Quality in Practice: Improving Customer Satisfaction at a Software Support Call Center[i]

In this case study, we describe how one company improved customer satisfaction statistics at a software support call center by 43% in one month in an industry where the monthly norm is a low single digit percentage. The company, which provides technical support services to software publishing companies, had managed to slowly improve customer satisfaction figures over a three-year period. The improvement had occurred through efforts initiated by management to implement common sense business practices such as consistent follow-up, better technical training, and improvement of what they call the personal factor. But the improvement had plateaued and a crisis emerged -- a major customer, who accounted for 45% of revenues, threatened to pull the account unless further improvement was made.

Facing a potential disaster, management formed committees of technical support staff and asked them for suggestions. The results were initially discouraging. Suggestions only mirrored those already thought of by management. The logjam was broken when someone discovered the work of Gary Klein.[ii] Klein specializes in the anatomy of expert decision making with the aim of developing ways to speed up the training process. He reasons that experts in high-pressure emergency occupations are often unable to explain how they arrive at a decision because the process by which decisions are made is mostly intuitive. Although incidents of help line support do not ordinarily qualify as emergencies, the company assumed that this use of intuition was the reason even expert staff members directly involved in customer service could not explain their success. The best support engineers, like experts in other fields, were usually unable to completely explain their reasons for success in making good instant decisions. The company carefully analyzed calls, looking for patterns that would explain the success of the most successful support engineers. Once they could effectively describe these patterns to the non-expert support engineers, they were able to improve their customer satisfaction ratings quickly.

They began by interviewing experts in a department that supported a popular word processing package. They could do no better than management in suggesting new

approaches. They were, however, able to examine more closely the faulty assumptions implicit in the suggestions. The suggestions fell into two groups -- more adequate training and more personalization with the customer.

The suggestions in the first group rested on one basic assumption: Since the customer's objective is to get the right answer, technicians could produce satisfied customers by simply giving the right answer. This assumption seemed reasonable because all technicians could recall certain types of problems that could be resolved instantaneously, and customers were always very satisfied with these calls. There were, however, three problems with this solution:

1. Some technicians of below average technical ability nevertheless consistently earned high customer satisfaction ratings. Even more puzzling, some technicians earned high ratings even when they could not supply the correct answer. Paul, for example, sometimes spent days on a problem before informing the customer that he could not supply a solution, yet customers usually gave him high satisfaction ratings.
2. Customers frequently gave low satisfaction ratings despite the fact that they received the correct answer. Take, for example, Joe, a technically competent support engineer who solved a customer's problem with a complex development software product in 10 minutes. This was excellent service for the level of complexity involved, yet the customer complained that the service was inadequate because "it took too long."
3. Even with the best training, most technicians need some experience with troubleshooting before they can give correct answers. Beginners, then, will always degrade the overall performance of the group. This can produce a significant hit because turnover rates in the software support business are frequently high (almost 50% in our case), and some products may require up to six months of ramp time

The company concluded, that although the correct answer could be important under certain conditions, another factor seemed to be involved -- the personal factor. These suggestions assumed that the technician did something on a personal level; such as using a particular tone of voice or a friendly greeting that satisfied the customer. One successful technician said it was just a matter of "loving the customer to death." The company had already heavily emphasized the personal factor. Management had prominently displayed signs in the office that read, "Treat the customers as you would like to

be treated," "Smile because a customer can hear your smile" and "Don't say no." These types of guidelines and suggestions comprised 60% of the procedures listed in the employee manual. Yet these apparently did little for customer satisfaction ratings.

In the ensuing research, 20 engineers responsible for supporting a well-known word processing package were chosen out of 200. Were divided into two groups -- experts and subexperts. The first group was comprised of those engineers who achieved an average customer satisfaction rating of greater than 74 percent during a six-month period. The second group achieved ratings below 75% during the same period. The company taped and analyzed two support calls per engineer. Because the engineers were not chosen randomly, the focus was not on gathering a statistically significant sample, but trying to discover what information would be most useful and effective for training support engineers. They compared and contrasted the two groups to seek out the distinctive factors that led to the higher satisfaction rates of the experts, and validated the initial results with interviews of supervisors and the support engineers themselves.

The greatest surprise was discovering how little the personal factor counted. They found that those experts who do establish a more personal relationship fare no better than those who do not. One engineer acted almost robotic while another took every opportunity to use the customer's name in an effort to be more personal. Yet these differences did not affect the ratings. It seemed that short of openly obnoxious behavior, the personal dimension was irrelevant. This result surprised us until they thought about it further. It appears that the customer seeking help for a technical problem has no reason to expect someone who can or will display social skills. Rather, he or she simply wants a solution to a technical problem. That this is often not possible because of the current low quality of software support facilities only makes a correct answer even more appreciated. This idea led them to the central method that all experts used to solve problems whether they were personable or not, which they called the transparency principle.

When following the transparency principle, the expert support engineer signals throughout the call that he or she is constantly applying troubleshooting skills to the customer's problem. Some of these signals are obvious --restating the problem in the beginning, asking questions to clarify the problem, and giving explanations. Some are less obvious -- admitting that a wrong course of action has been recommended, or simply

grunting something like "uh huh" instead of restating the entire problem initially. This method is simple but powerful – so powerful in fact, that when it is properly executed, the customer may walk away highly satisfied even after the engineer has failed to solve his or her problem because the customer believes that the engineer did everything possible to solve the problem. They called this the transparency principle because the expert gives the impression that he or she is not hiding any thoughts and actions from the customer.

Although simple in principle, this method is more complex in practice. Following is an example of how it~ works when the engineer must probe a moderate amount to find a solution for a caller.

Stage 1- Immediate establishment of status of engineer as a problem solver and expert. The expert engineer communicates that he or she understands the-problem the customer is describing. The standard previously set by management was that the engineer should explicitly restate the problem. Only about half the experts in this study did this consistently. The other half simply issued a series of uh huhs. The important thing is for the engineer to issue audible signals as evidence of paying attention to the customer's words.

This technique may present a problem for the beginner who encounters a particularly complex situation. In this case, the engineer should act exactly as an expert who has encountered a difficult problem by asking the customer to describe the steps that led to the problem. The engineer should then explain that this exercise may lead to the discovery of a simple mistake that has a simple solution.

Stage 2: Initiating the troubleshooting. After completing Stage 1, the engineer must quickly show evidence of mental activity directed toward problem resolution. There are several ways to indicate this activity, and the engineer may use any combination of them:

1. The engineer asks the customer to do a series of operations on his or her computer. The commands are issued rapidly and without hesitation. The implicit message is, "I want you to do x, y, and z to verify a hunch I have about what is wrong."
2. In a variation of No. 1, the engineer offers an explicit diagnosis before issuing the commands. This sets the expectation that the engineer has a definite solution in mind.
3. In a variation of No. 2, the engineer gives a hint that he or she is thinking of an as yet ill-defined solution by talking to the customer almost as if he or she were thinking

aloud. The engineer then issues the commands. This sets the expectation that he or she might have a solution.

4. The engineer cannot think of a way to attack the problem and must consult information databases or other technicians. He or she apprises the customer of this so the customer understands the reason for the long silence. If Stage 2 leads to a solution, the call produces a satisfied customer. If not, Stage 3 begins.

Stage 3 Open admission of a wrong hypothesis. The engineer immediately shows he or she is working on finding another strategy by recycling through a variation of Stage 2.

The research showed that the difference between the expert and subexpert groups in terms of adherence to the transparency principle was large enough to suggest training. The effects of the training were immediate and dramatic -- a 90 percent satisfaction rate one month after the training. This was 43 percent above the average and 27 percent higher than ever before. The training brought a gratifying result and an unexpected surprise -- the experts themselves also improved significantly. It seemed that once the high performing experts realized the reasons for their success, they used this knowledge to do even better.

Joe, the support engineer mentioned previously, exemplified this improvement. Despite his expertise as an engineer, at least 20% of the surveyed customers gave him a low rating. After learning about transparency, he realized where he went wrong. Whenever Joe put a customer on hold to research a problem, he did not apprise him or her of his plans. The customer, therefore, had no idea why he or she was on hold-at times for as long as 10 minutes. From this point on Joe made sure always to describe his research plan before putting the customer on hold, and he found that the customer sounded more relaxed when he returned to the phone.

Key Issues for Discussion

1. What does this case suggest about the importance of understanding customers' true needs and expectations?
2. What are the implications of the "transparency principle"?

3. Joseph Juran has long advocated the concept of studying the best people in an organization as a method of learning to improve others' skills. How might the learning from this case be applied to other organizations?

---

[1] Adapted from Charles Palson and Dale Seidlitz, "Customer Satisfaction At a Software Support Call Center," Quality Progess, June 2000, 71-75.

[2] Gary Klein, *Sources of Power: How People Make Decisions*, (Cambridge, MA: MIT Press, 1990)