

Tally Ref Product Nation

Bharat Goenka(Tally Solutions) talks to us about the company's 'stubborn' decision to stay focussed on products - ProductNation

pn.ispirt.in

11-14 minutes

Bharat Goenka is the architect of what is arguably India's most successful business solution — Tally. Co-Founder and Managing Director of Tally Solutions, Mr. Goenka developed the famous accounting solution under the guidance of his father, the late Sri S S Goenka. Today, the product is the de facto accounting solution for many SMEs and Mr. Goenka serves as an inspiration for many aspiring software product entrepreneurs. In an interview with pn.ispirt.in, Mr. Goenka talks to us about the company's 'stubborn' decision to stay focussed on products, the non-DIY nature of the Indian SME and the necessity for product companies to stay focussed on the product mentality.

Tally is one of India's most successful product stories, and it definitely appears to have ticked all the right product story boxes: responded to a genuine market need, stayed focused and evolved with the needs of users. Given the benefit of 20:20 hindsight, would you have done anything differently?



POWER OF SIMPLICITY

The reality is that one doesn't really learn from the past. We continue to do audacious things, we continue to get some success out of that as well as failure. Over our 25 year history, this has happened multiple times. Multiple times, we have taken a decision and it has gone wrong — but if the circumstance arose again would I take the same decision? In all likelihood, yes — I would have no reason to expect success, but I'd still have the optimism and think just because it went wrong in the past doesn't mean it also has to go wrong this time. So although I would say it's unlikely that one would have really done anything different, I can give you an example of a decision not working out for us. In 2004-2005, we changed the price of the software from 22,000 to 4,950 thinking that we would be able to sell software as a commodity. The reality was that for that time, it was difficult to sell software as a commodity in India in the B2B space. And so we suffered, massively. That proved our belief that we couldn't sell software as a commodity,

but it didn't stop us from trying. We lost almost 50 crores in those one and half – two years, so I would say our single biggest mistake was that.

Tally – or rather Peutronics — was founded in 1986 at a time when much of the Indian software industry's focus was on services. The decision to remain a product company when the tide seemed to be going the other way couldn't have been easy – why did you make this decision?

Actually when we started off, virtually every company had a product. Whether it was [TCS](#), [Wipro](#) or [Mastek](#) — everyone had a business product. The shift to services took place in the mid-90s, particularly towards the edge of the Y2K environment. We were one of the few stubborn companies who believed that while there was a lot of money to be made in services, we would never be able to address a lot of customers. So the mandate with which my father and I started the company in 1986 was that we were going to change the way millions of people do their business. We were clear that by moving to services, we would never be able to achieve the objective. We were unclear how long it would take us to get to a million — 25 years later, we are still trying to reach even the 1 million mark. But in 1986 we were clear that we want to be able to touch millions of customers. Therefore we remained focussed on our product line.

So what was that inspiring moment for you? Did you wake up one morning and decide that this was what you wanted to do — to change the way these millions of customer did their business, or was it a gradual evolution?

In the months before we got the product Tally out, one was into the product mindset but for developing systems related products like compilers and operating systems. So I was preparing myself to do those kind of products. At that time, my father was searching for a business product for our own small-scale industry business. He examined multiple products, but couldn't make sense of any of them. He very famously said: "When I'm buying a car I want to be a driver and not a mechanic." Similarly, he was looking for a product that would help him run his business — not his computer! Every product that he was looking at required him to change the way he thought about his business. So because I was interested in software, he said these guys can't do anything can you do something? So I was trying to solve his problem. After six months of development, I would say that it was his inspiration and thinking that formed the idea and belief that the product should be something that the country should also use.

The belief is that Indian SME's need to be "sold to" – the job that's conventionally handled by IT resellers who are critical to Tally's business model. What are your thoughts on the changes that Cloud technology might bring to this scenario, with the whole "self-service" angle coming into play?

India is not a DIY country, and this is unlikely to change in the SME sector.

The way the market works in India is like this : SME's expect people to come and sell something to them, even if it's bottled water. You expect it to be delivered, and you expect to pay for it in a different way. In India, SME's behave identical to the way enterprises behave abroad. Abroad, SME's behave identical to consumers. That's why in most MNCs, you see that the SME and SO/HO market being handled by a common head while the enterprise head is separate, because they need to be sold to. In India — actually, in all developing markets —

the SME and the enterprise behave similarly. In the west, the cost arbitrage of selling to a business is so high that the small business has no other option but to behave like a consumer. In developing markets, the cost arbitrage is low enough to send people to do the sales. And therefore, the buyer expects someone to come and do the sales. It is not about whether the visit is required because of the software complexity or the commercial complexity — it is an expected visit.

In your opinion, what are the three most common things that mislead or cause the downfall of Indian product companies today? What advice would you give them to overcome these?

I think it would boil down to one — which is to be clear about which business you're in.

Most people believe they are in the business of making money. Okay, even I am in the business of making money but my point is this: you can never be in the business of making money, you have to be in a business — money is an outcome of that. To explain it better, imagine that you are a software developer who wants to start your own product company. Capital costs are not very high — a single computer will cost about 20k, and assuming you develop the skill, it will some months to develop a software, and you'll get your software out. You might put together an infrastructure, sales people etc and you'll put up a monthly expenditure of about 25 – 30k. You start seeking customers — you find me. You sell me your product for say 10k. In all likelihood, I bought your product because I like your software development style and perhaps your product solved two or three problems I had — but I still have twenty more. Now because I like your software development style, I'll ask you to do more work for me. I might ask you to expand the product features, solve some HR problem that I have which this software doesn't solve and I'm willing to pay you for it.

Your first ten customers will give you so much work, you won't have time to go out and find your next 100. Or even if you find your next 100, they will give you so much work that you won't be able to look for your next 1000.

So ultimately, you will still continue to successfully make money, but you will never be able to create a successful product company. This is the single trap that I see almost all product companies fall into today. They all make money, and that's why they're still in the business but they stop eyeing the fact that they were supposed to be in the product business and not the services business. Now imagine taking a strategic decision like this in the early days when there was no competition in the market— today you can take a decision to change over night. But in the early days, while we did do services for companies (if someone asked you to do something extra, you did do it) we refused to take a single penny for any services that we did. That forced us to focus on selling new licenses. Otherwise once you're able to get money from services, there's no requirement to sell new licenses!

In your opinion, what's the reason behind Tally's popularity? At the risk of being politically incorrect, is it because of its "accessibility" due to piracy? Or is it largely because it's simple and user-friendly?

Pirated software doesn't become popular — popular software gets pirated. We strongly believe in one thing: if my software is not valuable to you, your money is not valuable to me. So customers are able to see tangible value in our software after they've paid for it, and therefore they tell their friends to also buy our software. Word of mouth has been the principle pivot of popularity, and we've told people on a number of occasions that if our

software has not been of value to them, we would return their money. Even after three years, people have returned and we have returned their money. In 25 years, this has happened nine times to us. But fundamentally, if our software doesn't work for them, their money doesn't work for us.

We see a lot of product start-ups coming up in both the enterprise and consumer space. What would be your advice to start-ups — where do you think they are lacking, and how should they go about correcting these issues?

I would ask them this: are they solving the problem for someone else vs are they solving the problem for themselves? If they are unable to be the most prolific users of their own solutions, they will find it difficult to put it elsewhere. It's the problem of architects, right? The architect is building for you — so they build and go away, but you have to live in the mess. I think as a company we had the privilege of this insight from my father. My most famous depiction of his words was in this context: in the early days, I had asked me a question against a certain context and when I was trying to explain to him that it was very difficult to solve the problem in that manner in software, which was why it was done in a particular way he asked me "Are you writing programs to make the life of the programmer easier or the life of the user easier?". The general tendency I have seen is that very few start-ups are willing to take the challenge of solving the complexity of the product themselves so that they give simplicity to the end-customer — and this is a fundamental requirement of the product.

The second problem that I find with product start-ups in the country is that most people design the software as if they are going to be present when the software is going to be used. It makes great sense for them to explain to someone how to use it, but if you want to be a software product company you have to design a product that can be used when you are not there. So, from a technical viewpoint fundamentally I would say that it is about being able to sit back and reflect upon these issues that impact your design. From an operational viewpoint, from day one you have to design as if you are not selling. It's easy for you to design a product and for you to go sell it, because you'll design your sales processes which are centered around your ability to sell. And this ability, because of your intimate knowledge of the product, will always be higher than someone else. So be able to design sales and service processes that are not operated by you will truly bring the product into the product category