



Software Quality Assurance

Process Improvement

Prepared for

Fluke Networks
Bridgewater, NJ

Brendan Heffernan

by Lisa Hauser

1.0 Introduction

The acquisition of LoopExpert Technologies, Inc. by Fluke Networks in July, 2003 and the recent addition of the Camarillo-based Harris Corporation operations in May, 2004 brought about many changes in our organization. One obvious change was the realignment of our development teams. These teams have continued work to maintain LoopExpert's solid reputation within the Telecom OSS market.

As part of Fluke Networks, we now follow the Danaher Business System (DBS), which strives to continuously improve quality, delivery, cost and growth by listening to the voice of the customer. According to Danaher, in order to achieve the goal of the DBS (world-class excellence in customer satisfaction) all associates should be involved with process improvement.

With this said, and the fact that we are still integrating with the Fluke Networks and Danaher way of business, now is an opportune time to review our software development processes for improvement.

You will be interested to know that an informal, internal review of the software quality process has been performed. Through interviews with team managers, research and a review of past experiences, I have evaluated the processes we currently implement for Software Quality Assurance (SQA). Specifically, I focused on processes related to documentation, training, interface design, usability, testing, production and other aspects supporting the overall SQA process.

This document summarizes my findings and proposes a broad plan for process improvement of the SQA aspects relating to documentation, training, user interface design, usability, testing and production processes of our overall software development process.

1.1 What is SQA?

Software Quality Assurance is the aspect of software development and engineering that is dedicated to ensuring the deliverables from a development project meet acceptable standards of completeness and quality.¹ According to Roger Pressman, a well-known software engineering consultant and author, software quality is:

Conformance to explicitly, stated functional and performance requirements, explicitly documented development standards, and implicit characteristics that are expected of all professionally developed software.

In other words, it is a system that is of high quality, delivered on time, within budget, and most importantly works properly. However, the system that is delivered includes more than just the executable code.

The entire range of deliverables that result from a development project also include the written specifications (narrative descriptions of the requirements and structure, graphical models of the system, and process specifications), system designs, test plans, source code documentation, user/installation manuals, online help and training documents.

In essence, SQA functions as a support mechanism for the development team, **providing not only the deliverables, but also the process for delivering quality software**. It provides the means to deliver software with quality built in, rather than tested in, at the end of the development process.

From an economic perspective, building quality into software makes sense. Many projects cost much more to complete than originally estimated. According to Karl Wieggers, this is due to the fact that rework costs are rarely, if ever figured into the budgets of time and resources needed for the overall project. In most cases, the cost of the project is based only on the initial build of the system, not the amount of money or time it will take to fix errors in the system or maintain it in the future.

1. Karl E. Wieggers, Ph.D. from *Software Errors: Prevention and Detection*

So, in order to build quality into software, these basic SQA goals should be acknowledged:

- Requirements Analysis
- Software Specifications
- Design
- Coding
- Testing
- Maintenance/Support

In addition to the basic goals outlined above, the following aspects of the software process are equally important and should not be overlooked:

- Documentation
- User Interface Design
- Usability
- Training
- Production

The above list outlines areas of the SQA process that are often taken for granted or given the least priority, though they are indeed a crucial part of the support mechanism that makes up the SQA process as a whole. More importantly, deliverables such as end-user documentation, training, interface design and usability are often the only things our customers base their opinion of the FNET brand and quality of our products on. Unfortunately, first impressions are difficult to alter, which makes delivering quality products even more crucial.

The advantages of improving our processes will result in products that are predictable, consistent, and of high quality. By making development activities repeatable and measurable, we will be able to gauge the progress of projects easily, enabling us to perform risk management confidently in the future.

Improving upon a few of the SQA goals listed or even the most important aspects of them will enhance our overall software development process and ensure that we will deliver the best possible product on time and within budget. Thus, we can continue to exceed our customer's needs and expectations.

2.0 Current Processes

Our current SQA processes were evaluated via interviews with team managers and associates; research regarding best practices for software development, quality assurance, technical communication, and knowledge management; and review of past experiences. Strengths, weaknesses and areas in need of improvement were identified within the SQA aspect of the current software development process.

Though the entire SQA process as a whole is important, this document focuses on the processes related to the following:

- Documentation
- User Interface Design
- Usability
- Training
- Testing
- Production.

2.1 Documentation

Much progress has recently been made regarding the standardization of the documentation process used to create user guides, installation guides, online help and training materials. These improvements bring us much closer to delivering consistent and high quality documentation to our customers.

2.1.1 Strengths

A style guide, templates, and sample documents have been created and are located on a shared network directory for use by documentation authors. Adobe FrameMaker is now the sole publishing software used to create all user documentation, which is now delivered only in electronic format.

In addition, a new process for creating online help within our products has been reviewed, tested, and approved for use, which brings us one step closer to the single-sourcing arena. A utility called WebWorks Publisher that integrates with FrameMaker has been purchased and used with much success. A template has also been created and is ready to be rolled out to the documentation group. Now the responsibility for the creation and updating of help is under the documentation authors, not the developers. This

streamlines the process by making the creation of documentation and help a simultaneous task, which as mentioned above, is a first step towards single-sourcing.

2.1.2 Weaknesses

Though the resources needed to produce documentation have been created and shared, there are still issues that need to be addressed in implementing the documentation process across the OSV offices. A communication avenue and training on how to implement the templates and style guides for both documentation and online help still need to be organized. Training on how to use FrameMaker and WebWorks Publisher also needs to be scheduled for those not proficient in its use.

Monthly meetings are set to be scheduled to begin addressing these issues and to improve communication between the various authors across the OSV offices. With time, communication and training, our documentation will eventually take on a solid, consistent look and feel that will reinforce the FNET brand.

2.2 User Interface Design

Similar to our documentation, our user interfaces have recently seen a vast improvement, specifically our web interfaces. A handful of web-based products have been developed using a new FNET web interface design that has received much attention internally and from our customers.

2.2.1 Strengths

The new FNET web interface design has not only improved the appearance of our interfaces, but has also improved the consistency of the FNET look and feel across all of the products in which it has been used. Thus, we have taken the first steps in coordinating our software product line. Additionally, an icon library has been compiled which makes it easy to take inventory of and search through icons and splash screens for our desktop applications.

2.2.2 Weaknesses

Creating the design for the new FNET web interface was just the beginning. We still need to communicate the new web interface standard to the project managers, web developers, and other OSV associates. A basic standards document has been created and a cross-functional user interface team has already begun monthly

meetings to slowly grow the standards by tackling one focus target each month. Training on how to utilize the new standards is set to be scheduled in the near future.

Once the web interface standards have been implemented, the team will then focus on creating standards for other user interfaces such as desktop systems, hand-held devices and test equipment. Additionally, a graphics library that will integrate with the already existing icon library is set to be created and shared.

All of these improvements will eventually strengthen the FNET brand and pull all of our software interfaces together under one consistent look.

2.3 Usability

Usability and user interface design go hand-in-hand. Unfortunately, usability is an area of our overall development process that has not received much attention in the past. However, usability, like our user interfaces and documentation are often the things our customers base their opinion of the FNET brand and quality of our products on. Thus, it is crucial to put just as much emphasis on it as we do our other deliverables.

2.3.1 Strengths

In order to steer us into thinking about usability, conversations have been sparked with FNET associates in various areas of OSV about how to best capture the Voice of the Customer (VOC). For example, several conversations with our Sales Associates, Project Managers, and Harvey Trager (DBS Trainer) have taken place about how to put a process in place to capture the VOC. Of particular interest is obtaining feedback and approval from the actual users of the system (not just the managers or project managers on the customer side) at every phase of development.

To kick off our focus on usability, two field trips have been scheduled to meet, observe and talk with the users of a few of our systems. Though a small step, it is a step in the direction of getting to know our users and listening to the VOC in order to build high quality software that meets their needs.

2.3.2 Weaknesses

Though there is currently no process in place to capture the VOC and realizing that every project is different, a goal has been set to begin an information exchange via face-to-face contact with the users to better understand their needs. This includes performing needs analyses and interviews; creating storyboards and prototype walkthroughs; and conducting evaluations with the users. It also means examining the purpose our products serve and how they impact the way our users perform their daily jobs. Part of this examination should be determining if a product is replacing a process that they currently do manually, replacing a system, interfacing to an already existing system, or is a completely novel application. Depending on the situation, we can gather crucial information from the user about how to smoothly integrate their methods and procedures into our software.

If we start an open dialogue with the users where they feel involved with the design of our products from beginning to end, then ultimately we can best mesh their needs with our functionality.

Ideally, a usability team should be formed to focus on the VOC and the process to be created to best meet our customer's needs, specifically regarding the integration of usability and the VOC into our user interfaces.

2.4 Training

Product training is another high-impact deliverable that easily influences how our customers perceive the FNET brand and the quality of our products. It is important not to undervalue the effectiveness of our current training initiative.

2.4.1 Strengths

Currently, I am aware of one dedicated trainer and several ad hoc trainers that provide instruction on our software products. These individuals either have an intimate understanding of the system they are training on or have a telephony background that aids them in their training activities.

2.4.2 Weaknesses

Though they get the job done, we could improve on this focus area in a number of ways. To my knowledge, we have never done any

post-training evaluation to rate the effectiveness of our training sessions. We also have not formally identified standards for training. Currently, the person performing the training creates their own lessons, slides, and other training materials independent of other trainers. A standardized process and the creation of a training team would increase the effectiveness and consistency of our training. It would also improve communication and sharing of best practices as well as provide a forum to review how our trainers organize and conduct training sessions. There is also interest in educating the identified trainers on cognitive strategies and conditions of learning for adult learners, something that is standard knowledge for almost all corporate trainers. Additionally, it would be beneficial to explore the feasibility of Web-Based Training (WBT), which is quickly becoming a common means for delivering product training.

2.5 Testing

Testing obviously plays a large role in delivering quality software. Unfortunately, limited resources and time often prevent our development teams from putting the necessary focus on this important SQA function.

2.5.1 Strengths

We have, in the past, successfully delivered our software to the customer in a timely manner. When resources were available, we also delivered test plans and performed some form of testing before delivery.

A few development teams do have informal testing processes in place. This includes the utilization of issue tracking software, the creation of test plans for function and validation testing (delivered to the customer), and the testing of the systems themselves. Testing is broken out into back-end and front-end activities. Developers and a few rogue associates perform the majority of the back-end testing for the most part, with front-end testing performed by documentation authors, trainers, and/or the user interface designers.

2.5.2 Weaknesses

Due to our lack of resources, testing processes are almost non-existent, communication between testers is weak, test documents are

inconsistent, and resource allocation is inefficient resulting in testing being performed “by the seat of our pants” or not at all.

To change the way testing is performed, associates who perform these functions need to be identified and brought together. By meeting on a regular basis, team members can start to build a testing process. Via communication, sharing of best practices, and creation of processes/standards they can start to tackle issues causing inefficiencies. With time, we would be able to strengthen the testing aspect of the overall SQA process.

2.6 Production

Production processes include those areas of the SQA process that result in the physical deliverables made to the customer. This includes CD/tape labels, delivery documentation, testing of installations/upgrades/releases, and the preparation of shipping materials.

2.6.1 Strengths

Though not much thought is currently put into production, these activities are performed regularly. Currently, a couple associates serve as key contacts that assist project managers organizing deliveries. They update and print labels, copy data to CDs and tapes, test installations, and assist with packaging.

2.6.2 Weaknesses

Just as testing is done “by the seat of our pants”, our production activities follow a similar pattern. There is no formalized process followed, so the way one project manager coordinates a delivery may be different than that of another.

To alleviate discrepancies, a production process needs to be written and communicated. As part of this process a delivery checklist should be followed for every delivery. In addition, label-making processes should be created, along with a label library and a customer contact list.

If these few things are implemented, then our production activities will become more consistent, making our deliveries part of the overall process rather than an after-thought.

3.0 Proposed SQA Process Improvements

Where do we go from here?: A Broad Plan for Improvement

Recognizing the strengths and weaknesses in our current SQA process, specifically regarding the areas outlined in this document, has proven useful in determining the areas in which we should focus process improvement initiatives. Prioritizing and selecting a few of the most important needs to initially improve upon will enhance our overall software development process.

The main goal of this document is to spark thought and discussion on the best ways to go forward with process improvement activities. This is just the inception of a long-term process improvement plan that will integrate with the overall SQA process, DBS and other FNET processes. With open communication and coordination, we can work toward slowly building a stronger overall SQA process.

3.1 Organization of Technical Communications Group

Reorganizing our resources to improve certain aspects of the SQA process is one proposed solution for process improvement. Realigning the people that work directly with documentation, interfaces, training, and testing into a Technical Communications Group (TechComm) is a logical step. By bringing these associates together under one collaborative team, we will be better able to manage and allocate our already limited resources.

There are many benefits to aligning a cross-functional team. First, it alleviates the project managers from having to worry about parts of the SQA process that are not directly related to development. For example, the project manager would no longer have to manage resources for documentation, interface design, training and testing. This would be the TechComm manager's role.

Another benefit would be that project managers would no longer have to "steal" resources from other teams. In fact, the TechComm team would eliminate down-time for associates. Instead of the associate dropping from the product manager's radar and having nothing to do (as often happens now), the TechComm manager would be able to manage the associate's assignments, even pairing associates up on collaborative projects to get a job done faster.

Cross-training of skills would also be an advantage. Team members would be responsible for several focus areas, allowing them to actively participate in several aspects of the SQA process. Building a skills matrix, utilizing people's strengths and developing their weak areas will grow a solid collaborative team.

These are just a few of the advantages of forming a collaborative, cross-functional team focused on documentation, design, training and testing. Outlined below is a breakdown of the purpose, strategy, and focus areas for the proposed alignment of the TechComm Group.

3.1.1 Purpose

To create, communicate and manage high quality information related to:

- Documentation
- User Interface Design
- Usability
- Training
- Testing
- Production

3.1.2 Strategy

Create a cross-functional team with a common vision centered around the Voice of the Customer.

Field the best team via:

- Common Vision
- Communication
- Processes/Standardization
- Training/Education
- Resource Allocation

Create sub-teams based on focus areas, utilizing the strengths of every team member while building-up areas of weakness, eventually creating a solid team capable of delivering products that are of high quality, on-time, within budget and meet the needs of our customers.

3.1.3 Who

The team would consist of current associates already involved with documentation, interface design, usability, training, testing and production activities. TechComm as a whole, would be comprised of static and revolving members. A core team of static members supported by revolving team members would make up the group. The cross-trained static members would perform most of the core functions on a full-time basis. The revolving team members would move in and out on an “as needed” basis performing specialized tasks while their primary responsibilities would remain with their main teams.

3.1.4 Focus Areas

Documentation

The creation of user guides, installation guides, help, test plans, standards/processes.

User Interface Design

The creation of user interfaces that adhere to the design standards, are consistent, usable, and match requirements.

Training

The delivery of product training including the creation of lesson plans and training materials.

Testing

The testing of systems including the creation of test plans and the use of issue tracking processes.

Production

The production of labels and delivery documentation, testing of installations/upgrades/releases, preparation of shipping materials, and the adherence to a delivery checklist.

4.0 Conclusion

Part of SQA is making sure the deliverables from a development project meet acceptable standards and quality. Recognizing that the deliverables for our projects include more than the executable code, we can now begin to look at the whole SQA picture. The SQA process also includes the documentation, user interfaces, training, testing and production processes, all of which have a direct impact on how our customers perceive FNET and the quality of our products.

With customer perception and first impressions being so persuasive, it is crucial that we value the VOC. This document outlined the strengths and weaknesses in our current SQA process. We need to prioritize and select a few of the most important needs to target for improvement. If we take these first steps, then our overall SQA process will improve and ensure that we will deliver the best possible product on time and within budget while continuing to exceed our customer's needs and expectations.

It is hoped that this document will spark thought and discussion on the ways to go forward with process improvement activities. One of those being the alignment of a Technical Communications Group dedicated to the improvement and roles related to documentation, user interfaces, training, testing and production.

This is just a small part of a long-term process improvement plan that will integrate with the overall SQA process, DBS and other FNET processes. With continued communication and coordination, we can work toward slowly building a stronger overall SQA process.