

Assessing the Assessors: Evaluating Quality Audit Tracking Software Systems

Ed Lyons

President

*Lyons Information Systems, Inc.
Raleigh, NC, USA 27603*

Summary

Effective Quality Assurance systems are the foundation for successful manufacturers. Beyond brand name, price, and market share, poor product quality can sink customer demand for a company's goods. Increased off-shore manufacturing, company mergers of products and information systems, quicker on-shelf finished goods timetables, and skyrocketing transportation costs have all added to the complexity and costs of quality assurance. QA software systems can help, but manufacturers must be able to assess quality software systems to make sure their needs are met. The first step is to develop a clear understanding of what is needed. Next, analyze the following topics: vendor relations, costs, solution fit, customization versus configuration, access, security, implementation, queries, reports, communication of performance, maintenance and support. Implementing a new software system involves managing change. Your software solution vendor can be a partner with you in this process. Asking the right questions and doing your homework will magnify your chances of selecting the correct assessor.

Introduction

Today, companies require a global means to accurately capture and analyze product quality audit information real-time so that problems are quickly identified and acted upon to save time and money. However, gaining a competitive edge by improving a product quality software system often involves purchasing, building, or rebuilding a system. Since company IT departments are often unable to assist users with the tasks of implementing new quality software solutions, in-house quality assurance professionals are left with the mission, although they often have little if any background in evaluating IT solutions. In order to provide support for these staff, this paper provides a discussion of points for non-MIS staff to consider when assessing quality audit tracking software systems operating in manufacturing environments.



It is important to begin by focusing on your goals. A successful quality audit tracking system targets the following major goals:

- report product quality problems early,

- provide consistency in quality processes and metrics,
- communicate quality expectations,
- contribute to increased profits,
- provide opportunities for companies to work more collaboratively with suppliers and business units.

First things first

Companies engage in a long and often arduous process before getting to the point of assessing potential solutions. This includes engaging all of the stakeholders in assessing whether to purchase a solution or build one in-house, evaluating the resources available within IT as well as in other parts of the organization, dedicating funds in the budget for the solution and its support, and so on. These processes are political, financial and managerial. These topics could be papers in their own right, so in order to reduce the scope to something manageable, the current discussion will be based upon the assumption that you have moved through these initial processes and are now at the point of assessing prospective quality software system solutions from vendors. Specifically, I base this paper on the following assumptions:

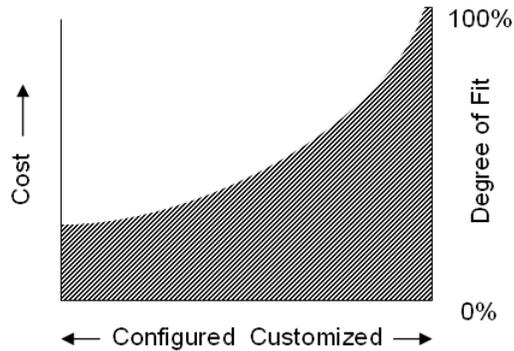
1. You have decided that you will search for a quality auditing system from software vendors who offer off-the-shelf, configurable solutions rather than build the solution in-house using your IT organization.
2. You have determined the extent to which you want a system that will interface with data that exists in data systems within your company that will feed the quality audit system solution.
3. You have developed a clear understanding of what is needed in the system. This includes a concise understanding of what quality assurance information should be collected, what reports are to be produced, and where the sources of data reside that will be used in the system.
4. You have identified who will use the system i.e., a single factory, multiple factories, suppliers, etc.
5. Important questions of ownership, support, and funding of the project have been evaluated.
6. You have gathered project buy-in from those who will use and support the system. These include the important players such as senior management and IT.
7. You have created a timeline and a budget for the project.

Now you are ready to go shopping and are prepared to evaluate the offerings of software vendors that you find in the marketplace.

First, write down characteristics and features of your proposed system that you cannot live without. This important step should emerge from the work you have done with stakeholders and

users while examining your business. These “must have” items will serve as a first filter to pass over initial vendors who court your business.

Confusion sometimes exists between the ideas of adding new capabilities to purchased software (*customization*) and *configuring* off-the-shelf software to business requirements. My experience with a hat stretcher that was used on a western hat that I purchased last year might help clarify the difference between customization and configuration. The hat was *configured* to fit on my



head by applying heat and pressure to change its circumference. The hat itself – its color, style, brim size and shape – were unchanged. However, I could have hired a hat maker, picked out a fabric, had my head measured, and spent much more money for a *customized* western hat built just for me. Similarly, configured software comes with a “built-in” approach to performing quality audit tasks. Although some parameters can be changed in the software, or configured, basically your business model must be able to comply with the software’s approach.

Customized software, however, is essentially rebuilt to better fit your business. In other words, if the solution source code is changed, then it is *customization*. If only environmental settings are changed, i.e., file paths, email server addresses, etc., then the software is *configured*.

A compromise must be made between cost and utility. Initially, customized software is typically more expensive than configured software, but experience has shown that it is generally easier to change the software than it is to change your business model. Over the long run, the cost of building a custom solution may well be worthwhile when matched against the costs and disruption of trying to implement a software solution that is a poor fit for your business model.

Everything is relationships

When implementing any new software system, you are not only confronted with business and technology challenges, but most important of all you are faced with managing change. It is imperative that you find a vendor who will partner with you in the change process and who listens well. Vendors who immediately try to persuade you that they have exactly what you need before hearing and understanding those needs should be avoided. Consider the following:

- How much time and energy has the prospective vendor spent in understanding and defining the unique requirements of your quality management operations? Is the vendor willing to meet with you to discuss your ideas and requirements? Do they make you feel like just another customer or do you feel that they really wish to solve your business problem?
- How flexible is their solution? Are they willing and able to customize their software to create a better fit or is only configuration available? If available, what costs are associated with customization?

- What level of experience does the vendor have in your industry? Have they created quality audit solutions for other companies like yours? How skilled do they seem at learning your business?
- What kind of warranty and after-sale user support is available from the vendor for their software? Often, software companies offer maintenance contracts to cover user and software support after the sale. These contracts are generally offered on a yearly basis and run approximately 14%-18% of the software price.
- Are upgrades to the product included in the yearly maintenance contract or treated as additional costs? Upgrades are usually defined as enhancements made to the base product and offered to all customers.
- How will user training be accomplished? What types of system documentation are available from the vendor?
- What recourse do you have if the vendor should stop supporting your product or cease doing business? Would you have access to the product source code through a software escrow account or some other means? Would your IT department have enough information about the solution to be able to support it?

No matter how complete the requirements planning has been, unexpected challenges will emerge during the development and implementation process. Count on it to happen. Here is where a critical ingredient of success is found. If both parties are as dedicated as they should be to achieving the same goal, then a give and take dynamic will occur so that unexpected requirements will be solved quickly and easily. Remember that you are looking for a software system solution and your vendor should be looking for a long-time relationship with a new happy customer.

What about costs?

Implementing a new quality software system involves costs in addition to the price of the solution itself. Four main cost elements generally make up the total cost of a purchased software solution: product cost, in-house costs associated with implementation, training and configuration costs. Beyond these costs, consider the following:

- If customization is required (and available from the vendor), how much will it cost?
- To what extent can the solution be configured in-house rather than requiring involvement of the vendor? For example, should you decide to install the solution at another plant site, will the vendor charge you for another instance of the product and therefore another license fee? Will their assistance be necessary to perform the additional installation?
- What does support cost? If a maintenance contract is not available, then what is offered in its place? Hourly fees for vendor support can add up quickly. To what extent is in-

house staff expected to provide level one support prior to contacting the vendor with user problems and questions?

- Is the system available as hosted, un-hosted, or both? An un-hosted system means that the software product and database(s) are housed inside of your company's data firewall and managed by your IT department. A hosted system on the other hand means that the system software and database(s) will be housed by the vendor or another service provider on their servers outside of your company's data firewall. Un-hosted systems place the responsibilities of database management and security on your IT department while hosted systems place those responsibilities on the vendor or service provider. One attractive aspect of hosted systems is the fact that your company IT department does not get involved in daily database backup, user access, and other system related issues. Also, the monthly application hosting or leasing fee is often reasonable.
- What are the IT costs associated with implementing the new system? As mentioned above, the cost of involving in-house IT staff with tasks such as consulting with the vendor about technical issues, writing database script programs to export legacy database information to the quality system, data reduction and download to the new system, etc. will contribute to the total cost. Generally, the internal IT costs can range from 2% to 9% of the purchase price of the software.

Where's the beef? The system solution.

Overview

Excluding cost, the essential question in choosing any quality audit solution is:

To what extent does the product fit our existing business model?

What items in your "must have" list are not met? To what extent can shortcomings be tolerated? Which expectations cannot be met?

The elements of a quality audit system and its ease of management must be considered when accessing vendors. Consider these functional areas:

Access

Consistency in data collection, analyses and reporting requires an enterprise-wide solution. Currently, the easiest way to provide enterprise-wide access to a centrally distributed and managed database is through a internet (or intranet) solution that runs on a company supported web browser, and with managed user access. Your assessment related to user access should include examination of the following questions:

- How is access to information within the vendor's solution managed? For example, may suppliers and others from outside of your company's firewall access the quality system and view only their specific performance information?

- How is system security managed? Are user login ID's assigned to individual users or may a group of users have a single login ID? How much effort is involved in adding and removing users within the system?

Setup

Many companies use spreadsheets to collect, analyze, and report their product manufacturing quality data. Information contained in in-house databases are sometimes combined in analysis and reporting processes. Setup involves planning how to access and load existing quality audit information contained in spreadsheets and legacy databases into the new system. Explore how vendors propose to load existing "core data" into their product without the need for users to manually enter the information into the system. Explore the vendors cost to assist in the process.

Data Entry

Meaningful data analysis only occurs when information is entered correctly, consistently, and in a timely manner. Ask your prospective vendor the following questions:

- May predictable choices be pre-loaded into list boxes and combo boxes for users to select during data entry to minimize manual data entry?
- Is data validated before it is saved? Systems should validate both the presence of required information as well as the integrity of what is entered.
- How easy may users navigate to the form(s) where data is entered? Must they drill down five menus deep to get to the desired input form?
- Is the user interface intuitive and simple to use?

Data Capture

Make sure that the solution captures the information you need. Examine the reports closely. Is important information absent from the reports? This could mean that the data is not being captured or is captured but not reported. Compare your "must have" list and audit data elements that you composed against the contents of the reports.

Reports

These make up the fruits of data collection. The primary reason for the costs and efforts expended on a quality audit system are meaningful reports. It's that simple. This may be one of the first places to evaluate when considering a vendor's software. Consider these points when reviewing the reports in a solution:

- Are they easy to understand? Most solutions come with "canned" quality reports while some also include a "report writer" which give users the ability to craft a more custom report with content which they choose.
- Are they meaningful? Can you answer your major quality assurance questions from the reports that are produced?
- May reports be made available to selected users?

- Is there a sufficient range of report criteria available to filter report contents?
- May users specify different formats for the output, such as HTML, Excel, MSWord, PDF, CSV, etc?

Queries

A query is a quick way of returning a group of related quality information. Queries allow users to assemble information that is usually not already shown in a report. Examine the query capability of the proposed solution and ask the following questions:

- How flexible are the querying capabilities? For example, may queries be sought on multiple fields of data?
- Are queries reported in formats that are easy to understand and communicate?
- May data that is returned from a query be sorted and drilled down to see further details?

Communication and Feedback

Audit measurements are indeed important, but an important benefit of a successful quality audit system is the ability to show change. This translates into demonstrating performance and cost savings. Therefore, quality audit results and quality expectations must be clearly communicated. You should ask:

- Is supplier performance easily communicated?
- May automatic notifications of non-compliance be generated and sent by the system? How easy is it to manage who gets notified?
- Can corrective actions be shared with supplier and others whose products are being evaluated?
- Is graphical product quality performance displayed easily and clearly?

System Maintenance

Quality auditing means that you are always comparing a specification (an expectation) with the actual (performance). Therefore, it is important to examine:

- How easily may specification requirements be changed?
- To what degree do you need the vendor's assistance in changing business rules, such as altering number of defects related to pass/fail decisions within quality audits?
- How scalable and flexible is the software solution to new business requirements?
- Will the solution operate on your company operating system or is new operating system software required to support the vendor's solution?

Conclusion

As you can see, implementing a new software solution involves managing change across many different areas. During the process of analyzing your business in preparation for purchasing a

new quality audit software solution, new issues may arise such as discovering inconsistencies in your existing business model; users may need to learn new technologies and new ways of doing their jobs; some jobs may become unnecessary and staff may become worried about their employment security; and access to improved quality data may drive changes in relationships with your suppliers.

Your software solution vendor can be a partner with you in this change process. Your selected software vendor should work in collaboration with you and your stakeholders to insure that your new solution fits your business model, allows for future change, and meets your cost and delivery expectations. Asking the right questions and doing your homework will magnify your chances of selecting the correct “assessor”.