This article is part of the IEEE Reliability Society 2009 Annual Technology Report.

Some Observations About Design for Six Sigma and Design for Reliability

Gary Jing

I have been leading or heavily involved in DFSS effort at several companies and made some interesting observations. Many of them are popular misunderstandings or fallacies that many people pay no much attention to. Some of them are worth to be shared with the Six Sigma related communities.

The Most Wanted Stuff: Non-technical vs. Technical

One interesting phenomenon I notice is that when people talk about DFSS, they are more frequently referring to technical stuff or tools, yet VOC research shows the more frequently cited needs might be the non-technical stuff or "soft" tool. Some Lean practitioners claim that for Lean activities, 80% needed are people skills and 20% are technical. I'll say for Six Sigma including DFSS, although more technical in nature, it's somewhat true as well, maybe 60% people and 40% technical. To be successful, people side of soft skills or activities are more critical than technical stuff. It's probably true in most of the initiatives or activities. This may in part explain why executives do not need to be technical to be successful.

So what are people's needs out of DFSS? A survey over key stakeholders of DFSS at a company showed eight major categories as an example. They are:

- Design for commonality
  - Platform design to meet multiple needs
  - Modulated design to avoid variant
  - Lack of business focus: Long tail of product SKU

- True and solid VOC
  - Latten or future needs
  - Global or similar needs
  - Evolving, fragmented or no performance requirements

- Knowledge sharing and organizational learning
  - Current VOC being fragmented and tribal knowledge
  - Needing a mechanism to encourage vertical and horizontal knowledge sharing

- Balance between speed (TTM) and quality
  - "Do it right the 1st time"
  - Time constraint to cut corner

- Fostering cross-functional collaboration between front end and back end.
  - Communication / visibility not as good
  - Disconnect between prototype and manufacturing

- Battling constraints of shared resources
  - Bottle neck
  - Different priorities among stakeholders

- Manual manufacturing process
  - Low R&R
  - Low capability
- Low use of statistics

As can be seen, majority of the VOC's is about collaboration / coordination, planning / sharing, system / structure, those non-technical "soft" stuff. Only the last two of the eight categories are technical. These categories are fairly generic and echoed at some other companies. This brings up the question of the focus of DFSS.

The Focus and Scope of DFSS

Although branded as DESIGN for (Lean) Six Sigma, the ultimate goal of DFSS should be sustainably successful new product introduction. The "sustainable" is a broad high mark here, including economical success over product lifetime and even beyond. Anything goes, as long as affects that goal. From this perspective, besides broad business management and execution need, at least four layers of activities are needed. They are:

- Portfolio Level:
  - Product portfolio management
    - Spend limited resources on the right products (projects) with right amounts (allocations).
    - Monitor progress with more visibility and rigor; kill as needed.
  - Resource management: Critical path centered coordination for shared resources.

- Project Level: Emphasize structured approach and tool applications to foster synchronized collaboration and improve deliveries.
  - Encourage information integration and activity coordination.
  - Manage progress and deliverables with more rigor.

- System Level (business dependent, optional):
  - Critical requirement management (requirement flowdown).
  - Transfer function between layers of system(s).

- Problem Solving Level:
  - DFSS tools:
    - Core tools (70%): VOC / QFD, FMEA, TRIZ, DOE / statistics …
    - Business dependent tools (30%): DFX (Design for xxx: reliability, manufacturability, assembly, modularity, lean)…
  - Overlap with traditional LSS extensively.

To sustain a good product introduction, all levels are needed. Usually each company has different level of competency and maturity in each of these areas. In a given environment, some of these areas may be already addressed effectively by existing system(s), such as new product development (NPD) process or portfolio management. DFSS in a way tries to plug holes in the existing system(s). It adopts what already existing and tries to bridge gaps, to make all areas reach the needed effective level. Since the development of four layers varies from company to company, the focus and scope of DFSS are likely to vary as well.

DFSS Curriculum: Generic vs. Industry Dependent Tools.

Surveying the field, the DFSS program isn't as matured as the DMAIC side. The DFSS structure varies from IDOV, DMADV, CDOV, etc. The length of the curriculum varies from three days (basic overview level) to four weeks (~BB level). The way I see DFSS body of knowledge is that there are some generic core activities / tools that everyone should have, which may account as few as only four items but may take as much as 70% of curriculum (mostly on statistics). And there are some highly business / industry dependent activities / tools, which may mean totally different things for different industries or products. For example,

- Core Generically Applicable Tools / Activities:

  - VOC – Measurable requirements
  - FMEA - Preventive & proactive activities
  - TRIZ - Creativities
  - DOE / statistics – Design effectiveness and efficiency

- Business / Industry Dependent Tools / Activities:
  - DFX - DFR, DFM, DFA, DFL, DFMdl (design for reliability, manufacturability, assembly, lean manufacturing, modularity, etc.)
  - Requirement flowdown
  - Modeling / transfer function
  - Simulation

Note some of the activities, such as VOC and FMEA, are not technical and should be participated by all parties / stakeholders; some, such as DOE, are highly technical and are only needed for technical people. With that I highly recommend the DFSS training to be tier based: the non-technical tier for the whole product development team (that can be equivalent to green belt level); and the deep diving technical tier for technical staff (that can be equivalent to black belt level). This concept seems to be pretty straightforward, but to my knowledge, it's not the case, at least not what's being done. DFSS programs across industry are typically single tiered. As promoted in one of my earlier articles [1], I highly advocate for tiered approach in Lean Sigma integration. Similar concept applies to DFSS too.

Notice the traditional Lean Sigma training (DMAIC) is Team Leader based, meaning only the belts are need to receive the training to be effective. Yet due to the nature of product development, DFSS training is better to be Team based, meaning the whole core members of product development team should take it, to promote and enable the collaboration of the team.

Design for Reliability, An Example of Heavily Industry Dependent Activities.
I notice many DFSS openings cite Design for Reliability (DFR) as one of the major responsibilities. A closer examination reveals that "this" DFR is not "that" DFR. The same terminology meanings quite different thing at different company or industry.

About a decade ago, when DFSS was just brought up and IDOV was yet coined, some people consider DFSS as the combination of a series of Design For X (DFX): such as reliability and manufacturability, etc. Yet the voice for a structured approach outweighed the voice for specific DFX techniques. The history went for a different direction. For quite a long time the emphasis of DFSS was on IDOV or DMADV type of structures and DFX has been left out of the picture until a much later time. I speculate the driver behind this is largely the urge to find a universally applicable practice, similar to that of DMAIC. DFX was left out largely due to its heavily industry dependent contents, which is hard to reconcile. Yet the nature of design and development requires more diversity in practices. That's the reason why there is a need for DFSS the first place. - DMAIC doesn't work as well in design and development environments. Up to today, the community couldn't settle on a single structure and is unlikely to settle in the future as well.

To illustrate the industry dependency, using Disk Drive industry as an example, which reliability is a major concern, DFR means four layers of activities:

- Reliability curve modeling
- Accelerated stress testing
- Reliability allocation
- Design for robustness

Among the four layers, the last two are more directly tied to DFSS, but the first two are not, which were started way before DFSS era.

For telecommunication network terminals, the need for reliability is quite different. There is no moving part, or frequent reconnections. Indoor products mostly operate under controlled environment. It's on or off at the time of connection. Once connected, it'll stay with the same performance wise. Failures afterward are more likely collateral damages caused by unwanted human interaction. The industry didn't see the need for allocation or reliability curve modeling under the current level of use. Reliability in this industry, besides accelerated stress testing, may mean design for easy connection, to avoid collateral damages.

Balancing Between the Risk and Efficiency.
One fine line DFSS effort needs to walk is to maintain a balance between risk control and efficiency. Failure to do so can quickly sink the momentum, and is a formula to fail. Many companies behave like a pendulum, swing back and forth between the two, reflecting the challenge and difficulty to maintain that balance. Almost all companies I ran into struggled with that. Most initiatives at the beginning have the tendency to over emphasize their advantages, strengths and benefits, and downplay their disadvantages and weakness, which may cause mis-assumptions and false expectations that will end up with dis-appointment.

One characteristic with DFSS (or NPD) is it emphasizes to front loading resources, to do a more solid job up front and to avoid more costly failures at the end. One typical pitfall is it swings the pendulum too far toward risk control side and gives up too much at the efficiency side, which will backfire a big time. It'll be much better off to strike a balance to begin with, which is easy to see at the hind side, but extremely hard to judge beforehand. In fact most of the companies aimed at and believed they reached a balance at the beginning, and only found themselves over shooting afterward. Measures suggested in this article, such as tierd approach and identifying limited core activities, help to strike and maintain a good balance during the process.

The Relationship Between Lean and Six Sigma
The discussion on balance brings up another interesting side note about the relationship between Lean and Six Sigma. I had the honor to meet with Jack Welch personally as a MBB of the Year finalist. I asked him to verify a report that he regretted for not being exposed to Lean first. Interestingly he side-stepped from that and quickly reaffirmed that if you do Six Sigma well, you'll get what you need. This made me thinking about why there are many different and sometime conflicting to each other claims about Lean and Sigma, such as follows.

- "Lean and Six Sigma are essentially doing the same thing…"
- "Lean and Six Sigma are complementary to each other …" - Not the same thing.
- "Lean and Six Sigma are against each other…"
- "We don't need Six Sigma if we really do Lean well…" - From a Shingo award winner.
- "If you do Six Sigma well, you can get what you get through Lean…" - Jack Welch.

It turned out to be the relationship between Lean and Six Sigma does vary depending on the level of cognitions.

- At tool level, they are specialized, "**complementary**" to each other.
- At execution level, **conflicts** are observed.
- At business / philosophical level, as strategies to achieve business excellence, they are generalized, all-inclusive, and "**mutually-inclusive**". ("Doing the same thing").

Different levels of cognitions occur in DFSS as well. The four layers of DFSS applications is a reflection to that.

References:

Gary Jing, A Lean Six Sigma Breakthrough, *Quality Progress*, May 2009, 24 – 31.