Marc Mentat Python API: PyPost

Mentat bietet neben dem „standard“ Post Processing auch die Möglichkeit mit Hilfe von Python Ergebnisse zu extrahieren und bedarfsgerecht zu speichern. Neben der Flexibilität beim Format der erzeugten Ergebnisse ist auch eine Automatisierung der Auswertung denkbar. Dies wiederum ermöglicht eine Prozessintegration.

Eine detaillierte Beschreibung und einige Beispiele zu PyPost sind im „Python Manual“ zu finden. In Mentat ist dieses unter
„Help ➔ Python Manual ➔ Marc Python Tutorial“ verfügbar. In diesem Artikel konzentrieren wir uns auf einige grundlegende Funktionalitäten, um einen möglichst reibungslosen Einstiegt in die Thematik zu ermöglichen.

Als erstes importieren wir das Modul PyPost bzw. die PyPost Funktionen:

```
from py_post import *
```

Nun müssen wir die Ergebnissdatei (t16) öffnen und benötigen außerdem die Anzahl der Inkremente:

```
p = post_open('test.t16')  # open the post file

nincs = p.increments()      # get number of increments
```

Mit diesen Informationen können wir die Ergebnissdatei durchlaufen und Ergebnisse extrahieren (beginnend vom Inkrement 1):

```
for ninc in range(nincs)[1:]:
  try:
    # go to increment number
    p.moveto(ninc)
  except:
    print "Error opening post file: test.t16"
    return
```

Die wichtigsten Ergebnisse sind nachfolgend zusammengefasst:

- Node Scalars (z.B. Verschiebung): `p.node_scalars()`
- Node Vectors (z.B. Contact Force): `p.node_vectors()`
- Element Scalars (z.B. Stress): `p.element_scalars()`
- Element Tensors (z.B.: Total Strain): `p.element_tensors()`
- Global Values (z.B. Volumen): `p.global_values()`
- Contact Body Displacement, Force etc.: `p.cbody_force(contact body ID)`

Die komplette Liste ist im „Python Manual ➔ Marc Python Reference ➔ Chapter 2: PyPost References“ festgehalten.

MSC Software ist Teil der Division Manufacturing Intelligence von Hexagon (Nasdaq Stockholm: HEXA B; hexagon.com). Das Unternehmenslogo von MSC Software und MSC sind Marken oder eingetragene Marken der MSC Software Corporation und/oder ihrer Tochtergesellschaften in den USA und/oder anderen Ländern. NASTRAN ist eine eingetragene Marke der NASA. Alle anderen Markennamen, Produktnamen oder Warenzeichen gehören ihren jeweiligen Eigentümern. © 2018 MSC.Software Corporation. All rights reserved.

Für die Elementergebnisse kann noch die Extrapolationsmethode festgelegt werden. Lineare Extrapolation p.extrapolation('linear') ist der Default. Zusätzlich gibt es noch "translate" und "average".

Die Vorgehensweise bei der Extraktion der Ergebnisse ist immer ähnlich. Wir erläutern das anhand von „Node Vectors".
Wir starten also mit der Extraktion der „Node Vectors" und Knotennummern:

```
nv = p.node_vectors()   # get available node vectors
n = p.nodes()           # get node count
```

Für jedes Inkrement, also innerhalb der oben abgebildeten Schleife, wird eine zusätzliche Schleife gebraucht.
In diesem Fall eine die über alle Knoten geht. Wir extrahieren hier die „Contact Normal Force":

```
# for each node
for j in xrange(0,n):
  # for each node vector
  for i in xrange(0,nv):
    # extract the needed node vector based on label
    if p.node_vector_label(i) == 'Contact Normal Force':
      v = p.node_vector(j,i)
      # get vector components [x, y, z]
      v_x = v.x
      v_y = v.y
      v_z = v.z
```

Somit sind der Vektor und die Komponenten für den jeweiligen Knoten verfügbar und können in einem beliebigen Format in eine Datei geschrieben bzw. weiterverarbeitet werden.

Abschließend noch einige Details zur Konfiguration von Python auf Windows. Oft ist bereits eine Standard Python Installation vorhanden. Damit diese mit der Mentat Python Installation verwendet werden kann muss man sicherstellen, dass:

- Es eine Python 64-bit Version (x86-64) ist (v2018 benötigt Python 3.5.1)
- Den Pfad zur **PATH** Umgebungsvariable hinzufügen, z.B. „C:\MSC.Software\Marc\2017.1.0\mentat2017\python"
- Eine zusätzliche Umgebungsvariable **PYTHONPATH** definieren mit dem Pfad zum Mentat **shlib**-Verzeichnis, z.B. „C:\MSC.Software\Marc\2017.1.0\mentat2017\shlib\win64". Diese Umgebungsvariable stellt sicher, dass die Mentat Python Module **PyPost** und **PyMentat** importiert werden können.

MSC Software ist Teil der Division Manufacturing Intelligence von Hexagon (Nasdaq Stockholm: HEXA B; hexagon.com). Das Unternehmenslogo von MSC Software und MSC sind Marken oder eingetragene Marken der MSC Software Corporation und/oder ihrer Tochtergesellschaften in den USA und/oder anderen Ländern. NASTRAN ist eine eingetragene Marke der NASA. Alle anderen Markennamen, Produktnamen oder Warenzeichen gehören ihren jeweiligen Eigentümern. © 2018 MSC.Software Corporation. All rights reserved.