By Sharad Sharma

Sharad has left Symantec recently where he was the India GM and VP – Product Operations. He is taking a planned break before his next assignment. He was earlier a co-founder and CEO of a successful wireless infrastructure startup. The views expressed here are his own. He can be reached at sharadsharma_dps@hotmail.com

Product Managers' Changing Relationships

One big opportunity is to turn product engineering from a cost-center model to a revenue-sharing model.

The relationship between product management and product engineering is a complex one. At times there is some healthy tension between the product manager and the engineering manager. Yet good collaboration between the two is essential for success. It is somewhat like what happens in the U.S. Congress. Most key bills have two sponsors – a credible Democrat and a respected Republican – working across party lines. Sarbanes-Oxley Disclosure Act, McCain-Feingold Campaign Act or Bayh-Dole Patent Act are all results of such collaboration. In the product world, most product strategy decisions or feature calls also require a product manager and a engineering manager to come together. This kind of successful collaboration takes real work. This is particularly the case when the product manager and the engineering manager are not co-located physically.

Whether or not we like it, a new organizational architecture is taking hold. It calls for keeping product management and other customer-facing jobs close to the key markets and the engineering management. Most recent startups, called "micro-multinationals" or "hybrids" have embraced this trend. So have mature multinationals. Many of them have product management in the U.S. and the engineering team and its leadership in India or Israel. The new norm will be to have a product leadership team that is geographically dispersed.

So increasingly product managers have to work with remote engineering managers as partners. While this makes a McCain-Feingold type of collaboration more difficult it also presents opportunities.

One big opportunity is to turn product engineering from a cost-center model to a revenue-sharing model. This is particularly attractive for products with revenues of about $20-$100 million. Here is how it works. Typically when you transition an engineering team to India you can do that to either your own captive development center or to a foster-home run by one of the many product engineering services companies like Wipro or MindTree. In case you choose the foster-home option, then you can pay that foster-home not by the traditional time-and-material basis but on a revenue share basis subject to a certain floor. The floor guarantees a minimum engineering headcount that will be maintained at all times. Now there are two things that result from this. One is the normal reduction in the R&D-spend as a percentage of revenues falls. Say the R&D-spend was 20 percent earlier; now it can be half due to the lower-cost structure in India. Second, there is a powerful new dynamic to re-invigorate the product. The engineering team in an India foster-home starts looking at ways of changing the trajectory of revenues. My experience has been that the teams in foster-homes are a lot hungrier than the teams in captive development centers provided the business incentives are setup the right way.

The revenue-share foster-home model is picking up momentum because strong indications are that it helps revive a lethargic revenue line. The shared incentives between the product manager and the foster home help speed up the upgrade cycle and that usually generates more revenues. In other cases, there have been dramatic improvements in technical support through the creation of internal supportability features. Of course the new revenue-share foster-home model isn't a panacea. If the product doesn't suffer from a lethargic revenue line and has a good line-of-sight between new features and incremental revenues then this revenue-share foster-home model isn't particularly attractive.

We are in the midst of a major change on how products are developed and sold. On the development side there are the forces of globalization and open source that are playing out. On the selling side, software as a service is a disruptive force. In this state of flux those who take a thoughtful approach, innovate and assimilate the learning of others will win.