

Ausführen von Kommandos

Inhalt

Einführung.....	3
Aufbau der Kommandodatei.....	3
Kommentare	3
Selektion.....	3
Parameter.....	5
Kommandos	5
Beschreibung der Kommandos	6
Kommando „Delete“	6
Beispiel	6
Parameter	6
Kommando „SetColor“	6
Beispiel	6
Parameter	6
Kommando „SetLineType“	6
Beispiel	6
Parameter	6
Kommando „SetDiameter“	7
Beispiel	7
Parameter	7
Kommando „SetProcessingType“	7
Beispiel	7
Parameter	7
Kommando „Rotate“.....	7
Beispiel	7
Parameter	7
Kommando „Translate“	8
Beispiel	8
Parameter	8
Kommando „Mirror“.....	8
Beispiel	8
Parameter	8
Kommando „InvertBend“.....	8
Beispiel	8
Kommando „PlaceText“ und „RetryPlaceText“	8
Beispiel	9

Parameter	9
Kommando „RenotchBends“	9
Beispiel	9
Parameter	10
Kommando „ReplaceSlits“	11
Beispiel	11
Parameter	11
Kommando „RenotchBendsAndReplaceSlits“	11
Beispiel	11
Parameter	11
Kommando „SmoothBendZones“	12
Beispiel	12
Parameter	12
Kommando „ExtractSheet“	12
Beispiel	12
Parameter	12
Kommando „ExtractMaterial“	12
Beispiel	12
Parameter	12
Kommando „ExtractData“	13
Beispiel	15
Parameter	15
Zeichnungsanalyse	16
Beispiel	16
Parameter	16
Kommando „ShiftTab“	16
Beispiel	16
Parameter	16
Kommando „ReplaceTemplates“	16
Beispiel	16
Parameter	17

Einführung

tc_conv.exe ist ein Tool zur Konvertierung von Dateiformaten z.B. von Dxf nach Geo. Es unterstützt eine Vielzahl von Optionen.

Eine besondere Option ist die Ausführung einer Kommandodatei. Die Kommandodatei wird zeilenweise abgearbeitet und mit der Option -X übergeben.

Beispielaufruf:

```
tc_conv.exe -i dxf -o geo -X command.txt demo.dxf
```

Kommandodateien können auch im KONvertierungstool in TruTops Boost verwendet werden.

Aufbau der Kommandodatei

Die Kommandodatei ist zeilenweise aufgebaut und wird Zeile für Zeile ausgeführt.

Es gibt 4 Arten von Zeilen:

- Kommentare
- Selektionen
- Parameter
- Kommandos

Innerhalb der Zeilen wird nicht auf Groß- und Kleinschreibung geachtet. Die hier vorgestellte Schreibweise dient nur der besseren Lesbarkeit.

Ebenso spielt es keine Rolle, wie viele Leerzeichen man als Trennung zwischen Ausdrücken verwendet und dort wo ein „=“ als Trenner verwendet wird, ist gar kein Leerzeichen nötig.

Es können beliebig viele Kommandos in einer Kommandodatei stehen. Da die Datei zeilenweise abgearbeitet wird, können diese sich gegenseitig beeinflussen. Färbt man z.B. alle grünen Elemente rot ein und löscht dann alle roten Elemente, dann sind am Schluss die grünen Elemente auch weg.

Kommentare

Kommentarzeilen beginnen mit dem Zeichen „#“. Ihr Inhalt ist beliebig und wird ignoriert. Sie dient dazu die Kommandodatei zu umschreiben.

Selektion

Die meisten Kommandos werden auf selektierten Elementen ausgeführt. Es gibt dabei eine Vielzahl von Möglichkeiten, um eine Selektion zu definieren.

Es gibt 4 Grundarten der Selektion. Sie werden durch die folgenden Schlüsselworte eingeleitet.

- Selection – Führt eine einfache Selektion aus.
- AddSelection – Erweitert die bisherige Auswahl um eine weitere Auswahl.
- SubSelection – Reduziert die bisherige Auswahl um eine Auswahl.
- DeSelection – Reduziert die Auswahl auf eine Unterauswahl.

Es gibt 8 verschiedene Auswahlverfahren.

- All – Wählt alles aus
- Type – Wählt bestimmte Elementtypen aus (line, arc, circle, text, marker, bend).
- Color – Wählt bestimmte Farben aus (white, red, blue, green, magenta, cyan, yellow).
- LineType – Wählt Elemente mit bestimmtem Linientyp aus (solid, dash, dot, dashdot).
- Length – Wählt Elemente einer gegebenen Länge aus.
- Diameter – Wählt Elemente mit einem gegebenen Durchmesser aus.
- Contour – Wählt bestimmte Konturtypen aus (open, closed, inner, outer).
- ProcessingType – Wählt Bearbeitungsarten aus (cut, mark, none, form, foil_removal)

Die beiden Angaben werden mit einem „=" getrennt.

Beispiele:

Selection=All

Wählt alle Elemente aus.

Selection=LineType dash

AddSelection=LineType dot

Wählt alle strichlierten und gepunkteten Elemente aus

Selection=Type line

SubSelection=Color red

Wählt alle roten Linien aus.

Selection=Type line

DeSelection=Color red

Wählt alle Linien aus, die nicht rot sind.

Es können beliebig viele Selektion hintereinander folgen. Wichtig zu beachten ist hierbei, dass die Datei zeilenweise abgearbeitet wird. Somit haben die folgenden Zeilen die Auswirkung, dass erst alle roten und dann alle grünen Elemente gewählt werden – was bedeutet, dass am Ende nur die grünen Elemente gewählt wurden und die erste Zeile keine Auswirkung hat.

Selection=Color red

Selection=Color green

Außerdem gilt, dass ausgeführte Kommandos in der Regel keinen Einfluss auf die Selektion haben. (Einzige Ausnahme ist natürlich das Kommando zum Löschen.) Somit kann man mit einer Selektion mehrere Kommandos ausführen.

Parameter

Mit einer Parameterzeile kann man einem Parameter einen bestimmten Wert zuordnen. Zwischen dem Parameternamen und dem Wert steht ein „=".

Es wird nur ein fester Satz an Parameternamen unterstützt. Diese Namen sind den jeweiligen Kommandobeschreibungen zu entnehmen.

Beispiel:

TargetColor = Red

Setze Rot als Zielfarbe.

Kommandos

In einer Kommandozeile wird das eigentliche Kommando ausgeführt. Es verwendet in der Regel die aktuell getroffene Selektion unter Beachtung gewisser Parameter.

Das Kommando wird eingeleitet durch das Schlüsselwort „Action“, gefolgt von einem „=" und anschließend dem Kommandonamen.

Beispiel:

Action=Delete

Löscht die aktuelle Auswahl.

Beschreibung der Kommandos

Nachfolgend eine Liste aller unterstützten Kommandos mit Angabe der Parameter, die sie verwenden.

Kommando „Delete“

Löscht die aktuell selektierten Elemente.

Beispiel

Selection=Color red
Action=Delete

Löscht alle roten Elemente.

Parameter

Keine

Kommando „SetColor“

Setzt die Zielfarbe für die aktuell selektierten Elemente.

Beispiel

Selection=Color red
TargetColor=green
Action=SetColor

Färbt alle roten Elemente grün.

Parameter

TargetColor – zu setzende Farbe (white, red, blue, green, magenta, cyan, yellow)

Kommando „SetLineType“

Setzt den Ziellinientyp für die aktuell selektierten Elemente.

Beispiel

Selection=All
TargetLineType=solid
Action=SetLineType

Sorgt dafür, dass alle Elemente den Linientyp durchgezogen erhalten.

Parameter

TargetLineType – zu setzender Linientyp (solid, dash, dot, dashdot)

Kommando „SetDiameter“

Setzt den Durchmesser für die aktuell selektierten Kreise.

Hinweis: Falls nicht nur Kreise selektiert wurden, so werden die zusätzlichen Elemente ignoriert. Dies gilt auch für Bögen.

Beispiel

```
Selection=Type Circle  
SubSelection=Diameter 10  
TargetDiameter=8  
Action=SetDiameter
```

Sorgt dafür, dass alle Kreise mit Durchmesser 10 auf Durchmesser 8 gesetzt werden.

Parameter

TargetDiameter – zu setzender Durchmesser (Default -1 – ignoriert Kommando)

Kommando „SetProcessingType“

Setzt die Bearbeitungsart (und passende Farbe) für die aktuell selektierten Elemente.

Beispiel

```
Selection=Color cyan  
ProcessingType=none  
Action=SetProcessingType
```

Alle cyan-farbene Elemente erhalten die Bearbeitungsart „keine“ (und werden rot eingefärbt).

Parameter

ProcessingType – zu setzende Bearbeitungsart
(cut, mark, none, form, foil_removal – default none)

Kommando „Rotate“

Rotiert die selektierten Elemente.

Beispiel

```
Selection=All  
RotationCenterX=10  
RotationCenterY=10  
RotationAngle=30  
Action=Rotate
```

Rotiert die gesamte Zeichnung um den Punkt (10,10) um 30 Grad.

Parameter

RotationCenterX – x-Koordinate des Rotationsmittelpunkts (Default 0)
RotationCenterY – y-Koordinate des Rotationsmittelpunkts (Default 0)
RotationAngle – Rotationswinkel (Default 0)

Kommando „Translate“

Bewegt die selektierten Elemente.

Beispiel

Selection=Type circle

TranslationX=10

Action=Translate

Bewegt alle Kreise um 10 Einheiten in x-Richtung.

Parameter

TranslationX – Verschiebung in x-Richtung (Default 0)

TranslationY – Verschiebung in y-Richtung (Default 0)

Kommando „Mirror“

Spiegelt die selektierten Elemente.

Beispiel

Selection=All

MirrorAngle=90

MirrorOffset=-10

Action=Mirror

Spiegelt um Spiegelachse, die parallel zur y-Achse läuft (90 Grad) und die x-Achse bei 10 schneidet (offset -10). Biegungen werden nur in ihrer Position gespiegelt, nicht in ihrer Biegerichtung (siehe InvertBend).

Parameter

MirrorAngle – Winkel der Spiegelachse (Default 0)

MirrorOffset – Offset der Spiegelachse (Default 0)

Kommando „InvertBend“

Invertiert Biegungen.

Beispiel

Selection=All

Action=InvertBend

Invertiert alle Biegungen.

Kommando „PlaceText“ und „RetryPlaceText“

Das Kommando „PlaceText“ versucht einen Text zu erzeugen. Der Text wird so platziert, dass er innerhalb des Teils und nicht innerhalb einer Innenkontur liegt.

Das Kommando „RetryPlaceText“ bietet exakt die selbe Funktionalität wie „PlaceText“. Einziger Unterschied ist, dass das Kommando nur ausgeführt wird, falls das vorangegangene „PlaceText“ Kommando – und alle folgenden „RetryPlaceText“ Kommando – fehlgeschlagen ist.

Damit kann man z.B. versuchen einen Text zu platzieren und immer kleinere Textgrößen verwenden. Der erste erfolgreiche Versuch wird dann genommen und platziert einen Text, der so groß wie möglich ist und so klein wie nötig.

Man kann für den Text eine Referenzposition angeben und es wird versucht möglichst nahe an diese Position zu kommen. Die Referenzposition wird – pro Koordinate – als Verhältniszahl angegeben, mit 0 als kleinste Position und 1 als größte Position. Entsprechend erhält man mit 0.5 die Teilemitte.

Für diese Kommandos wird keine Selektion benötigt.

Beispiel

TextContent=Test

TextHeight=20

Action=PlaceText

TextHeight=10

Action=RetryPlaceText

TextHeight=5

Action=RetryPlaceText

Versucht einen Text mit dem Inhalt „Test“ mit Fontgröße 20 innerhalb des Teils zu platzieren. Falls dies fehlschlägt, wird ein erneuter Versuch mit Größe 10 versucht. Schlägt auch dieser fehl, wird ein letzter Versuch mit Textgröße 5 probiert.

Parameter

TextContent – Inhalt des Textes (Default leer – keine Textplatzierung)

TextHeight – Höhe des Textes (Default 10)

TextFontNr – Fontnummer des Textes (Default 1)

1 – iso.fnt

130 – isoprop.fnt

131 – bold.fnt

TextColor – Textfarbe (Default yellow)

TextHorizontal – horizontale Ausrichtung (Default true)

TextBorder – Randabstand zur Teilegrenze (Default 1)

TextPosXRatio – Verhältniszahl der Referenzposition in x-Richtung

TextPosYRatio – Verhältniszahl der Referenzposition in y-Richtung

Kommando „RenotchBends“

Erzeuge automatische Biegeklinkungen.

Auf Wunsch kann die Originalkontur behalten werden, allerdings erhält sie ein Attribut, dass sie nicht bearbeitet werden soll und wird rot eingefärbt.

Beispiel

RenotchLevel=1

RenotchSize=5

Action=RenotchBends

Erzeuge falls nötig Rundklinkungen mit einem Durchmesser 5.

Parameter

RenotchLevel – Klinkungsstrategie pro Klinkung (Default -1)

- 1 – keine Änderung
- 0 – Ecke schließen
- 1 – Kreisklinkung erzeugen
- 2 – Quadratklinkung erzeugen
- 3 – Dreiecksklinkung erzeugen
- 4 – Schlitzklinkung erzeugen

RenotchSize – Defaultgröße für Klinkungen (Default 5)

RenotchLocalFit – lokale Klinkungsstrategie (Default 0)

- 0 – fixe Größe (RenotchSize)
- 1 – Klinkung einpassen in Geometrie
- 2 – Geometrie mit Klinkung umhüllen

RenotchListFit – setze Listenstrategie (Default 3)

- 0 – Liste ignorieren
- 1 – nächst kleineren Eintrag verwenden (reduzieren)
- 2 – nächst größeren Eintrag verwenden (vergroessern)
- 3 – nächsten Eintrag verwenden (runden)

RenotchGlobalFit – übergreifende Klinkungsstrategie (Default 0)

- 0 – jede Klinkung einzeln behandeln
- 1 – kleinsten Wert für alle übernehmen
- 2 – größten Wert für alle übernehmen

RenotchAddSize – definiere Listeneintrag (Default leer)

Kann mehrfach gesetzt werden, um eine Liste zu erzeugen

VerboseMode – Soll Originalkontur erhalten bleiben (Default false)

Kommando „ReplaceSlits“

Ersetze Schlitze durch einfache mittige Schnittgeometrie.

Auf Wunsch kann die Originalkontur behalten werden, allerdings erhält sie ein Attribut, dass sie nicht bearbeitet werden soll und wird rot eingefärbt.

Beispiel

SlitWidth=10

SlitFlag=3

Action=ReplaceSlits

Ersetze alle Rechteck- und Langlochkonturen mit einer Breite von bis zu 10 durch eine einfache mittige Linie.

Parameter

SlitWidth – Breite der zu ersetzenden Schlitze (Default 0)

SlitFlag – Auswahl welche Schlitze ersetzt werden sollen (Default 0)

1 – ersetze rechteckige Schlitze

2 – ersetze Langlochschnitte

4 – ersetze anteilige (Langloch/Rechteck) Schlitze in der Außenkontur

3,5,6,7 – Kombination der oberen Auswahl (einfach addieren)

8 – beliebige Schlitze

VerboseMode – Soll Originalkontur erhalten bleiben (Default false)

Kommando „RenotchBendsAndReplaceSlits“

Führt die Kommandos „RenotchBends“ und „ReplaceSlits“ simultan durch.

Die Kombination der beiden Kommandos kann unter Umständen sinnvoll sein, wenn Schlitze an Außenkonturen und potentiell Biegeklinkungen sich schneiden.

Beispiel

SlitWidth=10

SlitFlag=4

RenotchLevel=4

Action=RenotchBendsAndReplaceSlits

Ersetze alle anteiligen Schlitze an der Außenkontur mit einer Breite von bis zu 10 durch eine einfache mittige Linie. Außerdem setze Schlitze an den Biegeklinkungen.

Parameter

Siehe „RenotchBends“ und „ReplaceSlits“

Kommando „SmoothBendZones“

Glätte den Randbereich von Biegezonem.

Beispiel

Action=SmoothBendZones

Glättet den Randbereich von Biegezonem.

Parameter

keine

Kommando „ExtractSheet“

Analysiere eine Zeichnung mit verschachtelter Tafel und extrahiere die Tafel und die Einzelteile auf der Tafel.

Beispiel

SheetName=Tafel.taf

SheetPartName=Teil

Action=ExtractSheet

Extrahiere die Tafel als Tafel.taf und die Einzelteile als Teil_01.geo, Teil_02.geo, ..

Parameter

SheetName – Name der Tafel (default „sheet.taf“)

SheetPartName – Name für die Teile (default „part“)

SheetMaterial – Rohmaterial der Tafel

SheetThickness – Dicke des Materials

SheetFormat – Format der Tafeldatei (JSON,Taf – Default Taf)

SheetFindTwins – Flag ob gleiche Teile gefunden werden sollen (default 1)

Kommando „ExtractMaterial“

Sucht in der Zeichnung nach Texten mit definierbaren Schlüsselwörtern und extrahiert aus diesem das Rohmaterial, Werkstoff und die Dicke des Teils. Die Texte werden anschließend gelöscht. Das Schlüsselwort und das gesuchte Datum können getrennt werden durch ein Gleich, ein Bindestrich oder ein Doppelpunkt.

Beispiel

RawMaterialKey=RawMaterial

MaterialKey=Material

ThicknessKey=Thickness

Action=ExtractMaterial

Extrahiere die Materialdaten aus Texten die z.B. folgenden Inhalt haben: „Material=1.4301“.

Parameter

RawMaterialKey – Schlüssel für Rohmaterial (default „RawMaterial“)

MaterialKey – Schlüssel für Werkstoff (default „Material“)

ThicknessKey – Schlüssel für Dicke (default „Thickness“)

Kommando „ExtractData“

Analysiere eine Zeichnung und extrahiert eine Liste von Daten. Die Daten werden in Form einer Komma-separierten Liste (csv-Format) ausgegeben.

Der Ausgabedateiname kann über den Parameter „-x“ definiert werden (Default output.csv).

Es gibt fest definierte Key-Wörter, die man ausgeben kann.

Manche Keys kommen mehrfach vor (z.B. Biegewinkel). Diese sind dann indiziert und man muss den Index mit angeben. So lautet der Key für den dritten Biegewinkel z.B. „bend_angle3“.

Bei indizierten Keys kann man statt einem Index auch das „#“ Zeichen angeben. Dieser wird dann durch einen Index ersetzt, den man beim Kommando definieren kann. Dazu muss man nach dem Kommandonamen einen minimalen Index und optional einen maximalen Index anfügen. Beide werden durch Komma getrennt. Falls minimaler und maximaler Index sich unterscheiden, erzeugt der Aufruf mehrere Ausgabezeilen.

Der minimale und maximale Index kann durch eine fixe Zahl, aber auch durch Keys definiert werden.

Es ist ebenfalls möglich einfach „#“ als Key anzugeben. In diesem Fall wird der Index selbst ausgegeben.

Im Gegensatz zu dem üblichen Vorgehen, werden alle Keys nach Ausführung des Kommandos gelöscht, so dass man einfacher neue Keys definieren kann.

Falls man einen zu hohen Index verwendet (für die gegebene Datei) wird ein leerer Eintrag ausgegeben.

Die Liste der verfügbaren Keys ist wie folgt (Beschreibung in Englisch):

- "part_width" - part size in x-direction
- "part_length" - part size in y-direction
- "part_area" - area of the part
- "part_units" - units of the part ("inch"/"metric")
- "part_thickness" - thickness of the part
- "part_scrap_outer" - outer scrap (box size minus outer contour area)
- "part_scrap" - scrap (box size minus part area)
- "part_seam_count" - count of found plausible seams
- "part_seam_length" - total length of found plausible seams
- "part_material" - material of the part
- "part_raw_material" - raw material of the part
- "part_id" - (Fab) id of the part
- "file_type" - file type
- "file_name_org" - original name of the file (if it is an imported file)
- "file_size_org" - original size of the file (if it is an imported file)
- "file_name" - name of the drawing
- "file_guid" – guid of file
- "file_laser_prepared" - flag if part is laser prepared
- "outer_contour_length" - length of outer contour
- "outer_contour_element_count" - number of elements in outer contour
- "outer_contour_type" - type of outer contour
(11-circle, 12-rounded rectangle, 13-rectangle, 14-thread, 15-oblong, 16-form, 17-text, 18-arbitrary)
- "bendedge_count" - number of bends in part
- "bend_orientation_span" - maximum difference of bend orientations
- "text_count" - number of text elements in part
- "foil_removal_count" - number of foil removals in part
- "contour_count" - total number of contours in part
- "total_inner_contour_length" - length of all inner contours summed up

"total_bendedge_length" - length of all bends summed up
"bendedge_tec_count" - number of bends with tec data in part
"warped_contour_count"- number of contours in bend zones

Contours are grouped if they are identical and separated into standard, arbitrary and form contours.

"standard_count" - number of different standard contour groups found
"arbitrary_count" - number of different arbitrary contour groups found
"form_count" - number of form contours in part
"form_tec_count" - number of form contours with tool id in part

All following entries are followed by an index to specify a bend
(e.g. "bend_angle1" for bend angle of first bend)

"bend_mfg_radius" - manufacturing radius of bend
"bend_angle" - angle of bend
"bend_factor" - bend factor of bend
"bend_type" - type of bend (0-normal, 1-hem, 2-radius, 4-flat, 5-Z, 6-special, 7.double wall, 8-swing, 9-auxiliary)
"bend_method" - method of bend (0-air, 1-coin, 2-hem with gap, 3-hem without gap, 4-swing
"bend_technique" - technique of bend (0-normal, 1-pre bend, 4-bumping,)
"bend_upper_tool" - upper tool of bend
"bend_lower_tool" - lower tool of bend
"bend_orientation" - angle of bend line on plane
"bendedge_length" - length of bend

All following entries are followed by an index to specify a standard, arbitrary or form contour.
The contours are sorted by count and only the top 9 are exported.
(e.g. standard_hole_area7 for the area of the 7th most common standard hole)

"standard_hole_area", "arbitrary_hole_area", "form_hole_area" - area of contour
"standard_hole_length", "arbitrary_hole_length", "form_hole_length" - length of contour
"standard_hole_count", "arbitrary_hole_count", "form_hole_count" - number of identical holes
"standard_hole_distance", "arbitrary_hole_distance", "form_hole_distance" - average distance between holes (missing if just one hole)
"standard_hole_is_mark", "arbitrary_hole_is_mark" - "true" if contour is marked
"arbitrary_hole_element_count", "form_hole_element_count" - number of elements in contour
"standard_hole_type" - type of standard contour (see outer contour)

Beispiel

```
AddDataKey=bendedge_count
Action=ExtractData
AddDataKey=bend_angle1
AddDataKey=bend_angle2
AddDataKey=bend_angle3
AppendData=true
Action=ExtractData
AddDataKey=#
AddDataKey=bend_angle#
AddDataKey=bend_factor#
Action=ExtractData,1,bendedge_count
```

Würde für ein Teil mit 4 Biegungen vielleicht folgende Ausgabe erzeugen:

```
bendedge_count
4
bend_angle1, bend_angle2, bend_angle3
90,60,45
#,bend_angle#,bend_factor#
1,90,-1.92
2,60,-1.22
3,45,-0.88
4,-90,-1.92
```

Parameter

AddDataKey – Füge ein neues Datum in Liste ein, welches extrahiert werden soll.
ShowDataTitle – Soll eine Überschriftenzeile ausgegeben werden (Default – true)
AppendData – Sollen Daten ans Ende der Datei angefügt werden.
 true – Daten werden angefügt
 false – Datei wird überschrieben (Default)

Zeichnungsanalyse

Beim Konvertieren einer Zeichnung mit tc_conv kann man diese analysieren und das Ergebnis in einer Protokolldatei ablegen. Dies geschieht nicht über eine Kommandozeile, sondern über andere Kommandozeilenoptionen (-m und -e).

Bei dieser Prüfung wird untersucht, ob eine Zeichnung kleine Elemente enthält. Die Toleranzen für diese Prüfung können über die Kommandozeile verändert werden. Als Default wird eine Meldung generiert, wenn mindestens 100 Elemente kleiner als 2mm sind oder mindestens 1 Element kleiner als 0.1mm ist.

Beispiel

```
CheckSmallTolerance=1  
CheckSmallCount=50  
CheckTinyTolerance=0.1  
CheckTinyCount=5
```

Melde, wenn 50 Elemente kleiner als 1mm sind oder 5 Elemente kleiner als 0.1mm.

Parameter

CheckSmallTolerance – Elemente kleiner als dieser Wert gelten als klein

CheckSmallCount – Melde ab dieser Anzahl an kleinen Elementen

CheckTinyTolerance – Elemente kleiner als dieser Wert gelten als winzig

CheckTinyCount – Melde ab dieser Anzahl an winzigen Elementen

Kommando „ShiftTab“

Verschiebt eine Lasche, um Verschnitt und die Schweißnahtlänge zu reduzieren. Die Differenz zwischen der alten und neuen Schweißnahtlänge bzw. dem Verschnitt wird jeweils mit einem Faktor multipliziert. Die Summe muss jeweils positiv sein.

Beispiel

```
ShiftTabScrapWeight = 0  
Action=ShiftTab
```

Verschiebe Lasche, falls die Schweißnaht dadurch verkürzt wird. Der Verschnitt wird ignoriert.

Parameter

ShiftTabScrapWeight – Faktor für Verschnittergebnis (Default 1)

ShiftTabSeamWeight – Faktor für Nahtlänge (Default 1)

Kommando „ReplaceTemplates“

Ersetzt Konturmuster durch einen Platzhaltertext.

Beispiel

```
TemplateScalable=false  
AddTemplate = c:\Vlgs\Muster.vlg  
ReplaceThreads = none  
Action=ReplaceTemplates
```

Setzt die Vorlage "Muster.vlg" und ersetzt geometrische Instanzen von dieser durch den Text „Muster“.

Parameter

TemplateScalable – Flag ob Muster skaliert werden darf (Default false)

AddTemplate – Name der Vorlage die ersetzt wird (darf mehrfach gerufen werden)

VerboseMode – Soll Originalkontur erhalten bleiben (Default false)

ReplaceTemplates – Flag ob Vorlagen ersetzt werden sollen (Default true)

ReplaceThreads – Flag welche Gewinde ersetzt werden sollen (none, metric, imperial)
(Default none)