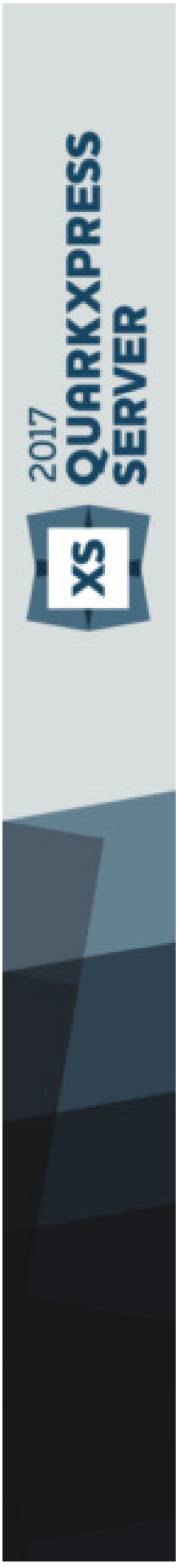

QuarkXPress Server 2017
March 2018 Update ReadMe

Contents

Introduction 1

Related Documents 2

System Requirements 3

System requirements: Mac OS X 3

System requirements: Windows 4

Installing QuarkXPress Server 5

Installing QuarkXPress Server: Mac OS X 5

Installing QuarkXPress Server: Windows 5

Upgrading QuarkXPress Server 7

Deploying QuarkXPress Server Manager 8

Installing QuarkXPress Server Manager 8

Installing QuarkXPress Server Manager: Mac OS X 9

Installing QuarkXPress Server Manager: Windows 9

Deploying QXPSM in a shared Spring context 9

Deploying in an external Web container 12

Deploying QXP Server in external Tomcat 12

Deploying QXPSM in external Tomcat 13

Deploying QXP Server in WebSphere 14

Prerequisites 14

Installing the IBM-WebSphere developer version 15

Configure the Java 7 SDK in the IBM WebSphere 15

Start IBM WebSphere 16

QXPS WebApp deployment into IBM WebSphere UI 16

QXPS WebApp silent deployment into IBM WebSphere 18

Deploying QXP Server in WebLogic 18

QXPS WebApp deployment into WebLogic UI 20

Introduction to QLA Server 21

Uninstalling 22

Uninstalling: Mac OS X 22

Uninstalling: Windows....22

Changes in this version....23
New for QuarkXPress Server 2017....23
New for QuarkXPress Server 2017 October Update....24
New for QuarkXPress Server 2017 March 2018 Update....24

Known and resolved issues....25

Contacting Quark....26
In the Americas....26
Outside the Americas....26

Legal notices....27

Introduction

QuarkXPress® Server is a high-fidelity content-composition and rendering server application. It is built on multiple technologies derived from QuarkXPress. Through a network connection and a browser, QuarkXPress Server can return rendered versions of QuarkXPress layouts and QuarkCopyDesk articles in a variety of formats. This document provides information about QuarkXPress Server software.

With QuarkXPress Server 2017, our release process was changed to provide maximum value to our customers and the earliest possible access to new software updates.

- **Software First Candidate** (FC) - our golden master software release. This release includes software components plus an initial documentation set:
    - QuarkXPress Server ReadMe
    - QuarkXPress Server Known and Resolved Issues
- **Software First Article** (FA) – the formal public software release. This includes the same golden master software release plus all documentation. This release typically arrives six weeks after the FC release.

Related Documents

For more information about using the QuarkXPress Server, you may need to reference the other QuarkXPress Server documentation/resources listed below.

Document/Item Name	Description	Where to find
QuarkXPress Server User Guide	A guide written for the end user of the QuarkXPress Server. Contains all of the QuarkXPress Server User Guide QXPS documentation, including documentation for QXPSM, and Modifier schema. This guide also contains all of the information which previously was contained in the Web Integration Guide.	*https://support.quark.documentation.Server*
XDK	This is the C/C++ XTensions Developer Kit. It is comprised of XTension Headers, libraries and documentation.	Once installed, SDK is available at: `[installed directory]/Documents/SDK`
QXPSM Javadocs	This set of Javadocs provides information about the QuarkXPress Server Manager java interface. This documentation is available with the QuarkXPress Server Manager installer.	Once installed, Javadocs are available from the home page of QuarkXPress Server Manager: *http://<servernameorip>:8090/*

System Requirements

This section lists baseline minimum requirements for evaluation and demonstration. Like any enterprise system, performance is highly contextual based on data complexity, number of requests, and IT infrastructure. Your specific requirements for production shall require development as part of the overall solution planning process.

For more information, please contact *http://enterprisesupport@quark.com* or your Quark sales contact.

The following is a list of factors to consider when scaling up for production:

Number of requests per minute

- Content richness
  - Rendered page count
  - Number of raster images rendered
  - Number of vector images rendered
  - Number of images with transparency
  - Number of tables
  - Number of tables which span across multiple pages
  - Number of landscape pages
  - Number of footnotes in table cells
  - Number of cross references
  - Number of image callouts or text callouts
- Output richness
  - PDF/print output styles
  - Digital output, e.g. number and size of interactive artifacts

System requirements: Mac OS X

Software

- Mac OS X 10.11 (El Capitan™), macOS 10.12 (Sierra™) or macOS 10.13.x (High Sierra™)

- Java J2SE™ Runtime Environment (JRE™) 8.0

Hardware

- 1 processor core and 4 sub-renderers with 4GB of RAM for each renderer*
- 10GB of additional disk space per renderer for temporary files

*Like any enterprise system, performance is highly contextual based on data complexity, number of requests, and IT infrastructure. Contact your account manager for sizing your organization's publishing needs.

System requirements: Windows

Software

- Microsoft® Windows® Server 2012 R2 64-bit and Windows® Server 2016 64-bit.
- .NET Framework 4.6.2 SP1 or later
- Java Development Kit (JDK) 1.8
- Java J2SE™ Runtime Environment (JRE™) 8.0
- Quark License Administrator (QLA) 4.5.0.2

Hardware

- *1 processor core and 4 sub-renderers with 4GB of RAM for each renderer
- 10GB of additional disk space per renderer for temporary files
- 5GB to 10GB hard disk space for installation
- CPU: 4 Core Intel Xeon, 64-bit
- TCP/IP network (Gigabit Ethernet recommended)

Power policy settings

QuarkXPress Server achieves better throughput under "High Performance" power policy setting as compared to a "Balanced" power policy setting.

- If QuarkXPress Server is installed on a bare metal machine, the power policy should be configured to "High Performance" on that machine.
- If QuarkXPress Server is installed on a virtual machine (Hyper V or ESXi), the power policy settings of the host machine should be configured to "High Performance" so that all virtual machines running on this host inherit this property.

*Like any enterprise system, performance is highly contextual based on data complexity, number of requests, and IT infrastructure. Contact your account manager for sizing your organization's publishing needs.

Installing QuarkXPress Server

The topics below describe how to install QuarkXPress Server.

Installing QuarkXPress Server: Mac OS X

To install QuarkXPress Server:

1. Disable any virus protection software. If you are concerned about a virus infecting your computer, run a virus check on the installer file and then disable your virus detection software.
2. Double-click the Installer icon and follow the instructions on the screen.
3. When prompted, enter the validation code.

Do not discard the validation code. You will need it if you have to reinstall.

4. In the **Quark License Server Details** screen, enter the IP address and port number of the main Quark® License Administrator (QLA) server in the **License Server Host Name/IP Address** and **License Server Port** fields. Optionally, enter the IP address and port number of a backup QLA server in the **Backup License Server Host Name/IP Address** and **Backup License Server Port** fields.
5. In the **Port and Folder Settings** screen, enter a value in the **Port** field and specify the location of the QuarkXPress Server document pool.
6. In the **Subrenderer Configuration** screen, enter the number of renderers you want to run. The number of available renderers is determined by the QLA license file.
7. Complete the installation.

After its initial launch, QuarkXPress Server creates a "Preferences" folder for each renderer. For example, if you have two renderers, QuarkXPress Server creates two folders named "Preferences sub1" and "Preferences sub2". QuarkXPress Server creates preferences files at the following location:
`[drive]/Users/[username]/Library/Preferences/Quark/QuarkXPress Server Renderer`

Installing QuarkXPress Server: Windows

To install QuarkXPress Server:

1. Disable any virus protection software. If you are concerned about a virus infecting your computer, run a virus check on the installer file and then disable your virus detection software.
2. Double-click the Installer icon and follow the instructions on the screen.
3. In the **Software Verification** screen, enter the validation code.

Do not discard the validation code. You will need it if you have to reinstall.

4. When the **License Server Details** screen displays, enter the IP address and port number of the main Quark License Administrator (QLA) server in the **Host Name or IP Address** and **License Server Port** fields. Optionally, enter the IP address and port number of a backup QLA server in the **Backup Host Name or IP Address** and **Backup License Server Port** fields.
5. When the **Configuration Settings** screen displays:
   - Enter a port number for QuarkXPress Server.
   - To install QuarkXPress Server as an automatic service that launches when the computer starts up, check **Install QuarkXPress Server as a service**. You can choose **Local** or **Domain** from the **Choose System** drop-down menu. If you choose **Domain**, enter values in the **Domain**, **User**, and **Password** fields.
6. When the Destination screen displays:
   - The screen displays the drive and folder where QuarkXPress Server will be installed. To install QuarkXPress Server elsewhere, click **Browse** and navigate to the desired folder.
   - In the **Number of renderers to launch** field, enter the number of renderers you want to run. The number of available renderers is determined by the QLA license file.
7. Complete the installation.
8. When the installation completes, click **Finish**.

After its initial launch, QuarkXPress Server creates a "Preferences" folder for each renderer. For example, if you have two renderers, QuarkXPress Server creates two folders named "Preferences sub1" and "Preferences sub2". The `getserverinfo` request returns the path of the QuarkXPress Server preferences under the element `PREFERENCESPATH`.

The URL for getting the QuarkXPress Server Preferences Path: `http://localhost:8080/getserverinfo`

Upgrading QuarkXPress Server

1. Uninstall your current version of QuarkXPress Server as described in "*Uninstalling*".
2. Install the new version of the software as described in "*Installing QuarkXPress Server*".

Deploying QuarkXPress Server Manager

There are three ways to deploy QuarkXPress Server Manager:

- As a standalone server. To deploy QuarkXPress Server Manager in this manner, install the software as described in "*Installing QuarkXPress Server Manager*," then double-click the `server/QXPSMServerStart.command` (Mac OS X) or `server/QXPSMServerStart.bat` (Windows) file in the QuarkXPress Server Manager application folder.
- As an independent Web application in an existing Tomcat Web container. For more information, see "*Deploying QXPSM in external Tomcat*."
- As a Web application that shares the same Spring context. For more information, see "*Deploying QXPSM in a shared Spring context*."

Installing QuarkXPress Server Manager

QuarkXPress® Server Manager coordinates rendering requests in a multiple QuarkXPress Server instance environment, using load-balancing, fail-safe, and caching capabilities to determine which server in the pool can best process each document request. If the first QuarkXPress Server instance in an array is unable to process a render request, QuarkXPress Server Manager sends the request to another instance. QuarkXPress Server Manager also provides an object-oriented programming environment for working with QuarkXPress Server, and it exposes a set of Web services and Web objects that can be used in any SOAP-compatible environments such as .NET, JavaScript™, Java™, and Objective-C.

The QuarkXPress Server Manager installation includes three primary components:

- QuarkXPress Server Manager Server (the server application that provides services for load-balancing, fault-tolerance, and the SOAP server that allows third-party communication with QuarkXPress Server instances through Web services).
- The QuarkXPress Server Manager Web-based administrative client.
- The QuarkXPress Server SDK, which includes Web services API documentation; Web service stubs and samples for .NET, Java, and Objective-C; and an Extensibility Tool for generating custom stubs. This component is helpful for developing client-side solutions using the QuarkXPress Server Manager Web services and servlet interface.

Installing QuarkXPress Server Manager: Mac OS X

You must uninstall the previous version of QuarkXPress Server Manager before installing this version. See "*Uninstalling*" for more information.

To install QuarkXPress Server Manager:

1. Disable any virus protection software. If you are concerned about a virus infecting your computer, run a virus check on the installer file and then disable your virus detection software.
2. Double-click the Installer icon and follow the instructions on the screen.
3. In the **QuarkXPress Server Manager Server Port Settings** screen, enter a port number for the QuarkXPress Server Manager server's Tomcat server and specify the location of the QuarkXPress Server Manager cache folder. (The cache folder will store temporary copies of content generated by the QuarkXPress Server instances that are managed by this QuarkXPress Server Manager server.)
4. Complete the installation.

Installing QuarkXPress Server Manager: Windows

You must uninstall the previous version of QuarkXPress Server Manager before installing this version. See "*Uninstalling*" for more information.

To install QuarkXPress Server Manager:

1. Disable any virus protection software. If you are concerned about a virus infecting your computer, run a virus check on the installer file and then disable your virus detection software.
2. Double-click the Installer icon and follow the instructions on the screen.
3. When the **Destination Folder** screen displays:
   - The screen displays the drive and folder where QuarkXPress Server will be installed. To install QuarkXPress Server elsewhere, click **Browse** and navigate to the desired folder.
   - Enter a port number for QuarkXPress Server Manager.
   - Specify a cache folder location. (The cache folder will store temporary copies of content generated by the QuarkXPress Server instances that are managed by this QuarkXPress Server Manager server.)
   - To install QuarkXPress Server Manager Server as a service, check **Install QuarkXPress Server Manager Server as a Service**.
4. Complete the installation.

Deploying QXPSM in a shared Spring context

In this type of deployment, QuarkXPress Server Manager can be embedded in a Spring application and can share the same Spring container instance and Spring context. This deployment model also allows the consuming of QuarkXPress Server Manager services (`RequestService` and `AdminService`) as POJOs (plain old Java objects).

This type of deployment model is possible only when the application in which QuarkXPress Server Manager is to be embedded is a Spring application and is deployed with an embedded or external Tomcat Web container.

To install QuarkXPress Server Manager in a shared Spring context:

1. Copy the `qxpsm` and `qxpsmadmin` Web application to the webapps folder for your existing application, so that QuarkXPress Server Manager can be recognized as a Web application in the existing Web container.
   - Set `cacheFolderContextFile` to the path of the "cache.xml" file relative to the current working directory.
   - Set s`erverXmlFilePath` to the path of the "Server.xml" file relative to the current working directory.
2. Copy all QXPSM-dependent jar files from `[QXPSM_Home]/Server/dependencies and [QXPSM_Home]/Server/lib` to a common location, then add all of these jar files to the application classpath so that they are available to other Web applications that want to use the QuarkXPress Server Manager API locally.
3. Import the QuarkXPress Server Manager Spring context file, "ManagerContainerConfig.xml," into the existing application Spring context.
4. Create a Java class named `CustomContextLoaderListener` that extends `org.springframework.web.context.ContextLoaderListener`. Override the method `loadParentContext()` as shown here:

```
public class CustomContextLoaderListener extends
org.springframework.web.context.ContextLoaderListener
{
   private static final Logger logger =
Logger.getLogger(GlobalContextLoaderListener.class);
   public void contextInitialized(ServletContextEvent
servletContextEvent)
   {
      super.contextInitialized(servletContextEvent);
      String contextName =
servletContextEvent.getServletContext().getServletContextName(
);
   }
   protected ContextLoader createContextLoader()
   {
      return new CustomContextLoader();
   }
   class CustomContextLoader extends ContextLoader
   {
      protected ApplicationContext
loadParentContext(ServletContext servletContext)
      {

//=============================================================
=
         //Here, returns the spring context to be used as
parent context

//=============================================================
=
         return
com.xyz.myapplication.MyApplicationContext.getContext();
      }
      protected WebApplicationContext
```

```
createWebApplicationContext(ServletContext servletContext,
ApplicationContext       parent)
      {
         return
super.createWebApplicationContext(servletContext, parent);
      }
   }
}
```

5. Create a file named “applicationContext.xml” in `webapps/qxpsm/WEB-INF` and fill it with the following content:

```
<?xml version=”1.0” encoding=”UTF-8”?>
<!DOCTYPE beans PUBLIC “-//SPRING//DTD BEAN//EN”
“http://www.springframework.org/dtd/spring-beans.dtd”>
<beans default-lazy-init=”false” default-autowire=”no”
default-dependency-check=”none”>
</beans>
```

6. Open the QuarkXPress Server Manager “web.xml” file and update `contextConfigLocation` and `listener-class` as follows:

```
<context-param>
<param-name>contextConfigLocation</param-name>
<param-value>/WEB-INF/applicationContext.xml</param-value>
</context-param>
<listener>
<listener-class>CustomContextLoaderListener</listener-class>
</listener>
```

7. If the existing application is already using Spring’s `PropertyPlaceholderConfigurer`, open “ManagerContainerConfig.xml” and specify the parent in the `placeholderConfig` bean as follows:

```
<bean id=” placeholderConfig “ parent=”placeholderConfig”
class=”org.springframework.beans.factory.config.PropertyPlaceh
olderConfigurer”>
```

Deploying in an external Web container

For development, and simple test deployments, QuarkXPress Server and QuarkXPress Server Manager are provided pre-installed in an instance of Apache Tomcat. However, for production deployment alongside other Web applications, QuarkXPress Server and QuarkXPress Server Manager can be deployed in an existing Tomcat, WebSphere, or JBoss Web container.

Deploying QXP Server in external Tomcat

To install QuarkXPress Server as an independent Web application in an existing Tomcat Web container:

1. Verify that Tomcat 7.0.77 is installed.
2. Set up the following environment variables:
   - `JAVA_HOME` = Java installation folder. For example: `E:\Program Files\Java\jdk1.8.0_121`
   - `CATALINA_HOME` = Tomcat installation folder. For example: `E:\apache-tomcat-7.0.77`
3. Copy contents of the `[QXPS_BUILD]/webapps` folder to the `[TOMCAT_HOME]/webapps` folder.
4. Copy the `[QXPS_BUILD]/conf` folder to `[TOMCAT_HOME]/bin`.
5. *(Mac OS X only)* Open the file `[Tomcat_Home]/bin/conf/ServerApp.properties` and make the following changes:
   - Uncomment the Mac OS X settings section:

     ```
     #qxpserver.serverRendererPath=./webapps/ROOT/QuarkXPressServer
     Renderer/QuarkXPress Server.app/Contents/MacOS
     #qxpserver.serverRendererExecutableName=QuarkXPress Server
     #qxpserver.DocumentRootFolder=/QuarkXPress Server
     Documents
     ```

   - Comment out the Windows settings section:

     ```
     qxpserver.serverRendererExecutableName=QuarkXPress Server
     Renderer.exe
     qxpserver.DocumentRootFolder=C\:\\QuarkXPress Server
     Documents
     ```

6. In the `[TOMCAT_HOME]/bin/conf/Qla.properties` file, set the host name, port number, and serial number of your instance of QLA Server.

- Enter the IP address or hostname of the QLA Server in the `QlaServer.machinename=` field.
- Enter the port number of the QLA Server in the `QlaServer.port=` field. If you have a backup QLA server, enter the IP address (or hostname) and port number in the `Backup.QlaServer.machinename=` and `Backup.QlaServer.port=` fields.

7. *(Windows OS only)* Execute `vcredist_x64.exe` provided with the build before starting QXPS.
8. Copy the `Documents` folder to `[TOMCAT_HOME]`.
9. Open `[TOMCAT_HOME]/conf/server.xml` and add the following inside the `<Host name="localhost" appBase="webapps".....>` tag:

```
<Context path="/qxps/documents"
docBase="../Documents/Documentation" >
<Manager pathname="‟/>
</Context>Uncomment the Mac OS X settings section.
```

10. Make the following change in the respective config file:
    - Remove tag `<import resource="classpath:com/quark/azure/auth/AzureAuthContext.xml"/>` from the `$(apache-tomcat_DIR)\bin\conf\QXPSLoadComponents.xml` file.
    - Remove tag `<import resource="classpath:com/quark/qxps/azure/config/AzureConfiguration.xml"/>` from the `$(apache-tomcat_DIR)\bin\conf\QXPSLoadComponents.xml` file.
    - Remove tag `<import resource="classpath:com/quark/qxps/azure/config/AzureConfiguration.xml"/>` from the `$(apache-tomcat_DIR)\bin\conf\QXPSLoadComponents.xml` file.
    - Remove tag `<value>file:./conf/AzureAuthenticate.properties</value>` from the `$(apache-tomcat_DIR)\`
    - `bin\conf\PropertiesFilePlaceholders.xml` file.
    - Remove the following tags from the `$(apache-tomcat_DIR)\webapps\qxps\WEB-INF\qxps-servlet.xml` file:

```
<value>file:./conf/AzureAuthenticate.properties</value>,
<property name="authManager"
ref="azureAuthenticationManager"/>,
<prop
key="/secure/azureauth">azureAuthReplyController</prop>,
<property name="authResultCBHandler"
ref="qxpsAzureAuthResultCBHandler"/>,
<property name="authManager"
ref="azureAuthenticationManager"/>
```

Deploying QXPSM in external Tomcat

To install QuarkXPress Server Manager as an independent Web application in an existing Tomcat Web container:

1. Verify that Tomcat 7.0.77 is installed.
2. Locate the Tomcat installation folder (referred to here as `[Tomcat_Home]`).
3. Locate the QuarkXPress Server Manager external Tomcat installation folder (referred to here as `[QXPSM_Tomcat]`).
4. Drag the contents of `[QXPSM_Tomcat]/conf` to `[Tomcat_Home]/conf`.
5. Drag the contents of `[QXPSM_Tomcat]/webapps` to `[Tomcat_Home]/webapps`.
6. Drag `[QXPSM_Tomcat]/qxpsm` to `[Tomcat_Home]`.
7. Open the file `[Tomcat_Home]/conf/ManagerContainerConfig.xml` and make the following changes in the `ContainerAdapter` bean:
   - Set `cacheFolderContextFile` to the path of the "cache.xml" file relative to the current working directory.
   - Set `serverXmlFilePath` to the path of the "Server.xml" file relative to the current working directory.
8. Open the file `[Tomcat_Home]/conf/Catalina.properties` and specify the path of the shared class loader. For example: `shared.loader=${catalina.home}/qxpsm/lib/*.jar,${catalina.home}/conf`
9. Open the file `[Tomcat_Home]/conf/Catalina/localhost/cache.xml` and set the docbase attribute to the path of the appropriate cache folder.
10. Double-click the "startup.sh" shell file (Mac OS) or the "startup.bat" file (Windows). These files can be found in the `TOMCAT_HOME/bin` folder.

Deploying QXP Server in WebSphere

To install QuarkXPress Server as an independent Web application in an IBM WebSphere Web container:

Prerequisites

1. Download the IBM Installation Manager 1.6.2 software and extract its contents to your local folder (for example, `IBM_MANAGER_DIR`).
2. Download the IBM WebSphere 8.5.5.0 developer version software. This is in 3 parts in an archived (.zip) format.
3. Extract all 3 parts of IBM WebSphere 8.5.5.0 to your local folder (for example, `IBM_WEBSPHERE_DIR`).
4. Download the *SDK for Java 1.7* for IBM WebSphere 8.5.5.0, This is in 3 parts in an archived (.zip) format.
5. Extract all 3 parts of *SDK for Java 1.7* to your local folder (for example, `SDK-FOR-Java-1.7_DIR`).

Installing the IBM-WebSphere developer version

1. Go to the `$(IBM_MANAGER_DIR)` folder and run "install.exe" with administrator privilege.
2. Launch the IBM Installation Manager 1.6.2 with administrator privilege.
3. Go to **File->Preferences**. The **Preferences** dialog displays. Select the **Repositories** option on the left side.
4. Click the **Add Repository** button from the right side. Browse to and select the "respository.config" file from the `$(IBM_WEBSPHERE_DIR)/was.repo.8550.developers.ilan_part1` folder.
5. Click **Apply** and then click **OK**.
6. Go to the IBM Installation Manager, select the **Install** option.

   IBM WebSphere 8.5.5.0 will be installed in your Program Files location. `C:\Program Files (x86)\IBM\WebSphere`.
7. During the installation, you will be asked to enter the configuration file for Part 2 and Part 3 of IBM WebSphere. Select the files from their respective folders:
   - `$(IBM_WEBSPHERE_DIR)/was.repo.8550.developers.ilan_part2`
   - `$(IBM_WEBSPHERE_DIR)/was.repo.8550.developers.ilan_part3`
8. Go to **Start->WebSphere Customization Toolbox**. Select the **Create** option on the right side.

   The Profile Management Tool dialog displays.
9. Select `Application Server` as the WebSphere Application Server and click **Next**.
10. Select T**ypical file creation** and click **Next**.
11. Select **Disable Administrative Security** and click **Next**.
12. Select **Create a profile**.

Configure the Java 7 SDK in the IBM WebSphere

1. Open the cmd window. Navigate to the `$(WS_INSTALLED_DIR)/AppServer/bin` folder.
2. Type the command `managesdk.bat -listAvailable`.
3. Launch the IBM installation manager.
4. Go to **File->Preferences**. The **Preferences** dialog displays. Select the **Repositories** option on the left side.
5. Click the **Add Repository** button from the right side. Browse to and select the "repository.config" file from the `$(SDK-FOR-Java-1.7_DIR)/was.repo.8550.java7_part1` folder.
6. Click **Apply** and then click **OK**.
7. Go to the IBM Installation Manager, select the **Install** option.

IBM WebSphere 8.5.5.0 will be installed in your Program Files location. `C:\Program Files (x86)\IBM\WebSphere`.

8. During the installation, you will be asked to enter the configuration file for Part 2 and Part 3 of IBM WebSphere. Select the files from their respective folders:
   - `$(SDK-FOR-Java-1.7_DIR)/was.repo.8550.java7_part2`
   - `$(SDK-FOR-Java-1.7_DIR)/was.repo.8550.java7_part3`
9. The Java7 SDK for IBM WebSphere will be installed successfully. Check it at the `$(WS_INSTALLED_DIR)/AppServer/java_1.7_64` folder
10. Open the cmd window. Navigate to the `$(WS_INSTALLED_DIR)/AppServer/bin` folder.
11. Type the command `managesdk.bat -listAvailable`.
12. Type the command `managesdk.bat -enableProfile -profileName AppSrv01-sdkname 1.7_64`. This binds IBM WebSphere into Java7 runtime.

Start IBM WebSphere

1. Enter `First Step.bat` from the start menu or from `$(WS_INSTALLED_DIR)/AppServer/profiles/AppSrv01/firststeps/firststeps.bat`.

   The **WebSphere Application server** dialog displays.
2. Start the Server.

QXPS WebApp deployment into IBM WebSphere UI

1. Copy the `[QXPS_BUILD]/conf` folder to WebSphere's current directory (for example, `$(WS_INSTALLED_DIR)/AppServer/profiles/AppSrv01`. Open "ServerApp.properties" and update the following:
   - `qxpswebserver.embeddedwebcontainer= false`
   - `qxps.webapp.context.name= /qxps`
2. Open the "QXPSLoadComponents.properties" file located at `$(WS_INSTALLED_DIR)/AppServer/profiles/AppSrv01/conf` and remove the following entries:

```
<import
resource="classpath:com/quark/azure/auth/AzureAuthContext.xml"
/>
<import
resource="classpath:com/quark/qxps/azure/config/AzureConfigura
tion.xml"/>
<import
resource="classpath:com/quark/qxps/azure/config/AzureConfigura
tion.xml"/>
```

3. Open the "PropertiesFilePlaceholders.properties" files located at `$(WS_INSTALLED_DIR)/AppServer/profiles/AppSrv01/conf` and remove the following entries:
   - `<value>file:./conf/AzureAuthenticate.properties</value>`

4. In the `[TOMCAT_HOME]/bin/conf/Qla.properties` file, set the host name, port number, and serial number of your instance of QLA Server.
   - Set `QlaServer.machinename` to the IP address or hostname of the QLA server.
   - Set `QlaServer.port` to the port number of the QLA server.
   - If you have a backup QLA server, set `Backup.QlaServer.machinename` and `Backup.QlaServer.port` to its IP address or hostname and port number.
5. Extract the "EAR (qxps-2017.ear)" file. Open the "qxps-servlet.xml" file located at `$(QXPS-EAR_DIR)\qxps-2017\source\qxps\WEB-INF\` and remove the following entries:

```
<value>file:./conf/AzureAuthenticate.properties</value>
<property name="authManager"
ref="azureAuthenticationManager"/>
<prop key="/secure/azureauth">azureAuthReplyController</prop>
<property name="authResultCBHandler"
ref="qxpsAzureAuthResultCBHandler"/>
<property name="authManager"
ref="azureAuthenticationManager"/>
```

6. Create the WAR bundle by using the "CreateQXPS_EAR.bat" batch file located at the extracted folder: `$(QXPS-EAR_DIR)`
7. Open *http://localhost:9080/ibm/console/* in a browser and login.
8. Select the **Application** tab from the right and select **New Application**.
9. Select **New Enterprise Application**. When asked for the path to the new application, choose the file and click **Next**.
10. When asked "**How do you want to install the application?**", select Fast Path and click **Next**.
11. Change the application name if required and click **Next**.
12. Select both the modules and click **Next**.
13. Select both the web modules and click **Next**.
14. Select the qxpsadmin module and click **Next** and then **Finish**.
15. Save the changes to the master configuration.
16. Select the **Server** tab from the right and select **WebSphere Application Servers> server1 > Web server plug-in properties**.
17. Set the maximum number of connections that can be handled by the application server to 150.
18. Save the changes.
19. Select the **Application** tab from the right and select **Application Types -> WebSphere Enterprise applications**. Select **qxps** and **Start**.

QXPS WebApp silent deployment into IBM WebSphere

1. Copy the `[QXPS_BUILD]/conf` folder to WebSphere's current directory (for example, `$(WS_INSTALLED_DIR)/AppServer/profiles/AppSrv01. Open ServerApp properties` and update the following:
   - `qxpswebserver.embeddedwebcontainer=` false
   - `qxps.webapp.context.name=` /qxps
2. In the `[TOMCAT_HOME]/bin/conf/Qla.properties` file, set the host name, port number, and serial number of your instance of QLA Server.
   - Set `QlaServer.machinename` to the IP address or hostname of the QLA server.
   - Set `QlaServer.port` to the port number of the QLA server.
   - If you have a backup QLA server, set `Backup.QlaServer.machinename` and `Backup.QlaServer.port` to its IP address or hostname and port number.
3. Save the following script as "install.jython":

```
earLoc='C:\QXPSTestAutomation\WebSphere\$(EAR_NAME)
appName='$(EAR_NAME)'
print AdminApp.install(earLoc, ['-MapWebModToVH', [['.*',
'.*',
'default_host']]])
AdminConfig.save()
```

4. Save the following script as "unnstall.jython":

```
appName=' $(EAR_NAME)'
print AdminApp.uninstall(appName)
AdminConfig.save()
```

5. Open the cmd window with admin privilege, go to the `$(WS_INSTALLED_DIR)/AppServer/profiles/AppSrv01/bin` folder and run the following commands:

```
wsadmin.bat -f '$(PATH_TO_INSTALL_JYTHON_SCRIPT)' -lang jython
```

6. Restart IBM WebSphere to launch the QuarkXPress Server.

Deploying QXP Server in WebLogic

To install QuarkXPress Server as an independent Web application in an Oracle WebLogic Web container:

1. Download the WebLogic Server 12c from *http://www.oracle.com*.
2. Extract the downloaded archive and go through the steps given in the "README.txt" file in the `[$wls]/wlsxxxx` folder.

   Your domain username and password are those that you entered while setting up WebLogic.
3. Start the WebLogic Server using the `startWebLogic`.cmd cmd in the `$wls]/wlsxxxx/user_projects/domains/base_domain/` folder.
4. Copy the `[QXPS_BUILD]/conf` folder to WebLogic's current directory (for example, `[$wls]/wlsxxxx/user_projects/domains/base_domain`).

5. Open the "ServerApp.properties" file in the `[$wls]/wlsxxxx/user_projects/domains/base_domain/conf` folder and update the following entries:

   - `qxpswebserver.embeddedwebcontainer= false`
   - `qxps.webapp.context.name= /qxps`

6. Open the "'QXPSLoadComponents.properties"' file located at `[$wls]/wlsxxxx/user_projects/domains/base_domain /conf` and remove the following entries:

```
<import
resource="classpath:com/quark/azure/auth/AzureAuthContext.xml"
/>
<import
resource="classpath:com/quark/qxps/azure/config/AzureConfigura
tion.xml"/>
<import
resource="classpath:com/quark/qxps/azure/config/AzureConfigura
tion.xml"/>
```

7. Open the "PropertiesFilePlaceholders.properties"' file located at `[$wls]/wlsxxxx/user_projects/domains/base_domain /conf` and remove the following entries:

   - `<value>file:./conf/AzureAuthenticate.properties</value>`

8. In the `[$wls]/wlsxxxx/ user_projects/domains/base_domain /conf/Qla.properties` file, set the host name, port number, and serial number of your instance of QLA Server:

   - Set `QlaServer.machinename` to the IP address or hostname of the QLA server.
   - Set `QlaServer.port` to the port number of the QLA server.
   - If you have a backup QLA server, set `Backup.QlaServer.machinename` and `Backup.QlaServer.port` to its IP address or hostname and port number.

9. If you are running WebLogic on Windows, execute the file "vcredist_x86.exe"

10. Extract the "'EAR (qxps-2017.ear)"' file. Open the "qxps-servlet.xml"' file located at `$(QXPS-EAR_DIR)\qxps-2017\source\qxps\WEB-INF\` and remove the following entries:

```
<value>file:./conf/AzureAuthenticate.properties</value>
<property name="authManager"
ref="azureAuthenticationManager"/>
<prop key="/secure/azureauth">azureAuthReplyController</prop>
<property name="authResultCBHandler"
ref="qxpsAzureAuthResultCBHandler"/>
<property name="authManager"
ref="azureAuthenticationManager"/>
```

11. Create the WAR bundle by using the "CreateQXPS_EAR.bat"' batch file located in the extracted folder: `$(QXPS-EAR_DIR)`

QXPS WebApp deployment into WebLogic UI

1. Navigate to the administrative console page for your server (*http://localhost:7001/console*) and log in using the domain credentials that you entered during the WebLogic setup in "Deploying QXP Server in WebLogic".
2. Go to domain structure > Deployments > Install, then select the QuarkXPress Server EAR file (`[QXPS_BUILD]\QXPS-EAR\qxps-2017.ear`) and select **Next**.
3. Select the **Install this deployment as an application option** button and click **Next**.
4. Click **Next**, leaving all default settings.
5. Select the **Yes, take me to the deployment's configuration screen** button and click **Finish**.
6. Click **Save** to finish the installation and save the configuration settings.
7. Go to the **Deployments** option listed on the left panel under **Domain Structure**.

   Observe the status of the qxps-2015 webapp. It will be shown in an active state after a successful installation.

Introduction to QLA Server

QLA Server is the software component that issues licenses to the Quark application clients. After you install Quark License Administrator Server, download your license file from the Quark Web site. The server can be any computer on the network that meets the minimum system requirements. The server monitors license usage over the network and grants licenses to the client computers.

QLA clients are the computers on which Quark applications are installed — for example, QuarkXPress, Quark Publishing System® (QPS®), or QuarkXPress Server software. When a user attempts to launch a Quark application on a client computer, the client sends a request to the QLA Server. The server, which is always listening for requests, determines whether there is an available license. If there is, then the server grants a license and the client computer can launch the Quark application. If no licenses are available, the server does not grant a license. When this occurs, the Quark application does not launch.

In the QLA Control Panel application, the **Configuration Options** tab lets you specify how QLA monitors licenses. This tab includes the **Server Configuration** area and the **Extended Check-out Configuration** area.

In the **Server Configuration** area:

- The **Heartbeat Interval (Seconds)** value lets you specify the number of seconds between attempts by the client to reach the server. If a Quark application fails to successfully connect to the server within this time, then that application automatically attempts to reconnect.
- The **Heartbeat Retries** field lets you specify the number of times a client can attempt to make a connection before reporting a connection failure.

For example: If the **Heartbeat Interval (Seconds)** value is 3600 (one hour) and the **Heartbeat Retries** value is 10, the system waits 10 hours before failing. If the client is able to successfully connect to the server on or before the tenth attempt, outstanding licenses are restored. If not, the application checks in all allocated licenses.

Uninstalling

If the uninstaller detects a running instance of the application, an alert displays. Close the application before you begin to uninstall.

Uninstalling: Mac OS X

To uninstall QuarkXPress Server, open the `Applications` folder on your hard drive (or the folder in which QuarkXPress Server was installed) and drag the `QuarkXPress Server` folder to the trash.

To uninstall QuarkXPress Server Manager, open the `Applications` folder on your hard drive (or the folder in which QuarkXPress Server Manager was installed) and drag the `QuarkXPress Server Manager` folder to the trash.

Uninstalling: Windows

To uninstall QuarkXPress Server or QuarkXPress Server Manager:

1. Choose **Start > All Programs > QuarkXPress Server 2017 > Uninstall** or **Start > All Programs > QuarkXPress Server Manager > Uninstall**.
2. Click **Yes**.

   The uninstaller gathers the information necessary for uninstalling and then an alert displays to notify you that the contents of the application folder will be deleted.
3. Click **Yes** to continue or **No** to cancel.

Changes in this version

This section will cover the various changes introduced in different versions of QuarkXPress Server.

For details about all of these changes, see *A Guide to QuarkXPress Server*.

New for QuarkXPress Server 2017

Changes in version 13.0 included the following:

- QuarkXPress Server is compatible with QuarkXPress 2017 including support for the following QuarkXPress 2017 features:
  - Export as Responsive HTML5 Publication (Multi-device Output)
  - Paragraph and text shading
  - Column Flow - Column Split and Span
- Added support for Item Styles
- Rendering metrics and monitoring - telemetry records metrics for executed rendering jobs:
  - Average & Peak Response Time
  - Busy Period
  - Utilization %
  - Total Transactions Processed by Renderers
  - Rendering Throughput
  - Arrival Rate of rendering requests
  - Average & Peak Queue Time
  - Average, Current, and Peak Queue Length
  - Failed Requests and Error %
  - Average Time per Page
  - Total Pages Rendered
  - Total Images Processed
  - Total Renditions to PDF, HTML5, QXPDOC, etc.

- Request timeout and ability to collect data for failed requests in a rescue folder to simplify troubleshooting
- Transaction-level font management

New for QuarkXPress Server 2017 October Update

Changes in version 13.1 included the following:

- QuarkXPress Server is compatible with QuarkXPress 13.1 and QuarkCopyDesk 13.1.
- Configuration changes:
  - New option to configure the path for the QuarkXPress Server "PublishingRescueFolder" to store publishing data to accelerate troubleshooting of publishing failures:
  - Two ways to configure the path
    - Edit the "PublishingUtilContext.xml" file (`[installpath]/Server/publishing`) to set the path as a value for the rescueFolderPath property (requires server restart)
    - Use JConsole (no server restart required)

New for QuarkXPress Server 2017 March 2018 Update

Changes in version 13.2.2 included the following:

- Compatible with QuarkXPress and QuarkCopyDesk 13.2.1
- **Modifier Schema Changes**
  - Added the ability to represent master spreads and its items in XML rendition.
  - Added the ability to create or modify items on master spreads using modifier XML.
  - Added the ability to represent and modify gradients.
  - Added the ability to represent blend modes in XML and apply blend modes using modifier XML.
  - Added the ability to represent page and spread guides in modifier deconstruct XML.
  - Added the ability to represent lock type in deconstruct XML and lock items on the layout using Modifier XML.
  - Added the ability to represent text stroke in deconstruct XML and apply text stroke using Modifier XML.
  - Added the ability to apply a hyperlink in a table cell.

Known and resolved issues

For lists of known issues and resolved issues in this version of the software, visit the Quark Web site at *http://www.quark.com*, *http://euro.quark.com*, or *http://japan.quark.com*. Product documentation, including lists of known and resolved issues, is available under **Support > Product Documentation**.

Contacting Quark

The support portal allows you to log support tickets, track tickets, receive status notifications, chat with a technical support representative, search the Knowledge Base, and access product documentation.

With direct access to documentation across all Quark software - from QuarkXPress and App Studio to Quark Enterprise Solutions - you can find answers to your questions at your convenience. Our support team is also available to help, either through our support portal or via phone for our maintenance contract customers.

If you are a Quark customer and have a current maintenance or support contract your account has already been created for you using your registered email address. If you do not have a support contract you can purchase a single support incident to get your problem resolved. If you have purchased or registered a supported product, you are eligible for free support for the first 90 days.

In the Americas

For more details, please check out our support website *http://www.quark.com/support*.

Outside the Americas

For countries outside the Americas, please visit the following sites to access your support account:

Support Website

- France - *http://www.quark.com/fr/support*
- Germany - *http://www.quark.com/de/support*

Legal notices

©2022 Quark Software Inc. and its licensors. All rights reserved.

Protected by the following United States Patents: 5,541,991; 5,907,704; 6,005,560; 6,052,514; 6,081,262; 6,633,666 B2; 6,947,959 B1; 6,940,518 B2; 7,116,843; 7,463,793; and other patents pending.

Quark, the Quark logo, and QuarkXPress are trademarks or registered trademarks of Quark Software Inc. and its affiliates in the U.S. and/or other countries. All other marks are the property of their respective owners.