Managing expectations
By: Harris Kern's Enterprise Computing Institute

Each year *Information Week* takes a poll of its readers on the top issues affecting CIO's and IT departments. Each year the issue of aligning IT to the business is at or near the top of the list. Key to the issue of aligning IT is the ability to maintain credibility with which requires properly managing customer expectations.

We have watched many IT organizations struggle to gain and maintain credibility with their customers. We hear the frustrations on both sides – IT managers frustrated at the pressure and lack of appreciation from their customers, and customers frustrated at the slow pace of systems implementation and poor quality of service. These disconnects are disappointing but are often understandable given how IT professionals are often unable to effectively manage their role of service providers by aligning their customers expectations with their department's ability to deliver services. Here are four lessons; we have learned in working with IT departments and our own customers.

Attitude
Attitude is everything in business and in life, and there are hundreds of books available on how to build and maintain a success-oriented attitude. We will not duplicate that literature, nor is it our place to help you feel good about yourself. Managing your customers is essential to your success, however, so we need to cover a few basics.

IT professionals often disconnect with customers early by not having the right perspective. IT professionals consider themselves computer scientists rather than business people or service providers. We are in the service business, not the science business. Your customers care as much about computer technology as automobile drivers care about combustion technology. It's your job to align yourself with your customer's needs, and explain your services in *their* terms, not yours.

The second attitude disconnect is to not understand the basic human psychology of expectation and satisfaction. Keep commitments. If you promise five widgets and deliver six, the customer is happy. If you promise five and deliver four, they're disappointed. They may get over their disappointment and adjust to the reality, but you won't get full credit for your hard work.

Resist the temptation to alleviate short-term stress by over-promising future products.

Planning
In managing expectations, planning means the art of anticipating your customers' next moves. Woe to the IT manager who is blindsided by a key customer who announces one day he is going to consolidate 12 warehouses into three during the next 6 months, or is doubling the size of his organization in the next 60 days. We are continually amazed by the number of IT managers and CIO's who have no idea where their business is headed or how to respond when it takes a sharp turn. This is a recipe for disaster, because the last thing you

want to be is on the critical path to business success. The pressures and "assistance" from upper management can ruin even the best IT shop.

How can you prevent these kinds of disasters?

Understand your business and your industry. Read the trade press and understand the forces that are acting on your industry. Listen to your internal customers, learn their plans and concerns, and find what keeps them up at night. This sounds sophomoric and preachy, but it seems that many companies do not involve their IT departments as an element of their business planning. Without integrating IT, IT can become a bottleneck. Through planning, you can meet customer expectations and become strategically valuable to your business.

Communicate, communicate, and communicate

You cannot manage expectations without communication. In fact, you cannot manage *anything* without communication. If some is good, more is better. The customer continually needs to know how you are making progress on their problems, what you will deliver, and what you need. Don't let daily pressures crowd out the continuing dialog that is essential for success.

Don't confuse customer contracts with communication. Having a signed requirements document and an approved budget is only the beginning of communication not the end. Never let your team wander off into a vacuum to "do their thing" and deliver when the product is ready. Insist on frequent short deliverables from your team, and insist on customer involvement at each delivery however small. Without continuing discussions, IT inevitably diverges from customer wishes and leads to the traditional IT complaint, "We gave them just what they asked for, but it wasn't what they wanted!"

Communication forces alignment at two levels – the IT professionals with their customers, and just as important, the customers with themselves. Customers almost never have a clear idea of the details of what they want, and if they get it, they will change their minds.

As much as IT managers complain about indecisive and demanding customers, there are sound reasons for this behavior. First, people almost always interpret their needs in terms of what they know. They are often best able to define their needs in terms of what they don't like rather than what they do like. The old saying "I'll know it when I see it" is literally true.

It would appear that the best approach to satisfying the customer is to allow them to shop for the best answer. But IT managers are co-developing new systems and introducing new technologies, and don't have the time or budget to allow their customers who demand to see the final product, critique it, and then discard what they don't like. This process is a recipe for delay, budget overruns, and frustration. By negotiating continually over the features, functions, and priorities and acceptance of a new system, the IT manager can manage customer expectations to ensure timely completion within budget.

Communication is the biggest problem within IT today. IT simply doesn't know how to effectively communicate with their customers nor do they effectively communicate within

their own organization. Communication is not highly regarded as a priority and mind you effective communication takes time!

Speed
Long-term IT projects are the exception rather than the rule, and one sure way to disconnect with customers is to begin a long-term, monolithic project developed on the traditional "waterfall" life cycle. Such a process assumes the product is a point solution that once specified can be developed in a vacuum, which is never true. Successful IT systems must evolve with the enterprise, which means that we assume that they will change continuously to meet customer's evolving needs. If this is true, customers will be happiest if the product is a continuous string of small incremental deliverables, developed quickly enough that the customers cannot become dissatisfied. This implies that systems are always under construction, and that the development process involves constant rework. With older technologies this was not possible, but with modern objects, messaging, and multi-tiered architectures, it is possible.

Two examples to prove our point: one a success and one a failure. One of the best client/server system developers we know set a standard with his customers that his development team would make any customer-requested change of a minor nature in two weeks. Major revisions would happen at 3-month intervals. He would meet with the customer, learn of their complaints, and offer them a prototype solution in a week. If they liked it, he would put the change in production in another week. Customers never had time to be dissatisfied. His was the most successful system in the department, although the CIO was frustrated because it was never "finished".

By contrast, we learned of a large utility that spent 5 years and tens of millions developing a new customer information system – including a year just to test it. The slow pace development meant that by the time the system was completed the business environment for the utility had changed. Deregulation is eminent and the system is not capable of accommodating the new reality without extensive rework. Customers are demanding major changes and large portions of the system are now being rebuilt at great expense.

The project managers and CIO should have *never* agreed to spend so much time in development.