



Authors:  
BOPF Team

## Technical documentation

# BOPF Enabling for Enterprise Search

Document Version	Status	Date

Template Version: AC 2.2

**INTERNAL / CONFIDENTIAL**

## Contents

<b>1</b>	<b>Revision Log .....</b>	<b>Error! Bookmark not defined.</b>
<b>2</b>	<b>Stakeholders and Roles .....</b>	<b>Error! Bookmark not defined.</b>
<b>3</b>	<b>Overview .....</b>	<b>3</b>

# 1 Overview

## 1.1 Design time

To enable a business object for enterprise search, the following actions have to be performed:

### 1.1.1 Enable BO for enterprise search in BOPF design time

The screenshot shows the 'Business Object' configuration tab in SAP Business Object Designer. The 'Business Object' field is set to '/BOBF/TST\_SEARCH\_EXTRACTION' and the 'Description' is 'test enterprise search data extraction'. The 'Object Category' is 'not classified'. Under the 'Enterprise Search Enablement' section, the 'Business Object is Enter. Search enabled' checkbox is checked. The 'Model Name' is 'BOBF\_TST\_SEA\_EXTR' and the 'Data Extractor Class' is '/BOBF/CL\_SEA\_BO\_DATA\_EXTRACT'. Other checkboxes include 'Business Object Model generated', 'Business Object can be enhanced', 'Business Object has Authorization checks', 'Business Object is abstract', 'Business Object is final', and 'Business Object is GenIL enabled', all of which are currently unchecked.

Flag the Business Object as Enterprise Search enabled. Enter the model name and the name of the extraction class.

### 1.1.2 Create the enterprise search model from the BOPF BO model

Import the BO model in transaction ESH\_MODELER as shown below.

The screenshot shows the 'Modeler for Search and Analytics' interface. The 'List of Models' section is visible, and the 'Import' button is highlighted. The 'Import' dialog is open, showing a list of models. The model '/BOBF/TST\_SEARCH\_EXTRACTION' is selected in the list. Other models in the list include '/BOBF/TST\_SEARCH\_CPOINTER1', '/BOBF/TST\_SEARCH\_METADATA', 'ESH\_TEST\_MATERIAL', 'USER\_AUTHORITY', 'USER\_AUTHORITY\_DATA', 'USER\_MAPPING', 'USER\_PROFILE', 'ZHHA\_BO', and 'ZTB\_SIMPLE\_SEARCH'.

The import transfers the persistency view of the BO model into the enterprise search model. That means, the database tables are used as a DDIC reference. As the BOPF BO model doesn't contain search specific metadata, the model has to be adjusted in the Enterprise Search modeler. When re-importing the data from BOPF, changes in the enterprise search model are overwritten.

## 1.2 Data extraction

Data extraction is needed to replicate the data to TREX. For that reason, interface /BOBF/IF\_SEA\_BO\_DATA\_EXTRACT needs to be implemented. The default implementation is /BOBF/CL\_SEA\_BO\_DATA\_EXTRACT, which extracts the persistency view of the BO model and fills corresponding nodes and fields of the enterprise search model. Data extraction can be separated into the following parts:

### 1.2.1 Full indexing

For an initial load of the TREX, the full indexing is needed. This step doesn't need to be repeated at a later point in time, as long as there are no inconsistencies (e.g. caused by a wrong implementation of the extraction class).

### 1.2.2 Delta indexing

After full indexing is completed, delta indexing is required only to keep the data on TREX up to date. BOPF notifies enterprise search automatically about BO changes, as soon as BO data is newly persisted. It is therefore important that the enterprise search model name is maintained in the BOPF model, even if the enterprise search model is not created via the import function.

### 1.2.3 Administration

In the Enterprise Search administration cockpit, a connector has to be created for the enterprise search model. Afterwards, the connector has to be scheduled for indexing.

The screenshot shows the SAP Connector Administration Cockpit interface. At the top, there is a header with the SAP logo and the title 'Connector Administration Cockpit'. Below the header, a sub-header reads 'Here you can monitor and administrate search object connectors.' The main area is titled 'Search Object Connectors' and contains a table with columns for Connector Name, Connector ID, Modified, Search, Analytics, and Status. The table lists several connectors, with 'test enterprise search data extraction' highlighted in blue. Below the table, there is a section for 'Details of "test enterprise search data extraction"' with tabs for 'General', 'Schedules', 'Job Log', and 'Categories'. The 'General' tab is active, showing fields for Connector Name, Connector ID, Connector Type, Model Name, and Model.

Connector Name	Connector ID	Modified	Search	Analytics	Status
BCV Dashboard	BFI001~BCV_DASHBOARD~	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Prepared
BCV Query Statistics	BFI001~ZSH_BCV_QUERY_STAT~	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Prepared
ODP Catalog	BFI001~0BS_ANLY_ODP_CONT~	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Prepared
Package	BFI001~0BS_ANLY_DEVC~	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Prepared
Software Component	BFI001~0BS_ANLY_CVERS~	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Prepared
test enterprise search data extraction	BFI001~BOBF_TST_SEA_EXTR~	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Active

**Details of "test enterprise search data extraction"**

General Schedules Job Log Categories

Connector Name: test enterprise search data extraction Edit

Connector ID: BFI001~BOBF\_TST\_SEA\_EXTR~

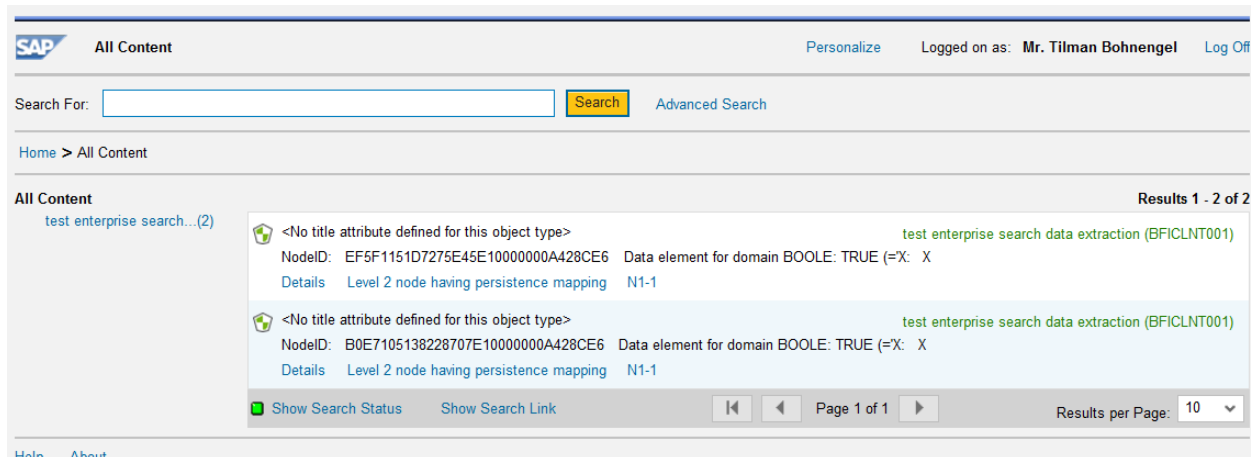
Connector Type: Embedded Search

Model Name: test enterprise search data extraction

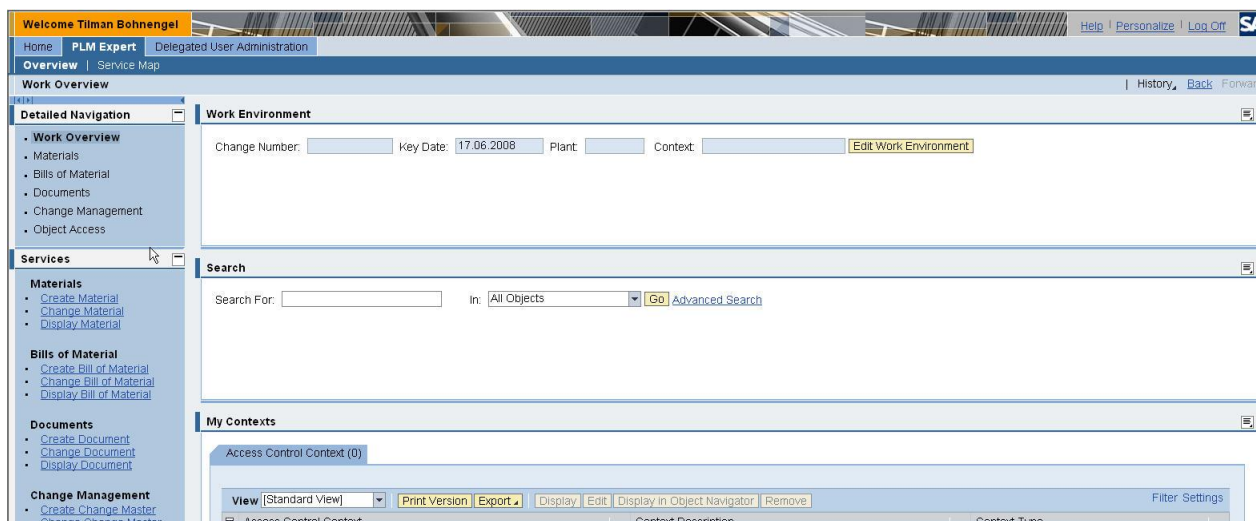
Model: BOBF\_TST\_SEA\_EXTR

### 1.3 Search

The BO is searchable, if its connector has the status Active. Enterprise Search delivers several generic UIs, which can be used to search the BO. For the local system search (embedded search), a web dynpro application exists that can be embedded into other web dynpro applications. It can also be started with transaction ESH\_SEARCH.













It is also possible to use the APIs provided by enterprise search to seamlessly embed search into the applications. The figures below show embedding of enterprise search into PLM solution with an application specific UI.



Full text search across all relevant PLM Objects embedded into PLM control center

## Search

Search For:  In:   [Advanced Search](#)

Search Results			
Results per Page			
10	<input type="button" value="Display Preview"/>	<input type="button" value="Additional Functions"/>	
<input type="checkbox"/> <a href="#">497</a>	Description: BIKE BELT Created By: VOIGT Type Description: Semifinished Product	Changed On: 26.05.2008 Mati Category:	Created On: 23.01.2008 Changed By: SCHNEIDERMAX Prod.Hierarchy: 
<input type="checkbox"/> <a href="#">495</a>	Description: BIKE SEAT Created By: VOIGT Type Description: Semifinished Product	Changed On: 26.05.2008 Mati Category:	Created On: 23.01.2008 Changed By: SCHNEIDERMAX Prod.Hierarchy: 
<input type="checkbox"/> <a href="#">494</a>	Description: BIKE FORK Created By: VOIGT Type Description: Semifinished Product	Changed On: 26.05.2008 Mati Category:	Created On: 23.01.2008 Changed By: SCHNEIDERMAX Prod.Hierarchy: 
<input type="checkbox"/> <a href="#">492</a>	Description: BIKE HANDLEBAR Created By: VOIGT Type Description: Semifinished Product	Changed On: 26.05.2008 Mati Category:	Created On: 23.01.2008 Changed By: SCHNEIDERMAX Prod.Hierarchy: 
<input type="checkbox"/> <a href="#">485</a>	Description: BIKE CRANK Created By: VOIGT Type Description: Semifinished Product	Changed On: 26.05.2008 Mati Category:	Created On: 23.01.2008 Changed By: SCHNEIDERMAX Prod.Hierarchy: 
<input type="checkbox"/> <a href="#">483</a>	Description: BIKE WHEEL SPOKE Created By: VOIGT Type Description: Semifinished Product	Changed On: 26.05.2008 Mati Category:	Created On: 23.01.2008 Changed By: SCHNEIDERMAX Prod.Hierarchy: 
<input type="checkbox"/> <a href="#">482</a>	Description: BIKE WHEEL RIM Created By: VOIGT Type Description: Semifinished Product	Changed On: 26.05.2008 Mati Category:	Created On: 23.01.2008 Changed By: SCHNEIDERMAX Prod.Hierarchy: 
<input type="checkbox"/> <a href="#">481</a>	Description: BIKE WHEEL CENTER Created By: VOIGT Type Description: Semifinished Product	Changed On: 26.05.2008 Mati Category:	Created On: 23.01.2008 Changed By: SCHNEIDERMAX Prod.Hierarchy: 
<input type="checkbox"/> <a href="#">491</a>	Description: BIKE SPROCKET Created By: VOIGT Type Description: Semifinished Product	Changed On: 26.05.2008 Mati Category:	Created On: 23.01.2008 Changed By: SCHNEIDERMAX Prod.Hierarchy: 
<input type="checkbox"/> <a href="#">488</a>	Description: BIKE PEDAL Created By: VOIGT Type Description: Semifinished Product	Changed On: 26.05.2008 Mati Category:	Created On: 23.01.2008 Changed By: SCHNEIDERMAX Prod.Hierarchy: 

Total Number of Hits: 79 Page: 2 Of: 8 << First | < Previous | **Next** >

Result list as a modal dialog window

## 2 Design

All enterprise search relevant objects can be found in package /BOBF/SEARCH.

### 2.1 Data provider

The data provider /BOBF/CL\_SEA\_DATA\_PROVIDER represents the central link between enterprise search and BOPF. It is called by enterprise search for design time actions (import function in the modeler) as well as for the extraction of data. It works also as a factory for:

- The object type provider
- The Enterprise Search metadata descriptor
- The BO data extractor

### 2.2 Metadata descriptor

The abstract metadata descriptor /BOBF/CL\_SEA\_BO\_METADATA contains the general mapping rules between the BOPF BO and the Enterprise Search Model. Besides that, the class /BOBF/CL\_SEA\_BO\_METADATA\_PERS, which inherits from that, adds the details, which are specific to the persistency view of the BO model. These are

- The data base table on the node as DDIC reference
- The association binding of compositions

## 2.3 Writer of change pointers

Writing of change pointers is necessary for delta indexing in order to keep the TREX indexes up to date.

The class /BOBF/CL\_SEA\_CHANGE\_POINTER notifies changes during the save phase to enterprise search. It is realized as a plugin of the transaction manager. In case of synchronous update, the plugin directly calls method CL\_ESH\_IF\_DATA\_PROVIDER\_TOOLS=>SET\_CHANGEPOINTERS. In case of asynchronous update, the function module /BOBF/SEA\_SAVE\_CPOINTERS is called in update task in order to do that job. Changes are aggregated on BO level, as data extraction is always executed for complete BOs. This ensures consistency at least within the BO. Change pointers are only written for BOs, which are marked as enterprise search enabled.

## 2.4 BO data extractor

The BO data extractor /BOBF/CL\_SEA\_BO\_DATA\_EXTRACT is implemented to generically extract the persistency view of a business object. That means, the enterprise search model must also reflect the persistency view, if this extractor should be used. The class is the default implementation of interface /BOBF/IF\_SEA\_BO\_DATA\_EXTRACT. Another class can be defined in the BO configuration. Full indexing is done in packages. One call of method ESH\_IF\_OBJECT\_ITERATOR~NEXT extracts one package during the full indexing, whereas IF\_ESH\_IF\_OBJECT\_DATA\_BY\_ID~GET\_DATA retrieves the BO data for a list of keys during the delta indexing.

## 3 Further hints and sources of information

### 3.1 Important transactions

ESH_MODELER	Design time
ESH_COCKPIT	Connector Administration
ESH_SEARCH	Generic search UI
TREX_ADMIN	Trex administration

### 3.2 Wikis

Enterprise Search @ Business Suite:

<https://wiki.wdf.sap.corp/wiki/display/BSExcellence/Enterprise+Search>

NW Enterprise Search

<https://wiki.wdf.sap.corp/wiki/display/EPSEARCH/Home>

TREX

[http://trexweb.wdf.sap.corp:1080/wiki/index.php/Main\\_Page](http://trexweb.wdf.sap.corp:1080/wiki/index.php/Main_Page)