

**OXID eSales
Documentation
ERP interface (CSV)
Protocol version 2.14.0**

Copyright

Copyright © 2017 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.

Legal Notice

OXID eSales AG

Bertoldstraße 48

79098 Freiburg

Germany

Fon: +49 (761) 36889 0

Fax: +49 (761) 36889 29

Represented by the Management Board: Roland Fesenmayr (CEO), Dr. Marcus Klosterberg

Supervisory Board: Michael Schlenk (Chairman)

Commercial Register Freiburg

No. HRB 701648

Table of Contents

Copyright	2
Conventions.....	2
Legal Notice	2
Table of Contents	3
About this document	6
About the interface	6
1 Installation	6
1.1 Activate module in OXID eShop	6
2 File format conventions for import	7
2.1 Data field specifications.....	7
2.2 Replacement table.....	7
2.3 Import file requirements	7
3 Importing	7
3.1 By Cronjob	7
4 Import interface	7
4.1 Changes in multishop handling.....	7
4.2 Version line	9
4.3 Create/update products	9
4.4 Update extended properties of products	16
4.5 Delete products.....	16
4.6 Create/Update categories	17
4.7 Delete categories	19
4.8 Create/Update vendors	19
4.9 Delete Vendor.....	20
4.10 Create/Update manufacturers.....	20
4.11 Delete Manufacturer	21
4.12 Create/update crossselling assignment	22
4.13 Delete Crossselling assignment	22
4.14 Create/Update Accessory assignment	22
4.15 Delete Accessory Assignment.....	22
4.16 Create/Update File records	23
4.17 Delete File record	24
4.18 Create/Update Category assignments	24
4.19 Set main category	25
4.20 Delete Category assignment	25
4.21 Create/update action assignment	25
4.22 Delete action assignment	26
4.23 Create/update scale prices	26
4.24 Delete scale prices	26
4.25 Create/update orders.....	27
4.26 Delete orders.....	31

4.27	Create/update product assignments to orders.....	31
4.28	Delete product assignments to orders	33
4.29	Create/Update file assignments to orders	34
4.30	Delete file assignments to orders	34
4.31	Change order status	35
4.32	Create/update product assignments to vendors.....	35
4.33	Delete product assignments to vendors.....	35
4.34	Create/update product assignments to manufacturers	36
4.35	Delete product assignments to manufacturers.....	36
4.36	Updating stock.....	36
4.37	Create/update users	36
4.38	Delete users	38
4.39	Create/Update countries	39
4.40	Delete Countries	40
4.41	Create/Update attributes.....	40
4.42	Delete Attributes	40
4.43	Create/Update Attribute assignments.....	40
4.44	Delete Attribute assignment	41
4.45	Create/Update contents	41
4.46	Delete Contents	42
4.47	Create/Update contents	43
5	Export interface.....	44
5.1	Changes in multishop handling.....	44
5.2	Sorting export data	44
5.3	Login	44
5.4	OXERPGetCategory.....	45
5.5	OXERPGetCategories	45
5.6	OXERPGetArticle	46
5.7	OXERPGetArticles	46
5.8	OXERPGetFile	46
5.9	OXERPGetFiles	47
5.10	OXERPGetArticleCategory	47
5.11	OXERPGetArticleAction	48
5.12	OXERPGetArticleCrossselling	48
5.13	OXERPGetArticleScalePrice	49
5.14	OXERPGetArticleAccessoire.....	49
5.15	OXERPGetArticleStock.....	50
5.16	OXERPGetOrder	50
5.17	OXERPGetOrderArticle	50
5.18	OXERPGetOrderFiles	51
5.19	OXERPGetVendor	51
5.20	OXERPGetArticleVendor.....	51
5.21	OXERPGetVendorArticles	52
5.22	OXERPGetManufacturer.....	52

5.23	OXERPGetArticleManufacturer	52
5.24	OXERPGetManufacturerArticles	53
5.25	OXERPGetUser	53
5.26	OXERPGetUsers	54
5.27	OXERPGetOrders	55
5.28	OXERPGetVendors	55
5.29	OXERPGetManufacturers	55
5.30	OXERPCallPlugin.....	56
5.31	OXERPGetAttributes.....	56
5.32	OXERPGetArticleAttributes.....	57
5.33	OXERPGetContent	57
6	Using custom fields.....	57
6.1	Example.....	58
6.2	Extending ERP with plugins.....	58
6.3	Creating a plugin.....	58
6.4	Sample plugin.....	58
6.5	Data flow of the sample plugin	59
6.6	oxGetActionArticlesPlugin	60
6.7	oxGetArticleVariantsPlugin.....	60
6.8	oxGetConfigParametersPlugin	60
6.9	oxSearchPlugin	61
7	Expandability of CSV interface.....	62
7.1	Define new attribute object	62
7.2	Implementing function in oxerpgateway.php.....	63
7.3	Import definition in oxerpcsv.php	63
7.4	Using new OXERPSetAttribute function.....	63
7.5	Registering function in oxerpcsvexport.php.....	64

About this document

This document describes the protocol version 2.14.0 of the ERP CSV interface for OXID eShop version 4.9.1/5.2.1 and higher in 5.x generation.

Older protocol versions are further on supported. Valid protocol versions are:

- 2.2.0
- 2.3.0
- 2.4.0
- 2.5.0
- 2.6.0
- 2.7.0
- 2.8.0
- 2.9.0
- 2.10.0
- 2.11.0
- 2.12.0
- 2.13.0
- 2.14.0

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

About the interface

The CSV interface is used for importing and exporting eShop data (e.g. products, categories, users etc.).

- The import is made from files which were uploaded to the server.
- The export is done by calling a particular URL.
- This URL must contain the version parameter `&version=X.X.X`

In `testdata_examples` directory sample import files for each protocol version are given.

1 Installation

Copy all files from *copy_this* to the main directory of your eShop and open eShop admin.

1.1 Activate module in OXID eShop

- Go to *Extensions -> Modules*
- Select module in *Overview* tab and press *Activate* button
- Activation is required for module to work correctly with all features

2 File format conventions for import

- Text files with Windows line breaks
- character encoding: ANSI

2.1 Data field specifications

- All data fields are enclosed in double quotes and separated by a semicolon.
- For numbers, a point is used as decimal separator.
- Leading zeros are omitted.
- For fields filled with 0 or 1, 0 always means no, 1 means yes.
- Named data types are taken from MySQL database.

2.2 Replacement table

In Import files, the following chars have to be replaced:

char	Description	ASCII-Code	Replacement
CR	Carriage Return	13	
LF	Line Feed	10	

"	Double Quote	34	"
'	Single Quote	39	'

2.3 Import file requirements

The first line has to contain version information. Following data can be in any order.

3 Importing

3.1 By Cronjob

If import is to be run by a cronjob, adjust

`modules/path/to/ERP_package/examplescripts/oxerpccron.php` according to your needs. Change file permissions to executable.

4 Import interface

At the beginning of each dataset, a char which indicates the data type is set. This char will be called "Indicator char" in the following text.

4.1 Changes in multishop handling

Starting from ERP interface version 2.14.0, the database fields OXSHOPINCL/OXSHOPEXCL are no longer supported. Instead of having these fields, the multishop tables now have mapping tables (i.e. oxarticles table has oxarticles2shop mapping table) and the new field OXINHERITEDSHOPIDS is used.

When importing data from version 2.14.0, OXINHERITEDSHOPIDS will contain serialized array of shop IDs where this object is assigned to (i.e. oxarticle for shop 1 would have this value - 'a:1:{i:0;i:1;}'). When importing from older versions, where OXSHOPINCL/OXSHOPEXCL fields are still present, the conversion to the new multishop functionality will be handled from within ERP itself.

Note (For Professional Edition):

OXINHERITEDSHOPIDS must be left blank.

4.2 Version line

The first line always has to contain information about the used protocol version. Example:

"v";"2.14.0"

#	Content/Example	Data type	Range	Field name
0.	Indicator char	char(1)	[V]	
1.	Version number	char(255) NOT NULL default ""	[0-9.]	

4.3 Create/update products

If a product with the given OXID already exists, the product is updated with the given values. Otherwise, a new product is created.

#	Content/Example	Data type	Range	Field name ¹
0.	Indicator char	char(1)	[A]	
1.	Product ID	char(32) NOT NULL	[0-9a-f.]	OXID
2.	Shop ID	int(11) NOT NULL default 1	1 - 64	OXSHOPID ²
3.	Inherited Shop IDs	See Chapter 4.1	See Chapter 4.1	OXINHERITEDSHOPIDS
4.	Parent-ID For variants: OXID of parent product	char(32) NOT NULL default ""	[0-9a-f.]	OXPARENTID
5.	Active	tinyint(1) NOT NULL default '1'	[0,1]	OXACTIVE
6.	Active from	datetime default 0000-00-00 00:00:00	YYYY-MM-DD HH:MM:SS	OXACTIVEFROM
7.	Active to	datetime default 0000-00-00 00:00:00	YYYY-MM-DD HH:MM:SS	OXACTIVETO
8.	Product number	varchar(255) NOT NULL default ""	[SPACE-~] ³	OXARTNUM
9.	EAN	varchar(13) NOT NULL default ""	[0-9]	OXEAN
10.	EAN of distributor	varchar(13) NOT NULL default ""	[0-9]	OXDISTEAN
11.	EAN of manufacturer	varchar(100) NOT NULL default ""	[0-9]	OXMPN
12.	Title	varchar(255) NOT NULL default ""	[SPACE-~]	OXTITLE

¹ Table: oxarticles

² Always 1 in OXID eShop Professional Edition

³ ASCII chr(32) – chr(126)

#	Content/Example	Data type	Range	Field name ¹
13.	Short description	varchar(255) NOT NULL default ''	[SPACE-~]	OXSHORTDESC
14.	Price	double NOT NULL default '0'	- 1.7976931348623157 E+308 to - 2.2250738585072014 E-308, 0, and 2.2250738585072014 E-308 to 1.7976931348623157 E+308 ⁴	OXPRICE
15.	Product has fixed price	tinyint(1) NOT NULL default '0'	[0,1]	OXBLFIXEDPRICE
16.	Price A	double NOT NULL default '0'	- 1.7976931348623157 E+308 to - 2.2250738585072014 E-308, 0, and 2.2250738585072014 E-308 to 1.7976931348623157 E+308	OXPRICEA
17.	Price B	double NOT NULL default '0'	see above	OXPRICEB
18.	Price C	double NOT NULL default '0'	see above	OXPRICEC
19.	Purchase price	double NOT NULL default '0'	see above	OXBPRICE
20.	RRP	double NOT NULL default '0'	see above	OXTPRICE
21.	Unit	varchar(32) NOT NULL default ''	[SPACE-~]	OXUNITNAME
22.	Quantity	double NOT NULL default '0'	see field 15	OXUNITQUANTITY
23.	External URL	varchar(255) NOT NULL default ''	valid URL without <i>http://</i>	OXEXTURL

⁴ Ranges defined by IEEE. The range that can be used is determined by hardware and operating system.

#	Content/Example	Data type	Range	Field name ¹
24.	Short description for external URL	varchar(255) NOT NULL default ""	[SPACE-~]	OXURLDESC
25.	URL of picture for external URL	varchar(128) NOT NULL default ""	[SPACE-~]	OXURLIMG
26.	Special VAT for this product	float default NULL	see field 15	OXVAT
27.	Filename (without path), Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ""	[SPACE-~] ⁵	OXTHUMB
28.	Filename (without path), Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ""	[SPACE-~]	OXICON
29.	Filename (without path), Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ""	[SPACE-~]	XPIC1
30.	Filename (without path) , Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ""	[SPACE-~]	XPIC2
31.	Filename (without path), Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ""	[SPACE-~]	XPIC3
32.	Filename (without path), Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ""	[SPACE-~]	XPIC4
33.	Filename (without path), Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ""	[SPACE-~]	XPIC5
34.	Filename (without path), Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ""	[SPACE-~]	XPIC6
35.	Filename (without path), Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ""	[SPACE-~]	XPIC7
36.	Filename (without path), Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ""	[SPACE-~]	XPIC8
37.	Filename (without path), Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ""	[SPACE-~]	XPIC9

⁵ Dependant on the file system

#	Content/Example	Data type	Range	Field name ¹
38.	Filename (without path), Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ''	[SPACE-~]	OXPIC10
39.	Filename (without path), Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ''	[SPACE-~]	OXPIC11
40.	Filename (without path), Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ''	[SPACE-~]	OXPIC12
41.	Weight	double NOT NULL default '0'	see field 15	OXWEIGHT
42.	Stock	int(11) NOT NULL default '-1' ⁶	0 – 2147483647	OXSTOCK
43.	Delivery Status	enum('1','2','3','4') NOT NULL default '1'	1. Standard 2. If out of Stock, offline 3. If out of Stock, not orderable 4. External Storehouse	OXSTOCKFLAG
44.	In Stock Message	varchar(255) NOT NULL default ''	[SPACE-~]	OXSTOCKTEXT
45.	Out of Stock Message	varchar(255) NOT NULL default ''	[SPACE-~]	OXNOSTOCKTEXT
46.	Available on	date NOT NULL default '0000-00-00'	YYYY-MM-DD	OXDELIVERY
47.	Date of creation (is ignored, actual date taken instead)	date NOT NULL default '0000-00-00'	YYYY-MM-DD	OXINSERT
48.	Date of last change (is ignored, actual date taken instead)	timestamp NOT NULL default CURRENT_TIMESTAMP on update CURRENT_TIMESTAMP	YYYY-MM-DD HH:MM:SS	OXTIMESTAMP
49.	Length	double NOT NULL default '0'	see field 15	OXLENGTH
50.	Width	double NOT NULL default '0'	see field 15	OXWIDTH

⁶ -1 was a special value in older versions. Negative values are not to be used in Enterprise Edition any more.

#	Content/Example	Data type	Range	Field name ¹
51.	Height	double NOT NULL default '0'	see field 15	OXHEIGHT
52.	File	varchar(128) NOT NULL default ''	[SPACE-~]	OXFILE
53.	Search Terms	varchar(255) NOT NULL default ''	[SPACE-~]	OXSEARCHKEYS
54.	Alt. Template	varchar(128) NOT NULL default ''	[SPACE-~]	OXTEMPLATE
55.	Alt. Contact	varchar(255) NOT NULL default ''	valid e-Mail address	OXQUESTIONEMAIL
56.	Searchable	tinyint(1) NOT NULL default '1'	[0,1]	OXISSEARCH
57.	Is product configurable with persistent parameters, Default '0'	tinyint(1) NOT NULL default '0'	[0,1]	OXISCONFIGURABLE
58.	For variants: Name of variant selection (e.g. Color)	varchar(255) NOT NULL default ''	[SPACE-~]	OXVARNAME
59.	Stock of all variants	int(5) NOT NULL default '0'	0 – 2147483647	OXVARSTOCK
60.	Amount of variants	int(1) NOT NULL default '0'	0 – 2147483647	OXVARCOUNT
61.	Name of variant (e.g. <i>green</i>)	varchar(255) NOT NULL default ''	[SPACE-~]	OXVARSELECT
62.	lowest price of all variants	double NOT NULL default '0'	see field 15	OXVARMINPRICE
63.	highest price of all variants	double NOT NULL default '0'	see field 15	OXVARMAXPRICE
64.	Name of variant selection in language 1	varchar(255) NOT NULL default ''	[SPACE-~]	OXVARNAME_1
65.	Name of variant in language 1	varchar(255) NOT NULL default ''	[SPACE-~]	OXVARSELECT_1
66.	Name of variant selection in language 2	varchar(255) NOT NULL default ''	[SPACE-~]	OXVARNAME_2
67.	Name of variant in language 2	varchar(255) NOT NULL default ''	[SPACE-~]	OXVARSELECT_2
68.	Name of variant selection in language 3	varchar(255) NOT NULL default ''	[SPACE-~]	OXVARNAME_3
69.	Name of variant in language 2	varchar(255) NOT NULL default ''	[SPACE-~]	OXVARSELECT_3
70.	Title in language 1	varchar(255) NOT NULL default ''	[SPACE-~]	OXTITLE_1
71.	Short description in language 1	varchar(255) NOT NULL default ''	[SPACE-~]	OXSHORTDESC_1

#	Content/Example	Data type	Range	Field name ¹
72.	URL description in language 1	varchar(255) NOT NULL default ''	[SPACE-~]	OXURLDESC_1
73.	Search terms in language 1	varchar(255) NOT NULL default ''	[SPACE-~]	OXSEARCHKEYS_1
74.	Title in language 2	varchar(255) NOT NULL default ''	[SPACE-~]	OXTITLE_2
75.	Short description in language 2	varchar(255) NOT NULL default ''	[SPACE-~]	OXSHORTDESC_2
76.	URL description in language 2	varchar(255) NOT NULL default ''	[SPACE-~]	OXURLDESC_2
77.	Search terms in language 2	varchar(255) NOT NULL default ''	[SPACE-~]	OXSEARCHKEYS_2
78.	Title in language 3	varchar(255) NOT NULL default ''	[SPACE-~]	OXTITLE_3
79.	Short description in language 3	varchar(255) NOT NULL default ''	[SPACE-~]	OXSHORTDESC_3
80.	URL description in language 3	varchar(255) NOT NULL default ''	[SPACE-~]	OXURLDESC_3
81.	Search terms in language 3	varchar(255) NOT NULL default ''	[SPACE-~]	OXSEARCHKEYS_3
82.	(always empty)	varchar(32) NOT NULL default ''	always empty	OXFOLDER
83.	class	varchar(32) NOT NULL default ''	always <i>oxarticle</i>	OXSUBCLASS
84.	In Stock Message in language 1	varchar(255) NOT NULL default ''	[SPACE-~]	OXSTOCKTEXT_1
85.	In Stock Message in language 2	varchar(255) NOT NULL default ''	[SPACE-~]	OXSTOCKTEXT_2
86.	In Stock Message in language 3	varchar(255) NOT NULL default ''	[SPACE-~]	OXSTOCKTEXT_3
87.	Out of Stock Message in language 1	varchar(255) NOT NULL default ''	[SPACE-~]	OXNOSTOCKTEXT_1
88.	Out of Stock Message in language 2	varchar(255) NOT NULL default ''	[SPACE-~]	OXNOSTOCKTEXT_2
89.	Out of Stock Message in language 3	varchar(255) NOT NULL default ''	[SPACE-~]	OXNOSTOCKTEXT_3
90.	Sorting	int(5) NOT NULL default '0'	0 – 2147483647	OXSORT
91.	Amount of sold units	double NOT NULL default '0'	see field 15	OXSOLDAMOUNT
92.	Intangible Product	int(1) NOT NULL default '0'	[0,1]	OXNONMATERIAL
93.	Free Shipping	int(1) NOT NULL default '0'	[0,1]	OXFREESHIPPING

#	Content/Example	Data type	Range	Field name ¹
94.	Send e-mail if Stock falls below Value	int(1) NOT NULL default '0'	[0,1]	OXREMINDACTIVE
95.	Value for low Stock e-mail	double NOT NULL default '0'	see field 15	OXREMINDAMOUNT
96.	(always empty)	varchar(32) NOT NULL default ''	always empty	OXAMITEMID
97.	(always empty)	varchar(16) NOT NULL default '0'	always empty	OXAMTASKID
98.	Vendor ID	varchar(32) NOT NULL default ''	[0-9a-f.]	OXVENDORID
99.	Manufacturer ID	char(32) NOT NULL default ''	[0-9a-f.]	OXMANUFACTURERID
100.	Skip all negative Discounts	tinyint(1) NOT NULL default '0'	[0,1]	OXSKIPDISCOUNTS
101.	Additional info in order confirmation	varchar(255) NOT NULL	[SPACE-~]	OXORDERINFO
102.	Export to pixi*	tinyint(4) NOT NULL default '0'	[0,1]	OXPIXIEXPORT
103.	Export to pixi* timestamp	timestamp NOT NULL default '0000-00-00 00:00:00'	YYYY-MM-DD HH:MM:SS	OXPIXIEXPORTED ⁷
104.	Packaging unit	int(11) NOT NULL default '1'	0 – 2147483647	OXVPE
105.	Average rating	Double NOT NULL default '0'	see field 15	OXRATING
106.	Amount of ratings	Int (11) NOT NULL default '0'	0 – 2147483647	OXRATINGCNT
107.	Minimal delivery time	Int (11) NOT NULL default '0'	0 – 2147483647	OXMINDELTIME
108.	Maximal delivery time	Int (11) NOT NULL default '0'	0 – 2147483647	OXMAXDELTIME
109.	Delivery time unit	varchar(255) NOT NULL default ''	[SPACE-~]	OXDELTIMEUNIT
110.	Long description	text	[SPACE-~]	OXLONGDESC
111.	Long description in language 1	text NOT NULL	[SPACE-~]	OXLONGDESC_1
112.	Long description in language 2	text NOT NULL	[SPACE-~]	OXLONGDESC_2
113.	Long description in language 3	text NOT NULL	[SPACE-~]	OXLONGDESC_3

7

In OXID eShop Professional Edition always 0000-00-00 00:00:00

#	Content/Example	Data type	Range	Field name ¹
114.	Update price	double NOT NULL default '0'	see field 16	OXUPDATEPRICE
115.	Update Price A	double NOT NULL default '0'	see field 16	OXUPDATEPRICEA
116.	Update price B	double NOT NULL default '0'	see field 16	OXUPDATEPRICEB
117.	Update price C	double NOT NULL default '0'	see field 16	OXUPDATEPRICEC
118.	Update price time	timestamp NOT NULL default '0000-00-00 00:00:00'	YYYY-MM-DD HH:MM:SS	OXUPDATEPRICETIME
119.	Is this product downloadable	tinyint(1) NOT NULL default '0'	[0,1]	OXISDOWNLOADABLE
120.	Has custom agreement	tinyint(1) NOT NULL default '1'	[0,1]	OXSHOWCUSTOMAGREEMENT

4.4 Update extended properties of products

With this function long descriptions and tags can be imported for existing products.

#	Content/Example	Data type	Range	Field name ⁸
0.	Indicator char	char(1)	[Y]	
1.	Product ID	char(32) NOT NULL	[0-9a-f.]	OXID
2.	long description	text	[SPACE-~]	OXLONGDESC
3.	Long description in language 1	text NOT NULL	[SPACE-~]	OXLONGDESC_1
4.	Long description in language 2	text NOT NULL	[SPACE-~]	OXLONGDESC_2
5.	Long description in language 3	text NOT NULL	[SPACE-~]	OXLONGDESC_3
6.	Tags	varchar(255) NOT NULL	[SPACE-~]	OXTAGS
7.	Tags in language 1	varchar(255) NOT NULL	[SPACE-~]	OXTAGS_1
8.	Tags in language 2	varchar(255) NOT NULL	[SPACE-~]	OXTAGS_2
9.	Tags in language 3	varchar(255) NOT NULL	[SPACE-~]	OXTAGS_3

4.5 Delete products

When deleting a product, all assignments for this product (crossselling, categories etc.) are deleted from database aswell.

#	Content/Example	Data type	Range	Field name
---	-----------------	-----------	-------	------------

⁸ Table: oxartextends

0	Indicator char	char(2)	[XA]	
1	Product number	char(32) NOT NULL default ''	[0-9a-f.]	OXID

4.6 Create/Update categories

If a category with the given OXID already exists, the category is updated with the given values. Otherwise, a new category is created.

#	Content/Example	Data type	Range	Field name ⁹
0.	Indicator char	char(1)	[K]	
1.	Category ID	varchar(32) NOT NULL default ''	[0-9a-f.]	OXID
2.	Parent category ID	varchar(32) NOT NULL default ''	[0-9a-f.]	OXPARENTID
3.	Nested category tree left	int(11) NOT NULL default '0'	0 to 2147483647	OXLEFT
4.	Nested category tree right	int(11) NOT NULL default '0'	0 to 2147483647	OXRIGHT
5.	Nested Category tree root	varchar(32) NOT NULL default ''	[0-9a-f.]	OXROOTID
6.	Sorting	int(11) NOT NULL default '9999'	0 to 2147483647	OXSORT
7.	Active	tinyint(1) NOT NULL default '1'	[0,1]	OXACTIVE
8.	Hidden	tinyint(1) NOT NULL default '0'	[0,1]	OXHIDDEN
9.	Shop-ID of Shop in which Category was created	int(11) NOT NULL default 1	1 to 64	OXSHOPID
10.	Inherited Shop IDs	See Chapter 4.1	See Chapter 4.1	OXINHERITEDSHOPS
11.	Title	varchar(254) NOT NULL default ''	[SPACE-~]	OXTITLE
12.	Short description	varchar(255) NOT NULL default ''	[SPACE-~]	OXDESC
13.	Long description	text NOT NULL	[SPACE-~]	OXLONGDESC
14.	Filename of thumbnail (without path)	varchar(128) NOT NULL default ''	[SPACE-~]	OXTHUMB
15.	Filename of thumbnail (without path) in language 1	varchar(128) NOT NULL default ''	[SPACE-~]	OXTHUMB_1

⁹ Table: oxcategories

#	Content/Example	Data type	Range	Field name ⁹
16.	Filename of thumbnail (without path) in language 2	varchar(128) NOT NULL default ''	[SPACE-~]	OXTHUMB_2
17.	Filename of thumbnail (without path) in language 3	varchar(128) NOT NULL default ''	[SPACE-~]	OXTHUMB_3
18.	External URL	varchar(255) NOT NULL default ''	valid URL	OXEXTLINK
19.	Alt. Template	varchar(128) NOT NULL default ''	[SPACE-~]	OXTEMPLATE
20.	Fast sorting(field name from oxarticles)	varchar(64) NOT NULL default ''	[a-z]	OXDEFSORT
21.	sorting mode (asc/desc)	tinyint(1) NOT NULL default '0'	[0,1]	OXDEFSORTMODE
22.	Price from	double	- 1.7976931348623157 E+308 to - 2.2250738585072014 E-308, 0, and 2.2250738585072014 E-308 to 1.7976931348623157 E+308	OXPRICEFROM
23.	Price to	double NOT NULL	see above	OXPRICETO
24.	Active in language 1	char(1) NOT NULL default ''	[0,1]	OXACTIVE_1
25.	Title in language 1	varchar(255) NOT NULL default ''	[SPACE-~]	OXTITLE_1
26.	Short description in language 1	varchar(255) NOT NULL default ''	[SPACE-~]	OXDESC_1
27.	Long description in language 1	text NOT NULL	[SPACE-~]	OXLONGDESC_1
28.	Active in language 2	char(1) NOT NULL default ''	[0,1]	OXACTIVE_2

#	Content/Example	Data type	Range	Field name ⁹
29.	Title in language 2	varchar(255) NOT NULL default ''	[SPACE-~]	OXTITLE_2
30.	Short description in language 2	varchar(255) NOT NULL default ''	[SPACE-~]	OXDESC_2
31.	Long description in language 2	text NOT NULL	[SPACE-~]	OXLONGDESC_2
32.	Active in language 3	char(1) NOT NULL default ''	[0,1]	OXACTIVE_3
33.	Title in language 3	varchar(255) NOT NULL default ''	[SPACE-~]	OXTITLE_3
34.	Short description in language 3	varchar(255) NOT NULL default ''	[SPACE-~]	OXDESC_3
35.	Long description in language 3	text NOT NULL	[SPACE-~]	OXLONGDESC_3
36.	Filename of category image (without path)	varchar(128) NOT NULL	[SPACE-~]	OXICON
37.	Filename of category promotion icon	varchar(128) NOT NULL	[SPACE-~]	OXPROMOICON
38.	Special VAT	float default NULL		OXVAT
39.	Skip all negative Discounts	tinyint(1) NOT NULL default '0'	[0,1]	OXSKIPDISCOUNTS
40.	Show SEO Suffix in Category	tinyint(1) NOT NULL default '1'	[0,1]	OXSHOWSUFFIX

4.7 Delete categories

#	Content/Example	Data type	Range	Field name
0.	Indicator char	char(2)	[XK]	
1.	Category ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID

4.8 Create/Update vendors

If a vendor with the given OXID already exists, the vendor is updated with the given values. Otherwise, a new vendor is created.

#	Content/Example	Data type	Range	Field name ¹⁰
0.	Indicator char	char(1)	[H]	
1.	Vendor ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID
2.	Shop ID of Shop in which vendor was created	int(11) NOT NULL default 1	1 to 64	OXSHOPID
3.	Inherited Shop IDs	See Chapter 4.1	See Chapter 4.1	OXINHERITEDSHOPIDS
4.	Active	tinyint(1) NOT NULL default '1'	[0,1]	OXACTIVE
5.	Filename of Icon	char(128) NOT NULL default ''	[SPACE-~]	OXICON
6.	Title	char(255) NOT NULL default ''	[SPACE-~]	OXTITLE
7.	Short description	char(255) NOT NULL default ''	[SPACE-~]	OXSHORTDESC
8.	Title in language 1	char(255) NOT NULL default ''	[SPACE-~]	OXTITLE_1
9.	Short description in language 1	char(255) NOT NULL default ''	[SPACE-~]	OXSHORTDESC_1
10.	Title in language 2	char(255) NOT NULL default ''	[SPACE-~]	OXTITLE_2
11.	Short description in language 2	char(255) NOT NULL default ''	[SPACE-~]	OXSHORTDESC_2
12.	Title in language 3	char(255) NOT NULL default ''	[SPACE-~]	OXTITLE_3
13.	Short description in language 3	char(255) NOT NULL default ''	[SPACE-~]	OXSHORTDESC_3
14.	Show SEO suffix	tinyint(1) NOT NULL default '1'	[0,1]	OXSHOWSUFFIX

4.9 Delete Vendor

#	Content/Example	Data type	Range	Field name
	Indicator char	char(2)	[XH]	
	Vendor ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID

4.10 Create/Update manufacturers

If a manufacturer with the given OXID already exists, the manufacturer is updated with the given values. Otherwise, a new manufacturer is created.

#	Content/Example	Data type	Range	Field name ¹¹
---	-----------------	-----------	-------	--------------------------

¹⁰

Table: oxvendor

#	Content/Example	Data type	Range	Field name ¹¹
15.	Indicator char	char(1)	[V]	
16.	Manufacturer ID	char(32) NOT NULL	[0-9a-f.]	OXID
17.	Shop ID of Shop in which manufacturer was created	int(11) NOT NULL default 1	1 to 64	OXSHOPID
18.	Inherited Shop IDs	See Chapter 4.1	See Chapter 4.1	OXINHERITEDSHOPIDS
19.	Active	tinyint(1) NOT NULL default '1'	[0,1]	OXACTIVE
20.	Filename of Icon	char(128) NOT NULL default ''	[SPACE-~]	OXICON
21.	Title	char(255) NOT NULL default ''	[SPACE-~]	OXTITLE
22.	Short description	char(255) NOT NULL default ''	[SPACE-~]	OXSHORTDESC
23.	Title in language 1	char(255) NOT NULL default ''	[SPACE-~]	OXTITLE_1
24.	Short description in language 1	char(255) NOT NULL default ''	[SPACE-~]	OXSHORTDESC_1
25.	Title in language 2	char(255) NOT NULL default ''	[SPACE-~]	OXTITLE_2
26.	Short description in language 2	char(255) NOT NULL default ''	[SPACE-~]	OXSHORTDESC_2
27.	Title in language 3	char(255) NOT NULL default ''	[SPACE-~]	OXTITLE_3
28.	Short description in language 3	char(255) NOT NULL default ''	[SPACE-~]	OXSHORTDESC_3
29.	Show SEO suffix	tinyint(1) NOT NULL default '1'	[0,1]	OXSHOWSUFFIX

4.11 Delete Manufacturer

#	Content/Example	Data type	Range	Field name
	Indicator char	char(2)	[XV]	
	Manufacturer ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID

¹¹

Table: oxmanufacturers

4.12 Create/update crossselling assignment

If a crossselling assignment with the given OXID already exists, the crossselling assignment is updated with the given values. Otherwise, a new crossselling assignment is created.

#	Content/Example	Data type	Range	Field name ¹²
0.	Indicator char	char(1)	[C]	
1.	ID of product which is assigned as crossselling	char(32) NOT NULL default ''	[SPACE-~]	OXOBJECTID
2.	ID of the product to which the other product is assigned to.	char(32) NOT NULL default ''	[SPACE-~]	OXARTICLEID
3.	Sorting	int(5) NOT NULL default '0'	0 to 2147483647	OXSORT
4.	Crossselling assignment ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID

4.13 Delete Crossselling assignment

#	Content/Example	Data type	Range	Field name
	Indicator char	char(2)	[XC]	
	Crossselling assignment ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID

4.14 Create/Update Accessory assignment

If an accessory assignment with the given OXID already exists, the accessory assignment is updated with the given values. Otherwise, a new accessory assignment is created.

#	Content/Example	Data type	Range	Field name ¹³
0.	Indicator char	char(1)	[Z]	
1.	ID of product which is assigned as accessory	char(32) NOT NULL default ''	[SPACE-~]	OXOBJECTID
2.	ID of the product to which the other product is assigned to.	char(32) NOT NULL default ''	[SPACE-~]	OXARTICLEID
3.	Sorting	int(5) NOT NULL default '0'	0 to 2147483647	OXSORT
4.	Accessory assignment ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID

4.15 Delete Accessory Assignment

#	Content/Example	Data type	Range	Field name
	Indicator char	char(2)	[XZ]	
	Accessory assignment ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID

¹² Table: oxobject2article

¹³ Table: oxaccessoire2article

4.16 Create/Update File records

If a file record with the given OXID already exists, the file record is updated with the given values. Otherwise, a new file record is created.

# ¹⁴	Content/Example	Data type	Range	Field name ¹⁵
0.	Indicator char	char(1)	[J]	
1.	File ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID
2.	Product ID	char(32) NOT NULL default ''	[0-9a-f.]	OXARTID
3. ¹⁶	File name (without path)	varchar(128) NOT NULL default ''	[0-9a-f.]	OXFILENAME
4. ¹⁷	File store content MD5 hash, for manual uploads leave empty	char(32) NOT NULL default ''	[0-9a-f.]	OXSTOREHASH
5.	File can be downloaded only after it's paid for it	tinyint(4) NOT NULL default '0'	[0,1]	OXPURCHASEDONLY
6.	Maximum downloads count	int(11) NOT NULL default '0'	-1 to 2147483647	OXMAXDOWNLOADS
7.	Maximum unregistered user download	int(11) NOT NULL default '0'	-1 to 2147483647	OXMAXUNREGDOWNLOADS
8.	Link expire time	int(11) NOT NULL default '0'	-1 to 2147483647	OXLINKEXPTIME
9.	Download expire time	int(11) NOT NULL default '0'	-1 to 2147483647	OXDOWNLOADEXPTIME

Example of creating new downloadable product file record:

1. Upload file via FTP or other file transfer protocol to shop folder: "/out/downloads/uploads/".
2. For manually uploaded files field "OXSTOREHASH" **must be empty**.
3. Specify only file name with extension (without path) in the field "OXFILENAME".
4. Call CSV import with similar data:

"V";"2.14.0"

"J";"myFileID";"myProductID";"myUploadedFileName.ext";"";"0";"-1";"-1";"-1";"-1"

Important: Do not change field "OXSTOREHASH" content for files, uploaded via admin interface, because this field specifies file download location.

¹⁴ Note for 6, 7, 8, 9 attributes: "-1" value, means that shop general configuration options will be used.

Expiration time (8, 9 attributes) is set in hours.

¹⁵ Table: oxfiles

¹⁶ For manually uploaded files specifies file name, placed in folder "/out/downloads/uploads/". For files, uploaded via admin interface only specifies download file name, file path information is encoded in OXSTOREHASH field.

¹⁷ For manually uploaded files, placed in folder "/out/downloads/uploads/" this field **must be empty**, for files, uploaded via admin interface leave as it is.

4.17 Delete File record

#	Content/Example	Data type	Range	Field name
	Indicator char	char(2)	[XJ]	
	File ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID

Before deleting record make sure, that there are no orders with reference to file record.

4.18 Create/Update Category assignments

If a category assignment with the given OXID already exists, the category assignment is updated with the given values. Otherwise, a new category assignment is created.

#	Content/Example	Data type	Range	Field name ¹⁸
0.	Indicator char	char(1)	[T]	
1.	category assignment ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID
2.	Shop-ID	int(11) NOT NULL default 1	1 to 64	OXSHOPID
3.	Inherited Shop IDs	See Chapter 4.1	See Chapter 4.1	OXINHERITEDSHOP IDS
4.	Product ID	char(32) NOT NULL default ''	[0-9a-f.]	OXOBJECTID
5.	Category ID	char(32) NOT NULL default ''	[0-9a-f.]	OXCATNID
6.	Sorting	int(11) NOT NULL default '0'	0 to 2147483647	OXPOS
7.	Timestamp (The entry with the oldest timestamp is automatically set as main category)	int(11) NOT NULL default '0'	0 to 2147483647	OXTIME

¹⁸ Table: oxobject2category

4.19 Set main category

If a main category is already set, the setting is overwritten.

#	Content/Example	Data type	Range	Field name
0.	Indicator char	char(1)	[M]	
1.	Product ID	char(32) NOT NULL default ''	[0-9a-f.]	OXOBJECTID
2.	Category ID	char(32) NOT NULL default ''	[0-9a-f.]	OXCATNID

4.20 Delete Category assignment

#	Content/Example	Data type	Range	Field name
	Indicator char	char(2)	[XT]	
	Category Assignment ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID

4.21 Create/update action assignment

If an action assignment with the given OXID already exists, the action assignment is updated with the given values. Otherwise, a new action assignment is created.

#	Content/Example	Data type	Range	Field name ¹⁹
0.	Indicator char	char(1)	[I]	
1.	Shop-ID	int(11) NOT NULL default 1	1 – 64	OXSHOPID
2.	Action ID	char(32) NOT NULL default ''	<u>oxstart</u> front page <u>oxtopstart</u> Top offer front page <u>oxfirststart</u> Big offer front page <u>oxbargain</u> Bargain <u>oxtop5</u> Top of the Shop <u>oxcatoffer</u> Category offer <u>oxnewest</u> Just arrived	OXACTIONID
3.	Product ID	char(32) NOT NULL default ''	[0-9a-f.]	OXARTID
4.	Sorting	int(11) NOT NULL default '0'	0 to 2147483647	OXSORT
5.	Action assignment ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID

¹⁹

Table: oxactions2article

4.22 Delete action assignment

#	Content/Example	Data type	Range	Field name
	Indicator char	char(2)	[XI]	
	Action Assignment ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID

4.23 Create/update scale prices

If a scale price with the given OXID already exists, the scale price is updated with the given values. Otherwise, a new scale price is created.

#	Content/Example	Data type	Range	Field name ²⁰
0.	Indicator char	char(1)	[P]	
1.	Scale price ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID
2.	Shop ID	int(11) NOT NULL default 1	1 to 64	OXSHOPID
3.	Product ID	char(32) NOT NULL default ''	[0-9a-f.]	OXARTID
4.	Absolute surcharge	double NOT NULL default '0'	- 1.797693134862 3157 E+308 to - 2.225073858507 2014 E-308, 0, und 2.225073858507 2014 E-308 to 1.797693134862 3157 E+308	OXADDABS
5.	Relative surcharge in %	double NOT NULL default '0'	siehe Feld 3	OXADDPERC
6.	Amount from	double NOT NULL default '0'	siehe Feld 3	OXAMOUNT
7.	Amount to	double NOT NULL default '0'	siehe Feld 3	OXAMOUNTTO

4.24 Delete scale prices

#	Content/Example	Data type	Range	Field name
	Indicator char	char(2)	[XP]	
	Scale price ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID

²⁰ Table: oxprice2article

4.25 Create/update orders

If an order with the given OXID already exists, the order is updated with the given values. Otherwise, a new order is created.

If `OXORDERNR` is set to 0, the eShop automatically creates a new order number: The highest existing order number is taken and incremented by 1.

#	Content/Example	Data type	Range	Field name
0.	Indicator char	char(1)	[0]	
1.	Order ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID
2.	Shop ID	int(11) NOT NULL default 1	1 to 64	OXSHOPID
3.	User ID	char(32) NOT NULL default ''	[0-9a-f.]	OXUSERID
4.	Order date	datetime NOT NULL default '0000-00-00 00:00:00'	YYYY-MM-DD HH:MM:SS	OXORDERDATE
5.	Order number	int(11) NOT NULL default '0'	0 to 2147483647	OXORDERNR
6.	Billing address: company	varchar(255) NOT NULL default ''	[SPACE-~]	OXBILLCOMPANY
7.	Billing address: e-mail	varchar(255) NOT NULL default ''	Gültige E-Mail- Adresse	OXBILLEMAIL
8.	Billing address: first name	varchar(255) NOT NULL default ''	[SPACE-~]	OXBILLFNAME
9.	Billing address: last name	varchar(255) NOT NULL default ''	[SPACE-~]	OXBILLNAME
10.	Billing address: Street	varchar(255) NOT NULL default ''	[SPACE-~]	OXBILLSTREET
11.	Billing address: street number	varchar(16) NOT NULL default ''	[SPACE-~]	OXBILLSTREETNR
12.	Billing address: Additional information	varchar(255) NOT NULL default ''	[SPACE-~]	OXBILLADDINFO
13.	Billing address: VAT ID	varchar(255) NOT NULL default ''	[SPACE-~]	OXBILLUSTID
14.	VAT ID status	tinyint(1) NOT NULL default '0'	[0;1]	OXBILLUSTIDSTATUS
15.	Billing address: city	varchar(255) NOT NULL default ''	[SPACE-~]	OXBILLCITY
16.	Billing address: country	varchar(32) NOT NULL default ''	[SPACE-~]	OXBILLCOUNTRYID

#	Content/Example	Data type	Range	Field name
17.	Billing address: state	varchar(32) NOT NULL default ''	[SPACE-~]	OXBILLSTATEID
18.	Billing address: ZIP	varchar(16) NOT NULL default ''	[SPACE-~]	OXBILLZIP
19.	Billing address: phone	varchar(128) NOT NULL default ''	[SPACE-~]	OXBILLFON
20.	Billing address: fax	varchar(128) NOT NULL default ''	[SPACE-~]	OXBILLFAX
21.	Billing address: title	varchar(128) NOT NULL default ''	[SPACE-~]	OXBILLSAL
22.	Shipping address: company	varchar(255) NOT NULL default ''	[SPACE-~]	OXDELCOMPANY
23.	Shipping address: first name	varchar(255) NOT NULL default ''	[SPACE-~]	OXDELFNAME
24.	Shipping address: last name	varchar(255) NOT NULL default ''	[SPACE-~]	OXDELLNAME
25.	Shipping address: street	varchar(255) NOT NULL default ''	[SPACE-~]	OXDELSTREET
26.	Shipping address: street number	varchar(16) NOT NULL default ''	[SPACE-~]	OXDELSTREETNR
27.	Shipping address: additional information	varchar(255) NOT NULL default ''	[SPACE-~]	OXDELADDINFO
28.	Shipping address: city	varchar(255) NOT NULL default ''	[SPACE-~]	OXDELCITY
29.	Shipping address: country	varchar(32) NOT NULL default ''	[SPACE-~]	OXDELCOUNTRYID
30.	Shipping address: state	varchar(32) NOT NULL default ''	[SPACE-~]	OXDELSTATEID
31.	Shipping address: ZIP	varchar(16) NOT NULL default ''	[SPACE-~]	OXDELZIP
32.	Shipping address: phone	varchar(128) NOT NULL default ''	[SPACE-~]	OXDELFON
33.	Shipping address: fax	varchar(128) NOT NULL default ''	[SPACE-~]	OXDELFAX
34.	Shipping address: title	varchar(128) NOT NULL default ''	[SPACE-~]	OXDELSAL
35.	Payment ID (from table oxuserpayments)	varchar(32) NOT NULL default ''	[SPACE-~]	OXPAYMENTID
36.	Payment type (from table oxpayments)	varchar(32) NOT NULL default ''	[SPACE-~]	OXPAYMENTTYPE

#	Content/Example	Data type	Range	Field name
37.	Net sum	double NOT NULL default '0'	- 1.79769313486231 57 E+308 to - 2.22507385850720 14 E-308, 0, and 2.22507385850720 14 E-308 to 1.79769313486231 57 E+308	OXTOTALNETSUM
38.	Gross sum	double NOT NULL default '0'	see above	OXTOTALBRUTSUM
39.	Total sum (incl. Shipping costs etc.)	double NOT NULL default '0'	see above	OXTOTALORDERSUM
40.	VAT percent 1	double NOT NULL default '0'	see above	OXARTVAT1
41.	VAT price 1	double NOT NULL default '0'	see above	OXARTVATPRICE1
42.	VAT percent 2	double NOT NULL default '0'	see above	OXARTVAT2
43.	VAT price 2	double NOT NULL default '0'	see above	OXARTVATPRICE2
44.	Shipping costs	double NOT NULL default '0'	see above	OXDELCOST
45.	VAT of shipping costs	double NOT NULL default '0'	see above	OXDELVAT
46.	Payment costs	double NOT NULL default '0'	see above	OXPAYCOST
47.	VAT of payment costs	double NOT NULL default '0'	see above	OXPAYVAT
48.	Gift wrapping costs	double NOT NULL default '0'	see above	OXWRAPCOST
49.	VAT of gift wrapping	double NOT NULL default '0'	see above	OXWRAPVAT
50.	Greeting card ID	varchar(32) NOT NULL default ''	[0-9a-f.]	OXCARDID
51.	Greeting card text	text NOT NULL	[SPACE-~]	OXCARDTEXT
52.	Discount	double NOT NULL default '0'	see field 35	OXDISCOUNT
53.	Export status for Lexware	tinyint(4)	0,1	OXEXPORT

#	Content/Example	Data type	Range	Field name
54.	Invoice number	varchar(128) NOT NULL default ''	[0-9]	OXBILLNR
55.	Tracking code	varchar(128) NOT NULL	[SPACE-~]	OXTRACKCODE
56.	Shipping date	Datetime	YYYY-MM-DD HH:MM:SS	OXSENDDATE
57.	Remark	text NOT NULL	[SPACE-~]	OXREMARK
58.	Coupon discount	double NOT NULL default '0'	see field 35	OXVOUCHERDISCOUNT
59.	Currency	char(32) NOT NULL default ''	[SPACE-~]	OXCURRENCY
60.	Conversion factor	double NOT NULL default '0'	see field 35	OXCURRATE
61.	Order folder in admin	varchar(32)	[SPACE-~]	OXFOLDER
62.	Payment gateway Transaction ID	varchar(64) NOT NULL default ''	[SPACE-~]	OXTRANSID
63.	Payment gateway payment ID	varchar(64) NOT NULL default ''	[SPACE-~]	OXPAYID
64.	Payment gateway X-ID	varchar(64) NOT NULL default ''	[SPACE-~]	OXXID
65.	Paid on	varchar(64) NOT NULL default ''	YYYY-MM-DD HH:MM:SS	OXPAID
66.	Order cancelled	tinyint(1)	[0,1]	OXSTORNO
67.	IP number	varchar(64) NOT NULL default ''	valid IP	OXIP
68.	Payment gateway transaction status	varchar(30) NOT NULL default ''	[SPACE-~]	OXTRANSSTATUS
69.	language	int(2) NOT NULL default '0'	0 to 99	OXLANG
70.	Invoice number	integer(11) NOT NULL	0 to 2147483647	OXINVOICENR
71.	Shipping method	char(32) NOT NULL default ''	[SPACE-~]	OXDELTYPE
72.	Export status for pixi*	tinyint(4) NOT NULL	[0,1]	OXPIXIEXPORT
73.	Trusted shops Buyers Protection product id, e.g. TS080501_500_30_EUR	char(64) NOT NULL default ''	[~]	OXTSPROTECTID
74.	Buyers Protection cost	double NOT NULL default '0'	2.22507385850720 14 E-308 to 1.79769313486231 57 E+308	OXTSPROTECTCOSTS

4.26 Delete orders

#	Content/Example	Data type	Range	Field name
	Indicator char	char(2)	[XO]	
	Order ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID

4.27 Create/update product assignments to orders

If a product assignment with the given OXID already exists, the product assignment is updated with the given values. Otherwise, a new product assignment is created.

#	Content/Example	Data type	Range	Field name ²¹
0.	Indicator char	char(1)	[R]	
1.	Product assignment ID	char(32) NOT NULL	[0-9a-f.]	OXID
2.	Order ID	char(32) NOT NULL	[0-9a-f.]	OXORDERID
3.	Amount	double NOT NULL default '0'	see field 9	OXAMOUNT
4.	Product ID	char(32) NOT NULL	[0-9a-f.]	OXARTID
5.	Product number	varchar(255) NOT NULL	[SPACE-~]	OXARTNUM
6.	Title	varchar(255) NOT NULL	[SPACE-~]	OXTITLE
7.	Short description	varchar(255) NOT NULL	[SPACE-~]	OXSHORTDESC
8.	Selected variant	varchar(255) NOT NULL	[SPACE-~]	OXSELVARIANT
9.	Total sum net	double NOT NULL default '0'	1.7976931348623157 E+308 to - 2.2250738585072014 E-308, 0, und 2.2250738585072014 E-308 to 1.7976931348623157 E+308 ²²	OXNETPRICE
10.	Total sum brut	double NOT NULL default '0'	see field 9	OXBRUTPRICE
11.	VAT Total	double NOT NULL default '0'	see field 9	OXVATPRICE
12.	Special VAT	double NOT NULL	see field 9	OXVAT
13.	Persistent parameters (not used in standard eShop)	Text NOT NULL	[SPACE-~]	OXPERSPARAM

²¹ Table: oxorderarticles

²² Ranges defined by IEEE. The range that can be used is determined by hardware and operating system

#	Content/Example	Data type	Range	Field name ²¹
14.	Product price	double NOT NULL default '0'	see field 9	OXPRICE
15.	Brut product price	double NOT NULL default '0'	see field 9	AXBPRICE
16.	Net product price	double NOT NULL default '0'	see field 9	OXNPRICE
17.	Gift wrapping ID	varchar(32)	[SPACE-~]	OXWRAPID
18.	External URL	varchar(255) NOT NULL default ''	valid URL without <i>http://</i>	OXEXTURL
19.	Description for external URL	varchar(255) NOT NULL default ''	[SPACE-~]	OXURLDESC
20.	URL of picture for external URL	varchar(128) NOT NULL default ''	see field 23	OXURLIMG
21.	Filename (without path) , Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ''	[SPACE-~] ²³	OXTHUMB
22.	Filename (without path) , Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ''	[SPACE-~]	OXPIC1
23.	Filename (without path) , Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ''	[SPACE-~]	OXPIC2
24.	Filename (without path) , Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ''	[SPACE-~]	OXPIC3
25.	Filename (without path) , Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ''	[SPACE-~]	OXPIC4
26.	Filename (without path) , Default: <i>nopic.jpg</i>	varchar(128) NOT NULL default ''	[SPACE-~]	OXPIC5
27.	Weight	double NOT NULL default '0'	see field 9	OXWEIGHT
28.	Stock	int(11) NOT NULL default '-1' ²⁴	-2147483648 to 2147483647	OXSTOCK
29.	Available on	date NOT NULL default '0000-00-00'	YYYY-MM-DD	OXDELIVERY
30.	Date of creation (is ignored, actual date is used instead)	date NOT NULL default '0000-00-00'	YYYY-MM-DD	OXINSERT

²³ Dependent on operating system

²⁴ -1 was a special value in older versions. Negative values are not to be used in Enterprise Edition any more.

#	Content/Example	Data type	Range	Field name ²¹
31.	Date of last change (is ignored, actual date is used instead)	timestamp NOT NULL default CURRENT_TIMESTAMP on update CURRENT_TIMESTAMP	YYYY-MM-DD HH:MM:SS	OXTIMESTAMP
32.	Length	double NOT NULL default '0'	see field 9	OXLENGTH
33.	Width	double NOT NULL default '0'	see field 9	OXWIDTH
34.	Height	double NOT NULL default '0'	see field 9	OXHEIGHT
35.	File	varchar(128) NOT NULL default ''	[SPACE-~]	OXFILE
36.	Search terms	varchar(255) NOT NULL default ''	[SPACE-~]	OXSEARCHKEYS
37.	Alt. template	varchar(128) NOT NULL default ''	[SPACE-~]	OXTEMPLATE
38.	Alt. contact	varchar(255) NOT NULL default ''	valid e-mail address	OXQUESTIONEMAIL
39.	Searchable	tinyint(1) NOT NULL default '1'	[0,1]	OXISSEARCH
40.	(always empty)	varchar(32) NOT NULL default ''	always empty	OXFOLDER
41.	Class	varchar(32) NOT NULL default ''	always <i>oxarticle</i>	OXSUBCLASS
42.	Cancelled	Tinyint(1) NOT NULL default '1'	[0,1]	OXSTORNO
43.	Shop-ID	varchar(32) NOT NULL	[SPACE-~]	OXORDERSHOPID
44.	VAT Total	Double NOT NULL	see field 9	OXTOTALVAT
45.	Order status	text	See 4.29	OXERPSTATUS ²⁵
46.	Bundled article	Tinyint(1) NOT NULL default '0'	[0,1]	OXISBUNDLE

4.28 Delete product assignments to orders

#	Content/Example	Data type	Range	Field name
	Indicator char	char(2)	[XR]	
	Product assignment ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID

²⁵ Always empty in OXID eShop Professional Edition

4.29 Create/Update file assignments to orders

If a file record with the given OXID already exists, the file record is updated with the given values. Otherwise, a new file record is created.

#	Content/Example	Data type	Range	Field name ²⁶
0.	Indicator char	char(1)	[W]	
1.	File assignment ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID
2.	Order ID	char(32) NOT NULL default ''	[0-9a-f.]	OXORDERID
3.	File name (without path)	varchar(128) NOT NULL default ''	[SPACE-~]	OXFILENAME
4.	File ID (existing record ID)	char(32) NOT NULL default ''	[0-9a-f.]	OXFILEID
5.	Shop-ID	int(11) NOT NULL default 1	1 to 64	OXSHOPID
6.	Ordered Product ID	char(32) NOT NULL default ''	[0-9a-f.]	OXORDERARTICLEID
7.	First download timestamp	timestamp NOT NULL default '0000-00-00 00:00:00'	YYYY-MM-DD HH:MM:SS	OXFIRSTDOWNLOAD
8.	Last download timestamp	timestamp NOT NULL default '0000-00-00 00:00:00'	YYYY-MM-DD HH:MM:SS	OXLASTDOWNLOAD
9.	Current download count	int(10) NOT NULL UNSIGNED default '0'	0 to 2147483647	OXDOWNLOADCOUNT
10.	Maximum allowed download count	int(10) NOT NULL UNSIGNED default '0'	0 to 2147483647	OXMAXDOWNLOADCOUNT
11.	Download expire time	int(10) NOT NULL UNSIGNED default '0'	0 to 2147483647	OXDOWNLOADEXPIRATIONTIME
12.	Link expire time	int(10) NOT NULL UNSIGNED default '0'	0 to 2147483647	OXLINKEXPIRATIONTIME
13.	Reset count	int(10) NOT NULL UNSIGNED default '0'	0 to 2147483647	OXRESETCOUNT
14.	Valid until date	Datetime	YYYY-MM-DD HH:MM:SS	OXVALIDUNTIL

4.30 Delete file assignments to orders

#	Content/Example	Data type	Range	Field name
	Indicator char	char(2)	[XW]	
	File assignment ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID

²⁶ Table: oxorderfiles

4.31 Change order status

If an order status with the given OXID already exists, the order status is updated with the given values. Otherwise, a new order status is created.

#	Content/Example	Data type	Range	Field name ²⁷
0.	Indicator char	char(1)	[S]	
1.	ID from oxorderarticles	char(32) NOT NULL default "	[0-9a-f.]	OXID
2.	Status text	text NOT NULL	[SPACE-~]	OXERPSTATUS_STATUS
3.	Date and time	Datetime	YYYY-MM-DD HH:MM:SS	OXERPSTATUS_TIME
4.	Tracking ID	char(32) NOT NULL default "	[SPACE-~]	OXERPSTATUS_TRACKID

4.32 Create/update product assignments to vendors

If a product is already assigned to a vendor, this assignment is overwritten.

#	Content/Example	Data type	Range	Field name
0.	Indicator char	char(1)	[D]	
1.	Product ID	char(32) NOT NULL default "	[0-9a-f.]	OXID
2.	Vendor ID	char(32) NOT NULL default "	[0-9a-f.]	OXVENDORID

4.33 Delete product assignments to vendors

#	Content/Example	Data type	Range	Field name
	Indicator char	char(2)	[XD]	
	Product assignment ID	char(32) NOT NULL default "	[0-9a-f.]	OXID

²⁷ Table: oxorderarticles

4.34 Create/update product assignments to manufacturers

If a product is already assigned to a manufacturer, this assignment is overwritten.

#	Content/Example	Data type	Range	Field name
3.	Indicator char	char(1)	[Q]	
4.	Product ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID
5.	Manufacturer ID	char(32) NOT NULL default ''	[0-9a-f.]	OXMANUFACTURER ID

4.35 Delete product assignments to manufacturers

#	Content/Example	Data type	Range	Field name
	Indicator char	char(2)	[XQ]	
	Product assignment ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID

4.36 Updating stock

#	Content/Example	Data type	Range	Field name
0.	Indicator char	char(1)	[E]	
1.	Product ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID
2.	Stock	Integer(11) NOT NULL default -1	-2147483648 to 2147483647	OXSTOCK

4.37 Create/update users

If a user with the given OXID already exists, the user is updated with the given values. Otherwise, a new user is created.

#	Content/Example	Data type	Range	Field name ²⁸
0.	Indicator char	char(1)	[U]	
1.	User ID	char(32) NOT NULL	[0-9a-f.]	OXID
2.	Active	tinyint(1) NOT NULL default 0	[0,1]	OXACTIVE
3.	Rights	char(32)	'user', 'malladmin' or subshop ID	OXRIGHTS
4.	Shop ID	int(11) NOT NULL default 1	[SPACE-~]	OXSHOPID

²⁸ Table: oxuser

#	Content/Example	Data type	Range	Field name ²⁸
5.	Login/e-mail address	varchar(255) NOT NULL	[SPACE-~]	OXUSERNAME
6.	Password	varchar(128) NOT NULL	[SPACE-~]	OXPASSWORD
7.	Salt (see also chapter 5.22)	char(128)	[SPACE-~]	OXPASSSALT
8.	Customer number	Integer(11)	0 to 2147483647	OXCUSTNR
9.	VAT ID	varchar(255) NOT NULL	[SPACE-~]	OXUSTID
10	VAT ID status	tinyint(1) NOT NULL default '0'	[0,1]	OXUSTIDSTATUS
11	Company	varchar(255) NOT NULL	[SPACE-~]	OXCOMPANY
12	First name	varchar(255) NOT NULL	[SPACE-~]	OXFNAME
13	Last name	varchar(255) NOT NULL	[SPACE-~]	OXLNAME
14	Street	varchar(255) NOT NULL	[SPACE-~]	OXSTREET
15	Street number	varchar(16) NOT NULL	[SPACE-~]	OXSTREETNR
16	Additional information	varchar(255) NOT NULL	[SPACE-~]	OXADDINFO
17	City	varchar(255) NOT NULL	[SPACE-~]	OXCITY
18	Country ID	varchar(32) NOT NULL	[SPACE-~]	OXCOUNTRYID
19	State ID	varchar(32) NOT NULL	[SPACE-~]	OXSTATEID
20	ZIP	varchar(16) NOT NULL	[SPACE-~]	OXZIP
21	Phone number	varchar(128) NOT NULL	[SPACE-~]	OXFON
22	Fax number	varchar(128) NOT NULL	[SPACE-~]	OXFAX
23	Salutation	varchar(128) NOT NULL	[SPACE-~]	OXSAL
24	Credit rating	integer(11)	0 to 2147483647	OXBONI
25	Date of creation	datetime	YYYY-MM-DD HH:MM:SS	OXCREATE
26	Date of registration	datetime	YYYY-MM-DD HH:MM:SS	OXREGISTER

#	Content/Example	Data type	Range	Field name ²⁸
27	Private phone number	varchar(64) NOT NULL	[SPACE-~]	OXPRIVFON
28	Cellular phone number	varchar(64) NOT NULL	[SPACE-~]	OXMOBFON
29	Date of birth	date	YYYY-MM-DD	OXBIRTHDATE
30	URL	varchar(255) NOT NULL	[SPACE-~]	OXURL
31	Automatic user group assignment	tinyint(1) NOT NULL	[0,1]	OXDISABLEAUTOGRP
32	LDAP-Key	varchar(128) NOT NULL	[SPACE-~]	OXLDAPKEY
33	Amount of invalid login attempts	integer(11)	0 to 2147483647	OXWRONGLOGINS
34	Key for „forgot password“ function	char(32)	[SPACE-~]	OXUPDATEKEY
35	Timestamp when key from field 33 expires	integer(11)	0 to 2147483647	OXUPDATEEXP
36	„Credit points“ from the users invitations	double NOT NULL default '0'	- 1.797693134862315 7 E+308 to - 2.225073858507201 4 E-308, 0, and 2.225073858507201 4 E-308 to 1.797693134862315 7 E+308	OXPOINTS

4.38 Delete users

#	Content/Example	Data type	Range	Field name
0.	Indicator char	char(2)	[XU]	
1.	User ID	char(32) NOT NULL	[0-9a-f.]	OXID

4.39 Create/Update countries

If a country with the given OXID already exists, the country is updated with the given values. Otherwise, a new country is created.

#	Content/Example	Data type	Range	Field name ²⁹
0.	Indicator char	char(1)	[N]	
1.	Country ID	char(32) NOT NULL	[0-9a-f.]	OXID
2.	Active	tinyint(1) NOT NULL default 0	[1,0]	OXACTIVE
3.	Name	char(128) NOT NULL	[SPACE-~]	OXTITLE
4.	ISO 2	char(2) NOT NULL	[A-Z]	OXISOALPHA2
5.	ISO 3	char(3) NOT NULL	[A-Z]	OXISOALPHA3
6.	ISO UN number	char(3) NOT NULL	[0-999]	OXUNNUM3
7.	Sorting	int(11) NOT NULL default 9999	0 to 2147483647	OXORDER
8.	Short description	char(255) NOT NULL	[SPACE-~]	OXSHORTDESC
9.	Description	char(255) NOT NULL	[SPACE-~]	OXLONGDESC
10.	Name in language 1	char(128) NOT NULL	[SPACE-~]	OXTITLE_1
11.	Name in language 2	char(128) NOT NULL	[SPACE-~]	OXTITLE_2
12.	Name in language 3	char(128) NOT NULL	[SPACE-~]	OXTITLE_3
13.	Short description in language 1	char(128) NOT NULL	[SPACE-~]	OXSHORTDESC_1
14.	Short description in language 2	char(128) NOT NULL	[SPACE-~]	OXSHORTDESC_2
15.	Short description in language 3	char(128) NOT NULL	[SPACE-~]	OXSHORTDESC_3
16.	Description in language 1	char(255) NOT NULL	[SPACE-~]	OXLONGDESC_1
17.	Description in language 2	char(255) NOT NULL	[SPACE-~]	OXLONGDESC_2
18.	Description in language 3	char(255)	[SPACE-~]	OXLONGDESC_3
19.	VAT Handling	tinyint(1) NOT NULL default 0	[0,1]	OXVATSTATUS

²⁹

Table: oxcountry

4.40 Delete Countries

#	Content/Example	Data type	Range	Field name
	Indicator char	char(2)	[XN]	
	Country ID	char(32) NOT NULL	[0-9a-f.]	OXID

4.41 Create/Update attributes

If a attribute with the given OXID already exists, the attribute is updated with the given values. Otherwise, a new attribute is created.

#	Content/Example	Data type	Range	Field name
0.	Indicator char	char(1)	[B]	
1.	Attribute ID	char(32) NOT NULL	[0-9a-f.]	OXID
2.	Shop-ID	int(11) NOT NULL default 1	1 - 64	OXSHOPID
3.	Inherited Shop IDs	See Chapter 4.1	See Chapter 4.1	OXINHERITEDSHOPIDS
4.	Name	char(128) NOT NULL	[SPACE~]	OXTITLE
5.	Name in language 1	char(128) NOT NULL	[SPACE~]	OXTITLE_1
6.	Name in language 2	char(128) NOT NULL	[SPACE~]	OXTITLE_2
7.	Name in language 3	char(128) NOT NULL	[SPACE~]	OXTITLE_3
8.	Sorting	integer(11)	0 to 2147483647	OXPOS

4.42 Delete Attributes

#	Content/Example	Data type	Range	Field name
0.	Indicator char	char(2)	[XB]	
1.	Attribute ID	char(32) NOT NULL	[0-9a-f.]	OXID

4.43 Create/Update Attribute assignments

If a attribute assignment with the given OXID already exists, the attribute assignment is updated with the given values. Otherwise, a new attribute assignment is created.

#	Content/Example	Data type	Range	Field name
0.	Indicator char	char(1)	[F]	
1.	attribute assignment ID	char(32) NOT NULL default "	[0-9a-f.]	OXID
2.	Product ID	char(32) NOT NULL default "	[0-9a-f.]	OXOBJECTID

3.	Attribute ID	char(32) NOT NULL default ''	[0-9a-f.]	OXATTRID
4.	Attribute value	char(255) NOT NULL default ''	[SPACE-~]	OXVALUE
5.	Sorting	int(11) NOT NULL default '0'	0 to 2147483647	OXPOS
6.	Attribute value in language 1	char(255) NOT NULL default ''	[SPACE-~]	OXVALUE_1
7.	Attribute value in language 2	char(255) NOT NULL default ''	[SPACE-~]	OXVALUE_2
8.	Attribute value in language 3	char(255) NOT NULL default ''	[SPACE-~]	OXVALUE_3

4.44 Delete Attribute assignment

#	Content/Example	Data type	Range	Field name
0.	Indicator char	char(2)	[XF]	
1.	Attribute Assignment ID	char(32) NOT NULL default ''	[0-9a-f.]	OXID

4.45 Create/Update contents

If a content with the given OXID already exists, the content is updated with the given values. Otherwise, a new content is created.

#	Content/Example	Data type	Range	Field name ³⁰
0.	Indicator char	char(1)	[G]	
1.	Content ID	char(32) NOT NULL	[0-9a-f.]	OXID
2.	Identification	char(32) NOT NULL	[0-9a-f.]	OXLOADID
3.	Shop-ID	int(11) NOT NULL default 1	1 - 64	OXSHOPID
20	Snippet is active	tinyint(1) NOT NULL default 1	[0,1]	OXSNIPPET
21	Content type (snippet, main menu, category, manually)	tinyint(1) NOT NULL default 0	[0,1,2,3]	OXTYPE
22	Active	tinyint(1) NOT NULL default 0	[0,1]	OXACTIVE
23	Active in language 1	tinyint(1) NOT NULL default 0	[0,1]	OXACTIVE_1
24	Active in language 2	tinyint(1) NOT NULL default 0	[0,1]	OXACTIVE_2
25	Active in language 3	tinyint(1) NOT NULL default 0	[0,1]	OXACTIVE_3
26	Name	varchar(255) NOT NULL	[SPACE-~]	OXTITLE

³⁰

Table: oxcountry

#	Content/Example	Data type	Range	Field name ³⁰
27	Name in language 1	varchar(255) NOT NULL	[SPACE-~]	OXTITLE_1
28	Name in language 2	varchar(255) NOT NULL	[SPACE-~]	OXTITLE_2
29	Name in language 3	varchar(255) NOT NULL	[SPACE-~]	OXTITLE_3
30	Sorting	varchar(32) NOT NULL	[SPACE-~]	OXPOSITION
31	Content	text NOT NULL	[SPACE-~]	OXCONTENT
32	Content in language 1	text NOT NULL	[SPACE-~]	OXCONTENT_1
33	Content in language 2	text NOT NULL	[SPACE-~]	OXCONTENT_2
34	Content in language 3	text NOT NULL	[SPACE-~]	OXCONTENT_3
35	Category ID	varchar(32)	[0-9a-f.]	OXCATID
36	Folder name	varchar(32) NOT NULL	[SPACE-~]	OXFOLDER
37	AGB version	char(32) NOT NULL	[SPACE-~]	OXTERMVERSION

4.46 Delete Contents

#	Content/Example	Data type	Range	Field name
	Indicator char	char(2)	[XG]	
	Content ID	char(32) NOT NULL	[0-9a-f.]	OXID

4.47 Create/Update contents

If a content with the given OXID already exists, the content is updated with the given values. Otherwise, a new content is created.

#	Content/Example	Data type	Range	Field name ³¹
4.	Indicator char	char(1)	[G]	
5.	Content ID	char(32) NOT NULL	[0-9a-f.]	OXID
6.	Identification	char(32) NOT NULL	[0-9a-f.]	OXLOADID
7.	Shop-ID	int(11) NOT NULL default 1	1 - 64	OXSHOPID
38	Snippet is active	tinyint(1) NOT NULL default 1	[0,1]	OXSNIPPET
39	Content type (snippet, main menu, category, manually)	tinyint(1) NOT NULL default 0	[0,1,2,3]	OXTYPE
40	Active	tinyint(1) NOT NULL default 0	[0,1]	OXACTIVE
41	Active in language 1	tinyint(1) NOT NULL default 0	[0,1]	OXACTIVE_1
42	Active in language 2	tinyint(1) NOT NULL default 0	[0,1]	OXACTIVE_2
43	Active in language 3	tinyint(1) NOT NULL default 0	[0,1]	OXACTIVE_3

³¹

Table: oxcountry

5 Export interface

All objects (products, users, categories and so on) can be exported. The structure of the exported file is the same as described above. For accessing import, a particular URL is called via browser. We strongly recommend to password-protect and SSL-protect (https://) the export so it's not openly accessible.

5.1 Changes in multishop handling

When exporting values with ERP version 2.14.0, make sure to pass the additional parameter 'blIncludeInheritedShopIds' => true into request to get the value of OXINHERITEDSHOPIDS, as otherwise it will not be retrieved.

Example:

```
$aParams = array(
    "sid"                => $sid,
    "id"                 => "oxtestcat1",
    'blIncludeInheritedShopIds' => true,
);
$sRes = $this->_doRequest("OXERPGetCategory", $aParams);
```

5.2 Sorting export data

In some exports you can set by which field and in which direction (ASC or DESC) the data is to be sorted. If no sorting parameter is given, the fields are sorted by OXID.

5.3 Login

Before CSV Data can be exported, you have to login first. This is done with OXERPLogin function:
<https://www.youreshop.com/modules/ERP/oxerpcsvexport.php?fnc=OXERPLogin&user=admin&pw=password&version=2.14.0&shopid=1&langid=0>

The following parameters are given:

Parameter	Description
fnc=	Name of function (here: OXERPLogin)
user=	Admin name (here: admin)
pw=	Password (here: password)
shopid=	Shop ID of Shop in which login is done (here: 1)
langid=	Language ID (here: 0)

The function returns a Session ID which has to be used in every other export function.

5.4 OXERPGetCategory

This function exports a category.

Parameters

Parameter	Description
fnc=OXERPGetCategory	Function name
sid=	Session ID returned by OXERPLLogin.
id=	ID of the category.
blIncludeInheritedShopIds= (optional)	Flag, which indicates whether to include inherited shop ID's in response. By default it is false.

Example

<https://www.youreshop.com/modules/ERP/oxerpcsvexport.php?fnc=OXERPGetCategory&sid=xxxxxxxxx&version=2.14.0&id=yyyyyyyyyyy>

5.5 OXERPGetCategories

This function exports several categories at once.

Parameters

Parameter	Description
fnc=OXERPGetCategories	Function name
sid=	Session ID returned by OXERPLLogin.
sSortFieldName= (optional)	The field the returned categories are sorted by.
sSortType= (optional)	ASC (ascending) or DESC (descending)
iStart= (optional)	Amount of objects which are skipped and not returned.
iCount= (optional)	Limits amount of returned objects.
blIncludeInheritedShopIds= (optional)	Flag, which indicates whether to include inherited shop ID's in response. By default it is false.

Example

<https://www.youreshop.com/modules/ERP/oxerpcsvexport.php?fnc=OXERPGetCategories&sid=xxxxx&version=2.14.0&sSortFieldName=OXID&sSortType=DESC&iStart=0&iCount=5>

5.6 OXERPGetArticle

This function exports a product.

Parameters

Parameter	Description
fnc=OXERPGetArticle	Function name
sid=	Session ID returned by OXERPLLogin.
id=	ID of the product.
blIncludeInheritedShopIds= (optional)	Flag, which indicates whether to include inherited shop ID's in response. By default it is false.

5.7 OXERPGetArticles

This function exports several products at once.

Parameters

Parameter	Description
fnc=OXERPGetArticles	Function name
sid=	Session ID returned by OXERPLLogin.
sSortFieldName= (optional)	The field the returned products are sorted by.
sSortType= (optional)	ASC (ascending) or DESC (descending)
iStart= (optional)	Amount of objects which are skipped and not returned.
iCount= (optional)	Limits amount of returned objects.
id= (optional)	Category ID. Only products assigned to this category are exported.
blIncludeInheritedShopIds= (optional)	Flag, which indicates whether to include inherited shop ID's in response. By default it is false.

5.8 OXERPGetFile

This function exports a file record.

Parameters

Parameter	Description
fnc=OXERPGetFile	Function name
sid=	Session ID returned by OXERPLLogin.
id=	ID of the file record.

5.9 OXERPGetFiles

This function exports several files at once.

Parameters

Parameter	Description
fnc=OXERPGetFiles	Function name
sid=	Session ID returned by OXERPLLogin.
sSortFieldName= (optional)	The field the returned files are sorted by.
sSortType= (optional)	ASC (ascending) or DESC (descending)
iStart= (optional)	Amount of objects which are skipped and not returned.
iCount= (optional)	Limits amount of returned objects.
id= (optional)	Product ID. Only files assigned to this product are exported.

5.10 OXERPGetArticleCategory

This function exports the categories a product is assigned to.

Parameters

Parameter	Description
fnc=OXERPGetArticleCategory	Function name
sid=	Session ID returned by OXERPLLogin.
id=	Product ID
sSortFieldName= (optional)	The field the returned categories are sorted by.
sSortType= (optional)	ASC (ascending) or DESC (descending)
iStart= (optional)	Amount of objects which are skipped and not returned.
iCount= (optional)	Limits amount of returned objects.
blIncludeInheritedShopIds= (optional)	Flag, which indicates whether to include inherited shop ID's in response. By default it is false.

5.11 OXERPGetArticleAction

This function exports the actions a product is assigned to.

Parameters

Parameter	Description
fnc=OXERPGetArticleAction	Function name
sid=	Session ID returned by OXERPLLogin.
id=	The ID of the product.
sSortFieldName= (optional)	The field the returned actions are sorted by.
sSortType= (optional)	ASC (ascending) or DESC (descending)
iStart= (optional)	Amount of objects which are skipped and not returned.
iCount= (optional)	Limits amount of returned objects.

5.12 OXERPGetArticleCrossselling

This function exports the products which are assigned as crosssellings to another product.

Parameters

Parameter	Description
fnc=OXERPGetArticleCrossselling	Function name
sid=	Session ID returned by OXERPLLogin.
id=	The ID of the product.
sSortFieldName= (optional)	The field the returned products are sorted by.
sSortType= (optional)	ASC (ascending) or DESC (descending)
iStart= (optional)	Amount of objects which are skipped and not returned.
iCount= (optional)	Limits amount of returned objects.

5.13 OXERPGetArticleScalePrice

This function exports scale prices assigned to a product.

Parameters

Parameter	Description
fnc=OXERPGetArticleScalePrice	Function name
sid=	Session ID returned by OXERPLogin.
id=	The ID of the product.
sSortFieldName= (optional)	The field the returned scale prices are sorted by.
sSortType= (optional)	ASC (ascending) or DESC (descending)
iStart= (optional)	Amount of objects which are skipped and not returned.
iCount= (optional)	Limits amount of returned objects.

5.14 OXERPGetArticleAccessoire

This function returns products which are assigned as accessories to another product.

Parameters

Parameter	Description
fnc=OXERPGetArticleAccessoire	Function name
sid=	Session ID returned by OXERPLogin.
id=	The ID of the product.
sSortFieldName= (optional)	The field the returned products are sorted by.
sSortType= (optional)	ASC (ascending) or DESC (descending)
iStart= (optional)	Amount of objects which are skipped and not returned.
iCount= (optional)	Limits amount of returned objects.

5.15 OXERPGetArticleStock

This function exports the stock of a product.

Parameters

Parameter	Description
fnc=OXERPGetArticleStock	Function name
sid=	Session ID returned by OXERPLLogin.
id=	The ID of the product.

5.16 OXERPGetOrder

This function exports an order.

Parameters

Parameter	Description
fnc=OXERPGetOrder	Function name
sid=	Session ID returned by OXERPLLogin.
id=	ID of the order.

5.17 OXERPGetOrderArticle

This function exports products of assigned to an order.

Parameters

Parameter	Description
fnc=OXERPGetOrderArticle	Function name
sid=	Session ID returned by OXERPLLogin.
id=	The OXID of the order
sSortFieldName= (optional)	The field the returned products are sorted by.
sSortType= (optional)	ASC (ascending) or DESC (descending)
iStart= (optional)	Amount of objects which are skipped and not returned.
iCount= (optional)	Limits amount of returned objects.

5.18 OXERPGetOrderFiles

This function exports files assigned to an order.

Parameters

Parameter	Description
fnc=OXERPGetOrderFiles	Function name
sid=	Session ID returned by OXERPLLogin.
id=	The OXID of the order
sSortFieldName= (optional)	The field the returned files are sorted by.
sSortType= (optional)	ASC (ascending) or DESC (descending)
iStart= (optional)	Amount of objects which are skipped and not returned.
iCount= (optional)	Limits amount of returned objects.

5.19 OXERPGetVendor

This function exports a vendor.

Parameters

Parameter	Description
fnc=OXERPGetVendor	Function name
sid=	Session ID returned by OXERPLLogin.
id=	ID of the vendor.
blIncludeInheritedShopIds= (optional)	Flag, which indicates whether to include inherited shop ID's in response. By default it is false.

5.20 OXERPGetArticleVendor

This function exports the vendor of a product.

Parameters

Parameter	Description
fnc=OXERPGetArticleVendor	Function name
sid=	Session ID returned by OXERPLLogin.
id=	ID of the product.
blIncludeInheritedShopIds= (optional)	Flag, which indicates whether to include inherited shop ID's in response. By default it is false.

5.21 OXERPGetVendorArticles

This function returns all products assigned to a vendor.

Parameters

Parameter	Description
fnc=OXERPGetVendorArticles	Function name
sid=	Session ID returned by OXERPLLogin.
id=	The ID of the vendor.
sSortFieldName= (optional)	The field the returned products are sorted by.
sSortType= (optional)	ASC (ascending) or DESC (descending)
iStart= (optional)	Amount of objects which are skipped and not returned.
iCount= (optional)	Limits amount of returned objects.
blIncludeInheritedShopIds= (optional)	Flag, which indicates whether to include inherited shop ID's in response. By default it is false.

5.22 OXERPGetManufacturer

This function exports a manufacturer.

Parameters

Parameter	Description
fnc=OXERPGetManufacturer	Function name
sid=	Session ID returned by OXERPLLogin.
id=	ID of the manufacturer.
blIncludeInheritedShopIds= (optional)	Flag, which indicates whether to include inherited shop ID's in response. By default it is false.

5.23 OXERPGetArticleManufacturer

This function exports the manufacturer of a product.

Parameters

Parameter	Description
fnc=OXERPGetArticleManufacturer	Function name
sid=	Session ID returned by OXERPLLogin.
id=	ID of the product.
blIncludeInheritedShopIds= (optional)	Flag, which indicates whether to include inherited shop ID's in response. By default it is false.

5.24 OXERPGetManufacturerArticles

This function returns all products assigned to a manufacturer.

Parameters

Parameter	Description
fnc=OXERPGetManufacturerArticles	Function name
sid=	Session ID returned by OXERPLLogin.
id=	The ID of the manufacturer.
sSortFieldName= (optional)	The field the returned products are sorted by.
sSortType= (optional)	ASC (ascending) or DESC (descending)
iStart= (optional)	Amount of objects which are skipped and not returned.
iCount= (optional)	Limits amount of returned objects.
blIncludeInheritedShopIds= (optional)	Flag, which indicates whether to include inherited shop ID's in response. By default it is false.

5.25 OXERPGetUser

This function exports a user.

Since OXID eShop version 4.9.0/5.2.0, passwords are SHA-2 encrypted with a salt.

These passwords cannot be decrypted. They cannot be imported in older eShop versions. The salt is always a HEX value.

Parameters

Parameter	Description
fnc=OXERPGetUser	Function name
sid=	Session ID returned by OXERPLLogin.
id=	The ID of the user.

5.26 OXERPGetUsers

This function exports several users at once.

Parameters

Parameter	Description
fnc=OXERPGetUsers	Function name
sid=	Session ID returned by OXERPLogin.
sSortFieldName= (optional)	The field the returned users are sorted by.
sSortType= (optional)	ASC (ascending) or DESC (descending)
iStart= (optional)	Amount of objects which are skipped and not returned.
iCount= (optional)	Limits amount of returned objects.
timestamp=yyyy-mm-dd hh:mm:ss (optional)	Only users created after this point of time are exported.

5.27 OXERPGetOrders

This function exports several orders at once.

Parameters

Parameter	Description
fnc=OXERPGetOrders	Function name
sid=	Session ID returned by OXERPLLogin.
sSortFieldName= (optional)	The field the returned orders are sorted by.
sSortType= (optional)	ASC (ascending) or DESC (descending)
iStart= (optional)	Amount of objects which are skipped and not returned.
iCount= (optional)	Limits amount of returned objects.
timestamp=yyyy-mm-dd hh:mm:ss (optional)	Alle Bestellungen werden exportiert, die an und nach diesem Zeitpunkt angelegt wurden.

5.28 OXERPGetVendors

This function exports several vendors

Parameters

Parameter	Description
fnc=OXERPGetVendors	Function name
sid=	Session ID returned by OXERPLLogin.
sSortFieldName= (optional)	The field the returned vendors are sorted by.
sSortType= (optional)	ASC (ascending) or DESC (descending)
iStart= (optional)	Amount of objects which are skipped and not returned.
iCount= (optional)	Limits amount of returned objects.
blIncludeInheritedShopIds= (optional)	Flag, which indicates whether to include inherited shop ID's in response. By default it is false.

5.29 OXERPGetManufacturers

This function exports several manufacturers

Parameters

Parameter	Description
fnc=OXERPGetManufacturers	Function name

Parameter	Description
sid=	Session ID returned by OXERPLLogin.
sSortFieldName= (optional)	The field the returned manufacturers are sorted by.
sSortType= (optional)	ASC (ascending) or DESC (descending)
iStart= (optional)	Amount of objects which are skipped and not returned.
iCount= (optional)	Limits amount of returned objects.
blIncludeInheritedShopIds= (optional)	Flag, which indicates whether to include inherited shop ID's in response. By default it is false.

5.30 OXERPCallPlugin

OXERPCallPlugin invokes an external plugin. Sample call:

`http://shopurl/modules/erp/oxerpcsvexport.php?sid=MYSID&fnc=OXERPCallPlugin&sPluginName=examplePlugin&aRequestData[var1]=val1&aRequestData[var2]=val2`

Parameters

Parameter	Description
fnc=OXERPCallPlugin	Function name
sid=	Session ID returned by OXERPLLogin.
sPluginName=	The name of a plugin to be called.
aRequestData=	Data passed to the plugin

5.31 OXERPGetAttributes

This function exports several attributes which are defined in sIn.

Parameters

Parameter	Description
fnc=OXERPGetAttributes	Function name
sid=	Session ID returned by OXERPLLogin.
sIn=	List of attributnames
sSortFieldName= (optional)	The field by which the returned attributes are sorted
sSortType= (optional)	ASC (ascending) or DESC (descending)
iStart= (optional)	Amount of objects which are skipped and not returned.
iCount= (optional)	Limits amount of returned objects.

Parameter	Description
blIncludeInheritedShopIds= (optional)	Flag, which indicates whether to include inherited shop ID's in response. By default it is false.

5.32 OXERPGetArticleAttributes

This function returns product assignments to attribute.

Parameters

Parameter	Description
fnc=OXERPGetArticleAttributes	Function name
sid=	Session ID returned by OXERPLLogin.
sArticleID=	ID of the product. May be left empty.
sAttributeID=	ID of the attribute. May be left empty.
sSortFieldName= (optional)	The field the returned data is sorted by.
sSortType= (optional)	ASC (ascending) or DESC (descending)
iStart= (optional)	Amount of objects which are skipped and not returned.
iCount= (optional)	Limits amount of returned objects.

5.33 OXERPGetContent

This function exports a content.

Parameters

Parameter	Description
fnc=OXERPGetContent	Function name
sid=	Session ID returned by OXERPLLogin.
id=	ID of the content.

6 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.

6.1 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',
'OXACTIVE'       => 'OXACTIVE',
```

This definition is extended with *articles_amount*:

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

6.2 Extending ERP with plugins

With plugins, you can extend the functionality of the CSV interface. Plugins are called using the function *OXERPCallPlugin* (See 5.29).

6.3 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 default 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). In ERP v2.14.2 or higher, plugins could be located in custom modules. Plugin class still must extend the class *oxErpPluginBase* and implement *invoke()* function, and also be registered in metadata of a custom module.

6.4 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);
}
}

```

6.5 Data flow of the sample plugin

Assuming data of a csvexport request is:

```
http://...&fnc=OXERPCallPlugin&sPluginName=examplePlugin&aRequestData[myvarname1]=value1&aRequestData[myvarname2]=value2
```

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

```

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

```

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.

CSV client would get duplicated input in csv format:

```

myvarname1,myvarname2
value1,value2
value1,value2

```

The first line of the output file is a header line, containing variable names (array key values of the first data record, returned by the plugin, if resulting lines column count is equal). If column count in data is not equal, header line is skipped.

6.6 oxGetActionArticlesPlugin

This function returns article id's of an action.

Parameters

Parameter	Description
fnc=OXERPCallPlugin	Function name
sPluginName=	Plugin name (oxGetActionArticlesPlugin)
sid=	Session ID returned by OXERPLLogin.
aRequestData[sActionID]=	ID of an action.

6.7 oxGetArticleVariantsPlugin

This function returns variant id's of an article.

Parameters

Parameter	Description
fnc=OXERPCallPlugin	Function name
sPluginName=	Plugin name (oxGetArticleVariantsPlugin)
sid=	Session ID returned by OXERPLLogin.
aRequestData[sArticleID]=	ID of an article

6.8 oxGetConfigParametersPlugin

This function returns configuration parameters:

- user config parameters:
 - loaded from datenbank:
 - facebook_apID (Facebook Application ID: 'sFbAppId');

- facebook_sKey (Application Secret Key: 'sFbSecretKey');
- rating (Ratings (Yes, No): 'bl_perfLoadReviews');
- review (Review (Yes/no): 'bl_perfLoadReviews');
- parentArticleBuyable (Buyable article (Yes/no): 'blVariantParentBuyable').
- loaded from config.inc.php file:
 - eFireID (eFireID: 'sMobAppFireID');
 - paypalmerchantID (PayPal Merchant ID: 'sMobAppPaypalMerchantID');
 - pictures:
 - splash (SplashScreen URL: 'sMobAppSplash');
 - header (Header logo path: 'sMobAppHeader');
 - default (Name of default article picture: 'sMobAppDefault').
 - color (Color values: 'aMobAppColor');
 - facebook (Facebook Yes/No: 'sMobAppFacebookActive');
 - twitter (Twitter Yes/No: 'sMobAppTwitterActive');
 - twitterConsumerKey (Application ID: 'sMobAppTwitterConsumerKey');
 - twitterConsumerSecret (Application Secret Key: 'sMobAppTwitterConsumerSecret');
 - email (Email (Yes, No): 'sMobAppEmail');
 - active shop parameters:
 - currency (Currency Sign);
 - currencyrate (Exchange rate);
 - lang (Supportet Languages).

Parameters

Parameter	Description
fnc=OXERPCallPlugin	Function name
sPluginName=	Plugin name (oxGetConfigParametersPlugin)
sid=	Session ID returned by OXERPLugin

6.9 oxSearchPlugin

This function searches for products and returns article fields:

- oxid (article ID);
- oxtitle (title of an article);
- oxprice (price of an article);
- oxrating (rating);
- oxmanufacturerid (manufacturer ID);
- oxunitname (unit);
- oxunitquantity (quantity);
- oxtprice (RRP);

Parameters

Parameter	Description
fnc=OXERPCallPlugin	Function name
sPluginName=	Plugin name (oxSearchPlugin)

sid=	Session ID returned by OXERPLLogin.
aRequestData[sSearchParam]=	ID of an article
aRequestData[iStart]= (optional)	Amount of objects which are skipped and not returned.
aRequestData[iCount]= (optional)	Limits amount of returned objects.

7 Expandability of CSV interface

You can also extend the CSV interface with own functions, but we advise you to use plugins (See chapter 6.2) for keeping the ERP easily updatable.

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

7.1 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',
            '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.

7.2 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);
}
```

7.3 Import definition in `oxerpcsv.php`

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

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

In `oxerpcsv.php` the indicator chars are defined in `$_aObjects` array. For attribute import, we choose 'B' as indicator char and add it to `$_aObjects` array:

```
'B' => 'attribute',
```

Take care that each indicator char is used only once.

7.4 Using new `OXERPSetAttribute` function

If an attribute with the given OXID already exists, the attribute is updated with the given values. Otherwise, a new attribute is created.

#	Content/Example	Data type	Range	Field name ³²
---	-----------------	-----------	-------	--------------------------

³² Table: `oxattribute`

#	Content/Example	Data type	Range	Field name ³²
0.	Indicator char	char(1)	[B]	
1.	Attribute ID	char(32) NOT NULL	[0-9a-f.]	OXID
2.	Shop-ID	int(11) NOT NULL default 1	1 - 64	OXSHOPID
3.	Name	char(128) NOT NULL	[SPACE-~]	OXTITLE
4.	Name in language 1	char(128) NOT NULL	[SPACE-~]	OXTITLE_1
5.	Name in language 2	char(128) NOT NULL	[SPACE-~]	OXTITLE_2
6.	Name in language 3	char(128) NOT NULL	[SPACE-~]	OXTITLE_3
7.	Sorting	integer(11)	0 to 2147483647	OXPOS

7.5 Registering function in oxerpcsvexport.php

In oxerpcsvexport.php, all get functions are defined. `OXERPSetAttribute` is not an export function, nothing has to be changed.

If a get function `OXERPGetAttribute` was implemented, in the field list at the beginning of the file the following line had to be added:

```
"OXERPGetAttribute" => "sAttributeID",
```