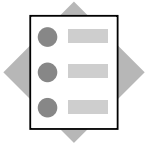




Contents:

- **Develop International Web Dynpro Applications**



After completing this lesson, you will be able to:

- **Internationalize Web Dynpro applications.**
- **Create simple types that contain the interface texts.**

Developing International Web Dynpro Applications

The image displays two side-by-side screenshots of SAP Web Dynpro user interfaces, illustrating the process of internationalization. The left screenshot is in German, titled 'Willkommen bei Web Dynpro!', and the right is in English, titled 'Welcome to Web Dynpro!'. Both forms are for a car rental reservation.

German Version (Left):

- Personendaten:** Vorname * (DE_LXRY.jpg), Nachname * (Schmitt)
- Reservierungsdaten:** Abholdatum (06.08.2004), Rückgabedatum (06.08.2004), Fahrzeugtyp (Luxury dropdown), and a car image.
- Buttons:** Sichern, Zurücksetzen

English Version (Right):

- Person data:** Firstname * (John), Lastname * (Smith)
- Reservation data:** Pickup Date (8/6/2004), Dropdown Date (8/6/2004), Vehicle Type (Luxury dropdown), and a car image.
- Region:** Radio buttons for Asia, Europe, South America, Africa, North America (selected), and Oceania.
- Location:** Pickup location (AMN_NY), Dropdown location (AMN_NY)
- Buttons:** Save, Reset

© SAP AG 2004, Internationalization / 3

THE BEST-RUN BUSINESSES RUN SAP



■ Internationalization

The purpose of internationalization is to adapt existing applications so that they can be used in several languages and countries (or regions) without the need for changes to the source code. Specifically, at SAP internationalization deals with the issue of text strings hard-coded in applications. Using the internationalization tools, you can store these text strings separately from the application source code, in a format that can be processed in the standard translation system.

To make the translation process smoother, we recommend that you store any frequently used language-specific texts that you have defined in the Layout editor, such as labels or table headers, in *Simple Types*.

Developing International Web Dynpro Applications

▶ In Web Dynpro, the text elements that you create at design time are isolated and grouped automatically in resource bundles.

▶ These *.xlf files are created for the following Web Dynpro elements.

WD element	Name of *.xlf file
MessagePool	<WDComponentName>MessagePool.wdmessagepool.xlf
View	<View Name>.wdview.xlf<View Name>.wdcontroller.xlf (text of the actions, for example)
Window	<Window Name>.wdwindow.xlf (title of the window, for example)
SimpleType	<SimpleType Name>.dtsimpletype.xlf

■ Internationalization

To internationalize the Web Dynpro application, copy the automatically generated *.xlf files and save them under a new name in the same directory. The new name must meet the following convention:

- <old name>_<language key>.xlf

Choose an appropriate language key. For example, if you are creating *.xlf files for German, use the language key **de**.

Example: Simple Types

VehicleType Defined in Java Dictionary

Context

Root Node

VehicleType

Simple Type VehicleType

Key	Displaytext
CMPC	Compact
ECNM	Economy

View Layout

Vehicle Type

Data Binding

VehicleType.dtsimpletype.xlf

Resource Name	Resource Type	Text	Max Le
SchemaType/Enumera...	technicaltextlong	Minivan	255
SchemaType/Enumera...	technicaltextlong	<Select vehicle ty...	255
SchemaType/Enumera...	technicaltextlong	Compact	255
SchemaType/Enumera...	technicaltextlong	Cabrio	255
SchemaType/Enumera...	technicaltextlong	Economy	255
SchemaType/Enumera...	technicaltextlong	Luxury	255

- Copy the automatically generated *.xlf files
- Save them under a new name in the same directory. Convention: <old name>_<language key>.xlf
- Open new files and translate entries.

© SAP AG 2004, Internationalization / 5

THE BEST-RUN BUSINESSES RUN SAP

SAP

Internationalization

Among other things, simple types are suitable for encapsulating frequently used texts that are defined in the Layout Editor at design time.

The graphic shows you an example of how to use a simple type. The *ReturnDate* simple type has the type *date* and encapsulates the texts *Return Date* (a label text) and *Choose a date* (a quick info text). You can now assign various context attributes to this simple type, such as *ReturnDate*. The context attribute can itself be bound with an input field. In this way, the quick info text of the simple type is used for the input field. If you define a label for the input field (you must set the *labelFor* property for the label), the label text of the simple type is used.

You can also define enumerations for simple types. You can also assign this simple type to a context attribute, and bind it to a dropdown list. In this way, all values included in the simple type can be displayed in the dropdown list.

Determine Language Specific Text Information at Runtime

User	Session Locale			System	VM default	WD calc.
	User	Browser	default			
authenticated	de	en	fr	it	ru	de
anonymous	-	en	fr	it	ru	en
anonymous	-	-	fr	it	ru	fr
anonymous	-	-	-	it	ru	it
anonymous	-	-	-	-	ru	ru

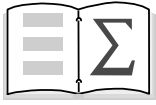
■ Determine Language Specific Text Information at Runtime

The session locale is the locale that is specified for the current user by the user management engine (UME). The fallback sequence is:

- 1) In case of an authenticated user, the locale specified for this user is returned.
- 2) Otherwise, the locale specified by the browser settings ("accept-language" HTTP header) is returned if existing.
- 3) Otherwise the default locale specified in the application properties is returned if existing.
- 4) Otherwise the default locale specified in the Web Dynpro system properties is returned if existing.
- 5) Otherwise the default locale of the VM is returned.

The locale calculated by the Web Dynpro Runtime (last column) is passed as input to the `java.lang.ResourceBundle` class loaded by the SAP J2EE Engine. This class uses the following fallback sequence to load a physically existing resource bundle:

- load resource bundle for WD calculated locale
- if not existing: load resource bundle for the default locale of the VM
- if not existing: load resource bundle without language suffix, which always exists



You should now be able to:

- **Internationalize Web Dynpro applications.**
- **Create simple types that contain the interface texts.**