



E-Mail Setup Guide

WECO E-Commerce™ **Sales&Service**

SAP® Certified
Powered by SAP NetWeaver®

SAP® Certified
Integration with Applications on SAP HANA®

Release 8.00

Date 09/11/2018

Icons

Icon	Meaning
	Caution
	Example
	Advise
	Source Code
	Tipp

Copyright

Copyright (©) 2010-2018 WECO Software GmbH. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of WECO Software GmbH. The information contained herein may be changed without prior notice.

WECO E-Commerce Suite, WECO E-Selling, WECO Mobile Sales, WECO E-Catalog and WECO E-Procure are trademarks of WECO Software GmbH.

SAP, R/3, SAP NetWeaver, ABAP, BAPI and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany or an SAP affiliate company.

All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only.

Index

Index	3
Introduction	4
Overview available emails	4
Sender	4
Receiver.....	4
Note for upgrade customers	5
Setup	5
RFC Destination to WebAs.....	5
SICF Service.....	5
Create SICF Alias	5
Usage of custom alias	5
Note BSP URL for email rendering	5
Email class.....	6
CONSTRUCTOR.....	6
EXECUTE.....	6
SET_DATA	7
BSP-Application.....	8
View.....	8
View Redefinition.....	8
View Template.....	8
Eventhandler	9
Embed static HTML content	9

Introduction

This document describes how to create and maintain a new email template and how to send individual and dynamic emails.

With the WECO email system it is possible to easily create and send new emails by using HTML. To use this function it is only necessary to generate design and email content by using a BSP view and to outsource the entire email-handling in a separate class. Therefore WECO offers an easy and flexible solution.

Overview available emails

Type	Sending-System	Description	Sender
Catalog print	Core	Send catalog per email	Email E-Service
Password forgotten	Core	Password forgotten link at logon screen	Email E-Service
Account activation	Core	Account activation, Remove order lock, Send new credentials	Email E-Service or dealer email address
B2B User Request	Core	User requests B2B at logon screen	Email E-Service
Call me back	Core	Call me back button at shop starting page	Active user
Contact form	Core	Email link at homepage	Active user
Order confirmation	Core	After sending an order	Email E-Service or dealer email address
Order confirmation multi basket	Core	After sending an order	Email E-Service or dealer email address
Recommend article	Core	Article recommendation in product catalog	Active user

Sender

Here the customized email address of the web application E-Service is used:

B2C register New cust. lock
 E-Mail E-Service:

Or the email address of the active user (see overview available emails) or the email address of the dealer (only when using dealer scenario)

Receiver

The receiver determination is only relevant for user request. It is possible to maintain the receiver of a new user request per country.

Change View "Assign responsible person to registration request":

New Entries

Assign responsible person to registration request				
Application ID	Cty	Counter	User	E-Mail Address
WECO_SHOP		0		office@weco.at
WECO_SHOP	GB	0		office@weco.com

You can maintain these email addresses in the WECO IMG by using the following path:
WECO IMG: Web user -> Registration -> Assign responsible person to registration request

Note for upgrade customers

The new email service is used in WECO standard. To enable sending emails from WECO you have to follow the steps below. We recommend to adopt modifications of already existing emails to the new service.

Setup

RFC Destination to WebAs

In a 2-system-landscape scenario you have to create a RFC connection to the WebAs, because emails of the ERP system are generated on the WebAs. Therefore you can use the same RFC connection as you use for catalogue publishing.

Maintain the RFC connection in field *Dest.WebAs* per application:
WECO IMG: Web application -> Define system specific URLs

SICF Service

Activate the following service within the transaction SICF on your WebAs:
[/default_host/sap/bc/bsp/weco/w_email](#)

Create SICF Alias

Now create a SICF alias for the email service:

Alias `/email` `/default_host/sap/bc/bsp/weco/w_email`



CAUTION

It is very important to define valid logon credentials for the alias. You can use the same technical user as for the webshop alias.

Usage of custom alias

If it is not possible to name the alias `/email` you can use a custom alias name. Therefore define the common parameter ALWMA (alias WAS email) in WECO IMG:
WECO IMG: Common Settings -> Define common parameters

On instantiation of WECO email classes the aliases `/email` will be overwritten with your customized alias name.

In a 2-system-landscape scenario you have to maintain the parameter on your WebAs.

Note BSP URL for email rendering

By default the URL of the BSP application for email rendering is created implicitly. If you have to use different URLs for BSP applications in your system environment (e.g. load balancing, proxy, HTTPS), you have the possibility to define a HTTP RFC destination with a different URL.

In the WECO IMG you can use the common parameter RFCMA to set your HTTP RFC destination.
WECO IMG: Common Settings -> Define common parameters

If the BSP application is called for email rendering the RFC destination will be used.

For instance, HTTP RFC destination WECOMAIL:

The screenshot shows the 'RFC Destination WECOMAIL' configuration window. It includes a 'Connection Test' button, a 'Description' section with three text boxes, and a 'Target System Settings' section with fields for 'Target Host' (mysapsystem.weco.at), 'Service No.' (443), and 'Path Prefix'.

RFC Destination WECOMAIL	
Connection Test	
RFC Destination	WECOMAIL
Connection Type	G HTTP Connection to External Serv Description
Description	
Description 1	<input type="text"/>
Description 2	<input type="text"/>
Description 3	<input type="text"/>
Administration Technical Settings Logon & Security Special Options	
Target System Settings	
Target Host	mysapsystem.weco.at Service No. 443
Path Prefix	<input type="text"/>

In a 2-system-landscape this destination and the common parameter RFCMA have to be customized on the WebAs.

Email class

WECO provides the model class /WECO/XCL_EMAIL_BASE. To create an own email class, it just has to inherit from this model class.

CONSTRUCTOR

Creating an email object needs some general information for the email delivery, e.g. application ID, language and the alias for the email BSP application. If no alias is given to the constructor the standard alias **/email** will be used (it does not matter if exists or not). The logical system or a message object are optional parameters which can be given to the constructor.

EXECUTE

The method EXECUTE has to be created manually (name is not relevant). All data which are important for sending the specific email should be given to this method. To enable access to this data for later usage it should be written to global variables.

The Method EXECUTE is responsible for setting data and sending an email.

Possible method calls within this method:

SET_OPTIONS:

Set email options like RFC-destination, document type or if BCS should be used.

SET_TITLE:

Set the email title

A structure of type /WECO/XS_SODOCCHGI1 is expected. The email title should be found in field ZOBJ_DESCR_LONG.

SET_BODY:

Here the whole email body is generated by calling the responsible view via http and giving back the content and passing it to the email body. This method just needs the name of the used email view (always use the original view to enable view redefinitions). Maintain the redefined view within WECO customizing.

SET_SENDER:

This method can be called optionally to set an email sender. If this method is not called the logged on SAP user will be used.

To set a sender explicitly this method needs a structure of type /WECO/XS_PARTNER_ADDR. The email address has to be found in field SMTPA. The sender name will be built by using the fields NAME1, NAME2 and NAME3.

SET_RECEIVER:

With this method the email receiver is set and it just needs an email address.

To set more than one receiver this method can be called as often as needed.

SEND:

Send the generated email

SET_DATA

This method is called explicitly in the SET_BODY method and must (!) be refined. It is responsible for exporting the needed data to a database. To find the right data in the email view please use the importing parameter IV_LGUID as index. Before exporting to the database the needed data have to be transformed to a string.

Example:

```
1 method set_data.
2
3     data: lv_recsite type string.
4
5     * Daten in XML umwandeln
6     call transformation id source examples = gs_recsite
7         result xml lv_recsite.
8
9     * Daten in Memory exportieren
10    export recsite from lv_recsite link from gv_link to database indx(00) id iv_lguid.
11
12    commit work.
13
14 endmethod.
```

BSP-Application

For rendering emails the BSP application /WECOW_EMAIL is used.

View

For every different email it is possible to use an own view with an own CSS. So a dynamic email can be designed quite easily and flexibly. Furthermore it is possible to use all functions of a view, just as includes, images etc.

To have access to pictures of the /WECOW_SHP1 application please maintain the system specific parameter "URL Mail Images" in the WECO customizing for the used shop application. Here the absolute URL for the image folder of the shop application is used.

URL Content	S4/content/
Host URL	
URL Mail Images	http://weco02.weco.at:8000/sap/bc/bsp/weco/w_shp1/S4/img/

To have access to this path use APPLICATION->GS_CUST_APPL-URLMI.

ADVISE

The URL for email images can be redefined with themes. For instance, it is possible to use a different logo for each theme.

View Redefinition

To adjust an already existing view please copy it into the customer name space and redefine the original view within the WECO customizing. Then you can change the redefined view as desired.

View Template

In order to display emails on all email clients, CSS has been implemented as inline styles and the email structure has been defined as tables.

If you would like to create your own HTML emails we strongly recommend the following template:

SOURCE CODE

```
<%@page language="abap" forceEncode="html" %>
<%@extension name="/WECO/SHP" prefix="zshp"%>

<!-- Inline Style Variables -->
<%@ include file = "V2/style_variables_incl.htm" %>
<!-- Email Header -->
<%@ include file = "V3/layout_header_incl.htm" %>

* Implement your content here.

<!-- Email Footer -->
<%@ include file = "V3/layout_footer_incl.htm" %>
```

Eventhandler

To enable generating an email an instance of the controller `init.do` has to be created in the event `OnCreate` of every view. After doing this it is possible to use view redefinitions in the WECO customizing and all short and long texts as usual.

```
DATA: lo_init TYPE REF TO cl_bsp_controller.

* call initialization (init.do)
lo_init ?=
create_controller( controller_name = 'init.do' ).
call_controller( lo_init ).
```

Furthermore all needed data should be imported again from the database to use them in the views. Use the GUID from the URL parameters to get the right data. To use the data correctly the imported strings have to be transformed to the original structures/tables.

```
15: * PARAMETERÜBERNAHME FÜR HTML AUSGABE
16: lv_guid = application->go_app->gs_urlparm-lguid.
17: import recsite to lv_recsite link to gv_link from database indx(00) id lv_guid.
18: delete from database indx(00) id lv_guid.
19:
20: commit work.
21:
22: call transformation id source xml lv_recsite
23:     result examples = gs_recsite.
```

Embed static HTML content

You can extend emails with static HTML content. By default, static HTML content is embedded in the email footer.

You can maintain the HTML content in the admin panel (Content Manager) of your webshop.