

# Dokumentation zum ET\_XML\_GENERATOR

Erstellt für:  
**SKODA AUTO Deutschland GmbH**

Datum: 04.03.2016

Technische Dokumentation	ET_XML_GENERATOR	
--------------------------	------------------	--

<b><u>Autor:</u></b>	
Andrey Boivalenko	

## Inhaltsverzeichnis

- 1.1 Allgemeine Beschreibung ..... 4
- 1.2 Beschreibung des Programmes ..... 4
- 1.3 Installation des Programms ..... 4
- 1.4 Ablauf des Programmes ..... 5
  - 1.4.1 Übergabeparameter des Programmes ..... 5
  - 1.4.2 Ein Beispiel von der BAT Datei mit Übergabeparameter ..... 5
  - 1.4.3 Start des Programmes ..... 6
  - 1.4.4 Beschreibung der Programmlogik ..... 10
- 1.5 Logs ..... 11

## 1.1 Allgemeine Beschreibung

Es werden im SAP System «ET2000» mehrere Rechnungen erstellt. Die Rechnungen werden an RVS abgeschickt. Das RVS System macht die folgenden 3 Schritte.

Schritt 1: Die PDF-Dateien werden auf einem Pfad vom SAD Share abgelegt.

Schritt 2: Es wird das Programm ET\_XML\_GENERATOR aufgerufen.

Schritt 3: Wenn das Programm ET\_XML\_GENERATOR beendet und alle XML Dateien mit allen PDF Dateien in einem Pfad im SAD SHARE abgelegt wurden, werden die Belege (PDF und entsprechende XML Datei) durch den RVS-Rechner-Batch an ELO übergeben.

## 1.2 Beschreibung des Programmes

ET\_XML\_GENERATOR – ist eine Java Konsole Anwendung, die von einer Bat Datei gestartet wird. Das Ziel des Programmes ist es zu jeder PDF Datei eine entsprechende XML Datei mit gleichem Namen wie die PDF Datei zu generieren. Dies erfolgt, wenn eine PDF Datei sowohl der Namenskonvention entspricht als auch eine gültige Händlernummer beinhaltet. Dabei werden zu diesen PDF Dateien noch eine VZ-Nummer hinzugefügt.

## 1.3 Installation des Programms

Das Programm ist circa 10 MB groß und beinhaltet die 3 Elemente:

1. Das Verzeichnis mit notwendigen JAR Bibliotheken.
2. Eine BAT Datei. Sie beinhalten die 4 Übergabeparameter, die es möglich machen, die JAR Datei bzw. das Programm auszuführen.
3. Eine JAR Datei. Das ist der Kern des Programmes.

Das Programm ist circa 10 MB groß und kann überall im System installiert werden. Dabei ist zu beachten, dass die 3 obengenannten Elemente in einem Verzeichnis abgelegt sein müssen (siehe Abbildung 1).

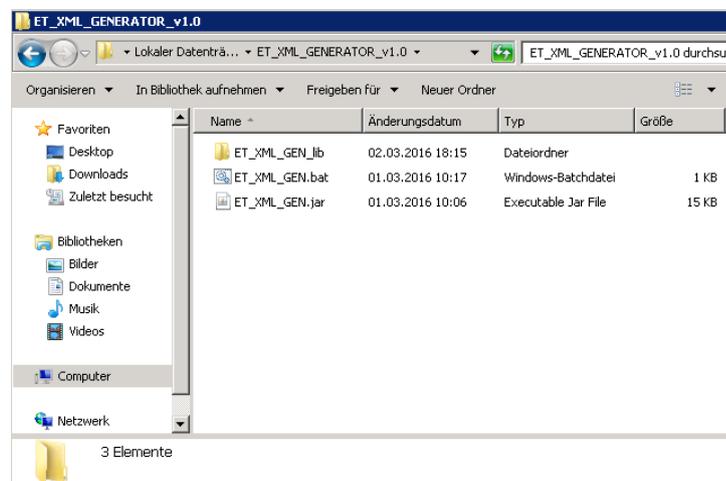


Abbildung 1

## 1.4 Ablauf des Programmes

### 1.4.1 Übergabeparameter des Programmes

Die Eingangsparameter (Übergabeparameter) des Programms ET\_XML\_GENERATOR sind:

1) path\_to\_files

Dieser Parameter muss vom Typ String sein. Der Dateipfad zu den PDF Dateien.

2) ip\_of\_database

Dieser Parameter muss vom Typ String sein. IP-Adresse zur DB2 Datenbank.

3) dd\_login\_user

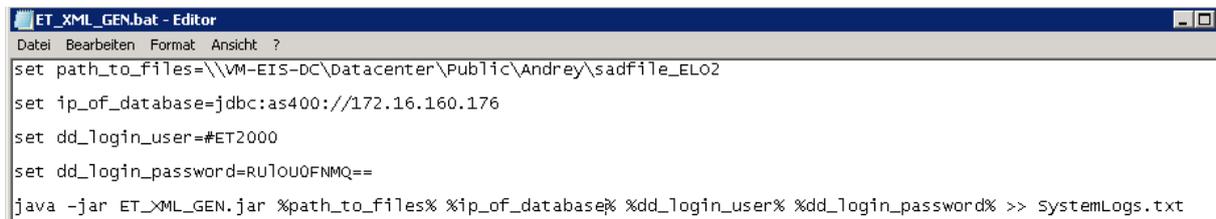
Dieser Parameter muss vom Typ String sein. Mit dem meldet sich das Programm an der Datenbank an.

4) dd\_login\_password

Dieser Parameter muss vom Typ String sein. Base 64 codiertes Kennwort des dd\_login\_user

### 1.4.2 Ein Beispiel von der BAT Datei mit Übergabeparameter

Auf der Abbildung 2 wird gezeigt, wie eine BAT Datei mit Übergabeparameter des Programmes aussieht.



```
ET_XML_GEN.bat - Editor
Datei Bearbeiten Format Ansicht ?
set path_to_files=\\VM-EIS-DC\Datacenter\Public\Andrey\sadfile_ELO2
set ip_of_database=jdbc:as400://172.16.160.176
set dd_login_user=#ET2000
set dd_login_password=RU7OU0FNMQ==
java -jar ET_XML_GEN.jar %path_to_files% %ip_of_database% %dd_login_user% %dd_login_password% >> systemLogs.txt
```

Abbildung 2

Die ersten 4 Zeilen stellen Übergabeparameter ein.

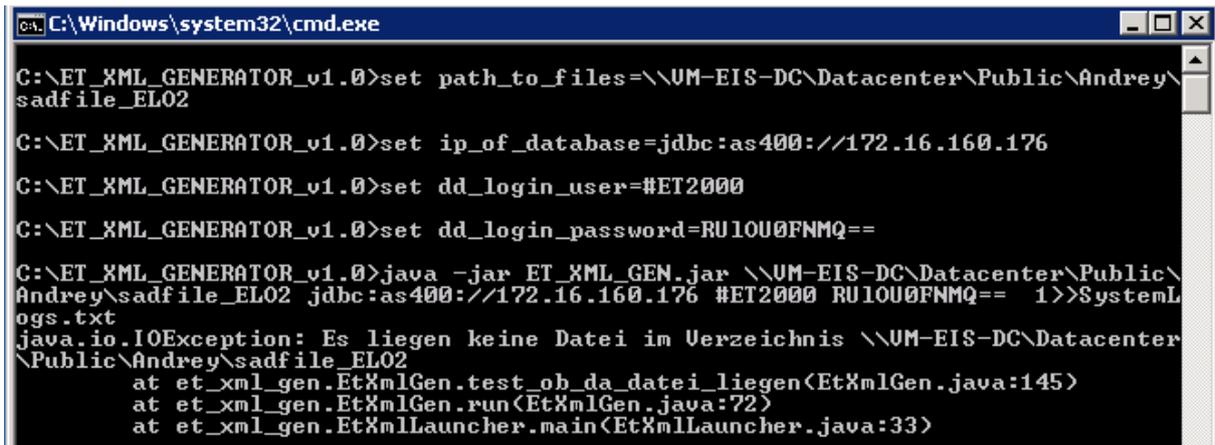
Die letzte Zeile der BAT Datei führt die JAR Datei bzw. das Programm mit der Übergabeparameter aus.

### 1.4.3 Start des Programmes

#### 1.4.3.1 PDF-Dateien in das Verzeichnis anlegen

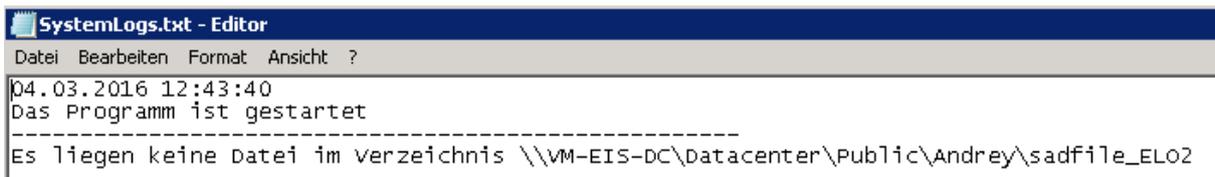
Damit das Programm korrekt arbeitet, müssen bereits PDF Dateien im SAD Share Verzeichnis existieren.

Wenn dies nicht der Fall ist, schreibt das Programm einen Fehler in die Konsole und in das Systemlog der TXT Datei, die automatisch nach dem ersten Aufruf des Programms erzeugt wird (siehe die Abbildung 3 und die Abbildung 4)



```
C:\Windows\system32\cmd.exe
C:\ET_XML_GENERATOR_v1.0>set path_to_files=\\UM-EIS-DC\Datacenter\Public\Andrey\sadfile_ELO2
C:\ET_XML_GENERATOR_v1.0>set ip_of_database=jdbc:as400://172.16.160.176
C:\ET_XML_GENERATOR_v1.0>set dd_login_user=#ET2000
C:\ET_XML_GENERATOR_v1.0>set dd_login_password=RUI0U0FNMQ==
C:\ET_XML_GENERATOR_v1.0>java -jar ET_XML_GEN.jar \\UM-EIS-DC\Datacenter\Public\Andrey\sadfile_ELO2 jdbc:as400://172.16.160.176 #ET2000 RUI0U0FNMQ== 1>>SystemLogs.txt
java.io.IOException: Es liegen keine Datei im Verzeichnis \\UM-EIS-DC\Datacenter\Public\Andrey\sadfile_ELO2
    at et_xml_gen.EtXmlGen.test_ob_da_datei_liegen(EtXmlGen.java:145)
    at et_xml_gen.EtXmlGen.run(EtXmlGen.java:72)
    at et_xml_gen.EtXmlLauncher.main(EtXmlLauncher.java:33)
```

Abbildung 3

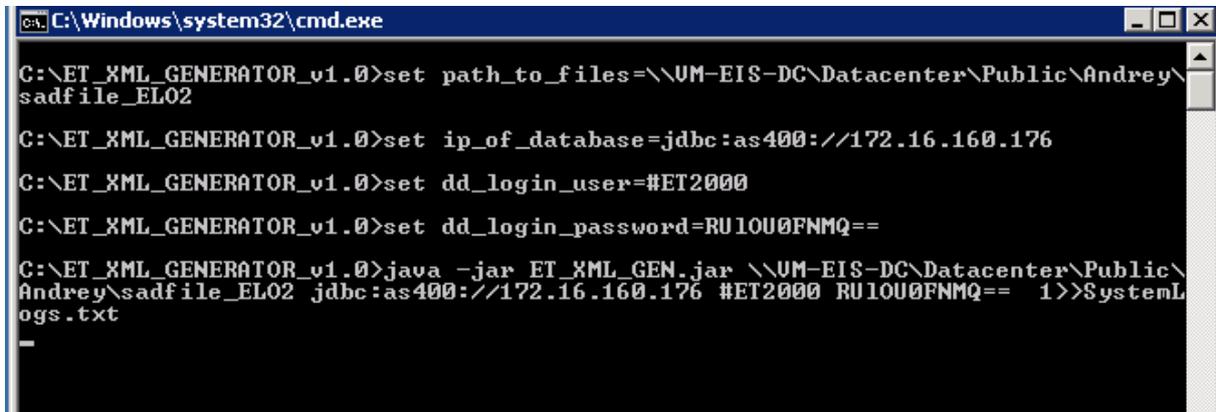


```
SystemLogs.txt - Editor
Datei Bearbeiten Format Ansicht ?
04.03.2016 12:43:40
Das Programm ist gestartet
-----
Es liegen keine Datei im Verzeichnis \\UM-EIS-DC\Datacenter\Public\Andrey\sadfile_ELO2
```

Abbildung 4

### 1.4.3.2 Bat Datei ausführen

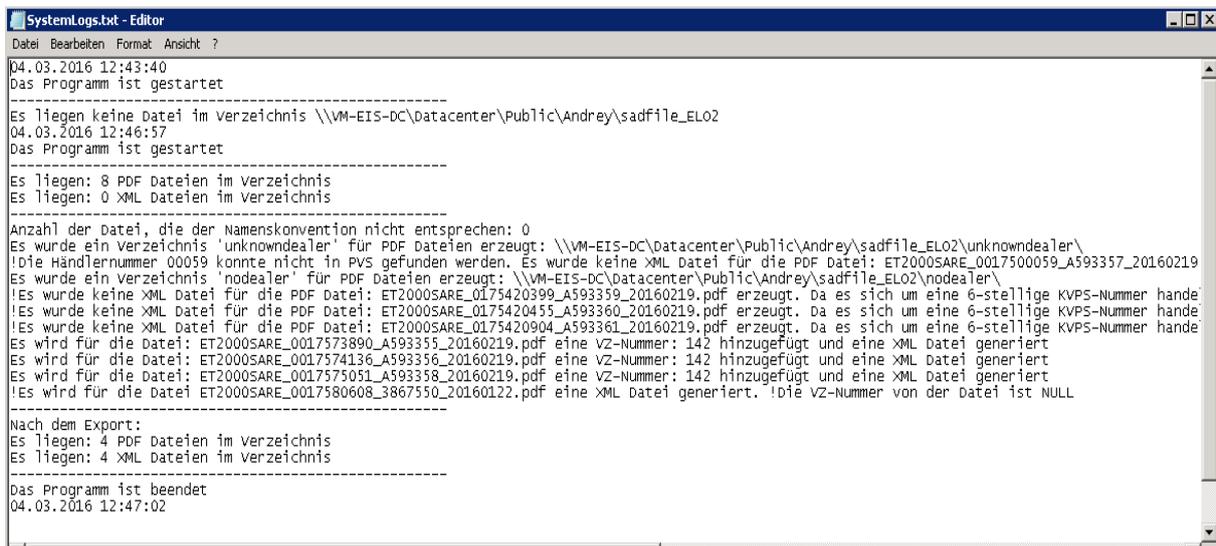
Wenn PDF Dateien im Verzeichnis liegen, startet das Programm ohne Fehlermeldung. Es muss dafür die BAT Datei ausgeführt werden. Es dauert einige Zeit (circa 1 Minute für 300 Belege) bis alle Belege durchgelaufen sind (siehe Abbildung 5).



```
C:\Windows\system32\cmd.exe
C:\ET_XML_GENERATOR_v1.0>set path_to_files=\\UM-EIS-DC\Datacenter\Public\Andrey\sadfile_ELO2
C:\ET_XML_GENERATOR_v1.0>set ip_of_database=jdbc:as400://172.16.160.176
C:\ET_XML_GENERATOR_v1.0>set dd_login_user=#ET2000
C:\ET_XML_GENERATOR_v1.0>set dd_login_password=RUI0U0FNMQ==
C:\ET_XML_GENERATOR_v1.0>java -jar ET_XML_GEN.jar \\UM-EIS-DC\Datacenter\Public\Andrey\sadfile_ELO2 jdbc:as400://172.16.160.176 #ET2000 RUI0U0FNMQ== 1>>SystemLogs.txt
```

Abbildung 5

Man sieht auf der Abbildung 6, dass 8 PDF Dateien im Verzeichnis liegen. Zu jeder PDF Datei wurde eine entsprechende XML Datei mit gleichem Namen wie die PDF Datei generiert. Dabei wurde in jeder PDF-Datei, die sowohl der Namenskonvention entspricht als auch eine gültige Händlernummer beinhaltet, eine entsprechende VZ-Nummer hinzugefügt (siehe die Abbildung 6).



```
SystemLogs.txt - Editor
Datei Bearbeiten Format Ansicht ?
04.03.2016 12:43:40
Das Programm ist gestartet
-----
Es liegen keine Datei im Verzeichnis \\UM-EIS-DC\Datacenter\Public\Andrey\sadfile_ELO2
04.03.2016 12:46:57
Das Programm ist gestartet
-----
Es liegen: 8 PDF Dateien im Verzeichnis
Es liegen: 0 XML Dateien im Verzeichnis
-----
Anzahl der Datei, die der Namenskonvention nicht entsprechen: 0
Es wurde ein Verzeichnis 'unknowndealer' für PDF Dateien erzeugt: \\UM-EIS-DC\Datacenter\Public\Andrey\sadfile_ELO2\unknowndealer\
!Die Händlernummer 00059 konnte nicht in PVS gefunden werden. Es wurde keine XML Datei für die PDF Datei: ET2000SARE_0017500059_A593357_20160219
Es wurde ein Verzeichnis 'nodealer' für PDF Dateien erzeugt: \\UM-EIS-DC\Datacenter\Public\Andrey\sadfile_ELO2\nodealer\
!Es wurde keine XML Datei für die PDF Datei: ET2000SARE_0175420390_A593359_20160219.pdf erzeugt. Da es sich um eine 6-stellige KVPS-Nummer handelt
!Es wurde keine XML Datei für die PDF Datei: ET2000SARE_0175420455_A593360_20160219.pdf erzeugt. Da es sich um eine 6-stellige KVPS-Nummer handelt
!Es wurde keine XML Datei für die PDF Datei: ET2000SARE_0175420904_A593361_20160219.pdf erzeugt. Da es sich um eine 6-stellige KVPS-Nummer handelt
Es wird für die Datei: ET2000SARE_0017573890_A593355_20160219.pdf eine VZ-Nummer: 142 hinzugefügt und eine XML Datei generiert
Es wird für die Datei: ET2000SARE_0017574136_A593356_20160219.pdf eine VZ-Nummer: 142 hinzugefügt und eine XML Datei generiert
Es wird für die Datei: ET2000SARE_0017575051_A593358_20160219.pdf eine VZ-Nummer: 142 hinzugefügt und eine XML Datei generiert
!Es wird für die Datei ET2000SARE_0017580608_3867550_20160122.pdf eine XML Datei generiert. !Die VZ-Nummer von der Datei ist NULL
-----
Nach dem Export:
Es liegen: 4 PDF Dateien im Verzeichnis
Es liegen: 4 XML Dateien im Verzeichnis
-----
Das Programm ist beendet
04.03.2016 12:47:02
```

Abbildung 6

Auf der Abbildung 6 wird gezeigt, dass 3 PDF Dateien eine falsche Händlernummer (6-stellige KVPS-Händlernummer) hatten. Es geht in diesem Fall um eine 6-stellige KVPS Händlernummer von Belegen. Diese Dateien wurden in das Verzeichnis ...nodealer verschoben. (Wenn dieses Verzeichnis noch nicht existiert, wird ebenfalls generiert).

Wenn die KVPS-Nummer 5 stellig ist, aber in PVS unbenannt ist, wird dann die PDF Datei nach dem Verzeichnis ...unknowndealer zur weiteren Bearbeitung verschoben. (Wenn dieses Verzeichnis noch nicht existiert, wird ebenfalls generiert).

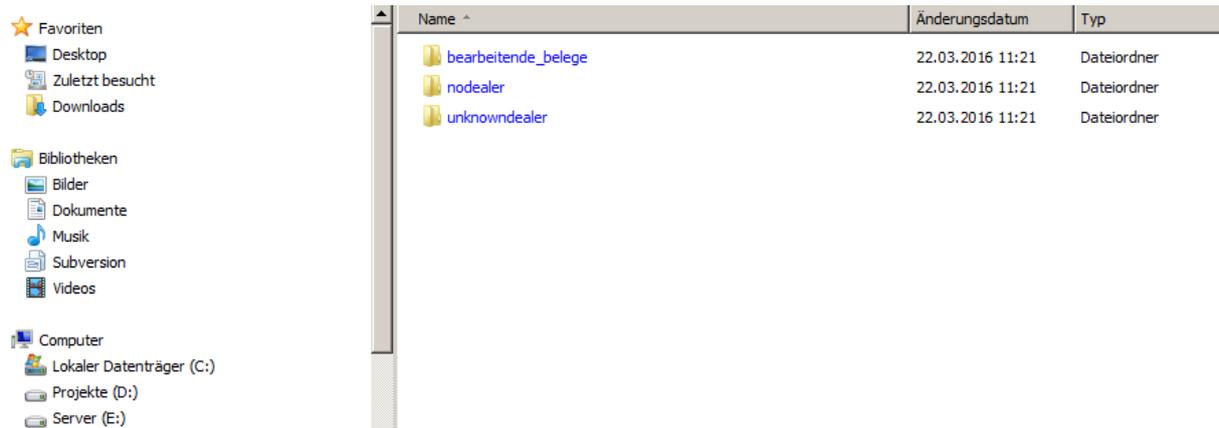


Abbildung 7

Auf der Abbildung 7 sieht man das Verzeichnis „bearbeitende\_belege“ mit schon bearbeitenden Belegen.

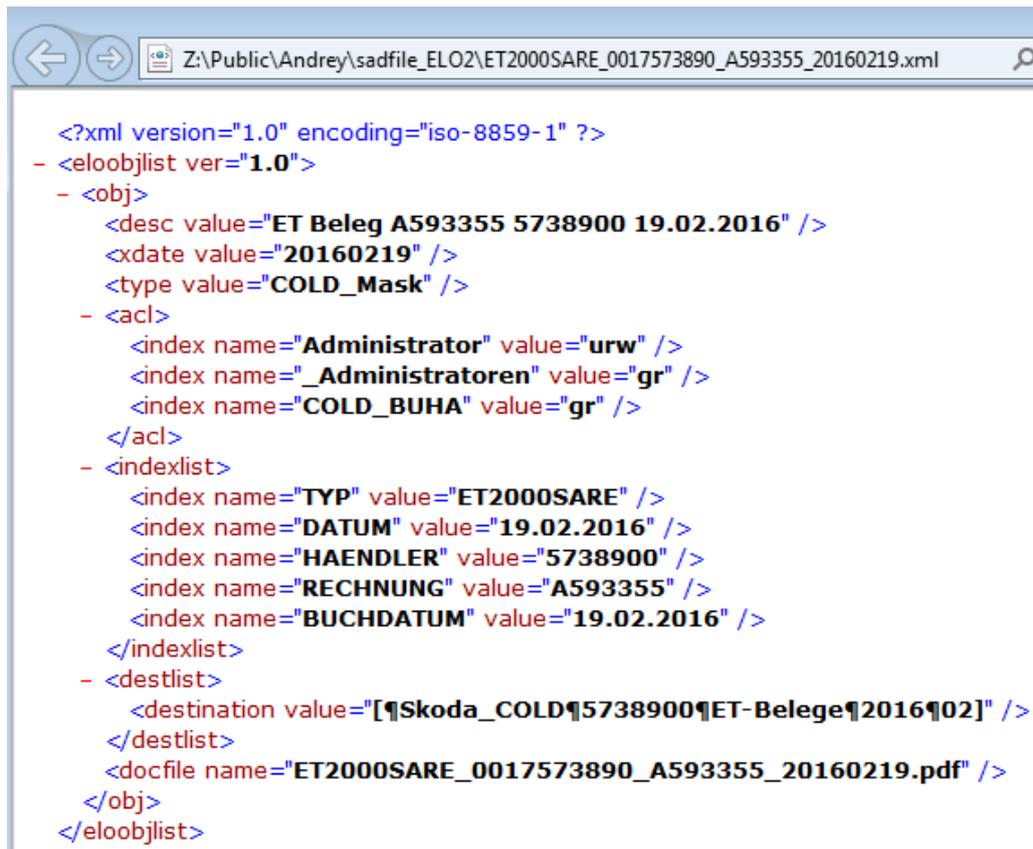
Auf den Abbildungen 8 und 9 sieht man den gleichen Beleg, zunächst ohne VZ-Nummer und dann (nach dem Ablauf des Programmes) mit einer VZ-Nummer.

Wiederholungsdruck		Sammelrechnung Nr. A593355		Datum 19.02.2016		Seite: 1		
Autohaus Goss GmbH Steuernummer:00722810800 Weiherhausstraße 8 64646 Heppenheim		Kundennummer: <b>17573890</b>		Route: 18/H01 Kundenleitzahl: <b>Y16</b> Sparte(Fabrikat): 13		SkodaAuto Deutschland GmbH Max-Planck-Str. 3-5 64331 Weiterstadt		
BWL-Nr.	Gespr.o.USt.	Handelssp.inkl. Zusatzrabatte	Einstandspreis	Handlingskosten sonst.Zu/Abschl.	Nettowert	Umsatzsteuer inkl. Altwertsteuer	Gesamtbetrag	Altwert
966552659	6,13	0,00	6,13	0,00	6,13	1,16	7,29	0,00 EUR

Abbildung 8. Beleg ohne VZ-Nummer

Wiederholungsdruck		Sammelrechnung Nr. A593355		Datum 19.02.2016		Seite: 1		
Autohaus Goss GmbH Steuernummer:00722810800 Weiherhausstraße 8 64646 Heppenheim		Kundennummer: <b>17573890</b>		Route: 18/H01 Kundenleitzahl: <b>Y16</b> Sparte(Fabrikat): 13 <b>VZ: 142</b>		SkodaAuto Deutschland GmbH Max-Planck-Str. 3-5 64331 Weiterstadt		
BWL-Nr.	Gespr.o.USt.	Handelssp.inkl. Zusatzrabatte	Einstandspreis	Handlingskosten sonst.Zu/Abschl.	Nettowert	Umsatzsteuer inkl. Altwertsteuer	Gesamtbetrag	Altwert
966552659	6,13	0,00	6,13	0,00	6,13	1,16	7,29	0,00 EUR
966559190	10,95	1,75	9,20	0,00	9,20	1,75	10,95	0,00 EUR

Abbildung 9. Beleg mit einer VZ-Nummer



```
<?xml version="1.0" encoding="iso-8859-1" ?>
- <elobjlist ver="1.0">
- <obj>
  <desc value="ET Beleg A593355 5738900 19.02.2016" />
  <xdate value="20160219" />
  <type value="COLD_Mask" />
- <acl>
  <index name="Administrator" value="urw" />
  <index name="_Administratoren" value="gr" />
  <index name="COLD_BUHA" value="gr" />
</acl>
- <indexlist>
  <index name="TYP" value="ET2000SARE" />
  <index name="DATUM" value="19.02.2016" />
  <index name="HAENDLER" value="5738900" />
  <index name="RECHNUNG" value="A593355" />
  <index name="BUCHDATUM" value="19.02.2016" />
</indexlist>
- <destlist>
  <destination value="[¶Skoda_COLD¶5738900¶ET-Belege¶2016¶02]" />
</destlist>
  <docfile name="ET2000SARE_0017573890_A593355_20160219.pdf" />
</obj>
</elobjlist>
```

Abbildung 10

Auf der Abbildung 10 wird eine generierte XML Datei gezeigt, die dem Beleg von der Abbildung 9 entspricht.

#### 1.4.4 Beschreibung der Programmlogik

Anhand der Eingangsparameter prüft das Programm die folgenden Bedingungen um das Programm weiter durchzuführen:

- 1) Ob eine Verbindung zum SAD SHARE aufgebaut werden kann.
- 2) Ob PDF-Dateien im Pfad vom SAD SHARE vorhanden sind.
- 3) Ob PDF-Dateien die in SAD SHARE liegen, der Namenskonvention entsprechen.
- 4) Ob in SAD SHARE Rechte für Lesen/Schreiben existieren.
- 5) Ob eine Connection zur DB2/AS/400 aufgebaut werden kann (es wird sowohl „dd\_login\_user“ als auch „dd\_login\_password“ geprüft).

Wenn die oben genannten 5 Parameter korrekt und ohne Exceptions durchgeführt werden, macht das Programm die folgenden 6 Schritte. Wenn dies nicht der Fall ist, wird eine Fehlermeldung mit Verweis auf die Ursache generiert. Es werden alle aufgetretene Fehler in SystemLogs TXT Datei aufgeschrieben. Wenn diese Datei nicht existieren sollte, wird sie erzeugt.

- 1) Es werden alle Partnernummern (Händlernummern) aus der EXPORT VIEW von der Tabelle PVS.Partner von der AS/400 Datenbank (EIS2) gelesen und in ein lokales Array gespeichert.
- 2) Es werden alle PDF-Dateien von SAD SHARE gelesen und folgende Werte von den Namen diesen PDF Dateien ermittelt und entsprechend in ein lokales Array gespeichert:
  - a) Belegtyp
  - b) Händlerner\_5
  - c) Belegnummer
  - d) Belegdatum
- 3) Es wird festgestellt, ob die Händlernerummern der PDF Dateinamen und die Händlernerummern von der EXPORT VIEW übereinstimmen.

Wenn das nicht der Fall ist, kann das zwei Ursachen haben:

- A) Es wurde für die PDF Datei ein falscher Name vergeben.
- B) Es handelt sich um eine 6-stellige KVPS-Nummer.

In beiden Fällen (A und B) werden die entsprechenden Dateien in bestimmte Verzeichnisse mit einem Logeintrag verschoben.

- 4) Es werden anhand der Namen der PDF- Dateien die entsprechenden XML- Dateien generiert.
- 5) Die Logs dazu werden (siehe 4.) in die Tabelle ET\_BELEGE\_LOG gespeichert.

```
SystemLogs.txt - Editor
Datei Bearbeiten Format Ansicht ?
04.03.2016 15:10:55
Das Programm ist gestartet
-----
Es liegen: 4 PDF Dateien im Verzeichnis
Es liegen: 4 XML Dateien im Verzeichnis
-----
Anzahl der Datei, die der Namenskonvention nicht entsprechen: 0
Es wurde schon für die Datei: ET2000SARE_0017573890_A593355_20160219.pdf eine VZ-Nummer: 142 hinzugefügt und eine XML Datei generiert
Es wurde schon für die Datei: ET2000SARE_0017574136_A593356_20160219.pdf eine VZ-Nummer: 142 hinzugefügt und eine XML Datei generiert
Es wurde schon für die Datei: ET2000SARE_0017575051_A593358_20160219.pdf eine VZ-Nummer: 142 hinzugefügt und eine XML Datei generiert
Die VZ-Nummer von der Datei: ET2000SARE_0017580608_3867550_20160122.pdf ist NULL
-----
Nach dem Export:
Es liegen: 4 PDF Dateien im Verzeichnis
Es liegen: 4 XML Dateien im Verzeichnis
-----
Das Programm ist beendet
04.03.2016 15:10:57
```

Abbildung 11

6) Es wird in jeder PDF-Datei eine VZ-Nummer eingeschrieben. Wird das Programm ein zweites Mal ausgeführt, erkennt das Programm, dass bereits eine VZ-Nummer erstellt wurde. Diese Information wird in das Log geschrieben (siehe Abbildung 11)

## 1.5 Logs

Es werden nach jedem Ablauf des Programms Log-Dateien in der SystemLogs.txt gespeichert. Das Programm protokolliert folgendes:

- 1) Ob eine Verbindung zum SAD SHARE aufgebaut werden kann. Wenn es nicht der Fall ist, wird dies protokolliert.
- 2) Ob PDF-Dateien im Pfad vom SAD SHARE vorhanden sind. Wenn es nicht der Fall ist, wird dies protokolliert.
- 3) Ob PDF-Dateien die in SAD SHARE liegen, der Namenskonvention entsprechen. Wenn dies nicht der Fall ist, wird dies protokolliert.
- 4) Ob in SAD SHARE Berechtigungsrechte für Lesen/Schreiben existieren. Wenn dies nicht der Fall ist, wird dies protokolliert.
- 5) Ob eine Connection zur DB2/AS/400 existiert (es wird sowohl „dd\_login\_user“ als auch „dd\_login\_password“ geprüft). Wenn dies nicht der Fall ist, wird es protokolliert.
- 6) Es wird zudem protokolliert, wenn für PDF Dateien keine XML Dateien erzeugt werden bzw. wenn die PDF Dateien keine gültige Händlernummer haben.