

a:control Fileparser / XML Funktionalität



Allen Coding GmbH
Friedrich-Bergius-Ring 30
97076 Würzburg
Germany

Phone: +49 (0) 931-25076-0
FAX: +49 (0) 931 25076-50

E-Mail: support@allencoding.com
Web: www.allencoding.com

Inhaltsverzeichnis

1. SYSTEMVORAUSSETZUNGEN	3
2. FILEPARSER	5
• Aktivierung des Fileparsers	5
• Einstellungen im a:control	6
3. XML	7
Action Tags:	9
Beispiele:	10
Kundenbeispiel mit Füllen von Variablen aus einem Excel-File:	11
Beispiel mit Füllen von variablen Kundendaten aus einer XML-Datei:	12

Hinweis:

Sämtliche Screenshots und Abbildungen (Bildschirmfotos) der Anwendung in diesem Dokument sind als beispielhafte Oberflächen anzusehen, und unterliegen dem Copyright der Allen Coding GmbH.

1. Systemvoraussetzungen

PC-Hardware	Anforderungen
Installiertes a:control ab Version 1.8.3	CPU mit mindestens 1.0 GHz Prozessorgeschwindigkeit
Physikalischer Hauptspeicher	Mindestens erforderlich, um die Installation abzuschließen: 1 GB Empfehlung: 2 GB
Freier Festplattenspeicher	30 MB für Programminstallation + Speicherplatz für Druckformate
Maus, Tastatur, Monitor oder Touch-Screen	Auflösung Monitor mindestens 800x600, 16 bit Farbtiefe
Netzwerkkarte	Ethernet
RS232 / COM Ports	Je ein COM-Port für jeden seriell angeschlossenen Drucker
USB- Ports	USB A für Datenaustausch mit USB-Stick
Software	Windows 7 SP1 Windows XP embedded ab Service Pack 3 Windows 10 Microsoft .NET Framework 4.0

Es ist durchaus möglich, dass die Software auf abweichenden Systemen einwandfrei läuft, jedoch könnte die Programmpformance unter schwächerer Hardware leiden. Vor allem ausreichend freier Arbeitsspeicher + Festplattenspeicher sind ein wichtiger Faktor für eine einwandfreie Programmausführung.

Drucker-Hardware	Anforderungen
Druckertyp	NG2, NG2+, NG4, NG4+, NG6, NG6E, NG8, NG8E
Firmware	Aktuellste Druckerfirmware
Konnektivität	Seriell oder Ethernet Netzwerk

53-System

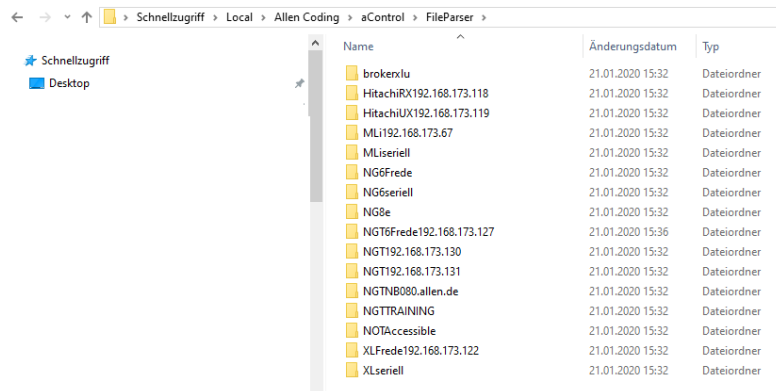
Drucker -Hardware	Anforderungen
Druckertyp	53LTi, 53XL80, 53LTc 53XL40, 53XL80u, 53XLC, XL5000, XL5000C
Konnektivität	Seriell

MLi-System

Drucker -Hardware	Anforderungen
Druckertyp	MLi
Konnektivität	Seriell oder Ethernet Netzwerk

2. FileParser

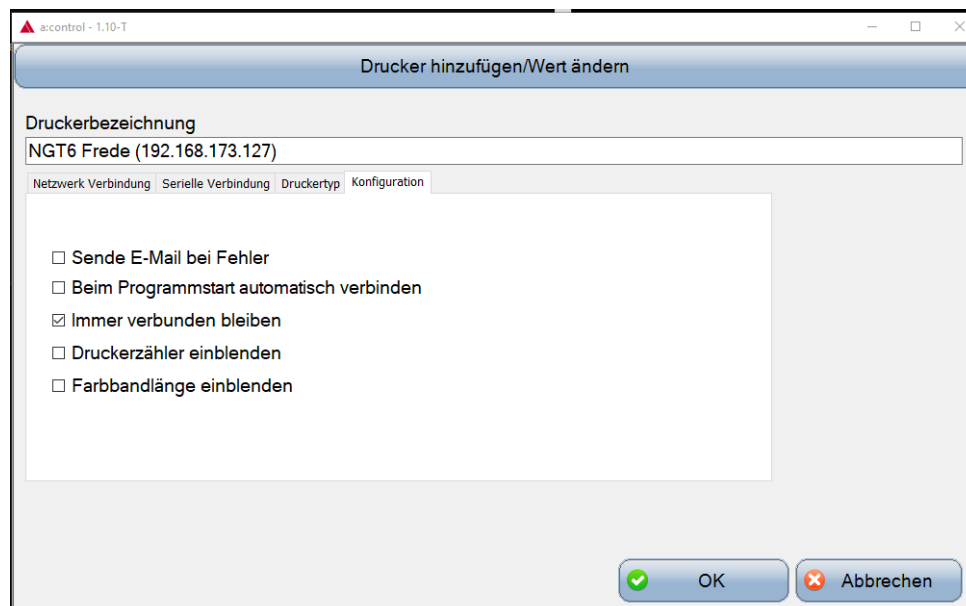
a:control legt für jeden Drucker automatisch ein Fileparserverzeichnis an.



Name	Änderungsdatum	Typ
brokerxlu	21.01.2020 15:32	Dateiordner
HitachiRX192.168.173.118	21.01.2020 15:32	Dateiordner
HitachiUX192.168.173.119	21.01.2020 15:32	Dateiordner
MLi192.168.173.67	21.01.2020 15:32	Dateiordner
MLiseriall	21.01.2020 15:32	Dateiordner
NG6Frede	21.01.2020 15:32	Dateiordner
NG6seriall	21.01.2020 15:32	Dateiordner
NG8e	21.01.2020 15:32	Dateiordner
NGT6Frede192.168.173.127	21.01.2020 15:36	Dateiordner
NGT192.168.173.130	21.01.2020 15:32	Dateiordner
NGT192.168.173.131	21.01.2020 15:32	Dateiordner
NGTNB080.allen.de	21.01.2020 15:32	Dateiordner
NGTTRAINING	21.01.2020 15:32	Dateiordner
NOTAccessibile	21.01.2020 15:32	Dateiordner
XLFrede192.168.173.122	21.01.2020 15:32	Dateiordner
XLseriall	21.01.2020 15:32	Dateiordner

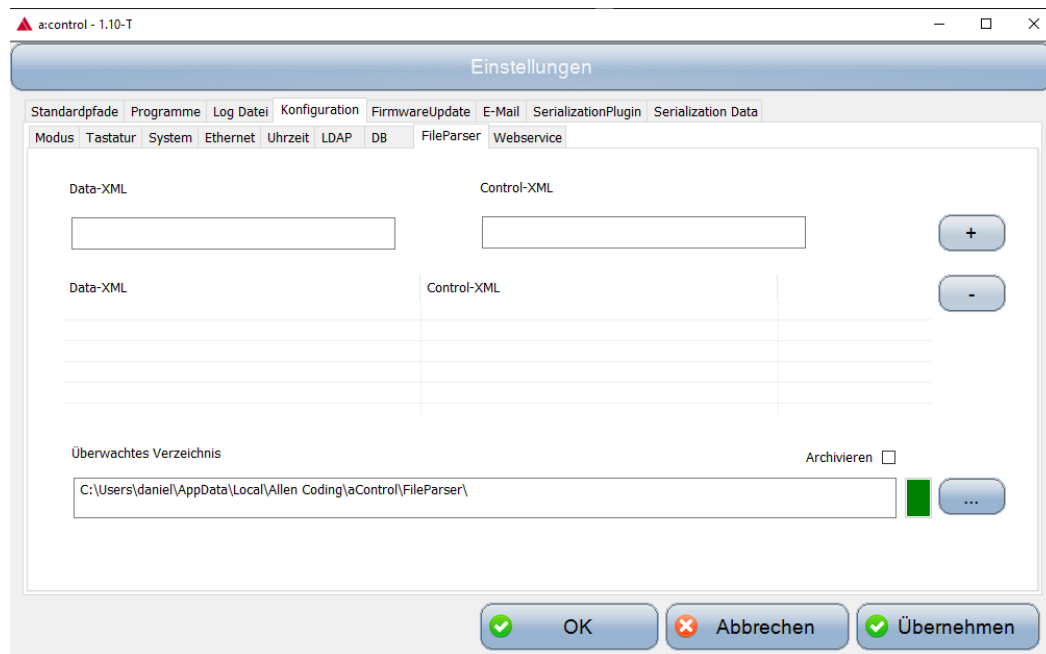
- Aktivierung des Fileparsers**

Sobald ein Drucker verbunden wird, wird im entsprechenden Druckerverzeichnis nach Bildern oder Fileparser-XML Dateien gesucht. Es ist daher oft sinnvoll, den Drucker automatisch bei Programmstart zu verbinden. Dies wird in der Konfiguration der Druckereinstellungen definiert:



- **Einstellungen im a:control**

Die Einstellungen für den Fileparser finden Sie in den Programmeinstellungen:



- **Anwenden von Druckformaten / XML-Dateien**

wenn nun ein Druckformat in ein solches Unterverzeichnis kopiert wird, werden die Formate auf die entsprechenden Drucker geladen. Ist die Datei eine gültige a:control XML Datei, wird diese angewendet.

Eine XML-Datei kann auch auf einen Drucker per Drag-Drop gezogen werden.

3. XML

Erzeugen eines XML Templates:

Verbinden Sie sich mit einem oder mehreren Druckern und laden Sie dort ein Format, z.B. ein Format mit Variablen. Nachdem Sie das Format geladen haben, können Sie ein Template durch „Formate exportieren“ erzeugen:



Durch diese Aktion wird ein XML-File erzeugt, das als Template bearbeitet werden kann:

Beispiel: FileParser130207_092356.xml

```
<?xml version="1.0" encoding="utf-8"?>  
  <!--FileParser export: date 130207_092356 (a:control1.5.10-T) -->  
  <aControl xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xsi:noNamespaceSchemaLocation="aControl.xsd">  
    <Printer id="126AF8F4-1F41-4FB6-83DA-39285405CF04" name="NG8IP">  
      <CMD action="LoadConfiguration" id="A6D02A1C6EA04E6C9D942F427542BF74" />  
      <CMD action="SetVariable" name="Density" value="020, 3 bar" global="false"  
sql="" />  
      <CMD action="SetVariable" name="Folientyp" value="000" global="false" sql=""  
/>  
      <CMD action="SetVariable" name="Zaehler1" value="0" global="false" sql="" />  
      <CMD action="SetCounter" name="Zaehler1" start="0" stop="999999" step="1"  
duration="1" value="000000" />  
      <CMD action="LoadFormat" name="Muster pol">  
        <Format>  
          <Content sql=""  
base64="fjEwMDA0NzgbRk11c3RlciBwb2wbVkJREYXR1bVplaXQxCzAxMTcuAzA1MTcuAzEwC0YwMDAwMDAwMDAbVkN  
aYWVobGVyMQtTMDAwMDAwMDAwMDA5OTk5OTk5MDAwMDAwMDAxMDAwMDAwMDAxMDAwMDAwMDAwMDAwMEYmZQ1Njc4OQswAD  
AwMDCLCxtWSUZvbGllbnR5cAswMDALRm9saWVudHlwCzAAMDAMDAwMjULCxtWSURlbnpdHkLMDIwLCAzIGJhcgtEZ  
W5zaXR5CzAAMDAMDAwMjULCxsqJjA0MDAxODcwMDA3NjAwRjEyMzQ1Njc4CzAwNTIwMDkxMDAwGyNUMDQwMDA4NzAw  
NjcxMDBWRERhdHVtWmVpdDELMjBMMDAwGyNUMDQwMDc2MjAwNjUwMDBGTVVTVEVSFRJVQ0sLMTRMMDAwGyNUMDQwMDU  
4NTAwNzIyMDBWQ1phZWhsZXIxXzEwTDAMBSjVDA1MDAwOTAwMDg2NTAwRkZvbGllbnR5cCA6IAsxOEwwMDAbI1QwNT  
AwNTEzMMA4NzUwMFZJRm9saWVudHlwCzE4TDAMBSjVDA1MDA4NzYwMDg2NTAwVk1EZW5zaXR5CzE4TDAMMBtTMjAwG  
1gwNzA1MP8bUTAMDAw" />  
        </Format>  
      </CMD>  
      <CMD action="SetParameter" name="Speed" value="200" global="false" sql="" />  
      <CMD action="SetParameter" name="Density" value="050" global="false" sql="" />  
    </Printer>  
    <Configuration>  
      <CMD action="Configuration" id="A6D02A1C6EA04E6C9D942F427542BF74" database=""  
dbtype="SQLite|OleDb|SQLServer" />  
    </Configuration>  
  </aControl>
```

Die XML-Datei wird gegen eine gültige a:control DTD geprüft.

Action Tags:

Im Folgenden wird eine Auflistung der gültigen Action-Tags für Printer erklärt. Die Action Tags befinden sich als Unterknoten unter dem Printer-Tag

Download der gegenwärtigen Konfiguration._

Hier wird eine XML-Datei mit der aktuellen Druckerkonfiguration erzeugt. Dies kann evtl. zur Überprüfung herangezogen werden. Das File wird im Druckerverzeichnis abgelegt: C:\Users\<Benutzername>\AppData\Local\Allen Coding\AControl\FileParser\<printername>\info\B29695F57B8C4E2290E99C4B02ECCF8B.xml

```
<CMD action="GetInfo" />
```

Setzen einer Variable

```
<CMD action="SetVariable" name="Variable2" value="123"/>
```

Setzen eines Zählers

```
<Counter name="GlobalCounter1" value="0" start="0" stop="999999999" step="1" duration="1"/>
```

Setzen einer Freien Variable

```
<CMD action="SetFreeVariable" name="free1" value="hello"
```

Setzen eines Parameters (Speed)

```
<CMD action="SetParameter" name="Speed" value="160" />
```

Setzen eines Parameters (Density)

```
<CMD action="SetParameter" name="Density" value="160" />
```

Setzen eines Parameters (Quantity)

```
<CMD action="SetParameter" name="Quantity" value="10" />
```

Laden einer Konfiguration mit der angegebenen ID.

Der Configuration-Tag wird auf derselben Ebene wie Printer Tags angegeben. In der Konfiguration sind für den Drucker Informationen wie z.B. Datenbankverbindungen, globale Variableninformationen und anderes definiert.

```
<CMD action="LoadConfiguration" id="9FCAC9CC8A154D0785DDA94DFDB47098" />
```

Laden eines Druckbildes.

Es wird das Druckbild mit dem angegebenen Namen geladen.

```
<CMD action="LoadFormat" name="AZ_NERI_LA">
```

Das Druckbild kann auch Base64-codiert direkt angegeben werden.

```
<CMD action="LoadFormat" name="AZ_NERI_LA">  
  <Format>
```

```
<Content sql="" base64="fjEwMDAyNzIbRkFaX05FUKlfTEEbVklURVhUTE1ORTELMtIzNDU2Nzg5C1RFWFRMSU5FMQswADAwMDAwMDI1CwsbVklURVhUTE1ORTILMTIzNDU2Nzg5C1RFWFRMSU5FMgswADAwMDAwMDI1CwsbVklURVhUTE1ORTMLMTIzNDU2Nzg5C1RFWFRMSU5FMwswADAwMDAwMDI1CwsbI1QwMDAwMTU2MDAwNDgwMFZJJEVYVExJTkUxCzA3TDAwMBSjVDAwMDAxNTYwMDA4NDAwVklURVhUTE1ORTILMDdMMDAwGyNUMDAwMDE1NjAwMTIxMDBWsvRFWFRMSU5FMwswN0wwMDAbUzE2MBtYMDcwNTD/G1EwMDAwMA==" />  
</Format>
```

</CMD>

Beispiele:

Eine XML-Datei muss nicht ein Format beinhalten, es können auch nur Variablen oder Counter geändert wie in folgendem Beispiel:

```
<?xml version="1.0" encoding="UTF-8"?>
<aControl xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="aControl.xsd">
    <Printer name="Printer1" id="C3E20533-6C04-47bb-9B7E-7BD2BA6AFB05">
        <CMD action="SetVariable" name="Variable2" value="123"/>
        <CMD action="SetVariable" name="Variable2" keyfield="123" table="Tabelle1"
dbtype="ArtNr"/>
        <CMD action="SetCounter" name="Counter1" value="123" start="0"
stop="999999999" step="1" duration="1" global="true"/>
    </Printer>
</aControl>
```

In einer Configuration Section des XMLs können druckerunabhängige a:control Einstellungen getätigt werden, wie z.B. Globale Variablen oder Datenbank Connectionstrings

```
<Configuration>
    <!--Setup Configuration-->
    <CMD action="Configuration" id="A6D02A1C6EA04E6C9D942F427542BF74"
name="Beispiel.cfg"
database="Server=technik4\sqlexpress;Database=toennies;Trusted_Connection=True;"
dbtype="SQLServer">
        <Parameters>
            <Parameter name="GlobalSpeed" value="200"/>
            <Parameter name="GlobalGlobalDensity" value="60"/>
            <Parameter name="GlobalQuantity1" value="00001"/>
            <Parameter name="GlobalQuantity2" value="00011"/>
        </Parameters>
    <!--Setup Global Variables-->
    <Variables>
        <Variable name="GTINGlobal" value="AB1345678901234"/>
        <Variable name="GlobalVariable1" value="Text"/>
        <Counter name="GlobalCounter1" value="0" start="0"
stop="999999999" step="1" duration="1"/>
        <Counter name="GlobalCounter2" value="0" keyfield="1234"
sql="select art_nr from Tabelle1" table="Tabelle1" fields="ArtNr" start="0" stop="999999999"
step="1" duration="1"/>
    </Variables>
    </CMD>
</Configuration>
```

Kundenbeispiel mit Füllen von Variablen aus einem Excel-File:

- Die Excel-Tabelle „Knob.xls“ ist ein Extrakt aus einer SAP-Datenbank.
- Der Name des Druckers ist „192.168.173.70“
- Es existiert ein Druckbild auf diesem Drucker („Knob“). Die darin definierten freien Variablen sollen nun über ein XML aus der Excel Tabelle Blatt „Product specs\$“ befüllt werden.
- die Datei „Knauf.xml“, die in das Fileparser Verzeichnis kopiert wird, hat folgenden Inhalt:

```
<?xml version="1.0" encoding="utf-8"?>
<!--FileParser export: date 140311_135223 (a:control1.5.12-R) -->
<aControl xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="aControl.xsd">
  <Printer id="6F4F8FD0-8CDC-433D-8410-55C2F72BFBA0" name="192.168.173.70">
    <CMD action="LoadConfiguration" id="B52827A9830949B69686B88ADA1B72CD" />

    <CMD action="SetFreeVariable" name="AREA" value="" fields="AREA" />
    <CMD action="SetFreeVariable" name="BARCODE- roll" value="" fields="BARCODE- roll" />
    <CMD action="SetFreeVariable" name="Lambda" value="" fields="Lambda" />
    <CMD action="SetFreeVariable" name="Length mm" value="" fields="Length mm" />
    <CMD action="SetFreeVariable" name="PRODUCT" value="" fields="PRODUCT" />
    <CMD action="SetFreeVariable" name="RValue" value="" fields="RValue" />
    <CMD action="SetFreeVariable" name="SAPCODE" value="" fields="SAPCODE"/>
    <CMD action="SetFreeVariable" name="Thick" value="" fields="Thick" />
    <CMD action="SetFreeVariable" name="Width full" value="" fields="Width full" />

    <CMD action="LoadFormat" name="knob"/>
    <CMD action="SetParameter" name="Speed" value="100" sql="" />
    <CMD action="SetParameter" name="Density" value="050" sql="" />
  </Printer>
  <Configuration>
    <CMD action="Configuration" id="B52827A9830949B69686B88ADA1B72CD"

    database="Provider=Microsoft.ACE.OLEDB.12.0;Data source=C:\Users\knob\Desktop\Knob.xls;Extended Properti
es='Excel 12.0 Xml;HDR=YES'"
    dbtype="OleDB"
    sql="Select * from [Product specs$] where SAPCODE='426005'" />
  </Configuration>
</aControl>
```

Beispiel mit Füllen von variablen Kundendaten aus einer XML-Datei:

-Das Format mit dem Namen "lloreale" ist ein Druckformat im Drucker
- Der Name des Druckers lautet "NGTSERIELL". - Dies sind zwei Druckervariablen, die dem XML zugeordnet sind (CHARG-> CHARG, MATNR-> MATNR).
-Im Konfigurationsabschnitt wird der Speicherort der XML-Datei definiert, der dbtype wird auf XML gesetzt
Jedes Mal, wenn die XML-Datei in das Datei-Parser-Verzeichnis kopiert wird, wird der SAP-Export auf das Format angewendet.
(Normalerweise unter C:\Users\<username>\AppData\Local\Allen Coding\<aControl>\FileParser)

```
<?xml version="1.0" encoding="utf-8"?>
<!--FileParser export: date 20190218_092836 (a:control1.9-R) -->
<aControl xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="aControl.xsd">
  <Printer id="F5048111-3EB7-D467-CC55-8F6AD1C94BB1" name="NGTSERIELL">
    <CMD action="LoadConfiguration" id="01354130A0BF4653BB03D4FC84370EEC" />
    <CMD action="SetVariable" name="CHARG" value="" global="false" sql="CHARG" />
    <CMD action="SetVariable" name="MATNR" value="" global="false" sql="MATNR" />
    <CMD action="LoadFormat" name="lloreale"/>
  </Printer>
  <Configuration>
    <CMD action="Configuration" id="01354130A0BF4653BB03D4FC84370EEC"
database="C:\Users\AllUsers\SAP_Export\sap.xml" dbtype="XML" />
  </Configuration>
</aControl>
```

Kundendatei in C:\Users\AllUsers\SAP_Export\sap.xml

```
<?xml version="1.0" encoding="utf-8"?>
<Lot xmlns="http://www.Apriso.com/Lot.xsd">
  <WERKS>280</WERKS>
  <AUFNR>300001962400</AUFNR>
  <RSHID>BD63</RSHID>
  <VERAN>UP2</VERAN>
  <MATNR>D1254003</MATNR>
  <GSTRI>20190110</GSTRI>
  <CHARG>123456</CHARG>
  <VFDAT>20220109</VFDAT>
</Lot>
```

Richten Sie Kopiertrigger so ein, dass die Fileparser-xml jedes Mal ausgeführt wird, wenn die Kundendatei geändert wurde:

