

# Создание многоязычных приложений, переводимых пользователями.

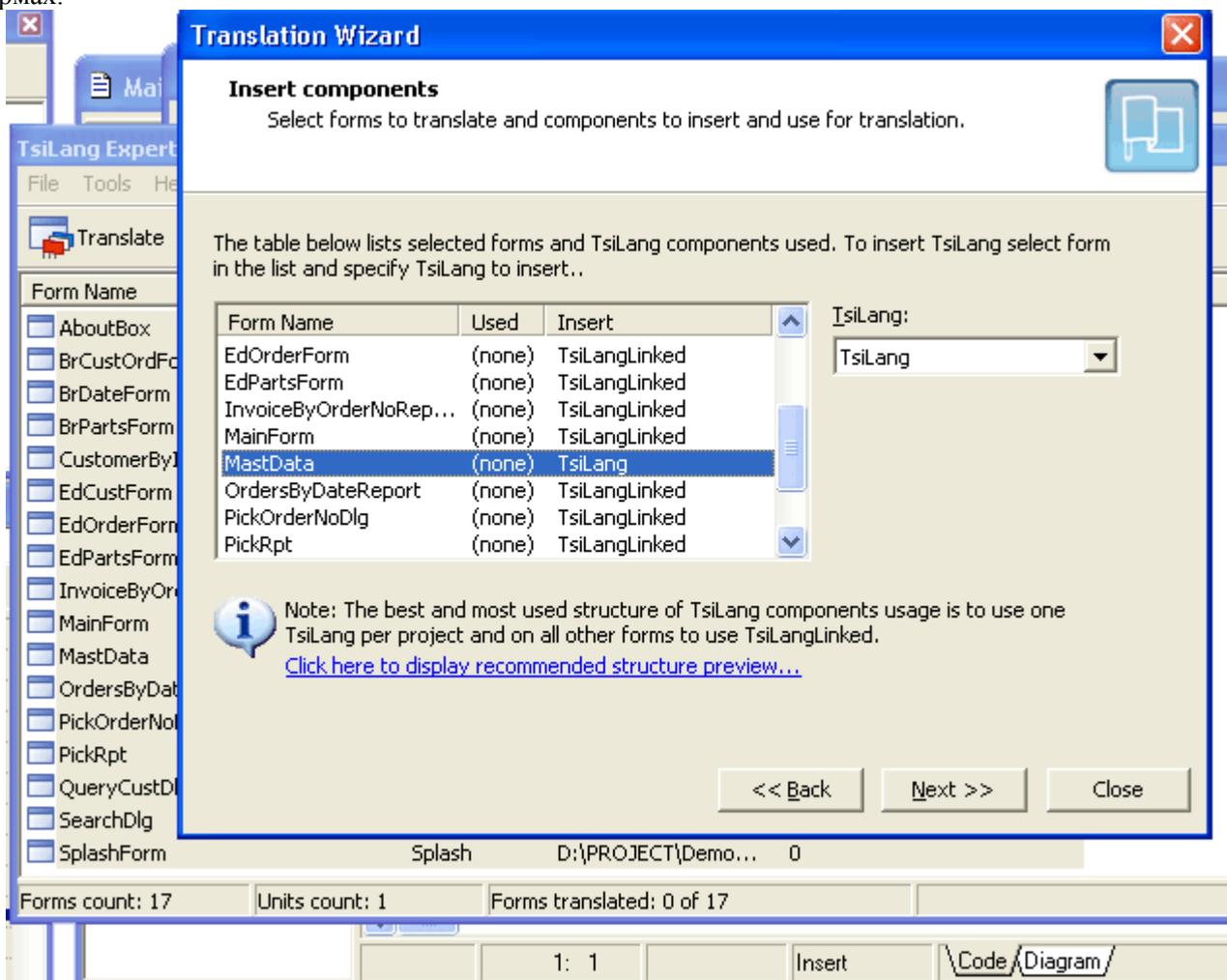
Задача создания многоязычных приложений, в которые пользователи могут добавлять свои переводы и языки, гораздо проще, чем видится на первый взгляд. Конечно, учитывая, что мы используем TsiLang Components Suite в своих разработках.

Рассмотрим решение данной задачи на примере проекта MastApp из папки примеров Delphi.

## 1. Перевод проекта

На переводе проекта мы не будем долго останавливаться, подразумевая, что Вы уже освоили эту часть. Можно только обратить внимание на используемые компоненты.

В проекте мы будем использовать один TsiLangDispatcher, один TsiLang и TsiLangLinked на остальных формах.



Изображение 1: Мастер перевода

Вставляем TsiLang и TsiLangDispatcher на модуль MastData, так как он создается автоматически и используется всеми модулями проекта.

## 2. Создание файла переводов.

После ввода переводов для всех форм, а можно и не вводить переводы, оставив только английские

термины, мы можем экспортировать переводы во внешний файл. Для этого воспользуемся функцией File | Save/Load Translations|Save Project из TsiLang Expert (TsiLang Expert находится в меню Tools). Мы рекомендуем использовать SIB файлы для хранения переводов, так как работа с ними гораздо быстрее, по сравнению с SIL файлами. Но Вы можете выбрать и SIL файлы, так как они хранят данные в обычном текстовом виде и могут быть отредактированы любым текстовым редактором.

### 3. Использование внешнего файла и динамическое изменение доступных языков.

Вы можете использовать два варианта:

1. Задать имя файла в свойстве FileName для TsiLangDispatcher
2. Проверять наличие файла в установленном месте в коде и потом загружать его самостоятельно.

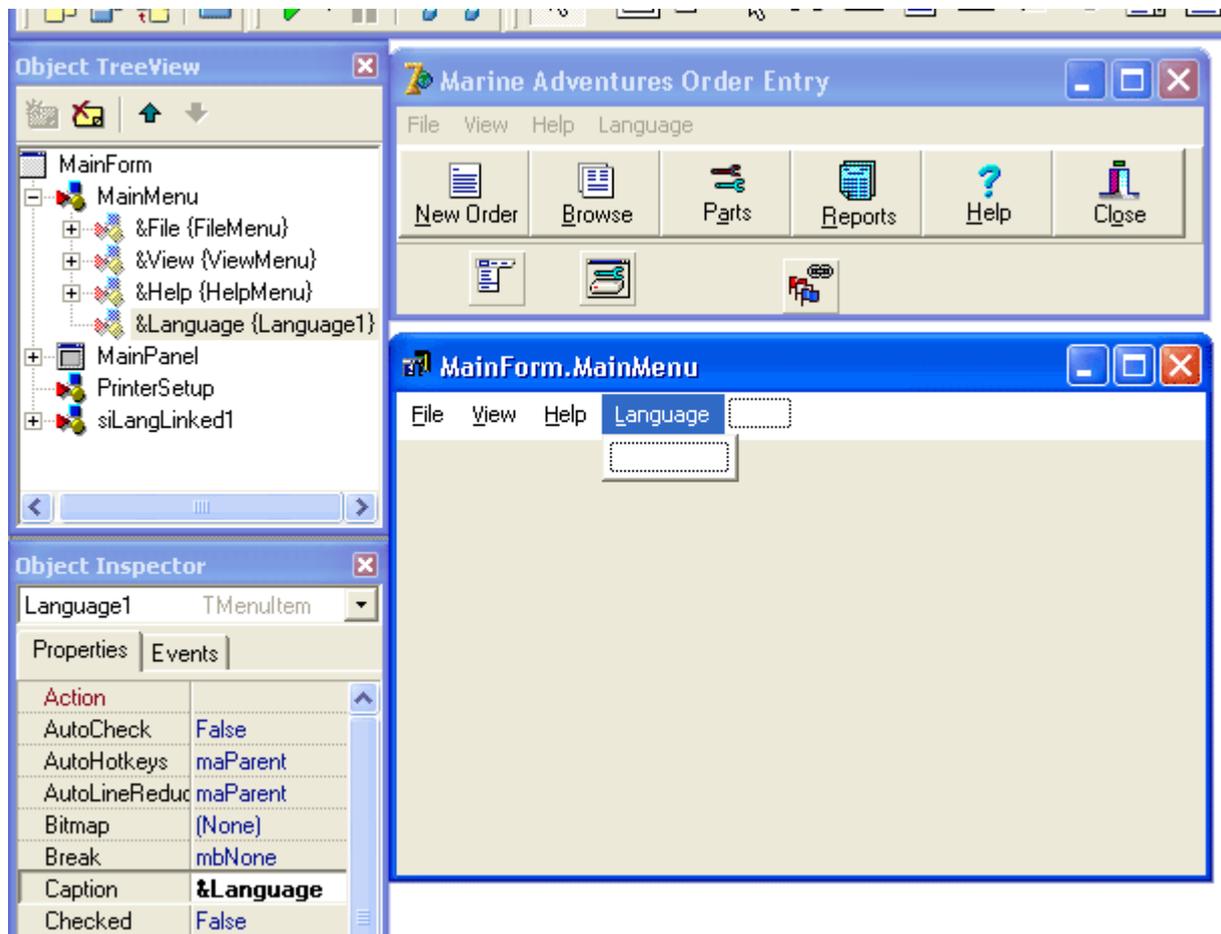
В первом варианте, диспетчер будет самостоятельно проверять наличие файла, указанного в свойстве FileName и загружать его во все формы. Для второго варианта можно использовать следующий код:

```
procedure TMastData.DataModuleCreate(Sender: TObject);  
var  
    sOurSibFile: string;  
begin  
    // вычисление названия файла  
    sOurSibFile := ExtractFilePath(Application.ExeName) + 'TheNameOfYourFile.sib';  
    // проверяем, существует ли такой файл  
    if FileExists(sOurSibFile) then  
        begin  
            // выставляем значение свойства  
            // это позволит автоматически загружать формы, созданные позднее  
            siLangDispatcher1.FileName := sOurSibFile;  
            // загружаем переводы в уже созданные формы  
            siLangDispatcher1.LoadAllFromFile(sOurSibFile);  
        end;  
end;
```

Вы просто должны определиться, какой из методов будет использован, исходя из собственных предпочтений.

Для динамического отображения доступных языков мы:

1. Добавим новый пункт меню в главное меню приложения.



Изображение 2: Добавление меню языков

2. Добавим код, настраивающий пункты меню, соответственно имеющимся языкам. Его можно добавить в код события OnShow для главной формы проекта:

```
// изменение языка при выборе пункта меню
procedure TMainForm.LanguageMenuItemClick(Sender: TObject);
begin
  MastData.siLangDispatcher1.ActiveLanguage := TMenuItem(Sender).Tag;
  TMenuItem(Sender).Checked := True;
end;

procedure TMainForm.FormShow(Sender: TObject);
var
  MenuItem: TMenuItem;
  I: Integer;
begin
  for I := 1 to MastData.siLangDispatcher1.NumOfLanguages do
  begin
    MenuItem := TMenuItem.Create(Self);
    // устанавливаем заголовок
    MenuItem.Caption := MastData.siLangDispatcher1.LangNames[I - 1];
    // устанавливаем тэг для удобства переключения языков
    MenuItem.Tag := I;
    MenuItem.RadioItem := True;
    // отображаем текущий выбор языка
    MenuItem.Checked := I = MastData.siLangDispatcher1.ActiveLanguage;
    // устанавливаем обработчик выбора языка
    MenuItem.OnClick := LanguageMenuItemClick;
    // добавляем пункт меню
    Language1.Add(MenuItem);
  end;
end;
```

Внимание! Код обновления и отображения языков должен выполняться после загрузки файла переводов.

## 4. Заключение.

Теперь запускаем проект на выполнение и «вот и все»! Как видите, задача, казавшаяся сложной, решена в течение нескольких минут, используя TsiLang Components Suite. Пользователи смогут переводить Ваше приложение без Вашего участия и перекомпиляции. Вы можете предложить пользователям использовать SIL Editor, который может быть загружен со страницы <http://www.sicomponents.com/download.html>

Домашняя страница TsiLang Components Suite: <http://www.tsilang.ru>

Скачать последнюю версию можно с <http://www.tsilang.ru/download.html>