

2015

**PUBLISHING
PLATFORM**



Quark Publishing Platform 2015 September 2016 Update System Administration Handbuch

Inhaltsverzeichnis

Einführung in die administrativen Anforderungen von Quark Publishing Platform. 6

Einsatz von Quark Publishing Platform Server für externe Server Container. 7

Einrichten von Umgebungsvariablen.....7

Konfigurieren der JVM.....7

Vorbereiten der EAR Datei für den Einsatz.....7

Einsatz mit IBM WebSphere.....8

Einsatz mit Oracle WebLogic.....10

Verwendung mit externem Tomcat.....11

Anforderungen und Setup.....11

Set-up von Tomcat unter Windows.....11

Serverstart und Verifizierung.....14

Anwendung von Quark Publishing Platform Server im Multiserver-Betrieb.... 14

Konfigurieren von Cluster Server Instanzen oder Knoten.....15

Konfigurieren von IIS als HTTP Load Balancer.....19

Konfigurieren von Apache Web Server als HTTP Loadbalancer.....21

Elasticsearch.....22

Download und Installation von Elasticsearch.....22

Configuring Platform server to leverage Elasticsearch.....23

Konfiguration von Elasticsearch für eine Multi-Server-Umgebung.....23

Starten von Elasticsearch.....24

Eine Schnittstelle für Elasticsearch.....25

Aktivieren des Secure Sockets Layer (SSL) für Quark Publishing Platform Server. 26

SSL-Unterstützung.....26

Aktivieren von SSL für Quark Publishing Platform Server.....26

Konfiguration von Platform Clients zur Anmeldung am SSL-aktiviertem Server.....27

Prüfen und Verwenden von SSL.....28

Quark Publishing Platform Server — Manuelle Konfiguration

.....2 9

Bearbeiten von „ServerApp.properties“29

Bearbeiten von „PublishingPool.properties“30

Konfigurieren von Quark Publishing Platform Renderer für die Verwendung mit Quark Publishing Platform. 31

Bearbeiten von "Qla.properties"31

Erweiterte Konfigurationsdateien.....32

Konfigurieren des Anzeigennamens des Benutzers im Bereich Benutzeraktivität im Admin Web Client. 32

Konfiguration von Platform Server für die Verwendung zum Publishing mit Quark XML Author–Sharepoint Adapter. 33

Konfiguration von Platform Server für die Verwendung zum Publishing mit Quark XML Author FileNet Adapter	33
JVM Memory Allocation unter Windows	33
Verwenden der Quark Publishing Platform Server Console oder des Quark Publishing Platform Server Windows Service	34
Verwenden von „Serverstartup.bat“	34
Konfigurieren der Windows Authentifizierung	34
Schritte zum Aktivieren von SSO in Platform Clients	35
Konfiguration des Web Client für Mixed-Mode-Authentifizierung in mehreren Schritten	36
Bearbeiten von „log4j.xml“	36
Hinzufügen von Benutzerinformationen zu protokollierten Ereignissen	36
Konfigurieren der Anmeldestufen	37
Modifizieren der Evaluierungseinstellungen für Abfragebenachrichtigungen	39
Datenbankeigenschaften	41
Transformationseigenschaften	42
Session-Timeout	42
Aktualisieren des Assetspeicher-Status	43
Umzug von Quark Publishing Platform Renderer	43
Integrieren von Quark Publishing Platform mit LDAP	44
Verwenden der Kerberos Authentifizierung	44
Verwenden der Einfachen Authentifizierung	45
Verbinden von LDAP Benutzerpasswörtern mit Quark Publishing Platform Server	45
Konfiguration von Platform für MSSQL Server Windows Authentifizierung	46
Beschränken der Workspace Browser Paletten	47
Ändern der Suchergebnis-Einstellungen	47
Einstellen der Erkennung individueller Inhaltstypen	47
Festlegen eines Standard-PDF-Ausgabestils	48
Einstellen der Delivery-Channel Anzeige im Webclient	48
Änderung der Unterscheidung zwischen Groß- und Kleinschreibung für Quark Publishing Platform Passwörter	48
Die Handhabung von Filtern und Index Service Einstellungen	49
Einstellungen für den Indexdienst	50
ASPOSE Filter	51
APS-Filter	52
POI Filter	52
Konfiguration der Verarbeitung von MS Office Dokumenten	53
Drosselungsmechanismus für parallele Requests durch OfficeService und ChartingService	54
QuarkXPress Server Filter	56
JAWS Filtereinstellungen	57
XML Author Filtereinstellungen	57
Die Verzeichnisse ImageMagick, Jaws und DITA OT	57
Konfiguration der Volltextindexierung	58
Charting Service	58
Integrieren von QLA mit Quark Publishing Platform	59
Dynamische Konfiguration	59
Aktivieren der IPTC-Unterstützung	60
Nur RMI und CORBA Clients	60
Ändern der von Quark Publishing Platform Server verwendeten Ports	60
Mehrere Netzwerkkarten	61

Firewalls mit NAT.....	63
Failover-Einrichtung.....	63
Verschlüsseln eines Klartext-Passworts.....	67
Aktivieren des Zwangs-Logoffs während Inaktivität.....	68
Konfiguration des WebAdmin zur Aktivierung des Zwangs-Logoffs.....	68
Konfiguration des Workspace zur Aktivierung des Zwangs-Logoffs.....	68
Konfiguration der Benachrichtigung.....	69
Quark Publishing Platform Web Client: Manuelle Konfiguration.70	
Konfiguration – Übersicht.....	70
Einstellungen der Anwendungsebene.....	70
Multi-Channel Voransicht.....	72
Attribute für den Bereich Allgemein.....	73
Rollenbasierte Konfiguration der Werkzeugleiste.....	74
Einschränken des Webclient-/Admin-Zugangs.....	76
Quark Publishing Platform Clients – Manuelle Konfiguration....78	
Erstellen und Verwalten der Log-Datei (nur Mac OS X).....	78
Erstellen und Verwalten der Log-Datei (nur Windows).....	78
Erstellen und verwalten der Log-Datei (Quark XML Author für Platform).....	80
Unterdrücken der Warnmeldung für Bedienungshilfen.....	81
Anzeigen von Versionskommentaren.....	81
Anzeigen von Vor- und Nachnamen.....	82
Ändern der Schrift und Größe von Voransichten (nur Windows).....	83
Einstellen der maximalen Anzahl Assets, die abgerufen werden (nur Windows).83	83
Festlegen, ob Chunked Encoding verwendet werden soll (nur Windows).....	83
Festlegen der Unterstützung für Lazy Loading bei der Suche (nur Windows).84	84
Einstellen der Lazy Loading Blockgröße (nur Windows).....	84
Einstellen der Service-Timeout-Werte für alle Remote Service-Referenzen (Nur Windows).84	84
Einstellen des Service Timeout-Werts für den Publishing Service (nur Windows).84	84
Festlegen der Schriftgröße der Textprobenzeile (nur Windows).....	85
Festlegen des Symbols für eine Dateierweiterung (nur Windows).....	85
Einstellen der Passwortspeicherung (nur Mac OS X).....	85
Verwenden von Mac Clients mit einem Proxyserver.....	86
Verwenden von Windows Clients mit einem Proxyserver.....	86
Spiegeln der Sammlungshierarchie beim Auschecken/Laden.....	86
Ausschalten des Spiegeln von Sammlungen: Mac OS X.....	86
Ausschalten des Spiegeln von Sammlungen: Windows.....	87
Konfigurieren der Publishing Channels.....	87
Konfigurieren der Publishing Channels: Mac OS X.....	87
Konfigurieren der Publishing Channels: Windows.....	88
Konfiguration der Delivery-Channels.....	88
Konfiguration der Delivery-Channels: Mac OS X.....	88
Konfiguration der Delivery-Channels: Windows.....	89
Einstellen der Vorgaben für Quark XML Author für Platform.....	89

Einstellen des Speicherorts für das Auschecken.....	89
Einstellen der Datei-Löschen-Vorgabe beim Sichern und Schließen.....	89
Einstellen der Vorgaben für die Schnellsuche.....	90
Einstellen der Vorgabe Versionskommentare anzeigen beim Sichern und Schließen.	90
Konfigurieren von Platform Adapters für Microsoft Office Komponenten zum Web Sharing.	91
Konfiguration des Quark Publishing Platform Adapters für Microsoft Office – Word.	91
Konfiguration des Quark Publishing Platform Adapters für Microsoft Office – Excel.	92
Konfigurieren von Quark Publishing Platform Adapter für Microsoft Office – PowerPoint.	92
Aktualisieren veröffentlichter ClickOnce-Bereitstellungen.....	92
Manuelle Konfiguration von QuarkXPress und QuarkCopyDesk XTensions....	94
Die Handhabung von Backups und Dateispeicherung.....	96
Backup von Quark Publishing Platform Server.....	96
Backup Ihrer Datenbank.....	96
Backup von Assets.....	97
Backup von Indexdateien (Volltextsuche).....	97
Wiederherstellen von Quark Publishing Platform Server.....	97
Wiederherstellen von Assets.....	97
Wiederherstellen der Quark Publishing Platform Server Datenbank.....	98
Wiederherstellen von Volltext-Indexen.....	98
Verschieben des Quark Publishing Platform Assetspeichers (Repository).....	98
Rechtliche Hinweise.....	100

Einführung in die administrativen Anforderungen von Quark Publishing Platform

Die Administrierung einer Quark® Publishing Platform™ Umgebung erfordert zusätzlich zur Nutzung der Einstellmöglichkeiten von Quark Publishing Platform eine umfangreiche Hard- und Softwarewartung. Diese Anleitung beschreibt die Aufgaben, die Administratoren im Hinblick auf die Sicherheit, die Einstellungen der Systemkonfiguration und die Integration mit anderen Systemen gestellt werden. Informationen zur Administrierung mithilfe der Benutzerschnittstelle von Quark Publishing Platform finden Sie in Ihrer *Anleitung zu Quark Publishing Platform*. Information zur Installation der Quark Publishing Platform Software finden Sie in der *Quark Publishing Platform ReadMe-Datei*.

Einsatz von Quark Publishing Platform Server für externe Server Container

Sie können Quark Publishing Platform Server in einer externen Tomcat Installation, in IBM WebSphere und in Oracle WebLogic betreiben.

Einrichten von Umgebungsvariablen

Stellen Sie als Vorbereitung für den Einsatz des Systems folgende Umgebungsvariablen ein.

➔ Wenn Sie Quark Publishing Platform Server als Dienst betreiben wollen, müssen diese Variablen als globale Systemumgebungsvariablen eingerichtet werden.

- `MAGICK_HOME`: `[ImageMagick Installationsverzeichnis]\bin`
- `MAGICK_FILTER_MODULE_PATH`: `%MAGICK_HOME%/modules/filters`
- `MAGICK_CODER_MODULE_PATH`: `%MAGICK_HOME%/modules/coders`

Konfigurieren der JVM

Unabhängig davon, welchen Application-Server Sie verwenden, müssen Sie den der JVM in Tomcat, IBM WebSphere und Oracle Weblogic zur Verfügung stehenden Speicher folgendermaßen vergrößern, damit er Quark Publishing Platform Server genügt:

- Stellen Sie die Mindestgröße für Java Heap Memory auf 2.048 MB ein.
- Stellen Sie die Mindestgröße für PermGenSpace auf 512 MB ein.

Vorbereiten der EAR Datei für den Einsatz

Ein Platform EAR Build besteht aus folgendem:

- Konfigurationsdateien für Platform Server.
- Binärdateien von Dependencies.
- Einem Tool zur Erzeugung der JEE EAR Datei.

EINSATZ VON QUARK PUBLISHING PLATFORM SERVER FÜR EXTERNE SERVER CONTAINER

Ein EAR Build enthält folgende Ordner:

- `conf`: Enthält Konfigurationsdateien für Platform Server.
- `database`: Enthält die Erstellungs-, Update- und Supportdateien der Datenbank.
- `ImageMagick`: Enthält ImageMagick Applikations-Binärdateien.
- `Jaws`: Enthält JAWS Applikations-Binärdateien.
- `DITA-OT1.6`: Enthält DITA OT Binärdateien.
- `publishing`: Enthält Konfigurations- und Ressourcen-Dateien für das Quark Publishing Platform Framework.

Die Platform Server Application besteht aus folgenden Modulen:

- `admin.war`: Die Quark Publishing Platform Administrations-Webanwendung.
- `workspace.war`: Die Quark Publishing Platform Workspace-Webanwendung.
- `webservices.war`: Das Webservices-Modul von Quark Publishing Platform Server.
- `rest.war`: Das REST Interface-Modul von Quark Publishing Platform Server.
- `messaging.war`: Die HTTP-basierte Messaging-Schnittstelle.
- `qxpsm.war`: QuarkXPress Server Manager.
- `qxpsmadmin.war`: Die QuarkXPress Server Manager Administrations-Webanwendung.
- `pluginwiris_engine.war`: Die Unterstützung für das wiris Plugin für mathematische Formeln.

So erstellen Sie das Quark Publishing Platform Server JEE Enterprise Archiv:

- 1 Kopieren Sie das Quark Publishing Platform EAR Build auf den Computer, auf dem der Application Server läuft.
- 2 Aktualisieren Sie die Konfigurationsdateien (einschließlich „Database.properties“, „managerconfig.xml“, „PluginContext.xml“, und „QLA.properties.xml“) wie unter „[Quark Publishing Platform Server — Manuelle Konfiguration](#)“ beschrieben.
- 3 Aktualisieren Sie die Werte für den JMX Agent **Benutzernamen** und das **Passwort** in „serverApp.properties“. Die Werte für **Benutzername** und **Passwort** sind dieselben, die für die Anmeldung an der WebSphere Admin-Console benutzt werden.
- 4 Führen Sie `RunEARUpdater.bat` aus. Es wird ein Ordner „For Deployment“ erzeugt, der eine einsatzfähige Quark Publishing Platform EAR Datei enthält.

Einsatz mit IBM WebSphere

So betreiben Sie IBM WebSphere (Unterstützt ist Version 8.5.5.8):

- 1 Kopieren Sie den Ordner „qpp“ aus „For Deployment“ in das Arbeitsverzeichnis der Application-Server-Instanz, in der die Datei EAR zum Einsatz kommen wird. Bei IBM WebSphere ist das typischerweise der zum Zielprofil gehörende Ordner.

- ➔ Um Ressourcen (Quark Publishing Platform oder andere) an IBM FileNet zu übergeben, fügen Sie folgendes am Ende der Datei `[working directory]\properties\wsjaas.conf` ein:

```
FileNetP8WSI {  
    com.filenet.api.util.WSILoginModule required;  
};
```

- 2 Starten Sie die Application Server Console von IBM WebSphere, indem Sie zu `http://localhost:port/ibm/console` navigieren.
- 3 Klicken Sie unter **Applications** auf **New Application**.
- 4 Klicken Sie auf **New Enterprise Application**.
- 5 Klicken Sie unter **Path to the new application** auf **Choose File** und wählen Sie die Datei `.ear` im Ordner „For Deployment“. (Weitere Informationen finden Sie unter *„Vorbereiten der EAR Datei für den Einsatz.“*)
- 6 Klicken Sie unter **Preparing for the application installation** auf **Next**.
- 7 Klicken Sie **Next** für die Schritte 1, 2, 3 und 4. Belassen Sie alle anderen Einstellungen in ihren Standardwerten.
- 8 Klicken Sie in Schritt 5 auf **Finish**.
- 9 Klicken Sie auf **Sichern**.
- 10 Navigieren Sie zu **Applications > Application Types > WebSphere enterprise applications** und klicken Sie erneut auf `qpp-server-12.0.ear`.
- 11 Klicken Sie unter **References** auf **Shared library references**.
- 12 Markieren Sie die Box neben `qpp-server-12.0.ear` und klicken Sie auf **Reference shared libraries**.
- 13 Klicken Sie im nächsten Bildschirm auf **New**.
- 14 Wählen Sie einen beliebigen Namen. Geben Sie als Classpath die relativen Pfade der Ordner für das Platform Publishing und ext ein.
So verweisen Sie beispielsweise auf dieses Verzeichnis:
`C:\ProgramFiles\IBM\WebSphere\AppServer\profiles\AppSrv01\qpp\publishing`
... in diesem Fall geben Sie `./qpp/publishing` und `./qpp/ext` ein.
- 15 Klicken Sie auf **Apply** oder **OK**, um die Einstellungen zu sichern.
- 16 Weisen Sie diese Library-Referenz der Anwendung `qpp-server-12.0.ear` zu.
- 17 Starten Sie die Webapp `qpp-server-12.0.ear`.
- 18 Klicken Sie auf das in der Liste angezeigte QPP EAR, wählen Sie anschließend **Class loading and update detection**, um die Konfiguration des Ladevorgangs zu ändern.
Navigieren Sie zu **Applications > Application Types > WebSphere enterprise applications**:
 - Ändern Sie die Class Loader Reihenfolge in **Classes loaded with local class loader first (parent last)**.

EINSATZ VON QUARK PUBLISHING PLATFORM SERVER FÜR EXTERNE SERVER CONTAINER

- Wählen Sie für die WAR Class Loader Methode die Option **Class loader for each WAR file in application**.
- Klicken Sie auf **Apply** und sichern Sie die Änderungen als Master-Konfiguration.
- Starten Sie QPP ear erneut.

19 Navigieren Sie zum Prüfen der Installation zu folgenden URLs:

```
http://localhost:9080/admin  
http://localhost:9080/workspace
```

Einsatz mit Oracle WebLogic

So funktioniert der Betrieb von Quark Publishing Platform Server in Oracle WebLogic (unterstützt ist Version 12.1):

- 1** Starten Sie den WebLogic Server.
- 2** Erstellen Sie eine neue WebLogic Domain mit dem Namen `QPP_DOMAIN`.
- 3** Zum Einstellen des JVM Parameters für die Festlegung des Speicherortes der Datei „krb5.conf“ öffnen Sie die Datei `[WEB_LOGIC_DIR]\domains\base_domain\bin\setDomainEnv.cmd` und fügen folgendes an das Ende an:

```
set JAVA_OPTIONS=%JAVA_OPTIONS%  
-Djava.security.krb5.conf=./qpp/conf/krb5.conf
```
- 4** Kopieren Sie den Ordner „qpp“ aus dem Ordner „For Deployment“ in das WebLogic Server Domain Zielverzeichnis (z. B. `C:\oracle\Middleware\user_projects\domains\QPP_DOMAIN`).
- 5** Setzen Sie die globale Systemumgebungsvariable `CLASSPATH` auf dem WebLogic Computer auf folgenden Wert:

```
[WEB_LOGIC_DIR]\domains\QPP_DOMAIN\qpp\publishing
```

und

```
[WEB_LOGIC_DIR]\domains\QPP_DOMAIN\qpp\ext
```
- 6** Starten Sie WebLogic neu.
- 7** Öffnen Sie die Oracle Weblogic Application Server Console unter Verwendung folgender URL:

```
http://localhost:port/console
```
- 8** Markieren Sie die Domain (`QPP_DOMAIN`) und klicken Sie auf das Register **Web Applications**.
- 9** Markieren Sie **Archived Real Path Enabled**.
- 10** Klicken Sie auf **Sichern**.
- 11** Klicken Sie unter **Domain Structure** auf **Deployments**.
- 12** Klicken Sie rechts unter **Deployments** auf **Install** und geben Sie den Pfad zur Datei `.ear` im Ordner "For Deployment" an. (Weitere Informationen finden Sie unter *„Vorbereiten der EAR Datei für den Einsatz“*.)

- 13 Klicken Sie im nächsten Bildschirm auf **Weiter** bei gewählter Standardoption (**Install this deployment as an application**).
- 14 Klicken Sie im nächsten Bildschirm auf **Weiter**.
- 15 Klicken Sie auf **Fertigstellen**.
- 16 Klicken Sie auf **Sichern**.
- 17 Starten Sie die Quark Publishing Platform Web Application.
- 18 Navigieren Sie zum Prüfen der Installation zu folgenden URLs:
`http://localhost:7001/admin` `http://localhost:7001/workspace`

Verwendung mit externem Tomcat

Der von der Apache Software Foundation entwickelte Apache Tomcat™ dient als Standard-Referenzimplementierung für die Java™ Servlet und JavaServer™ Pages Technologien. *Tomcat™* ist ein Servlet-Container für die Verwaltung von Web-Applications.

Wenn Sie die Standalone-Version von Quark Publishing Platform Server installieren, wird eine Instanz von Tomcat in die Java Virtual Machine (JVM™) von Quark Publishing Platform Server eingebettet, um Quark Publishing Platform Webanwendungen wie Quark Publishing Platform Web Client zu verwalten.

Wenn Sie jedoch bereits einen Tomcat Server für andere Webanwendungen nutzen und möchten, dass die Quark Publishing Platform Webanwendungen den vorhandenen Tomcat Server verwenden, können Sie Quark Publishing Platform Server mit dieser Instanz von Tomcat betreiben. Die Verwendung von Quark Publishing Platform Server mit Ihrem vorhandenen (also: externen) Tomcat Server bedeutet, dass Sie auf dem Servercomputer keinen separaten Quark Publishing Platform Server Prozess ausführen müssen. Wenn Sie Quark Publishing Platform Server mit Ihrem externen Tomcat betreiben möchten, können Sie ein anders Build verwenden, das sich in „Server (External Web Container)“ in Ihrem Software-Paket befindet.

Anforderungen und Setup

Quark Publishing Platform Server benötigt JVM 1.5.x oder 1.6.x und Apache Tomcat 7.0.41, um als Webserver konfiguriert zu werden. Der Computer, auf dem Tomcat ausgeführt wird, sollte ein 64-Bit Computer sein. Sie können Quark Publishing Platform Server einer vorhandenen Tomcat-Installation hinzufügen.

Set-up von Tomcat unter Windows

- ➔ Der weiter unten erwähnte, existierende Apache Tomcat Installationsordner wird `TOMCAT_HOME` genannt.

So setzen Sie Quark Publishing Platform in einer vom Tomcat Installer erzeugten, externen Tomcat-Installation ein:

- 1 Kopieren Sie den Ordner `[QPP_BUILD]/qpp` in den Ordner `[TOMCAT_HOME]`.

- 2 Kopieren Sie den Inhalt des Ordners `[QPP_BUILD]/webapps` in den Ordner `[TOMCAT_HOME]/webapps`.
- 3 Falls der Ordner `[TOMCAT_HOME]/endorsed` nicht vorhanden ist, erstellen Sie ihn.
- 4 Kopieren Sie den Inhalt des Ordners `[QPP_BUILD]/endorsed` in den Ordner `[TOMCAT_HOME]/endorsed`.
- 5 Wenn Tomcat mit dem Installer installiert wurde, müssen Sie in den Dateien `[TOMCAT_HOME]/qpp/publishing/AS-Busdoc.xslt`, `[TOMCAT_HOME]/qpp/publishing/BusDoc2QCD.xslt`, `[TOMCAT_HOME]/qpp/publishing/BusDoc2QXPS.xslt` und `[TOMCAT_HOME]/qpp/publishing/SmartDoc2QXPS.xslt` folgende Pfade wie unten beschreiben aktualisieren:

```
<xsl:include href="./qpp/publishing/xref-dita-anchors.xslt"/>
<xsl:include href="./qpp/publishing/AS-StyleSheets.xslt"/>
<xsl:include href="./qpp/publishing/AS-Transformations.xslt"/>
<xsl:include href="./qpp/publishing/BusDocsWordTableStyles.xslt"/>
```

- 6 Wenn Tomcat mit Binärcodes im Einsatz ist, müssen Sie in den Dateien `[TOMCAT_HOME]/qpp/publishing/AS-Busdoc.xslt`, `[TOMCAT_HOME]/qpp/publishing/BusDoc2QCD.xslt`, `[TOMCAT_HOME]/qpp/publishing/BusDoc2QXPS.xslt` und `[TOMCAT_HOME]/qpp/publishing/SmartDoc2QXPS.xslt` folgende Pfade aktualisieren:

```
<xsl:include href=" ../qpp/publishing/xref-dita-anchors.xslt"/>
<xsl:include href=" ../qpp/publishing/AS-StyleSheets.xslt"/>
<xsl:include href=" ../qpp/publishing/AS-Transformations.xslt"/>
<xsl:include href=" ../qpp/publishing/BusDocsWordTableStyles.xslt"/>
```

- 7 In der Datei `[TOMCAT_HOME]/qpp/conf/ServerApp.properties`:
 - Geben Sie den Wert `webServer.port` ein, den Sie für Tomcat konfiguriert haben (z. B. 8080).
 - Setzen Sie den Wert `webServer.embeddedWebContainer` auf `false`.
- 8 In der Datei `[TOMCAT_HOME]/qpp/conf/ManagerConfig.xml`:
 - Geben Sie die **IP address** oder den **Host-Namen** des QuarkXPress Servers in das Element `<name>` des Abschnitts `<connectioninfo>` ein.
 - Geben Sie den Port für QuarkXPress Server in das Element `port` ein.

➔ Denken Sie daran, dass Ihr Tomcat und der QuarkXPress Server nicht auf demselben Port laufen dürfen, wenn QuarkXPress Server und Tomcat auf demselben Computer laufen.

- 9 In der Datei `[TOMCAT_HOME]/qpp/conf/Qla.properties`:
 - Geben Sie den **Host-Namen**, die **Port-Nummer** und die **Seriennummer** Ihrer Instanz des QLA Servers an.
- 10 Zur Konfiguration der Datenbank bearbeiten Sie in der Datei `[TOMCAT_HOME]/qpp/conf/Database.properties` die folgenden Werte bzw. fügen Sie sie hinzu:

- Für Oracle:

```
qpp.jdbc.driverClassName = oracle.jdbc.driver.OracleDriver
qpp.jdbc.url = jdbc:oracle:thin:@<hostname>:<portnumber>:<oracle_sid>
qpp.jdbc.userName = QppOracleDB
qpp.jdbc.password = QppPassword
```

- Für SQL Server:

```
qpp.jdbc.driverClassName = com.microsoft.sqlserver.jdbc.SQLServerDriver
qpp.jdbc.url = jdbc:sqlserver://<your-host-name>\\<instanceName>;databaseName=qppdb
qpp.jdbc.userName = QppMSSQLDB
qpp.jdbc.password = QppPassword
```

- 11** Bearbeiten Sie die Datei [TOMCAT_HOME]/qpp/conf/PluginsContext.xml, indem Sie den Standardeintrag des HSQL DAO Kontexts durch den erforderlichen Datenbank-DAO-Kontext ersetzen:

- Für Oracle:

```
<import
resource="classpath:com/quark/qpp/common/dao/rdbms/oracle/OracleDaoContext.xml"/>
```

- Für SQL Server:

```
<import
resource="classpath:com/quark/qpp/common/dao/rdbms/sqlserver/SqlServerDaoContext.xml"/>
```

- 12** Legen Sie die folgenden globalen Umgebungsvariablen auf dem Computer fest:

- **MAGICK_HOME** : [Tomcat_Home]\qpp\ImageMagick\bin.
- **MAGICK_FILTER_MODULE_PATH** : %MAGICK_HOME%/modules/filters.
- **MAGICK_CODER_MODULE_PATH** : %MAGICK_HOME%/modules/coders.

- 13** Öffnen Sie die Datei [TOMCAT_HOME]/conf/catalina.properties und fügen Sie folgende Zeile hinzu:

```
Shared.Loader
:${catalina.home}/qpp/conf,${catalina.home}/qpp/lib/*.jar,
${catalina.home}/qpp/publishing, ${catalina.base}/qpp/ext
```

- 14** Öffnen Sie die Datei [TOMCAT_HOME]/conf/catalina.properties und suchen Sie nach org.apache.catalina.startup.TldConfig.jarToSkip= und aktualisieren Sie:

```
org.apache.catalina.startup.TldConfig.jarToSkip=a*.jar,b*.jar,c*.jar,d*.jar,e*.jar,f*.jar,g*.jar,
h*.jar,i*.jar,k*.jar,l*.jar,m*.jar,n*.jar,o*.jar,p*.jar,q*.jar,r*.jar,s*.jar,t*.jar,u*.jar,v*.jar,w*.jar,x*.
jar,y*.jar,z*.jar,ja*.jar,jc*.jar,jd*.jar,je*.jar,ji*.jar,jn*.jar
```

- 15** Wenn Tomcat mit dem Installer installiert wurde:

Starten Sie den Tomcat Monitor. Im Register JAVA legen Sie CATALINA_OPTS im Bereich Java Options fest:

- Wenn Platform Server mit einer Oracle Datenbank betrieben wird:
Doracle.jdbc.J2EE13Compliant=true
- Für die Integration von LDAP (oder geben Sie den absoluten Pfad an):
Djava.security.krb5.conf=./qpp/conf/krb5.conf
- Für den Java PermGen Speicher: XX:MaxPermSize=256m

16 Wenn Tomcat mit Binärcodes betrieben wird:

Öffnen Sie die Datei `[TOMCAT_HOME]/bin/catalina.bat` und stellen Sie folgende Parameter ein:

- `JRE_HOME=C:\Program Files\Java\jdk1.7.0_76`
- `JAVA_OPTS=-server -Xmx2048m -XX:MaxPermSize=128m`
- `CATALINA_OPTS=-Doracle.jdbc.J2EE13Compliant=true
-Djava.security.krb5.conf=../qpp/conf/krb5.conf`

17 Starten Sie den Tomcat Monitor. Im Register JAVA stellen Sie folgende Parameter ein:

- `Initial memory pool=1024 MB.`
- `Maximum memory pool=2048 MB.`

Serverstart und Verifizierung

Nach der Installation von Quark Publishing Platform Server mit Tomcat und der Festlegung des Ports für den Quark Publishing Platform Server Zugriff können Sie Quark Publishing Platform Server starten und Ihre Konfiguration prüfen. Quark Publishing Platform Server und Tomcat sind miteinander gekoppelt. Zum Starten und Stoppen von Quark Publishing Platform Server müssen Sie Tomcat starten und stoppen.

Um den Quark Publishing Platform Web Client Zugriff zu bestätigen, geben folgendes in das Adressfeld Ihres Browsers ein.

`http://[ComputerName]:[webServerPort]/workspace`

Um den Quark Publishing Platform Web Admin Zugriff zu bestätigen, geben folgendes in das Adressfeld Ihres Browsers ein.

`http://[ComputerName]:[webServerPort]/admin`

- ➔ QuarkCopyDesk, QuarkXPress, Quark Publishing Platform Client, Quark Publishing Platform Web Client und Quark Publishing Platform Script Manager verwenden den Tomcat Port für die Anmeldung am Quark Publishing Platform Server.

Anwendung von Quark Publishing Platform Server im Multiserver-Betrieb

Ein Quark Publishing Platform Multiserver-Cluster ist eine Gruppe von Quark Publishing Platform Server Installationen, die für die Nutzung einer gemeinsamen Datenbank, eines gemeinsamen Assetspeichers und einer gemeinsamen Message-Queue konfiguriert sind. Ein Multiserver-Cluster bietet folgende Vorteile:

- Es ermöglicht das Bedienen einer großen Anzahl von Requests durch das Hinzufügen von Hardware.
- Es erlaubt eine aktive Lastverteilung.
- Es bietet ein redundantes, verlässliches Setup.
- Das Versagen einer einzelnen Instanz in einem Cluster führt nicht zu einem kompletten Ausfall des Service. Nur ein Teil der aktiven Sessions sind durch den

Ausfall einer Instanz betroffen und nachfolgende Anfragen (Requests) können an eine aktive Instanz von Quark Publishing Platform Server geleitet werden.

- Es besteht Transparenz für Clients, die mit dem HTTP-Server kommunizieren, der die Last verteilt.

Um die Last von Anfragen zu verteilen, benötigen Sie einen HTTP-Server und einen beliebigen HTTP-Load-Balancer, der das „Sticky Session“ Feature unterstützt. Folgende HTTP-Server wurden als lastverteilende Server getestet:

- Microsoft IIS 7 mit der neuesten Version des IIS Tomcat Connectors.
- Apache 2.2 mit der neuesten Version des JK Tomcat Connectors für Apache 2.2.
- Apache 2.2 mit integrierten mod_proxy und mod_proxy_balancer DSO-Modulen von Apache 2.2.

➔ Nur durch einen HTTP-Load-Balancer geroutete Requests werden zwischen den verschiedenen Instanzen verteilt. An eine bestimmte Instanz gerichtete Requests werden ungeachtet der vorhandenen Last stets durch diese Instanz bedient.

➔ RMI Clients müssen sich direkt mit einer spezifischen Serverinstanz verbinden.

Konfigurieren von Cluster Server Instanzen oder Knoten

Der Quark Publishing Platform Server kann als Standalone-Server oder in einem externen Tomcat eingesetzt werden und somit die Datenbanken der ursprünglichen Computer zu nutzen, die sogenannten Knoten des Platform Multi-Server Clusters. Der erste Schritt beim Einrichten von Cluster Servern ist das Setup einer Quark Publishing Platform Server Datenbank für Oracle oder MS SQL Server.

Platform Server im Betrieb als Standalone-Server, der als Knoten läuft

Auf jedem Quark Publishing Platform Server Computer:

- 1 Öffnen Sie die Datei „ActiveMQ.xml“, die sich im Ordner `qpp/conf` befindet.
- 2 Fügen Sie die Netzwerk-Konnektoren auf folgende Weise in der Datei „ActiveMQ.xml“ hinzu, um die Message Queue der Server-Instanz mit einer oder mehreren Server-Instanzen im Cluster zu integrieren (wobei `qpp-node-N` der Name eines Computers ist, auf dem eine weitere Instanz von Quark Publishing Platform Server zum Einsatz kommt):

Es muss einen Konnektor für jeden Knoten des Clusters geben, außer dem Knoten selbst. Nachfolgend ist ein Cluster aus drei Knoten aufgelistet:

Knoten 1 sollte die folgenden Einträge enthalten:

```
<networkConnectors> <networkConnector dynamicOnly="true" duplex="false"
name="node1-1" networkTTL="1" uri="static:(tcp://qpp-Node2:61401)"/>
<networkConnector dynamicOnly="true" duplex="false" name="node1-2"
networkTTL="1" uri="static:(tcp://qpp-node3:61401)"/> </networkConnectors>
```

Knoten 2 sollte die folgenden Einträge enthalten:

```
<networkConnectors> <networkConnector dynamicOnly="true" duplex="false"
name="node2-1" networkTTL="1" uri="static:(tcp://qpp-Node1:61401)"/>
<networkConnector dynamicOnly="true" duplex="false" name="node2-2"
networkTTL="1" uri="static:(tcp://qpp-node3:61401)"/> </networkConnectors>
```

Knoten 3 sollte die folgenden Einträge enthalten:

```
<networkConnectors> <networkConnector dynamicOnly="true" duplex="false"
name="node3-1" networkTTL="1" uri="static:(tcp://qpp-Node1:61401)"/>
<networkConnector dynamicOnly="true" duplex="false" name="node3-2"
networkTTL="1" uri="static:(tcp://qpp-node2:61401)"/> </networkConnectors>
```

➔ Die Namen der Konnektoren (z. B. node1-1 & node1-2) sollten im gesamten Cluster eindeutig sein.

- In der Datei „ActiveMQ.xml“ kommentieren Sie die Transport-Konnektoren aus, um die Message Queues der Server Instanzen zu integrieren.

Es sollte folgendermaßen aussehen:

```
<transportConnector
uri="tcp://localhost:${jms.openWirePort}?wireFormat.maxInactivityDuration=0"
updateClusterClients="true" rebalanceClusterClients="true"
updateClusterClientsOnRemove="true"/> <transportConnector
uri="tcp://${server.machinename}:${jms.openWirePort}?wireFormat.maxInactivityDuration=0"/>
```

- Öffnen Sie die Datei „web.xml“ der Workspace- und Admin-Webapps, die sich im Ordner `WEB-INF` jeder Webapp befindet und entfernen Sie den Kommentar bei `SessionResetFilter`, um Sessions im Falle eines Knoten-Fehlers zu verarbeiten.

```
<filter> <filter-name>SessionResetFilter</filter-name>
<filter-class>com.quark.web.security.servlet.SessionResetFilter</filter-class>
<init-param> <param-name>jvmRoute</param-name>
<param-value>qpp1</param-value> </init-param> <init-param>
<param-name>sessionCookieName</param-name>
<param-value>JSESSIONID</param-value> </init-param> <init-param>
<param-name>sessionInitUrls</param-name> <param-value>
/workspace/reconnectUser.jsp /workspace/login.jsp </param-value>
</init-param> </filter> <filter-mapping>
<filter-name>SessionResetFilter</filter-name> <url-pattern>/*</url-pattern>
</filter-mapping>
```

- Weisen Sie in der Datei „ServerApp.properties“ im Ordner `[QPP Server]/conf` dem Attribut `server.id.prefix` einen eindeutigen Wert zu, der die Server-Instanz bezeichnet.
- Weisen Sie in der Datei „ServerApp.properties“ im Ordner `[QPP Server]/conf` als `server.machinename` die IP des Computers zu, um damit die Server-Instanz zu identifizieren.
- Weisen Sie in der Datei „server.xml“ im Ordner `[server-path]/conf` dem Attribut `jvmRoute` des Tags `<Engine>` einen eindeutigen Wert zu.

```
<Engine name="Catalina" defaultHost="localhost" jvmRoute="node1">
```

➔ Die Werte von `jvmRoute` in den Dateien „web.xml“ (die in Schritt 4 bearbeitete Datei) und „server.xml“ sollten identisch sein.

- Ändern Sie den Namen des Ordners „services“ (in `{TOMCAT_HOME}/webapps/webServices/WEB-INF`) in „services_old“.
- Ändern Sie den Namen des Ordners „serviceArchivesMultiServer“ (in `{TOMCAT_HOME}/webapps/webServices/WEB-INF`) in „services“.
- In der Datei „ServerStartup.bat“ im Installationsordner des Servers fügen Sie `-Dspring.profiles.active=multiserver` hinzu, um den Server im Cluster-Modus zu betreiben:

```
java -Xmx2048m -XX:MaxPermSize=256m -classpath "%JavaClassPath%"
-Dspring.profiles.active=multiserver -Doracle.jdbc.J2EE13Compliant=true
-Djava.security.krb5.conf=conf/krb5.conf -Djava.endorsed.dirs="endorsed"
```

```
-Djavax.xml.stream.XMLInputFactory=com.ctc.wstx.stax.WstxInputFactory
com.quark.gpp.Server PluginsContext.xml
```

- 11 In der Datei „wrapper.conf“ im Installationsordner des Servers entfernen Sie den Kommentar unterhalb des Abschnitts „Java Additional Parameters“:

```
wrapper.java.additional.7=-Dspring.profiles.active=multiserver
```

- ➡ Wiederholen Sie die oben angeführten Schritte für alle Quark Publishing Platform Knoten.

- 12 Löschen Sie die Browser-Cookies, bevor Sie Ihre Arbeit beginnen.

- 13 Führen Sie die Schritte zum Bereinigen der Caches über alle Server-Instanzen aus wie im folgenden Abschnitt „*Konfiguration des Server Cache Flush Mechanismus*“ beschrieben.

Konfiguration des Server Cache Flush Mechanismus

Platform Server bietet zwei Methoden Caches über alle Server-Instanzen im Falle eines Updates zu leeren:

- Anhand von Benachrichtigungen - Diese Methode löscht den Cache asynchron. Das Cache Flushing anhand von Benachrichtigungen wird realisiert durch die Übermittlung einer Nachricht des Typs `CACHE` mit der Eigenschaft `FLUSH_INTERCEPTOR`. Das ist der standardmäßig konfigurierte Mechanismus.
- Anhand von REST-Anfragen - Diese Methode löscht den Cache synchron. Die REST-basierte Methode stellt eine HTTP-Anfrage an jeden Peer-Knoten, um dessen Cache-Bereinigung anzustoßen:
`http://{peerNode}:61400/rest/service/xcache/flush/{interceptorName}`.
Diese Methode ist nicht die Standard-Konfiguration, kann aber bei Bedarf konfiguriert werden.

Um den REST-basierten Cache Flush Mechanismus zu konfigurieren, nehmen Sie folgende Änderungen vor:

- 1 Öffnen Sie die Datei „CacheManager.xml“; Sie finden sie in `{install_path}/Server/lib/qpp-server-common-{version}.jar`
- 2 Setzen Sie einen Kommentar bei `asyncRemoteMessenger` und entfernen Sie den Kommentar bei `synchronousRemoteInvoker`. Fügen Sie diese Bean anstelle von `asyncRemoteMessenger` in `baseCacheFlushingInterceptor` ein.

```
<bean id="synchronousRemoteInvoker"
  class="com.quark.gpp.common.caching.SynchronousRemoteInvoker">
  <property name="remoteHosts" value="{server.peer.url}"/>
  <property name="httpContext" value="{server.cache.remoteFlush.context}"/>
</bean>

<!-- <bean id="asyncRemoteMessenger"
class="com.quark.gpp.core.messaging.service.impl.AsyncRemoteMessenger">
  <property name="messagePublisher" ref="messagePublisher"/>
</bean> -->
```

- 3 Suchen Sie nach `baseCacheFlushingInterceptor` und fügen Sie die `synchronousRemoteInvoker` Bean in `flushinginterceptor` ein:

```
<bean id="baseCacheFlushingInterceptor"
class="com.quark.gpp.common.caching.BaseCacheFlushingAdvice" abstract="true">
```

```
<property name="remoteInvoker" ref="synchronousRemoteInvoker"/>
</bean>
```

- 4 Sichern Sie die Datei und schließen Sie sie.
- 5 Öffnen Sie die Datei „ServerApp.properties“; Sie finden sie im Ordner [QPP Server]/conf, entfernen Sie folgenden Kommentar

```
# Uncomment following properties to configure QPP cluster with synchronous
cache update
# Set comma separate URIs of peer server instances which are part of Cluster
except the node itself
server.peer.url=http://{peerHostN 1}:61400,http://{peerHostN 2}:61400

# Http context which may be used to invoke cache flushing
server.cache.remoteFlush.context=rest/service/xcache/flush
```

- 6 Für jeden Knoten öffnen Sie die Datei „rest-servlet.xml“ im Ordner [QPP Server]/webapps/rest/WEB-INF und entfernen Sie folgenden Kommentar:

```
<!-- Following code should be uncommented for synchronous REST based cache
flushing in clustered environment.
Damit aktivieren Sie den „RemoteFlushController“, der die HTTP-Anfragen
zur Cache-Bereinigung von entfernten Peers verarbeitet. -->
<context:component-scan base-package="com.quark.qpp.common.caching"/>
```

- 7 Starten Sie den Server neu.

Platform Server im Einsatz im externen Tomcat

Auf jedem Quark Publishing Platform Server Computer:

- 1 Lassen Sie eine Standalone-Instanz von Quark Publishing Platform Server im externen Tomcat laufen und verwenden Sie die Datenbank auf dem ersten Computer.
 - 2 Öffnen Sie die Datei „ActiveMQ.xml“ im Ordner qpp/conf und fügen Sie Netzwerk-Konnektoren hinzu, um die Message Queue der Instanz des Servers mit einer oder mehreren Server-Instanzen im Cluster zu integrieren (wobei qpp-node-N der Name eines Computers ist, auf dem eine weitere Instanz von Quark Publishing Platform Server zum Einsatz kommt):
- ➡ Fügen Sie diese Netzwerk-Konnektoren auf die gleiche Art und Weise hinzu, wie Sie das bereits in Schritt 2 des folgenden Abschnitts gemacht haben: „[Platform Server im Betrieb als Standalone-Server, der als Knoten läuft](#)“.
- 3 Entfernen Sie den Kommentar von Transport Connector(s) in der Datei „ActiveMQ.xml“, um die Message Queues der Server-Instanzen mit der IP-Adresse und dem Namen der Server-Instanzen im Cluster zu integrieren.

```
<transportConnector
uri="tcp://localhost:${jms.openWirePort}?wireFormat.maxInactivityDuration=0"
updateClusterClients="true" rebalanceClusterClients="true"
updateClusterClientsOnRemove="true"/> <transportConnector
uri="tcp://{server.machinename}:${jms.openWirePort}?wireFormat.maxInactivityDuration=0"/>
```

- 4 Weisen Sie in der Datei „ServerApp.properties“ im Ordner [QPP Server]/conf dem Attribut server.id.prefix einen eindeutigen Wert zu, um damit die Server-Instanz zu identifizieren.
- 5 Weisen Sie in der Datei „ServerApp.properties“ im Ordner [QPP Server]/conf dem Attribut server.machinename die IP des Computers zu, um damit die Server-Instanz zu identifizieren.

- 6 Weisen Sie in der Datei „server.xml“ im Ordner `{TOMCAT_HOME}/conf` dem Attribut `jvmRoute` des Tags `<Engine>` einen eindeutigen Wert zu.

```
<Engine name="Catalina" defaultHost="localhost" jvmRoute="node1">
```

- 7 Bearbeiten Sie die Datei `tomcat_home/conf/context.xml` folgendermaßen:

```
<Context sessionCookiePath="/">
```

- 8 Ändern Sie den Namen des Ordners „serviceArchivesMultiServer“ (in `{TOMCAT_HOME}/webapps/webServices/WEB-INF`) in „services“.

- 9 Bearbeiten Sie die Datei „Catalina.bat“ im Ordner `{TOMCAT_HOME}/bin`. Stellen Sie die folgenden Runtime-Parameter ein, um den Server im Multi-Server-Modus zu betreiben:

```
Definieren Sie CATALINA_OPTS=-Dspring.profiles.active=multiserver
```

- ➔ Wenn Apache Tomcat als „Dienst“ läuft, fügen Sie diesen Parameter im Apache Tomcat Monitor im Register **Java** im Textfeld **Java-Option** ein:

```
-Dspring.profiles.active=multiserver
```

- 10 Öffnen Sie die Datei „web.xml“ der Workspace- und Admin-Webapps, die sich im Ordner `{TOMCAT_HOME}/webapps/WEB-INF` jeder Webapp befindet und entfernen Sie den Kommentar bei `SessionResetFilter`, um Sessions im Falle eines Knoten-Fehlers wie unten beschrieben zu verarbeiten.

```
<filter> <filter-name>SessionResetFilter</filter-name>
<filter-class>com.quark.web.security.servlet.SessionResetFilter</filter-class>
  <init-param> <param-name>jvmRoute</param-name>
  <param-value>qpp1</param-value> </init-param> <init-param>
  <param-name>sessionCookieName</param-name>
  <param-value>JSESSIONID</param-value> </init-param> <init-param>
  <param-name>sessionInitUrls</param-name> <param-value>
  /workspace/reconnectUser.jsp /workspace/login.jsp </param-value>
</init-param> </filter> <filter-mapping>
<filter-name>SessionResetFilter</filter-name> <url-pattern>*/</url-pattern>
</filter-mapping>
```

- ➔ Der Wert des Parameters `<param-value>qpp1</param-value>` sollte mit dem in der Datei „server.xml“ in Schritt 6 übereinstimmen.

- 11 Wiederholen Sie die oben angeführten Schritte für alle Quark Publishing Platform Knoten.

Konfigurieren von IIS als HTTP Load Balancer

So konfigurieren Sie IIS 7 als HTTP Load Balancer:

- 1 Als erstes muss der Tomcat Connector gestartet werden. Einzelheiten dazu finden Sie unter http://tomcat.apache.org/connectors-doc/webserver_howto/iis.html. Erstellen Sie einen Ordner mit dem Namen „TomcatConnector“ und kopieren Sie die Datei „isapi_redirect.dll“ aus dem Tomcat Connector für IIS Build hinein.
- 2 Erstellen Sie eine Datei mit dem Namen „isapi_redirect.properties“ in demselben Ordner und fügen Sie folgendes in diese Datei ein:

```
extension_uri=/jakarta/isapi_redirect.dll
log_file=C:\TomcatConnector\Log\isapi.log
# Make sure the directory listed for log files in the
# "isapi_redirect.properties" file exists
log_level=info
```

EINSATZ VON QUARK PUBLISHING PLATFORM SERVER FÜR EXTERNE SERVER CONTAINER

```
worker_file=C:\TomcatConnector\worker.properties
worker_mount_file=C:\TomcatConnector\uriworker.properties
```

- Erstellen Sie in demselben Ordner eine Datei mit dem Namen „workers.properties“ und fügen Sie folgendes in die Datei ein, wobei `qpp-node1` und `qpp-node2` die Werte des Attributs `jvmRoute` sind - wie in der Datei „server.xml“ für jeden Quark Publishing Platform Server angegeben:

```
worker.list=TomcatBalancer
worker.TomcatBalancer.type=lb
worker.TomcatBalancer.balance_workers=qpp-node1,qpp-node2
worker.TomcatBalancer.sticky_session=True
worker.TomcatBalancer.sticky_session_force=True
worker.TomcatBalancer.method=Request
worker.TomcatBalancer.lock=Pessimistic

worker.qpp-node1.type=ajp13
worker.qpp-node1.host=qpp-node1

# This should be the same as the port in the AJP connector defined
# in the "server.xml" file for the Tomcat server in which Quark
# Publishing Platform is deployed
worker.qpp-node1.port=61398

worker.qpp-node1.lbfactor=3

worker.qpp-node2.type=ajp13
worker.qpp-node2.host=qpp-node2

# This should be the same as the port in the AJP connector defined
# in the "server.xml" file for the Tomcat server in which Quark
# Publishing Platform is deployed
worker.qpp-node2.port=61398

worker.qpp-node2.lbfactor=3
```

- Erstellen Sie eine Datei mit dem Namen „uriworker.properties“ in demselben Ordner und fügen Sie folgendes in diese Datei ein:

```
/admin/*=TomcatBalancer
/admin=TomcatBalancer
/workspace/*=TomcatBalancer
/workspace=TomcatBalancer
/webservices/*=TomcatBalancer
/webservices=TomcatBalancer
/rest/*=TomcatBalancer
/rest=TomcatBalancer
/messaging/*=TomcatBalancer
/messaging=TomcatBalancer
/qxpsmadmin/*=TomcatBalancer
/qxpsmadmin=TomcatBalancer
/favicon.ico=TomcatBalancer
/*=TomcatBalancer
/qxpsm/*=TomcatBalancer
/qxpsm=TomcatBalancer
/pluginwiris_engine/*=TomcatBalancer
/pluginwiris_engine=TomcatBalancer
```

➔ Diese Datei enthält die Mappings der URLs, die von IIS verarbeitet und an die Quark Publishing Platform Servers weitergeleitet werden sollen.

- Fügen Sie unter Verwendung der IIS Management-Console Ihrer IIS Website ein neues virtuelles Verzeichnis mit dem Namen „jakarta“ hinzu mit dem physikalischen Pfad des Verzeichnisses, in dem Sie die Datei „isapi_redirect.dll“ platziert haben.
- Geben Sie dem virtuellen Verzeichnis das Recht „execute“. Markieren Sie den virtuellen Ordner, doppelklicken Sie **Handler Mappings** und klicken Sie auf **Edit Feature Permissions** im Bereich **Actions**. Markieren Sie **Execute** in der Dialogbox **Edit Feature Permissions** und klicken Sie auf **OK**.

- 7 Fügen Sie der IIS Website einen ISAPI filter hinzu. Markieren Sie die Website, doppelklicken Sie auf **ISAPI Filters** und klicken Sie auf **Add** im Bereich **Actions**. Geben Sie in die Dialogbox **Add ISAPI Filter** einen Namen und den Pfad der Datei „isapi_redirect.dll“ ein und klicken Sie auf **OK**.
- 8 Konfigurieren Sie die ISAPI und CGI Restrictions Funktionen. Navigieren Sie zum Homescreen des Servers, doppelklicken Sie auf **ISAPI and CGI Restrictions** und klicken Sie auf **Add** im Bereich **Actions**. Geben Sie in die Dialogbox **Add ISAPI or CGI Restriction** einen Namen und den Pfad der Datei „isapi_redirect.dll“, markieren Sie **Allow extension path to execute** und klicken Sie auf **OK**.
- 9 Aktivieren Sie die Windows Authentifizierung. Navigieren Sie zu Ihrer Website und doppelklicken Sie auf **Authentication**. Rechtsklicken Sie auf **Windows Authentication** und wählen Sie **Enable**, deaktivieren Sie die übrigen Authentifizierungsoptionen und starten Sie IIS neu.
- 10 Starten Sie zum Prüfen der Installationen Quark Publishing Platform Server auf allen Knoten und greifen Sie mithilfe des Computernamens oder mit der IP-Adresse und dem Port auf den IIS Server zu. Der IIS Server sollte unter Verwendung des Tomcat Connectors eine Lastverteilung ausführen. Einzelheiten entnehmen Sie der in der Datei „isapi_redirect.properties“ angegebenen Logdatei.

Konfigurieren von Apache Web Server als HTTP Loadbalancer

So konfigurieren Sie Apache Web Server 2.2 mit der neuesten Version des JK Tomcat Connector für Apache 2.2 als HTTP Loadbalancer:

- 1 Als erstes muss der Tomcat Connector gestartet werden. (Einzelheiten dazu finden Sie unter <http://tomcat.apache.org/connectors-doc/miscellaneous/faq.html>.) Kopieren Sie die Datei „mod_jk.so“ vom Download-Ort des Tomcat Connectors nach `{Apache_2.2_installation}/modules`.
- 2 Öffnen Sie die Datei `{Apache_2.2_installation}/conf/httpd.conf` in einem Texteditor und fügen Sie folgendes ein:

```
LoadModule jk_module modules/mod_jk.so

<IfModule jk_module>
    JkWorkersFile conf/workers.properties
    JkLogFile logs/mod_jk.log
    JkLogStampFormat "[%H:%M:%S] "
    JkRequestLogFormat "%T"
    JkLogLevel error
    JkOptions +ForwardKeySize +ForwardURICompat -ForwardDirectories
    <Directory />
        AllowOverride All
        <Limit GET HEAD POST PUT DELETE OPTIONS>
            Order Allow,Deny
            Allow from all
        </Limit>
    </Directory>
    JkMount /workspace TomcatBalancer
    JkMount /workspace/* TomcatBalancer
    JkMount /webservices TomcatBalancer
    JkMount /webservices/* TomcatBalancer
    JkMount /admin TomcatBalancer
    JkMount /admin/* TomcatBalancer
    JkMount /rest TomcatBalancer
    JkMount /rest/* TomcatBalancer
    JkMount /messaging TomcatBalancer
    JkMount /messaging/* TomcatBalancer
    JkMount /qxpsm TomcatBalancer
    JkMount /qxpsm/* TomcatBalancer
    JkMount /qxpsmadmin TomcatBalancer
    JkMount /qxpsmadmin/* TomcatBalancer
```

```
JkMount /pluginwiris_engine/*=TomcatBalancer
JkMount /pluginwiris_engine=TomcatBalancer
</IfModule>
```

3 Erstellen Sie eine Datei mit dem Namen

`{Apache_2.2_installation}/conf/workers.properties` und fügen Sie ihr folgenden Inhalt hinzu, wobei `qpp-node1` und `qpp-node2` die Werte des Attributs `jvmRoute` sind - wie in der Datei „server.xml“ für jeden Quark Publishing Platform Server angegeben:

```
worker.list=TomcatBalancer

worker.TomcatBalancer.type=lb
worker.TomcatBalancer.balance_workers=qpp-node1, qpp-node2
worker.TomcatBalancer.sticky_session=True

# This should be commented out, if it is not, you cannot connect to the
# Apache Web server which
# acts as the load balancer.
worker.TomcatBalancer.sticky_session_force=True

worker.TomcatBalancer.method=Request
worker.TomcatBalancer.lock=Pessimistic

worker.qpp-node1.type=ajp13

# This should be the IP address of the first QPP node
worker.qpp-node1.host=Server 1

# 8009 is for external tomcat server
# 61398 is for embedded tomcat with QPP
worker.qpp-node1.port=61398

worker.qpp-node1.lbfactor=3

worker.qpp-node2.type=ajp13

# This should be the IP address of the first QPP node
worker.qpp-node2.host=Server 2

# 8009 is for external tomcat server
# 61398 is for embedded tomcat with QPP
worker.qpp-node2.port=61398

worker.qpp-node2.lbfactor=3
```

4 Starten Sie Apache 2.2 httpd neu.

5 Starten Sie zum Prüfen der Installationen Quark Publishing Platform Server auf allen Nodes und greifen Sie mithilfe des Computernamens oder mit der IP-Adresse und dem Port auf den Apache Server zu. Der Apache Server sollte unter Verwendung des Tomcat Connectors eine Lastverteilung ausführen. Einzelheiten finden Sie in den in Logfiles unter `{Apache_2.2_Installation}/logs`.

Elasticsearch

Ein Multi-Server-Einsatz kann Elasticsearch so beeinflussen, dass eine erneute Generierung des Suchindexes für jeden Anwendungsknoten des Platform Multi-Server-Clusters vermieden werden kann.

Download und Installation von Elasticsearch

1 Laden Sie Elasticsearch v2.4.0 herunter und extrahieren Sie alle Dateien:

www.elastic.co/downloads/elasticsearch

2 So installieren Sie Elasticsearch:

- Unter Unix: Starten Sie `bin/elasticsearch`.
- Unter Windows: Starten Sie `bin\elasticsearch.bat`.
- Starten Sie `curl -X GET http://localhost:9200/`

Configuring Platform server to leverage Elasticsearch

A Multi-server deployment can leverage Elasticsearch to avoid regenerating the search index for each application node of the Platform multiserver cluster.

- ➔ If you are deploying Platform server as a single instance with Elasticsearch, there is no need to make any configuration changes as that deployment is covered by the default configuration.

Perform the following steps on each Quark Publishing Platform Server computer:

- 1 Open the "ESIndexingConfig.properties" file located in the `[QPP Server]/conf` folder.

- 2 Specify where Elasticsearch is running:

```
#Comma delimited list of host:tcp_port entries pointing to Elasticsearch
cluster nodes
es.cluster.nodes=platform2k12:9300
```

- ➔ The default port for Elasticsearch is 9300.

- 3 Specify the name of the index to be used for attributes:

```
es.attribute.index.name=qps-attribute-index
```

- 4 Specify the name of the index to be used for text:

```
es.text.index.name=qps-text-index
```

- 5 In the "ESAttributeIndexSettings.json" file located in the `[QPP Server]/conf` folder, specify the number of shards and replicas required for the index:

```
{ "index":
  { "number of shards":5,
    "number of replicas":1
  }
  ...
  ...
}
```

- ➔ Shards are per index and replicas are per node.

- 6 In the "PluginsContext.xml" file located in the `[QPP Server]/conf` folder, comment out the following line to specify that you will no longer be using the Lucene indexer:

```
<!--import
resource="classpath:com/quark/qpp/text/indexing/adapter/impl/LuceneTextIndexerContext.xml"/>-->
```

Konfiguration von Elasticsearch für eine Multi-Server-Umgebung

Beim Betrieb in einer Multi-Server-Umgebung, in der auf jedem Server eine eigene Instanz von Elasticsearch läuft, wird der Cluster nach einem Master-Child-Konzept betrieben. Eine Instanz von Elasticsearch wird als Master betrachtet (je nachdem

welcher Node zuerst gestartet wurde) und alle anderen Instanzen als Child-Instanzen. Wenn die Master-Instanz beendet wird, wird eine der Child-Instanzen zum Master.

Um jede Elasticsearch-Instanz als Teil eines Clusters zu konfigurieren, führen Sie die folgenden Schritte für jeden Elasticsearch-Betrieb durch:

- 1 Öffnen Sie die Datei „elasticsearch.yml“ im Ordner

```
{Elasticsearch_install_path}/config.
```

- 2 Geben Sie den Namen des Clusters an:

```
cluster.name: myESclustername
```

- 3 Geben Sie den Namen dieses Nodes an:

```
node.name: nameofnode1
```

- 4 Geben Sie den Netzwerk-Hostnamen an.

```
network.host: mynetworkhost
```

- 5 Identifizieren Sie weitere Knoten (Instanzen von Elasticsearch) in Ihrem Cluster:

```
discovery.zen.ping.unicast.hosts: ["nameofnode2", "nameofnode3"]
```

- 6 Auf jedem Quark Publishing Platform Server Computer befindet sich die Datei „ESIndexingConfig.properties“ im Ordner [QPP Server]/conf, geben Sie den Namen des Clusters an:

```
#name of the Elasticsearch cluster to which this instance of Elasticsearch  
client will connect to.  
es.cluster.name=myESclustername
```

Starten von Elasticsearch

Elasticsearch erfordert mindestens Java 7 Update 55 oder neuer. Die Umgebungsvariable `JAVA_HOME` sollte entsprechend eingestellt sein.

Elasticsearch kann mit einer der folgenden Methoden gestartet werden:

- Mithilfe einer Batch-Datei:

Um Elasticsearch im Vordergrund als neuen Prozess zu starten, sollten Sie einfach die Datei „elasticsearch.bat“ im Ordner `/bin` aufrufen.

- Als Service:

Um Elasticsearch im Vordergrund als neuen Prozess zu starten, sollten Sie einfach die Datei „elasticsearch.bat“ im Ordner `/bin` aufrufen.

Um Elasticsearch als Dienst zu starten, rufen Sie „service.bat“ im Ordner `/bin` mit folgenden Befehlen auf:

- So installieren Sie Elasticsearch als Dienst:

```
C:\elasticsearch-2.4.0\bin>service.bat install.
```

- Nach der Installation starten Sie Elasticsearch als Dienst:

```
C:\elasticsearch-2.4.0\bin>service.bat start
```

- So stoppen Sie den Elasticsearch-Dienst:

```
C:\elasticsearch-2.4.0\bin>service.bat stop
```

- So entfernen Sie den installierten Elasticsearch Dienst (und stoppen den Dienst, falls er läuft): `C:\elasticsearch-2.4.0\bin>service.bat remove`
- So starten Sie eine grafische Benutzeroberfläche für die Verwaltung der installierten Dienste: `C:\elasticsearch-2.4.0\bin>service.bat manager`

➔ Weitere Informationen zur Konfiguration von Elasticsearch im Einsatz mit Platform finden Sie im www.elastic.co/guide/en/elasticsearch/reference/current/setup-service-win.html#setup-service-win

Quark Publishing Platform SysAdmin Handbuch.

Eine Schnittstelle für Elasticsearch

Elasticsearch stellt keine Benutzerschnittstelle zur Verfügung, aus diesem Grund bietet Quark ein Drittanbieter-Plugin dafür an.

Um das Plugin zu nutzen, führen Sie folgende Schritte bei jedem Betrieb von Elasticsearch durch:

- 1** Kopieren Sie das Head-Plugin und setzen Sie es im Ordner `{Elasticsearch_install_path}/plugins` ein.
- 2** So können Sie auf die Benutzerschnittstelle zugreifen:
`{network host}:9300/_plugin/head/`

Aktivieren des Secure Sockets Layer (SSL) für Quark Publishing Platform Server

Sie können Quark Publishing Platform mit unterschiedlichen Sicherheitsoptionen konfigurieren. Über die Sicherheitsspezifikationen Ihres eigenen Netzwerks hinaus können Sie das SSL-Protokoll für Ihre Quark Publishing Platform Client-Anwendungen vorgeben.

SSL-Unterstützung

Sie können Ihre Application Server Container und alle Quark Publishing Platform Clients so konfigurieren, dass er im Sicherheitsmodus mit aktivierter Secure Sockets Layer (SSL) Technologie läuft. In diesem Abschnitt wird der Konfigurationsprozess erläutert.

- ➔ Es ist auch möglich, Quark Publishing Platform ohne Einbettung von Tomcat in die JVM zu betreiben. Unter "[Verwendung mit externem Tomcat](#)" finden Sie Informationen zur Einrichtung von Quark Publishing Platform ohne Einbettung von Tomcat.

Für die Handhabung von Webanwendungen der Quark Publishing Platform Umgebung bettet Quark Publishing Platform Server eine Instanz von Apache Tomcat 7.0.61 in seine JVM ein. Die vier Webanwendungen in Quark Publishing Platform sind Quark Publishing Platform Web Client, Quark Publishing Platform Console, Quark Publishing Platform Renderer Manager und Quark Publishing Platform Web Services.

Wenn Sie SSL aktivieren, wirkt sich dies auf alle Quark Publishing Platform Client-Anwendungen aus, die in Quark Publishing Platform Server laufen.

Aktivieren von SSL für Quark Publishing Platform Server

Die unten folgende Anleitung berücksichtigt zwei Szenarien. Die Datei „server.xml“, die Sie bearbeiten, enthält XML-Tags für beide Szenarien, die Sie durch Kommentieren und Aufheben der Kommentierung bestimmter Tags aktivieren oder deaktivieren müssen.

So aktivieren Sie SSL für sicheres HTTP für alle Quark Publishing Platform Webanwendungen:

AKTIVIEREN DES SECURE SOCKETS LAYER (SSL) FÜR QUARK PUBLISHING PLATFORM SERVER

1 Öffnen Sie die Datei „server.xml“, die sich im Ordner {Installationspfad}/conf befindet.

2 Kommentieren Sie folgendes Tag:

```
<Connector port="61400" maxHttpHeaderSize="8192"maxThreads="150"
minSpareThreads="25" maxSpareThreads="75"enableLookups="false"
redirectPort="61399" acceptCount="100"connectionTimeout="20000"
disableUploadTimeout="true" />
```

3 Entfernen Sie den Kommentar von folgendem Tag:

```
<Connector port="61399" maxHttpHeaderSize="8192"MaxThreads="150"
minSpareThreads="25" maxSpareThreads="75"enableLookups="false"
disableUploadTimeout="true"acceptCount="100" scheme="https"
secure="true"clientAuth="false" sslProtocol="TLS" />
```

4 Ersetzen Sie 61399 durch 61400 (oder einen anderen Port, auf dem Tomcat Secure Connections überwachen soll).

5 Speichern Sie „server.xml“ und schließen Sie die Datei.

6 Auf dem Computer mit Quark Publishing Platform Server öffnen Sie die Eingabeaufforderung und führen folgende Befehle aus:

```
%JAVA_HOME%\bin\keytool -genkey -alias tomcat -keyalg RSA
```

7 Legen Sie die Details in den Eingabeaufforderungen fest. Das Passwort sollte `changeit` lauten.

8 Starten Sie den Quark Publishing Platform Server erneut.

9 Rufen Sie die Admin-Webseite auf: <https://servername:61399/admin>

➔ Diese Konfiguration erstellt einen Schlüsselspeicher mit einem privaten und einem öffentlichen Schlüsselpaar. Dabei handelt es sich um ein selbstsigniertes Zertifikat.

10 Weitere Informationen über die SSL-Konfiguration und den Einsatz einer Zertifizierungsstelle finden Sie hier in der Apache Tomcat SSL Information:

<http://tomcat.apache.org/tomcat-5.5-doc/ssl-howto.html>.

➔ Diese Änderung bedeutet, dass Quark Publishing Platform Client-Anwendungen HTTPS verwenden können. So wäre beispielsweise die URL für einen Quark Publishing Platform Web Client Benutzer folgende: [https://\[Servername\]:61399/workspace](https://[Servername]:61399/workspace).

Konfiguration von Platform Clients zur Anmeldung am SSL-aktiviertem Server

Benutzer von QuarkXPress und QuarkCopyDesk unter Mac OS X müssen ein SSL-Zertifikat vom Quark Publishing Platform Server holen, bevor sie sich anmelden können. Dazu sollte jeder Benutzer Terminal starten und folgenden Befehl ausführen, der die IP-Adresse des Quark Publishing Platform Servercomputers für [server name] ersetzt:

```
echo | openssl s_client -connect[server name]:443 > [server name].pem
```

für El-Capitan und Sierra MAC OS lautet der Befehl:

```
echo | openssl s_client -connect[server name]:[Port] -servername[server name] > [server name].pem
```

AKTIVIEREN DES SECURE SOCKETS LAYER (SSL) FÜR QUARK PUBLISHING PLATFORM SERVER

Dieser Befehl ruft eine Kopie des Serverzertifikats „[servername].pem“ ab. Legen Sie diese Datei im Ordner `~/Library/Application Support/Quark/QPP/Certificates` ab (oder, wenn Sie die Datei plist unter `~/Library/Application Support/Quark/QPP/[QPP Framework Version]` angepasst haben, dann in diesem Ordner).

- ➔ Wenn die Ordner „Quark Publishing Platform“ und „Certificates“ an den oben angegebenen Orten noch nicht existieren, erstellen Sie sie dort manuell.
- ➔ Quark Publishing Platform Clients unter Windows benötigen kein Server-basiertes SSL-Zertifikat.

Prüfen und Verwenden von SSL

So wird SSL geprüft und verwendet:

- 1 Starten Sie den Quark Publishing Platform Server.
- 2 Testen Sie den Quark Publishing Platform Web Client Zugriff durch folgende Eingabe:
`https://[machine IP/name]:61399/workspace.`

Quark Publishing Platform Server — Manuelle Konfiguration

Sie können die Standardkonfiguration nach der Installation von Quark Publishing Platform Server ändern. Über das Einstellen von Parametern mit JConsole während des Betriebes von Quark Publishing Platform Server hinaus können Sie in verschiedenen .xml und .properties Dateien Einstellungen vornehmen. Des weiteren ist die Einstellung der Speicherzuweisung für Ihre JVM Konfiguration möglich.

Bearbeiten von „ServerApp.properties“

So bearbeiten Sie „ServerApp.properties“:

- 1 Öffnen Sie die Datei „ServerApp.properties“, die sich im Ordner `[QPP Server]/conf` befindet.
- 2 Stellen Sie den Wert `rmi.port` auf die Nummer des Ports ein, den die RMI Registry überwacht. Java-basierte RMI Clients wie ScriptManager verbinden sich über diesen Port.
- 3 Stellen Sie den Wert `rmi.servicePort` auf die von dem RMI Server verwendete Portnummer ein, auf dem Quark Publishing Platform Service Objects registriert sind.
- 4 Stellen Sie den Wert `namingservice.port` auf die Portnummer ein, die für das Überwachen auf Object Resolution Requests durch den CORBA Naming Service verwendet wird.
- 5 Stellen Sie den Wert von `serverORB.port` auf die Portnummer ein, die von dem ORB verwendet wird, auf dem Quark Publishing Platform Service Objects aktiviert sind.
- 6 Stellen Sie den Wert `jms.openWirePort` auf die Portnummer ein, die für die JMS Kommunikation über das OpenWire Protokoll geöffnet ist. Java Clients wie QPS Script Manager verbinden sich mit diesem Port, um auf Server Benachrichtigungen zu warten.
- 7 Stellen Sie den Wert `webServer.port` auf die Portnummer ein, auf der Tomcat HTTP Verbindungen überwacht. Quark Publishing Platform Web Client und SOAP Clients verbinden sich über diesen Port. Der Wert sollte auf denselben Wert eingestellt werden, der für den HTTP Connector in der Tomcat-Datei „server.xml“ festgelegt wurde.
- 8 Stellen Sie den Wert `socketStreaming.port` auf die Portnummer ein, die für Dateiübertragungen genutzt werden soll (Upload/Download).

- 9 Um Quark Publishing Platform Server an eine bestimmte IP-Adresse zu binden, stellen Sie den Wert `server.machinename` auf die entsprechende IP-Adresse ein und setzen Sie `server.bindtoip=true`. Und falls Sie über mehrere Netzwerkkarten verfügen und Quark Publishing Platform an alle IP-Adressen binden möchten, stellen Sie `server.machinename=localhost`, `server.bindtoip=false` und `server.additionalnames=[non-default-ip1],[non-default-ip2]` ein.
- 10 Stellen Sie den Wert `server.additionalnames` so ein, dass er die globale IP-Adresse erhält, auf der die Firewall geöffnet ist.
- 11 Geben Sie den Wert `webServer.port` ein, den Sie für Tomcat konfiguriert haben (z. B. 8080).
- 12 Um es Benutzern zu ermöglichen, sich auch dann an Quark Publishing Platform Server anzumelden, wenn der Directory Server nicht läuft, setzen Sie `authentication.external.cacheTicket = true`.
- 13 Um festzulegen, dass beim Vergleich von Passwörtern keine Groß-/Kleinschreibung berücksichtigt werden soll, stellen Sie den Wert `server.password.case.sensitive` auf false. Diese Option ist nicht gültig, wenn LDAP für die externe Authentifizierung benutzt wird.
- 14 Um ein Session-TimeOut zu konfigurieren, stellen Sie den Wert `session.maxIdle` für die Dauer des Session Timeouts in Sekunden ein und den Wert `session.eviction.thread.delay`, um die Häufigkeit des Session Eviction Threads (zur Bereinigung der Sessions) in Sekunden festzulegen.
- 15 Der Repository Status Updater ist der Hintergrund-Thread, der mit einer festgelegten Frequenz läuft, um zu testen, ob das dazugehörige Repository aktiv ist oder nicht und um dessen Status in der Server-Datenbank zu aktualisieren. Um das Sleep Intervall des Repository Status Updaters zu konfigurieren, setzen Sie den Wert `respository.status.updator.sleepInterval`, mit dem Sie das Intervall in Sekunden angeben, nach dem der Repository Status Update Thread gestartet werden soll.
- 16 Wenn `Realm Verif. für Admin. Requests` in QuarkXPress Server spezifiziert wurde, muss der Benutzername und das Passwort für QuarkXPress Server in dieser Datei festgelegt werden. Geben Sie folgende Eigenschaften ein, um den QuarkXPress Server zu konfigurieren:
 - Setzen Sie den Wert `qxps.username`, um den QuarkXPress Server Benutzernamen anzugeben.
 - Setzen Sie den Wert `qxps.password`, um das QuarkXPress Server Passwort anzugeben.
 - Setzen Sie den Wert `qxps.locale`, um den Speicherort von QuarkXPress ServerXPS anzugeben.

Bearbeiten von „PublishingPool.properties“

So bearbeiten Sie „PublishingPool.properties“:

- 1 Öffnen Sie die Datei „PublishingPool.properties“ im Ordner `conf` des Server-Installationsordners.
- 2 Setzen Sie den Wert `publishingThread.pool.maxActive`, um die maximale Anzahl von gleichzeitig ablaufenden Hintergrund-Publishing-Threads anzugeben.
- 3 Setzen Sie den Wert `publishingThread.pool.maxIdle`, um die maximale Anzahl von Idle-Threads im Pool festzulegen.
- 4 Setzen Sie den Wert `publishingThread.pool.minIdle`, um die minimale Anzahl von Idle-Threads im Pool festzulegen.
- 5 Setzen Sie den Wert `publishingThread.pool.maxWait`, um die Zeit in Millisekunden festzulegen, die die Publishing-Anfrage warten sollte, während ein Thread vom Pool geborgt wird.
- 6 Setzen Sie den Wert `publishingThread.pool.minEvictableIdleTimeMillis`, um die Zeit in Millisekunden festzulegen, die ein Thread im Pool bleiben muss, bevor er exmittiert („herausgeworfen“) werden kann.
- 7 Setzen Sie den Wert `publishingThread.pool.timeBetweenEvictionRunsMillis`, um die Zeit in Millisekunden festzulegen, nach der der Vertreiber-Thread gestartet werden sollte, um Idle Threads zu entfernen.

Konfigurieren von Quark Publishing Platform Renderer für die Verwendung mit Quark Publishing Platform

So konfigurieren Sie Quark Publishing Platform Renderer für die Verwendung mit Quark Publishing Platform:

- 1 Öffnen Sie den QXPSM Admin Client, indem Sie in einem Webbrowser folgende URL aufsuchen: `http://[QPP server name]:[port]/qxpsmadmin`.
- 2 Klicken Sie im Bereich **Server verwalten** auf **Server hinzufügen** und fügen Sie die Quark Publishing Platform Instanz von Quark Publishing Platform Renderer hinzu. (Weitere Informationen finden Sie in Ihrer *Anleitung zu Quark Publishing Platform Renderer*.)

Bearbeiten von "Qla.properties"

So bearbeiten Sie "Qla.properties":

- 1 Öffnen Sie die Datei "Qla.properties" im Ordner `[QPP Server]/conf`.
- 2 Geben Sie die IP-Adresse oder den Host-Namen des QLA Servers in das Feld `QlaServer.machinename=` ein.
- 3 Geben Sie die Portnummer des QLA Servers in das Feld `QlaServer.port=` ein.
- 4 Falls Sie einen QLA Backupserver besitzen, geben Sie die IP-Adresse (oder den Host-Namen) und die Portnummer in die Felder `Backup.QlaServer.machinename=` und `Backup.QlaServer.port=` ein.

- 5 Geben Sie die Quark Publishing Platform Seriennummer in das Feld `Qla.SerialNumber=` ein.

➔ Die QLA Server Console und QLA Client Anwendungen zeigen die Quark Publishing Platform Seriennummer an.

- 6 Speichern Sie "Qla.properties" und schließen Sie die Datei.

Erweiterte Konfigurationsdateien

Konfigurationsdateien sind für den komfortablen Betrieb in *Basis* und *Erweitert* aufgeteilt.

Anwenderspezifische **Beans**, **Prozesse** und **Publishing Channels** sollten in *.ext*-Dateien im Ordner *ext* der Server-Installation definiert werden, um anwenderdefinierte Erweiterungen zu separieren und diese nach einem Software-Upgrade beizubehalten.

- ChannelConfig-ext.xml
- content-mimetype-mappings-ext.xml
- custom-xml-types-ext.xml
- IndexingChannels-ext.xml
- PluginsContext-ext.xml
- ProcessConfig-ext.xml
- PublishingConfig-ext.xml

Konfigurieren des Anzeigennamens des Benutzers im Bereich Benutzeraktivität im Admin Web Client

Sie können Quark Publishing Platform so konfigurieren, dass der Benutzername im Bereich **Benutzeraktivität** auf drei verschiedene Arten angezeigt wird:

- `[Benutzername]`
- `[Benutzername] ([Vorname] [Nachname])`
- `[Benutzername] ([Nachname], [Vorname])`

Um diese Einstellung zu ändern, öffnen Sie die Datei „WebAdminConfig.properties“ im Ordner `[QPP Server]/webapps/admin/WEB-INF/classes`. Stellen Sie den Wert der Eigenschaft `userNameFormatting` folgendermaßen ein:

- 0 für `[Benutzername]`.
- 1 für `[Benutzername] ([Vorname] [Nachname])`.
- 2 für `[Benutzername] ([Nachname], [Vorname])`.

Konfiguration von Platform Server für die Verwendung zum Publishing mit Quark XML Author – Sharepoint Adapter

Konfiguration von Quark Publishing Platform Server für die Verwendung zum Publishing mit Quark XML Author – Sharepoint Adapter:

- 1 Öffnen Sie die Datei „sharepoint.properties“ im Ordner [QPP Server]/publishing.
- 2 Geben Sie für den Wert `sharepoint.username` den Login-Namen eines Benutzers mit Zugang zu den Sharepoint Sites an.
- 3 Setzen Sie den Wert für `sharepoint.userpassword` auf das Login-Passwort des oben angegebenen Benutzers.
- 4 Setzen Sie den Wert für `sharepoint.userdomain` auf die Domain des oben angegebenen Benutzers.
- 5 Setzen Sie den Wert für `sharepoint.sitecollection` auf die URL einer Sharepoint Site Collection, die Dokumente enthält, die während des Publishings benötigt werden.
- 6 Starten Sie den Server neu.

Konfiguration von Platform Server für die Verwendung zum Publishing mit Quark XML Author – FileNet Adapter

So konfigurieren Sie Quark Publishing Platform Server für eine Verwendung zum Publishing mit Quark XML Author - FileNet Adapter:

- 1 Öffnen Sie die Datei „contentengine.properties“ im Ordner [QPP Server]/publishing.
- 2 Setzen Sie den Wert für `filenet.stanza`, damit er die Stanza (Konfiguration) für die FileNet Content Engine Verbindung ist. Sie müssen den Parameter nur bearbeiten, wenn der FileNet-Server so konfiguriert ist, dass er andere Stanza als FileNet verwendet.
- 3 Geben Sie als Wert für `filenet.username` den FileNet Benutzernamen ein.
- 4 Geben Sie als Wert für `filenet.userpassword` das FileNet-Passwort des oben angegebenen Benutzers an.
- 5 Stellen Sie als Wert für `filenet.connectionuri` die Connection URI des FileNet Content Engine Webservice ein.
- 6 Starten Sie den Server neu.
- 7

JVM Memory Allocation unter Windows

Unter Windows stellen Sie die JVM Speicherverwaltung an unterschiedlichen Stellen ein, abhängig davon, wie Sie Quark Publishing Platform Server starten. Bei 64-bit Betriebssystemen können Sie dem Quark Publishing Platform Java Prozess mehr Speicher zuteilen. Sie sollten jedoch in keinem Falle mehr als 50 Prozent des verfügbaren Speichers zuweisen.

Verwenden der Quark Publishing Platform Server Console oder des Quark Publishing Platform Server Windows Service

- 1 Stoppen Sie Quark Publishing Platform Server.
- 2 Wenn Sie Quark Publishing Platform Server mit Quark Publishing Platform Server Console oder dem Quark Publishing Platform Server Windows Service starten, öffnen Sie die Datei „wrapper.conf“.
- 3 Suchen Sie nach `wrapper.java.maxmemory`.
- 4 Stellen Sie den Wert ein. Bei einem 64-Bit Betriebssystem darf der Wert größer sein.
- 5 Sichern Sie die Änderungen und starten Sie Quark Publishing Platform Server neu.

Verwenden von „Serverstartup.bat“

- 1 Stoppen Sie Quark Publishing Platform Server.
- 2 Falls Sie Quark Publishing Platform Server mit der Datei „ServerStartup.bat“ im Installationsordner von Quark Publishing Platform Server starten, öffnen Sie „ServerStartup.bat“.
- 3 Suchen Sie nach `java -server -Xmx2048m -XX:MaxPermSize=256m -classpath.2048m`. entspricht 1.536 MB RAM, das Quark Publishing Platform Server zugewiesen wird.
- 4 Bei einem 64-Bit Betriebssystem darf der Wert größer sein.
- 5 Sichern Sie die Änderungen und starten Sie Quark Publishing Platform Server neu.

Konfigurieren der Windows Authentifizierung

Windows User können sich bei Quark Publishing Platform transparent mit ihren Windows Anmeldedaten anmelden, ohne einen Anmeldedialog zu Gesicht zu bekommen. Quark Publishing Platform unterstützt alle Authentifizierungsschemata von Windows, einschließlich NTLM-v1/NTLM-v2 und Negotiate/Kerberos. Die Plattform unterstützt darüber hinaus den Mixed-Mode. Daher können die Windows Authentifizierung und die Authentifizierung der Plattform koexistieren.

Sie konfigurieren die Windows Authentifizierung mit einem HTTP Servlet-Filter (als Plugin). Sie können die Authentifizierung ganz einfach zum Deployment-Zeitpunkt aktivieren und deaktivieren, indem Sie der Datei "web.xml" jeder einzelnen Webapplication einen Sicherheitsfilter hinzufügen.

```
<filter>
  <filter-name>SecurityFilter</filter-name>
  <filter-class>com.quark.web.security.servlet.ApplicationSecurityFilter</filter-class>
  <init-param>
    <param-name>provider</param-name>
    <param-value>com.quark.web.security.waffle.WaffleAuthenticationProvider</param-value>
  </init-param>
  <init-param>
    <param-name>provider/protocols</param-name>
    <param-value>Negotiate NTLM</param-value>
  </init-param>
</filter>
<filter-mapping>
  <filter-name>SecurityFilter</filter-name>
  <url-pattern>*/</url-pattern>
</filter-mapping>
```

Jede Webapplication von Quark Publishing Plattform lässt sich so konfigurieren, dass unterschiedliche Windows Authentifizierungs-Schemata verwendet werden (siehe unten).

Webapplication	Nur NTLM	Nur Negotiate/Kerberos	NTLM und Kerberos
Arbeitsbereich	NTLM	Negotiate	NTLM Negotiate
Admin	NTLM	Negotiate	NTLM Negotiate
Webservices	NTLM	Kerberos	NTLM Negotiate
QXPSM	k. A.	k. A.	k. A.
Messaging	k. A.	k. A.	k. A.
REST	k. A.	k. A.	k. A.

```

  <param-name>provider</param-name>
  <param-value>com.quark.web.security.waffle.WaffleAuthenticationProvider</param-value>
</init-param>
<init-param>
  <param-name>provider/protocols</param-name>
  <param-value>Negotiate NTLM</param-value>
</init-param>
</filter>
<filter-mapping>
  <filter-name>SecurityFilter</filter-name>
```

Schritte zum Aktivieren von SSO in Platform Clients

Für die Admin Web App:

- 1 Gehen Sie zu C:\Program Files\Quark\Quark Publishing Platform\Server\webapps\admin\WEB-INF
- 2 Öffnen Sie Web.xml und suchen Sie nach folgendem Codeschnipsel:

```
</p><p> <filter></p><p> <filter-name>SecurityFilter</filter-name></p><p>
<filter-class>com.quark.web.security.servlet.ApplicationSecurityFilter</filter-class></p><p>
  <init-param></p><p> <param-name>provider</param-name></p><p>
  <param-value>com.quark.web.security.waffle.WaffleAuthenticationProvider</param-value></p><p>
  </init-param></p><p> <init-param></p><p>
  <param-name>provider/protocols</param-name></p><p>
  <param-value>com.quark.web.security.waffle.WaffleAuthenticationProvider</param-value></p><p>
  </init-param></p><p> </filter></p><p>
  <filter-mapping></p><p> <filter-name>SecurityFilter</filter-name></p><p>
  <url-pattern>*/</url-pattern></p><p> </filter-mapping> </p><p>
```

- 3 Entfernen Sie die Kommentierung des oben angeführten Codeschnipsels und sichern Sie die Datei.

Für Workspace und Desktop Clients gehen Sie zu der entsprechenden Web App, öffnen Sie die Datei web.xml und entfernen die Kommentierung des genannten Codeschnipsels.

- Für Desktop Clients: `C:\Program Files\Quark\Quark Publishing Platform\Server\webapps\webservices\WEB-INF`
- Für Workspace: `C:\Program Files\Quark\Quark Publishing Platform\Server\webapps\workspace\WEB-INF`

➔ Der Platform Server sollte als Dienst mit einem Konto als Lokales System laufen.

Konfiguration des Web Client für Mixed-Mode-Authentifizierung in mehreren Schritten

- 1 Gehen Sie zu `C:\Program Files\Quark\Quark Publishing Platform\Server\webapps\workspace`. Kopieren Sie die Datei PreLogin.jsp und benennen Sie sie um in local.jsp.
- 2 Gehen Sie zu `C:\Program Files\Quark\Quark Publishing Platform\Server\webapps\workspace\WEB-INF`
- 3 Öffnen Sie Web.xml und suchen Sie nach folgendem Codeschnipsel:

```
< filter > < filter-name > SecurityFilter </ filter-name > < filter-class > com.quark.web.security.servlet.ApplicationSecurityFilter </ filter-class > < init-param > < param-name > provider </ param-name > < param-value > com.quark.web.security.waffle.WaffleAuthenticationProvider </ param-value > </ init-param > < init-param > < param-name > provider/protocols </ param-name > < param-value > NTLM </ param-value > </ init-param > <init-param> <param-name>exclude-url-patterns</param-name> <param-value>/local.jsp,/Login.jsp</param-value> </init-param> </ filter > < filter-mapping > < filter-name > SecurityFilter </ filter-name > < url-pattern > /* </ url-pattern > </ filter-mapping >
```
- 4 Entfernen Sie die Kommentierung des oben angeführten Codeschnipsels und sichern Sie die Datei. Die fettgedruckten Zeilen des oben gezeigten Codes müssen hinzugefügt werden.
- 5 Starten Sie den Server neu und greifen Sie auf die entsprechenden Web-Anwendungen mit folgenden URLs zu:
 - Für einen Zugang über die native Platform Benutzer Authentifizierung verwenden Sie die URL: `http://localhost:61400/workspace/local.jsp`
 - Für einen Zugang über die Windows Authentifizierung verwenden Sie die URL: `http://localhost:61400/workspace/local.jsp`

➔ Die oben gezeigten Schritte können für Web Admin oder andere Web-Anwendungen wiederholt werden.

Bearbeiten von „log4j.xml“

Hinzufügen von Benutzerinformationen zu protokollierten Ereignissen

Sie können die Datei log4j.xml so konfigurieren, dass jedes protokollierte Ereignis mit der Benutzer-ID und dem Benutzernamen ergänzt wird:

- 1 Öffnen Sie die Datei „log4j.xml“ im Ordner {QPP Server}/conf.
- 2 Legen Sie den Parameter `conversionPattern` fest:

```
<appender name="QpsServerAllFileAppender"
class="org.apache.log4j.RollingFileAppender" > <param name="file"
value="log/QppServer.log"/> <param name="maxFileSize" value="10MB"/> <param
name="maxBackupIndex" value="10"/> <layout
class="org.apache.log4j.EnhancedPatternLayout"> <param
name="conversionPattern" value="%d %p [%c][%t]
[User (Id:%properties{qpp.user.id},Name:%properties{qpp.user.name})] - %
m%n"/> </layout> </appender>
```

Konfigurieren der Anmeldestufen

Sie können die Datei "log4j.xml" zum Einstellen der Anmeldestufen bearbeiten und JConsole verwenden, um die Anmeldestufen (logging levels) nach dem Start von Quark Publishing Platform Server zu ändern. Darüber hinaus ist es möglich, unterschiedliche Anmeldestufen für Ausnahmefälle einzustellen.

Ändern der Protokollierungsstufen in „log4j.xml“

Sie können die Protokollierungsstufen für Quark Publishing Platform Web Client und Quark Publishing Platform Server ändern. Die vorhandenen Optionen sind **ERROR**, **INFO**, **WARN**, **DEBUG**, **SQLTRACE** und **TRACE**.

- **ERROR** = enthält Meldungen, die unterbrochene und fehlgeschlagene Datenanforderungen (Requests) anzeigen.
- **INFO** = enthält Meldungen, die den Status von Diensten (Services) anzeigen.
- **WARN** = enthält unkritische Fehlermeldungen mit Diensten.
- **DEBUG** = enthält Meldungen, die die Benutzung von Server-Ressourcen anzeigen.
- **SQL_TRACE** = enthält Meldungen anhand der Aktivität im Hinblick auf SQL-Requests.
- **TRACE** = enthält Meldungen anhand der Aktivität im Hinblick auf Requests.

In der Java Dokumentation finden Sie weitere Informationen zu Protokollierungsstufen.

So ändern Sie die Protokollierungsstufe:

- 1 Öffnen Sie die Datei „log4j.xml“ im Ordner {QPP Server}/conf.
- 2 Um die Protokollierungsstufe für die Quark Publishing Platform Web Client Aktivität festzulegen, scrollen Sie bis `<logger name=com.quark.qpp.web.webeditor`. Der Eintrag besitzt folgende Struktur:

```
<logger name="com.quark.qpp.web.webeditor" additivity="false"> <level
value="INFO" /> <appender-ref ref="WebHubAsyncAppender" /> </logger>
```

- 3 Um die Protokollierungsstufe für die Quark Publishing Platform Server Aktivität festzulegen, scrollen Sie bis `<logger name=com.quark.qpp`. Der Eintrag besitzt folgende Struktur:

```
<logger name="com.quark.qpp"> <level value="INFO" /> </logger>
```

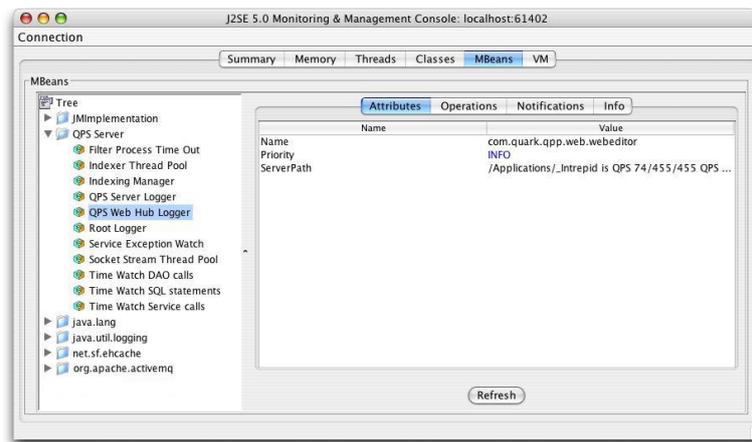
- Um die Protokollierungsstufe für andere Aktivitäten festzulegen, scrollen Sie bis "<root>". Der Eintrag besitzt folgende Struktur:

```
<root> <priority value="ERROR" /> <appender-ref ref="QppServerAsyncAppender" /> </root>
```

- Sichern Sie die Datei „log4j.xml“ und schließen Sie sie.

Ändern der Protokollstufen nach dem Start von Quark Publishing Platform Server

- Rufen Sie bei laufendem Quark Publishing Platform Server die **Platform Server Console** auf.
- Klicken Sie auf **JConsole**, um ein Fenster für die Überwachung unterschiedlicher Details der Leistung von Quark Publishing Platform Server aufzurufen.
- Klicken Sie auf das Register **MBeans**. Auf der linken Seite des Registers **MBeans** werden einzelne Funktionen von Quark Publishing Platform Server als Baumansicht angezeigt.
- Öffnen Sie **Platform Server > Logging** in der Baumstruktur.
- Klicken Sie auf das Register **Operationen**.



Verwenden Sie JConsole, um die Prioritätsstufen der Protokollierung einzustellen

- Um die Protokollstufen von Quark Publishing Platform Server oder des Quark Publishing Platform Web Clients zu ändern, stellen Sie die Protokollstufe in der Textbox ein und klicken auf die entsprechende Schaltfläche.
- ➔ Die in JConsole vorgenommenen Änderungen werden sofort wirksam, nach einem Neustart von Quark Publishing Platform Server werden jedoch die Änderungen in „log4j.xml“ angewendet.

Ändern der Protokollierung von Ausnahmezuständen

Sie können die Protokollierung bekannter und unbekannter Ausnahmezustände einstellen, indem Sie zwei Werte in der Datei „ServerApp.properties“ bearbeiten.

- Öffnen Sie den Ordner „conf“ im Quark Publishing Platform Server Ordner.
- Öffnen Sie „ServerApp.properties“ in einer Textbearbeitungsanwendung.

- 3 Falls Quark Publishing Platform Ausnahmestände nicht im Quark Publishing Platform Server Protokoll aufgezeichnet werden sollen, stellen Sie `server.logppserviceexception` auf `false` ein.
- 4 Falls die Protokollierung unbekannter Ausnahmestände unterbunden werden soll, stellen Sie `server.logthrowable` auf `false` ein.
- 5 Speichern Sie „ServerApp.properties“ und schließen Sie die Datei.

Modifizieren der Evaluierungseinstellungen für Abfragebenachrichtigungen

Quark Publishing Platform Server benachrichtigt alle offenen **Suchergebnispaletten**, wenn die in der Palette angezeigten Assets verändert wurden. Sie können verschiedene Parameter bearbeiten, um die für die Evaluierung von Bedingungen und die Übergabe dieser Abfragebenachrichtigungen verwendeten Strategien und Ressourcen zu beeinflussen. Der optimale Wert für die meisten Parameter hängt von der für Quark Publishing Platform Server ausgewählten Datenbank und Hardware ab.

➔ Nur erfahrene Administratoren sollten die unten beschriebenen Einstellungen ändern. Falls Sie Unterstützung benötigen, wenden Sie sich bitte an den Quark Enterprise Support.

- 1 Öffnen Sie den Ordner "conf" im Quark Publishing Platform Server Ordner.
- 2 Öffnen Sie "Query.properties" in einer Textbearbeitungsanwendung.
- 3 Sie können die Leistung mit einer HSQL Datenbank optimieren, indem Sie Quark Publishing Platform Server anweisen, Asset-Metadaten in einer temporären Tabelle zu speichern, bevor die Abfragebenachrichtigungen übergeben werden. Um die Suchleistung mit einer HSQL Datenbank zu optimieren, scrollen Sie zum Parameter `query.notification.useTempTable=` und setzen Sie den Wert auf `true`.

➔ Wenn Sie keine HSQL Datenbank verwenden, setzen Sie den Wert auf `false`. Standardmäßig stellt der Quark Publishing Platform Server Installer diesen Parameter für die von Ihnen gewählte Quark Publishing Platform Datenbankoption korrekt ein.

- 4 Quark Publishing Platform Server verwendet zwei Pools mit Threads für die Evaluierung von Abfragebenachrichtigungen — die "Generic Notification Evaluator Thread Pool Configuration" und die "Simple Notification Evaluator Thread Pool Configuration". Der "Generic" Thread Pool evaluiert Abfragebenachrichtigungen mithilfe einer Datenbank. Der "Simple" Thread Pool verwendet eine einfachere Strategie, um Abfragebenachrichtigungen ohne Beteiligung einer Datenbank zu evaluieren. Stellen Sie folgende Eigenschaften im Bereich "Generic Notification Evaluator Thread Pool Configuration" ein:

- Um die maximale Anzahl gleichzeitiger Threads festzulegen, die im Hintergrund ablaufen, um Benachrichtigungen zu evaluieren, die einen Datenbankzugriff erfordern, stellen Sie den Wert `query.notification.generic.pool.maxActive` ein. Erhöhen Sie diesen Wert, um die Leistung zu verbessern, wenn Sie Hardware mit mehreren Prozessoren und umfangreichem Systempeicher verwenden.

- Um die maximale Anzahl von "Idle"-Threads im Pool festzulegen, stellen Sie den `query.notification.generic.pool.maxIdle` ein.
 - Um die minimale Anzahl von "Idle"-Threads im Pool festzulegen, stellen Sie `query.notification.generic.pool.minIdle` ein.
 - Um die Mindestdauer einzustellen, für die ein Thread "idle" sein kann, bevor er aus dem Pool entfernt werden kann, stellen Sie die Dauer für `query.notification.generic.pool.minEvictableIdleTimeMillis` in Millisekunden ein.
 - Um die Anzahl der Millisekunden anzugeben, nach denen ein Background-Evictor-Thread zum Entfernen der Idle-Threads ausgeführt werden soll, stellen Sie `query.notification.generic.pool.timeBetweenEvictionRunsMillis` ein.
- 5** Abfragenachrichtigungen für bestimmte Assetänderungen können effizient mithilfe einer einfacheren Strategie evaluiert werden, die keinen Datenbankzugriff erfordert. Stellen Sie folgende Eigenschaften im Bereich "Simple Notification Evaluator Thread Pool Configuration" ein:
- Um die maximale Anzahl von gleichzeitig vorhandenen im Hintergrund ausgeführten Threads festzulegen, stellen Sie `query.notification.simple.pool.maxActive` ein. Erhöhen Sie diesen Wert, um die Leistung zu verbessern, wenn Sie Hardware mit mehreren Prozessoren und umfangreichem Systempeicher verwenden.
 - Um die maximale Anzahl von "Idle"-Threads im Pool festzulegen, stellen Sie `query.notification.simple.pool.maxIdle` ein.
 - Um die minimale Anzahl von "Idle"-Threads im Pool festzulegen, stellen Sie `query.notification.simple.pool.minIdle` ein.
 - Um die Mindestdauer einzustellen, für die ein Thread "idle" sein kann, bevor er aus dem Pool entfernt werden kann, stellen Sie die Dauer für `query.notification.simple.pool.minEvictableIdleTimeMillis` in Millisekunden ein.
 - Um die Anzahl der Millisekunden anzugeben, nach denen ein Background-Evictor-Thread zum Entfernen der Idle-Threads ausgeführt werden soll, stellen Sie `query.notification.simple.pool.timeBetweenEvictionRunsMillis` ein.
- 6** Zum Konfigurieren der JMS Topologie für die Abfragenachrichtigung stellen Sie die Eigenschaft `query.notification.topicPerSession` ein. Wenn das Verhalten der Quark Publishing Platform Versionen vor 8.1 nachgebildet werden soll, stellen Sie diesen Wert auf `false` ein.
- 7** Bearbeiten Sie die Werte anderer Eigenschaften in der Konfigurationsdatei "Query.properties" nicht.
- 8** Speichern Sie "Query.properties" und schließen Sie die Datei.
- 9** Starten Sie Quark Publishing Platform Server neu, damit die neuen Einstellungen wirksam werden.

Datenbankeigenschaften

Sie können die URL der Datenbank, den Datenbank-Benutzernamen, das Datenbank-Benutzerpasswort und die Größe des Datenbank-Verbindungs-Pools manuell vorgeben.

So ändern Sie die Datenbankeigenschaften:

- 1 Öffnen Sie die Datei „Database.properties“; Sie finden sie im Ordner `[QPP Server]/conf`.
 - 2 Scrollen Sie bis zum Abschnitt `Database related configuration`.
 - 3 Um den Driver-Class-Namen anzugeben, ersetzen Sie den Wert `qpp.jdbc.driverClassName`. Für eine HSQL Datenbank würden Sie zum Beispiel `qpp.jdbc.driverClassName=org.hsqldb.jdbcDriver` verwenden.
 - 4 Um die URL der Datenbankverbindung anzugeben, ersetzen Sie den Pfad hinter `qpp.jdbc.url`.
 - 5 Um den Datenbank-Benutzernamen anzugeben, ersetzen Sie den Wert `qpp.jdbc.userName`.
 - 6 Um das Datenbank-Benutzerpasswort anzugeben, ersetzen Sie den Wert `qpp.jdbc.password`.
 - 7 Um die Größe des Datenbankverbindungs-Pools anzugeben, ändern Sie den Wert `qpp.jdbc.maxActive`.
 - 8 Um die Mindest- und die Höchstanzahl von Verbindungen festzulegen, die idle sein können, ändern Sie die Werte `qpp.jdbc.maxIdle` und `qpp.jdbc.minIdle`.
 - 9 Um die Mindestdauer in Millisekunden festzulegen, wie lange sich eine Verbindung im Pool befinden muss, bevor sie entfernt werden kann, ändern Sie den Wert `qpp.jdbc.minEvictableIdleTimeMillis`.
 - 10 Um die Dauer in Millisekunden festzulegen, nach der der Evictor-Thread ausgeführt werden sollte, um eine Leerlaufverbindung zu entfernen, ändern Sie den Wert `qpp.jdbc.timeBetweenEvictionRunsMillis`.
 - 11 Um das Bündeln von prepared Statements anzupassen, ändern Sie den Wert `qpp.jdbc.poolPreparedStatements`.
 - 12 Um die Höchstanzahl von Verbindungen einzustellen, die im No-Autocommit-Pool hergestellt werden dürfen, ändern Sie den Wert `qpp.jdbc.maxActive_noAutoCommit`.
 - 13 Sichern Sie „Database.properties“ und schließen Sie die Datei.
- ➔ Falls eine Firewall zwischen dem Platform Server und dem Datenbank-Server konfiguriert ist, die TCP-Sessions beenden kann, wenn sie längere Zeit idle waren, sollten folgende Konfigurationsänderungen in der Datei „Database.properties“ vorgenommen werden.
- Setzen Sie den Wert `qpp.jdbc.minIdle` auf `0`.
 - Setzen Sie den Wert `qpp.jdbc.minEvictableIdleTimeMillis` auf `60000`.
 - Setzen Sie den Wert `qpp.jdbc.timeBetweenEvictionRunsMillis` auf `60000`.

Platform Server muss neu gestartet werden, damit diese Einstellungen wirksam werden.

Transformationseigenschaften

Sie können Transformationseigenschaften verwenden, um die Ausgabe verschiedener vordefinierter Transformationen zu modifizieren. Änderungen an der Datei „Transformation.properties“ verändern die externen Befehle, die für die Transformation von Assets aufgerufen werden. Befehle in dieser Datei können vordefinierte Makros wie `<temp>`, `<source>`, `<extension>` und `<output>` enthalten.

So ändern Sie die Transformationseigenschaften:

- 1 Öffnen Sie die Datei „Transformation.properties“, die sich im Ordner `[QPP Server]/conf` befindet.
- 2 Um ein Muster einzugeben, das nicht als Wert (oder Teil eines Wertes) für eine andere Eigenschaft wahrgenommen wird, definieren Sie einen Wert für `transformer.spaceEncodingPattern`. Dieses Muster wird verwendet, um Leerzeichen in Befehlen und anderen Eigenschaftswerten zu umgehen.
- 3 Um festzulegen, welche Dateitypen mit ImageMagick verarbeitet werden dürfen, geben Sie eine durch Kommas separierte, Groß- und Kleinschreibung nicht berücksichtigende Liste der Dateierweiterungen in `imTransformer.extensions` ein.
- 4 Um den temporären Dateispeicherort festzulegen, der von ImageMagickTransformer verwendet werden soll, geben Sie einen Pfad für `imTransformer.tempDir` an.
- 5 Um den ImageMagick Befehl festzulegen, der für die Erzeugung von Output mit ImageMagickTransformer ausgeführt werden soll, geben Sie einen Wert für `imTransformer.jpg.imCommand` ein.
- 6 Um festzulegen, welche Dateitypen mit JawsTransformer verarbeitet werden dürfen, geben Sie eine durch Kommas separierte, Groß- und Kleinschreibung nicht berücksichtigende Liste der Dateierweiterungen in `jawsTransformer.extensions` ein.
- 7 Um den temporären Dateispeicherort festzulegen, der von JawsTransformer verwendet werden soll, geben Sie einen Pfad für `jawsTransformer.tempDir` an.
- 8 Um den Jaws Befehl festzulegen, der für die Erzeugung von Output mit JawsTransformer ausgeführt werden soll, geben Sie einen Wert für `jawsTransformer.jpg.command` ein.
- 9 Sichern Sie „Transformation.properties“ und schließen Sie die Datei.

Session-Timeout

Sie können festlegen, wie lange einzelne Sessions idle bleiben dürfen, bevor ein Timeout eintritt, und außerdem den Abstand zwischen den Tests auf idle Sessions einstellen.

So legen Sie die Timeout-Parameter fest:

- 1 Öffnen Sie die Datei „ServerApp.properties“, die sich im Ordner `[QPP Server]/conf` befindet.
 - 2 Scrollen Sie bis zum Eintrag `session.maxIdle=`.
 - 3 Geben Sie die Anzahl der Sekunden ein, die eine Session idle sein darf, bevor ein Timeout eintritt.
 - 4 Um die Häufigkeit einzustellen, mit der Quark Publishing Platform Server eine Prüfung im Hintergrund auf idle Sessions vornimmt, geben Sie eine Zahl (von Sekunden) in `session.eviction.thread.delay` ein.
 - 5 Speichern Sie „ServerApp.properties“ und schließen Sie die Datei.
- ➔ Webclient Timeout-Einstellungen können geändert werden, indem Sie den Eintrag `ajaxTimeout` in der Datei „Workspace-Config.xml“ anpassen (in Millisekunden). Sie finden diese Datei im Ordner `{Installationsverzeichnis}/webapps/workspace/WEB-INF/classes`

Aktualisieren des Assetspeicher-Status

Sie können einen Hintergrund-Thread festlegen, der regelmäßig ausgeführt wird, um den Status des Quark Publishing Platform File Servers zu prüfen.

So legen Sie das Aktualisierungsintervall für den Assetspeicher-Status fest:

- 1 Öffnen Sie den Ordner "conf" im Quark Publishing Platform Server Ordner.
- 2 Öffnen Sie "ServerApp.properties" in einer Textbearbeitungsanwendung.
- 3 Scrollen Sie bis zum Eintrag `repository.status.updator.sleepInterval=`.
- 4 Geben Sie einen Wert ein, um die Anzahl der Sekunden festzulegen, nach der der Assetspeicher-Status-Thread ausgeführt wird.
- 5 Speichern Sie "ServerApp.properties" und schließen Sie die Datei.

Umzug von Quark Publishing Platform Renderer

Falls Sie Quark Publishing Platform Renderer auf einen anderen Computer bringen müssen, brauchen Sie Quark Publishing Platform Server nicht erneut zu installieren. Stattdessen können sie die Datei „ManagerConfig.xml“ folgendermaßen ändern:

- 1 Öffnen Sie die Datei „ManagerConfig.xml“, die sich im Ordner `[QPP Server]/conf` befindet.
- 2 Scrollen Sie zum Abschnitt `<connectioninfo>` unten in der Datei „ManagerConfig.xml“.
- 3 Ändern Sie den Wert des Eintrags `name` in die IP-Adresse oder in den Host-Namen des neuen Quark Publishing Platform Renderers um.
- 4 Ändern Sie den Wert des Eintrags `port` zur Portnummer des neuen Quark Publishing Platform Renderers.

- 5 Sichern Sie die Datei „ManagerConfig.xml“ und starten Sie Quark Publishing Platform Server.
 - 6 Suchen Sie in der Datei „QppServer.log“ nach der Zeile `Successfully registered with QXPS`, um die Änderung zu prüfen.
- ➔ Alternativ können Sie den Quark Publishing Platform Renderer Manager Client mit der URL `http://[server]:[port]/qxpsmadmin` öffnen und diese Änderungen dort vornehmen.
- ➔ Solange Sie **Realm verif. für Admin. Requests** nicht aktivieren und einen Benutzernamen und ein Passwort in das Register **HTTP** der QuarkXPress Dialogbox **Serverkonfiguration** eingeben, können Sie die Einträge `<user>` und `<password>` frei lassen. Dies gilt ebenfalls für die Felder Benutzername und Passwort in der Datei „ServerApp.properties“ (`qxps.username` und `qxps.password`).

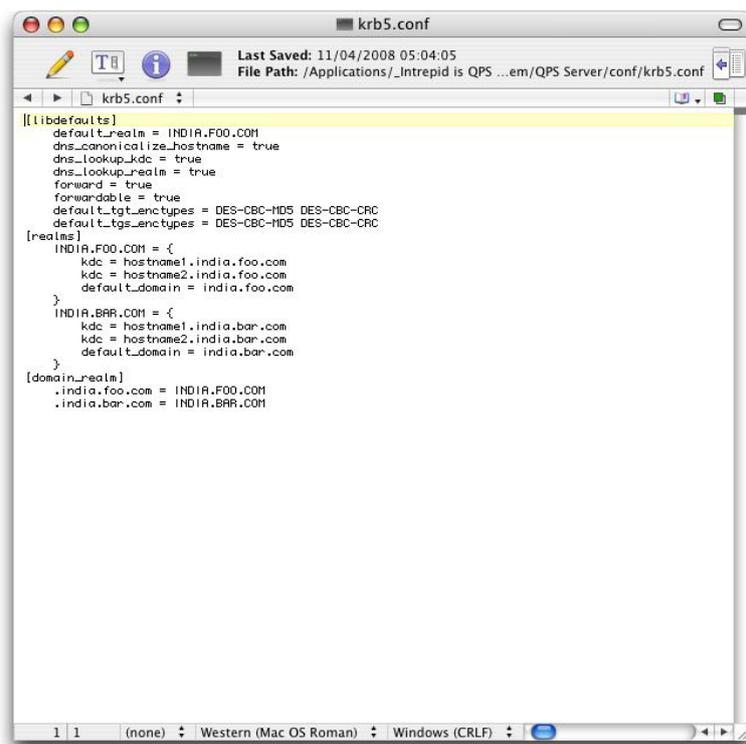
Integrieren von Quark Publishing Platform mit LDAP

Informationen zur Integration von Quark Publishing Platform und LDAP finden Sie unter "Verwalten von Benutzerlisten mit LDAP" in Ihrem *Handbuch zu Quark Publishing Platform*.

Verwenden der Kerberos Authentifizierung

So verwenden Sie die Kerberos Authentifizierung:

- 1 Öffnen Sie „krb5.conf“ in einer Textbearbeitungsanwendung. Die Datei „krb5.conf“ enthält die Konfigurationsdaten für Kerberos, darunter den Speicherort des Key Distribution Center (KDC) und die Admin Server für die Kerberos Realms. Kerberos ist ein sicheres Verfahren zur Authentifizierung von Anforderungen (Requests) eines Dienstes über ein Computernetzwerk.



Führen Sie die Realm-Einstellungen in der Datei „krb5.conf“ durch.

- 2 Bearbeiten Sie „krb5.conf“, um alle Realm-Einstellungen vorzunehmen. Zu diesen zählen die Einstellungen der Realm-Namen, des Standard-Realms, der Realm-Domain-Mappings und der KDCs für die Realms.
- 3 Sichern Sie „krb5.conf“ und schließen Sie die Datei.

➔ Nach der Konvention werden Realm-Namen in Großbuchstaben und Domain-Namen in Kleinbuchstaben eingegeben.

Verwenden der Einfachen Authentifizierung

Wenn Sie einen anderen Directory Server als Active Directory unter Windows vorziehen, benutzen Sie in den meisten Fällen das Einfache Authentifizierungs-Schema. So verwenden Sie die Einfache Authentifizierung:

- 1 In der LDAP Attribut-Zuordnungsgruppe definieren Sie den Authentifizierungsnamen = :uid: **User Name**=uid.
- 2 Für die Einfache Authentifizierung ist **Realm Name** (Bereichsname) optional.

Verbinden von LDAP Benutzerpasswörtern mit Quark Publishing Platform Server

Wenn Sie Quark Publishing Platform anhand der unten folgenden Anleitung konfigurieren und die Verbindung zwischen Quark Publishing Platform Server und dem Directory Server unterbrochen wird, können Benutzer sich dennoch am Quark Publishing Platform Server anmelden, indem sie ihre Directory Server Anmeldedaten verwenden.

Wenn ein Benutzer sich erstmals erfolgreich am Quark Publishing Platform Server anmeldet, werden in Quark Publishing Platform verschlüsselte User Credentials gespeichert. Diese Daten können später für Anmelde-Operationen verwendet werden, wenn keine Verbindung zwischen Quark Publishing Platform Server und dem Directory Server besteht.

So legen Sie fest, dass das LDAP Benutzerpasswort abgefragt, verschlüsselt und in der Datenbank von Quark Publishing Platform Server gespeichert werden kann:

- 1 Öffnen Sie die Datei „ServerApp.properties“; Sie finden sie im Ordner `[QPP Server]/conf`.
- 2 Wenn Sie zulassen möchten, dass Benutzer sich am Quark Publishing Platform Server anmelden, wenn die Verbindung zwischen Quark Publishing Platform Server und dem Directory Server unterbrochen ist, ändern Sie den Wert für `authentication.external.cacheTicket=` in `true`. Die Standardeinstellung für `authentication.external.cacheTicket=` ist `false` (deaktiviert).
- 3 Um darauf hinzuweisen, dass es erforderlich ist, das Flag „CollectionInfo#isAccessibleChildrenAvailable()“ während der Laufzeit für jede Sammlung im Kontext zu evaluieren, sollte der Wert von `collections.evaluateAccessibleChildrenFlag=` `true` sein (Standard).
- 4 Sobald ein Element vom LDAP Server geholt wurde, wird es im Platform Server gecached und vom Cache für eine angegebene Anzahl von Sekunden bereitgestellt und nach dieser Zeit erneut geholt. Um die Lebensdauer in Sekunden für LDAP Cache-Elemente anzugeben, geben Sie einen Wert für `ldap.cache.timetolive=` ein.
- 5 Um die Verzögerungszeit in Sekunden für die LDAP Synchronisation anzugeben, setzen Sie den Wert für `ldap.synchronization.thread.delay=` auf `true`. Wenn keine LDAP Synchronisation erforderlich ist, setzen Sie den Wert auf `-1`.
- 6 Speichern Sie „ServerApp.properties“ und schließen Sie die Datei.

Konfiguration von Platform für MSSQL Server Windows Authentifizierung

Die folgenden Änderungen sind erforderlich, damit Quark Publishing Platform Server mit einer SQL Server-Datenbank mit aktiver Windows-Authentifizierung laufen kann:

- 1 Öffnen Sie die Datei „database.properties“ im Ordner `[QPP Server]/conf`.
- 2 Entfernen Sie den Kommentar bei folgender Eigenschaft und geben Sie die passenden Daten ein:

```
qpp.jdbc.url=jdbc:sqlserver://<your-host-name>\<instanceName>;databaseName=qppdb;integratedSecurity=true;
```
- 3 Fügen Sie die Java Classpath Variable mit einem Pointer auf `sqljdbc_auth.dll` hinzu, die Sie bei den Microsoft JDBC Treibern finden.
 Download hier: [msdn.microsoft.com/en-us/library/mt683464\(v=sql.110\).aspx](https://msdn.microsoft.com/en-us/library/mt683464(v=sql.110).aspx).
- 4 Starten Sie den Server.

Beschränken der Workspace Browser Paletten

So beschränken Sie die Anzahl von **Workspace Browser**-Fenstern, die gleichzeitig vom Anwender geöffnet werden können:

- 1 Öffnen Sie die Datei „Query.properties“, die sich im Ordner [QPP Server]/conf befindet.
- 2 Geben Sie einen Wert für den Parameter `query.maxWatchedQueryCountPerSession` ein.
- 3 Speichern Sie „Query.properties“ und schließen Sie die Datei.

Ändern der Suchergebnis-Einstellungen

- Um die Zahl der Ergebnisse einzugrenzen, die für eine auf *Assignments* basierende Suche ausgegeben werden, geben Sie einen positiven Wert (nicht Null) für den Parameter `query.maxAssignmentFetchCount` ein.
- Um die Zahl der Ergebnisse einzugrenzen, die für eine auf *Anwender* basierende Suche ausgegeben werden, geben Sie einen positiven Wert (nicht Null) für den Parameter `query.maxRowFetchCount` ein.

Einstellen der Erkennung individueller Inhaltstypen

Standardmäßig ist Quark Publishing Platform so konfiguriert, dass Formate von Quark, verbreitete Bildformate, bekannte Content-Formate (Microsoft Office, Plain Text, RTF, PDF, HTML), XML, XHTML, DITA Dokumente und BusDoc Dokumente erkannt werden. Sie können Quark Publishing Platform jedoch so konfigurieren, dass versucht wird, automatisch weitere, individuelle Inhaltstypen zu erkennen. Gehen Sie dazu folgendermaßen vor:

- 1 Um einen bestimmten Inhaltstyp durch das Untersuchen des MIME-Typs einer Datei zu erkennen, stellen Sie sicher, dass der MIME-Typ in der Datei „tika-mimetypes.xml“ aufgeführt ist. Sollte der MIME-Typ in dieser Datei nicht vorkommen, können Sie ihn in „custom-mimetypes“ hinzufügen. Dateitypen lassen sich anhand ihrer Dateinamenserweiterung, mit Magic Bytes oder (bei XML Dateien) mithilfe des Root-Elements erkennen. Beispiel:

```
<mime-type type="application/dita+xml; format=busdoc"> <sub-class-of
type="application/dita+xml;format=topic"/> <_comment>Business
Documents</_comment> </mime-type>
```

Strukturell gleichen, jedoch semantisch unterschiedlichen XML-Dateien lassen sich unterschiedliche MIME-Typen zuordnen. XML-Dateitypen lassen sich anhand der in der entsprechenden XML angewendeten Verarbeitungsanweisungen unterscheiden. Fügen Sie dazu einen MIME-Typ-Eintrag zur Datei „custom-xml-types-ext.xml“ im Ordner „ext“ auf dem Platform Server hinzu. Beispiel:

```
<mime-type type="application/dita+xml; format=busdoc"
sub-class-of="application/dita+xml; format=topic"> <pi
name="Xpress">productLine="busdoc"</pi> </mime-type>
```

Weitere Informationen finden Sie unter „<http://tika.apache.org>“.

- 2 Konfigurieren Sie einen Quark Publishing Platform Inhaltstyp auf der Basis des erkannten MIME-Typs. Fügen Sie dazu einen Mapping-Eintrag zur Datei „content-mimetype-mappings-ext.xml“ im Ordner „ext“ auf dem Platform Server hinzu. Assets des spezifizierten, anwenderdefinierten MIME-Typs werden mit dem entsprechenden Inhaltstyp eingecheckt. Beispiel:

```
<content-type name="Business Document"> <mime-type>application/dita+xml;
format=busdoc</mime-type> </content-type>
```

- 3 Fügen Sie dazu einen passenden Mapping-Eintrag zur Datei „Indexing Channels-ext.xml“ im Ordner „ext“ auf dem Platform Server hinzu. Beispiel:

```
<mapping contenttype="Ratings Document" channel="ratingsPreviewChannel">
<parameter name="channelParam1">param value</parameter> <parameter
name="channelParam2">param value</parameter> </mapping>
```

Festlegen eines Standard-PDF-Ausgabestils

Benutzer von Quark Publishing Platform Web Client können Kopien von QuarkXPress Projekten oder QuarkCopyDesk Artikeln als PDFs abrufen. Die Einstellungen für solche PDFs werden ausschließlich im Standard-PDF-Ausgabestil für QuarkXPress Server festgelegt. Um den Standard-PDF-Ausgabestil festzulegen, aktualisieren Sie die Job Jackets Standarddatei wie unter „Dialogbox Job Jackets“ in Ihrer *Anleitung zu QuarkXPress Server* beschrieben. Der Standard-PDF-Stil kann ebenfalls in dem für die PDF verwendeten Publishing Channel vermerkt werden. Der Standard-Channel für die Ausgabe einer PDF aus QuarkXPress Projekten lautet „qxpPdf“.

Einstellen der Delivery-Channel Anzeige im Webclient

Delivery-Channels sind standardmäßig nicht sichtbar. Um sie verfügbar zu machen, stellen Sie die Eigenschaft `displaydeliverychannels` in der Datei `QPP_HOME\webapps\workspace\WEB-INF\classes\Workspace-Config.xml` auf `true`:

```
<property name="enableDeliveryChannels" value="true" />
```

Ändern der Unterscheidung zwischen Groß- und Kleinschreibung für Quark Publishing Platform Passwörter

So legen Sie die Unterscheidung zwischen Groß- und Kleinschreibung für Quark Publishing Platform Benutzerpasswörter fest:

- 1 Öffnen Sie die Datei „ServerApp.properties“; Sie finden sie im Ordner `[QPP Server]/conf`.
- 2 Wenn Sie festlegen möchten, dass bei Benutzerpasswörtern zwischen Groß- und Kleinschreibung unterschieden werden soll, setzen Sie `server.password.case.sensitive` auf `true`. Andernfalls geben Sie `false` ein.
- 3 Speichern Sie „ServerApp.properties“ und schließen Sie die Datei.

- ➔ Die Standardeinstellung für `server.password.case.sensitive` ist `true`. Quark Publishing Platform Administratoren sollten Benutzer darüber informieren, ob die Groß- und Kleinschreibung bei Passwörtern berücksichtigt wird.

Die Handhabung von Filtern und Index Service Einstellungen

Quark Publishing Platform Server verwendet vier Filter für die Erstellung von Voransichten; Sie können die Werte für jeden Filter einzeln in den „properties“-Dateien ändern. Es handelt sich um folgende Dateien:

- Die Datei „AsposeFilterServiceConfig“ liefert Voransichten und Miniaturen von Dateien der Formate Word, RTF, EXCEL, POWER POINT und Plain-Text.
- Die Datei „QxpsFilterServiceConfig.properties“ liefert Voransichten und Miniaturen von QuarkXPress Projekten und QuarkCopyDesk zur Verfügung.
- Die Datei „IMFilterServiceConfig.properties“ liefert Voransichten und Miniaturen der meisten Bilddateien.
- Die Datei „JawsFilterServiceConfig.properties“ liefert Voransichten und Miniaturen bestimmter Bilddateien – PDF Dateien, EPS Dateien und Adobe® Illustrator® Dateien.
- Die Datei „XADocFilterServiceConfig.properties“ liefert Voransichten und Miniaturen von XML-Dateien.

Folgende Befehle gelten für alle vier Filter:

- `<FILTER_NAME>filter.generateAttributes`: Stellen Sie den Wert auf `true` ein, um filterspezifische Attribute zu erzeugen.
- `<FILTER_NAME>filter.generatePreview`: Stellen Sie den Wert auf `true` ein, um Voransichten zu erzeugen.
- `<FILTER_NAME>filter.generateThumbnail`: Stellen Sie den Wert auf `true` ein, um Miniaturen zu erzeugen.
- `<FILTER_NAME>filter.generateText`: Stellen Sie diesen Wert auf `true` ein, um Text zu extrahieren und eine Volltextsuche zu ermöglichen.
- `<FILTER_NAME>filter.previewWidth`: Geben Sie mit diesem Wert die Breite von Voransichten (in Pixeln) an.
- `<FILTER_NAME>filter.previewHeight`: Geben Sie mit diesem Wert die Höhe von Voransichten (in Pixeln) an.
- `<FILTER_NAME>filter.thumbnailHeight`: Geben Sie mit diesem Wert die Höhe von Miniaturen (in Pixeln) an.
- `<FILTER_NAME>filter.thumbnailWidth`: Geben Sie mit diesem Wert die Breite von Miniaturen (in Pixeln) an.
- `<FILTER_NAME>filter.previewPages`: Geben Sie mit diesem Wert die Anzahl der zu erzeugenden Vorschauseiten an.

- `<FILTER_NAME>filter.processTimeout`: Geben Sie mit diesem Wert die Anzahl der Millisekunden an, nach der ein unvollständiger Indexvorschauprozess abgebrochen wird.

Einstellungen für den Indexdienst

Sie können die Parameter für `asset indexing` eingeben. Als separater Hintergrundprozess, der in Quark Publishing Platform Server ausgeführt wird, kann die Indexierung die Systemressourcen erheblich beanspruchen. Durch die Bearbeitung der Datei „Indexing.properties“ (wie im Folgenden beschrieben) können Sie die Indexierungsprozesse („Indexing Thread Pools“) verwalten, indem Sie die Anzahl der gleichzeitig vorhandenen Thread Pools und die Zeitintervalle zwischen den Thread Pools einstellen.

So verwalten Sie das Timing und die Threads der Asset-Indexierung:

- 1 Öffnen Sie die Datei „Indexing.properties“, die sich im Ordner `[QPP Server]/conf` befindet.
 - 2 Um die maximale Anzahl gleichzeitig im Hintergrund ablaufender Indexing Threads festzulegen, stellen Sie den Wert `indexingThread.pool.maxActive=` ein.
 - 3 Um die Zeit (in Millisekunden) anzugeben, die auf einen verfügbaren Thread aus dem Pool gewartet werden soll, stellen Sie den Wert `indexingThread.pool.maxWait=` ein. Ein Wert von `-1` bedeutet unbegrenztes Warten.
 - 4 Um die maximale Anzahl von „Idle“-Threads im Pool festzulegen, stellen Sie den Wert `indexingThread.pool.maxIdle=` ein.
 - 5 Um die Mindestanzahl von „Idle“-Threads im Pool festzulegen, stellen Sie den Wert `indexingThread.pool.minIdle=` ein.
 - 6 Um die Anzahl der Millisekunden anzugeben, für die sich ein Thread im Pool befinden kann, bevor er entfernt wird, stellen Sie den Wert `indexingThread.pool.minEvictableIdleTimeMillis=` ein.
 - 7 Um die Anzahl der Millisekunden anzugeben, nach denen der Evictor-Thread zum Entfernen der Idle-Threads ausgeführt werden soll, stellen Sie den Wert `indexingThread.pool.timeBetweenEvictionRunsMillis=` ein. Der Evictor-Thread läuft im Hintergrund.
- ➔ Wenn der Evictor-Thread ausgeführt wird, gibt er alle Idle-Threads aus dem Pool jenseits des Wertes `indexingThread.pool.maxIdle` frei. Die Anzahl der als `indexingThread.pool.minIdle` angegebenen Idle-Threads bleibt im Pool jedoch erhalten.
- 8 Um zuvor nicht indexierte Assets zu indexieren, wenn Quark Publishing Platform Server startet, geben Sie `true` für den Wert `indexing.indexPendingAssetsOnStartup=` ein.
 - 9 Sichern Sie „Indexing.properties“ und schließen Sie die Datei.

ASPOSE Filter

Der ASPOSE Filter verarbeitet Textdateien, Microsoft Word, Excel und PowerPoint Dokumente, RTF-Dateien und Plain-Text-Dateien. So passen Sie die ASPOSE Filter-Einstellungen an:

- 1 Öffnen Sie die Datei „AsposeFilterServiceConfig.properties“; Sie finden sie im Ordner [QPP Server]/conf.
- 2 `asposefilter.previewType=png`. Unterstützte Vorschautypen sind **png** und **jpg**.
- 3 `asposefilter.generatePreview=true`.
- 4 `asposefilter.generateThumbnail=true`.
- 5 `asposefilter.generateTableImages=false`.
- 6 `asposefilter.generateChartImages=false`.
- 7 `asposefilter.thumbnailHeight= Wert`.
- 8 `asposefilter.thumbnailWidth= Wert`.
- 9 `asposefilter.previewDpi= Wert`.
- 10 `asposefilter.reditionPreviewDpi= Wert`.
- 11 Zum Festlegen der Dauer, die ein Prozess warten soll, bevor er endet, wenn keine Indexierung stattfindet, geben Sie einen Wert in den Eintrag `asposefilter.processTimeOut= ein` (in Millisekunden).
- 12 Um die maximale Anzahl Seiten anzugeben, für die eine Vorschau erzeugt werden soll, geben Sie einen Wert in den Eintrag `asposefilter.previewPages= ein`.
- 13 Zur Definition des Layoutrandes der Vorschauseite für Dateien des Typs RTF, XML, HTML, HTM und TXT, geben Sie Werte in die folgenden Einträge ein:
 - `asposefilter.text.topMargin= Wert`.
 - `asposefilter.text.bottomMargin= Wert`.
 - `asposefilter.text.rightMargin= Wert`.
 - `asposefilter.text.leftMargin= Wert`.
- 14 Sichern und schließen Sie „AsposeFilterServiceConfig.properties“.

Konfiguration des ASPOSE-Filters für die PDF-Vorschau

Der Plattform Server bietet zwei Implementierungen des Filters für die Indexierung von PDF-Dokumenten:

- Jaws Filter
- ASPOSE Filter

Als Standard ist der JAWS Filter für die Indexierung von PDF-Dokumenten konfiguriert. So konfigurieren Sie den ASPOSE-Filter für die Indexierung von PDF-Dokumenten:

1 Öffnen Sie die Datei „AsposeFilterServiceConfig.properties“; Sie finden sie im Ordner [QPP Server]/conf.

2 Suchen Sie die nachfolgenden Zeilen und entfernen Sie den Kommentar:

```
<!-- <supported-file-type> <mac-os-type/>
<file-extension>pdf</file-extension> <mime-type/> <creator-code/>
</supported-file-type> <supported-file-type> <mac-os-type/> <file-extension/>
<mime-type>application/pdf</mime-type> <creator-code/> </supported-file-type>
<supported-file-type> <mac-os-type/> <file-extension/>
<mime-type>application/x-pdf</mime-type> <creator-code/>
</supported-file-type> -->
```

3 Öffnen Sie "JawsFilterServiceConfig.properties" in einer Textbearbeitungsanwendung.

4 Suchen Sie nach dem unterstützten Dateityp und versehen Sie ihn mit einem Kommentar:

```
<supported-file-type> <mac-os-type/> <file-extension>pdf</file-extension>
<mime-type/> <creator-code/> </supported-file-type>
```

5 Sichern und schließen Sie „AsposeFilterServiceConfig.properties“.

6 Sichern und schließen Sie „JawsFilterServiceConfig.properties“.

APS-Filter

So passen Sie die APS-Filter-Einstellungen an:

1 Öffnen Sie die Datei „ApsFilterServiceConfig.properties“; Sie finden sie im Ordner [QPP Server]/conf.

2 `apsfilter.generatePreview=true`.

3 `apsfilter.generateThumbnail=true`.

4 `apsfilter.generateAttributes=true`.

5 `apsfilter.generateText=true`.

6 Zum Festlegen der Dauer, die ein Prozess warten soll, bevor er endet, wenn keine Indexierung stattfindet, geben Sie einen Wert in den Eintrag `apsfilter.processTimeOut=` ein (in Millisekunden).

7 Um die maximale Anzahl Seiten anzugeben, für die eine Vorschau erzeugt werden soll, geben Sie einen Wert in den Eintrag `apsfilter.previewPages=` ein.

8 Speichern Sie "ApsFilterServiceConfig.properties" und schließen Sie die Datei.

POI Filter

So passen Sie die POI Filter-Einstellungen an:

1 Öffnen Sie die Datei „POIFilterServiceConfig.properties“, die sich im Ordner [QPP Server]/conf befindet.

2 `poifilter.approximateCharactersPerPage`. Legen Sie die Anzahl Zeichen fest, aus der ungefähr eine Textseite besteht. Dieser Wert bestimmt den Seitenumbruch für Textdateien.

- 3 `poifilter.previewMaxLineLength`. Legen Sie die maximale Anzahl Zeichen fest, die für die Erzeugung eines Vorschaubildes von Textdateien in einer Zeile platziert werden sollen.
- 4 `poifilter.fontName`. Der Name der Schrift, die für die Erzeugung eines Bildes aus Text verwendet werden soll.
- 5 `poifilter.previewFontSize`. Die Schriftgröße die für Text in einem Bild benötigt wird.
- 6 `poifilter.lineSpacing`. Die Abstände zwischen Zeilen in Vielfachen der in der Eigenschaft `poifilter.previewFontSize` definierten Schriftgröße.
- 7 Sichern Sie „POIFilterServiceConfig.properties“ und schließen Sie die Datei.

Konfiguration der Verarbeitung von MS Office Dokumenten

So passen Sie die MS Office Filter-Einstellungen an:

- 1 Öffnen Sie die Datei „OfficeServiceConfig.properties“ im Ordner `[QPP Server]/conf`.
- 2 Bearbeiten Sie die Eigenschaft `office.defaultImageFormat`, um den Standard-Ausgabeformat-Typ festzulegen. Das Standard-Format ist PNG.
- 3 Bearbeiten Sie die Eigenschaft `office.defaultImageFormat`, um den Standard-Ausgabeformat-Typ festzulegen.
- 4 Bearbeiten Sie die Eigenschaft `office.defaultVerticalResolution`, um die standardmäßige vertikale Auflösung von Voransichten festzulegen.
- 5 Setzen Sie die Eigenschaft `office.onePagePerSheet` auf `true` und der gesamte Inhalt des Arbeitsblatts wird auf einer einzigen Seite ausgegeben.
- 6 Bearbeiten Sie die Eigenschaft `office.showHiddenWorksheets`, um die Sichtbarkeit verborgener Arbeitsblätter zu steuern.
- 7 Um die Option einzustellen, mit der versteckte Spalten/Zeilen in den Ausgabeformaten HTML, XHTML, SMARTTABLE und CALS entfernt oder verborgen werden, bearbeiten Sie folgende Eigenschaften:
 - `office.defaultHiddenColumnDisplayType`=Eintrag. Die erlaubten Werte für diese Option sind `hide` oder `remove`.
 - `office.defaultHiddenRowDisplayType`=Eintrag. Die erlaubten Werte für diese Option sind `hide` oder `remove`.
- 8 Definieren Sie die Eigenschaft `office.defaultHorizontalResolution` `office.ppt.document.defaultImageFormat`, um das Standardbildformat für die Ausgabe von PowerPoint-Dokumenten festzulegen.
- 9 Definieren Sie die Eigenschaft `office.defaultHorizontalResolution` `office.ppt.document.image.defaultHorizontalScalingFactor`, um den voreingestellten horizontalen Skalierungsfaktor für die Ausgabe von PowerPoint-Dokumenten festzulegen. Diese Eigenschaft ist anwendbar für die Bildformate JPEG, GIF, PNG und BMP.

- 10 Definieren Sie die Eigenschaft `office.defaultHorizontalResolution` `office.ppt.document.image.defaultVerticalScalingFactor`, um den voreingestellten vertikalen Skalierungsfaktor für die Ausgabe von PowerPoint-Dokumenten festzulegen. Diese Eigenschaft ist anwendbar für die Bildformate JPEG, GIF, PNG und BMP.
- 11 Definieren Sie die Eigenschaft `office.defaultHorizontalResolution` `office.ppt.component.defaultImageFormat`, um das Standardbildformat für die Ausgabe von PowerPoint-Folien festzulegen.
- 12 Definieren Sie die Eigenschaft `office.defaultHorizontalResolution` `office.ppt.component.image.defaultHorizontalScalingFactor`, um den voreingestellten horizontalen Skalierungsfaktor für die Ausgabe von PowerPoint-Folien festzulegen. Diese Eigenschaft ist anwendbar für die Bildformate JPEG, GIF, PNG und BMP.
- 13 Definieren Sie die Eigenschaft `office.defaultHorizontalResolution` `office.ppt.component.image.defaultVerticalScalingFactor`, um den voreingestellten vertikalen Skalierungsfaktor für die Ausgabe von PowerPoint-Folien festzulegen. Diese Eigenschaft ist anwendbar für die Bildformate JPEG, GIF, PNG und BMP.
- 14 Definieren Sie die Eigenschaft `office.defaultHorizontalResolution` `office.visio.document.defaultImageFormat`, um das Standardbildformat für die Ausgabe von Visio-Dokumenten festzulegen.
- 15 Definieren Sie die Eigenschaft `office.defaultHorizontalResolution` `office.visio.document.image.defaultResolution`, um die Standardauflösung des Bildes für die Ausgabe von Visio-Dokumenten festzulegen. Diese Eigenschaft ist anwendbar für die Bildformate JPEG, GIF, PNG und BMP.
- 16 Definieren Sie die Eigenschaft `office.defaultHorizontalResolution` `office.visio.component.defaultImageFormat`, um das Standardbildformat für die Ausgabe von Visio-Seiten festzulegen.
- 17 Definieren Sie die Eigenschaft `office.defaultHorizontalResolution` `office.visio.component.image.defaultResolution`, um die Standardauflösung des Bildes für die Ausgabe von Visio-Seiten festzulegen. Diese Eigenschaft ist anwendbar für die Bildformate JPEG, GIF, PNG und BMP.

Drosselungsmechanismus für parallele Requests durch OfficeService und ChartingService

Um OfficeService und ChartingService für die Ausführung paralleler Requests zu konfigurieren, ändern Sie die beiden Property-Dateien:

- "ChartingPool.properties"
- "OfficePool.properties"

Konfiguration von „ChartingPool.properties“

- 1 Öffnen Sie die Datei „ChartingPool.properties“, die sich im Ordner `[QPP Server]/conf` befindet.

- 2 Mit der Eigenschaft `chartingThread.pool.maxActive` legen Sie die maximale Anzahl Threads fest, die zu jedem beliebigen Zeitpunkt aktiv sein können.
- 3 Mit der Eigenschaft `chartingThread.pool.whenExhaustedAction` bestimmen Sie, was geschehen soll, wenn die Poolgröße bereits erreicht wurde.
 - Wählen Sie `0`, um sofort mit einer Fehlermeldung `NoSuchElement` zu antworten.
 - Wählen Sie `1`, um einige Zeit abhängig vom Wert `maxWait` auf einen verfügbaren Thread zu warten.
 - Wählen Sie `2`, um den Pool zu vergrößern und eine neue Thread-Instanz zu liefern.
- 4 Mit der Eigenschaft `chartingThread.pool.maxWait` legen Sie die Zeitspanne in Millisekunden fest, die auf einen verfügbaren Thread gewartet werden soll. Wählen Sie `-1`, um auf unbestimmte Zeit zu warten.
- 5 Mit der Eigenschaft `chartingThread.pool.maxIdle` legen Sie die maximale Anzahl von Idle-Threads im Pool fest.
- 6 Mit der Eigenschaft `chartingThread.pool.minIdle` legen Sie die Mindestzahl von Idle-Threads im Pool fest.
- 7 Mit der Eigenschaft `chartingThread.pool.minEvictableIdleTimeMillis` legen Sie die Mindestzeit in Millisekunden fest, die ein Thread im Pool bleiben muss, bevor er exmittiert („herausgeworfen“) werden kann.
- 8 Mit der Eigenschaft `chartingThread.pool.timeBetweenEvictionRunsMillis` legen Sie die Zeit in Millisekunden fest, nach der der Eviction-Thread gestartet werden sollte, um Idle-Threads zu entfernen. Wählen Sie `-1`, um ihn niemals anzustoßen.

Konfiguration der Datei „OfficePool.properties“

- 1 Öffnen Sie die Datei „OfficePool.properties“, die sich im Ordner `[QPP Server]/conf` befindet.
- 2 Mit der Eigenschaft `officeThread.pool.maxActive` legen Sie die maximale Anzahl Threads fest, die zu jedem beliebigen Zeitpunkt aktiv sein können.
- 3 Mit der Eigenschaft `officeThread.pool.whenExhaustedAction` bestimmen Sie, was geschehen soll, wenn die Poolgröße bereits erreicht wurde.
 - Wählen Sie `0`, um sofort mit einer Fehlermeldung `NoSuchElement` zu antworten.
 - Wählen Sie `1`, um einige Zeit abhängig vom Wert `maxWait` auf einen verfügbaren Thread zu warten.
 - Wählen Sie `2`, um den Pool zu vergrößern und eine neue Thread-Instanz zu liefern.
- 4 Mit der Eigenschaft `officeThread.pool.maxWait` legen Sie die Zeitspanne in Millisekunden fest, die auf einen verfügbaren Thread gewartet werden soll. Wählen Sie `-1`, um auf unbestimmte Zeit zu warten.
- 5 Mit der Eigenschaft `officeThread.pool.maxIdle` legen Sie die maximale Anzahl von Idle-Threads im Pool fest.

- 6 Mit der Eigenschaft `officeThread.pool.minIdle` legen Sie die Mindestzahl von Idle-Threads im Pool fest.
- 7 Mit der Eigenschaft `officeThread.pool.minEvictableIdleTimeMillis` legen Sie die Mindestzeit in Millisekunden fest, die ein Thread im Pool bleiben muss, bevor er exmittiert („herausgeworfen“) werden kann.
- 8 Mit der Eigenschaft `officeThread.pool.timeBetweenEvictionRunsMillis` legen Sie die Zeit in Millisekunden fest, nach der der Eviction-Thread gestartet werden sollte, um Idle-Threads zu entfernen. Wählen Sie `-1`, um ihn niemals anzustoßen.

QuarkXPress Server Filter

So passen Sie die QuarkXPress Filtereinstellungen an:

- 1 Öffnen Sie die Datei „QxpsFilterServiceConfig.properties“, die sich im Ordner `[QPP Server]/conf` befindet.
 - 2 Setzen Sie den Wert `qxpsfilter.useSpreadForArticles=` auf `true`, um festzulegen, dass Voransichten für Artikel für jede Doppelseite erzeugt werden.
 - 3 Setzen Sie den Wert `qxpsfilter.useSpreadForProjects=` auf `true`, um Projektvoransichten mit Doppelseiten festzulegen.
 - 4 Standardmäßig ist die XML Dekonstruktion aktiviert. Die Standardeinstellung für den Eintrag `qxpsfilter.generateDeconstructedXML=` ist `true`. Um die XML Dekonstruktion zu deaktivieren, setzen Sie den Eintrag auf `false`.
 - 5 Standardmäßig ist die Full Text Search (FTS – Volltextsuche) aktiviert. Die Standardeinstellung für den Eintrag `qxpsfilter.generateText=` ist `true`. Um die FTS zu deaktivieren, setzen Sie den Eintrag auf `false`.
- ➔ Der Wert `true` für `qxpsfilter.generateDeconstructedXML` aktiviert auch FTS für QuarkXPress Projekte und QuarkCopyDesk Artikel. Die Texterzeugung hängt von der XML-Erzeugung ab. Falls Sie `generateDeconstructedXML` auf `false` setzen, empfiehlt Quark, `qxpsfilter.generateText` ebenfalls auf `false` einzustellen.
- 6 Um die Ansichtsgröße von Vorschaubildern anzugeben, stellen Sie den Wert `qxpsfilter.previewScale=` ein. Der Standardwert von `1` führt zu einer Vorschaugröße von 100 Prozent. Die Werte `2`, `3` und `6` bedeuten 200 Prozent, 300 Prozent und 600 Prozent.
 - 7 Um die Ansichtsgröße von Miniaturen anzugeben, stellen Sie den Wert `qxpsfilter.thumbnailScale=` ein. Der Standardwert von `1` führt zu einer Vorschaugröße von 100 Prozent. Die Werte `.5`, `2`, `3` und `6` bedeuten 50 Prozent, 200 Prozent, 300 Prozent und 600 Prozent.
 - 8 Um die Bildqualität von JPEG-Vorschaubildern anzugeben, stellen Sie den Wert `qxpsfilter.jpegQuality=` ein. Geben Sie `1` für die höchste Qualität, `2` für hohe Qualität, `3` für mittlere Qualität und `4` für die niedrigste Qualität ein.
 - 9 Speichern Sie „QxpsFilterServiceConfig.properties“ und schließen Sie die Datei.

JAWS Filtereinstellungen

Der JAWS Filter ist Teil der ImageMagick Filter® Service Configurations für EPS-, PDF- und Adobe® Illustrator® Dateien. So nehmen Sie die Einstellungen für den JAWS Filter vor:

- 1 Öffnen Sie die Datei „JawsFilterServiceConfig.properties“ im Ordner [QPP Server]/conf.
- 2 Um die Auflösung für Miniaturen und Voransichten festzulegen (in dpi), stellen Sie den Wert `jawsfilter.resolution=` ein.
- ➔ Der Wert `jawsfilter.resolution=` ist die Fallback-Auflösung, die verwendet wird, um das Vorschaubild auf die angegebene Größe zu skalieren, wenn die genaue Auflösung nicht berechnet werden kann.
- 3 Speichern Sie „JawsFilterServiceConfig.properties“ und schließen Sie die Datei.

XML Author Filtereinstellungen

Der Filter von XML Author ist Teil der Konfiguration von XML-Dateien. Mit XADocFilter werden Voransichten und Miniaturbilder von XML-Dateien erzeugt. So nehmen Sie die Einstellungen für den XML Author Filter vor:

- 1 Öffnen Sie die Datei „XADocFilterServiceConfig.properties“, die sich im Ordner [QPP Server]/conf befindet.
- 2 Um die Zeit festzulegen, die ein Prozess auf den Abschluss der Indexierung (mit XADocFilter) wartet, stellen Sie den Wert `xaDocfilter.processTimeOut=` ein
- 3 Um die Höhe und Breite von Voransichten anzugeben, passen Sie die Werte `xaDocfilter.previewHeight=` und `xaDocfilter.previewWidth=` an.
- 4 Um die maximale Anzahl Seiten anzugeben, für die eine Vorschau erzeugt werden soll, passen Sie den Wert `xaDocfilter.previewPages=` an.
- 5 Sichern Sie „XADocFilterServiceConfig.properties“ und schließen Sie die Datei.

Die Verzeichnisse ImageMagick, Jaws und DITA OT

Standardmäßig gehören ImageMagick, Jaws und das DITA Open Toolkit zum Lieferumfang von Quark Publishing Platform. Wenn Sie jedoch bereits vorhandene Installationen von ImageMagick oder Jaws besitzen, führen Sie folgende Arbeitsschritte durch:

- 1 Öffnen Sie die Datei „ServerApp.properties“, die sich im Ordner [QPP Server]/conf befindet.
- 2 Scrollen Sie bis zum Eintrag `IMAGE_MAGICK_HOME=`.
- 3 Geben Sie den Pfad zu Ihrem bereits vorhandenen ImageMagick Bin Ordner ein.
- 4 Scrollen Sie bis zum Eintrag `JAWS_HOME=`.
- 5 Geben Sie den Pfad zu Ihrem bereits vorhandenen Jaws Bin Ordner ein.
- 6 Scrollen Sie bis zum Eintrag `DITA_HOME=`.

- 7 Geben Sie den Pfad zu Ihrem bereits vorhandenen DITA Open Toolkit Ordner ein.
- 8 Speichern Sie „ServerApp.properties“ und schließen Sie die Datei.

Konfiguration der Volltextindexierung

So konfigurieren Sie die Filtereinstellungen für FTSIndex:

- 1 Öffnen Sie die Datei „LuceneTextIndexingConfig.properties“, die sich im Ordner `[QPP Server]/conf` befindet.
 - 2 Der standardmäßige Speicherort für Indexdateien ist der Quark Publishing Platform Installationsordner. Sie können diesen Speicherort jedoch ändern, indem Sie für den Parameter `lucene.index.dir=` den Ordner angeben, in dem Sie Indexdateien speichern möchten.
 - 3 Geben Sie die am häufigsten für die Textindexierung und die Analyse der Abfragebegriffe verwendete Sprache (Klasse) an, indem Sie den Parameter `lucene.analyzerClass=` ändern. Die Standardeinstellung – `StandardAnalyzer` – erkennt bei der Volltextsuche die englische Semantik. Wenn der Hauptanteil des Textes in Ihrem Workflow jedoch nicht in englischer Sprache vorliegt, können Sie eine andere Sprache angeben, um anhand der Semantik dieser Sprache eine genauere Suche zu ermöglichen. Die Sprachen sind in der Datei „LuceneTextIndexingConfig.properties“ aufgelistet.
- ➔ „LuceneTextIndexingConfig.properties“ enthält Daten zu jedem Parameter und Verknüpfungen zur Apache Dokumentation.
- 4 Um die Anzahl der Änderungen festzulegen, nach der eine Indexoptimierung durchgeführt werden soll, stellen Sie einen Wert für `lucene.maxModificationWithoutOptimisation` ein.
 - 5 Sichern Sie „LuceneTextIndexingConfig.properties“ und schließen Sie die Datei.

Charting Service

So konfigurieren Sie die Eigenschaften für den Charting Service:

- 1 Stellen Sie das Standard-Ausgabeformat (PDF/HTML/IMAGE/JSON) mithilfe des Parameters `charting.defaultOutputFormat` ein.
- 2 Legen Sie die Standardwerte für die Bildausgabe-Eigenschaften in folgenden Parametern fest:
 - `charting.defaultImageFormat=png`
 - `charting.defaultWidth=600`
 - `charting.defaultScale=`
- 3 Geben Sie den Pfad zur Standard-HTML-Vorlage an:
`charting.defaultHTMLTemplateURI=classpath\DefaultHTMLTemplate.html`

Integrieren von QLA mit Quark Publishing Platform

So konfigurieren Sie die Einstellungen für den Primär- und den Backup-Server von Quark® License Administrator (QLA) mit Quark Publishing Platform neu:

- 1 Öffnen Sie die Datei „Qla.properties“, die sich im Ordner `[QPP Server]/conf` befindet.
 - 2 Geben Sie die aktuelle IP-Adresse oder den Host-Namen des Computers, auf dem Sie QLA installiert haben, als Parameter `QlaServer.machinename=` ein.
 - 3 Geben Sie die Portnummer als Parameter `QLAServer.port=` ein.
 - 4 Falls Sie einen QLA Backupserver besitzen, legen Sie für die korrekte IP-Adresse oder den Host-Namen und die Werte für den Port die Parameter `Backup.QlaServer.machinename=` und `Backup.QlaServer.port=` fest.
 - 5 Geben Sie als Parameter `Qla.SerialNumber=` die QLA Seriennummer für Quark Publishing Platform Server ein. Die QLA Server Console und QLA Client Anwendungen zeigen Ihre Seriennummer an.
- ➔ Die Seriennummer wird in „Qla.properties“ auf der Basis des Validierungscodes aktualisiert, den Sie bei der Installation oder Aktualisierung von Quark Publishing Platform Server angegeben haben.
- 6 Speichern Sie „Qla.properties“ und schließen Sie die Datei.

Dynamische Konfiguration

Der Konfigurationsdienst wird zur Verwaltung der Publishing-Konfiguration verwendet:

- Hinzufügen/Aktualisieren neuer Publishing-Kanäle
- Hinzufügen/Aktualisieren neuer Publishing-Prozesse
- Hinzufügen/Aktualisieren neuer MIME-Typen für Inhaltstypen

Der Konfigurationsdienst ist verantwortlich für folgende Aktionen:

- Verwaltung der Konfigurationsdateien.
- Erfassen von Updates.
- Neu-Initialisierung dazugehöriger Bean-Definitionen, Vermeidung von Neustarts.
- Mithilfe der Neu-Initialisierungs-API werden Neustarts vermieden, nachdem die Konfigurationsdateien auf dem Server manuell geändert wurden.

In einer Multi-Server-Umgebung muss jeder Knoten in einem Cluster Zugriff auf einen .ext-Ordner haben und der gemeinsam genutzte Speicherplatz muss der Systemumgebungs-Variablen `classpth` hinzugefügt werden.

Erstellen eines anwenderdefinierten XML Mime-Typs

```
https://http://localhost:61400/rest/service/config/xmlmimetypes?
op=create&mimetype=application/xml;
format= researchreport&parentmimetype=application/xml;
format= smartcontent&xpath =/*[local-name()='section']/@type= ' researchreport
```

Abbilden des XML Mime-Typs auf einen Inhaltstyp in Platform

```
https://http://localhost:61400/rest/service/config/xmlmimetypes?  
op=mapcontenttype&contenttype=Research_Report&mimetypes=application/xml;  
format=researchreport
```

Erstellen von Publishing-Kanälen

```
https://http://localhost:61400/rest/service/config/publishingchannels?  
op=create&publishingchannellist=<publishingChannelList><publishingChannelInfoId="researchReportPdf"
```

Definition von Publishing-Kanälen für die Indexierung von Assets

```
https://http://localhost:61400/rest/service/config/publishingchannels?  
contenttype=Research_Report&loginname=Admin&loginpassword=Admin
```

Aktivieren der IPTC-Unterstützung

IPTC-Werte in Bilddateien können ausgelesen und der Attributliste der Bilder beigefügt werden. Um die IPTC-Funktionalität zu aktivieren oder zu deaktivieren öffnen Sie die Datei `\Server\conf\IMFilterServiceConfig.properties` und setzen die Eigenschaft `imfilter.generateIPTCAttributes` auf `true` oder `false`. Sie können darüber hinaus die Auflösung von Voransichten und Miniaturen einstellen, indem Sie einen Wert für `imfilter.resolution` eingeben.

Nur RMI und CORBA Clients

Standardmäßig ist für Quark Publishing Platform Clients nur Port 61400 erforderlich. Wenn Sie jedoch eigene RMI und CORBA Clients verwenden oder entwickeln, lesen Sie bitte das Nachfolgende.

Ändern der von Quark Publishing Platform Server verwendeten Ports

Standardmäßig ist für Quark Publishing Platform Clients nur Port 61400 erforderlich. Wenn Sie jedoch eigene RMI und CORBA Clients verwenden oder entwickeln, können Sie die Einstellungen für die Server- und Filestreaming-Ports in Ihrer Quark Publishing Platform Umgebung vornehmen. Als Standard bietet Quark Publishing Platform Server Services auf den Netzwerk-Ports 61400 bis 61407, jedoch lassen sich alle diese Ports einstellen.

Ändern der Standard-Ports

Standardmäßig ist für Quark Publishing Platform Clients nur Port 61400 erforderlich. Wenn Sie jedoch eigene RMI und CORBA Clients verwenden oder entwickeln, ändern Sie die Ports folgendermaßen:

- 1 Öffnen Sie die Datei „ServerApp.properties“; Sie finden sie im Ordner `[QPP Server]/conf`.
- 2 Um den Port für laufende, Java basierte RMI Clients (wie Quark Publishing Platform Script Manager) festzulegen, geben Sie einen Wert in `rmi.port=` ein. Remote Method Invocation (RMI) ist ein Standard-Remote-Prozeduraufruf, der es Java Objekten in Ihrem Netzwerk ermöglicht, remote ausgeführt zu werden.

- 3 Um den Port festzulegen, an den Dienste gebunden sind, geben Sie einen Wert in `rmi.serviceport=` ein.
- 4 Um den Port für die Handhabung von IIOP Requests festzulegen, geben Sie einen Wert in `namingservice.port=` ein. Internet Inter-Orb Protocol (IIOP) ist ein Message-Protokoll für den Umgang mit Objekten in einem TCP/IP-Netzwerk. QuarkXPress, QuarkCopyDesk und Quark Publishing Platform Client stellen Verbindungen über diesen Port her.
- 5 Um den Port für den Export von CORBA Objekten festzulegen, geben Sie einen Wert in `serverORB.port=` ein.
- 6 Um den Port für die gesamte Kommunikation von Quark Publishing Platform Client Anwendungen festzulegen, geben Sie einen Wert in `jms.openWirePort=` ein. In einer Quark Publishing Platform Umgebung verwendet der Java Messaging Service (JMS) das OpenWire® Protokoll.
- 7 Um festzulegen, auf welchem Port der Tomcat Server ausgeführt wird, geben Sie einen Wert in `webServer.port=` ein. Quark Publishing Platform Web Client verbindet sich über diesen Port.
- 8 Speichern Sie „ServerApp.properties“ und schließen Sie die Datei.

Festlegen des Serverports für Tomcat

Nach dem Ändern der Standardports müssen Sie auch den Port für das Ausführen des Tomcat Servers in einer separaten Datei festlegen:

- 1 Öffnen Sie die Datei „server.xml“, die sich im Ordner `[QPP Server]/conf` befindet.
- 2 Ändern Sie den Webserver-Port.
- 3 Speichern Sie „server.xml“ und schließen Sie die Datei.
- 4 Starten Sie den Server neu.

Ändern des Datei-Streaming-Ports

So ändern Sie den Datei-Streaming-Port:

- 1 Öffnen Sie den Ordner „conf“ im Quark Publishing Platform Server Ordner.
- 2 Öffnen Sie „SocketStreaming.properties“ in einer Textbearbeitungsanwendung.
- 3 Um den Port für Datei-Uploads und Datei-Downloads festzulegen, geben Sie einen Wert in `socketStreaming.port=` ein.
- 4 Speichern Sie „SocketStreaming.properties“ und schließen Sie die Datei.

Mehrere Netzwerkkarten

Standardmäßig ist für Quark Publishing Platform Clients nur Port 61400 erforderlich. Wenn Sie jedoch eigene RMI und CORBA Clients verwenden oder entwickeln und der Computer, auf dem Quark Publishing Platform Server läuft, über mehr als eine Netzwerkkarte verfügt, können Sie für Quark Publishing Platform Server eine Bindung an eine bestimmte Karte oder an alle IP-Adressen auf dem Computer vornehmen.

Bindung an eine bestimmte IP-Adresse

So stellen Sie eine Bindung an eine bestimmte IP-Adresse her:

- 1 Öffnen Sie den Ordner „conf“ im Quark Publishing Platform Server Ordner.
- 2 Öffnen Sie „ServerApp.properties“ in einer Textbearbeitungsanwendung.
- 3 Scrollen Sie bis zum Eintrag `server.machinename=`. Geben Sie die IP-Adresse der Netzwerkkarte ein, für die die Bindung erfolgen soll.
- 4 Scrollen Sie bis zum Eintrag `server.bindtoip=`. Geben Sie `true` ein, wenn Quark Publishing Platform Server nur an die IP-Adresse und an den Namen gebunden sein soll, die durch den Eintrag „server.machinename=“ festgelegt sind.
- 5 Speichern Sie „ServerApp.properties“ und schließen Sie die Datei.

Bearbeiten von "server.xml"

Auch die Datei "server.xml" muss folgendermaßen bearbeitet werden:

- 1 Öffnen Sie den Ordner "conf" im Quark Publishing Platform Server Ordner.
- 2 Öffnen Sie "server.xml" in einer Textbearbeitungsanwendung.
- 3 Geben Sie in das Tag "connector" die IP-Adresse ein. Ändern Sie beispielsweise `<Connector port=61400 in <Connector address = "<IP Address>" port="61400"`.
- 4 Speichern Sie "server.xml" und schließen Sie die Datei.

Binden aller IP-Adressen an einen einzigen Computer

So binden Sie alle IP-Adressen an einen bestimmten Computer:

- 1 Öffnen Sie den Ordner "conf" im Quark Publishing Platform Server Ordner.
 - 2 Öffnen Sie "ServerApp.properties" in einer Textbearbeitungsanwendung.
 - 3 Scrollen Sie bis zum Eintrag `server.machinename=`. Geben Sie `localhost` ein.
 - 4 Scrollen Sie bis zum Eintrag `server.bindtoip=`. Geben Sie `false` ein, damit Quark Publishing Platform Server eine Bindung mit allen IP-Adressen des Computers eingeht.
 - 5 Scrollen Sie bis zum Eintrag `server.additionalnames=`. Geben Sie die IP-Adresse der Netzwerkkarte ein, für die die Bindung erfolgen soll. Falls Sie über mehrere IP-Adressen verfügen, können Sie die Einträge durch ein Komma trennen (z. B. `server.additionalnames= 10.91.43.266,10.X.Y.Z`). Falls Ihr Computer nur eine Netzwerkkarte besitzt, lassen Sie dieses Feld frei.
- ➡ In dieser Liste werden nur Nicht-Standard-IPs verzeichnet. Geben Sie hier nicht die Standard-IP des Computers an, da Quark Publishing Platform Server die Standard-IP-Adresse automatisch verwendet. Das Hinzufügen einer Standard-IP wird nicht empfohlen.
- 6 Speichern Sie "ServerApp.properties" und schließen Sie die Datei.

Firewalls mit NAT

Wenn Sie RMI und CORBA Clients verwenden oder entwickeln und über das Internet durch eine Firewall mit Network Address Translation (NAT) Services Zugriff auf Quark Publishing Platform Server gewähren möchten, müssen Sie die öffentliche IP-Adresse festlegen, mit der die private IP-Adresse von Quark Publishing Platform Server verknüpft wird.

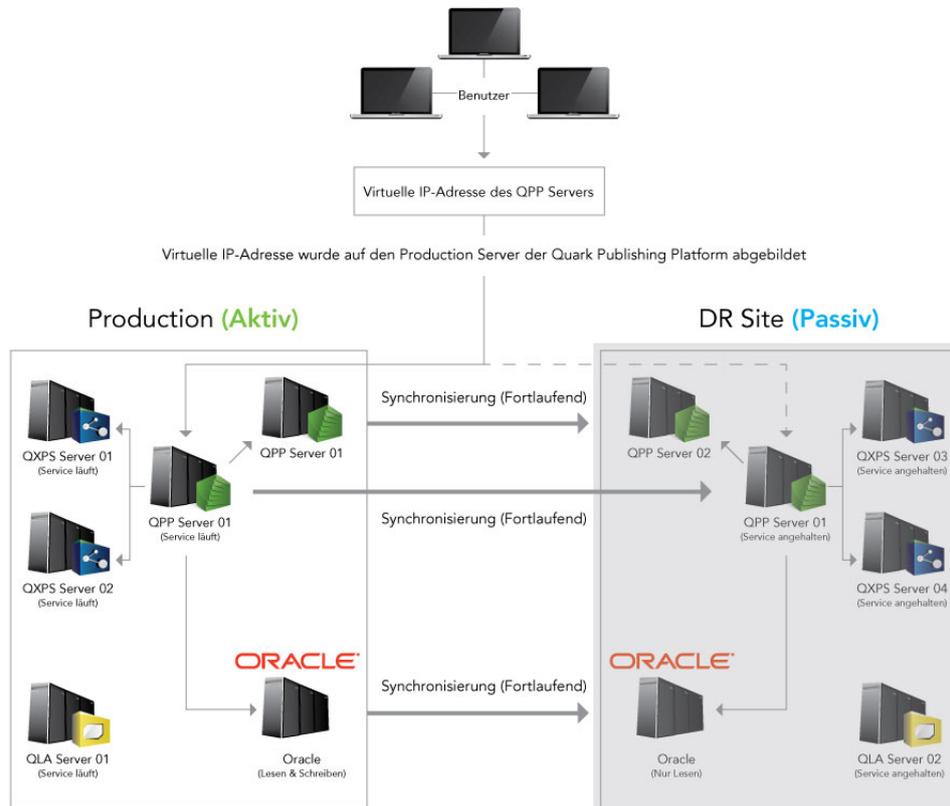
So legen Sie die öffentliche IP-Adresse fest:

- 1 Öffnen Sie den Ordner "conf" im Quark Publishing Platform Server Ordner.
- 2 Öffnen Sie "ServerApp.properties" in einer Textbearbeitungsanwendung.
- 3 Scrollen Sie bis zum Eintrag `server.additionalnames=`.
- 4 Geben Sie die öffentliche IP-Adresse ein, mit der die private IP-Adresse von Quark Publishing Platform Server verknüpft ist.
- 5 Speichern Sie "ServerApp.properties" und schließen Sie die Datei.

Failover-Einrichtung

Im Folgenden wird eine Möglichkeit beschrieben, wie Sie eine Quark Publishing Platform Installation für ein Failover einrichten.

Der Schlüssel zu einem solchen Setup liegt in der Verwendung einer virtuellen IP-Adresse für den Quark Publishing Platform Servercomputer.



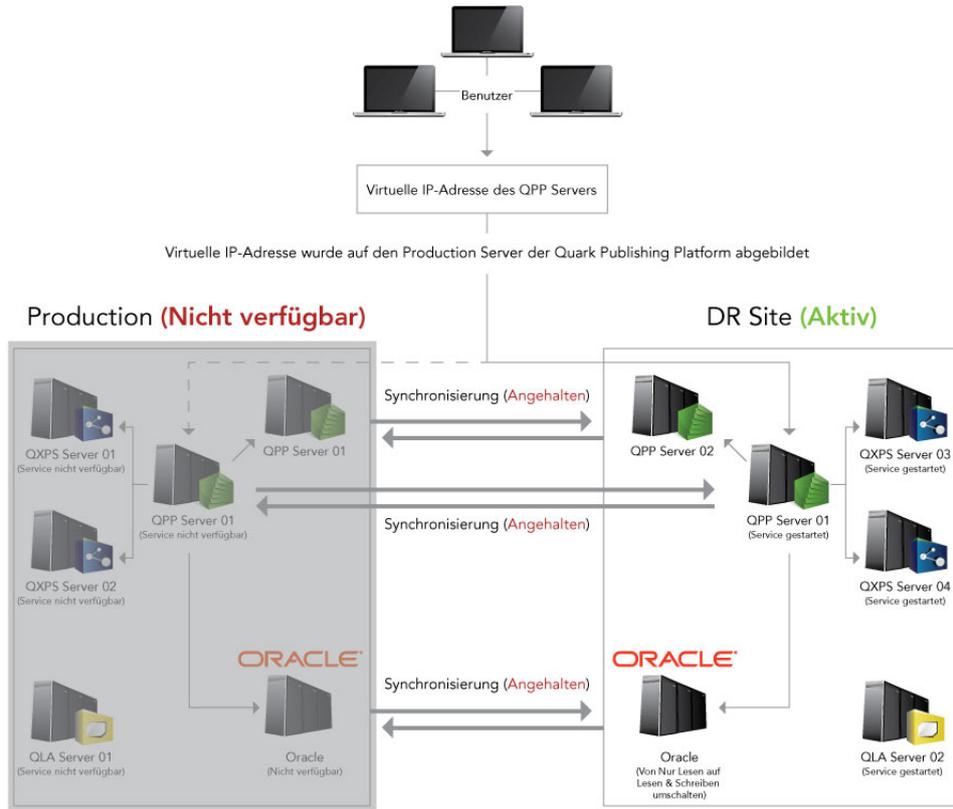
Failover-Einrichtung: Normalbetrieb

Während des normalen Betriebes läuft die Production-Site und die virtuelle IP-Adresse ist auf diesen Server gemappt. Die DR-Site (Disaster Recovery) ist im Schlafmodus (dormant), alle Quark Services sind angehalten. Folgende Objekte werden zwischen den Production-Servern und dem DR-Server synchronisiert:

- Quark Publishing Platform Repository-Ordner
- Quark Publishing Platform Datenbank
- Ordner `QPP Server/conf`
- Ordner `QPP Server/Index`

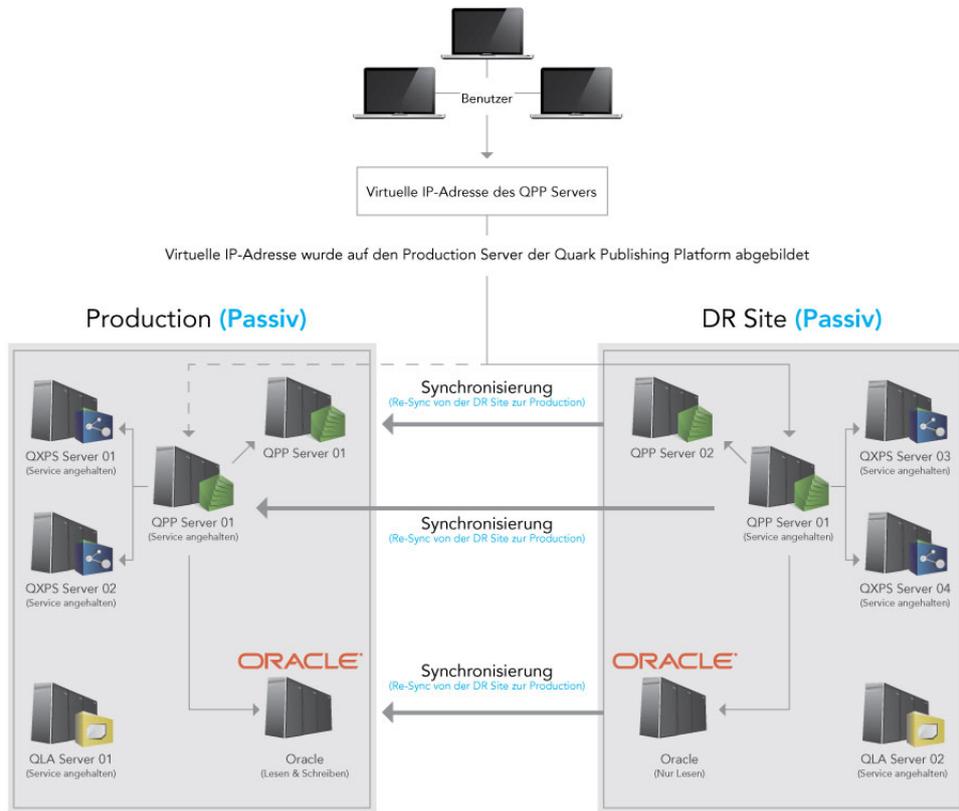
➔ Die Synchronisierung der Quark Publishing Platform File Server Ordner und der Quark Publishing Platform Oracle Datenbank muss gleichzeitig und in identischen Intervallen durchgeführt werden, um die Datenintegrität zu bewahren.

Wenn die Production-Site ausfällt, wird die Synchronisierung angehalten und die virtuelle IP-Adresse auf die DR-Site umgeleitet. Endanwender sollten diese Änderung nicht bemerken, da sie weiterhin dieselbe virtuelle IP-Adresse verwenden.



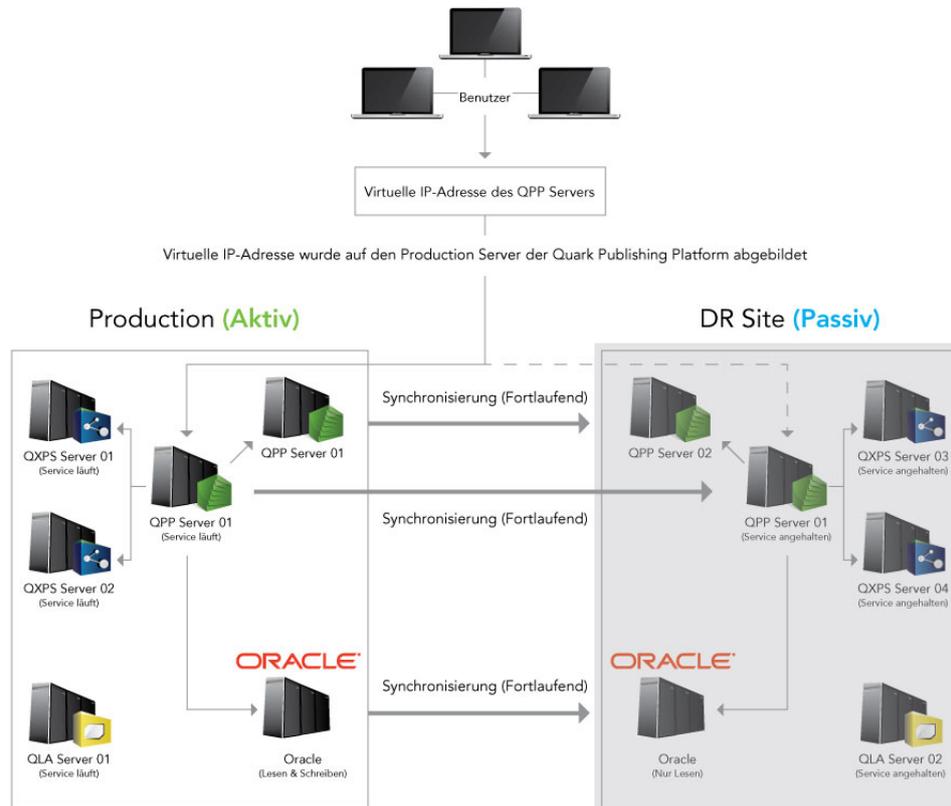
Failover-Einrichtung: Production Server fällt aus

Zu diesem Zeitpunkt sollte ein Administrator ein reguläres Herunterfahren der DR Site durchführen und dann die Synchronisierung von der DR Site zur Production Site neu starten.



Failover-Einrichtung: Synchronisieren von der DR Site zur Production Site

Nach Abschluss der Synchronisierung zwischen der DR Site und der Production Site wird mappen Sie die virtuelle IP-Adresse wieder auf die Production Site und kehren die Synchronisierung wieder so um, dass Änderungen vom Production Server an die DP Site übergeben werden. Starten Sie nun alle Komponenten von Quark Publishing Plattform.



Failover-Einrichtung: Normalbetrieb wiederhergestellt

Verschlüsseln eines Klartext-Passworts

In Quark Publishing Platform 10.1 und neuer können Sie Ihre sämtlichen in Klartext vorliegenden Passwörter verschlüsseln. Wenn Sie zum Beispiel einen SQL Server oder Oracle als Datenbank im Platform Server benutzen, wird Ihr Datenbank-Benutzerpasswort in Klartext in der Datei „database.properties“ abgelegt. Sie können diese Passwort verschlüsseln.

So verschlüsseln Sie Klartext-Passwörter:

- 1 Öffnen Sie die Eingabeaufforderung und navigieren zum Platform Server Installationsordner.
- 2 Starten Sie die Datei „encrypt.bat“ mit Ihrem Passwort (`encrypt.bat password`), um seinen verschlüsselten Wert abzurufen.
- 3 Kopieren Sie den erzeugten verschlüsselten Wert des Passworts und verwenden diesen in allen properties Dateien (zum Beispiel „database.properties“).
- 4 Öffnen Sie den Ordner „lib“ im Ordner Quark Publishing Platform Server.
- 5 Öffnen Sie „qpp-server-app-11.2.jar“ und navigieren Sie zu `\com\quark\qpp\app` innerhalb der jar-Datei.

- 6 Öffnen Sie „ServerStartupContext.xml“ in einer Textverarbeitungsanwendung und fügen Sie unter dem Knoten `securePlaceholderConfig` einen Eintrag für jede properties Datei hinzu, in der Sie den verschlüsselten Wert benutzt haben. Entfernen Sie - falls vorhanden - diesen Eintrag `placeholderConfig` aus dem Knoten.
 - 7 Sichern Sie die Datei, aktualisieren Sie die jar-Datei und starten Sie den Server erneut.
- ➔ Sie können einen beliebigen Algorithmus benutzen, der in der „ServerStartupContext.xml“ genannt ist, und Sie können den Schlüssel modifizieren, der zur Verschlüsselung des Passworts verwendet wurde (Standard ist QUARK).

Aktivieren des Zwangs-Logoffs während Inaktivität

Konfiguration des WebAdmin zur Aktivierung des Zwangs-Logoffs

Ein Zwangs-Logoff kann aktiviert werden durch Einschalten eines vordefinierten Servlet-Filters in der Datei „Web.xml“ im Ordner `[QPP Server]/webapps/admin`:

- `session-idle-time`: Legt die Inaktivitätszeit (in Sekunden) fest, nach der ein Benutzer von WebAdmin zwangsabgemeldet wird.
- `exclude-url-patterns`: Definiert URL-Patterns, die nicht als Benutzeraktivität betrachtet werden.
- `pre-expiry-mag-time`: Legt die Zeit (in Sekunden) fest, nach der eine Warnung oder Countdown-Nachricht angezeigt wird, bevor anstehende Inactivity Bases abgemeldet werden.

```
<filter> <filter-name>SessionTimeoutCookieFilter</filter-name>
<filter-class>com.quark.web.activity.servlet.SessionTimeoutCookieFilter</filter-class>
  <init-param> <param-name>session-idle-time</param-name>
  <param-value>60</param-value> </init-param> <init-param>
  <param-name>exclude-url-patterns</param-name>
  <param-value>/admin/keepAlive.qsp</param-value> </init-param> <init-param>
  <param-name>pre-expiry-mag-time</param-name> <param-value>40</param-value>
  </init-param> </filter> <filter-mapping>
<filter-name>SessionTimeoutCookieFilter</filter-name>
<url-pattern>/*</url-pattern> </filter-mapping>
```

Zwangs-Logoff kann auch in der Datei „Admin-Home.jsp“ QPP

Home/webapps/admin/jsp aktiviert werden:

```
<!-- Entfernen Sie den Kommentar für „Zwangs-Logoff bei Inaktivität“ -->
<script type="text/javascript" src="resources/js/SessionTimer.js"></script>
  <link type="text/css" href="resources/js/SessionTimer.css
  re;"stylesheet"></link>

function logoutDueToInactivity () { window.location.href = "logout.qsp";
}; <!-- Initialise inactivity monitor --%>
SessionTimer.init(logoutDueToInactivity);
```

Konfiguration des Workspace zur Aktivierung des Zwangs-Logoffs

Ein Zwangs-Logoff kann aktiviert werden durch Einschalten eines vordefinierten Servlet-Filters in der Datei „Web.xml“ im Ordner `[QPP Server]/webapps/workspace`:

- `session-idle-time`: Legt die Inaktivitätszeit (in Sekunden) fest, nach der ein Benutzer zwangsabgemeldet wird.

- `exclude-url-patterns`: Definiert URL-Patterns, die nicht als Benutzeraktivität betrachtet werden.
- `pre-expiry-mag-time`: Legt die Zeit (in Sekunden) fest, nach der eine Warnung oder Countdown-Nachricht angezeigt wird, bevor anstehende Inactivity Bases abgemeldet werden.

```
<filter> <filter-name>SessionTimeoutCookieFilter</filter-name>
<filter-class>com.quark.web.activity.servlet.SessionTimeoutCookieFilter</filter-class>
  <init-param> <param-name>session-idle-time</param-name>
  <param-value>300</param-value> </init-param> <init-param>
  <param-name>exclude-url-patterns</param-name>
  <param-value>/workspace/keepAlive.qsp/workspace/assetHeaderUpdate.qsp</param-value>
  </init-param> <init-param> <param-name>pre-expiry-mag-time</param-name>
  <param-value>60</param-value> </init-param> </filter> <filter-mapping>
<filter-name>SessionTimeoutCookieFilter</filter-name>
<url-pattern>/*</url-pattern> </filter-mapping>
```

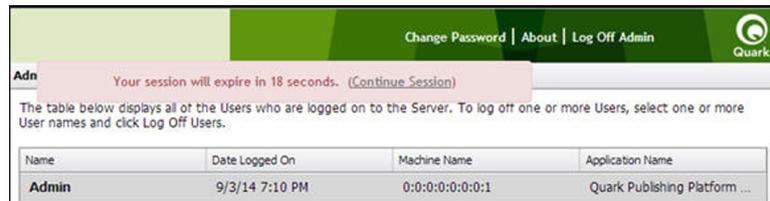
Zwangs-Logoff kann auch in der Datei „User-Home.jsp“ QPP

Home/webapps/workspace/jsp aktiviert werden:

```
<!-- Entfernen Sie den Kommentar für „Zwangs-Logoff bei Inaktivität“ -->
<script type="text/javascript" src="resources/js/SessionTimer.js"></script>
  <link type="text/css" href="resources/js/SessionTimer.css
  rel="stylesheet"></link>

<%-- Initialisieren Sie die Inaktivitätsüberwachung --%>
SessionTimer.init(UserHomeUI.logout);
```

Die Countdown-Nachricht zeigt die verbleibenden Sekunden. Der Anwender kann die Zwangsabmeldung abwenden, indem er auf **Session fortführen** klickt.



Zwangs-Logoff-Countdown

Konfiguration der Benachrichtigung

So konfigurieren Sie Quark Publishing Platform Server für die Anzeige von `Asset Workflow geändert`-Benachrichtigungen, wenn ein beliebiges aufgelistetes Attribut des entsprechenden Assets geändert wird:

- 1 Öffnen Sie die Datei „AssetWorkflowChangedMessageAttributes.xml“, die sich im Ordner `{Installationspfad}/Server/conf` befindet.
- 2 Legen Sie die Attribut-IDs oder -Namen fest, um eine `Asset Workflow geändert`-Benachrichtigung zu generieren, wenn ein beliebiges aufgelistetes Attribut des entsprechenden Assets geändert wird:

```
<!-- List of attribute ids or names to be considered for "ASSET WORKFLOW
CHANGED MESSAGE" The "ASSET WORKFLOW CHANGED MESSAGE" will be published if
there is any changes in the attributes of the mentioned asset/> <util:list
id="assetWorkflowAttributes"> <value>name</value> <value>Status</value>
<value>Workflow</value> <value>Routed to</value> </util:list> </beans>
```

Quark Publishing Platform Web Client: Manuelle Konfiguration

In den folgenden Abschnitte erfahren Sie, wie Sie die erweiterten Einstellungen des Quark Publishing Platform Web Client konfigurieren.

Konfiguration – Übersicht

Die Konfiguration der Arbeitsumgebung ist in zwei Kategorien eingeteilt:

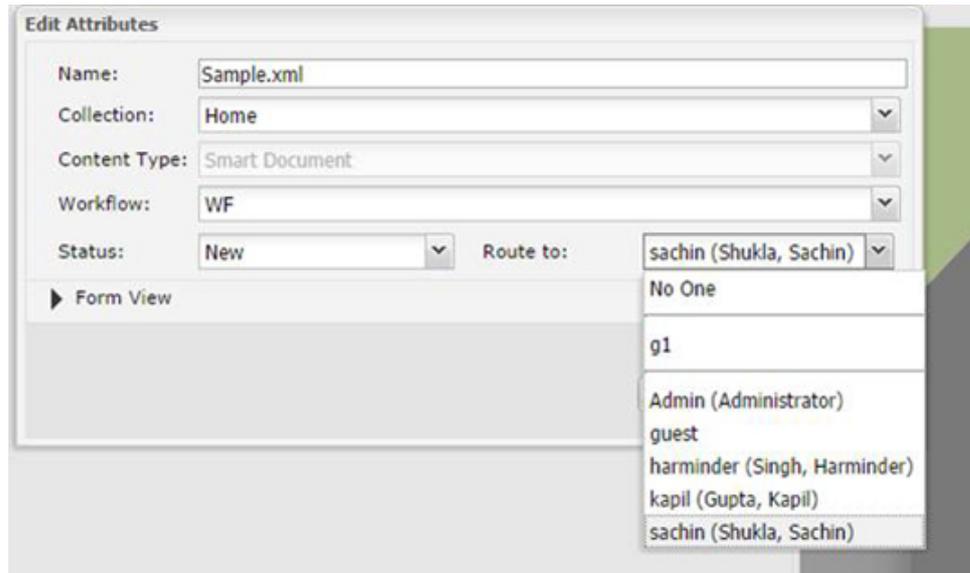
- Einstellungen auf Anwendungsebene
- Einstellungen auf Dokumentebene

Die Einstellungen werden in der Konfigurationsdatei „Workspace-Config.xml“ im Ordner `[QPP Server]/webapps/workspace/WEB-INF/classes` vorgenommen.

Einstellungen der Anwendungsebene

Die Attribut-Elemente von `ApplicationSettings` definieren verschiedene Einstellungen der Anwendungsebenen:

- `userNameFormatting`: Legt das Format des **Benutzernamens** fest, wie er in allen Dialogen angezeigt werden soll:
 - 1 0, um den Loginnamen `<benutzername>` anzuzeigen
 - 2 1, um `<Vorname> <Nachname>` anzuzeigen
 - 3 2, um `<Nachname> <Vorname>` anzuzeigen



- `layoutPreview`: Definiert das Erscheinungsbild des Vorschau Fensters für Assets, die zum Inhaltstyp QuarkXPress gehören:
 - 1 `True`, um die Vorschau in einem separaten Browser-Fenster anzuzeigen
 - 2 `False`, um die Vorschau in einem Javascript-Fenster anstelle eines Browsers anzuzeigen.
- `viewrevision_expandAll`: Legt fest, ob alle Versionskommentare im Dialog Versionen anzeigen erweitert werden müssen.
- `supported_picture_extension`: Definiert die möglichen Dateitypen, die für den Import eines Bildes/Textes in den Artikel-/Projekt-Editoren erlaubt sind.
- `picture_editing_extension`: Definiert die möglichen Dateiformate, die für die Bearbeitung von Bildern in den Artikel-/Projekt-Editoren erlaubt sind.
- `defaultPreviewScale`: Legt die Standard-Zoomeinstellung für die Vorschau fest, die beim Bearbeiten von Artikeln/Projekten und Betrachten von Assets (Live-Voransicht) verwendet werden soll. Gültige Werte für diesen Parameter reichen von 0.1 bis 5.
- `ajaxTimeout`: Legt die Dauer für alle Hintergrundanfragen aus dem Browser fest, nach deren Ablauf die Anfragen als Timeout gekennzeichnet werden. Dieser Wert wird in Millisekunden angegeben.
- `showFormView`: Gibt an, ob der Bereich Formularansicht im **Einchecken** -Dialog angezeigt werden soll. Geben Sie hier `true` ein, um den Bereich Formularansicht einzublenden.
- `topBannerJspPath`: Spezifiziert den Pfad zu der .jsp-Datei, die eingebunden werden muss, um das obere Banner in der Zuweisungsseite anzuzeigen.
- `logoFilePath`: Gibt den Pfad zum Logo-Bild an, das im oberen Banner in der Zuweisungsseite angezeigt werden soll.
- `enabledPublishingChannels`: Definiert die Liste von Publishing Channels, die bei der Auswahl eines Assets berücksichtigt werden sollen.

- `enableDeliveryChannels`: Definiert, ob Delivery Channels aktiviert werden müssen oder nicht.
- `enabledDeliveryChannels`: Definiert die Liste von Delivery Channels, die bei der Auswahl eines Assets berücksichtigt werden sollen. Dieses wird aktiv, wenn die Option `enableDeliveryChannels` auf `true` gesetzt wurde.
- `allowPublishedRenditionDownload`: Definiert, ob die angezeigte Download-Ausgabe für die unterschiedlichen Voransichten erlaubt sein soll, wenn das für das ausgewählte Asset definierte Publishing verwendet wird.

```
<ApplicationSettings> <Add key="viewrevision_expandAll" value="false"/>
<Add key="supported_picture_extension" value="bmp;jpg;jpeg;tif;tiff;gif;/>
<Add key="picture_editing_extension" value="jpg;jpeg;tif;tiff;eps"/> <Add
key="userNameFormatting" value="0"/> <Add key="layoutPreview" value="true"/>
<Add key="defaultPreviewScale" value="0.8"/> <Add key="ajaxTimeout"
value="300000"/> <Add key="showFormView" value="false"/> <Add
key="topBannerJspPath" value="Header.jsp"/> <Add key="logoFilePath"
value="images/login/topbanner-innerpage-left.png"/> <Add
key="enabledPublishingChannels"
value="qxpPdf;qxpEpub;qxpAppStudio;qxpAppStudioPackage;busDocPdf;busDocHtml;busDocQxp/>
<Add key="enableDeliveryChannels" value="false"/> <Add
key="enabledDeliveryChannels"
value="checkInToSharepoint;checkInToFileNet;checkInToDocumentum;sendEmail;sendToFTPServer"/>
<Add key="allowPublishedRenditionDownload" value="true"/>
</ApplicationSettings>
```

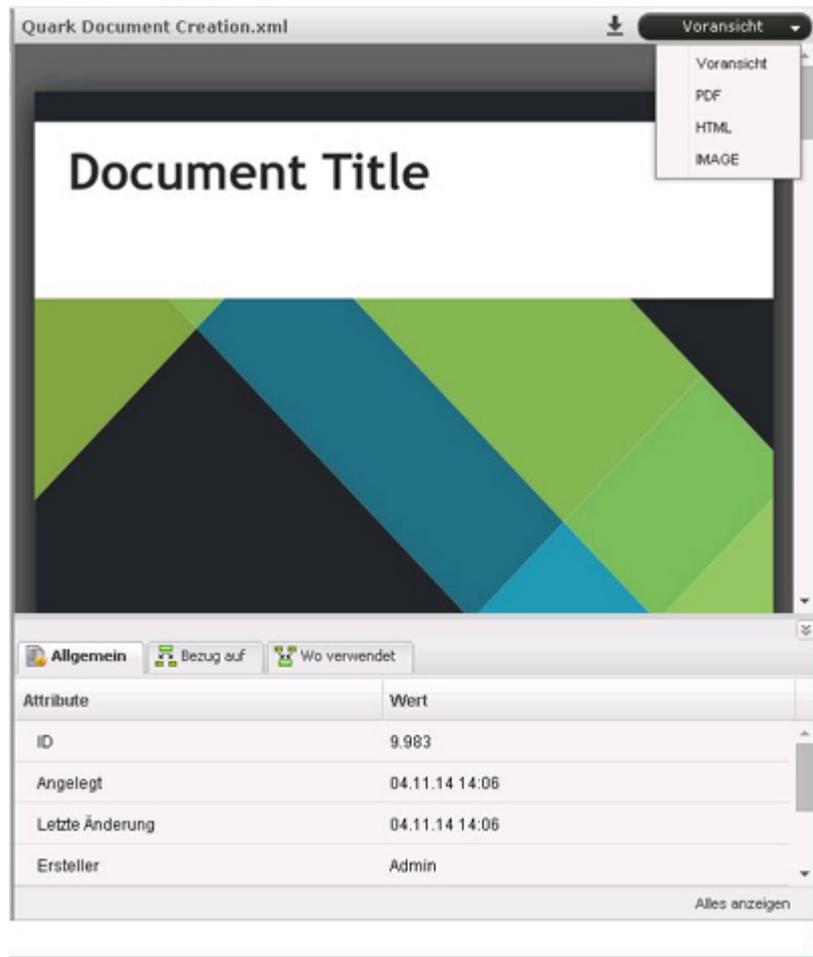
Multi-Channel Voransicht

Für jeden Inhaltstyp geben Sie die Publishing Channel an, die für den Anwender im Register **Voransicht** der Zuweisungsseite verfügbar sein sollen. Publishing Channels für die Vorschau werden im Element `<PreviewSettings>` konfiguriert:

- `displayName`: Legt den im User Interface angezeigten Channel-Namen fest. (optional) Wenn er nicht vergeben wurde, wird `outputFormat` verwendet.
- `Id`: Legt die Publishing Channel ID wie im Platform Server definiert fest.
- `ContentType`: Legt den Inhaltstyp des Assets fest.
- `ApplyToChildContentTypes`: Legt fest, ob untergeordnete Inhaltstypen eingeschlossen werden sollen.
- `outputFormat` : Unterstützte Werte:
 - 1 IMAGE_ARCHIVE: In Bildarchiv für die veröffentlichte Ausgabe, das in einer Webseite verarbeitet wird.
 - 2 HTML_ARCHIVE: Ein HTML-Archiv für die veröffentlichte Ausgabe, das unverändert berechnet wird, indem es auf den Dateinamen zeigt, der im HTML-Archiv vermerkt ist.
 - 3 PDF_ARCHIVE: Die veröffentlichte Ausgabe-PDF, die unverändert berechnet wird.
- `downloadChannel`: (optional) wird genutzt, falls ein anderer Channel für den Download einer ausgewählten Channel-Voransicht aufgerufen werden muss.

```
<PreviewSettings> <ChannelMappings> <ChannelMapping contentType ="Business
Document" applyToChildContentTypes ="true"> <Channels> <Channel
id="busDocPdf" outputFormat="PDF_ARCHIVE" displayName="PDF"/> <Channel
id="busDocHtml" outputFormat="HTML_ARCHIVE" displayName="HTML"/> <Channel
id="busDocJpeg" outputFormat="IMAGE_ARCHIVE" displayName="IMAGE"/>
</Channels> </ChannelMapping> <ChannelMapping contentType="Smart Content"
applyToChildContentTypes="true"> <Channels> <Channel id="smartDocPdf"
```

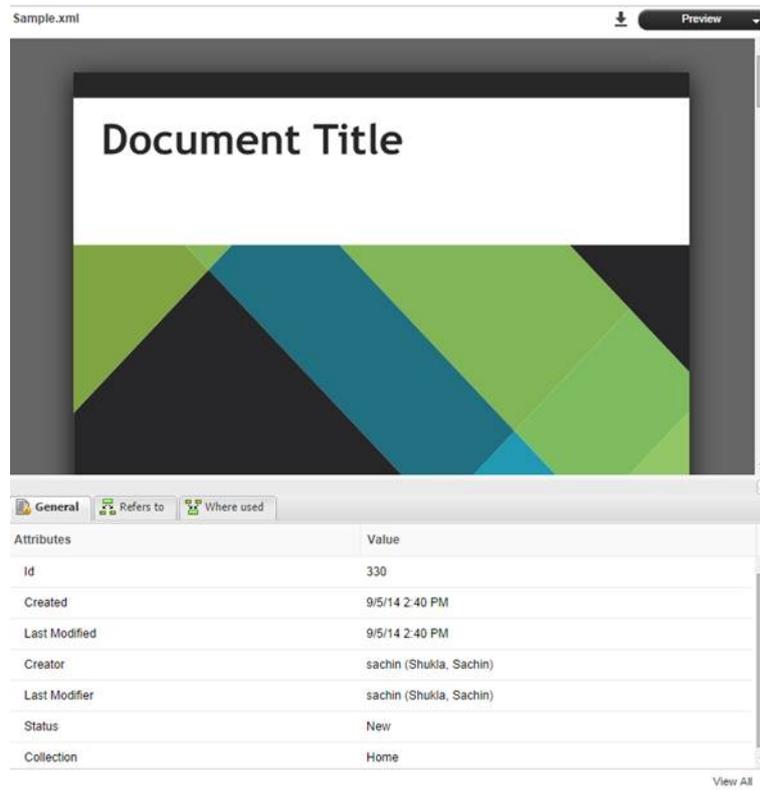
```
outputFormat="PDF_ARCHIVE" displayName="PDF"/> <Channel id="smartDocHtml"
outputFormat="HTML_ARCHIVE" displayName="HTML"/> <Channel id="smartDocJpeg"
  outputFormat="IMAGE_ARCHIVE" displayName="IMAGE"/> </Channels>
</ChannelMapping> </ChannelMappings> </PreviewSettings>
```



Attribute für den Bereich Allgemein

Die Attribute für das Register **Allgemein** werden mithilfe des Elements `<PreviewAttributes>` konfiguriert. Die Werte bezeichnen die Liste der Attribute, die im Register **Allgemein** des ausgewählten Assets angezeigt wird.

```
<PreviewAttributes> <PreviewAttribute>Id</PreviewAttribute>
<PreviewAttribute>Created</PreviewAttribute> <PreviewAttribute>Last
modified</PreviewAttribute> <PreviewAttribute>Creator</PreviewAttribute>
<PreviewAttribute>Last modifier</PreviewAttribute>
<PreviewAttribute>Status</PreviewAttribute>
<PreviewAttribute>Collection</PreviewAttribute> ..... </PreviewAttributes>
```



Rollenbasierte Konfiguration der Werkzeugleiste

In diesem Abschnitt erfahren Sie, wie die Werkzeugleistenobjekte für eine bestimmte Rolle definiert werden können. Die rollenbasierte Konfiguration von Werkzeugleistenobjekten erfolgt mithilfe des Elements `<ToolbarConfig>`.

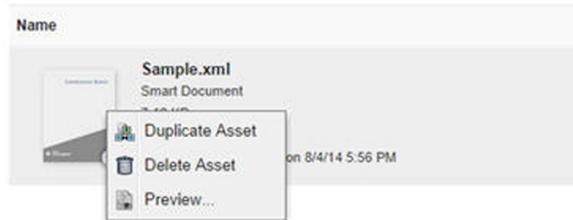
- **Role name:** Geben Sie den Namen der Rolle ein, deren Konfiguration festgelegt werden soll.
- **Item ID:** Geben Sie die ID des Objekts ein, dessen Anzeige konfiguriert werden soll.
- **showInToolbar:** Um das Menüobjekt in der Werkzeugleiste zu verbergen, setzen Sie dieses Attribut auf „false“.
- **showInContext:** Um das Menüobjekt im Kontextmenü zu verbergen, setzen Sie dieses Attribut auf „false“.
- **showInNewMenu:** Um das Menüobjekt im Menü „Neues Asset“ zu verbergen, setzen Sie dieses Attribut auf „false“.
- **showInTemplateMenu:** Wenn das Menüobjekt nicht in „Neues Asset aus Vorlage“ gezeigt werden soll, setzen Sie dieses Attribut auf „false“.

Die XML-Struktur der Werkzeugleisten-Konfiguration sieht folgendermaßen aus:

```
<Role name = "[ROLE NAME]"> <Item id = "[ITEM ID]" showInToolbar = "[true/false (defaults to true)]" showInContextMenu = "[true/false (defaults to true)]" showInNewMenu = "[true/false (defaults to true)]" showInTemplateMenu = "[true/false (defaults to true)]"/> </Role>
```

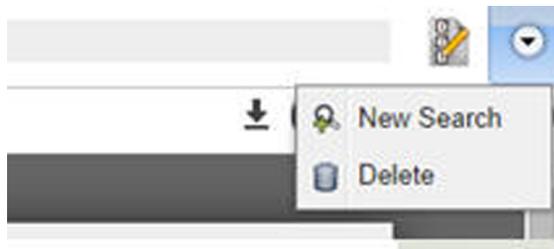
```
<ToolbarConfig> <Role name ="Guest"> <Item id ="new_qcd_menu_item"/> <Item id ="new-search-btn"/> <Item id ="duplicate-asset" showInToolbar="false"/> <Item id ="delete-asset"/> <Item id ="show-edit-attributes" showInContextMenu="false"/> <Item id ="show-Asset_preview" showInToolbar="false"/> ..... </Role> </ToolbarConfig>
```

Das Kontextmenü für die Rolle „Gast“ sieht etwa so aus:



Kontextmenü für Gast

Die Werkzeugleistenobjekte für die Rolle „Gast“ sehen etwa so aus:



Werkzeugleistenobjekte für Gast

Es folgt die Liste der Objekt-IDs:

- `checkin`: Einchecken des Assets
- `show-check-out`: Auschecken des Assets
- `cancel-checkout`: Auschecken des Artikels abbrechen
- `get-asset`: Asset laden
- `new-search-btn`: Neue Suche (nicht verfügbar als Kontextmenüobjekt)
- `show-Asset-Preview`: Asset-Vorschau
- `show-edit-attributes`: Asset-Attribute bearbeiten
- `attachment-info`: Zuordnungsinformationen zeigen
- `view-revisions`: Asset-Versionen zeigen
- `duplicate-asse` : Asset duplizieren
- `delete-asset`: Asset löschen
- `reindex-asset`: Asset erneut indexieren (in der Werkzeugleiste nicht standardmäßig sichtbar)
- `publish-menu-btn`: Asset veröffentlichen
- `deliver-menu-btn`: Asset übertragen
- `Wiederherstellen`: Assets wiederherstellen

- `archive-asset`: Assets archivieren
- `open-collection`: Zum Asset gehörende Sammlung öffnen
- `refresh-datadoc`: Datadoc aktualisieren
- `unlink-datadoc`: Datadoc trennen
- `open-readonly`: Asset schreibgeschützt öffnen
- `new_qcd_menu_item`: Neuen QuarkCopyDesk Artikel erstellen
- `new_qxp_menu_item`: Neues QuarkXpress Projekt aus Vorlage erstellen

➔ Die Attribute `showInToolBar` und `showInContextMenu` kommen nur zum Tragen bei Menüobjekten, die in der Werkzeugleiste und den Kontextmenüs angezeigt werden. Für Schaltflächen, die in der Datei `ui-extension.xml` konfiguriert werden, ist `showInToolBar` nicht zulässig, da die Konfiguration dazu dient, eine Werkzeugschaltfläche an sich hinzuzufügen. Zusätzlich kann die Schaltfläche einem Kontextmenü hinzugefügt werden, folglich ist nur `showInContextMenu` für `ui-extension.xml`-Schaltflächen zulässig. Die Attribute `showInNewMenu` und `showInTemplateMenu` sind nur zulässig für Menüobjekte im Dropdown-Menü **Neues Asset** und nicht beim Setup einer rollenbasierten Konfiguration für `ui-extension.xml`-Menüs in diesem Abschnitt.

➔ Die übrigen konfigurierbaren Elemente, die in „Workspace-config.xml“ definiert sind, dienen zur Integration mit Quark Author Web Edition. Weitere Informationen finden Sie in der „Anleitung zur Konfiguration von Quark Author Web Edition“.

Einschränken des Webclient-/Admin-Zugangs

Um die Webanwendung so zu beschränken, dass auf den Arbeitsbereich und auf Admin-Seiten nur aus Ihrem eigenen Netzwerk zugegriffen werden kann, verwenden Sie ein Tomcat Valve für die Zugangsbeschränkung auf der Basis eines Bereichs von IP-Adressen.

- 1 Öffnen Sie die Datei „server.xml“, die sich im Ordner `{Installationspfad}/conf` befindet.
- 2 Bearbeiten Sie die Kontextelemente, die den Admin- und Workspace-Anwendungen entsprechen, um ein `RemoteAddrValve` (ein Ventil für Remote-Adressen) einzustellen. Das Ventil muss so konfiguriert werden, dass es nur Intranet-Datenverkehr erlaubt. Die endgültige Config-Datei wird etwa folgendermaßen aussehen:

```
<Context path="/qxpsm" docBase="qxpsm"> <Manager pathname="" /> </Context>
<Context path="/workspace" docBase="workspace"> <Manager pathname="" />
<Valve className="org.apache.catalina.valves.RemoteAddrValve" allow="
10\.91\.\.\.*" /> </Context> <Context
path="/webservices" docBase="webservices"> <Manager pathname="" /> </Context>
<Context path="/messaging" docBase="messaging"> <Manager pathname="" />
</Context> <Context path="/admin" docBase="admin"> <Manager pathname="" />
<Valve className="org.apache.catalina.valves.RemoteAddrValve" allow="
10\.91\.\.\.*" /> </Context> <Context path="/rest" docBase="rest"> <Manager
pathname="" /> </Context>
```

- ➔ Der Wert des Attributs `allow` ist ein regulärer Ausdruck (RegEx) für passende IP-Adressen. Im oben angeführten Beispiel wird vorausgesetzt, dass die Intranet-IPs im Bereich 10.91.x.x liegen. Sie müssen diesen Wert basierend auf Ihre Netzwerk-IP-Bereiche aktualisieren.

Quark Publishing Platform Clients – Manuelle Konfiguration

Sie können eine Vielzahl von Konfigurationsoptionen für Quark Publishing Platform Clients und Platform XTensions nach der Installation ändern. In den nachfolgenden Abschnitten finden Sie hierzu weitere Informationen.

Erstellen und Verwalten der Log-Datei (nur Mac OS X)

Die Einstellung zum Erstellen und Verwalten der Log-Datei ist in der Datei `com.quark.qpp.client.Quark.QPP.Client.config.plist` vermerkt. Sie finden diese Datei hier: `~/Library/Preferences/Quark\QPP\{version}`.

Den Speicherort der Protokolldatei ändern Sie, indem Sie den neuen Speicherort mit Attribut `LogFileName` definieren. Der Standardspeicherort für die Log-Datei lautet `~/Library/Logs/Quark Publishing Platform Client.Log`

Sie ändern die Größe der Protokolldatei, indem Sie die neue Größe mit dem Attribut `LogFileSize` definieren.

Benutzer können die Fehler-Protokollierung mit dem Attribut `LogException` aktivieren und deaktivieren.

Folgende Werte werden unterstützt:

- (NO) Deaktiviert die Fehlerprotokollierung
- (YES) Aktiviert die Fehlerprotokollierung

Erstellen und Verwalten der Log-Datei (nur Windows)

Die Einstellung zum Erstellen und Verwalten der Log-Datei ist in der Datei `Quark.QPP.Client.config` vermerkt. Diese Datei befindet sich an dem Speicherort, an dem die Anwendung installiert ist. Benutzer können die Protokollierung mit dem Attribut `QPSLogIsEnabled` aktivieren und deaktivieren.

Folgende Werte werden unterstützt:

- 0 (NEIN)
- 1 (JA)

Den Speicherort der Protokolldatei ändern Sie, indem Sie den neuen Speicherort mit Attribut `QPSLogFile` definieren. Der Standard-Speicherort der Log-Datei:

- für den Quark Publishing Platform Client – `AppData\Quark Inc\Quark Publishing Platform\{version}\Logs`
- für zum Platform Client gehörende Operationen in QuarkXPress – `AppData\Roaming\Quark Software Inc\QuarkXPress {version}\{version}\Logs`

Die Größe der Protokolldatei ändern Sie, indem Sie die neue Größe mit dem Attribut `QPSLogFileSize` definieren. Die Größe wird in `Bytes` angegeben. Der Standardwert der Out-of-the-Box-Konfiguration beträgt 5242880 Byte.

Mit dem Attribut `QPSLogFileMode` geben Sie an, wie die Log-Datei geschrieben werden soll.

Erlaubte Werte sind:

- 1 (NeuErstellen). Gibt an, dass das Betriebssystem eine neue Datei erstellen soll. Falls die Datei bereits vorhanden ist, wird die Meldung `System.IO.IOException` ausgegeben.
- 2 (Erstellen). Gibt an, dass das Betriebssystem eine neue Datei erstellen soll. Falls die Datei bereits existiert, wird sie überschrieben.
- 3 (Öffnen). Gibt an, dass das Betriebssystem eine vorhandene Datei öffnen soll. Falls die Datei noch nicht vorhanden ist, wird die Meldung `System.IO.FileNotFoundException` ausgegeben.
- 4 (ÖffnenOderErstellen). Gibt an, dass das Betriebssystem eine Datei öffnen soll, wenn sie existiert; andernfalls soll eine neue Datei erstellt werden.
- 5 (Beschneiden). Gibt an, dass das Betriebssystem eine vorhandene Datei öffnen soll. Wenn sie geöffnet sind, sollen die Dateien so beschnitten werden, sodass ihre Größe 0 Bytes beträgt.
- 6 (Anfügen). Öffnet die Datei, falls sie existiert und durchsucht sie bis zum Dateiende oder erstellt eine neue Datei.

➔ DateiModus - ÖffnenOderErstellen (4) ermöglicht das Öffnen einer Datei mit folgenden Berechtigungen: `DateiZugriff.Lesen`, `DateiZugriff.Schreiben`, `DateiZugriff.LesenSchreiben`, `DateiZugriff.Anfügen`. Der Versuch, eine Datei zu lesen, die mit DateiModus - Beschneiden (5) geöffnet wurde, führt zu einer Fehlermeldung.

Das Attribut `QPSLogFileShare` bietet die Möglichkeit, die von anderen Prozessen erstellte Log-Datei einzuschränken oder gemeinsam zu nutzen. Der Standardwert ist 1 („Lesen“)

Erlaubte Werte sind:

- 0 (Ohne). Verweigert die gemeinsame Nutzung der aktuellen Datei.
- 1 (Lesen). Der Standardwert. Erlaubt das Öffnen der Datei zum Lesen.
- 2 (Schreiben). Erlaubt das Öffnen der Datei zum Schreiben.

- 3 (LesenSchreiben). Erlaubt das Öffnen der Datei zum Lesen oder Schreiben.
- 4 (Löschen). Erlaubt das anschließende Löschen einer Datei.
- 16 (Vererbbar). Sorgt dafür, dass der File-Handle durch Kindprozesse vererbbar ist.

Zum Festlegen der maximalen Anzahl von Backup-Log-Dateien verwenden Sie das Attribut `QPSTLogFileBackupNumber`.

Erstellen und verwalten der Log-Datei (Quark XML Author für Platform)

Die folgende Konfiguration in der Datei `Quark.CMSAdapters.config` erlaubt dem Systemintegrator oder Anwender den Pfad zur Log-Datei einzustellen: `<!-- Definiert den Pfad der Log-Datei.--> <add key="LogFilePath" value="%APPDATA%\Quark\XML Author\Logs\CMS Adapter Log.txt"/>`

Die Einstellung zum Erstellen und Verwalten der LogDatei ist in der Datei `Quark.QPP.Client.config` vermerkt. Diese Datei befindet sich im Anwendungsordner. Benutzer können die Protokollierung mit dem Attribut `QPSTLogIsEnabled` aktivieren und deaktivieren.

Folgende Werte werden unterstützt:

- 0 (NEIN)
- 1 (JA)

Den Speicherort der Log-Datei ändern Sie, indem Sie den neuen Speicherort im Attribut `QPSTLogFile` definieren. Der Speicherort der Protokolldatei ist `AppData\Quark\XML Author\Logs\CMS Adapter Log.txt`

Die Größe der Protokolldatei ändern Sie, indem Sie die neue Größe mit dem Attribut `QPSTLogFileSize` definieren. Die Größe ist in Bytes angegeben. Der Standardwert der Out-of-the-Box-Konfiguration beträgt 5242880 Byte.

Mit dem Attribut `QPSTLogFileMode` geben Sie an, wie die Protokolldatei geschrieben werden soll.

Erlaubte Werte sind:

- 1 (NeuErstellen). Gibt an, dass das Betriebssystem eine neue Datei erstellen soll. Falls die Datei bereits vorhanden ist, wird die Meldung `System.IO.IOException` ausgegeben.
- 2 (Erstellen). Gibt an, dass das Betriebssystem eine neue Datei erstellen soll. Falls die Datei bereits existiert, wird sie überschrieben.
- 3 (Öffnen). Gibt an, dass das Betriebssystem eine vorhandene Datei öffnen soll. Falls die Datei noch nicht vorhanden ist, wird die Meldung `System.IO.FileNotFoundException` ausgegeben.
- 4 (ÖffnenOderErstellen). Gibt an, dass das Betriebssystem eine Datei öffnen soll, wenn sie existiert; andernfalls soll eine neue Datei erstellt werden.

- 5 (Beschneiden). Gibt an, dass das Betriebssystem eine vorhandene Datei öffnen soll. Wenn sie geöffnet sind, sollen die Dateien so beschnitten werden, sodass ihre Größe 0 Bytes beträgt.
- 6 (Anfügen). Öffnet die Datei, falls sie existiert und durchsucht sie bis zum Dateiende oder erstellt eine neue Datei.

➔ DateiModus - ÖffnenOderErstellen (4) ermöglicht das Öffnen einer Datei mit folgenden Berechtigungen: DateiZugriff.Lesen, DateiZugriff.Schreiben, DateiZugriff.LesenSchreiben, DateiZugriff.Anfügen. Der Versuch, eine Datei zu lesen, die mit DateiModus - Beschneiden (5) geöffnet wurde, führt zu einer Fehlermeldung.

Das Attribut `QPSLogFileShare` bietet die Möglichkeit, die von anderen Prozessen erstellte Protokolldatei einzuschränken oder gemeinsam zu nutzen. Der Standardwert ist 1 ("Lesen")

Erlaubte Werte sind:

- 0 (Ohne). Verweigert die gemeinsame Nutzung der aktuellen Datei.
- 1 (Lesen). Der Standardwert. Erlaubt das Öffnen der Datei zum Lesen.
- 2 (Schreiben). Erlaubt das Öffnen der Datei zum Schreiben.
- 3 (LesenSchreiben). Erlaubt das Öffnen der Datei zum Lesen oder Schreiben.
- 4 (Löschen). Erlaubt das anschließende Löschen einer Datei.
- 16 (Vererbbar). Sorgt dafür, dass der File-Handle durch Kindprozesse vererbbar ist.

Unterdrücken der Warnmeldung für Bedienungshilfen

Standardmäßig wird beim Start in Quark Publishing Platform Client für Mac OS X unter Umständen die Warnmeldung angezeigt „Bedienungshilfen (Accessibility-Services) sind nicht aktiviert“.

- 1 Control-klicken Sie auf das Anwendungssymbol von Quark Publishing Platform Client und wählen Sie **Paketinhalt zeigen**. Es erscheint ein neues Fenster.
- 2 Öffnen Sie die Datei „Info.plist“ in einem Texteditor.
- 3 Suchen Sie nach folgenden Zeilen:

```
<key>QPPDisableAccessibilityWarning</key> <string>0</string>
```
- 4 Ändern Sie die Null im Element `<string>` in eine 1 um.

Anzeigen von Versionskommentaren

Wenn Sie die Dialogbox **Versionen ansehen** aufrufen, müssen Sie in der Standardeinstellung jede Version erweitert darstellen, um den jeweiligen Versionskommentar sehen zu können. So sorgen Sie dafür, dass Versionskommentare unter Mac OS X automatisch angezeigt werden:

- 1 Navigieren Sie zu `~/Library/Preferences/Quark/QPP/[QPP Framework-Version]`.
- 2 Öffnen Sie die Datei „`com.quark.qpp.client.[Anwendungsname].config.plist`“ in einem Texteditor.
- 3 Suchen Sie nach folgenden Zeilen:

```
<key>ExpandAllRevisionComments</key> <false/>
```
- 4 Ändern Sie `<false/>` in `<true/>`.

Unter Windows öffnen Sie die Datei „`[Anwendungsname].exe.config`“, suchen folgende Zeile und ändern die Null in eine 1 um.

```
<add key = "ExpandAllRevisionComments" value="0"/>
```

Anzeigen von Vor- und Nachnamen

Sie können Quark Publishing Platform auf drei verschiedene Arten so konfigurieren, dass Benutzernamen angezeigt werden:

- `[Benutzername]`
- `[Benutzername] ([Vorname] [Nachname])`
- `[Benutzername] ([Nachname], [Vorname])`

So ändern Sie diese Einstellung:

- Für **Quark Publishing Platform Client für Mac OS X** öffnen Sie die Datei `~/Library/Preferences/Quark/QPP/[QPP Framework Version]/com.quark.qpp.client.{Anwendung}.config.plist`, suchen nach dem folgenden Text und stellen den Wert `<string>` auf 0 für `[Benutzername]`, 1 für `[Benutzername] ([Vorname] [Nachname])` oder auf 2 für `[Benutzername] ([Nachname], [Vorname])` ein:

```
<key>UserNameFormattingStyle</key> <string>0</string>
```
- Bei **Quark Publishing Platform Client für Windows** öffnen Sie die Datei „Quark Publishing Platform Client.exe.config“ (im Quark Publishing Platform Client Anwendungsordner), suchen nach dem folgenden Text und stellen das Attribut `value` auf `DEFAULT` für `[Benutzername]`, auf `FIRSTNAME_LASTNAME` für `[Benutzername] ([Vorname] [Nachname])` oder auf `LASTNAME_FIRSTNAME` für `[Benutzername] ([Nachname], [Vorname])` ein.

```
<add key="UserNameFormattingStyle" value="DEFAULT"/>
```
- Für **QuarkXPress für Mac OS X** öffnen Sie die Datei `~/Library/Preferences/Quark/QPP/[QPP Framework Version]/com.quark.qpp.client.{Anwendung}.config.plist`, suchen nach dem folgenden Text und stellen den Wert `<string>` auf 0 für `[Benutzername]`, 1 für `[Benutzername] ([Vorname] [Nachname])` oder auf 2 für `[Benutzername] ([Nachname], [Vorname])` ein:

```
<key>UserNameFormattingStyle</key> <string>0</string>
```

- Für **QuarkCopyDesk für Mac OS X** öffnen Sie die Datei
`~/Library/Preferences/Quark/QPP/[QPP Framework Version]/com.quark.qpp.client.{Anwendung}.config.plist`, suchen nach dem folgenden Text und stellen den Wert `<string>` auf 0 für `[Benutzername]`, 1 für `[Benutzername] ([Vorname] [Nachname])` oder auf 2 für `[Benutzername] ([Nachname], [Vorname])` ein:
`<key>UserNameFormattingStyle</key> <string>0</string>`
- Bei **QuarkXPress und QuarkCopyDesk für Windows** öffnen Sie die Datei „Quark.QPP.client.config“ (im Anwendungsordner), suchen nach dem folgenden Text und stellen das Attribut `value` auf `DEFAULT` für `[Benutzername]`, auf `FIRSTNAME_LASTNAME` für `[Benutzername] ([Vorname] [Nachname])` oder auf `LASTNAME_FIRSTNAME` für `[Benutzername] ([Nachname], [Vorname])` ein.
`<add key="UserNameFormattingStyle" value="DEFAULT"/>`

Ändern der Schrift und Größe von Vorsichten (nur Windows)

Um die für Attributwerte verwendete Schrift und Größe auf von der Standardsuchpalette abweichende Werte zu ändern, fügen Sie dem Abschnitt `<appSettings>` der Datei „Quark Publishing Platform Client.exe.config“ folgenden Schlüssel hinzu, wobei das Attribut `"value"` die gewünschte Schrift und Größe enthält:

```
<add key="FontName_Text Preview" value="Arial, 18"/>
```

Einstellen der maximalen Anzahl Assets, die abgerufen werden (nur Windows)

Sie ändern die maximale Anzahl Assets, die pro Sammlung abgerufen werden, wenn der Anwender den Befehl `Asset abrufen` oder den Befehl `Sammlung abrufen` ausführt, indem Sie folgenden Schlüssel im Abschnitt `<appSettings>` der Datei „Quark Publishing Platform Client.exe.config“ mit der gewünschten Anzahl im Attribut `„value“` hinzufügen:

```
<add key="MaximumAssetFetchPerCollection" value="50"/>
```

➔ Der Standardwert ist 50.

Festlegen, ob Chunked Encoding verwendet werden soll (nur Windows)

Um festzulegen, ob **Chunked Encoding** beim Übertragen von Dateien auf HTTP verwendet werden soll, fügen Sie den folgenden Schlüssel im Abschnitt `<appSettings>` der Datei „Quark Publishing Platform Client.exe.config“ hinzu. Unterstützte Werte sind 0 und 1.

```
<add key="UseChunkedEncoding" value="0"/>
```

➔ Der Standardwert ist 0.

Festlegen der Unterstützung für Lazy Loading bei der Suche (nur Windows)

Um die Unterstützung für die Lazy Loading Strategie (Nachladen bei Bedarf) bei der Suche festzulegen, fügen Sie im Abschnitt `<appSettings>` der Datei „Quark Publishing Platform Client.exe.config“ folgenden Schlüssel hinzu, wobei das Attribut „value“ die gewünschte Strategie enthält:

```
<add key="LazyLoadingMode" value="LAZY_LOADING_SCROLLBAR" />
```

➔ Folgende Werte werden unterstützt:

- `NO_LAZYLOADING` (kein Lazy Loading): Alle Assets werden gleichzeitig durch die Blockgröße konfiguriert.
- `LAZY_LOADING_HYPERLINK` (Nachladen über Hyperlink): Die spezifizierten Ergebnisse werden auf der Basis der Blockgröße abgerufen; wenn die durch die Blockgröße spezifizierten Ergebnisse abgerufen werden, erscheint ein Hyperlink, der das Abrufen des nächsten Ergebnisblocks aktiviert.
- `LAZY_LOADING_SCROLLBAR` (Nachladen mit Rollbalken): Die spezifizierten Ergebnisse werden auf der Basis der Blockgröße abgerufen; durch den Rollbalken wird das Abrufen des nächsten Ergebnisblocks aktiviert.

➔ Der Standardwert ist `LAZY_LOADING_SCROLLBAR`.

Einstellen der Lazy Loading Blockgröße (nur Windows)

Sie ändern die Blockgröße für die abzurufenden Lazy Loading Suchergebnisse, indem Sie den folgenden Schlüssel im Abschnitt `<appSettings>` der Datei „Quark Publishing Platform Client.exe.config“ mit der gewünschten Größe im Attribut „value“ hinzufügen:

```
<add key="LazyLoadingChunkSize" value="50" />
```

➔ Der Standardwert beträgt 50.

Einstellen der Service-Timeout-Werte für alle Remote Service-Referenzen (Nur Windows)

Sie ändern den Service-Timeout-Wert für alle Remote Service-Referenzen, indem Sie den folgenden Schlüssel im Abschnitt `<appSettings>` der Datei „Quark Publishing Platform Client.exe.config“ mit der gewünschten Zeit (in Sekunden) im Attribut „value“ hinzufügen:

```
<add key="ServiceTimeoutInSeconds" value="0" />
```

➔ Der Standardwert beträgt 0. Verwenden Sie den Standardwert, der in der Platform Server Konfiguration festgelegt wurde.

Einstellen des Service Timeout-Werts für den Publishing Service (nur Windows)

Sie ändern den Service-Timeout-Wert für den Publishing Service, indem Sie den folgenden Schlüssel im Abschnitt `<appSettings>` der Datei „Quark Publishing Platform

Client.exe.config“ mit der gewünschten Zeit (in Sekunden) im Attribut „value“ hinzufügen:

```
<add key="PublishingServiceTimeoutInSeconds" value="600"/>
```

- ➔ Der Standardwert ist 600. Der Wert 0 gibt an, dass die Einstellungen von `ServiceTimeoutInSeconds` wirksam sind.

Festlegen der Schriftgröße der Textprobenzeile (nur Windows)

Um die für Schriftgröße für die Textprobenzeile zu ändern, fügen Sie im Abschnitt `<appSettings>` der Datei „Quark Publishing Platform Client.exe.config“ folgenden Schlüssel hinzu, wobei das Attribut "value" die gewünschte Schriftgröße enthält:

```
<add key="CopyTastingRowFontSize" value="25"/>
```

Festlegen des Symbols für eine Dateierweiterung (nur Windows)

Um ein Symbol mit einer speziellen Dateierweiterung zu verknüpfen fügen Sie den folgenden Schlüssel im Abschnitt `<appSettings>` der Datei „Quark Publishing Platform Client.exe.config“ mit dem Pfad zum gewünschten Symbol im Attribut „value“ hinzu:

```
<add key="icon_<Dateierweiterung>" value="Pfad zum Symbol"/>
```

- ➔ Der Schlüssel definiert die „Dateierweiterung“ und der Wert gibt den Dateipfad an.

Einstellen der Passwortspeicherung (nur Mac OS X)

Quark Publishing Platform Clients können so eingestellt werden, dass der Benutzername von Login zu Login gespeichert bleibt, das Benutzerpasswort jedoch nicht. So konfigurieren Sie diese Option:

- Für **Quark Publishing Platform Client** öffnen Sie die Datei
`~/Library/Preferences/Quark/QPP/[QPP
Framework-Version]/com.quark.qpp.client.{Application}.config.plist,`
suchen folgenden Text und ändern `<false/>` in `<true/>` um:
`<key>RememberPassword</key> <false/>`
- Für **QuarkXPress für Mac OS X** öffnen Sie die Datei
`~/Library/Preferences/Quark/QPP/[QPP
Framework-Version]/com.quark.qpp.client.{Application}.config.plist,`
suchen folgenden Text und ändern `<false/>` in `<true/>` um:
`<key>RememberPassword</key> <false/>`
- Für **QuarkCopyDesk für Mac OS X** öffnen Sie die Datei
`~/Library/Preferences/Quark/QPP/[QPP
Framework-Version]/com.quark.qpp.client.{Application}.config.plist,`
suchen folgenden Text und ändern `<false/>` in `<true/>` um:
`<key>RememberPassword</key> <false/>`

Verwenden von Mac Clients mit einem Proxyserver

Wenn Sie wünschen, dass sich Mac OS X Clients außerhalb der Firewall über einen Proxyserver mit Quark Publishing Platform Server verbinden können, führen Sie folgendes durch:

- Für **Quark Publishing Platform Client** öffnen Sie die Datei `~/Library/Application Support/Quark/QPP/[QPP Framework Version]/com.quark.qpp.client.{Anwendung}.config.plist`, suchen Sie den folgenden Text auf und ändern Sie `<false/>` in `<true/>`:
`<key>UseProxy</key> <false/>`
- Für **QuarkXPress für Mac OS X** öffnen Sie die Datei `~/Library/Application Support/Quark/QPP/[QPP Framework Version]/com.quark.qpp.client.{Anwendung}.config.plist`, suchen folgenden Text und ändern `<false/>` in `<true/>` um:
`<key>UseProxy</key> <false/>`
- Für **QuarkCopyDesk für Mac OS X** öffnen Sie die Datei `~/Library/Application Support/Quark/QPP/[QPP Framework Version]/com.quark.qpp.client.{Anwendung}.config.plist`, suchen folgenden Text und ändern `<false/>` in `<true/>` um:
`<key>UseProxy</key> <false/>`

Verwenden von Windows Clients mit einem Proxyserver

So verwenden Sie einen Windows Client mit einem Proxyserver:

- 1 Öffnen Sie die Datei "[Programmname].exe.config" und suchen Sie folgende Zeile:
`<!-- <add key="ProxyAddress" value="http://<proxyname>:<portnumber>" /> -->`
- 2 Entfernen Sie die Auskommentierung der Zeile und fügen Sie die entsprechenden Proxydaten ein.
- 3 Sichern Sie die Datei und schließen Sie sie.

Spiegeln der Sammlungshierarchie beim Auschecken/Laden

Standardmäßig spiegeln Quark Publishing Platform Clients die Sammlungshierarchie auf dem lokalen Laufwerk, wenn Assets ausgecheckt oder geladen werden. Diese Option lässt sich jedoch ändern.

Ausschalten des Spiegeln von Sammlungen: Mac OS X

So stellen Sie das Spiegeln von Sammlungen unter Mac OS X ab:

- 1 Für Quark Publishing Platform Client öffnen Sie die Datei `~/Library/Preferences/Quark/QPP/[QPP Framework Version]/com.quark.qpp.client.Quark Publishing Platform`

`Client.config.plist`, suchen folgenden Text und ändern `<true/>` in `<false/>` um:

```
<key>MirrorCollectionHierarchy</key> <true/>
```

- 2 Für QuarkXPress öffnen Sie die Datei `~/Library/Preferences/Quark/QPP/[QPP Framework Version]/com.quark.qpp.client.QuarkXPress.config.plist`, suchen folgenden Text und ändern `<true/>` in `<false/>` um:

```
<key>MirrorCollectionHierarchy</key> <true/>
```

- 3 Für QuarkCopyDesk öffnen Sie die Datei `~/Library/Preferences/Quark/QPP/[QPP Framework Version]/com.quark.qpp.client.QuarkCopyDesk.plist`, suchen folgenden Text und ändern `<true/>` in `<false/>` um:

```
<key>MirrorCollectionHierarchy</key> <true/>
```

Ausschalten des Spiegels von Sammlungen: Windows

So stellen Sie das Spiegeln von Sammlungen unter Windows ab:

- 1 Für Quark Publishing Platform Client öffnen Sie die Datei "Quark Publishing Platform Client.exe.config", suchen folgenden Key und stellen `value` auf 0.

```
<add key="MirrorCollectionHierarchy" value="1"/>
```

- 2 Für QuarkXPress öffnen Sie die Datei "Quark.QPP.Client.config", suchen folgenden Key und stellen `value` auf 0.

```
<add key="MirrorCollectionHierarchy" value="1"/>
```

- 3 Für QuarkCopyDesk öffnen Sie die Datei "Quark.QPP.Client.config", suchen folgenden Key und stellen `value` auf 0.

```
<add key="MirrorCollectionHierarchy" value="1"/>
```

Konfigurieren der Publishing Channels

Sie können einstellen, welche Publishing Channels von Quark Publishing Platform unterstützt werden.

Konfigurieren der Publishing Channels: Mac OS X

So konfigurieren Sie die Publishing Channels unter Mac OS X:

- 1 Öffnen Sie für alle Mac OS Clients folgende Datei:

```
~/Library/Preferences/Quark/QPP/[QPP Framework Version]/com.quark.qpp.publishing.preferences.v7.plist
```

- 2 Suchen Sie folgenden Key, der die Liste der vom Client akzeptierten Publishing Channels festlegt und fügen Sie Channels hinzu oder entfernen Sie Channels.

```
<key>EnabledPublishingChannels</key>
<array>
  <string>qxpPdf</string>
  <string>qxpEpub</string>
  <string>qxpAppStudio</string>
  <string>qxpAppStudioPackage</string>
  <string>busDocPdf</string>
  <string>busDocHtml</string>
  <string>busDocQxp</string>
```

```
<string>busDocAppStudio</string>
<string>busDocAppStudioPackage</string>
<string>ditaDocPdf</string>
<string>ditaDocWordRTF</string>
<string>ditaDocHtml</string>
<string>collectBusdocForOutput</string>
<string>collectDitaForOutput</string>
<string>smartDocHtml5Publication</string>
<string>busDocHtml5Publication</string>
<string>visioDocPdf</string>
<string>fetchVisioPage</string>
<string>fetchPowerPointSlide</string>
</array>
```

- 3 Hier können Sie die gewünschten Publishing Channels hinzufügen oder entfernen.

Konfigurieren der Publishing Channels: Windows

So konfigurieren Sie die Publishing Channels unter Windows:

- 1 Öffnen Sie für Quark Publishing Platform Client die Datei „Quark Publishing Platform Client.exe.config“ und suchen Sie folgenden Key, der die Liste der vom Client akzeptierten Publishing Channels festlegt.

```
<add key="EnabledPublishingChannels"
value="qxpPdf,qxpEpub,qxpAppStudio,qxpAppStudioPackage,
busDocPdf,busDocHtml,busDocQxp,busDocAppStudio,
busDocAppStudioPackage,ditaDocPdf,ditaDocWordRTF,ditaDocHtml,
qxpAppStudio,collectBusdocForOutput,collectDitaForOutput,
smartDocHtml5Publication,
busDocHtml5Publication,visioDocPdf,fetchVisioPage,fetchPowerPointSlide"/>
```

- 2 Hier können Sie die gewünschten Publishing Channels hinzufügen oder entfernen.
- 3 Für QuarkXPress und QuarkCopyDesk öffnen Sie die Datei „Quark.QPP.Client.config“ und nehmen dieselben Änderungen vor.

Konfiguration der Delivery-Channels

Sie können einstellen, welche Delivery-Channels von Quark Publishing Platform unterstützt werden.

Konfiguration der Delivery-Channels: Mac OS X

So konfigurieren Sie die Delivery Channels unter Mac OS X:

- 1 Öffnen Sie für alle Mac OS X Clients folgende Datei:

```
~/Library/Preferences/Quark/QPP/[QPP Framework
Version]/com.quark.qpp.publishing.preferences.v5.plist
```

- 2 Suchen Sie nach folgenden Zeilen:

```
<key>EnableDeliveryChannels</key> <false/>
```

- 3 Ändern Sie das Element <false/> in <true/>

- 4 Suchen Sie folgenden Key, der die Liste der vom Client akzeptierten Delivery-Channels angibt:

```
<key>EnabledDeliveryChannels</key> <array>
<string>checkInToSharepoint</string> <string>checkInToFileNet</string>
<string>sendEmail</string> <string>checkInToDocumentum</string>
<string>sendToFileSystem</string> <string>sendToFTPServer</string> </array>
```

- 5 Hier können Sie die gewünschten Delivery-Channels hinzufügen oder entfernen.

Konfiguration der Delivery-Channels: Windows

So konfigurieren Sie die Publishing Channels unter Windows:

- 1 Öffnen Sie für alle Windows Clients folgende Datei:

```
~application name.exe.config
```

- 2 Suchen Sie nach folgenden Zeilen:

```
<add key = "EnableDeliveryChannels" value = "0" >
```

- 3 Ändern Sie die 0 in eine 1

- 4 Suchen Sie folgende Zeile, die die Liste der vom Client akzeptierten Delivery-Channels angibt:

```
<add key = "EnabledDeliveryChannels" value = "checkInToSharepoint,checkInToFileNet,sendEmail,checkInToDocumentum,sendToFTPServer,sendToFileSystem"/>
```

- 5 Hier können Sie die gewünschten Delivery-Channels hinzufügen oder entfernen.

Einstellen der Vorgaben für Quark XML Author für Plattform

Einstellen des Speicherorts für das Auschecken

Sie legen die Vorgabe „Speicherort für das Auschecken“ fest, indem Sie den Schlüssel `CheckoutLocation` im Abschnitt `<appSettings>` der Datei „Quark.CMS.Adapters.config“ bearbeiten. Damit definieren Sie die anfänglichen Vorgaben beim ersten Start der Anwendung.

Ändern Sie den Speicherort für das Auschecken, indem Sie **Ablage/Datei > Vorgaben > Allgemein** aufrufen und den entsprechenden Speicherort aufsuchen.

So setzen Sie diese Voreinstellung zurück:

- Auf Standard zurücksetzen: Anwender können die Vorgaben ändern, indem Sie den Schlüssel `CheckoutLocation` im Abschnitt `appSettings` der Datei „Quark.CMS.Adapters.config“ ändern und die Funktion **Erweitert > Auf Standard zurücksetzen** nutzen.

Einstellen der Datei-Löschen-Vorgabe beim Sichern und Schließen

Mithilfe des Schlüssels `FileDeletionOption` im Abschnitt `<appSettings>` der Datei „Quark.CMS.Adapters.config“ stellen Sie die Vorgaben für die Löschung von Dateien ein. Damit definieren Sie die anfänglichen Vorgaben beim ersten Start der Anwendung.

Um die Vorgabe „Datei löschen“ zu ändern, rufen Sie **Ablage/Datei > Vorgaben > Allgemein** auf und wählen eine der folgenden Optionen:

- 1 Ohne Warnmeldung löschen
- 2 Löschen nie zulassen
- 3 Vor dem Löschen fragen

Diese Vorgabe kann wie folgt zurückgesetzt werden:

- Auf Standard zurücksetzen: Anwender können die Vorgaben ändern, indem Sie den Schlüssel `FileDeletionOption` im Abschnitt `appSettings` der Datei „Quark.CMS.Adapters.config“ ändern und die Funktion `Erweitert > Auf Standard zurücksetzen` nutzen.
- Konfiguration ändern: Ein Administrator aktualisiert die Dateien und teilt anschließend die aktualisierte Datei „Quark.CMS.Adapters.config“ an allen Speicherorten und bittet den Anwender, die neuen Vorgaben mit der Funktion `Erweitert > Konfiguration ändern` zu laden.

Einstellen der Vorgaben für die Schnellsuche

Mithilfe des Schlüssels `QuickSearchOption` im Abschnitt `<appSettings>` der Datei „Quark.CMS.Adapters.config“ stellen Sie die Vorgaben für die Schnellsuche ein. Damit definieren Sie die anfänglichen Vorgaben beim ersten Start der Anwendung.

Um die Vorgabe „Datei löschen“ zu ändern, rufen Sie `Ablage/Datei > Vorgaben > Suche` auf und wählen eine der folgenden Optionen:

- 1 Name
- 2 Inhalt
- 3 Name und Inhalt

Diese Voreinstellung kann folgendermaßen zurückgesetzt werden:

- Auf Standard zurücksetzen: Anwender können die Vorgaben ändern, indem Sie den Schlüssel `QuickSearchOption` im Abschnitt `appSettings` der Datei „Quark.CMS.Adapters.config“ ändern und die Funktion `Erweitert > Auf Standard zurücksetzen` nutzen.

Einstellen der Vorgabe Versionskommentare anzeigen beim Sichern und Schließen

Mithilfe des Schlüssels `RevisionCommentsDisplayOption` im Abschnitt `<appSettings>` der Datei „Quark.CMS.Adapters.config“ stellen Sie die Vorgaben für die Anzeige von Versionskommentaren ein. Damit definieren Sie die anfänglichen Vorgaben beim ersten Start der Anwendung.

Um die Vorgabe „Datei löschen“ zu ändern, rufen Sie `Ablage/Datei > Vorgaben > Suche` auf und wählen eine der folgenden Optionen:

- 1 Immer
- 2 Nie
- 3 Nur für neue Aufgaben

Diese Vorgabe kann wie folgt zurückgesetzt werden:

- Auf Standard zurücksetzen: Anwender können die Vorgaben ändern, indem Sie den Schlüssel `RevisionCommentsDisplayOption` im Abschnitt `appSettings` der Datei „Quark.CMS.Adapters.config“ ändern und die Funktion `Erweitert > Auf Standard zurücksetzen` nutzen.
- Konfiguration ändern: Ein Administrator aktualisiert die Dateien und teilt anschließend die aktualisierte Datei „Quark.CMS.Adapters.config“ an allen

Speicherorten und bittet den Anwender, die neuen Vorgaben mit der Funktion `Erweitert > Konfiguration ändern` zu laden.

Konfigurieren von Platform Adapters für Microsoft Office Komponenten zum Web Sharing

- ➔ Die Web Sharing Konfiguration wird nur für Standalone Server-Installationen empfohlen und nicht für die Bereitstellung externer Web-Container oder den EAR-basierten Einsatz. Wenn eine Web Sharing Konfiguration in diesen Szenarien erforderlich ist, müssen Sie sich an den Technischen Support von Quark wenden.

In diesem Kapitel finden Sie eine Anleitung, wie die folgenden Microsoft Office Komponenten installiert werden:

- Quark Publishing Platform Adapter für Microsoft Office – Word
- Quark Publishing Platform Adapter für Microsoft Office – Excel
- Quark Publishing Platform Adapter für Microsoft Office – PowerPoint

Der Speicherplatz für die Bereitstellung befindet sich im Webapps-Ordner von „Quark Publishing Platform Server“ auf dem Platform Server Computer.

Vor der Installation einer dieser Komponenten müssen folgende Schritte abgeschlossen sein:

- 1 Kopieren und Extrahieren Sie die Archivdatei „Platform<version and Build>_Adapter_for_Office.zip“.
- 2 Kopieren/überschreiben Sie alle Dateien aus dem Ordner „webapps\ROOT“ des extrahierten Archivs in den Ordner „Quark Publishing Platform Server\webapps\ROOT“.
- 3 Kopieren Sie den Inhalt des Ordners „webapps\clientinstallers“ in den Ordner „Quark Publishing Platform Server\webapps\clientinstallers“.

Folgen Sie anschließend den Schritten in den nächsten Abschnitten, um die einzelnen Komponenten zu installieren.

Konfiguration des Quark Publishing Platform Adapters für Microsoft Office – Word

So installieren Sie den Platform Adapter für Microsoft Word:

- 1 Extrahieren Sie den Inhalt der Archivdatei „Word Adapter.zip“ im Ordner `webapps\clientinstallers` auf dem Platform Server Computer in den Ordner `webapps\clientinstallers\Word Adapter`.
- 2 Öffnen Sie die Datei „ConfigureAdapter.bat“, die sich im Ordner `clientinstallers/Word Adapter` befindet.
- 3 Aktualisieren Sie `HostName/IP` und den `Port` und starten Sie anschließend die Batch-Datei.

Konfiguration des Quark Publishing Platform Adapters für Microsoft Office – Excel

So installieren Sie den Platform Adapter für Microsoft Excel:

- 1 Extrahieren Sie den Inhalt der Archivdatei „Excel Adapter.zip“ im Ordner `webapps\clientinstallers` auf dem Platform Server Computer in den Ordner `webapps\clientinstallers\Excel Adapter`.
- 2 Öffnen Sie die Datei „ConfigureAdapter.bat“, die sich im Ordner `clientinstallers/Excel Adapter` befindet.
- 3 Aktualisieren Sie `HostName/IP` und den `Port` und starten Sie anschließend die Batch-Datei.

Konfigurieren von Quark Publishing Platform Adapter für Microsoft Office – PowerPoint

So installieren Sie den Platform Adapter für Microsoft PowerPoint:

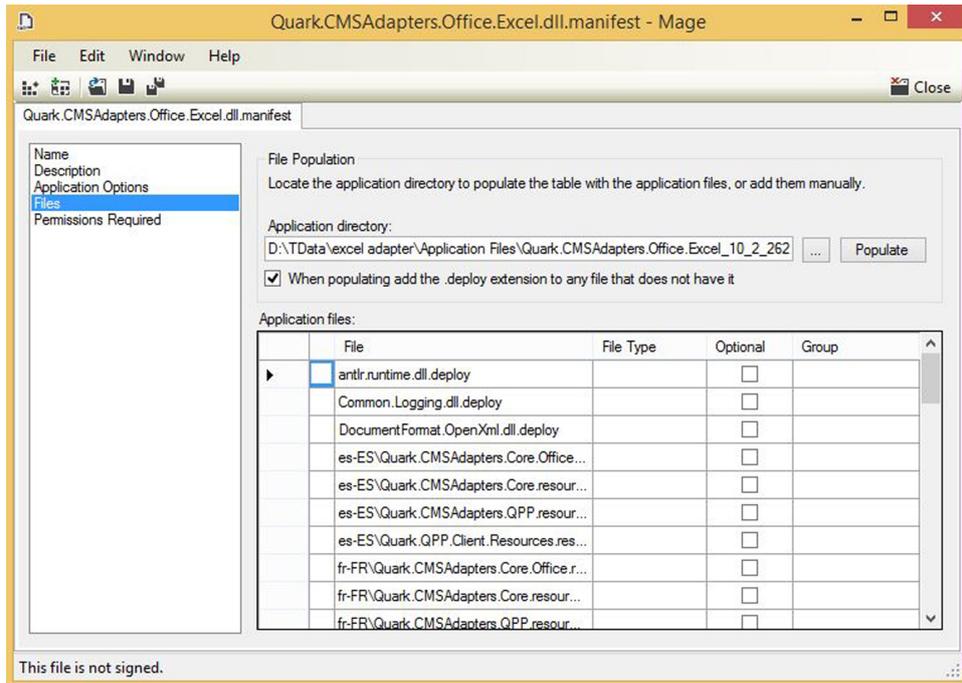
- 1 Extrahieren Sie den Inhalt der Archivdatei „PowerPoint Adapter.zip“ im Ordner „webapps\clientinstallers“ auf dem Platform Server Computer in den Ordner „webapps\clientinstallers\PowerPoint Adapter“.
- 2 Aktualisieren Sie die Parameter `HostName/IP` und `Port` in der Batchdatei „ConfigureAdapter.bat“ unter „clientinstallers/PowerPoint Adapter“ und starten Sie anschließend die Batchdatei.

Aktualisieren veröffentlichter ClickOnce-Bereitstellungen

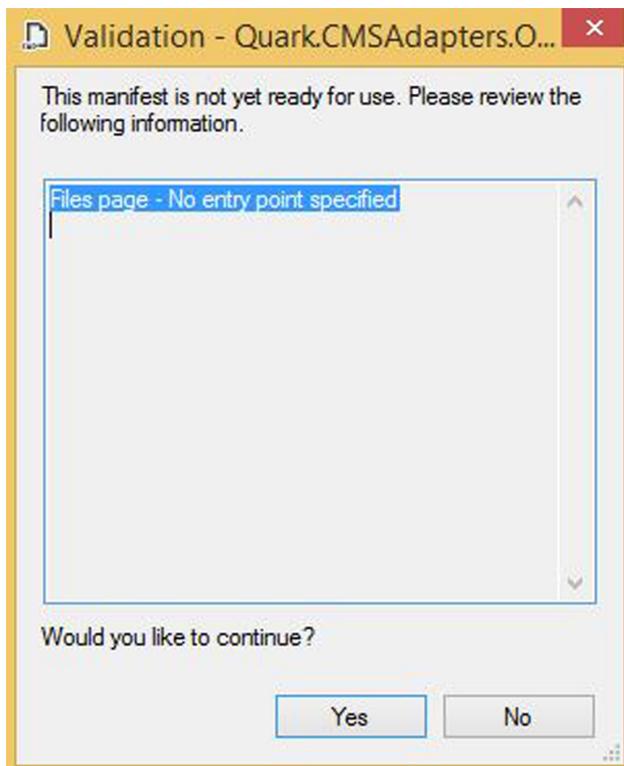
Zum Aktualisieren eines veröffentlichten ClickOnce-Bereitstellungs-Package verwenden Sie das Tool zum Generieren und Bearbeiten von Manifesten (`mage.exe`), das mit Microsoft Visual Studio installiert wurde.

Aktualisieren des Package und des Anwendungs-Manifest

- 1 Gehen Sie zu `Application Files\Quark.CMSAdapters.Office.<Komponente>_%HöchsteVersion%` und löschen Sie die Datei „Quark.CMSAdapters.Office.<Komponente>.vsto“. Für Komponente wird dabei Word, Excel oder PowerPoint eingesetzt.
- 2 Nehmen Sie nun die erforderlichen Updates an den Konfigurationsdateien vor.
- 3 Run the Mage tool. Verwenden Sie die Visual Studio-Eingabeaufforderung der Visual Studio Tools und geben Sie `mage.exe` ein.
- 4 In Mage öffnen Sie das Anwendungs-Manifest „Quark.CMSAdapters.Office.<Komponente>.dll.manifest“ im Ordner `\Application Files\App_%HöchsteVersion%\`.



- Wählen Sie **Dateien** und aktivieren Sie die Option **Beim Füllen die Erweiterung ".deploy"** zu allen Dateien ohne diese Erweiterung hinzufügen. Klicken Sie anschließend auf die Schaltfläche **Auffüllen** und sichern Sie die Datei entweder durch Klick auf **Speichern** oder **Datei > Speichern**.



- Klicken Sie auf **Ja** im Dialog **Validierung**.



- 7 Geben Sie den Speicherpfad der Zertifikatsdatei und deren Passwort (falls vergeben) ein, wenn Sie die Option **Mit Zertifikatsdatei signieren** auswählen. Klicken Sie auf **OK**.

Update der VSTO

- 1 Zum Update des im Package-Ordner vorhandenen „Quark.CMSAdapters.Office.<Komponente>.vsto“ verwenden Sie folgenden Befehl:


```
mage -Update "<Pfad>\Quark.CMSAdapters.Office.<Komponente>.vsto"
-AppManifest
"<Pfad>\Quark.CMSAdapters.Office.<Komponente>.dll.manifest" -CertFile
"<Zertifikatpfad>\<Zertifikat>.pfx" -Password "<Passwort>"
```
- 2 Kopieren Sie die Datei „Quark.CMSAdapters.Office.Word.vsto“ aus dem Package-Ordner nach `\Application Files\App_%HöchsteVersion%`.

Manuelle Konfiguration von QuarkXPress und QuarkCopyDesk XTensions

Die XML-basierten Vorgaben von QuarkXPress und QuarkCopyDesk XTensions sind in den Dateien `QPPXPressXT.xml` und `QPPCopyDeskXT.xml` im Vorgabenordner der Anwendung gespeichert.

- ➔ In Windows befindet sich der QuarkXPress Vorgabenordner hier:


```
C:\Users\<username>\AppData\Local\Quark\QuarkXPress 2016
```

 und der QuarkCopyDesk Vorgabenordner hier: `C:\Users\<username>\AppData\Local\Quark\QuarkCopyDesk 2016`
- ➔ In MAC OS X befindet sich der QuarkXPress Vorgabenordner hier:


```
/Volume/users/username/Library/Preferences/Quark/QuarkXPress 2016.
```

Der Vorgabenordner von QuarkCopyDesk befindet sich hier:

`/Volume/users/username/Library/Preferences/Quark/QuarkCopydesk 2016.`

Die meisten Vorgaben werden mithilfe der Benutzeroberfläche eingestellt, die Festlegung des Speicherorts der Platform-Menüobjekte jedoch wird manuell vorgenommen.

So setzen Sie den Speicherort der Platform-Menüobjekte:

- Definieren Sie das Attribut `location` im Node `PlatformMenuItemsLocation` innerhalb des Abschnitts `<Basic>` der Dateien „QPPXPressXT.xml“ und „QPPCopyDeskXT.xml“. Wählen Sie einen der folgenden Werte:

- 1** 0 Menü Ablage/Datei (Standard)
- 2** 1 Menü Platform
- 3** 2 sowohl im Menü Ablage/Datei als auch im Menü Platform

Die Handhabung von Backups und Dateispeicherung

Sie wählen die Backup-Software und legen die Backup-Intervalle für Ihre Quark Publishing Platform Datenbank, Ihre Quark Publishing Platform Assets und wichtige Dateien fest (z. B. für Quark® Job Jackets® Dateien und Skript-Dateien). Quark empfiehlt die Synchronisierung der Backups für Ihre Datenbank, die Assets und die wichtigen Dateien, um Inkonsistenzen zu vermeiden, wenn ein Backup wiederhergestellt werden soll. Quark empfiehlt darüber hinaus ein regelmäßiges Prüfen der Wiederherstellbarkeit Ihrer Backups.

Falls Sie Ihren Assetspeicher verschieben, folgen Sie der Anleitung in "[Verschieben des Quark Publishing Platform Assetspeichers \(Repository\)](#)".

Backup von Quark Publishing Platform Server

Quark empfiehlt das Anhalten von Quark Publishing Platform Server vor einem Backup, jedoch ist das Stoppen von Quark Publishing Platform Server nicht zwingend erforderlich. Führen Sie ein Backup Ihrer Datenbank, Ihrer Assets und der wichtigen Dateien (z. B. von FTS Index-Dateien) auf einem separaten Speichergerät durch. Sie können zwar den gesamten Ordner "Quark Publishing Platform Server" sichern, dies sind jedoch die wichtigsten Ordner:

- Ordner "conf" (enthält System-Konfigurationsdateien, darunter solche, die nach der Installation manuell bearbeitet wurden)
- Ordner "index" (enthält die Indexdateien für Volltextsuche)

Backup Ihrer Datenbank

Die Datenbank enthält alle Metadaten für Quark Publishing Platform Assets.

Wenn Sie eine Microsoft® SQL Datenbank oder eine Oracle Datenbank verwenden, nutzen Sie die MS-SQL oder Oracle beigefügten Backup-Tools und Anleitungen.

Wenn Sie für Quark Publishing Platform Server eine eingebettete HSQL Datenbank verwenden, werden die Daten der Datenbank ausschließlich im Ordner "database" des Ordners "Quark Publishing Platform Server" gesichert und Sie müssen ein Backup des Ordners "database" anfertigen, um die Metadaten zu sichern und Ihre Workflow-Konfiguration zu schützen. Wenn der Ordner "database" nicht wiederhergestellt werden kann, müssen Sie eine manuelle Identifikation der Quark Publishing Platform Assets durchführen.

Backup von Assets

Sie wählen die Software und die Intervalle für die Backups Ihrer Quark Publishing Platform Assetspeicher aus.

➔ Assetnamen sind im Quark Publishing Platform Assetspeicher verschlüsselt abgelegt.

Backup von Indexdateien (Volltextsuche)

Quark Publishing Platform Server indexiert alle in die Datenbank eingeecheckten Dateien, daher können Sie in den Textinhalten von Quark Publishing Platform Assets eine Suche durchführen. Quark Publishing Platform Server speichert Indexdaten für Volltextsuchoperationen im Ordner "index". Der Ordner "index" befindet sich auf der Stammebene Ihres Quark Publishing Platform Server Ordners und ist der Standard-Speicherort für die für Volltext-Indexe erforderlichen Dateien. Unter "[Konfiguration der Volltextindexierung](#)" erfahren Sie, wie sie den Speicherort der Volltext-Indexe durch die Bearbeitung von "LuceneTextIndexingConfig.properties" ändern können. Wenn Sie den Speicherort ändern, legen Sie Sicherungskopien des neuen Speicherortes an.

Wiederherstellen von Quark Publishing Platform Server

Falls Ihre Quark Publishing Platform Assetspeicher nicht wiederhergestellt werden müssen, sind die Speicherpfade zu den Assetspeichern noch gültig, wenn Sie Ihre Quark Publishing Platform Datenbank wiederhergestellt haben. Falls Ihre Quark Publishing Platform Assetspeicher und Ihre Datenbank wiederhergestellt werden müssen, definieren Sie den aktualisierten Speicherort anhand der Anleitung in "[Verschieben des Quark Publishing Platform Assetspeichers \(Repository\)](#)".

Wenn beispielsweise auf dem Computer, auf dem Quark Publishing Platform Server betrieben wird, die Festplatte ausfällt, sollte das neueste Backup an einem anderen Speicherort wiederhergestellt werden. Reinstallieren Sie Quark Publishing Platform Server anhand der Anleitung in der *Quark Publishing Platform ReadMe*. Stellen Sie nach der erneuten Installation von Quark Publishing Platform Server sicher, dass es nicht läuft, bevor Sie die Datenbank, die Assets und andere Dateien wiederhergestellt haben.

Wiederherstellen von Assets

Verwenden Sie nach Möglichkeit denselben Pfad, den Sie für den alten Assetspeicher verwendet haben. Wenn Sie zum Beispiel die Festplatte des Computers ersetzen, auf dem Quark Publishing Platform Server läuft, können Sie das Backup des Assetspeichers an denselben Speicherort kopieren. Versuchen Sie auch dann, wenn Sie einen neuen Computer benötigen, denselben Pfad zu verwenden (z. B. `C:\QPP_Assets`).

Wenn Sie nach Quark Publishing Platform Assets suchen können und die verschlüsselten Dateinamen betrachten möchten, sehen Sie sich nach Abschluss Ihrer Suche das Attribut "File Path" an. Die Masterdatei und alle Assetversionen befinden sich an dem selben Speicherort. Die Verschlüsselung der Assetnamen besitzt ein einheitliches Muster (z. B. `34.1.1.1.JPG`). Die erste Zahl des verschlüsselten Namens repräsentiert die Asset-ID. Die zweite Zahl ist die Versionsnummer, die dritte der Darstellungstyp für Voransichten und die vierte Zahl gibt die erste Seite der Voransicht an.

- ➔ Falls Sie Hardware wiederherstellen müssen, starten Sie Quark Publishing Platform Client und richten Sie die Verbindung zu einem oder mehreren Assetspeichern im Register **Assetspeicher** des Bildschirms **Administration: Speicherung** wieder neu ein. Löschen Sie den Assetspeicher nicht und versuchen Sie nicht, einen neuen anzulegen.

Wiederherstellen der Quark Publishing Platform Server Datenbank

Wenn Sie eine Microsoft SQL Datenbank oder eine Oracle Datenbank verwenden, nutzen Sie die MS-SQL oder Oracle beigefügten Tools und Anleitungen zur Wiederherstellung (Restore). Wenn Sie Quark Publishing Platform Server erneut installieren, können Sie die korrekten MS-SQL- oder Oracle-Daten während des Installationsprozesses eingeben.

Wiederherstellen von Volltext-Indizes

Stellen Sie den Ordner "index" an dem in der Datei "LuceneTextIndexingConfig.properties" angegebenen Speicherort wieder her.

Verschieben des Quark Publishing Platform Assetspeichers (Repository)

Wenn Sie Ihren Quark Publishing Platform Assetspeicher verschieben, können Sie den Pfad zum Assetspeicher mit Quark Publishing Platform Client aktualisieren. Unter "Konfigurieren von Speicheroptionen" in Ihrem *Handbuch zu Quark Publishing Platform* finden Sie weitere Informationen zum Einrichten der Assetspeicherung.

So aktualisieren Sie den Pfad zum Assetspeicher:

- 1 Navigieren Sie in einem Webbrowser zu [http://\[server\]:\[port\]/admin](http://[server]:[port]/admin) und melden Sie sich mit Administratorrechten an.
- 2 Rufen Sie den Bildschirm **Administration** auf und klicken Sie auf **Speicherung**. Der Bildschirm **Administration: Speicherung** wird angezeigt.
- 3 Rufen Sie das Register **Assetspeicher** auf, das einen oder mehrere Einträge in der Spalte **Assetspeicher-Name** enthält.
- 4 Markieren Sie einen Assetspeicher und wählen Sie **Bearbeiten > Assetspeicher bearbeiten**. Es erscheint eine Warnmeldung.



Dieses Meldung wird angezeigt, wenn Sie einen Assetspeicher bearbeiten

- 5 Klicken Sie auf **Ja**. Die Dialogbox **Assetspeicher bearbeiten** wird angezeigt.



Die Dialogbox **Assetspeicher bearbeiten**

- 6 Geben Sie einen neuen Ort in das Feld **Speicherort** ein.
- 7 Wiederholen Sie die Schritte 4 bis 6 für alle Assetspeicher im Register **Assetspeicher**.

Rechtliche Hinweise

© 2022 Quark Software Inc. und seine Lizenzgeber. Alle Rechte vorbehalten.

Geschützt durch folgende Patente der Vereinigten Staaten: 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. Weitere Patente sind angemeldet.

Quark, das Quark Logo und Quark Publishing Platform sind Marken oder eingetragene Marken von Quark Software Inc. und der verbundenen Unternehmen in den U.S.A. und/oder anderen Ländern. Alle anderen Marken sind Eigentum ihrer jeweiligen Besitzer.

Index

A

Authentifizierung 44, 45

B

Benutzernamen, anzeigen 32, 82

D

Datenbank-Konfiguration 41

Delivery-Channels 48

DITA Open Toolkit 57

E

Einsatz 7, 8, 10

Email 48

F

FileNet 48

I

ImageMagick 57

IPTC 60

J

Jaws 57

JVM 11

JVM Konfiguration 7

K

Kerberos Authentifizierung 44

N

Namen, anzeigen 32, 82

P

Passwörter, Speicherung 85

S

SharePoint 48

T

Tomcat 11

Transformationen, konfigurieren 42

V

Volltextindexierung 58

W

Web Client 70, 74

WebLogic 10

WebSphere 8