



Handbuch Webservices

Produktinfo und Support

Telefon: + 49 (0) 40/32 80 86 - 0
E-Mail: batchman@honico.com
Web: www.honico.com
Release: BatchMan 5.0
Stand: Februar 2018

SAP® Certified
Powered by SAP NetWeaver®

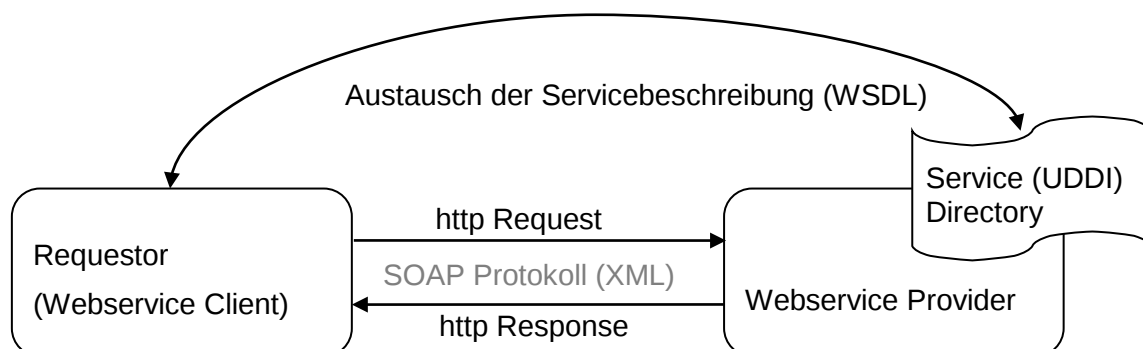


Inhaltsverzeichnis

1.	Webservices	3
1.1.	Allgemein	3
1.2.	Kommunikation	3
2.	BatchMan Webservice Client	4
2.1.	Webservice Setup	4
2.2.	Webservice Client	5
3.	Webservice Beispiel	7
3.1.	WSDL Extraktion	7
3.2.	BatchMan Webservice Setup	9
3.3.	BatchMan Webservice Client	10
4.	Alerttool Webservice	13
4.1.	Webservice Variante	13
4.2.	Alerttool Setup	14

1. Webservices

1.1. Allgemein



Ein Webservice ist ein über ein Netzwerk bereitgestellter Dienst, welcher eine Interaktion mit anderen Netzwerkteilnehmern unterstützt. Die Interaktionen zwischen den Prozessen erfolgt dabei über SOAP (Simple Object Access Protocol), dabei werden die Daten in XML-Form meist über http(s) ausgetauscht.

In welcher Form die Daten ausgetauscht werden, definiert jeder Webservice in seiner Servicebeschreibung, der WSDL (Web Service Definition Language).

1.2. Kommunikation

Um mit einem Webservice über SOAP zu kommunizieren, benötigt man folgende Informationen, welche man aus der WSDL (Servicebeschreibung) extrahieren kann:

- Webservice Endpoint
 - o URL über die der Webservice angesprochen werden kann
- SOAP-Action
 - o Auswahlkriterium, falls ein Webservice mehrere Dienste bereitstellt
- SOAP-Message
 - o Inputparameter in XML-Form

Da der manuelle Aufbau der SOAP Message aus einer WSDL-Datei, abhängig von der Komplexität des betreffenden Webservices, meist sehr kompliziert ist, empfiehlt sich hierbei die Nutzung von Hilfsprogrammen (WSDL-Parser). Diese Programme extrahieren oben beschriebene Informationen aus der WSDL-Datei.

Ein Beispiel für einen relativ populären WSDL-Parser, welcher auch im unten stehenden Teil zum Einsatz kommt, ist SoapUI (soapui.org). Die Basisversion unterliegt der GNU Lesser General Public License und ist kostenfrei. Sie gestattet neben dem Extrahieren der WSDL-Datei auch funktionale Tests des Webservices.

2. BatchMan Webservice Client

2.1. Webservice Setup

Um in das Setup für den BatchMan Webservice Client zu gelangen, wählen Sie:

BatchMan®-Funktionsbaum → *Customizing* → *Technisches Customizing* → *Setup WebService*

The screenshot shows the 'BatchMan Setup WebServices: Einstieg' dialog box in the SAP environment. The window has a title bar with 'SAP' and standard window controls. Below the title bar is a menu bar with 'Bearbeiten', 'System', and 'Hilfe'. A toolbar contains various icons for file operations and help. The main area is divided into several sections:

- BatchMan: Setup Webservice** (Section Header)
- Name**: A text input field with a yellow background and a lock icon.
- Beschreibung**: A text input field.
- Endpoint URL**: A text input field.
- SOAP Action**: A text input field.
- Authentication**: Two checkboxes, 'http Auth' and 'Proxy', both currently unchecked.
- SOAP Message**: A large text area for entering the SOAP message.
- Navigation**: A row of icons for navigating between different parts of the setup.
- Status/Log**: A section at the bottom showing 'Ze 1, Sp 1' and 'Ze 1 - Ze 1 von 1 Zeilen'.
- Timing**: Two input fields for 'Erfassung' and 'Änderung', both showing '00:00:00'.
- Footer**: A status bar at the very bottom showing 'HOD (1) 000', 'BMD1', and 'INS'.

Bitte geben Sie einen Namen für den Webservice ein. Wenn Sie auf „Anlegen“ klicken und eine Beschreibung eintragen, verbleiben die 3 oben beschriebenen Felder „Endpoint URL“ „SOAP Action“ und „SOAP Message“ (vgl. 1.2 Kommunikation).

Ab SAP Rel. 7.01 wird beim Anlegen die URL der WSDL abgefragt und anschließend geparkt – das manuelle Parsen (vgl. 1.2 Kommunikation) entfällt somit. Zusätzlich gibt es die Felder „http Auth“ mit

„Benutzer“ und „Password“. Diese Felder benötigen Sie nur, wenn der http-Server des Webservices eine Authentifizierung (Basic Authentication nach RFC 2617) benötigt.

Die Felder „Proxy“ mit „Proxy Rechner“, „Proxy Benutzer“ und „Proxy Password“, benötigen Sie, falls für die Kommunikation mit dem http-Server ein Proxy-Server notwendig ist.

2.2. Webservice Client

Nachdem Sie einen Webservice im BatchMan eingerichtet haben, können Sie ihn über den BatchMan Webservice Client aufrufen.

Um zu dem Webservice Client zu gelangen, wählen Sie:

BatchMan®-Funktionsbaum → Customizing → Technisches Customizing → Variante für Webservice

[illegible]

Dieser Report kann sowohl im Dialog als auch im Batch, sowie bei der Alarmierung als Alerttool verwendet werden.

Um den Report im Batch oder als Alerttool zu verwenden, müssen Sie eine Variante anlegen. Unter dem Feld „Name“ haben Sie eine Listbox mit allen im BatchMan definierten Webservices (vgl. 2.1 Webservice Setup).

Haben Sie einen Webservice ausgewählt, erscheinen in der Parametertabelle die zugehörigen Input Parameter. An dieser Stelle können Sie den Inputparametern feste oder dynamische Werte zuordnen.

Um einen Parameter einen festen Wert zuzuordnen, tragen Sie diesen in der betreffenden Tabellenzeile ein. Um einem Parameter einen dynamischen Wert zuzuordnen, tragen Sie in der betreffenden Tabellenzeile den Ersetzungsparameter ein. Folgende Ersetzungsparameter sind dabei möglich (vgl. BM4 UserHB, Zeile 227):

<OPSYS>	Betriebssystem
<INSTANCE>	Instanz der R/3- Anwendung
<SYSID>	Name der R/3- Anwendung laut Systemfeld SY- SYSID.
<DBSYS>	Datenbanksystem laut Systemfeld SY- DBSYS
<SAPRL>	R/3- Release laut Systemfeld SY- SAPRL
<HOST>	Rechnername laut Systemfeld SY- HOST
<CLIENT>	Mandant laut Systemfeld SY- MANDT
<LANGUAGE>	Anmeldesprache laut Systemfeld SY- LANGU
<DATE>	Datum laut Systemfeld SY- DATUM
<YEAR>	Jahr laut Systemfeld SY- DATUM, vierstellig
<SYEAR>	Jahr laut Systemfeld SY- DATUM, zweistellig
<MONTH>	Monat laut Systemfeld SY- DATUM
<DAY>	Tag laut Systemfeld SY- DATUM
<WEEKDAY>	Wochentag laut Systemfeld SY- FDAYW
<TIME>	Uhrzeit laut Systemfeld SY- UZEIT
<STIME>	Stunde und Minute laut Systemfeld SY- UZEIT
<HOUR>	Stunde laut Systemfeld SY- UZEIT
<MINUTE>	Minute laut Systemfeld SY- UZEIT
<SECOND>	Sekunde laut Systemfeld SY- UZEIT
<P=name>	Wert eines Profilparameters des laufenden Systems
<V=name>	Wert einer Variablen gemäß Variablen-tabelle (TCODE FILE)
<F=name>	Rückgabewert eines Funktionsbausteins
<T=name>	TVARV- Variable
<L= logische Datei>	definiert einen logischen Dateinamen (TCODE FILE)
<A=logischer Pfad>	definiert einen logischen Pfad (TCODE FILE)

Für die Verwendung als Alerttool stehen folgende, weitere Ersetzungsparameter zur Verfügung:

&HOST	Hostname (Feld SY-HOST aus der Systemtabelle SYST)
&SID	SAP System Ident (Feld SY-SYSID aus der Systemtabelle SYST)
&MDT	Mandant (Feld SY-MANDT aus der Systemtabelle SYST)
&RFC	RFC Verbindung (Zielsystem) eines Jobs
&JOB	Jobname
&COUNT	Jobcount
&DATE	Datum
&TIME	Zeit
&CHECKP	Name des Checkpoints der den Alarm ausgelöst hat
&OPA	Nummer des Auftrags
&USER	Benutzername aus Jobstep (nur Mastersystem) als erste Zuordnung, sonst Benutzer aus Tabelle Nachrichtenempfänger
&TELEFON	Telefonnummer aus Tabelle Nachrichtenempfänger
&FAX	Faxnummer aus Tabelle Nachrichtenempfänger
&PAGER	Mobil- oder Pager Nummer aus Tabelle Nachrichtenempfänger
&MAIL	Mailadresse aus Tabelle Nachrichtenempfänger
&AClass	Alarmklasse aus Tabelle Nachrichtenempfänger
&MESS	enthält den zusammengesetzten Text, um ihn als Betreffzeile zu verwenden

&TID	Zuordnung einer sogenannten Transaktions-ID (laut SAP: „einer weltweit eindeutigen Kennung der aktuellen 'unit of work'")
&MAILFILE	Platzhalter für den Mailtext des Alerttools
&LOGFILE	Platzhalter für das Joblog
&LOGFILE[]	Platzhalter für das Joblog, BASE64 Codiert

Unter der Tabelle mit den Inputparametern kann die Antwort des Webservice behandelt werden. Sie können hier definieren, welche Output-Parameter wann zum Abbruch führen.

Die Namen der Output-Parameter können Sie dabei ebenfalls der WSDL-Datei entnehmen.

Vereinfacht können Sie den Webservice auch testweise ausführen, um an die Output-Parameter zu gelangen.

Als Ausgabeoptionen steht eine strukturierte (nur Parameter) und eine unstrukturierte Ausgabe (komplettes XML) zur Option. Bei der strukturierten Ausgabe können Sie zusätzlich die Output-Parameter selektieren, welche ausgegeben werden sollen. Nutzen Sie den Webservice für die Alarmierung ist hierbei die Besonderheit, dass der erste an dieser Stelle definierte Outputparameter in der Alarmierungshistorie gespeichert wird.

3. Webservice Beispiel

3.1. WSDL Extraktion

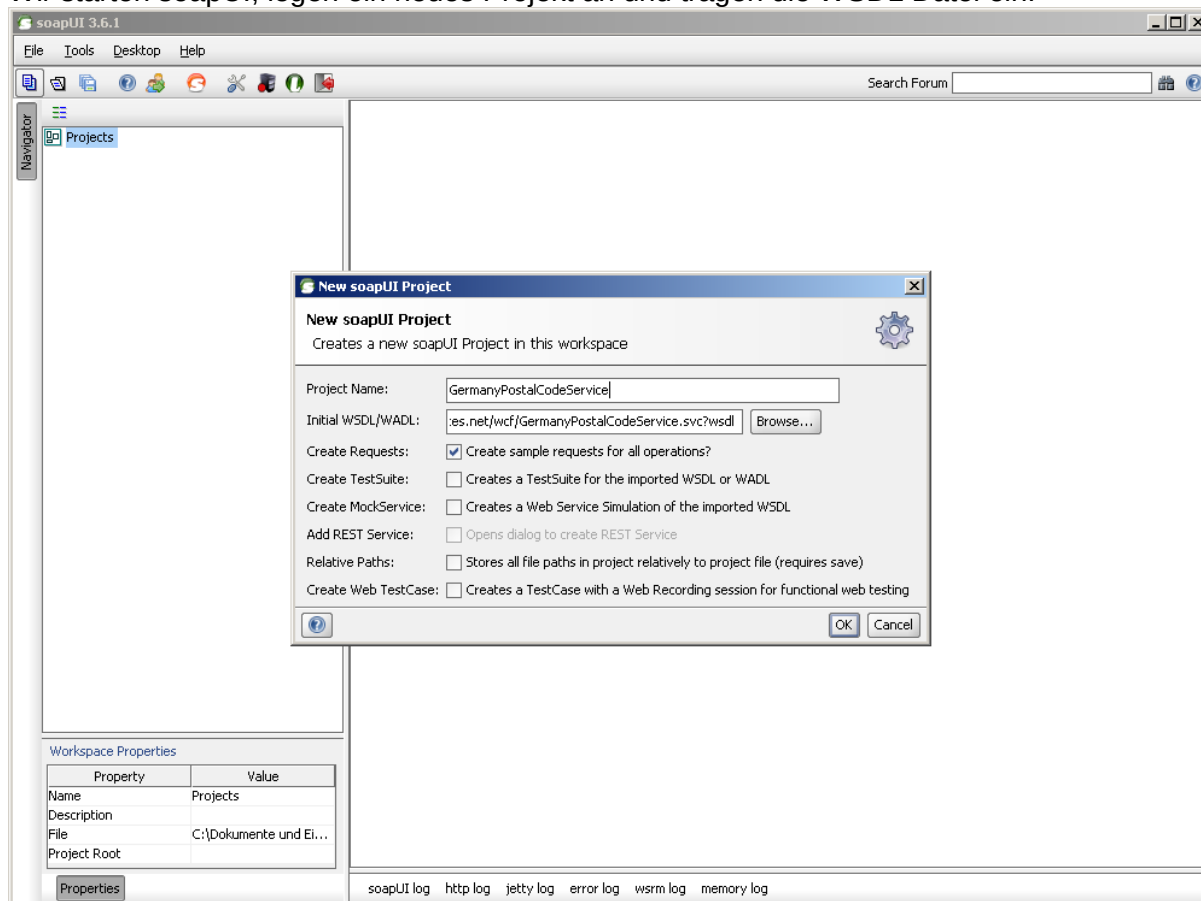
Wir wollen nun exemplarisch über BatchMan einen Test Webservice aufrufen.

Unser Test Webservice hat den Namen „GermanyPostalCodeService“ und folgende WSDL:

<http://www.restfulwebservice.net/wcf/GermanyPostalCodeService.svc?wsdl>

Als Erstes benötigen wir die 3 Parameter „Webservice Endpoint“, „SOAP-Action“ und „SOAP-Message“ (vgl. 1.2 Kommunikation).

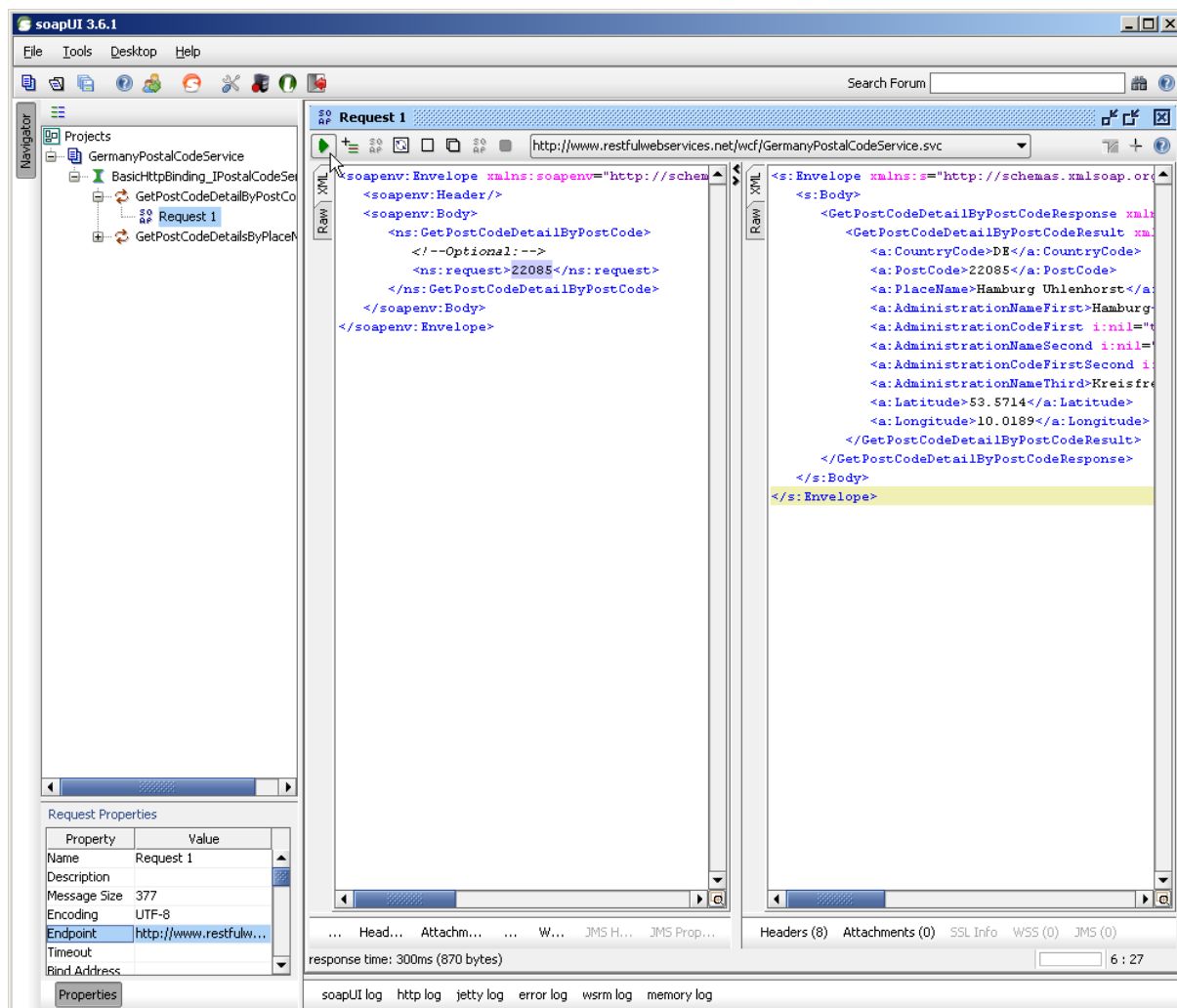
Wir starten soapUI, legen ein neues Projekt an und tragen die WSDL Datei ein.



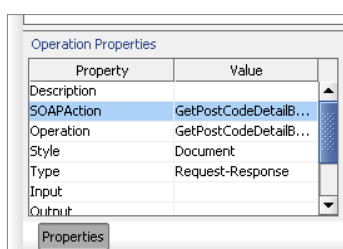
Wir sehen links im Baum, dass unser Webservice 2 Methoden anbietet und wollen uns die Methode „GetPostCodeDetailByPostCode“ näher ansehen.

Mit einem Doppelklick auf den Punkt „Request 1“ wird uns die SOAP Input Message generiert. Wir sehen im rechten Doppelfenster links die Input Message (Request), rechts die Output Message (Response).

Die Output Message bekommen wir logischer Weise erst, wenn wir die Input Message abgesendet haben. Testweise können wir einen Inputparameter für <ns:request> eintragen und den Webservice austesten, dies ist jedoch nicht zwingend notwendig, da wir ja lediglich an die Inputmessage wollen.



Rechts unten sehen wir den Eintrag für „Endpoint“ - somit haben wir schon 2 von unseren 3 Parametern.



Um an den dritten Parameter zu gelangen, klicken wir links im Baum, statt auf „Request 1“ wieder auf „GetPostCodeDetailByPostCode“ und bekommen unten links den letzten Parameter „SOAP Action“ angezeigt. Mit unseren 3 Parametern verlassen wir das Programm soapUI und gehen in den BatchMan.

3.2. BatchMan Webservice Setup

Wir starten das BatchMan Webservice Setup (vgl. 2.1 Webservice Setup) und tragen einen frei wählbaren Namen ein. Anschließend klicken wir auf „Anlegen“ und tragen eine Beschreibung ein. Wir übernehmen jetzt in die übrigen Felder unsere kopierten Werte „Webservice Endpoint“, „SOAP-Action“ und „SOAP-Message“ (vgl. 3.1 WSDL Extraktion).

BatchMan Setup Webservices: Anlegen

BatchMan: Setup Webservice

Name: GET_DETAIL_BY_POST_CODE

Beschreibung: Get detail by post code

Endpoint URL: http://www.restfulwebservice.net/wcf/GermanyPostalCodeService.svc

SOAP Action: GetPostCodeDetailByPostCode ☐ http Auth ☐ Proxy

SOAP Message:

```
<?xml version='1.0' encoding='utf-8' ?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns="http://www.restfulwebservice.net/PostalCodeService" >
  <soapenv:Header/>
  <soapenv:Body>
    <ns:GetPostCodeDetailByPostCode>
      <!--Optional:-->
      <ns:request>22085</ns:request>
    </ns:GetPostCodeDetailByPostCode>
  </soapenv:Body>
</soapenv:Envelope>
```

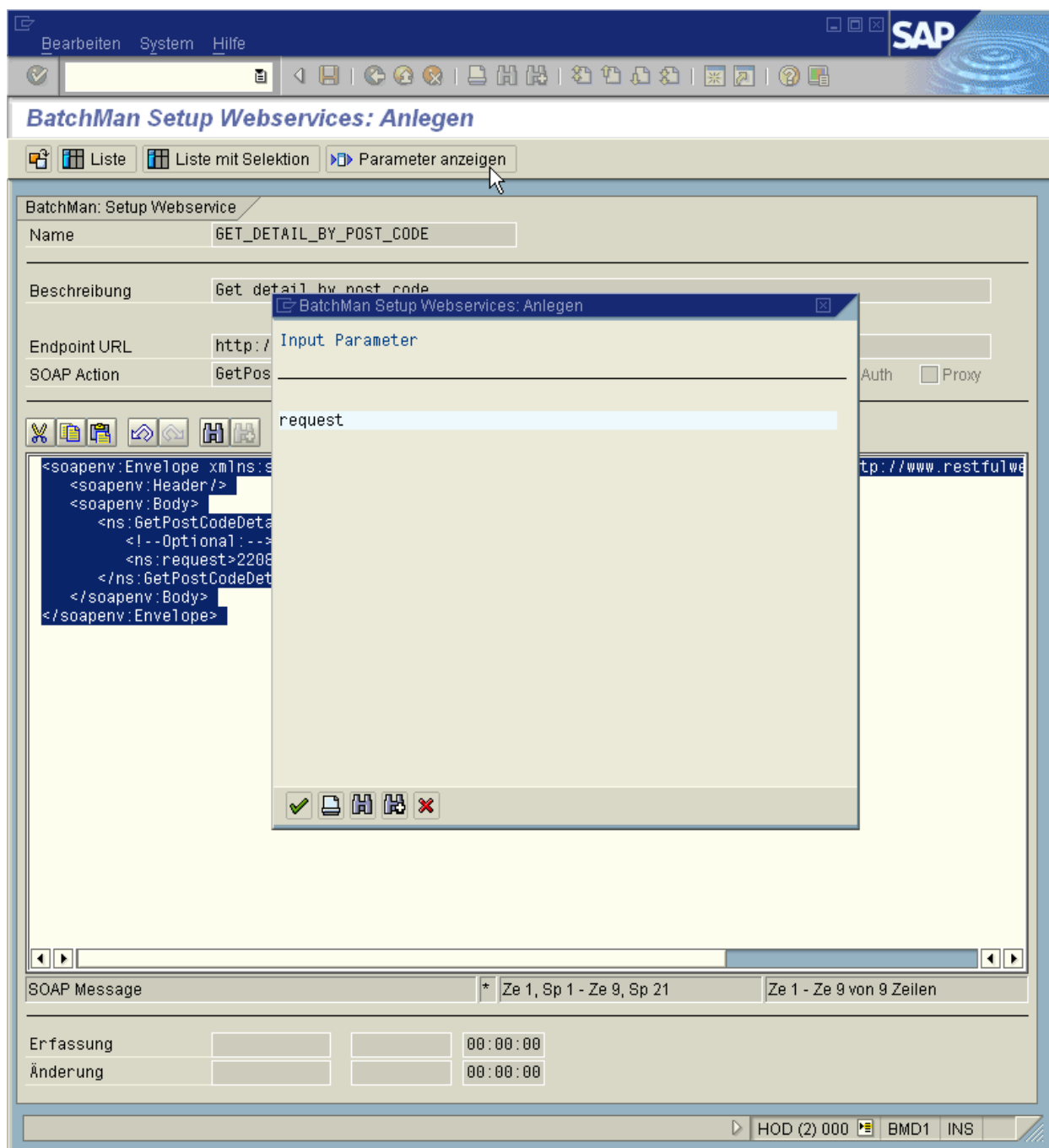
SOAP Message: * Ze 9, Sp 20 Ze 1 - Ze 9 von 9 Zeilen

Erfassung: 00:00:00

Änderung: 00:00:00

HOD (2) 000 BMD1 INS

Über den Button „Parameter anzeigen“ können wir uns die Parameter aus der SOAP-Message anzeigen lassen. Unser Test-Webservice hat nur einen Inputparameter mit dem Namen „request“. Diesen Parameter können wir im nächsten Absatz Werte oder Variablen zuteilen.



3.3. BatchMan Webservice Client

Wir starten den BatchMan Webservice Client (vgl. 2.2 Webservice Client).

Aus der Listbox wählen wir die zuvor definierte Webservice Konfiguration (vgl. 3.2 BatchMan Webservice Setup). In der Parametertabelle bekommen wir wieder unseren Parameter „request“ angezeigt und können diesem jetzt einen Wert zuordnen. Möglich ist hierbei ein fester Wert wie „22085“ oder auch ein dynamischer Wert, bspw. aus einer TVARV.

Wir entscheiden uns für eine Mischung aus beiden: „220<T=RANDOM_NUMBER>5“, wobei der Wert von „<T=RANDOM_NUMBER>“ zur Laufzeit durch den entsprechenden Wert aus der TVARV(C) – Tabelle (in unserem Fall der Wert „8“) ersetzt wird.
Zur Laufzeit haben wir somit den Wert „22085“.

The screenshot displays the SAP BatchMan (BM) interface for calling a web service. The main window is titled "BM: Webservice aufrufen". It shows the configuration for the web service "GET_DETAIL_BY_POST_CODE". The input parameter "request" is set to the value "220<T=RANDOM_NUMBER>5".

Below the input parameters, the "Web service Outputparameter" section is visible. It includes a table for the output parameters, which is currently empty. The "Ausgabe als:" (Output as) section is set to "XML (unparsed)".

In the foreground, a smaller window titled "Tabelle TVARVC anzeigen" (Display TVARVC Table) is open. It shows the configuration for the TVARVC table, including the table name "RANDOM_NUMBER", the type "P", and the number of digits "8". The table is currently empty.

Wir speichern die Selektion in einer Variante und führen den Report aus. Als Ausgabeliste erhalten wir die Antwort des Webservice.

BM: Webservice aufrufen

Parameter	Value
CountryCode	DE
PostCode	22085
PlaceName	Hamburg Uhlenhorst
AdministrationNameFirst	Hamburg
AdministrationCodeFirst	
AdministrationNameSecond	
AdministrationCodeFirstSecond	
AdministrationNameThird	Kreisfreie Stadt Hamburg
Latitude	53.5714
Longitude	10.0189

Webservice Aufruf erfolgreich! HOD (2) 000 BMD1 INS

Hier können wir uns Parameter aussuchen, welche zu einem Abbruch führen sollen. Meist gibt der Webservice Returncodes als Parameter zurück – in unserem Beispiel ist dies nicht der Fall.

Trotzdem sollten wir auf bestimmte Parameter triggern, da ansonsten ein Abbruch nur bei fehlerhaften http-Status erfolgt – Webservice standardmäßig jedoch den http-Status meist nur bei syntaktischen Problemen auf Fehler setzen.

Für unser Beispiel triggern wir auf die Parameter „PlaceName“, „CountryCode“ und „AdministrationNameFirst“.

Ein Abbruch erfolgt somit nur, wenn die PLZ „22085“, nicht irgendwo in Hamburg liegt, als Land nicht Deutschland oder als Bundesland Brandenburg zurückgegeben wird.

Webservice Outputparameter

Abbruch wenn:

Parameter	Wert		
PlaceName	Hamburg*	→	OR
CountryCode	DE	→	OR
AdministrationNameFirst	Brandenburg	→	

Ausgabe als:

☐ XML (unparsed) ☒ Parameter (geparsed) *

4. Alerttool Webservice

4.1. Webservice Variante

Zum Einrichten einer Webservice-Variante für die Alarmierung gehen Sie wie in Punkt 2.2 Webservice Client beschrieben vor (Voraussetzung Webservice Setup, vgl. 2.1).

BM: Webservice aufrufen

Webservice
Name: **SOLMAN_SERVICE_DESK**

Webservice Inputparameter

Parameter	Wert
AddInfoAttribute	SAPUserStatus
AddInfoValue	E0001SLFN0001
AttachmentGuid	&TID
Filename	joblog_&JOB_&COUNT.txt
MimeType	text/plain
Data	&LOGFILE[]
Timestamp	
PersonId	117
Url	
Language	
Delete	
IncidentGuid	&TID
RequesterGuid	E03AB580C0419BF19091000C29FB8A52
ProviderGuid	E03AB580C0419BF19091000C29FB8A51
AgentId	
ReporterId	117
ShortDescription	BatchMan Alarmierung
Priority	2

Webservice Outputparameter

Abbruch wenn:

Parameter	Wert
PrdctId	

Ausgabe als:

☐ XML (unparsed) ☒ Parameter (parsed) PrdctId

HOD (2) 000 BMD1 INS

Ordnen Sie den Parametern die gewünschten Werte zu. Sie können hier zusätzlich die Ersetzungsparameter für die Alarmierung benutzen.

Definieren Sie aussagekräftige Abbruchparameter, bspw. (Rückgabe ID ist leer (Bild), RC ungleich 0 etc.).

Rechts unten, definieren Sie den Output-Parameter der später in die Alarmierungshistorie übernommen werden soll.

Ein Webservice Abbruch bei der Alarmierung, können Sie an 3 Stellen nachverfolgen.

- SLG1 Anwendungslog
- Joblog des betreffenden Jobs
- Output-Parameter in der Alarmierungshistorie

4.2. Alerttool Setup

Das Alerttool Setup erfolgt, wie im Alerthandbuch beschrieben. Als Systemtyp wählen Sie bitte „Webservice“. Als Webservice Variante können Sie hierbei alle Varianten des Webservice Client (vgl. 4.1 Webservice Variante) verwenden.

Alert-Tools: Pflegen

BatchMan: Alert-Tools

Alert-Tool: SOLMAN

Bezeichnung: Create solman notification

Systemtyp: Webservice

Systemtypabhängige Eingaben

Webservice Variante: SOLMAN

Erfolgreich Meldung: Der Job &JOB mit dem Jobcount &COUNT ist erfolgreich gelaufen.

Fehlerhaft Meldung: Der Job &JOB mit dem Jobcount &COUNT ist abgebrochen.

Recoveryjob Meldung:

NoExec Meldung:

Minimale Laufzeit:

Maximale Laufzeit:

Maxim. Laufzeit erneut:

Langtext: Langtext ar

Erfassung: MZWIERZ 09.

letzte Änderung: MZWIERZ 11.

Systemtypabhängige Eingaben

Webservice Variante: SOLMAN

Langtext: Meldungen anzeigen

fuer Mail oder Datei: Sehr geehrte Damen und Herren,

&MESS

Das Joblog zu dieser Servicemeldung finden Sie Dokumente (&JOB_&COUNT.txt)

Mit freundlichen Grüßen