

API Dokumentation

**Accantum GmbH
Äußere Oberastr. 36/4
83026 Rosenheim**

Titel:	API Dokumentation
MC:	ACC_API
Erstelldatum:	25.08.17; HZI
Letzte Bearbeitung:	27.04.20, HZI
Revisionsnummer:	7.0
Speicherort:	API Kurzanleitung V70 April 2020.docx

© Accantum GmbH

Die Weitergabe, sowie Vervielfältigung dieser Unterlage, auch von Teilen, Verwertung und Mitteilung ihres Inhaltes ist nicht gestattet, soweit nicht ausdrücklich durch die [accantum]GmbH zugestanden. Zuwiderhandlung verpflichtet zu Schadenersatz. Alle Rechte vorbehalten.

Inhalt

1	Einleitung	3
1.1	Ressourcen	4
2	Entwicklungsumgebung.....	5
3	WCF-API: Erste Schritte	6
3.1	Authentifizierung	6
3.2	Dokument-Metadaten speichern.....	6
3.3	Binärdaten übertragen	6
3.4	Stammdaten abrufen	7
3.5	Metadaten eines Dokuments abrufen	7
3.6	Binärdaten eines Dokuments abrufen.....	7
3.7	Suche nach einem Dokument	7
3.8	Dokument ablegen	7
4	Programmierbeispiel - Sample.....	8
5	CHM Hilfedatei	13
6	Überprüfung der SOAP-Nachrichten	15
7	Empfehlungen.....	17
8	Wissenswertes.....	18
9	Beantragung eines persönlichen Identifier.....	19

1 Einleitung


Dieses Dokument ist für Software Entwickler verfasst, die sich einen ersten Überblick über die API von [accantum] und deren Möglichkeiten verschaffen möchten.

Das [accantum] Archivsystem stellt Ihnen über einen WebService eine Standard API zur Verfügung, über die Sie Archivfunktionalitäten in Ihr bestehendes System integrieren können.

Über diese API erhalten Sie Zugriff auf die nahezu gesamte Funktionalität des [accantum] Archivsystems.

[accantum] V7.0 API Dokumentation

IApiArchive.GetDocument Methode



Liefert ein Dokument zurück, wenn die eingeloggte Person die entsprechenden Rechte darauf hat.
Das Dokument ist im [AccDocumentResult](#) enthalten.

Namensraum: [Accantum.AccAPI](#)
Assembly: AccAPI (in AccAPI.dll) Version: 7.0.0.24 (7.0.0.24)

▲Syntax

C# VB C++ F#

```
AccDocumentResult GetDocument(  
    Guid documentID,  
    AccGetDocumentContext context,  
    string ticket  
)
```

Copy

Die [accantum] API unterstützt neben C#, VB, C++ und F#. Es können weitere Programmiersprachen verwendet werden, mit denen SOAP-Nachrichten versendet und empfangen werden können.

SOAP-Nachrichten sind Nachrichten im XML-Format und werden über das http-Protokoll versendet.

ACHTUNG

Sollten Sie das XML der SOAP-Nachricht selbst erstellen, dann müssen Sie die Reihenfolge der Request-Parameter beachten. So muss beispielsweise der Dokumentname *<Name>* vor dem Dokumentstatus *<DocState>* angegeben werden! Dies scheint eine Eigenart der .NET-Kommunikationsplattform WCF zu sein.

1.1 Ressourcen

Folgende Dokumentation / Ressourcen stellen wir für die Entwicklung zur Verfügung.

- [accantum] V7.0 API Dokumentation.chm (Helpfile)
- Beispiel-Projekt „WCFClientSample“

[Download API Ressourcen für \[accantum\] V7.0.0](#)

▲Beispiel

Codebeispiel zur Erstellung eines Tickets:

```
C# Copy  
  
private string CreateTicket(string a_strUser, string a_strPwd, string a_strSystemIdentifier)  
{  
    string strCredentials = string.Format("{0}:{1}", a_strUser, a_strPwd);  
    byte[] oBytes = Encoding.UTF8.GetBytes(strCredentials);  
    string strEncodedCredentials = Convert.ToBase64String(oBytes);  
  
    string strTemp = strEncodedCredentials.Replace('+', '-').Replace('/', '_').TrimEnd('=');  
    string strTicket = string.Format("{0}@{1}", strTemp, a_strSystemIdentifier);  
    return strTicket;  
}
```

Hinweise:

Das Helpfile *.chm ist im Download von [accantum] im Ordner Dokumentation enthalten.

Das Helpfile *.chm muss auf einem lokalen Laufwerk installiert werden.

„Microsoft Security Updates 896358 & 840315 block display of CHM file contents when opened from a network drive (or a UNC path)“

Siehe auch unter <http://stackoverflow.com/questions/11438634/opening-a-chm-file-produces-navigation-to-the-webpage-was-canceled>

2 Entwicklungsumgebung

Für die Nutzung der API ist eine Installation der [accantum] Dienste notwendig. Mit der mitgelieferten Demo-Lizenz ist ein Demo-Konnektor freigeschaltet, mit welchem die API verwendet werden kann. Für die Verwendung mit einer „scharfen“ Lizenz muss ein eigenes Modul bei Accantum beantragt werden.

Die API besteht aus der Web- und WCF-API.

Die Web-API besteht lediglich aus einer Web-Url, mit der je nach angegebenen Parametern ein Dokument gesucht oder geöffnet werden kann. Für die Nutzung der Web-API ist eine vollständig installierte Accantum-Version notwendig (Web-GUI und Dienste). Im Gegensatz zur WCF-API sind hier keine Programmierkenntnisse notwendig. Um z.B. ein Dokument anzuzeigen, ist lediglich der Aufruf einer URL nötig. Als Parameter werden die Authentifizierungsart, die WCF-Service-URL und die Dokument-ID angegeben:

```
http://server/accantum/webapi?auth=windows&archive=http%3a%2f%2fserver%3a50000%2f&docid= FABF5D84-5BE4-47d8-BCEB-274CDEBBF67D
```

Detailliertere Informationen erhalten Sie in unserer API-Dokumentation ([accantum] V6.0 API Dokumentation.chm) unter „How To's > Verwendung der Web-API“.

Die WCF-API wird durch unseren Windowsdienst „AccServices.exe“ gehostet. Mit dem [accantum] Configuration-Manager können Sie den Port einstellen, unter dem der Dienst zur Verfügung steht. Bei einer Standard-Installation mit nur einem Archiv wird die API unter der Adresse <http://server:42000> gehostet. Die WCF-API bildet die vollständige Accantum-Funktionalität ab.

Mit einer .NET Programmiersprache kann im Microsoft Visual Studio relativ einfach ein WCF-Service in ein eigenes Projekt eingebunden werden. Durch das Hinzufügen des WCF-Service werden automatisch Proxy-Klassen generiert, über die dann die Schnittstelle benutzt werden kann. Sollte Ihre Programmiersprache keine Proxy-Klassen aus einem WCF-Service erzeugen können, lässt sich unsere API mit etwas mehr Aufwand auch direkt per SOAP nutzen.

3 WCF-API: Erste Schritte

Wurde die Service-Reference in das Projekt eingebunden, dann kann in wenigen Schritten die API genutzt werden. Die meist benötigten Aufrufe sollen hier kurz aufgeführt werden. Code-Beispiele hierzu finden Sie in unserer kleinen Anwendung „WCFClientSample“, in der noch weitere Funktionalitäten rudimentär implementiert sind.

- Authentifizierung
- Dokument-Metadaten aufbereiten und speichern
- Binärdaten übertragen
- Stammdaten abrufen
- Metadaten eines Dokuments abrufen
- Binärdaten eines Dokuments abrufen
- Suche nach einem Dokument
- Dokument ablegen

3.1 Authentifizierung

Bei jedem Aufruf einer API-Methode muss ein Ticket übergeben werden. Dieses Ticket setzt sich aus einer Session-ID und einem Modul-Identifizier zusammen. Die Session-ID erhält man durch die Authentifizierung mit der Methode „Login()“. Der Modul-Identifizier wird fest von uns vorgegeben und lautet für den **Demo-Konnektor**: „4uaQ7b0lwdqT7xylHU15jg“. Der Demo-Konnektor ist nur in einer Demo-Lizenz freigeschaltet. Für eigene Module muss der Identifizier und damit auch das Modul bei Accantum beantragt werden (siehe auch Kap. 9).

Ein Beispiel für das Erstellen eines Tickets finden Sie auch im Helpfile unter „How To's > Authentifizierung“.

3.2 Dokument-Metadaten speichern

Ein Dokument in Accantum besteht aus einem Metadaten-Satz für das Dokument (AccDocument), einem Content (AccContent) und einem Ablagepfad (AccPath). Um ein Dokument zu archivieren müssen alle Objekte erstellt werden. Erst wenn diese Objekte mit der Methode „StoreDocument()“ gespeichert wurden, können die Binärdaten übertragen werden.

Ein Beispiel finden Sie auch im Helpfile unter „How To's > Archivierung eines Dokuments“.

3.3 Binärdaten übertragen

Da die WCF-Requests einer Größenbeschränkung unterliegen, können die Binärdaten in einer Schleife „Häppchenweise“ übertragen werden. Mit der Übertragung des letzten Häppchens muss die Übertragung als „committed“ gekennzeichnet werden.

Ein Beispiel finden Sie auch im Helpfile unter „How To's > Archivierung eines Dokuments“.

Weitere Beispiele finden Sie in unserer kleinen Beispiel-Anwendung „WCFClientSample“, in der ein paar wesentliche Funktionalitäten rudimentär implementiert sind.

3.4 Stammdaten abrufen

Wird ein Dokument archiviert, dann muss z.B. auch eine Kategorie gesetzt werden, welche in den Stammdaten gepflegt wird. Um eine Liste von Stammdaten abzurufen, wird die Methode „GetLists()“ verwendet. Über Parameter wird angegeben, welche Art von Stammdaten ausgelesen werden soll.

3.5 Metadaten eines Dokuments abrufen

Soll auf ein bereits archiviertes Dokument zugegriffen werden, können mit der Methode „GetDocument()“ die Metadaten eines Dokuments abgerufen werden.

3.6 Binärdaten eines Dokuments abrufen

Soll das Dokument dagegen angezeigt werden, können mit der Methode „GetBinaryData()“ die Binärdaten des Dokuments abgerufen und z.B. lokal gespeichert und PDF-Dokumente mit dem installierten PDF-Viewer angezeigt werden.

3.7 Suche nach einem Dokument

Natürlich kann auch nach Dokumenten gesucht werden. Mit der Methode „FindDocuments()“ wird zunächst einmal die Suche angestoßen, und liefert ein erstes Ergebnis zurück. Da eine Suche viele Ergebnisse zurückliefern kann, welche die technischen Grenzen der Übertragungsmenge sprengen können, verwenden wir einen Paging-Mechanismus, um immer nur einen Teil des Suchergebnisses zurückzugeben (Umfang kann bei den Parametern der Parameter angegeben werden). Ob eine „erweiterte Suche“ oder eine Attribut-Suche ausgeführt werden soll, wird über die Suchparameter gesteuert.

Durch die zurückgegebene Gesamtzahl (Property *TotalItemCount*) kann festgestellt werden, ob noch mehr Datensätze als die gelesenen vorhanden sind. Diese lassen sich mit der Methode „FindMoreDocuments()“ abrufen. Der hier notwendige Parameter „SearchId“ wird beim Aufruf der Methode „FindDocuments()“ zurückgegeben. Beim PagingContext muss der *RowIndex* entsprechend hochgesetzt werden.

3.8 Dokument ablegen

Soll ein Dokument in einen endgültigen Ablageort verschoben oder das Kennzeichen „revisionssicher“ gesetzt werden, so wird dies mit der Methode „BulkEdit()“ durchgeführt.

4 Programmierbeispiel - Sample

Technologie:

- .NET Framework
- Programmiersprache C#
- Anbindung an WCF-Schnittstelle der Accantum-Dienste

Voraussetzung:

- Lauffähiges Accantum (Demo-Version)
- Accantum-Lizenz mit freigeschaltetem Modul „Demo-Konnektor“

Das Beispiel-Projekt "WCFClientSample" enthält folgende Funktionalität:

- Login/Logout
- Abrufen von
 - Ablageorte
 - Kategorien
 - Ablageregeln
- Archivieren eines Dokuments
- Suche nach Dokumenten
- Abrufen der Dokument-Metadaten (inkl. Benutzerdefinierte Attribute)
- Abrufen des Dokument-Binärdaten (zur Anzeige)
- Neue Version erstellen

Folgende Screenshots zeigen die häufigsten Funktionen zur Verwendung der API auf:

1. Archivieren eines Dokuments

The screenshot shows a web application window titled "Mainframe". It has three tabs: "Archivieren", "Dokument anzeigen", and "Neue Version". The "Archivieren" tab is active. The form contains the following fields and values:

Field	Value
Datei	C:\Users\hzi\Downloads\Lieferschein15021218.PDF
Ablageort	\Ablageorte\
Relativer Pfad	[YEAR]-[Month]\day]
Dokumentstatus	Filed
Dokumentname	Lieferschein15021218.PDF
Kategorie	Lieferschein
Ablageregeln	AccCapture ohne OCR

At the bottom right of the form is a button labeled "Hochladen". At the bottom of the window, there are buttons for "Login" and "Logout", and a status indicator that says "Angemeldet als: Administrator".

Abrufen von Stammdaten aus [accantum], erzeugen von Dokumentmetadaten, Übertragung von Binärdaten sowie Ablage von Dokumenten.

2. Suche nach Dokumenten

Suchen in : <alle Ablageorte>

Suchbegriff : Lieferschein 15021

☒ Dokumentname
☐ Dokumentenbeschreibung
☐ Suchbegriffe
☒ OCR-Text Dokument
☐ OCR-Text Anhang
☐ Steuerelevante Dokumente

☐ Verzeichnisstruktur
☐ Suchen in Ablageort ...
☐ Beschreibung des Ablageorts
☐ Name des Anhangs
☐ Archiviert

☐ Erzeugungsdatum Zwischen 12.05.2016 und 12.05.2016
☐ Letzte Änderung Zwischen 12.05.2016 und 12.05.2016
☐ Aufbewahrungsfrist Zwischen 12.05.2016 und 12.05.2016

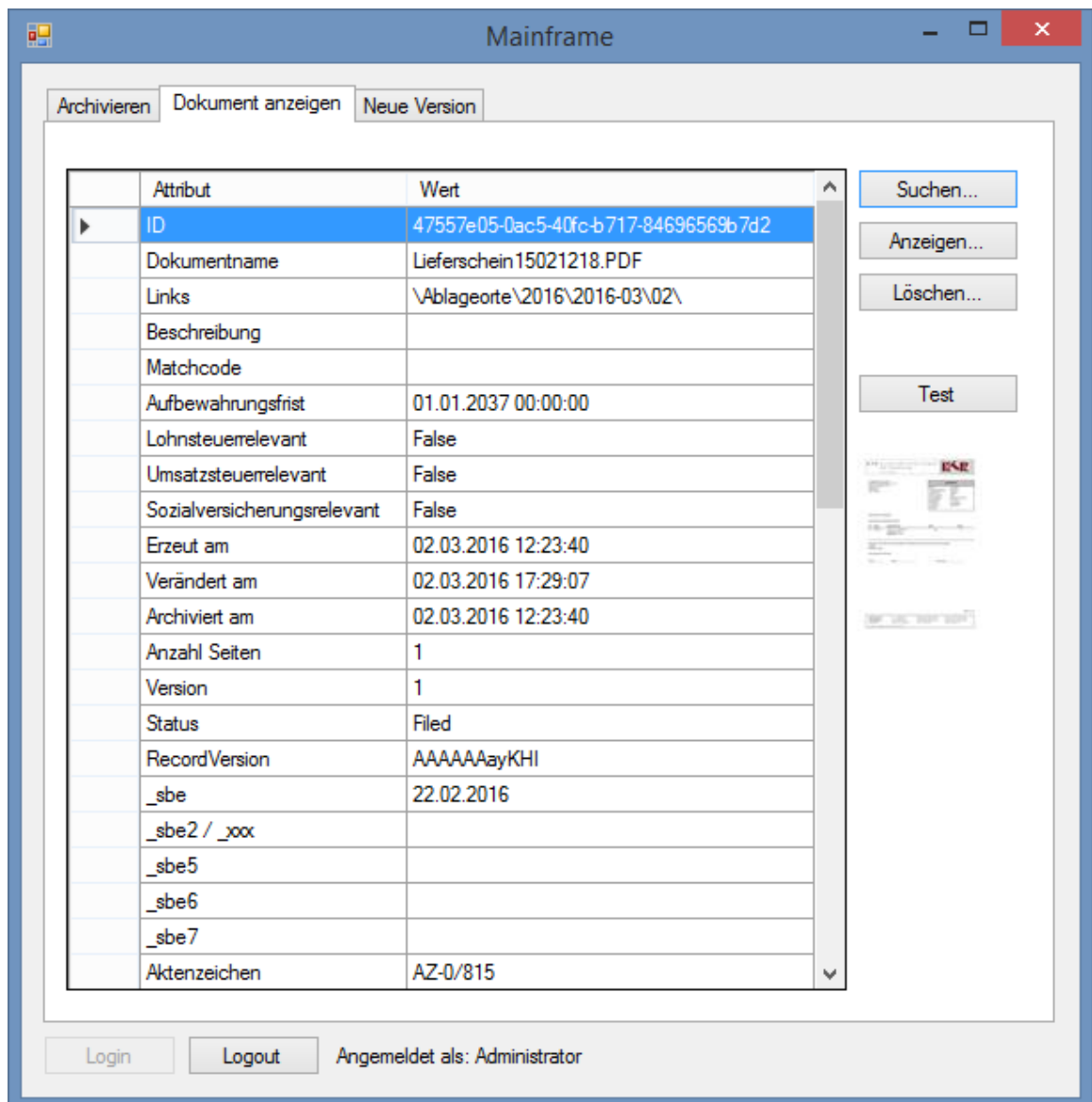
Suche starten

	Name	Path	LinkID	DocID
▶	Lieferschein15021218.PDF	\\Ablageorte\\2016\\2016-03\\02\\	9e0c302e-1a93-4022-9a25-2d2a581ce1f3	47557e05-0a

OK Abbrechen

Suche von Dokumenten mittels erweiterter Suchparameter. Dies ergänzt die Suche mittels Attributfelder.

3. Anzeige des Dokument-Inhalts



Abrufen der Dokumentbinär- und Metadaten.

4. Neue Version erstellen

The screenshot shows a web application window titled 'Mainframe'. It has three tabs: 'Archivieren', 'Dokument anzeigen', and 'Neue Version', with the last one being active. The 'Neue Version' tab contains the following elements:



- A text input field labeled 'Dokument-ID' with the value '47557e05-0ac5-40fc-b717-84696569b7d2'.
- A text input field labeled 'Datei' with the value 'C:\Users\hzi\Downloads\211591.pdf' and a blue ellipsis button to its right.
- A checkbox labeled 'Dateiname als neuen Dokumentname übernehmen' which is currently unchecked.
- A 'Hochladen' button located at the bottom right of the form area.

At the bottom of the window, there is a status bar containing a 'Login' button, a 'Logout' button, and the text 'Angemeldet als: Administrator'.

Hochladen einer neuen Dokumentversion.

5 CHM Hilfedatei

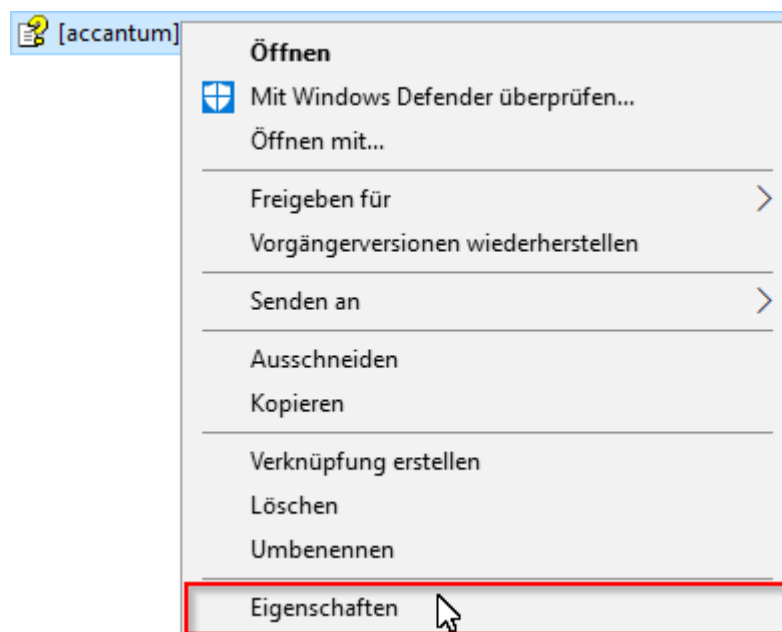
Bei der API Dokument handelt es sich um die Datei „[accantum] V7.0 API Dokumentation.**chm**“, diese wird mit einem Doppelklick geöffnet. Diese Datei ist ebenfalls im Download / Setup von [accantum] enthalten.

Accantum V700 > Documentation				
Name	Änderungsdatum	Typ	Größe	
 [accantum] V7.0 API Dokumentation.chm	18.04.2019 15:18	Kompilierte HTML...	4.079 KB	
 Acc70_API-Changes.pdf	26.04.2019 16:19	PDF-Datei	168 KB	

Compiled Helpfile

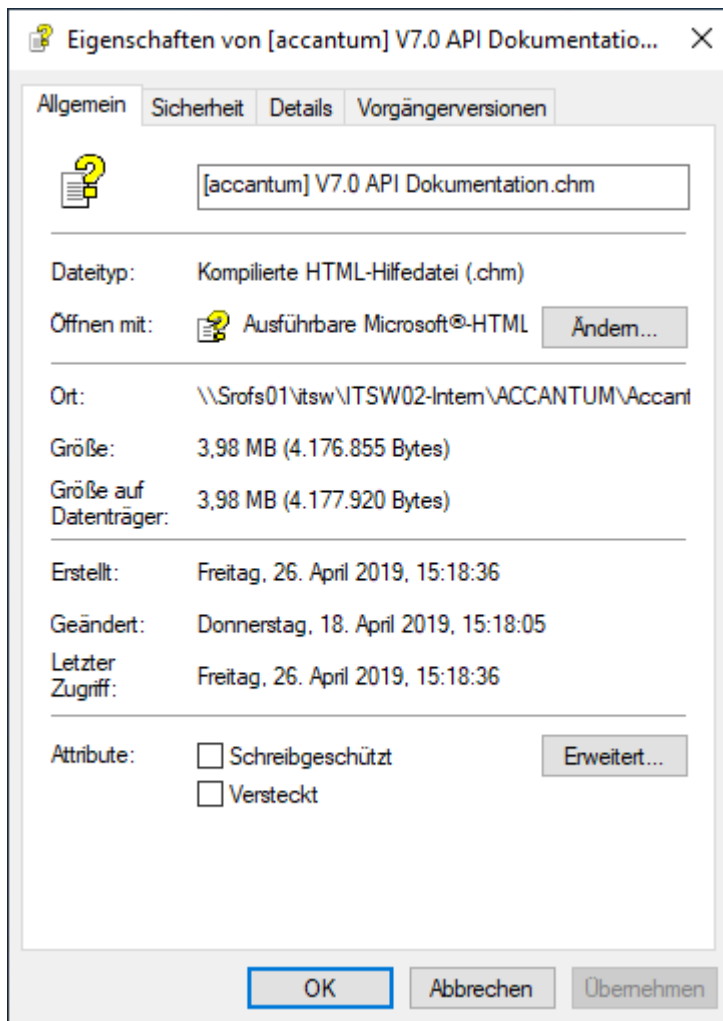
Die Datei [accantum] V7.0 API Dokumentation.chm muss auf dem **lokalen Rechner** ausgeführt werden, da über ein Netzlaufwerk der Inhalt der Dokumentation nicht angezeigt wird.

Wird unter Windows 10 die Dokumentation nicht korrekt angezeigt, müssen in den Dateieigenschaften die Sicherheitseinstellungen angepasst werden. Hierzu mit der rechten Maustaste „**Eigenschaften**“ auswählen:



Dateieigenschaften

Der Haken „**Zulassen**“ muss im Bereich **Sicherheit** gesetzt werden. Anschließend kann die API Dokumentation auch unter Windows 10 auf dem lokalen Rechner ausgeführt werden.

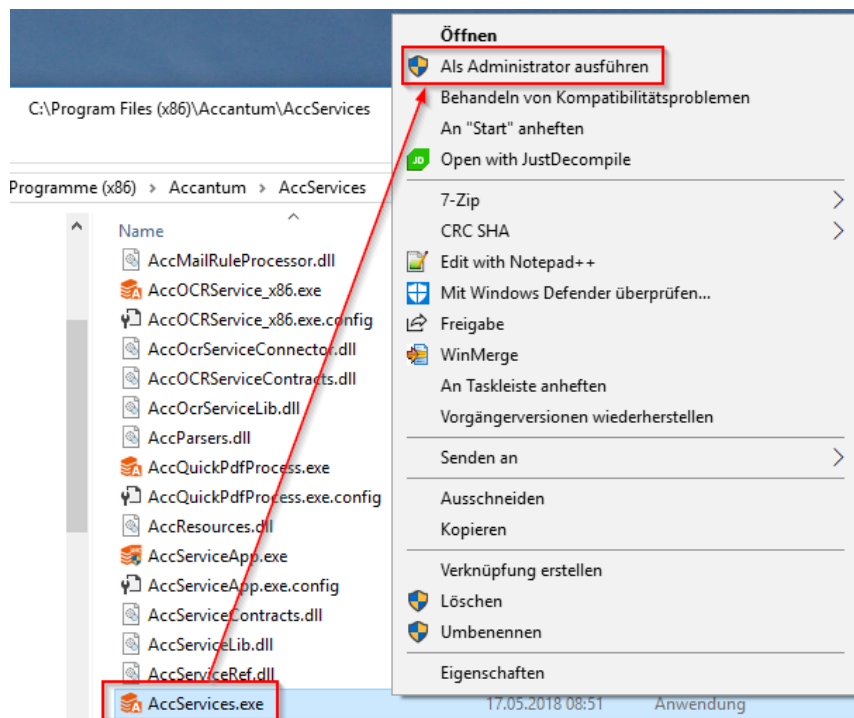


Sicherheit zum Datei öffnen „Zulassen“

6 Überprüfung der SOAP-Nachrichten

Sie haben die Möglichkeit, die SOAP-Nachrichten unseres Accantum-Service einzusehen. Dies ist vor allem dann ein bewährtes Hilfsmittel, wenn eine Funktion nicht so funktioniert, wie von Ihnen erwartet. In diesem Falle können Sie eine gleiche bzw. ähnliche Aktion mit der Web-GUI von Accantum durchführen und dann die Inhalte der SOAP-Nachrichten mit denen von Ihrer Software vergleichen. Hierzu gehen Sie wie folgt vor:

- Beenden Sie den Windowsdienst „[accantum] Verarbeitungsdienste“
- Starten Sie den Dienst AccServices.exe als Administrator (dieser liegt bei einer Standard-Installation unter C:\Programme (x86)\Accantum\AccService)



- Der Dienst öffnet sich nun mit einer Oberfläche, in der Sie neben internen Informationen auch eine Liste mit den SOAP-Nachrichten einsehen können. In dieser Liste werden alle über die API eingehenden und ausgehenden SOAP-Nachrichten protokolliert (aus Sicht des Dienstes).

AccServices.exe läuft. Zum Beenden diesen Dialog bestätigen.

Löschen DefaultLogging

Info Log Soap

370.248 bytes

Nr	In/Out	Zeit	Dauer	Aufruf	Bytes	Vollständig
202	OUT	11:29:42.303	00:00.007	GetEffectivePermissionsResponse	29932	<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap..."
201	IN	11:29:42.296		GetEffectivePermissions	613	<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap..."
200	OUT	11:29:42.288	00:00.077	GetDocumentResponse	3662	<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap..."
199	IN	11:29:42.212		GetDocument	684	<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap..."
198	OUT	11:29:42.204	00:00.017	GetNextQueueEntriesResponse	644	<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap..."
197	IN	11:29:42.187		GetNextQueueEntries	751	<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://sche..."
196	OUT	11:29:42.139	00:00.350	StoreDocumentResponse	3677	<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap..."
195	IN	11:29:41.788		StoreDocument	4327	<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap..."

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Header>
    <To mustUnderstand="1" xmlns="http://schemas.microsoft.com/ws/2005/05/addressing/none">http://wshzi:50000/Windows/Archive svc</To>
    <Action mustUnderstand="1" xmlns="http://schemas.microsoft.com/ws/2005/05/addressing/none">http://www.accantum.de/accapi/soap/IapiArchive/StoreDocument</Action>
  </s:Header>
  <s:Body>
    <StoreDocument xmlns="http://www.accantum.de/accapi/soap">
      <doc xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <ID>f33c17a-ccc0-4c3e-abe4-cd9d5058bb3c</ID>
        <RecVersion>AAAAAqLQ-Q</RecVersion>
        <CreationDate i:nil="true"></CreationDate>
        <CreationUser i:nil="true"></CreationUser>
        <ModifyDate i:nil="true"></ModifyDate>
        <ModifyUser i:nil="true"></ModifyUser>
        <AnalysisRuleID i:nil="true"></AnalysisRuleID>
        <ArchivingDate i:nil="true"></ArchivingDate>
        <Attributes>
          <AccBaseValue i:type="AccBoolValue">
            <DefID>cc2ac085f7f0-4ff4-ba5b-2f9914328bdc</DefID>
            <Value>false</Value>
          </AccBaseValue>
          <AccBaseValue i:type="AccStringValue">
            <DefID>d5dfa2d3-1502-426d-931e-a7bc28a8f3c3</DefID>
            <Value>12345</Value>
          </AccBaseValue>
          <AccBaseValue i:type="AccStringValue">
            <DefID>804e9641-4aed-4124-9746-d4de4b2c06fc</DefID>
            <Value>Itinion. Soft. GmbH</Value>
          </AccBaseValue>
          <AccBaseValue i:type="AccIntValue">
            <DefID>adaee667-69c9-469b-85af-7b3a5c323a52</DefID>
            <Value>25</Value>
          </AccBaseValue>
          <AccBaseValue i:type="AccGuidValueList">
            <DefID>fa98747-7c4e-4834-9573-c7586e4937bb</DefID>
            <ValueList xmlns:a="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
              <a:guid>08f7944d-1cbf-43f1-995b-d131d9ebfe49</a:guid>
            </ValueList>
          </AccBaseValue>
          <AccBaseValue i:type="AccNullValue">
            <DefID>54f6acbc-ab14-4e14-b82d-8155b0f8ac76</DefID>
          </AccBaseValue>
          <AccBaseValue i:type="AccBoolValue">
            <DefID>fecb28c1-7d84-4fc9-ba7a-8d0e271c477c</DefID>
            <Value>false</Value>
          </AccBaseValue>
        </Attributes>
      </StoreDocument>
    </s:Body>
  </s:Envelope>
```

- Führen Sie nun die gewünschte Aktion in Accantum aus (z.B. Speichern eines Dokuments, Suche nach einem Dokument, etc.).
- Jetzt sollten Sie alle für die Aktionen notwendigen Funktionsaufrufe inkl. Parameter bei den SOAP-Nachrichten einsehen können.
- Wenn Sie nun dieselben Aktionen mit Ihrer Software ausführen, können Sie die SOAP-Nachrichten vergleichen und anhand der Unterschiede ggf. fehlerhafte Parameter identifizieren.

7 Empfehlungen

- **Dauer einer WCF-Verbindung**
Grundsätzlich sollte versucht werden, die WCF-Verbindung so kurz wie möglich zu halten. Da jedoch das Erzeugen einer WCF-Verbindung aus Sicht der Performance relativ teuer ist, sollten die WCF-Verbindung solange gehalten werden, solange API-Aufrufe erfolgen (z.B. bei einem Rechnungsdruck sollte die Verbindung nicht pro Dokument geöffnet und wieder geschlossen werden, sondern für den gesamten Rechnungsdruck).
- **Automatische Reparatur einer WCF-Verbindung**
Es kann vorkommen, dass die WCF-Verbindung beendet wird. Es sollte deshalb dafür gesorgt werden, dass die Verbindung wieder automatisch erstellt wird. Das Ticket bleibt auch durch eine beendete WCF-Verbindung weiterhin gültig.
- **Nahezu alle API-Methoden geben ein Objekt vom Typ AccResult (oder ein davon abgeleitetes Objekt) zurück. Dieses ist auf Fehler zu prüfen.**
- **Wird mit AccResult ein Fehler zurückgegeben (HasErrors = true), konnte die aufgerufene Funktion nicht korrekt durchgeführt werden. Wird jedoch eine Warnung zurückgegeben (HasWarnings = true), so wurde die Funktion zwar vollständig ausgeführt, aber der Benutzer sollte auf ein aufgetretenes Problem aufmerksam gemacht werden. Alle weiteren zurückgegebenen Meldungen dienen lediglich der Information.**

8 Wissenswertes

- Ein Dokument besteht aus einem oder mehreren sogenannten *Contents* (entspricht im Groben einem Link auf die Binärdaten des Dokuments)
- Ein *Content* kann das Original-Dokument (pro Version), eine Seite, ein Anhang oder weitere Contents darstellen.

Dies ist aus der Historie gewachsen, als ein Dokument noch aus mehreren TIFF-Seiten bestanden hat und somit aus mehreren Original-Contents bestand. Ein PDF-Dokument besteht immer nur aus einem Original-Content pro Version. Ein Seiten-Content beinhaltet den OCR-Text und weitere Metadaten einer einzelnen Seite

9 Beantragung eines persönlichen Identifizier

Für die Erstellung von eigenen Konnektoren muss bei [accantum] ein persönlicher Identifizier beantragt werden. Mit diesem Identifizier kann Ihr Konnektor mit der Accantum-Lizenz freigeschaltet werden. Für die Ausstellung eines Identifizier benötigen wir folgende Informationen:

- Name des Moduls (max. 10 Zeichen)
- Beschreibung des Moduls (1-2 Sätze)
- Name Ihres Unternehmens (ggf. zusätzlich einen Ansprechpartner)
- Läuft der Connector
 - als Server-Dienst (1 User) oder
 - benutzt jeder User den Konnektor unter seinem Account (x User; jeder Benutzer authentifiziert sich unter seinem Namen, z.B. bei Add-Ins)
- Benutzt der Konnektor die Accantum-Benachrichtigung?

Hinweis

Der Identifizier ist nur für Ihren Connector bestimmt und darf nicht an Dritte weitergegeben werden!