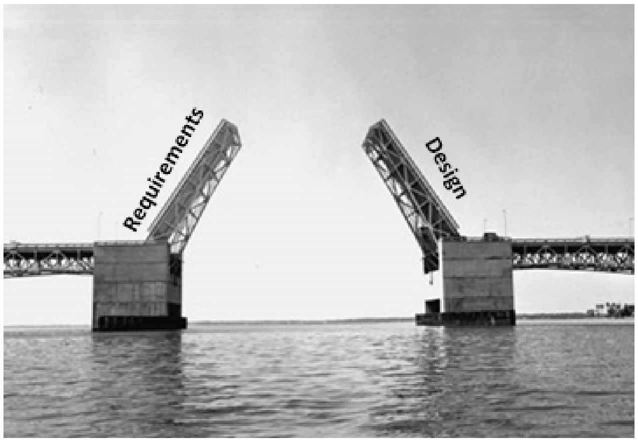
TECHNOLOGY
Leveraging
Strategic by Design

Best Practices in Requirements Engineering

The Requirements Dilemma Part II: Bridge the Gap

By Sandy Kleinberg, Practice Leader, Business Architecture

In this two part series, we discuss the two bookends of the requirement phase. In Part I, we explored what we should have in place to enter the requirements phase. In Part II, we will look at creating requirements that work for the first consumer of the requirements, the architect.



A few years ago, I was part of a very large transformation project at a health plan which included the replacement of its core claims processing system. The health plan wanted to maximize their chances of success so they brought in a requirements consultant. Among their many humorous approaches to requirements (in hind sight only, I assure you) was to have the team pretend that they knew nothing about the system we intended to put in place. This approach lead to mass confusion, several failed attempts at creating the requirements, millions of lost dollars and ultimately the premature departure of the consultant. I often use this anecdote to illustrate what goes wrong when we ignore reality when creating requirements.

Although that's an extreme example, I see companies essentially making the same mistake regularly in the year 2014. It happens every time they engage in requirements work using a one-size-fits-all method or don't provide analysts with scenario-based techniques to articulate requirements in a specific and repeatable way. Although all "downstream consumers" of the requirements suffer, the architect bears the brunt of it because they are the first person to have to use the requirements to produce their designs. The architect essentially has to mitigate the deficiencies in the requirements by asking the questions that should have been asked during that phase.

What's the answer? First, always frame your requirements in the context of the gap. Next, use the "factory" approach to developing requirements. The factory approach is an iterative, pattern-based requirements methodology that is tightly coupled with design. It offers flexibility and the benefit of past experience to create the kind of requirements that may cause your architect to do a happy dance.

The Requirement Factory Approach

1. We begin the requirements factory with a thorough examination of the gap and produce a preliminary inventory of pattern-based gaps (e.g. a real-time system interface or an executive dashboard). Starting from here helps us avoid the nasty situation from the health plan where we pretend a system hasn't already been identified.
2. We produce overriding artifacts such as process and information models that pull together all of the gaps in a coordinated way and identify missing or inconsistent requirements later.

3. We create a set of business and user requirements for each gap using the pattern. As we progress through these first two iterations, we can question whether we have identified the correct pattern and adjust if necessary. We also identify which sub-patterns are in play and refine using them. Once we get to the point where SMEs agree on business and user requirements, the architect has the opportunity to identify design patterns they intend to use. Now we can finish with the implementation requirements that define what must be built.
4. Last, we integrate all of the requirements and check for consistency and completeness.

The Requirement Factory Benefits

Activity Sequencing – The factory approach allows us to iterate between requirements and design in a way that connects them in a very potent way. It allows us to look at needs (requirements) and solutions (design) as two perspectives of the same thing: the gap. Resolving this basic understanding leads to…

Better Relationships – The factory approach allows the various roles like the business analyst and the architect to participate in both phases and take an approach I call, "co-design." By this I simply mean that the SMEs develop more trust in the architect because they are present from the beginning of the project and truly understand the needs and the drivers behind the needs. On the other hand, the architect is now able to produce designs in a friendlier environment and can more easily frame the design in a way that can demonstrate it's optimized for the organization vs what the SME may have assumed would be built.

Results – Perhaps the most telling benefit is the outcome: higher quality requirements and reduced timelines. The beauty of starting from a known set of patterns is that they help produce more consistent results because we are starting from a known place. Architects are much happier when they know what to expect and what they expect is directly aligned to their design patterns. Timelines are drastically reduced because we aren't starting from a blank whiteboard.

Follow Sandy Kleinberg on LinkedIn for more insights and practical applications of business architecture. Sandy welcomes your questions and comments; contact him at skleinberg@leveragingtechnology.com or 585.454.4250 x145.

Working with Leveraging Technology is different than with other consulting firms because we look at each client's organization and not just the technology. We partner with clients in unique ways to help them design and build internal capabilities and sustain them in the most cost-effective way.

About Leveraging Technology

We are a business consulting firm founded in 1998 that fully understands how all aspects of the information technology lifecycle can impact business results. We employ a practical, partnering approach to transforming the design and architecture of your business and information technology environment to drive business performance.