

Organizing Translations in SAP Implementation Roll-out Projects



Applies to:

The methodology and various translation approaches discussed in this document applies to all SAP applications based on ABAP.

For more information, visit the [ABAP homepage](#).

Summary

This document describes the different tools available for carrying out translation. This document will also consider the various approaches available for translation during multi-country Roll-out projects. This document also describes the procedure to set-up the translation environment which is a prerequisite for carrying out translation using Worklists.

Author: Prasanna Rangaraju

Company: SAP Global Delivery, SAP India

Created on: 31 March 2009

Author Bio



Prasanna Rangaraju is working as SRM Consultant in SAP Global Delivery, Bangalore. He has around 6 years of experience. He has undertaken projects for customers based in Sweden, France and US. His experience also includes working as a Solution Management lead in SAP Labs India.

Table of Contents

Overview	4
Purpose of Document	4
Scope of the document	4
Introduction	5
Transaction SE63: Translation Initial screen	5
Transaction LXE_MASTER / SLWA – Translation Environment Admin.....	6
Language Availability of SAP Applications	7
Overview of Translation	8
Development Objects.....	8
SAPscript Objects	9
Customizing Data.....	9
Master and Transaction Data.....	10
Prerequisites Before Starting the Translation	10
Language Supplementation	11
Different Approaches for Translation.....	12
Object Translation in Corresponding Development Transactions	12
Individual Object translation.....	13
Procedure to translate single development object in SE63.....	13
Procedure to translate customizing entries	15
Procedure to translate Master data	17
Translation Using Worklists	18
On-the Fly Translation.....	18
Translation Using Worklists	19
Setting up the Translation Environment – Transaction LXE_MASTER.....	20
Step 1: Define languages into which translation needs to be carried out.....	20
Step 2: Create translation graphs	21
Step 3: Define the client(s) in which translations need to be done.....	22
Step 4: Define the object types that each target language is to translate	23
Step 5: Assign collections to translation graphs.....	24
Step 6: Create or Activate translator profiles.....	25
Step 7: Create translators	26
Step 8: Run an evaluation to create worklists and statistics in the system	27
Access the Worklist in SE63 and Perform the Translation	31
Step 1: Check the default settings in SE63 transaction	31
Step 2: Open the worklist using menu Standard → Worklist.....	32
Step 3: Choose the Worklist number and navigate to next screen.....	32
Step 4: Load the Worklist using “Object List” tab page	33
Step 5: Carry out translation for all objects in Worklist.....	34
Advantages of Translation Using Worklists	35
Appendix and Recommendations.....	36
Recommendation	36
Supplementation	36

Related Content..... 38
Copyright..... 39

Overview

Purpose of Document

This document discusses the various approaches and methodologies available for carrying out translations effectively during Roll-out projects. In this document all the four methodologies available for translation of objects in SAP Web AS system are compared with each other.

Scope of the document

The translation procedure described in this document can be applied for translating any SAP application running on SAP Web AS ABAP for ex: ERP, SCM, SRM, CRM etc. The various approaches discussed here are applicable for all ABAP-objects which are relevant for translation.

This document deals with the basic set-up of the translation environment. It is suggested to refer the provided online help links to know more about the various options available for setting up and carrying out translations. Some of the advanced topics of translations are out-of the scope of this document.

The following are out-of the scope of this document:

- Translations in BI specific objects
- Translation in Enterprise Portal – Iviews/Roles etc.,
- Translation of applications developed in Web AS Java
- Authorizations required to carry out the set-up and translations explained in this document
- Any non-ABAP translations – for ex: translations for developments done in Visual composer, Composite application framework (CAF), Guided procedures (GP) etc.,

Introduction

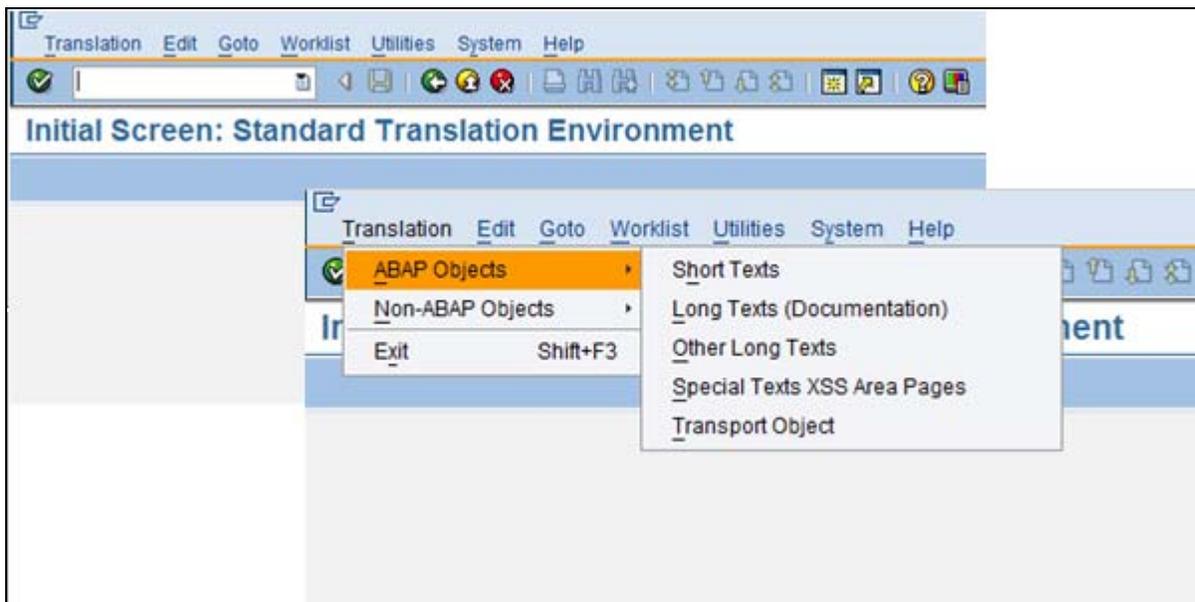
In general SAP implementation projects involve extensive custom developments. All the developments will obviously involve language specific texts for the user interface. During the roll-out of the solution to a new country, all these texts will have to be translated to the local language.

Translation is one of the important phases of the Localization and in case of roll-outs to multiple countries; translation itself will be another big and challenging project. In this document, the different possibilities and approaches to efficiently translate the SAP system during roll-outs and localization are discussed. The standard SAP provided translation infrastructure will be used to carry out the translation.

The standard SAP system provides two transactions as part of translation infrastructure:

Transaction SE63: Translation Initial screen

Transaction SE63 enables to access the translation tools. Individual objects can be accessed for translation from the initial screen of SE63.



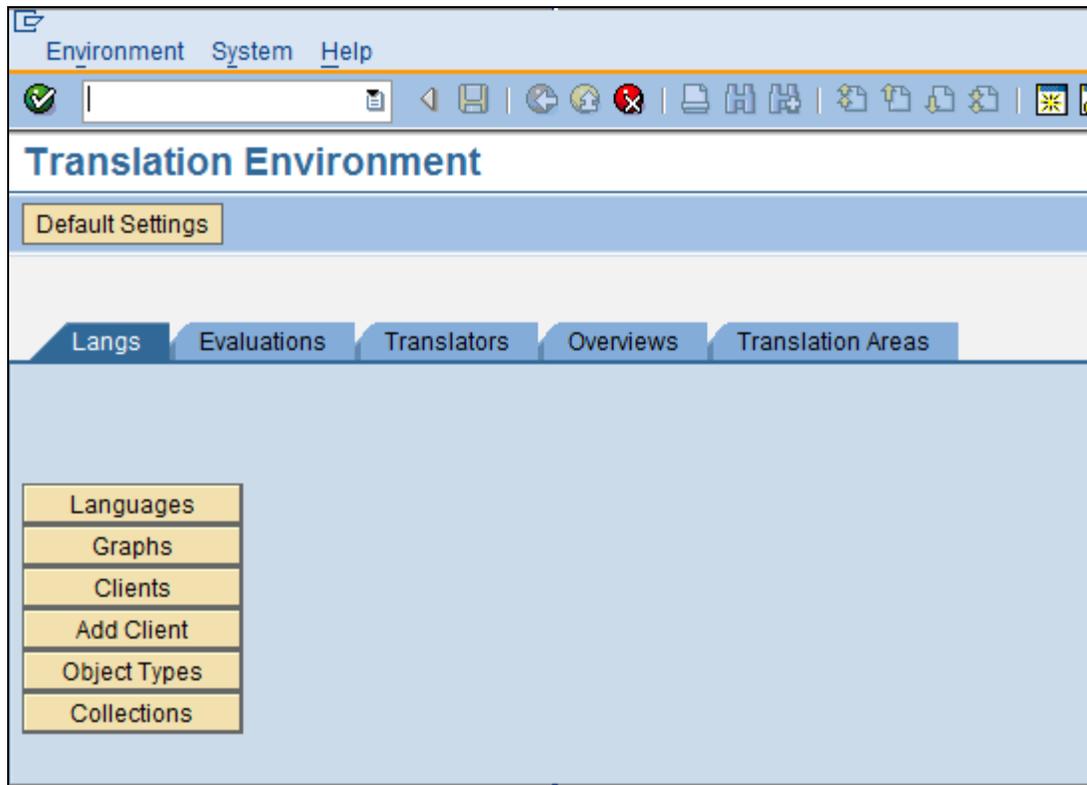
Transaction SE63 enables to perform the following activities:

- Call up individual objects directly for translation
- Call up a worklist to find and translate objects that need translating in the collections assigned to translator. The prerequisite is that the translation environment has been set up already.
- Define settings to suit the way translator want to work with the translation tools

Transaction LXE_MASTER / SLWA – Translation Environment Admin

Before translation can start in an SAP system, the translation environment in accordance with customer's translation requirements has to be set-up in transaction LXE_MASTER. This transaction allows creating [worklists](#) and [statistics](#) for each target language into which translation has to be carried out. The translation-relevant objects are then translated via worklists which can be assigned to translators.

The objects to be translated can be distributed among one or more translators and for each target language. Statistics can be generated through which monitoring the progress of the translation project can be carried out.



Language Availability of SAP Applications

[Product Availability matrix](#) in SAP Service Marketplace provides more detailed information on the languages supported by every SAP application.

For ex: the below screenshot shows the languages supported by SAP ERP 6.0:

General data		Languages		Countries		Database Platforms		Operating System	
SAP ERP 6.0									
Print...		 Help							
SAP ECC Server		SAP Application Platform - IPC			SAP XSS (Self Services)				
Language	Status (Internal)	Date (Internal)	Status (External)	Date (External)	System	Platform	Database	OS	Kernel
Arabic	Released	24.10.2005	Released	24.10.2005					
Bulgarian	Released	24.10.2005	Released	24.10.2005					
Catalan	Released	06.11.2006	Released	06.11.2006					
Chinese	Released	24.10.2005	Released	24.10.2005					
Chinese trad.	Released	24.10.2005	Released	24.10.2005					
Croatian	Released	24.10.2005	Released	24.10.2005					
Czech	Released	24.10.2005	Released	24.10.2005					
Danish	Released	24.10.2005	Released	24.10.2005					
Dutch	Released	24.10.2005	Released	24.10.2005					
English	Released	24.10.2005	Released	24.10.2005					
Estonian	Planned	02.03.2009	Planned	02.03.2009					
Finnish	Released	24.10.2005	Released	24.10.2005					
French	Released	24.10.2005	Released	24.10.2005					
German	Released	24.10.2005	Released	24.10.2005					
Greek	Released	24.10.2005	Released	24.10.2005					
Hebrew	Released	24.10.2005	Released	24.10.2005					
Hungarian	Released	24.10.2005	Released	24.10.2005					
Italian	Released	24.10.2005	Released	24.10.2005					

Overview of Translation

Localization or Roll-out projects involve the below phases:

1. Adapting to the Legal Requirements of the local country
2. Adapting the Local Best Business Practices
3. Translation

Translation is one the major and important phase during the Roll-out project where all the language dependent texts have to be translated. Broadly the objects which need translation can be classified as following:

Development Objects

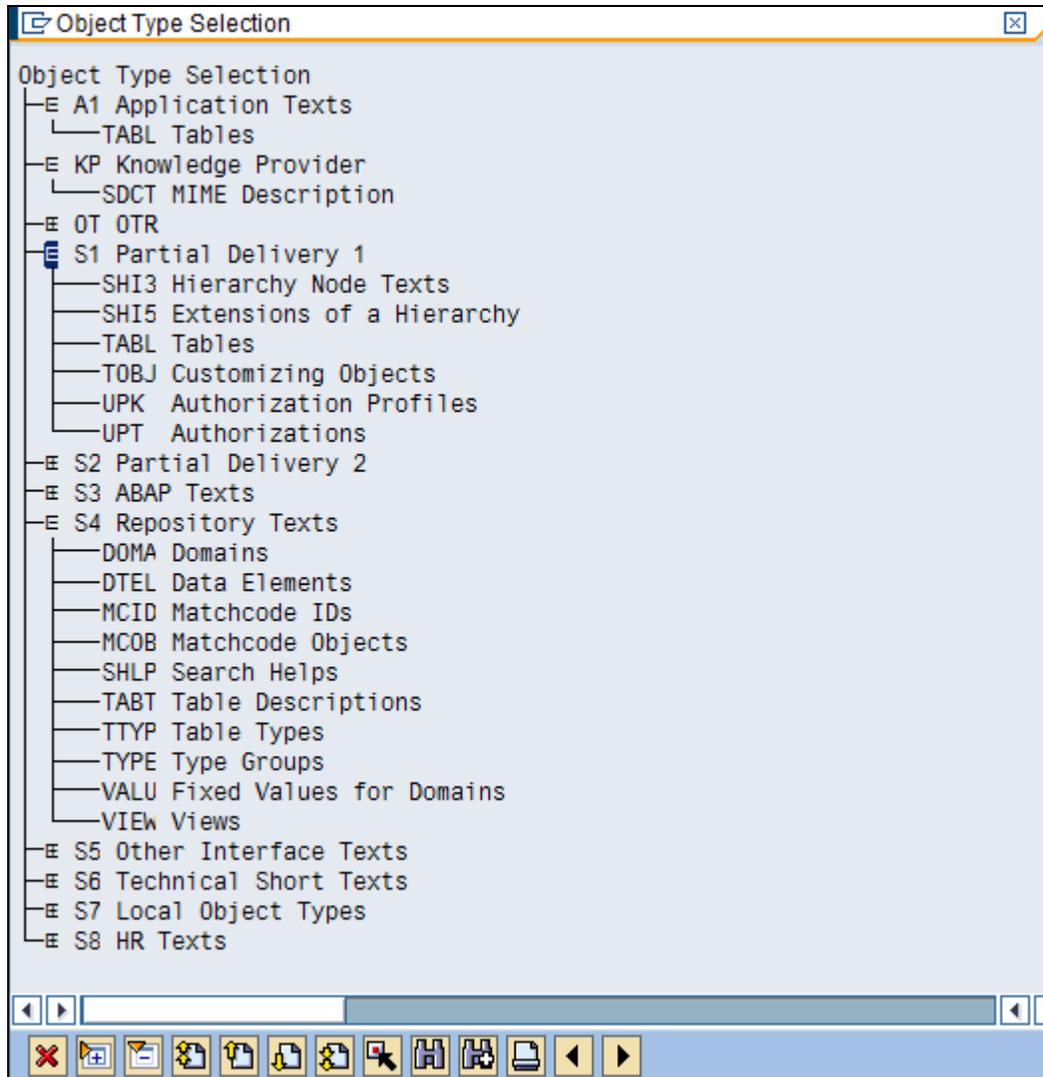
Development objects are primarily language-independent. However their assigned text elements (e.g. report text elements, messages, selection texts, short text description of repository objects) are language-dependent. When they are created, development object texts are assigned in the user's logon language ("source language"). Every additional text change can only be done either in the original language or in the logon language of the current logon session. In order to avoid confusion, a single language should be used as the original language for development objects.

The following list gives some of the most important translations that can be carried out on repository objects:

1. Domains and Fixed values at domain
2. Data elements
3. Messages (SE91 messages)
4. Transactions
5. HTML IAC Resource parameters
6. Screen elements (Fields, buttons etc)
7. Report texts, attributes, text symbols
8. Workflow and work item texts
9. Search helps
10. Tables, Structure, Views, Table types
11. Repository objects
12. Webdynpro text elements

Note that the above list is just an indicative list and not the comprehensive list. Please refer to transaction SE63 to get an overview of all objects that can be translated.

The following screenshot provides a glimpse of all short texts that can be translated using SE63:



SAPscript Objects

For SAPscript objects, the same recommendations apply as mentioned above. However, SAPscript objects can be maintained in languages other than the user's logon language. Regarding translation of a SAPscript object, please consider that - for example - changes in text lengths may require a redesign of the layout of the complete SAPscript form.

Customizing Data

Customizing data can be translated via the translation functionality within SM30 (specifying the translation target language) or in the corresponding IMG with the appropriate logon language.

Master and Transaction Data

Master and business data can be maintained manually via the appropriate application transaction using the translation target language as logon language. For large amounts of data, however, this is not a feasible procedure. A system-wide scan for text elements will be available from which a translation worklist can be created.

For ex: the descriptions for Products, Commodity codes, Product categories, Material Master, Material group etc., and all customer specific master data in custom tables belong to this category.

Prerequisites Before Starting the Translation

Before the translation project can be started:

- The target language should be installed and configured using SMLT transaction on the system so that the language is ready to use. This ensures that logon in, and translation into, these target language is possible. See notes 42305 and 73606 for more information.
- Import the translations that are available on the language CDs delivered by SAP for the software components available in for the system. This prevents the unnecessary translation of SAP objects, for which SAP has already delivered a translation, so that translations for the customer specific objects and data can be concentrated. This takes place using the *Language Management* functionality in transaction SMLT. For more information, see [Language Imports Using Transaction SMLT](#).
- If translation was already carried out in another system, it is recommended to reuse the [proposal pools](#) (translation memories) that were created during translation, as this will save the translation time and ensure consistent translations throughout systems.

When the new language is installed the translations for following will be made automatically available for all the standard SAP objects:

- Repository objects (user interface, error messages, F1 help, IMG)
- Standard Customizing data
- Language supplementing of client-independent texts (client 000)
- Language import into logon client (client-dependent Customizing data)
- Supplementing client-dependent Customizing data

Language Supplementation

English and German are the only two languages that are complete in every SAP system. In general, any other languages are not completely translated. This means that at least some of the texts do not exist in these other languages. Since missing texts can sometimes cause processes to crash, they need to be replaced in some way. The language supplementation function in the language transport tools enables to do this.

The language supplementation function enables to replace texts that do not exist in the target language with another language. This language is known as the *supplementation language*. It is always recommended to use one of the two standard languages (English or German) as the supplementation language. After the supplementation is carried out for the new target language, the text is available in the supplementation language (such as English), but under the language key of the target language (such as French – FR).

Language supplementing is used to copy language-dependent texts from a source language to a target language. This may be useful for user interface text elements in a language for which SAP does not provide a complete translation. Via transaction SMLT, the source and target languages of the supplement can be specified. Details can be found in SAP Note 111750. In the same transaction the supplement jobs can be scheduled. For more details, please refer to [Language supplementation](#) in online help.

Different Approaches for Translation

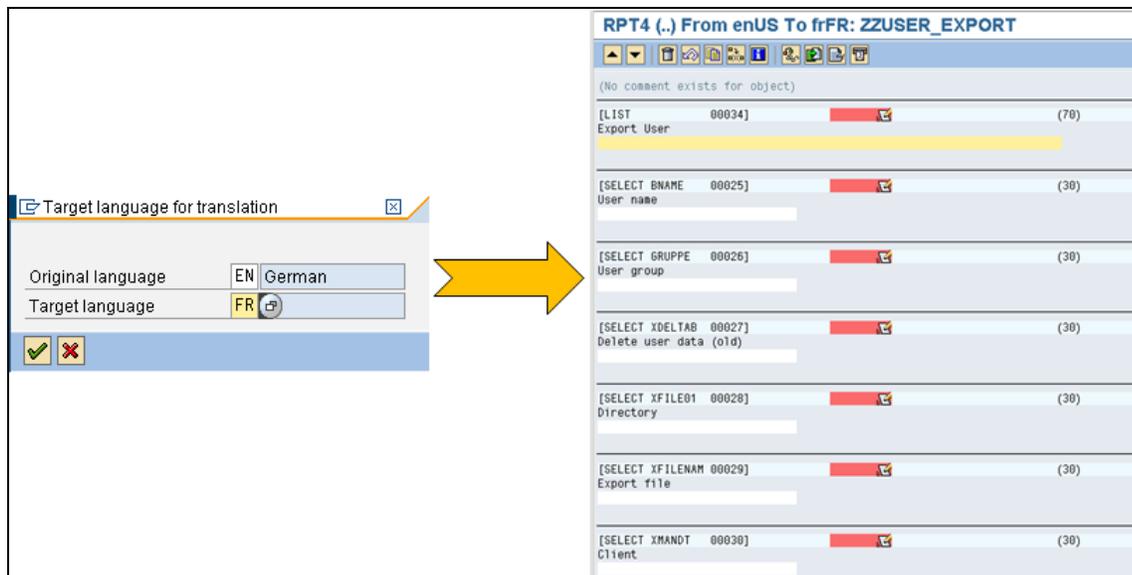
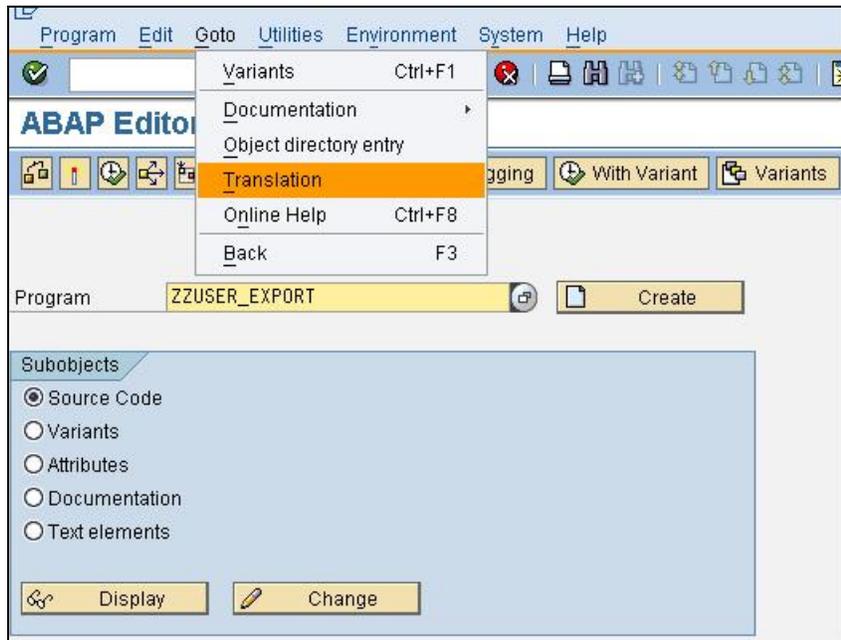
There are four ways of translating objects/texts in an SAP system as discussed below:

Object Translation in Corresponding Development Transactions

Objects developed in ABAP can be translated directly after they are created them by choosing **Goto → Translation** from the respective development transaction. For ex: SE38 for reports or SE37 for function modules etc.

This option is applicable during the initial development phase. But this methodology is too cumbersome and expensive when the system is already live for some countries and Roll-out is planned for any new countries as each and every object have to be translated individually navigating through their corresponding transactions. Hence this approach is best suited during the time of development.

The following screenshots provide an example of how to carry out translation using this approach:



Individual Object translation

Individual objects for translation can be directly translated in transaction SE63. For ex: Call up SE63 for individual data elements, domains, fixed values at domains, screen elements, buttons and messages.

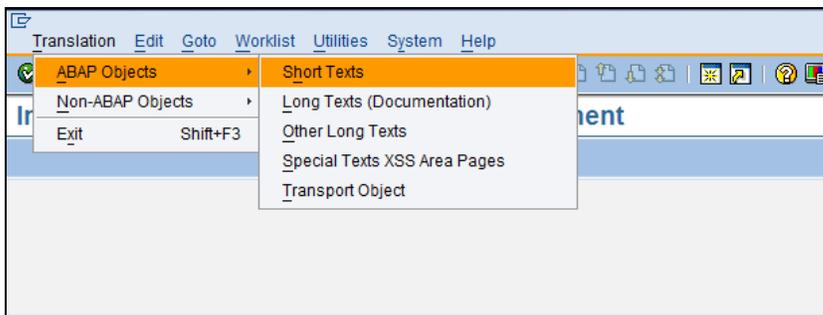
The prerequisite is that the translator must know the object type, and the technical name of the object to carry out translation. If the number of objects to be translated is too large, then this methodology of translation during the roll-out project will become inefficient and impractical. Hence this methodology can be used to translate objects individually on a case-to-case basis.

For more information, see [Translating Objects Directly](#).

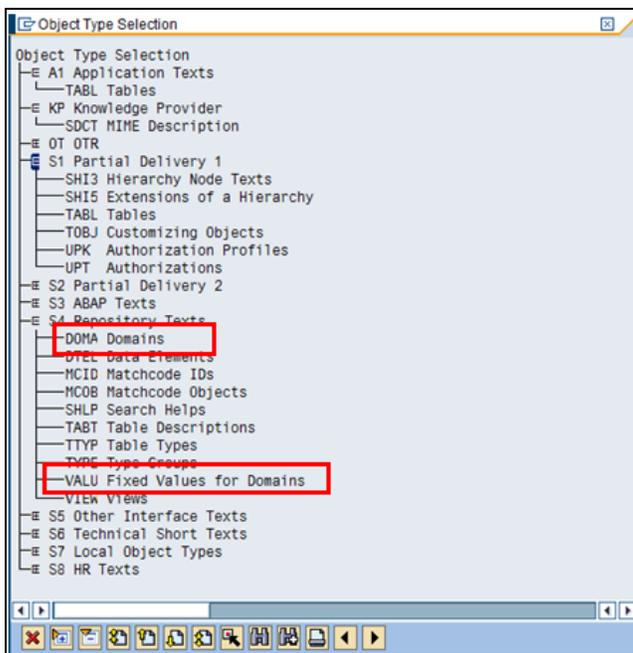
Procedure to translate single development object in SE63

The following procedure describes how translate individual objects can be carried out - for ex: domain/fixed values at domain in SE63. The same procedure can be used to translate all other texts.

1. Execute SE63 and choose "Short Texts" as shown below:



2. Now choose the object to be translated (for ex: domain/data element/Fixed values at domain etc..)



- In the next screen enter the object to be translated, choose the source and target language and then click on "Edit" to carry out translation

ABAP Short Texts: Domains

Object
 Object Name: ZCATALOGNAME
 Translation Area: 999999 | ABAP-System

Translation
 Source Language: enUS | English US
 Target language: frFR | French FR

DOMA (..) From enUS To frFR: ZCATALOGNAME

(No comment exists for object)

[DDTEXT 00001]	Catalog name	(60)
	Nom de Catalogue	

Another example to translate fixed values at domain is as shown below:

ABAP Short Texts: Fixed Values for Domains

Object
 Object Name: ZOPERATOR
 Translation Area: 999999 | ABAP-System

Translation
 Source Language: enUS | English US
 Target language: frFR | French FR

VALU (..) From enUS To frFR: ZOPERATOR

(No comment exists for object)

[DDTEXT 00001]	Equal	(60)
[DDTEXT 00002]	Not Equal	(60)
[DDTEXT 00003]	Contains Pattern	(60)
[DDTEXT 00004]	Does not Contain Pattern	(60)
[DDTEXT 00005]	Greater	(60)
[DDTEXT 00006]	Less	(60)

Procedure to translate customizing entries

1. Open the view either in SM30 or navigate through SPRO IMG Path

Document Types Purchase order Change

Type	Doc. Type	Doc. Type Descript.	Itmint	NoRgeint	NoRge Ext	NoRgeALE	Updat.	FieldSel.
DP25	Electronic commerce		10	45	25		SAP	NBF
DP26	Electronic commerce		10	45	26		SAP	NBF
DP27	Electronic commerce		10	45	27		SAP	NBF
DP28	Electronic commerce		10	45	28		SAP	NBF
DP29	Electronic commerce		10	45	29		SAP	NBF
DP30	Electronic commerce		10	45	30		SAP	NBF
EC	Electronic commerce		10	45	PO		SAP	NBF
ECDP	Electronic commerce		10	45	DM		SAP	NBF
ENB	Standard PO DFPS		10	45	41		SAP	NBF
EUB	Stock Transp. Order		10	45	41		SAP	UBF
F0	Framework order		10	45	41		SAP	FOF
GNB	GTS Standard PO(old)		10	45	41		SAP	NBF
GST	GTS Stock Tr PO(old)		10	45	41		SAP	UBF
IN	Internet commerce		10	45	41		SAP	NBF
NB	Standard PO		10	45	41		SAP	NBF
NB2	Standard PO		10	45	41		SAP	NBF
NB2C	Standard PO		10	45	41		SAP	NBF2
UB	Stock transport ord.		10	45	41		SAP	UBF
UB2	Stock Transp. Order		10	45	41		SAP	UBF2
ZGT1	IDES - GTS PO		10	45	41		SAP	NBF
ZNB	Test for PO Release		10	45			SAP	NBF
ZS	SNC PO		10	45	41		SAP	NBF

Now navigate through the menu path Goto → Translation

Table View Edit Goto Selection Utilities System Help

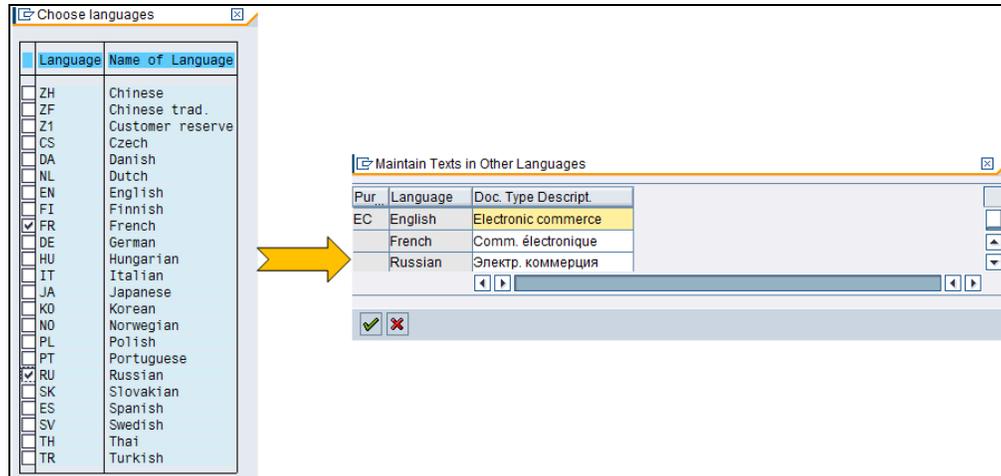
Document Types Purchase order Change

Details

- Next Entry
- Previous Entry
- Position Cursor... Shift+F7
- Translation**
- Address
- Form
- Back F3

Type	Doc. Type	Doc. Type Descript.	Itmint	NoRgeint	NoRge Ext	NoRgeALE	Updat.	FieldSel.
DP25	Electronic commerce		10	45	25		SAP	NBF
DP26	Electronic commerce		10	45	26		SAP	NBF
DP27	Electronic commerce		10	45	27		SAP	NBF
DP28	Electronic commerce		10	45	28		SAP	NBF
DP29	Electronic commerce		10	45	29		SAP	NBF
DP30	Electronic commerce		10	45	30		SAP	NBF
EC	Electronic commerce		10	45	PO		SAP	NBF
ECDP	Electronic commerce		10	45	DM		SAP	NBF
ENB	Standard PO DFPS		10	45	41		SAP	NBF
EUB	Stock Transp. Order		10	45	41		SAP	UBF
F0	Framework order		10	45	41		SAP	FOF
GNB	GTS Standard PO(old)		10	45	41		SAP	NBF
GST	GTS Stock Tr PO(old)		10	45	41		SAP	UBF
IN	Internet commerce		10	45	41		SAP	NBF

2. Choose the target languages and carry out the translation:



Procedure to translate Master data

Master data will be translated in the corresponding master data transactions. An example of how to maintain the language specific data for the Product (Part) in a CRM is as shown below. In the same way all other master data can be translated in the corresponding master data maintenance transactions (for ex: MM02 for materials etc.)

The screenshot displays the SAP CRM Master Data Maintenance interface for a Product (Part). The interface is divided into several sections:

- General Data:** Material: MATLOCAL1, Description: BlackBox.
- Product Categories:** A table with columns: Category ID, Hierarchy ID, Category Description.
- Product Status:** Status, Active/Inact, To archive, Locked, Deleted.
- Descriptions:** A table with columns: Language, Product Description. This section is highlighted with a red border.
- Product Taxes:** A table with columns: Cou..., Region, Tax Type, Tax Group, Tax Scale Code.

Category ID	Hierarchy ID	Category Description
MAT_ROH	R3PRODSTYP	Raw material
LOCAL1	R3MATCLASS	Office Material (L)

Language	Product Description
English	BlackBox
German	BlackBox
French	

Translation Using Worklists

This option is recommended for large translation projects where a large number of objects need to be translated into multiple languages. For example, over a thousand objects need to be translated into 3 different languages.

In this approach [worklists](#) and [statistics](#) for each target language into which translation will be done are created. The translation-relevant objects are then translated via worklists. Objects to be translated can be divided among more than one translator for each target language. Statistics are generated so that translation progress can be monitored.

The prerequisite for this approach is to set up the translation environment in transaction LXE_MASTER. In the next section of this document the step-by-step procedure to set-up the environment is described. For more detailed and additional information, see [Setting up the Translation Environment](#).

The first two options discussed above do not require setting up the translation environment, but fewer translation functions are available as a result. Translation using worklist requires the set up of the translation environment in transaction LXE_MASTER, but opens up a wide range of functionality to enable to translate efficiently and consistently.

On-the Fly Translation

In this approach all or some objects in a transport request can be called up in a temporary worklist to translate them quickly and without having to set up the translation environment. For more information, see [On-The-Fly Translation](#).

This approach is a variant of the above approach (Translation using Worklist) but with the exception that the worklist itself is not saved and it is generated at run-time.

In case there is no time to set up the translation environment, create translation graphs, create translators and assign collections to translators, and run regular evaluations, “on-the-fly” translation functions in transaction SE63 can be used for translation though all the functionality of translation using worklist are not available here.

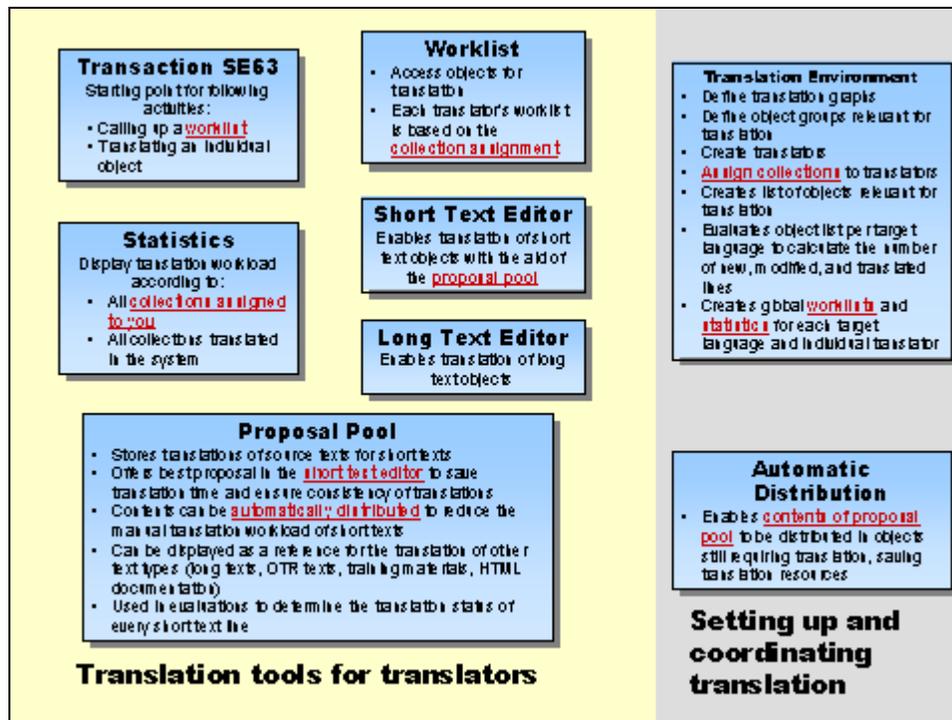
Translation Using Worklists

This section describes in detail the steps required to set-up the system in the LXE_MASTER translation administrator transaction (as an administrator) and the steps to load and translate the worklist (as a translator).

The following links and documents can be used as reference for setting up and performing translation using this approach:

1. [Working with Collections in LXE_MASTER and SLLS](#)
2. [Translation Tools for Translators \(BC-DOC-TTL\)](#)
3. [Setting Up and Coordinating Translation \(BC-DOC-TTL\)](#)
4. [Translation Tools for Coordinators](#)
5. [Translation Tools for Translators](#)

The graphic below lists the functions in the translation tools, and shows how they interact with each other:



Setting up the Translation Environment – Transaction LXE_MASTER

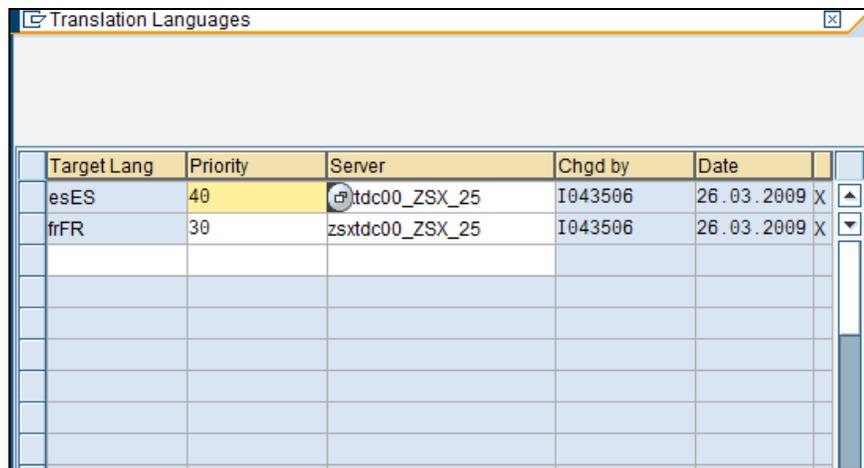
Before translation can start in an SAP system, the translation environment has to be defined in accordance with customer's translation requirements. The following activities need to be performed in the following sequence:

Step 1: Define languages into which translation needs to be carried out

This is the first step of defining the translation environment in transaction LXE_MASTER where the languages into which translation has to be carried out are defined. The prerequisite is that the target language must be installed and configured.

In the case of languages where more than one option is offered in the hierarchy (for example, for English, both *enGB* and *enUS* are available), double-click on the two-character language ID (in this case *en*), which is the top node for the target language in the hierarchy. This ensures that the system selects the correct version of the target language, into which translation of ABAP objects is possible.

For more information please refer to [Defining Target Languages](#)



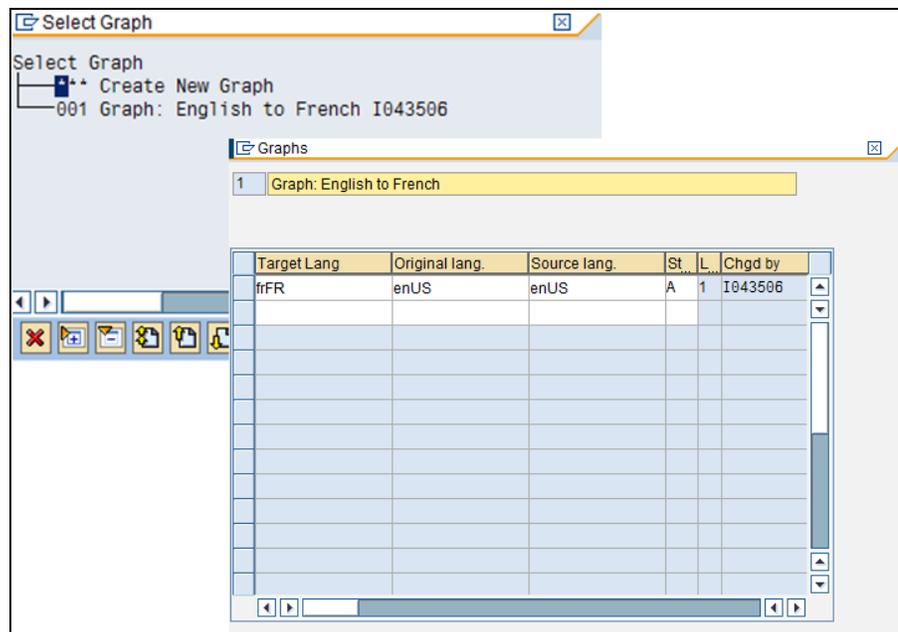
Target Lang	Priority	Server	Chgd by	Date	
esES	40	itdc00_ZSX_25	I043506	26.03.2009	X
frFR	30	zsxtdc00_ZSX_25	I043506	26.03.2009	X

Step 2: Create translation graphs

In this second step the aim is to define the source language for each target language and group together collections according to translation relevance. The following table gives additional information about the terminologies used in this transaction.

Term	Definition
Target language	The language <i>into</i> which translation is done
Source language	The language <i>from</i> which translation is done
Original language	The language in which a developer created an object. In other words, the development language. It is possible to have more than one development language (for example, both German developers and developers who develop in English are working in a project).

Translation graphs define the translation directions for a group of collections (packages in the development environment). For more information please refer to [Defining Translation Graphs](#)



Step 4: Define the object types that each target language is to translate

Every object created in an SAP system has an *object type* (for example, data element and message). In the translation environment, each object type is assigned to an *object group*. An object group collects object types of a similar translation priority.

Object types and object groups enable to define the *scope* of translation for each target language. For more information please refer to [Defining Object Types for Translation](#)

The screenshot shows the 'Object Types' configuration window in SAP. The 'Target Language' pane is set to 'frFR'. The main table lists object types and their distribution settings for the 'frFR' language.

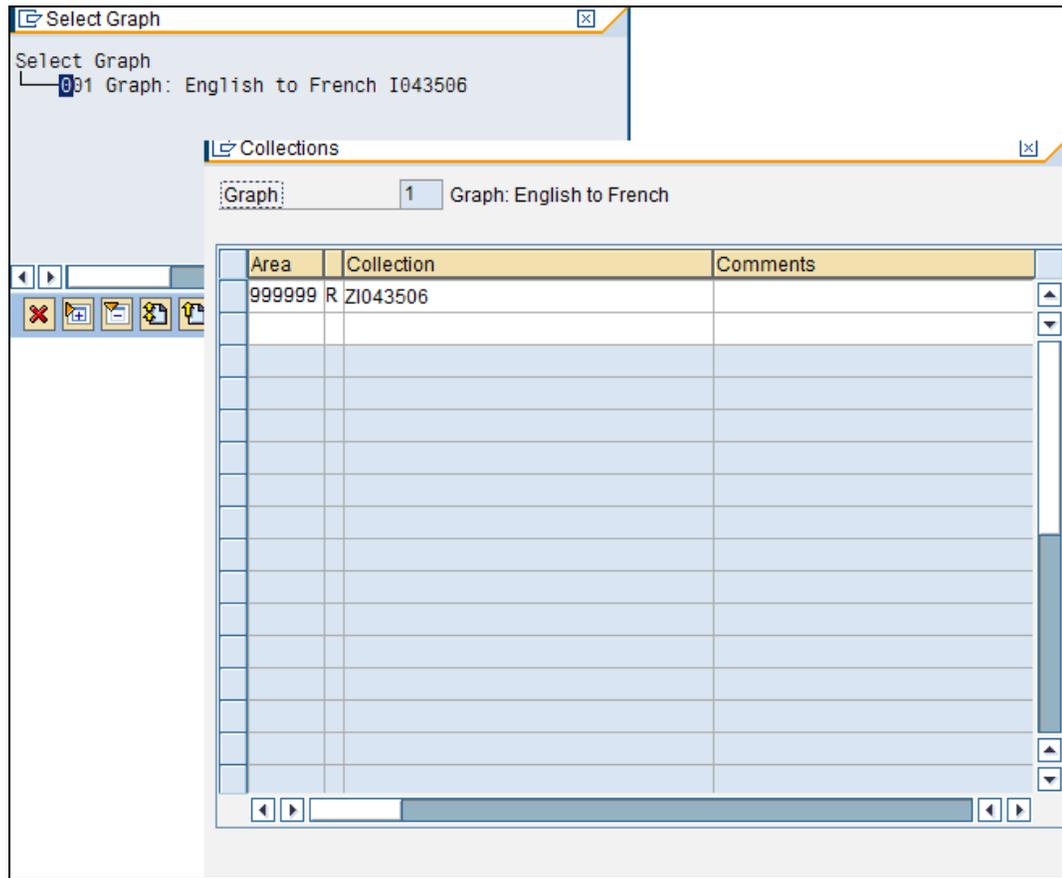
Obj. Type	Dist. New	Dist. Mod	Dist. Tra	Chgd by	Date	Lang
ACGR				I043506	26.03.2009	frFR
BEZD				I043506	26.03.2009	frFR
BSP				I043506	26.03.2009	frFR
BSP1				I043506	26.03.2009	frFR
BSP2				I043506	26.03.2009	frFR
CA1				I043506	26.03.2009	frFR
CA2				I043506	26.03.2009	frFR
CA3				I043506	26.03.2009	frFR
CA4				I043506	26.03.2009	frFR
CA5				I043506	26.03.2009	frFR
CA7				I043506	26.03.2009	frFR
CAD1				I043506	26.03.2009	frFR
CAD2				I043506	26.03.2009	frFR
CAD3				I043506	26.03.2009	frFR
CAD4				I043506	26.03.2009	frFR
CAD5				I043506	26.03.2009	frFR

The bottom toolbar contains the following options: Object Types, Dist. New, No Distr. New, Dist. Mod, No Distr. Mod, Distribution in Translated Lines, and No Distribution in Translated Lines.

Step 5: Assign collections to translation graphs.

After defining a translation graph, collections have to be assigned to it. Only collections that are assigned to an active graph arrive in the worklists and statistics for the target languages defined in the graph. A collection can only be assigned to one translation graph. If a collection that is already assigned to another graph is selected, it will be removed from that graph and jump to this one.

For more information please refer to [Assigning Collections to a Translation Graph](#)



Step 6: Create or Activate translator profiles

A profile has to be assigned to every translator and coordinators created in the system. Each translator has one profile per target language. Translators cannot be created until translator profiles are available.

Either customer specific translator profiles can be created or SAP sample profiles can be activated which can then be modified in accordance with customer specific requirements, if necessary. Translator profiles control the activities that each translator / coordinator can perform for one or more or all target languages in the translation environment. For more information please refer to [Working with Translator Profiles](#)

Aut.	Val.	Chgd.
AVML	Multiple Call of Worklists	
AVUP	Worklist Status	
COGA	Graph Assignment to Collections	
COSE	Collection Selection	
GRAP	Graphs	
INFO	Information Display	
LACL	Client Assignment to Translation Language	
LANG	Translation Languages	
LA0B	Maintain Object Type Assignments / Display	
LASS	Secondary Language Status	
OBJL	Object Lists	
PPD0	Free Domains	
PPQ1	Assign Quality Status	
PPQ2	Distribute Quality Status	
PREL	Settings for Language	

Step 8: Run an evaluation to create worklists and statistics in the system

Most objects requiring translation are assigned to development classes. When worklist is created, one or more development classes can be included. All the objects belonging to the development classes are then included in the worklist.

Evaluation run evaluates all the objects in these development classes which are relevant for translation. This step involves two sub-steps as described below:

Step 8.1: Creating and activating Object list

An object list should contain exactly those objects that need to be translated. When object list is created, parameters are defined to ensure that exactly those objects relevant for translation are included in the object list. If development is ongoing, and objects are still being created in / transported to the translation system, new object list has to be created at regular intervals (ideally once a week until the development deadline). For more information, see [Creating Object Lists](#).

The screenshot shows two SAP dialog boxes. The top one is 'Object Lists' and the bottom one is 'Parameters'.

Object Lists Dialog:

Object List	Object List - 0001	Object List - 0001	Object List - 0001
1	N X	Object List - 0001	I043506 26.03.2006

Parameters Dialog:

Object List: 1 | Object List - 0001

Renew Domain Assignment

Global

Object Types	10 Entries
Collections	1 Entries

ABAP System

Transports	No Entries
Transport Objects	No Entries

Local Lists

Local Lists	No Entries
-------------	------------

Object List Merge

Object Lists	No Entries
Object Types	No Entries
Collections	No Entries
Object Name	*

Buttons: Parameters, Activate

Activation of the above created object list generates a batch job:

Job overview from: 26.03.2009 at: : : : to: 26.03.2009 at: : : : Selected job names: * Selected user names: I043506							
<input type="checkbox"/> Scheduled <input checked="" type="checkbox"/> Released <input checked="" type="checkbox"/> Ready <input checked="" type="checkbox"/> Active <input checked="" type="checkbox"/> Finished <input checked="" type="checkbox"/> Canceled <input type="checkbox"/> Event controlled Event ID: <input type="checkbox"/> ABAP program Program name :							
Job	Ln	Job CreatedB	Status	Start date	Start time	Duration(sec.)	Delay (sec.)
<input checked="" type="checkbox"/> OBJLIST_00001		I043506	Finished	26.03.2009	12:59:35	158	0
*Summary						158	0

The job log shows the actual objects gathered which are relevant for translation:

Graphical display of spool request 24926 in system ZSX				
26.03.2009	Generate Object List			1
00001			26.03.2009 13:00:35	00:00:00
SYSTEM			26.03.2009 13:00:35	00:00:00
999999			26.03.2009 13:00:35	00:00:00
	DOMA	1	26.03.2009 13:01:16	00:00:41
	VALU	1	26.03.2009 13:02:13	00:00:57
MERGE *			26.03.2009 13:02:13	00:00:00
LOCAL TRANSPORT			26.03.2009 13:02:13	00:00:00
TRANSPORTS			26.03.2009 13:02:13	00:00:00
LOCAL-LISTS			26.03.2009 13:02:13	00:00:00

Step 8.2: Creating and activating Evaluation

After an object list is created, an [evaluation](#) analyzes each object and calculates the translation status of each line in each object for each target language. The result of the evaluation is worklists and statistics. If development is ongoing, and objects are still being created in / transported to the translation system, new evaluations have to be created in regular intervals (ideally once a week until the development deadline). This adds new and modified objects to the worklists and updates the statistics. For more information, see [Creating Evaluations](#).

The screenshot displays the SAP Evaluation configuration interface. The main window shows a table with the following data:

Num.	S	Description	Creator	
1		X Evaluation	I043506	26.03.2009

The 'Parameters' dialog is open, showing the following settings:

- Evaluation: 1
- Worklist Number: 1 (Worklit - 0001)
- Distribution:
- Initial Statistics:
- Statistics:
- Deletion:
- Renew Domain Assignment:
- No Status Calculation:

The 'Object Selection' panel shows the following counts:

Object Selection	Count
Object Lists	1 Entries
Object Types	286 Entries
Graphs	1 Entries
Target Languages	2 Entries

Activation of Evaluation created above generates a job as shown below:

Job	Ln	Job CreatedB	Status	Start date	Start time	Duration(sec.)	Delay (sec.)
<input type="checkbox"/> EVALUATION_00001_000		I043506	Finished	26.03.2009	13:06:08	0	0
<input type="checkbox"/> EVALUATION_00001_000_frFR_ONE		I043506	Finished	26.03.2009	13:06:08	1	0
<input type="checkbox"/> EVALUATION_00001_001		I043506	Finished	26.03.2009	13:06:39	3	0
<input type="checkbox"/> OBJLIST_00001		I043506	Finished	26.03.2009	12:59:35	158	0
*Summary						162	0

At the end of this step, the system is successfully set up for carrying out the translation. Now translator can login and carry out the translation.

Caution: Note that the jobs that get executed in background when the Object list and Evaluation are activated should have finished successfully before translator starts the Worklist. In case any of these jobs fail, the translator may not be able to load the objects in worklist.

Access the Worklist in SE63 and Perform the Translation

Now translator can access the worklist and perform the translation of the objects that are assigned to him/her.

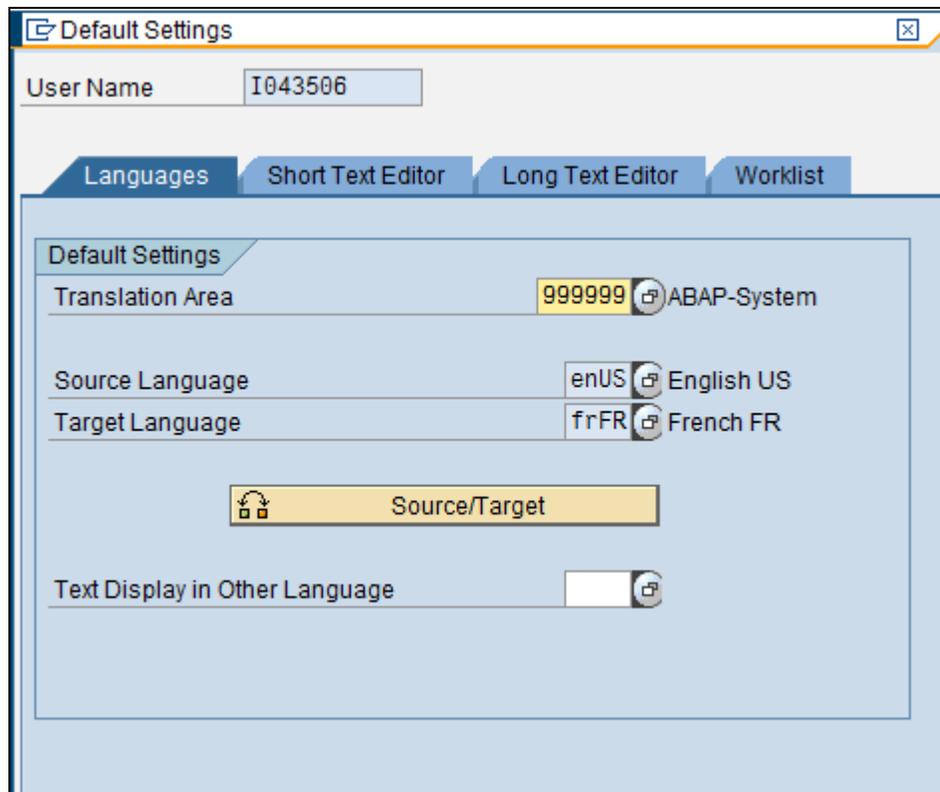
The concept is analogous to the Test Package assignment in Solution manager where a set of test cases are grouped into a Test package which is assigned to a user. User accesses the test package assigned to him/her and then performs the testing. In same way here user will access the Worklist assigned to him/her to carry out the translation.

If the translator calls up a personal worklist using the standard worklist option, translator can only access objects if they have been evaluated, if the collections to which the objects belong are assigned to translator, and if no other translator has reserved the objects in his or her personal worklist. For more information, please refer to online help link below: [Accessing a Personal Worklist](#)

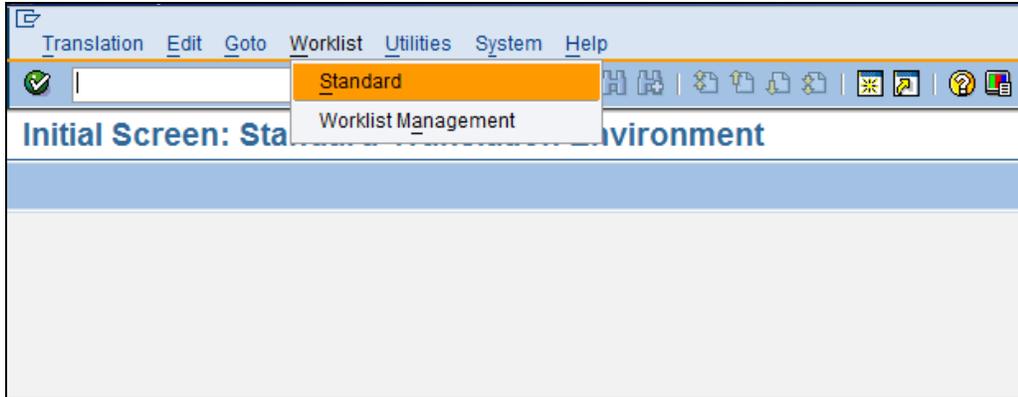
The below steps describe the procedure for the translator to load and execute the translation:

Step 1: Check the default settings in SE63 transaction

Before the Worklist is started, check the Default settings to make sure that proper Source and Target language are selected. The settings can be checked using Menu Utilities → Settings in SE63:



Step 2: Open the worklist using menu Standard → Worklist

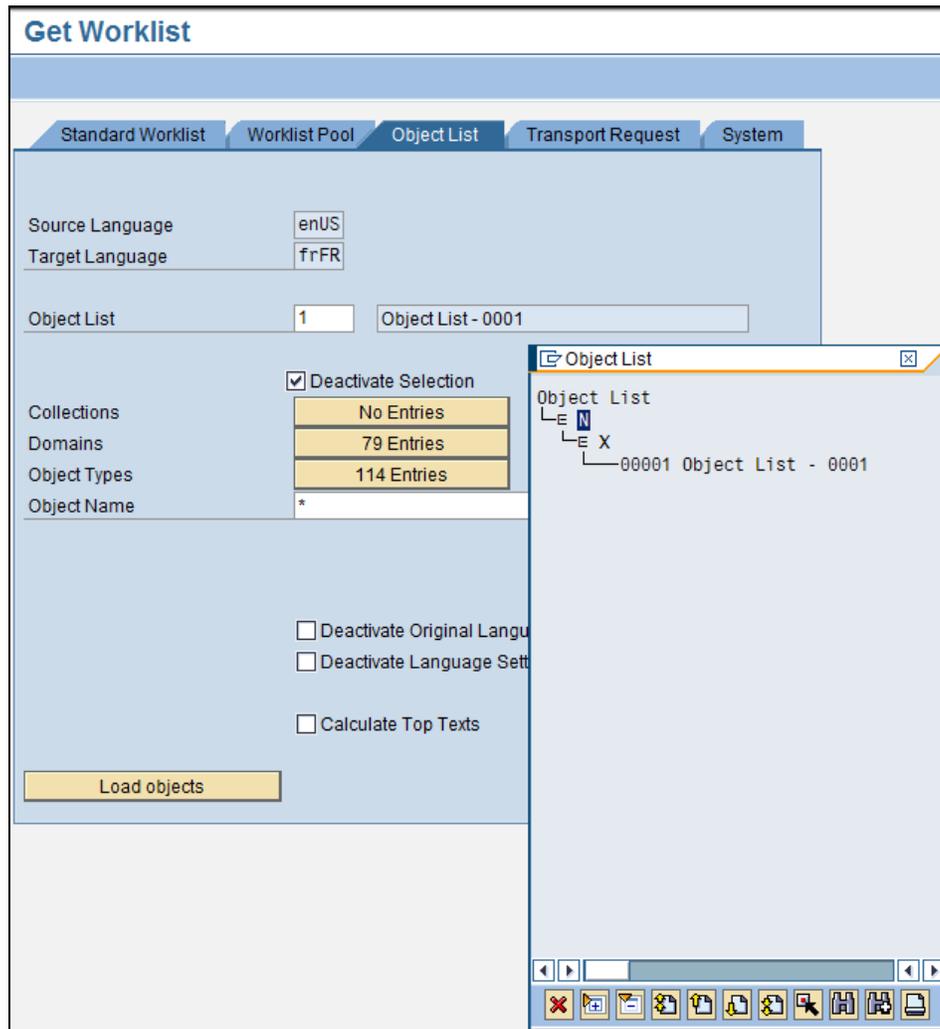


Step 3: Choose the Worklist number and navigate to next screen



Step 4: Load the Worklist using “Object List” tab page

In this step choose the “Object list” which has been assigned to a particular user. The possible Object lists can be chosen using the search help provided.



Now click on the button “Load Objects” which loads all the objects that needs translation based on the Worklist assigned to translator during set-up

The objects relevant for translation are shown in a hierarchical way grouped according to the object types shown below. This enables the translator to easily and efficiently manage and perform translation.

Worklist: Object List For I043506 From enUS To frFR

Sequential Get/Extend Remove Objects from Worklist

Worklist for I043506 (2 objects found)

- Repository Texts
 - <DOMA> Domains
 - ZZ_DOMAIN_1
 - <VALU> Fixed Values for Domains
 - ZZ_DOMAIN_1

Step 5: Carry out translation for all objects in Worklist

Double clicking on the individual objects will take into the actual translation screen as shown below where translations can be carried out

VALU (..) From enUS To frFR: ZZ_DOMAIN_1

(No comment exists for object)

[DDTEXT 00001]			(60)
Domain1_EN_01			
Domain1_FR_01			
[DDTEXT 00002]			(60)
Domain1_EN_02			
Domain1_EN_02_FR			
[DDTEXT 00003]			(60)
Domain1_EN_03			
Domain1_EN_03			

Advantages of Translation Using Worklists

Translation using Worklists is recommended for large translation projects where a large number of objects need to be translated into multiple languages. For example, over a thousand objects need to be translated into 3 different languages. The main advantages with this approach are:

- Worklists can be created and assigned to specific users
- Worklists can be created separately for different languages, users and objects. This will allow more flexibility, management and proper tracking of the translation. A typical example would be:
 - Translations for all domains from EN → FR to be done by User-A
 - Translations for all data elements from EN → FR to be done by User-B
 - Translations of all objects from EN → ES to be done by User-C
- There is a possibility to include an entire development package in an object list. This means all the objects in this package which can be translated will be automatically included. And hence there is no need to remember the objects that need translation.
- Worklist provides all objects to be translated in an organized tree structure which is very helpful for the translator to carry out the translation.
- Statistics can be generated to monitor the progress of the translation project.
- Translation can be carried out efficiently and consistently

Appendix and Recommendations

Recommendation

- Standard objects will be translated automatically when new language is installed. Additionally language supplementation can be done to avoid manual translation of all standard objects which are not provided by SAP. The recommendation would be to either supplement with German (DE) or English (EN).
- For all the custom developed objects in Z* customer namespace, translation using worklists is recommended where worklists can be created and translation can be carried out efficiently for all the objects in worklist in SE63.
- For performing translation of standard objects which were enhanced / modified, either Object translation in the respective development transactions or individual objects' translation (SE63) can be used.
- Master data can be translated using corresponding transactions.
- Customizing will be translated again in SM30 directly.

Supplementation

Before translation, it is recommended to use supplementation, in order to limit translation works. Use the language supplementation function to fill in the gaps in a language that has not been translated completely.

The texts stored in cross-client database tables can also be accessed from all clients simultaneously. The default setting specifies that cross-client tables are supplemented when logged on to client 000.

The screenshot shows the SAP Language Management interface. The left pane displays a tree view of languages, including Chinese, Czech, Danish, Dutch, English, Finnish, French, German, Greek, Hungarian, Italian, Japanese, Korean, Norwegian, Polish, Portuguese, Russian, Slovakian, Spanish, Swedish, Thai, and Turkish. The right pane shows a table of supplementation jobs.

Supplementation Job	Client	System	Date	Description
CLIENT_SUP	700	SAP*	31.01.06	Mandantenversorgung: Mandant 800
CLIENT_SUP	700	MIZUNO	18.11.05	Client Maintenance: Client 810
CLIENT_SUP	700	MIZUNO	18.11.05	Client Maintenance: Client 805
CLIENT_SUP	700	MIZUNO	18.11.05	Client Maintenance: Client 800

- A. Go to the language that needs to be supplemented and choose  or *Language → Supplement Language*.
- B. On the next screen, enter a description for the supplementation. This helps to recognize the supplementation that was started in the transaction SMLT overview.
- C. In the center of the dialog box, enter the conditions for selecting the tables that needs to be supplemented. These fields are set with SAP defaults that depend on the logon client; however, they can be changed. Client-specific Customizing tables are subject to special handling during language imports. This has resulted in the following supplementation strategies being created:
- *Combined with RSREFILL* (standard setting when logging on to clients other than 000)
 - *Combined with Client Maintenance*
 - *Supplement Only*
- D. Enter a start time and (optional) target server and it's now ready to supplement the language by choosing *Execute*. This action schedules a background job that performs the supplementation.
- E. The overview screen of transaction SMLT creates a supplementation entry for the appropriate language. Choose the corresponding icon to administer and monitor the supplementation.

The overview contains the following icons:

	Display supplementation log (in table)
	Display job information
	Display job log
	Restart action, if it has terminated for technical reasons

- F. The icon at the start of the line indicates the status of the import. The icons have the following meanings:

	The selected table was supplemented without errors.
	There were warnings during the supplementation. These warnings can be analyzed by displaying the supplementation log. To display the supplementation log, choose  or double-click the name of the supplementation action.
	There were errors during supplementation. Analyze them as described above.
	The action is scheduled. Choose  to display job information, such as the start time and target server.
	The action is currently running.
	The action has terminated for technical reasons. To find out the cause of the error, see the job log (choose ).

- G. The supplementation log gives the detailed information about the tables that have been supplemented.

Each line of the log contains the name, class, description, and other information about a supplemented table. The status, client, and supplementation mode appear first, followed by statistics about how many rows in the table have been processed, inserted, and modified. Return code can also be used for analysis.

All tables that have been processed fully are given a status based on the return code. This status is indicated by a green, yellow, or red light. For information about the meaning of the return codes, display the F1 help for the return code (RC) column.

Related Content

SAP Service Market place – <http://service.sap.com>
 SAP Online Help – <http://help.sap.com>
 SAP Developer Network – <http://sdn.sap.com>
 SRM RKT - <http://service.sap.com/rkt-srm>
 SAP Language Services - <http://service.sap.com/translation>

Documents	Version No.
OSS Note 920085 <i>Additional languages for countries without SAP</i>	
OSS Note 885571 <i>Translation connection in system is 'Not changeable'</i>	
OSS Note 481289 <i>No language supplementation program in translation</i>	

Copyright

© Copyright 2009 SAP AG. All rights reserved.

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

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Excel, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, System i, System i5, System p, System p5, System x, System z, System z10, System z9, z10, z9, iSeries, pSeries, xSeries, zSeries, eServer, z/VM, z/OS, i5/OS, S/390, OS/390, OS/400, AS/400, S/390 Parallel Enterprise Server, PowerVM, Power Architecture, POWER6+, POWER6, POWER5+, POWER5, POWER, OpenPower, PowerPC, BatchPipes, BladeCenter, System Storage, GPFS, HACMP, RETAIN, DB2 Connect, RACF, Redbooks, OS/2, Parallel Sysplex, MVS/ESA, AIX, Intelligent Miner, WebSphere, Netfinity, Tivoli and Informix are trademarks or registered trademarks of IBM Corporation.

Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

SAP, R/3, SAP NetWeaver, Duet, PartnerEdge, ByDesign, SAP Business ByDesign, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and other countries.

Business Objects and the Business Objects logo, BusinessObjects, Crystal Reports, Crystal Decisions, Web Intelligence, Xcelsius, and other Business Objects products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Business Objects S.A. in the United States and in other countries. Business Objects is an SAP company.

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

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.