

# Glossar

Hier werden die Begriffe, die in dieser Dokumentation verwendet werden, näher erklärt.

## A

### Abschnitt

Mit **Abschnitt** wird ein Bereich in der Primus.CFG.XML bezeichnet. Ein Abschnitt beginnt mit einem **Start-Tag** und endet mit einem **End-Tag**.

```
<GUI_Data_Lists> <!--Start-Tag-->
<!--Inhalt des Abschnitts-->
</GUI_Data_Lists> <!--End-Tag-->
```

Ausnahme sind die Abschnitte die aus einem sogenannten **Leer-Tag** bestehen. Hier wird der Inhalt des Abschnitts zusammen mit Start- und End-Tag in eine Zeile geschrieben. Als Inhalt kommen in dem Fall nur **Attribute** in Frage.

```
<ListViewSettings splitterdist="376" mode="0" showbehinddatalist="1"
splitterorient="Ver" />
```

### Attribut

Ein **Attribut** ist Bestandteil eines **Tag**. Ein Attribut muß innerhalb eines Tags immer eindeutig sein, hierbei wird die Groß-/Kleinschreibung unterschieden. Einem Attribut ist stets ein Wert zugewiesen. Die Wertzuweisung erfolgt mittels =, der Wert selbst muß in " gesetzt sein. Die Wertzuweisung kann auch leer sein, die Zuweisung erfolgt dann durch zwei aufeinanderfolgende "".

```
<Preview mode="viewer" />
<Filename label="" /> <!--Zuweisung eines leeren Werts-->
```

Das **Tag** ist in diesem Fall **Preview**, das **Attribut** **mode** und der **Wert** **viewer**. Bestimmte Zeichen sind innerhalb eines Wertes nicht erlaubt. Falls diese in einem Wert jedoch trotzdem benötigt werden, müssen sie durch sog. Entitäten ersetzt werden. Die folgende Tabelle zeigt die nicht zulässigen Zeichen und den Ersatztext.

Verbotenes Zeichen	Entität	Bemerkung
<	&lt;	lower than (kleiner als)
>	&gt;	greater than (größer als)
'	&apos;	apostroph
„	&quot;	quota (Anführungszeichen)
&	&amp;	ampersand (kaufm. und)

## D

### Dateiendung

Bei der Dateiendung handelt es sich um den rechten Teil des Dateinamens der mit einem Punkt (.) vom linken Teil des Dateinamens getrennt ist. Programme erkennen an der Dateiendung um welche Art Datei es sich handelt und ob sie diese verarbeiten können. In Windows können zu bestimmten Dateiendungen, Programme eingetragen werden die zum Verarbeiten der Datei aufgerufen werden.

### Dienst

Ein Dienst, gelegentlich auch Daemon genannt, ist ein Prozess der auf einem Server im Hintergrund ohne Benutzerinteraktion läuft. Dienste werden von Programmen aufgerufen um bestimmte Funktionen, die der Dienst zur Verfügung stellt, zu nutzen oder ihm bestimmte Aufgaben zu übergeben. Dienste können auch über das Netzwerk von anderen Rechnern aus genutzt werden.

### DLL

Die Abkürzung **DLL** steht für **Dynamic Link Library**, und bezeichnet eine Programmbibliothek die dynamisch, also nur bei Bedarf, vom aufrufenden Programm in den Speicher geladen wird. Bei Primus werden z.B. [Addins](#) auf diese Weise eingebunden. Dies betrifft sowohl Primus eigene als auch Kundenspezifische Zusatzfunktionen.

## E

### End-Tag

Ein **End-Tag** kennzeichnet das Ende eines Tag. Es besteht aus `</Tag-Name>`. Ausnahme: Bei einem [Leer-Tag](#) wird ein vereinfachter Abschluss ohne Name, nur mit `/>` verwendet.

### ERP-System

Ein **ERP-System** ist ein System für Enterprise-Resource-Planning (ERP). Die Aufgabe eines solchen Systems ist vereinfacht ausgedrückt, die Ressourcen die einem Unternehmen zur Verfügung stehen, zeitlich und mengenmäßig so zu steuern das die betrieblichen Abläufe möglichst störungsfrei ablaufen können, um so den Wertschöpfungsprozess zu optimieren.

## F

## Freigabe

Eine **Freigabe** ist ein Verzeichnis auf einem Dateiserver welches unter einem bestimmten Namen aus dem Netzwerk heraus angesprochen werden kann. Die Freigabe kann einen anderen Namen haben als das Verzeichnis im Dateisystem der Platte, auch kann ein Verzeichnis verschiedene Freigaben haben. Jede Freigabe kann eigene Berechtigung für die Benutzer mit sich bringen. Freigaben werden z.B. in der [Primus.CFG.XML](#) im Abschnitt [Folder\\_Paths](#) verwendet.

## I

### Icon

Ein Icon ist ein Symbolbild das z.B. einen bestimmten Dateityp darstellt, oder auch als grafisches Element in der Primus Oberfläche verwendet wird.

## K

### Kommentar

Ein **Kommentar** ist ein speziell markierter Text im Quellcode eines Programms oder einer Konfigurationsdatei. Dieser Text soll die Verständlichkeit für den Bearbeiter erhöhen. Ein Kommentar wird vom jeweils lesenden Programm ignoriert. In [XML](#) beginnt ein Kommentar mit `<!--` und endet mit `-->`. Alles was zwischen Kommentaranfang und -ende steht wird vom [XML-Parser](#) ignoriert. Beispiel:

```
<GUI_Data_Lists>
  <!-- Das ist ein Kommentar -->
  <GUI_VirtualFolders_List>
```

### Kontextmenüs

Um das [kontextmenue](#) aufzurufen müssen Sie die rechte Maustaste auf einem ausgewählten Objekt kurz drücken. Die Befehle unterscheiden sich je nach ausgewähltem Objekt.

## L

### Lebenszyklus

### Leer-Tag

Bei einem **Leer-Tag**, stehen Anfangs- und End-Tag in einer Zeile. Ein Leer-Tag kann keine weitere

Unterteilung enthalten. Ein Leer-Tag kann ausschließlich [Attribute](#) enthalten. Einträge in [Lists](#) sind z.B. Leer-Tags. Beispiel:

```
<entry value="0" text="Normal" />
```

## S

### Stammelement

Das **Stammelement** ist ein spezielles [Primusspezifisches Tag](#).

### Start-Tag

Ein **Start-Tag** definiert den Beginn eines [Tags](#). Es besteht immer aus [<Tag-Namen>](#). Ausnahme: Bei einem [Leer-Tag](#) entfällt das [>](#) hinter dem Tag-Namen und es folgen direkt die [Attribute](#).

## T

### Tag

Bei einem **Tag** handelt es sich um ein XML Strukturelement. Ein Tag hat einen [Anfang](#), einen Namen und ein [Ende](#). Darüber hinaus können auch noch [Attribute](#) zugewiesen werden.

```
<Das_Tag das_Attribut="Wert des Attributs" />
```

### Tag-Name

Der **Tag-Name** ist ein notwendiger Bestandteil eines [Tag](#).

## X

### XML

Die Abkürzung **XML** bedeutet *Extensible Markup Language* und beschreibt eine Textdatei, deren Inhalt hierarchisch und strukturiert gegliedert ist. Dadurch ist der Inhalt sowohl für Menschen als auch Maschinen gleichermaßen lesbar. Die Daten werden in [Tags](#) und [Attributen](#) abgelegt.

### XML-Parser

Bei einem **XML-Parser** handelt es sich um ein Programm oder den Teil eines Programms das/der eine XML-Datei liest und zur weiteren Verarbeitung zur Verfügung stellt. Der XML-Parser ist dabei so aufgebaut, dass er nur die jeweils gewünschten Informationen ausliest und alle anderen Informationen ignoriert.

[klaerungsbedarf](#)

From:

<https://dokuwiki.unitec.de/> - **Unitec Wiki für Produktdokumentationen**

Permanent link:

<https://dokuwiki.unitec.de/doku.php?id=primus:glossar&rev=1675857672>

Last update: **2023/02/08 12:01**

