

Dynamische Mehrsprachigkeit für Forms und Reports

Ronald Steiner
PITSS GmbH
Stuttgart

Schlüsselworte

Multilanguage , Oracle Forms, Datenbank, Oracle Reports.

Warum dynamische Mehrsprachigkeit oder Multilanguage?

Viele Softwareentwickler stoßen früher oder später auf die Notwendigkeit der Übersetzung des Softwareprodukts in die Sprache des Landes, wo es später verwendet oder verkauft wird. Seit der Globalisierung der Unternehmen tritt dieser Fall öfters ein. Betrachten Sie die folgende Beispielsituation in der sich das Hauptunternehmen in Deutschland und die Tochtergesellschaft in den Vereinigten Staaten von Amerika befindet. Die Software, die von beiden Unternehmen verwendet wird, wird in den USA entwickelt. Die deutsche Firma braucht jedoch eine übersetzte Version des Programms und eine Möglichkeit, mit der es relativ einfach ist eine Anpassung des Programms (Hinzufügen, Bearbeiten vorhandener Übersetzung von bestimmten Blöcken) durchzuführen.

Es wird die Übersetzung einer Form mit Hilfe der Oracle Software betrachtet. Als Beispiel wird aus verschiedenen Feldern und Labels eine Form mit zwei Schaltflächen (Deutsch und Englisch) erstellt.

Voraussetzung für die Übersetzung

Die Übersetzung der Software (basierend auf Oracle Forms) anhand der Übertragung des Wertes des Parameters NLS_LANG (National Language Support) ist der einfachste Weg. Für die Anwendung, die übersetzt wird, sollen auch Tabellen erstellt werden, in denen sich unter anderem folgendes befindet:

- originale der Felder und Ihre Übersetzung (z.B. Save und Speichern)
- mögliche Sprachen zur Übersetzung der Anwendung
- etc.

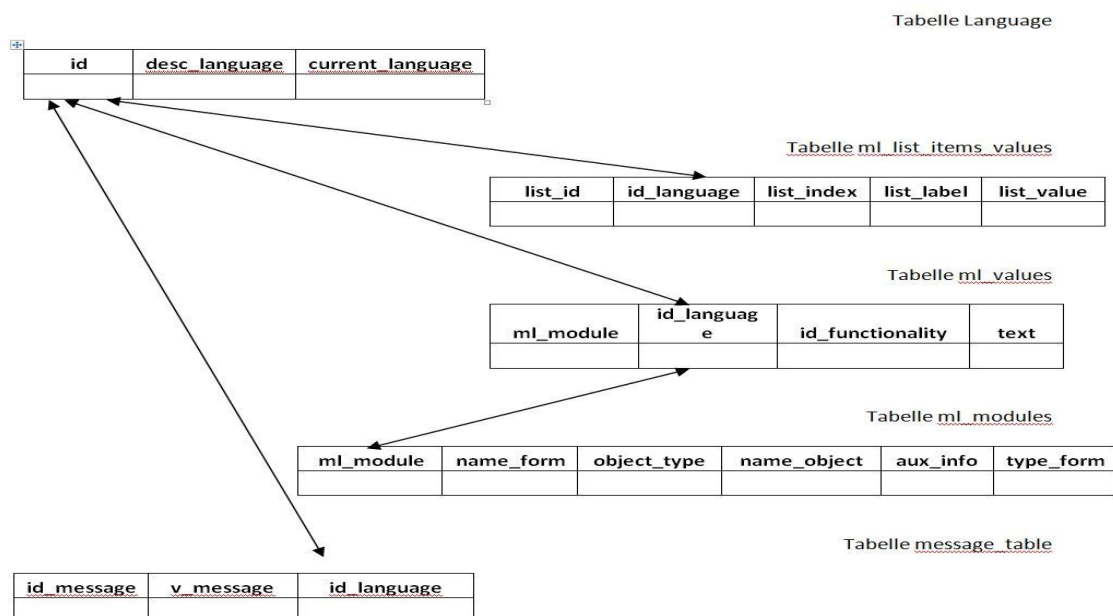


Abb. 1: benötigte Tabellen

Am Beispiel dieser Tabellen lässt sich leicht erkennen:

1:1 Beziehung - eine Sprache entspricht einer Beschreibung

1:n Beziehung - viele Übersetzungen (in unterschiedlichen Sprachen) entsprechen einem Wort, das sein Feld in der Originalbedeutung beschreibt.

Jetzt wird eine Prozedur beschrieben, die Messages der Ausgangssprache in eine Zielsprache übersetzt. Alle Messages, die die Anwendung enthält, müssen unbedingt in eine Tabelle (message_table) herausgezogen werden.

In dieser Tabelle werden benötigt:

- Die Messages in Originalsprache des Programms und alle Übersetzungen in den jeweils ausgewählten Sprachen(v_message)
- die Ordnungsnummer der Message(id_message)
- die Nummer die einer Sprache zu Übersetzung entspricht (id_language)

Dann wird eine Funktion mit Parameter (z.B. get_string) erstellt. Diese muss:

- den Text der Message aus der Tabelle message_table der aktuellen Sprache herausfiltern. Als aktuelle Sprache wird die Sprache gesetzt in die übersetzt werden soll.
- den ausgewählten Text der zusätzlichen Variablen zuweisen.
- den Text in der Variablen durch den Eingabewert der Funktion ersetzen. Auch wird davon ausgegangen, dass die Länge der Zeichenkette nicht mehr wie 80 Symbole hat.

Am Ende soll die Funktion den String zurückgeben.

Nachdem die Tabellen mit den notwendigen Daten (Beschreibung der Felder (Original + Übersetzung), Textnachrichten (Original + Übersetzung)) befüllt wurden, wird die Prozedur der Anwendung hinzugefügt. Die Prozedur liest aus der Tabelle, in welcher Sprache das Programm ausgeführt werden muss und welche ursprünglichen Werte die Felder der Form annehmen müssen.

Weiter werden die nächsten Schritte beschrieben die durchgeführt werden müssen um die Form vom Englischen ins Deutsche zu übersetzen. Als Beispiel nehmen wir folgende Formsanwendung:

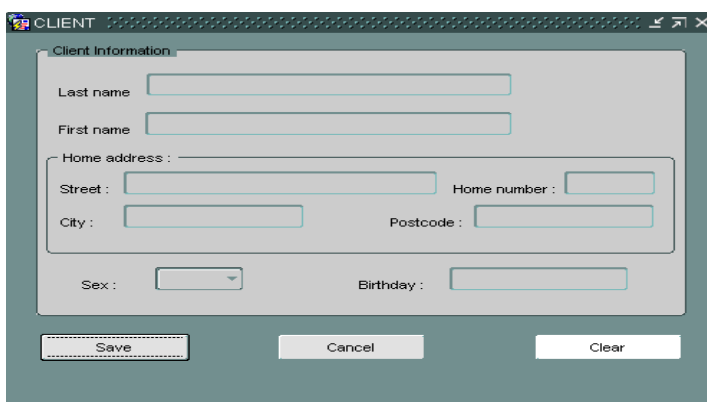


Abb. 2: Beispielanwendung

Auf dieser Maske sind die Elemente so wie Labels (Last_name, First_name, Street usw.), Textobjekte, Listenobjekte (Sex) und Schaltflächen (Save, Cancel, Clear) zu sehen.

1.Schritt: Attributkorrektur (Konvertierung Boilerplates in Prompts)

Es wird das Textobjekt "LAST_NAME" ausgewählt. Mit der rechten Maustaste wird das Menü geöffnet. Dort wird die Attributpalette ausgewählt.

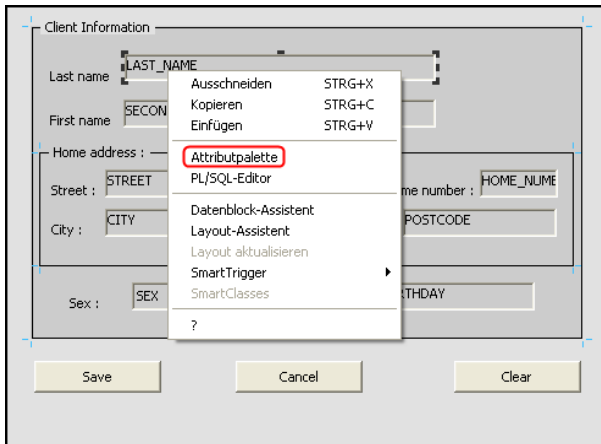


Abb. 3: Attributkorrektur

In der Zeile "Prompt" wird der Labelname eingetragen den man in Prompt konvertieren will (z.B. Last_Name). Mit dieser Methode werden alle Labels und Boilerplates (oder Standardtexte) in Prompts konvertiert.

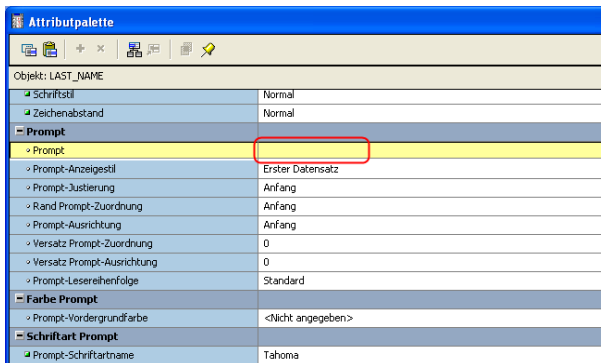


Abb. 4: Attributkorrektur

2.Schritt: Anlegen der Tabellen

In den folgenden Tabellen werden alle Informationen über Übersetzungen, Sprachen. Modul Eigenschaften und Anwendungsmeldungen aufbewahrt.

Zum Beispiel wird die Tabelle **language** erstellt. In dieser dreispaltigen Tabelle sind die Nummer der Sprache, die Sprachbeschreibung und die aktuelle Sprache (mit Buchstaben Y für Yes oder N für No) eingetragen.

Die zweite Tabelle ist **ml_modules**. Diese Tabelle enthält die Informationen über die Module der Anwendung.

Die dritte Tabelle ist **ml_values** mit den Übersetzungen der Eigenschaften der Module der Anwendung.

Die vierte Tabelle **ml_list_items_values** beinhaltet die Übersetzung der List Items.

Abschliessend die Tabelle **message_table** die die Messages der Anwendung enthält.

Ausserdem werden fünf temporäre Tabellen erstellt(**tmp_language**, **tmp_ml_values**, **tmp_ml_list_items_values**, **tmp_message_table**), die für den Import der Sprachen, Module der Anwendung, Eigenschaften der Module, Werte der List Items und Messages verwendet werden können.

3.Schritt: Befüllen der Tabellen

Die Tabelle **tmp_language** wird mit der Hauptsprache, in der die Anwendung geschrieben ist, und die Sprache in der die Anwendung übersetzt werden soll ergänzt.

```
insert into tmp_language (id_language, desc_language, current_language) values (1, 'American', 'Y')
```

```
insert into tmp_language (id_language, desc_language, current_language) values (23, 'German', 'N')
```

Die erstellten Tabellen werden mit den Informationen über die vorhandenen Objekte der Form sowohl in der Originalsprache (Englisch) als auch in der Zusatzsprache (Deutsch) gefüllt:

für Tabelle **tmp_ml_modules**:

```
insert into tmp_ml_modules (pk_ml_modules, name_form, object_type, name_object, aux_info1, aux_info2, type_form) values (63, 'CLIENT', 13, 'SAVE', 'BLOCK3', "", 10)
```

```
insert into tmp_ml_modules (pk_ml_modules, name_form, object_type, name_object, aux_info1, aux_info2, type_form) values (64, 'CLIENT', 13, 'CANCEL', 'BLOCK3', "", 10)
```

für die Tabelle **tmp_ml_values** (mit Übersetzung):

```
insert into tmp_ml_values (fk_ml_modules, id_language, id_functionality, text) values ( 63, 1, 7, 'Save')
```

```
insert into tmp_ml_values (fk_ml_modules, id_language, id_functionality, text) values ( 63, 1, 6, 'Save')
```

```
insert into tmp_ml_values (fk_ml_modules, id_language, id_functionality, text) values ( 63, 1, 15, "")
```

```
insert into tmp_ml_values (fk_ml_modules, id_language, id_functionality, text) values ( 63, 1, 8, "")
```

```
insert into tmp_ml_values (fk_ml_modules, id_language, id_functionality, text) values ( 63, 23, 7, 'Speichern')
```

```
insert into tmp_ml_values (fk_ml_modules, id_language, id_functionality, text) values ( 63, 23, 6, 'Speichern')
```

```
insert into tmp_ml_values (fk_ml_modules, id_language, id_functionality, text) values ( 63, 23, 15, "")
```

```
insert into tmp_ml_values (fk_ml_modules, id_language, id_functionality, text) values ( 63, 23, 8, ")
```

Wenn die Anwendung Warnungen oder Kommentare, enthält, müssen diese auch übersetzt werden und in die Tabelle **message_table** aufgenommen werden.

Weiter werden die Informationen aus den Tabellen mit dem Präfix tmp_ in die Tabellen ohne dieses Präfix kopiert.

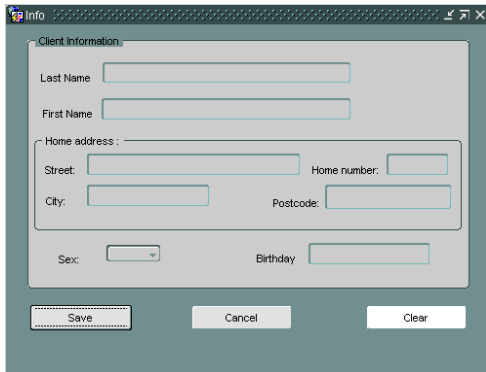
```
INSERT INTO ml_modules(pk_ml_modules, name_form, object_type, name_object, aux_info1, aux_info2, type_form)
```

```
SELECT pk_ml_modules + (l_max_production - l_min_temporary + 1),  
                                             name_form,  
                                             object_type,  
                                             name_object,  
                                             aux_info1,  
                                             aux_info2,  
                                             type_form  
  
FROM tmp_ml_modules;
```

Anschließend wird die zu betrachtende Anwendung noch durch die Prozedur CONFIG_LANG ergänzt, in der:

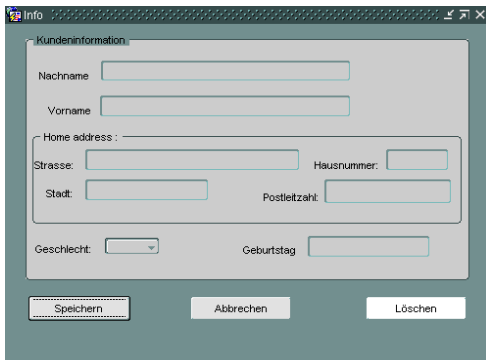
- Cursor für Listenelementen, die Werte der Listenelemente, die Werte der Eigenschaften deklariert werden
- die Funktion definiert wird, die die Sprache, auf der die gestartete Form dargestellt werden soll, bestimmt;
- die Eigenschaften für die Warnungen, den Hinweistext usw. vorgegeben werden.

Und dann wird die Anwendung ausgeführt:



The screenshot shows a window titled 'Info' with a form titled 'Client Information'. The form contains the following fields: 'Last Name' (text input), 'First Name' (text input), 'Home address' (grouped box containing 'Street' and 'Home number' text inputs), 'City' (text input), 'Postcode' (text input), 'Sex' (dropdown menu), and 'Birthday' (text input). At the bottom, there are three buttons: 'Save', 'Cancel', and 'Clear'.

Abb. 5: Originalanwendung



The screenshot shows a window titled 'Info' with a form titled 'Kundeninformation'. The form contains the following fields: 'Nachname' (text input), 'Vorname' (text input), 'Home address' (grouped box containing 'Strasse' and 'Hausnummer' text inputs), 'Stadt' (text input), 'Postleitzahl' (text input), 'Geschlecht' (dropdown menu), and 'Geburtsdag' (text input). At the bottom, there are three buttons: 'Speichern', 'Abbrechen', and 'Löschen'.

Abb. 6: Übersetzte Version

Auf den oben angeführten Bildern wird die Darstellung der Formsanwendung vor und nach der Übersetzung dargestellt. Es ist auch zu berücksichtigen, dass es eine sehr einfache Version einer Anwendung ist. In der Praxis ist es viel komplizierter. Anwendungen bestehen aus mehreren miteinander verbundenen Forms. Eine Applikation, die in eine andere Sprache übersetzt werden muss, kann bereits am Anfang Schnittstellen in mehreren Sprachen beinhalten. All dies erschwert den oben beschriebenen Prozess. Mit der Weiterentwicklung der Anwendung ist es möglich, vorhandene übersetzte Objekte (aus dem sog. Wörterbuch) zu verwenden.

Die oben beschriebenen einzelnen Umwandlungsschritte der Anwendung sind im Programm PITSS.CON automatisiert:

- vereinfachte Möglichkeit die Sprachen hinzuzufügen,
- Konvertierung von Boilerplates in Prompts,
- Skripterstellung,
- die Erstellung von dynamischen Wörterbüchern usw.

Kontaktadresse:

Ronald Steiner
PITSS GmbH
Zettachring 2
D-70567 Stuttgart

Telefon: +49 (0) 711 - 875213
Fax: +49 (0) 711 - 72875201
E-Mail: rsteiner@pitss.de
Internet: www.pitss.de