

Stephen A. Wise
Director of Statistical Applications

Oops! Did Your SPC Software Forget about the Process?

This article describes the problems encountered when using a single SPC file to collect data from a part that is run on different machines. This situation may create an SPC deployment that unintentionally reports invalid Cp/Cpk ratios and generate shop floor control charts with incorrect control limits. Failing to recognize the problem, most SPC software companies promote methods that place their customers' SPC program at risk. This major blunder may invalidate the SPC software you are currently using. After you finish reading this exposé, we invite you to call your SPC software's degreed industrial statistician to find out what they have to say.



If you think your SPC deployment is at risk, please read on....

Lets say that you've set up a part profile¹ within your SPC software. A profile is the front-end setup routine used to specify elements of an SPC data file. Not all SPC software calls their setup routines *profiles*. Within that profile you specify the following primary items:

- part number
- characteristic(s) to be measured
- specification limit(s)
- subgroup size
- tag fields (machine serial number, lot, operator name and so forth)

¹ A profile is a common name used to describe the front-end setup routine used to specify elements of an SPC data file. Not all SPC software companies call their setup routines *profiles*.

Once these fields are defined you begin collecting and charting data.

Sample	Part	Machine	Lot	Date	Time	Test	OD - Loc A
1	Blue Part	Lathe 167	49	7/13/99	3:03:17 PM	1	1.4976
1	Blue Part	Lathe 167	49	7/13/99	3:03:17 PM	2	1.4902
1	Blue Part	Lathe 167	49	7/13/99	3:03:17 PM	3	1.5029
2	Blue Part	Lathe 167	49	7/13/99	3:20:17 PM	1	1.5071
2	Blue Part	Lathe 167	49	7/13/99	3:20:17 PM	2	1.489
2	Blue Part	Lathe 167	49	7/13/99	3:20:17 PM	3	1.5051
3	Blue Part	Lathe 167	49	7/13/99	3:37:17 PM	1	1.4975
3	Blue Part	Lathe 167	49	7/13/99	3:37:17 PM	2	1.4984
3	Blue Part	Lathe 167	49	7/13/99	3:37:17 PM	3	1.4978
4	Blue Part	Lathe 167	32	7/13/99	3:54:17 PM	1	1.4998
4	Blue Part	Lathe 167	32	7/13/99	3:54:17 PM	2	1.5093
4	Blue Part	Lathe 167	32	7/13/99	3:54:17 PM	3	1.4999
5	Blue Part	Lathe 167	15	7/13/99	4:11:17 PM	1	1.4984
5	Blue Part	Lathe 167	15	7/13/99	4:11:17 PM	2	1.5018
5	Blue Part	Lathe 167	15	7/13/99	4:11:17 PM	3	1.4934
6	Blue Part	Lathe 167	15	7/13/99	4:28:17 PM	1	1.4948
6	Blue Part	Lathe 167	15	7/13/99	4:28:17 PM	2	1.4956
6	Blue Part	Lathe 167	15	7/13/99	4:28:17 PM	3	1.499
7	Blue Part	Lathe 167	15	7/13/99	4:45:17 PM	1	1.4978
7	Blue Part	Lathe 167	15	7/13/99	4:45:17 PM	2	1.502
7	Blue Part	Lathe 167	15	7/13/99	4:45:17 PM	3	1.4962
8	Blue Part	Lathe 167	54	7/13/99	5:02:17 PM	1	1.4928
8	Blue Part	Lathe 167	54	7/13/99	5:02:17 PM	2	1.5016
8	Blue Part	Lathe 167	54	7/13/99	5:02:17 PM	3	1.5008
9	Blue Part	Lathe 167	32	7/13/99	5:19:17 PM	1	1.4886
9	Blue Part	Lathe 167	32	7/13/99	5:19:17 PM	2	1.4963
9	Blue Part	Lathe 167	32	7/13/99	5:19:17 PM	3	1.5017
10	Blue Part	Lathe 167	49	7/13/99	5:36:17 PM	1	1.5016
10	Blue Part	Lathe 167	49	7/13/99	5:36:17 PM	2	1.4982
10	Blue Part	Lathe 167	49	7/13/99	5:36:17 PM	3	1.5001
11	Blue Part	Lathe 167	49	7/13/99	5:53:17 PM	1	1.4935
11	Blue Part	Lathe 167	49	7/13/99	5:53:17 PM	2	1.4933
11	Blue Part	Lathe 167	49	7/13/99	5:53:17 PM	3	1.4978
12	Blue Part	Lathe 167	66	7/14/99	11:03:01 AM	1	1.496
12	Blue Part	Lathe 167	66	7/14/99	11:03:01 AM	2	1.4953
12	Blue Part	Lathe 167	66	7/14/99	11:03:01 AM	3	1.4978

Table 1

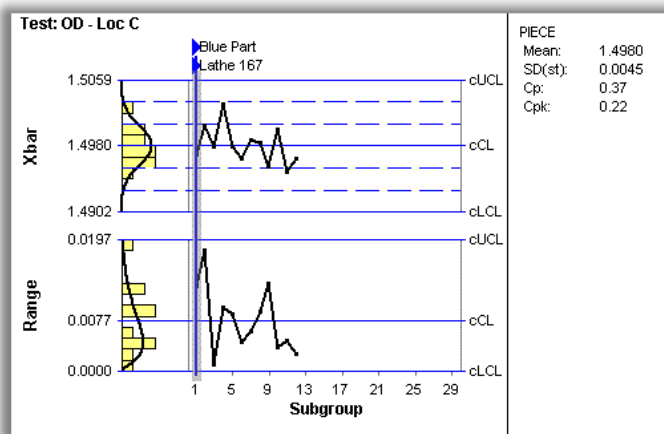


Figure 1

Did Somebody Forget the P in SPC? - continued

The run is over and everything is fine, you've got your raw data and control charts. Now, lets change the situation. A few days pass and the part needs to be run again, except this time the part is setup on a different machine (lathe 225). Switching to a different machine is no problem because you anticipated this. That is why the machine number was set up as a tag field. Collecting more data (starting with sample 13):

Sample	Part	Machine	Lot	Date	Time	Test	OD - Loc A
13	Blue Part	Lathe 225	15	2/13/00	4:18:50 PM	1	1.4975
13	Blue Part	Lathe 225	15	2/13/00	4:18:50 PM	2	1.4979
13	Blue Part	Lathe 225	15	2/13/00	4:18:50 PM	3	1.4985
14	Blue Part	Lathe 225	15	2/13/00	4:18:59 PM	1	1.4992
14	Blue Part	Lathe 225	15	2/13/00	4:18:59 PM	2	1.4957
14	Blue Part	Lathe 225	15	2/13/00	4:18:59 PM	3	1.4983
15	Blue Part	Lathe 225	32	2/13/00	4:19:10 PM	1	1.4961
15	Blue Part	Lathe 225	32	2/13/00	4:19:10 PM	2	1.5014
15	Blue Part	Lathe 225	32	2/13/00	4:19:10 PM	3	1.4996
16	Blue Part	Lathe 225	49	2/13/00	4:19:19 PM	1	1.5
16	Blue Part	Lathe 225	49	2/13/00	4:19:19 PM	2	1.4971
16	Blue Part	Lathe 225	49	2/13/00	4:19:19 PM	3	1.4969
17	Blue Part	Lathe 225	54	2/13/00	4:19:27 PM	1	1.4978
17	Blue Part	Lathe 225	54	2/13/00	4:19:27 PM	2	1.4986
17	Blue Part	Lathe 225	54	2/13/00	4:19:27 PM	3	1.5005
18	Blue Part	Lathe 225	32	2/13/00	4:19:37 PM	1	1.4962
18	Blue Part	Lathe 225	32	2/13/00	4:19:37 PM	2	1.5022
18	Blue Part	Lathe 225	32	2/13/00	4:19:37 PM	3	1.4984
19	Blue Part	Lathe 225	49	2/13/00	4:19:45 PM	1	1.5002
19	Blue Part	Lathe 225	49	2/13/00	4:19:45 PM	2	1.5
19	Blue Part	Lathe 225	49	2/13/00	4:19:45 PM	3	1.498
20	Blue Part	Lathe 225	54	2/13/00	4:19:52 PM	1	1.4967
20	Blue Part	Lathe 225	54	2/13/00	4:19:52 PM	2	1.499
20	Blue Part	Lathe 225	54	2/13/00	4:19:52 PM	3	1.4997
21	Blue Part	Lathe 225	15	2/13/00	4:20:02 PM	1	1.4982
21	Blue Part	Lathe 225	15	2/13/00	4:20:02 PM	2	1.4952
21	Blue Part	Lathe 225	15	2/13/00	4:20:02 PM	3	1.4974
22	Blue Part	Lathe 225	32	2/13/00	4:20:10 PM	1	1.498
22	Blue Part	Lathe 225	32	2/13/00	4:20:10 PM	2	1.4985
22	Blue Part	Lathe 225	32	2/13/00	4:20:10 PM	3	1.4964
23	Blue Part	Lathe 225	49	2/13/00	4:21:43 PM	1	1.4985
23	Blue Part	Lathe 225	49	2/13/00	4:21:43 PM	2	1.5009
23	Blue Part	Lathe 225	49	2/13/00	4:21:43 PM	3	1.4996
24	Blue Part	Lathe 225	54	2/13/00	4:21:53 PM	1	1.5009
24	Blue Part	Lathe 225	54	2/13/00	4:21:53 PM	2	1.501
24	Blue Part	Lathe 225	54	2/13/00	4:21:53 PM	3	1.4999

Table 2

Resulting control chart showing both processes:

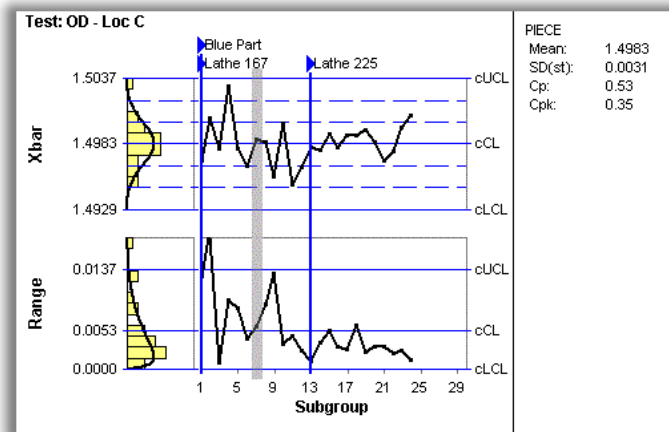


Figure 2

Something's Not Right...

At this point, the control chart contains data from both machines. Is this control chart correct...one control chart displaying data across more than one process? Some feel right. Being a good quality engineer, you do a little research. You go to your old SPC text books to look for control chart examples containing data from more than one machine. Much to your disappointment, all the examples assume the data came from the same process. If you were lucky enough to find a multiple machine example, the data were probably kept separate. That is, the differences between the two machines were illustrated using two control charts, two histograms or two box and whisker plots. You see, the professors know that different processes produce different statistical results. Therefore, mixing their data on the same control chart is not acceptable. Here is the root of the problem – shared control limits. If the data from two processes are plotted on the same control chart, both processes would be sharing the same control limits. The range chart must assume that the variability between the two machines are similar and the xbar chart must assume that the central tendencies are similar. Under certain conditions, these assumptions may hold true, but should be verified using only appropriate statistical tests such as the KW (Kruskall-Walis) test.

At this point, you have to make a choice – do you go ahead and keep collecting data from different machines in the same part file, or do you bite the bullet and set up separate files for each different part/machine combination. Here is the tradeoff: mixing machines in the same part file is good in a way because all the data from the same part is in one place, but this approach makes your control charts suspect and compromises the validity of the resulting Cp/Cpk ratios. On the other hand, setting up separate part files for each part/machine combination is very time-consuming and could potentially confuse the data entry users when trying to locate the correct file. But worst of all, the data from one part would reside in multiple files. Running customer Cpk reports would become a daunting recurring task.

Both choices have problems. This dilemma exists for one reason, the software developers simply forgot about the process. Yes, even though SPC stands for Statistical *Process* Control, the traditional SPC file organization logic ignores the statistical effect of the

process. If you step back a little and look at the traditional SPC file organization, you will realize they are Part/Test based. That is, the specification limits and control limits are unique for each part/test combination. Within the same part file, you specify the specification limits and control limits for each unique part/test combination. If the process changes, the control limits do not reflect this change. Surprisingly, the process used to create the item being measured is relegated to the tag field level. Basically, forgotten. Yes, you can sort the data by these tag fields, but the control limits are not specified at the tag field level. We all know that a different machine will produce different results. From a statistical perspective, different machines will generate different means and different levels of variability (sigma). Why then would an SPC software company manufacture and promote a product that ignores the process effect in the control limits? Without the process, SPC becomes Statistical *Part* Control. I think these so-called SPC software companies have some explaining to do!

Any auditor that has certified an SPC deployment that uses a software where the process is not an integral part of the control limits should re-think that certification. Any SPC software company that advises their customer to mix different processes in the same file by specifying the process as a tag field is providing advice that could lead to false conclusions about the outgoing product quality.

Is there are work around?

Yes. You could work around this problem by establishing a different set of control limits when ever the machine is changed. Although statistically correct, this approach has strings attached, 1) someone will actually maintain these control limit updates and 2) the software is capable of establishing different control limits effective at each machine change interval. Even though this may “fix” a given control chart, still little is known about a given machine because machine-specific data is interspersed among several part files. This approach is not statistical process control.

Any good industrial statistician understands this dilemma.

How could this blunder have occurred?

Here it is, the middle of 2000 and most SPC software still stores data using flat file mentality (a legacy carried over from their DOS roots). The solution (a software that provides both part control *and* process control) is too complex for flat files. Its not that they don't want to fix their problems, they simply cannot with their existing code. They're faced with a total re-write.

Is there a real solution?

Yes. The programmers and industrial statisticians that designed InfinityQSi SPC software have always understood the importance of the process. For this reason, InfinityQSi has never stored data using flat file logic and because of this, control limits are part/process/test based. That is, a different set of control limits can be stored to the database for each different part/process/test combination. For example the Outside Diameter from Part A run on Lathe 167 can (and probably will) have a different set of control limits from the same Outside Diameter from Part A run on Lathe 225.

Lets take another look at the two-process dataset, InfinityQSi allows you to display both processes on the same chart. If the two processes are similar, the same control limits can be shared. Otherwise, InfinityQSi will *automatically* change the control limits whenever the process changes.

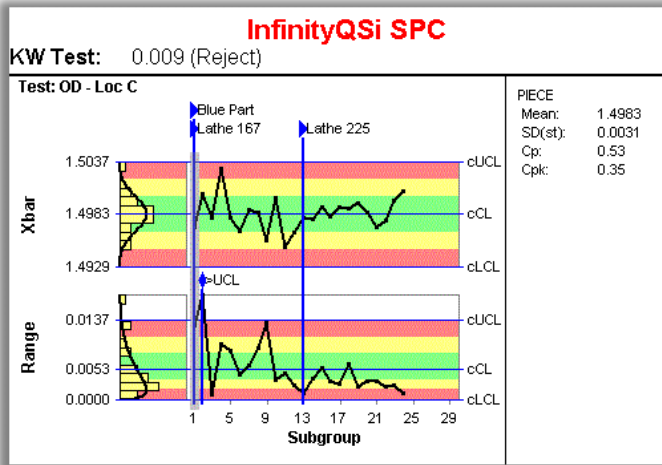


Figure 3

Notice that the KW test (a non-parametric statistic used to verify if the variability across both processes is similar) states to reject with a probability value of 0.009. This means that there is a 99.1% chance that the two process' variability is different. Therefore, the correct way to display this data is to use separate control limits for each process.

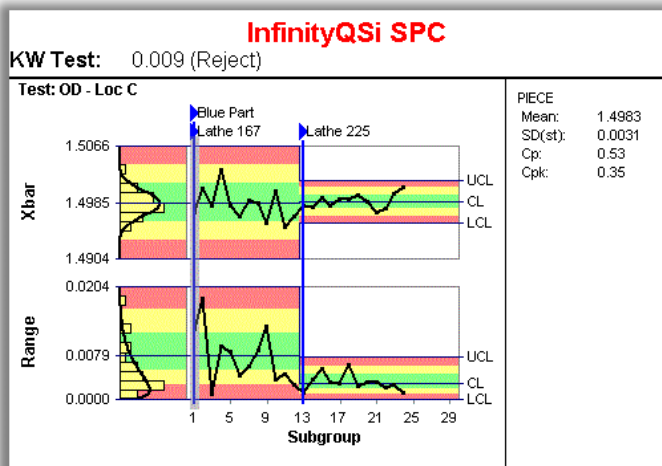


Figure 4

In Figure 4 you clearly see the differences between Lathe 167 and Lathe 225. This chart, however, is not a process control chart. It is merely a part control chart. To see process control, you need a way to display all the parts run on a single machine, regardless of feature size. This type of chart is easily managed within InfinityQSi because of the way data gets stored to the database (see the article, *Dirty Little Secret* found on www.infinityqs.com for more details).

Figure 5, a standardized control chart (plots data in units of standard deviations) shows how two parts, with different nominal sizes, ran on Lathe 167. There are hundreds of specialized control charts supported by InfinityQSi.

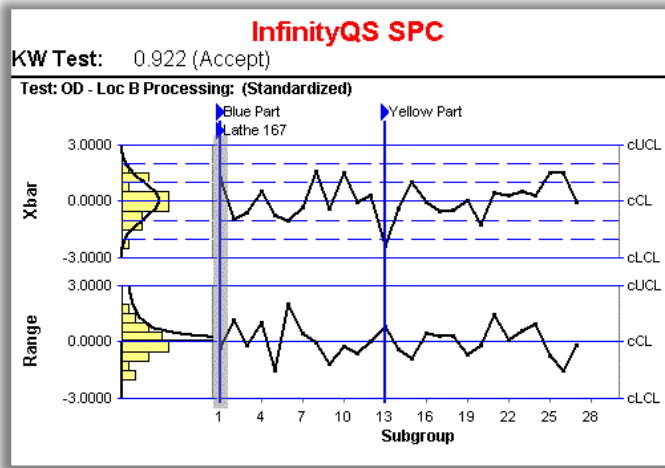


Figure 5

With InfinityQSi customers get the best of both worlds, they get part control analysis for their customer reports and process control analysis for themselves. No more dilemma, no more compromise.

<<>>

About the Author:

Stephen A. Wise is the Director of Statistical Applications at Lyle-Kearsley Systems. He holds a B.S. in Industrial Statistics. Prior to joining Lyle-Kearsley Systems, Mr. Wise worked as an industrial statistician at The Boeing Company, where he co-authored the groundbreaking D1-9000® specification, Advanced Quality System® for Boeing Suppliers. He is also co-author of the book *Innovative Control Charting*, published by ASQ Quality Press. Today, he helps companies on a daily basis implementing real-time production floor SPC and advanced statistical tools.