

# LP3 Commander LP3 Commander Editor



**Die frei konfigurierbare  
Benutzeroberfläche für Labels Platform 3**

**Version 3.25**

## Inhaltsverzeichnis

<b><u>1</u></b>	<b><u>Einleitung.....</u></b>	<b><u>3</u></b>
<b><u>2</u></b>	<b><u>LP3 Commander Editor - Überblick .....</u></b>	<b><u>3</u></b>
<b><u>3</u></b>	<b><u>LP3 Commander - Überblick .....</u></b>	<b><u>3</u></b>
<b><u>4</u></b>	<b><u>LP3 Commander Aktionen .....</u></b>	<b><u>4</u></b>
4.1	Aktion: Druckstart.....	4
4.2	Aktion: Druckstop .....	5
4.3	Aktion: Textdatei-Import.....	6
4.4	Aktion: Serieller Datenimport.....	6
4.5	Aktion: Druckerüberwachung Automation.....	7
4.6	Aktion: Dateiüberwachung Automation .....	7
4.7	Aktion: Serielle Überwachung Automation .....	8
4.8	Aktion: VBA Script .....	8
4.9	Aktion: Aktionen Sequenz .....	9
4.10	Aktion: Datei Sequenz .....	9
4.11	Aktion: OPC Client .....	10
<b><u>5</u></b>	<b><u>LP3 Commander Variablen.....</u></b>	<b><u>10</u></b>
<b><u>6</u></b>	<b><u>LP3 Commander Felder .....</u></b>	<b><u>11</u></b>
6.1	Eingabefeld.....	11
6.2	Druckknopf.....	12
6.3	Auswahlliste .....	14
6.4	Anzeigefeld.....	16
6.5	Tabellenfeld.....	17
6.6	Allgemeine Einstellungen.....	18
<b><u>7</u></b>	<b><u>LP3 Commander Benutzerverwaltung.....</u></b>	<b><u>19</u></b>
<b><u>8</u></b>	<b><u>LP3 Commander Funktionstasten .....</u></b>	<b><u>19</u></b>
<b><u>9</u></b>	<b><u>LP3 Commander Eingabefolge .....</u></b>	<b><u>20</u></b>

## **1 Einleitung**

Produktionsabläufe bedürfen meist individuelle Benutzerführungen und auch der Datenfluss (z.B. Datenübernahme aus Fremdsystemen) ist oft unterschiedlich und muss angepasst werden.

Mittels des „LP3 Commander Editors“ kann eine Bedienoberfläche für kundenspezifische Applikationen erstellt werden. Der Funktionsumfang dieser Programme („LP3 Commander“) entspricht nicht nur dem Funktionsumfang des LP3 „Produktmanager StandardDB“, sondern enthält (neben der frei gestaltbaren Optik) auch noch zusätzliche Möglichkeiten, z.B. Zugriff auf mehrere Datenbank-Quellen/Tabellen in derselben Oberfläche, Aufruf von VB-Script-Programmen, flexibler Datenimport, etc.

Das erstellte Commander-Projekt kann mit jeder LP3 PREMIUM Lizenz ausgeführt werden. Zudem sind auch Optionen erwerbbar, welche ein Commander-Projekt auch mit ECO und STAR-Lizenzen lauffähig machen.

## **2 LP3 Commander Editor - Überblick**

Der Aufbau ist ähnlich wie der LP3 Etiketteneditor. Ein frei vergrößerbare Fenster mit Menü und Symbolleiste. Der Hauptbereich zeigt das aktuell in Bearbeitung befindliche Commander-Fenster. Rechts daneben ist die Feld-/Variablenliste.

Felder werden identisch wie im LP3 Etiketteneditor angelegt. Modus in Menü oder Symbolleiste wählen, mit Maus in Hauptbereich klicken, und das neue Feld wird dort platziert. Im Auswahlmodus kann man die Felder verschieben und in der Größe ändern. Per Rechtsklick bekommt man ein Popupmenu, mit Kopieren/Einfügen/etc und dem Punkt „Eigenschaften“.

Die Eigenschaften werden in der von LP3 gewohnten Eigenschaften-Liste angezeigt, dort kann der Benutzer die Position und Optik (z.B. Schriftart und Farben) beeinflussen. Auch die weiteren spezielleren Eigenschaften (Datenbezüge) eines Feldes kann er hier einstellen.

Die fertig gestaltete Oberfläche wird in einer Datei abgespeichert („LPC“ – LP Commander). Dem Commander-Programm (LPCommander.exe) kann man diesen Dateinamen als Kommandozeilenparameter übergeben, oder ihn in der Commander-INI-Datei einstellen, oder der Commander sucht automatisch im Startverzeichnis nach der ersten LPC-Datei. Die gefundene Datei wird vom LPCommander geladen, das Programm führt die darin definierte Funktionalität und Optik aus.

## **3 LP3 Commander - Überblick**

Die Funktionalität eines Commander-Projektes setzt sich aus folgenden Hauptbestandteilen zusammen:

### **1. Aktionen:**

a) Hintergrund Aktionen : Prozesse, die ständig oder per Knopfdruck ablaufen können, z.B. Datenimport oder Automations-Überwachungen. Jeder Prozess bekommt vom Benutzer einen Namen, es können mehrere Prozesse parallel ablaufen.

b) Druckstart Aktionen : man kann Listen mit Aktionen festlegen, die bei einem „Druckstart“ ablaufen sollen. Jede dieser Aktionslisten bekommt vom Benutzer einen eigenen Namen, es können mehrere

Druckstart-Aktionen definiert werden (für z.B. Druckstart auf verschiedenen Linien).

c ) Druckstop Aktionen : man kann Listen mit Aktionen festlegen, die bei einem „Druckstop“ ablaufen sollen. Jede dieser Aktionslisten bekommt vom Benutzer einen eigenen Namen.

2. Variablen. Diese werden zum Datenaustausch zwischen den Commander-Feldern, dem LP3 Etikett, usw. benutzt
3. Felder. Anzeige/Bedienfelder wie Knöpfe, Eingabefelder, usw.
4. Benutzerverwaltung

## 4 LP3 Commander Aktionen

Aktionen sind nicht an Felder gebundene Prozesse, die der Benutzer definieren und unter einem Namen ablegen kann. Je nach Aktionstyp kann die Aktion dann ständig im Hintergrund ausgeführt werden, oder auch per Knopfdruck gestartet (und ggf. auch gestoppt) werden.

Mögliche Aktionen sind:

- 1) Druckstart-Aktion
- 2) Druckstop-Aktion
- 3) Textdatei Import
- 4) Serieller Datenimport
- 5) Drucker Überwachung - Automation
- 6) Dateiüberwachung - Automation
- 7) Serielle Überwachung - Automation
- 8) VBA-Script Ausführung
- 9) Aktionen Sequenz
- 10) Datei-Sequenz
- 11) OPC Client

Hierzu gibt es im Commander Editor ein „Aktionen Manager“ Fenster, in dem eine Liste mit allen bisher definierten Aktionen angezeigt wird, sowie den Knöpfen „Neu...“, „Ändern...“, „Löschen...“.

### 4.1 Aktion: Druckstart

Hier hat der Benutzer ein Fenster, in dem er folgendes einstellen kann:

- Eine Liste mit drei Infos pro Spalte:
  - a) fester Druckername, oder Variable, die Druckernamen enthält
  - b) feste Druckanzahl, oder Variable, die Druckanzahl enthält
  - c) fester Layoutname, oder Variable, die Layoutname enthält

In dieser Liste kann der Benutzer also festlegen, welche(r) Drucker beim Druckstart mit welchem Layout und welcher Anzahl aktiviert werden soll. Die Reihenfolge der Druckjobs gibt auch die Reihenfolge an, in welcher die Jobs an den Spooler übergeben werden. Damit sind also auch mehrere Jobs für denselben Drucker hintereinander aktivierbar.

- Optional: serielle Ausgabe.

Hier kann der Benutzer festlegen, ob bei Druckstart auf eine serielle Schnittstelle ein Datenstring geschickt werden soll. Einstellbar sind: Schnittstelle, Startzeichen, Stopzeichen, Daten (fester Text oder eine Variable). Per LPCOMMANDER.INI sind die Schnittstellen-Einstellungen auch außerhalb

des Commander-Editors einstellbar („Aktionsname\_Port=...“). Siehe Kapitel „INI-Datei“.

- Optional: Datei Ausgabe.

Hier kann der Benutzer festlegen, ob bei Druckstart ein Text in eine Datei geschrieben werden soll. Einstellbar sind: Dateiname, „Daten anhängen“, Daten (fester Text oder eine Variable). XML

- Optional: Start eines VBA-Scripts

Hier kann der Benutzer das Script festlegen (Dateiname des Scripts oder direkt den Script-Source) Falls Dateiname, so kann per LPCOMMANDER.INI Eintrag „Aktionsname\_ScriptFile=...“ auch den Dateinamen überschreiben. Siehe Kapitel „INI-Datei“.

- Optional: Liste mit Commander Vars->LP3 Layout (externe) Vars, die vor Druckstart transferiert werden sollen. Hintergrund: LP3 Layout wird externe Vars haben, die die Druckdaten empfangen sollen. Wenn man nicht gleich in Commander-Feldern diese externe Vars befüllt, kann man dies auch bei Druckstart von der Druckstart-Aktion ausführen lassen.

Sind mehrere Optionen aktiv, so wird die Druckstart-Aktion in folgender Reihenfolge ausgeführt:

1. Variablentransfer
2. VBScript
3. Druckauftrag erzeugen und Start der Übertragung
4. Schreiben Daten in Datei
5. Serielle Daten Ausgabe

Diese Reihenfolge deckt die meisten Standard-Anforderungen ab. Benötigt man eine andere Reihenfolge, so kann man sich mit Aktionen Sequenzen (siehe unten) behelfen.

## 4.2 Aktion: Druckstop

Hier hat der Benutzer ein Fenster, in dem er folgendes einstellen kann:

- Eine Liste mit einer Spalte:

a) fester Druckername, oder Variable, die Druckernamen enthält

In dieser Liste kann der Benutzer also festlegen, welche(r) Drucker beim Druckstop noch zurückgesetzt werden soll

- Optional: serielle Ausgabe.

Hier kann der Benutzer festlegen, ob bei Druckstop auf eine serielle Schnittstelle ein Datenstring geschickt werden soll. Einstellbar sind: Schnittstelle, Startzeichen, Stopzeichen, Daten (fester Text oder eine Variable). Per LPCOMMANDER.INI sind die Schnittstellen-Einstellungen auch außerhalb des Commander-Editors einstellbar („Aktionsname\_Port=...“). Siehe Kapitel „INI-Datei“.

- Optional: Datei Ausgabe.

Hier kann der Benutzer festlegen, ob bei Druckstop ein Text in eine Datei geschrieben werden soll. Einstellbar sind: Dateiname, „Daten anhängen“, Daten (fester Text oder eine Variable).

- Optional: Start eines VBA-Scripts

Hier kann der Benutzer das Script festlegen (Dateiname des Scripts oder direkt den Script-Source) Falls Dateiname, so kann per LPCOMMANDER.INI Eintrag „Aktionsname\_ScriptFile=...“ auch den

Dateinamen überschreiben. Siehe Kapitel „INI-Datei“.

Sind mehrere Optionen aktiv, so wird die Druckstop-Aktion in folgender Reihenfolge ausgeführt:

1. Drucker Stop
2. VBScript
3. Schreiben Daten in Datei
4. Serielle Daten Ausgabe

Diese Reihenfolge deckt die meisten Standard-Anforderungen ab. Benötigt man eine andere Reihenfolge, so kann man sich mit Aktionen Sequenzen (siehe unten) behelfen.

### 4.3 Aktion: Textdatei-Import

Hier hat der Benutzer ein Fenster, in dem er folgendes einstellen kann:

- Ziel-Datenbank Tabelle (d.h. Datasource-Name, ggf Username/PW, Tabellenname)
- Pfad + Name der zu importierenden Textdatei (Wildcards möglich), optional aus Variable
- Zuordnung Spalte <> Dateielement / XML
- Datei-Art (mit Trennzeichen oder ohne oder XML)
- Falls mit Trennzeichen: Wahl des Zeichens
- Option: Datei nach Import löschen (ja/nein)
- Festlegbare Liste der „Remove Chars“
- Festlegbare Liste der „Replace Chars“
- Einstellung: „Import nach n Sekunden“ (0 Sekunden=Prozess muss manuell per Knopf gestartet werden)
- Import-Modus: a) neue Daten immer anhängen, b) neue Daten überschreiben ggf. alte Daten (in diesem Modus muss das Key-Feld angegeben werden), c) neue Daten ersetzen komplett alte Daten (hierbei wird beim Import der alte Tabelleninhalt komplett gelöscht).

Diese Aktion kann manuell (per Knopfdruck) gestartet werden, oder auch zyklisch im Hintergrund laufen (Sekundenzahl > 0). Falls sie zyklisch läuft, kann der Benutzer sie auch per „Aktion Start/Stop“-Knopf anhalten und wieder starten.

Falls der Import-Dateinamen fest ist (also nicht aus einer Variablen kommt), kann man ihn per LPCOMMANDER.INI Eintrag „Aktionsname\_File=...“ auch überschreiben. Siehe Kapitel „INI-Datei“.

### 4.4 Aktion: Serieller Datenimport

Hier hat der Benutzer ein Fenster, in dem er folgendes einstellen kann:

- Ziel-Datenbank Tabelle (d.h. Datasource-Name, ggf Username/PW, Tabellenname)
- Serielle Schnittstelle + Schnittstellenparameter (Comx, Baud, Bits, Protokoll), Netzwerk
- Start-Zeichen
- Stop-Zeichen
- Trenn-Zeichen
- Antwort-Text: a) keiner, b) fester Text (einstellbar), c) Antwort-Text=Empfangstext
- Option: alle Datensätze der Ziel-Tabelle bei Empfang einer angebbaren Sequenz löschen
- Übernahme-Modus: a) neue Daten immer anhängen, b) neue Daten überschreiben ggf. alte Daten (in diesem Modus muss das Key-Feld angegeben werden), c) neue Daten ersetzen komplett alte Daten (hierbei wird bei der Übernahme der alte Tabelleninhalt komplett gelöscht).

Die Einstellungen der seriellen Schnittstelle können per LPCOMMANDER.INI Eintrag „Aktionsname\_Port=...“ auch überschrieben werden. Siehe Kapitel „INI-Datei“.

## 4.5 Aktion: Druckerüberwachung Automation

Hier hat der Benutzer ein Fenster, in dem er folgendes einstellen kann:

- Tabelle-Variable (gibt Datenbank und Tabelle), die zur Automation genutzt werden soll. Zudem die Unter-Tabellenvariable, die den Schlüsselwert enthält (wird zum Löschen benötigt). Ist der Schlüssel in der Tabelle ein „Zahl“-Typ, so muss man das auch angeben.
- Zur Automation muss der Benutzer entweder fest einen LP3 Druckernamen angeben, oder festlegen „Tabellenspalte X enthält Druckernamen“ (Tabellenspalte wird über weitere Tabellen-Untervariable eingestellt).  
Im ersten Fall (fester Druckername) überwacht der Automationsprozess den gegebenen Drucker, hat dieser keinen Druckjob, so wird der erste Datensatz der Tabelle übertragen, und aus der Tabelle gelöscht. Im zweiten Fall (variabler Druckername aus Tabelle) filtert der Automationsmodus die Datensätze nach den Druckern, und führt immer den nächsten ersten Datensatz auf dem jeweiligen Drucker aus, wenn dieser keinen Druckjob hat. Auch hier wird der entsprechende Datensatz aus der Tabelle gelöscht.
- Pro Druckeranschluss muss man hierzu noch bis zu zwei Angaben machen: a) die „Druckstart-Aktion“ (aus der Liste der definierten Druckstart-Aktionen) und (optional) b) die „Druckstop-Aktion“ (aus der Liste der definierten Druckstop-Aktionen). Die Druckstop-Aktion wird ausgeführt, wenn der Batchcount des zu überwachenden Druckers auf „0“ geht, die Druckstart-Aktion, wenn der neue Auftrag auf dem Drucker aktiviert wird. Vor Ausführung der Druckstart Aktion wird natürlich auch die angegebene Tabellenvariable aufgefrischt (alle Untervariablen erhalten die Werte des automatisch gewählten Datensatzes).

Besonderheit: auch eine „Druckerüberwachung ohne Drucker“ ist möglich. D.h.: wenn man nur möchte, dass sobald ein Datensatz in einer Tabelle vorhanden ist, mit diesem Datensatz eine Druckstart-Aktion durchgeführt werden soll (und diese Druckstart-Aktion überträgt z.B. nur die Daten des Datensatzes über die serielle Schnittstelle), und danach der eine Datensatz gelöscht werden soll, so kann man als Druckername einfach „-“ benutzen. Damit weiß der Commander, dass er einfach die Datensätze nacheinander bearbeiten soll, ohne auf eine Druckerrückmeldung zu warten.

## 4.6 Aktion: Dateiüberwachung Automation

Hier hat der Benutzer ein Fenster, in dem er folgendes einstellen kann:

- Tabelle-Variable (gibt Datenbank und Tabelle), die zur Automation genutzt werden soll. Zudem die Unter-Tabellenvariable, die den Suchwert enthält. Ist das Suchfeld in der Tabelle ein „Zahl“-Typ, so muss man das auch angeben.
- Zur Automation muss der Benutzer noch folgende Angaben machen: Fester Dateinamen; Druckstart-Aktion; Fehler-Aktion. Der Ablauf gestaltet sich dann folgendermassen: der Automationsprozess testet zyklisch (Sekundentakt einstellbar), ob eine neue Datei existiert. Wenn ja, so wird die Datei ausgelesen, und gelöscht. Dann wird der gelesene Dateiinhalt in der gegebenen Tabellen-Spalte gesucht. Falls Suche erfolgreich, so wird die die Tabellenvariable aufgefrischt und die Druckstart-Aktion ausgeführt.  
Der Benutzer kann zudem wählen, ob der Datensatz gelöscht werden soll oder nicht (Unterschied zwischen Auftragsnummer-Abarbeitung und Artikelnummer-Aufruf).  
Werden die gelesenen Daten nicht gefunden, so wird die Fehler-Aktion durchgeführt (falls vorhanden).

Der feste Dateiname kann per LPCOMMANDER.INI Eintrag „Aktionsname\_File=...“ auch überschrieben werden. Siehe Kapitel „INI-Datei“.

## 4.7 Aktion: Serielle Überwachung Automation

Hier hat der Benutzer ein Fenster, in dem er folgendes einstellen kann:

- Tabelle-Variable (gibt Datenbank und Tabelle), die zur Automation genutzt werden soll. Zudem die Unter-Tabellenvariable, die den Suchwert enthält. Ist das Suchfeld in der Tabelle ein „Zahl“-Typ, so muss man das auch angeben.
- Zur Automation muss der Benutzer folgende Angaben machen: Schnittstelle mit Parametern (Baud, etc); Start- und Stopzeichen; Druckstart- bzw Fehler-Aktion. Der Ablauf gestaltet sich dann folgendermassen: der Automationsprozess überwacht die angegebene Schnittstelle. Wird da eine Startzeichen-Text-Stopzeichen-Sequenz empfangen, so wird der empfangene Text in der gegebenen Tabellen-Spalte gesucht. Falls Suche erfolgreich, so wird die Tabellenvariable aufgefrischt und die angegebene Druckstart-Aktion ausgeführt. Der Benutzer kann zudem wählen, ob der Datensatz gelöscht werden soll oder nicht (Unterschied zwischen Auftragsnummer-Abarbeitung und Artikelnummer-Aufruf). Werden die empfangenen Daten nicht gefunden, so wird die Fehler-Aktion durchgeführt (falls vorhanden).

Die Einstellungen der seriellen Schnittstelle können per LPCOMMANDER.INI Eintrag „Aktionsname\_Port=...“ auch überschrieben werden. Siehe Kapitel „INI-Datei“.

## 4.8 Aktion: VBA Script

Hier hat der Benutzer ein Fenster, in dem er folgendes einstellen kann:

- VBA-Script (Filename oder direkt den Script-Befehlscode)
- Einstellung, ob dieses VBA-Script alle n Sekunden gestartet werden soll
- Einstellung, ob dieses VBA-Script einmal bei Programmstart ausgeführt werden soll

Falls Dateiname, so kann per LPCOMMANDER.INI Eintrag „Aktionsname\_ScriptFile=...“ auch den Dateinamen überschreiben. Siehe Kapitel „INI-Datei“.

Diese Aktion kann auch über Druckknöpfe oder Aktionen-Sequenzen ausgeführt werden.

Man kann alle Befehle von VBScript nutzen, je nach installierter Windows Scripting Engine. Unter Windows XP ist derzeit die Version 5.6 aktuell (sollte auch auf allen XP Rechnern installiert sein, da sie Bestandteil des Betriebssystems ist). Innerhalb der LP Commanders kann man zudem zwei zusätzliche Befehle direkt einsetzen: „SetLPVar“ und „GetLPVar“. Damit kann man im Skript auf Variableninhalte des Commanders zugreifen (x= GetLPVar(“Variablenname“)) bzw. Werte in interne und externe Variablen setzen (Call SetLPVar(“Variablenname“,“Wert“)).

Einfaches VBScript-Beispiel: holt Wert aus interner Variable „IT\_Zähler“, erhöht den Wert um 1, und setzt ihn wieder in die Variable „IT\_Zähler“ zurück:

```
Dim strV
```

```
strV=GetLPVar("IT_Zähler")
```

```
strV=strV+1
```

```
Call SetLPVar("IT_Zähler",strV)
```

Ein weiterer Script-Befehl ist SetLPFocus("Feldname"). Damit kann man den Eingabefokus gezielt auf ein Eingabefeld im Commander setzen.

Beispiel:

```
Call SetLPFocus("Eingabefeld1")
```

Um die LP3 Datenbank-Manager Variablen zu setzen, stehen die beiden Befehle SetDBMTable(Datenquellename, Tabellename) und SetDBMVar(Feldname,Wert) zur Verfügung. Zur Nutzung muss einmalig im Script der SetDBMTable Befehl aufgerufen werden, und dann n-mal dahinter der SetDBMVar-Befehl mit den einzelnen Feldern und Werten.

Beispiel:

```
Call SetDBMTable("MeineDatenbank","Artikel")
```

```
Call SetDBMVar("Artikelnummer",strArtikel)
```

```
Call SetDBMVar("Artikelname",strName)
```

```
Call SetDBMVar("Artikeltext",strText)
```

...

## 4.9 Aktion: Aktionen Sequenz

Aktionen Sequenzen sind Listen mit anderen Aktionen und ggf. Pausen. Diese Sequenz kann dann anstelle einer „normalen“ einmaligen Aktion (Druckstart/Druckstop/VBSkript/Dateiimport) z.B. per Druckknopf abgefahren werden. Eine solche Liste kann notwendig sein, wenn man mit einem Knopfdruck mehrere Aktionen hintereinander ausführen muss, oder wenn z.B. die Standard-Ausführungsreihenfolge einer Druckstart/Druckstop-Aktion nicht passt. Zudem kann man jede zu startende Aktion noch an eine Bedingung (Variablenvergleich) knüpfen, so dass man z.B. „DruckStartOK“ ausführt, wenn Variable „DatenbankSelektion“ größer 0 ist, und „DruckStartFehler“ ausführt, wenn die Variable 0 ist.

Beispiel 1: bei Druckstop soll der aktuelle Druckjob gelöscht werden, und unmittelbar danach noch ein „Abschluss-Etikett“ gedruckt werden. Man kann also nun eine Sequenz zusammenstellen, die zunächst den Druckstart durchführt, und danach noch einen Druckstart für das Abschlussetikett. Diese Sequenz wird dann anstelle des einfachen Druckstops auf einen Knopf gelegt.

Beispiel 2: Bei einer Druckstart-Aktion werden die einzelnen Bestandteile wie folgt hintereinander ausgeführt: Zuerst werden die Etiketten-Variablen gesetzt, dann das VBSkript ausgeführt, dann der Druckauftrag erzeugt und übertragen, und zuletzt Daten in eine Datei/Schnittstelle geschrieben. Nun benötigt man aber nach der Erzeugung des Druckauftrages noch einen VBSkript-Aufruf (z.B. um im Etikettenlayout geänderte Externe Variablen auszuwerten). Damit ist der im Druckstart integrierte VBSkript nicht nutzbar (da er vor dem Druckauftrag ausgeführt wird). Mit Hilfe einer Sequenz kann man aber noch eine VBSkript-Aktion nach der Druckstart-Aktion ausführen lassen.

## 4.10 Aktion: Datei Sequenz

Datei Sequenzen sind Listen mit Dateivariablen und ggf. Pausen. Diese Sequenz kann dann anstelle einer „normalen“ einmaligen Aktion (Druckstart/Druckstop/VBSkript/Dateiimport) z.B. per Druckknopf abgefahren werden, und damit beispielsweise Daten über eine „schreibende“ Dateivariable in eine Datei abgelegt werden. Es sind auch „lesende“ Dateivariablen nutzbar (wenn man zu einem bestimmten Zeitpunkt eine Dateilese-Aktion durchführen möchte), der Hauptaugenmerk liegt bei der Datei-Sequenz aber bei Schreibvariablen: diese werden nämlich im normalen Commander-Ablauf niemals ausgeführt, sondern nur, wenn eine entsprechende Dateisequenz sie aktiviert.

Zudem kann man jede Datei-Variable noch an eine Bedingung (Variablenvergleich) knüpfen, so dass man z.B. die Variable „SchreibeLogGut“ ausführt, wenn Variable „ScanGut“ 1 ist, und „SchreibeLogSchlecht“ ausführt, wenn die Variable 0 ist.

#### **4.11 Aktion: OPC Client**

Der LP Commander kann auch als OPC Client fungieren, d.h. Daten mit beliebigen OPC Servern austauschen. OPC Server sind zumeist eigenständige Windows-Applikationen, die als Bindeglied zu beliebiger Hardware (SPSen, IO-Geräte, Bussysteme) und OPC fähigen Applikationen dienen. Im LP Commander kann man in einer „OPC Client“ Aktion definieren, welche OPCServer genutzt werden sollen (lokale OPCServer oder auch Remote per Netzwerk), und welche OPC Prozesspunktvariablen gelesen/geschrieben werden sollen.

Hierzu wählt man im Konfigurationsfenster einen OPCServer aus, und betätigt „OPC Prozesspunkte auflisten“. Nun erscheinen in den Listen die verfügbaren Prozesspunkte. Die gewünschten Punkte sind mit der Maus auszuwählen, und per „Übernehmen“ Knopf in das Commander Projekt zu integrieren, sie werden dann in der unteren Liste angezeigt. Zu jedem OPC Prozesspunkt kann man nun a) eine LP3 Schreibvariable, b) eine Empfangsvariable und c) eine Empfangsaktion wählen.

Zur Laufzeit des Commanders werden diese Zuordnung automatisch ausgeführt:

Ändert sich im Commander der Wert der Schreibvariablen, so wird der neue Wert auf die zugeordneten OPC Prozesspunkte geschrieben. Ändert sich ein OPC Prozesspunkt, so wird dessen Wert auf die Empfangsvariable gegeben, und zudem die Empfangsaktion ausgeführt.

## **5 LP3 Commander Variablen**

Die Variablen, die im Commander zur Verfügung stehen, dienen zum Datenaustausch zwischen Feldern und Aktionen. Die Felder haben oftmals eine „Eingangsvariable“ und „Ausgangsvariable“, und können per Variable Werte erhalten bzw. schreiben.

Folgende Variablen sind verfügbar:

- Konstanten
- Formeln
- Datenbank
- Textdatei
- Zeit/Datum
- Extern
- Intern
- Tabelle

Die meisten Commander-Vars sind dabei identisch zu den Variablen im LP3 Etiketteneditor. Ein leichter Unterschied besteht bei der Tabellenvariable: Sie liefert die Information „Aktuell gewählter Datensatz“ in ihrer Hauptvariablen, ihre Untervariablen enthalten immer die Daten der aktuellen Tabellenselektion.

Neu ist zudem die „Interne“ Variable. Diese entspricht der „Externen“, übergibt ihren Wert aber nicht an das LP3 Layout. „Interne“ Variablen sind also dann zu nutzen, wenn sie rein zum Datenaustausch zwischen Commander Feldern/Aktionen verwendet werden.

## 6 LP3 Commander Felder

Felder werden auf der Fensterfläche platziert, und bilden das Benutzerinterface (Wertanzeigen, Werteingaben). Felder können statische Infos anzeigen (z.B. eine Kundenlogo-Grafik), aber auch dynamische Werte (aus Variablen). Die meisten Felder reagieren auf Maus- bzw. Tastatur-Eingaben (Ausnahme: das reine Anzeigefeld).

Felder haben in ihren Eigenschaften eine änderbare „Funktionseinstellung“, die das Grundverhalten des Feldes bestimmt.

Feldtypen:

- Eingabefeld
- Druckknopf
- Auswahl-Klappliste
- Grafik-Anzeigefeld
- Tabellenanzeige

### 6.1 Eingabefeld

Ein Eingabefeld wird für Tastatur-Eingaben benutzt, z.B. für zusätzliche Druck-Dateneingaben, Eingabe Anzahl Etiketten, oder als Eingabe eines Suchbegriffes in einer Datenbank-Tabellen. Grundsätzlich hat ein Eingabefeld deshalb eine „Ausgangsvariable“ (die den eingegebenen Text bekommt), und auch eine „Eingangsvariable“ (damit man das Eingabefeld mit Werten vorbesetzen kann).

Das Commander-Eingabefeld umfasst die Möglichkeiten der LP3 Text-Eingabevariablen (also eine Eingabe-Maske, oder Min-/Max-Wert).

- Name: Frei definierbarer Name des Feldes
- Statuszeilentext: Frei definierbare Info für Statuszeile
- X, Y: Position des Eingabefelds (in Pixel)
- Breite, Höhe: Größe
- Sperrbedingung: Identisch zu der LP3 Layout-„Druckbedingung“, schaltet Feld je nach Variablen aktiv
- Textfarbe: Knopf „...“: Änderung der Textfarbe
- Textfarbe (gesperrt): Knopf „...“: Änderung der Textfarbe für gesperrtes Feld
- Hintergrundfarbe: Knopf „...“: Änderung der Hintergrundfarbe
- Schriftart: Knopf „...“: Änderung der Schriftgröße/Schriftart
- Rahmen: Auswahl „Ja“ / „Nein“, bei „Ja“ wird Rahmen um Feld gemalt
- Eingabeart: Texteingabe oder Ganzzahl Eingabe oder Fließpunkt Eingabe
- Eingabefokus: links/rechts/alles selektieren (wie bei Eingabevariable)
- Eingabeabschluss: Verhalten bei „ENTER“-Taste: Schließen/Sprung nächstes Feld/Offen halten
- Eingabemaske: Klappliste: ja oder nein
- Maskentext: (falls „Eingabemaske nutzen auf „ja“): wie üblich (##@??kg\*)
- Eingabebegrenzung: Klappliste: ja oder nein
- Minimum Zeichen/Zahl: (falls Eingabebegrenzung auf „ja“)
- Maximum Zeichen/Zahl: (falls Eingabebegrenzung auf „ja“)
- Mindest-Benutzerstufe: Nur Benutzer mit eingestellter Stufe (oder höher) können Eingaben tätigen
- Funktion: Klappliste mit folgenden Funktionen:
  - \* a) Eingabefeld oder
  - \* b) Sucheingabe Tabellen-Feld

In letzterem Falle wird eine interne „Verbindung“ zwischen dem Eingabefeld und dem wählbaren Tabellen-Feld eingegangen. Eine Änderung hier spiegelt sich dann im Tabellenfeld wieder (Auswahl ändert sich) und umgekehrt.

  - \* c) Linker Etikettenrand
  - \* d) Oberer Etikettenrand
- Eingangsvariable: (bei Funktion a+b)Knopf „...“: öffnet Variablenauswahlfenster

- Ausgangsvariable: (bei Funktion a)Knopf „...“: öffnet Variablenauswahlfenster
- Tabellen-Feld: (bei Funktion b)Eingabe des Names des zugehörigen Tabellenfelds
- Tabellen-Spaltennr: (bei Funktion b)Nummer der Tabellen-Spalte, die mit dem Eingabefeld übereinstimmt
- Etikettenvariable: (bei Funktion c+d)Wahl der Variablen, in der der Layoutname steht
- Druckervariable: (bei Funktion c+d) Wahl der Variablen, in der der Druckername steht

Wenn Eingabemaske/Begrenzung aktiv ist, und der Benutzer gerade eine Eingabe tätigt, so wird eine Masken/Begrenzungsinfo im Statusleisten-Anzeigefeld angezeigt.

Wenn als Funktion „Sucheingabe Tabellen-Feld“ eingestellt ist, und eine Eingangsvariable benutzt wird, so kann die zugehörige Tabelle nicht mehr per Maus/Tastatur umselektiert werden, der Datensatz bleibt dann auf dem exakten Wert der Eingangsvariablen.

Die Funktionen „Linker Etikettenrand“/„Oberer Etikettenrand“ laden intern sofort das Etikett, und zeigen die Randwerte je nach Drucker an. Geänderte Randwerte werden in das interne Etikett zurück geschrieben (nur im RAM, nicht in der Etikettendatei gespeichert!), und beim darauf folgenden Druck benutzt. Dies bedeutet auch, dass diese Randfunktionen nur sinnvoll nutzbar sind, wenn man nur mit einer Etikettenauswahl im Commander arbeitet, bei mehreren Etikettenauswahlen würden ständig neue Etiketten intern geladen, und dabei die vorher eingestellten Werte überschrieben.

## 6.2 Druckknopf

- Name: Frei definierbarer Name des Feldes
- Statuszeilentext: Frei definierbare Info für Statuszeile
- X, Y: je ein Eingabefeld: Position des Druckknopfes (in Pixel)
- Breite, Höhe: je ein Eingabefeld: Größe des Druckknopfes
- Sperrbedingung: Identisch zu der LP3 Layout-„Druckbedingung“, schaltet Feld je nach Variablen aktiv
- Anzeigemodus: Klappliste: Grafik oder Text
- Anzeigetext „gedrückt“: Eingabefeld: Beschriftungstext 1 (nur verfügbar im Anzeigemodus „Text“)
- Anzeigetext „nicht g.“: Eingabefeld: Beschriftungstext 2 (nur verfügbar im Anzeigemodus „Text“)
- Anzeigetext „gesperrt“: Eingabefeld: Beschriftungstext 3 (im Anzeigemodus „Text“, wenn Knopf gesperrt)
- Textfarbe: Knopf „...“: Änderung der Textfarbe (nur verfügbar im Anzeigemodus „Text“)
- Textfarbe (gedrückt): Knopf „...“: Änderung der Textfarbe für gedrücktes Feld
- Textfarbe (gesperrt): Knopf „...“: Änderung der Textfarbe für gesperrtes Feld
- Hintergrundfarbe: Knopf „...“: Änderung der Hintergrundfarbe (nur im Anzeigemodus „Text“)
- Hintergrund (gedrückt): Knopf „...“: Änderung der Hintergrundfarbe für gedrücktes Feld
- Schriftart: Knopf „...“: Änderung der Schriftgröße/Schriftart (nur im Anzeigemodus „Text“)
- Grafik „nicht gedrückt“: Knopf „...“: Auswahl einer Grafikdatei für Zustand „nicht gedrückt“ (Grafik-Modus)
- Grafik „gedrückt“: Knopf „...“: Auswahl einer Grafikdatei für Zustand „gedrückt“ (Grafik-Modus)
- Grafik „gesperrt“: Knopf „...“: Auswahl einer Grafikdatei für Zustand „gesperrt“ (Grafik-Modus)
- Eingangsvariable: Knopf „...“: öffnet Variablenauswahlfenster: setzt Zustand per Var (Zahländerung = simulierter Knopfdruck)
- Ausgangsvariable: Knopf „...“: öffnet Variablenauswahlfenster: liefert Zustand in Var (0=oben,1=unten)
- Mindest-Benutzerstufe: Nur Benutzer mit eingestellter Stufe (oder höher) können Knopf betätigen
- Funktion: Klappliste mit folgenden Funktionen:
  - \* a) Start/Stop Druck
  - \* b) Start/Stop Aktion
  - \* c) Start Aktion einmalig
  - \* d) Druckereinstellungen
  - \* e) Randeinstellungen
  - \* f) Etikettenvorschau
  - \* g) Programm beenden
  - \* h) Benutzerwechsel
  - \* i) Tabellen-Variable neu auffrischen
  - \* j) Anwendung starten
  - \* k) Seite wechseln

#### Parameter für Funktion a) Start-Stop Druck

- Druckstart-Aktion: Auswahl einer definierten Druckstart-Aktion
- Druckstop-Aktion: Auswahl einer definierten Druckstop-Aktion

Bei dieser Funktion wird der Knopf zunächst Anzeigtext/grafik „nicht gedrückt“ anzeigen. Betätigt der Benutzer den Knopf, so wird die gewählte Druckstart-Aktion aktiviert, und der Anzeigetext/grafik „gedrückt“ angezeigt. Ein nochmaliges Betätigen startet die gewählte Druckstop-Aktion, und wechselt wieder auf „nicht gedrückt“. Hinweis: die Aktionen müssen nicht zwingend tatsächliche Druckstart- / Druckstop-Aktionen sein. Genauso könnten z.B. zwei VBSkript-Aktionen zugeordnet werden. Keinen Sinn machen jedoch Automations-Aktionen, da diese ja keine einmalige Aktion ausführen, sondern ständig laufen müssen.

#### Parameter für Funktion b) Start-Stop Aktion

- Aktion: Auswahl einer definierten Aktion (Datei-Import, Automation)

Ist die gewählte Aktion zur Zeit im Hintergrund aktiv, so zeigt den Anzeigetext/grafik „gedrückt“. Ansonsten zeigt er Zustand „nicht gedrückt“. Bei Betätigung wird die Aktion entweder gestartet oder angehalten (Umschaltung des derzeitigen Zustands). Hinweis: diese Knopf-Funktion macht keinen Sinn mit „Druckstart“ und „Druckstop“ Aktionen, da diese ja immer nur einmalige – und keine ständigen – Aktionen durchführen. Alle anderen Import- und Automations-Aktionen können jedoch hiermit gestartet/angehalten werden, genauso auch VBSkript-Aktionen, welche zyklisch (im angegebenen Sekundenraster) ablaufen.

#### Parameter für Funktion c) Start Aktion einmalig

- Aktion: Auswahl einer definierten Aktion (Datei-Import, Automation, Druckstart)

Hier kann eine beliebige der definierten Aktionen einmalig ausgeführt werden. Typische Fälle: Druckstarts mit einer bestimmten Etikettenanzahl aktivieren, oder Datei-Import auf Knopfdruck. Hinweis: Aktionen, die ständig laufen müssen (z.B. die „Serielle Datenimport“-Aktion, oder die Automatismen), können nicht einmalig gestartet werden.

#### Parameter für Funktion d) Druckereinstellungen

- Layoutvariable: Auswahl einer der Variablen, die den Layoutnamen enthält
- Druckervariable: Auswahl einer der Variablen, die den Druckernamen enthält

Das Druckereinstell-Fenster für das gegebene Layout/Drucker erscheint.

#### Parameter für Funktion e) Randeinstellungen

- Layoutvariable: Auswahl einer der Variablen, die den Layoutnamen enthält
- Druckervariable: Auswahl einer der Variablen, die den Druckernamen enthält

Ein Fenster erscheint, in dem der Benutzer den linken Rand/oberen Rand für das gegebene Layout/Drucker ändern kann.

#### Parameter für Funktion f) Druckvorschau

- Layoutvariable: Auswahl einer der Variablen, die den Layoutnamen enthält
- Druckstart-Aktion: Auswahl einer definierten Druckstart-Aktion

Das Druckvorschau Fenster des Layouts erscheint. Die Druckstart-Aktion ist optional, wenn diese angegeben wird, wird vor Anzeige der Druckvorschau diese



Druckstart-Aktion durchlaufen. Hintergrund: eventuell müssen ja noch Commander Vars -> LP3 Extern Vars transferiert werden, und diese definiert ja die Druckstart-Aktion.

Parameter für Funktion g) Programm beenden

- keine zusätzlichen Parameter

Das Commander-Programm wird beendet.

Parameter für Funktion h) Benutzerwechsel

- keine zusätzlichen Parameter

Die Benutzer-Name Eingabe/Passwort Eingabe erscheint.

Parameter für Funktion i) Tabellenvariable auffrischen

- Tabellen-Variable: Auswahl einer Tabellen-Variablen (nicht Untervariable!)

Damit wird bei Betätigung des Druckknopfes die angegebene Tabellen-Variable frisch vom Commander eingelesen (und damit verbundene Tabellen-Felder zeigen dann die neuen Datensätze an).

Parameter für Funktion j) Anwendung starten

- Anwendung: Angabe einer Anwendung, ggf. mit Pfad und Kommandozeile (z.B. ‚notepad.exe c:\hilfe.txt‘)

Bei Betätigung des Knopfes wird die Anwendung gestartet. Hilfreich auch zum Starten einer externen Tastatur.

Parameter für Funktion k) Seite wechseln

- Seite: Nummer der Seite, zu der man wechseln möchte. Die Funktion ist nur sinnvoll, wenn man den LPCommander mit mehreren Bildschirmseiten nutzt.

### 6.3 Auswahlliste

- Name: Frei definierbarer Name des Feldes
- Statuszeilentext: Frei definierbare Info für Statuszeile
- X, Y: je ein Eingabefeld: Position der Klappliste (in Pixel)
- Breite, Höhe: je ein Eingabefeld: Größe der Klappliste
- Sperrbedingung: Identisch zu der LP3 Layout-„Druckbedingung“, schaltet Feld je nach Variablen aktiv
- Listenanzeige: Nein=Klappliste, Ja=Zeilenliste
- Textfarbe: Knopf „...“: Änderung der Textfarbe
- Textfarbe (gesperrt): Knopf „...“: Änderung der Textfarbe für gesperrtes Feld
- Hintergrundfarbe: Knopf „...“: Änderung der Hintergrundfarbe

- Schriftart                    Knopf „...“: Änderung der Schriftgröße/Schriftart
- Eingangsvariable:        Knopf „...“: öffnet Variablenauswahlfenster (Eingangsvariable bestimmt Vorauswahl der Liste)
- Ausgangsvariable:      Knopf „...“: öffnet Variablenauswahlfenster (Ausgangsvariable enthält gewähltes Element der Liste)
- Funktion:                 Klappliste mit folgenden Funktionen:
  - \* a) Auswahl Druckeranschluss
  - \* b) Auswahl Etikett
  - \* c) Feste Auswahlliste
  - \* d) Variable Auswahlliste
  - \* e) Dateiwahl

Parameter für Funktion a) Auswahl Druckeranschluss

- keine zusätzlichen Parameter

Es werden alle LP3 Drucker aufgelistet. Automatisch vorgewählt wird der Drucker, welcher der Eingangsvariablen entspricht. Der ausgewählte Druckername wird in die Ausgangsvariable geschrieben.

Parameter für Funktion b) Auswahl Etikett

- Verzeichnis-Variable

Es werden alle Dateien des LP3-Layoutverzeichnisses gelistet. Automatisch vorgewählt wird das Layout, welches der Eingangsvariablen entspricht. Der ausgewählte Layoutname (inklusive Pfad) wird in die Ausgangsvariable geschrieben. Möchte man die Dateien ein anderes als das LP3-Layoutverzeichnis listen, so kann man eine Variable angeben, die das Verzeichnis enthält (fehlt diese, oder ist sie leer -> Standard-Layout-Verzeichnis).

Parameter für Funktion c) Feste Auswahlliste

- Listenelemente        Hier kann der Benutzer die festen Listenelemente Semikolon-separiert eingeben

Alle Elemente werden einzeln gelistet. Automatisch vorgewählt wird das Element, welches der Eingangsvariablen entspricht. Das ausgewählte Element wird in die Ausgangsvariable geschrieben.

Man kann zudem auch die Listenelemente mit „unsichtbaren Werten“ versehen (durch „=-“ Zeichen vom Anzeigetext getrennt), dann werden diese Werte in die Ausgangsvariable geschrieben (bzw. von einer Eingangsvariablen erwartet):

Beispiel 1: „Test1;Test2;Test3“ ergibt eine Auswahl zwischen diesen drei Texten, der gewählte Text wird in Ausgangsvariable geschrieben.

Beispiel 2: „Test1=Jochen;Test2=Peter;Test3=Hugo“ ergibt ebenfalls eine Auswahl zwischen den drei „Test“-Texten (alles hinter dem Gleichheitszeichen wird nicht angezeigt), bei einer Auswahl von „Test3“ würde jedoch der Text „Hugo“ in die Ausgangsvariable geschrieben. Dies ist z.B. praktisch einsetzbar bei einer Verzeichnisauswahl: „Standard =;Deckel=C:\Deckel;Seite=C:\Seite“. Der Benutzer sieht dann nur Standard/Deckel/Seite, wählt aber tatsächlich das entsprechende Verzeichnis.

#### Parameter für Funktion d) Variable Auswahlliste

- Listenelement-Variable      Knopf „...“: öffnet Variablenauswahlfenster

Ist die Variable eine Haupt-Tabellvariable, so werden die einzelnen Datensätze der Tabelle (1. Spalte) aufgelistet. Wird die Tabellvariable aufgefrischt (z.B. über Druckknopf mit entsprechender Funktion), so wird auch die Liste neu gefüllt.

Ist die Variable eine andere Variable, so erwartet die Klappliste einen Inhalt a la „Element1;Element2;Element3;...“ und listet dann diese einzelnen Elemente (so wie bei Funktion „feste Ausgangsliste“). Ändert sich der Inhalt der Listenelement-Variablen, so ändert sich auch der Inhalt der Klappliste entsprechend.

#### Parameter für Funktion e) Dateiwahl

- Pfad/Filter-Variable

Ein Pfad/Filter ist notwendig, damit die Dateiwahl die entsprechenden Dateien auflisten kann. Beispiel: alle TXT Dateien eines Verzeichnisses: C:\TEST\\*.TXT  
Möchte man die Liste auffrischen, so muss man die Pfad/Filter Variable kurz leer setzen, und dann wieder das korrekte Verzeichnis/Filter setzen.

## 6.4 Anzeigefeld

Das Anzeige-Feld dient rein zur Anzeige von Grafiken oder auch von konstanten Text. Sie kann also nicht per Maus/Tastatur bedient werden.

- Name      Frei definierbarer Name des Feldes
- Statuszeilentext      Frei definierbare Info für Statuszeile
- X, Y:      je ein Eingabefeld: Position des Anzeigefeldes (in Pixel)
- Breite, Höhe:      je ein Eingabefeld: Größe
- Anzeigebedingung      Identisch zu der LP3 „Druckbedingung“, macht Feld je nach Variablen sichtbar
- Hintergrundfarbe:      Knopf „...“: Änderung der Hintergrundfarbe
- Rahmen:      Ja/Nein. Falls ja, wird ein Rahmen um das Feld angezeigt.
- Funktion:      Klappliste:
  - \* a) Text
  - \* b) Grafik
  - \* c) Druckerstatus
  - \* d) Layout-Vorschau
  - \* e) Infozeile
  - \* f) Web-Zeile

#### Parameter für Funktion a) Text

- Textausrichtung:      Auswahl: links/mittig/rechts
- Mehrzeilig:      Auswahl: ja/nein. Bei mehrzeiligen Texten wird bei Zeilenumbruch-Zeichen oder bei Erreichen der maximalen Feldbreite umgebrochen.
- Textfarbe:      Knopf „...“: Änderung der Textfarbe
- Schriftart      Knopf „...“: Änderung der Schriftgröße/Schriftart
- Anzeigetext:      Eingabefeld: Beschriftungstext des Feldes
- Eingangsvariable      Knopf „...“: öffnet Variablenauswahlfenster

Ist keine Eingangsvariable gesetzt, so wird der konstante „Anzeigetext“ angezeigt; ansonsten der aktuelle Wert der Variablen.

#### Parameter für Funktion b) Grafik

- Grafik Knopf „...“: Auswahl einer Grafikdatei
- Seitenverhältnis Proportional, 1:1, Beliebig, Proportional-Anker. Bestimmt, wie die Grafik dem Feld angepasst wird
- Transparentmodus Auswahl: ja/nein. Wenn auf „ja“ gesetzt, so bestimmt die Einstellung „Hintergrundfarbe“, welche Farbe in der Grafik transparent dargestellt wird.
- Eingangsvariable Knopf „...“: öffnet Variablenauswahlfenster

Hat man keine Variable angegeben, so zeigt das Feld immer die ausgewählte Grafikdatei. Ansonsten wird versucht, der in der Variablen enthaltene Grafikname zu laden und anzuzeigen. Gibt es die Grafik dabei nicht, so wird die eingestellte Grafik angezeigt.

#### Parameter für Funktion c) Druckerstatus

- Textausrichtung: Auswahl: links/mittig/rechts
- Textfarbe: Knopf „...“: Änderung der Textfarbe
- Schriftart Knopf „...“: Änderung der Schriftgröße/Schriftart
- Übertragungsbalken Auswahl: ja/nein
- Druckernamen-Variable Knopf „...“: öffnet Variablenauswahlfenster

Dieses Feld „weiß“ durch den Druckernamen in der Variablen, den Status welchen Druckers es anzeigen soll. Ist „Übertragungsbalken“ auf „nein“, so wird beim Spoolen der Fortschritt nur als Prozent-Zahl angezeigt. Ist der Balken auf „ja“, so wird am rechten Ende des Anzeigefeldes der typische Übertragungsbalken sichtbar.

#### Parameter für Funktion d) Layout-Vorschau

- Seitenverhältnis Proportional, 1:1, Beliebig. Bestimmt, wie die Layout-Grafik dem Feld angepasst wird
- Layoutnamen-Variable Knopf „...“: öffnet Variablenauswahlfenster

Das „Laden“-Vorschaubild, das in der angegebenen Layoutdatei enthalten ist, wird intern geladen und angezeigt. Existiert die Layoutdatei nicht, so wird ein leeres Feld (Hintergrundfarbe) angezeigt.

#### Parameter für Funktion e) Infozeile

- Textausrichtung: Auswahl: links/mittig/rechts
- Textfarbe: Knopf „...“: Änderung der Textfarbe
- Schriftart Knopf „...“: Änderung der Schriftgröße/Schriftart

Dieses Textfeld wird zur Laufzeit immer den Infotext des aktuellen Feldes (=mit Mauszeiger berührt, oder per TAB angesprungen) anzeigen.

#### Parameter für Funktion f) Web-Anzeige

- URL: die anzuzeigende URL, z.B. [www.google.de](http://www.google.de), oder eine Drucker-IP-Adresse, etc.
- Eingangsvariable: falls die URL variabel sein soll, so ist hier eine entsprechende Variable mit dem URL-Inhalt anzugeben. Sobald sich die Variable ändert, wird auch die Web-Anzeige aufgefrischt.

## 6.5 Tabellenfeld

Dient zum Anzeigen von n Spalten und m Zeilen, erhält diese Daten von einer Tabellenvariablen. Der Benutzer kann dann eine der Zeilen auswählen, und füllt damit automatisch die Tabellen-Untervariablen mit

den Werten des gewählten Datensatzes. Ist kein Datensatz gewählt, so sind alle Untervariablen leer. Die Haupt-Tabellenvariable kann zudem auch in Anzeigefeldern/Formeln/Sperrbedingungen benutzt werden: Sie enthält die Nummer des aktuell gewählten Datensatzes (1 bis n), ist kein Datensatz gewählt, so enthält sie „0“.

- Name: Frei definierbarer Name des Feldes
- Statuszeilentext: Frei definierbare Info für Statuszeile
- X, Y: je ein Eingabefeld: Position der Tabelle (in Pixel)
- Breite, Höhe: je ein Eingabefeld: Größe
- Sperrbedingung: Identisch zu der LP3 Layout-„Druckbedingung“, schaltet Feld je nach Variablen aktiv
- Rahmen: Ja/Nein. Falls ja, wird ein Rahmen um das Feld angezeigt.
- Hintergrundfarbe: Knopf „...“: Änderung der Hintergrundfarbe
- Textfarbe: Knopf „...“: Änderung der Textfarbe
- Textfarbe (gesperrt): Knopf „...“: Änderung der Textfarbe für gesperrtes Feld
- Linienfarbe: Knopf „...“: Änderung der Linienfarbe
- Selektionsfarbe: Knopf „...“: Änderung der Selektionsfarbe
- Schriftart: Knopf „...“: Änderung der Schriftgröße/Schriftart
- Spaltenkopf anzeigen: Auswahl: ja/nein zur Anzeige des Spaltennamen
- Spaltenkopffarbe: Knopf „...“: Änderung der Spaltenkopf-Hintergrundfarbe
- Spaltenkopfschriftart: Knopf „...“: Änderung der Spaltenkopf-Schriftgröße/Schriftart
- Tabellenvariable: Knopf „...“: öffnet Variablenauswahlfenster

Tabellenfelder können auch direkt mit Eingabefeldern und Druckknöpfen und Aktionen in Verbindung stehen:

Eingabefelder mit Funktion „Sucheingabe Tabellen-Feld“ sind mit dem jeweiligen Tabellenfeld gekoppelt: tippt der Benutzer im Eingabefeld, so wird der „nächst beste“ Eintrag im Tabellenfeld selektiert. Klickt der Benutzer im Tabellenfeld einen Datensatz an, so wird das Eingabefeld ebenso aufgefrischt.

Druckknöpfe mit Funktion „Tabellenvariable auffrischen“ erzwingen ein neues Laden der Tabellenvariablen-Inhalte, ist mit dieser Tabellenvariable ein Tabellenfeld verbunden, so wird auch dieses Feld aufgefrischt.

Die Aktionen „Textdatei-Import“, „Serieller Import“ und „Druckerüberwachung-Automation“ ändern ebenfalls Ziel-Tabellen. Ist eine dieser Zieltabellen in einer Tabellenvariablen benutzt, so wird die entsprechende Tabellenvariable aufgefrischt. Ist mit dieser Tabellenvariable ein Tabellenfeld verbunden, so wird auch dieses Feld aufgefrischt.

## 6.6 Allgemeine Einstellungen

Vergleichbar mit den Etiketten-Eigenschaften (wenn Benutzer Rechtsklick auf leere Fläche im Etiketteneditor tätig), so gibt es auch allgemeine „Commander-Eigenschaften“ bei Rechtsklick auf eine leere Fläche.

- Titeltext: Text, der im Titel des Commander Fensters erscheinen soll
- Prg. beenden: Auswahl: ohne Rückfrage; nur nach Rückfrage; nur über Druckknopf. Entscheiden, wie der Benutzer das Commander-Fenster schließen kann.
- Breite: Breite des Commander-Fensters in Pixel
- Höhe: Höhe des Commander-Fensters in Pixel
- Seitenanzahl: Falls mehrere Seiten benötigt werden, so ist die Anzahl (bis zu 20) hier einzustellen. Man wechselt per Knopf-Feld, Funktion „Seite wechseln“, zwischen den einzelnen Seiten.
- Start-Seite: Verfügt man über mehrere Seiten, so kann man hier angeben, welche Seite bei Programmstart angezeigt wird.
- Titelleiste: Auswahl: ja/nein. Will man den Commander für beispielsweise eine Touch-Anwendung Bildschirmfüllend darstellen, so kann man die Titelleiste abschalten.
- Feste Fenster-Position: Standardmäßig wird das Commander-Fenster bei Start auf dem Desktop zentriert. Möchte man eine feste Position vorgeben, so ist diese Option auf „ja“ zu stellen. Man erhält dann zwei weitere Eigenschaften (X Position und Y Position), in welcher man die gewünschte Position in Pixel angeben kann. Bei einer

bildschirmfüllenden Anwendung kann es Sinn machen, die Position auf negative Werte (z.B. -2, -2) zu setzen, um den Programmfensterrahmen unsichtbar zu machen.

- Hintergrundfarbe Grundfarbe des Commander-Fensters
- Raster/Rahmen/Lineal-Einstellungen wie im Etiketteneditor

## 7 LP3 Commander Benutzerverwaltung

Innerhalb des LP3 Commander Editors gelangt man per Menüpunkt „Benutzerverwaltung“ zu einer eigenen Benutzerverwaltung innerhalb des aktuellen Commander-Projektes. Dort kann man Benutzer anlegen (Benutzername, Passwort, Berechtigungsstufe 0 – 100). Diese Benutzer sind unabhängig von den LP3 Etiketteneditor-Benutzern, sie gelten nur innerhalb des Commander-Fensters!.

Ist die Commander-Benutzerverwaltung aktiv, so können nur Benutzer, deren Berechtigungsstufe der Mindest-Stufe des zu bedienenden Feldes entsprechen, das Feld per Maus/Tastatur bedienen.

## 8 LP3 Commander Funktionstasten

Die Taste „F1“ ist Windows-seitig mit der Hilfe-Funktion belegt. Im Commander kann man seine eigene Hilfe-Anwendung + Hilfedatei auf Druck von F1 starten. Hierzu ist in der Datei „LPCommander.INI“ der Eintrag

[Settings]

...

HelpFile=ANWEDUNG.exe PARAMETER

Einzugeben. Zum Starten des Windows Notepad mit einer Hilfe-Textdatei kann man z.B. folgenden Eintrag vornehmen:

HelpFile=notepad.exe "C:\Meine Hilfe\Hilfe.txt"

Alle anderen Funktionstasten (F2-F12) sind frei mit einer Aktion oder einer Feldaktivierung konfigurierbar. Hierzu das Funktionstasten-Fenster im Commander Editor (unter Menüpunkt „Einstellungen“) aufrufen, und die Tasten entsprechend konfigurieren.

Eingestellte Aktionen werden dann auf Tastendruck einmalig durchlaufen (vergleichbar mit der Druckknopf-Funktion „Start Aktion einmalig“). Eingestellte Felder werden bei Tastendruck aktiviert, d.h.: bei Eingabefeldern wird der Eingabecursor in das Feld gesetzt. Bei Druckknöpfen wird die eingestellte Druckknopf-Funktion ausgeführt. Bei Auswahllisten wird die Liste angezeigt. Tabellenfelder werden mit dem Tastatur-Eingabefokus versehen. Um die Felder einstellen zu können, bedürfen sie eindeutige Namens-Bezeichnungen (einstellbar in den jeweiligen Feldeigenschaften).

Bitte beachten: die F10 Taste wird parallel von Windows zum Aktivieren der Menüleiste benutzt. Da der Commander keine Menüleiste hat, passiert bei F10 zunächst nicht augenscheinliches, allerdings wechselt Windows den Eingabefokus auf die nicht existierende Menüleiste. D.h. das nun alle nachfolgenden Tastatureingaben vom Commander-Hauptfenster nicht beachtet werden, bis wieder die F10 Taste betätigt wird (oder der Benutzer per Maus wieder in das Hauptfenster klickt).

## 9 LP3 Commander Eingabefolge

Der LP Commander unterstützt Windows-gemäß die Tabulator-Taste zum Navigieren zwischen einzelnen Feldern per Tastatur. TAB springt zum nächsten Feld, SHIFT-TAB zum vorherigen. Die Springreihenfolge entspricht hierbei der Feldreihenfolge, so wie sie in der Feldliste im Editor angezeigt wird. Durch Verschieben der Felder in der Liste wird also auch die Sprungreihenfolge geändert.

Bei speziellen Anforderungen ist es aber notwendig, nach Eingabeende in einem Feld definiert zu einem anderen zu springen. Typische Fälle sind Eingabefelder, die mit einem Tastatur-Scanner bedient werden sollen. Um hier per Default immer auf dem Scan-Eingabefeld zu stehen, kann man die anderen Feldern in der Commander-Oberfläche anweisen, immer wieder das Scan-Eingabefeld zu aktivieren. Dies vermeidet unnötige Mausclicks bzw. Tab-Betätigungen in der produktiven Anwendung.

Die Eingabefolgen werden im gleichnamigen Fenster (unter Menüpunkt „Einstellungen“) definiert. Hier gibt man jeweils ein Startfeld und ein Zielfeld an (die Felder benötigen dann eindeutige Namensbezeichnungen). Sobald im Commander dann die Benutzereingabe im Startfeld abgeschlossen wurde (bei Eingabefeldern: ENTER-Taste, bei Druckknöpfen ENTER bzw. Mausclick, bei Auswahllisten ebenfalls ENTER oder Mausclick), wird automatisch der Eingabefokus auf das eingestellte Zielfeld gesetzt, so als wäre es als nächstes Feld per TAB angesprungen worden).

## 10 Projekt-Transfer / INI-Datei

Im Editor befindet sich ein „Test“-Knopf, welcher die LPCommander.exe mitsamt der aktuellen Projektdatei (Projektname.LPC) aktiviert. Dadurch kann man also schnell ein Projekt laufen lassen und dessen Funktion testen.

Für den Transfer eines Commander-Projektes muss (a) die .LPC-Datei des Projektes kopiert werden, und (b) alle eventuell benutzten Grafikdateien (Textdateien/Datenbank-Dateien...). Grafikdateien sollten aus dem LP3-Grafik-Verzeichnis „Pictures“ benutzt werden (um keine absolute Pfade im Projekt zu hinterlegen, die auf dem Zielrechner dann vielleicht nicht existieren).

Zum LPCOMMANDER.EXE gehört zudem eine INI-Datei gleichen Namens (ändert man den EXE Namen, so kann man auch eine INI mit entsprechenden Namen nutzen). Auch möglich ist es, diese INI Datei im gleichen Verzeichnis wie die LPC Datei, mit gleichen Dateinamen, zu legen.

In der INI merkt sich ein laufendes Commander-Projekt beispielsweise die Spalten-Größen eines Tabellenfeldes, darin kann aber auch der Benutzer bestimmte Aktionseinstellungen abändern, ohne den Commander Editor benutzen zu müssen.

Generell gilt: die Einstellungen für ein Commander-Projekt befinden sich in der INI-Datei in der Sektion „LPC-Dateiname“ (Dateiname ohne Pfad), also beispielsweise [TEST.LBC].

Darin hinterlegt der Commander die Spaltengrößen eines Tabellenfeldes per „Size\_Feldname\_Spaltennummer“, also beispielsweise „Size\_Tabfeld\_001=100“.

Andere Einträge kann man manuell vornehmen, um Aktionen zu beeinflussen. Beispielsweise kann man die Script-Dateinamen umändern (da auf dem Zielrechner andere Pfade existieren), oder die Einstellungen der Aktions-Schnittstelle (z.B. anstatt von COM1 eben COM2 nutzen). Hier nun die verschiedenen Einstellmöglichkeiten:

Aktion: Datei-Import/Druckstart/Druckstop/Datei-Automation

Aktionsname\_File=C:\XYZ\TEST.TXT

➔ Ändern der Import-Datei bzw. der Schreib-Datei

Aktion: Druckstart/Druckstop/VBScript

Aktionsname\_ScriptFile=C:\XYZ\VB.TXT

➔ Ändern der VBScript-Datei

Aktion: Serieller Import/Druckstart/Druckstop/Serielle Automation

Aktionsname\_Port=...

➔ Ändern der Port-Einstellungen. Je nach Grundtyp (serieller Port, Netzwerk-Port, USB) ist ein anderer Aufbau der Einstellungen notwendig:

- seriell:

COMx:Baud,Parity(0=N,1=E,2=O),Handshake(0=none,1=XonXoff,2=HW1,3=HW2,4=HW3),  
Databits,Stopbits

Bsp: X\_Port=COM2:19200,0,1,8,1

- Netzwerk:

IP-Adresse:Port

Bsp: Y\_Port=192.168.0.1:9100

- Netzwerk:

USB-Device-Name

Bsp: U\_Port=Hugo

Aktion: OPC Client

Man kann zwei Zeitangaben für eine OPC Client-Aktion in der INI hinterlegen. Bitte beachten: diese Zeiten sollten nur bei Problem-Fällen geändert werden!

Polltime\_Aktionsname=500 stellt die OPC Server Poll-Zeit (Abfragerate) auf 500 ms. Standardwert sind 750 ms.

Inittime\_Aktionsname=1000 gibt an, dass der LP3 Commander 1000 ms nach Start des OPC Servers warten soll, bevor er Zugriffe auf diesen ausführt. Standardwert sind 0 ms.

Aktion: Commander Start/Ablauf

Man kann festlegen, ob der Commander beim Programmstart gleich minimiert werden soll:

StartMinimized=1

Es ist zudem bestimmbar, ob der Commander sich beim Minimieren in den System Tray oder in die Taskbar (Standardeinstellung) legen soll:

TrayMinimized=1

Diverse Fehler im Ablauf werden dem Benutzer per Meldungsfenster (Message Box) automatisch angezeigt, z.B. „Kein Etikett gewählt“. Möchte man diese Meldungen unterdrücken, so ist der Eintrag:

NoMessageBox=1

zu setzen.

Neben den [\*LPC] Sektionen gibt es noch eine allgemeine Sektion [Settings]. Hier kann man folgende Einstellungen vornehmen:

DefLPC=C:\xyz\Test.LPC

Der Eintrag „DefLPC“ wird genutzt, falls das LPCommander.exe gestartet wird, und kein LPC-Dateiname per Kommandozeile übergeben wird. Fehlt sowohl der DefLPC-Eintrag, als auch die Kommandozeile, so sucht



und nutzt der Commander das erste LPC-File im Startverzeichnis. Gibt es auch kein LPC File im Verzeichnis, so wird ein leeres Commander-Fenster mit entsprechendem Hinweis angezeigt.