

AppAttributes

Im Verzeichnis **AppAttributes** liegen die Ini-Dateien mit denen der Abgleich der Datenfelder zwischen [Solid Edge](#) und Primus konfiguriert wird. Eine Liste der Felder und ihrer Schreibweise in Solid Edge, liegt im Verzeichnis und heißt *Properties.txt*.

se_getprop_XXX.ini

Steuert den Abgleich der mit dem [Menübefehl seGetAllProperties](#) aufgerufen wird. Das XXX oben steht für den jeweiligen Dateityp von Solid Edge.

Syntax

Die Ini-Datei besteht nur aus einer Section mit dem Namen **[PropertiesToPrimus]**. Jedes Zuweisungspärchen muß in einer eigenen Zeile stehen. Links steht das Primus [Datenfeld](#) und rechts durch ein = getrennt die Solid Edge Dateieigenschaft. Diese wird mit dem Schema **Eigenschaftsbereich . Eigenschaftsname** angegeben.

Bereich	Eigenschaft
SummaryInformation	Titel
	Thema
	Autor
	Kommentar
ProjectInformation	Revisionsnummer
	Dokumentnummer
	Projektname
FileProperties	CreatingDate
Custom	Benutzerdefinierte Eigenschaft

Beispiel

```
[PropertiesToPrimus]
Title=SummaryInformation.Titel
Revision=ProjectInformation.Revisionsnummer
Dimension=Custom.Dimension
```

se_getprop_XXX_ADDIN.ini

Steuert den Abgleich für **Speichern unter** mit dem [Solid Edge Addin](#) damit werden bestimmte [predata](#) Felder von der bestehenden XML-Datei und nicht nur aus den Solid Edge Dateieigenschaften gelesen. Dadurch werden auch nicht nur die Felder in die neue Datei übertragen, die in dem [GUI_NewDocument_DataSheet](#) definiert sind, sondern auch die Felder die in der **_ADDIN.ini** zusätzlich angegeben worden sind. Z.B. `Note=# (Note)`. Die Syntax ist identisch zu

[se_getprop_XXX.ini](#)

se_prop_XXX.ini

Diese Ini Datei steuert den Abgleich der vom Solid Edge Addin beim Öffnen über Primus vorgenommen wird. Die Ini-Datei besteht aus mehreren Sections die für die einzelnen Eigenschaftsbereiche der Solid Edge Dateieigenschaften stehen, sowie dem Bereich [\[PropertiesToPrimus\]](#). Letzterer ist wie unter [se_getprop_XXX.ini](#) beschrieben aufgebaut und steuert welche Werte beim Speichern einer Solid Edge Datei in Primus zurück geschrieben werden sollen.

[PropertiesToPrimus]

In der `se_prop_par.ini`, `se_prop_psm.ini` und `se_prop_asm.ini` kann `[PropertiesToPrimus] Weight=Weight.kg(##.##.##0.000)` ausgelassen werden, wenn nicht bei jedem Speichern das Gewicht übertragen werden soll. Zusätzlich gibt es die Menü-Funktion [seUpdateWeight](#), die in dem Fall für alle selektierten Dateien das Gewicht aktualisiert.

Listeninhalte verwenden

Beim Übertragen der Wert von Primus an [Solid Edge](#), ist es möglich statt des eigentlichen Wertes des Datenfelds stattdessen einen zugeordneten Wert aus einer Liste zu übertragen. Dies ist beispielsweise sinnvoll bei dem Benutzernamen. Dieser ist in Primus der Windows-Loginname des Anwenders ohne davorgestellte Domäne. Für den Eintrag in Solid Edge kann es aber sinnvoll sein z.B. für die Historientabelle auf der Zeichnung eine abweichende Schreibweise zu verwenden. Abweichende Werte für die Ersetzung müssen hier jedoch mit einem Pipe (|) und nicht mit einem Doppelpunkt abgetrennt werden, wie dies in der CFG.XML üblich ist. Dafür stehen folgende Funktionen zur Verfügung:

- [GetListText](#)
- [GetListTextFromDict](#)

Mittels Listeninhalten ist es auch möglich [Bedingte Werte von Primus an Solid Edge zu übertragen](#).

Beispiele

- `CheckedName=CALL(GetListText CheckedName)`
- `Autor=CALL(GetListText CreatedName)` - Weist dem *Autor* in Solid Edge den im Attribut `text` in der zugehörigen Liste hinterlegten Wert zu.
- `ChangeName#=#8#CALL(GetListText ChangeName|histext)` - Weist dem *ChangeName* in Solid Edge den im Attribut `histext` in der zugehörigen Liste hinterlegten Wert zu.

Die Definition der Liste selbst, wird aus der [GUI_Document_List](#) der [CFG.XML](#) genommen, in diesem Fall beidesmal `List_User`.

```
<CreatedName data="CreatedName" type="Text" list="List_User"
position="19" width="104" visible="1" />
```

```
<ChangeName data="ChangeName" type="List" list="List_User"
position="39" width="104" visible="0" />
```

Sonderfall `se_prop_dft.ini`

Für die Übergabe der Werte der Versionshistorie in die Benutzerdefinierten Eigenschaften ([Custom]) der Draftdatei, kann man steuern wie viele Revisionen in Draft angezeigt werden und ob die Startversion berücksichtigt werden soll oder nicht. Dazu dient das Nummernzeichen (#) als Platzhalter am Eigenschaftsnamen. `Rev#=#8#Revision` bedeutet das die 8 neuesten Revisionen in die Draftdatei eingetragen werden und das Feld mit Rev beginnt und eine der fortlaufenden Nummern direkt angehängt wird. Soll dabei mit der Startversion statt der ersten Revision begonnen werden, dann wird der Angabe der Anzahl der Versionen ein Punkt (.) vorangestellt. `Rev#=#.8#Revision`

Ein typischer Eintrag mit Kommentarzeilen (#) in der `se_prop_dft.ini` sieht dann so aus:

```
# Das #-Zeichen steht für die Nummer der jeweiligen Revision.
# Die 8 steht für die letzten 8 Dateien. Ein Punkt vor der 8 (.8) heißt eine
Version niedriger.
Rev#=#8#Revision
Change#=#8#Change
ChangeDate#=#8#ChangeDate
ChangeName#=#8#CALL(GetListText ChangeName)
```

Siehe auch

- [Listenmapping](#)

From:

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

Permanent link:

<https://www.dokuwiki.unitec.de/doku.php?id=primus:appattributes&rev=1702998991>

Last update: **2023/12/19 15:16**

