**ASNA Technical Design Document**

**Version 1.0**



eTouch Systems

6627 Dumbarton Circle

Fremont CA 94555

Phone: 510.795.4800

Fax: 510.795.4803

www.etouch.net

|  |
| --- |
| **Synopsis** |
| This is the Technical Design Document for modernization AS/400 green screens to Web UI using ASNA and customization using JavaScript/jQuery |

|  |  |
| --- | --- |
| **Based on** |  |
| **Replaces** |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision History** | | | |
| **Version** | **Date** | **Author** | **Revision Comments** |
| 1.0 | 04 Jan 2016 | eTouch Systems |  |
|  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **Approvals** | | |
| **Name** | **Project Role** | **Approval Date** |
| <<TBD>> |  |  |
|  |  |  |
|  |  |  |

**Table of Contents**

[1. Introduction 4](#_Toc440391599)

[2. Products Used for Modernization 5](#_Toc440391600)

[3. UI Developer environment 6](#_Toc440391601)

[4. Conversion using ASNA Wings 7](#_Toc440391602)

[5. How ASNA Wings works(Flow) 11](#_Toc440391603)

[6. Flow from AS/400 to browser 12](#_Toc440391604)

[7. Flow from Browser to AS/400 13](#_Toc440391605)

[8. UI modernization and development specifications 14](#_Toc440391606)

[9. Modifying the Sign-on Logic 16](#_Toc440391607)

[10. Deployment Diagram 17](#_Toc440391608)

# Introduction

This technical design document provides technical details of AS/400 green screen conversion process and flow using ASNA Wings. Also it states about the environment, products, software required for modernization and UI developers.

# Products Used for Modernization

* 1. Modernization environment

These will be installed on the PC that will be used for conversion:

Visual Studio 2012

ASNA Wings Design Aid

This is the Windows client component of Wings. Upon installation, the Wings Design Aid tightly integrates itself with Visual Studio. During application modernization, you will use the Wings Design Aid exclusively inside Visual Studio since Wings does not have its own user interface.

Web Browser

Wings requires Windows Internet Explorer 8 or higher, Mozilla Firefox 3.6 or higher, or Google Chrome 8.0 or higher.

ASNA Monarch Collector

* 1. Software requirements for deployment/testing/Production

IBMi V6R1 or higher

IBM Open Access RPG Edition

ASNA DataGate for IBMi

Windows Server version 2012

ASNA DataGateWebpak

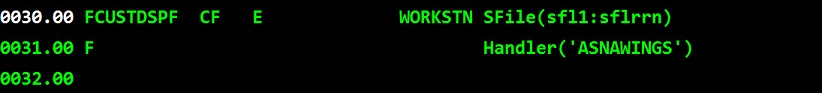
# UI Developer environment

Visual Studio 2012

Web Browser

# Conversion using ASNA Wings

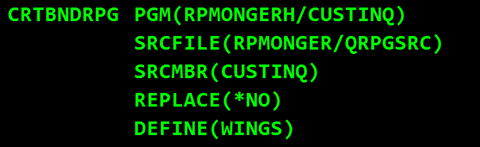
* 1. Adding Wings handler to RPG ILE program without condition:
     1. However, many Wings customers want their users to be able to use either the Wings version of the application or to be able to use the original green-screen version of the application. Wings can accommodate that by using the same source code to create a Wings-ready program objects while also retaining the ability to create the original green screen program objects. ILE RPG's conditional compilation option is the recommended way to do this.
     2. Setting up a RPG program to work with Wings is a matter of adding a single continuation line. All you need to do is insert Handler ('ASNAWINGS') into to the WORKSTN file declaration in the ILE RPG program.



* + 1. Re-Compile the RPG Program
  1. Adding Wings handler to RPG ILE program with conditional compilation:
     1. The ILE RPG compiler can conditionally include code based on compilation “conditions.” These conditions are specified with compile commands such as [CRGBNDRPG](https://www-01.ibm.com/support/knowledgecenter/ssw_i5_54/cl/crtbndrpg.htm) using the DEFINE keyword. Conditional compilation, along with the ability to target a specific object output library, provides the basis of the scheme to create two program objects from a single ILE RPG source member; one for use with Wings and one for use with the 5250 display file.
     2. Preparing an application to be conditionally compiled as a Wings-ready program object is a matter of adding three lines of code as shown below in lines 31-33.

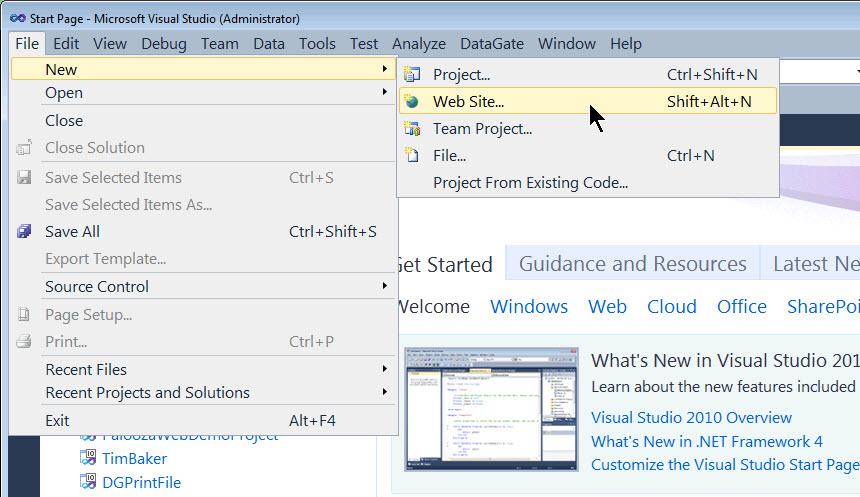


* + 1. Any program with the above logic included can be compiled as Wings-ready using (as an example) [CRTBNDRPG](https://www-01.ibm.com/support/knowledgecenter/ssw_i5_54/cl/crtbndrpg.htm), as shown in below figure. Notice the “WINGS” condition described with the DEFINE keyword; or the source can be compiled without the WINGS condition to create a non-Wings program.

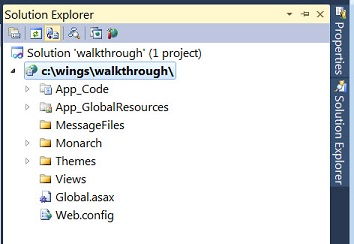


* 1. Start the Visual Studio and create new ASNA Wings Website using

*File->New->Website->ASNA Wings Web Site*

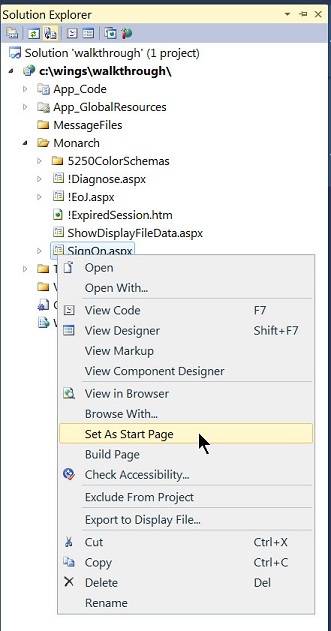


* 1. The Visual Studio Solution Explorer window (*View->Solution Explorer*) shows you the current Web site contents (which include all of its files and folders)

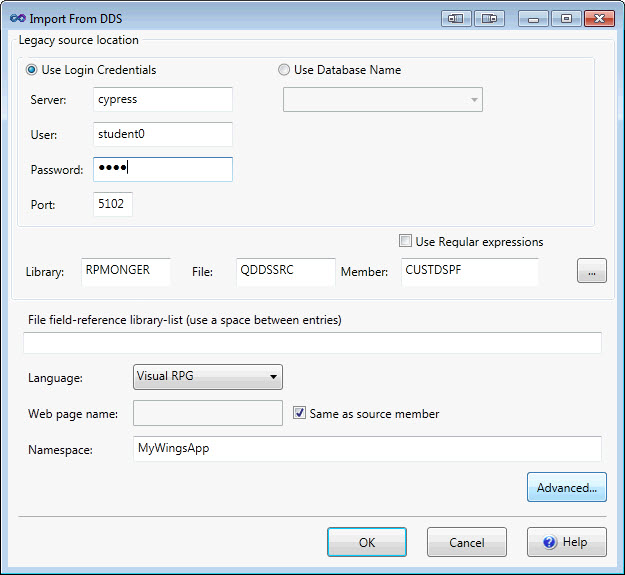


* 1. To set SignOn page as a default application page,

Explore the *Monarch folder->Right click on the SignOn.aspx->Set As Start Page*.

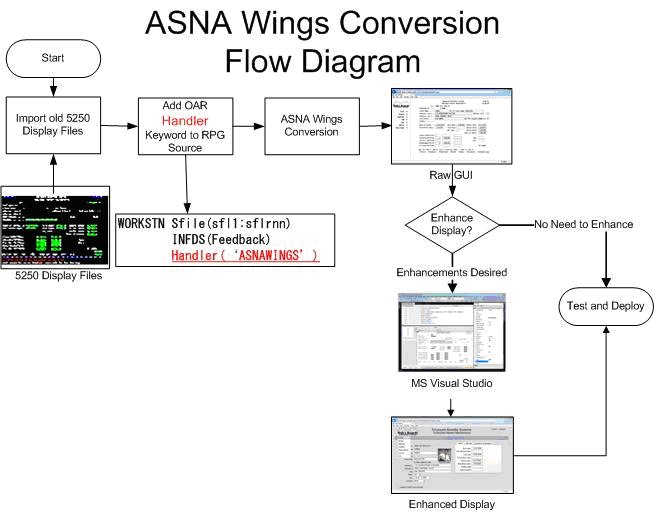


* 1. To import DDS Right click on the view folder in the created website, in the "Import From DDS Dialog" fill in the login details and fill in the other fields as shown in the below image
     1. Library – <RPG Library Name - Same as Directory in windows>
     2. File – <RPG Display File>
     3. Member – <RPG Program>
     4. Language – Visual RPG/C#/VB
     5. Web page name – Select the checkbox for Same as source member
     6. Namespace – <Namespace e.g. MyWingsAppimport>



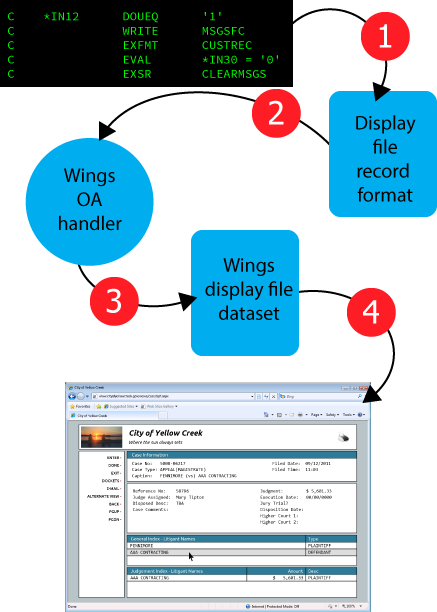
* 1. Now you are ready to run website using VS built in web server(No need to install any additional web server at this phase)

# How ASNA Wings works(Flow)



# Flow from AS/400 to browser

A wing uses IBM's Open Access to intercept RPG workstation file data. This intercepted data is then sent to the Wings presentation layer, which was derived from the DDS display file source definition. The four primary steps that move the workstation file to the Wings UI are shown below.

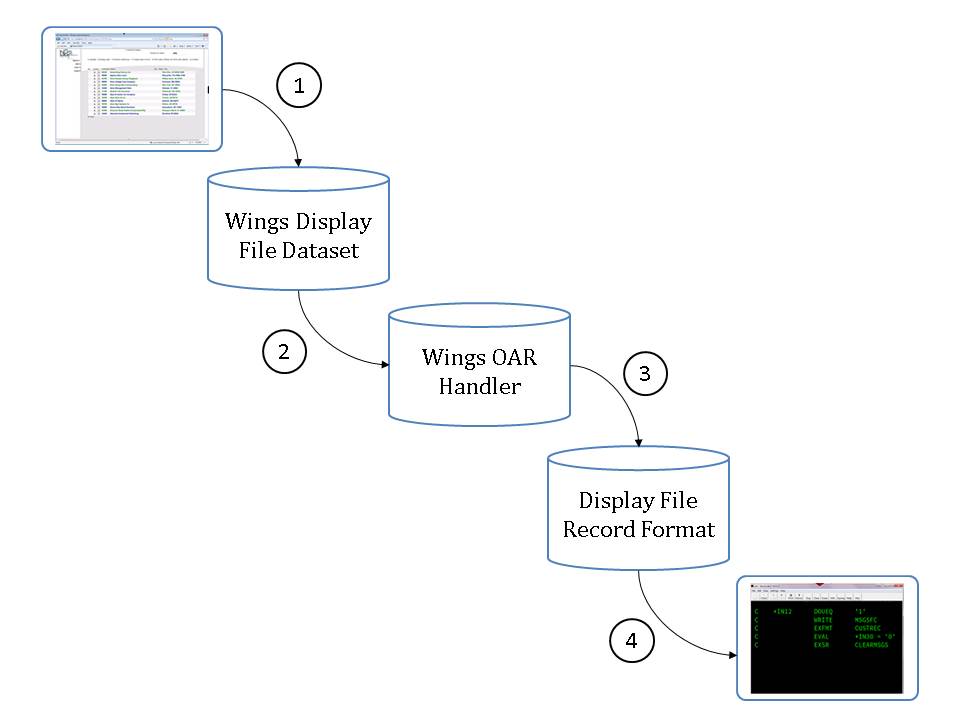
1. The RPG program writes to a display file record format--just as it always has. You don't change any logic in your RPG to use Wings. This results in a populated display file record format ready to send to a display file. At this step, the RPG has no knowledge that this record format data will be redirected to the Wings display.
2. In normal circumstances, the display file record format data is passed to the workstation controller in Step 2. However, the IBM i OS knows that the Wings Open Access handler is registered with the RPG program which causes IBM i to pass the display file record format data to the Wings Open Access handler.
3. The Wings Open Access handler translates the raw display file record format data into a Wings display file dataset. Unlike typical screen scrapers, Wings does not emit a 5250 data stream. The Wings dataset contains all of the fields in the display file record format, hidden fields, and all 99 indicators.
4. The Wings browser page receives the Wings display dataset and scatters the field values to the appropriate elements on the page. DDS rules, generally driven by indicator values, are enforced. For example, if in the original display indicator 40 caused a given field to have the cursor position, that same field will have focus with Wings.

# Flow from Browser to AS/400

* 1. Wings mimics IBM i function keys smoothly into the Window’s world by providing an on-screen buttons that corresponds to each enabled function key by retaining their functionality.
  2. So when user clicks a particular Wings UIbutton it will perform same action as F-key invoked on the AS/400
  3. Below diagram/flow explains end to end to flow from Browser action(click) to AS/400(RPG Program)
     1. The Wings display file dataset is captured from the user by the modernized UI
     2. Wings display file dataset is passed to the Wings Open Access handler
     3. The Wings Open Access handler translates the Wings display file dataset into a raw

Display file record format data

* + 1. Raw display file record format passed to the RPG program



# UI modernization and development specifications

* 1. Adding HTML/CSS
  + New HTML elements and components will be added in the process of modernization adhering HTML5 W3C standards e.g.HTML tables, buttons etc.
  + Custom CSS2/CSS3 will be added in separate stylesheet file in the process of modernization adhering CSS2/CSS3 coding standards
  + Modification in the default ASNA Wings CSS e.g. To rearrange the action buttons provided by ASNA Wings (if required)
  + New JavaScript/jQuery code for event handling
  + New JavaScript /jQuery for DOM manipulations wherever required
  + New JavaScript/jQuery plugins will be incorporated in the process of modernization for example:
    1. Bootstrap jQuery plugin for responsive design
    2. Bootstrap jQuery for pop-ups
  1. Green Screen F-keys mapping with UI Action Buttons

Use Case:

If F6=Createopens a new customer entry in the green screen, how this will be implemented as the “Create Customer” button using modernized UI

Approach:

* + ASNA Wings enablessame function keys from green screen to perform the same F Key operation through modernized UI
  + Same JavaScript code provided by ASNA Wings for Fkey F6 will be used on a “click”

Event.

For Example “Create Customer” button, Code with jQuery will be:

$(“#create-customer”).click(function() {

//Call ASNA Wings f-key F6 JS code/JS function

})

* 1. Infinite scrolling(Content on the page continually load as a user navigates down the page)

Plugin sample:

* + **jScroll** – jQuery Plugin for Infinite Scrolling / Auto-Paging

Example Code:

The HTML (abbreviated)

<div class="scroll">

.

.

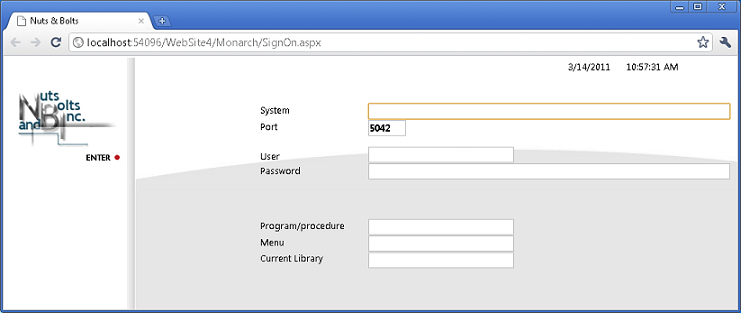
.

</div>

The JavaScript for implementing this is:

$('.scroll').jscroll();

# Modifying the Sign-on Logic

* 1. The default sign-on screen provided by [ASNA Wings Design Aid](http://devnet.asna.com/documentation/Help110/Wings/_HTML/WingsDesignAid.htm) looks like following,
  2. The display file and logic provided with the [ASNA Wings Design Aid](http://devnet.asna.com/documentation/Help110/Wings/_HTML/WingsDesignAid.htm) may be functional for the programmer in a testing environment but it does not provide restricted access to the other servers that may not be available to all users and it can be cumbersome for most daily users in a runtime environment. To eliminate both of these issues the sign on display and logic can be changed in following way,

The variablesand values for vairables System, Port, Program/procedure, Menu and Current Library will be stored in the **web.config** which will be pre-loaded in the initilization phase of the application.

* + 1. Following variables values will be put in the **web.config** file,
* **System**
* The IBM i server name containing the RPG/ILE programs with the ASNA Wings Design Aid if in modernization mode, or the server that is used for production after your application has been deployed.
* **Port**
* The port number of the server connection. The default is 5042.
* **Program/procedure and Menu**
* The cl program or procedure you want to execute or the menu program.
* **Current Library**
* The library the program, procedure, or menu is located in. This would be the library containing the RPG ILE programs that have the Handler keyword.

# Deployment Diagram

<<TBD>> will be updated when the environment details are available.