Rocket U2 Web Development Environment

Installation and Configuration

Version 5.3.0

July 2017
WDE-530-ALL-IM-01
Notices

Edition

Publication date: July 2017  
Book number: WDE-530-ALL-IM-01  
Product version: Version 5.3.0

Copyright

© Rocket Software, Inc. or its affiliates 1996-2017. All Rights Reserved.

Trademarks

Rocket is a registered trademark of Rocket Software, Inc. For a list of Rocket registered trademarks go to: www.rocketsoftware.com/about/legal. All other products or services mentioned in this document may be covered by the trademarks, service marks, or product names of their respective owners.

Examples

This information might contain examples of data and reports. The examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

License agreement

This software and the associated documentation are proprietary and confidential to Rocket Software, Inc. or its affiliates, are furnished under license, and may be used and copied only in accordance with the terms of such license.

Note: This product may contain encryption technology. Many countries prohibit or restrict the use, import, or export of encryption technologies, and current use, import, and export regulations should be followed when exporting this product.
**Corporate information**

Rocket Software, Inc. develops enterprise infrastructure products in four key areas: storage, networks, and compliance; database servers and tools; business information and analytics; and application development, integration, and modernization.

Website: [www.rocketsoftware.com](http://www.rocketsoftware.com)

Rocket Global Headquarters
77 4th Avenue, Suite 100
Waltham, MA 02451-1468
USA

To contact Rocket Software by telephone for any reason, including obtaining pre-sales information and technical support, use one of the following telephone numbers.

<table>
<thead>
<tr>
<th>Country</th>
<th>Toll-free telephone number</th>
</tr>
</thead>
<tbody>
<tr>
<td>United States</td>
<td>1-855-577-4323</td>
</tr>
<tr>
<td>Australia</td>
<td>1-800-823-405</td>
</tr>
<tr>
<td>Belgium</td>
<td>0800-266-65</td>
</tr>
<tr>
<td>Canada</td>
<td>1-855-577-4323</td>
</tr>
<tr>
<td>China</td>
<td>400-120-9242</td>
</tr>
<tr>
<td>France</td>
<td>08-05-08-05-62</td>
</tr>
<tr>
<td>Germany</td>
<td>0800-180-0882</td>
</tr>
<tr>
<td>Italy</td>
<td>800-878-295</td>
</tr>
<tr>
<td>Japan</td>
<td>0800-170-5464</td>
</tr>
<tr>
<td>Netherlands</td>
<td>0-800-022-2961</td>
</tr>
<tr>
<td>New Zealand</td>
<td>0800-003210</td>
</tr>
<tr>
<td>South Africa</td>
<td>0-800-980-818</td>
</tr>
<tr>
<td>United Kingdom</td>
<td>0800-520-0439</td>
</tr>
</tbody>
</table>

**Contacting Technical Support**

The Rocket Community is the primary method of obtaining support. If you have current support and maintenance agreements with Rocket Software, you can access the Rocket Community and report a problem, download an update, or read answers to FAQs. To log in to the Rocket Community or to request a Rocket Community account, go to [www.rocketsoftware.com/support](http://www.rocketsoftware.com/support).

In addition to using the Rocket Community to obtain support, you can use one of the telephone numbers that are listed above or send an email to support@rocketsoftware.com.
Chapter 5: Copying RedBeans components to a UNIX computer

Setting up the web server computer
Removing previous versions of the RedBeans (UNIX)
Copying RedBeans components

Chapter 6: Installing the stand-alone JSP RBOScope testing tool

JSP RBOScope testing tool overview
Installing the JSP RBOScope testing tool
Configuring the encoding of the JSP RBOScope testing tool

Chapter 7: Authorizing Web DE

Authorizing Web DE
Clearing violation codes

Chapter 8: Configuring the RedBack gateway

RedBack gateway overview
Configuring the RedBack gateway
Modifying default global settings
Inserting an account in the gateway configuration file
Modifying account settings
Chapter 1: Rocket U2 Web Development Environment overview

Rocket U2 Web Development Environment (Web DE) is an application development toolkit for building interactive client/server applications for deployment on the World Wide Web or on a corporate intranet.

Introduction

This document provides a complete step-by-step guide to installing and setting up U2 Web Development Environment. It contains steps for installing components of Web DE on Windows, UNIX, deploying the RedBack Object Server component to a UNIX or Windows computer, and copying RedBeans components to a UNIX computer. This document also provides instructions for installing two optional features: a RedPages COM Wrapper and a stand-alone JSP RBOScope testing tool. Additional topics include upgrading older U2 accounts to the Web DE 5.x architecture, and licensing and authorizing the product.

Web DE components


The following table describes the role of each Web DE component.

Table 1: Web DE components

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RedBack Object Server</td>
<td>The RedBack Object Server manages access to RedBack applications and implements dynamic runtime interfaces to server-side application resources. The RedBack Object Server also maintains the repository that contains all the definitions and code that define an application.</td>
</tr>
<tr>
<td>Java Scheduler (Connection Manager)</td>
<td>The Java Scheduler manages data transfer between the RedBack Object Server and the web server, handling requests from RedBeans and RedPages.NET. It also performs the job of load balancing, processing the queue of connection requests by allocating U2 licenses as webshares.</td>
</tr>
<tr>
<td>Admin API Server</td>
<td>The Admin API web server hosts a Web API and accompanying Web UI for Java Scheduler (Connection Manager) administration.</td>
</tr>
<tr>
<td>RedBeans and RedPages.NET</td>
<td>The RedBeans and RedPages.NET components are the Java and .NET application programming interfaces (APIs) that enable you to connect your application to RedBack Objects (RBOs). These components are the interface between the web server and the RedBack Object Server. The role of these components is to establish and maintain connection pool connections or webshare connections between the web server and the RedBack Object Server.</td>
</tr>
<tr>
<td>Web Designer</td>
<td>This suite of designer tools runs in an Eclipse development environment and provides an interface for designing RBOs. For more information, see Web Designer User’s Guide.</td>
</tr>
</tbody>
</table>
Web DE architecture

Web DE supports two different methods for managing web requests and responses:

- **Web DE installation with Java Scheduler, on page 7** - Uses a webshare model similar to that implemented in Web DE 4.x
- **Web DE installation with connection pooling, on page 7** - Uses the connection pooling technology of UniData 7.x and UniVerse 10.3 and later.

Web DE installation with Java Scheduler

The following figure illustrates Web DE with the Java Scheduler.

![Web DE with the Java Scheduler](image)

Web DE installation with connection pooling

The following figure illustrates Web DE with connection pooling.
Web DE offers the alternative of using the connection pooling feature of UniData or UniVerse to perform load balancing, supported in UO.NET and UOJ.

**RedPages COM Wrapper**

Web DE provides an optional RedPages Component Object Model (COM) Wrapper for compatibility with ASP and other COM-based applications. The wrapper connects classic ASP and other COM-based applications to the Web DE 5.x architecture, which relies on the Microsoft .NET Framework version 4.0.
Chapter 2: Installation overview

The following sections provide information and steps to help you install Web DE.

Web DE is installable on both Windows and UNIX. For Windows information, see Installing Web DE on Windows, on page 9. For UNIX information, see Installing Web DE on UNIX, on page 17.

For upgrade information, see Upgrade overview, on page 23.

Installing Web DE on Windows

The following sections provide instructions for setting up your computer for Web DE and installing components of the product on Windows.

Preinstallation

Before installing, you must verify your system meets the requirements described in the following sections. You can also uninstall older versions of Web DE if you choose.

System requirements (Windows)

Verify the system requirements to support Web DE.

The following figure illustrates how components of Web DE can be installed on one or several computers.

*Web Designer and RedPages.NET are Windows-only components.
In some cases, you can install server components on the same computer. For example, you can install the Java Scheduler on the same computer that runs the web server or the RedBack Object Server. However, supported operating systems, web server computer requirements, and disk space requirements are listed for each server component because you might install components on separate computers.

**U2 Web Designer requirements**

**Supported operating systems:**
- Windows Server 2012 (R2)
- Windows 8.1
- Windows 8
- Windows 7 (SP1)

**Web server requirements:**
Requirements for .NET applications (RedPages.NET):
- Microsoft Internet Information Services (IIS) 6.x or later
- Microsoft .NET Framework 3.5 or later
- Microsoft .NET Framework 4.0 or later (for RedPages COM Wrapper)
- Microsoft Visual Studio 2010 or later

Requirements for Java applications (RedBeans), one of the following:
- Java Development Kit (JDK) 8 or later
- Java Runtime Environment (JRE) 8 or later

**Disk space requirements:** 300 MB

**Java Scheduler requirements**

**Supported operating systems:**
- Windows Server 2012 (R2)
- Windows Server 2008 (R2 SP1)
- Windows 10
- Windows 8.1
- Windows 7 (SP1)

**Runtime environment:** JRE 8 or later

**Disk space requirements:** 1 MB

**RedBack Object Server requirements**

**Supported operating systems:**
- Windows Server 2012 (R2)
- Windows Server 2008 (R2 SP1)
- Windows 10
- Windows 8.1
- Windows 7 (SP1)

**Runtime environment:** JRE 8 or later
**Disk space requirements:** 20 MB

**Database server requirements:**
- UniData 8.1 or higher
- UniVerse 11.2 or higher

(Optional) Uninstalling previous versions of Web DE

If you installed a previous version or deployed components of Web DE to the target Windows computer, and want to remove the components to prepare for a clean installation, complete the following steps:

**Note:** These steps are not required unless you want only one version of Web DE on your computer at a time.

1. On the target Windows computer, click **Start → Control Panel → Programs and Features.**
2. In the list of installed programs, right-click **U2 Web Development Environment** and select **Uninstall.**

Running multiple versions of Web DE

You do not need to uninstall previous versions of Web DE before installing Web DE 5.x. If you have a limited number of computers available, it is possible to run previous and current versions of Web DE on the same computer. This includes 5.1.1 running on the same computer where 5.1.0 is installed.

The U2WDE environment variable contains the path to the last product installed. If you want to use a previously installed product, you must manually change the path to that version in the U2WDE variable. Each version must be installed to a different location and must be treated as a separate product.

You can upgrade existing U2 accounts used in your applications to the Web DE 5.x architecture. For more information, see [Upgrading existing U2 accounts to 5.x, on page 22](#).

If you have reason to run legacy Web DE architecture for some U2 accounts while upgrading others to the 5.x architecture, the accounts can coexist without any problems. However, note that any specific account can run only with the pre-5.x architecture or with the 5.x architecture—not both.

Reusing RBOs from previous versions

RBOs created in previous versions of Web DE can be reused in 5.x.

**Prerequisites**

- Creating a Module, as described in the *Administrator’s Guide.*

After you have created a 5.x account and module (such as EXMOD in rbexamples), perform the following steps.

1. Browse to the equivalent module at the operating system level in your old account.
2. Browse to the file `xxCLASSES` (where `xx` represents the module ID, such as `EXMODCLASSES`) and copy this file to the 5.x account.
New connections made to the 5.x account in the Web Designer now show the RBOs from the old account in the appropriate module under U2 Web DE Elements.

Installing Web DE (Windows)

After you have verified that your system meets the requirements to install Web DE, you can proceed with these installation steps.

Prerequisites

- Review the requirements, described in System requirements (Windows).
- (Optional) Uninstalling previous versions of Web DE, on page 11

Procedure

1. Start Windows on the computer on which you want to install components of Web DE.
2. Turn off any virus protection software as it can interfere with the installation.
3. Close all Windows programs.
4. Start the World Wide Web Publishing Service on the computer before you run the AutoRun.exe file to install Web DE.
   a. Open the Services control panel by clicking Start → Control Panel → Administrative Tools → Services.
   b. In the Services control panel, right-click World Wide Web Publishing Service and select Start. A status of “Started” indicates that Windows started the service successfully.
5. From the installation folder, run the AutoRun.exe file.
7. Follow the steps in the installation wizard.
   a. When the Select Features window is displayed, scroll through the list of available component features. Select the check box of each component to install. The most commonly used components are selected by default. If you do not want to install a component, clear the check box.

The following table describes each component feature.

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>U2 Web Designer</td>
<td>An Eclipse-based visual design tool for designing and building RBOs for web access to U2 accounts.</td>
</tr>
<tr>
<td>RedPages.NET</td>
<td>A set of .NET classes to deploy from a Windows-based web server to provide access to U2 accounts using RBOs. The RedPages COM Wrapper files are included in the RedPages.NET component. For additional steps in configuring the RedPages COM Wrapper, see (Optional) Configuring the RedPages COM Wrapper, on page 13.</td>
</tr>
<tr>
<td>RedBeans</td>
<td>A set of Java classes to deploy from a Java-based web server or call from a Java application to provide access to U2 accounts using RBOs.</td>
</tr>
</tbody>
</table>
### Component Description

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Java Scheduler (Connection Manager)</td>
<td>A Java-based scheduler that processes the queue of connections using a webshare licensing scheme.</td>
</tr>
<tr>
<td>Admin API Server</td>
<td>The Admin API web server hosts a Web API and accompanying Web UI for Java Scheduler (Connection Manager) administration.</td>
</tr>
<tr>
<td>RBO Data sources</td>
<td>This component enables RBOs to participate in ASP.NET data binding. RBO Data sources integrates directly with Visual Studio to enhance usability in working with RBOs.</td>
</tr>
<tr>
<td>Sample Web Pages</td>
<td>Contains sample JSP and ASP.NET-based web pages showing example code connecting to RBOs.</td>
</tr>
<tr>
<td>RedBack Object Server for UniData</td>
<td>Contains the RedBack Object Server account rbdefn, providing access to the UniData database server.</td>
</tr>
<tr>
<td>UniData Example</td>
<td>The rbexamples account for use with the RedBack Object Server for UniData.</td>
</tr>
<tr>
<td>RedBack Object Server for UniVerse</td>
<td>Contains the RedBack Object Server account rbdefn, providing access to the UniVerse database server.</td>
</tr>
<tr>
<td>International Licenses</td>
<td>The internationalized license agreement files for Web DE.</td>
</tr>
</tbody>
</table>

b. If a previous version of Web DE is on your computer, the wizard alerts you to change the installation path so the previous version does not corrupt the current installation. By default, the suggested path is U2WDEnnn, where nnn is the current version number being installed. Accept the current path or change it.

c. When the program finishes copying files, the message “Installation is finished” indicates that the installation was successful. Click **OK**.

### (Optional) Configuring the RedPages COM Wrapper

Web DE offers the optional RedPages COM Wrapper for compatibility with classic ASP and other COM-based applications. The RedPages COM Wrapper connects classic ASP and other COM-based applications to the Web DE 5.x architecture, which relies on the Microsoft .NET Framework.

#### Prerequisites

Install Microsoft .NET Framework 4.0 or later on the web server computer.

#### About this task

This feature is designed for application developers who have written ASP or other COM-based applications with the RedPages 3.x or 4.x API and do not have the option of moving the existing code into ASP.NET. The wrapper allows you to upgrade Web DE server components to 5.x and employ features of the new architecture without modifying the application front end.

To begin, install the RedPages COM Wrapper on the web server computer on which you run your classic ASP or other COM-based applications. The RedPages COM Wrapper libraries replace the
registered classes for classic RedPages on the web server. With the RedPages COM Wrapper libraries installed, COM-based applications run against the Web DE 5.x architecture with no modifications.

**Note:** The ADO calling interface from classic RedPages has not been implemented in the RedPages COM Wrapper. Applications that use the ADO calling interface need some rework to convert them to the COM interface used with the RedPages COM Wrapper.

To configure the RedPages COM Wrapper, complete the following steps:

**Procedure**

1. Determine whether a 32-bit or a 64-bit COM Wrapper is required. Navigate to [IIS Manager](#) → **Application Pools** → **Advanced Settings** and view the value of the **Enable 32-bit Applications** parameter.
   - If this option is set to **True**, you will register the RedPages COM Wrapper in the 32-bit version of .NET Framework 4.0.
   - If this option is set to **False**, you will register the RedPages COM Wrapper in the 64-bit version of .NET Framework 4.0.

2. On the web server computer, stop the World Wide Web Publishing Service:
   a. Open the Services control panel by clicking **Start** → **Control Panel** → **Administrative Tools** → **Services**.
   b. In the Services control panel, right-click **World Wide Web Publishing Service** and select **Stop**. The Service Control window provides a status as Windows stops the service.

3. Navigate to the Web DE installation directory. The default installation directory is: `C:\U2\U2WDEnnn`, where `nnn` is the version number you installed.
   In the installation directory, open the `rgw` folder, which contains supporting files for the RedPages COM Wrapper:
   - `RedPagesCCW.dll`
   - `RedPagesNET.dll`
   - `UODOTNET.dll`
   - `redpagesCCW32_setup.bat`
   - `redpagesCCW64_setup.bat`
   - `redpagesCCW32_uninstall.bat`
   - `redpagesCCW64_uninstall.bat`
   To make use of RedPages.NET in your application, add the `RedPagesNET.dll` and `UODOTNET.dll` as references in your preferred .NET development environment.

4. Open the command prompt on the web server computer by using the **Run as administrator** option.

5. At the command line, navigate to the `rgw` folder in the Web DE installation directory and run the `redpagesCCW32_setup.bat` file or the `redpagesCCW64_setup.bat` file, as determined in the first step.
   A Registry editor window displays, alerting you that an entry will be added to the Registry. Click **Yes** to continue.
   A second Registry editor window displays, confirming registration of UO.NET components. Click **OK** to continue.
In the following example, the output prompts to view full details of the registration process in the batch file log redpagesCCW64_setup.txt:

C:\U2\U2WDE\nnn\rgw>RedPagesCCW64_setup.bat Welcome to RedPages COM Wrapper setup...
Registering RedPages COM Wrapper components...
Checking .NET framework version...
32-bit Microsoft.NET framework used. Calling regasm...
Registering UO.NET components...
Registration of RedPages COM Wrapper components complete!
Please view redpagesCCW_setup.txt for full registration details.

7. Upgrade existing U2 accounts from version 3.x or 4.x to Web DE 5.x, if necessary. For steps, see (Optional) Upgrading U2 accounts, on page 22.
8. Open a web browser and point it to your ASP application. Confirm that you can read and write data in your application account.

Setting up the Admin API Server

The Admin API web server hosts a Web API and accompanying Web UI for Java Scheduler (Connection Manager) administration. Access the application.properties file to set up connection, configuration, and monitoring functions on the Admin API Server.

1. Open the application.properties file in a text editor. By default, this file resides in the following location: C:\U2\U2WDE5xx\apiserver.
2. In the [CM] section of the file, ensure cm.adminport is set to 7071.
3. In the [DB] section of the file, review the following parameters and revise values as necessary.
   - db.user: Enter a valid OS user name. The db.user and db.pass credentials will be used to connect to the rbdefn account and validate credentials from the WWUSERS file.
     
     Note: If you need to use a "\\" character in the user name, use two slashes ("\\") instead of one in order for the path to be understood by the application.properties file.

   - db.host: Ensure this property is set to the host where the rbdefn account resides.
   - db.service: Ensure this property is set to either uvcs for UniVerse, or udcx for UniData.
   - db.account: Set this property to the location of your rbdefn account.

   Note: If you are configuring it to point to a path on a remote Windows machine, the application.properties file expects two backslashes “\\” when specifying a backslash “\\” character. For UNIX, use a single slash.
   
   For example:
   - For Windows, use C:\U2\U2WDE5xx\UVserver\rbdefn.
   - For UNIX, use /disk1/wde5xx/rbdefn.

4. Generate an encrypted form of the password for the db.user specified in the previous step by running the following program, which accepts a password and then returns an encrypted string for the password:

   $ java -jar hazify.jar -i
   Enter password: <hidden-password> [Enter]
Enter password again: <hidden-password> [Enter]
Encrypted password is: 5ngy20fm3M0kXUWU2SdU1a==
5. Copy the encrypted password string and paste it into the **db.pass** field.
6. Save and close the **application.properties** file.

**Security between the Admin API Server and the database**

To secure the connection between your dataserver and the Admin API Server, you must first configure your database connection with the Admin API Server for SSL. This entails creating or obtaining a valid certificate and placing it in your truststore. See either your UniVerse or UniData *Security Features* guide for additional instructions on creating a certificate on the database.

**Procedure**

1. Open the **application.properties** file (stored by default in the `connection-manager` directory) in a text editor.
2. In the **[DB]** section, make the following changes to enable SSL security between your dataserver and the Admin API Server:
   a. Set the **db.usingssl** property to **true**.
   b. In the **db.ssl-trust-store** property, append the path to the trust-store for your dataserver certificate.
   c. In the **db.ssl-trust-store-password** property, enter the password for your truststore.
3. Save the **application.properties** file.
4. If the Admin API Server is already running, you must restart it to make these changes effective.

**Security between the client and the Admin API Server**

**Procedure**

1. Obtain a valid certificate and private key to secure the connection between the Admin API Server and your web browser. Specific instructions for creating or obtaining a certificate are outside the scope of this documentation.
2. Place the certificate and the private key from Step 1 into your keystore.
3. In the **[SSL]** section of the **application.properties** file, make the following changes to enable an HTTPS connection between the Admin API Server and the client:
   a. In the **server.port** field, designate the server port to be used for secure access to the Admin API Server. The default is 7043.
   b. In the **server.ssl.key-store** field, enter the path to the keystore referenced in Step 2.
   c. In the **server.ssl.key-store-password** field, enter the password for your keystore.
   d. In the **server.ssl.key-password**, enter the password you obtained in Step 1.
4. Save the **application.properties** file.
5. If the Admin API Server is already running, you must restart it to make these changes effective.
6. Verify that your secure connection is working by opening your web browser and entering `HTTPS://Admin API Server location:port number designated in Step 3a`, for example: `HTTPS://localhost:7043`.

**Starting and stopping the Admin API Server (Windows)**

On Windows, you can start and stop the Admin API Server (the Connection Manager Admin service) from the **Windows Services** menu.
Installing Web DE on UNIX

The following sections provide instructions for setting up your computer for Web DE and installing components of the product on UNIX.

Alternatively, you can use the Web Designer to deploy the RedBack Object Server to the UNIX machine. For more information, see Deploying the RedBack Object Server to a UNIX or Windows computer, on page 25.

Preinstallation

Before installing, you must verify your system meets the requirements described in the following sections. You can also uninstall older versions of Web DE if you choose.

System requirements (UNIX)

Verify the system requirements to support Web DE.

For more information on how Web DE can be used across multiple computers, see Figure 3: Web DE installation components on one or more computers, on page 9.

In some cases, you can install server components on the same computer. For example, you can install the Java Scheduler on the same computer that runs the web server or the RedBack Object Server. However, supported operating systems, web server computer requirements, and disk space requirements are listed for each server component because you might install components on separate computers.

Web server requirements

Supported operating systems (all 64-bit):

- AIX
  - 7.2
  - 7.1
- AP/UX Intel Itanium
  - 11.31
- Linux
  - Red Hat Enterprise Linux 6, 7
  - SUSE Linux Enterprise Server 11 (SP2)
- Solaris
  - 11 x86
  - 11 SPARC

Web server software: Java-based web server software, such as IBM WebSphere Application Server or Tomcat. Otherwise, if you only want to run client/server applications on this computer, one of the following:

1. From the Control Panel, navigate to Services → U2 Web DE Connection Manager Admin → Start the Service.
2. To stop the Admin API Server on Windows, access the Control Panel, navigate to Services → U2 Web DE Connection Manager Admin → Stop the Service.
Chapter 2: Installation overview

- Java Development Kit (JDK) 8 or later
- Java Runtime Environment (JRE) 8 or later

**Disk space requirement:** 120 KB

**Java Scheduler requirements**

**Supported operating systems:**

- AIX
  - 7.2
  - 7.1
- AP/UX Intel Itanium
  - 11.31
- Linux
  - Red Hat Enterprise Linux 6, 7
  - SUSE Linux Enterprise Server 11 (SP2)
- Solaris
  - 11 x86
  - 11 SPARC

**Runtime environment:** JRE 8 or later

**Disk space requirement:** 1 MB

**RedBack Object Server requirements**

**Supported operating systems:**

- AIX
  - 7.2
  - 7.1
- AP/UX Intel Itanium
  - 11.31
- Linux
  - Red Hat Enterprise Linux 6, 7
  - SUSE Linux Enterprise Server 11 (SP2)
- Solaris
  - 11 x86
  - 11 SPARC

**Database server requirements:**

- UniData 8.1 or higher
- UniVerse 11.2 or higher

**Runtime environment:** JRE 8 or later if you run the Java Scheduler to manage client/server applications on this computer.

**Disk space requirement:** 12 MB
(Optional) Uninstalling previous versions of Web DE

If you installed a previous version or deployed components of Web DE to the target UNIX computer and want to remove the components to prepare for a clean installation, complete the following steps:

**Note:** These steps are not required unless you want only one version of Web DE on your computer at a time.

1. On the UNIX machine where Web DE is installed, open a command window and change directory to the Web DE path.
2. Use the remove program command (for example, `rm`) to remove the Web DE files.

Preparing to install

You must perform some preliminary setup work before installing Web DE 5.x.

You need two source files to install Web DE on UNIX: an install script and a `.tar` file. These files are located in the Web DE installation folder in the `\server\unix\udt` or `\server\unix\uv` directory.

The installation script, `rbappsvr`, is common to all operating systems. It allows you to install all components of Web DE quickly using the `.tar` file. The `.tar` file contains all the example accounts, directories, and files for Web DE. The `.tar` file uses the following naming convention:

`rb521_udt73_3094.tar`, where:

- `521` is the version number.
- `udt73` is the U2 account used (in this case, UniData; UniVerse uses “uv”) and the minimum version number required, in this case 7.3, indicating that release 7.3 and later are supported.
- `3094` is the build number.

These files must be copied from the extracted installation folder or copied from another computer.

Copying source files from the installation folder

Copy the source files from the installation folder to the temp directory.

1. Change to the installation folder containing the required source files.
2. Copy the two required files to your temp directory, for example,
   ```
   $ cp rbappsvr /tmp/
   $ cp rb521_udt73_3086.tar /tmp/
   ```

Copying source files from another computer

Alternatively, you can copy the source files from a Windows computer through FTP network transfer on a TCP/IP network.

1. Extract the installation folder on the Windows computer.
2. On the Windows computer, open an FTP session to the UNIX host.
3. Change the local directory to the path of the required source files on the installation folder.
4. Change the directory to your temp directory.
5. In binary transfer mode, transfer the two source files to the temp directory.
Chapter 2: Installation overview

Installing Web DE (UNIX)

After you have verified that your system meets the requirements to install Web DE, you can proceed with these installation steps.

Prerequisites

- Review the requirements, described in System requirements (UNIX).
- (Optional) Uninstalling previous versions of Web DE, on page 19.

Procedure

1. Start UNIX on the computer on which you want to install components of Web DE.
2. Change to the temp directory, and enter `rbappsvr`.
3. Follow the script’s prompts for input for the license details, installation directory, U2 data server installation directory, and time zone. Typically, the default options are acceptable.
   a. For license details, locate the license details provided with the product and enter the numeric portion of your webshare serial number as the installation serial number for Web DE.
   b. For the installation directory, enter the directory to which you want to install all of the Web DE files.
   c. Accept or change the installation path of the UniData or UniVerse database. For UniData, the script detects `UDTHOME`. For UniVerse, the script detects `UVHOME`. If the environment variable is not set, the installation process reads the `.uvhome` file and extracts the location of the UniVerse installation.
   d. Accept or change the time zone. The script detects the time zone from the `TZ` environment variable.
4. Accept the prompt to install from the `.tar` file.
   The .tar file contains all the example accounts, directories, and files for Web DE, such as the rbdefn, rbexamples, and rbodemo directories, and the Java Scheduler.

Setting up the Admin API Server

The Admin API web server hosts a Web API and accompanying Web UI for Java Scheduler (Connection Manager) administration. Access the `application.properties` file to set up connection, configuration, and monitoring functions on the Admin API Server.

1. Open the `application.properties` file in a text editor. By default, this file resides in the following location: `C:\U2\U2WDE5\apiserver`.
2. In the `[CM]` section of the file, ensure `cm.adminport` is set to `7071`.
3. In the `[DB]` section of the file, review the following parameters and revise values as necessary.
   - `db.user`: Enter a valid OS user name. The `db.user` and `db.pass` credentials will be used to connect to the rbdefn account and validate credentials from the `WWUSERS` file.
   - `db.host`: Ensure this property is set to the host where the rbdefn account resides.
   - `db.service`: Ensure this property is set to either `uwcs` for UniVerse, or `udcs` for UniData.
   - `db.account`: Set this property to the location of your rbdefn account.

*Note: If you need to use a "\" character in the user name, use two slashes ("\") instead of one in order for the path to be understood by the `application.properties` file.*
4. Generate an encrypted form of the password for the `db.user` specified in the previous step by running the following program, which accepts a password and then returns an encrypted string for the password:

```
$ java -jar hazify.jar -i
Enter password: <hidden-password> [Enter]
Enter password again: <hidden-password> [Enter]
Encrypted password is: 5ngyZOfm3M0kxWUi2SdUlA==
```

5. Copy the encrypted password string and paste it into the `db.pass` field.

6. Save and close the `application.properties` file.

Security between the client and the Admin API Server

Procedure

1. Obtain a valid certificate and private key to secure the connection between the Admin API Server and your web browser. Specific instructions for creating or obtaining a certificate are outside the scope of this documentation.

2. Place the certificate and the private key from Step 1 into your keystore.

3. In the [SSL] section of the `application.properties` file, make the following changes to enable an HTTPS connection between the Admin API Server and the client:
   a. In the `server.port` field, designate the server port to be used for secure access to the Admin API Server. The default is 7043.
   b. In the `server.ssl.key-store` field, enter the path to the keystore referenced in Step 2.
   c. In the `server.ssl.key-store-password` field, enter the password for your keystore.
   d. In the `server.ssl.key-password`, enter the password you obtained in Step 1.

4. Save the `application.properties` file.

5. If the Admin API Server is already running, you must restart it to make these changes effective.

6. Verify that your secure connection is working by opening your web browser and entering HTTPS://Admin API Server location:port number designated in Step 3a, for example: HTTPS://localhost:7043.

Security between the Admin API Server and the database

To secure the connection between your dataserver and the Admin API Server, you must first configure your database connection with the Admin API Server for SSL. This entails creating or obtaining a valid certificate and placing it in your truststore. See either your UniVerse or UniData Security Features guide for additional instructions on creating a certificate on the database.

Procedure

1. Open the `application.properties` file (stored by default in the connection-manager directory) in a text editor.
2. In the [DB] section, make the following changes to enable SSL security between your dataserver and the Admin API Server:
   a. Set the `db.usingssl` property to `true`.
   b. In the `db.ssl-trust-store` property, append the path to the trust-store for your dataserver certificate.
   c. In the `db.ssl-trust-store-password` property, enter the password for your trust-store.
3. Save the `application.properties` file.
4. If the Admin API Server is already running, you must restart it to make these changes effective.

Starting and stopping the Admin API Server (UNIX)

On UNIX, you can start and stop the Admin API Server (the Connection Manager Admin service) from the command line.
1. To start the server on UNIX, enter the following command in the command line:
   
   $ start-admin.sh &
   
   2. To stop the server, enter the following command in the command line:
   
   $ stop-admin.sh

(Optional) Upgrading U2 accounts

U2 accounts created with Web DE 3.x or 4.x can be used with version 5.x by completing a short procedure to convert them.

Upgrading existing U2 accounts to 5.x

Complete the following procedure to upgrade existing U2 accounts used in your applications from version 3.x or 4.x to Web DE 5.x.

Prerequisites

- Existing U2 accounts used in Web DE 3.x or 4.x.
- Installing Web DE (Windows), on page 12 or Installing Web DE (UNIX), on page 20
  This includes installing or deploying the RedBack Object Server component to a Windows or UNIX computer.

About this task

It is not necessary to uninstall Web DE 3.x or 4.x before installing 5.x. If you have reason to run the legacy Web DE architecture for some U2 accounts while upgrading others to the 5.x architecture, the accounts can coexist without any problems. However, note that any specific account can run only with the pre-5.x architecture or with the 5.x architecture, but not both.

Procedure

1. Before proceeding to the next step, make note of the path of all application accounts to be upgraded.
2. Click Start → All Programs → Rocket U2 → Web Development Environment → U2 Web Designer.
3. In the Web Designer, connect to a U2 server. For steps, see the information about making a U2 server connection in the *Administrator’s Guide*.

4. In the U2 Resource view, expand the Accounts tree, right-click *rbdefn* and select Properties.

5. In the U2 Account Properties window, confirm that the path to the rbdefn account points to the current installation of rbdefn on your system, and click Finish.

   If the path does not point to the 5.x location of rbdefn, correct the path as follows:
   a. Close the U2 Account Properties window.
   b. In the U2 Resource view, expand the Accounts tree, right-click *rbdefn* and select Delete to remove the rbdefn account link. In the Confirm Account Delete window, select Do not delete contents option and click Yes.
   c. In the Accounts tree, right-click Accounts and select New U2 Account.
   d. Use the Create a New U2 Account wizard to create a link to the rbdefn account, pointing to the current 5.x location for rbdefn, and click Finish.

6. In this step, you can create a link to the U2 account to be upgraded, if necessary. This process adds an entry for the U2 account in the *UV.ACCOUNT* or *UD.ACCOUNT* file.

   If the U2 account is listed in the Accounts tree, it already has an entry in *UV.ACCOUNT* or *UD.ACCOUNT* from previous usage. Proceed to the next step. If the U2 account does not appear in the Accounts tree, create a link to it as follows:
   a. In the U2 Resource view, right-click Accounts and select New U2 Account.
   b. In the Create a New U2 Account wizard, enter the account name and path to the U2 account to be upgraded, and click Finish.

   The new U2 account link is now listed in the U2 Resource view Accounts tree.

7. Enable the U2 account for Web DE as follows:
   a. In the U2 Resource view, expand the Accounts tree. Right-click the entry for the U2 account to be upgraded and select Enable for U2 Web DE.
   b. If prompted for the path to rbdefn, enter the path to the current location of rbdefn on your system.

8. Verify that the upgraded U2 account is working correctly as follows:
   a. In the Accounts tree, expand the tree view for the upgraded U2 account.
   b. Expand the U2 Web DE Elements node to confirm that RBOs are accessible.

9. On the web server computer, add an entry for the upgraded U2 account to the *rgw5.ini* file. For steps, see the information about editing the initialization file in the *Administrator’s Guide*.

10. Repeat steps 6 through 9 for each U2 account to be upgraded.

---

**Upgrade overview**

To successfully upgrade to v5.2.0 and keep your applications from a previous version, you must complete the following steps.

For a video tutorial of these steps, see link.

1. Deploy the RBO Server. There are a few different ways to perform this step.
   - On Windows, you can do this using the installation wizard’s default options on the Select Features screen.
See step 7a in Installing Web DE (Windows), on page 12 for instructions.

- On Windows or UNIX, you can use the Deploy to U2 Server option in Web Designer. See Deploying the RedBack Object Server to a UNIX or Windows computer, on page 26 for instructions.

- On UNIX, you can use the `rbappsvr` install script. See Installing Web DE (UNIX), on page 20 for instructions.

2. Once the RBO Server is deployed, update your web applications. There are a few different ways to perform this step.

   - If you created your applications using ASP or COM, you can use the COM wrapper to run ASP applications without changing older code. This is a stop-gap implementation that gives you time to update classic ASP with COM to ASP.NET. See (Optional) Configuring the RedPages COM Wrapper, on page 13 for instructions.

   - If you created your applications using Java technology, replace the previous RedBeans .jar files with the RedBeans.jar and asjava.jar files provided in the RedBeans directory of the installation folder.

   **Note:** The jri-provided `libjrgw.so` or `jgrw.dll` files are no longer needed.
Chapter 3: Deploying the RedBack Object Server to a UNIX or Windows computer

With Web DE now installed, you can use a wizard in the Web Designer to deploy the RedBack Object Server component to a UNIX or Windows computer.

Setting up a UNIX computer

If you want to deploy the RedBack Object Server component to a UNIX computer, you must set up a UNIX computer for that purpose.

To verify your UNIX computer meets all the requirements necessary, see System requirements (UNIX).

Removing previous versions of the RedBack Object Server (UNIX)

If you previously deployed the RedBack Object Server component to the target UNIX computer, remove the component to prepare for a clean installation.

1. On the target UNIX computer, navigate to the parent directory in which the RedBack Object Server component was deployed (for example, \U2\U2\WDE\nnn, where nnn is the version number you installed).
2. In the parent directory, delete the entire subdirectory that contains the RedBack Object Server component.

Setting up a Windows computer

Deploying the RedBack Object Server component to a Windows computer is optional.

You can install the RedBack Object Server and all other components of Web DE on a Windows computer through the Windows installation program. For details, see Installing Web DE on Windows, on page 9.

However, it is possible to use the wizard to deploy only the RedBack Object Server component to a Windows computer. You must first set up the Windows computer on which you want to deploy the RedBack Object Server.

To verify your Windows computer meets all the requirements necessary, see System requirements (Windows). System requirements (UNIX), on page 17

Removing previous versions of the RedBack Object Server (Windows)

If you previously deployed the RedBack Object Server component to the target Windows computer, remove the component to prepare for a clean installation.

1. If the RedBack Object Server was installed on the Windows computer, do the following:
   a. Click Start \> Control Panel \> Programs and Features.
   b. In the list of installed programs, right-click U2 Web Development Environment and select Uninstall.
2. If the RedBack Object Server was deployed to the Windows computer, do the following:
a. Navigate to the parent directory in which the RedBack Object Server component was deployed (for example, C:\U2\U2WDEnnn, where nnn is the version number you installed).
b. In the parent directory, delete the entire subdirectory that contains the RedBack Object Server component.

Deploying the RedBack Object Server to a UNIX or Windows computer

You deploy the RedBack Object Server component by using a wizard in the Web Designer.

Prerequisites

▪ Verify the system requirements, as described in System requirements (UNIX) and System requirements (Windows).
▪ Create a U2 server in the U2 Resource view of the Web Designer for the UNIX or Windows computer on which to deploy the RedBack Object Server component. If you have not already done so, create a U2 server before continuing. For detailed steps, see the information about defining a U2 server in the Administrator’s Guide.

About this task

Deployment is an optional, efficient method of installing this server component on a Windows computer. For UNIX, you can optionally install the RedBack Object Server as described in Installing Web DE on UNIX, on page 17 Installing Web DE (UNIX), on page 20.

Procedure

1. On the computer on which the Web Designer is installed, click Start → All Programs → Rocket U2 → Web Development Environment → U2 Web Designer.
2. In the U2 Resource view, open a connection to an existing U2 server on the UNIX or Windows computer on which to deploy the RedBack Object Server component.
3. With the name of the U2 server selected, click Tools → Deploy to U2 Server.
4. If the server software is not at or above the release level required by the deployer software, a dialog box gives you the option to upgrade the server software.

The user credentials (either the credentials defined for this server connection in the U2 Resource view or the Optional Server User Credentials field) must have sufficient privilege to write to the database server installation directory’s BP (for UniData) or BP.O (for UniVerse) and catalog area on the server, or the upgrade will fail and the deployment will also fail.

To proceed with the upgrade, click Yes.
5. From the wizard, in the Choose to Deploy a Product, Folder, or File page, the Deploy a packaged product option is selected by default. Accept the default option and click Next.
6. In the Choose a Product to Deploy page, select the logging level and product to deploy:
   a. From the Logging Level list, select a logging option for this deployment.

   The logging option sets the level of detail captured in the deployer logs on both the client and server computers. The client logs are stored in the U2 Web Designer installation directory on the client computer.

   The log names are formatted as follows:
DeployToServer_yyyymmdd_hhmmss.log where yyyymmdd is the four-digit year, two-digit month, and two-digit day the log was created. hhmmss is the two-digit hour, two-digit minute, and two-digit second the log was created.

For example, DeployToServer_20130924_104012.log is the file name for a log created on September 24, 2013 at 10:40:12 AM.

A nearly identical log is created on the server. The log is created in the account defined in the U2 Resource view's server connection in the “_PH_” folder (for UniData) or “&PH&” folder (for UniData), with the same naming convention.

If the &PH& folder is a Type 1 file in UniData, the length of the log name causes it to be broken into several subfolders under &PH&, but the subfolders contain all the data.

The client and server logs are usually close to identical, but individual products might log specific details to the client log or server log only. The two logs can be compared to see where an error occurs in the case of failure. The server log is created only if the &PH&/_PH_ folder is writable by the user establishing the connection.

This example shows the Default logging option, which provides enough detail to verify success or failure. If the deployment fails, the log indicates which operation was under way when the deployment failed.

b. In the Available Products list, click **RedBack Object Server**.

c. Click **Next**.

7. In the Choose Destination for Deployment page, specify the destination folder for each component to be deployed:

![Choose Destination for Deployment page](image)

- **Base Path for Server Folder** field: Specify a destination path to an existing folder on the server for the selected components. Enter the path name or click **Browse** to browse to and select a path.

- **Deploy Component?** list:
  - [ ] libxml
  - [ ] libexamples
  - [ ] libdemo

- **Select All** and **Clear Selected** buttons:

- **Next** and **Finish** buttons:

In the Base Path for Server Folder field, specify a destination path to an existing folder on the server for the selected components. Enter the path name or click Browse to browse to and select a path.
The path of the selected folder on the server is populated in the Base Path for Server Folder field in the wizard. The selected path is also propagated to each of the selected components. If a specific component needs to go to a different path, select a path for the component. Select the Parent Folder On Server field for that component, click Edit, and enter the path.

b. In the Deploy Component list, all components of the RedBack Object Server package are selected for deployment by default. Clear the check box for any component that you do not want to deploy.

c. After selecting a destination folder for each component, click Next.

Tip: Click inside the Parent Folder On Server column next to the component you want to enable an Edit button. Here you can change the destination folder for each individual component.

8. In the U2 Deployment Summary page, review your selections and enter different server credentials, if necessary.

In the Optional Server User Credentials group box, you can enter a different set of user credentials to the U2 server. This might be necessary if the connection for the U2 server in the U2 Resource view contains credentials that do not have sufficient privileges to perform some part of the deployment on the server, such as:

- Writing to the destination directory.
- Performing logging to the connected account’s &PH&/._PH_. directory.
- Upgrading the transfer software (XTOOLSUB) on the server if it is out of date.

9. If you need to make changes to your selections, click Back to return to previous pages of the wizard. Otherwise, click Finish to proceed with deployment.
Chapter 4: Deploying the Java Scheduler to a UNIX or Windows computer

The following sections describe the steps for using a wizard in the Web Designer to deploy the Java Scheduler component to a UNIX or Windows computer.

Setting up a UNIX computer

If you want to deploy the Java Scheduler component to a UNIX computer, you must set up a UNIX computer for that purpose.

To verify your UNIX computer meets all the requirements necessary, see System requirements (UNIX).

Removing previous versions of the Java Scheduler (UNIX)

If you previously deployed the Java Scheduler component to the target UNIX computer, remove the component to prepare for a clean installation.

1. On the target UNIX computer, navigate to the parent directory in which the Java Scheduler component was deployed (for example, \U2\U2WE\nnn, where nnn is the version number you installed).
2. In the parent directory, delete the entire subdirectory that contains the Java Scheduler component.

Setting up a Windows computer

Deploying the Java Scheduler component to a Windows computer is optional.

You can install the Java Scheduler and all other components of Web DE on a Windows computer through the Windows installation program. For details, see Installing Web DE on Windows, on page 9

However, it is possible to use the wizard to deploy only the Java Scheduler component to a Windows computer. You must first set up the Windows computer on which you want to deploy the Java Scheduler.

To verify your Windows computer meets the requirements necessary, refer to System requirements (Windows).

Removing previous versions of the Java Scheduler (Windows)

If you previously deployed the Java Scheduler component to the target Windows computer, remove the component to prepare for a clean installation.

1. If the Java Scheduler was installed on the Windows computer, do the following:
   a. Click Start > Control Panel > Programs and Features.
   b. In the list of installed programs, right-click U2 Web Development Environment and select Uninstall.
2. If the Java Scheduler was deployed to the Windows computer, do the following:
Deploying the Java Scheduler to a UNIX or Windows computer

You deploy the Java Scheduler component by using a wizard. Deployment is the only method of installing the Java Scheduler component on a UNIX computer. It is an optional, efficient method of installing this server component on a Windows computer.

Prerequisites

- Verify the software requirements, as described in System requirements (UNIX) and System requirements (Windows).
- Create a U2 server in the U2 Resource view of the Web Designer for the UNIX or Windows computer on which to deploy the RedBack Object Server component. If you have not already done so, create a U2 server before continuing. For steps, see the information about defining a U2 server in the Administrator’s Guide.

Procedure

1. In the U2 Resource view, open a connection to an existing U2 server on the UNIX or Windows computer on which to deploy the Java Scheduler component.
2. With the name of the U2 server selected, click Tools → Deploy to U2 Server.
3. If the server software is not at or above the release level required by the deployer software, a dialog box gives you the option to upgrade the server software.

   The user credentials (either the credentials defined for this server connection in the U2 Resource view or the Optional Server User Credentials field) must have sufficient privilege to write to the database server installation directory’s BP (for UniData) or BP.O (for UniVerse) and catalog area on the server, or the upgrade will fail and the deployment will also fail.

   To proceed with the upgrade, click Yes.
4. From the wizard, in the Choose to Deploy a Product, Folder, or File page, the Deploy a packaged product option is selected by default. Accept the default option and click Next.
5. In the Choose a Product to Deploy page, select the logging level and product to deploy:
   a. From the Logging Level list, select a logging option for this deployment.

      The logging option sets the level of detail captured in the deployer logs on both the client and server computers. The client logs are stored in the U2 Web Designer installation directory on the client computer.

      The log names are formatted as follows:
      DeployToServer_yyyymmdd_hhmms.log where yyyymmdd is the four-digit year, two-digit month, and two-digit day the log was created. hhmms is the two-digit hour, two-digit minute, and two-digit second the log was created.

      For example, DeployToServer_20130924_104012.log is the file name for a log created on September 24, 2013 at 10:40:12 AM.
A nearly identical log is created on the server. The log is created in the account defined in the U2 Resource view’s server connection in the “_PH_” folder (for UniData) or “&PH&” folder (for UniData), with the same naming convention.

If the &PH& folder is a Type 1 file in UniData, the length of the log name causes it to be broken into several subfolders under &PH&, but the subfolders contain all the data.

The client and server logs are usually close to identical, but individual products might log specific details to the client log or server log only. The two logs can be compared to see where an error occurs in the case of failure. The server log is created only if the &PH/&_PH_ folder is writable by the user establishing the connection.

This example shows the Default logging option, which provides enough detail to verify success or failure. If the deployment fails, the log indicates which operation was under way when the deployment failed.

b. In the Available Products list, click Java Scheduler.

c. Click Next.

6. In the Choose Destination for Deployment page, specify the destination folder for each component to be deployed:

   **Note:** The following figure displays a UNIX path as an example.

   ![Choose Destination for Deployment page](image)

   In the **Base Path for Server Folder** field, specify a destination path to an existing folder on the server for the selected components. Enter the path name or click **Browse** to browse to and select a path.

   The path of the selected folder on the server is populated in the Base Path for Server Folder field in the wizard. The selected path is also propagated to each of the selected components. If a specific component needs to go to a different path, select a path for the component. Select the **Parent Folder On Server** field for that component, click **Edit**, and enter the path.
b. In the **Deploy Component** list, all components of the Java Scheduler package are selected for deployment by default. Clear the check box for any component that you do not want to deploy.

c. After selecting a destination folder for each component, click **Next**.

**Tip:** Click inside the **Parent Folder On Server** column next to the component you want to enable an **Edit** button. Here you can change the destination folder for each individual component.

7. In the **U2 Deployment Summary** page, review your selections and enter different server credentials, if necessary.

In the **Optional Server User Credentials** group box, you can enter a different set of user credentials to the U2 server. This might be necessary if the connection for the U2 server in the U2 Resource view contains credentials that do not have sufficient privileges to perform some part of the deployment on the server, such as:

- Writing to the destination directory.
- Performing logging to the connected account’s &PH&/_PH_/ directory.
- Upgrading the transfer software (XTOOLSUB) on the server if it is out of date.

8. If you need to make changes to your selections, click **Back** to return to previous pages of the wizard. Otherwise, click **Finish** to proceed with deployment.

**Results**

The wizard copies the selected components on the target UNIX or Windows computer as follows to the Java Scheduler folder within the base path:

For UNIX:

- getRequests.jar
- JavaScheduler.ini
- restartAccount.jar
- Scheduler.jar
- StopScheduler.jar
- StartScheduler.sh
- StopScheduler.sh
- template.ini
- U2WebDESchedulerScript

For Windows:

- getRequests.jar
- restartAccount.jar
- Scheduler.jar
- StopScheduler.jar
- StartScheduler.bat
- StopScheduler.bat
- template.ini
Configuring, starting, and stopping the Admin API Server

The `application.properties` file is the configuration file for the Admin API Server, and is located in the `apiserver` directory of the Web DE installation directory.

Refer to the `Web DE Installation and Configuration` guide for details regarding this file.

1. Open the `application.properties` file with a text editor.
2. In the `[DB]` section, set the properties to connect to the rbdefn account and validate credentials from the `WWUSERS` file.
   - Ensure `db.port` is set to the database server port that stores the rbdefn account.
   - Ensure `db.user` contains the user name of a person with permission to access the rbdefn account.
   - Ensure `db.host` contains the host of the database server that stores the rbdefn account.
   - Ensure `db.account` contains the complete path to the rbdefn account on the database server.
   - Ensure `db.service` contains the name to use to connect to the database server storing the rbdefn account. Use `uvcs` for UniVerse, or `udcs` for UniData.
3. Generate an encrypted form of the password for the `db.user` specified in the previous step by running the following program, which accepts a password and then returns an encrypted string for the password:
   ```
   $ java -jar hazify.jar -i
   Enter password: <hidden-password> [Enter]
   Enter password again: <hidden-password> [Enter]
   Encrypted password is: 5ngyZOfm3M0kxWUi2SdUlA==
   ```
4. Copy the encrypted password string and paste it into the `db.pass` field.
5. Save the `application.properties` file.
6. Start the Admin API Server.
   - On Windows, start the Admin API Server from Windows Services.
   - On UNIX, start the Admin API Server from the command line using the following shell command:
     ```
     $ start-admin.sh &
     ```

It might take several minutes for the service to start, depending on your installation configuration. Runtime errors are stored in the `cm-admin.log` file, which is located in the `apiserver` logs directory.

By default, the Admin API server is listening on port 7077. You can access it in a browser at `http://host:7077`. The default login user name and password are `rbadmin` and `redback`.

7. Secure the connections between the rbdefn account and the Admin API server, and the connections between the Admin API Server and the browser client. Refer to the `Web DE Installation and Configuration` guide for instructions on securing these connections.
8. You can stop the Admin API Server at any time.
▪ On Windows, stop the Admin API Server from Windows Services.
▪ On UNIX, stop the Admin API Server from the command line using the following command:

```bash
$ stop-admin.sh
```
Chapter 5: Copying RedBeans components to a UNIX computer

The following sections describe the steps for copying and configuring the RedBeans components on a UNIX computer.

The RedBeans installation files shipped are:
- **RedBeans.jar**: Main RedBeans class files for development and run time
- **asjava.zip**: Contains the UniObjects for Java class files in zip format
- **asjava.jar**: Contains the UniObjects for Java class files in jar format
- **rgw5.ini**: Holds the RedBeans connection information
- **rbexamples.war**: Contains JSP RBExample pages

You can find these files RedBeans folder of the install.

Setting up the web server computer

If you want to run RedBeans on UNIX, you must set up the web server computer for that purpose.

To verify your UNIX computer meets all the requirements necessary, see [System requirements (UNIX)](#).

Removing previous versions of the RedBeans (UNIX)

If you previously copied the RedBeans component to the target UNIX computer, remove the component to prepare for a clean installation.

1. On the target UNIX computer, navigate to the parent directory in which the RedBeans component was deployed (for example, `\U2\U2WDE\nnn`, where `nnn` is the version number you installed).
2. In the parent directory, delete the entire subdirectory that contains the RedBeans component.

Copying RedBeans components

To copy the RedBeans files to a UNIX computer and configure the environment, do the following steps:

1. **Locate the RedBeans folder in the Web DE installation folder.**
2. **Transfer the RedBeans.jar and asjava.jar files to the UNIX computer on which your web server software is running.**
3. **Place the RedBeans.jar and asjava.jar files in the library location required by your web server.**
4. **Add RedBeans.jar and asjava.jar to the CLASSPATH environment variable.**
5. **Locate and open the rgw5.ini file in the RedBeans folder.** Set the path in the `SUWDE` environment variable, and then copy the `rgw5.ini` file to that path on your UNIX computer.
6. **Edit the contents of the rgw5.ini file to match the server, user, account path, and other details to those of your system.** For steps, see [Configuring the RedBack gateway, on page 41](#).
Chapter 6: Installing the stand-alone JSP RBOScope testing tool

The following sections describe the procedure for installing the stand-alone Java Server Pages (JSP) RBOScope testing tool.

JSP RBOScope testing tool overview

Web DE includes a JSP-based version of the RBOScope testing tool. This stand-alone tool provides an environment for testing RBOs outside of the Web Designer. The JSP RBOScope testing tool can be deployed in your preferred JSP web server and run through any web browser.

This version of the RBOScope testing tool does not require users to have administrative permissions on the RedBack Object Server computer and does not require the Web Designer.

The JSP RBOScope testing tool is contained in the RedBeans directory of the Web DE product download directory, in the `rboscope.war` file. The web application archive (WAR) file contains the following essential files:

- `RBOScopeError.jsp`
- `RBOScopeLogin.jsp`
- `RBOScopeMain.jsp`
- `RBOScopeQuery.jsp`
- `WEB-INF/lib/asjava.jar`
- `WEB-INF/lib/rbtools.jar`
- `WEB-INF/lib/RedBeans.jar`

Installing the JSP RBOScope testing tool

To set up an environment for testing RBOs outside of the Web Designer, install the JSP RBOScope testing tool on your web server computer.

Prerequisites

JSP web server software (such as Apache Tomcat or IBM WebSphere) must be installed on the web server computer. For help with installing or setting up the web server software, consult the product documentation for the JSP web server.

Note: The following steps show Apache Tomcat as an example.

Procedure

1. Deploy the `rboscope.war` file:
   a. In the Deploy section of the Apache Tomcat Web Application Manager, browse to the `rboscope.war` file on the Web DE product download directory, as shown in the figure.
Installing the JSP RBOScope testing tool

Figure 6: Apache Tomcat Web Application Manager

b. Click **Deploy** to deploy the contents of the **rboscope.war** file to the Apache Tomcat server.

2. In the Path column, verify that the Apache Tomcat Applications list includes the “rboscope” application, as shown in the following figure:

Figure 7: Applications list

<table>
<thead>
<tr>
<th>Path</th>
<th>Display Name</th>
<th>Running</th>
<th>Sessions</th>
<th>Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>/</td>
<td>Welcome to Tomcat</td>
<td>true</td>
<td>0</td>
<td>Start, Stop, Reload, Undeploy</td>
</tr>
<tr>
<td>/docs</td>
<td>Tomcat Documentation</td>
<td>true</td>
<td>0</td>
<td>Start, Stop, Reload, Undeploy</td>
</tr>
<tr>
<td>/manager</td>
<td>Tomcat Manager Application</td>
<td>true</td>
<td>1</td>
<td>Start, Stop, Reload, Undeploy</td>
</tr>
<tr>
<td>/rboscope</td>
<td>JSP RBOScope</td>
<td>true</td>
<td>0</td>
<td>Start, Stop, Reload, Undeploy</td>
</tr>
</tbody>
</table>

The JSP RBOScope testing tool is ready to use for testing RBOs outside of the Web Designer. For information about using the stand-alone JSP RBOScope testing tool, see *Web Designer User’s Guide*. 
Chapter 6: Installing the stand-alone JSP RBOScope testing tool

Configuring the encoding of the JSP RBOScope testing tool

You can configure how the JSP RBOScope testing tool handles encoding by updating the `rgw5.ini` file. When configured to a specific encoding type, the JSP RBOScope testing tool will support any session encoding and display the information correctly.

1. Open the `rgw5.ini` file in a text editor.
2. In the `[accountname]` section, add the following:
   
   \[ encoding=type, where type is the encoding type. For example, UTF-8, or ISO 8859-1. \]
3. Save and close the file.
Chapter 7: Authorizing Web DE

The following sections describe how to authorize Web DE and to clear license violations.

Authorizing Web DE

The RedBack Object Server installation process on Windows uses the information you enter to automatically license Web DE for a 10-day trial period. Within this trial period, you must authorize Web DE. You can authorize Web DE from an option in the Web Designer.

Prerequisites

Within the 10-day trial period, you must enter your license details on the U2 Tech Connect site at https://u2tc.rocketsoftware.com to obtain an authorization code. After you have an authorization code, you can enter it through the Manage Object Server License option in the Web Designer.

Procedure

1. Click Start → All Programs → Rocket U2 → Web Development Environment → U2 Web Designer.
2. In the U2 Resource view, double-click the name of the U2 server to which you want to connect.
3. Enter your U2 server credentials and click Connect.
4. Expand the tree view for a Web DE-enabled account and double-click U2 Web DE Elements.
5. Enter your credentials for the RedBack Object Server.
   The default user name of the administrator is rbadmin and the default password is redback.
   
   **Remember:** Change the administrator name and password for increased security. For information on changing the password of the administrator or any other user, see the information about creating and maintaining users in the Administrator’s Guide.

6. Click Finish.
7. In the U2 Resource view, right-click U2 Web DE Elements, and select Manage Object Server License.
8. In the RBO Editor pane, the License tabbed page contains the current license details. Change the details as needed.

Figure 8: License tab
a. If the RedBack Object Server was deployed to this computer using the Deploy to U2 Server wizard, change the temporary serial number displayed in the U2 Web DE Serial Number field to the serial number issued to you by U2.

b. In the **Number of Webshares** field, change the number of webshares covered by your license, if necessary.

c. In the **Authorization Code** field, enter the authorization code generated on the Rocket Business Connect site at [https://rbc.rocketsoftware.com](https://rbc.rocketsoftware.com).

d. To save the changes in the License tabbed page, click the **Save** icon.

### Clearing violation codes

Violation codes are issued when abnormalities are detected with a Web DE installation, such as a missing or tampered license file. You cannot access the RedBack Object Server until the violation is cleared.

1. Contact Global Technical Support and provide the violation code to obtain a response code. Alternatively for violation code 3 or 11, you can obtain a response code on the Rocket Business Connect site at [https://rbc.rocketsoftware.com](https://rbc.rocketsoftware.com).

2. Enter the response code, and click **OK**.

   If the RedBack Object Server is in violation status, the **Clear License Violation** dialog box appears, prompting you to enter the violation response code there.
Chapter 8: Configuring the RedBack gateway

The following sections describe the steps for configuring the gateway by modifying the configuration file.

RedBack gateway overview

The key role of the RedBack gateway is to manage data transfer between the gateway and the RedBack Object Server.

The RedBack gateway components run on the web server or the client computer. The gateway uses information in the `rgw5.ini` configuration file for the following purposes:

- Global parameters for performance monitoring and logging. These parameters apply to all Web DE-enabled U2 accounts on the RedBack Object Server computer.
- Reference to the location at which each U2 account resides.

Configuring the RedBack gateway

Global and account-related parameters provide the settings that the RedBack gateway requires to make connections to the RedBack Object Server. After installing Web DE, configure the Java Scheduler, and then configure the gateway.

Prerequisites

Before you can configure the RedBack gateway, you need to have Web DE installed with the RedBack Object Server and Java Scheduler deployed to a Windows or UNIX computer. A U2 server must also be defined, as described in Administrator’s Guide. If you are using the Java Scheduler to process HTTP requests, configure the Java Scheduler on Windows or UNIX.

Procedure

1. Modifying default global settings, on page 41
2. Modifying account settings, on page 43
3. Inserting an account in the gateway configuration file, on page 42

Modifying default global settings

The `rgw5.ini` file contains global parameters that apply to all U2 accounts on the RedBack Object Server computer.

1. On the Windows computer on which Web DE is installed, start the Web Designer.
2. In the U2 Resource view, connect to the U2 server on which the RedBack Object Server component is installed.
3. On the Windows computer on which Web DE is installed, start the Web Designer.
4. From the Tools menu, select Edit Gateway Configuration.
5. From the Open window, browse to the folder containing the `rgw5.ini` file, and double-click it to open this configuration file in the Editor pane.
On Windows computers, the `rgw5.ini` file is located in the directory specified by the U2WDE environment variable. The installation directory is `C:U2\U2WDe.nn` by default, where `nnn` is the version number you installed.

On UNIX computers, the `rgw5.ini` file must be created manually or created on your local client computer and then copied to a UNIX system. The preferred location for this file is in `/etc`.

6. In the `rgw5.ini` tabbed page, edit the global parameters:
   a. Maximize the `rgw5.ini` tabbed page.
   b. In the Global Settings area at the bottom of the `rgw5.ini` tabbed page, set the performance monitor option. To log performance statistics for all Web DE-enabled U2 accounts, in the Performance Monitor area, select the `_perfstats` check box. The `Perfstats.log` file includes the same statistics available from the Web DE performance monitor API, such as wait time and number of requests processed for each account.
   c. In the Logging Settings area at the bottom of the `rgw5.ini` tabbed page, select the check box for each type of information to write to the client log files for Web DE-enabled U2 accounts:
      - `RedBeans_accountname.log`
      - `RedPages_accountname_processid.log`, where `processid` is the ID of the Windows process with `RedPagesNet.dll` loaded.

The following table describes each parameter. Each parameter is selected by default. If you do not want the parameter, clear the check box.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Panic</td>
<td>To log unrecoverable errors, select the <code>Panic</code> check box.</td>
</tr>
<tr>
<td>Error</td>
<td>To log serious errors, select the <code>Error</code> check box.</td>
</tr>
<tr>
<td>Warning</td>
<td>To log warning messages, select the <code>Warning</code> check box.</td>
</tr>
<tr>
<td>Inf</td>
<td>To log informational messages, select the <code>Inf</code> check box.</td>
</tr>
<tr>
<td>Init</td>
<td>To log initialization/termination messages, select the <code>Init</code> check box.</td>
</tr>
<tr>
<td>Trace</td>
<td>To log program flow tracing messages, select the <code>Trace</code> check box.</td>
</tr>
<tr>
<td>Times</td>
<td>To log the time (in milliseconds) for running the RBO method calls, select the <code>Times</code> check box.</td>
</tr>
</tbody>
</table>

7. To save the changes to the `rgw5.ini` file, select `File` → `Save`.

**Inserting an account in the gateway configuration file**

After creating a U2 account, you can insert a reference to the account in the `rgw5.ini` file and configure the account settings.

1. In the `rgw5.ini` tabbed page, add an account to the `JavaScheduler.ini` file.
   a. Maximize the `rgw5.ini` tabbed page.
   b. Right-click anywhere in the `Selection/Property Name` grid, and select `Insert Account`.
   c. Complete the form in the wizard, following the same instructions for step 1 as described in the `Modifying account settings, on page 43`.

2. To apply the new account details, click `Finish`.
3. In the `rgw5.ini` tabbed page, review the changes in the new account section and make corrections if necessary.
4. To save the changes to the `rgw5.ini` file, select `File` → `Save`. 
Modifying account settings

Each account section in the `rgw5.ini` file contains parameters that apply to connections made to the associated U2 account only.

**Tip:** The `rbexamples` section is an example that shows the values to enter for each account connection.

The RedBack gateway components, which run on the web server or the client computer, use the `rgw5.ini` file as a reference to the location of each U2 account.

If you increase the number of accounts, the load on the RedBack Object Server and the Web server might increase.

1. In the `rgw5.ini` tabbed page, edit the parameters in the account section:
   a. Maximize the `rgw5.ini` tabbed page and expand the section for the account to be modified.
   b. Double-click any property in the associated account section.
      In the Establish server details for this account connection page of the account editor wizard, enter server details for the account connection.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Account name</strong></td>
<td>Enter the name to use for this account section in the <code>rgw5.ini</code> file. This name is typically the same as the account name that has been assigned to the account.</td>
</tr>
<tr>
<td><strong>Server</strong></td>
<td>Enter the name or IP address of the RedBack Object Server computer for this account.</td>
</tr>
<tr>
<td><strong>Server Type</strong></td>
<td>Select the type of U2 data server running on the RedBack Object Server computer: <strong>UniData</strong> or <strong>UniVerse</strong>.</td>
</tr>
</tbody>
</table>

In the Connection area, select an option for allocating licenses when making connections between the account and the RedBack Object Server, as described in the following table.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Use Web Shares</strong></td>
<td>Select this option and enter the associated details:</td>
</tr>
<tr>
<td></td>
<td>1. In the Scheduler port field, enter the port number on which the Java Scheduler accepts requests from this account.</td>
</tr>
<tr>
<td></td>
<td>2. In the Scheduler Connections area, enter or select the minimum and maximum number of responders available to the account.</td>
</tr>
</tbody>
</table>
Option | Description
--- | ---
Use Connection Pooling | Select this option and enter the associated details:
1. **Account path** field, enter the path to the account directory on the RedBack Object Server computer.
2. **User** field, enter the operating system-level user ID for the RedBack Object Server computer.
3. **Password | Verify Password** fields, enter the password associated with the user ID and reenter the password to verify it.
4. To test the settings entered for the account connection parameters, click **Test Connection Pooling Settings**. If the connection succeeds, the wizard returns the message “Successful connection.” If the connection fails, make changes to correct the error condition and test again.
5. In the Database Connections area, enter or select the minimum and maximum number of pooled connections available to the account.
   - In the **Minimum** field, the minimum number of pooled connections cannot exceed the total number of connection pooling licenses on the U2 database server computer.
   - In the **Maximum** field, the maximum number of pooled connections cannot exceed the total number of connection pooling licenses on the U2 database server computer.

**Password | Verify Password** | Enter the password associated with the user ID, and reenter the password to verify it.

d. In the **Local log folder** field, enter or browse to the full path of the directory in which to store log files on the client computer.
e. To modify the account’s connection pool parameters, click **Next** and enter details in the Establish account connection details page in the wizard.

**Table 6: Connection pool parameters**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Idle Remove Threshold</strong></td>
<td>To set an idle remove threshold parameter for the account, select the <strong>Idle Remove Threshold</strong> check box. In the field, enter or select the number of milliseconds that a connection pool thread can remain idle before it is flagged for removal.</td>
</tr>
<tr>
<td><strong>Idle Remove Exec Interval</strong></td>
<td>To set an idle remove execution interval for the account, select the <strong>Idle Remove Exec Interval</strong> check box. In the field, enter or select the number of milliseconds of the interval at which UOJ runs an executable to remove from a connection pool any threads that have exceeded the idle threshold.</td>
</tr>
<tr>
<td><strong>Open session</strong></td>
<td>To set an open session parameter for the account, select the <strong>Open session</strong> check box. In the field, enter or select the number of milliseconds for which UOJ attempts to acquire a session from the connection pool before timing out.</td>
</tr>
</tbody>
</table>

f. To modify optional parameters for the account, click **Next** and complete the Set optional account and connection properties page in the wizard.
Table 7: Optional parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| Deactivate account                        | To deactivate the U2 account, select this check box. You can clear the check box to reactivate the account later, if needed.  
Deactivating a U2 account prevents it from consuming server or connection pooling licenses. It disables use of the account’s RBOs in web applications and in the RBOScope testing tool. However, users can access a deactivated account from the U2 Resource view and work with its RBOs in the RBO Editor. |
| Enable SSL for this account’s connection(s) to server | To use the SSL protocol when making connections to the U2 server, select this check box.                                                    |
| Enable server-side logging for this account | To log activity for this account on the U2 server, select this check box.                                                                  |
| Create the uoj_trace.log file             | To keep a log of UOJ processes for the account, select this check box.                                                                      |

2. To apply the changes, click **Finish**.
3. In the rgw5.ini tabbed page, review the changes in the account section and make corrections if necessary.
4. To save the changes to the **rgw5.ini** file, select **File → Save**.