

Erstellen eines parametrisches 3D-Gewinde in MegaCAD 2014

Mit der Parametrik lassen sich häufig verwendete Teile, die in unterschiedlichen Größen aber immer in der gleichen Form gegeben sind, leicht aus einer Datei laden und schnell anpassen. Eine Mutter hat z.B. immer 6 Ecken, nur die Kantenlänge ändert sich. Alle Normteile eignen sich gut um diese im Vorhinein zu zeichnen und zu Parametrisieren.

Hat man mehrere Normteile schon parametrisiert muss man diese nur noch öffnen, die gebrauchte Größe an zu geben und in die Zeichnung ein zu fügen. Dazu sind Baugruppen gedacht und gut geeignet. Weshalb wird an passender Stelle erklärt.

In diesem Tutorial gehe nehme ich als Beispiel das metrische ISO-Gewinde. Es wird mit einem Gewindeauslauf und einer Fase am unteren Ende versehen.

Nun benötigt man alle Tabellen, die unser Teil beschreiben. In diesem Fall ist es die DIN 13-1 für das Gewinde und die DIN 76-1 für Gewindeausläufe.

Bild Tabellenbuch Seite 85; Bild Tabellenbuch Seite 190

Bevor man anfängt muss man sich im Klaren sein welche Werte müssen vom Benutzer eingegeben werden bzw. sollten von ihm änderbar sein. Bei dem Gewinde wären das der Gewindedurchmesser und die Gewindelänge. Im optimalen Fall sieht der Benutzer auch nicht mehr als diese Werte. Danach macht man sich Gedanken, wie man das Teil am banalsten aufbaut. Hier ist es eine Helix für die Gewindeflanken, ein Rotationskörper für den Gewindeauslauf, ein Zylinder als Kern und ein weiterer Rotationskörper für die Fase am Ende des Gewindes.

Man baut eine Parametrik direkt in eine Baugruppe rein, da diese noch eine Besonderheit aufweist, welche man bei Zeichnungen nicht hat.

1. Parametrische Skizzen

Als erstes Fangen wir mit dem wichtigsten Objekt an, der Helix die das Gewinde darstellt. *An dieser Stelle sei erwähnt, dass wenn eine Helix keine Zwischenräume haben soll ein besseres Ergebnis erzielt werden kann, wenn man die auf 2 aufteilt.*

Zudem sollte man Radien und grade Flächen bei einer Helix nicht Teilen, daraus ergibt sich, dass wir 2 Basisprofile zeichnen müssen:

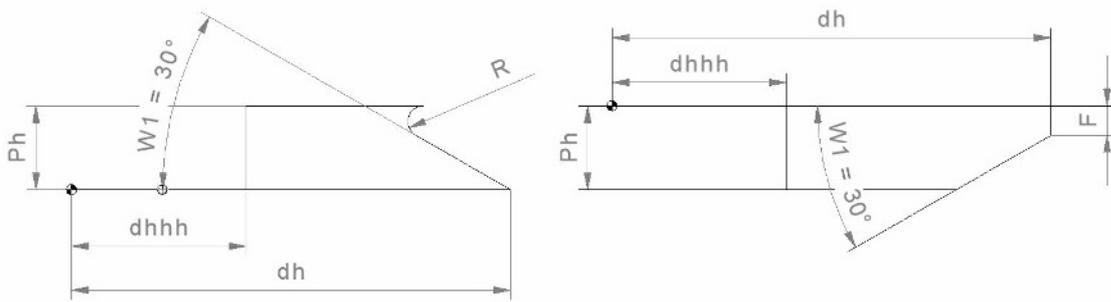


Der Bezugspunkt ist in diesem Fall in der Mitte, damit wir nicht groß umrechnen müssen.

Es empfiehlt sich bei der Parametrik möglichst immer das kleinste Abmaß als erste Skizze zu nehmen, da es gelegentlich passiert, dass MegaCAD beim Verkleinern der Skizze Verknüpfungen zwischen den Zeichenelementen verliert.

Nun müssen wir die Skizzen parametrisieren. Dazu erzeugen wir aus den Skizzen jeweils eine Variante.

Dann werden die benötigten Maße durch klicken auf die entsprechende Linie gesetzt.



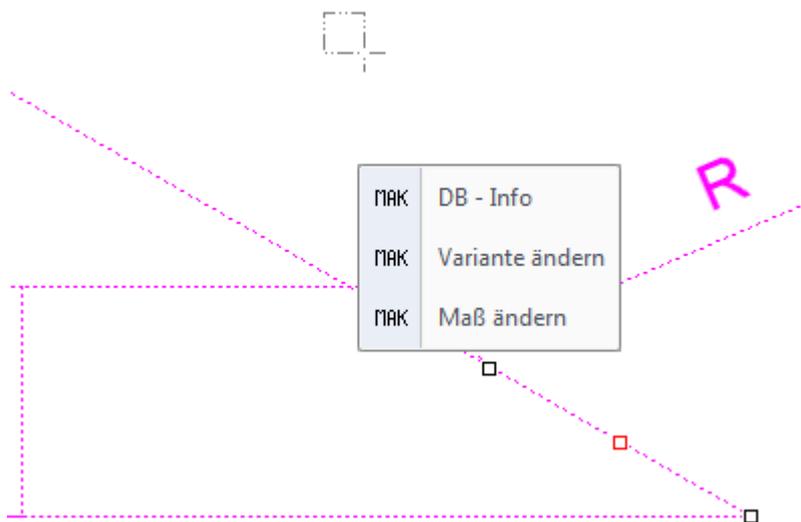
Dabei sollte man direkt bei den Maßen den Namen ändern entsprechend der Norm nach der man zeichnet. Das erhöht die Übersichtlichkeit und man muss nachher nicht mehr ständig nachsehen welche Variable welches Maß in der Norm ist.

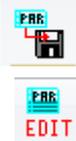
Zusatzzeichen sind auch der Übersichtlichkeit zuträglich. Hier eine kleine Liste nach der ich benenne (# = Bezeichnung nach Norm):

- #s = Wert durch Variable auf Richtigkeit überprüft (Verhindert das die Parametrik durch falsche Eingaben beschädigt wird)
- #h = Wert durch 2 um z.B. aus dem Durchmesser den Radius zu bekommen
- #d = Wert mal 2 um z.B. aus dem Radius den Durchmesser zu bekommen
- list# = Variablen gesteuerte Liste (Nur Sinnvoll bei Wertebereichen und nicht direkten Tabellen)
- #2 = Maß wird mehr als 1 mal in der Skizze verwendet
- Wert# = Für den Benutzer wichtige Information

2. Listen in MegaPara

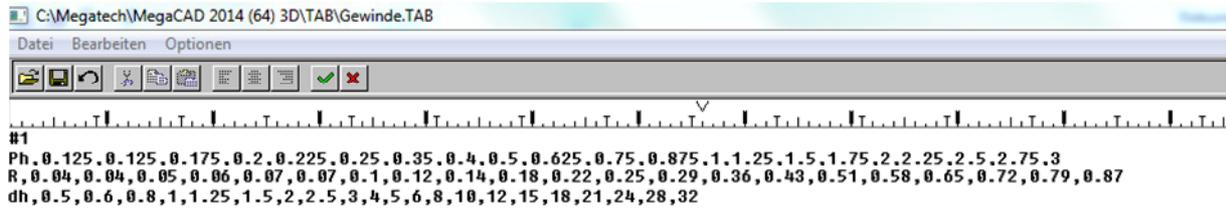
Listen lassen sich leichter in MegaPara erstellen, darum wechseln wir jetzt zu MegaPara indem wir auf eine der Zeichnungen klicken, dort „Baugruppe“ auswählen und danach „Variante ändern“.





Es öffnet sich Megapara. Dort wählen wir auf der Startseite „Maße in Tabelle schreiben“ und sagen ihm wie die Tabelle heißen soll. Dann öffnen wir die Tabelle zum Bearbeiten indem wir auf „Maße in Tabelle ändern“ klicken.

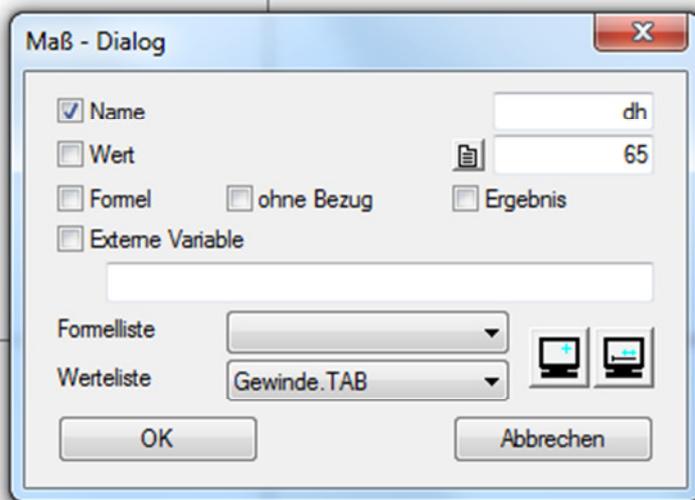
Die Tabelle die sich jetzt öffnet ist zwar nicht ganz so schön wie eine Excel-Tabelle aber um



Lang wird die Tabelle nicht, da wir nur Werte eintragen müssen, die wir nicht aus den vom Benutzer gegebenen Werten errechnen können. Das wären dh als Eingang, Ph und R. dhhh wird aus ds berechnet und W1 ist immer 30° somit müssen diese nicht mit in die Liste.

Dann speichern wir die Tabelle in dem wir auf das grüne Häkchen klicken.

Wir müssen noch die Tabelle bei unseren Maßen auswählen. Dazu öffnen wir das Wertemenü eines Maßes, das in der Tabelle vorkommt und wählen unter „Werteliste“ unsere Tabelle aus.



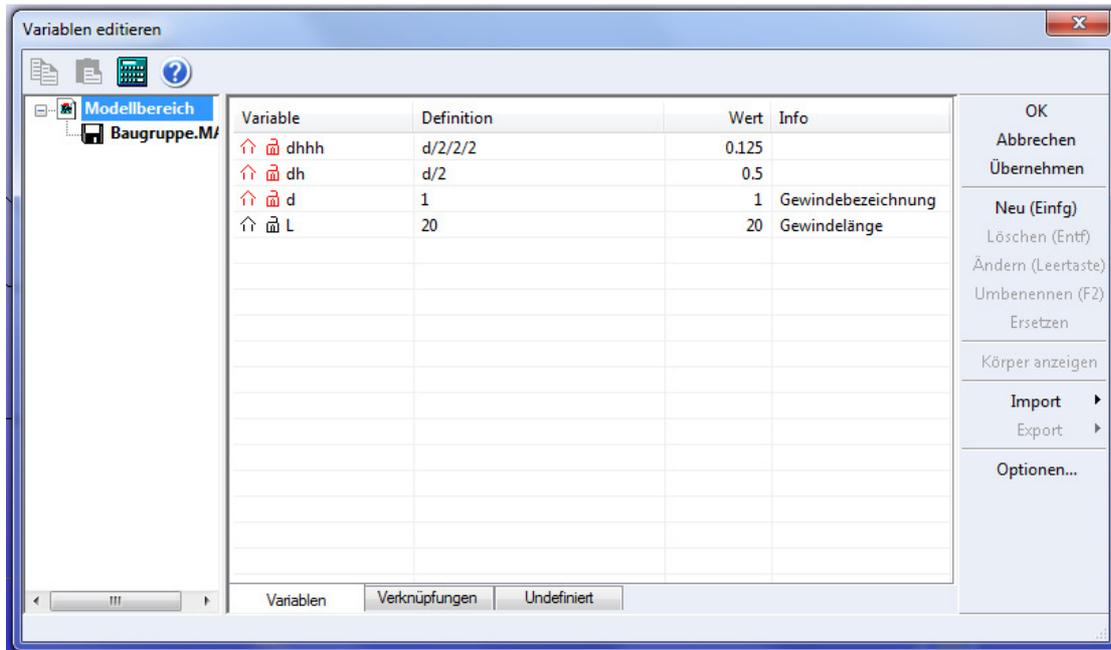
Alle restlichen Werte, die in der Tabelle vorkommen, werden automatisch von MegaPara Verknüpft.

Jetzt schließen wir MegaPara. Die Parametrisierung muss nicht gespeichert werden aber zum weiteren gebrauch muss sie an MegaCad übergeben werden.

Das muss jetzt nur noch für die 2 Skizze wiederholt werden, wobei auch die Tabelle der ersten Skizze geladen werden kann.

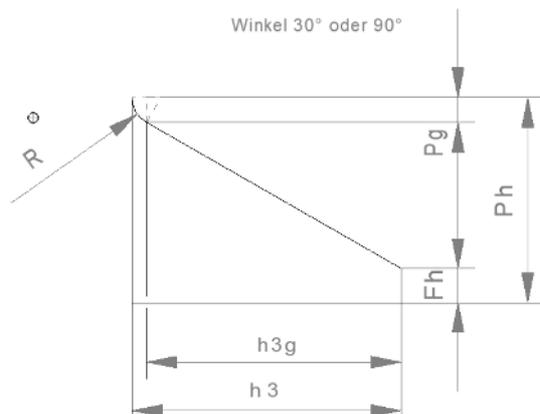
Die Variablen mit dem Pfeil davor tauchen jetzt auch im Modellbereich auf und können nur noch von da auf geändert werden.

Dort tragen wir dann unter Definition dann die Funktion ein mit der diese Variable berechnet wird. Da wo eine Variable benötigt wird, wird der Name der Variable einfach eingetragen. Man kann ganz normal +, -, / und * verwenden wie in Excel. Man muss und darf nicht ein = am Anfang setzen.



Fehlt uns noch die 2 Skizze. Da fällt uns auf, dass nicht nur dh und dhhh berechnet werden müssen sondern auch F. Um F berechnen zu können müssen wir auf die Trigonometrie zurückgreifen. Zum Glück bietet uns MegaCad hier schon einige Funktionen an um dies zu tun (Liste aller Funktionen im Anhang).

Um nun F heraus zu finden empfiehlt es sich das Problem als erstes auf zu skizzieren. Danach löst man da kleinere Unterlösungen heraus, die dann zu einer großen Formel zusammengefasst werden.

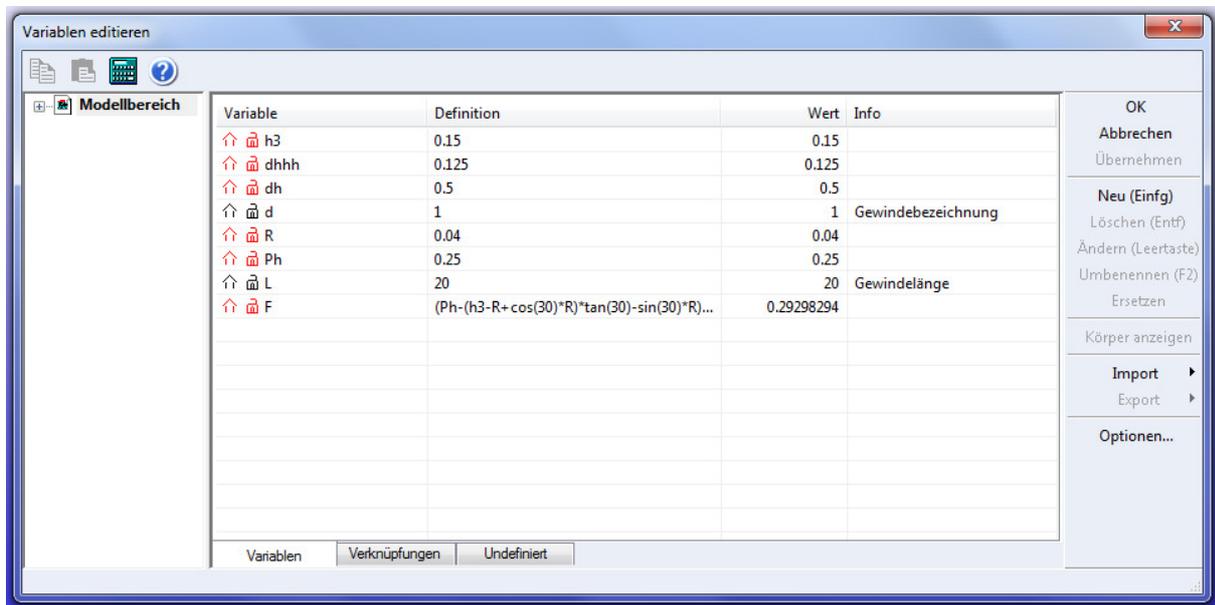


$$P_g = \sin(30) * R$$

$$h_{3g} = h_3 - R + \cos(30) * R$$

$$F = (P_h - (h_3 - R + \cos(30) * R) * \tan(30) - \sin(30) * R) * 2$$

Wir holen uns jetzt also F aus der Baugruppe hoch und fügen unsere Formel dafür ein. MegaCad erklärt uns, dass es die Variablen h3, R und Ph nicht existieren und daher nicht verwendet werden können. Um dies zu beheben erzeugen wir diese Variablen eben neu und fügen die Formel erneut ein.



Da die Werte R und Ph jetzt auch über die Variablen gesteuert werden benötigen wir die Liste nicht mehr. Also beide Baugruppen öffnen und die Tabelle wieder aus den Maßeinstellungen entfernen.

Sehen wir uns jetzt die Gewindedarstellung im Tabellenbuch an, fallen uns die Formeln neben der Darstellung auf. Daraus lässt sich entnehmen, dass sich alle Werte aus d und P berechnen lassen. Daher benötigen wir nur eine Liste, die die Werte P einem bestimmten Wert d zuweisen.

Um dies zu gewerkstelligen müssen wir eine Abfrage einbauen. In MegaCad sieht das wie gefolgt aus:

$$(d==1)*0.25+(d==1.6)*0.35$$

wenn d gleich 1 dann 0,25 oder d gleich 1,6 dann 0,35

Das bedeutet für jeden Wert in der Liste benötigen wir den Block (d==?)*?+.

Da kommt am Ende dann der P-Wert für den entsprechenden Durchmesser raus. Und weil es eine Liste ist bekommt es am Anfang das Kürzel list.

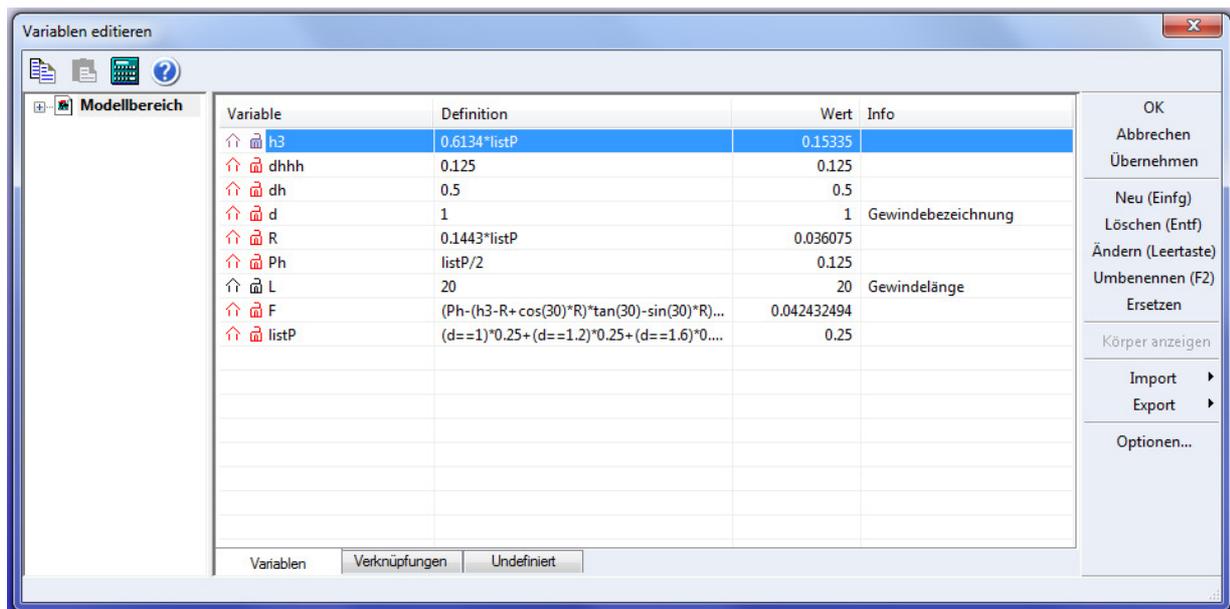
Abfrage-operationen:

== : ist gleich

<= : ist kleiner oder gleich

=> : ist größer oder gleich

<> : ist ungleich



Solche Listen bringen aber einen Nachteil mit. Wenn der Benutzer einen Wert eingibt, der nicht in der Liste definiert ist gibt die Liste 0 aus, aber alle Zeichenelmente die eine Länge von 0 haben werden gelöscht, wodurch auch die Parametrik kaputt geht. Darum müssen wir noch eine Sicherheitsabfrage einbauen, die im Fall, dass d nicht in der Liste ist, einen Standardwert einsetzt.

Hier kommt ds ins Spiel. ds ist ähnlich wie listP eine Liste, mit einem entscheidenden Unterschied. Mit der Klammer (d==?) wird nur noch überprüft ob der Wert stimmt. Tut er das kommt bei der Klammer 1 raus. Somit lässt sich sagen: wenn alle Klammern gleich 0 dann wurde ein falscher Wert eingegeben.

$$(((d==1)+(d==1.2)+(d==1.6))=0)*10+(((d==1)+(d==1.2)+(d==1.6))=0)*d$$

Für die, die zu faul sind die Klammern doppelt ein zu geben können den ersten Teil in eine zweite Variable einsetzen

$$dss = (d==1)+(d==1.2)+(d==1.6)$$

$$ds = (dss==0)*1+(dss<>0)*d$$

Nachdem man diese Sicherheitsabfrage eingebaut hat muss man nachher auch überall wo der ursprüngliche Wert verwendet wurde ausgetauscht werden.

Variable	Definition	Wert	Info
listP	$(ds=1)*0.25+(ds=1.2)*0.25+(ds=1.6)...$	0.25	
h3	$0.6134*listP$	0.15335	
dss	$(d=1)+(d=1.2)+(d=1.6)+(d=2)+(d...$	1	
ds	$(dss=0)*10+(dss<0)*d$	1	
dhhh	$ds/2/2/2$	0.125	
dh	$ds/2$	0.5	
d	1	1	Gewindebezeichnung
R	$0.1443*listP$	0.036075	
Ph	$listP/2$	0.125	
L	20	20	Gewindelänge
F	$(Ph-(h3-R+\cos(30)*R)*\tan(30)-\sin(30)*R)...$	0.042432494	

Jetzt sind wir so weit, dass wir bei d irgendwelche Werte eintragen können und zusehen wie sich unsere Skizze verändert.

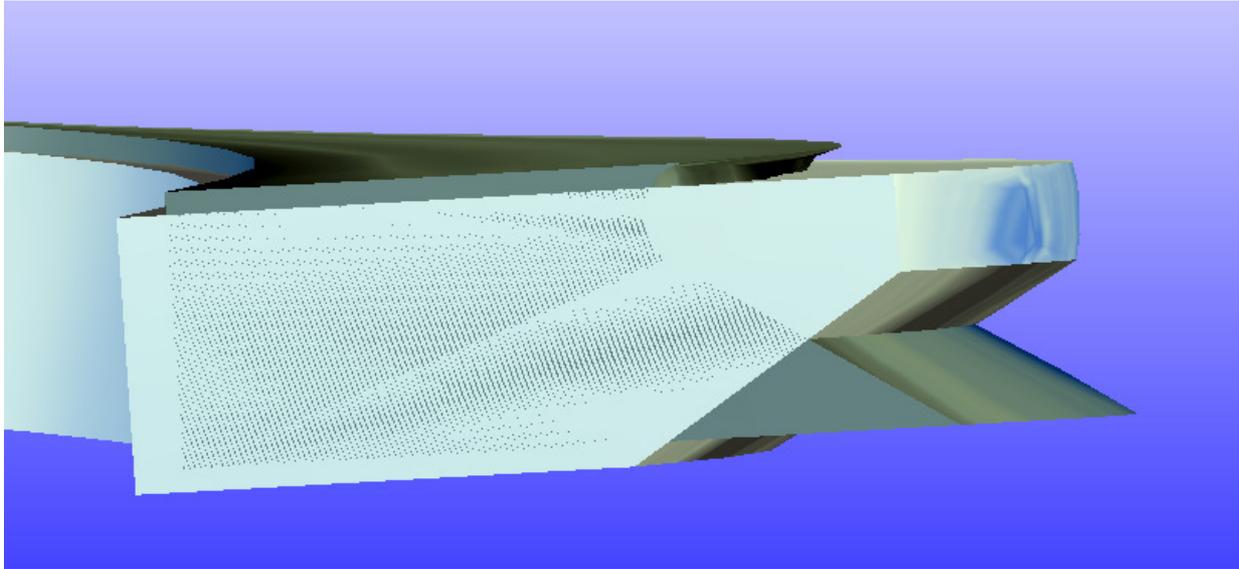
Es ist wichtig zwischendurch immer mal wieder nach zu sehen ob die Parametrik noch funktioniert, da man sich sonst noch sehr viel Arbeit macht um zu sehen, dass es nachher nicht geht. Zudem sollte man immer mal zwischen speichern. Auch ruhig V1, V2.... Machen, da Fehler die erst spät auftauchen ihre Wurzel sehr weit vorne haben kann. Zudem ist es immer wieder nett, wenn man nach den wilden testen mit den Variablen noch die erste funktionierende Form vorliegen hat wenn die neue Formel nicht so ganz will.

4. Verknüpfungen von 3D-Objekten zueinander

Ich empfehle hier die alte Helixfunktion, da wir keine wilden Besonderheiten in dem gleichmäßigen Gewinde haben und da die Helix(adv.) zu unübersichtlich ist, wenn man nur Höhe und Steigung eingeben will

 Wir haben jetzt unsere Basisskizzen fertig. Somit bleibt nur noch eins: Helixfunktion und hoch damit. Eben schnell noch ein wenig mit den Eingangswerten rum spielen um zu sehen ob es noch geht und Ups! Die Helixen verschieben sich immer ineinander, da sich die Helix sich nicht merken kann wo ihr Bezugspunkt ist und den immer wieder in den Flächenschwerpunkt verschiebt.

Dadurch dass der Flächenschwerpunkt genommen wird ist es auch schwer den richtigen Radius heraus zu finden. Um diesen genau zu bestimmen muss man für jeden Durchmesser das Basisprofil anpassen und dann den Wert vom Bezugspunkt zum Flächenschwerpunkt in eine Liste einschreiben. Was das bringt wird in Kapitel 7 erklärt.



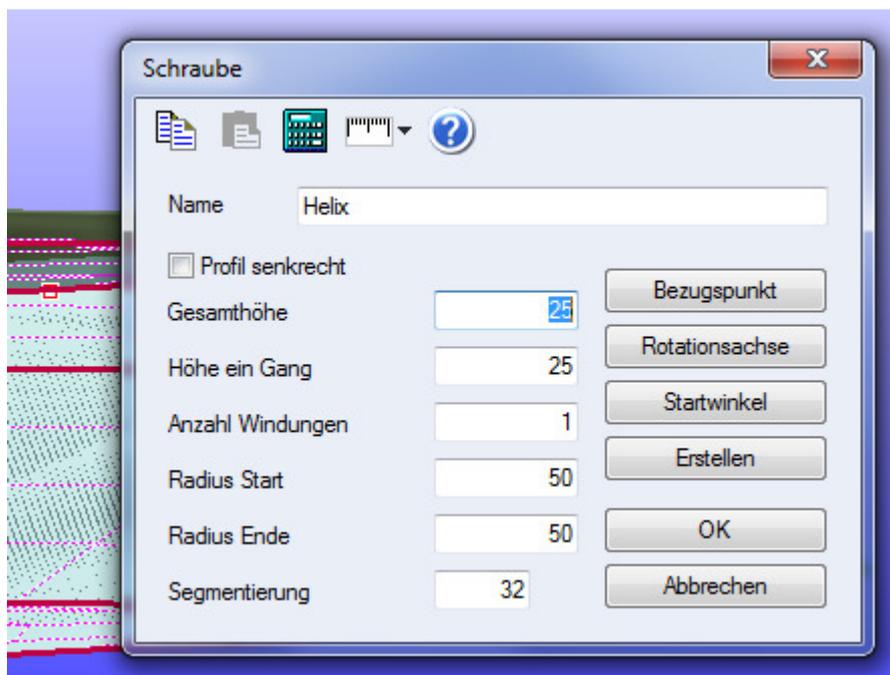
kann Ecken von Flächen zu anderen Ecken einer anderen Fläche zueinander abhängig machen bzw die eine von der anderen abhängig.



Die Funktion die wir suchen heißt in MegaCad „Verknüpfungen“ und wird durch zwei Büroklammern dargestellt.

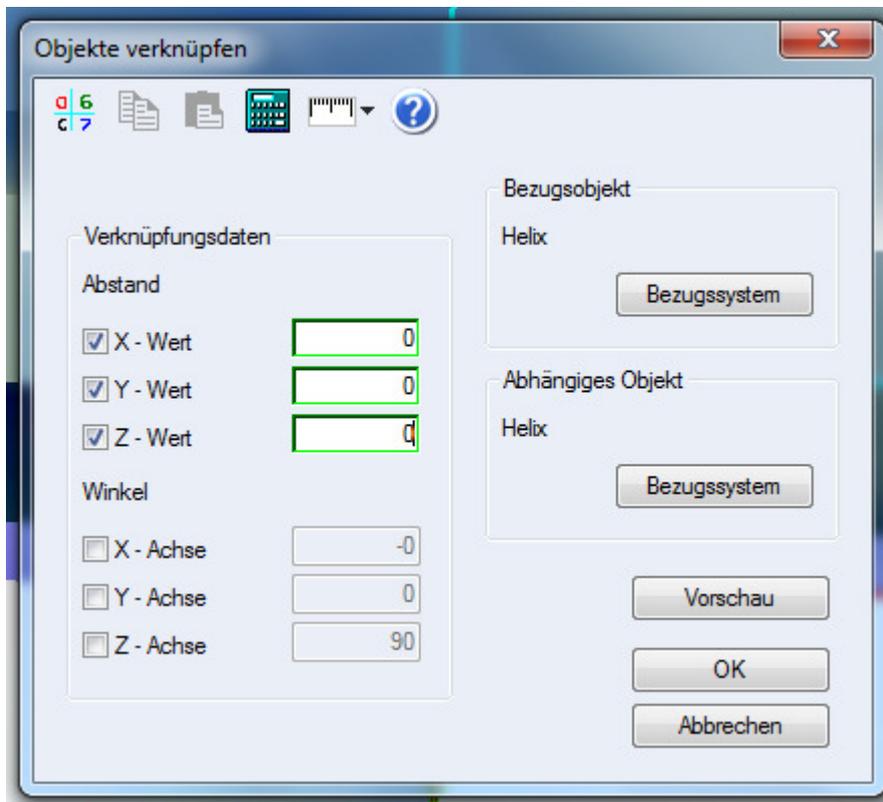
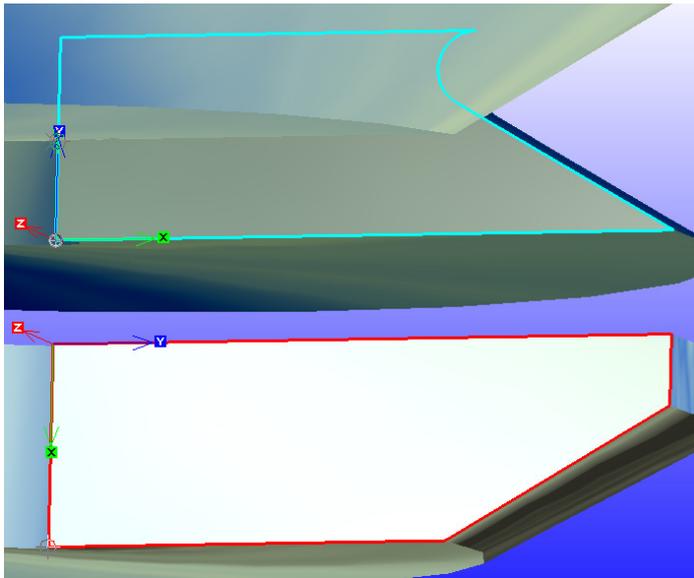
Diese kann aber nur auf graden Flächen agieren was für uns bedeutet, dass wir bei der Helixfunktion das Häkchen bei „Profil senkrecht“ wegnehmen müssen. Damit ist die Start- und die Endfläche wieder Parallel zur Z-Achse.

Das Basisprofil muss dafür auf der ZX-Ebene liegen.



Wir sollten zudem noch die Helix, die oben sein soll so weit nach oben verschieben, bis beide Flächen voneinander getrennt sind.

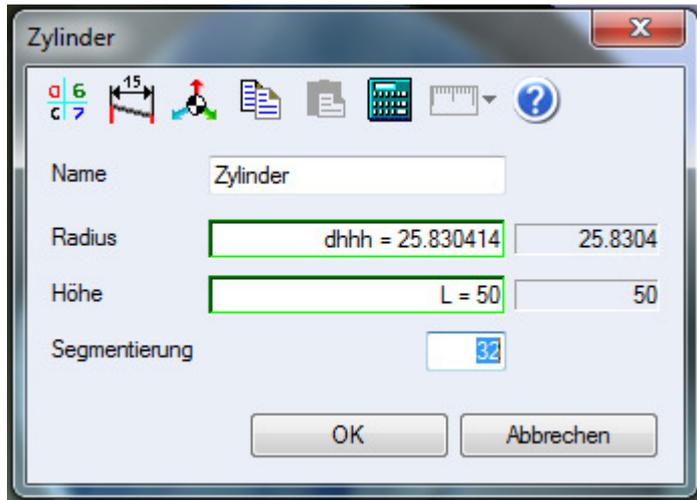
Dann die untere Fläche auswählen und den Punkt, der mit dem anderen dann identisch sein soll, dann die andere Fläche und den zweiten Punkt. Dann in dem Fenster, dass sich öffnet alle Koordinaten auf 0 setzten und die Winkel deaktivieren durch entfernen des Häkchens davor.



Siehe da die Helixen sind fest miteinander verbunden und bleiben dies auch, selbst wenn man die Werte ändert.

5. Setzen von Variablen in 3D-Objekten

Das Gewinde sieht soweit schon gut aus. Nur gibt es sehr wenige Gewinde mit einem Loch in der Mitte. Daher muss noch ein Kern her. Dieser soll sich auch direkt mit den Variablen mit ändern. Dazu ziehen wir in der Mitte des Gewindes einen Zylinder ganz normal auf. In dem Fenster, in dem man die Maße des Zylinders sieht kann man einfach die Namen der Variablen hinein schreiben und der Zylinder zieht sich automatisch die richtigen Maße.



6. Fase und Gewindeauslauf

Jetzt kommt der Übungsteil. Die Fase entspricht $45^\circ \times P$ und die Angaben zum Gewindeauslauf sind im Tabellenbuch zu finden. Dabei sollte man aufpassen, dass das Gewinde stumpf anfängt und die Helixen höher sein müssen, da sie ja unten abgeschnitten werden und oben in den Gewindeauslauf gehen muss. Die Gewindelänge geht von der graden Fläche unten bis zum Anfang des Gewindeauslaufes.

7. Userinterface und Hilfwerte

Wie am Anfang schon angesprochen sollte der Benutzer möglichst wenig von den Formeln sehen, die mittlerweile eine ganze Menge sind und nur zum Suchen der Eingabevariablen auffordern. Um zu erreichen, dass der Benutzer nur die Eingabevariablen zu Gesicht bekommt muss man erst die Baugruppe speichern.

ACHTUNG! Baugruppe nicht schließen sondern mit einer zweiten Instance von MegaCad überprüfen ob auch alle Variablen mit gespeichert wurden. Sollten diese nicht gespeichert worden sein muss man diese jetzt per Copy-Paste von der alten Zeichnung in die neue kopiert werden. Dann kann man die alte Zeichnung schließen und die neue nach dem Speichern auch.

Jetzt öffnen wir eine leere Baugruppe und fügen über Datei → Laden → Baugruppe einfügen die grade gespeicherte Baugruppe ein. Dann gehen wir auf Variablen und setzen vor d und L den Pfeil und schon sind in dem Modellbereich nur noch die Eingaben für den Benutzer.

Wie in unserem Beispiel, wo eine Funktion keine Variablen nutzen kann und den Bezugspunkt immer wieder verschiebt wird der Benutzer nachträglich noch aufgefordert an den entsprechenden Funktionen die Werte zu ändern. Dazu muss er von der Parametrik vorgegeben bekommen welche Werte er ein zu geben hat.

Dafür werden bei den Benutzereingaben noch die Wert#-Variablen eingesetzt.

In unserem Fall sind das die neue Länge, der neue Radius und die Steigung der Helixen. Diese werden genauso wie listP oder durch eine Formel aufgebaut.

Dadurch dass die beiden Basisprofile unterschiedlich sind haben beide unterschiedliche Helixradiendies kann man vereinfachen, in dem man die Teilung der Helix anders setzt, so dass man das Basisprofil spiegeln kann.

8. Schlusswort

Warum ein ganzes Tutorial, wenn der Gewindebolzen doch schon existiert und einsatzbereit ist? Dafür gibt es mehrere Gründe und dazu zählen nicht weder soziale Menschenfreundlichkeit oder meine Gier nach Anerkennung und Ehre sondern ganz banal meine Faulheit. Ich könnte das Wissen für mich behalten und horten wie Gollum doch wurde das auch bedeuten, dass jedes Mal wenn jemand einen Ring braucht ich schufteln muss. Stattdessen werfe ich mein Wissen in die Welt hinaus in der Hoffnung das ein oder andere zurück zu bekommen (Menschen geben einem mehr Geld wenn sie glauben man würde das nur aus Nettigkeit tun. Das habe ich damals als kleiner Junge beim Schnee schaufeln bei den Nachbarn gelernt.). Ich jedenfalls hoffe dass noch mindestens das Innengewinde mit Gewindebohrung, Gewindeauslauf und Fase bereit zum Dividieren in diesem Forum auftaucht. Ein Trapezgewinde wäre auch cool oder andere Normteile.

Warum im letzten Kapitel (7) keine Bilder mehr sind? Ich doof habe aus Versehen die Zeichnung mit den ganzen Variablen, Formeln und Listen gelöscht und hatte keine Lust mehr diese alle neu zu machen. Daher entschied ich das Worte Erklärung genug sind.

Und ganz wichtig und zu Letzt: Die Variablenbenennung obliegt nur bei euch. Die hier aufgeführte Weise ist nur ein Beispiel wie man das ganze übersichtlich machen kann.

9. Anhang (Originaltext von ww3.cad.de aus dem MegaCad-forum, habe den Thread nicht gefunden)

Verwenden von Formeln in MegaCAD

In den Eingabefeldern können sowohl Zahlen als auch Formeln verwendet werden. Dem Anwender stehen die dazu die

Funktionen: "sin", "cos", "tan", "asin", "acos", "atan", "sqrt", "log10", "log", "exp", "abs", sowie die Operatoren "<";

>", "==" , "<>", "<=", ">=", "+", "-", "*", "/" und "^" zur Verfügung.

Im Zusammenhang mit der 3D-Parametrik kann der Anwender Variablen definieren und auch diese in Formeln verwenden.

Einige Schlüsselwörter sind reserviert und stehen dem Anwender als Variablennamen nicht zur Verfügung:

"pow", "ln", "exp", "sinh", "cosh", "tanh", "abs", "atan2", "div", "mul", "_hypot", "hypot", "and", "or", "not", "nand",

"nor", "xor", "and_eq", "or_eq", "xor_eq", "true", "false".

Trigonometrische Funktionen

Hinweis: Bei der Verwendung von Variablen in Verbindung mit trigonometrischen Funktionen sollte die Einstellung

von 360 Grad im Vollkreis nicht verändert werden. Bei der Beschreibung der Funktionen wird die Standardeinstellung

vorausgesetzt.

sin

Liefert den Sinus einer Zahl.

Syntax:

`sin(Zahl)`

Zahl ist der im Gradmaß angegebene Winkel, dessen Sinus Sie berechnen möchten.

Beispiele:

`sin(180)` ergibt 0. Der Sinus von 180 Grad ist 0.

`sin(90)` ergibt 1.

`sin(30)` ergibt 0.5.

cos

Liefert den Kosinus einer Zahl.

Syntax:

`cos(Zahl)`

Zahl ist der im Gradmaß angegebene Winkel, dessen Kosinus Sie berechnen möchten.

Beispiele:

`cos(90)` ergibt 0.

`cos(60)` ergibt 0.5, den Kosinus von 60 Grad.

tan

Liefert den Tangens einer Zahl.

Syntax:

`tan(Zahl)`

Zahl ist der im Gradmaß angegebene Winkel, dessen Tangens Sie berechnen möchten.

Beispiele:

`tan(0)` ergibt 0.

`tan(45)` ergibt 1.

Verwenden von mathematische Funktionen in MegaCAD 1 Von 6

asin

Liefert den Arkussinus einer Zahl. Der Arkussinus einer Zahl ist der Winkel, dessen Sinus die Zahl ist. Der als Ergebnis

gelieferte Winkel wird im Gradmaß mit einem Wert zwischen -90 und 90 ausgegeben.

Syntax:

`asin(Zahl)`

Zahl ist der Sinus des Winkels, den Sie berechnen möchten, und muß im Bereich -1 bis 1 liegen.

Beispiele:

`asin(0.5)` ergibt 30.

`asin(-0,5)` ergibt -30.

`asin(1)` ergibt 90.

acos

Liefert den Arkuskosinus einer Zahl. Der Arkuskosinus einer Zahl ist der Winkel, dessen Kosinus die Zahl ist. Der

Ergebniswinkel wird im Gradmaß im Wertebereich von 0 bis 180 angegeben.

Syntax:

`acos(Zahl)`

Zahl ist der Kosinus des berechneten Winkels und muß zwischen -1 und 1 liegen.

Beispiele:

`acos(-0.5)` ergibt 120.

`acos(0.5)` ergibt 60.

atan

Liefert den Arkustangens einer Zahl. Der Arkustangens einer Zahl ist der Winkel, dessen Tangens die Zahl ist. Der als

Ergebnis gelieferte Winkel wird im Gradmaß im Wertebereich zwischen -90 und 90 ausgegeben.

Syntax:

`atan(Zahl)`

Zahl ist der Tangens des Winkels, den Sie berechnen möchten.

Beispiele:

`atan(1)` ergibt 45.

`atan(-1)` ergibt -45

Verwenden von mathematische Funktionen in MegaCAD 2 Von 6

Mathematische Funktionen

sqrt

Liefert die Quadratwurzel einer Zahl.

Syntax:

`sqrt(Zahl)`

Zahl ist die Zahl, deren Quadratwurzel Sie berechnen möchten. Zahl darf nicht negativ sein!

Beispiel:

$\sqrt{16}$ ergibt 4.

log10

Liefert den Logarithmus einer Zahl zur Basis 10.

Syntax:

$\log_{10}(\text{Zahl})$

Zahl ist die positive reelle Zahl, deren Logarithmus zur Basis 10 Sie berechnen möchten.

Beispiele:

$\log_{10}(86)$ ergibt 1,9344985.

$\log_{10}(10)$ ergibt 1.

$\log_{10}(1E5)$ ergibt 5.

$\log_{10}(10^5)$ ergibt 5.

log

Liefert den natürlichen Logarithmus einer Zahl. Natürliche Logarithmen haben die Konstante $e = 2.71828183$ als Basis.

Syntax:

$\log(\text{Zahl})$

Zahl ist die positive reelle Zahl, deren natürlichen Logarithmus Sie berechnen möchten.

Beispiele:

$\log(2.71828183)$ ergibt 1.

$\log(2.71828183^3)$ ergibt 3.

$2.71828183^{\log(4)}$ ergibt 4.

exp

Liefert den Wert der natürlichen Exponentialfunktion einer Zahl e^{Zahl} mit $e = 2.71828183$.

Syntax:

$\exp(\text{Zahl})$

Liefert den Wert der natürlichen Exponentialfunktion für Zahl.

Beispiele:

$\exp(1)$ ergibt 2,7182818 (Näherungswert für e).

$\exp(2)$ ergibt e^2 (7,3890561).

$\exp(\log(3))$ ergibt 3.

Verwenden von mathematische Funktionen in MegaCAD 3 Von 6

abs

Liefert die Absolutwert einer Zahl.

Syntax:

$\text{abs}(\text{Zahl})$

Zahl ist die Zahl, deren Absolutwert Sie berechnen möchten.

Beispiele:

$\text{abs}(a = -3)$ ergibt 3.

$\text{abs}(-1)$ ergibt 1.

Operatoren

Arithmetische Operatoren

+ (Pluszeichen) Addition

Beispiel:

$3 + 3$ ergibt 6.

- (Minuszeichen) Subtraktion / Negation

Beispiele:

$10 - 3$ ergibt 7.

$-(3 + 2)$ ergibt -5.

$-(a = 50)$ ergibt -50.

*** (Sternchen) Multiplikation**

Beispiel:

$3 * 3$ ergibt 9.

/ (Schrägstrich) Division

Beispiel:

$3 / 3$ ergibt 1.

% (Prozentzeichen) Modulodivision

Liefert den Divisionsrest von $\text{Zahl1}/\text{Zahl2}$, so daß $\text{Zahl1} = i * \text{Zahl2} + \text{Rest}$, wobei i eine Ganzzahl ist. Rest hat das

gleiche Vorzeichen wie Zahl1 . Der Absolutwert von Rest ist kleiner als der Absolutwert von Zahl2 .

Beispiele:

$3 \% 2$ ergibt 1.

$13 \% 5$ ergibt 3.

$-6.7 \% 2$ ergibt -0.7.

-6.7%(-2) ergibt -0.7.

^ (Caret-Zeichen) Potenzierung

Beispiel:

$3 \wedge 2$ ergibt 9.

Verwenden von mathematische Funktionen in MegaCAD 4 Von 6

Zuweisungsoperator

= (Gleichheitszeichen) Zuweisung/Werte anzeigen

Beispiele:

Eingabe:

$a = 100$ der Variablen a wird der Wert 100 zugewiesen.

Ausgabe:

$a = 100$ die Variable a hat den Wert 100.

Vergleichsoperatoren

Vergleichsoperatoren vergleichen zwei Werte und liefern dann den Wert 1 (wahr) oder den Wert 0 (falsch).

(siehe auch: Beispiel zur Verwendung von Vergleichsoperatoren in einer MegaCad - Zeichnung)

== (Doppelte Gleichheitszeichen) Gleich

Beispiele:

$10 == 10$ ergibt 1 (wahr).

$(2 + 3) == 5$ ergibt 1 (wahr).

$a == a$ ergibt 1 (wahr).

$3 == 7$ ergibt 0 (falsch).

< > (Ungleich-Zeichen) Ungleich

Beispiele:

$10 < > 10$ ergibt 0 (falsch).

$(2 + 3) < > 5$ ergibt 0 (falsch).

$a < > a$ ergibt 0 (falsch).

$3 < > 7$ ergibt 1 (wahr).

> (Größer-als-Zeichen) Größer als

Beispiele:

$10 > 5$ ergibt 1 (wahr).

$(a = 10) > 20$ ergibt 0 (falsch).

< (Kleiner-als-Zeichen) Kleiner als

Beispiele:

$10 < 5$ ergibt 0 (falsch).

$(a = 10) < 20$ ergibt 1 (wahr).

>= (Größer-gleich-Zeichen) Größer oder gleich

$4 >= 4$ ergibt 1 (wahr).

$3 >= (a = 3) + 0.01$ ergibt 0 (falsch).

<= (Kleiner-gleich-Zeichen) Kleiner oder gleich

Beispiele:

$4 <= 4$ ergibt 1 (wahr).

$3 <= (a = 3) - 0.01$ ergibt 0 (falsch).

Verwenden von mathematische Funktionen in MegaCAD 5 Von 6

Operatorrang

Reihenfolge der Auswertung von Operatoren in Formeln.

Wenn innerhalb einer Formel mehrere Operatoren verwendet werden, werden sie entsprechend ihres Ranges

ausgewertet. Operatoren mit höherem Rang werden zuerst ausgewertet. Werden innerhalb eines Ausdrucks Operatoren

gleichen Ranges verwendet, so werden sie von links nach rechts berechnet. Durch Setzen von Klammern kann man die

Reihenfolge der Auswertung verändern. Im Folgenden sind die Operatoren nach Rang geordnet, beginnend mit dem

höchsten Rang, aufgeführt:

"-" Negation

"^" Potenzierung

"*", "/" Multiplikation, Division

"%" ModuloDivision

"+", "-" Addition, Subtraktion

"==", "<>", "<", ">", "<=", ">=" Vergleich

"=" Zuweisung