



Masoud Aghadavoodi Jolfaei

July 27, 2015 3 minute read

Specification of the Push Channel Protocol (PCP)

[Follow](#)

[RSS feed](#)

2 Likes 2,495 Views 5 Comments

The Push Channel Protocol is a message based protocol, which is used in context of [ABAP Channels](#), i.e. [ABAP Messaging Channels](#) and [ABAP Push Channels](#). The message structure is very similar to a simple HTTP message (see RFC2616) without having the start-line (request-line, response-line) entry. In other words the PCP messages consist of (header) fields and a plain body which is separated by line feeds (LF, i.e. \n – 0x0A (hexadecimal) – 10 (decimal)). A field entry contains name-value pair separated by “:” character, i.e. <name>:<value>. The names and values are in UTF-8 characters and case-sensitive. The names and values should not contain the special characters LF, “:” and “\”. In that case the special characters must be escaped with the escape character “\” whereas LF is replaced by “\n”. Each field entry, i.e. <name>:<value>, is terminated by a line feed. The last field entry is completed with double LF, i.e. LFLF, independently of the existence of a body. In case of the existence of a body this content is appended after the double LFs. The body contains only UTF8 characters. A body in binary is encoded in Base64 format. Additionally field name or values must not contain leading or trailing spaces, i.e. white spaces. The field names starting with the prefix “pcp-” are reserved for PCP library and must not be used by application.

For the usage of the PCP protocol in context of ABAP Push Channel, i.e. the WebSocket support in ABAP Application Server, the WebSocket subprotocol “v10.pcp.sp.com”, i.e. version 1.0 of the PCP protocol, is to

be used.

Augmented Backus-Naur Form (BNF)

A Push Channel Protocol (PCP) message can be more formally described using the Backus-Naur Form (BNF) grammar used in HTTP/1.1 **RFC 2616**. Figure 1 shows the structure of a PCP message.

```
LF           = <US-ASCII LF, linefeed (decimal 10, hexadecimal 0A)>
escaping-char = "\" used to escape special-char, e.g. LF, ":" and "\"
pcp-message  = pcp-fields LF
               *( fields LF ) LF
               ( body-text | body-binary )
pcp-fields   = "pcp-action" ":" pcp-action-value LF
               *( pcp-attributes )
               "pcp-body-type" ":" pcp-body-type LF
               *( fields )
pcp-action-value = "MESSAGE" | ... (tbd)
pcp-body-type   = "text" | "binary"
pcp-attributes  = "pcp-channel" ":" field-value LF
fields          = field-name ":" field-value
field-name      = any UTF-8 characters except special-char
field-value     = any UTF-8 characters except special-char
special-char    = LF | ":" | "\"
escape-sequence = replace "\" by "\\\" | replace ":" by "\:" | replace LF by "\n"
body-text       = any UTF-8 characters
body-binary     = any bytes encoded in Base64 format
```

Figure 1: BNF grammar of Push Channel Protocol message structure

The WebSocket log of a WebSocket connection to an ABAP Push Channel application using subprotocol “v10.pcp.sap.com” will show following records, when the client sends a PCP message with two fields and a body and the following values.

- Field 1 consisting of field name “field1” and field value “value1”
- Field 2 consisting of field name “field2” and field value “field2”
- A body containing the text “this is the body !”

In Chrome developer tools will show the following log entries:

pcp-action:MESSAGE pcp-body-type:text field1:value1 field2:value2 this is the body !	85	15:57:34.007
--	----	--------------

with the content:

```
1 pcp-action:MESSAGE
2 pcp-body-type:text
3 field1:value1
4 field2:value2
5
6 this is the body !
```

In this example the APC application returns the received message including the additional field with the field name “counter” and the field value “1” back:

pcp-action:MESSAGE pcp-channel:apc\://ping_pcp pcp-body-type:text field1:value1 field2:value2 counter:1 this is the body !	124	15:57:34.010
--	-----	--------------

with the content:

```
1 pcp-action:MESSAGE
2 pcp-channel:apc\://ping_pcp
3 pcp-body-type:text
4 field1:value1
5 field2:value2
6 counter:1
7
8 this is the body !
```

PCP Libraries

There exist up to now the following libraries for different use-cases:

- In ABAP Application Server and in the context of ABAP Channels the class CL_AC_MESSAGE_TYPE_PCP and interface IF_AC_MESSAGE_TYPE_PCP provide the necessary APIs.

Furthermore the PCP message type is an inbuilt message type in in [ABAP Messaging Channels](#) and in [ABAP Push Channels](#) as WebSocket subprotocol “v10.pcp.sap.com”.

- In SAP UI5 the class [sap.ui.core.ws.SapPcpWebSocket](#) provides the necessary PCP methods as an extension to the JavaScript WebSocket class.
- In other alternative UI technologies, e.g. Web Dynpro, Business Server Pages, SAP GUI HTML, provides “sap-pcp-websocket.js” the necessary PCP methods. This library is available in the mime repository (under the repository path /sap/public/bc/ur) and in ABAP Application Server with 740 SP10. For accessing the library refer to note [2050139](#) (SAP Push Channel Protocol (PCP) JavaScript library for ABAP Push Channel).

Like: [Alert Moderator](#)

Assigned tags

[ABAP Connectivity](#) | [abap](#) | [amc](#) | [apc](#) | [channels](#) |

[View more...](#)

Related Blogs

[Introduction to ABAP Channels](#)

By **Olga Dolinskaja** , Nov 27, 2014

[ABAP Push / Messaging Channel and SAPUI5 Demo Application](#)

By **Gregor Wolf** , Sep 07, 2015

[ABAP Channels Part 2: Publish/Subscribe Messaging Using ABAP Messaging Channels](#)

By **Masoud Aghadavoodi Jolfaei** , Mar 26, 2014

Related Questions

[ABAP Push Channel is not supported in the selected project](#)

By **Former Member** , Nov 03, 2016

[ABAP Push Channel ...How to deal with data?](#)

By **Amol Samte** , Oct 21, 2016

[ABAP Push Channel... Send data to UI5 application once data changes..](#)

By **Former Member** , Nov 10, 2016

5 Comments

You must be [Logged on](#) to comment or reply to a post.



Former Member

August 19, 2015 at 1:17 pm

Hi Masoud,

Thanks again for a useful blog post. It is just what I was looking for 😊 .

Regarding using V1 of the PCP protocol with ABAP Push Channel, giving the mention of a specific PCP protocol version, I was wondering whether another version of PCP is available? If so, could you please give me some quick hints about where I could find out more info about this?

Thank you!

Sergiu

Like (0)



Masoud Aghadavoodi Jolfaei | Post author

August 19, 2015 at 1:46 pm

Hi Sergiu,

the v10 version is the latest PCP protocol version and there is no other version of PCP protocol available.

Cheers,

Masoud

Like (0)



Gregor Wolf

September 4, 2015 at 9:27 pm

Dear Masoud,

I've tried to implement the example shown in the Videos [How to implement ABAP Push Channel \(APC\)](#) and [How to implement collaboration using ABAP Push Channel \(APC\) and ABAP Messaging Channel \(AMC\)](#). Unfortunately the code line:

```
this.oWs = new sap.ui.core.ws.SapPcpWebSocket("/sap/bc/apc/sap/zapc_echo",  
sap.ui.core.ws.SUPPORTED_PROTOCOLS.v10);
```

causes the Java Script error:

Uncaught TypeError: Cannot read property 'SapPcpWebSocket' of undefined

Which minimum Version of SAPUI5 do I need to make it work? On my ABAP Stack it is only 1.22.3.

Best regards

Gregor

Like (0)



Matthias Osswald

September 8, 2015 at 6:23 am

Hi Gregor,

did you miss to require the module first?

As it is not a control (lazyloading) and not loaded by default you have to manually require it:

```
jQuery.sap.require("sap.ui.core.ws.SapPcpWebSocket");
```

The WebSocket Classes should be available since 1.16 (protocol parameter since 1.20).

Cheers,

Matthias

Like (0)



Gregor Wolf

September 8, 2015 at 7:18 am

Hi Matthias,

thank you for your tip. In the meantime I've figured it out myself and already blogged about it:

[ABAP Push / Messaging Channel and SAPUI5 Demo Application](#)

Best regards

Gregor

Like (0)

Share & Follow

[Privacy](#)

[Terms of Use](#)

[Legal Disclosure](#)

[Copyright](#)

[Trademark](#)

[Cookie Preferences](#)

[Sitemap](#)

[Newsletter](#)