

ELO Suite for SAP

ArchiveLink® (SAP

NetWeaver® & SAP S/

4HANA®)

ELO Connectivity Pack for SAP® ERP -
Indexdownload

Table of contents

ELO Connectivity Pack for SAP® ERP – Index	download	3
Getting started		3
Installation		5
Functionality		7
ELO configuration		19
Process		29
Additional functions		31
Required permissions		48

ELO Connectivity Pack for SAP® ERP – Indexdownload

Getting started

This document describes how to install and configure ELO Connectivity Pack – Indexdownload.

Security and licensing requirements

Depending on the configuration, you can use ELO Connectivity Pack – Indexdownload to store sensitive data in ELO. A secure permissions concept is therefore crucial. Export and use of SAP data in external systems, especially by users without an SAP account, can affect the SAP license of the connected system. If in doubt, consult SAP and/or the responsible IT systems specialist. ELO is not liable for any license violations.

Basics

Using ELO Connectivity Pack – Indexdownload requires customization of SAP ArchiveLink on the SAP system connected to ELO. Document types, document classes, a content repository, and link entries have to be created for this purpose. An ELO ArchiveLink interface or ELO Smart Link interface with configured content repositories is also required.

ELO Connectivity Pack – Indexdownload must be configured in ELO Smart Link or ELO ArchiveLink and the transport has to be imported into the corresponding SAP system. The configured SAP user must have permission to run this function module (see the section SAP function module).

To use the entire ELO Connectivity Pack for SAP® ERP, your computer must also meet the following system requirements and release statuses of ELO and the SAP system:

- ELOprofessional/ELOenterprise version 10.02.000 and higher
- ELO Business Solutions Common 1.05.002 and higher
- The following SAP versions or higher:
 - SAP_BASIS 702 SAPKB70213
 - SAP_BASIS 730 SAPKB73009
 - SAP_BASIS 731 SAPKB73106

The following requirements have to be met to use the ELO Integration Client within the SAP system:

- ELOprofessional or ELOenterprise version 12 and higher
- Depending on the version, the following modules are required:
 - ELO 20:
 - ELOWf 20.05.000
 - ELO REST Service 20.02.000
 - ELO 12:
 -

ELOWf 12.06.000

■ ELO REST API 12.04.000

- ELO Business Solutions Common version 1.07.000 or higher

The ELO license must also include the ELO Integration Client.

Installation

A license for ELO Connectivity Pack for SAP ERP includes two functions: *Indexdownload* and *Datatransfer*. ELO Connectivity Pack – Indexdownload contains two packages.

- An SAP transport that has to be imported to the SAP target system.
- An ELO Business Solutions package customized for SAP.

You will find the installation files on the ELO SupportWeb under:

Integration > SAP® ERP > ELO Suite for SAP ArchiveLink® > ELO Connectivity Pack for SAP® ERP > Downloads

SAP components

Inform your customer or the SAP partner about the SAP transport to be installed and provide it to them.

The current SAP transport is available on the ELO SupportWeb.

Information

Inform your customers or SAP partner of the following:

- The standard SAP transport routes must be used.
- All developments and functions within the transport in ELO were developed in a separate SAP namespace, called */ELO/*.

You may have to enable the *Ignore Invalid Component Version* option when importing the SAP transport.

It is also possible that you will encounter warnings when importing the transport due to different release statuses (return code 8), which can be ignored.

Once an SAP transport has successfully been transported to the SAP systems, you can provide your customer/the SAP partner with the technical description for the */ELO/BO_FIELDS* transaction.

You will find it in the section '*/ELO/BO_FIELDS*' transaction.

Common package

First, install *Business Solution Common 1.05.002* package or higher if it has not already been installed on your ELO system. This package provides the class framework, namespaces, and architectural concepts for services, actions, and function modules.

1. To start installation, log on to the ELO Java Client as Administrator. You can install the individual packages from the ELO SupportWeb with a single click.

Alternative: You can download the package containing the .eloinst files on the ELO SupportWeb. At this point, ELO Connectivity Pack – Indexdownload only requires the following two files:

- 00_sol.common_1.xx.xxx.eloinst
- Custom_sol.common_1.xx.xxx.eloinst

2. Next, install the current ELO Connectivity Pack for SAP® ERP – Indexdownload package. You will now find the configuration for ELO Connectivity Pack – Indexdownload in the *Business Solutions Custom* folder in ELO.

3. You can now proceed with the chapter ELO configuration.

- erp.sap.indexdownload_1.XX.XXX.eloinst
- custom_erp.sap.indexdownload_1.XX.XXX.eloinst

Installation information

When installing ELO Connectivity Pack – Indexdownload, the web app installation may not finish completely. You can tell that this is the case if there is no interface when you open the configuration. To fix this, switch to the ELO Application Server and open the ELOwf status report. You will see the *App Manager* entry in the menu bar on the right.

In the ELO App Manager, the *App status* indicates whether the web app installation for Indexdownload was successful. If this is the case, the entry *Installed* is shown at the location. If it is set to *Archived*, you will have to enable/install, and refresh the web app.

Functionality

ELO Connectivity Pack – Indexdownload is based on an SAP function module that can call data for SAP business objects from the SAP system generically. ELO Connectivity Pack – Indexdownload uses new metadata forms and fields created in ELO with SAP attribute names. For more information on the SAP attribute names, refer to the section '[/ELO/BO_FIELDS](#)' transaction.

This data can vary depending on the business object and possibly the release versions. Table data (e.g. on order items) is also offered. When used in a field, it is listed separated by pilcrow characters (¶).

Metadata forms in ELO

ELO Connectivity Pack – Indexdownload contains three new metadata forms specifically for an SAP business object: an SAP folder form, the new SAPDATA form, and a template form that ELO Business Partners can use as a basis for customizing metadata forms. By default, these metadata forms are defined for the *Everyone* permissions group. The ELO Business Partner has to make any necessary changes based on the existing permissions concept.

SAP-PurchaseOrder - business object BUS2012

The *SAP-PurchaseOrder* metadata form was defined to index documents for the *PO (BU2012)* business object. This form has three tabs with the following fields:

SAP-PurchaseOrder

Name	SAP-PurchaseOrder	ID	38
Translation variable	erp.sap.mask.saporder.maskname	GUID	(AEC47024-2FAD-3FF1-6360-B142B81207DB)
Last change	22.05.2020 12:24		Save data as a table ?

[Usage](#)[Fields](#)

Header	ArchiveLink	Objectdata
Short name	Short name	
Date	Date	Version
Filing date	Filing date	Editor
Description	SAP_DESCR (Description, L5)	
Document Type Clearname	SAP_ARO_TXT (Document Type Clearname, L2)	
Creditor name	VENDOR_NAME (Creditor name, L3)	
Lieferantennummer (Kreditor)	VENDOR_VENDORNO (Lieferantennummer (Kreditor), L4)	
Order Creation Date	POHEADER_	
Creator	POHEADER_CREATED_BY (Creator, L20)	
Company Code	POHEADER_COMP_CODE (Company Code, L21)	
Purchase Organisation	POHEADER_PURCH_ORG (Purchase Organisation, L22)	
Purchase Group	POHEADER_PUR_GROUP (Purchase Group, L23)	

Header	ArchiveLink X	Objectdata X
SAP Archivepath	SAPPATH (SAP Archivepath, L9)	
SAP Creation Date	SAPCRDATE (SAP Creation Date, L10)	
SAP Alterationdate	SAPMODATE (SAP Alterationdate, L11)	
SAP Version	SAPAPVER (SAP Version, L12)	
SAP ArchiveLink Rights	SAPPRCH (SAP ArchiveLink Rights, L13)	
SAP Filename	SAPFILEN (SAP Filename, L14)	
SAP Filesize	SAPELODATA (SAP Filesize, L15)	
SAP Content-Type	SAPCONT (SAP Content-Type, L16)	
SAP Barcode	SAPBARC (SAP Barcode, L17)	
ArchiveLink Protocol Version	SAPALVER (ArchiveLink Protocol Version, L18)	

Header	ArchiveLink	Objectdata
Object ID		OBJECT_ID (Object ID, L6)
Object Type		SAP_OBJECT (Object Type, L7)
Dokumentart		AR_OBJECT (Dokumentart, L8)
SAP Object Clearname		SAP_OBJ_TXT (SAP Object Clearname, L1)
Creator		SAP_CREATOR (Creator, L24)

SAP-BillingDocument - business object VBRK

The *SAP-BillingDocument* metadata form was defined to index documents for the *billing document* (VBRK) business object. This form has four tabs with the following fields:

SAP-BillingDocument			
Name	SAP-BillingDocument	ID	36
Translation variable	erp.sap.mask.billingdocument.maskname	GUID	(2AB6CE32-D8C9-8F1A-8D5F-9CD9A8377667)
Last change	22.05.2020 12:24	Save data as a table	?
Usage			
Fields			
Header	Creditor	ArchiveLink	Objectdata
Short name	Short name		
Date	Date	Version	Version
Filing date	Filing date	Editor	Editor
Description	SAP_DESCR (Description, L17)		
Document Type Clearname	SAP_ARO_TXT (Document Type Clearname, L15)		
Faktura	BILLINGDOCUMENT (Faktura, L18)		
Billing Kind	BILLINGDOCUMENTDETAIL_BILL_TYPE (Billing Kind, L19)		
Billing Type	BILLINGDOCUMENTDETAIL_BILLCATEG (Billing Type, L20)		
Billing Document	BILLINGDOCUMENTDETAIL_BILLINGDOC (Billing Document, L21)		
Currency	BILLINGDOCUMENTDETAIL_CURRENCY (Currency, L22)		
Netvalue	BILLINGDOCUMENTDETAIL_NET_VALUE (Netvalue, L23)		
Taxvalue	BILLINGDOCUMENTDETAIL_TAX_VALUE (Taxvalue, L24)		
Debtor No.	SOLDTOPARTY_CUSTOMERNO (Debtor No., L25)		
Debtor Name	SOLDTOPARTY_NAME (Debtor Name, L26)		
Street	SOLDTOPARTY_STREET (Street, L27)		
City	SOLDTOPARTY_CITY (City, L28)		
Postalcode	SOLDTOPARTY_POSTALCODE (Postalcode, L29)		
Country	SOLDTOPARTY_COUNTRY (Country, L30)		
Header	Creditor	ArchiveLink	Objectdata
Companycode	COMPANYCODE_COMPANYCODEID (Companycode, L32)		
Name	COMPANYCODE_DESCRIPTION (Name, L33)		
City	COMPANYCODE_CITY (City, L34)		
Land	COMPANYCODE_COUNTRYKEY (Land, L35)		

Header	Creditor	ArchiveLink	Objectdata
SAP Archivepath		SAPPATH (SAP Archivepath, L1)	
SAP Creation Date		SAPCRDATE (SAP Creation Date, L2)	
SAP Alterationdate		SAPMODATE (SAP Alterationdate, L3)	
SAP Version		SAPAPVER (SAP Version, L4)	
SAP ArchiveLink Rights		SAPPRCH (SAP ArchiveLink Rights, L5)	
SAP Filename		SAPFILEN (SAP Filename, L6)	
SAP Filesize		SAPELODATA (SAP Filesize, L7)	
SAP Content-Type		SAPCONT (SAP Content-Type, L8)	
SAP Barcode		SAPBARC (SAP Barcode, L9)	
ArchiveLink Protocol Version		SAPALVER (ArchiveLink Protocol Version, L10)	

Header	Creditor	ArchiveLink	Objectdata
Object ID		OBJECT_ID (Object ID, L11)	
Object Type		SAP_OBJECT (Object Type, L12)	
Dokumentart		AR_OBJECT (Dokumentart, L14)	
SAP Object Clearname		SAP_OBJ_TXT (SAP Object Clearname, L13)	
Creator		SAP_CREATOR (Creator, L16)	

SAP-Vendor (supplier) - business object LFA1

The *SAP-Vendor* metadata form was defined to index documents for the *vendor (LFA1)* business object. This form has three tabs with the following fields:

SAP-Vendor																																																							
Name	SAP-Vendor	ID	34																																																				
Translation variable	erp.sap.mask.vendor.maskname	GUID	(63F5B4E2-0C02-AB28-DE6E-2E3971F7D28D)																																																				
Last change	22.05.2020 12:24	Save data as a table ?																																																					
➤ Usage																																																							
▼ Fields																																																							
<table border="1"> <thead> <tr> <th>General</th> <th>ArchiveLink</th> <th>Objectdata</th> <th>X</th> </tr> </thead> <tbody> <tr> <td>Short name</td> <td colspan="3">Short name</td> </tr> <tr> <td>Date</td> <td>Date</td> <td>Version</td> <td>Version</td> </tr> <tr> <td>Filing date</td> <td>Filing date</td> <td>Editor</td> <td>Editor</td> </tr> <tr> <td>Description</td> <td colspan="3">SAP_DESCR (Description, L17)</td> </tr> <tr> <td>Document Type Clearname</td> <td colspan="3">SAP_ARO_TXT (Document Type Clearname, L15)</td> </tr> <tr> <td>Vendor No.</td> <td colspan="3">GENERALDETAIL_VENDOR (Vendor No., L18)</td> </tr> <tr> <td>Name</td> <td colspan="3">GENERALDETAIL_NAME (Name, L19)</td> </tr> <tr> <td>Street</td> <td colspan="3">GENERALDETAIL_STREET (Street, L20)</td> </tr> <tr> <td>City</td> <td colspan="3">GENERALDETAIL_CITY (City, L21)</td> </tr> <tr> <td>Postalcode</td> <td colspan="3">GENERALDETAIL_POSTL_CODE (Postalcode, L22)</td> </tr> <tr> <td>Country</td> <td colspan="3">GENERALDETAIL_COUNTRY (Country, L23)</td> </tr> <tr> <td>Bank Detail</td> <td colspan="3">BANKDETAIL (Bank Detail, L24)</td> </tr> </tbody> </table>				General	ArchiveLink	Objectdata	X	Short name	Short name			Date	Date	Version	Version	Filing date	Filing date	Editor	Editor	Description	SAP_DESCR (Description, L17)			Document Type Clearname	SAP_ARO_TXT (Document Type Clearname, L15)			Vendor No.	GENERALDETAIL_VENDOR (Vendor No., L18)			Name	GENERALDETAIL_NAME (Name, L19)			Street	GENERALDETAIL_STREET (Street, L20)			City	GENERALDETAIL_CITY (City, L21)			Postalcode	GENERALDETAIL_POSTL_CODE (Postalcode, L22)			Country	GENERALDETAIL_COUNTRY (Country, L23)			Bank Detail	BANKDETAIL (Bank Detail, L24)		
General	ArchiveLink	Objectdata	X																																																				
Short name	Short name																																																						
Date	Date	Version	Version																																																				
Filing date	Filing date	Editor	Editor																																																				
Description	SAP_DESCR (Description, L17)																																																						
Document Type Clearname	SAP_ARO_TXT (Document Type Clearname, L15)																																																						
Vendor No.	GENERALDETAIL_VENDOR (Vendor No., L18)																																																						
Name	GENERALDETAIL_NAME (Name, L19)																																																						
Street	GENERALDETAIL_STREET (Street, L20)																																																						
City	GENERALDETAIL_CITY (City, L21)																																																						
Postalcode	GENERALDETAIL_POSTL_CODE (Postalcode, L22)																																																						
Country	GENERALDETAIL_COUNTRY (Country, L23)																																																						
Bank Detail	BANKDETAIL (Bank Detail, L24)																																																						
<table border="1"> <thead> <tr> <th>General</th> <th>ArchiveLink</th> <th>Objectdata</th> <th>X</th> </tr> </thead> <tbody> <tr> <td>SAP Archivepath</td> <td colspan="3">SAPPATH (SAP Archivepath, L1)</td> </tr> <tr> <td>SAP Creation Date</td> <td colspan="3">SAPCRDATE (SAP Creation Date, L2)</td> </tr> <tr> <td>SAP Alterationdate</td> <td colspan="3">SAPMODATE (SAP Alterationdate, L3)</td> </tr> <tr> <td>SAP Version</td> <td colspan="3">SAPAPVER (SAP Version, L4)</td> </tr> <tr> <td>SAP ArchiveLink Rights</td> <td colspan="3">SAPPRCH (SAP ArchiveLink Rights, L5)</td> </tr> <tr> <td>SAP Filename</td> <td colspan="3">SAPFILEN (SAP Filename, L6)</td> </tr> <tr> <td>SAP Filesize</td> <td colspan="3">SAPELODATA (SAP Filesize, L7)</td> </tr> <tr> <td>SAP Content-Type</td> <td colspan="3">SAPCONT (SAP Content-Type, L8)</td> </tr> <tr> <td>SAP Barcode</td> <td colspan="3">SAPBARC (SAP Barcode, L9)</td> </tr> <tr> <td>ArchiveLink Protocol Version</td> <td colspan="3">SAPALVER (ArchiveLink Protocol Version, L10)</td> </tr> </tbody> </table>				General	ArchiveLink	Objectdata	X	SAP Archivepath	SAPPATH (SAP Archivepath, L1)			SAP Creation Date	SAPCRDATE (SAP Creation Date, L2)			SAP Alterationdate	SAPMODATE (SAP Alterationdate, L3)			SAP Version	SAPAPVER (SAP Version, L4)			SAP ArchiveLink Rights	SAPPRCH (SAP ArchiveLink Rights, L5)			SAP Filename	SAPFILEN (SAP Filename, L6)			SAP Filesize	SAPELODATA (SAP Filesize, L7)			SAP Content-Type	SAPCONT (SAP Content-Type, L8)			SAP Barcode	SAPBARC (SAP Barcode, L9)			ArchiveLink Protocol Version	SAPALVER (ArchiveLink Protocol Version, L10)										
General	ArchiveLink	Objectdata	X																																																				
SAP Archivepath	SAPPATH (SAP Archivepath, L1)																																																						
SAP Creation Date	SAPCRDATE (SAP Creation Date, L2)																																																						
SAP Alterationdate	SAPMODATE (SAP Alterationdate, L3)																																																						
SAP Version	SAPAPVER (SAP Version, L4)																																																						
SAP ArchiveLink Rights	SAPPRCH (SAP ArchiveLink Rights, L5)																																																						
SAP Filename	SAPFILEN (SAP Filename, L6)																																																						
SAP Filesize	SAPELODATA (SAP Filesize, L7)																																																						
SAP Content-Type	SAPCONT (SAP Content-Type, L8)																																																						
SAP Barcode	SAPBARC (SAP Barcode, L9)																																																						
ArchiveLink Protocol Version	SAPALVER (ArchiveLink Protocol Version, L10)																																																						

Basis	ArchiveLink X	Objectdata X
Object ID		OBJECT_ID (Object ID, L11)
Object Type		SAP_OBJECT (Object Type, L12)
Dokumentart		AR_OBJECT (Dokumentart, L14)
SAP Object Clearname		SAP_OBJ_TXT (SAP Object Clearname, L13)
Creator		SAP_CREATOR (Creator, L16)

SAP-Templatemask metadata form

The template metadata form *SAP-Templatemask* was created to add metadata to SAP business objects. ELO Business Partners can use this form as a basis for their customizations. This template form contains the mandatory fields required to download an index, divided into three tabs:

SAP-Templatemask

Name	SAP-Templatemask	ID	37
Translation variable	erp.sap.mask.saptemplate.maskname	GUID	(E7786D68-520C-68AC-91F6-168A47900415)
Last change	22.05.2020 12:24	Save data as a table (i)	
> Usage ▼ Fields			

Basis	ArchiveLink X	Objectdata X	
Short name	Short name		
Date	Date	Version	Version
Filing date	Filing date	Editor	Editor
Description	SAP_DESCR (Description, L17)		
Document Type Clearname	SAP_ARO_TXT (Document Type Clearname, L15)		

Basis	ArchiveLink	Objectdata
SAP Archivepath		SAPPATH (SAP Archivepath, L1)
SAP Creation Date		SAPCRDATE (SAP Creation Date, L2)
SAP Alterationdate		SAPMODATE (SAP Alterationdate, L3)
SAP Version		SAPAPVER (SAP Version, L4)
SAP ArchiveLink Rights		SAPPRCH (SAP ArchiveLink Rights, L5)
SAP Filename		SAPFILEN (SAP Filename, L6)
SAP Filesize		SAPELODATA (SAP Filesize, L7)
SAP Content-Type		SAPCONT (SAP Content-Type, L8)
SAP Barcode		SAPBARC (SAP Barcode, L9)
ArchiveLink Protocol Version		SAPALVER (ArchiveLink Protocol Version, L10)

General	ArchiveLink	Objectdata
Object ID		OBJECT_ID (Object ID, L11)
Object Type		SAP_OBJECT (Object Type, L12)
Dokumentart		AR_OBJECT (Dokumentart, L14)
SAP Object Clearname		SAP_OBJ_TXT (SAP Object Clearname, L13)
Creator		SAP_CREATOR (Creator, L16)

To retrieve data from the SAP function module, all you have to do is create fields for the SAP attribute names and assign them to the metadata form. You can use the SAP transaction ELO Connectivity Pack – Indexdownload report contained in the SAP transport for this.

'SAPDATA' SAP form

General SAP document metadata form for all SAP objects with the required basic information. This form is mandatory when filing documents from SAP and should only be changed if necessary.

Please note

Do not delete any fields in the metadata form **SAPDATA**.

SAPDATA

Name	SAPDATA	ID	35
Translation variable	erp.sap.mask.sapdata.maskname	GUID	(F77E9B3B-1010-71FC-1051-F41EC19644BC)
Last change	22.05.2020 12:24	Save data as a table i	

> Usage

▼ Fields

ArchiveLink	Objectdata	X
Short name	Short name	
Date	Date	Version
Filing date	Filing date	Editor
SAP Archivepath	SAPPATH (SAP Archivepath, L1)	
SAP Creation Date	SAPCRDATE (SAP Creation Date, L2)	
SAP Alterationdate	SAPMODATE (SAP Alterationdate, L3)	
SAP Version	SAPAPVER (SAP Version, L4)	
SAP ArchiveLink Rights	SAPPRCH (SAP ArchiveLink Rights, L5)	
SAP Filename	SAPFILEN (SAP Filename, L6)	
SAP Filesize	SAPELODATA (SAP Filesize, L7)	
SAP Content-Type	SAPCONT (SAP Content-Type, L8)	
SAP Barcode	SAPBARC (SAP Barcode, L9)	
ArchiveLink Protocol Version	SAPALVER (ArchiveLink Protocol Version, L10)	

ArchiveLink	Objectdata
Object ID	OBJECT_ID (Object ID, L11)
Object Type	SAP_OBJECT (Object Type, L12)
SAP Object Clearname	SAP_OBJ_TXT (SAP Object Clearname, L13)
Dokumentart	AR_OBJECT (Dokumentart, L14)
Document Type Clearname	SAP_ARO_TXT (Document Type Clearname, L15)
Creator	SAP_CREATOR (Creator, L16)
Description	SAP_DESCR (Description, L17)
Name	SAP_INST_NAME (Name, L18)
Geschäftsjahr	FISCALYEAR (Geschäftsjahr, L19)
Creditor name	VENDOR_NAME (Creditor name, L20)
Lieferantennummer (Kreditor)	VENDOR_VENDORNO (Lieferantennummer (Kreditor), L21)

SAP metadata form 'SAP-Folder'

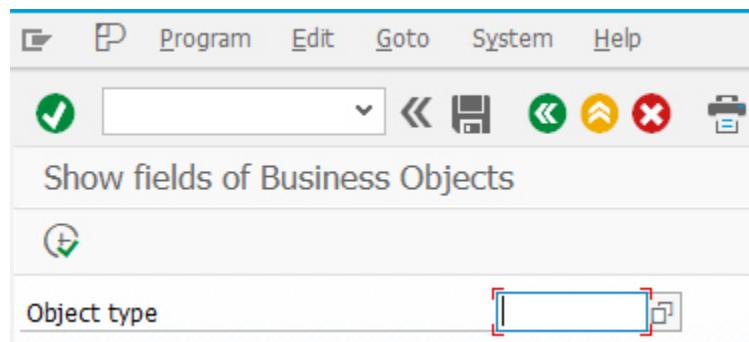
The *SAP-Folder* metadata form is used for the bottom folder of the new structure. This folder only contains documents.

SAP-Folder			
Name	SAP-Folder	ID	39
Translation variable	erp.sap.mask.folder.maskname	GUID	(577DECF4-0725-5C47-1FEE-52BC71C6DC31)
Last change	22.05.2020 12:24	Save data as a table ?	
> Usage			
< Fields			
Basis			
Short name	Short name		
Date	Date	Version	Version
Filing date	Filing date	Editor	Editor
ELOINDEX	ELOINDEX (ELOINDEX, L1)		
Object Type	SAP_OBJECT (Object Type, L2)		
Object ID	OBJECT_ID (Object ID, L3)		
Dokumentart	AR_OBJECT (Dokumentart, L4)		

'/ELO/BO_FIELDS' transaction

This transaction enables you to show all SAP attribute fields for SAP business objects.

1. After importing transports on your SAP system, start the transaction via the transaction code /
n/ELO/BO_FIELDS.



2. To search business objects, you can use the F4 help or enter the object type. In this example, we will use the object type *KNA1 (Customer)*.

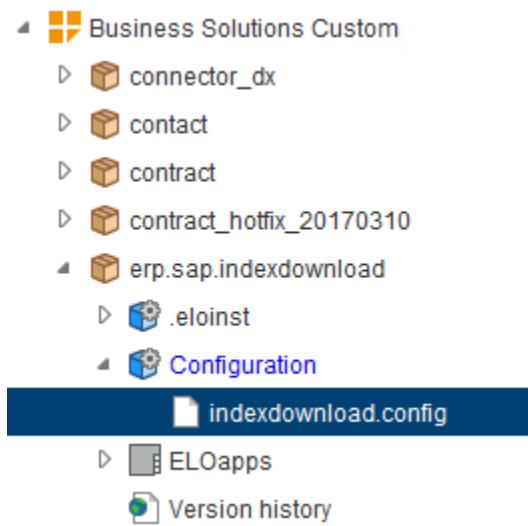
Show fields of Business Objects

           	
Technical Name (index group field)	Description
OBJECT_ID	SAP ArchiveLink: Object ID (object identifier)
SAP_OBJECT	SAP ArchiveLink: Object type of business object
AR_OBJECT	Document type
SAP_CREATOR	Creator Name
SAP_DESCR	Description for Browser Item (Display Attribute)
SAP_OBJ_TXT	Name
SAP_ARO_TXT	SAP ArchiveLink: Document type
PURCHASEORDER	Purchase Order
PURCHASINGGROUP_DESCRIPTION	Name
PURCHASINGGROUP_OUTPUTDEVICE	Output Device
PURCHASINGGROUP_PHONENUMBER	Telephone Number
PURCHASINGGROUP_PURCHASINGGROUP	Purchasing Group
PURCHORGANIZATION_DESCRIPTION	Object Description
PURCHORGANIZATION_PURCHORGANIZATION	Purch. Organization
RELEASEGROUP	Release Group
RELEASEINDICATOR	Release Indicator
RELEASESTATUS	Release Status
RELEASESTRATEGY	Release Strategy
SUBJECTTORELEASE	Subject to Release
VENDOR_CITY	City
VENDOR_COUNTRY	Country
VENDOR_NAME	Name
VENDOR_POSTALCODE	Postal Code
VENDOR_PURCHASINGBLOCK	Purchasing Block
VENDOR_STREET	Street

Once you have selected a business object and run the function, all possible attributes for this SAP business object are shown.

3. Now, enter the technical name of the desired field as a field in your metadata form.

ELO configuration



Once you have implemented the *erp.sap.indexdownload* package, go to *Configuration > indexdownload.config*. The ELO Connectivity Pack – Indexdownload configuration interface opens in ELO.

Please note

Only edit the configuration in the *Business Solutions Custom* path to ensure the system can be updated in the future. When a module is updated, the configuration in the *Business Solutions* folder can be overwritten.



The configuration interface contains two areas: one for general settings and the *Structure and rights* area. In addition, you can enable additional configuration settings by selecting the *Expert mode* option.

Information

When installing the solution with ELO 10, you may not be able to open the configuration interface. For more information, refer to the Installation information.

Configuration - General

General

<input type="radio"/> Custom	Move enabled	<input checked="" type="checkbox"/>
<input type="radio"/> Custom	URL (ELO Smart Link Content Server)	http://10.49.110.199:9060/sl-DEMO12/ContentServer
<input type="radio"/> Default	Form name	SAPDATA

Move enabled: Enable/disable document sorting from the chaos folder based on specified criteria.

URL (ELO Smart Link Content Server): URL to the ELO Smart Link or ELO ArchiveLink interface (server URL):

Form name: Default metadata form for Indexdownload. The same metadata form as the one in the ELO Smart Link or ELO ArchiveLink configuration must be entered here.

Index download button

<input type="radio"/> Default	Metadata field to check whether the index has already been downloaded	OBJECT_ID
<input type="radio"/> Default	Restrict to user group	User group name

Metadata field to check whether the index has already been downloaded: Check for downloading the index again (button).

Restrict to user group: Here, you can enable the button to download an index (see chapter the Additional functions) for a specific user group. If the field is left blank, all ELO users can use the button.

Indexdownload

<input type="radio"/> Default	Clear all fields before re-downloading that are not required to link the document to SAP	<input checked="" type="checkbox"/>
-------------------------------	--	-------------------------------------

Clear all fields before re-downloading that are not required to link the document to SAP: This means that the index fields (besides the ArchiveLink system fields) are cleared before the index is

downloaded for a document again. If this function is disabled, the metadata are continuously added to the fields.

View in expert mode

The screenshot shows a configuration interface for SAP ArchiveLink. At the top, there is a blue header bar with the text "Indexdownload". Below this, there are two tabs: "Default" and "Clear all fields before re-downloading that are not required to link the document to SAP". The second tab is selected and has a checked checkbox. Below these tabs, there is a section titled "SAP document link fields" with a "Default" tab selected. A list of fields follows:

- SAPPATH
- SAPCRDATE
- SAPMODATE
- SAPALVER
- SAPAPVER
- SAPPRCH
- SAPFILEN
- SAPELODATA
- SAPCONT
- SAPBARD

SAP document link fields: List of all default SAP ArchiveLink fields that are not cleared when the option above is selected.

Configuration - Structure and rights

In the following area, you can create different scenarios, e.g. for setting up a structure or assigning permissions.

Move actions

Metadata fields for determining the name

In the first part of the *Structure and rights* configuration interface, you can configure the fields that are used to determine names.

Metadata fields for determining the name

Default Metadata fields

SAP_DESCR

SAP_ARO_TXT

+ Add entry

Metadata fields: The short name of the document is generated based on these entries. The fields entered here are checked. If the first field is blank, the second field is checked, etc. By default, the *SAP_DESCR* field contains a field from the SAP table *TOAAT* (*description*).

Template for determining the new path

In the second area, you can create a new document path using Handlebar syntax. You can use the fields configured in *SAPDATA* for this. It is also possible to move the documents to the *SAP* path at this point and, after successfully changing the metadata form and downloading a new index, create a structure with your own scripts. See also the chapter Process.

Template for determining the new path

Default Path (Handlebars support)

```
if(SAP){{{(sord.objKeys.SAP_OBJ_TXT)}} {{{(sord.objKeys.SAP_OBJECT)}}}}||{{(#if sord.objKeys.FISCALYEAR)}{{(sord.objKeys.FISCALYEAR)}}}
||{{(else if sord.objKeys.VENDOR_VENDORNO)}{{(sord.objKeys.VENDOR_VENDORNO)}} - {{{(sord.objKeys.VENDOR_NAME)}}}}{{(else)}}{{(#if sord.objKeys.SAP_INST_NAME)}}{{(sord.objKeys.OBJECT_ID)}} - {{{(sord.objKeys.SAP_INST_NAME)}}}}{{(else)}}{{(sord.objKeys.OBJECT_ID)}}}}{{(#if}}
```

Mandatory fields (in the path; also used for the search)

SAP_OBJ_TXT

SAP_OBJECT

OBJECT_ID

+ Add entry

Path (Handlebars support): Path for building the folder structure based on field values of the form *SAPDATA*.

Mandatory fields (in the path; also used for the search): If you have defined mandatory fields in the path, enter them here. ELO creates the folder structure based on these fields.

Form change (document)

The next step is to change the metadata form. At this point, you can specify which metadata fields and values to assign to the desired metadata form.

Form change (document)		
Custom	Metadata field and value assignment to form	
Metadata field	Value	Form name
SAP_OBJECT	BUS2012	SAP-PurchaseOrder
SAP_OBJECT	LFA1	SAP-Vendor
SAP_OBJECT	VBRK	SAP-BillingDocument
SAP_OBJECT	BUS2032	SAP-CustomerOrder

+ Add entry

Metadata field and value assignment to form: Each metadata field in the default form *SAPDATA* can be used to change forms in order to assign documents not only new forms but also new fields.

Recommendation: Set a metadata form for an SAP business object or document type.

When you change the form, the index is downloaded again, as the new form may contain or use other fields.

Folder form

Folder form		
<input checked="" type="checkbox"/> Default	Form name	SAP-Folder

Form name: The form assigned to the bottom folder of the new structure can be defined here. This folder only contains documents. By default, ELO uses the *SAP-Folder* form.

If the *Expert mode* field under the menu is selected, two additional menu items appear.

Folder form

Default	Form name	SAP-Folder
Default	Update short name	<input type="checkbox"/>
Default	Update repository path	<input type="checkbox"/>
Default	Apply metadata from document	
Folder metadata field	Document metadata field	Ignore during retrieval
SAP_OBJECT	SAP_OBJECT	<input type="checkbox"/> Ignore during retrieval ▲ ▼ ✕
OBJECT_ID	OBJECT_ID	<input type="checkbox"/> Ignore during retrieval ▲ ▼ ✕
AR_OBJECT	Document metadata field	<input type="checkbox"/> Ignore during retrieval ▲ ▼ ✕
+ Add entry		

Update short name: If this option is enabled, the [Path \(Handlebars support\)](#) setting is interpreted differently. The last entry is no longer used to set the path, but instead defines the short name of the item to be filed.

Example with template path:

```
SAP||Orders||{{{{sord.objKeys.OBJECT_ID}}}}
```

The OBJECT_ID has the value 450001000.

Previously, the document was filed to the *SAP||Orders||450001000* structure. By enabling the option, the document is now filed to the *SAP||Orders* structure and is assigned the value 450001000 as its short name. If you want to keep the previous structure when enabling this option, you will have to add an element to the template path. The example would then look like this:

```
SAP||Orders||{{{{sord.objKeys.OBJECT_ID}}}}||{{{{sord.objKeys.OBJECT_ID}}}}
```

The result: The document would be filed to *SAP||Orders||450001000* as before, but also assigned the short name *450001000*.

It is also possible to initiate this via an event type linkage in the SAP system.

Update repository path: This function can be used to configure whether the short name is checked and modified on document storage and the subsequent index download, and whether it is modified and moved if there is new metadata. It is also possible to initiate this via an event type linkage in the SAP system.

Apply metadata from document: Defines the fields from the first document of the respective folder which are then applied to the *SAP-Folder* form defined above.

Information

The *AR_OBJECT* (*document type*) field is required for future developments. This field is left blank in the current configuration.

Information

If you make any adjustments to the configuration or script, you need to restart the ELO Indexserver and ELO Automation Services Tomcat services.

Assignment of user and group rights

In this area, you can assign user and group rights to a document or its structure based on specific values.

The screenshot shows a user interface for assigning rights. At the top, a blue header bar contains the title 'Assignment of user and group rights'. Below it, a 'Custom' tab is selected. Underneath, there's a section titled 'Rules and matches' with four checkboxes: 'Document', 'Structure', 'Replace rights', and 'RegEx'. To the right of these checkboxes are two input fields: 'Metadata field' and 'Value', both highlighted with a red border. Below these fields are two buttons: a blue one labeled '+ Add entry' and a white one also labeled '+ Add entry'.

Document: If the box is checked, the rights are set for the document.

Structure: If this box is checked, the rights are assigned to the folder structure in which the document is stored. The structure used here only affects the structure that is explained in the chapter [Assignment of template structures](#).

Replace rights: ELO takes on the rights of the parent element by default. If this option is enabled, the rights of the parent element are discarded and only the set rights are assigned. Please note that this function only works if a folder structure does not yet exist (see section [Assignment of template structures](#)).

Metadata field: The technical name of the metadata field to be checked is entered here.

Value: The value that is compared with the value from the specified metadata field. The rights are set if there is a match.

RegEx: If this box is checked, the entry in the *Value* field must be a regular expression. The convention for the regular expression is without slashes.

Example: *BUS201[0,2]* would result in matches with the values *BUS2010* and *BUS2012*.

Rights for users and groups

User or group name	<input type="checkbox"/> R	<input type="checkbox"/> W	<input type="checkbox"/> D	<input type="checkbox"/> E	<input type="checkbox"/> L	<input type="checkbox"/> P	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
+ Add entry								

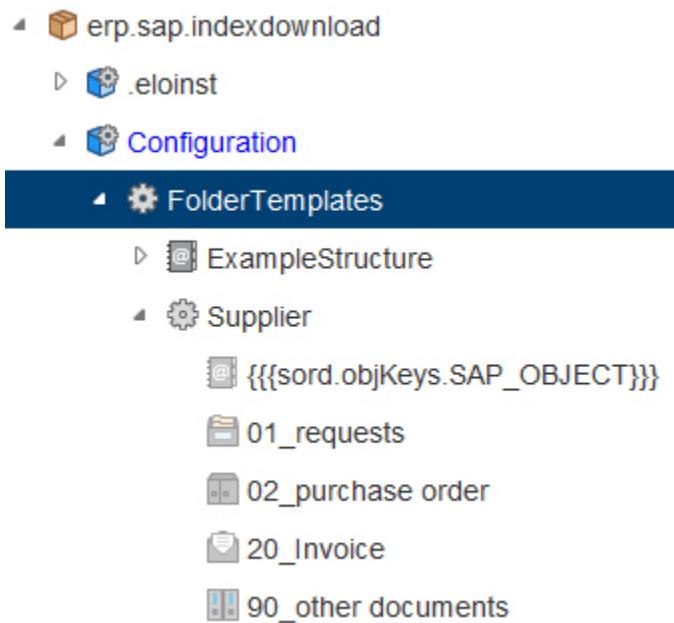
User or group name: Enter the desired users or groups that you want to assign rights to here.

RWDELP: Use the check boxes to set the different permissions types for users or groups.

- R = View
- W = Change metadata
- D = Delete
- E = Edit
- L = Edit list
- P = Set permissions (from ELO 12)

For more information, refer to the documentation [Concept for assigning rights and permissions](#).

Assignment of template structures



Besides creating a simple structure based on metadata, this is where you automatically create a folder structure based on a configured template when filing a document. The required template can be created in the configuration area for the Indexdownload function at the following path:

Administration // Business Solutions Custom // erp.sap.indexdownload // Configuration // FolderTemplates

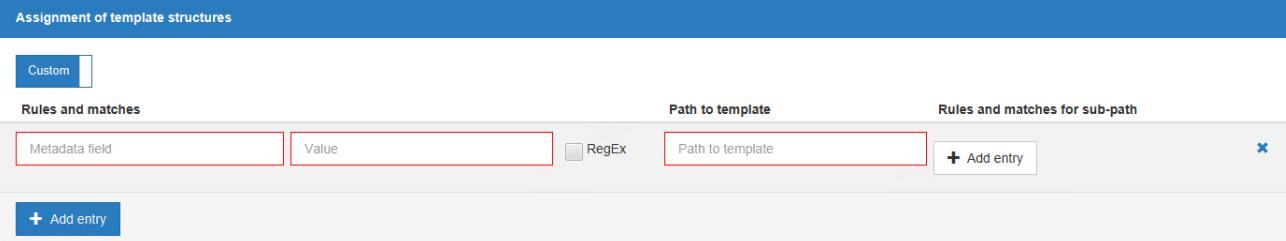
If a document is stored for an object for the first time, a template structure is created based on the configuration.

Information

It is also possible to create the name of a folder in the folder template using Handlebars syntax. Example text for an SAP object:

```
◀  Supplier
  @ {{{sord.objKeys.SAP_OBJECT}}}
```

Rules and matches



Assignment of template structures					
Custom		Rules and matches		Path to template	
Metadata field	Value	<input type="checkbox"/> RegEx	Path to template	Rules and matches for sub-path	
<input type="button" value="+ Add entry"/> <input type="button" value="X"/>					

Metadata field: Enter the technical name of the metadata field to be checked here.

Value: Enter the value here that is compared with and must match the value in the specified metadata field to create the template structure. If the values match, the structure configured under *Path to template* is created.

RegEx: If you want to use a regular expression in the *Value* field, you must check this box. The convention for the regular expression is without slashes. Example: *BUS201[0,2]* would result in matches with the values *BUS2010* and *BUS2012*.

Path to template: Enter the path to the template structure here. For example, if you created a template structure *PurchaseOrder*, this would mean that when you file a corresponding document, the structure is created at the following path provided all the conditions are met:

Administration // Business Solutions Custom // erp.sap.indexdownload // Configuration // FolderTemplates // PurchaseOrder

At this point, you can use a separator to define a deeper folder structure. However, the separator used must be leading, e. g. if you enter */Vendor/Orders*, the slash is recognized as a separator. Handlebars syntax is also allowed, which enables you to control templates flexibly.

If you use Handlebars syntax to create the structure, you can only use metadata from the fields of the initial template, in most cases *SAPDATA*. If more information is required at this point than is available in the default system, the form *SAPDATA* and the index download need to be customized.

Example

Assignment of template structures

Custom	Rules and matches	Path to template
	SAP-OBJECT BUS201[0,2]	<input checked="" type="checkbox"/> RegEx /{{{sord.objKeys.SAP_OBJECT}}}
+ Add entry		

The system checks the field *SAP-OBJECT* for the contents *BUS2010* and *BUS2012*. If this condition is met, the corresponding template structure is created from the following path:

```
/{{{sord.objKeys.SAP\_OBJECT}}}
```

In this case it is the structure below */BUS2010* or */BUS2012*, as defined by the rule.

Assignment of documents in the template structure

There is another function that allows you to store documents (e.g. by document type) in child directories based on rules. You can create these directories using the template structure. If a specified child directory does not exist within the template structure, it is created.

Rules and matches for sub-path

Metadata field	Value	<input type="checkbox"/> RegEx	Subpath (first character = separator, handlebars supported)	x	x
+ Add entry					

Metadata field: The technical name of the metadata field to be checked is entered here.

Value: Enter the value to be compared in the specified metadata field here. If there is a match, the template structure of the defined subpath is created, see below.

RegEx: If you want to use a regular expression in the *Value* field, you must check this box. The convention for the regular expression is without slashes. Example: *BUS201[0,2]* would result in matches with the values *BUS2010* and *BUS2012*.

Subpath: The subpath defined here is based on the path of the template structure. Again, the first character to be defined is the separator. You can specify any depth, and also use Handlebars syntax. If the path does not exist, the structure is created.

Process

After filing a document from SAP, initially only the standard information such as *SAPPATH* (SAP Archivepath) or *SAPCRDATA* (SAP Creation Date) is stored. Here are two possible scenarios of how you can automate the index download:

- With the configuration setting in ELO Smart Link. You will find details in the configuration documentation of ELO Smart Link for SAP® ERP.
- With event linkage from the SAP system. This function is also used, for example, in the ELO Integration Client environment to obtain metadata from the SAP system and build structures in ELO, which the ELO Integration Client requires. For more information, refer to the [ELO Toolbox for SAP® ERP](#) documentation.

The screenshot shows the 'Edit instance' configuration page for the 'Instance scheduler'. It includes sections for 'Interval in seconds in for queue processing (Indexdownload, Barcode Upload, Datatransfer)', a note about restarting the instance, and a 'BarcodeTimer' configuration section.

Interval in seconds in for queue processing (Indexdownload, Barcode Upload, Datatransfer)

31 Seconds

Note
Changes to this value must be confirmed by restarting the instance.

BarcodeTimer

If the index download function is enabled in the configuration settings of ELO Smart Link, there is a system delay until the index is download. The download interval can be set in the ELO Smart Link configuration.

The function module in SAP is now contacted and an initial index download is performed based on the standard information. As the SAP function module can identify the referenced document in SAP based on *SAPPATH*, it returns all available index values to ELO. Based on the metadata fields that have the same attribute names in SAP, ELO can now complete the required metadata fields (e.g. *SAP_OBJECT* or *OBJECT_ID*).

If the function *Move enabled* is selected, the data is transferred to the ELO structure via an ELOas script based on the metadata fields in the form *SAPDATA*. In addition, the template structure is created if the configuration is stored. The defined permissions in the configuration are also set. The next step is to change the metadata form, if configured.

Information

The structure is based on the data from the first index download. If additional or different fields are required to create the structure, these can be added in the base form (*SAPDATA*).

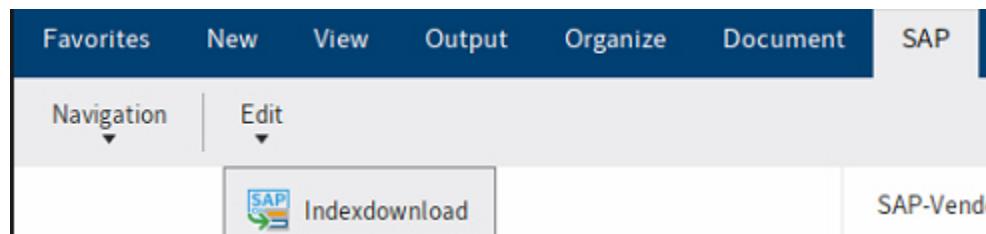
After the new form has been selected, a second index download is performed and the available metadata fields of the new form are completed. The index download also uses the same SAP attribute names here.

Information

If you only want to change the metadata form, you can modify the corresponding script so that a dummy path (SAP) is used instead of the chaos folder, for example.

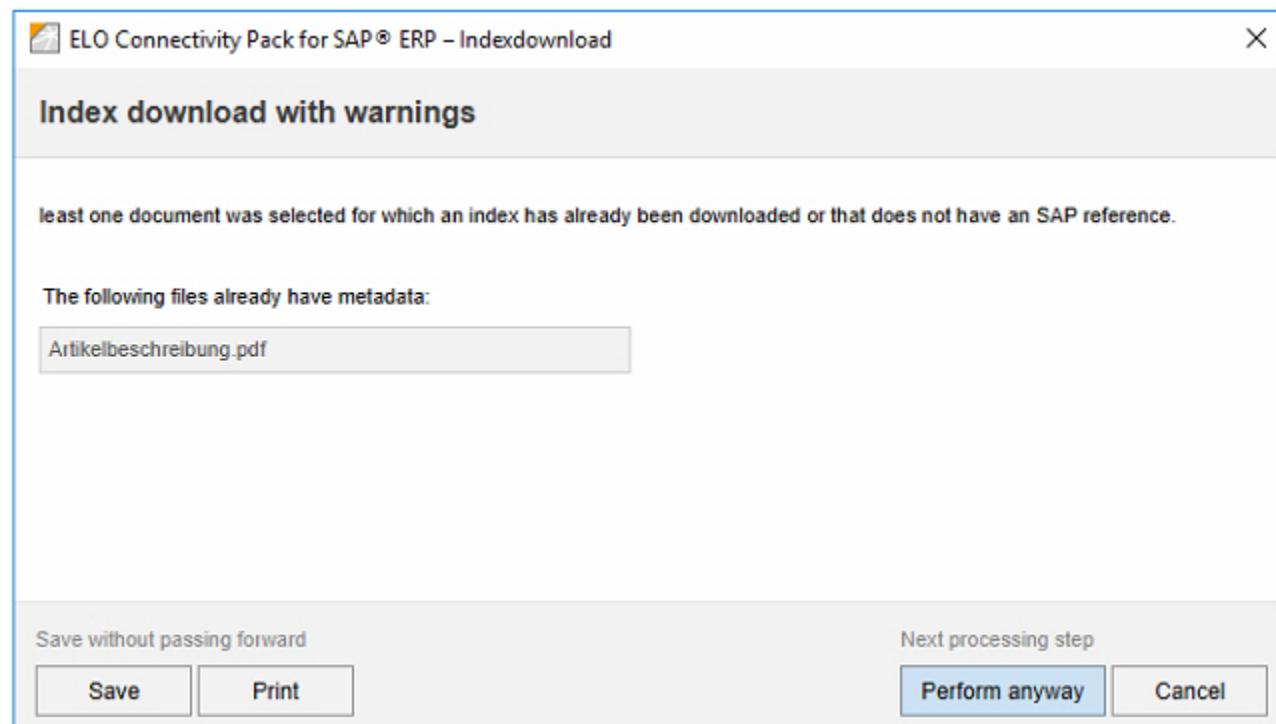
Additional functions

'Indexdownload' button



You will find the *Indexdownload* button on the new SAP tab. Click this button to download the index of the selected documents again. Depending on the configuration, the current metadata values are either deleted or retained. The values are separated with a semicolon.

The function checks whether the selected documents have a reference to the SAP system and whether they have already been indexed. If either of the two cases applies, these documents are listed:



You now have the option to abort the process or continue. Another dialog box appears, which either confirms the process was successful or lists all documents for which the index download was not completed.

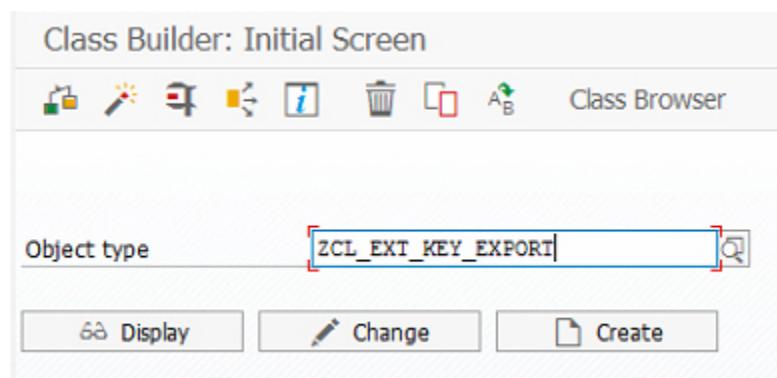
Please note

The index download requires a certain amount of time for indexing. During this time, do not lock any documents that you are performing the index download on. In addition, when using a manual index download, you should limit the number of documents, otherwise the process may take a long time and can cause the Indexserver process to abort. We recommend selecting no more than 500 entries.

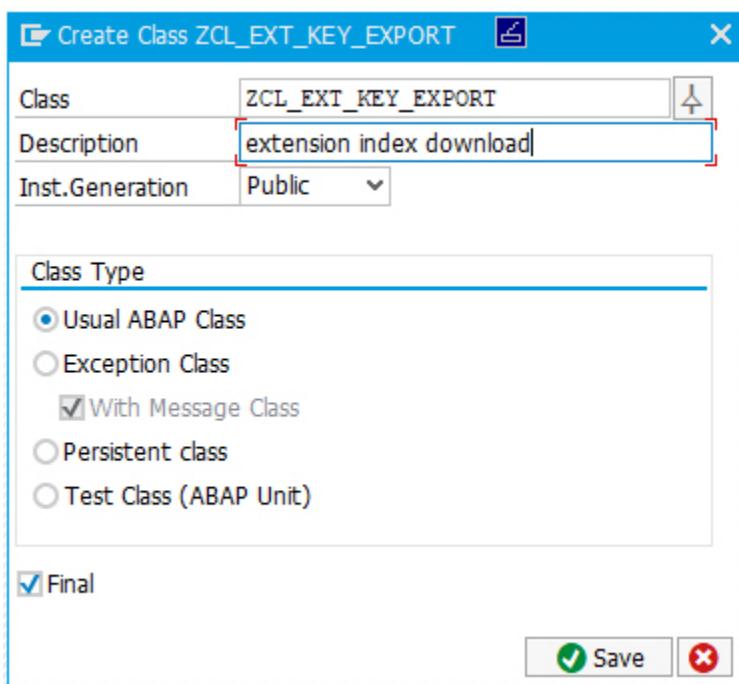
ELO Connectivity Pack – Indexdownload extensions

If you would like to add more fields or implement an existing development to exchange data, you can create an extension for ELO Connectivity Pack – Indexdownload. ELO only recommends this option if you have extensive knowledge of SAP-ABAP programming.

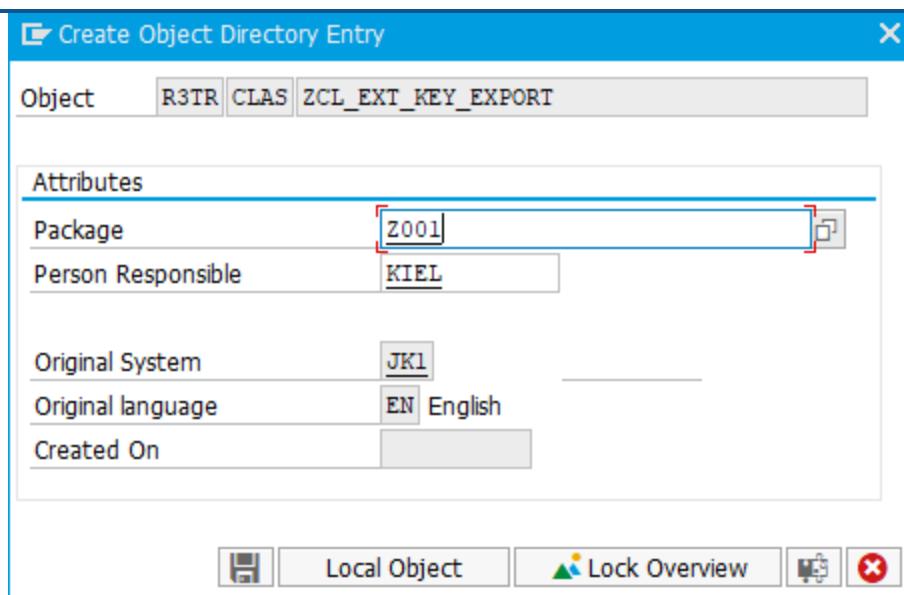
Create class



1. Create a new class (e.g. *ZCL_EXT_KEY_EXPORT*) in the transaction *SE24*.



2. Enter the description and confirm with *Save*.



3. Enter the desired/required package.
4. Select the desired transport order or create a new one.

Properties	Interfaces	Friends	Attributes	Methods	Events	Types	Aliases
	(selected)						
<input type="checkbox"/> Filter							
Interface	Abstract	Final	Model...	Description			
/ELO/IF_EXTEND_KEY_EXPORT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interface für eigene Erweiterung			

5. On the *Interfaces* tab, you can now enter the */ELO/IF_EXTEND_KEY_EXPORT* interface.
6. Next, you have to implement the two methods on the *Methods* tab.

The */ELO/IF_EXTEND_KEY_EXPORT~GET_KEYS* method is called following standard processing when downloading an index. Here, you can add fields and values or replace the entire content. The */V_TOA* import parameter is assigned the entry from the link table (e.g. content repository, document ID, SAP object, object ID, etc.). The *CT_KEYTAB* changing parameter contains the completed value pair tables.

Class Builder: Class ZCL_EXT_KEY_EXPORT Change

The screenshot shows the SAP Class Builder interface for the class ZCL_EXT_KEY_EXPORT. At the top, there's a toolbar with various icons. Below it is a table showing parameters and their descriptions. The main area displays the implementation code for the /ELO/IF_EXTEND_KEY_EXPORT~GET_KEYS method.

Ty.	Parameter	Type spec.	Description
►□	IV_TOA	TYPE TOAV0	SAP ArchiveLink: Verknüpfungstabelle
►□	CT_KEYTAB	TYPE /ELO/CL_KEY_EXPORT=>TT_KEYTAB	Die modifizierbare Wertetabelle

Method: /ELO/IF_EXTEND_KEY_EXPORT~GET_KEYS

```

1 METHOD /elo/if_extend_key_export~get_keys.
2
3   FREE: ct_keytab.
4   CALL FUNCTION 'Z_KEYEXPORT'
5     EXPORTING
6       contentrep = iv_toa-archiv_id
7       docid      = iv_toa-arc_doc_id
8       mode       = ' '
9     TABLES
10    keytab     = ct_keytab[].
11
12 ENDMETHOD.

```

This figure shows an implementation example that deletes all default values and calls an implemented function module.

In addition, the name and descriptions of possible fields for the implementation should be maintained in the */ELO/IF_EXTEND_KEY_EXPORT~GET_POSSIBLE_FIELDS* method.

Class Builder: Class ZCL_EXT_KEY_EXPORT Change

Ty.	Parameter	Type spec.	Description
IV_SAP_OBJECT		TYPE TOAV0-SAP_OBJECT	SAP ArchiveLink: Objekttyp des Business Objekts
CT_KEYTAB		TYPE /ELO/CL_KEY_EXPORT=>TT_KEYTAB	Modifizierbare Wertetabelle

Method /ELO/IF_EXTEND_KEY_EXPORT~GET_POSSIBLE_FIELDS Inactive

```

1  METHOD /elo/if_extend_key_export~get_possible_fields.
2   DATA: ls_row LIKE LINE OF ct_keytab.
3   .
4   FREE: ct_keytab.
5
6   IF iv_sap_object EQ 'BKPF'.
7
8     ls_row-name = 'BELEGDATUM'.
9     ls_row-wert = 'The date of the document'.
10    APPEND ls_row TO ct_keytab.
11
12    "ls_row-name = 'FIELDNAME'.
13    "ls_row-wert = 'Description'.
14    "APPEND ls_row to ct_keytab.
15
16    "ESLEIF iv_sap_object EQ 'XXXXXX'.
17    "...
18
19  ENDIF.
20  ENDMETHOD.

```

The figure shows an implementation for the case that the called function module from the previous example only returns one field named *DATEDOCUMENT* for the *BKPF* business object.

Implementation example

An implementation example that can be used is available in the */ELO/CL_KE_EXT_BKPF* class. This implementation outputs the fields *Document type*, *Document date*, and *Posting date* for the *BKPF* (*accounting document*) business object.

Maintaining extensions



The */ELO/EXT_KE* transaction is available to maintain developed extensions. This transaction starts the maintenance dialog box for the */ELO/KE_EXT* table.

In the transaction, you can maintain the class created for a business object. You also have the option to create an asterisk entry (*). This is then used for all business objects without a specific entry.

New Entries: Overview of Added Entries	
?	

ObjectType	Class/Interface
BKPF	/ELO/CL_KE_EXT_BKPF

The figure shows a corresponding entry.

Using the ELO Integration Client within the SAP system

With the ELO Integration Client, ELO and documents filed in ELO can now be used within SAP transactions. This requires the products ELO Connectivity Pack for SAP® ERP and ELO Toolbox for SAP ERP®. The requirements listed in the chapter Basics also have to be met.

The ELO Integration Client is part of the ELO installation packages as well as SAP transports described in the chapter Installation. Configuration of the ELO Integration Client within ELO is described in the documentation for the ELO Toolbox for SAP® ERP.

Indexdownload transfer (see also ELO Toolbox for SAP® ERP)

To use the ELO Integration Client in an SAP transaction, ELO requires a defined path to the corresponding SAP document. Creating this path requires an RFC destination within the SAP system. The path defined in ELO Connectivity Pack – Indexdownload is used to create the path. For more details on creating a path, refer to the chapter ELO configuration.

1. Create a new RFC destination via SAP transaction SM59.

RFC Destination ELO20

Connection Test		
RFC Destination	ELO20	
Connection Type	G HTTP Connection to External Serv	
Description	Description 1: ELO Description 2: Description 3:	

Administration Technical Settings Logon & Security Special Options

Target System Settings

Target Host	SAPDEMO	Service No.	9090
Path Prefix:			

HTTP Proxy Options

Global Configuration	
Proxy Host	
Proxy Service	
Proxy User	
Proxy PW Status	is initial
Proxy Password	*****

'Technical Settings' tab

Name	Value
RFC destination	"Basic entry"
Connection type	http connection to ext. server
Description 1	"Basic entry"
Target machine	ELO servers
Service no.	Port number of the ELO REST Service on the ELO server

"Logon & Security" tab

RFC Destination ELO20

Connection Test	
RFC Destination	ELO20
Connection Type	G HTTP Connection to External Serv
Description	
Description 1	<input type="text"/>
Description 2	<input type="text"/>
Description 3	<input type="text"/>

Administration Technical Settings **Logon & Security** Special Options

Logon Procedure

Logon with User

Do Not Use a User
 Basic Authentication

User	ELO Service
PW Status	saved
Password	<input type="password"/> ****

Logon with Ticket

Do Not Send Logon Ticket
 Send Logon Ticket Without Ref. to a Target System
 Send Assertion Ticket for Dedicated Target System

System ID Client

Name Value

User ELO user

Password Password of the relevant user

Configuring the Indexdownload destination

The transfer is realized via the *ELO REST* module.

Object Type	Destination Name	Pfadpräfix
*	ELO20	/ix-SAPDEMO/plugin/de.elo.ix.plugin.proxy/rest/

To establish a connection, the entry to the *ELO REST* module has to be configured as the destination name and path prefix via SAP transaction */ELO/KE_DESTINATIONS*.

Column	Explanation
Object type	Specification of the desired SAP business object, such as <i>BUS2012</i> for orders. Alternative: * as a wildcard
Destination name	Name of the ELO destination
Path prefix	Path to the ELO REST Service module via the ELOix proxy plug-in

The path also points to the delivered RF within ELO REST:

```
/ix-SAPDEMO/plugin/de.elo.ix.plugin.proxy/rest/api/misc/functions/
RF_erp_sap_indexdownload_service_CreatePath
```

From version 1.0.9

The table was extended by the column role to get to different Integration Clients belonging to ELO systems, depending on the user role.

ELO Key Export - Destination Maintenance				
Object Type	Role	Destination Name	Path Prefix	Remote Function Name (neue REST-API)
*	Z_TC_ELO_2	ZSAPDEMO1	/ix-Sapdemo1/plugin/de.elo.ix.plugin.proxy/rest/api/misc/functions/RF_erp_sap_indexdownload_service_CreatePath	
*	Z_TC_ELO_2	ZSAPDEMO2	/ix-Sapdemo2/plugin/de.elo.ix.plugin.proxy/rest/api/misc/functions/RF_erp_sap_indexdownload_service_CreatePath	
BUS2012	*	ZSAPDEMO3	/ix-Sapdemo3/plugin/de.elo.ix.plugin.proxy/rest/api/misc/functions/RF_erp_sap_indexdownload_service_CreatePath	
BUS2012	Z_TC_ELO_4	IX_SAPDEMO4	/ix-Sapdemo4/plugin/de.elo.ix.plugin.proxy/rest/api/misc/functions/RF_erp_sap_indexdownload_service_CreatePath	RF_erp_sap_indexdownload_service_CreatePath

Column	Explanation
Object type	Specification of the desired SAP business object, such as <i>BUS2012</i> for orders. Alternative: * as a wildcard
Role	Role of the user
Destination name	Name of the ELO destination
Path prefix	Path to the ELO REST Service module via the ELOix proxy plug-in
Remote Function Name	Access to new REST-API via Indexserver

From ELO 21, the Rest-API interface can be called via the *Remote Function Name* column with the following function:

```
RF_erp_sap_indexdownload_service_CreatePath
```

If an entry exists in this field, the field *Path prefix* is ignored. The Indexserver must also be entered under the destination name.

Please note

The roles used must be valid and added in the user base accordingly. If the role of the user is not valid, a search is performed for the role *.

Example for determining the valid entry

The following example explains how the most precise match is compared to the roughest match to use the corresponding destination:

1. First, a search is performed for the biggest object type and object role match. Our example only refers to system 4 if the object type is *BUS2012* and the role *ZICELO_4* was added in the user's master record.
2. If no results are found, a check is performed to determine whether an entry with the available object type and role exists. Therefore, all users who do not have the role *ZICELO_4* for the object type *BUS2012* are forwarded to system 3.
3. If there is no match here either, a search is performed for a result for the object type * and the suitable role, and therefore, the user is forwarded to system 2 if the user has the role *ZICELO_2*, independently of the object type.
4. A default entry should be created so that a destination can be referred to if there is no match, in this example system 1.

Event-based structure

For the ELO Integration Client to have a starting point to display the structure within an SAP object, it requires a structure built based on SAP business object information. See also the documentation for ELO Toolbox for SAP® ERP.

This structure can also be used so that it builds on SAP events. Creating the same structures (e.g. a purchase file) in ELO when creating an order in the SAP system, for example, does not require the ELO Integration Client in this case.

Information

You can start background jobs with an event. An event is a signal to the background control that a certain status has been reached in the system. The background control then starts all jobs waiting for this event. For example, you can start an external program in the background with this event.

Even if you don't want to use the ELO Integration Client, the event-based structure also requires the configurations from the section [Using the ELO Integration Client within the SAP system](#).

The events to be used also have to be implemented/triggered (thrown) within an SAP system. If such an event occurs, a process is started via the receiver function module */ELO/SEND_KEYS_FROM_BO*. The configuration of the *Indexdownload* function is used. If a structure was

defined within the *indexdownload.config* file or the folder template is used for the corresponding SAP business object (see section Assignment of template structures), the defined structures are created in ELO.

Implementation example for event-controlled structure creation

In the following example implementation, we use the event *Order created* to create a defined order file from the folder template.

In the first step, you have to identify the event you want to use within the SAP system.

Event	Description
PurchaseOrder.changed	Changed
PurchaseOrder.canceled	Canceled
PurchaseOrder.created	Created
PurchaseOrder.released	Purchase Order Released
PurchaseOrder.releaseStepCreated	Release Step for Purchase Order Created
PurchaseOrder.significantlyChanged	Purchase Order Changed Significantly
PurchaseOrder.reset	Not Used
PurchaseOrder.rejection_start	Purchase Order Rejected
PurchaseOrder.rejection_stop	Cancel rejection
PurchaseOrder.parked	Purchase Order Parked
PurchaseOrder.completed	Purchase Order Complete

1. Call the desired SAP business object using SAP transaction *SWO1* and then open the *Events* menu item.

For SAP business object *BUS2012*, there is the event *PurchaseOrder.created*, which triggers a new order on creation.

In the next step, an event type linkage has to be created for this event.

2. Go to SAP transaction *SWETYPV* and create a new event type linkage.

Please note

Event type linkages may already be enabled on a customer system. Only make changes after consulting with the SAP partner.

Change View "Event Type Linkages": Details

New Entries

Object Category	BOR Object Type
Object Type	BUS2012
Event	CHANGED
Receiver Type	TRIGGER_INDEXDOWNLOAD

Linkage Setting (Event Receiver)

Receiver Call	Function Module
Receiver Function Module	/ELO/trigger_download
Check Function Module	
Receiver Type Function Module	
Destination of Receiver	

Event delivery	Using tRFC (Default)
----------------	----------------------

Linkage Activated
 Enable Event Queue

Behavior Upon Error Feedback	System defaults
Receiver Status	No errors

1. Create a new entry with the information from the following table and save it.

Name	Value
Object category	BOR object type
Object type	BUS2012
Event	CREATED
Receiver type	CREATE STRUCTURE (description)
Receiver call	Function module
Receiver function module	/ELO/SEND_KEYS_FROM_B0
Event delivery	Using tRFC (default)
Linkage enabled	X
Enable event queue	-
Behavior in event an error is returned	System default
Receiver status	No errors

Information

The receiver function module named here is part of the installation package for ELO Connectivity Pack for SAP® ERP – Indexdownload and is located in the respective SAP transports.

 Standard PO created under the number 4500022699

- ↳  Purchase Order (BUS2012)
 - ↳  E0003 - Elektronik OHG
 - ↳  4500022699
 -  Order offers
 -  Original documents
 -  Other documents
 -  Shipping documents

If the event type linkage is enabled and the configurations have been made for the business object *Purchase order BUS2012* in ELO, the structure is automatically created in ELO when an order is created in the SAP system (in our example, as a folder template).

Information

It is possible that an event required for an SAP business object may not exist. In this case, you can create a Z business object to define the desired event. However, you should only do this if you have extensive knowledge of SAP.

Automatic metadata update

The event type linkage can also be used to update metadata for a document already stored in ELO, for example if you want to update the metadata after changing an order. A new index download is performed for all documents of the corresponding SAP business object.

This event type linkage can also be used to download metadata after successfully linking a barcode via the barcode upload function.

Developed for this purpose, the function module */ELO/TRIGGER_DOWNLOAD* is part of the installation packages and the corresponding SAP transport.

Implementation example for updating metadata

As in the example for creating a structure (see section [Implementation example for event-controlled structure creation](#)), a relevant event has to be identified to update the metadata.

1. As described, you can use the SAP transaction *SWO1* for the corresponding object.

In the next step, you create the event type linkage for the identified event.

2. Go to SAP transaction *SWETYPV* and create a new event type linkage. This example uses the event *PurchaseOrder.changed*.

Please note

Event type linkages may already be enabled on a customer system. Only make changes after consulting with the SAP partner.

Change View "Event Type Linkages": Details

		New Entries						
Object Category	BOR Object Type							
Object Type	BUS2012							
Event	CREATED							
Receiver Type	CREATE STRUCTURE							
Linkage Setting (Event Receiver)								
Receiver Call	Function Module							
Receiver Function Module	/ELO/SEND_KEYS_FROM_BO							
Check Function Module								
Receiver Type Function Module								
Destination of Receiver								
Event delivery	Using tRFC (Default)							
<input checked="" type="checkbox"/> Linkage Activated								
<input type="checkbox"/> Enable Event Queue								
Behavior Upon Error Feedback	System defaults							
Receiver Status	No errors							

1. Create a new entry with the following information and save it.

Name	Value
Object category	<i>BOR object type</i>
Object type	BUS2012
Event	CHANGED
Receiver type	TRIGGER INDEXDOWNLOAD (description)
Receiver call	<i>Function module</i>

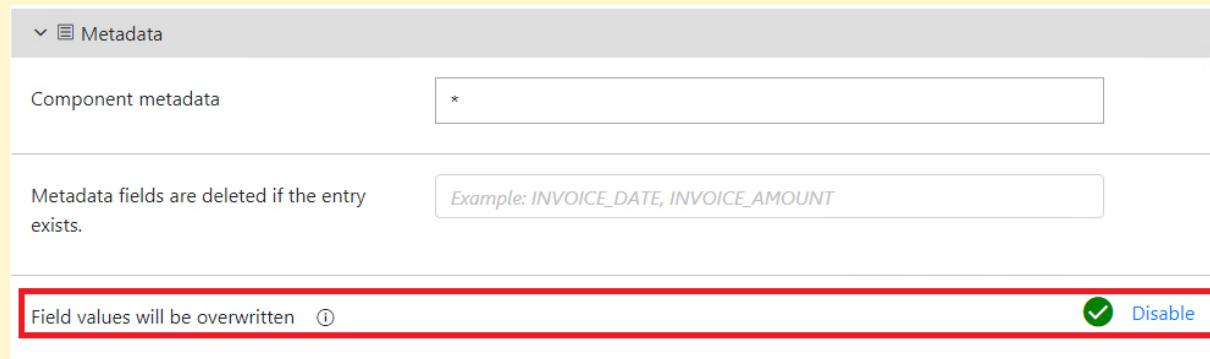
Name	Value
Receiver function module	/ELO/TRIGGER_DOWNLOAD
Event delivery	Using tRFC (default)
Linkage enabled	X
Enable event queue	-
Behavior in event an error is returned	<i>System default</i>
Receiver status	<i>No errors</i>

If the event type linkage is enabled and the configurations have been made for the business object *Purchase order BUS2012* in ELO, the index download function is carried out again when an order is changed in the SAP system.

Please note

Purchasing Group
007¶001

If the function *Field values will be overwritten* is disabled for the respective content repository in the *Metadata* area of the ELO Smart Link for SAP ERP® configuration, performing an index download again adds the metadata to the fields separated by a pilcrow character ¶.



Only enable this setting if you want the newly downloaded metadata to be maintained. In this case, the changes are documented in the *change history*.

Additional configuration parameters from the ELO Smart Link for SAP® ERP documentation can also be used.

Implementation example based on Barcode Upload

You can also use the function module */ELO/TRIGGER_DOWNLOAD* to respond to linking of a barcode in the SAP system. A destination does not have to be configured for this function.

In the first step, you have to configure the SAP system to create/throw an event upon successful barcode linking.

Generate Event BARCODE.ASSIGNED

1. Switch to SAP transaction *OAG1* and disable the entry *Create event BARCODE.ASSIGNED* in the bottom area.

If a new link entry is now created via the known standard barcode upload scenarios (see also the ELO Smart Link for SAP® ERP documentation), the event *BARCODE.ASSIGNED* is thrown/executed and can then be implemented as an event type linkage for the receiver function module */ELO/TRIGGER_DOWNLOAD*.

Example

New Entries: Details of Added Entries

Object Category	BOR Object Type			
Object Type	BARCODE			
Event	ASSIGNED			
Receiver Type	TRIGGER_INDEXDOWNLOAD_BARCODE			

Linkage Setting (Event Receiver)

Receiver Call	Function Module
Receiver Function Module	/ELO/TRIGGER_DOWNLOAD
Check Function Module	
Receiver Type Function Module	
Destination of Receiver	
Event delivery	Using tRFC (Default)
<input checked="" type="checkbox"/> Linkage Activated <input type="checkbox"/> Enable Event Queue	
Behavior Upon Error Feedback	System defaults
Receiver Status	No errors

Name	Value
Object category	BOR object type
Object type	BARCODE
Event	ASSIGNED

Name	Value
Receiver type	TRIGGER INDEXDOWNLOAD BARCODE (description)
Receiver call	<i>Function module</i>
Receiver function module	/EL0/TRIGGER_DOWNLOAD
Event delivery	<i>Using tRFC (default)</i>
Linkage enabled	X
Enable event queue	-
Behavior in event an error is returned	<i>System default</i>
Receiver status	<i>No errors</i>

Required permissions

The SAP service user entered in the configuration of ELO Smart Link or ELO ArchiveLink for SAP (e.g. *CPIC_ELO*) requires different authorizations to execute and read the index data in your SAP system. The authorizations listed here are derived from the supplied templates for *purchase order BUS2012*, *billing document VBRK*, and *vendor LFA1*. If fields or SAP objects are added to the index download, other authorizations are required for the SAP service user. You can use the SAP transaction *stauthtrace* to analyze this.

Permissions for 'purchase order BUS2012'

The following table lists the permissions for *purchase order BUS2012*.

Object	Field # (1)	Value # (1)	Field # (2)	Value # (2)
S_RFC	RFC_TYPE	FUGR	RFC_NAME	SYST
S_RFC	RFC_TYPE	FUGR	RFC_NAME	RFC1
S_RFC	RFC_TYPE	FUGR	RFC_NAME	SDIFRUNTIME
S_RFC	RFC_TYPE	FUGR	RFC_NAME	/ELO/CONNECT_EXPORT
F_LFA1_GEN	ACTVT	3		
CPE_SETTIN	ACTVT	3		
M_BEST_EKO	ACTVT	3	EKORG	*
M_BEST_EKO	ACTVT	8	EKORG	*
M_BEST_EKO	ACTVT	9	EKORG	*
M_BEST_EKG	ACTVT	3	EKGGRP	*
M_BEST_EKG	ACTVT	8	EKGGRP	*
M_BEST_EKG	ACTVT	9	EKGGRP	*
M_BEST_BSA	ACTVT	3	BSART	*
M_BEST_BSA	ACTVT	8	BSART	*
M_BEST_BSA	ACTVT	9	BSART	*
M_BEST_WRK	ACTVT	3	WERKS	*

Information

The permissions can be set even more restrictively at this point, e.g. by configuring restrictions on *EKORG* or *BSART*.

Permissions billing document VBRK

The following table lists the permissions for *billing document VBRK*.

Object	Field # (1)	Value # (1)	Field # (2)	Value # (2)	Field # (3)	Value # (3)
S_RFC	RFC_TYPE	FUGR	RFC_NAME	SYST	ACTVT	16
S_RFC	RFC_TYPE	FUGR	RFC_NAME	/ELO/ CONNECT_EXPORT	ACTVT	16
S_RFC	RFC_TYPE	FUGR	RFC_NAME	SYSU	ACTVT	16
S_RFC	RFC_TYPE	FUGR	RFC_NAME	/ELO/ CONNECT_EXPORT	ACTVT	16
S_TABU_DIS	DICBERCLS	FC01	ACTVT	3		
F_KNA1_GEN	ACTVT	3				

Permissions for 'vendor LFA1'

The following table lists the permissions for *vendor LFA1*.

Object	Field # (1)	Value # (1)	Field # (2)	Value # (2)	Field # (3)	Value # (3)
S_RFC	RFC_TYPE	FUGR	RFC_NAME	SYST	ACTVT	16
S_RFC	RFC_TYPE	FUGR	RFC_NAME	/ELO/ CONNECT_EXPORT	ACTVT	16
S_RFC	RFC_TYPE	FUGR	RFC_NAME	SYSU	ACTVT	16
S_RFC	RFC_TYPE	FUGR	RFC_NAME	/ELO/ CONNECT_EXPORT	ACTVT	16
F_LFA1_GEN	ACTVT	3				
F_LFA1_APP	ACTVT	3	APPKZ	*		
F_LFA1_GRP	ACTVT	1	KTOKK	*		