Validation2DMIS Users Guide

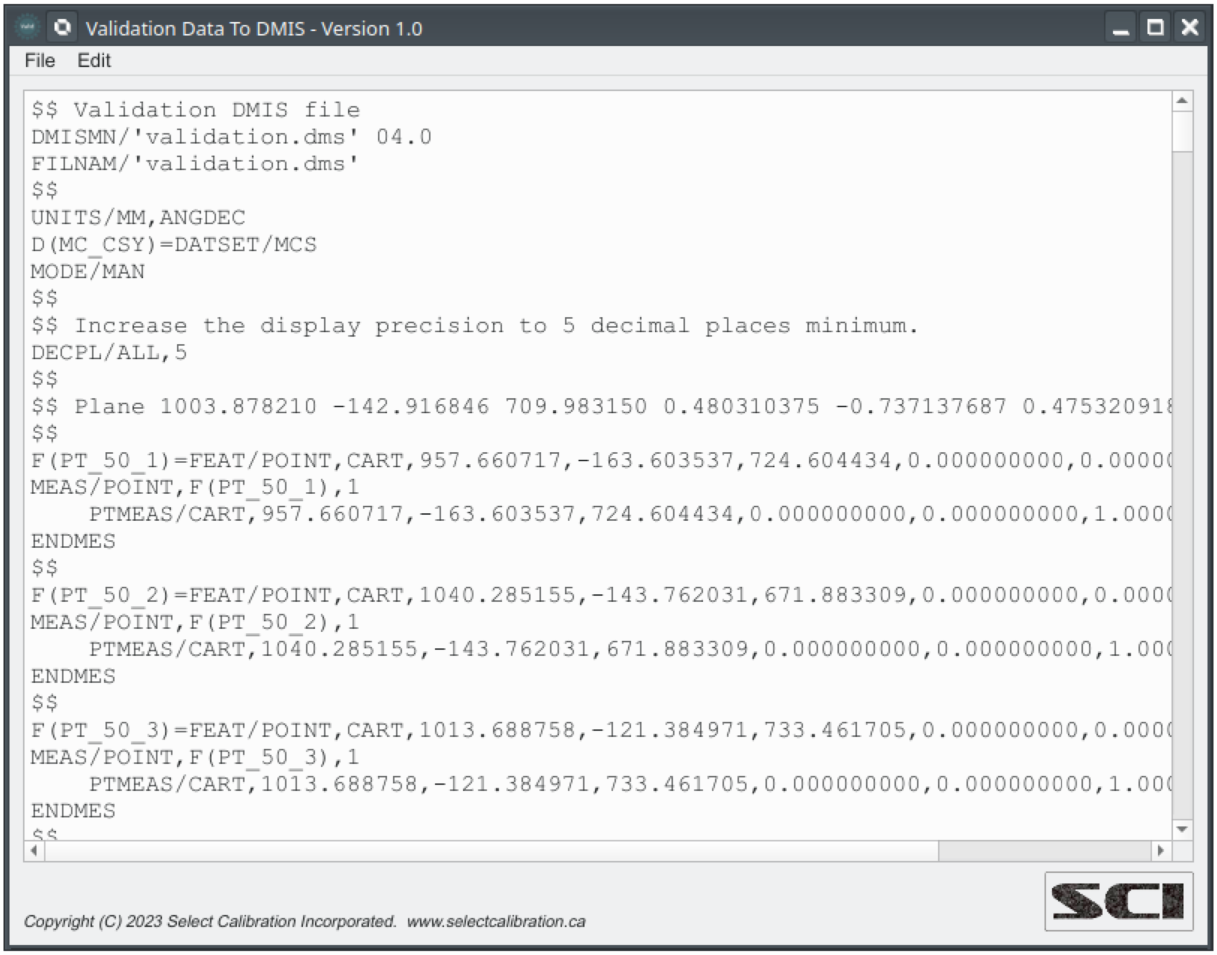
Table of Contents

Purpose

The *Validation2DMIS* utility creates a DMIS part program based on the validation data from *MeasureDirect*. The DMIS program contains the point data and constructed features of the type specified in the validation data file. Creation of a DMIS program from the validation data allows importing into any software that supports the DMIS standard.

User Interface

The user interface of *Validation2DMIS* is shown in illustration 1. This utility is spartan in terms of options as it has a very specific purpose.



```
$$ Validation DMIS file
DMISMN/'validation.dms' 04.0
FILNAM/'validation.dms'
$$
UNITS/MM,ANGDEC
D(MC_CSY)=DATSET/MCS
MODE/MAN
$$
$$ Increase the display precision to 5 decimal places minimum.
DECPL/ALL,5
$$
$$ Plane 1003.878210 -142.916846 709.983150 0.480310375 -0.737137687 0.475320918
$$
F(PT_50_1)=FEAT/POINT,CART,957.660717,-163.603537,724.604434,0.000000000,0.000000
MEAS/POINT,F(PT_50_1),1
    PTMEAS/CART,957.660717,-163.603537,724.604434,0.000000000,0.000000000,1.0000
ENDMES
$$
F(PT_50_2)=FEAT/POINT,CART,1040.285155,-143.762031,671.883309,0.000000000,0.0000
MEAS/POINT,F(PT_50_2),1
    PTMEAS/CART,1040.285155,-143.762031,671.883309,0.000000000,0.000000000,1.000
ENDMES
$$
F(PT_50_3)=FEAT/POINT,CART,1013.688758,-121.384971,733.461705,0.000000000,0.0000
MEAS/POINT,F(PT_50_3),1
    PTMEAS/CART,1013.688758,-121.384971,733.461705,0.000000000,0.000000000,1.000
ENDMES
$$
```

Copyright (C) 2023 Select Calibration Incorporated. www.selectcalibration.ca

Illustration 1: View of Validation2DMIS after loading the validation data.

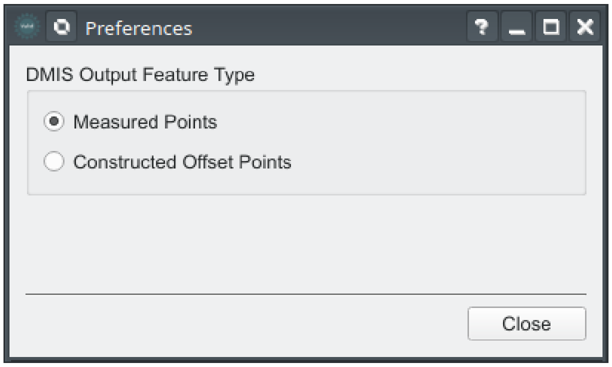
Options:

Menu Option	*Description*
File – Open	Open a validation data file. The DMIS program is automatically generated if the selected file is valid.

Menu Option	*Description*
File – Export DMIS	Create a DMIS file. The default extension of the DMIS file is DMS.
File – Quit	Close the Validation2DMIS program.
Edit – Preferences	Open the preferences dialog to allow customization of the generated DMIS program. See *Preferences* below.

Preferences

The preferences dialog allows customization of the generated DMIS file. Illustration 2 shows the current options for creating DMIS files.



DMIS Output Feature Type

● Measured Points

○ Constructed Offset Points

Illustration 2: Preferences Dialog.

Options:

Option	*Description*
Measured Points	The points for each set of generated features uses the DMIS command FEAT/POINT. The inspection software that loads this data will try to physically measure this point if the program is executed.
Constructed Offset Points	The points for each set of generated features uses the DMIS command CONST/POINT with the offset option. The location of each point is offset from feature PT_ZERO which is defined near the top of the DMIS generated file. Running this program does not require the CMM to measure anything.

DMIS Notes

The version of DMIS generated is 4.0. The generated DMIS program is probably compatible with software that supports older versions of DMIS but this has not been tested.

The display resolution is changed to 5 decimal places with the following command:

```
$$
$$ Increase the display precision to 5 decimal places minimum.
DECPL/ALL,5
```

The preferred resolution is 6 or more decimal places but some software, such as PC-DMIS, will only allow a maximum of 5 decimal places.

PC-DMIS does not recognize the command DECPL/ when importing the DMIS program. The display resolution must be manually changed after importing the DMIS program.

Point Types

The type of point features created can be measured points or constructed offset points based on the preference settings. The following shows examples of each type.

Measured Points

```
$$
F(PT_50_1)=FEAT/POINT,CART,-552.921805,-1113.843456,-554.504267,0.000000000,0.000000000,1.000000000
MEAS/POINT,F(PT_50_1),1
    PTMEAS/CART,-552.921805,-1113.843456,-554.504267,0.00000000,0.00000000,1.00000000
ENDMES
```

Constructed Points

```
$$
F(PT_ZERO)=FEAT/POINT,0,0,0,0,0,1
...
$$
F(PT_50_1)=FEAT/POINT,CART,-552.921805,-1113.843456,-554.504267,0.00000000,0.00000000,1.00000000
CONST/POINT,F(PT_50_1),MOVEPT,F(PT_ZERO),-552.921805,-1113.843456,-554.504267
```

PC-DMIS Specific Notes

In order to properly compare results it is necessary to access the constructed feature values with more precision than PC-DMIS allows by default. The following subroutines were created to output results with sufficient precision. The subroutine output files are *sphere_results.txt*, *plane_results.txt*, *line_results.txt*, and *circle_results.txt.*

Subroutines

```
STARTUP    =ALIGNMENT/START,RECALL:USE_PART_SETUP,LIST=YES
            ALIGNMENT/END
            MODE/MANUAL
            MOVESPEED/ 100
            FORMAT/TEXT,OPTIONS, ,HEADINGS,SYMBOLS, ;NOM,TOL,MEAS,DEV,OUTTOL, ,
            LOADPROBE/SERVICE_PROBE
            TIP/T1A0B0, SHANKIJK=0, 0, 1, ANGLE=0
$$ NO,
            ====================================================================
S1         =GENERIC/SPHERE,DEPENDENT,CARTESIAN,OUT,$
            NOM/XYZ,<0,0,0>,$
            MEAS/XYZ,<0,0,0>,$
            NOM/IJK,<0,0,1>,$
            MEAS/IJK,<0,0,1>,$
            RADIUS/0,0
P1         =GENERIC/PLANE,DEPENDENT,CARTESIAN,$
            NOM/XYZ,<0,0,0>,$
            MEAS/XYZ,<0,0,0>,$
            NOM/IJK,<0,0,1>,$
            MEAS/IJK,<0,0,1>
L1         =GENERIC/LINE,DEPENDENT,CARTESIAN,$
            NOM/XYZ,<0,0,0>,$
            MEAS/XYZ,<0,0,0>,$
            NOM/IJK,<0,0,1>,$
            MEAS/IJK,<0,0,1>,$
            DISTANCE/0,0
```

```
C1         =GENERIC/CIRCLE,DEPENDENT,CARTESIAN,OUT,$
            NOM/XYZ,<0,0,0>,$
            MEAS/XYZ,<0,0,0>,$
            NOM/IJK,<0,0,1>,$
            MEAS/IJK,<0,0,1>,$
            RADIUS/0,0
            ROUTINE/END
$$ NO,
            ====================================================================
            SUBROUTINE/VALIDATION_SPHERE_CLEAR,
                 =
V1         =FILE/EXISTS,"sphere_results.txt"
            IF/V1 <> 0
              FILE/DELETE,"sphere_results.txt"
            END_IF/
            ENDSUB/
$$ NO,
            ====================================================================
            SUBROUTINE/VALIDATION_PLANE_CLEAR,
                 =
V1         =FILE/EXISTS,"plane_results.txt"
            IF/V1 <> 0
              FILE/DELETE,"plane_results.txt"
            END_IF/
            ENDSUB/
$$ NO,
            ====================================================================
            SUBROUTINE/VALIDATION_LINE_CLEAR,
                 =
V1         =FILE/EXISTS,"line_results.txt"
            IF/V1 <> 0
              FILE/DELETE,"line_results.txt"
            END_IF/
            ENDSUB/
$$ NO,
            ====================================================================
            SUBROUTINE/VALIDATION_CIRCLE_CLEAR,
                 =
V1         =FILE/EXISTS,"circle_results.txt"
            IF/V1 <> 0
              FILE/DELETE,"circle_results.txt"
            END_IF/
            ENDSUB/
$$ NO,
            ====================================================================
            SUBROUTINE/VALIDATION_SPHERE_OUT,
                FEATURE = {S1} : SPHERE,
                 =
DIM LOC1= LOCATION OF SPHERE {FEATURE}  UNITS=MM ,$
GRAPH=OFF  TEXT=OFF  MULT=10.00  OUTPUT=BOTH  HALF ANGLE=NO
AX    NOMINAL       +TOL       -TOL       MEAS        DEV     OUTTOL
RN       0.000      0.050      0.000      0.000      0.000      0.000 #--------
END OF DIMENSION LOC1
FPTR       =FILE/OPEN,"sphere_results.txt",APPEND
            ASSIGN/TEXT=FORMAT("Sphere %12.6f",DOUBLE(FEATURE.X))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.6f",DOUBLE(FEATURE.Y))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.6f",DOUBLE(FEATURE.Z))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.6f",DOUBLE(FEATURE.D))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.6f",DOUBLE(LOC1.RN.MEAS))
            FILE/WRITELINE,FPTR,STR(TEXT)
            FILE/CLOSE,FPTR,KEEP
            ENDSUB/
$$ NO,
            ====================================================================
            SUBROUTINE/VALIDATION_PLANE_OUT,
                FEATURE = {P1} : PLANE,
                 =
DIM LOC2= LOCATION OF PLANE {FEATURE}  UNITS=MM ,$
GRAPH=OFF  TEXT=OFF  MULT=10.00  OUTPUT=BOTH  HALF ANGLE=NO
AX    NOMINAL       +TOL       -TOL       MEAS        DEV     OUTTOL
FL       0.000      0.050      0.000      0.000      0.000      0.000 #--------
END OF DIMENSION LOC2
FPTR       =FILE/OPEN,"plane_results.txt",APPEND
            ASSIGN/TEXT=FORMAT("Plane %12.6f",DOUBLE(FEATURE.X))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.6f",DOUBLE(FEATURE.Y))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.6f",DOUBLE(FEATURE.Z))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.9f",DOUBLE(FEATURE.I))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.9f",DOUBLE(FEATURE.J))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.9f",DOUBLE(FEATURE.K))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.6f",DOUBLE(LOC2.FL.MEAS))
            FILE/WRITELINE,FPTR,STR(TEXT)
            FILE/CLOSE,FPTR,KEEP
            ENDSUB/
$$ NO,
            ====================================================================
            SUBROUTINE/VALIDATION_LINE_OUT,
                FEATURE = {L1} : LINE,
                 =
DIM LOC3= LOCATION OF LINE {FEATURE}  UNITS=MM ,$
GRAPH=OFF  TEXT=OFF  MULT=10.00  OUTPUT=BOTH  HALF ANGLE=NO
AX    NOMINAL       +TOL       -TOL       MEAS        DEV     OUTTOL
ST       0.000      0.050      0.000      0.000      0.000      0.000 #--------
END OF DIMENSION LOC3
FPTR       =FILE/OPEN,"line_results.txt",APPEND
            ASSIGN/TEXT=FORMAT("Line %12.6f",DOUBLE(FEATURE.X))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.6f",DOUBLE(FEATURE.Y))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.6f",DOUBLE(FEATURE.Z))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.9f",DOUBLE(FEATURE.I))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.9f",DOUBLE(FEATURE.J))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.9f",DOUBLE(FEATURE.K))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.6f",DOUBLE(LOC3.ST.MEAS))
            FILE/WRITELINE,FPTR,STR(TEXT)
            FILE/CLOSE,FPTR,KEEP
            ENDSUB/
$$ NO,
            ====================================================================
            SUBROUTINE/VALIDATION_CIRCLE_OUT,
                FEATURE = {C1} : CIRCLE,
                 =
DIM LOC4= LOCATION OF CIRCLE {FEATURE}  UNITS=MM ,$
GRAPH=OFF  TEXT=OFF  MULT=10.00  OUTPUT=BOTH  HALF ANGLE=NO
AX    NOMINAL       +TOL       -TOL       MEAS        DEV     OUTTOL
RN       0.000      0.050      0.000      0.000      0.000      0.000 #--------
END OF DIMENSION LOC4
FPTR       =FILE/OPEN,"circle_results.txt",APPEND
            ASSIGN/TEXT=FORMAT("Circle %12.6f",DOUBLE(FEATURE.X))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.6f",DOUBLE(FEATURE.Y))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.6f",DOUBLE(FEATURE.Z))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.9f",DOUBLE(FEATURE.I))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.9f",DOUBLE(FEATURE.J))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.9f",DOUBLE(FEATURE.K))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.6f",DOUBLE(FEATURE.D))
            ASSIGN/TEXT=TEXT + FORMAT(" %12.6f",DOUBLE(LOC4.RN.MEAS))
            FILE/WRITELINE,FPTR,STR(TEXT)
            FILE/CLOSE,FPTR,KEEP
            ENDSUB/
$$ NO,
            ====================================================================
```

Example Subroutine Call Function

The following shows examples of calling the subroutine for 3D line features. The first call to

CLEAR is used to delete all previous results where each subsequent call to *OUT* adds an entry to the output file.

```
CS1 =CALLSUB/VALIDATION_LINE_CLEAR,C:\feature_output_subroutines.PRG:,,
CS2 =CALLSUB/VALIDATION_LINE_OUT,C:\feature_output_subroutines.PRG:{LN_50},,
CS3 =CALLSUB/VALIDATION_LINE_OUT,C:\feature_output_subroutines.PRG:{LN_49},,
...
CS50 =CALLSUB/VALIDATION_LINE_OUT,C:\feature_output_subroutines.PRG:{LN_1},,
```

The subroutine examples use the Legacy output of PC-DMIS for form results. There is a known issue with the form result of PC-DMIS 3D lines so that data should be ignored when performing comparisons. The position and direction should always match regardless of the feature type.

The generic features are used as place-holders for the actual feature when the subroutine program is loaded as a part program. Without the generic features the form dimensions will show errors.

Revision History		
Date	*Version*	*Changes*
May 3, 2023	1.0	New program.