

"Bridges To Trust: Achieving Corporate Expectations within a Skeptical Environment"

by Richard Inch, Technical Documentation Manager, Leybold Inficon, Inc., Syracuse NY

Given at CNY STC's October 1999 monthly meeting

Contact Info: rinch@inficon.com (phone: 315-434-1100)

Think about this title for a moment: Can there be anyone in technical communications who has not faced exactly this challenge in their jobs? When Technical Documentation Manager Richard Inch joined Leybold Inficon in Syracuse several years ago, he faced a daunting, dual challenge: winning the trust of a skeptical staff of engineers who were accustomed to doing all their own tech writing, and planning and implementing a new, long-term strategy for managing Leybold Inficon's technical communication operations.

At our October 20 meeting, Richard explained some of the tough challenges, novel approaches and successful procedures he implemented at Leybold Inficon—actions that worked there, and may also be helpful to you in building a more tightly-coordinated technical communication function within your company.

Transcript:

Leybold needed someone to write a software manual, and I was brought in as a consultant. Working in my cubicle for a year, I realized that I was being tolerated by the engineering staff in my group, but it was clear that rest of the company's engineers really didn't like technical writers. They had had a lot of past problems with on-staff technical writers who had not met deadlines yet didn't appear busy, didn't seem to do a whole lot and were constantly hounding the engineering staff for information. Leybold had finally eliminated their in-house tech writers and, for 14 years, their engineers had become accustomed to doing their own technical publications.

Step One - Building Relationships

I spent that entire first year working on building relationships. I made friends. I went to lunch with people. I plugged myself into their community as best as I could. My Marketing Communication Manager observed that I was winning some people's trust and asked if I would like to manage all technical communications, acting as the company's sole technical documentation person and designing a system from the ground up which would help their engineer-authors and ensure that no product would ever be late to market because of lack of a user guide. I accepted this daunting challenge.

Initially, the situation was not good. Without a Technical Communication Manager to lead them, the engineers had descended into writing documentation using every writing tool imaginable - Starwriter, WordPerfect, and almost every version of Microsoft Word ever released. It was incredible! Also, there was no network at Leybold. My previous Syracuse employer had a very high tech shop: I was used to working in Interleaf on a SUN 4 system running over a full UNIX network. Leybold had seen the work I had done on that system, told me they liked my work and that this was exactly what they wanted. They then sat me down at a 386 PC with 8 megs of ram and a 9-pin dot matrix printer, located in a room that was essentially a closet with a desk, chairs, coat racks and other junk. This was my new office!

While Leybold's publications looked pretty good at the time, there were no review systems in place, no organized methodologies for producing technical documentation. Engineers did their own writing, put it on floppies, and carried it to a graphics person in the Marketing area who ran it through Pagemaker or Express, or whatever she felt like using at the time, to give it a corporate

standard look, then sent the files to the printer for output to paper. The graphic artist was not a technical publications person; she never read or edited anything. The resulting manuals were hard to maintain because since each was done with a different program, you couldn't easily identify specific chapters.

A positive was that the engineers were accustomed to working as teams, and the product managers were used to writing articles for magazines. They were good writers and understood structured writing, but some were overconfident. As soon as I got a decent computer, and Leybold installed a computer network, I began working over the network with people on projects. I took the approach that we already had 220 good writers on staff; my job was to put a methodology and a set of tools in place to help them be good writers.

Step Two - Finding the Right Tools

At this critical point, the company happened to decide upon the adoption of Microsoft Word as a corporate standard word processor. This was a stroke of luck in that without any influence from me, it forced them to abandon their old tools and settle on a single standard tool. While everyone was still grumbling about the changeover, I grabbed the opportunity to do lunches with them, pushing the use of templates and structured documentation.

During this period, I gave Word the very best shot any human could be expected to give a software program, but it didn't work! Word is an illusionary program: When you first use it, you become excited by its possibilities, but a few weeks down the line, all the fancy features only get in your way, and beyond that, the features don't hold. Graphics start moving and links shift each time you close and reopen the document.

Step Three - Setting Standards

I tracked my time and discovered I was losing 8 hours a week playing with Word. I brought this to my manager's attention, and we ended up bringing Framemaker in-house. Today, we're an all-Framemaker company. All technical publications are done in Framemaker, assembled over our network.

I began to establish content, style and graphics standards. The first thing I would do is build a road map in the form of a Table of Contents. This arduous stage is indispensable if you're going to produce effective documentation and not waste time.

Once we have a living-document Table of Contents, I use it to identify the primary tech writer, the engineering project manager, the marketing product manager and the quality assurance engineer. With their help, I then identify all the Subject Matter Experts (SMEs) responsible for each Table of Contents topic, and call the first team meeting.

Next, we map out three formal reviews for the project (fewer, if the project is very simple). Work begins, and to manage the timely submission of materials, I employ a work-in-progress directory on our network. I give team members read/write access to their files.

Step Four - Getting Engineering to Budget Time for Documentation

A key step which really marked a turning point for me was when I was able to establish a procedure saying that "...the engineering project manager will include time for user guide preparation as a separate line item in each individual's project development schedule."

It took a tough fight to make this happen! The Project Managers did not want to commit any of their engineers' time to technical documentation! It took meetings with my boss plus the two division Presidents and the most powerful individual in the entire company-the Chief Financial Officer-to make this happen.

My argument went as follows: "You have been doing technical documentation anyway, for 14 years.

Have you just been pulling it out of thin air? Have you been losing time? Have you been cheating on your time schedules? Are you forecasting correctly?" Then the people above us picked up the ball: "How have you been accounting for your time? Exactly where have you been charging this time? It's obvious that you've been charging your time wrong."

So, I said, "Let's allocate things properly." Very soon the engineering managers realized that this could be advantageous: if writing was charged against the manual budget, more time would be freed up for project design.

Step Five - Becoming an all-PDF Shop

Another key step came when we decided to make the changeover to PDF-based documentation. We had previously spent a lot of money developing one razzle-dazzle online program that users could click through, with lots of interactive links. But when we asked our customers, they said they didn't want that kind of interaction.

At that time, I had just gotten Acrobat and a clean room manual project came in. I knew in my heart we would end up with a PDF-based manual for this product because it's enormously costly and difficult to make a paper manual you can take into a clean room: it takes special paper, binders, ink and wrappers. I saw PDF as a way to get manuals into the clean room, saving lots of printing costs and making customers happy. The PDF manual was given to a customer, who promptly demanded that all future manuals come to him in PDF format only.

This has worked out so well that now we post PDFs of all our manuals on our intranet. Field service and customers alike can instantly be on the 'same page' and everyone loves them. We are now in the process of converting all our engineering documents to PDF.

The system we follow is simple. Everything is written in Framemaker. We convert Framemaker document to PDF files - some for online viewing, others for sending to our printer (because we print exclusively from PDF now, too.) We also automatically generate online help systems for our software manuals directly from Framemaker, by porting our Framemaker files through Omni Systems' MIF-to-RTF converter and then feeding the RTF directly into the WinHelp compiler.

Step Six - Backups

A final key step was for me to start relying on our Information Systems for all backups. With all our publications up on the network, I can now dare to let our IS people handle backups. Our fiber-optic network is so fast, they can do this easily. Also, we keep a second set of all our PDF files offsite, at our printer's location, for added security.