



QuarkXPress Server 2016 ReadMe

Contents

QuarkXPress Server 2016 ReadMe.....	4
Related Documents.....	5
System requirements.....	6
System requirements: Mac OS X.....	7
System requirements: Windows.....	7
Installing QuarkXPress Server.....	9
Installing QuarkXPress Server: Mac OS X.....	9
Installing QuarkXPress Server: Windows.....	9
Upgrading QuarkXPress Server.....	11
Deploying QuarkXPress Server Manager.....	12
Installing QuarkXPress Server Manager.....	12
Installing QuarkXPress Server Manager: Mac OS X.....	13
Installing QuarkXPress Server Manager: Windows.....	13
Deploying QXPSM in a shared Spring context.....	14
Deploying in an external Web container.....	16
Deploying QXP Server in external Tomcat.....	16
Deploying QXPSM in external Tomcat.....	17
Deploying QXP Server in WebSphere.....	18
Prerequisites.....	18
Installing the IBM-WebSphere developer version.....	18
Configure the Java 7 SDK in the IBM WebSphere.....	19
Start IBM WebSphere.....	19
QXPS WebApp deployment into IBM WebSphere UI.....	20
QXPS WebApp silent deployment into IBM WebSphere.....	20
Deploying QXP Server in WebLogic.....	21
QXPS WebApp deployment into WebLogic UI.....	22
Introduction to QLA Server.....	23
Uninstalling.....	24

Uninstalling: Mac OS X.....24
Uninstalling: Windows.....24

Changes in this version.....25
New for 2016.....25

Known and resolved issues.....27

Contacting Quark.....28
In the Americas.....28
Outside the Americas.....28

Legal notices.....29

QuarkXPress Server 2016 ReadMe

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.

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 QXPS documentation, including documentation for QXPSM, and Modifier schema. This guide also contains all of the information which previously was contained in the Web Intregation Guide.	support.quark.com/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: <a href="http://<servernameorip>:8090/">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 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 10.8.x (Mountain Lion™), Mac OS X 10.9 (Mavericks™), or Mac OS X 10.10 (Yosemite™), or Mac OS X 10.11 (El Capitan™)
- Java J2SE™ Runtime Environment (JRE™) 7.0 or 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 to specify your organization's publishing needs.

System requirements: Windows**Software**

- Microsoft® Windows® Server 2008 R2 64-bit and Windows® Server 2012 R2 64-bit.
- .NET Framework 4.6.2 SP1
- Java Development Kit (JDK) 1.7
- 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.

SYSTEM REQUIREMENTS

- ➔ *Like any enterprise system, performance is highly contextual based on data complexity, number of requests, and IT infrastructure. Contact your account manager to specify 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/[user name]/Library/Preferences/Quark/QuarkXPress Server
Renderer 10
```

Installing QuarkXPress Server: Windows

To install QuarkXPress Server:

INSTALLING 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** field.
 - 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

To install QuarkXPress Server Manager:

- ➔ You must uninstall the previous version of QuarkXPress Server Manager before installing this version. See "[Uninstalling](#)" for more information.

- 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 `serverXmlFilePath` 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, return 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 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.PropertyPlaceholderConfigurer">
```

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 7.0.61 is installed.
- 2 Set up the following environment variables:
 - `JAVA_HOME` = Java installation folder. For example: `E:\Program Files\Java\jdk1.7.0_21`
 - `CATALINA_HOME` = Tomcat installation folder. For example: `E:\apache-tomcat-7.0.61`
- 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.
```

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 6.0.18 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
```

DEPLOYING IN AN EXTERNAL WEB CONTAINER

- 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 `$(IBM_MANAGER_DIR)` folder and run "install.exe" with administrator privilege.
- 2 Launch the IBM Installation Manager 1.6.2 with administrator privilege.
- 3 Go to **File->Preferences**. The **Preferences** dialog displays. Select the **Repositories** option on the left side.
- 4 Click the **Add Repository** button from the right side. Browse to and select the `respoistory.config` file from the `$(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 **Typical 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 `$(WS_INSTALLED_DIR)/AppServer/bin` folder.
- 2 Type the command `managesdk.bat -listAvailable`.
- 3 Launch the IBM installation manager.
- 4 Go to **File->Preferences**. The **Preferences** dialog displays. Select the **Repositories** option on the left side.
- 5 Click the **Add Repository** button from the right side. Browse to and select the `respoistory.config` file from the `$(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.

DEPLOYING IN AN EXTERNAL WEB CONTAINER

- 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 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 Open <http://localhost:9080/ibm/console/> in a browser and login.

- 4 Select the **Application** tab from the right and select **New Application**.

- 5 Select **New Enterprise Application**. When asked for the path to the new application, choose the file and click **Next**.

- 6 When asked "How do you want to install the application?", select **Fast Path** and click **Next**.

- 7 Change the application name if required and click **Next**.

- 8 Select both the modules and click **Next**.

- 9 Select both the web modules and click **Next**.

- 10 Select the `qxpsadmin` module and click **Next** and then **Finish**.

- 11 Save the changes to the master configuration.

- 12 Select the **Server** tab from the right and select **WebSphere Application Servers> server1 > Web server plug-in properties**.

- 13 Set the maximum number of connections that can be handled by the application server to 150.

- 14 Save the changes.

- 15 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" 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, `[$wls]/wlsxxxx/user_projects/domains/base_domain`).
 - 5 Open the ServerApp properties in the `[$wls]/wlsxxxx/user_projects/domains/base_domain /conf` folder and update the following entries:
 - `qxpswebserver.embeddedwebcontainer= false`

DEPLOYING IN AN EXTERNAL WEB CONTAINER

- `qxps.webapp.context.name= /qxps`

6 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.

7 If you are running WebLogic on Windows, execute the file `vcredist_x86.exe`

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-10.2.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 the **Save** button 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 with 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 12 > Uninstall** or **Start > All Programs > QuarkXPress Server Manager 10 > 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 2016

Changes in version 2016 Update included the following:

- QuarkXPress Server is compatible with QuarkXPress 2016.
- Introduced new attributes (`MINWIDTH`, `MAXWIDTH`, `SPAN`) under the `INLINETABLE` modifier. These new attributes provide greater control on table layout.
- Added support for index generation.
- Added support for equation publishing with Wiris Math Equations.
- Added support for table auto-fit rules.
- Added support for Azure Cloud Service configuration.
- Added support for using the DITA Open Toolkit as a publishing engine for your DITA content.
- Added the ability to apply a hyperlink on `INLINEBOX`.
- Added the ability to process nested `<PARAGRAPH>` elements.
- Added the ability to specify `ELLIPSIS/CLIP/CUSTOM STRING` for text overflow.
- Added new elements to Modifier Schema to support the cross-references feature introduced in QuarkXPress 2016.
- Introduced a new attribute (`TEXTOVERFLOW`) under the `STORY` modifier. Possible values are `ELLIPSIS` | `CLIP` | `CUSTOM STRING`. This attribute lets you control the representation of overflow text. The option `ELLIPSIS` renders an ellipsis "..." to represent over flow text. The option `CLIP` clips the text. The option `CUSTOM STRING` allows you to supply custom text. (e.g Continue Reading... or Read more...) to represent over flow text. These are particularly useful when you have a finite sized box designated to hold title text. If the title text is too lengthy to fit into the designated box on the Master Page, using the `TEXTOVERFLOW` attribute allows you to specify the behavior you want to see in the rendered output.

CHANGES IN THIS VERSION

- Added a new namespace ([HTML5](#)) and url parameter (webreader), that generate HTML5 output from QuarkXPress 2016 digital layouts.

API Changes

- Migration from Ver-11.2.3 to QuarkXPress & QuarkXPress Server 2016 (12.2)
 - Migration from Ver-11.2.2 to QuarkXPress & QuarkXPress Server 2016 (12.2)
 - Migration from Ver-11.2.1 to QuarkXPress & QuarkXPress Server 2015 (11.2.2)
 - Migration from Ver-11.1 to QuarkXPress & QuarkXPress Server 2015 (11.2.1)
- ➔ No migration anticipated unless you want to use new attributes. However, testing is recommended if you are using callouts ([CALLOUTANCHOR](#)).
- Migration from Ver-11.0. to QuarkXPress & QuarkXPress Server 2015 (11.1)
- ➔ No migration anticipated unless you want to use new attributes. However, testing is recommended if you are using callouts ([CALLOUTANCHOR](#)).

Recommendations for Migration:

Custom implementations done either in the XSL layer or by a Custom XTension to achieve the following listed use cases can be functionally migrated to product functionality:

- Make use of Paragraph Styles and associated Bullets and Outline Numbering in QuarkXPress to achieve section numbering.
- Generate a table of contents based upon the lists feature available in QuarkXPress. A TOC generated in this way would also possess the outline numbering as well.
- Rely on the cross-references functionality offered in QuarkXPress and QuarkXPress Server 2016 (v 12.2) to achieve use cases related to cross references with in the document. For example "Refer to Section 12.1.3 on page 15 above".
- Make use of the content variable functionality for the following use cases:
 - Running headers and footers.
 - Use **Last Page Number** content variable to achieve "Page 4 of 60".
- Migrate all of the pagination rules that might exist in XSL's to the QuarkXPress Publishing Template by re-defining the same rules using the pagination rule user Interface.
- Consider migrating the index terms marked in the source XML to map to the new INDEXTERM markup introduced in QuarkXPress Server 2016. The QuarkXPress Server 2016 automatically compiles all index terms in the document and builds the index list that gets appended at the end of the document.
- Migrate foot notes use cases (achieved using a custom XTension) based upon the QuarkXPress and QuarkXPress Server 2015/2016 feature built into the product.

Known and resolved issues

For lists of known issues and resolved issues in this version of the software, visit the Quark Web site at www.quark.com, euro.quark.com, or 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 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 - www.quark.com/fr/support
- Germany - www.quark.com/de/support

Legal notices

©2016 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.