

Daniel Shefer  
[www.shefer.net](http://www.shefer.net)

[DS\\_PM/delete-this/spamex.com](http://DS_PM/delete-this/spamex.com)

## **Disruptive Customer Demands: When, Why and How to Deal With Them**

All Product Managers experience product-related requests from customers. These requests help align the product with customer's needs, shed light on how they use the product and generally improve the attractiveness of the product to its target market. However, sometimes, these requests become disruptive. This can happen when providing a solution is done outside the product roadmap. Many product organizations do this because the "request" includes a threat that unless the feature is added, changed, fixed etc., the customer will not buy the product, will stop testing it or discontinue its use.

This article discusses when and why customers make disruptive requests, how to deal with them and how minimize their disruptive impact.

### **Disruptive Demands During the Sales Cycle**

*"New feature requests are basically like the beginning of a new sales cycle. The request needs to be addressed like an objection."*

*Merrill R. (Rick) Chapman, author of [In Search of Stupidity: Over 20 Years of High-Tech Marketing Disasters](#) and [The Product Marketing Handbook for Software](#).*

### **Corporate Culture and the ShriII Factor**

Sales reps mirror the corporate culture they are part of. If a sales rep feels that the only way to get things done in her company is by yelling and pounding her fists on the table, then this is exactly how she will behave when under pressure. When this is the environment, every rudimentary feature request turns into a "I must have this to close the deal". Dealing with these requests becomes harder and the first job of the Product Manager is to do triage – figuring out which request is really critical and which is not.

### **"I Need This Feature to Close the Deal..."**

Sales reps tend to see everything from the perspective of their last sales call. The feature they "need" to close a deal may be forgotten tomorrow and it's the Product manager's job to add perspective to the request. When a sales rep comes from a prospect saying, "they won't buy unless we add this or that feature", someone has to act like the adult and call a spade a spade if it's not really critical. The PM has to ask - Does the customer really need this feature? What is the customer really trying to achieve? Is there a workaround within the existing product? Etc.

*"The most common reason for "need this feature" during a sales cycle, in my experience, is that the sales person is not selling into the chosen target market (or the company hasn't focused on a target). So often, we build a set of features for a specific market and those features don't completely satisfy another market. Or, we don't have a market in mind and we build a generic product with 80% of what people need but not 100% of what anyone needs. So sales people sell the next release hoping that the missing feature for this customer will be there. So selling futures means: pick a target market, meet their needs, and enforce selling*

*into that market. One president was great at this. He said, 'Sell to whomever you want, but I only pay commissions on deals in this target market.' And we had no problems; imagine that!"*

*Steve Johnson, VP, Pragmatic Marketing*

### **The Value Proposition is not Clear**

When the customer declares that he will not buy the product unless a certain feature is added this means that the customer does not see enough value in the existing offering. This is a problem with communicating the value proposition of the product to the prospect.

*"One way to deal with prospects who make feature requests that the sale supposedly hinges upon, is to return to the customer and say: 'let's get up and running and we'll give you this down the road'. We have found that stressing the current value of the product and having short release cycles, we can present the prospect with a roadmap that they are comfortable with without disrupting our development cycles."*

*Parker Harris, SVP Products, Salesforce.com*

On the vendor's side, the willingness to add such a feature has consequences that go beyond responding to customers' needs. Many times the willingness to be "flexible" is an expression of lack of internal consensus as to where customers derive value from the product.

### **Pressure to Perform**

When sales reps are under a lot of pressure to perform, they apply this pressure wherever they can. While some pressure is necessary to get the sales force to perform, too much will have negative effects. The main system used to pressure reps is the quota system. Reps will feel additional pressure if there is a large disparity between their quota and their pipeline. Reps may react to this by increased sensitivity to their customers' requests and any potential problem with an account, and specifically product requests, is immediately declared a major fire.

### **Disruptive Demands After The Deal was Closed**

#### **The Product Champion Speaks Loudest**

Part of the issue of "critical" feature requests is the psychology of the product champion. For example, during a pilot, an Interwise prospect was using the product in front of a group of people. When she did something unconventional and it didn't work as expected, she was embarrassed. Had this not been the product champion, the incident would have resulted in a mental note not to do this again and as a standard request to change the feature's behavior. However, due to the embarrassment involved, the incident resulted in an unequivocal demand for a change before purchasing the product.

## Putting Out Fires

Good account managers are aware of how the customer uses their product. By gently pointing out the product's limitations and available workarounds, they can help avoid any misunderstandings. If they let customers inadvertently encounter the product's limitations, they can cause small fires. These in turn may become major fires where the emotional volume of the customer is much louder than it should have been. At this point the customer becomes hypersensitive and a simple request turns into a disruptive demand.

## The Critical Feature That Wasn't

Now that the new feature was added, the customer may refuse to roll out the new version. There are two reasons for this. The first is that the critical feature, really wasn't critical in the first place. For whatever reason, the customer demanded a feature that they didn't really need. Delivering this feature is a failure of Product Management. The second reason involves the difficulties in rolling out a new version of software in large organizations. Enterprise customers hate rolling out new versions more than once a year due to the overhead and threat to network and system stability. Therefore, if the feature is not all that "critical", the new version won't be rolled out and will wait till the next maintenance round.

## How Selling to Large Customers Impacts Their Demands

### Disparity of Company Sizes – Who's In Charge?

In most markets, when there is a disparity in size between a large customer and a small vendor, the customer will have significant leverage over the vendor especially if the latter is new to the market. This impacts both the dynamics of the sales cycle and the nature of the product requests. The larger company will tend to flex their muscle in the buying process even if just to show who is driving the sales cycle. Large customers also tend to have a "vision" of how the product should work for them and they are more than willing to pressure the vendor to fill that "vision".

*"Being a small company selling large enterprise systems is a very difficult proposition to sustain. Large customers expect to get their vision of the product. Small companies, who many times live and die by each large deal, have difficulty in pushing back at these requests and are caught between the devil and the deep blue sea. If they give the customer what they want, they may continue to fight the next day but will have disrupted, sometimes severely, their own vision and roadmap."*

*Bill Peyton, Principal, IP-Digital*

### Large Customers Don't like "Me Too" Solutions

When selling to large companies, many times they get satisfaction knowing that the vendor created something special for them. This is a cultural issue – buying a "Me too" product vs. wanting to be unique. As the customer senses their leverage, they will tend to use it to drive the product in the direction they want.

## Bully Mentality

If the customer's corporate mentality is that they can achieve results by threatening, this will manifest itself in the buying process. The moment they sense reluctance from the vendor to come through with a feature, they will quickly revert to bullying.

## Other Factors That Impacts Disruptive Demands

### The Length of the Release Cycles

Requests from customers come at any time in the release cycle. With long release cycles, vendors deliver new features after a long delay. This tends to give customers the feeling that the vendor is not responsive to their needs. If the pressure to deliver is too great to bear, an interim release is created to satisfy the customer's demands. This interim release is inherently disruptive to the development and release processes.

*"A proven way to deal with disruptive customer requests is to have a time-based release schedule with bite-sized projects or milestones. To make this work, requirements gathering must be continuous and effective. Time-based releases help prevent feature creep and instill trust in customers that the next release will be on time and with the promised features. Furthermore, this track record can be used as a sales advantage."*

*Alyssa Dver, VP & Chief Marketing Officer, Sedona Corporation, author of ["Software Product Management Essentials"](#).*

### Lack of Vision

Customers like to feel that they are driving the vendor's product roadmap. Just as dogs sense fear in humans, customers can sense a lack of vision in their vendor and will attempt to drive the product where they think it should go. This is a double whammy as adding new features drives up the development costs of the product and reduces the ability of the vendor to respond to future feature requests. Another fundamental issue is that the customer may be driving the vendor in a direction that they are not interested in moving in.

## Recommendations

### Processes

- There is no release process that is a panacea for all companies and situations. However, there is evidence that a timed-release schedule (vs. a feature based release plan) with bite sized milestones works to create customer willingness to wait until the next release as well as confidence in the company's ability to deliver.
- Handle requests for changes to the product through the company's consulting/services arm and make sure that the customer pays for any changes to the product. If the customer does not share the burden of the cost, nothing will stop them from demanding features, regardless of the feature's importance to their use of the product.

- Handle requests for changes to the product roadmap with a disciplined process. The highest product authority in the company should be involved in all decisions that impact the product roadmap.
- Don't devote resources to make changes to the product before the customer signs the contract. A sales rep's promise that the customer is "just about to sign" or that "this is a closed deal" should leave you unfazed. Many things can and do go wrong at the last minute. A sales rep's word is not money in the bank. Just a reminder – if the contract includes a commitment to a future feature, this will prevent revenue recognition for part or the entire contract until the feature is delivered.

### Product Architecture

- Build the product so it sits on top of an API. This way, changes can be made by a third party or a services arm without disrupting R&D. By using an API, the code does not have to be torn open for every change.
- Build the product in a modular fashion. This will save a lot of effort when changes have to be made and the product will be more resilient to change.
- ASP based applications have a distinct advantage over boxed software because making changes to them is much easier and they do not require downloading patches, shipping CDs or installing new versions locally.

### Other

- If it's decided to deliver a patch for the customer, it must be agreed upon, in writing, that the customer will roll out the new patch within a prescribed timeframe. Also, the customer should agree to act as a reference if the new feature is relevant to other prospects. Requesting such an agreement has the additional benefit of cooling the customer's enthusiasm about demanding the patch.
- Entering a market with a high volume and low price point product has an advantage in this context. When customers buy low cost items, they expect less so their requests tend to be less disruptive.
- If your product has "gaps" in its functionality, prepare pre-defined workflows for your customers and train the product champions accordingly. This will prevent surprises down the road.

### Summary

Customer requests can be a double-edged sword. On the one hand they can help point the way of where the market wants a company to go. On the other, requests can become disruptive and distracting. By understating the factors behind customer requests, the dynamics of the relationship and how these requests impact the process, companies can channel the "request energy" into positive channels leading to a better product that customers are excited about and willing to pay for.

## **A Note to My Readers**

I'm excited to see that my readership is global and I'm eager to hear from you. Please tell me - who you are? Where you are from? What you do? Your thoughts on the article? Etc.

Drop me a note at: [DS\\_PM@spamex.com](mailto:DS_PM@spamex.com).

This article and its contents copyright (c) 2003 by Daniel Shefer.