DataNet QUALITY SYSTEMS

WHITE PAPER:

Six questions to ask before linking Coordinate Measuring Machines (CMM) to Statistical Process Control (SPC) Software

Table of Contents

Introduction

Coordinate Measuring Machines (CMM) can help you slash inspection costs, greatly improve inspection capabilities and improve final product quality. But for all their benefits, CMMs do not improve the manufacturing process.

There is a big difference between "making quality products" and "inspecting quality into your products". CMMs do the later. If your process makes 90 good parts and 10 bad parts, then your CMMs will be able to sort the good from bad. But you still suffer the loss of investment in time, energy, wages and raw material costs by producing defective parts In short, while CMMs do help improve final product quality, CMMs alone do not improve yields.

Statistical Process Control (SPC), however, is a methodology that enables you to improve the processes that make your products, which will in turn produce higher quality yields. SPC software systems can automate that methodology, making it easier to reduce scrap and rework, increase productivity and improve quality.

The first task in implementing SPC is to establish a baseline of your process capabilities. Then identify areas of variation and focus your attention on those areas that will have the greatest impact on your production yields.

SPC software will continually monitor these areas of variation and notify the appropriate personnel (in real time) whenever there is a violation or a drift in the process. This is a key value provided by real-time SPC software --helping you to be proactive in spotting process drifts early instead of reacting to scrapped product.

Linking CMMs to an SPC system is a very rewarding "next step" for a company's improvement strategy. Collecting the highly accurate data captured by a CMM and then analyzing the information with an SPC system can help quickly pinpoint the root causes of costly process variations and offer insights on how to correct them.

Companies that use CMMs and SPC together to eliminate defects, reduce down time and improve overall product quality gain a significant competitive edge and ultimately help improve the bottom line.

So why aren't more manufacturers linking CMMs to SPC systems? Marrying a CMM to an SPC software package can pose several challenges. This white paper is dedicated to helping you successfully address some of these integration challenges.

Six questions you should ask before linking CMMs to SPC

1) Will my SPC software "talk" to my CMMs?

Hopefully this question was asked before your SPC system was purchased and implemented. All CMMs have output capabilities, and many have standardized their outputs to industry guidelines. A good SPC software product will support these various outputs, making it compatible with any brand of CMM.

But be cautious because the answer to this question could be: "Yes, we work with all XYZ models". What if you don't have those models? What if you wanted to upgrade to a different model? What if you needed to purchase a different brand of CMM for a specialized job?

Unfortunately, if your SPC software is tightly integrated to only one or two CMM models, you are married to that solution and, like the saying goes, "it's for better or worse." Although this approach might work well in the short run, it can mean turbulent times in your future.

The correct answer to this question should be, "Yes, we work with all of the following outputs..." The sign of a robust SPC software product is its flexibility in linking with many types of CMMs and their outputs. (See Appendices A and B for typical CMM output examples.)

In summary, be sure to find out what file types are supported by your SPC software package and compare them to your CMM output capabilities. A software vendor who works closely with only one or two CMM providers will either limit your freedom of choice or, even worse, not be compatible with your CMM.

2) Who configures the initial CMM-to-SPC link?

Ordinarily, the CMM operators are responsible for the day-to-day activities of a CMM. These activities include among other things: adjusting the CMM program routines to add a new datum, removing an obsolete datum or modifying the definition of an existing datum. These are simple tasks for a properly trained CMM operator.

Similarly, Quality Managers are responsible for configuring SPC software. They have a view of the "big picture" and understand how the individual components (the CMM being one of them) fit into the company's overall quality objectives. By monitoring key variables (datums) they can detect and predict potential problems before they're realized. For someone trained in the various quality concepts, this is a simple task.

So whose responsibility is it to identify which features to gather data against and to configure the initial CMM-to-SPC link? The dynamic nature of a CMM and the sheer volume of data that it could potentially generate would overwhelm any SPC system if it required manual administration. A strategy employing automation could produce a "zero administration" environment.

SPC software that is truly Zero-Administration must be able to automatically create Parts, Variables, and Tags whenever a new Part or Variable is encountered. It must be able to "dynamically mask" an output file whenever the layout of that file changes. It must support an unlimited number of Parts, Variables, and Tags. And, it must be able to run unattended and recover from errors without adversely affecting the SPC data.

Once a Zero Administration link is conceived, there are three possible situations that you might find at your plant:

a) All features of the CMM need to be monitored by the SPC software.

If you find yourself in this situation, you're in a good position. Why? Because your SPC software should be able to automatically input Parts, Variables and Tags based on the data found in the CMM output file. The CMM output file contains all of the detailed information required to create those Parts and Variables. Most SPC software packages need at least the target and either an upper or lower specification limit for a given feature. Check with your SPC vendor to see which "Auto-Create" features they support.

b) Only selected CMM features need to be monitored and the CMM Operator has already flagged them.

If your CMM Operator has flagged the key features to monitor, those flags will be embedded in the CMM output file. Again, your SPC software should be able to parse the output file for only these features that have the appropriate flag and automatically create the corresponding Parts, Variables and Tags.

c) Only selected CMM features need to be monitored and only the Quality Manager knows which ones.

And finally, if you are in the situation where the Quality Manager has the information, the simplest solution is to treat the CMM data as if you needed all of the features (see a)). Let the SPC software do the work of setting up the Parts and Variables for you. Then, after these Parts have been created, you can delete the Variables that you don't need.

So, back to the question: Who configures the initial CMM-to-SPC link? The CMM! Of course, the CMM operator still needs to create a CMM program, but the output generated from those programs will drive the SPC software.

3) Who is responsible for the day-to-day administration of my CMM-to-SPC software link?

After the initial link from CMM to SPC has been established, there will be ongoing tasks including adding, removing and changing the CMM and SPC variables. Who is responsible for the day-to-day administration of both systems? Before we answer this question, it's important to understand the dynamic nature of any CMM program and how this can play havoc with most SPC software packages.

A CMM program, or routine, is required to operate the CMM hardware. These routines instruct the CMM to follow a path to specified feature locations or "datums". Based on where the datums are found, we can measure these features and write their locations to a file. If more information is needed about a feature of a part, you add a new datum to the routine. If the tolerances of a feature have been adjusted, you modify the tolerances stored in the routine. If a feature of your product is no longer important, you remove the datum from the routine. And so forth

Likewise, the corresponding output file generated by a CMM program will reflect any changes you have made. Two common methods for reading this output file are called "Positional masking" and "Key Identifier masking". Positional masking refers to the technique of locating a value within a file by its relative position (i.e. 4th row, 16th column, 10 characters long). Key Identifier masking uses a key word to identify where in the file the values exist (e.g. Look for the word "Datum 1X" and whatever follows is the measurement).

Initially, the administrative overhead of using either technique is to identify and mask every feature to a corresponding SPC variable. The more variables you have, the more time-consuming this task becomes.

But the actual measure of administrative overhead should be: 'How much time does it take to make a change to a CMM program?' For example, if you decide to insert a new datum into your CMM program you would expect this to take a few minutes. But the resultant output file shifts the data making the "positional mask" useless, and in many cases, renames the Datum IDs making the "key identifier mask" useless as well. The new output file must be "masked" again! This re-masking could take hours depending on the actual number of features and the SPC software masking capabilities.

What if someone suggested not to change the CMM program once it has been masked? Ask any CMM operator and you will find that CMM programs change frequently. They are always adding, removing, or modifying an existing datum. A CMM program is a living and breathing entity that is constantly evolving. See Appendix C for an example of the consequences of dynamic CMM data.

The correct solution is to use the appropriate CMM output file with a "dynamic" mask. A dynamic mask will adjust itself whenever the CMM routines are adjusted. If you insert a new datum into your CMM program, the SPC software automatically shifts its mask to account for the new feature. If you remove one, the SPC software will adjust again.

In order for the CMM data to be truly "free" (i.e., automatically passing the CMM data already being collected to an SPC software package without requiring additional operator intervention) then there also must be no additional overhead to collecting that data. That includes any administrative overhead required to keep the CMM file linked to your SPC software. Your SPC software, therefore, must be able to dynamically mask itself. If your SPC software package requires you to modify the mask every time a CMM routine has changed, then your data is not free.

So, back to the question: Who is responsible for the day-to-day administration of the CMM programs to SPC software link? The answer is: The CMM. Of course, the CMM operator will continue to modify CMM programs, but the output generated from those programs will drive the SPC software.

4) Will my CMM-to-SPC link run unattended?

Assuming that the link between your CMM and SPC software is in place, the next step in improving your overall quality is to take all of the data that has been gathered and distill it down to meaningful nuggets of information. This is the reason why we do SPC in the first place.

To derive such information, you need to answer a series of questions which describe the capabilities of your SPC software to 1) validate data in real time, 2) warn/predict future events, and 3) trap errors and continue running.

Please take the following survey. Give yourself 1 point for every "Yes" answer and 0 points for every "No".

____ If data collected from the CMM is unreasonable (i.e., grossly too large or too small) can your software detect AND reject the offending data?)

____ When data has been rejected, are there records of the activity for follow up?

____ When a trend or out of control condition is detected, are there mechanisms (e.g., e-mail, pager, etc.) to notify the appropriate people in real time?

____ Can you create custom trend rules to monitor specific situations that may be unique to your CMM or company?

____ If the CMM "hiccups" and sends erroneous data, will SPC software trap and ignore these errors?

____ If an error does occur, whether it is hardware or software related, does the system have the ability to recover?

Score:

5 – 6: Your CMM-to-SPC link is robust enough to operate unattended.

3 – 4: Your CMM-to-SPC link is adequate but must be actively monitored.

< 3: The value of linking your CMM-to-SPC software is questionable.

5) Is my CMM data accessible?

The value of any data collected from a CMM is related directly to who has access to it and when it is available. Being able to respond to adverse conditions quickly might be the difference between saving money and incurring costs.

The data should be stored in a centrally located, relational database. The most common databases are MS-SQL and Oracle but others can be used, provided their database schema is public and there are ways to query the data. Any ODBC compliant database will meet these requirements.

The data needs to be available in real time. If your system consists of gathering data to a holding area and then updating the SPC database on a specified time interval (e.g., once a day, once an hour, etc.) then you will be limited as to how quickly you can respond to problems. Ideally, you should have your data available the moment the CMM has completed writing its results to a file.

The data needs to be visible by all. The purpose of collecting this data cannot be to build more fiefdoms. Rather, the goal is to share your findings with everyone in the company and with people upstream and downstream in the process. The SPC system, therefore, must have robust reporting capabilities with easy access to the information for operators, managers and other business decision makers.

6) When should I start my CMM-to-SPC implementation?

The most successful companies aren't those who wait until they've developed a "perfect" solution. Successful companies start with pilots and base requirements and expand from there. Almost without exception, Quality leaders have started with limited resources and a simple set of SPC goals. What ultimately makes them leaders is they get started sooner rather than later and continuously improve their processes. By taking an incremental approach, they begin to experience the benefits of improving quality and then expand those benefits to other areas over time.

CMMs ensure the accuracy and reliability of your products. SPC offers tremendous bottom-line benefits to companies by reducing scrap, eliminating waste, and increasing yields. Linking your CMM to an SPC system is critical to maximizing those benefits. But no one gets there without taking the first step.

When getting started, remember to:

a) Identify the Key Quality Characteristics early on in your implementation. Focus only on variables that contribute to the bottom line or are required by your customer.

b) Be conscious of the current workload of an average CMM operator/programmer. Adding more work is out of the question. Reducing their workload should be the goal.

c) Understand the dynamic nature of a CMM. Key variables can be added to or removed from the CMM program at will. This inherent flexibility of the CMM is crucial in understanding how to implement the CMM-to-SPC link.

d) Understand the ongoing administration requirements to maintain the CMM-to-SPC link.

e) Coordinate who is responsible for responding to SPC violations and how they should be addressed.

Finally, don't hesitate to ask for help from your SPC vendor to assist with the CMM integration. If you are interested in testing SPC software in your environment you may obtain a free 60-day software trial by calling 248.357.2200 or visiting www.winspc.com

About the Author

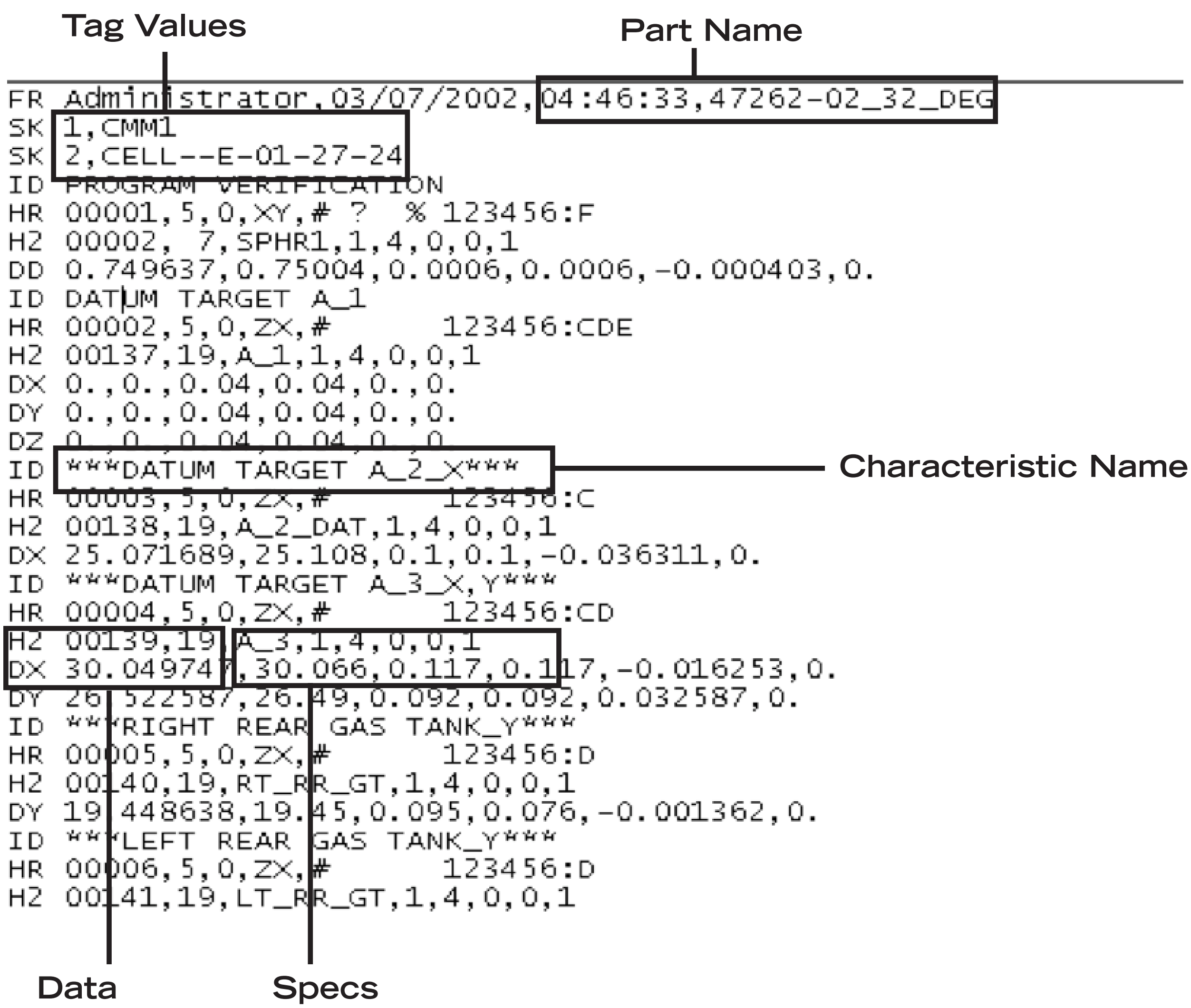
Frank Tappen is vice president of solution delivery services for DataNet Quality Systems. Tappen has more than 17 years of SPC software development, training and sales experience. Tappen oversees the coordination of professional field and integration personnel who are actively building customer solutions for DataNet clients. Tappen is a Certified Quality Engineer, Six Sigma Black Belt and a Certified Quality Auditor through ASQ and holds a Bachelor of Science degree from the University of Michigan.

DataNet Quality Systems

DataNet Quality Systems delivers continuous improvement software and services that empower manufacturers to improve their product and process quality through real-time Statistical Process Control (SPC). Our award-winning WinSPC and QualTrend software allow shop-floor personnel to identify, control and prevent manufacturing process and product defects in real-time, while providing manufacturing engineers and management with advanced statistical analysis and reporting tools. With over 2500 installed facilities worldwide and distributors throughout the world, DataNet is dedicated to delivering a high level of customer service and support, shop-floor expertise, and professional continuous improvement, Six Sigma, and lean Six Sigma services.

Appendix A

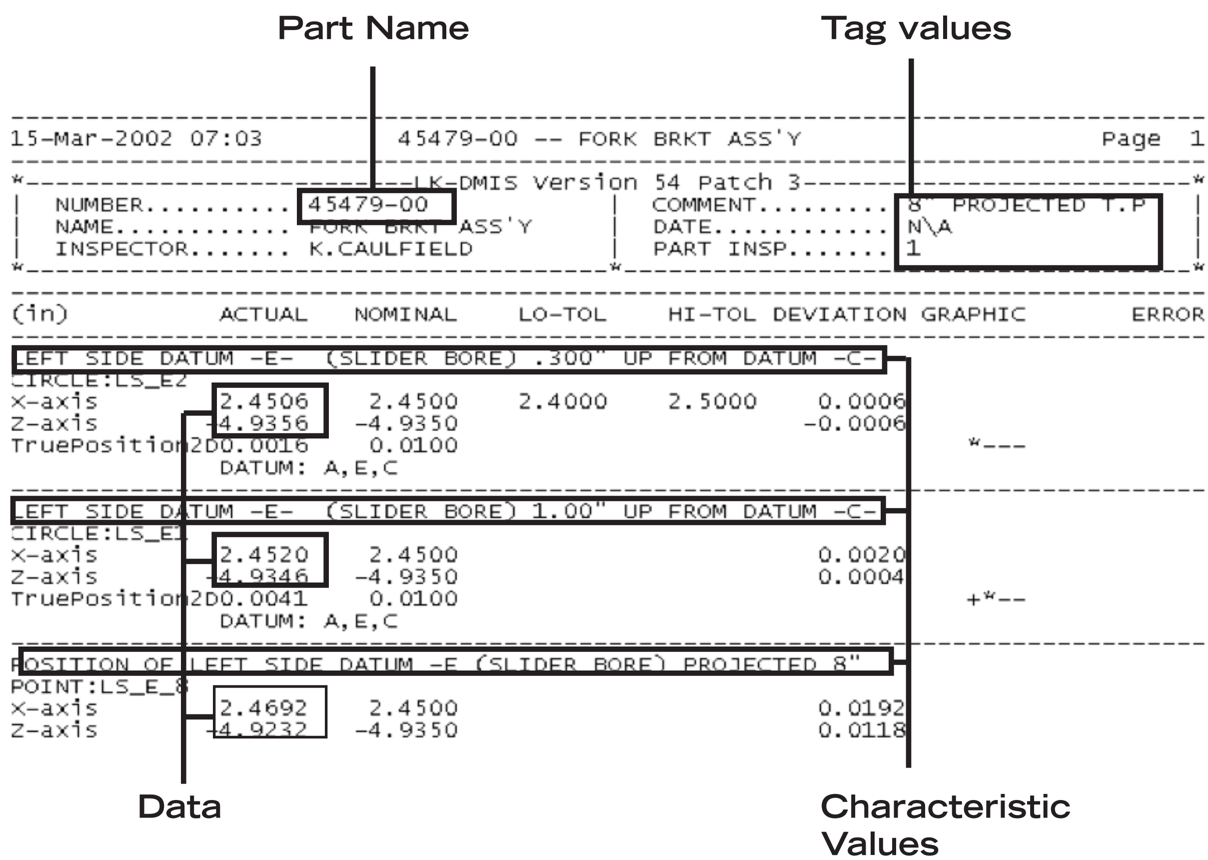
Example: .RTF file taken from a CMM




```
FR Administrator,03/07/2002,04:46:33,47262-02_32_DEG
SK 1,CMM1
SK 2,CELL--E-01-27-24
ID PROGRAM VERIFICATION
HR 00001,5,0,XY,# ?  % 123456:F
H2 00002,  7,SPHR1,1,4,0,0,1
DD 0.749637,0.75004,0.0006,0.0006,-0.000403,0.
ID DATUM TARGET A_1
HR 00002,5,0,ZX,#       123456:CDE
H2 00137,19,A_1,1,4,0,0,1
DX 0.,0.,0.04,0.04,0.,0.
DY 0.,0.,0.04,0.04,0.,0.
DZ 0.,0.,0.04,0.04,0.,0.
ID ***DATUM TARGET A_2_X***
HR 00003,5,0,ZX,#       123456:C
H2 00138,19,A_2_DAT,1,4,0,0,1
DX 25.071689,25.108,0.1,0.1,-0.036311,0.
ID ***DATUM TARGET A_3_X,Y***
HR 00004,5,0,ZX,#       123456:CD
H2 00139,19,A_3,1,4,0,0,1
DX 30.049747,30.066,0.117,0.117,-0.016253,0.
DY 26.522587,26.49,0.092,0.092,0.032587,0.
ID ***RIGHT REAR GAS TANK_Y***
HR 00005,5,0,ZX,#       123456:D
H2 00140,19,RT_RR_GT,1,4,0,0,1
DY 19.448638,19.45,0.095,0.076,-0.001362,0.
ID ***LEFT REAR GAS TANK_Y***
HR 00006,5,0,ZX,#       123456:D
H2 00141,19,LT_RR_GT,1,4,0,0,1
```

Appendix B

Example: .RTF file taken from a CMM




```
-----------------------------------------------------------------------------------
15-Mar-2002 07:03          45479-00 -- FORK BRKT ASS'Y                       Page  1
-----------------------------------------------------------------------------------
*-------------------------------LK-DMIS Version 54 Patch 3-------------------------*
|  NUMBER.......... 45479-00             |  COMMENT......... 8" PROJECTED T.P     |
|  NAME............ FORK BRKT ASS'Y      |  DATE............ N\A                  |
|  INSPECTOR....... K.CAULFIELD          |  PART INSP....... 1                    |
*----------------------------------------*----------------------------------------*
-----------------------------------------------------------------------------------
(in)          ACTUAL    NOMINAL     LO-TOL     HI-TOL  DEVIATION GRAPHIC      ERROR
-----------------------------------------------------------------------------------
LEFT SIDE DATUM -E-  (SLIDER BORE) .300" UP FROM DATUM -C-
CIRCLE:LS_E2
X-axis         2.4506    2.4500     2.4000     2.5000     0.0006
Z-axis        -4.9356   -4.9350                          -0.0006
TruePosition2D0.0016     0.0100                                      *---
              DATUM: A,E,C
-----------------------------------------------------------------------------------
LEFT SIDE DATUM -E-  (SLIDER BORE) 1.00" UP FROM DATUM -C-
CIRCLE:LS_E1
X-axis         2.4520    2.4500                           0.0020
Z-axis        -4.9346   -4.9350                           0.0004
TruePosition2D0.0041     0.0100                                      +*--
              DATUM: A,E,C
-----------------------------------------------------------------------------------
POSITION OF LEFT SIDE DATUM -E (SLIDER BORE) PROJECTED 8"
POINT:LS_E_8
X-axis         2.4692    2.4500                           0.0192
Z-axis        -4.9232   -4.9350                           0.0118
```

Appendix C

Example: Handling constantly changing CMM data

For example, lets start by tracking three datums

(1) Big Hole

(2) Little Hole

(3) Left Side

The sample output below shows the X & Y locations of the three datums

Original output file

	Big Hole		Little Hole		Left Side	
Nominal	-1.151	-5.112	-5.18	-4.512	-2.055	-1.2
Plus Tol	0.01	0.01	0.001	0.001	0.0015	0.0015
Minus Tol	0.01	0.01	0.001	0.001	0.0015	0.0015
Seq# DataType	1X	1Y	2X	2Y	3X	3Y
	-1.15239	-5.11411	-5.17995	-4.51195	-2.05531	-1.19964
	-1.15042	-5.11458	-5.18006	-4.51205	-2.05553	-1.19971
	-1.1515	-5.11174	-5.18006	-4.51205	-2.05529	-1.1998

Then imagine you are asked to insert a new datum (Medium Hole) into the CMM routine. Doing so will shift several of the columns to the right and adjust the numbering. Initially "2X" and "2Y" were associated to "Little Hole" in our original output file. But now "2X" and "2Y" are assigned to "Medium Hole". See the example below.

Output file after insertion of new datum

	Big Hole		Medium Hole		Little Hole		Left Side	
Nominal	-1.151	-5.112	-3.28	-3.612	-5.18	-4.512	-2.055	-1.2
Plus Tol	0.01	0.01	0.001	0.001	0.001	0.001	0.0015	0.0015
Minus Tol	0.01	0.01	0.001	0.001	0.001	0.001	0.0015	0.0015
Seq# DataType	1X	1Y	2X	2Y	3X	3Y	4X	4Y
	-1.15239	-5.11411	-3.27995	-3.61195	-5.17995	-4.51195	-2.05531	-1.19964
	-1.15042	-5.11458	-3.28006	-3.61205	-5.18006	-4.51205	-2.05553	-1.19971
	-1.1515	-5.11174	-3.28006	-3.61205	-5.18006	-4.51205	-2.05529	-1.1998