



# KENDOX XmlAPI

## Handbuch

**Version 4.0**  
(08.01.2013)



### **Haftung**

Dieses Handbuch wird ohne jegliche Haftung herausgegeben. Änderungen sind jederzeit möglich.

### **Copyright**

Dieses Handbuch ist urheberrechtlich geschützt. Der Nachdruck, auch auszugsweise, ist nur mit Genehmigung der Kendox AG gestattet. Alle Rechte vorbehalten.

© Kendox AG

### **Hersteller-, Produkte- und Warennamen**

Alle in diesem Handbuch erwähnten Hersteller oder Produktnamen sowie die verwendeten Software- und Hardwarebezeichnungen sind eingetragene Warenzeichen der jeweiligen Hersteller und werden ohne Gewährleistung der freien Verwendbarkeit benutzt.

### **Kendox AG**

Bahnhof-Strasse 7  
CH-9463 Oberriet SG

Tel.: +41 71 763 72 72  
Fax: +41 71 763 72 71

<http://www.kendox.com>

### Inhaltsverzeichnis

<b>1</b>	<b>Änderungen .....</b>	<b>5</b>
<b>2</b>	<b>Allgemein .....</b>	<b>6</b>
<b>3</b>	<b>XmlAPI-Schnittstellen .....</b>	<b>10</b>
<b>4</b>	<b>Allgemeine Informationen zu den Befehlen .....</b>	<b>11</b>
<b>5</b>	<b>Mit Server verbinden .....</b>	<b>12</b>
<b>6</b>	<b>Verbindung mit Server beenden .....</b>	<b>14</b>
<b>7</b>	<b>Dokumente einscannen .....</b>	<b>15</b>
<b>8</b>	<b>Dokumentanzeige/drucken/senden .....</b>	<b>19</b>
<b>9</b>	<b>Dokumentanzeige zurücksetzen .....</b>	<b>20</b>
<b>10</b>	<b>Dokument ablegen .....</b>	<b>21</b>
<b>11</b>	<b>Dokumentsuche .....</b>	<b>26</b>
<b>12</b>	<b>SearchAndListForm im Hintergrund laden .....</b>	<b>31</b>
<b>13</b>	<b>Spalteneinstellungen für alle Benutzer zu- rücksetzen .....</b>	<b>32</b>
<b>14</b>	<b>Dokumentsuche fortsetzen .....</b>	<b>33</b>
<b>15</b>	<b>Eigenschaften ändern .....</b>	<b>34</b>
<b>16</b>	<b>Dokumenteigenschaften auslesen .....</b>	<b>36</b>
<b>17</b>	<b>Eigenschaftstypen erstellen .....</b>	<b>38</b>
<b>18</b>	<b>Eigenschaftstypen löschen .....</b>	<b>40</b>
<b>19</b>	<b>Dokumente exportieren .....</b>	<b>41</b>
<b>20</b>	<b>Dokumente verknüpfen .....</b>	<b>43</b>
<b>21</b>	<b>Dokumentverknüpfung löschen .....</b>	<b>44</b>
<b>22</b>	<b>Propertytypen auflisten .....</b>	<b>45</b>
<b>23</b>	<b>Property-Typen mit Sprachen auflisten .....</b>	<b>47</b>
<b>24</b>	<b>InfoStores auflisten .....</b>	<b>48</b>
<b>25</b>	<b>Eigenschaftstypvorlagen auflisten .....</b>	<b>49</b>
<b>26</b>	<b>Importvorlagen auflisten .....</b>	<b>51</b>
<b>27</b>	<b>In eine Datei scannen .....</b>	<b>54</b>

# KENDOX XmlAPI

## Handbuch

28	Werte auf Korrektheit prüfen .....	56
29	Dokument löschen.....	58
30	Benutzer auflisten .....	59
31	Benutzer einer Benutzergruppe auflisten .....	61
32	Daten aus Verzeichnis auslesen .....	62
33	Daten aus Suchverzeichnis auslesen .....	65
34	Liste von Dokumenten anzeigen.....	68
35	Dokumente ausbuchen .....	70
36	Dokumentausbuchung aufheben .....	71
37	Dokument einbuchen .....	72
38	Signaturprofile ausgeben .....	74
39	Signaturen anzeigen .....	75
40	Signatur prüfen .....	76
41	Versionsliste anzeigen .....	78
42	Versionseigenschaften anzeigen .....	80
43	Dateiversion anzeigen.....	82
44	Dokumentenstatus anzeigen .....	83
45	Passwort ändern .....	84
46	Ordner erstellen .....	85
47	Ordneigenschaften anzeigen.....	86
48	Ordneigenschaften setzten .....	88
49	Prüfen ob Benutzer existiert .....	89
50	Benutzer löschen.....	90
51	Benutzer erstellen .....	91
52	Benutzer aktivieren.....	93
53	Benutzer deaktivieren .....	94
54	Prüfen ob Benutzergruppe existiert .....	95
55	Benutzergruppe erstellen.....	96

<b>56</b>	<b>Benutzergruppe löschen.....</b>	<b>97</b>
<b>57</b>	<b>Benutzer in Benutzergruppe einfügen .....</b>	<b>98</b>
<b>58</b>	<b>Benutzer aus Benutzergruppe entfernen.....</b>	<b>99</b>
<b>59</b>	<b>Alle Benutzergruppen eines Benutzers auflisten .....</b>	<b>100</b>
<b>60</b>	<b>Alle Rollen eines Benutzers auflisten .....</b>	<b>101</b>
<b>61</b>	<b>Signaturreport ausgeben .....</b>	<b>102</b>
<b>62</b>	<b>Such-Operatoren .....</b>	<b>103</b>

## **1 Änderungen**

Dieses Kapitel soll einen Überblick über die Neuerungen dieses Handbuchs verschaffen.

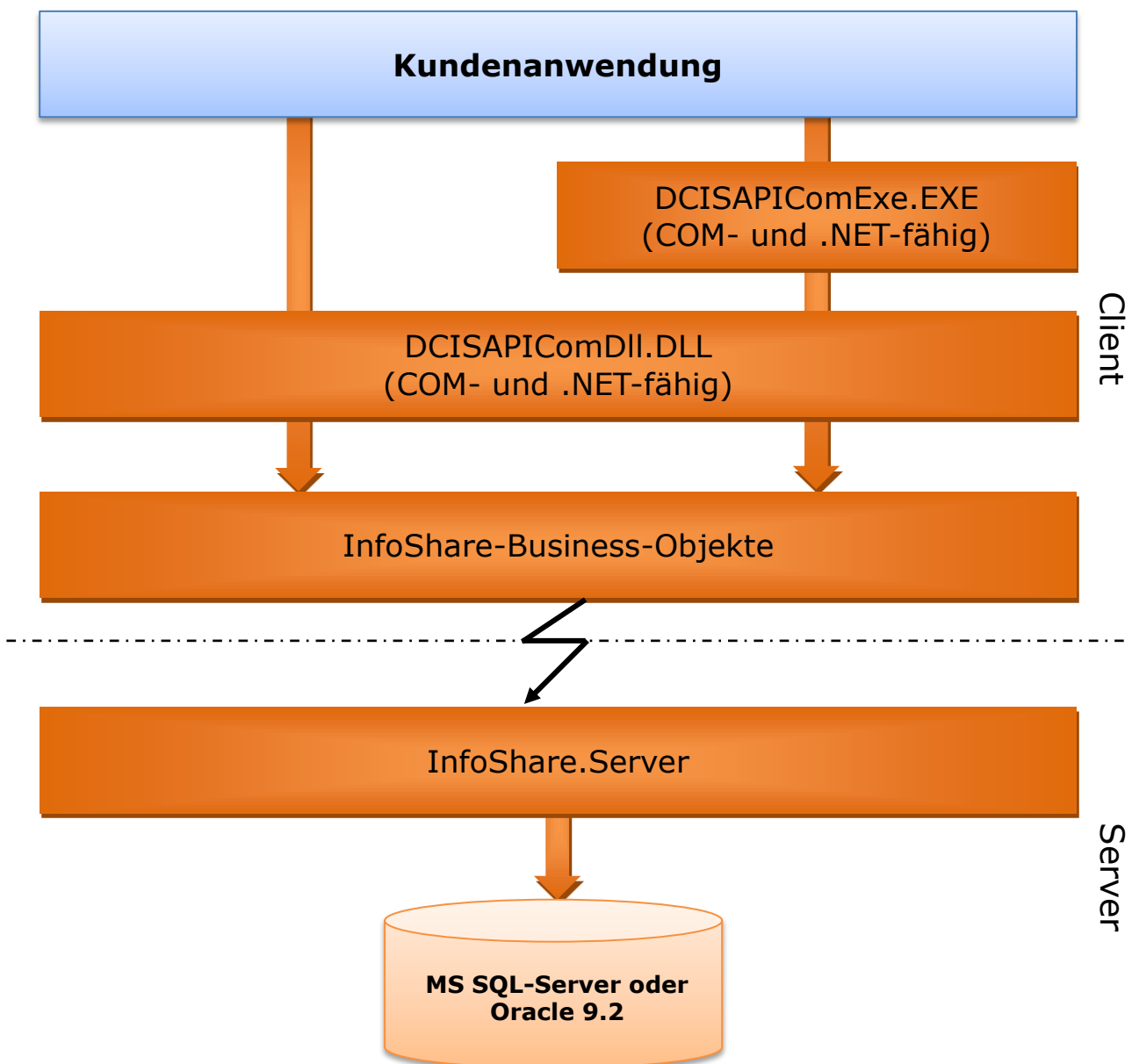
<b>Kapitel</b>	<b>Version</b>	<b>Änderung</b>
	4.0	Neues Handbuch erstellt

### 2 Allgemein

Die Kendox XmlAPI-Schnittstelle ermöglicht eine einfache Integration von Kendox InfoShare in eine Drittanwendung. Die XmlAPI-Schnittstelle wurde in Microsoft .NET entwickelt. Sie beinhaltet ebenfalls verschiedenste grafische Objekte, wie Dokumentauflistung, Dokumentablage mit Indexierungsdialog sowie eine TWAIN-Scanning Schnittstelle.

Es wurde Wert darauf gelegt, dass auch ältere Entwicklungsumgebungen, die das Microsoft COM-Modell unterstützten, auf diese Schnittstelle zugreifen können.

#### 2.1 Grafische Übersicht für die Integration



### 2.2 Aufruf DCISAPIComDll.DLL vs. DCISAPIComExe.EXE

Wie aus der grafischen Übersicht ersichtlich ist, kann ein Kendox XmlAPI-Aufruf über die Datei „DCISAPIComDll.DLL“ oder über das Programm „DCISAPIComExe.EXE“ aufgerufen werden.

Die Datei „DCISAPIComDll.DLL“ wie auch das Programm „DCISAPIComExe.EXE“ sind Microsoft COM fähig und können in ältere Entwicklungssprachen, wie z. B. VB6, eingebunden werden. Wird die „DCISAPIComDll.DLL“ direkt instanziiert, so läuft die DLL im selben Prozess wie die aufzurufende Applikation.

Wird das Programm „DCISAPIComExe.EXE“ instanziiert, so läuft das Programm „DCISAPIComExe.EXE“ in einem eigenen Prozessraum.

Beim direkten Aufruf der DCISAPIComDll.DLL müssen sämtliche benötigte InfoShare-Business-Objekte im selben Verzeichnis der aufrufenden Anwendung abgelegt sein.

Es ist ebenfalls darauf zu achten, dass bei einer Neuinstallation von InfoShare.Server auch das dazu passende Kendox XmlAPI verwendet wird (gleiche Version).

### 2.3 Registrierung von DCISAPICom-Dll.DLL/DCISAPIComExe.EXE COM-Aufrufe

Bevor „DCISAPIComDll.DLL“ oder „DCISAPIComExe.EXE“ verwendet werden können, müssen diese DLL's für den COM-Zugriff registriert werden. Dies kann über die Stapeldatei „RegisterDcisAPICom.cmd“ vorgenommen werden. Für die Nutzung aus einer Microsoft.NET Sprache ist keine Registrierung notwendig.

### 2.4 Benötigte DLLs/Objekte für Auslieferung

Im Installationsverzeichnis der Kendox XmlAPI sind alle notwendigen DLL's für die Integration vorhanden (inklusive der Unterverzeichnisse für die Sprachenabhängigen Ressourcen - DLL's, z. B. „de“). Wird das Kendox XmlAPI mit der Kundenanwendung ausgeliefert, müssen alle Objekte aus dem Installationsverzeichnis inklusive der Unterverzeichnisse mit den Ressourcen - DLL's (ausgenommen die Testapplikation TESTAPP.EXE) mit ausgeliefert werden.

### 2.5 Testapplikation

Für den Test des Kendox XmlAPI's kann das Programm „TESTAPP.EXE“ verwendet werden. Dieses Programm ermöglicht das Senden und Empfangen der XML-Aufrufe.

### 2.6 Dokumentanzeige mit InfoShare Viewer

Ab InfoShare 1.5 können Bildobjekte (TIFF, JPG, BMP, ...) und PDF-Dokumente über den mitgelieferten InfoShare Viewer angezeigt werden. Wird eine Dokumentanzeige über das XmlAPI initiiert, so muss sichergestellt werden, dass der InfoShare Viewer im Unterverzeichnis „Viewer“, welches bei der Installation des InfoShare.Client erstellt wird (Installationsverzeichnis), ebenfalls mit ausgeliefert wird. Der InfoShare Viewer muss ins Unterverzeichnis „Viewer“ unterhalb der Kendox XmlAPI-Dateien kopiert werden.



**Hinweis:** Ab Version 2.0 muss das Viewer-Verzeichnis nicht mehr kopiert werden!

### 2.7 Beispielaufruf

#### 2.7.1 Aus VB6

Nachfolgend wird ein Beispiel aufgezeigt, wie aus einer VB6-Anwendung ein InfoShare „Logon“ aufgerufen werden kann. Das Programm „DCISAPIComExe.Exe“ muss vorgängig in VB referenziert werden.

Option Explicit

```
Dim DciServer As DCISAPIComExe.XMLGateway
```

```
Dim XML As String
```

```
Dim ReturnXML As String
```

```
Dim RetNbr As Long
```

```
Dim RetMessage As String
```

```
Private Sub Form_Load()
```

```
    Set DciServer = New DCISAPIComExe.XMLGateway
```

```
End Sub
```

```
Private Sub Command1_Click()
```

```
    XML = "<?xml version='1.0' encoding='UTF-16?'><XmlAPI>" & _
```

```
    "<APICall>LOGON</APICall>" & _
```

```
    "<APIVersion>1.0</APIVersion><APILanguage>de-CH</APILanguage>" & _
```

```
    "<APIMultikeyDelimiter>;</APIMultikeyDelimiter><Logon><Request>" & _
```

```
    "<UserID>ms</UserID><Password></Password><ServerType>DCI.Server</ServerType>" & _
```

```
    "<ServerName>localhost</ServerName><Protocol>IP</Protocol>" & _
```

```
    "<IPPort>23450</IPPort></Request></Logon></XmlAPI>"
```

```
    ReturnXML = DciServer.XMLExecute(XML, RetNbr, RetMessage)
```

```
    If RetNbr <> 0 Then
```

```
        MsgBox "Fehler '" + RetMessage + "' aufgetreten", vbOKOnly + vbCritical, "Demo"
```

```
    Else
```

```
        Debug.Print ReturnXML
```

```
    End If
```

```
End Sub
```

### 2.7.2 Aus C-Sharp (.NET)

Bevor die DLL „DCISAPIComDll.Dll“ genutzt werden kann, muss diese im Projekt referenziert werden. Auch müssen alle weiteren DLL's/Objekte aus dem Kendox XmlAPI-Verzeichnis ins „\bin“-Verzeichnis der Anwendung kopiert werden.

```
using System;
using System.Diagnostics;
namespace DCISSample
{
    DCISAPIComDll.XMLGateway m_DCI = new DCISAPIComDll.XMLGateway();
    class Class1
    {
        [STAThread]
        static void Main(string[] args)
        {
            LogonKendoxInfoGate("localhost", 23450, "hal", "");
        }
        private void LogonKendoxInfoGate(string ServerName, int ServerPort, string User, string Password)
        {
            //init vars
            int myError = 0;
            string myMessage = null;
            string myReturnXml = "";
            //login to KendoxInfoGate
            myReturnXml = m_DCI.XMLExecute(this.CreateLoginXMLString(ServerName, ServerPort,
            User, Password), ref myError, ref myMessage);
            if (myError != 0)
                throw new Exception(myMessage + " (ReturnCode=" + myError.ToString());
        }
        private string CreateLoginXMLString(string ServerName, int ServerPort, string User, string Password)
        {
            return(
                "<?xml version=\"1.0\" encoding=\"UTF-16\"?>\r\n\r\n" +
                "<XmlAPI><APICall>LOGON</APICall><APIVersion>1.0</APIVersion>" +
                "<APILanguage>de-CH</APILanguage><APIMultikeyDelimiter>";
                "</APIMultikeyDelimiter><Logon><Request><UserID>" + User +
                "</UserID><Password>" + Password +
                "</Password><ServerType>DCI.SERVER</ServerType><ServerName>" + ServerName +
                "</ServerName><Protocol>IP</Protocol><IPPort>" + ServerPort.ToString() +
                "</IPPort></Request></Logon></XmlAPI>");
        }
    }
}
```

### 3 XmlAPI-Schnittstellen

#### 3.1 Aufbau Schnittstelle

Die Kommunikation zu XmlAPI erfolgt, wie bereits erwähnt über COM. Die Nachrichtenstruktur ist in XML abgebildet. Dies hat den Vorteil, dass Funktionserweiterungen ohne grossen Aufwand vorgenommen werden können und die bestehenden Programme davon nicht betroffen sind.

Nachfolgend ist der generelle Nachrichtenaufbau aufgeführt. Zu jeder abgesetzten Nachricht an XmlAPI wird eine Rückgabennachricht als Bestätigung geliefert.

#### 3.2 Anforderungsnachricht

Im XML-Header werden generelle Angaben zum API-Aufruf angegeben. Dies sind:

APICall:	Aufzurufende Funktion
APIVersion:	Version des APICalls; Zurzeit wird Version 1 unterstützt
APILanguage:	Sprache die verwendet werden soll; d. h. dass beschreibende Werte in der angegebenen Sprache geliefert werden; Mögliche Werte: „de-CH“, „en-US“, „it-CH“ und „fr-CH“
APIMultikey:	Trennsymbol für Mehrfachwerte

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>callfunction</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH|en-US|it-CH|fr-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <DocumentScan>
    <Request>
      . . .
    </Request>
  </DocumentScan>
</XmlAPI>
```

#### 3.3 Antwortnachricht

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <!-- Anforderungsspezifische Nachricht -->
  <DocumentScan>
    <Return>
      . . .
    </Return>
  </DocumentScan>
</XmlAPI>
```

## 4 Allgemeine Informationen zu den Befehlen

In diesem Kapitel werden allgemeine Informationen zu den XMLApi-Befehlen aufgeführt.

### 4.1 Optionale Befehle

Die XmlAPI-Befehle enthalten teilweise optionale Befehle (Tags). Diese werden mit der HTML-Kommentierung `<!-- TAG -->` und in grüner Farbe markiert.

Beispiel: `<!-- <APIDebug>yes</APIDebug> -->`

### 4.2 Rückgabe von Befehlen

Bei einigen Befehlen wird lediglich eine Bestätigung (z. B. der Ausführung des Befehls) zurückgegeben. Auf diese wird aus Platzgründen verzichtet. Das heisst, wenn keine Antwort eines Befehls beschrieben wird, sieht sie ähnlich des folgenden Beispiels aus.

#### 4.2.1 Beispiel

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <Logoff>
    <Return />
  </Logoff>
</XmlAPI>
```

### 4.3 Wiederkehrende XmlAPI-Befehle

Die folgenden Befehle werden im vorliegenden Dokument mehrfach verwendet und werden deshalb hier gebündelt beschrieben. Dabei kann es entsprechend zu unterschiedlichen Verwendungen der hier vorgestellten Tags kommen. Aus dem Grund der Übersichtlichkeit werden abweichende Verwendungen der Tags im jeweiligen Kapitel nochmals beschrieben.

#### 4.3.1 Dokument-ID

```
<DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
```

- Angabe der ID des Dokuments, welches verwendet werden soll oder welches verwendet wird
- Die ID kann aus dem InfoShare.Client bezogen werden („Dokumenteigenschaften“ → Reiter „Allgemein“ → Tag „Guid“)
- Die eindeutige Dokumentkennung ist vielfach zwingend (ausser z. B. beim Befehl „RESETVIEWER“)

#### 4.3.2 Ablage-ID

```
<StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID>
```

- Wird die eindeutige Ablage-ID ebenfalls angegeben, wird der Prozess für die angeforderte Funktion beschleunigt; d. h. es werden nicht alle Ablagen nach der angeforderten Dokumenten-ID durchsucht
- Die ID kann aus dem InfoShare.Client bezogen werden („Dokumenteigenschaften“ → Reiter „Allgemein“ → Tag „StoreID“)

#### 4.3.3 Session-ID

```
<SessionID>0000000a-38b2-11d8-a902-0050bf427e8f</SessionID>
```

- Nummer der Kommunikations-Sitzung (Session)
- Zurzeit ist nur eine Sitzung je APIDLL-Instanz möglich

## 5 Mit Server verbinden

(LOGON)

Mit dieser Funktion wird die Verbindung zum Server hergestellt.

### 5.1 Anforderung

```
<?xml version="1.0" encoding="utf-16"?>
<XmlAPI>
  <APICall>LOGON</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <Logon>
    <Request>
      <!-- <ConfigFile>D:\Config.xml</ConfigFile> -->
      <UserID>DCIadmin</UserID>
      <Password>meinpasswort</Password>
      <ServerType>DCI.SERVER</ServerType>
      <ServerName>KendoxServer</ServerName>
      <Tenant>KendoxDB 3.0.9</Tenant>
      <Protocol>IP</Protocol>
      <IPPort>23450</IPPort>
      <!-- <XMLRequestTraceFile>Z:\XmlAPI\myLogFile.txt</XMLRequestTraceFile> -->
      <!-- <PluginDirectory>Z:\XmlAPI\Plugins</PluginDirectory> -->
    </Request>
  </Logon>
</XmlAPI>
```

### 5.2 Code-Definitionen der Anforderung

#### 5.2.1 Konfigurationsdatei

```
<ConfigFile>D:\Config.xml</ConfigFile>
```

- Dieser optionale Parameter (Node) wird nur vom DCIAdmin benötigt
- Der Parameter dient dazu einen Pfad anzugeben, wo sich das ConfigFile vom DCI-Admin befindet; wird kein Pfad angegeben, wird im Pfad der Anwendung nach dieser Datei gesucht

#### 5.2.2 Benutzer-ID

```
<UserID>hansmeier</UserID>
```

- Benutzername, mit welchem die Sitzung (Session) gestartet werden soll
- InfoShare: Beim leer lassen des Benutzernamens wird die Anmeldung über den Windows-Benutzer ausgeführt (Single-Sign-on)

#### 5.2.3 Passwort

```
<Password>meinpasswort</Password>
```

- Passwort zur Benutzer-ID

#### 5.2.4 Server-Typ

```
<ServerType>DCI.SERVER</ServerType>
```

- Momentan werden bei diesem Tag die Servertypen „DCI.SERVER“, „ISRWS“ und „FileNet“ unterstützt

#### 5.2.5 Servername

```
<ServerName>KendoxServer</ServerName>
```

- Name oder IP-Adresse des Servers

# KENDOX XmlAPI

## Handbuch

### 5.2.6 Tenant

`<Tenant>KendoxDB 3.0.9</Tenant >`

- Name des Mandaten-Datenbankinstanz
- Das XML-Tag ist optional. Wenn im Kendox InfoShare.Server Windows Dienst mehrere Datenbankinstanzen hinterlegt wurden, muss dieser Parameter angegeben werden

### 5.2.7 Protokoll

`<Protocol>IP</Protocol>`

- Dieser Parameter (Node) wird von ISRws benötigt (Wert „DCOM“ oder „IP“)
- DCI.SERVER unterstützt ausschliesslich das IP-Protokoll (Wert „IP“)

### 5.2.8 IP-Port des Servers

`<IPPort>23450</IPPort>`

- Nummer des IP-Ports, welcher für die Kommunikation verwendet werden soll
- Falls Firewalls eingesetzt werden, muss dieser Port für den DCI.SERVER freigegeben werden

### 5.2.9 Trace-Datei

`<XMLRequestTraceFile>Z:\XmlAPI\myLogFile.txt</XMLRequestTraceFile>`

- Angabe der Trace-Datei, in welcher die XML-Anforderungen protokolliert werden
- Diese Angabe ist optional und ist grundsätzlich nur für Debug-Zwecke zu verwenden

### 5.2.10 Plugin-Pfad

`<PluginDirectory>Z:\XmlAPI\Plugins</PluginDirectory>`

- Dieses optionale Tag definiert den Pfad für Plugins

## 5.3 Antwort

`<?xml version="1.0" encoding="utf-16"?>`

`<XmlAPI>`

`<Logon>`

`<Return>`

`<SessionID>000000a-38b2-11d8-a902-0050bf427e8f</SessionID>`

`</Return>`

`</Logon>`

`</XmlAPI>`

## 6 Verbindung mit Server beenden

(LOGOFF)

Mit dieser Funktion wird die Verbindung zum Server unterbrochen. Das Tag <SessionID> muss die Nummer (GUID) der Sitzung enthalten, welche beendet werden soll.

### 6.1 Anforderung

```
<?xml version="1.0" encoding="utf-16"?>
<XmlAPI>
  <APICall>LOGOFF</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <Logoff>
    <Request>
      <!-- <SessionID>0000000a-38b2-11d8-a902-0050bf427e8f</SessionID> -->
    </Request>
  </Logoff>
</XmlAPI>
```

## 7 Dokumente einscannen

(SCAN/SCANDIALOG)

Mit dieser Funktion können Dokumente über einen TWAIN-fähigen Scanner oder eine Fotokamera eingelesen werden. Nach Aufruf dieser Nachricht wird ein Scan-Fenster (DeskSCAN) geöffnet, welches dem Benutzer die Möglichkeit gibt, Daten von einem Twain-fähigen Gerät zu lesen.

### 7.1 Anforderung

```
<?xml version="1.0" encoding="utf-16"?>
<XmlAPI>
  <APICall>SCAN|SCANDIALOG</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <DocumentScan>
    <Request>
      <StartScan>Yes</StartScan>
      <DocumentName>Scan.pdf</DocumentName>
      <ImportTemplate>
        <Template>Rechnungen</Template>
      </ImportTemplate>
      <Properties>
        <Property name="Dokumentart">
          <Value>Rechnungen</Value>
          <Value>Dokumente</Value>
        </Property>
        <Property name="Betreff">
          <Value>Rechnung4711</Value>
        </Property>
      </Properties>
      <!-- <Folders>
        <Folder>Dokumente\Rechnungen</Folder>
        <Folder>PERSONAL\Rechnungen</Folder>
      </Folders> -->
      <FilingStore>
        <Store>Kendox</Store>
      </FilingStore>
      <ProtectionDomain>
        <Domain>Produktion</Domain>
      </ProtectionDomain>
      <!-- <SigningProfile>
        <Profile>myProfile</Profile>
      </SigningProfile> -->
      <!-- <Signatures>
        <Signatur PluginClass="Com.Kendox.Dcis.BO.Crypto.Impl.Plugin.CryptoPluginAcapi,
          DcisCryptoPluginAcapi" CertID="Test" isTimeStamp="yes" HashAlg="SHA512"
          Name="SignaturName">Z:\Signaturen\Document4711.pdf.p7s</Signatur>
      </Signatures> -->
    </Request>
  </DocumentScan>
</XmlAPI>
```

## 7.2 Code-Definitionen der Anforderung

### 7.2.1 Scanvorgang unmittelbar nach Laden von DeskSCAN starten

```
<StartScan>Yes</StartScan>
```

- Ist in diesem Tag „Yes“, „True“ oder „1“ enthalten, wird der Scanvorgang direkt nach dem Laden des Scanfensters gestartet

### 7.2.2 Dokumentname

```
<DocumentName>Scan.pdf</DocumentName>
```

- Name der Datei, die im InfoShare.Client verwendet/angezeigt werden soll
- Die Dateiergung ist massgebend für das anzuwendende Anzeigeprogramm

### 7.2.3 Importvorlage

```
<ImportTemplate>
```

```
  <Template>MyFileTemplate1</Template>
```

```
</ImportTemplate>
```

- Das Importvorlagen-Tag kann optional angegeben werden
- Wird es weggelassen, müssen die Tags „<Folders>“, „<FilingStore>“ und „<ProtectionDomain>“ angegeben werden

### 7.2.4 Dokumenteigenschaften

```
<Properties>
```

```
  <Property name="Dokumentart">
```

```
    <Value>Rechnungen</Value>
```

```
    <Value>Dokumente</Value>
```

```
  </Property>
```

```
  <Property name="Betreff">
```

```
    <Value>Rechnung4711</Value>
```

```
  </Property>
```

```
</Properties>
```

- Das Übergabeformat der Dokumenteigenschaften richtet sich nach den Angaben gemäss Property-Typen-Plug-In

### 7.2.5 Ordner

```
<Folders>
```

```
  <Folder>Dokumente\Rechnungen</Folder>
```

```
  <Folder>PERSONAL\Rechnungen</Folder>
```

```
</Folders>
```

- Ordnerzuweisungs-Tag
- Optionale Angabe, wenn „<Import-Template>“ angegeben wird; wird das Tag zusätzlich zu „<ImportTemplate>“ angegeben, werden die Werte aus „<Folders>“ berücksichtigt; ist ein Dokument schon in diesem Folder vorhanden, wird die alte Referenz gelöscht

### 7.2.6 Ablage

```
<FilingStore>
```

```
  <Store>Kendox</Store>
```

```
</FilingStore>
```

- Definiert die Ablageortzuweisung
- Optionale Angabe, wenn „<Import-Template>“ angegeben wird; wird das Tag zusätzlich zu „<Import-Template>“ angegeben, werden die Werte aus „<FilingStore>“ berücksichtigt

### 7.2.7 Sicherheitsbereich

```
<ProtectionDomain>
  <Domain>Produktion</Domain>
</ProtectionDomain>
```

- Definiert die Rechtevergabe zum Dokument
- Optionale Angabe, wenn „<ImportTemplate>“ angegeben wird; wird das Tag zusätzlich zu „<ImportTemplate>“ angegeben, werden die Werte aus „<ProtectionDomain>“ berücksichtigt

### 7.2.8 Signaturprofil

```
<SigningProfile>
  <Profile>myProfile</Profile>
</SigningProfile>
```

- Dieses optionale Tag definiert das Profil, welches zum Signieren verwendet werden soll (im Dialog nicht vorhanden)

### 7.2.9 Signaturen

```
<Signatures>
  <Signatur PluginClass="Com.Kendox.Dcis.BO.Crypto.Impl.Plugin.CryptoPluginAcapi,
    DcisCryptoPluginAcapi" CertID="Test" isTimeStamp="yes" HashAlg="SHA512"
    Name="SignaturName">Z:\Signaturen\Document4711.pdf.p7s</Signatur>
</Signatures>
```

- PluginClass: optional
- certID: Beschreibung für Zertifikat (suchbar, im Dialog nicht vorhanden)
- Optional: isTimeStamp (Zeitstempel über Signatur) und HashAlg (der zur Signierung verwendete Hash-Algorithmus)

## 7.3 Antwort

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <DocumentScan>
    <DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
    <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID>
    <Renditions>
      <Rendition name="C:\Users\qs01\AppData\Local\Temp\Kendox\XmlAPI\Rechnung4711.tiff">
        d42c90b9-15e8-11e1-bf86-000c29b4469b
      </Rendition>
    </Renditions>
    <Properties>
      <Property name="Property1">
        <Value>Value1</Value>
        <Value>Value11</Value>
      </Property>
      <Property name="Property2">
        <Value>Value2</Value>
      </Property>
    </Properties>
  </DocumentScan>
</XmlAPI>
```

### 7.4 Code-Definitionen der Antwort

#### 7.4.1 Renditions

```
<Renditions>  
  <Rendition name="C:\Users\qs01\AppData\Local\Temp\Kendox\XmlAPI\Rechnung4711.tiff">  
    d42c90b9-15e8-11e1-bf86-000c29b4469b  
  </Rendition>  
</Renditions>
```

- Rückgabe der eindeutigen Dokumentobjekt-ID's; eine Dokumentobjekt-ID kennzeichnet das abgelegte Abbild

#### 7.4.2 Eigenschaften

```
<Properties>  
  <Property name="Property1">  
    <Value>Value1</Value>  
  </Property>  
  <Property name="Property2">  
    <Value>Value2</Value>  
  </Property>  
</Properties>
```

- Es werden sämtliche Dokumenteigenschaften zurückgeliefert

## 8 Dokumentanzeige/drucken/sendern

(VIEW, RESETANDVIEW, PRINT, PRINTDIALOG, MAILASIS, MAILZIPPED)

Dokumente können mittels dieser Funktion angezeigt, versendet oder gedruckt werden. Eine eindeutige Dokument-ID ist für diese Anforderung zwingend. Es werden alle Dokumentobjekte berücksichtigt. Bei Verwendung des Compart Viewers gibt es die Möglichkeit, alternativ zum VIEW Befehl, den Befehl RESETANDVIEW zu verwenden. Dieser schliesst alle bereits geöffneten Dokumente im Viewer bevor das neu angeforderte Dokument angezeigt wird.

### 8.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>VIEW</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <Document>
    <!-- DataRequest for API Call VIEW, RESETANDVIEW, PRINT, PRINTDIALOG, MAILASIS, MAILZIPPED -->
    <Request>
      <DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
      <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
    </Request>
  </Document>
</XmlAPI>
```

### 8.2 Code-Definitionen der Anforderung

#### 8.2.1 API-Aufruf

```
<APICall>VIEW</APICall>
```

- Mögliche Werte: VIEW (Anzeige), RESETANDVIEW, PRINT (Direktdruck), PRINTDIALOG (Drucken mit Dialog), MAILASIS (E-Mail „so wie sie ist“), MAILZIPPED (E-Mail als ZIP-Datei)

## 9 Dokumentanzeige zurücksetzen

*(RESETVIEWER)*

Mit diesem Befehl können alle im Compart Viewer geöffneten Dokumente geschlossen werden. Die „leere“ Dokumentanzeige bleibt jedoch geöffnet.

### 9.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>RESETVIEWER</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
</XmlAPI>
```

## 10 Dokument ablegen

(SAVE/SAVEDIALOG)

Mittels dieser Funktion kann ein Dokument im InfoShare-System abgelegt werden. Wird die Funktion SAVEDIALOG angewendet, werden unmittelbar vor der Ablage die Dokumenteigenschaften dargestellt.

### 10.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>SAVE|SAVEDIALOG</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <DocumentSave>
    <Request>
      <DocumentName>Rechnung4711</DocumentName>
      <Renditions>
        <Rendition main="yes">Z:\Dokumente\Rechnung4711.pdf</Rendition>
      </Renditions>
      <ImportTemplate>
        <Template>Rechnungen</Template>
      </ImportTemplate>
      <!-- <CreateVersion mode="explicit">true</CreateVersion> -->
      <!-- <Comment>Text von Version 1</Comment> -->
      <!-- <Version>1</Version> -->
      <Properties>
        <Property name="KDNR" useForVersioning="false" archive="false" add="true">
          <Value>4711</Value>
          <Value>4712</Value>
        </Property>
        <Property name="Betreff">
          <Value>MeinBetreff</Value>
        </Property>
      </Properties>
      <Folders>
        <Folder>PERSONAL\Rechnungen</Folder>
        <Folder>Dokumente\Rechnungen</Folder>
      </Folders>
      <FilingStore>
        <Store>Standard</Store>
      </FilingStore>
      <ProtectionDomain>
        <Domain>Produktion</Domain>
      </ProtectionDomain>
      <!-- <LifeCycle>myLifeCycle</LifeCycle> -->
      <!-- <LifeCycleStartDate>1.1.2012</LifeCycleStartDate> -->
      <SigningProfile>
        <Profile>Signatur</Profile>
      </SigningProfile>
      <!-- <Signatures>
        <Signatur Plugin-Class=
          "Com.Kendox.Dcis.BO.Crypto.Impl.Plugin.CryptoPluginAcapi,DcisCryptoPluginAcapi"
          CertID="Test" isTimeStamp="yes" HashAlg="SHA512"
          ReportFile="C:\SignatureReport.html" Name="SignaturName">
            Z:\Signaturen\Document4711.pdf.p7s
          </Signatur>
        </Signatures> -->
    </Request>
  </DocumentSave>
</XmlAPI>
```

## 10.2 Code-Definitionen der Anforderung

### 10.2.1 Dokumentname

```
<DocumentName>Rechnung4711</DocumentName>
```

- Definiert den Namen des abzulegenden Dokuments

### 10.2.2 Renditions

```
<Renditions>
```

```
  <Rendition main="yes">Z:\Dokumente\Rechnung4711.pdf</Rendition>
```

```
</Renditions>
```

- In diesem Tag wird das zu speichernde Dokument definiert
- Der Parameter main="yes" sollte nicht verändert werden

### 10.2.3 Import-Vorlage

```
<ImportTemplate>
```

```
  <Template>Rechnungen</Template>
```

```
</ImportTemplate>
```

- Definiert die zu verwendende Importvorlage

### 10.2.4 Neue Version erstellen

```
<CreateVersion mode="explicit">true</CreateVersion>
```

- Mit diesem Tag kann definiert werden, ob beim Ablegen des Dokumentes geprüft werden soll, ob ein ähnliches Dokument (mit gleichen Metadaten) bereits vorhanden ist; ist dies der Fall, wird eine neue Version erzeugt, andernfalls ein neues Dokument
- Mit „mode="explicit">true“ kann definiert werden, dass zwingend nur 1 Dokument vorhanden sein darf, um eine neue Version zu erzeugen; werden mehrere Dokumente mit den angegebenen Metadaten gefunden, wird ein Fehler ausgegeben
- **ACHTUNG:** Dieses Tag wird nur bei der Methode „SAVE“ berücksichtigt

### 10.2.5 Versionierungs-Kommentar

```
<Comment>Text von Version 1</Comment>
```

- Angabe des Versionierungs-Kommentars, der in InfoShare übernommen werden soll
- **ACHTUNG:** Dieses Tag wird nur bei der Methode „SAVE“ und mit dem Tag „CreateVersion“ berücksichtigt

### 10.2.6 Versionsnummer

```
<Version>1</Version>
```

- Angabe der Versionsnummer, die in InfoShare übernommen werden soll
- **ACHTUNG:** Dieses Tag wird nur bei der Methode „SAVE“ und mit dem Tag „CreateVersion“ berücksichtigt

### 10.2.7 Eigenschaften

```
<Properties>
  <Property name="KDNR" useForVersioning="false" archive="false" add="true">
    <Value>4711</Value>
    <Value>4712</Value>
  </Property>
  <Property name="Betreff">
    <Value>MeinBetreff</Value>
  </Property>
</Properties>
```

- Das Übergabeformat der Dokumenteigenschaften richtet sich nach den Angaben gemäss Eigenschaftstypen-Plug-In
- Attribut useForVersioning="false": Wird dieses Attribut mit „false“ angegeben, so wird diese Eigenschaft (Indexfeld) nicht für die Prüfung auf Vorhandensein eines Dokuments für die Versionierung verwendet;  
**ACHTUNG:** Dieses Tag darf nur in Verbindung mit dem „CreateVersion“-Tag angegeben werden
- Attribut archive="false": Mit diesem Attribut wird angegeben, dass das Indexfeld nicht aktualisiert/angelegt wird, sondern nur zur Prüfung auf eine bestehende Version verwendet wird;  
**ACHTUNG:** Dieses Tag darf nur in Verbindung mit dem „CreateVersion“-Tag angegeben werden
- Attribut add="true": Mit diesem Attribut kann definiert werden, ob ein nicht vorhandener Wert in einem Multikey-Feld hinzugefügt werden soll; mit „false“ wird angegeben, dass allfällig bestehende Werte überschrieben werden;  
**ACHTUNG:** Dieses Tag darf nur in Verbindung mit dem Tag „CreateVersion“ angegeben werden

### 10.2.8 Ordner

```
<Folders>
  <Folder>PERSONAL\Rechnungen</Folder>
  <Folder>Dokumente\Rechnungen</Folder>
</Folders>
```

- Optionales Ordnerzuweisungs-Tag falls „<Import-Template>“ angegeben wird
- Wird das Tag zusätzlich zu „<ImportTemplate>“ angegeben, werden die Werte aus „<Folders>“ berücksichtigt; ist ein Dokument schon in diesem Ordner vorhanden, wird die alte Referenz gelöscht
- Um das Dokument in einem privaten Ordner zu importieren, muss der erste Teil des Pfades mit „PRIVATE“ beginnen
- Um in einen öffentlichen Ordner zu importieren, muss mit dem Namen des öffentlichen Ordners begonnen werden (zum Beispiel „Öffentliche Ordner\Mein Verzeichnis“)

### 10.2.9 Ablage

```
<FilingStore>
  <Store>Standard</Store>
</FilingStore>
```

- Die Angabe ist optional, wenn „<ImportTemplate>“ angegeben wird
- Wird das Tag zusätzlich zu „<ImportTemplate>“ angegeben, werden die Werte aus „<FilingStore>“ berücksichtigt

### 10.2.10 Sicherheitsbereich

```
<ProtectionDomain>  
  <Domain>Produktion</Domain>  
</ProtectionDomain>
```

- Rechtevergabe zum Dokument
- Optionale Angabe, wenn „<ImportTemplate>“ angegeben wird
- Wird das Tag zusätzlich zu „<ImportTemplate>“ angegeben, werden die Werte aus „<ProtectionDomain>“ berücksichtigt

### 10.2.11 Lebenszyklus

```
<LifeCycle>myLifeCycle</LifeCycle>
```

- Lebenszyklus, der auf das Dokument gesetzt werden soll
- Dieses Tag ist optional

### 10.2.12 Lebenszyklus-Startdatum

```
<LifeCycleStartDate>1.1.2012</LifeCycleStartDate>
```

- Datum, ab dem der Lebenszyklus starten soll
- Kann als Zusatz für den angegebenen Lebenszyklus angegeben werden (im Dialog nicht ersichtlich)

### 10.2.13 Signaturprofil

```
<SigningProfile>  
  <Profile>Signatur</Profile>  
</SigningProfile>
```

- Dieses optionale Tag kann das Profil beinhalten, welches zum Signieren verwendet werden soll

### 10.2.14 Signaturen

```
<Signatures>  
  <Signatur Plugin-Class=  
    "Com.Kendox.Dcis.BO.Crypto.Impl.Plugin.CryptoPluginAcapi,DcisCryptoPluginAcapi"  
    CertID="Test" isTimeStamp="yes" HashAlg="SHA512"  
    ReportFile="C:\SignatureReport.html" Name="SignaturName">  
    Z:\Signaturen\Document4711.pdf.p7s  
  </Signatur>  
</Signatures>
```

- PluginClass: optional
- certID: Beschreibung für Zertifikat (suchbar, im Dialog nicht vorhanden)
- Optional: isTimeStamp (Zeitstempel über Signatur) und HashAlg (der zur Signierung verwendete Hash-Algorithmus)

### 10.3 Antwort

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <DocumentSave>
    <Return>
      <DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
      <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID>
      <Renditions>
        <Rendition name="Z:\Dokumente\Rechnung4711.pdf">
          3EE4F80C-3900-4e8f-B5C2-94F6EF79AFD4
        </Rendition>
      </Renditions>
      <Properties>
        <Property name="KDNR">
          <Value>4711</Value>
          <Value>4712</Value>
        </Property>
        <Property name=" Betreff ">
          <Value>MeinBetreff</Value>
        </Property>
      </Properties>
    </Return>
  </DocumentSave>
</XmlAPI>
```

### 10.4 Code-Definition der Antwort

#### 10.4.1 Renditions

```
<Renditions>
  <Rendition name="Z:\Dokumente\Rechnung4711.pdf">
    3EE4F80C-3900-4e8f-B5C2-94F6EF79AFD4
  </Rendition>
</Renditions>
```

- Rückgabe der eindeutigen Dokumentobjekt-ID
- Diese ID kennzeichnen das abgelegte Dokumentobjekt

#### 10.4.2 Eigenschaften

```
<Properties>
  <Property name="KDNR">
    <Value>4711</Value>
    <Value>4712</Value>
  </Property>
  ...
</Properties>
```

- Es werden sämtliche definierte Dokument-Eigenschaften des gespeicherten Dokuments zurückgeliefert

# 11 Dokumentsuche

(SEARCH/SEARCHANDLIST)

Mit diesem Befehl können Dokumente gesucht und über das mitgelieferte API-Programm dargestellt werden.

## 11.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
<APICall>SEARCH|SEARCHANDLIST</APICall>
<APIVersion>1.0</APIVersion>
<APILanguage>de-CH</APILanguage>
<APIMultikeyDelimiter>;</APIMultikeyDelimiter>
<!-- <APIDebug>yes</APIDebug> -->
<DocumentSearch>
  <Request>
    <SearchConditions>
      <Properties>
        <Condition property="Dokumentart" coperator="eq" relation="AND">Rechnungen
        </Condition>
        <Condition property="Kundenname" coperator="eq" relation="AND">Kendox
        </Condition>
      </Properties>
      <!-- <FullText>
        <Word>InfoShare.Client</Word>
      </FullText> -->
    </SearchConditions>
    <SearchOptions>
      <Language>ALL</Language>
      <MaxDocuments>1000</MaxDocuments>
      <FullTextOperator>AND</FullTextOperator>
    </SearchOptions>
    <SearchStores>
      <Store>Berlin</Store>
      <Store>Bern</Store>
      <Store>Wien</Store>
    </SearchStores>
    <SearchFolders>
      <Folder>PERSONAL\Rechnungen</Folder>
      <Folder subfolders="yes">PERSONAL\Rechnungen\Suchordner</Folder>
    </SearchFolders>
    <Title>Dokumentsuche</Title>
    <!-- <Grouping>
      <Property>Jahr</Property>
      <Property>Land</Property>
      <Property>Region</Property>
    </Grouping> -->
    <!-- <ExpandResult>true</ExpandResult> -->
    <PropertyList>
      <Property width="80">Jahr</Property>
      <Property width="120">Land</Property>
      <Property>Region</Property>
    </PropertyList>
    <Sorting>
      <Property sort="asc">Jahr</Property>
    </Sorting>
  </Request>
</DocumentSearch>
</XmlAPI>
```

### 11.2 Code-Definitionen der Anforderung

#### 11.2.1 Suchbedingungen

```
<SearchConditions>
  <Properties>
    <Condition property="Belegnummer" coperator="eq" relation="OR">4711</Condition>
    ...
  </Properties>
  <FullText>
    <Word>Rechnung</Word>
  </FullText>
</SearchConditions>
```

- Dokumentsuchen können über die Eigenschaften wie auch über Volltext, Synonym und Thematic vorgenommen werden
- Je nach Operator müssen keine, ein oder mehrere <Value>-Tags angegeben werden; werden zu viele <Value>Tags übergeben, werden die überzähligen Tags ignoriert
- Mit dem optionalen Tag „Fulltext“ kann die Volltextsuche mit einem definierten Wort aktiviert werden

Abkürzung	Englisch	Deutsch	<Value> Tag
<b>eq</b>	equal	gleich	1
<b>gt</b>	greater (than)	größer	1
<b>ge</b>	greater / equal	größer / gleich	1
<b>le</b>	less / equal	kleiner / gleich	1
<b>lt</b>	less (than)	kleiner	1
<b>ne</b>	not equal	ungleich	1
<b>lk</b>	like	wie	1
<b>ct</b>	contains	enthält	1
<b>bt</b>	between	zwischen	2 (von – bis)
<b>un</b>	undefined	Undefiniert	0
<b>in</b>	value list	Werteliste	1 bis n
<b>ni</b>	Not in value list	Ungleich Werteliste	1 bis n
<b>sx</b>	soundex	Soundex	1

#### 11.2.2 Suchoptionen

```
<SearchOptions>
  ...
</SearchOptions>
```

- In diesem Tag werden die Suchoptionen definiert (siehe folgende Tags)

#### 11.2.3 Suchsprache

```
<Language>DE</Language>
```

- Dieses Tag definiert, welche Sprache(n) bei der Suche unterstützt werden soll(en)
- Mögliche Werte: „DE“, „FR“, „IT“, „EN“ und „ALL“

#### 11.2.4 Maximale Suchtreffer

```
<MaxDocuments>1000</MaxDocuments>
```

- Angabe, wie viele Treffer zurückgegeben werden sollen
- Weitere Dokumente können mit „SEARCHNEXT“ abgerufen werden

#### 11.2.5 Volltext-Operator

```
<FullTextOperator>AND</FullTextOperator>
```

- Hier wird definiert, wie die Werte innerhalb des Volltext-Tags verknüpft werden
- Mit „AND“ werden sie UND-verknüpft; mit „OR“ ODER-verknüpft

### 11.2.6 Dokumentablagen

```
<SearchStores>  
  <Store>Berlin</Store>
```

...

```
</SearchStores>
```

- Dokumentablagen, welche in die Suche mit einbezogen werden sollen

### 11.2.7 Suchordner

```
<SearchFolders>  
  <Folder>PERSONAL\Rechnungen</Folder>  
  <Folder subfolders="yes">PERSONAL\Rechnungen\Suchordner</Folder>
```

...

```
</SearchFolders>
```

- Angabe der Ordner, in denen die Dokumente gesucht werden sollen
- Werden keine Ordner angegeben, wird das gesamte InfoShare-System durchsucht

### 11.2.8 Fensterüberschrift

```
<Title>Dokumentsuche</Title>
```

- Titel für die Fensterüberschrift
- Dieses XML-Tag wird nur bei „SEARCHANDLIST“ berücksichtigt

### 11.2.9 Gruppierung im Resultatdialog

```
<Grouping>  
  <Property>Jahr</Property>  
  <Property>Land</Property>  
  <Property>Region</Property>
```

```
</Grouping>
```

- Die Gruppierung wird nur berücksichtigt, wenn eine Suche mit Resultatdialog ausgeführt wird; d. h. dass dieses XML-Tag nur bei „SEARCHANDLIST“ berücksichtigt wird
- Es sind maximal drei Stufen möglich

### 11.2.10 Gruppierungsergebnis erweitern

```
<ExpandResult>true</ExpandResult>
```

- Bei Angabe von Gruppierungsspalten kann mit diesem Tag gesteuert werden, ob das Ergebnis erweitert dargestellt werden soll (true) oder nicht (false)

### 11.2.11 Eigenschaftsliste

```
<PropertyList>  
  <Property width="80">Jahr</Property>
```

...

```
</PropertyList>
```

- Angabe der Eigenschaften, die in der Trefferliste dargestellt resp. in der Rückgabeantwort geliefert werden sollen
- Die Anzahl der Angaben der Eigenschaftsfelder kann beliebig gross gewählt werden
- Mit „width=“ wird die Breite der Spalte definiert; ist die Breite nicht definiert, so wird sie automatisch angepasst

### 11.2.12 Sortierung der Trefferliste

```
<Sorting>
```

```
  <Property sort="asc">Jahr</Property>
```

```
</Sorting>
```

- Angabe, wie die Trefferliste sortiert dargestellt resp. in der Rückgabeantwort geliefert werden soll
- Mit „asc“ werden die Treffer aufsteigend, mit „desc“ absteigend aufgelistet
- Bei der normalen Suche funktioniert dieses Tag nicht, da bei der Suchanfrage keine Eigenschaft mitgegeben werden kann, nach dem die Suche sortiert werden kann

## 11.3 Antwort

### 11.3.1 Antwort bei SEARCH

```
<?xml version="1.0" encoding="UTF-16"?>
```

```
<XmlAPI>
```

```
  <DocumentSearch>
```

```
    <Return>
```

```
      <ResultList>
```

```
        <Columns>
```

```
          <Column typeid="01001000-38a7-11d8-a902-0050bf427e8f" type="String" name="Dokumentart" align="left">Dokumentart</Column>
```

```
          <Column typeid="01001010-38a7-11d8-a902-0050bf427e8f" type="String" name="Name" align="left">Name</Column>
```

```
        </Columns>
```

```
        <DocumentsFound>2</DocumentsFound>
```

```
        <NumberOfDocumentsReturned>2</NumberOfDocumentsReturned>
```

```
        <HasMore>False</HasMore>
```

```
        <Rows>
```

```
          <Row>
```

```
            <Document ID="7b808f6a-1b3b-11e1-a8c5-000c29b4469b"
```

```
            StoreID="9e30250d-e4f9-11e0-bd71-000c29b4469b"
```

```
            ProtectionDomain="Standardsicherheit"
```

```
            OriginalFilename="Rechnung4711.pdf" HasSignature="no"
```

```
            checkedout="false">Rechnung4711</Document>
```

```
            <Property name="Dokumentart" type="" plugin="String"
```

```
            multikey="true">Rechnungen</Property>
```

```
            <Property name="Rechnung4711" type="" plugin="String"
```

```
            multikey="false">Rechnung4711.pdf</Property>
```

```
          </Row>
```

```
          <Row>
```

```
            <Document ID="10942955-007f-11e1-a319-005056c00008"
```

```
            StoreID="db22a820-e4f5-11e0-b827-00155d051700"
```

```
            ProtectionDomain="Standardsicherheit"
```

```
            OriginalFilename="Rechnung4712.docx" HasSignature="no"
```

```
            checkedout="false">Rechnung4712</Document>
```

```
            <Property name="Dokumentart" type="" plugin="String"
```

```
            multikey="true">Test</Property>Rechnungen<Property name="Name"
```

```
            type="" plugin="String" multikey="false">Rechnung4712.docx
```

```
            </Property>
```

```
          </Row>
```

```
        </Rows>
```

```
      </ResultList>
```

```
    </Return>
```

```
  </DocumentSearch>
```

```
</XmlAPI>
```

### 11.3.2 Antwort bei SEARCHANDLIST

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <DocumentSearch>
    <Return />
  </DocumentSearch>
</XmlAPI>
```

## 11.4 Code-Definitionen der Antwort

### 11.4.1 Datenspalten (Columns)

```
<Columns>
  <Column typeid="01001000-38a7-11d8-a902-0050bf427e8f" type="String"
    name="Dokumentart" align="left">Dokumentart</Column>
  <Column typeid="01001010-38a7-11d8-a902-0050bf427e8f" type="String"
    name="Name" align="left">Name</Column>
</Columns>
```

- Angabe der einzelnen Datenspalten, die zurückgeliefert werden
- Die Kolonnenangaben enthalten die Ausrichtung, sowie die eindeutige Identifikation des Format-Plug-Ins
- Die Beschreibung zur Kolonne wird ebenfalls zurückgeliefert

### 11.4.2 Anzahl gefundener Dokumente

```
<DocumentsFound>2</DocumentsFound>
```

- Anzahl gefundener Dokumente zur angegebenen Suche

### 11.4.3 Anzahl zurückgelieferter Treffer

```
<NumberOfDocumentsReturned>2</NumberOfDocumentsReturned>
```

- Anzahl der Treffer, welche in XML zurückgeliefert werden

### 11.4.4 Weitere Dokumente

```
<HasMore>False</HasMore>
```

- Angabe, ob weitere Dokumente gefunden wurden (true) oder nicht (false)

### 11.4.5 Datenreihen

```
<Rows>...</Rows>
```

- Dataset mit den Dokumentauflistungen

### 11.4.6 Datensätze

```
<Row>
  <Document ID="7b808f6a-1b3b-11e1-a8c5-000c29b4469b"
    StoreID="9e30250d-e4f9-11e0-bd71-000c29b4469b" ProtectionDomain="Standardsicherheit"
    OriginalFilename="Rechnung4711.pdf" HasSignature="no" checkedout="false">
    Rechnung4711
  </Document>
  <Property name="Dokumentart" type="" plugin="String" multikey="true">
    Rechnungen
  </Property>
  <Property name="Rechnung4711" type="" plugin="String" multikey="false">
    Rechnung4711.pdf
  </Property>
</Row>
```

- Je Dokument wird eine Zeile (Row) erstellt; diese <Row> enthält die eindeutige Dokument-ID („<ID>“), die eindeutige Ablagekennung („<StoreID>“), Ausbuchungsstatus („<checked-out>“), Dokumentname und die dazugehörigen Eigenschaften

## 12 SearchAndListForm im Hintergrund laden

*(LOADSEARCHFORM)*

Lädt ein „leeres“ SearchAndList Formular im Hintergrund. Somit muss beim Befehl „SEARCHANDLIST“ nur noch die Suche ausgeführt und im Formular dargestellt werden. Dadurch wird die Ladezeit beim Befehl „SEARCHANDLIST“ erheblich verringert.

### 12.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>LOADSEARCHFORM</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
</XmlAPI>
```

### 13 Spalteneinstellungen für alle Benutzer zurücksetzen

*(RESETLISTVIEWCONFIGS)*

Löscht die lokalen Einstellungen der Trefferliste für alle Benutzer.

#### 13.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>RESETLISTVIEWCONFIGS</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug -->
</XmlAPI>
```

## 14 Dokumentsuche fortsetzen

### (SEARCHNEXT)

Bei der Funktion „SEARCH“ kann angegeben werden, dass nicht alle Treffer einer Suche zurückgegeben werden sollen („<MaxDocumentsReturn>“), was sich erheblich auf die Performance auswirkt. Mit „SEARCHNEXT“ können weitere Treffer vom Server geholt werden. Somit lässt sich ein einfaches Paging implementieren.

### 14.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>SEARCHNEXT</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <DocumentSearchNext>
    <Request>
      <SearchOptions>
        <MaxDocumentsReturn>50</MaxDocumentsReturn>
      </SearchOptions>
    </Request>
  </DocumentSearchNext>
</XmlAPI>
```

### 14.2 Code-Definition der Anforderung

#### 14.2.1 Maximal zurückgegebene Treffer

```
<MaxDocumentsReturn>50</MaxDocumentsReturn>
```

- Angabe, wie viele Treffer zurückgegeben werden sollen
- Werden keine weiteren Dokumente mehr gefunden, wird eine leere Resultatliste zurückgegeben

### 14.3 Antwort

Der Rückgabeparameter entspricht dem der Funktion „SEARCH“.

## 15 Eigenschaften ändern

(UPDATEPROPERTIES)

Mit dieser Funktion können Dokumenteigenschaften aktualisiert oder gelöscht werden.

### 15.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>UPDATEPROPERTIES</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <DocumentUpdProperties>
    <Request>
      <DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
      <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
      <Properties>
        <Property name="Betreff" add="yes">
          <Value>Rechnung4711</Value>
        </Property>
        <Property name="Kundenname">
          <Value>Kendox AG</Value>
        </Property>
        <Property name="Bearbeiter" add="yes">
          <Value>John Bünzli</Value>
          <Value>Gian Giacometti</Value>
          <Value>Fritz Müller</Value>
        </Property>
      </Properties>
      <Folders>
        <Folder>PERSONAL\Rechnungen</Folder>
        <Folder>Dokumente\Rechnungen</Folder>
      </Folders>
      <ProtectionDomain>
        <Domain>Standardsicherheit</Domain>
      </ProtectionDomain>
      <!-- <LifeCycle>myLifeCycle</LifeCycle> -->
    </Request>
  </DocumentUpdProperties>
</XmlAPI>
```

## 15.2 Code-Definitionen der Anforderung

### 15.2.1 Eigenschaften

<Properties>...</Properties>

- Das Übergabeformat der Dokumenteigenschaften richtet sich nach den Angaben gemäss Eigenschaftstypen-Plug-Ins
- Als Parameter müssen der „Property name“ und dessen Werte angegeben werden
- Standardmässig werden alle Eigenschaften ersetzt
- Mit dem Parameter add="yes" können zu einer bestehenden Eigenschaft Werte hinzugefügt werden

### 15.2.2 Ordner

<Folders>

<Folder>PERSONAL\Rechnungen</Folder>

<Folder>Dokumente\Rechnungen</Folder>

</Folders>

- Ordnerzuweisungs-Tag
- Das Dokument kann so den entsprechenden Ordnern zugewiesen werden; ist ein Dokument bereits in diesem Ordner vorhanden, wird die alte Referenz gelöscht

### 15.2.3 Sicherheitsbereich

<ProtectionDomain>

<Domain>Standardsicherheit</Domain>

</ProtectionDomain>

- Rechtevergabe zum Dokument

### 15.2.4 Lebenszyklus

<LifeCycle>myLifeCycle</LifeCycle>

- Lebenszyklus für das Dokument setzen

# 16 Dokumenteigenschaften auslesen

## (GETPROPERTIES)

Mit dieser Funktion können Dokumenteigenschaften ausgelesen werden.

### 16.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETPROPERTIES</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <DocumentGetProperties>
    <Request>
      <DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
      <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
    </Request>
  </DocumentGetProperties>
</XmlAPI>
```

### 16.2 Antwort

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <DCISDocumentGetProperties>
    <Return>
      <PhysicalPath>
        db22a931-e4f5-11e0-b827-00155d051700\2011\09\22\000\000\214Rechnung4711.pdf
      </PhysicalPath>
      <Properties>
        <Property name="Änderungszähler">
          <Value>1</Value>
          <Category>Standard</Category>
        </Property>
        <Property name="Version">
          <Value>1</Value>
          <Category>Standard</Category>
        </Property>
        <Property name="Lebenszyklus Zustand">
          <Category>Standard</Category>
        </Property>
        ...
        <Property name="Letztes Einbuchen">
          <Value>09.12.2010 13:43</Value>
          <Category />
        </Property>
        <Property name="Interner Name">
          <Value>Rechnung4711.pdf</Value>
          <Category>Standard</Category>
        </Property>
        <Property name="Ausgebucht von">
          <Value />
          <Category>Standard</Category>
        </Property>
      </Properties>
    </Return>
  </DCISDocumentGetProperties>
</XmlAPI>
```

### 16.3 Code-Definition der Antwort

#### 16.3.1 Physischer Pfad

```
<PhysicalPath>  
  db22a931-e4f5-11e0-b827-00155d051700\2011\09\22\000\000\214Rechnung4711.pdf  
</PhysicalPath>
```

- Liefert den Pfad zum physischen Dokument
- Der Root-Ordner, in dem sich die einzelnen Unterordner für die Ablagen befinden, kann der Datei „config.xml“ aus dem InfoShare.Server-Verzeichnis entnommen werden

#### 16.3.2 Eigenschaften-Container

```
<Properties>...</Property>
```

- Dieser Container enthält alle zurückgegebenen Eigenschaften

#### 16.3.3 Eigenschaften

```
<Property name="Änderungszähler">  
  <Value>1</Value>  
  <Category>Standard</Category>  
</Property>
```

- „name=...“ ist der Name des Eigenschaftstyps in der Sprache, welche im Tag <APILanguage>de-CH</APILanguage> definiert wurde
- „Value“ ist der Wert der Eigenschaft
- „Category“ ist die Eigenschaftskategorie

## 17 Eigenschaftstypen erstellen

### (CREATEPROPERTYTYPE)

Mit dieser Funktion können neue Eigenschaftstypen im InfoShare-System angelegt werden.

### 17.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>CREATEPROPERTYTYPE</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <CreatePropertyType>
    <Request>
      <PropertyTypeName>Lieferscheinnummer</PropertyTypeName>
      <FieldType>text</FieldType>
      <!-- <Category>Standard</Category> -->
      <ViewFormat>Text</ViewFormat>
      <InputFormat>Text</InputFormat>
      <IsMultikey>>false</IsMultikey>
      <IsSearchable>>true</IsSearchable>
      <IsActive>>true</IsActive>
      <!-- Additional information for field type „Text“ -->
      <MaxLength>200</MaxLength>
      <ForbiddenSigns>a;b;c;d</ForbiddenSigns>
      <CaseSensitiv>>false</CaseSensitiv>
      <!-- Additional information for field type „Number“ -->
      <NumberOfDigits>10</NumberOfDigits>
      <DigitsAfterDecimalPoint>2</DigitsAfterDecimalPoint>
      <InternalDigits>15</InternalDigits>
      <InternalDigitsAfterDecimalPoint>5</InternalDigitsAfterDecimalPoint>
      <!-- Additional information for field type „Date“ or „Long date“ -->
      <FormatInfo>dd.MM.yyyy</FormatInfo>
    </Request>
  </CreatePropertyType>
</XmlAPI>
```

### 17.2 Code-Definitionen der Anforderung

#### 17.2.1 Eigenschaftsname

```
<PropertyName>Lieferscheinnummer</PropertyName>
```

- Name des zu erstellenden Eigenschaftstyps

#### 17.2.2 Feldtyp

```
<FieldType>text</FieldType>
```

- Angabe des Feldtyps
- Mögliche Werte: text, number, date, longdate
- Die weiteren Angaben unterscheiden sich je nach hier angeführtem Feldtyp

#### 17.2.3 Kategorie

```
<Category>Standard</Category>
```

- Dieses optionale Tag ordnet den Eigenschaftstyp der entsprechenden Kategorie zu

#### 17.2.4 View-Format

```
<ViewFormat>Text</ViewFormat>
```

- Die Werte des Eigenschaftstyps werden in diesem Format dargestellt

# KENDOX XmlAPI

## Handbuch

### 17.2.5 Eingabeformat

<InputFormat>Text</InputFormat>

- Darstellung, während dem ein Wert editiert wird

### 17.3 Antwort

```
<?xml version="1.0" encoding="UTF-16" ?>
```

```
<XmlAPI>
```

```
  <CreateProperty>
```

```
    <Return>
```

```
      <PropertyTypeId>973a2b3e-200f-11e1-9b1f-000c29b4469b</PropertyTypeId>
```

```
      <PropertyName>Lieferscheinnummer</PropertyName>
```

```
    </Return>
```

```
  </CreateProperty>
```

```
</XmlAPI>
```

## 18 Eigenschaftstypen löschen

### (DELETEPROPERTYTYPE)

Mit dieser Funktion können im System angelegte Eigenschaftstypen gelöscht werden.

#### 18.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>DELETEPROPERTYTYPE</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <DeletePropertyType>
    <Request>
      <PropertyTypeName>Lieferscheinnummer</PropertyTypeName>
    </Request>
  </DeletePropertyType>
</XmlAPI>
```

#### 18.2 Code-Definition der Anforderung

##### 18.2.1 Eigenschaftsname

```
<PropertyTypeName>Lieferscheinnummer</PropertyTypeName>
```

- Name des zu löschenden Eigenschaftstyps

#### 18.3 Antwort

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <DeletePropertyType>
    <Return>
      <PropertyTypeId>973a2b3e-200f-11e1-9b1f-000c29b4469b</PropertyTypeId>
      <PropertyName>Lieferscheinnummer</PropertyName>
    </Return>
  </DeletePropertyType>
</XmlAPI>
```

#### 18.4 Code-Definitionen der Antwort

##### 18.4.1 Eigenschaftsname

```
<PropertyTypeId>973a2b3e-200f-11e1-9b1f-000c29b4469b</PropertyTypeId>
```

- ID des gelöschten Eigenschaftstyps

##### 18.4.2 Eigenschaftsname

```
<PropertyName>Lieferscheinnummer</PropertyName>
```

- Name des gelöschten Eigenschaftstyps

## 19 Dokumente exportieren

### (EXPORTRENDITION)

Mit dieser Funktion können Dokumentobjekte (Renditions) exportiert werden.

#### 19.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>EXPORTRENDITION</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <DocumentExportRendition>
    <Request>
      <DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
      <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
      <Renditions>
        <Rendition>C:\Export\Renditions\R4711.pdf</Rendition>
      </Renditions>
      <!-- <ExportSignatures>Yes</ExportSignatures> -->
      <!-- <ExportAnnotations>Yes</ExportAnnotations> -->
      <!-- <ExportParts>Yes</ExportParts> -->
    </Request>
  </DocumentExportRendition>
</XmlAPI>
```

#### 19.2 Code-Definitionen der Anforderung

##### 19.2.1 Dokument-ID

```
<DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
```

- Angabe der eindeutigen Dokumentkennung (zwingend), von welcher die Dokumentobjekte exportiert werden sollen
- Es können mehrere Objekte gleichzeitig exportiert werden

##### 19.2.2 Renditions

```
<Renditions>
  <Rendition>C:\Export\Renditions\R4711.pdf</Rendition>
</Renditions>
```

- Angaben der einzelnen Renditions, die exportiert werden sollen; „DCI.SERVER“ wird zurzeit nicht beachtet
- Der Export wird in das definierte Verzeichnis gemacht
- ISRWs: Der "name" wird nicht beachtet; das Dokument wird aber in das angegebene Verzeichnis exportiert
- Mit "original-filename="true/false" kann bestimmt werden, ob das zu exportierende Dokument mit dem Original-Dateinamen exportiert wird oder nicht

##### 19.2.3 Signaturen exportieren

```
<ExportSignatures>Yes</ExportSignatures>
```

- Exportiert zusätzlich alle vorhandenen Signaturen

##### 19.2.4 Annotationen exportieren

```
<ExportAnnotations>Yes</ExportAnnotations>
```

- Mischt beim Export die Annotationen in das Dokument ein

### 19.3 Antwort

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <DocumentExportRendition>
    <Return>
      <Renditions>
        <Rendition ID="d8918458-e517-11e0-bd71-000c29b4469b" Export-
          Path="C:\Export\Renditions\R4711.pdf" />
      </Renditions>
    </Return>
  </DocumentExportRendition>
</XmlAPI>
```

### 19.4 Code-Definitionen der Antwort

#### 19.4.1 Rendition

`<Rendition ... />`

- Pro exportierter Rendition wird ein „<Rendition>“-Eintrag erstellt

#### 19.4.2 Rendition-ID

`ID="d8918458-e517-11e0-bd71-000c29b4469b"`

- Eindeutige ID der exportierten Renditions

#### 19.4.3 Exportpfad

`ExportPath="C:\Export\Renditions\R4711.pdf"`

- Dateiname inklusive Export-Pfad, wo die Rendition-Datei abgelegt wurde

## 20 Dokumente verknüpfen

(LINKDOCUMENTS)

Mit dieser Funktion können mehrere Dokumentobjekte miteinander verknüpft werden.

### 20.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>LINKDOCUMENTS</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <LinkDocuments>
    <Request>
      <DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
      <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
      <DocumentLinkList>
        <Document>
          <ID>08c28af6-1f54-11e1-af56-005056c00008</ID>
          <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
        </Document>
        <Document>
          <ID>d89182b6-e517-11e0-bd71-000c29b4469b</ID>
          <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
        </Document>
        <Document>
          <ID>388ba221-e522-11e0-bd71-000c29b4469b</ID>
          <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
        </Document>
      </DocumentLinkList>
    </Request>
  </LinkDocuments>
</XmlAPI>
```

### 20.2 Code-Definition der Anforderung

#### 20.2.1 Dokument-Linkliste

```
<DocumentLinkList>
  <Document>
    <ID>08c28af6-1f54-11e1-af56-005056c00008</ID>
    <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID>
  </Document>
  ...
</DocumentLinkList>
```

- In dieser Dokumentliste werden alle Dokumente definiert, welche mit dem oben definierten Dokument verknüpft werden sollen

## 21 Dokumentverknüpfung löschen

(DELETEDOCUMENTLINK)

Mit dieser Funktion können Dokumentverknüpfungen gelöscht werden.

### 21.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>DELETEDOCUMENTLINKS</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <DeleteDocumentLinks>
    <Request>
      <DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
      <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
    </Request>
  </DeleteDocumentLinks>
</XmlAPI>
```

## 22 Propertytypen auflisten

(GETPROPERTYTYPES)

Mit dieser Funktion können alle Eigenschaftstypen ausgelesen werden. Diese Funktion wird nur vom DCI.Server unterstützt.

### 22.1 Anforderung

Mit der Variante 1 werden die Daten per XmlAPI zurückgegeben. Mit der Variante 2 können die Eigenschaftstypen in einer CSV-Datei gespeichert werden.

#### 22.1.1 Variante 1

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETPROPERTYTYPES</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <!-- <GetPropertyTypes>
    <Request>
      </Request>
    </GetPropertyTypes> -->
</XmlAPI>
```

#### 22.1.2 Variante 2

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETPROPERTYTYPES</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <GetPropertyTypes>
    <Request>
      <FileName>C:\Data\PropertyTypes.csv</FileName>
    </Request>
  </GetPropertyTypes>
</XmlAPI>
```

### 22.2 Code-Definition der Anforderung

Es muss keine spezielle Anfrageanforderung übergeben werden. Optional kann in der Anforderung ein Dateipfad angegeben werden. Wird dies gemacht, wird eine CSV-Datei mit einer Auflistung der Eigenschaften erstellt. In der Antwort-XML gibt es jedoch keine Auflistung mehr.

#### 22.2.1 Speicherpfad

```
<FileName>C:\Data\PropertyTypes.csv</FileName>
```

- Ablagepfad (inklusive Dateiname), wo die CSV-Datei, welche alle Eigenschaftstypen enthält, abgelegt werden soll

## 22.3 Antwort

### 22.3.1 Variante 1

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <GetPropertyTypes>
    <Return>
      <PropertyTypeList>
        <PropertyType typeid="01001003-38a7-11d8-a902-0050bf427e8f" name="Erstellt"
searchable="true" multikey="false">Erstellt</PropertyType>
        <PropertyType typeid="01001003-38a7-11d8-a902-0050bf427e8f" name="Geändert"
searchable="true" multikey="false">Geändert</PropertyType>
        <PropertyType typeid="01001000-38a7-11d8-a902-0050bf427e8f" name="Dateityp"
searchable="true" multikey="true">Dateityp</PropertyType>
        <PropertyType typeid="01001010-38a7-11d8-a902-0050bf427e8f" name="Name"
searchable="false" multikey="false">Name</PropertyType>
      </PropertyTypeList>
    </Return>
  </GetPropertyTypes>
</XmlAPI>
```

### 22.3.2 Variante 2

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <GetPropertyTypes>
    <Return />
  </GetPropertyTypes>
</XmlAPI>
```

## 22.4 Code-Definitionen der Antwort (Variante 1)

### 22.4.1 Eigenschaftstyp

```
<PropertyType PropertyId="01000000-38a7-11d8-a902-0050bf427e8f" Type="Date" TypeId="01001003-38a7-11d8-a902-0050bf427e8f" IsSearchable="True" IsActive="True" IsObsolete="False" IsMulti="False">Erstellt</PropertyType>
```

- Pro definiertem Eigenschaftstyp wird ein „<PropertyType>“-Eintrag erstellt
- Es werden vom Eigenschaftstypen die ID (PropertyId), der Datentyp (Type) und die dazugehörige ID (TypeId), die Definition ob der Eigenschaftstyp suchbar (IsSearchable) sowie aktiviert (IsActive) → d.h. nicht deaktiviert ist, die Angabe ob das Feld Multikey ist sowie der Eigenschaftstyp-Name zurückgegeben
- Wurde ein Eigenschaftstyp in InfoShare gelöscht, sind die zurückgegebenen Werte von IsActive="False" und IsObsolete="False"
- Wurde ein Eigenschaftstyp in InfoShare deaktiviert, sind die zurückgegebenen Werte von IsActive="True" und IsObsolete="True"

## 23 Property-Typen mit Sprachen auflisten

### (GETPROPERTYTYPEESLANGUAGES)

Mit dieser Funktion können alle Eigenschaftstypen inklusive aller hinterlegten Sprachen ausgelesen werden. Diese Funktion wird nur vom DCI.Server unterstützt.

### 23.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETPROPERTYTYPEESLANGUAGES</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
</XmlAPI>
```

### 23.2 Antwort

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <GetPropertyTypes>
    <Return>
      <PropertyTypeList>
        <PropertyType typeid="01000000-38a7-11d8-a902-0050bf427e8f">
          <PropertyTypeNames>
            <de>Erstellt</de>
            <en>Created</en>
            <fr>Crée</fr>
          </PropertyTypeNames>
        </PropertyType>
        <PropertyType typeid="01000001-38a7-11d8-a902-0050bf427e8f">
          <PropertyTypeNames>
            <de>Geändert</de>
            <en>Modified</en>
            <fr>Modifié</fr>
          </PropertyTypeNames>
        </PropertyType>
        <PropertyType typeid="01004008-38a7-11d8-a902-0050bf427e8f">
          <PropertyTypeNames>
            <de>Dateityp</de>
            <en>File type</en>
            <fr>Type de fichier</fr>
          </PropertyTypeNames>
        </PropertyType>
        ...
      </PropertyTypeList>
    </Return>
  </GetPropertyTypes>
</XmlAPI>
```

## 24 InfoStores auflisten

(GETINFOSTORES)

Mit dieser Funktion können alle definierten InfoShare-Ablagen ausgelesen werden. Diese Funktion wird nur vom DCI.Server unterstützt.

### 24.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETINFOSTORES</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <GetInfoStores>
    <Request>
      </Request>
    </GetInfoStores>
  </XmlAPI>
```

### 24.2 Code-Definitionen der Anforderung

Es muss keine spezielle Anfrageanforderung übergeben werden.

### 24.3 Antwort

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <GetInfoStores>
    <Return>
      <InfoStoreList>
        <InfoStore name="Standard">Standard</InfoStore>
        <InfoStore name="Berlin">Berlin</InfoStore>
        <InfoStore name="Bern">Bern</InfoStore>
        <InfoStore name="Geschäftsleitung">Geschäftsleitung</InfoStore>
        <InfoStore name="KxShare">KxShare</InfoStore>
        <InfoStore name="Personaldaten">Personaldaten</InfoStore>
        <InfoStore name="Wien">Wien</InfoStore>
      </InfoStoreList>
    </Return>
  </GetInfoStores>
</XmlAPI>
```

### 24.4 Code-Definitionen der Antwort

#### 24.4.1 InfoStore

```
<InfoStore name="Standard">Standard</InfoStore>
```

- Pro definierter Ablage wird ein „<InfoStore>“-Eintrag erstellt
- Im Wert wird der Name der Ablage zurückgegeben

## 25 Eigenschaftstypvorlagen auflisten

(*GETPROPERTYTYPETEMPLATES*)

Mit dieser Funktion können alle Eigenschaftstypvorlagen und die dazugehörigen Eigenschaftstypen ausgelesen werden. Diese Funktion wird nur vom DCI.Server unterstützt.

### 25.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETPROPERTYTYPETEMPLATES</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
</XmlAPI>
```

### 25.2 Code-Definitionen der Anforderung

Es muss keine spezielle Anfrageanforderung übergeben werden.

### 25.3 Antwort

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <GetPropertyTypeTemplates>
    <Return>
      <PropertyTypeTemplateList>
        <PropertyTypeTemplate name="Personalakte">
          <PropertyTypeList>
            <PropertyType typeid="01001000-38a7-11d8-a902-0050bf427e8f" type=""
              name="Belegart" searchable="true" multikey="false" required="true"
              visible="true" canEdit="true" constVal="">Nachname</PropertyType>
            <PropertyType typeid="01001001-38a7-11d8-a902-0050bf427e8f" type=""
              name="Personalnummer" searchable="true" multikey="false" required="true"
              visible="true" canEdit="true" constVal="">Personalnummer</PropertyType>
            ...
          </PropertyTypeList>
        </PropertyTypeTemplate>
        <PropertyTypeTemplate name="Rechnungen">
          <PropertyTypeList>
            <PropertyType typeid="01001000-38a7-11d8-a902-0050bf427e8f" type=""
              name="Belegart" searchable="true" multikey="false" required="true"
              visible="true" canEdit="true" constVal="">KdNr</PropertyType>
            <PropertyType typeid="01001001-38a7-11d8-a902-0050bf427e8f" type=""
              name="Personalnummer" searchable="true" multikey="false" required="true"
              visible="true" canEdit="true" constVal="">Firma</PropertyType>
            ...
          </PropertyTypeList>
        </PropertyTypeTemplate>
      </PropertyTypeTemplateList>
    </Return>
  </GetPropertyTypeTemplates>
</XmlAPI>
```

## 25.4 Code-Definitionen der Antwort

### 25.4.1 Liste der Eigenschaftstypvorlagen

`<PropertyTypeTemplateList>...</PropertyTypeTemplateList>`

- In diesem Container werden die Eigenschaftstypvorlagen zusammengefasst

### 25.4.2 Eigenschaftstypvorlagen

`<PropertyTypeTemplate name="Rechnungen">`

- Für jede Eigenschaftstypvorlage in InfoShare wird ein Eintrag (Container) erstellt
- Das Tag „name“ definiert dabei den für die Benutzer sichtbaren Eigenschaftstypvorlagen-Name

### 25.4.3 Eigenschaftstypvorlagen-Container

`<PropertyTypeList>...</PropertyTypeList>`

- In diesem Container werden die Eigenschaftstypvorlagen-Attribute zusammengefasst

### 25.4.4 `<PropertyType>`

`<PropertyType typeid="01001000-38a7-11d8-a902-0050bf427e8f" type="" name="Belegart" searchable="true" multikey="false" required="true" visible="true" canEdit="true" constVal="">Nachname</PropertyType>`

- In diesem Tag werden die einzelnen Eigenschaftstypen und deren Attribute innerhalb der Eigenschaftstypvorlage angezeigt

## 26 Importvorlagen auflisten

(GETIMPORTTEMPLATES)

Mit dieser Funktion können alle Importvorlagen und die dazugehörigen Eigenschaften ausgelesen werden. Diese Funktion wird nur vom DCI.Server unterstützt.

### 26.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETIMPORTTEMPLATES</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
</XmlAPI>
```

### 26.2 Code-Definitionen der Anforderung

Es muss keine spezielle Anfrageanforderung übergeben werden.

### 26.3 Antwort

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <GetImportTemplates>
    <Return>
      <ImportTemplateList>
        <ImportTemplate ID="aeb91289-e4fb-11e0-bd71-000c29b4469b" Name="Personalakte">
          <PropertyTypeTemplate>Personalakte</PropertyTypeTemplate>
          <ProtectionDomain CanChange="False">PersonaldatenAdmin</ProtectionDomain>
          <Store CanChange="False">Personaldaten</Store>
          <SigningProfile CanChange="True" />
          <LifeCycle CanChange="True" />
          <Process CanChange="True" />
          <StartProcess>False</StartProcess>
          <CanChangePropertiesCollection>True</CanChangePropertiesCollection>
        </ImportTemplate>
        <ImportTemplate ID="ca03aece-e4fc-11e0-bd71-000c29b4469b" Name="Rechnungen">
          <PropertyTypeTemplate>Rechnungen</PropertyTypeTemplate>
          <ProtectionDomain CanChange="True">Standardsicherheit</ProtectionDomain>
          <Store CanChange="False">Standard</Store>
          <SigningProfile CanChange="False">Signatur</SigningProfile>
          <LifeCycle CanChange="True">10 Jahre nicht löschen</LifeCycle>
          <Process CanChange="True">Rechnungsverarbeitung</Process>
          <StartProcess>False</StartProcess>
          <CanChangePropertiesCollection>True</CanChangePropertiesCollection>
        </ImportTemplate>
      </ImportTemplateList>
    </Return>
  </GetImportTemplates>
</XmlAPI>
```

## 26.4 Code-Definitionen der Antwort

### 26.4.1 Importvorlagen-Container

`<ImportTemplateList>...</ImportTemplateList>`

- In diesem Container werden die Importvorlagen zusammengefasst

### 26.4.2 Importvorlagen-ID

`ID="ca03aece-e4fc-11e0-bd71-000c29b4469b"`

- Wert der eindeutigen Importvorlagen-ID

### 26.4.3 Name der Importvorlage

`Name="Rechnungen"`

- Dies ist der Name der Importvorlage, der für die Administratoren und Benutzer überall sichtbar ist

### 26.4.4 Eigenschaftstypvorlage

`<PropertyTypeTemplate>Rechnungen</PropertyTypeTemplate>`

- Definierte Eigenschaftstypvorlage

### 26.4.5 Sicherheitsbereich

`<ProtectionDomain CanChange="True">Standardsicherheit</ProtectionDomain>`

- Hiermit wird angegeben, welcher Sicherheitsbereich verwendet wird
- Mit `CanChange="True"` wird definiert, dass der Benutzer den Sicherheitsbereich ändern darf

### 26.4.6 Ablage

`<Store CanChange="False">Standard</Store>`

- Hiermit wird angegeben, welche Ablage verwendet wird
- Mit `CanChange="True"` wird definiert, dass der Benutzer die Ablage ändern darf

### 26.4.7 Signaturprofil

`<SigningProfile CanChange="False">Signatur</SigningProfile>`

- Hiermit wird angegeben, welches Signaturprofil verwendet wird
- Mit `CanChange="True"` wird definiert, dass der Benutzer das Signaturprofil ändern darf

### 26.4.8 Lebenszyklus

`<LifeCycle CanChange="True">10 Jahre nicht löschen</LifeCycle>`

- Hiermit wird angegeben, welcher Lebenszyklus verwendet wird
- Mit `CanChange="True"` wird definiert, dass der Benutzer den Lebenszyklus ändern darf

### 26.4.9 Vorgang

`<Process CanChange="True">Rechnungsverarbeitung</Process>`

- Hiermit wird angegeben, welcher Vorgang nach dem Import gestartet wird
- Mit `CanChange="True"` wird definiert, dass der Benutzer den Vorgang ändern darf

### 26.4.10 Vorgang starten

`<StartProcess>False</StartProcess>`

- Dieses Tag definiert, ob der definierte Vorgang nach dem Import gestartet wird

### 26.4.11 Eigenschaftsauswahl änderbar

`<CanChangePropertiesCollection>True</CanChangePropertiesCollection>`

- Definiert, ob die Eigenschaftsauswahl für den Benutzer änderbar ist (True) oder nicht (False)

## 27 In eine Datei scannen

(SCANTOFILE)

Mit dieser Funktion kann der Scan-Vorgang (DeskSCAN) ausgelöst werden. Es wird keine Verbindung zum Kendox InfoShare.Server vorausgesetzt.

### 27.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>SCANTOFILE</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <DocumentScan>
    <Request>
      <StartScan>Yes</StartScan>
      <FileName>Z:\XmlAPI\MyScanDoc.pdf</FileName>
      <FileFormat>PDF</FileFormat>
    </Request>
  </DocumentScan>
</XmlAPI>
```

### 27.2 Code-Definitionen der Anforderung

#### 27.2.1 Scan-Vorgang automatisch starten

```
<StartScan>Yes</StartScan>
```

- Wenn in diesem Tag „Yes“, „True“ oder „1“ enthalten ist, dann wird der Scanvorgang direkt nach dem Laden des Scanfensters gestartet

#### 27.2.2 Dateiname

```
<FileName>Z:\XmlAPI\MyScanDoc.pdf</FileName>
```

- Angabe des Dateinamens, in welchem das eingescannte Dokument abgelegt werden soll

#### 27.2.3 Dateiformat

```
<FileFormat>PDF</FileFormat>
```

- Angabe, in welchem Format die Dokumentablage geschehen soll
- Wird keine Angabe gemacht, wird die Ablage im TIFF-Format vorgenommen
- Mögliche Optionen sind „TIF“ oder „PDF“

### 27.3 Antwort

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <DocumentScan>
    <Return>
      <File>Z:\XmlAPI\MyScanDoc.pdf</File>
    </Return>
  </DocumentScan>
</XmlAPI>
```

### 27.4 Code-Definition der Antwort

#### 27.4.1 Datei

`<File>Z:\XmlAPI\MyScanDoc.pdf</File>`

- Im Parameter „File“ wird der Dateiname inklusive Pfad des eingescannten und gespeicherten Dokuments zurückgegeben

## 28 Werte auf Korrektheit prüfen

(CHECKPROPERTIES)

Mit dieser Funktion können Werte eines Eigenschaftstyps auf ihre Gültigkeit geprüft werden. Die Prüfung wird für alle übergebenen Werte vorgenommen. Diese Funktion wird nur vom DCI.Server unterstützt.

### 28.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>CHECKPROPERTIES</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <APIDebug>yes</APIDebug>
  <CheckProperties>
    <Request>
      <Properties>
        <Property name="Kundennummer">
          <Value>4711</Value>
        </Property>
        <Property name="Datum">
          <Value>4711</Value>
          <Value>25.07.2012</Value>
        </Property>
      </Properties>
    </Request>
  </CheckProperties>
</XmlAPI>
```

### 28.2 Code-Definitionen der Anforderung

#### 28.2.1 Eigenschaftstyp

```
<Property name="Kundennummer">
```

- Im Tag „Property“ muss unter „name=“ der Name des Eigenschaftstyps angegeben werden

#### 28.2.2 <Value>

```
<Property name="Datum ">
  <Value>4711</Value>
  <Value>25.07.2012</Value>
</Property>
```

- Angabe der Werte, welche geprüft werden sollen
- Hier sind mehrere zu überprüfende Werte definierbar (siehe Beispiel)

### 28.3 Antwort

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <CheckProperties>
    <Return>
      <Properties>
        <Property name="Kundennummer">
          <Value message="">4711</Value>
        </Property>
        <Property name="Datum">
          <Value message="UnexpectedCharacter">4711</Value>
          <Value message="">25.07.2012</Value>
        </Property>
      </Properties>
    </Return>
  </CheckProperties>
</XmlAPI>
```

### 28.4 Code-Definitionen der Antwort

#### 28.4.1 Nachricht der Überprüfung

```
<Property name="Datum">
  <Value message="UnexpectedCharacter">4711</Value>
  <Value message="">25.07.2012</Value>
</Property>
```

- Im Tag „Property“ wird zusätzlich das Attribut „message=“ zurückgegeben; in diesem Attribut wird im Fehlerfall eine Meldung zurückgegeben
- Mögliche Fehlermeldungen: „TooManyCharacter“ (zu viele Zeichen), „UnexpectedCharacter“ (unerwartete[s] Zeichen)

## 29 Dokument löschen

(DELETE)

Mit dieser Funktion können InfoShare-Dokumente gelöscht werden. Diese Funktion wird nur vom DCI.Server unterstützt.

### 29.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>DELETE</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <DocumentDelete>
    <Request>
      <DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
      <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
    </Request>
  </DocumentDelete>
</XmlAPI>
```

### 29.2 Code-Definitionen der Antwort

#### 29.2.1 Dokument-ID

```
<DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
```

- Eindeutige ID des zu löschenden Dokuments

#### 29.2.2 Ablage-ID

```
<StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID>
```

- Wird die eindeutige Ablage-ID des zu löschenden Dokuments ebenfalls angegeben, wird der Prozess für die angeforderte Funktion beschleunigt; d. h. es werden nicht alle Ablagen nach der angeforderten Dokumenten-ID durchsucht
- Die ID kann aus dem InfoShare.Client bezogen werden („Dokumenteigenschaften“ → Reiter „Allgemein“ → Tag „StoreID“)
- Wird nur mit „DCI.SERVER“ berücksichtigt

## 30 Benutzer auflisten

(GETUSERS)

Mit dieser Funktion können alle Benutzer ausgelesen werden, die angelegt wurden. Diese Funktion wird nur vom DCI.Server unterstützt.

### 30.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETUSERS</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <GetUsers>
    <Request />
  </GetUsers>
</XmlAPI>
```

### 30.2 Code-Definitionen der Anforderung

Es muss keine spezielle Anfrageanforderung übergeben werden.

### 30.3 Antwort

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <GetUsers>
    <Return>
      <UserList>
        <User LogonName="dciadmin" isActive="true" loginWithoutPassword="false"
          loginInternalPassword="true" loginOsAccount="false" showDefaultSearch="true"
          canChangeSettings="true" defaultLanguage="de" emailAdress="admin@kendox.com" force-
          Language="false" emailNotification="true" todoNotification="false">
          DCI administrator
        </User>
        <User LogonName="dciadmin" isActive="true" loginWithoutPassword="false"
          loginInternalPassword="true" loginOsAccount="true" showDefaultSearch="true"
          canChangeSettings="true" defaultLanguage="de" emailAdress="buchhaltung@kendox.com "
          forceLanguage="false" emailNotification="true" todoNotification="false">
          Buchhaltung
        </User>
        ...
      </UserList>
    </Return>
  </GetUsers>
</XmlAPI>
```

### 30.4 Code-Definitionen der Antwort

#### 30.4.1 Benutzer

```
<User...>
  DCI administrator
</User>
```

- Pro definiertem Benutzer wird ein „<User>“-Eintrag aufgelistet
- Im Wert wird der Anzeigename des Benutzers zurückgegeben

#### 30.4.2 Anmeldename

```
LogonName="dciadmin"
```

- Anmeldename für diesen Benutzer

### 30.4.3 Anmeldenname

`isActive="true"`

- Angabe, ob der Benutzer aktiv ist

### 30.4.4 Anmelden ohne Passwort

`loginWithoutPassword="false"`

- Angabe, ob sich der Benutzer ohne Passwort anmelden darf

### 30.4.5 Anmelden mit internem Passwort

`loginInternalPassword="true"`

- Angabe, ob sich der Benutzer mit einem InfoShare-internen Passwort anmelden darf

### 30.4.6 Anmelden per Single-Sign-On

`loginOsAccount="false"`

- Ist dieses Tag auf „true“ gesetzt, kann sich der Benutzer, wenn alles korrekt konfiguriert wurde (siehe Handbuch von InfoShare.Client), per Single-Sign-On anmelden

### 30.4.7 Standardsuche

`showDefaultSearch="true"`

- Angabe, ob der Benutzer die Standardsuche sieht und ausführen darf

### 30.4.8 Benutzer darf Optionen ändern

`canChangeSettings="true"`

- Ist dieses Tag auf „true“ gesetzt, darf der Benutzer seine Optionen ändern

### 30.4.9 Benutzersprache

`defaultLanguage="de"`

- Definierte Sprache des Benutzers (Deutsch, Englisch oder Französisch)

### 30.4.10 E-Mail-Adresse

`emailAdress="admin@kendox.com"`

- Zugewiesene E-Mail-Adresse für den jeweiligen Benutzer

### 30.4.11 Benutzer darf Spracheinstellungen ändern

`forceLanguage="false"`

- Ist dieses Tag auf „true“ gesetzt, darf der Benutzer seine persönlichen Spracheinstellungen verändern

### 30.4.12 E-Mail-Benachrichtigung

`emailNotification="true"`

- Ist dieses Tag auf „true“ gesetzt, ist die E-Mail-Benachrichtigung aktiv

## 31 Benutzer einer Benutzergruppe auflisten

(*GETUSERSINGROUP*)

Mit dieser Funktion können alle Benutzer einer bestimmten Benutzergruppe ausgelesen werden. Diese Funktion wird nur vom DCI.Server unterstützt.

### 31.1 Anforderung

```
<?xml version="1.0" encoding="utf-16"?>
<XmlAPI>
  <APICall>GETUSERSINGROUP</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <GetUsersInGroup>
    <Request>
      <GroupName>Produktion</GroupName>
    </Request>
  </GetUsersInGroup>
</XmlAPI>
```

### 31.2 Antwort

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <GetUsersInGroup>
    <Return>
      <UserList>
        <User id="ce8eada2-3589-11df-a663-000c2955ee1c" LogonName="hro">
          Hans Roth
        </User>
        <User id="65d0c9b4-3595-11df-8096-000c29bbc049" LogonName="pha">
          Paul Haus
        </User>
        <User id="0100a003-38a7-11d8-a902-0050bf427e8f" LogonName="tbu">
          Tom Bunz
        </User>
      </UserList>
    </Return>
  </GetUsersInGroup>
</XmlAPI>
```

### 31.3 Code-Definitionen der Antwort

#### 31.3.1 Benutzer

```
<User id="ce8eada2-3589-11df-a663-000c2955ee1c" LogonName="hro">Hans Roth</User>
```

- Pro definierten Benutzer wird ein „<User>“-Eintrag erstellt
- Im Wert wird der Anzeigename (nicht Anmeldename) des Benutzers zurückgegeben

#### 31.3.2 Benutzer-ID

```
<User id="ce8eada2-3589-11df-a663-000c2955ee1c" ...
```

- Eindeutige GUID des Benutzers

#### 31.3.3 <LogonName>

```
LogonName="hro"
```

- Anmeldename für diesen Benutzer

## 32 Daten aus Verzeichnis auslesen

### (GETFOLDERDATA)

Mit dieser Funktion können alle Daten zu einem Verzeichnis aufgelistet werden. Diese Funktion wird nur vom DCI.Server unterstützt.

### 32.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETFOLDERDATA</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <GetFolderData>
    <Request>
      <FolderPath>Öffentliche Ordner</FolderPath>
      <!-- <FolderPath>.</FolderPath> -->
      <!-- <FolderPath>PERSONAL\MeineRechnungen</FolderPath> -->
      <!-- <FolderPath>Dokumente</FolderPath> -->
    </Request>
  </GetFolderData >
</XmlAPI>
```

### 32.2 Code-Definition der Anforderung

#### 32.2.1 Ordnerpfad

```
<FolderPath>Dokumente</FolderPath>
```

- Pfad des Verzeichnisses, aus dem die Daten ausgelesen werden sollen
- Wird nur ein „.“ (Punkt) als Anfrage übergeben, werden alle definierten öffentlichen Ordner zurückgegeben
- Auf das private Verzeichnis des Benutzers kann zugegriffen werden, wenn „PERSONAL\“ angegeben wird

### 32.3 Antwort

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <GetFolderData>
    <Return>
      <FolderList>
        <Folder>Rechnungen</Folder>
        <Folder>Lieferscheine</Folder>
      </FolderList>
      <SearchFolderList>
        <Folder>Rechnungssuche</Folder>
        <Folder>Lieferscheinsuche</Folder>
      </SearchFolderList>
      <DocumentList>
        <Columns>
          <Column typeid="01000a00-38a7-11d8-a902-0050bf427e8f" name="Name"
            align="left">Name</Column>
          <Column typeid="c9bca816-e4fd-11e0-bd71-000c29b4469b" name="Dokumentart"
            align="left">Dokumentart</Column>
          <Column typeid="8fdd42b2-e521-11e0-bd71-000c29b4469b"
            name="Zusatzinformationen" align="left">Zusatzinformationen</Column>
        </Columns>
        <Document ID="311e9619-0f89-11e1-b876-005056c00008"
          StoreID="db22a820-e4f5-11e0-b827-00155d051700" OriginalFilename="Invoice
          4711.pdf" HasSignature="no" checkedout="false">
          <Name>Invoice 4711.pdf</Name>
          <Property name="Sicherheitsbereich" type="" multikey="false">Standardsicherheit
          </Property>
          <Property name="Dateityp" type="" multikey="true">tif</Property>
          <Property name="Produkt" type="" multikey="true">InfoShare.Client</Property>
          <Property name="Dokumentart" type="" multikey="true">Rechnungen</Property>
          <Property name="Änderungszähler" type="" multikey="false">6</Property>
          <Property name="Version" type="" multikey="false">3</Property>
          <Property name="Lebenszyklus-Zustand" type="" multikey="false"> Prevent change
          </Property>
          <Property name="Lebenszyklus" type="" multikey="false"> Prevent change
          </Property>
          <Property name="Signaturen" type="" multikey="true" />
          <Property name="Aufbewahrungsfrist" type="" multikey="false" />
          <Property name="Letztes Einbuchen" type="" multikey="false">22.11.2011 09:33
          </Property>
          <Property name="Interner Name" type="" multikey="true">Invoice 4711.pdf
          </Property>
          <Property name="Erinnerungsdatum" type="" multikey="true" />
          <Property name="Name" type="" multikey="false">Invoice 4711.pdf</Property>
          <Property name="Ausgebucht von" type="" multikey="false" />
        </Document>
      </DocumentList>
    </Return>
  </GetFolderData>
</XmlAPI>
```

### 32.4 Code-Definitionen der Antwort

#### 32.4.1 Ordnerliste

```
<FolderList>  
  <Folder>Rechnungen</Folder>  
  <Folder>Lieferscheine</Folder>  
</FolderList>
```

- In diesem Tag werden alle vorhandenen Ordner aufgelistet

#### 32.4.2 Suchordnerliste

```
<SearchFolderList>  
  <Folder>Rechnungssuche</Folder>  
  <Folder>Lieferscheinsuche</Folder>  
</SearchFolderList>
```

- In diesem Tag werden alle vorhandenen Suchordner aufgelistet

#### 32.4.3 Suchordnerliste

```
<DocumentList>  
  ...  
</DocumentList>
```

- In diesem Tag werden alle vorhandenen Dokumente und deren Eigenschaften aufgelistet

### 33 Daten aus Suchverzeichnis auslesen

(GETSEARCHFOLDERDATA)

Mit dieser Funktion können die Daten eines Suchverzeichnisses ausgelesen werden. Diese Funktion wird nur vom DCI.Server unterstützt.

#### 33.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETSEARCHFOLDERDATA</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <GetSearchFolderData>
    <Request>
      <FolderPath>Öffentliche Ordner\Rechnungen</FolderPath>
      <!-- <FolderPath>PERSONAL\MeineRechnungen</FolderPath> -->
    </Request>
  </GetSearchFolderData>
</XmlAPI>
```

#### 33.2 Code-Definition der Anforderung

##### 33.2.1 Ordnerpfad

<FolderPath>Öffentliche Ordner\Rechnungen</FolderPath>

- Angabe vom Pfad des Verzeichnisses, aus welchem die Daten ausgelesen werden sollen
- Auf das private Verzeichnis des Benutzers kann zugegriffen werden, wenn „PERSONAL\“ angegeben wird

### 33.3 Antwort

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <GetSearchFolderData>
    <Return>
      <Title />
      <Comment>Suchordner für Rechnungen</Comment>
      <IsSimpleSearch>true</IsSimpleSearch>
      <IsFullTextEnabled>true</IsFullTextEnabled>
      <Columns>
        <Column typeid="c9bca816-e4fd-11e0-bd71-000c29b4469b" name="Dokumentart"
          align="left">Dokumentart</Column>
        <Column typeid="01000a00-38a7-11d8-a902-0050bf427e8f" name="Name" align="left"
          sorted="true" sortorder="ascending">Name</Column>
      </Columns>
      <CriterionList>
        <Criterion coperator="ge" relation="and">
          <PropertyType typeid="01000009-38a7-11d8-a902-0050bf427e8f"
            name="Rechnungssuche" displayname="Rechnungssuche" visible="True"
            editable="True">Letztes Einbuchen</PropertyType>
          <SearchValue>1.1.2000</SearchValue>
        </Criterion>
        <Criterion coperator="eq" relation="and">
          <PropertyType typeid="c9bca816-e4fd-11e0-bd71-000c29b4469b"
            name="Dokumentart" displayname="" visible="False" editable="False">
            Dokumentart</PropertyType>
          <SearchValue>Rechnungen</SearchValue>
        </Criterion>
      </CriterionList>
      <SearchStores>
        <Store>Produktion</Store>
        <Store>Standard</Store>
      </SearchStores>
    </Return>
  </GetSearchFolderData>
</XmlAPI>
```

### 33.4 Code-Definitionen der Antwort

#### 33.4.1 Suchordner-Beschreibung

`<Comment>Suchordner für Rechnungen</Comment>`

- Angabe vom Pfad des Verzeichnisses, aus welchem die Daten ausgelesen werden wurden

#### 33.4.2 Einfache Suche

`<IsSimpleSearch>true</IsSimpleSearch>`

- Dieses Tag ist auf „true“ gesetzt, wenn beim Suchordner die einfache Suche aktiviert wurde

#### 33.4.3 Volltext

`<IsFullTextEnabled>true</IsFullTextEnabled>`

- Sobald die Volltextsuche aktiviert wird, ist dieses Tag „true“

#### 33.4.4 Spalten

`<Columns>`

`<Column typeid="c9bca816-e4fd-11e0-bd71-000c29b4469b" name="Dokumentart"
 align="left">Dokumentart</Column>`

...

`</Columns>`

- Im Tag `<Columns>` werden alle Spalten aufgelistet, welche bei der Resultatliste dargestellt werden

### 33.4.5 Suchkriterien

```
<CriterionList>  
  <Criterion coperator="ge" relation="and">  
    <PropertyType typeid="01000009-38a7-11d8-a902-0050bf427e8f"  
      name="Rechnungssuche" displayname="Rechnungssuche" visible="True"  
      editable="True">Letztes Einbuchen</PropertyType>  
    <SearchValue>1.1.2000</SearchValue>  
  </Criterion>  
  ...  
</CriterionList>
```

- In diesem Tag werden die definierten Kriterien der Suche angezeigt

### 33.4.6 Ablagen

```
<SearchStores>  
  <Store>Produktion</Store>  
  <Store>Standard</Store>  
</SearchStores>
```

- Dieses Tag zeigt die vom Suchordner berücksichtigten Ablagen an

## 34 Liste von Dokumenten anzeigen

(VIEWLIST)

Mit dieser Funktion kann eine Liste von Dokumenten im gleichen Dialog wie die Suchresultate angezeigt werden bzw. angezeigt werden lassen. Diese Funktion wird nur vom DCI.Server unterstützt.

### 34.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>VIEWLIST</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <ViewList>
    <Request>
      <DocumentList>
        <Document>
          <ID>388ba221-e522-11e0-bd71-000c29b4469b</ID>
          <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
        </Document>
        <Document>
          <ID>d89182b6-e517-11e0-bd71-000c29b4469b</ID>
          <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
        </Document>
      </DocumentList>
      <PropertyList>
        <Property>Name</Property>
        <Property>Zusatzinformationen</Property>
        <Property>Letztes Einbuchen</Property>
      </PropertyList>
      <Grouping>
        <Property>Dokumentart</Property>
      </Grouping>
    </Request>
  </ViewList>
</XmlAPI>
```

### 34.2 Code-Definitionen der Anforderung

#### 34.2.1 Dokumentliste

```
<DocumentList>...</DocumentList>
```

- In diesem Container werden die einzelnen Dokumente definiert, welche angezeigt werden sollen

#### 34.2.2 Dokumentdefinition

```
<Document>
  <ID>d89182b6-e517-11e0-bd71-000c29b4469b</ID>
  <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID>
</Document>
```

- Angabe der Dokument- und Ablage-ID des anzuzeigenden Dokuments
- Hier können beliebig viele Dokumente definiert werden

### 34.2.3 Eigenschaftsliste

```
<PropertyList>  
  <Property>Name</Property>  
  <Property>Zusatzinformationen</Property>  
  <Property>Letztes Einbuchen</Property>  
</PropertyList>
```

- Angabe der Eigenschaften, die in der Trefferliste dargestellt werden sollen

### 34.2.4 Gruppierung

```
<Grouping>  
  <Property>Dokumentart</Property>  
</Grouping>
```

- Definiert die Gruppierung der Dokumente im Resultatdialog
- Es sind maximal drei Stufen möglich (wie in InfoShare.Client)

## 35 Dokumente ausbuchen

(CHECKOUT)

Mit dieser Funktion können Dokumente ausgebucht werden.

### 35.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>CHECKOUT</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <CheckOut>
    <Request>
      <DocumentID>388ba221-e522-11e0-bd71-000c29b4469b </DocumentID>
      <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
      <outputPath>Z:\XmlAPI</outputPath>
    </Request>
  </CheckOut>
</XmlAPI>
```

### 35.2 Code-Definitionen der Anforderung

#### 35.2.1 Dokument-ID

```
<DocumentID>388ba221-e522-11e0-bd71-000c29b4469b </DocumentID>
```

- Hier wird die ID des Dokuments angegeben, welches ausgebucht werden soll

#### 35.2.2 Ablage-ID

```
<StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID>
```

- Wird die eindeutige (optionale) Ablagekennung ebenfalls angegeben, wird der Prozess für die angeforderte Funktion beschleunigt; d. h. es werden nicht alle Ablagen nach der angeforderten Dokumenten-ID durchsucht

#### 35.2.3 Ausgabepfad

```
<outputPath>Z:\XmlAPI</outputPath>
```

- Dieser lokale Pfad definiert, wo die ausgebuchte Datei gespeichert werden soll

### 35.3 Antwort

```
<?xmlversion="1.0"encoding="UTF-16"?>
<XmlAPI>
  <GetFileForEdit>
    <Return>
      <OutPutFile>Z:\XmlAPI\Bildformat-Test.tif</OutPutFile>
    </Return>
  </GetFileForEdit>
</XmlAPI>
```

### 35.4 Code-Definition der Antwort

#### 35.4.1 <outputFile>

- Pfad und Dateiname, wo die ausgebuchte Datei gespeichert wurde

## 36 Dokumentausbuchung aufheben

(UNDOCHECKOUT)

Mit dieser Funktion können ausgebuchte Dokumente wieder eingebucht werden.

### 36.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>UNDOCHECKOUT</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <CheckIn>
    <Request>
      <DocumentID>388ba221-e522-11e0-bd71-000c29b4469b </DocumentID>
      <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
      <FileName>Z:\XmlAPI\Bildformat-Test.tif</FileName>
    </Request>
  </CheckIn>
</XmlAPI>
```

### 36.2 Code-Definition der Anforderung

#### 36.2.1 Dateiname

```
<FileName>Z:\XmlAPI\Bildformat-Test.tif</FileName>
```

- Das Tag „FileName“ ist optional und repräsentiert den Pfad zum ausgebuchten/exportierten Dokument
- Wird der Pfad angegeben, wird die Datei nach der Aufhebung der Dokumentausbuchung gelöscht

### 36.3 Antwort

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <GetFileForEdit>
    <Return>
      <OutPutFile>Z:\XmlAPI\Bildformat-Test.tif</OutPutFile>
    </Return>
  </GetFileForEdit>
</XmlAPI>
```

### 36.4 Code-Definition der Antwort

#### 36.4.1 <outputFile>

- Pfad und Dateiname, wo die ausgebuchte Datei gespeichert wurde

## 37 Dokument einbuchen

(CHECKIN)

Mit dieser Funktion können Dokumente eingebucht werden.

### 37.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>CHECKIN</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <CheckIn>
    <Request>
      <DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
      <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
      <!-- <Comment>Rechnung überprüft</Comment> -->
      <!-- <Version>2</Version> -->
      <FileName>Z:\XmlAPI\Bildformat-Test.tif</FileName>
      <!-- <SigningProfile>
        <Profile>Signaturen</Profile>
      </SigningProfile> -->
      <!-- <Signatures>
        <Signatur>Z:\XmlAPI\Bildformat-Test.pdf.p7s</Signatur>
      </Signatures> -->
    </Request>
  </CheckIn>
</XmlAPI>
```

### 37.2 Code-Definitionen der Anforderung

#### 37.2.1 Dokument-ID

```
<DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
```

- Hier wird die ID des Dokuments angegeben, welches ausgebucht werden soll
- Dieses Tag ist notwendig

#### 37.2.2 Ablage-ID

```
<StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID>
```

- Wird die eindeutige (optionale) Ablagekennung ebenfalls angegeben, wird der Prozess für die angeforderte Funktion beschleunigt; d. h. es werden nicht alle Ablagen nach der angeforderten Dokumenten-ID durchsucht

#### 37.2.3 Einbuchen-Kommentar

```
<Comment>Rechnung überprüft</Comment>
```

- Optionaler Kommentar zum Einbuchen

#### 37.2.4 Angabe der Version

```
<Version>2</Version>
```

- Optionale Version zum Einbuchen

#### 37.2.5 Dateiname

```
<FileName>Z:\XmlAPI\Bildformat-Test.tif</FileName>
```

- Kompletter Pfad der Datei, welche eingebucht werden soll

### 37.2.6 Signaturprofil

<SigningProfile>

<Profile>Signaturen</Profile>

</SigningProfile>

- Optionales Profil, welches zum Signieren verwendet werden kann (im Dialog nicht vorhanden)

### 37.2.7 <Signatures>

<Signatures>

<Signatur>Z:\XmlAPI\Bildformat-Test.pdf.p7s</Signatur>

</Signatures>

- Definiert das gesamte Verzeichnis der zu verwendenden Signatur
- PluginClass: optional
- certID: Beschreibung für Zertifikat (suchbar, im Dialog nicht vorhanden)

### 38 Signaturprofile ausgeben

(GETSIGNATUREPROFILES)

Gibt eine Liste mit den im InfoShare vorhandenen Signaturprofilen aus.

#### 38.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETSIGNATUREPROFILES</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <GetSignatureProfiles>
    <Request>
      </Request>
    </GetSignatureProfiles>
  </XmlAPI>
```

#### 38.2 Antwort

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <GetSignatureProfiles>
    <Return>
      <Profile>
        <Value>Persönliche Unterschrift und Zeitstempel</Value>
      </Profile>
      <Profile>
        <Value>Persönliche Unterschrift und 2 Zeitstempel</Value>
      </Profile>
    </Return>
  </GetSignatureProfiles>
</XmlAPI>
```

### 39 Signaturen anzeigen

(GETSIGNATURES)

Zeigt alle vorhandenen Signaturen eines bestimmten Dokumentes an.

#### 39.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETSIGNATUREPROFILES</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <GetSignatureProfiles>
    <Request>
      </Request>
    </GetSignatureProfiles>
  </XmlAPI>
```

#### 39.2 Antwort

Die Antwort variiert je nach Zertifikatsaussteller.

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <GetSignatures>
    <Return>
      <Signatures>
        <Node Value="Digitale Signatur">
          <Node Value="Signiert durch">
            <Node Value="Zeitpunkt : 23.11.2011 16:35" />
            <Node Value="Zertifikat">
              <Node Value="Aussteller">
                <Node Value="C=CH" />
                <Node Value="O=QuoVadis Trustlink Schweiz AG" />
                <Node Value="OU=Issuing Certificate Authority" />
                <Node Value="CN=QV Schweiz ICA" />
              </Node>
              <Node Value="Antragsteller">
                <Node Value="C=CH" />
                <Node Value="OU=Standard Commercial Certificate" />
                <Node Value="OU=Testzertifikat Kein Haftungsanspruch" />
                <Node Value="CN=TEST: Kendox" />
                <Node Value="E=signatur@kendox.com" />
              </Node>
              <Node Value="Gültig von : 08.02.2010 10:31" />
              <Node Value="Gültig bis : 08.02.2013 16:00" />
              <Node Value="Signierungsalgorithmus : SHA-1withRSA" />
              <Node Value="Version : 3" />
              <Node Value="Seriennummer : 12345" />
            </Node>
          </Node>
        </Node>
      </Signatures>
    </Return>
  </GetSignatures>
</XmlAPI>
```

## 40 Signatur prüfen

### (CHECKSIGNATURES)

Überprüft eine Signatur auf ihre Gültigkeit.

#### 40.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>CHECKSIGNATURES</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <CheckSignatures>
    <Request>
      <DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
      <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
      <!-- <SignatureID>d42c916c-15e8-11e1-bf86-000c29b4469b</SignatureID> -->
      <!-- <VersionID>7dcec6fc-e56f-11da-8f7c-0013a90dcd5a </VersionID> -->
    </Request>
  </CheckSignatures>
</XmlAPI>
```

#### 40.2 Code-Definitionen der Anforderung

##### 40.2.1 Signatur-ID

```
<SignatureID>d42c916c-15e8-11e1-bf86-000c29b4469b</SignatureID>
```

- Optionale Signatur-ID (Ausgabe von „GETSIGNATURES“)
- Es wird nur die entsprechende Signatur überprüft

##### 40.2.2 Dokument-Versions-ID

```
<VersionID>7dcec6fc-e56f-11da-8f7c-0013a90dcd5a </VersionID>
```

- Hiermit wird die entsprechende (originale) optionale Version des Dokumentes überprüft

#### 40.3 Antwort

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <CheckSignatures>
    <Return>
      <Signatures>
        <Signature>
          <SignatureID>d42c916c-15e8-11e1-bf86-000c29b4469b</SignatureID>
          <Valid>True</Valid>
        </Signature>
      </Signatures>
    </Return>
  </CheckSignatures>
</XmlAPI>
```

### 40.4 Code-Definitionen der Antwort

#### 40.4.1 Signatur-ID

`<SignatureID>d42c916c-15e8-11e1-bf86-000c29b4469b</SignatureID>`

- Mit diesem Tag wird die ID der Signatur (des Dokuments) zurückgegeben

#### 40.4.2 Gültigkeit der Signatur

`<Valid>True</Valid>`

- Wird mit diesem Tag „True“ zurückgegeben, ist die angegebene Signatur gültig

# 41 Versionsliste anzeigen

(GETVERSIONLIST)

Zeigt eine Liste mit allen Versionen eines Dokumentes an.

## 41.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETVERSIONLIST</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <GetVersionList>
    <Request>
      <DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
      <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
      <!-- <VersionsOnly>yes</VersionsOnly> -->
    </Request>
  </GetVersionList>
</XmlAPI>
```

## 41.2 Antwort

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <Versions>
    <Version>
      <VersionID>0f6ad2c5-226c-11e1-8aaa-00155d051700</VersionID>
      <Date>25.09.2011 14:45:55</Date>
      <Name>Buchhaltung</Name>
      <Version>2</Version>
      <Comment>Revidiert</Comment>
    </Version>
    <Version>
      <VersionID>d7acf860-e517-11e0-b827-00155d051700</VersionID>
      <Date>22.09.2011 14:38:56</Date>
      <Name>Scanner</Name>
      <Version>1</Version>
      <Comment>Eingescannt</Comment>
    </Version>
  </Versions>
</XmlAPI>
```

### 41.3 Code-Definitionen der Antwort

#### 41.3.1 Versionen-Container

`<Versions>...</Versions>`

- Dieser Tag-Container beinhaltet alle Angaben zu den Dokumentversionen
- Die Dokumentversionen werden nach Datum (absteigend) sortiert

#### 41.3.2 Version

`<Version>`

`<VersionID>0f6ad2c5-226c-11e1-8aaa-00155d051700</VersionID>`

`<Date>25.09.2011 14:45:55</Date>`

`<Name>Buchhaltung</Name>`

`<Version>2</Version>`

`<Comment>Revidiert</Comment>`

`</Version>`

- Dieser Sub-Tag-Container enthält alle Informationen zur jeweiligen Version
- Pro Dokumentversion wird ein Sub-Tag-Container geschrieben

## 42 Versioneigenschaften anzeigen

(*GETVERSIONPROPERTIES*)

Zeigt die Dateieigenschaften der ausgewählten Version an.

### 42.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETVERSIONPROPERTIES</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <GetVersionProperties>
    <Request>
      <DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
      <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
      <VersionID>3ae452f4-e522-11e0-b827-00155d051700</VersionID>
    </Request>
  </GetVersionProperties>
</XmlAPI>
```

### 42.2 Code-Definition der Anforderung

#### 42.2.1 Dokument-ID

```
<DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
```

- Hier wird die ID des Dokuments angegeben, welches ausgebucht werden soll
- Dieses Tag ist notwendig

#### 42.2.2 Versions-ID

```
<VersionID>0f6ad2c5-226c-11e1-8aaa-00155d051700</VersionID>
```

- ID-Nummer der Version; diese kann über „GetVersionList“ ausgelesen werden
- Dieses Tag ist notwendig

### 42.3 Antwort

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <VersionProperties>
    <ID>0f6ad2c5-226c-11e1-8aaa-00155d051700</ID>
    <Property name="Dokumentart">
      <Value>Rechnungen</Value>
      <Category>Datei</Category>
    </Property>
    <Property name="Dateityp">
      <Value>pdf</Value>
      <Category>Standard</Category>
    </Property>
    <Property name="Änderungszähler">
      <Value>2</Value>
      <Category>Standard</Category>
    </Property>
    <Property name="Version">
      <Value>2</Value>
      <Category>Standard</Category>
    </Property>
    <Property name="Lebenszyklus-Zustand">
      <Category>Standard</Category>
    </Property>
    <Property name="Lebenszyklus">
      <Category>Standard</Category>
    </Property>
    <Property name="Signaturen">
      <Category>Standard</Category>
    </Property>
    <Property name="Aufbewahrungsfrist">
      <Value />
      <Category>Standard</Category>
    </Property>
    <Property name="Letztes Einbuchen">
      <Value>09.12.2011 14:45</Value>
      <Category>Standard</Category>
    </Property>
    <Property name="Interner Name">
      <Value>Rechnung4711.pdf</Value>
      <Category>Standard</Category>
    </Property>
    <Property name="Erinnerungsdatum">
      <Category>Standard</Category>
    </Property>
    <Property name="Name">
      <Value>Rechnung4711.pdf</Value>
      <Category>Standard</Category>
    </Property>
  </VersionProperties>
</XmlAPI>
```

### 43 Dateiversion anzeigen

(GETVERSION)

Exportiert eine Dokumentenversion zur Ansicht und gibt den Pfad zurück.

#### 43.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETVERSION</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <GetVersion>
    <Request>
      <DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
      <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
      <VersionID>3ae452f4-e522-11e0-b827-00155d051700</VersionID>
      <!-- <outputPath>Z:\XmlAPI\Output</outputPath> -->
    </Request>
  </GetVersion>
</XmlAPI>
```

#### 43.2 Code-Definitionen der Anforderung

##### 43.2.1 Dokument-ID

```
<DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
```

- Hier wird die ID des Dokuments angegeben, welches ausgebucht werden soll

##### 43.2.2 Dokument-Versions-ID

```
<VersionID>3ae452f4-e522-11e0-b827-00155d051700</VersionID>
```

- Versions-ID, welche über „GetVersionList“ ausgelesen werden kann

##### 43.2.3 Ausgabepfad

```
<outputPath>Z:\XmlAPI\Output</outputPath>
```

- Optionaler Pfad, in welchem die Datei gespeichert werden soll

#### 43.3 Antwort

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <ID>d7acf860-e517-11e0-b827-00155d051700</ID>
  <Path>Z:\XmlAPI\Output\Rechnung.pdf</Path>
</XmlAPI>
```

#### 43.4 Code-Definitionen der Antwort

##### 43.4.1 Versions-ID

```
<ID>388ba221-e522-11e0-bd71-000c29b4469b</ID>
```

- Hier wird die ID des Dokuments angegeben, welches ausgebucht werden soll

##### 43.4.2 Ausgabepfad

```
<Path>Z:\XmlAPI\Output\Lorem Ipsum.doc</Path>
```

- Angabe des Pfades, wo das Dokument abgelegt wurde
- Da „outputPath“ optional ist, wird dieses Tag nicht immer dargestellt

## 44 Dokumentenstatus anzeigen

(CHECKOUTSTATUS)

Zeigt den aktuellen Dokumentenstatus an.

### 44.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>CHECKOUTSTATUS</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <CheckoutStatus>
    <Request>
      <DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
      <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
    </Request>
  </CheckoutStatus>
</XmlAPI>
```

### 44.2 Antwort bei ausgecheckten Dokumenten

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <CheckoutStatus>CheckedOutExclusive</CheckoutStatus>
</XmlAPI>
```

### 44.3 Antwort bei nicht ausgecheckten Dokumenten

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <CheckoutStatus>NotCheckedOut</CheckoutStatus>
</XmlAPI>
```

## 45 Passwort ändern

(*CHANGEPASSWORD*)

Ändert das Passwort des aktuell angemeldeten Benutzers.

### 45.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>CHANGEPASSWORD</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <ChangePassword>
    <Request>
      <OldPassword>AltesPasswort$2011</OldPassword>
      <NewPassword>NeuesPasswort$2012</NewPassword>
    </Request>
  </ChangePassword>
</XmlAPI>
```

### 45.2 Code-Definitionen der Anforderung

#### 45.2.1 Angabe des alten Passworts

```
<OldPassword>AltesPasswort$2011</OldPassword>
```

- Hier wird die ID des Dokuments angegeben, welches ausgebucht werden soll
- Dieses Tag ist notwendig

#### 45.2.2 Definition des neuen Passworts

```
<NewPassword>NeuesPasswort$2012</NewPassword>
```

- In diesem Tag wird das neue Passwort definiert
- Kendox empfiehlt die Verwendung von sicheren Passwörtern (siehe: <http://www.microsoft.com/de-ch/security/online-privacy/passwords-create.aspx>)

### 45.3 Antwort

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <ChangePassword>
    <Return>True</Return>
  </ChangePassword>
</XmlAPI>
```

### 45.4 Code-Definitionen der Antwort

#### 45.4.1 Return

```
<Return>True</Return>
```

- Wird mit diesem Tag der Wert „True“ zurückgegeben, wurde das Passwort des momentan angemeldeten Benutzers geändert
- Wird „False“ zurückgegeben, wurde das neue Passwort nicht gesetzt; Grund dafür könnte sein, dass das alte Passwort nicht korrekt angegeben wurde

## 46 Ordner erstellen

(CREATEFOLDER)

Erstellt den angegebenen Ordner. Existiert der Ordner bereits, wird eine Fehlermeldung zurückgegeben.

### 46.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>CREATEFOLDER</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <CreateFolder>
    <Request>
      <Path>Öffentliche Ordner\Rechnungen</Path>
    </Request>
  </CreateFolder>
</XmlAPI>
```

### 46.2 Code-Definition der Anforderung

#### 46.2.1 Angabe des Pfades

```
<Path>Öffentliche Ordner\Rechnungen</Path>
```

- Dieses Tag definiert den gesamten Pfad des neu zu erstellenden Ordners innerhalb von InfoShare

### 46.3 Antwort

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <CreateFolder>
    <Return>
      <Folder>Öffentliche Ordner\Rechnungen</Folder>
    </Return>
  </CreateFolder>
</XmlAPI>
```

### 46.4 Code-Definition der Antwort

#### 46.4.1 Pfad des erstellten Ordners

```
<Path>Öffentliche Ordner\Rechnungen</Path>
```

- Dieses Tag gibt den gesamten Pfad in InfoShare des soeben erstellten Ordners zurück

## 47 Ordneigenschaften anzeigen

(GETFOLDERPROPERTIES)

Liefert die angezeigten Eigenschaften des Ordners zurück.

### 47.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETFOLDERPROPERTIES</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <GetFolderProperties>
    <Request>
      <Path>Öffentliche Ordner\Rechnungen</Path>
    </Request>
  </GetFolderProperties>
</XmlAPI>
```

### 47.2 Code-Definition der Anforderung

#### 47.2.1 Angabe des Ordnerpfades

```
<Path>Öffentliche Ordner\Rechnungen</Path>
```

- Hier wird der Pfad des Ordners definiert, von welchem die Eigenschaften ausgelesen werden sollen

### 47.3 Antwort

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <GetFolderProperties>
    <Return>
      <PropertyTypes>
        <PropertyType>Name</PropertyType>
        <PropertyType>Dokumentart</PropertyType>
        <PropertyType>Zusatzinformationen</PropertyType>
      </PropertyTypes>
      <InheritFromParent>True</InheritFromParent>
      <IncludeSubFolders>True</IncludeSubFolders>
    </Return>
  </GetFolderProperties>
</XmlAPI>
```

### 47.4 Code-Definitionen der Antwort

#### 47.4.1 Eigenschaftstypen

```
<PropertyTypes>  
  <PropertyType>Name</PropertyType>  
  <PropertyType>Dokumentart</PropertyType>  
  ...  
</PropertyTypes>
```

- Dieses Tag gibt die Spalten Ordners zurück (Eigenschaftstypen)

#### 47.4.2 Spalteneinstellungen vom übergeordneten Ordner erben

```
<InheritFromParent>True</InheritFromParent>
```

- Ist dieses Tag auf „True“ gesetzt, werden die Spalteneinstellungen vom übergeordneten Ordner geerbt

#### 47.4.3 Elemente aller Unterordner anzeigen

```
<IncludeSubFolders>True</IncludeSubFolders>
```

- Ist dieses Tag auf „True“ gesetzt, werden auch die Elemente aller Unterordner angezeigt

## 48 Ordneigenschaften setzen

(SETFOLDERPROPERTIES)

Setzt die angezeigten Eigenschaften des Ordners.

### 48.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>SETFOLDERPROPERTIES</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <SetFolderProperties>
    <Request>
      <Path>Öffentliche Ordner\Rechnungen</Path>
      <IncludeSubFolders>True</IncludeSubFolders>
      <InheritFromParent>True</InheritFromParent>
      <PropertyTypes>
        <PropertyType>Name</PropertyType>
        <PropertyType>Dateityp</PropertyType>
      </PropertyTypes>
    </Request>
  </SetFolderProperties>
</XmlAPI>
```

### 48.2 Code-Definitionen der Anforderung

#### 48.2.1 Angabe des Ordnerpfades

```
<Path>Öffentliche Ordner\Rechnungen</Path>
```

- Hier wird der Pfad des Ordners definiert, dessen Eigenschaften geschrieben werden sollen

#### 48.2.2 Elemente aller Unterordner anzeigen

```
<IncludeSubFolders>True</IncludeSubFolders>
```

- Ist dieses Tag auf „True“ gesetzt, werden auch die Elemente aller Unterordner angezeigt

#### 48.2.3 Spalteneinstellungen vom übergeordneten Ordner erben

```
<InheritFromParent>True</InheritFromParent>
```

- Ist dieses Tag auf „True“ gesetzt, werden die Spalteneinstellungen vom übergeordneten Ordner geerbt

#### 48.2.4 Eigenschaftstypen

```
<PropertyTypes>
  <PropertyType>Name</PropertyType>
  <PropertyType>Dokumentart</PropertyType>
  ...
</PropertyTypes>
```

- Dieses Tag definiert die Spalten des Ordners (Eigenschaftstypen)

## 49 Prüfen ob Benutzer existiert

(USEREXISTS)

Prüft, ob der angegebene Benutzer existiert.

### 49.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>USEREXISTS</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <UserExists>
    <Request>
      <LoginName>DCIAdmin</LoginName>
    </Request>
  </UserExists>
</XmlAPI>
```

### 49.2 Antwort

#### 49.2.1 Antwort falls der Benutzer existiert

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <UserExists>
    <Return>
      <Exists>Yes</Exists>
    </Return>
  </UserExists>
</XmlAPI>
```

#### 49.2.2 Antwort falls der Benutzer nicht existiert

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <UserExists>
    <Return>
      <Exists>No</Exists>
    </Return>
  </UserExists>
</XmlAPI>
```

## 50 Benutzer löschen

*(USERDELETE)*

Löscht den im Tag <LoginName> angegebenen Benutzer. Hierbei muss der Name (nicht der Anzeigename) definiert werden, damit das Löschen funktioniert.

### 50.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>USERDELETE</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <UserDelete>
    <Request>
      <LoginName>scan</LoginName>
    </Request>
  </UserDelete>
</XmlAPI>
```

## 51 Benutzer erstellen

### (USERCREATE)

Erstellt einen neuen Benutzer mit der entsprechenden Konfiguration.

#### 51.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>USERCREATE</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <UserCreate>
    <Request>
      <LoginName>maxmuster</LoginName>
      <DisplayName>Max Mustermann</DisplayName>
      <Password>myMax2012</Password>
      <Active>yes</Active>
      <LoginWithoutPassword>no</LoginWithoutPassword>
      <LoginInternalPassword>yes</LoginInternalPassword>
      <LoginOSAccount>no</LoginOSAccount>
      <!-- <ShowDefaultSearch>true</ShowDefaultSearch> -->
      <!-- <CanChangeSettings>true</CanChangeSettings> -->
      <!-- <DefaultLanguage>de</DefaultLanguage> -->
      <!-- <EmailAdress>m.mustermann@muster.com</EmailAdress> -->
      <!-- <ForceLanguage>>false</ForceLanguage> -->
      <!-- <EmailNotification>true</EmailNotification> -->
      <!-- <ToDoNotification>true</ToDoNotification> -->
    </Request>
  </UserCreate>
</XmlAPI>
```

#### 51.2 Code-Definitionen der Anforderung

##### 51.2.1 Benutzername

```
<LoginName>maxmuster</LoginName>
```

- Definiert den Namen, mit welchem sich der Benutzer anmelden können soll

##### 51.2.2 Anzeigename

```
<DisplayName>Max Mustermann</DisplayName>
```

- Definiert den Anzeigenamen des Benutzers
- Dieser Wert kann sich vom Benutzernamen komplett unterscheiden und wird lediglich für die bessere/detailliertere Angabe des Benutzernamens verwendet

##### 51.2.3 Benutzerpasswort

```
<Password>myMax2012</Password>
```

- Definiert das Passwort für den Benutzer

##### 51.2.4 Aktivierungsstatus des Benutzers

```
<Active>yes</Active>
```

- Mit dem Setzen dieses Tags auf „Yes“ wird der Benutzer nach der Erstellung aktiviert
- Wird das Tag auf „No“ definiert, ist der Benutzer deaktiviert und kann später im InfoShare.Client in den Benutzereinstellungen wieder aktiviert werden

### 51.2.5 Anmeldung ohne Passwort

`<LoginWithoutPassword>no</LoginWithoutPassword>`

- Wird dieses Tag auf „Yes“ gesetzt, kann sich der Benutzer ohne Passwort anmelden
- Das Aktivieren der „Anmeldung ohne Passwort“ wird aus Sicherheitsgründen nicht empfohlen

### 51.2.6 Anmeldung über internes Passwort

`<LoginInternalPassword>yes</LoginInternalPassword>`

- Wird dieses Tag auf „Yes“ gesetzt, kann sich der Benutzer mit dem definierten Passwort bei InfoShare anmelden

### 51.2.7 Anmeldung über Betriebssystemkonto

`<LoginOSAccount>no</LoginOSAccount>`

- Wird dieses Tag auf „Yes“ gesetzt, kann sich der Benutzer mit dem Betriebssystemkonto (Single-Sign-On) bei InfoShare anmelden

### 51.2.8 Benutzer darf Standardsuche ausführen

`<ShowDefaultSearch>true</ShowDefaultSearch>`

- Beim Setzen dieses Tags auf „true“ darf der Benutzer die Standardsuche ausführen

### 51.2.9 Benutzer darf Optionen ändern

`<CanChangeSettings>true</CanChangeSettings>`

- Beim Setzen dieses Tags auf „true“ darf der Benutzer seine InfoShare-Optionen ändern

### 51.2.10 Benutzersprache

`<DefaultLanguage>de</DefaultLanguage>`

- Mit diesem Tag wird die Standard-Benutzersprache definiert
- Es sind die Werte „de“ (Deutsch), „en“ (Englisch) und „fr“ (Französisch)

### 51.2.11 E-Mail-Adresse

`<EmailAdress>m.mustermann@muster.com</EmailAdress>`

- Hier kann die optionale E-Mail-Adresse definiert werden

### 51.2.12 Benutzer darf Spracheinstellungen verändern

`<ForceLanguage>>false</ForceLanguage>`

- Wird dieses Tag auf „true“ gesetzt, darf der Benutzer die Spracheinstellungen nicht verändern
- Wird dieses Tag auf „false“ gesetzt, darf der Benutzer die Spracheinstellungen verändern

### 51.2.13 E-Mail-Benachrichtigung

`<EmailNotification>true</EmailNotification>`

- Beim Setzen dieses Tags auf „true“ wird die E-Mail-Benachrichtigung für diesen Benutzer aktiviert
- Damit dieses Feature funktioniert, muss es serverseitig konfiguriert werden (siehe Handbuch von InfoShare.Server)

## 52 Benutzer aktivieren

(USERACTIVATE)

Aktiviert den angegebenen Benutzer.

### 52.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>USERACTIVATE</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <UserActivate>
    <Request>
      <LoginName>maxmuster</LoginName>
    </Request>
  </UserActivate>
</XmlAPI>
```

### 52.2 Code-Definition der Anforderung

#### 52.2.1 Benutzername

```
<LoginName>maxmuster</LoginName>
```

- Definiert den Namen des zu aktivierenden Benutzers

### 53 Benutzer deaktivieren

(*USERDEACTIVATE*)

Deaktiviert den angegebenen Benutzer.

#### 53.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>USERDEACTIVATE</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <UserDeactivate>
    <Request>
      <LoginName>maxmuster</LoginName>
    </Request>
  </UserDeactivate>
</XmlAPI>
```

#### 53.2 Code-Definition der Anforderung

##### 53.2.1 Benutzername

```
<LoginName>maxmuster</LoginName>
```

- Definiert den Namen des zu deaktivierenden Benutzers

## 54 Prüfen ob Benutzergruppe existiert

### (GROUPEXISTS)

Prüft, ob die angegebene Benutzergruppe existiert. Hierbei ist zu beachten, dass der Wert des Tags <GroupName> case-sensitive ist.

### 54.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GROUPEXISTS</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <GroupExists>
    <Request>
      <GroupName>Produktion</GroupName>
    </Request>
  </GroupExists>
</XmlAPI>
```

### 54.2 Code-Definition der Anforderung

#### 54.2.1 Gruppenname

```
<GroupName>Produktion</GroupName>
```

- Definiert den Namen der Gruppe, welche überprüft werden soll

### 54.3 Antwort

#### 54.3.1 Antwort falls die Gruppe existiert

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <GroupExists>
    <Return>
      <Exists>Yes</Exists>
    </Return>
  </GroupExists>
</XmlAPI>
```

#### 54.3.2 Antwort falls die Gruppe nicht existiert

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <GroupExists>
    <Return>
      <Exists>No</Exists>
    </Return>
  </GroupExists>
</XmlAPI>
```

## 55 Benutzergruppe erstellen

(*GROUPOCREATE*)

Erstellt eine neue Benutzergruppe.

### 55.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GROUPOCREATE</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <GroupCreate>
    <Request>
      <GroupName>Produktion</GroupName>
      <GroupDisplayName>Produktionsgruppe</GroupDisplayName>
    </Request>
  </GroupCreate>
</XmlAPI>
```

### 55.2 Code-Definitionen der Anforderung

#### 55.2.1 Interner Gruppenname

```
<GroupName>Produktion</GroupName>
```

- Definiert den Namen der Gruppe, welche erstellt werden soll

#### 55.2.2 Anzeigename

```
<GroupName>Produktion</GroupName>
```

- Definiert den Anzeigenamen der Gruppe, welche erstellt werden soll

## 56 Benutzergruppe löschen

(GROUPDELETE)

Löscht die angegebene Benutzergruppe.

### 56.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GROUPDELETE</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <GroupDelete>
    <Request>
      <GroupName>Produktion</GroupName>
    </Request>
  </GroupDelete>
</XmlAPI>
```

### 56.2 Code-Definition der Anforderung

#### 56.2.1 Gruppenname

```
<GroupName>Produktion</GroupName>
```

- Definiert den Namen der Gruppe, welche gelöscht werden soll

## 57 Benutzer in Benutzergruppe einfügen

(ADDUSERTOGROUP)

Fügt einen vorhandenen Benutzer in die angegebene Benutzergruppe ein.

### 57.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>ADDUSERTOGROUP</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <AddUserToGroup>
    <Request>
      <LoginName>maxmuster</LoginName>
      <GroupName>Produktion</GroupName>
    </Request>
  </AddUserToGroup>
</XmlAPI>
```

### 57.2 Code-Definitionen der Anforderung

#### 57.2.1 Benutzername

```
<LoginName>maxmuster</LoginName>
```

- Definiert den Namen des Benutzers, welcher in die entsprechende Benutzergruppe eingefügt werden soll

#### 57.2.2 Gruppenname

```
<GroupName>Produktion</GroupName>
```

- Definiert den Namen der Gruppe, in welche der entsprechende Benutzer eingefügt werden soll

## 58 Benutzer aus Benutzergruppe entfernen

(REMOVEUSERFROMGROUP)

Entfernt den angegebenen Benutzer aus der angegebenen Benutzergruppe.

### 58.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>REMOVEUSERFROMGROUP</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <RemoveUserFromGroup>
    <Request>
      <LoginName>maxmuster</LoginName>
      <GroupName>Produktion</GroupName>
    </Request>
  </RemoveUserFromGroup>
</XmlAPI>
```

### 58.2 Code-Definitionen der Anforderung

#### 58.2.1 Benutzername

```
<LoginName>maxmuster</LoginName>
```

- Definiert den Namen des Benutzers, welcher von der entsprechenden Benutzergruppe entfernt werden soll

#### 58.2.2 Gruppenname

```
<GroupName>Produktion</GroupName>
```

- Definiert den Namen der Gruppe, von welcher der entsprechende Benutzer entfernt werden soll

## 59 Alle Benutzergruppen eines Benutzers auflisten

(*GETUSERGROUPS*)

Gibt alle Gruppen aus, denen der Benutzer zugehörig ist.

### 59.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETUSERGROUPS</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <GetUserGroups>
    <Request>
      <UserName>maxmuster</UserName>
    </Request>
  </GetUserGroups>
</XmlAPI>
```

### 59.2 Code-Definition der Anforderung

#### 59.2.1 Benutzername

```
<UserName>maxmuster</UserName>
```

- Definition des Benutzernamens, von welchem die zugeteilten Benutzergruppen zurückgegeben werden sollen

### 59.3 Antwort

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
<UserGroups>
  <Group groupId="9d6e93ac-41e6-11dd-90e7-005056c00008">Produktion</Group>
  <Group groupId="d43a0292-49e6-11dd-bec0-005056c00008">Mitarbeiter</Group>
</UserGroups>
</XmlAPI>
```

## 60 Alle Rollen eines Benutzers auflisten

(*GETUSERROLES*)

Gibt alle Rollen aus, denen der Benutzer zugehörig ist.

### 60.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETUSERROLES</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <GetUserRoles>
    <Request>
      <UserName>maxmuster</UserName>
    </Request>
  </GetUserRoles>
</XmlAPI>
```

### 60.2 Code-Definition der Anforderung

#### 60.2.1 Benutzername

```
<UserName>maxmuster</UserName>
```

- Definition des Benutzernamens, von welchem die zugeteilten Benutzerrollen zurückgegeben werden sollen

### 60.3 Antwort

```
<?xml version="1.0" encoding="UTF-16" ?>
<XmlAPI>
  <UserRoles>
    <Role roleId="0d17eb1a-e4f7-11e0-bd71-000c29b4469b">Produktion</Role>
    <Role roleId="1e41f874-e4f7-11e0-bd71-000c29b4469b">Mitarbeiter</Role>
  </UserRoles>
</XmlAPI>
```

# 61 Signaturreport ausgeben

(GETSIGNATUREREPORT)

Speichert den Signatur-Report in die angegebene HTML-Datei.

## 61.1 Anforderung

```
<?xml version="1.0" encoding="UTF-16"?>
<XmlAPI>
  <APICall>GETSIGNATUREREPORT</APICall>
  <APIVersion>1.0</APIVersion>
  <APILanguage>de-CH</APILanguage>
  <APIMultikeyDelimiter>;</APIMultikeyDelimiter>
  <!-- <APIDebug>yes</APIDebug> -->
  <GetSignatureReport>
    <Request>
      <DocumentID>388ba221-e522-11e0-bd71-000c29b4469b</DocumentID>
      <!-- <StoreID>db22a820-e4f5-11e0-b827-00155d051700</StoreID> -->
      <FileName>Z:\XmlAPI\SignaturReport_4711.html</FileName>
    </Request>
  </GetSignatureReport>
</XmlAPI>
```

## 61.2 Code-Definition der Anforderung

### 61.2.1 Dateiname

```
<FileName>Z:\XmlAPI\SignaturReport_4711.html</FileName>
```

- Definition des Verzeichnisses, in welches der Signatur-Report gespeichert werden soll (inklusive Dateiname und -endung)

## 62 Such-Operatoren

In dieser Liste sind die möglichen Operatoren aufgelistet, welche bei Abfragen / Suchen verwendet werden können. Zu beachten ist, dass die Operatoren nicht überall verwendet werden können.

<b>Abkürzung (operator)</b>	<b>Englische Bezeichnung</b>	<b>Deutsche Bezeichnung</b>
<b>bt</b>	between	zwischen
<b>ct</b>	contains	enthält
<b>eq</b>	equal	gleich
<b>ge</b>	greater / equal	grösser / gleich
<b>gt</b>	greater (than)	grösser
<b>in</b>	value list	Werteliste
<b>le</b>	less / equal	kleiner / gleich
<b>lk</b>	like	wie
<b>lt</b>	less (than)	kleiner
<b>ne</b>	not equal	ungleich
<b>un</b>	undefined	Undefiniert
<b>in</b>	value list	Werteliste
<b>ni</b>	not in value list	Ungleich Werteliste
<b>sx</b>	soundex	Soundex