



OnWeb 7.5.0

Migration Guide

Micro Focus (IP) Ltd.
The Lawn
22-30 Old Bath Road
Newbury, Berkshire RG14 1QN
UK
<http://www.microfocus.com>

Copyright 2010 Micro Focus (IP) Limited. All Rights Reserved.

MICRO FOCUS, the Micro Focus logo and RUMBA are trademarks or registered trademarks of Micro Focus (IP) Limited or its subsidiaries or affiliated companies in the United States, United Kingdom and other countries.

All other marks are the property of their respective owners.

Table of Contents

Chapter 1: Overview

Chapter 2: Migration Issues

Migration scenarios • 7

Migrating to OnWeb 7.x • 7

Migrating to OnWeb 6.x/7.x • 7

Migrating to OnWeb 5.2.4 • 9

Migrating to OnWeb 5.2.3 • 10

Migrating to OnWeb 5.2.1 • 12

Migrating to OnWeb 5.2.0 • 13

Migrating to OnWeb 5.1.1 • 14

Migrating to OnWeb 5.1.0 • 14

Migrating to OnWeb 5.0.0 • 14

Issues in Detail • 15

Updating customized hpstyles.css and kbhandl.js files • 15

Previously created Web services are not supported • 22

Modifying Refine script in the Business Logic object • 22

Changing location of the SalvoInit.srx file • 22

Scripts written in VBScript will not run • 22

Using 5250 emulation • 23

Preserving objects and HTML files from the previous version • 23

Designer.mdb updated to a new format • 23

Upgrading Host Publishing applications • 24

Upgrading Host Integration applications • 25

Importing User Services objects • 25

Back up HostProfile.ini • 25

Using the closing </FORM> tag • 25

Using the WaitFor function with VT emulation • 26



Appendix A: 3270 Emulation Upgrade

- Configuring the emulator version for OnWeb Server • 28
 - Changes in 3270 emulator version 2.0 • 28
 - Colour attributes • 28
 - Autoskip attributes • 29
 - Hidden and Protected attributes • 29
 - ReadScreen function • 29
 - Type function • 30
 - Press("Clear") function • 30
 - KeyboardLocked function • 30
-

Index • 31



This guide describes the process of migrating your Salvo® Impact® applications to OnWeb® and various issues related to upgrading your applications from one version of OnWeb to the next.

[Chapter 2, “Migration Issues”](#) describes issues related to migrating applications from Salvo to OnWeb and between OnWeb versions.

[Appendix A, “3270 Emulation Upgrade”](#) describes how recent upgrades to the OnWeb Server may affect scripts which interact with the 3270 terminal.

To learn how to use Designer, consult Designer online help and *OnWeb Developer’s Guide*.

Note: In this guide, the name “Designer” refers to OnWeb Designer.



This chapter describes various issues you should take into consideration when moving your applications either from Salvo to OnWeb or from one version of OnWeb to the next. In most cases this migration is transparent and does not need any adjustments on your part. However, because in some versions of OnWeb we've introduced new features and improvements to the existing features, you may need to make slight adjustments to your applications.

Migration scenarios

Relevant migration issues are listed under each migration scenario and are linked to its detailed description in the Issues in Detail section.

Migrating to OnWeb 7.x

OnWeb 7.2.2 to OnWeb 7.2.3

See “Updating customized hpstyles.css and kbhandl.js files” on page 15.

OnWeb 7.1 to OnWeb 7.2

See “Previously created Web services are not supported” on page 22.

Migrating to OnWeb 6.x/7.x

OnWeb 5.2.4 to OnWeb 6.x/7.x

See “Importing User Services objects” on page 25.

See “Using the closing </FORM> tag” on page 25.

See “Using the WaitFor function with VT emulation” on page 26.

OnWeb 5.2.3 to OnWeb 6.x/7.x

See “Importing User Services objects” on page 25.

See “Using the WaitFor function with VT emulation” on page 26.

OnWeb 5.2.1 to OnWeb 6.x/7.x

See “Importing User Services objects” on page 25.

See “Using the WaitFor function with VT emulation” on page 26.

OnWeb 5.2.0 to OnWeb 6.x/7.x

See “Importing User Services objects” on page 25.

See “Using the WaitFor function with VT emulation” on page 26.

OnWeb 5.1.1 to OnWeb 6.x/7.x

See “Changing location of the SalvoInit.srx file” on page 22.

See “Upgrading Host Publishing applications” on page 24.

See “Upgrading Host Integration applications” on page 25.

See “Importing User Services objects” on page 25.

See “Using the WaitFor function with VT emulation” on page 26.

OnWeb 5.1.0 to OnWeb 6.x/7.x

See “Changing location of the SalvoInit.srx file” on page 22.

See “Upgrading Host Publishing applications” on page 24.

See “Upgrading Host Integration applications” on page 25.

See “Importing User Services objects” on page 25.

See “Using the WaitFor function with VT emulation” on page 26.



OnWeb 5.0.0 to OnWeb 6.x/7.x

See “Changing location of the SalvoInit.srx file” on page 22.

See “Preserving objects and HTML files from the previous version” on page 23.

See “Upgrading Host Publishing applications” on page 24.

See “Importing User Services objects” on page 25.

See “Using the WaitFor function with VT emulation” on page 26.

Salvo 4.1 to OnWeb 6.x/7.x

See “Changing location of the SalvoInit.srx file” on page 22.

See “Scripts written in VBScript will not run” on page 22.

See “Using 5250 emulation” on page 23.

See “Using the WaitFor function with VT emulation” on page 26.

Migating to OnWeb 5.2.4

OnWeb 5.2.3 to OnWeb 5.2.4

See “Using the closing </FORM> tag” on page 25.

See “Using the WaitFor function with VT emulation” on page 26.

OnWeb 5.2.1 to OnWeb 5.2.4

See “Using the WaitFor function with VT emulation” on page 26.

OnWeb 5.2.0 to OnWeb 5.2.4

See “Using the WaitFor function with VT emulation” on page 26.

OnWeb 5.1.1 to OnWeb 5.2.4

See “Changing location of the SalvoInit.srx file” on page 22.

See “Upgrading Host Publishing applications” on page 24.

See “Upgrading Host Integration applications” on page 25.

See “Back up HostProfile.ini” on page 25.

See “Using the WaitFor function with VT emulation” on page 26.

OnWeb 5.1.0 to OnWeb 5.2.4

See “Changing location of the SalvoInit.srx file” on page 22.

See “Upgrading Host Publishing applications” on page 24.

See “Upgrading Host Integration applications” on page 25.

See “Using the WaitFor function with VT emulation” on page 26.

OnWeb 5.0.0 to OnWeb 5.2.4

See “Changing location of the SalvoInit.srx file” on page 22.

See “Preserving objects and HTML files from the previous version” on page 23.

See “Upgrading Host Publishing applications” on page 24.

See “Using the WaitFor function with VT emulation” on page 26.

Salvo 4.1 to OnWeb 5.2.4

See “Changing location of the SalvoInit.srx file” on page 22.

See “Scripts written in VBScript will not run” on page 22.

See “Using 5250 emulation” on page 23.

See “Using the WaitFor function with VT emulation” on page 26.

Migrating to OnWeb 5.2.3

OnWeb 5.2.1 to OnWeb 5.2.3

See “Using the WaitFor function with VT emulation” on page 26.



OnWeb 5.2.0 to OnWeb 5.2.3

See “Using the WaitFor function with VT emulation” on page 26.

OnWeb 5.1.1 to OnWeb 5.2.3

See “Changing location of the SalvoInit.srx file” on page 22.

See “Upgrading Host Publishing applications” on page 24.

See “Upgrading Host Integration applications” on page 25.

See “Back up HostProfile.ini” on page 25.

See “Using the WaitFor function with VT emulation” on page 26.

OnWeb 5.1.0 to OnWeb 5.2.3

See “Changing location of the SalvoInit.srx file” on page 22.

See “Upgrading Host Publishing applications” on page 24.

See “Upgrading Host Integration applications” on page 25.

See “Using the WaitFor function with VT emulation” on page 26.

OnWeb 5.0.0 to OnWeb 5.2.3

See “Changing location of the SalvoInit.srx file” on page 22.

See “Preserving objects and HTML files from the previous version” on page 23.

See “Upgrading Host Publishing applications” on page 24.

See “Using the WaitFor function with VT emulation” on page 26.

Salvo 4.1 to OnWeb 5.2.3

See “Changing location of the SalvoInit.srx file” on page 22.

See “Scripts written in VBScript will not run” on page 22.

See “Using 5250 emulation” on page 23.

See “Using the WaitFor function with VT emulation” on page 26.

Migrating to OnWeb 5.2.1

OnWeb 5.2.0 to OnWeb 5.2.1

See “Using the WaitFor function with VT emulation” on page 26.

OnWeb 5.1.1 to OnWeb 5.2.1

See “Changing location of the SalvoInit.srx file” on page 22.

See “Upgrading Host Publishing applications” on page 24.

See “Upgrading Host Integration applications” on page 25.

See “Back up HostProfile.ini” on page 25.

See “Using the WaitFor function with VT emulation” on page 26.

OnWeb 5.1.0 to OnWeb 5.2.1

See “Changing location of the SalvoInit.srx file” on page 22.

See “Upgrading Host Publishing applications” on page 24.

See “Upgrading Host Integration applications” on page 25.

See “Using the WaitFor function with VT emulation” on page 26.

OnWeb 5.0.0 to OnWeb 5.2.1

See “Changing location of the SalvoInit.srx file” on page 22.

See “Preserving objects and HTML files from the previous version” on page 23.

See “Upgrading Host Publishing applications” on page 24.

See “Using the WaitFor function with VT emulation” on page 26.

Salvo 4.1 to OnWeb 5.2.1

See “Changing location of the SalvoInit.srx file” on page 22.

See “Scripts written in VBScript will not run” on page 22.



See “Using 5250 emulation” on page 23.

See “Using the WaitFor function with VT emulation” on page 26.

Migrating to OnWeb 5.2.0

OnWeb 5.1.1 to OnWeb 5.2.0

See “Changing location of the SalvoInit.srx file” on page 22.

See “Upgrading Host Integration applications” on page 25.

See “Using the WaitFor function with VT emulation” on page 26.

OnWeb 5.1.0 to OnWeb 5.2.0

See “Changing location of the SalvoInit.srx file” on page 22.

See “Upgrading Host Integration applications” on page 25.

See “Using the WaitFor function with VT emulation” on page 26.

OnWeb 5.0.0 to OnWeb 5.2.0

See “Changing location of the SalvoInit.srx file” on page 22.

See “Preserving objects and HTML files from the previous version” on page 23.

See “Using the WaitFor function with VT emulation” on page 26.

Salvo 4.1 to OnWeb 5.2.0

See “Changing location of the SalvoInit.srx file” on page 22.

See “Scripts written in VBScript will not run” on page 22.

See “Using 5250 emulation” on page 23.

See “Using the WaitFor function with VT emulation” on page 26.

Migrating to OnWeb 5.1.1

OnWeb 5.0.0 to OnWeb 5.1.1

See “Preserving objects and HTML files from the previous version” on page 23.

See “Designer.mdb updated to a new format” on page 23.

Salvo 4.1 to OnWeb 5.1.1

See “Scripts written in VBScript will not run” on page 22.

See “Using 5250 emulation” on page 23.

Migrating to OnWeb 5.1.0

OnWeb 5.0.0 to OnWeb 5.1.0

See “Modifying Refine script in the Business Logic object” on page 22.

See “Preserving objects and HTML files from the previous version” on page 23.

Salvo 4.1 to OnWeb 5.1.0

See “Modifying Refine script in the Business Logic object” on page 22.

See “Scripts written in VBScript will not run” on page 22.

See “Using 5250 emulation” on page 23.

Migrating to OnWeb 5.0.0

Salvo 4.1 to OnWeb 5.0.0

See “Scripts written in VBScript will not run” on page 22.

See “Using 5250 emulation” on page 23.

Issues in Detail

Updating customized hpstyles.css and kbhandl.js files

An application created with OnWeb 7.2.2 may display the inserted screen incorrectly when you upgrade to OnWeb 7.2.3. This problem will occur only if you've used the "version=2.0" attribute in the OW:INSERT-SCREEN-HERE tag.

Because this problem is caused by the modifications introduced in OnWeb 7.2.3 to two files, hpstyles.css and kbhandl.js, these two files must be updated, using one of the following scenarios:

	hpstyles.css not customized	hpstyles.css customized
kbhandl.js not customized	follow Procedure 1	follow Procedure 2
kbhandl.js customized	follow Procedure 3	follow Procedure 4

Procedure 1

1. Open your application in OnWeb Designer.
2. Open the Components white-board for your application.
3. Right-click the hpstyles.css object and choose **Reset** from the menu.
4. Right-click the kbhandl.js object and choose **Reset** from the menu.
5. In the tree list, right-click the application name and choose **Rebuild Item**.

Procedure 2

1. Reset the kbhandl.js file as described in [Procedure 1](#).
2. In order to preserve your custom changes in the hpstyles.css file, manually update this file, as follow:

a) Insert the following classes after the `.hpinutgs` class:

```

/*
    The following classes are applied to unprotected fields
    generated by OW:INSERT-SCREEN-HERE tag version 2.0.
*/
/* Mandatory enter */
.hpinutgs_md {font-family: courier new; padding: 0;
line-height: 1.1em;height: 1.2em;background-color: black;
border-left-width:0px; border-right-width:0px; border-
top-width:0px; border-bottom-width:1px; border-style:
none; }
/* Mandatory fill */
.hpinutgs_mf {font-family: courier new; padding: 0;
line-height: 1.1em;height: 1.2em;background-color: black;
border-left-width:0px; border-right-width:0px; border-
top-width:0px; border-bottom-width:1px; border-style:
none; }
/* alphanumeric */
.hpinutgs_alphanumeric {font-family: courier new;
padding: 0; line-height: 1.1em;height: 1.2em;background-
color: black; border-left-width:0px; border-right-
width:0px; border-top-width:0px; border-bottom-width:1px;
border-style: none; }
/* numeric */
.hpinutgs_numeric {font-family: courier new; padding: 0;
line-height: 1.1em;height: 1.2em;background-color: black;
border-left-width:0px; border-right-width:0px; border-
top-width:0px; border-bottom-width:1px; border-style:
none; }
/* digits only */
.hpinutgs_digits {font-family: courier new; padding: 0;
line-height: 1.1em;height: 1.2em;background-color: black;
border-left-width:0px; border-right-width:0px; border-
top-width:0px; border-bottom-width:1px; border-style:
none; }
/* monospace field */
.field_monospace {text-transform: uppercase;}

```



b) Insert the following classes after the `.unprotected_reversemonochrome` class:

```

/*
The following classes are used when generating the color
information of unprotected fields by using the
OW:INSERT-SCREEN-HERE tag version 2.0.
This enables the designer to change the actual color
presented in the html browser.
*/
/* these are for Mandatory fill field */
.unprotected_default_mf {color:white;border-bottom-
color:white;}
.unprotected_blue_mf {color:blue;border-bottom-
color:blue;}
.unprotected_red_mf {color:red;border-bottom-color:red;}
.unprotected_pink_mf {color:pink;border-bottom-
color:pink;}
.unprotected_turquoise_mf {color:aqua;border-bottom-
color:aqua;}
.unprotected_green_mf {color:lime;border-bottom-
color:lime;}
.unprotected_yellow_mf {color:yellow;border-bottom-
color:yellow;}
.unprotected_white_mf {color:white;border-bottom-
color:white;}
.unprotected_monochrome_mf {color:white;border-bottom-
color:white;}
.unprotected_reverse_default_mf {color:black; background-
color:white;border-bottom-color:black;}
.unprotected_reverse_blue_mf {color:yellow; background-
color:blue;border-bottom-color:yellow;}
.unprotected_reverse_red_mf {color:lime; background-
color:red;border-bottom-color:lime;}
.unprotected_reverse_pink_mf {color:blue; background-
color:pink;border-bottom-color:blue;}
.unprotected_reverse_turquoise_mf {color:yellow;
background-color:aqua;border-bottom-color:yellow;}
.unprotected_reverse_green_mf {color:red; background-
color:lime;border-bottom-color:red;}
.unprotected_reverse_yellow_mf {color:blue; background-
color:yellow;border-bottom-color:blue;}

```

```
.unprotected_reverse_white_mf {color:black; background-  
color:white;border-bottom-color:black;}  
.unprotected_reverse_monochrome_mf {color:black;  
background-color:white;border-bottom-color:black;}  
/* these are for Mandatory enter field */  
.unprotected_default_md {color:white;border-bottom-  
color:white;}  
.unprotected_blue_md {color:blue;border-bottom-  
color:blue;}  

```



```

.unprotected_default_numeric {color:white;border-bottom-
color:white;}

.unprotected_blue_numeric {color:blue;border-bottom-
color:blue;}

.unprotected_red_numeric {color:red;border-bottom-
color:red;}

.unprotected_pink_numeric {color:pink;border-bottom-
color:pink;}

.unprotected_turquoise_numeric {color:aqua;border-bottom-
color:aqua;}

.unprotected_green_numeric {color:lime;border-bottom-
color:lime;}

.unprotected_yellow_numeric {color:yellow;border-bottom-
color:yellow;}

.unprotected_white_numeric {color:white;border-bottom-
color:white;}

.unprotected_monochrome_numeric {color:white;border-
bottom-color:white;}

.unprotected_reverse_default_numeric {color:black;
background-color:white;border-bottom-color:black;}

.unprotected_reverse_blue_numeric {color:yellow;
background-color:blue;border-bottom-color:yellow;}

.unprotected_reverse_red_numeric {color:lime; background-
color:red;border-bottom-color:lime;}

.unprotected_reverse_pink_numeric {color:blue;
background-color:pink;border-bottom-color:blue;}

.unprotected_reverse_turquoise_numeric {color:yellow;
background-color:aqua;border-bottom-color:yellow;}

.unprotected_reverse_green_numeric {color:red;
background-color:lime;border-bottom-color:red;}

.unprotected_reverse_yellow_numeric {color:blue;
background-color:yellow;border-bottom-color:blue;}

.unprotected_reverse_white_numeric {color:black;
background-color:white;border-bottom-color:black;}

.unprotected_reverse_monochrome_numeric {color:black;
background-color:white;border-bottom-color:black;}

/* these are for digits only field */

.unprotected_default_digits {color:white;border-bottom-
color:white;}

.unprotected_blue_digits {color:blue;border-bottom-
color:blue;}

```

```
.unprotected_red_digits {color:red;border-bottom-  
color:red;}  
.unprotected_pink_digits {color:pink;border-bottom-  
color:pink;}  
.unprotected_turquoise_digits {color:aqua;border-bottom-  
color:aqua;}  
.unprotected_green_digits {color:lime;border-bottom-  
color:lime;}  
.unprotected_yellow_digits {color:yellow;border-bottom-  
color:yellow;}  
.unprotected_white_digits {color:white;border-bottom-  
color:white;}  
.unprotected_monochrome_digits {color:white;border-  
bottom-color:white;}  
.unprotected_reverse_default_digits {color:black;  
background-color:white;border-bottom-color:black;}  
.unprotected_reverse_blue_digits {color:yellow;  
background-color:blue;border-bottom-color:yellow;}  
.unprotected_reverse_red_digits {color:lime; background-  
color:red;border-bottom-color:lime;}  
.unprotected_reverse_pink_digits {color:blue; background-  
color:pink;border-bottom-color:blue;}  
.unprotected_reverse_turquoise_digits {color:yellow;  
background-color:aqua;border-bottom-color:yellow;}  
.unprotected_reverse_green_digits {color:red; background-  
color:lime;border-bottom-color:red;}  
.unprotected_reverse_yellow_digits {color:blue;  
background-color:yellow;border-bottom-color:blue;}  
.unprotected_reverse_white_digits {color:black;  
background-color:white;border-bottom-color:black;}  
.unprotected_reverse_monochrome_digits {color:black;  
background-color:white;border-bottom-color:black;}  
/* these are for alphanumeric field */  
.unprotected_default_alphanumeric {color:white;border-  
bottom-color:white;}  
.unprotected_blue_alphanumeric {color:blue;border-bottom-  
color:blue;}  

```



```

.unprotected_turquoise_alphanumeric {color:aqua;border-
bottom-color:aqua;}

.unprotected_green_alphanumeric {color:lime;border-
bottom-color:lime;}

.unprotected_yellow_alphanumeric {color:yellow;border-
bottom-color:yellow;}

.unprotected_white_alphanumeric {color:white;border-
bottom-color:white;}

.unprotected_monochrome_alphanumeric {color:white;border-
bottom-color:white;}

.unprotected_reverse_default_alphanumeric {color:black;
background-color:white;border-bottom-color:black;}

.unprotected_reverse_blue_alphanumeric {color:yellow;
background-color:blue;border-bottom-color:yellow;}

.unprotected_reverse_red_alphanumeric {color:lime;
background-color:red;border-bottom-color:lime;}

.unprotected_reverse_pink_alphanumeric {color:blue;
background-color:pink;border-bottom-color:blue;}

.unprotected_reverse_turquoise_alphanumeric
{color:yellow; background-color:aqua;border-bottom-
color:yellow;}

.unprotected_reverse_green_alphanumeric {color:red;
background-color:lime;border-bottom-color:red;}

.unprotected_reverse_yellow_alphanumeric {color:blue;
background-color:yellow;border-bottom-color:blue;}

.unprotected_reverse_white_alphanumeric {color:black;
background-color:white;border-bottom-color:black;}

.unprotected_reverse_monochrome_alphanumeric
{color:black; background-color:white;border-bottom-
color:black;}

```

Procedure 3

1. Reset the hpstyles.css file as described in [Procedure 1](#).
2. Contact Micro Focus® Technical Support for assistance in updating the kbhandl.js file.

Procedure 4

1. Update the hpstyles.css file as described in [Procedure 2](#).
2. Contact Micro Focus Technical Support for assistance in updating the kbhandl.js file.

Previously created Web services are not supported

If you have published a Web service using OnWeb Server (Multiplatform) and have created an application that consumes this Web service, you have to recreate the Web service using OnWeb Object Builder and modify your application according to the new WSDL format. Previous versions of Web services are not supported because of the improvements in handling tables.

Modifying Refine script in the Business Logic object

If you have a Business Logic object in which the Information object is passed from one Refine script to another, the Refine script of the calling Business Logic object may have to be modified. Specifically, if the Refine script that has an Information object as an input parameter is written in VBScript and if the Refine script of the calling Business Logic object references any IObject by index, modify the Refine script of the calling Business Logic object to increase by one the index value when the IObject is referenced in the case where there are two Refine scripts.

Changing location of the SalvoInit.srx file

SalvoInit.srx is a user modifiable REXX startup script. It is located in the Server\Scripts\Source directory and is executed during startup. A new rebranded script called OnWebSystem.wpc checks for the old SalvoInit.srx script in the Server\Scripts\System directory and calls it if found. If you are using this startup script, you must manually move the SalvoInit.srx file from the Source directory to the System directory.

Scripts written in VBScript will not run

If you install OnWeb Server 5.x or later on a system that has Salvo 4.0.x or 4.1.x installed, you will be upgrading the Salvo VBScript components on the system. As a result, some scripts written in VBScript will no longer run on the original Salvo Server 4.0.x or 4.1.x installation. The scripts will run on the newer OnWeb Server 5.x or later installation.

Using 5250 emulation

If you install OnWeb Server 5.x or later on a machine that has Salvo Server 4.0.x or 4.1.x installed, you will not be able to use 5250 emulation with OnWeb Server 5.x or 6.0.0. To use 5250 emulation with OnWeb Server 5.x or later, you must first uninstall Salvo Server 4.x, and then install OnWeb Server 5.x or later.

Preserving objects and HTML files from the previous version

If you installed the new version of OnWeb Server in a different location than the old one, and you want to preserve your existing application objects and HTML files, you must:

- Manually move all your objects from the old Information_Rules directory to the Information_Rules directory created by the new OnWeb Server installation. You do not need to copy the Salvo.pks file and the Compiled subdirectory.
- Copy the HTML pages and scripts that were added or modified from the old location to the new HTML_Pages and Scripts directories.

Designer.mdb updated to a new format

If you update OnWeb Designer 2.0 to 2.1, using the same directory, when you start the DevServer, it updates your old metadata (designer.mdb) to the new 2.1 format. If you later try to access that designer.mdb file through Designer 2.0, you will get invalid SQL error messages and will not be able to open OnWeb Designer Advanced 2.0.

To solve this problem, the Designer installation backs up your old designer.mdb file and saves it as designer.001. You will get the message informing you about this fact during installation. If you install Designer 2.1 to a different directory, the old designer.mdb file will not be updated and is still compatible with Designer 2.0. However, your system's ODBC data source for Designer will be pointing to the new Designer metadata under the newly installed directory. If you wish to use your old version with your old metadata, you will need to change the ODBC data source "designer" to point to your 2.0 metadata designer.mdb. The latest DevServer should not be used with your old version of the metadata. Once you do so, that metadata will be updated to the OnWeb Advanced 2.1 format. You can then continue to use the DBServer as you did in 2.0.

Upgrading Host Publishing applications

In order to be able to edit Host Publishing applications originally created in Designer 2.1.x, they must be upgraded using the following procedure:

1. Back up the HostProfile.ini file found in the Designer 2.1.x system folder before uninstalling Designer 2.1.x and installing Designer 2.2.x.
2. After installing Designer 2.2.x, copy the hostProfile.ini file to the OnWeb Designer 2.2.x system folder (if you installed in the default folder, this will be "C:\Program Files\OnWeb\Designer").
3. Start Designer, create a new project, then close Designer. Creating a new project in Designer creates a folder under the OnWeb Source Server's Source\Debug folder with the project name you provided.
4. Copy the Applications and Templates folders from the Designer 2.1.x location to the newly created project folder in Designer 2.2.x (OnWeb\Source Server\Source\Debug\). The 2.1.x Application and Templates folders are located in the Designer 2.1.x source directory as indicated on the Designer 2.1.x Preferences tab.
5. For each Designer 2.1.x application that you want to add, make sure that you have a corresponding DSN entry in your systems ODBC Data Sources. These entries should be added on the System DSN tab in the ODBC Data Sources dialog box. Each DSN entry should:
 - › point to the application.mdb file in the corresponding application folder, for example:

```
OnWeb\Source Server\Source\Debug\ <project name>\Applications\
```
 - › have the same name as the <application name> folder. This name is case-sensitive.

You can remove these DSN entries after the upgrade process is successfully completed.
6. Start Designer and open the project created in step 3.
7. From the **Tools** menu, point to **Upgrade Applications**, then choose **Host Publishing**. If you did not copy the HostProfile.ini file into the Designer folder, a dialog box opens and allows you to locate this file.
8. After Designer locates the HostProfile.ini file, it imports all the applications specified in this file into your current project.



Upgrading Host Integration applications

In order to be able to edit Host Integration applications originally created in a version of Designer 2.1.0 or 2.1.1, the applications must be upgraded to version 2.2.0 using the Wizard. To access the Wizard, in Designer, from the **Tools** menu, point to **Upgrade Applications** and choose **Host Integration**. Then follow the instructions in the Wizard.

Importing User Services objects

Starting in version 6.0, OnWeb no longer offsets data tables in the User Services IObject by one when the Auto Disconnect is turned on. Therefore, if you are importing a User Services object that is in HTML format and has the Auto Disconnect turned on, you need to manually change table numbers in all the SalvoImport() or OnWebImport(), tags found in the HTML script.

For example, in the Designer 2.2.4 application, to show the first cell's value of the first table, you would use the following tag: `OnWebImport (2 , 1 , 1)` .

Once this User Service is imported into a Designer 6.0 application, you will have to change this line to read: `OnWebImport (1 , 1 , 1)` in order to display the same cell's value.

Use the new Find in Files feature of OnWeb 6.x or later to quickly find the OnWebImport/SalvoImport strings in the scripts.

Back up HostProfile.ini

If you created Host Publishing applications with the previous version of Designer and are planning to import those applications to the next version of Designer, you must back up the HostProfile.ini file before uninstalling old version of Designer. Then, after installing the new version of Designer, restore the file. This file is located in the Designer directory.

Using the closing </FORM> tag

Make sure that any FORM tags used in an HTML script have a closing </FORM> tag. To improve performance, OnWeb is now searching the HTML script for a closing </FORM> tag to insert the SessionID and SequenceID. If this tag is missing from the script, the SessionID and SequenceID are not inserted and the next request from the same client will cause a new Session to be created on the Server. The previous Session information will be lost.

Using the WaitFor function with VT emulation

Starting in OnWeb 5.2.0, the functionality of the WaitFor function for the VT emulation has changed. The old WaitFor function was similar to the CheckFor function and scanned the host screen until either the text is found or the timeout has passed. The new WaitFor function monitors the inbound data stream from the host until the text is discovered.

If you are using the WaitFor function in your scripts, read the following notes to determine the best course of action.

- If you are upgrading to OnWeb 5.2.0, you need to modify your scripts to accommodate the new WaitFor function.
- If you are upgrading to OnWeb 5.2.1 or later, a new tVTEmul.dll is installed that allows you to toggle between the old and the new WaitFor functionality by changing the value of the registry key.

► To use the old version of WaitFor

1. Open Registry Editor.
2. Under HKEY_LOCAL_MACHINE\SOFTWARE\NetManage\OnWeb\Emulator, add a new key named VT_WAITFOR with the DWORD value = 1.

OnWeb Server uses 3270 Emulator Version 2.0. Customers who are upgrading applications from Salvo Server may have applications written for 3270 Emulator Version 1.0.

The 2.0 version of the emulator represents significant improvements in memory usage and performance, especially under heavy load conditions, over the previous version, 1.0. As well, the newer version of the emulator corrects some minor emulation errors in the earlier version.

OnWeb Server 5.0.0 to 5.1.1 and Salvo Server versions 4.13 and up have both emulators installed. A registry setting is used to change the default emulator. Starting with OnWeb 5.1.2, only Emulator 2.0 is installed. The table below describes which emulator is active by default with the various software versions.

Version	Emulator 2.0 “Harbour”	Emulator 1.0 “Relay”
OnWeb 5.1.2 and higher	Only option	N/A
OnWeb 5.0.0 - 5.1.1	Default	Registry setting
Salvo 4.13+	Registry setting	Default
Salvo Server 4.12 -	Available as drop-in DLLs	Default

The result of these changes may affect existing information rule scripts that interact with the 3270 terminal. This appendix identifies those changes which may affect information rule or data source scripts. For assistance in modifying your OnWeb application for use with the 3270 emulator version 2.0, contact Micro Focus Technical Support (for details, see the back of the cover page).

Note: In this appendix, the word “function” refers to a function in REXX, or a method in VBScript.

Configuring the emulator version for OnWeb Server

This section contains information about editing the registry. Before you edit the registry, make sure you understand how to restore it if a problem occurs. For information about how to do this, view the “Restoring the Registry” Help topic in Regedit.exe or the “Restoring a Registry Key” Help topic in Regedt32.exe.

Using REGEDIT, open the following registry key:

HKEY_LOCAL_MACHINE\SOFTWARE\Simware\Salvo\Emulator

For 3270 Emulator Version 1.0 (Relay), set value to “1.0”

For 3270 Emulator Version 2.0 (Harbour), set value to “2.0”

There is no longer a need to copy emulator 2.0 DLLs into the BIN directory. Even if the DLLs are copied, the registry change will be required to activate the new emulator.

Note: Information in this section applies only to OnWeb 5.1.1 and earlier. In OnWeb 5.1.2 and later you must use the default setting of 2.0.

Changes in 3270 emulator version 2.0

Colour attributes

In version 2, non-extended attributes mode correctly handles the four colors available in this mode. Version 1 returned the default color for all colors in non-extended attributes mode.

In version 1, the Colour attribute in extended attribute mode of the blank character (space) could be inaccurate. Blank characters would have the Colour attribute of the previous non-blank character up to the next non-blank character. In version 2, the Colour attribute of the blank character is corrected.

Impacts

- Scripts which depend on the colors in non-extended attribute mode may be affected.
- Scripts which use the Colour attribute of text in extended color mode may be affected.

Autoskip attributes

Problems that may have occurred with Autoskip functionality in version 1 have been corrected in version 2.

Note: The Autoskip setting in the 3270 Terminal Data Source settings in Salvo Impact have no affect.

Impacts

- Any scripts that were coded to inaccurate Autoskip attributes may be affected.

Hidden and Protected attributes

In version 1, the Hidden Field attribute was incorrect in protected fields. Version 2 corrects this.

Impacts

- None expected.

ReadScreen function

The ReadScreen function, used to retrieve characters on the 3270 screen, now returns data in fields which are hidden-protected, that is, have both hidden and protected attributes. Previously, the data in hidden fields would be returned only for hidden-non-protected fields.

Impacts

- Any scripts that read and process a large region of the screen may have to be changed if there are hidden fields which also contain data. To check if a field is hidden, use the Attribute function.

Type function

In version 1, the Type function returned an error for keyboard locked for most error conditions. In version 2, the Type function returns the error code for the specific condition.

Impacts

- Scripts that check the return code of the Type command may be affected. Scripts can now also be improved to take advantage of the error reporting for conditions such as a disconnected emulation session.

Press("Clear") function

In version 1, there may have been a delay in the clear due to the way that it was processed.

In version 2, when Press("Clear") is called, the screen is cleared immediately. That is, the screen is cleared before the call to Press("Clear") returns control to the script.

Impacts

- If you have a routine which waits for a Press("Clear") to complete, this routine could be simplified.
- This may affect a routine that waits for response after pressing a key. Specifically, if you have a routine that, after a Press command, waits for a screen change using comparative readscreens and if this routine does an infinite wait for a change in the screen, then this script may have to be changed.

KeyboardLocked function

The KeyboardLocked function, in version 2, more accurately reflects a real terminal. The keyboard is locked for the interval between when the Press function returns and when the first host data is received. Previously, the keyboard-locked state was unpredictable during this interval.

Impacts

- Scripts that use the keyboard-locked state in a routine to wait for response from the host may be affected. Scripts can also be modified to take advantage of the more accurate reporting of keyboard-locked state.

Numerics

- 3270 emulator
 - Harbour • 27
 - Relay • 27
- 5250 emulation • 23

A

- attributes
 - Autoskip • 29
 - Colour • 28
 - Hidden • 29
 - protected • 29
- Autoskip attribute • 29

B

- Business Logic object • 22

C

- closing FORM tag • 25
- Colour attribute • 28

D

- Designer.mdb • 23

F

- functions
 - KeyboardLocked • 30
 - Press("Clear") • 30
 - ReadScreen • 29
 - Type • 30

H

- Harbour • 27
- Hidden attribute • 29
- Host Integration application, upgrading • 25
- Host Publishing application, upgrading • 24
- HostProfile.ini • 25

K

- KeyboardLocked function • 30

O

- OnWebSystem.wpc • 22

P

- Press("Clear") function • 30
- Protected attribute • 29

R

- ReadScreen function • 29
- Refine script • 22
- Relay • 27

S

- SalvoInit.srx • 22

T

- Type function • 30

W

- WaitFor function • 26
- Web services • 22