Materialdaten aus Abaqus ODB auslesen

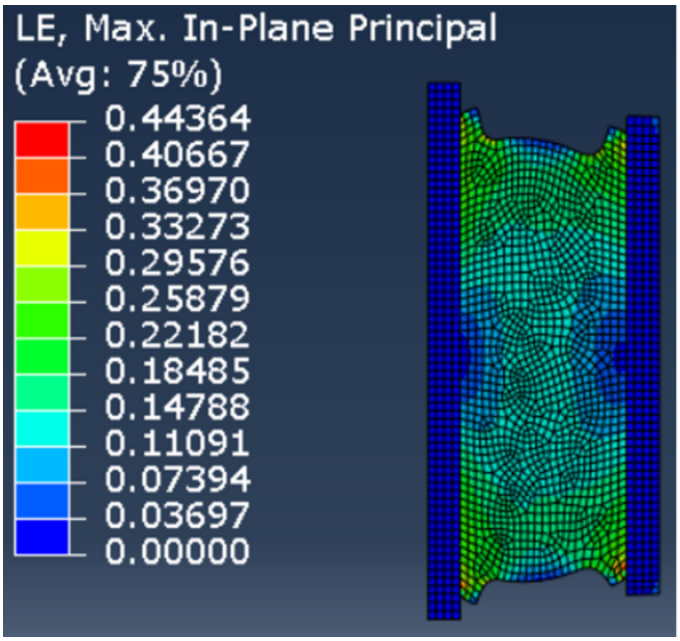
Simulation · https://plm.systemworkx.de/docs/materialdaten-aus-odb/

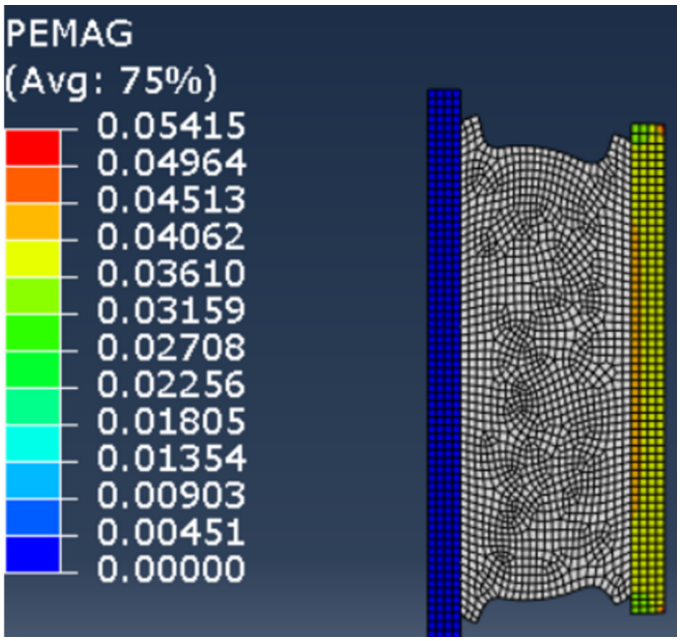
Grundsätzlich werden die für die Analyse mit Abaqus Unified FEA verwendeten Materialdaten in der ODB nicht angezeigt. Hier hilft ein einfaches Python-Skript.

Im vorliegenden Modell kommen zwei Materialien zur Verwendung:

RUBBER, Neo-Hooke, Hyperelastisches Materialmodell

STEEL, Stahl, elastisch-plastisches Materialmodell





Dieses kurze Skript ermöglicht das Anzeigen der verwendeten Materialdaten und –Modelle.
Die Eingabe erfolgt über die Kommandozeile (Kernel Command Line Interface) am unteren Rand von Abaqus/CAE (Module Visualization) oder Abaqus/Viewer.

Hole aktuellen Viewport, ODB-File des aktuellen Viewports, Dateinamen und Pfad

CV = session.viewports[session.currentViewportName]

odbFile = CV.displayedObject

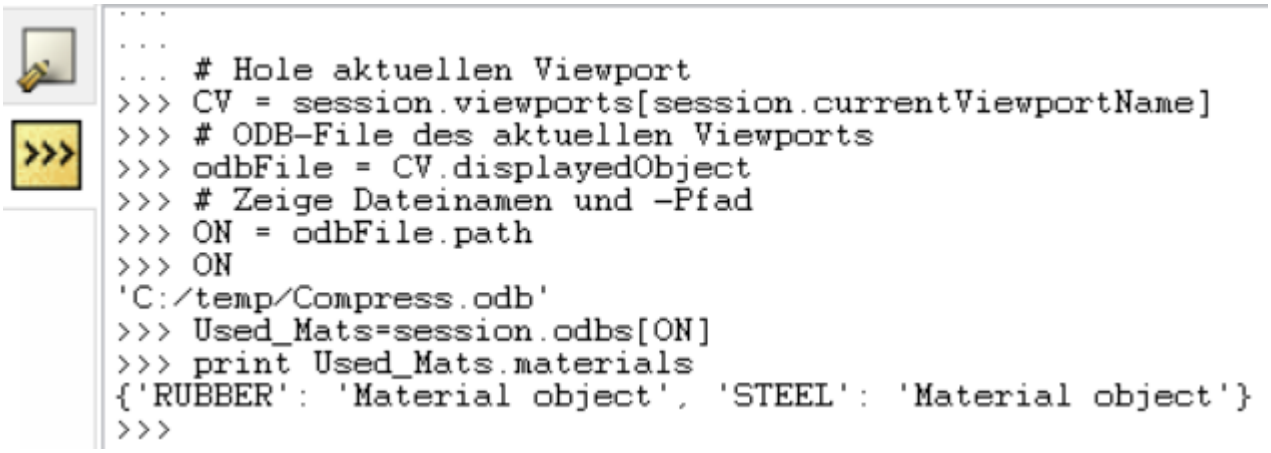
ON = odbFile.path

ON

Used_Mats=session.odbs[ON]

print Used_Mats.materials

Die Ausgabe sieht wie folgt aus :



Nun kann man, weiter über die Kommandozeile, nähere Informationen über die vohandenden Materialdaten anfordern:

print Used_Mats.materials['STEEL']

liefert

()

Informationen, welche Daten zu einem Material vorhanden sind. Hier zum Beispiel die Wärmeleitfähigkeit, die Dichte, Angaben zur Elastizität und der elektrischen Leitfähigkeit. Nun kann man weitere Informationen anfordern: print Used_Mats.materials['STEEL'].elastic

Beispiele:

print Used_Mats.materials['RUBBER'].hyperelastic

```
>>> print Used_Mats.materials['RUBBER'].hyperelastic
({'anisotropicType': FUNG_ANISOTROPIC, 'behaviorType': INCOMPRESSIBLE, 'beta': FITTED_VALUE, 'compressible': OFF, 'dependencies': 0, 'deviatoricResponse':
UNIAXIAL, 'formulation': STRAIN, 'localDirections': 0, 'materialType': ISOTROPIC, 'moduliTimeScale': LONG_TERM, 'n': 1, 'poissonRatio': 0.0, 'properties':
0, 'table': ((0.89, 0.01),), 'temperatureDependency': OFF, 'testData': OFF, 'type': NEO_HOOKE, 'volumetricResponse': DEFAULT})
>>>
```

print Used_Mats.materials['STEEL'].plastic

```
>>> print Used_Mats.materials['STEEL'].plastic
({'dataType': HALF_CYCLE, 'dependencies': 0, 'hardening': ISOTROPIC, 'numBackstresses': 0, 'rate': OFF, 'scaleStress': None, 'staticRecovery': OFF, 'strainRangeDependency': OFF,
'table': ((250.0, 0.0), (350.0, 0.1)), 'temperatureDependency': OFF})
>>> print Used_Mats.materials['STEEL'].plastic
({'dataType': HALF_CYCLE, 'dependencies': 0, 'hardening': ISOTROPIC, 'numBackstresses': 0, 'rate': OFF, 'scaleStress': None, 'staticRecovery': OFF, 'strainRangeDependency': OFF,
'table': ((250.0, 0.0), (350.0, 0.1)), 'temperatureDependency': OFF})
```

print Used_Mats.materials['STEEL‘].elastic

```
>>> print Used_Mats.materials['STEEL'].elastic
({'dependencies': 0, 'moduli': LONG_TERM, 'noCompression': OFF, 'noTension': OFF, 'table': ((210000.0, 0.3),), 'temperatureDependency': OFF, 'type': ISOTROPIC})
>>>
```

ACHTUNG:

Der Materialname innerhalb der eckigen Klammern muss in Hochkommata stehen! Je nach Editor kann das zu Problemen führen.