Portrait
Foundation

Localization Add-in User Guide

Edition 7.0

16 January 2013

Pitney Bowes
Software

Portrait Foundation
Localization Add-in User Guide

©2013
Copyright Portrait Software International Limited

All rights reserved. This document may contain confidential and proprietary information belonging to Portrait Software plc and/or its subsidiaries and associated companies.

Portrait Software, the Portrait Software logo, Portrait, Portrait Software's Portrait brand and Million Handshakes are the trademarks of Portrait Software International Limited and may not be used or exploited in any way without the prior express written authorization of Portrait Software International Limited.

Acknowledgement of trademarks

Other product names, company names, marks, logos and symbols referenced herein may be the trademarks or registered trademarks of their registered owners.

About Portrait Software

Portrait Software is now part of Pitney Bowes Software Inc..

Portrait Software enables organizations to engage with each of their customers as individuals, resulting in improved customer profitability, increased retention, reduced risk, and outstanding customer experiences. This is achieved through a suite of innovative, insight-driven applications which empower organizations to create enduring one-to-one relationships with their customers.

Portrait Software was acquired in July 2010 by Pitney Bowes to build on the broad range of capabilities at Pitney Bowes Software for helping organizations acquire, serve and grow their customer relationships more effectively. The Portrait Customer Interaction Suite combines world leading customer analytics, powerful inbound and outbound campaign management, and best-in-class business process integration to deliver real-time customer interactions that communicate precisely the right message through the right channel, at the right time.

Our 300 + customers include industry-leading organizations in customer-intensive sectors. They include 3, AAA, Bank of Tokyo Mitsubishi, Dell, Fiserv Bank Solutions, Lloyds Banking Group, Merrill Lynch, Nationwide Building Society, RACQ, RAC WA, Telenor, Tesco Bank, T-Mobile, Tryg and US Bank.

Pitney Bowes Software Inc. is a division of Pitney Bowes Inc. (NYSE: PBI).

For more information please visit: http://www.pitneybowes.co.uk/software/

UK

Portrait Software
The Smith Centre
The Fairmile
Henley-on-Thames
Oxfordshire, RG9 6AB, UK

Email: support@portraitsoftware.com
Tel: +44 (0)1491 416778
Fax: +44 (0)1491 416601

America

Portrait Software
125 Summer Street
16th Floor
Boston, MA 02110
USA

Email: support@portraitsoftware.com
Tel: +1 617 457 5200
Fax: +1 617 457 5299

Norway

Portrait Software
Portrait Million Handshakes AS
Maridalsveien. 87
0461 Oslo
Norway

Email: support@portraitsoftware.com
Tel: +47 22 38 91 00
Fax: +47 23 40 94 99

About this document

Purpose of document

This document explains the purpose of the **Portrait Localization Add-in** and the mechanism it employs for extracting resource strings from Portrait Web applications. It also describes the steps required to install and use the add-in.

Intended audience

Web developers and build controllers.

Related documents

Localizing .Net Applications User Guide (*Localizing_NET_Applications.pdf*)

Software release

Foundation Foundation 5.0 or later.

Contents

1 Introduction 6

2 Extraction process 7

3 Add-in installation 8

4 Using the Visual Studio Add-in 9

4.1 Configuring the Add-in 9

4.2 Running the Add-in 10

1 Introduction

Portrait Foundation Web applications and their related .NET components have been designed and implemented such that culture dependent text can be automatically extracted into separate resource catalogues. The extraction process is carried out by a Visual Studio automation tool that analyses the ASP.NET page implementation, extracts text resources and modifies the page to link its controls to the extracted resources.

This document outlines the mechanism used by the add-in for extracting the localizable text. It also provides a guide on how to use the add-in in a development environment.

It is recommended that the reader familiarise themselves with the guidelines provided in the Localizing .Net Applications User Guide.

Please refer to the release notes for the version of Microsoft Visual Studio that is currently supported.

2 Extraction process

⇒Note that the localization add-in is designed to only work with the controls provided as part of the SDK.

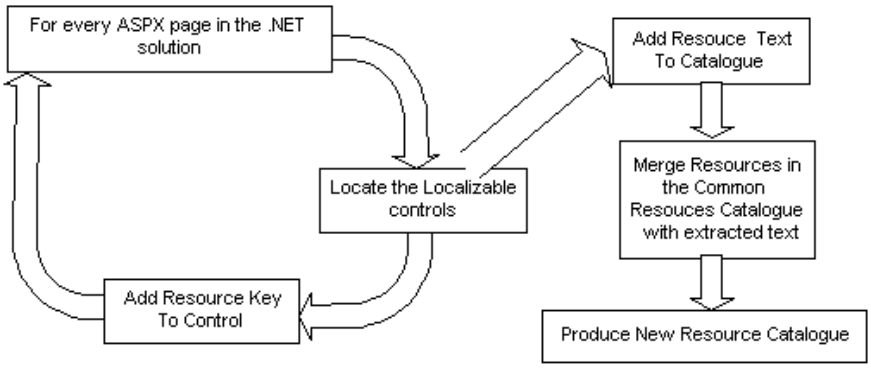
All Portrait ASP.NET implementations utilise the .NET controls provided in the SDK. These controls are all self-localizing in that they all retrieve localized text from a resource catalogue via unique identifiers (**ResourceKeys**) that indicate which strings to retrieve.

The generation of the resource catalogue and the setting of the controls' resource keys is carried out using the **Portrait Localization** Add-in.

The add-in is an automation tool that bolts onto Visual Studio and automates the development environment to retrieve text from the Web pages.

The Add-in's extraction mechanism is shown below

Figure 1 – Add-in extraction mechanism



The add-in is designed to consolidate resource files with the extracted text from the localizable controls. The resource files are used to store extra localizable strings that cannot be added using the SDK controls (for example information messages) and strings that have been extracted from the assemblies used by the Web project being localized (typically the assemblies associated with an included Portrait Foundation configuration package).

3 Add-in installation

In order to install the Portrait Localization Add-in, do the following:

- Close all instances of Visual Studio.
- Copy the "Software\Tools\Localization" folder from the release CD to your local drive.
- Open a command prompt and navigate to the local copy of the Localization folder.
- Run InstallMe.bat

Once the add-in is installed, two new **Portrait Localization** menu options should be available in Visual Studio's Tools menu.

4 Using the Visual Studio Add-in

In order for the add-in to run successfully, it has to be aware of certain information about the solution it is running on. The minimal amount of information required is as follows:

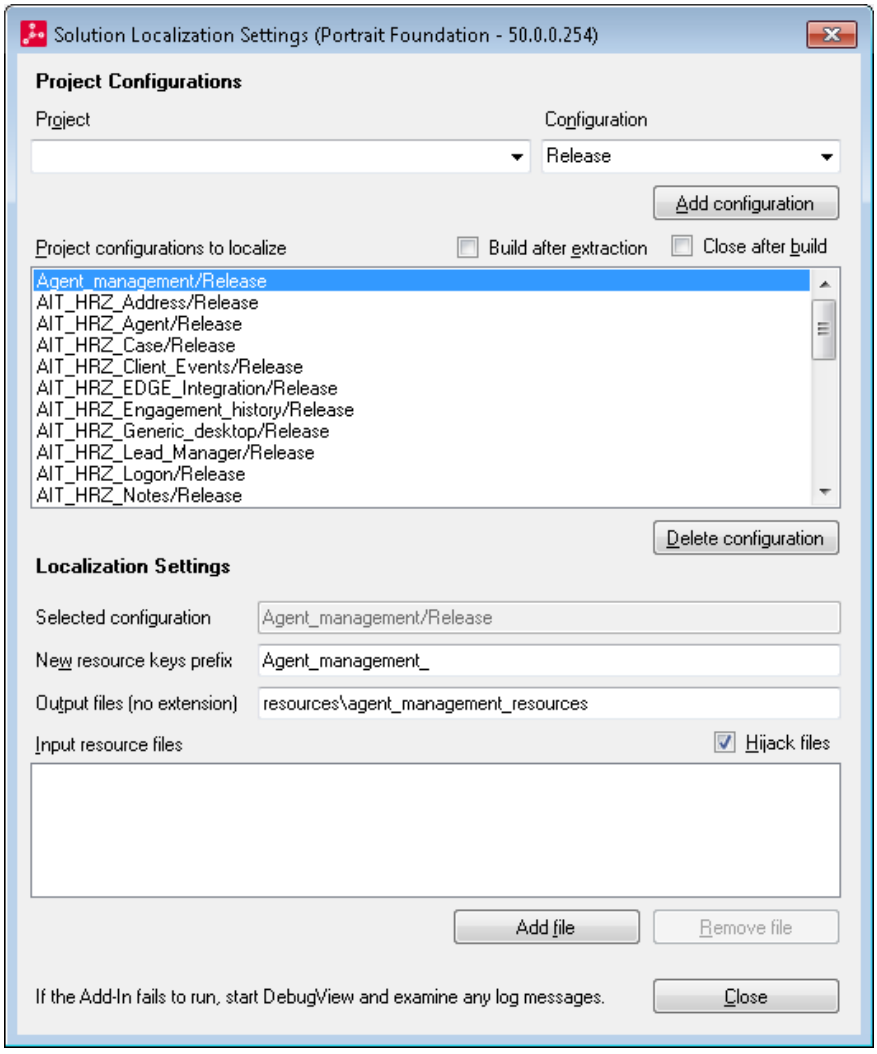
- Which projects inside the solution are to be localized? For every project, the following items have to be set.
  - Prefix for newly generated resource string keys
  - Output resource catalogue file

In most cases the default values can be used. To add all the projects in a solution simply select the desired **Configuration** (i.e. Release) and then click **Add configuration** without selecting a specific project.

4.1 Configuring the Add-in

The add-in retrieves all the information it requires from the .NET solution file. These settings can be changed by running **Portrait Localization Settings** from the Tools menu inside Visual Studio when the Portrait Foundation implementation project is loaded. The settings dialog box is shown in Figure 2.

Figure 2 – Portrait Localization Add-in settings dialog



The following section describes the different components in the dialog box.

4.1.1 Project specific settings

At least one configuration has to be added to the project configurations list in order for the add-in to work properly.

Each project that requires the add-in to run has to be added to the **Project configurations to localize** list box. This is achieved by selecting the project name and its appropriate configuration in the combo boxes and clicking the **Add configuration** button.

Each project/configuration pair can have the following options.

Input resource files are .NET resource files (RESX) to read and merge with the resource strings extracted from the web pages of the selected project. Typically the list contains a common resource file, which contains all special strings like message box texts, strings for common buttons like OK and Cancel, etc., and list of extracted .NET resource files for assemblies shipped with Portrait Foundation packages included into the Portrait Foundation configuration in question.

New resource keys prefix is a string that specifies a prefix for all newly generated entries in the output resource file. The Add-in parses the web pages of the selected project and extracts localizable strings to be add to the output resource file. Every entry in the resource file has to have a lookup key associated with it. That key is generated by the Add-in during the extration process and the specified prefix is used to ensure uniquenes of the key when merging with input resource files' entries.

Output files field indicates the name of the output resource files (without their extension) that will be produced once the extraction is finished successfully. The value of this setting contains a relative path of the files produced by the add-in. For example, if the value is set to **resources/commonresources** and the project files are in **c:\samplemortgage.net**, then the add-in will generate its output files in **c:\samplemortgage.net\resources** and these files will be named **commonresources**. Note that the add-in produces three different files when it is run. These are

- CAT format, which is a delimited file format that can be translated using a third party tool
- RESX format, which is the standard .NET resource catalogue format
- RESOURCES format, which is the binary resource catalogue that gets used by the application.

Hijack files is a flag that indicates whether the project files should have their read-only flags removed before the extraction process. This might be necessary if the files are under a source control system.

4.1.2 Solution specific settings

⇒These two options are usually used by build controllers who produce the final builds of the profiles.

Build after extraction and **Close after build** options are used in case the add-in is run from the command line rather from Visual Studio.NET menus. The **Close after build** attribute indicates that the add-in should close Visual Studio after extraction. **Build after extraction** is a flag to inform the add-in to run before any build process if the build configuration is in the **Configuration** list box. Both of these settings are needed if the add-in is run as part of a final build process.

4.1.3 Saving the add-in configuration

Clicking the OK button in the settings dialog box stores the settings in the solution file, which in turn can be saved from within Visual Studio.

4.2 Running the Add-in

Make sure the solution is open and every file of the project can be successfully opened with Visual Studio. Select **Portrait Localization** from the Tools menu. The add-in will run and generate the resource files as configured in the solution.