

# Allgemeines

## Inhaltsverzeichnis

- [1 Aufruf aus LogiSoft Batchpad](#)
- [2 Aufruf aus AppDesigner Makro](#)

Der LogiSoft Beleggenerator ist eine Erweiterung für die Sage 100 und kann entweder aus dem Batchpad oder in AppDesigner-Lösungen aus Makros heraus angesprochen werden.

Der LogiSoft Beleggenerator ist eine Erweiterung für die Sage 100 und kann entweder aus dem Batchpad oder in AppDesigner-Lösungen aus Makros heraus angesprochen werden.

Importdaten werden entweder aus Datenbanktabellen gelesen oder per DataTable übergeben. Aus den Daten wird entweder ein neues Belegobjekt samt Positionen in der Sage 100 erzeugt oder ein bestehender Beleg kann verändert werden. Als Quelle kann entweder eine gemeinsame Tabelle/DataTable für Kopf und Positionen oder zwei getrennte Tabellen angegeben werden. Dazu werden entweder Gruppierungsinformationen (eine Tabelle) oder Parent-Verknüpfungen (zwei Tabellen) benötigt.

Das Mapping zwischen Import- und Belegfeldern wird mittels Konfigurationsdateien im XML-Format definiert, die entweder eingelesen oder direkt als String beim Aufruf übergeben werden.

Nach erfolgreichem Import eines Beleges können Datenbankskripte ausgeführt werden um z.B. die importierten Daten als verarbeitet zu kennzeichnen und die Beleg-ID des erzeugten Beleges dort zu vermerken.

Der Beleggenerator füllt direkt und generisch das Sage Belegobjekt, sowohl im Einkauf als auch im Verkauf, d.h. prinzipiell können praktisch alle Felder im Beleg und den Positionen angesprochen werden. Die Erstellung der Objekte erfolgt dabei nach dem üblichen Muster in Sage. Abhängigkeiten werden dabei nur teilweise für die wichtigsten Felder geprüft, für wenig benutzte Felder muss selbst für die Korrektheit der Daten gesorgt werden, ebenso ggf. für Pflichtfelder in bestimmten Fällen. Es wird in jedem Fall die Validierung der Sage-Objekte angestoßen und nur bei Erfolg wird gespeichert. Lagerplatzzuordnungen werden nur eingeschränkt übernommen.

Fehler- und auch Debug-Informationen werden über den TraceLog-Manager der Sage 100 ausgegeben und können dort zur Fehlersuche herangezogen werden.

## 1 Aufruf aus LogiSoft Batchpad

Der Beleggenerator kann aus dem Batchpad heraus angestoßen werden sofern auf dem Applikations-Server das entsprechende Pugin „LSAppServerPlugin“ installiert wurde.

Dazu werden im Skript zuerst einmalig die Zugangsdaten für den Applikations-Server initialisiert, danach können beliebig viele Aufrufe zur Beleggenerierung erfolgen.

#### Initialisierung der Zugangsdaten

##### **Sage100Init**

|            |  |
|------------|--|
| Server     | Name des Sage Applikations-Servers   |
| Port       | Port des installierten Plugins   |
| Datasource | Datenbankbezeichnung der Sage 100 (aus dem Sage Administrator)   |
| Mandant    | Mandantenummer   |
| User       | Benutzer auf dem Applikations-Server. In der Sage 100 sind entsprechende Rechte zu vergeben.   |
| Kennwort   | Kennwort   |
| Timeout    | Zeitraum nach dem der Aufruf abgebrochen wird. Angabe im Format hh:mm:ss Default ist 00:02:00, d.h. falls nach 2 Minuten keine Rückmeldung des Applikations-Servers erfolgt bricht der Aufruf mit einem Fehler ab. |
| Https      | false oder true Falls auf true gesetzt wird https für den Aufruf verwendet, dazu muss ein entsprechender Endpunkt am Applikations-Server eingerichtet sein.  |

Beispiel:

Code

```
<Sage100Init Server="Server123" Port="7777" Datasource="OLDemoReweAbfD" Mandant="123" User=
```

#### Aufruf des Beleggenerators

##### **Sage100Execute**

|         |   |
|---------|---|
| Service | LogiSoft.Beleggenerator   |
| Config  | Dateiname der XML-Konfigurationsdatei (aus Sicht des Applikations-Servers!) oder der Inhalt als XML   |
| Data    | Kann eine Auflistung von DataTables für die Beleg- und/oder Positionsdaten enthalten. In diesem Fall werden die Importdaten aus den DataTables verwendet und nicht von der Datenbank gelesen. Die Bezeichnung der DataTables im Batchpad werden kommasapariert angegeben. Die erste DataTable gibt dabei die Belegdaten oder die kompletten Daten des Import an, die zweite die Postionsdaten. Bsp:<br>{@Beleg},{@Positionen} |
| Message | Gibt eine Variable an, in der die Rückgabemeldung des Beleggenerators gespeichert werden. Das ist entweder eine Angabe wie viele Belege generierte wurden oder die Fehlermeldung, falls ein Fehler aufgetreten ist.   |

|          |   |
|----------|---|
| Mode     | Wird nicht verwendet  |
| Args     | Wird nicht verwendet  |
| Variable | Gibt die Variable für den Rückgabewert der Ausführung an. Enthält true wenn die Ausführung erfolgreich war oder false wenn ein Fehler aufgetreten ist. In diesem Fall wird unter Message die Fehlermeldung übergeben. |

Beispiel:

Code

```
<Sage100Execute Service="LogiSoft.Beleggenerator" Config="{@ConfigFile}" Data="{@TableKopf}
```

Zusätzlich werden alle im Batchpad angelegten Variablen an den Beleggenerator übergeben und können dort in der Konfiguration als Werte verwendet werden.

Außerdem stehen nach der erfolgreichen Ausführung im Batchpad zwei Variablen zur Auswertung zur Verfügung:

{@GenerierteBelID} Enthält die kommaseparierten BelIDs der generierten Belege  
 {@GenerierteBelegnummer} Enthält die kommaseparierten Belegnummern der generierten Belege

## 2 Aufruf aus AppDesigner Makro

Die Beleggenerator-DLL kann aus einem AppDesigner-Makro heraus aufgerufen werden.

Verwenden Sie dazu den Makro-Befehl „AufrufenDLL“

Parameter 1 LogiSoft.Beleggenerator.dll  
 Parameter 2 LogiSoft.Beleggenerator.BeleggeneratorMacro  
 Parameter 3 "Config-Datei (optional)";[Variable1];[Variable2];...

Der Pfad zu der zu verwendenden XML-Configdatei wird an erster Stelle im Parameter 3 in Anführungszeichen übergeben, danach kommen, durch Semikolon getrennt, Makro-Variablen und Felder in der AppDesigner-Makro üblichen Form in [ ].

Alternativ kann der Datei-Pfad auch in einer Makro-Variable „\_ConfigDatei“ (mit Unterstrich) enthalten sein, in diesem Fall spielt die Reihenfolge der übergebenen Parameter keine Rolle.

Als weitere Option kann der XML-Inhalt auch direkt in der Variable „\_ConfigXML“ übergeben werden, in diesem Fall wird kein Dateipfad benötigt.

In der Variable „\_Config“ kann der Dateiname oder der Inhalt übergeben werden, wie in der vorigen Optionen, eine Erkennung ob Pfad oder XML erfolgt automatisch.

Als Rückgabe wert steht in der Makro-Variable „\_AnzahlBelege“ die Anzahl der generierten Belege zur Verfügung. Die Variable „\_GenerierteBelID“ enthält die BelID des erzeugten Beleges oder eine Liste, in „\_GenerierteBelegnummer“ steht die Belegnummer(n). Die Standard Makro-Variablen „\_Canceled“ und „\_CanceledMessage“ werden ebenfalls gesetzt und können zur Fehlerbehandlung ausgewertet werden.

Beispiele:

“C:\Beispiel\MyConfig.xml”;[BelegID]

[\_ConfigDatei];[BelegID]