MindTree
CONSULTING

Nine Key Factors Behind a Successful Technical Career

MindTree Lecture Series - 1

July 2004

The other day, I met a bright young engineer in MindTree and asked him what his ambition was. He was very clear. "I want to be an architect". My next question to him was what does he read? He looked surprised and then replied that he does not read much outside what appears on a computer screen. My next question to him was whom all does he admire in MindTree among the three best architects? He named the predictable three. Then I told him what the fundamental gap was between him and the best three. It was about the ability to make intelligent conversation about any subject under the sun – a capability borne out of serious reading habits.

The next thing I asked him to do was to poll these three on what were the six books they had read last. The result was amazing. The three named eighteen books in all – of which at least six were common. Ninety percent of the books had nothing to do with Information Technology. The exercise proves a key point – to be a great nerd, one has to have interests outside writing code. However, many engineers think that the path to a great technical career is about technical skills alone.

Long back, Bell Labs conducted an interesting study – closely watching the common characteristics among a group of technical professionals who rose to the top. The exercise revealed nine key factors outside just technical competence that differentiated brilliant technical folks from the masses. The study was conducted by Robert Kelly of Carnegie Mellon and Janet Caplan of Williams College. As I see the Indian industry today, I think the study done at Bell Labs remains relevant in every detail.

The Bell Labs engineers who did extremely well for themselves – as they progressed in their career, showed the following qualities that differentiated them from their peers: taking initiative, cognitive ability, networking, leadership, teamwork, followership, perspective, organization savvy and show-and-tell capability. Let us look at each of these and see what lies underneath.

Taking Initiative

This is about accepting responsibility above and beyond your stated job. It is about volunteering for additional activities and promoting new ideas. None of these will jump out as apparent as a young engineer gets in to her first job. She will tend to think that her career progress is really dependent only on the ability to write code. The concept of initiative begins by looking for technical and other opportunities in the organization and volunteering for them. The idea of volunteering is little

understood – both by organizations and individuals. In the days to come, it will gain increasing prominence in our professional lives.

Initiative is also about two other things – dealing constructively with criticism and planning for the future. The latter is a function of many things – a good starting point is to start mapping the environment, learning to understand how the future is unfolding and then stepping back to ask, how am I preparing myself?

Cognitive abilities

The concept of cognitive development is about understanding the interplay of technology and trends in how they are getting deployed. It is also about recognizing the business eco-system in which technology works. It is about situational understanding and consequence thinking. The importance of consequence thinking is very critical. It asks us to look beyond the immediate deliverable of a task and it is about asking who will be impacted by my work, what is the end state? People in our industry just think in terms of modules and seldom ask where is it going, who is my customer and more importantly – who is my customer's customer? Cognition is a key faculty that determines how much we are able to read patterns, make sense of things. Refining cognitive skills helps us to go beyond stated needs of our customers to explore unstated needs.

Networking

We tend to think of networking in a social sense. As one grows higher in life, we are often as powerful as is our network. Building a professional network requires us to step out of the comfort zone to look at whom can I learn from. Quite often, and more as one progresses in life, the learning has to come from unusual sources. At MindTree, we expose our people to social workers, architects, graphic designers, teachers, people who lead government organizations, leaders from client organizations. The interesting thing about benefiting from a network is that it works like a savings bank. I need to deposit in to it before I withdraw. We all have heard about how important internal and external knowledge communities are. Again, in MindTree – we encourage people to belong to 26 different knowledge communities that run on a non-project based agenda and are vehicles of learning. These create networking opportunities and open many doors.

Leadership

Next to networking is development of leadership skills. Many technical people associate it with "management" and shy away from developing key leadership skills like communication, negotiation, influencing, inter-personal skills, business knowledge, building spokespersonship and so on. Take for instance negotiating as a skill. Imagine that you are an individual professional contributor. Why should you learn to negotiate? Tomorrow, your organization becomes member of a standard body and you have to represent the organization as a technical expert. You will find yourself needing to negotiate with powerful lobbies that represent a competing viewpoint or a rival standard. Unless you have honed your capability alongside your hacking skills, you will be at a complete loss. Yet, you do not discover your negotiating capability one fine morning. You need to work on it from an early stage. Negotiating for internal resources is becoming another critical need. You can choose to remain an individual professional contributor but from time to time, you have to create mind share in the organization where resources are limited and claimants are many. Establishing thought leadership is another key requirement of growth and independent of whether I want to be a technical person or grow to be a manager, I need to develop as a leader who can influence others.

Teamwork

Our educational system does not teach us teamwork. If you ever tried to solve your test paper "collaboratively" – it was called copying. You and I spent all our school and college life fiercely competing to get the engineering school and seat of our choice. Then comes the workplace and you suddenly realize that it is not individual brilliance but collective competence that determines excellence. Collaboration is the most important part of our work life. Along with collaboration come issues of forming, norming, storming, performing stages of team life. Capability to create interdependencies, capability to encourage dialogue and dissension, knowledge sharing become critical to professional existence. All this is anti-thesis of what we learn in the formative years of life. Add to it, our social upbringing – our resource-starved system tells us to find ways and means to ensure self-preservation ahead of

teamwork. In Japan, the country comes first, the company (read team) comes next and I come last. In India, it is the other way round.

Followership

The best leaders are also great followers. We can be great leaders if we learn and imbibe the values of followership. Everywhere you go - there are courses that teach leadership. Nowhere you will find a business school teaching you followership. Yet, when solving complex problems in life, we have to embrace what is called "situational leadership". I have to be comfortable being led by others, I must learn to trust leadership. Many people have issues reporting to a test lead as a developer, or being led by a business analyst or a user interface designer. In different parts of a project life cycle, people of varied competence must lead. I must be comfortable when some one else is under the strobe light. I must have the greatness to be led by people younger than I, people with a different background or a point of view. That is how I learn.

Perspective

This is the hardest to explain. It begins with appreciating why I am doing what I am doing. Quite often, I find people having a very narrow view of their tasks; many do not see the criticality of their task vis-à-vis a larger goal. So, a tester in a project sees his job as testing code or a module designer's worldview begins and ends with the module. He does not appreciate the importance of writing meaningful documentation because he thinks it is not his job or does not realize that five years from now, another person will have to maintain it. I always tell people about the story of two people who were laying bricks. A passer by asked the first one as to what he was doing. He replied, "I am laying bricks". He asked the second one. He replied, "I am building a temple". This story explains what perspective is and how the resultant attitude and approach to work can be vastly different.

Organizational Savvy

As technical people grow up, they often feel unconnected to the larger organization. Some people develop a knack of exploring it, finding spots of influence, tracking changes, creating networks and in the process they learn how to make the organization work for them. The organization is not outside of me. If I
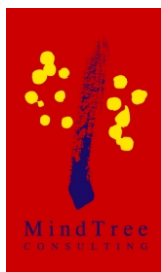
know it well, I can get it to work for me when I want. Think of the difference between one Project Manager and another or one technical lead from another. One person always gets the resources she needs – the other one struggles. One person knows who is getting freed from which client engagement and ahead of time blocks the person. One person reacts to an organizational change and finds himself allocated to a new project as a fait accompli – another person is able to be there ahead of the opportunity. Larger the organization, higher is the need to develop organization savvy. It begins with questioning ones knowledge about the larger business dynamic, knowing who does what, tracking the work of other groups, knowing leaders outside of my own sphere and a host of other things. Importantly, it is also about tracking what the competitors of the organization are doing and keeping abreast of directional changes.

Show and Tell

This is the bane of most Indian software engineers. We all come from a mindset that says; if you know how to write code, why bother about honing communication skills? Recently, we asked a cross section of international clients on what they think is the number one area of improvement for Indian engineers? They replied in unison, it is communication. Show and tell is about oral and written communication. Some engineers look down upon the need for communication skills and associate it with people who make up for poor programming prowess. It is the greatest misconception. Think of the best Chief Technology Officers of companies like Microsoft, Oracle, IBM Global Services or Sun. Their number one job is evangelizing. If they cannot forcefully present their technologies, nothing else will matter. So, every engineer must pay attention to improving the ability to present in front of people, develop the ability to ask questions and handle objections. In a sense, if you cannot sell the technology you create, it has no value. So, building salespersonship is a key requirement for technical excellence.

The foregoing points are not relevant if you have already filed your first patent at the age of eighteen. Everyone else, please take note.

(Subroto Bagchi is the Co-founder and Chief Operating Officer, MindTree Consulting. This article is based on the lecture delivered by Subroto as part of the MindTree Lecture Series held in Bangalore.)