecture is depicted in Figure 4.

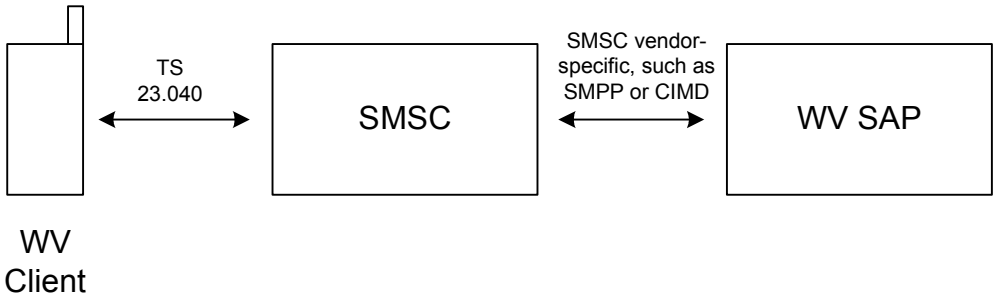


Figure 6. Architecture for SMS transport binding

The protocol between SMSC and WV SAP is one of the SMSC-vendor specific protocols, such as SMPP or CIMD.

The WV SAP is addressed by the recipient address in short message as encoded in TP-DA field in SMS-SUBMIT TPDU [TS 23.040]. The SMSC MUST be able to recognize this number and route it towards the WV SAP. The receiving SMSC is addressed in the RP-DA field in RP-DATA RPDU [TS 24.011].

9.2 Access point to WV SAP

The access point definition towards WV SAP requires normal SMSC access point definition as well as the special IN-type number identifying the WV SAP.

10.Registered Identifiers

10.1 Message Type

The WV message types for textual and binary XML, SMS Bindings as well as for CIR content type are registered through IANA.

Proposal for registration, as well as type used for experimental purposes SHALL be:

application/vnd.wv.csp.xml

application/vnd.wv.csp.wbxml

application/vnd.wv.csp.cir (for PAP push submission)

application/vnd.wv.csp.sms

10.2 WAP Push Application Id

The push application id is registered from the WAP Forum WINA registry.

For experimental purposes, the push application-id SHALL be 56731.

10.3 Port Number for Standalone UDP/IP CIR Channel

The port numbers for Standalone UDP/IP CIR channel will be registered through IANA.

For experimental purposes the default port number for UDP/IP binding is 56732

10.4 Port Number for SMS Binding

The port number for SMS binding will be registered through IANA

For experimental purposes the default port number is 56733.

10.5 Port Number for Standalone SMS CIR Channel

The port number for Standalone SMS CIR channel will be registered through IANA.

For experimental purposes the default port number for standalone SMS CIR channel binding is 56734.

11. Static Conformance Requirements for Transport Bindings

Req#	Description	C-Req	S-Req	Reference
TRANSP-1	Support for transport binding for data channel	M	M	
TRANSP-2	Support for transport binding for CIR channel. Not applicable if only TRANS-7, else mandatory.	C	C	
TRANSP-3	Support for HTTP binding in data channel	O	O	
TRANSP-4	Support for HTTP/S binding in data channel	O	O	
TRANSP-5	Support for WSP 1.2 binding in data channel	O	O	
TRANSP-6	Support for WSP 2.0 binding in data channel	O	O	
TRANSP-7	Support for SMS binding in data channel	O	O	
TRANSP-8	Support for WAP push SMS binding in CIR channel	O	O	
TRANSP-9	Support for WAP push UDP/IP binding in CIR channel	O	O	
TRANSP-10	Support for standalone UDP/IP binding in CIR channel	O	O	
TRANSP-11	Support for standalone TCP/IP binding in CIR channel.	O	O	
TRANSP-12	With WSP 1.2 or WSP 2.0 bindings for data channel, only WAP SMS binding or WAP UDP binding is used in CIR channel.	M	M	
TRANSP-13	Sending of Poll request when poll request is received in WV message inside the WSP/HTTP(S) POST response.	M	N/A	
TRANSP-14	Sending of Poll request when CIR is received	M	N/A	
TRANSP-15	Support standalone SMS binding for CIR channel	O	O	
TRANSP-16	If the server discovers that the CIR channel over standalone TCP/IP is disconnected, the server-originated primitive contains the CIR element with the value ‘F’ to notify the client.	N/A	O	
TRANSP-17	If a session has expired, and server originated notifications are delivered to the client through a Poll request, the server delivers a Disconnect notification to the client before closing any session related resources.	N/A	O	
TRANSP-18	Support for Standalone HTTP binding in CIR channel.	O	O	

Appendix A. Change History (Informative)

A.1 Approved Version History

Reference	Date	Description
OMA-WV-CSP_Transport-V1_1-20021001-A	01 Oct 2002	Version 1.1
OMA-IMPS-WV-CSP-Transport-V1_2-20050125-A	25 Jan 2005	Version 1.2 Ref TP Doc# OMA-TP-2004-0457-IMPS-V1_2-for-final-approval