The Softer Side of the Architect
by Joe Shirey

Summary
Most of us would agree that having strong technology skills is a key ingredient to being a competent architect. However, the most successful architects that I have met possess more than just great technical skills. They also have qualities that enable them to work well with people.

Developing and refining these “soft skills” can take ordinary architects to new levels of effectiveness in their careers. This article outlines a framework I developed for defining these soft skills and strategies for the architect based on my experiences and interactions with architects I admire.

Why Soft Skills are Relevant
We have all worked with one of them at some point in our careers—brilliant technologists who can solve just about any technical problem, but are so arrogant and insufferable that people despise working with them. Some of these ubertechs will reach a point in their careers where they no longer get promoted and their value is marginalized from a business perspective. The reason they hit this ceiling is due to their lack of the “people” and “business” skills that we often call soft skills.

If you look at the Microsoft Certified Architect program, you will notice that there is a set of competencies that go well beyond technology skills. These competencies were based on focus groups from companies large and small. One common theme from these focus groups was that the soft skills really matter. In fact, they identified more soft competencies than technical competencies. In their view, the soft skills are what separate the highly skilled technologist from the true architect.

The International Association of Software Architects (IASA) has also gone through a detailed analysis and polled its members to determine the skills necessary to be a successful architect. Of the five foundational skill areas the IASA identified, two are based in these soft skills.

It is apparent that attempts to define the critical skills of an architect contain some measure of soft skills. In my experience this is quite true; the most successful architects I know are able to increase their effectiveness by combining their technical and nontechnical skills.

In my mind, the successful technical solution also requires three distinct soft skills: business alignment, perspective awareness, and communication (Figure 1). Most architects acknowledge the importance of these areas yet fail to make them real priorities during the project life cycle. In this article, I’ll take a look at each of these areas and offer strategies for ensuring that they remain priorities during the project without significant additional effort.

Business Alignment
I doubt anyone would deny that business alignment is a critical success factor for any project. Most projects begin with some type of requirements document that drives most of the technical decisions or at least an architecture document that demonstrates how the architecture meets business needs. The issue generally isn’t lack of intent but with the alignment at the strategic level.

In my different roles, I have had the opportunity to review many projects and discuss them in detail with the architects. Usually, the architect can discuss the business requirements, but it is surprising how often the architect cannot explain the project in terms that the CFO would understand. There is a lack of understanding of the real business drivers and the detailed financial implications versus the business requirements. It is the critical factor that drives the real project decisions.

When I mention business drivers, I mean tangible and measurable items. Often, an architect will describe the project drivers as “increase customer satisfaction” which is quite nebulous. I believe that business drivers should boil down to financial terms when possible. In the above example, I would recommend a goal such as “increase repeat customer revenue by 10 percent and reduce call center support costs by 5 percent over the next year” with detail to support why these goals are attainable based on the project. These types of objectives will drive the true project benefits and can frame what a project should cost.

There are some projects that are not driven by purely financial terms, but there is always an underlying business reason for the project. For example, compliance projects often are not done based on a pure return on investment model. However, it is important to understand the business risks associated with not doing the project and use those risks to develop the appropriate solution.

By understanding the actual business drivers, the architect is able to make rational technical decisions during the project life cycle. Because many projects are based on business requirements, we treat them as a “contract.” We sometimes become inflexible about changing requirements because we have based all of our plans on that “contract,” but business situations and drivers can change during the course of a project. If we possess a true understanding of the business aspects, then we can help the business side of our organizations understand the implications of those changes in our solution development.
I often think back upon a project that I worked on for a niche software vendor a couple of years ago in which the customer was pushing us to complete a fairly large and complex project in a timeframe that was unrealistic for building a high quality and maintainable solution. After many heated discussions about the situation, we started to understand that the date was not arbitrary — it was about the company’s ability to survive. If the solution wasn’t done by a certain date, they would likely go out of business. That sense of urgency certainly drove the architecture of the solution (or some might say “lack of architecture”). We had to make important trade-offs, which was really a loftier goal. We also took the appropriate steps to ensure the customer understood the trade-offs made during the project and the future implications. Because we took the time to fully understand the issues that drove the business, we moved into a strategic rather than a tactical role.

In that example, I certainly wasn’t very proud of the technical solution, but we solved the customer’s problem and kept the business afloat, which was really a loftier goal. We also took the appropriate steps to ensure the customer understood the trade-offs made during the project and the future implications. Because we took the time to fully understand the issues that drove the business, we moved into a strategic rather than a tactical role.

In my opinion, agile methodologies are a strategy for aligning with the business. The methodology enforces regular and scheduled stakeholder interactions, resulting in better understanding of and alignment with the underlying business needs. Not to say that agile projects will always achieve this alignment, but the agile methodology does make it more likely. Nor is it true that non-agile methodologies cannot be aligned, but many of these methodologies do not enforce the regular involvement of the business.

**Summary Strategies for Business Alignment**

- Think about your project like a CEO and CFO. Invest the time up front to dissect the business drivers for the project, and if possible, determine the true financial impact of the costs and benefits of the project.
- Use business drivers instead of requirements as your guide for developing the solution architecture. Keep a finger on the pulse of the business throughout the project life cycle to maintain the appropriate flexibility in the project.
- Evaluate how your methodology maintains business alignment during the project life cycle. If needed, inject some regular touch points to keep the business close to the project.
- After your solution is put into production, look for ways to measure its ability to meet the defined objectives.

**Perspective Awareness**

Each member of the team, from the project stakeholders to the developers and testers, has views and motivations that have the potential to create conflict, posing problems for the architect that can even derail the project.

It is fairly common for the development team to dismiss views from outside of the team. This attitude is dangerous not only because the team might miss some important business aspects of the solution, but they also risk alienating others and gaining a reputation for being difficult. It is the role of the architect to bring these groups together to ensure that the team is building the solution that best meets the needs of the business.

Understanding others’ often diverse needs is key to ensuring that the solution is the optimal fit. The best way to understand their needs is to view the business issues from their perspectives. An end user of the system has concerns about usability while an operations manager wants to ensure the system performs properly in a production environment. Every stakeholder’s perspective should inform the shape of the solution architecture. Having a mindset that every individual has something to contribute facilitates genuine interactions and meaningful communication.

The architect should be cognizant that not everyone involved may understand or believe in the business drivers of the project. Personal motivations or organizational politics can sometimes work against fruitful interactions. As the architect, you cannot control all factors, but if you take the time to frame these interactions within the scope of the business drivers, then it becomes more difficult for others to inject non-important or divisive factors.

Many architects are in leadership roles with regard to the development team. Managing the attitude of the development team in some environments can be a challenge. Some development teams rarely interact with anyone outside their own teams; in their isolation, these teams become disconnected from the organization. Often it takes someone on the project team to continually remind people that others have a job to accomplish and everyone is on the same team. It is amazing to watch teams transform when leaders take the initiative to help them see other views and motivations instead of joining in when the group starts complaining.

**Summary Strategies for Perspective Awareness**

- Take the time to understand the perspectives and motivations of the individual.
- Ensure that multiple perspectives are taken into account.
- Frame the conversations with the guiding business drivers to alleviate personal agendas.
- Lead by helping others see alternate perspectives.
“OPEN AND HONEST COMMUNICATIONS ARE OF PARTICULAR IMPORTANCE WHEN COMMUNICATING WITH THE PROJECT SPONSOR. PROJECTS SHOULD BE DEVELOPED WITH A STRONG COMMUNICATION PLAN BETWEEN THE ARCHITECT AND THE PROJECT SPONSOR—WHEN SPONSORS ARE THE FIRST TO KNOW ABOUT POSITIVE OR NEGATIVE SITUATIONS, THEY CAN BE IN CONTROL OF THE PROJECT AND REACT WHEN NECESSARY.”

Communication
Most people would probably agree that communication is a key skill of the architect, but few architects employ deliberate strategies for successful communication as part of their roles. Since many of us have technical roots, we tend to revert to our technical strengths when problems occur.

Many problems stem from the inability to communicate with people in the terminology they use on a day-to-day basis. Some of the best architects I’ve known have the ability to meet with executives and discuss topics in business terms, then walk down the hall to the development team and dive deep into technology discussions. I believe this skill can be learned, but it takes discipline and listening skills.

If you have ever traveled to a foreign country where you don’t speak the language, you probably understand how daunting it can be to interact with others. Sometimes you are distrusted just because you don’t know the language. But if you learn a few basic words and seek to understand the culture and customs, you may even be welcomed. In general, people who are fluent in the language are much more likely to be trusted and embraced.

You have probably experienced a similar phenomenon when you have taken a job at a new company. You have to learn a new language, culture, and customs. For example, the term “architect” at one company can mean a radically different role from an architect at another company. Over time, you pick up on the language nuances and begin to use them in your everyday communications. You start to incorporate acronyms in your daily language that meant nothing to you before.

This kind of subgroup acculturation also happens within an enterprise. Each group develops its own “native” language. As an architect it is important to listen carefully for these language clues and understand their usage across the enterprise. You need to become multilingual, using the appropriate language and terminology based on the audience.

The skill to develop is the ability to pick up on the language in a group rapidly and to use it effectively, even within the course of a meeting. This means learning to listen effectively during the course of your time with a group and to start cataloging the terms. By communicating with someone in their “native tongue,” you get over a trust barrier and increase the possibility for open communication.

After becoming multilingual within an organization, you might want to restructure your meetings to bridge languages and needs. I have attended many meetings among business and technical groups in which one group will dominate the meeting and the other will lose interest and stop communicating. When the situation calls for bringing together disparate groups, you may need to play translator to ensure that each group is communicating effectively.

In all of your communications, one of your primary goals should be to build trust with your audience. The best way in my opinion is to be tactfully open and honest. In general, if people feel that they can trust what you have to say, they are more likely to give you their true thoughts even when they are unpopular.

Open and honest communications are of particular importance when communicating with the project sponsor. I have observed many projects developed with a strong communication plan between the architect and the project sponsor. In these cases, because the sponsors were always the first to know about positive or negative situations, they were in control of the project and could react when necessary.

These types of communications need to be very efficient. Sending a 20-page status report to the sponsor on a weekly basis with the minutiae of the project is not an efficient use of anyone’s time. I believe that a written status report on one side of one sheet of paper with the macro-level project statistics and the primary issues and risks delivered in person in 15 minutes a week sets up an effective communication channel and builds rapport. When using this process, I have had project sponsors comment that they felt more engaged with the project than any other in the past.

When communicating with others, it is important to avoid absolutes. Terms like “can’t” and “won’t” convey inflexibility. The alternative is to present options and implications of decisions in an open and honest manner. This approach also endows the conversation with a peer quality, opening the lines of communication to a discussion rather than a debate. If the business drivers are also a part of the discussion, then there is a framework for making joint decisions.

Summary Strategies for Communication
• Communicate with others in their “native” language.
• Foster an open and honest environment.
• Provide efficient communications to the project sponsor.
• Present alternatives and implications when possible.

Conclusion
Most of us could benefit from further developing our soft skills to improve our effectiveness in our day-to-day role. The framework I’ve described can help you develop strategies to improve these important skills.

About the Author
Joe Shirey is a senior architect evangelist for Microsoft based in Denver, Colo. Prior to joining Microsoft, Joe was a vice president at Interlink Group where he was responsible for services delivery for their largest market. In the past, Joe was a Microsoft Regional Director; he has served on the Microsoft Architect Advisory Board and the .NET Partner Advisory Council. He has more than eighteen years of hands-on technical and functional experience in project management, systems analysis, design, development, and implementation. Joe attained his Microsoft Certified Architect in Solutions in 2005.