

Closing the Gap Between Business and IT

by Ivar Jacobson and Kurt Bittner

From the time that computers were first applied to solving a business problem more than 50 years ago, there has been a communication gap between Business and IT. For almost as long we have sought solutions, but the gap grows only wider, until it now is a chasm that needs a fairly substantial bridge.

From the Business you may hear that “we have no confidence in IT’s ability to deliver useful solutions”, or “we have limited visibility of progress, risks and problems”, and “we don’t know how we should measure the value of our investments in IT.” From IT you may hear that “they (the Business) don’t fund the projects adequately”, or “they don’t know what they need”, or “they don’t know what is possible to develop”. Each side feels the other is responsible for the problem. And, you know, both are right.

Over the years many things have been tried to close the gap, from the one extreme to another. Some have viewed the gap as a soft problem: if only Business and IT could collaborate better and learn from one another the problem would be solved. Improvements in communication and sensitivity training were tried, but still the gap grew.

At the other extreme people have tried to apply engineering approaches to the problem, assuming that rigor would succeed where informal communication has failed. The result was usually a formal business process model, understandable only by its creators who were typically IT people. This usually made the business people feel ignorant and annoyed, and the gap grew wider.

So we know what doesn’t work - let’s try something different! Our view is that the solution lies somewhere in the middle. Starting with the end in mind, what everyone wants is an executable solution that supports and improves the ability of the business to succeed. That end is oft-forgotten, but no one wins if it is not achieved. All of the document sign-offs and hand-offs and other techniques to ensure accountability and transfer responsibility mean nothing if business results do not improve.

It has been said that a sign of insanity is doing the same thing time and again and expecting different results. We have ample proof that traditional approaches like writing and approving a comprehensive requirements document and then “throwing it over the wall” to be implemented does not produce good results. It is beyond human ability to specify all requirements upfront. But let’s go beyond that to the real heart of the matter - the very nature of the Business-IT relationship.

The Broken Business Model for IT

The dominant model in IT to date has been to think of IT as an internal “supplier” to the business, with the business as the “customer” of IT. This model is at the heart of the problem.

The reality is that IT is not a supplier. If it were a supplier, it would be free to pursue

other business if it did not find the terms the business offered attractive - and in most cases, the terms the business offers *are* unattractive: the business typically dictates cost and schedule and features. Or rather it tries to - it usually dictates the cost and the schedule but lets the features pile on so that the initial cost, usually unachievable, becomes burdened with extra features (“usually irrelevant”) so that the original cost and schedule are guaranteed to be unachievable.

If IT is not a supplier, then neither is the business “the customer”. If it were a customer it would be free to source services from any supplier, and it often tries to do so. The reality is that most suppliers would not accept the terms that the business imposes on its own IT organization: fixed cost and schedule and thoroughly variable scope. The Business wants, and has been allowed to believe it is reasonable to demand, the ability to endlessly change its mind, or at least to be imprecise about what it really needs without bearing the cost.

It should be obvious to anyone familiar with the situation that the current model does not work, and has never worked. We need to approach the problem in a different way!

Let Business and IT play on the same team, together, to win

Let’s consider a proven model for working together as a team to win. A soccer team has specific positions with specialized skills: goalkeeper, defender, forward. Despite the specialization, anyone can kick the ball, including the goalkeeper. If we look at Business and IT as a single “team” (let’s call this the *Solution Team*) we also have many kinds of specialists: people with knowledge of business processes and resources of various kinds (human, machines, etc.), people with knowledge of how to write code, how to test software, and how to understand needs and devise solutions. All of these specialties are essential to deliver the best solution, but everyone must collaborate and work as *one* team for everyone to be successful.

On a soccer team everyone must also understand the basic rules of the game in order to contribute to winning. On the *Solution Team* everyone must know how to play the “game” of software development for the team to win. This means understanding how software can and should be built: in many small steps based on a long-term roadmap. Business people especially need to understand how to participate in the development effort by communicating the *essence* of business processes and the outcomes the business needs to achieve. Working together, then, the *Solution Team* (which includes the business) explores different ways the outcomes can be achieved and together agrees on how to move ahead.

One of the reasons for the Business-IT gap is that typically the business is expected to define detailed requirements (which are really specifications for a solution) when they are poorly equipped to understand how technology can be creatively applied to deliver the end result that needs to be achieved. This causes everyone to lose focus on the real problems to be solved. By being clearer about the outcomes that the business wants to achieve it is possible for the whole team to have a clearer focus on what must be done. This does not mean that there is not a role for requirements and other specifications - there is, but as a record of the agreements made within the

team. These agreements can be as formal or as informal as the team collectively agrees, as they serve mainly as a convenient way to document discussions, not as a comprehensive communication mechanism.

This means that everyone, including and especially the IT team members, needs to understand business goals, strategies and business processes. Software has the ability to change and improve business processes, but not if it starts from a flawed or incomplete understanding of how the business works and what it needs to achieve.

Business and IT need to both change and learn new skills so that they can, despite their different backgrounds and perspectives, join together as one team working toward a single goal: to deliver creative solutions that create business value. This means that business people need to gradually come to better understand what technology can do for them, and technology people need to gradually come to understand how the business works so that they can devise creative solutions.

In order to reach this goal the team needs to agree on a way of working and on the practices they think they need to solve the problem at hand. These practices may involve proven techniques such as developing iteratively, formulating requirements and tests as use cases and test cases, planning product releases based on scenarios, measuring progress through successfully passed tests, and so on. These practices will involve both sides in meaningful collaborative work that, if correctly performed, leads to an excellent result. By working together as one team, Business and IT will win together.

Challenges in getting Business and IT to play on the same team

It's nice to talk about Business and IT playing on "one team", but the reality today is that if they were a soccer team they would probably not win many, if any, games. Extending the soccer analogy, the Business often acts like the absentee owner who wants the team to win but does not really want to take the time to be directly involved. Instead they try to micro-manage from a distance, demanding a detailed play-by-play plan for who is going to score and when, and they berate the team for not adhering to the plan. They will say that they will provide players (business representatives and product owners) but the players they assign are usually absent because they are too busy doing other things. As the team owner, they also don't want to spend too much to hire the best players and coaches, but they still want to win against teams that are willing to spend more.

But IT is often not any better. Instead of providing convincing proof that the team is making progress, they hide behind detailed play-books and technical jargon. They claim they cannot win unless they get the best uniforms, and a nice practice field, and the best shoes, and a new stadium. They spend lots of time coming up with new strategies and play-books and little time on the field actually practicing. They require the owners to review and sign-off on all plays, and if a play does not work they blame the owner for not providing good feedback.

Are you beginning to see the connections? Maybe soccer and software are not so different after all!

To be more direct, here is what often happens:

1. The Business often wants to dictate, or at least lock-in, the cost and schedule even before they really understand what they really need or what problem they are solving.
2. The Business is rarely accountable for the business benefits claimed in the business case. Business cases are often “rigged” to justify a project that the Business “wants to do” - but the project team is held accountable for cost and schedule estimates based on an often poorly conceived “wish list” of features, and on the “overly optimistic” business case.
3. The Business often feels that anything less than meeting all their initial expectations, no matter how poorly understood or unrealistic, is a failure.
4. The Business often is unclear about its real needs, and tends to demand a lot of things that it does not really need and that never get used.
5. When they do change their mind about what they want or need, they rarely adjust the original cost and schedule.
6. The Business is often not willing to invest time in better understanding their needs because that will “take too much time”, and yet failure to deliver will be considered IT’s failure.
7. The Business is often not willing to provide access to the most knowledgeable people because they are considered too valuable to take away from their regular jobs, even though the project cost is great and the need for success even greater.
8. The Business often regards software development as a commodity skill that they can outsource to the lowest cost provider. They do not value software development professionals who actually understand their business and they are often not willing to invest in developing the IT organization as a strategic resource.

But IT is not a blameless victim in this game:

1. IT usually accepts the situation, feeling that they are powerless to do anything about it. Even when they know they cannot provide estimates of cost and schedule with only a vague understanding of the needs, they provide one anyway, and this “vague estimate” becomes the measure against which the project is held accountable, even when it is a poorly-informed guess.
2. IT managers often do not really understand software development, and they don’t tend to take an interest in becoming better informed. As a result they think that people are interchangeable, that a good process can compensate for lack of skills and experience, that teams and teamwork are not important, and that success means delivering what the Business asks for as opposed to what they really need.
3. They often measure the wrong things - like production of artifacts - rather than whether the software has become better, whether it is developed faster, whether the development has become cheaper and whether the customer is happier.
4. IT’s engagement often tends to be low-level, not strategic, focusing on “cool technology” that is poorly connected to business benefits. They are not trusted by the Business to contribute as strategic problem solvers, mainly because they have not demonstrated that they really understand the business and how to apply technology to creating business value. They wait for the Business to

- tell them what to do, rather than proposing solutions.
5. The technology itself is immature and not well understood, and decisions are often driven by fads and the desire to “play with the latest technology”. This results in systems that age badly and often need replacement only a decade after their original deployment.
 6. And finally, the IT team members often do not respect each other’s contributions. Developers look down their noses at testers and business analysts. Business analysts tend to be unconcerned about whether something they specify can actually be developed (that is a mere detail), and testers often lack the skills to do more than manual testing, since that tends to require real development skills.

Given these challenges, it would seem impossible to get Business and IT to work together. Is there really a simple way to break down these barriers to really working as one team? Yes, and no. Some of the changes are simple in concept but hard to put into practice. Other changes require a fundamental re-thinking of the relationship between the Business and IT and will take changes in the cultures of both organizations.

Ten Fundamental Steps You Need to Take to Get Started

As noted above, the first step is to fundamentally rethink the relationship between Business and IT, and to change the relationship from being based on a “customer-supplier” model to one in which Business and IT are one team working toward one goal. The journey will be neither easy nor short, but there are some things you can do to get started.

Most fundamentally, you need to recognize that the problems are deeply rooted in, and protected by, an IT management approach that has come to be accepted as “simply the way it is”. In order to change the “IT as Supplier” model, you will need to create cross-cutting solution teams with members from both the business units and IT. Each contributes essential and unique skills and perspectives without which success is impossible. Unfortunately we have created a model that tries to minimize collaboration, and a culture in which business people seek to minimize interaction with IT instead of looking for opportunities to create value through creative application of technology to business problems.

There are a number of things you can do to *start* the necessary transformation:

1. Choose a technically competent CIO who understands and is respected by the business leaders, a CIO who can instill in these business leaders an appreciation of the value of modern technology, a CIO who can lead with enthusiasm, inspiration and passion. Giving such a leader a significant seat at the executive management table will help to establish an organizational appreciation of the potential for technology to enhance the creation of business value.
2. Fund IT related projects directly from the business’ budget, rather than from a centralized “pool” for all IT projects. Doing so has an immediate tendency to increase the attention paid to IT projects by the business - it is

now *their* money that is being spent, not “IT’s money” - and this creates a direct connection between cost and what the Business asks for.

3. Charge the business for the fully allocated cost of operating their applications, including maintenance. Today these costs tend to be buried in IT overhead, rendering it hard to see when an application is aging badly and costing more to support. Often the Business only sees the cost of new development, which is usually less than a quarter of all IT costs. The result of this is that the true cost of ownership, and therefore the cost of not modernizing, is hidden, often leading to poor investment decisions.
4. Prioritize the projects based on the business value that they will deliver *over their planned* lifetime, considering the full lifecycle costs and benefits not simply the initial costs and benefits. Most project funding requests fail to consider the long-term costs and benefits of applications, resulting in unfunded long-term support costs.
5. Let IT and Business collaborate during the chartering of projects to define the problems to solve, or the goals that are to be reached, and to work together to find creative technical solutions to these problems, stimulating innovation. Today, too often, the Business requests or demands a solution based on a limited understanding of what could be done, and often based on an incomplete or flawed understanding of the problems that need to be solved and the goals to be met. Working together, with a focus on really understanding what is needed and what can be done to meet those needs, better solutions can be devised. Only then should projects be funded and chartered.
6. Once a project is chartered, select a project lead from the Business - someone who really understands the Business and its needs, who can assemble the right team members to deliver the right solution. Some of these will come from the Business (such as marketing, product design, software development, financial, business process experts, and so forth) and others from IT (such as analysts, developers, testers, and so on). Resources should be brought together from different organizations into a single cross-functional team, focused on delivering business value - and measured against their success in delivering that business value. Once the project is under way, ensure that both Business and IT participate in the development project as if the Business depended on it - which it should if the project is worth doing. This does not mean that project team members cannot have other commitments (everyone does), but that commitments to the project need to be negotiated and honored, or someone else should be assigned.
7. Measure the performance of the project against the goals established in the business case, and hold the team, with representatives from both Business and IT, accountable for these results. Even before the project delivers, measure results in business terms such as the number of business scenarios that have been implemented and tested - don’t measure progress in terms of completion of documents. When you publish internal progress measures, always present them in business terms.
8. Recruit and promote IT leaders who have business sense, that is, who understand the Business and have a proven track record for delivering innovative solutions. In the business units, recruit and promote people

who can create value by solving problems using innovative applications of technology.

9. Promote IT discussions that value communication in business terms and not in technology terms by continually focusing the discussion on how technical solutions will create business value. Stimulate cooperation and collaboration between IT and Business by ensuring that representatives of both speak with one voice and avoid one-way 'monologues'. Apply practices that get Business and IT to collaborate and which, if correctly performed, result in measurable success.
10. Derive software architecture from the business model – starting from the business use cases and business process models, deriving a road map representing the system's major components and their collaborations. This road map will then be fulfilled over the course of many projects, with the road map providing organization and consistency of investment and effort.

This is not all that needs to be done, but it is a start. The broken relationship between Business and IT has been years, even decades, in the making, and it will take a long time and focused effort to fix the problem.

In our discussion we have consciously chosen the word “closing” instead of “bridging” when we speak of the gap between business and IT. A bridge acknowledges the gap, even accepts the gap, and tries to make a small connecting path across the gap. We think that the gap needs to be eliminated. Business, and therefore IT, cannot afford for there to be a gap - the two need to work as one in order to deliver better business solutions. The benefits can be substantial - not merely automated afterthoughts to improve existing business processes, but whole new ways of delivering value and fulfilling the needs of *real* customers.

Ivar Jacobson

Dr. Ivar Jacobson is a father of components and component architecture, use cases, aspect-oriented software development, modern business engineering, the Unified Modeling Language and the Rational Unified Process. His latest contribution to the software industry is a formal practice concept that promotes practices as the 'first-class citizens' of software development and views process simply as a composition of practices. He is the principal author of six influential and best-selling books. He is a keynote speaker at many large conferences around the world.

Kurt Bittner

Kurt is Chief Technology Officer of Ivar Jacobson International, Americas. He has worked in the software industry for more than 25 years in a variety of roles including developer, team leader, architect, project manager and business leader. He has led agile projects, run a large division of a software development company, survived and thrived in several start-ups, run an acquisition, and worked with clients in a variety of industries including aerospace, finance, energy and electronics. He was a key contributor to the early development of the Rational Unified Process as well as, more recently, IBM's Jazz project. His experience includes significant work in banking and finance, relational database system design and architecting, and consulting and mentoring a wide variety of clients on software development improvement strategies and approaches. He is the co-author of "Use Case Modeling", "Managing Iterative Software Development Projects" and "The Economics of Iterative Software Development".