

OXID eSales Documentation ERP interface (SOAP) Protocol version 2.5.0

Copyright

Copyright © 2008 OXID eSales AG, Germany

All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the prior written permission of OXID eSales AG.

Decompilation of source code, piracy as well as transfer to a third party is not allowed.

Use of this content for any other purpose is expressly prohibited and may result in severe civil and criminal penalties. Violators will be prosecuted to the maximum extent possible.

The information contained within this document was created accordingly to the state of the art technology.

OXID eSales AG assumes no liability or warranty for the accuracy, the completeness or reliability of any content.

In spite of all efforts, the occurrence of mistakes cannot be eliminated completely.

Thus we appreciate any tips.

Conventions

The following typographical conventions are used in this document:

cursive font

used for file names, file paths, email addresses and URLs.

`non-proportional font`

used for code examples and names of code elements.

Non-proportional cursive font

used for navigation steps.

Contact

OXID eSales AG

Bertoldstrasse 48

79098 Freiburg

Germany

Fon: +49 (761) 36889 0

Fax: +49 (761) 36889 29

Represented by the Supervisory Board: Roland Fesenmayr (CEO), Eric Jankowsky, Andrea Seeger

Registered Office Freiburg i. Br.

Commercial Register Freiburg

No. HRB 701648

Table of contents

Copyright	2
Conventions	2
Contact	2
Table of contents	3
About this document	4
System requirements	4
Installation	4
Show all available functions	4
Read WSDL.....	4
Special data types	5
Functions overview	6
Using custom fields.....	32
Extending ERP with plugins	33
Expandability of SOAP interface.....	35

About this document

This document describes the protocol version 2.5.0 of the ERP-Soap interface for OXID eShop Version 4.4.0 and above.

Older protocol versions are further on supported. For that purpose, the version parameter has to be set adequately when calling WSDL (see chapter 2).

Valid protocol versions are:

- 1
- 1.1
- 2
- 2.1.0
- 2.2.0
- 2.3.0
- 2.4.0
- 2.5.0

For details of each protocol version, refer to the according documentation.

1 System requirements

- The SOAP extension of PHP has to be installed.

2 Installation

- Copy all files from *copy_this* to the main directory of OXID eShop.

3 Show all available functions

- The list of all available functions can be shown by calling the URL
<https://www.myeshop.com/oxideshop/modules/erp/oxerpservice.php?functions&version=2.5.0>

4 Read WSDL

- For reading the WSDL, call the URL
<https://www.myeshop.com/oxideshop/modules/erp/oxerpservice.php?wsdl&version=2.5.0>

5 Special data types

5.1 Data type ‚OXERPTType‘

5.1.1 Get-functions

Each Get-function returns OXERPTType type or an array of OXERPTType types. Each OXERPTType contains one object from eShop, e.g. a user, a product, a category and so on. The structure is as follows:

OXERPTType	
aResult	String[][]
blResult	Bool
sMessage	String

- `aResult` is a 2-dimensional array containing the database field names and the data of each field. For example, if an user object from table `oxuser` is returned:

	Database field name		Field content
<code>aResult[0][0]</code>	OXID	<code>aResult[0][1]</code>	36944b76cc9604c53.04579642
<code>aResult[1][0]</code>	OXACTIV	<code>aResult[1][1]</code>	1
<code>aResult[2][0]</code>	OXRIGHTS	<code>aResult[2][1]</code>	User
<code>aResult[n][0]</code>	...	<code>aResult[n][1]</code>	...

Example `aResult` if a product is loaded:

	Database field name		Field content
<code>aResult[0][0]</code>	OXID	<code>aResult[0][1]</code>	36944b76cc9604c53.04579642
<code>aResult[1][0]</code>	OXSHOPID ¹	<code>aResult[1][1]</code>	1
<code>aResult[2][0]</code>	OXPARENTID	<code>aResult[2][1]</code>	2077
<code>aResult[n][0]</code>	...	<code>aResult[n][1]</code>	...

- `blResult` is `True` or `False` and indicates if the function was executed without errors.
- `sMessage` contains an error message if `blResult` = `False`.

5.1.2 Set- and Delete-functions

When using Set-Functions, objects (user, product, order etc.) are transferred in an OXERPTType. The structure of OXERPTType is the same as in Get-Functions.

If several objects are to be transferred, they have to be stored in an array containing OXERPTType types.

¹ This field only exists in OXID eShop Enterprise Edition

If no OXID is set in `OXERPTYPE`, the eShop creates an OXID and saves the object with this OXID to database.

The Set- and Delete-functions return `OXERPTYPE`, too. In each returned `OXERPTYPE` the OXID is saved in `aResult[0][1].blResult` is `True` or `False` and indicates if the function was run successfully. If errors occurred, `sMessage` contains the error message.

5.2 Data type ,OXERPSort'

In some Get-functions the results can be sorted. For doing so, an `OXERPSort` type has to be transferred. The Structure of `OXERPSort` is:

OXERPSort	Type	Sample values
sFieldName	String	OXID or OXPRICE etc.
sType	String	ASC or DESC

An Example for `OXERPSort` usage is given at `OXERPGetArticles` function description (chapter 6.16).

6 Functions overview

6.1 OXERPLLogin

`OXERPLLogin` logs in to eShop (or Subshop) and returns the session-ID. This session-ID is needed for all other function calls.

Only objects from the main shop/subshop the login is made for can be accessed².

6.1.1 Parameters

Name	Type	Content
sUserName	String	Benutzername eines Admin
sPassword	String	Passwort des Admin
iShopID	Integer	Die ID des (Sub-)shops
iLanguage	Integer	Die ID für die Sprache

6.1.2 Return values

Name	Type	Content
OXERPLLoginResult	OXERPTYPE	Contains the session-ID in <code>sMessage</code> . This session-ID is needed for other functions.

² If logged in as malladmin, objects from other shops can be loaded but not saved.

6.2 OXERPSetUser

With `OXERPSetUser` several users can be created/updated.

6.2.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLLogin</code> .
aUser	OXERPTYPE[]	Contains fields of <i>oxuser</i> table as described in 5.1.

6.2.2 Return values

Name	Type	Content
OXERPSetUserResult	OXERPTYPE[]	In <code>aResult[0][1]</code> the particular OXID is stored and <code>blResult</code> is True or False.

6.3 OXERPGetUser

`OXERPGetUser` returns a user object for a given OXID.

6.3.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLLogin</code> .
sUserID	String	The OXID of the user.

6.3.2 Return Values

Name	Type	Content
OXERPGetUserResult	OXERPTYPE	Contains fields of <i>oxuser</i> table as described in 5.1.

6.4 OXERPGetUsers

`OXERPGetUsers` returns several user objects.

6.4.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLLogin</code> .
OXERPSort	OXERPSort	Contains the field name used for sorting and ASC or DESC for ascending or descending sorting. (see chapter 5.2).
iStart	Integer	Amount of objects which are skipped and not returned.
iCount	Integer	Limits amount of returned objects.
Timestamp	dateTime	Timestamp after which users were , format: YYYY-MM-DD HH:MM:SS.

6.4.2 Return values

Name	Type	Content
OXERPGetUsersResult	OXERPTyp[]	Contains fields of <i>oxuser</i> table as described in 5.1.

6.5 OXERPDeleteUser

`OXERPDeleteUser` deletes a user from *oxuser* table.

6.5.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLLogin</code> .
sUserID	String	The <code>OXID</code> of the user to be deleted.

6.5.2 Return values

Name	Type	Content
OXERPDeleteUserResult	OXERPTyp	Contains the particular <code>OXID</code> . <code>blResult</code> indicates if delete was successful.

6.6 OXERPDeleteUsers

`OXERPDeleteUsers` deletes several users at once.

6.6.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLLogin</code> .
aUser	OXERPTyp[]	Each <code>OXERPTyp</code> must contain the <code>OXID</code> of a user.

6.6.2 Return values

Name	Type	Content
OXERPDeleteUsersResult	OXERPTyp[]	Contains the particular <code>OXID</code> for each deleted user. <code>blResult</code> indicates if a delete was successful.

6.6.3 Example

An example SOAP request for deleting 2 users:

```
<soapenv:Envelope
  xmlns:q0="OXERPService"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Header>
```

```

</soapenv:Header>
<soapenv:Body>
  <q0:OXERPDeleteUsers>
    <q0:sSessionID>vjtavolulu7rrjj7mqgok3cim7</q0:sSessionID>
    <q0:aUser>
      <q0:OXERPTType>
        <q0:aResult>
          <q0:ArrayOfString>
            <q0:string>OXID</q0:string>
            <q0:string>0m4479c88e2a2f2b4.53241736</q0:string>
          </q0:ArrayOfString>
        </q0:aResult>
        <q0:blResult>true</q0:blResult>
      </q0:OXERPTType>
      <q0:OXERPTType>
        <q0:aResult>
          <q0:ArrayOfString>
            <q0:string>OXID</q0:string>
            <q0:string>84t479c9f5e9c6715.46010945</q0:string>
          </q0:ArrayOfString>
        </q0:aResult>
        <q0:blResult>true</q0:blResult>
      </q0:OXERPTType>
    </q0:aUser>
  </q0:OXERPDeleteUsers>
</soapenv:Body>
</soapenv:Envelope>

```

6.7 OXERPSetCategory

OXERPSetCategory writes several categories at once to *oxcategories* table.

6.7.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
aCategory	OXERPTType[]	Contains the fields of <i>oxcategories</i> table. See also 5.1.

6.7.2 Return values

Name	Type	Content
OXERPSetCategoryResult	OXERPTType[]	Contains the particular OXID for each category. blResult indicates if setting a category was successful.

6.8 OXERPGetCategory

OXERPGetCategory loads a category object for a given OXID.

6.8.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
sCategoryID	String	The ID of the category.

6.8.2 Return values

Name	Type	Content
OXERPGetCategoryResult	OXERPTYPE	Contains fields of <i>oxcategories</i> table as described in 5.1.

6.9 OXERPGetCategories

OXERPGetCategories loads several categories at once.

6.9.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
OXERPSort	OXERPSort	Contains the field name used for sorting and ASC or DESC for ascending or descending sorting. (see chapter 5.2).
iStart	Integer	Amount of objects which are skipped and not returned.
iCount	Integer	Limits amount of returned objects.

6.9.2 Return values

Name	Type	Content
OXERPGetCategoriesResult	OXERPTYPE[]	Contains fields of <i>oxcategories</i> table as described in 5.1.

6.10 OXERPDeleteCategory

OXERPDeleteCategory deletes a category identified by its OXID.

6.10.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
sCategoryID	String	The ID of the category.

6.10.2 Return values

Name	Type	Content
OXERPDeleteCategoryResult	OXERPTYPE	Contains the particular OXID. blResult indicates if delete was successful.

6.11 OXERPDeleteCategories

OXERPDeleteCategories deletes several categories at once.

6.11.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
aCategory	OXERPTYPE[]	Each OXERPTYPE must contain the OXID of a category.

6.11.2 Return values

Name	Type	Content
OXERPDeleteCategoriesResult	OXERPTYPE[]	Contains the particular OXID for each deleted category. blResult indicates if a delete was successful.

6.12 OXERPSetArticle

OXERPSetArticle writes several products at once to *oxarticles* table.

Note: starting from ERP interface version 2.4.0 the OXSTOCKFLAG field has a default value 1 (if it is not specified while creating *new* article).

6.12.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
aArticle	OXERPTYPE[]	Contains the fields of <i>oxarticles</i> table. See also 5.1.

6.12.2 Return values

Name	Type	Content
OXERPSetArticleResult	OXERPTYPE[]	Contains the particular OXID for each product. blResult indicates if an operation was successful.

6.13 OXERPGetArticle

OXERPGetArticle loads a product object for a given OXID.

6.13.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
sArticleID	String	The ID of the product.

6.13.2 Return values

Name	Type	Content
OXERPGetArticleResult	OXERPTYPE	Contains the fields of <i>oxarticles</i> table. See also 5.1.

6.14 OXERPSetArtExtends

`OXERPSetArtExtends` writes extended product data of several products to the *oxartextends* table.

6.14.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
aArticle	OXERPTYPE[]	Contains the fields of <i>oxartextends</i> table. See also 5.1.

6.14.2 Return values

Name	Type	Content
OXERPSetArtExtendsResult	OXERPTYPE[]	Contains the particular <code>OXID</code> for each product. <code>blResult</code> indicates if an operation was successful.

6.15 OXERPGetArtExtends

`OXERPGetArtExtends` loads extended product data from *oxartextends* table.

6.15.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
sArticleID	String	The ID of the product.

6.15.2 Return values

Name	Type	Content
OXERPGetArtExtendsResult	OXERPTYPE	Contains the fields of <i>oxartextends</i> table. See also 5.1.

6.16 OXERPGetArticles

`OXERPGetArticles` loads several products. A category can be given as parameter: then, only products within this category are loaded.

6.16.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
OXERPSort	OXERPSort	Contains the field name used for sorting and ASC or DESC for ascending or descending sorting. (see chapter 5.2).
iStart	Integer	Amount of objects which are skipped and not returned.
iCount	Integer	Limits amount of returned objects.
sCategoryID (optional parameter)	String	Contains the ID of the category from which products are to be loaded.

6.16.2 Return values

Name	Type	Content
OXERPGetArticlesResult	OXERPType[]	Contains the fields of <i>oxartextends</i> table. See also 5.1.

6.16.3 Example

Example SOAP request loading the first 2 products (*iStart*: 0, *iCount*: 2) from the category with OXID 30e44ab82c03c3848.49471214 sorted ascending *OXPRICE*:

```
<soapenv:Envelope
  xmlns:q0="OXERPService"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Header>
  </soapenv:Header>
  <soapenv:Body>
    <q0:OXERPGetArticles>
      <q0:sSessionID>elm3gqhofrntlienl2vgqglfel</q0:sSessionID>
      <q0:OXERPSort>
        <q0:sFieldName>OXPRICE</q0:sFieldName>
        <q0:sType>ASC</q0:sType>
      </q0:OXERPSort>
      <q0:iStart>0</q0:iStart>
      <q0:iCount>2</q0:iCount>
      <q0:sCategoryID>30e44ab82c03c3848.49471214</q0:sCategoryID>
    </q0:OXERPGetArticles>
  </soapenv:Body>
</soapenv:Envelope>
```

6.17 OXERPDeleteArticle

`OXERPDeleteArticle` deletes a product.

6.17.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLLogin</code> .
sArticleID	String	The ID of the product to be deleted.

6.17.2 Return values

Name	Type	Content
OXERPDeleteArticleResult	OXERPTYPE	Contains the particular <code>OXID</code> . <code>blResult</code> indicates if delete was successful.

6.18 OXERPDeleteArticles

`OXERPDeleteArticles` deletes several products at once.

6.18.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLLogin</code> .
aArticle	OXERPTYPE[]	Each <code>OXERPTYPE</code> must contain the <code>OXID</code> of a product.

6.18.2 Return values

Name	Type	Content
OXERPDeleteArticlesResult	OXERPTYPE[]	Each <code>OXERPTYPE</code> contains the particular <code>OXID</code> . <code>blResult</code> indicates if a delete was successful.

6.19 OXERPSetArticleStock

`OXERPSetArticleStock` changes the stock of several products.

6.19.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLLogin</code> .
aArticleStock	OXERPTYPE[]	In each <code>OXERPTYPE</code> <code>OXID</code> and <code>OXSTOCK</code> must be set.

6.19.2 Return values

Name	Type	Content
OXERPSetArticleStockResult	OXERPTYPE[]	Each <code>OXERPTYPE</code> contains the particular <code>OXID</code> . <code>blResult</code> indicates if a stock update was successful.

6.20 OXERPGetArticleStock

OXERPGetArticleStock returns the stock of a product.

6.20.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
sArticleID	String	The OXID of the product.

6.20.2 Return values

Name	Type	Content
OXERPGetArticleStockResult	OXERPTYPE	OXERPTYPE contains the particular OXID. blResult indicates if operation was successful.

6.21 OXERPSetArticleCategory

OXERPSetArticleCategory assigns a Product to categories.

6.21.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
sArticleID	String	The OXID of the product.
aCategoryID	OXERPTYPE[]	Each OXERPTYPE has to contain a OXID of a category.

6.21.2 Return values

Name	Type	Content
OXERPSetArticleCategoryResult	OXERPTYPE[]	Each OXERPTYPE contains the particular OXID of the <i>oxobject2category</i> table row. blResult indicates if each operation was successful.

6.22 OXERPGetArticleCategory

OXERPGetArticleCategory returns the category a product is assigned to.

6.22.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
sArticleID	String	The OXID of the product.
OXERPSort	OXERPSort	Contains the field name used for sorting and ASC or DESC for ascending or descending sorting. (see chapter 5.2).
iStart	Integer	Amount of objects which are skipped and not returned.
iCount	Integer	Limits amount of returned objects.

6.22.2 Return values

Name	Type	Content
OXERPGetArticleCategoryResult	OXERPTYPE[]	Each OXERPTYPE contains the values from <i>oxcategories</i> table. See also 5.1.

6.23 OXERPSetMainArticleCategory

OXERPSetMainArticleCategory sets the main category for a product.

6.23.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
sArticleID	String	The OXID of the product.
sCategoryID	String	The OXID of the category to be set as main category.

6.23.2 Return values

Name	Type	Content
OXERPSetMainArticleCategoryResult	OXERPTYPE[]	Each OXERPTYPE contains the values from <i>oxcategories</i> table. See also 5.1.

6.24 OXERPDeleteArticleCategory

OXERPDeleteArticleCategory deletes a category assignment from a product.

6.24.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
sArticle2CategoryID	String	The OXID of the <i>oxobject2category</i> entry which is to be deleted.

6.24.2 Return values

Name	Type	Content
OXERPDeleteArticleCategoryResult	OXERPTYPE	Contains the particular OXID. blResult indicates if the operation was successful.

6.25 OXERPSetArticleAction

OXERPSetArticleAction assigns a product to one or more actions.

Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
sArticleID	String	The OXID of the product.

aAction	OXERPTYPE[]	Each OXERPTYPE contains the OXID of an action. See also 5.1.
---------	-------------	--

6.25.1 Return values

Name	Type	Content
OXERPSetArticleActionResult	OXERPTYPE[]	Each OXERPTYPE contains the OXID from <i>oxactions2article</i> table. See also 5.1.

6.26 OXERPGetArticleAction

OXERPGetArticleAction returns the actions a product is assigned to.

6.26.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
sArticleID	String	The OXID of the product.
OXERPSort	OXERPSort	Contains the field name used for sorting and ASC or DESC for ascending or descending sorting. (see chapter 5.2).
iStart	Integer	Amount of objects which are skipped and not returned.
iCount	Integer	Limits amount of returned objects.

6.26.2 Return values

Name	Type	Content
OXERPGetArticleActionResult	OXERPTYPE[]	Each OXERPTYPE contains the values from <i>oxactions</i> table. See also 5.1.

6.27 OXERPDeleteActionsArticle

OXERPDeleteActionsArticle deletes an assignment of a product to an action.

6.27.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
sActions2ArticleID	String	The OXID of the <i>oxactions2article</i> table entry which is to be deleted.

6.27.2 Return values

Name	Type	Content
OXERPDeleteActionsArticleResult	OXERPTYPE	Contains the particular OXID. blResult indicates if the operation was successful.

6.28 OXERPSetArticleCrossselling

OXERPSetArticleCrossselling assigns several crosssellings to a product.

6.28.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
sArticleID	String	The <code>OXID</code> of the product.
aArticle	OXERPTYPE[]	Each OXERPTYPE contains the <code>OXID</code> of a product to be assigned as crossselling. See also 5.1.

6.28.2 Return values

Name	Type	Content
OXERPSetArticleCrosssellingResult	OXERPTYPE[]	Each OXERPTYPE contains the particular <code>OXID</code> of the <code>oxobject2article</code> table entry. <code>blResult</code> indicates if each operation was successful.

6.29 OXERPGetArticleCrossselling

OXERPGetArticleCrossselling returns crosssellings assigned to a product.

6.29.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
sArticleID	String	The <code>OXID</code> of the product.
OXERPSort	OXERPSort	Contains the field name used for sorting and ASC or DESC for ascending or descending sorting. (see chapter 5.2).
iStart	Integer	Amount of objects which are skipped and not returned.
iCount	Integer	Limits amount of returned objects.

6.29.2 Return values

Name	Type	Content
OXERPGetArticleCrosssellingResult	OXERPTYPE[]	Each OXERPTYPE contains the values from <code>oxarticles</code> table. See also 5.1.

6.30 OXERPDeleteArticleCrossselling

OXERPDeleteArticleCrossselling deletes a crossselling assignment from a product.
(before v2.4.0 it was called OXERPDeleteCrossselling)

6.30.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
sCrosssellingID	String	The OXID of the <code>oxobject2article</code> entry to be deleted.

6.30.2 Return values

Name	Type	Content
OXERPDeleteCrosssellingResult	OXERPTYPE	Contains the particular OXID. <code>blResult</code> indicates if the operation was successful.

6.31 OXERPSetArticleScalePrice

`OXERPSetArticlesScalePrice` sets scale prices for a product.

6.31.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
sArticleID	String	The OXID of the product.
aArticleScalePrice	OXERPTYPE[]	Each OXERPTYPE has to contain the fields of <code>oxprice2article</code> table. See also 5.1.

6.31.2 Return values

Name	Type	Content
OXERPSetArticleScalePriceResult	OXERPTYPE[]	Each OXERPTYPE contains the particular OXID of the <code>oxprice2article</code> table entry. <code>blResult</code> indicates if each operation was successful..

6.32 OXERPGetArticleScalePrice

`OXERPGetArticleScalePrice` returns scale prices for a product.

6.32.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
sArticleID	String	The OXID of the product.
OXERPSort	OXERPSort	Contains the field name used for sorting and ASC or DESC for ascending or descending sorting. (see chapter 5.2).
iStart	Integer	Amount of objects which are skipped and not returned.
iCount	Integer	Limits amount of returned objects.

6.32.2 Return values

Name	Type	Content
OXERPGetArticleScalePriceResult	OXERPTYPE[]	Each OXERPTYPE contains the values of <i>oxprice2article</i> table fields. See also 5.1.

6.33 OXERPDeleteArticleScalePrice

OXERPDeleteArticleScalePrice deletes a scale price from a product.

6.33.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
sArticleScalePriceID	String	The OXID of the <i>oxprice2article</i> entry to be deleted.

6.33.2 Return values

Name	Type	Content
OXERPDeleteArticleScalePriceResult	OXERPTYPE	Contains the particular OXID. <i>blResult</i> indicates if the operation was successful.

6.34 OXERPSetArticleAccessoire

OXERPSetArticleAccessoire assigns several accessoires to a product.

6.34.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
sArticleID	String	The OXID of the product.
aArticle	OXERPTYPE[]	Each OXERPTYPE contains the OXID of a product to be assigned as accessory. See also 5.1.

6.34.2 Return values

Name	Type	Content
OXERPSetArticleAccessoireResult	OXERPTYPE[]	Each OXERPTYPE contains the OXID of new <i>oxaccessoire2article</i> table entries. <i>blResult</i> indicates if each operation was successful.

6.35 OXERPGetArticleAccessoire

OXERPGetArticleAccessoire returns accessory products of a product.

6.35.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
sArticleID	String	The <code>OXID</code> of the product.
OXERPSort	OXERPSort	Contains the field name used for sorting and ASC or DESC for ascending or descending sorting. (see chapter 5.2).
iStart	Integer	Amount of objects which are skipped and not returned.
iCount	Integer	Limits amount of returned objects.

6.35.2 Return values

Name	Type	Content
OXERPGetArticleAccessoireResult	OXERPTYPE[]	Each <code>OXERPTYPE</code> contains the values of <code>oxarticles</code> table fields. See also 5.1.

6.36 OXERPDeleteAccessoireArticle

`OXERPDeleteAccessoireArticle` deletes an asseccoire assignment from a product.

6.36.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
sAccessoireArticleID	String	The <code>OXID</code> of the <code>oxaccessoire2article</code> table entry to be deleted.

Return values

Name	Type	Content
OXERPDeleteAccessoireArticleResult	OXERPTYPE	Contains the particular <code>OXID</code> . <code>blResult</code> indicates if the operation was successful.

6.37 OXERPSetOrder

`OXERPSetOrder` creates new orders.

6.37.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
aOrder	OXERPTYPE[]	Each <code>OXERPTYPE</code> contains the values of <code>oxorder</code> table fields. See also 5.1.

If 0 or nothing is set for `OXORDERNR` field, the eShop automatically creates a new Order number. For this, the highest order number is taken and increased by 1.

6.37.2 Return values

Name	Type	Content
OXERPSetOrderResult	OXERPTYPE[]	Each OXERPTYPE contains the <code>OXID</code> of new <i>oxorder</i> table entries. Also contains the <code>OXORDERNR</code> value. <code>blResult</code> indicates if each operation was successful.

6.38 OXERPGetOrder

`OXERPGetOrder` returns an order for a given order `OXID`.

6.38.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
sOrderID	String	The <code>OXID</code> of the order.

6.38.2 Return values

Name	Type	Content
OXERPGetOrderResult	OXERPTYPE	Contains the values of <i>oxorder</i> table fields. See also 5.1.

6.39 OXERPGetOrders

`OXERPGetOrders` returns several orders. By `Timestamp` parameter, only orders made after a certain point of time can be returned.

6.39.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
OXERPSort	OXERPSort	Contains the field name used for sorting and ASC or DESC for ascending or descending sorting. (see chapter 5.2).
iStart	Integer	Amount of objects which are skipped and not returned.
iCount	Integer	Limits amount of returned objects.
Timestamp	dateTime	Point of time starting from which orders are returned.

6.39.2 Return values

Name	Type	Content
OXERPGetOrdersResult	OXERPTYPE[]	Each OXERPTYPE contains the values of <i>oxorder</i> table fields. See also 5.1.

6.40 OXERPDeleteOrder

`OXERPDeleteOrder` deletes an order.

6.40.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
sOrderID	String	The <code>OXID</code> of the order to be deleted.

6.40.2 Return values

Name	Type	Content
OXERPDeleteOrderResult	OXERPTYPE	Contains the particular <code>OXID</code> . <code>blResult</code> indicates if the operation was successful.

6.41 OXERPDeleteOrders

`OXERPDeleteOrders` deletes several orders at once.

6.41.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
aOrder	OXERPTYPE[]	Each <code>OXERPTYPE</code> contains the <code>OXID</code> of an order to be deleted.

6.41.2 Return values

Name	Type	Content
OXERPDeleteOrdersResult	OXERPTYPE[]	Each <code>OXERPTYPE</code> contains a particular <code>OXID</code> . <code>blResult</code> indicates if each operation was successful.

6.42 OXERPSetOrderArticle

`OXERPSetOrderArticle` sets products to an order.

6.42.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
aOrderArticle	OXERPTYPE[]	Each <code>OXERPTYPE</code> contains the values of <code>oxorderarticles</code> table fields. See also 5.1.

6.42.2 Return values

Name	Type	Content
OXERPSetOrderArticleResult	OXERPTYPE[]	Each <code>OXERPTYPE</code> contains a particular <code>OXID</code> . <code>blResult</code> indicates if each operation was successful.

6.43 OXERPGetOrderArticle

`OXERPGetOrderArticle` returns the articles of an order.

6.43.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
sArticleID	String	The <code>OXID</code> of the product.
OXERPSort	OXERPSort	Contains the field name used for sorting and ASC or DESC for ascending or descending sorting. (see chapter 5.2).
iStart	Integer	Amount of objects which are skipped and not returned.
iCount	Integer	Limits amount of returned objects.

6.43.2 Return values

Name	Type	Content
OXERPSetOrderArticleResult	OXERPTyp[]	Each <code>OXERPTyp</code> contains a particular <code>OXID</code> . <code>blResult</code> indicates if each operation was successful.

6.44 OXERPDeleteOrderArticle

`OXERPDeleteOrderArticle` deletes a product from an order.

6.44.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
sOrderArticleID	String	Contains the <code>OXID</code> of the <code>oxorderarticles</code> entry to be deleted.

6.44.2 Return values

Name	Type	Content
OXERPDeleteOrderArticleResult	OXERPTyp	Contains the particular <code>OXID</code> . <code>blResult</code> indicates if the operation was successful.

6.45 OXERPSetOrderStatus

`OXERPSetOrderStatus` sets values for `OXERPSTATUS` fields in `oxorderarticles` table.

6.45.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .

aOrderStatus	OXERPTType[]	Each OXERPTType contains the fields OXID, OXERPSTATUS_STATUS ³ , OXERPSTATUS_TIME Fehler! Textmarke nicht definiert. und OXERPSTATUS_TRACKID See also 5.1.
--------------	--------------	--

6.45.2 Return values

Name	Type	Content
OXERPSetOrderStatusResult	OXERPTType[]	Each OXERPTType contains a particular OXID. <code>blResult</code> indicates if each operation was successful.

6.46 OXERPSetVendor

OXERPSetVendor creates new vendors.

6.46.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
aVendor	OXERPTType[]	Each OXERPTType contains the values of <i>oxvendor</i> table fields. See also 5.1.

6.46.2 Return values

Name	Type	Content
OXERPSetVendorResult	OXERPTType[]	Each OXERPTType contains a particular OXID. <code>blResult</code> indicates if each operation was successful.

6.47 OXERPGetVendor

OXERPGetVendor returns a vendor.

6.47.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
sVendorID	String	The OXID of the vendor.

6.47.2 Return values

Name	Type	Content
OXERPGetVendorResult	OXERPTType	Contains the values of <i>oxvendor</i> table fields. See also 5.1.

6.48 OXERPGetVendors

OXERPGetVendors returns several vendors.

³ Always empty in OXID eShop Professional Edition

6.48.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
OXERPSort	OXERPSort	Contains the field name used for sorting and ASC or DESC for ascending or descending sorting. (see chapter 5.2).
iStart	Integer	Amount of objects which are skipped and not returned.
iCount	Integer	Limits amount of returned objects.

6.48.2 Return values

Name	Type	Content
OXERPGetVendorsResult	OXERPTyp[]	Each OXERPTyp contains the values of <i>oxvendor</i> table fields. See also 5.1.

6.49 OXERPDeleteVendor

`OXERPDeleteVendor` deletes a vendor.

6.49.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
sVendorID	String	The <code>OXID</code> of the vendor to be deleted.

6.49.2 Return values

Name	Type	Content
OXERPDeleteVendorResult	OXERPTyp	Contains the particular <code>OXID</code> . <code>blResult</code> indicates if the operation was successful.

6.50 OXERPDeleteVendors

`OXERPDeleteVendors` deletes several vendors at once.

6.50.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
aVendor	OXERPTyp[]	Each OXERPTyp has to contain the <code>OXID</code> of a vendor to be deleted.

6.50.2 Return values

Name	Type	Content
OXERPDeleteVendorsResult	OXERPTyp[]	Each OXERPTyp contains the particular <code>OXID</code> . <code>blResult</code> indicates if each operation was successful.

6.51 OXERPSetArticleVendor

`OXERPSetArticleVendor` assigns a vendor to a product.

6.51.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLLogin</code> .
sArticleID	String	The <code>OXID</code> of the product.
sVendorID	String	The <code>OXID</code> of the vendor.

6.51.2 Return values

Name	Type	Content
<code>OXERPSetArticleVendorResult</code>	<code>OXERPTType</code>	Contains the particular <code>OXID</code> . <code>blResult</code> indicates if the operation was successful.

6.52 OXERPGetArticleVendor

`OXERPGetArticleVendor` returns the vendor assigned to a product.

6.52.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLLogin</code> .
sArticleID	String	The <code>OXID</code> of the product.

6.52.2 Return values

Name	Type	Content
<code>OXERPGetArticleVendorResult</code>	<code>OXERPTType</code>	Contains the fields from <code>oxvendor</code> table. See also 5.1.

6.53 OXERPGetArticleVendors

`OXERPGetArticleVendors` returns the assigned vendors for an several products. Each vendor is returned only once.

6.53.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLLogin</code> .
aArticle	<code>OXERPTType[]</code>	Each <code>OXERPTType</code> contains the <code>OXID</code> of a product.

6.53.2 Return values

Name	Type	Content
<code>OXERPGetArticleVendorsResult</code>	<code>OXERPTType[]</code>	Each <code>OXERPTType</code> contains the fields from <code>oxvendor</code> table. See also 5.1.

6.54 OXERPGetVendorArticles

`OXERPGetVendorArticles` loads all products assigned to a vendor. The result list can be limited.

6.54.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLLogin</code> .
sVendorID	String	The <code>OXID</code> of the vendor.
OXERPSort	OXERPSort	Contains the field name used for sorting and ASC or DESC for ascending or descending sorting. (see chapter 5.2).
iStart	Integer	Amount of objects which are skipped and not returned.
iCount	Integer	Limits amount of returned objects.

6.54.2 Return values

Name	Type	Content
OXERPGetVendorArticlesResult	OXERPTYPE[]	Each <code>OXERPTYPE</code> contains the fields from <code>oxarticle</code> table. See also 5.1.

6.55 OXERPSetCountry

`OXERPSetCountry` creates new countries.

6.55.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLLogin</code> .
aCountry	OXERPTYPE[]	Each <code>OXERPTYPE</code> contains the fields from <code>oxcountry</code> table. See also 5.1.

6.55.2 Return values

Name	Type	Content
OXERPSetCountryResult	OXERPTYPE[]	Each <code>OXERPTYPE</code> contains the particular <code>OXID</code> . <code>blResult</code> indicates if each operation was successful.

6.56 OXERPGetCountry

`OXERPGetCountry` loads a country.

6.56.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLLogin</code> .
sCountryID	String	The <code>OXID</code> of the country.

6.56.2 Return values

Name	Type	Content
OXERPGetCountryResult	OXERPTYPE	Contains the fields from <i>oxcountry</i> table. See also 5.1.

6.57 OXERPGetCountries

OXERPGetCountries loads several countries.

6.57.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
OXERPSort	OXERPSort	Contains the field name used for sorting and ASC or DESC for ascending or descending sorting. (see chapter 5.2).
iStart	Integer	Amount of objects which are skipped and not returned.
iCount	Integer	Limits amount of returned objects.

6.57.2 Return values

Name	Type	Content
OXERPGetCountriesResult	OXERPTYPE[]	Each OXERPTYPE contains the fields from <i>oxcountry</i> table. See also 5.1.

6.58 OXERPDeleteCountry

OXERPDeleteCountry deletes a country.

6.58.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by OXERPLogin.
sCountryID	String	The OXID of the country to be deleted.

6.58.2 Return values

Name	Type	Content
OXERPDeleteCountryResult	OXERPTYPE	Contains the particular OXID. <i>blResult</i> indicates if the operation was successful.

6.59 OXERPSetObject2Category

OXERPSetObject2Category is used for assigning one or more products to a category.

6.59.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
aObject2Category	OXERPTyp[]	Each OXERPTyp contains the fields from <i>object2category</i> table. If OXTIME field is left empty, the current time is taken. The entry with the oldest OXTIME is automatically set as main category.

6.59.2 Return values

Name	Type	Content
OXERPSetObject2CategoryResult	OXERPTyp[]	Each OXERPTyp contains a particular OXID. blResult indicates if each operation was successful.

6.60 OXERPGetObject2Category

`OXERPGetObject2Category` returns product assignments to categories.

6.60.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
sObject2CategoryID	String	OXID of the <i>object2category</i> entry. May be left empty.
sArticleID	String	OXID of the product. May be left empty.
sCategoryID	String	OXID of the category. May be left empty.
OXERPSort	OXERPSort	Contains the field name used for sorting and ASC or DESC for ascending or descending sorting. (see chapter 5.2).
iStart	Integer	Amount of objects which are skipped and not returned.
iCount	Integer	Limits amount of returned objects.

6.60.2 Return values

Name	Type	Content
OXERPGetObject2CategoryResult	OXERPTyp[]	Each OXERPTyp contains the fields from <i>object2category</i> table. See also 5.1.

6.61 OXERPDeleteObject2Category

`OXERPDeleteObject2Category` deletes a product assignment to a category. It deletes a given *object2category* entry.

6.61.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
sObject2CategoryID	String	The OXID of the <code>oxobject2category</code> table entry to be deleted.

6.61.2 Return values

Name	Type	Content
OXERPDeleteObject2CategoryResult	OXERPTYPE	Contains the particular OXID. <code>blResult</code> indicates if the operation was successful.

6.62 OXERPCallPlugin

OXERPCallPlugin invokes an external plugin.

6.62.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
sPluginName	String	The name of invoked plugin.
aRequestData	OXERPTYPE[]	Data to be passed to the invoked plugin.

6.62.2 Return values

Name	Type	Content
OXERPCallPluginResult	OXERPTYPE[]	Plugin output data.

Using custom fields

If additional fields were created in tables, the ERP module files have to be adjusted. The files are stored in `/modules/erp/objects`. The filename indicates for which table file is used. At the beginning, the files contain a field list. This list has to be extended with the names of the additional fields.

6.63 For Enterprise Edition users

You have to update your views after making changes. To do so:

- Log in to eShop Admin
- Go to *Master Settings* -> *Core Settings* -> *Mall*
- Click *Update DB Views now*

6.64 Example

In this example, the table `oxcategories` was extended with the field `articles_amount`. This field contains the amount of products assigned to the category.

For using this field, `oxerptype_oxcategory.php` is adjusted:

The file contains a definition of database fields:

```
'OXID'           => 'OXID',  
'OXPARENTID'    => 'OXPARENTID',  
'OXORDER'       => 'OXORDER',  
'OXACTIV'       => 'OXACTIV',
```

This definition is extended with `articles_amount`:

```
.  
. .  
'OXSEOID_3'      => 'OXSEOID_3',  
'ARTICLES_AMOUNT' => 'ARTICLES_AMOUNT'  
);
```

When using Set-functions, the field `articles_amount` can also be set in `OXERPTType`. When using Get-functions, the `articles_amount` field is also returned.

Extending ERP with plugins

With plugins, you can extend the functionality of the SOAP interface. Plugins are called using the function `OXERPCallPlugin` (See 6.62).

6.65 Creating a plugin

A plugin class has to extend the class `oxErpPluginBase` and implement `invoke()` function (which is defined abstract by `oxErpPluginBase`). The class name has to contain *Plugin* at the end (e.g. *examplePlugin* as below).

The Plugins are stored in in the `<erp base>/plugins` directory. The filename has to be the same as the class name of the plugin **in lowercase** (e.g. *exampleplugin.php* for the example below).

6.66 Sample plugin

```
<?php
/**
 * plugin which duplicates input
 */
class examplePlugin extends oxErpPluginBase
{
    /**
     * plugin entry point
     *
     * @param array $aParams 'name=>value' based passed parameters map
     *
     * @return oxERPReturn[]
     */
    public function invoke($aParams)
    {
        $oResult = new oxERPReturn();
        $oResult->aResult = $aParams;
        $oResult->blResult = true;
        $oResult->sMessage = 'example msg';
        return array($oResult, clone $oResult);
    }
}
```

6.66.1 data flow of the sample plugin

Assuming data of a soap request is:

```
OXERPTType->aResult->ArrayOfString[0]->string[0] = 'myvarname1';
OXERPTType->aResult->ArrayOfString[0]->string[1] = 'value1';
OXERPTType->aResult->ArrayOfString[1]->string[0] = 'myvarname2';
OXERPTType->aResult->ArrayOfString[1]->string[1] = 'value2';
```

The plugin's **invoke** method gets:

```
array (  
    'myvarname1' => 'value1',  
    'myvarname2' => 'value2',  
)
```

If there is a need of custom input data processing, a plugin may override **invokeEntry** method, which wraps the call to the *invoke* and passes it the input data in the correct format.

The example plugin duplicates data, and returns it to the call handler. The result data is:

```
array (  
    0 =>  
        OXERPReturn::__set_state(array(  
            'blResult' => true,  
            'sMessage' => 'example msg',  
            'aResult' =>  
                array (  
                    'myvarname1' => 'value1',  
                    'myvarname2' => 'value2',  
                ),  
            ),  
        )),  
    1 =>  
        OXERPReturn::__set_state(array(  
            ... [same as 0th entry] ...  
        )),  
)
```

Such result is formatted (by *oxERPGateway*) according to the request type (SOAP or CSV) and returned to the client.

Expandability of SOAP interface

You can also extend the SOAP interface with own functions, but we advise you to use plugins (See chapter 0) for keeping the ERP easily updatable.

How to extend the SOAP interface is explained with an example: The import of attributes is implemented. As a guideline, oxcountry entries can be taken.

6.67 Define new attribute object

For accessing attributes, an attribute object has to be defined. Existing object files are stored in *modules/objects*. Here, a new file has to be created: *oxerptype_attribute.php*:

```
<?php

require_once( 'oxerptype.php' );

class oxERPTType_Attribute extends oxERPTType
{
    public function __construct()
    {
        global $myConfig;

        parent::__construct();

        $this->_sTableName = 'oxattribute';
        $this->_sShopObjectName = 'oxattribute';

        $this->_aFieldList = array(
            'OXID'          => 'OXID',
            'OXSHOPID'      => 'OXSHOPID',
            'OXSHOPINCL'    => 'OXSHOPINCL',
            'OXSHOPEXCL'    => 'OXSHOPEXCL',
            'OXTITLE'       => 'OXTITLE',
            'OXTITLE_1'     => 'OXTITLE_1',
            'OXTITLE_2'     => 'OXTITLE_2',
            'OXTITLE_3'     => 'OXTITLE_3',
            'OXPOS'         => 'OXPOS'
        );
    }
}
```

In `$this->_aFieldList`, all database fields from *oxattribute* table are defined.

6.68 Register new function in WSDL file

A new function called `OXERPSetAttribute` has to be entered in *oxerpservice.wsdl*:

6.68.1 Definition of data types (type)

```
<s:element name="OXERPSetAttribute">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="sSessionID" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="aCountry" type="tns:ArrayOfOXERPTYPE" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="OXERPSetAttributeResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="OXERPSetAttributeResult"
        type="tns:ArrayOfOXERPTYPE" />
    </s:sequence>
  </s:complexType>
</s:element>
```

6.68.2 Definition of messages (message)

```
<wsdl:message name="OXERPSetAttributeSoapIn">
  <wsdl:part name="parameters" element="tns:OXERPSetAttribute" />
</wsdl:message>
<wsdl:message name="OXERPSetAttributeSoapOut">
  <wsdl:part name="parameters" element="tns:OXERPSetAttributeResponse" />
</wsdl:message>
```

6.68.3 Definition of port types (portType)

```
<wsdl:operation name="OXERPSetAttribute">
  <wsdl:input message="tns:OXERPSetAttributeSoapIn" />
  <wsdl:output message="tns:OXERPSetAttributeSoapOut" />
</wsdl:operation>
```

6.68.4 Definition of binding (binding)

```
<wsdl:operation name="OXERPSetAttribute">
  <soap:operation soapAction="OXERPService/OXERPSetAttribute" style="document" />
  <wsdl:input>
    <soap:body use="literal" />
  </wsdl:input>
```

```

<wsdl:output>
  <soap:body use="literal" />
</wsdl:output>
</wsdl:operation>

```

and

```

<wsdl:operation name="OXERPSetAttribute">
  <soap12:operation soapAction="OXERPService/OXERPSetAttribute" style="document" />
  <wsdl:input>
    <soap12:body use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal" />
  </wsdl:output>
</wsdl:operation>

```

6.69 Implementing function in oxerpgateway.php

```

public function OXERPSetAttribute($oParameters)
{
    try{
        $this->_loadSession($oParameters->sSessionID);
        $aAttribute = $this->_formatInput($oParameters->aAttribute);
        $oResult = $this->_import('attribute',$aAttribute);
    }
    catch (Exception $e) {
        $oResult = $this->_exception2OxErpReturn($e);
    }
    return $this->_formatResult($oResult,'OXERPSetAttributeResult',true);
}

```

6.70 Import definition in oxerpsoap.php

In *oxerpsoap.php* the function `_ImportAttribute` has to be implemented:

```

protected function _ImportAttribute(oxERPTypes & $oType, $aRow)
{
    $sResult = $this->_Save($oType, $aRow);
    $this->_aResults[$this->_iIdx] = $sResult;
    return (boolean) $sResult;
}

```

6.71 Usage of new OXERPSetAttribute function

Now attributes can be created by `OXERPSetAttribute` function:

6.71.1 Parameters

Name	Type	Content
sSessionID	String	The session-ID returned by <code>OXERPLogin</code> .
aAttribute	OXERPTYPE[]	Each OXERPTYPE contains the fields of <i>oxattribute</i> table.

6.71.2 Return values

Name	Type	Content
OXERPSetAttributeResult	OXERPTYPE[]	Each OXERPTYPE contains a particular <code>OXID</code> . <code>blResult</code> indicates if each operation was successful.