**Dsg File - Key file for running COMPARE CORE**

Written by GUI

**comp.dsg file see lines 1195-1325**

ofstream dsg("comp.dsg",ios::out);

dsg<<"$ This is the file to run compare"<<endl;

this file contains all the information about the run, e.g., experiments, material parameter list, lower and upper, weights, etc.

Zeta Psi

Diagram

AI-generated content may be incorrect.

dsg<<"EXPR:"<<endl

<<" "<<numTests<<" "

dsg<<endl<<"NDV:"<<endl

<<" "<<numConstants<<endl

numConstants = 2 + 2\*nVE + 12 + 9\* nVP; part of 12 – n,mu,kappa, 4 stiff damage param, zeta, Psi, cut-off strength and cut-off stiff

Note numbering of parameters is as follows

**Es, Nu, Emi, Rhoi, kappa, Kbj, n, mu, mj, betaj, Rj, Hj, ce, Ye, mue, ne, cdj, Ydj, mudj, ndj, Zeta, Psi, Cutoff-Strength, Cutoff-Stiff, Dummy (purple are not in GUI now)**

<<"INIT:"<<endl<<" "

**Order of parameters**

**Es, Nu, Emi, Rhoi, kappa, Kbj, n, mu, mj, betaj, Rj, Hj, cdj, Ydj, mudj, ndj, ce, Ye, mue, ne, Zeta, Psi, Cutoff-Strength, Cutoff-Stiff, Dummy. Where i=1,nVE and j=1,nVP**

dsg<<endl<<"LOWE:"<<endl<<" ";

**Order of parameters**

**Es, Nu, Emi, Rhoi, kappa, Kbj, n, mu, mj, betaj, Rj, Hj, cdj, Ydj, mudj, ndj, ce, Ye, mue, ne, Zeta, Psi, Cutoff-Strength, Cutoff-Stiff, Dummy. Where i=1,nVE and j=1,nVP**

dsg<<endl<<"UPPE:"<<endl<<" ";

**Order of parameters**

**Es, Nu, Emi, Rhoi, kappa, Kbj, n, mu, mj, betaj, Rj, Hj, cdj, Ydj, mudj, ndj, ce, Ye, mue, ne, Zeta, Psi, Cutoff-Strength, Cutoff-Stiff, Dummy. Where i=1,nVE and j=1,nVP**

dsg<<endl<<"SUBP:"<<endl

<<" 1 "; No of active parameters; List of active parameters (note numbering order is sequential – see examples)

1 8 13 14 15 16 17 18 19 20 (given 25 total parameters – 13 through 20 corresponded to stiff damage and strength damage parameters)

1. 5 7 11 13 15 17 (given 34 total parameters (1 VE mech, 2 VP mech) Kb2, m2, beta2, R2, H2

1 5 5 6 8 11 12 (given 25 total parameters (1 VE mech, 1 VP mech) kappa, Kb, mu, R, H

dsg<<"LINK:"<<endl;

Exp#, NDV, list of parameters 1, 2, …. NDV

dsg<<"FACT:"<<endl;

Exp#, NDV, list of NDV’s 1.0s

//output the weights for each constant

dsg<<"MDO:"<<endl<<" ";

List of NumTests values --- each are typically 1.0/NumTests.

For example given 6 tests: 0.1666667, 0.1666667, 0.1666667, 0.1666667, 0.1666667, 0.1666667

If one test: 1.0

// this stuff is always the same too

dsg<<endl<<"OBJS:"<<endl<<" ";

for(i=0;i<numTests;i++)

{

dsg<<"1.0 ";

}

//always the same

dsg<<endl<<"ERR:"<<endl<<" 2"<<endl;

======================

Read by Main or subroutine DSGNRD (inside file Inputdat.for)

This then ultimately calls HYVIP which starts the analysis of each test. Opens multiple files which will contain the results for each test.

OPEN(UNIT=80, FILE=TNAM2(:ICH2)//'.tem',

& STATUS='UNKNOWN')

2000 Format (11(E11.4,1X) )

OPEN(UNIT=90, FILE=TNAM2(:ICH2)//'.plot',

& STATUS='UNKNOWN')

5000 FORMAT (45(E16.8E3,1X) )

5100 FORMAT (45(E16.8E3,1X) )

OPEN(UNIT=91, FILE=TNAM2(:ICH2)//'.alp',

& STATUS='UNKNOWN')

5100 FORMAT (45(E16.8E3,1X) )

OPEN(UNIT=93, FILE=TNAM2(:ICH2)//'.qs',

& STATUS='UNKNOWN')

5000 FORMAT (45(E16.8E3,1X) )

if(modl.eq.10) then

OPEN(UNIT=92, FILE=TNAM2(:ICH2)//'.dam',

& STATUS='UNKNOWN')

OPEN(UNIT=94, FILE=TNAM2(:ICH2)//'.damy',

& STATUS='UNKNOWN')

endif

5000 FORMAT (45(E16.8E3,1X) )

Plus creates a number of temporary files

Sim%d.in where %d runs from 1 to number of total experiments simulated

Sim%d.out where %d runs from 1 to number of total experiments simulated

u%d.tem where %d runs from 1 to number of total experiments simulated

final.val – file that contains the final values of parameters.