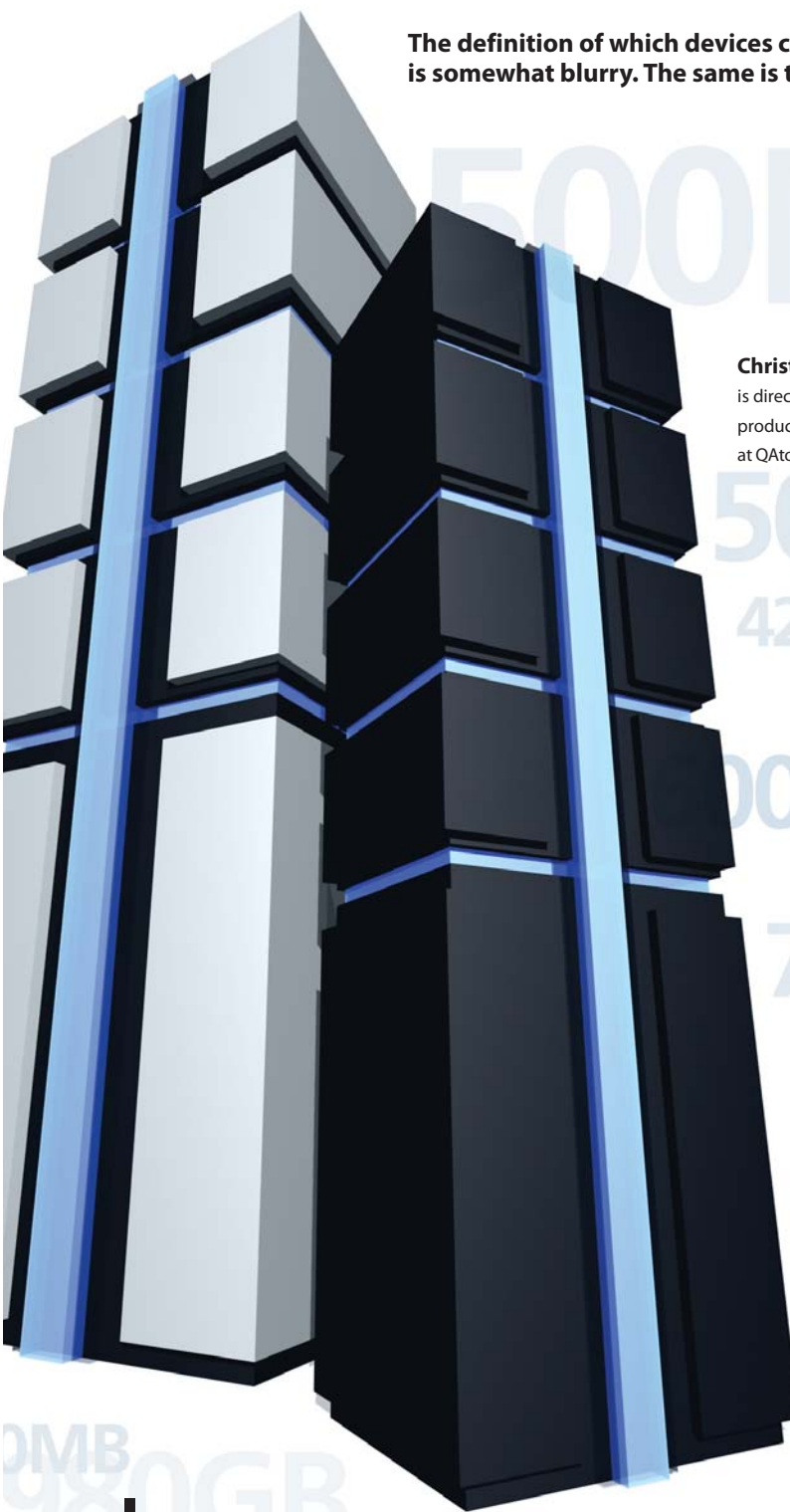


Regulatory Report

Like-for-Like Exchanges

The definition of which devices can be exchanged for others without revalidation is somewhat blurry. The same is true when it comes to computer systems.



Christian Stage
is director of QA
product development
at QAtor, Denmark.

In the pharmaceutical world it is generally accepted that a valve can be exchanged for another when it becomes defective, provided that the new one is exactly the same as the old one.

Occasionally, the supplier of the old valve no longer exists, but the same valve from the new supplier can be accepted as a like-for-like exchange. Also, valves that have the same diameter, are made of the same material and share other identical specifications are accepted too, even if it is a completely different valve. Personally, this is satisfactory as long as functionality is considered in terms of risk assessment.

But what happens when a computer system breaks down? Computer hardware tends to be unobtainable within a very short time frame. Furthermore, is it acceptable to exchange a Gb hard drive for a Mb one, or can the software run on a GHz computer instead of a MHz one?

Analysis

In my opinion, the most important tool for finding out how the exchange of a unit influences a system is risk assessment. For one client, who recently required assistance with their computer systems, we created a like-for-like standard operating procedure (SOP), which we took a little further.

We discovered that although numerous parts of the original documentation referred to a like-for-like change, nowhere was the term specifically defined. An effective and lasting solution was required. In this case it was computer-related systems that needed attention.

First, it was decided that the SOP should be as extensive as possible. The goal was to create an SOP that covered all changes of computer equipment and process logic controller (PLC) systems. But how possible is it to exchange systems or parts of systems for something equal, and what exactly is equal hardware?

In practice, it is quite difficult to find hardware that is exactly the same as that installed, and old software may not run on newer hardware. Therefore, system analysis is a good option. If the software requires Windows NT or newer it may not actually impact your system if you install Windows 2000 in place of the Windows NT. Also, if the software requires a minimum memory of 4 Mb, a new computer containing 512 Mb should not pose a problem, as long as the operating system can handle it. When talking about computer systems, we are, in fact looking at minimum requirements rather than specific requirements.

To establish what exactly the requirements are, some detective work is required at this stage.

Hardware. Gather the user requirement specification (URS) and

design specification to find out if they contain exact requirements regarding the system. Frequently, the hardware is poorly described, and the system software equally vague.

Software. Next, obtain information from the software supplier — this information is often found on the side of the original software distribution box or on the supplier's home page of their website. The text is often worded to the effect of: "The system will run on a computer with a 50 MHz 486 central processing unit (CPU) or faster, and minimum 64 Mb RAM." Unfortunately, these specs are long gone and the minimum available is a Pentium 4 and 1 Gb of memory, but the software may run successfully anyway.

Supplier. The next move is to contact the supplier. If, in their opinion (preferably based on a written

statement that could be attached to the like-for-like documentation), the software runs on the available hardware, we can proceed to the next step. If the supplier does not recommend a like-for-like exchange, the risk assessment has to be moved to a higher level (change request or change control case).

Risk Assessment

Almost anything can be 'passed' provided it is properly documented and based on adequate rationales. To document the considerations made, a risk assessment must be performed.

If professionals do the rationales that explain why the like-for-like is OK based on a predefined pattern, it will be possible to record them in a 'fill-in-the-blanks' template document, which makes it easily accessible. By doing this, we can consider all the general pitfalls that

Table 1 Sample rows from a sample SOP.

Req. #	Requirement	Method	Acceptance criteria	Observation	Accept [Yes/No]	Data	Signature
2.1	Minimum requirements exists for the system.	Find the documentation for the system application (SattLine, iFix, RSview32 etc.) and note the minimum requirements for the application system in question.	A list of minimum requirements are found in the rubric "Observation". Append a photocopy of the minimum requirements for the application software.	Appendix: The following minimum system requirements are observed in the documentation : CPU Hard disk CD-Rom Floppy drive Ethernet Ethernet I/F Communication interface Screen resolution Number of screen colours.			
2.2	Minimum requirements exists, URS (or IRS).	Find the user requirements (User Requirement Specification, Installation Requirement Specification or the equals) and note the requirements as listed.	A list of minimum requirements are found in the rubric 'observation'. Append a photocopy of the minimum requirements from the user.	Appendix: The following minimum system requirements are observed in the documentation : CPU Hard disk CD-ROM Floppy drive Ethernet Ethernet I/F Communication interface Screen resolution Number of screen colours.			

would occur during the actual implementation:

- hardware stress as result of use of increased system performance
- less security
- lack of documentation to perform the action.

Based on these considerations we can decide whether it would be feasible to do a like-for-like exchange, otherwise a change control case will be needed to be opened instead. As opposed to malfunction scenarios, it is possible to do the risk assessment in good time, and any slight indication

of problems regarding a seamless change of device/system/software will give the person evaluating the risks a better idea of choosing between the straightforward like-for-like option or a more paper-consuming change control case. Whatever the decision, the performed risk assessment may serve as an appendix for either, demonstrates the occurrence of thorough scrutinization and that rationales for either choice are in place.

Prewritten Protocols and Reports

To make the like-for-like exchange more easy to implement in real-life and to keep stress levels to a minimum, it is wise to prepare. A generic test protocol for verification of system settings and functionality (referring to the existing validation documentation for the system) should be possible to create. It is advisable to consider both physical settings, as mentioned in the system's design specification and verification of the installed software based on the system (according to the baseline for the system), and functionality tests (such as start of the application and communication towards other systems). Once the 'equalness' (like-for-like similarity) based on minimum requirements of the system are established, a final report can be written. If sufficiently prepared, a 'fill in the blanks' report may be produced prior to the test.

Important

At all stages of this system update it must be borne in mind that the like-for-like exchange is a 'cheap' version of the change control case. Provided that the considerations that the like-for-like exchange are based on are done before the exchange is necessary, it is quite easy to maintain a very high level of control, and, therefore, the 'validated state'. If the considerations are not made beforehand, it may be far more difficult to prove that full control has been maintained throughout the process.

Hands on

It may be worthwhile to build an SOP that contains:

- a prewritten generic document analysis (Table 1)
- a prewritten generic risk assessment
- a prewritten generic test protocol
- a prewritten summary report.

If these elements are predefined and the considerations are made beforehand, more degrees of freedom can be allowed. After all, that is what we are looking for, and with the authorities actually suggesting that we use the risk assessment in a more active manner, it might be a good way of being in compliant control, and fixing problems related to the age of the computer at the same time. It also complies with the life cycle approach, provided that we do make updates when necessary and the like-for-like exchanges when appropriate.

Conclusion

This topic is not debated nearly enough. In the area of computerized systems, there have not been sufficient considerations to decide whether like-for-like exchanges of equipment must be based on exact likeness or can be based on minimum requirement likeness.

I hope that this article gives your company a push in the direction of the latter, but urge you to make your own decisions based on risk assessment. [PTE](#)

