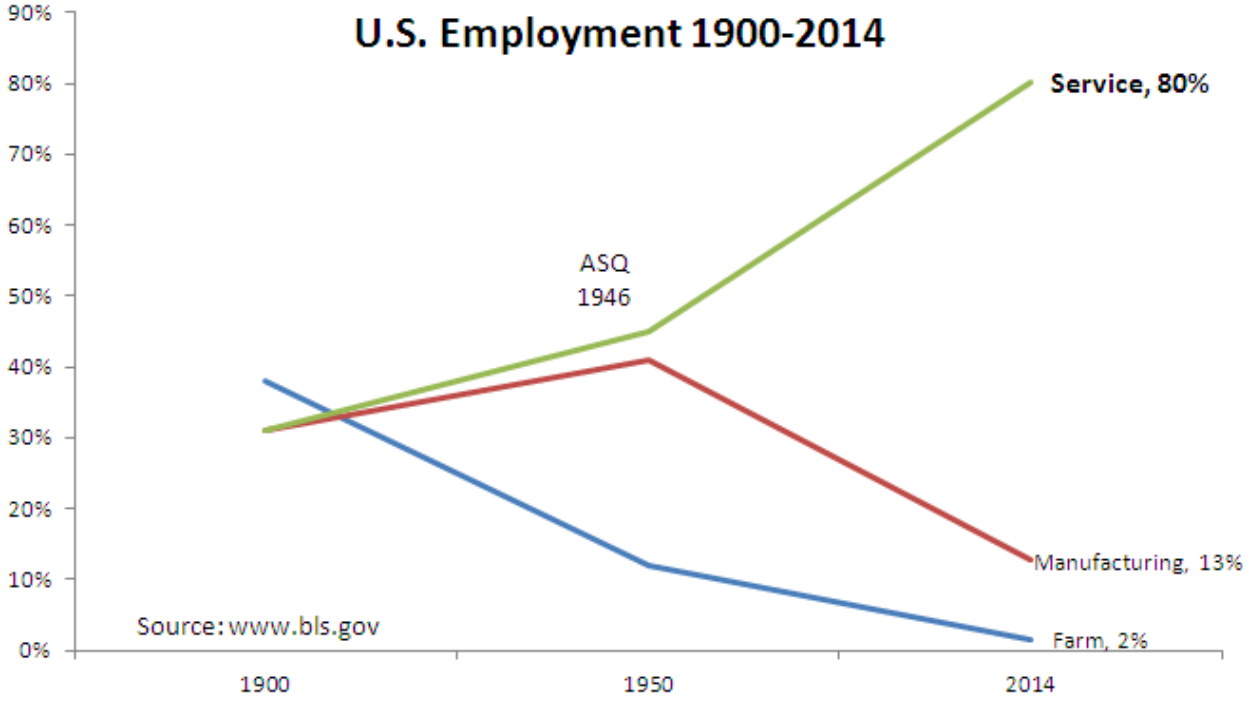
Agile Lean Six Sigma

By Jay Arthur

The January-February 2017 HBR article on problem solving found that 85% of C-suite executives say their companies struggle with *problem diagnosis*, not problem solving, and that this comes with a significant cost. (Joseph Juran often said that companies waste 25-33% of costs on waste and rework. This is the hidden cost of problems.) *Six Sigma is too complex and time-consuming to fit into a regular workday, s*ays Author, Thomas Wedell-Wedellsborg. *We need tools that don't require the entire organization to undergo weeks-long training programs.*

I'd like you to consider that the Six Sigma body of knowledge and training was designed for improving a manufacturing plant, although these jobs are now rare. That's why we need a simplified and streamlined approach to Lean Six Sigma that fits *services*.



Origins of Agile

Back in 2000, a rogue group of programmers got together and applied the principles of Lean to software development. They came up with a fresh approach to software that they named *Agile*. The existing method was called *Waterfall* development, It took too long and often failed to deliver the expected output.

The programmers came up with a new set of values and objectives.

The Agile Manifesto Values:

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation

- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan
- **Numerous small experiments** over a few large bets
- **Testing and data** over opinions and habits (sounds like Six Sigma doesn't it?)
- **Engagement and transparency** over posturing

I believe that we can adapt these values to Lean Six Sigma:

The Agile Lean Six Sigma Manifesto Values:

- **Results** over rigor
- **Working improvements** over comprehensive documentation
- **Cross-functional collaboration** over silo analysis
- **Responding to change** over following a plan
- **Numerous small experiments** over a few large bets
- **Testing and data** over opinions and conventions
- **Engagement and transparency** over posturing

Lean Six Sigma Agile Manifesto

By changing one word, *software* to *improvement,* I believe we can also adapt the Agile Manifesto. The Lean Six Sigma Agile Manifesto is based on twelve principles:

1. Satisfy customers by early and continuous delivery of valuable [improvements]
2. Welcome changing requirements, even late in the [improvement]
3. Deliver working [improvements] frequently (in hours or days rather than weeks)
4. Close, daily cooperation between business people and [improvers]
5. Build projects around motivated individuals, who should be trusted
6. Face-to-face conversation is the best form of communication (co-location)
7. Working [improvement] is the primary measure of progress
8. Sustainable [improvements], able to maintain a constant pace
9. Continuous attention to technical excellence and good design
10. Simplicity—the art of maximizing the amount of work *not done*—is essential
11. Best architectures, requirements, and designs emerge from self-organizing teams
12. Regularly, the team reflects on how to become more effective and adjusts accordingly

Hacking Lean Six Sigma

Hackers try to build the best services over the long term by quickly releasing and learning from smaller iterations rather than trying to get everything right all at once. -Mark Zuckerberg – *The Hacker Way* (https://www.wired.com/2012/02/zuck-letter/)

Hackers would rather just prototype something and see what works.

The spirit of hacking [can] be adapted and applied to general business management, not just technical innovation. Scott Brinker – Hacking Marketing

Modern performance improvement has a lot in common with software development.

Parallel revolutions

- From rigid processes to agile
- From maintaining the status quo to accelerating speed and adaptability
- From big projects to evolving customer experiences
- From silos to engagement
- From complexity to simplicity

Hacking Lean

Is it possible to accelerate the adoption and use of Lean methods and tools by *hacking Lean*? By *hacking*, I mean the good kind of hacking—simplifying, streamlining and optimizing the methods and tools to maximize results with minimal investment. I believe the answer to this question is YES!

While Lean came from the world of manufacturing—the Toyota Production System (TPS), 80 percent of American businesses are service businesses. Can these businesses use a subset of Lean to maximize performance? Yes, again.

We now live in a digital world. The key characteristics of the digital world, according to Scott Brinker, author of *Hacking Marketing* are speed and adaptability. Brinker says the challenges in a digital world are:

1. How do we execute faster?
2. How do we resist unwise knee-jerk reactions or overheated churn?

If you look at Lean Six Sigma training, it is essentially unchanged for decades. I believe that this model no longer serves U.S. service businesses and slows the tempo of change.

DOWNTIME

The speed bumps of Lean use the acronym DOWNTIME:

- **Delay**
- Overproduction
- Waste and Rework
- Non-value-added processing
- Transportation
- Inventory
- **Movement**
- Employee creativity (unused)

In manufacturing companies, most of the unnecessary delays and movement have been eliminated from the factory floor. But in service companies, delays between processing steps represent the vast majority of performance problems. To eliminate unnecessary delays, employees only need two tools: value stream mapping and spaghetti diagramming to redesign process flow and movement.

That's it! That's all you need to start hacking Lean in a service environment. Joseph Juran said: "Vital few, trivial many." It applies to methods and tools as well as improvements.

Lean Sprints

The week-long Kaizen blitz is a thing of the past unless you work someplace where you can shut down for five days. You just can't shut down a nursing unit for a week. You have to transform the unit in operation. This suggests the need for an *agile* approach to improvement. What is *agile*? A method that "is characterized by the division of tasks into short phrases of work and frequent reassessment and adaptation of plans." In other words, a *sprint*.

Using a 1-2 hour *timebox*, focus on one element of improvement at a time. A series of these low-stake bets will find the best solution more quickly.

All you need are a pad of Post-it® Notes and a flipchart to map the process flow or movement. This can be done in less than an hour. Once you know the current state, you can design a desired state that eliminates delay and movement. This can also be done in less than an hour. Then it's just a matter of changing procedures and layouts to achieve the desired state. Some can be done immediately; some may take a while.

In an hour or less, I've seen nurses redesign a nursing unit in ways that reduce nurse travel by 50 percent or more. This means more time with patients, better patient outcomes, better patient satisfaction and better nursing satisfaction.

In an hour or two, I've seen a computer operation group rethink nightly batch processing and reduce the cycle time from nine hours to one.

Eliminating unnecessary delays and movement will automatically reduce waste and rework because you eliminate the opportunity to miss a step or do a step twice. It may automatically reduce overproduction, inventory and non-value-added processing.

It doesn't matter what service you provide, there's always a way to reduce unnecessary delay and movement. And when there's nothing left to improve, turn your attention to the other elements of DOWNTIME.

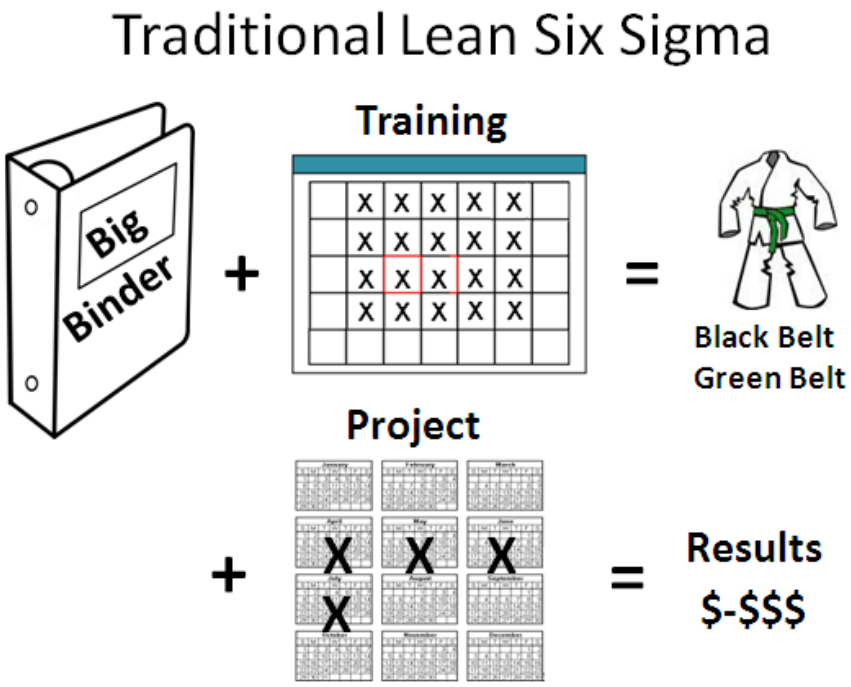
EIEIO – Experiment - improve early, improve often

Hacking Six Sigma

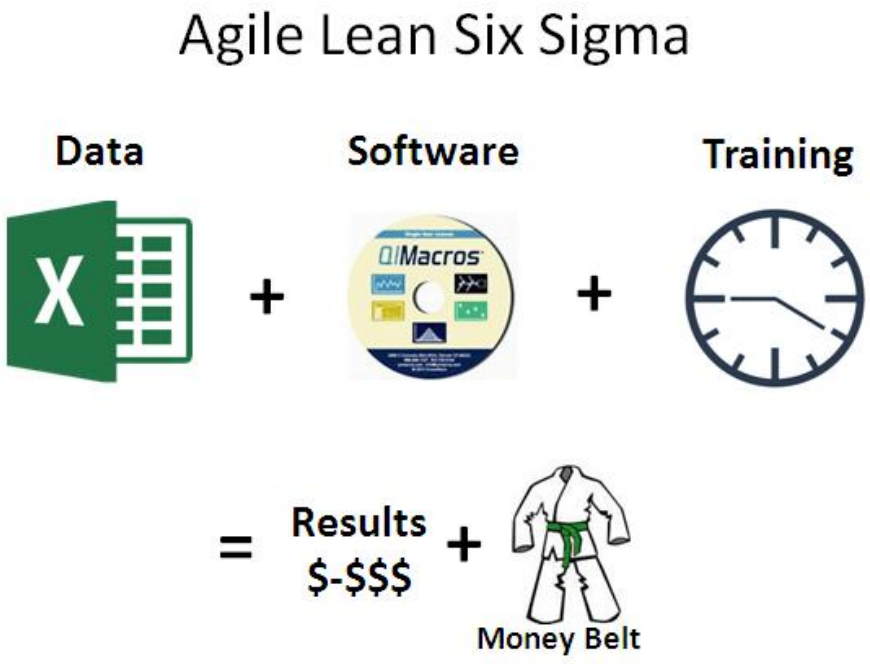
Is it possible to accelerate the adoption and use of Six Sigma methods and tools by *hacking Six Sigma*? I believe the answer to this question is YES!

While Lean Six Sigma came from manufacturing, 80 percent of American businesses are service businesses. Can these businesses use a subset of Lean Six Sigma methods and tools to maximize results? Yes, again.

Traditional Lean Six Sigma implementations take months or even years to see results:



Agile Lean Six Sigma training takes a different approach. Using existing defect data and software, we can train employees in a day or less *and* solve real problems:



Hacking is the art of invention. – Scott Brinker

Hacking just means building something quickly or testing the boundaries of what can be done.
-Mark Zuckerberg – *The Hacker Way*

Recommendation: Stop managing for activities (teams started) and start managing for outcomes (bottom-line, profit-enhancing results).

Vital Few, Trivial Many

A handful of tools will solve most business problems, especially in service industries:

- PivotTables – to find the *invisible low-hanging fruit*
- Control Charts – XmR chart of defect rates
- Pareto Charts – Types of defects
- Histograms – cycle (turnaround) times
- Fishbone (Ishikawa) Diagrams – root cause analysis
- Countermeasures and Action Plans (Matrices)

I have found that these five tools will diagnose and solve the vast majority of operational problems. Are there more *complicated* problems that need more exotic tools? Of course there are, but first solve all of the *uncomplicated* problems.

Agile Principles

Under an agile business management model, agile techniques, practices, principles and values are expressed across five domains:

1. **Cross-functional engagement** – to embed customers and suppliers within any project process to share accountability for product/service improvement.
2. **Facilitation-based management** – adopting agile management models to facilitate the day-to-day operation of teams.
3. **Agile work practices** – adopting specific iterative and incremental work practices.
4. **An enabling organizational structure** – with a focus on staff engagement, personal autonomy and outcomes based governance.
5. **An education model** that blends agile practices and philosophies to create micro-schools that emphasize collaborative culture creation and self-directed learning.

Agile Litmus Test – From Agile Project Management for Dummies, Wiley, 2012

1. Does what we're doing at this moment support early and continuous delivery of valuable improvements?
2. Does our process welcome change and take advantage of change.
3. Does our process lead to and support the delivery of working improvement?

4. Are improvers and doers working together daily? Are customers and stakeholders working closely with the team?
5. Does our environment give the improvement team the support it needs to get the job done.
6. Are we communicating face-to-face more than phone or email?
7. Are we measuring progress by the amount of working improvements produced?
8. Can we maintain this pace indefinitely?
9. Do we support technical excellence and good design that allows for future changes?
10. Are we maximizing the amount of work not done?
11. Is this development team self-organizing and self-managing? Does it have the freedom to succeed?
12. Are we reflecting at regular intervals and adjusting our behavior accordingly?

Benefits of Agile Lean Six Sigma

1. Better improvements
2. Higher satisfaction
3. Higher team morale
4. Increased collaboration and ownership
5. More relevant metrics
6. Improved performance
7. Improved project predictability

Agile Metrics

- Sprint success rates – LSS velocity
- Defects
- Total project duration
- Time to implement
- Total project cost
- ROI

How to Collapse the Lean Six Sigma Learning Curve

Have you ever tried to learn something one way and found it too hard, but then you tried another way and it was easy? I grew up in Tucson, Arizona in the heart of the Sonoran desert. When I moved to Denver, my coworkers hauled me up to the ski slopes, helped me rent 5.5 foot skis and sent me off to the bunny hill for lessons. The instructors used the *snowplow* method to teach us how to ski—you shape the skis into an inverted V and bend your knees in to carve a turn. I couldn't get it.

A couple of years later, the ski club at Bell Labs invited me up to Vermont with the promise of a magical new way to learn to ski called *graduated-length* method. I started a 9am on 3.5 ft. skis. Graduated to 4.5 ft. skis by lunch. Moved up to 5.5 ft. skis the next morning. *I went from zero to hero in 24 hours*.

Can we do the same thing with Lean Six Sigma?

The January-February 2017 HBR article on problem solving argues that Six Sigma is *too complex and time-consuming to fit into a regular workday. We need tools that don't require the entire organization to undergo weeks-long training programs.*

How do we solve the Six Sigma learning problem?

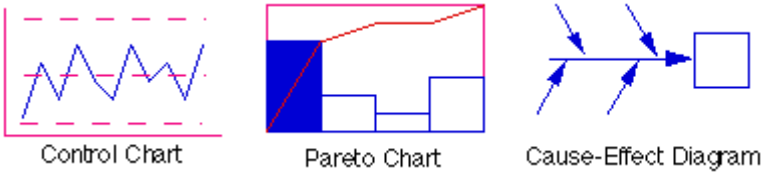
- First, remove everything that is purely manufacturing focused (e.g., hypothesis testing, DOE, etc.) because over 80% of U.S. employees work in service industries.
- Second, eliminate formulas and manual calculations. Six Sigma trainers forget that most people are *terrified* of math and statistics.
- Automate everything that can be automated.

How to Collapse DMAIC

Problem: Most Six Sigma Training spends at least three days on the methods and tools of Define, Measure and Analyze. When you spread the training out over multiple days, it makes Six Sigma seem slow and cumbersome.

Solution: Use the QI Macros Data Mining Wizard on Raw Data

- One minute of Data Mining Wizard to turn raw data into the PivotTables, control charts, Pareto charts and fishbones required for root cause analysis.



- Two hours of understanding the output and selecting projects

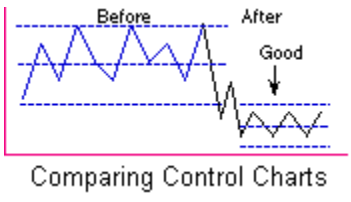
How to Collapse Control Chart Training

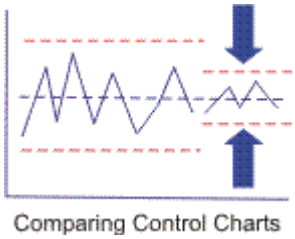
Problem: Five Day Control Chart Classes focus on manually calculating and drawing XmR, XbarR, XbarS, c, np, p, u charts. There is no reason to do this manually. I used to spend hours teaching people how to navigate a control chart decision tree; now I don't.

Solution: Use the QI Macros Control Chart Wizard to select the right chart automatically.

- One minute using QI Macros Control Chart Wizard to create a control chart.

- Two hours of *understanding the patterns* in control charts:
  - Special Causes – out-of-control points that should only occur 3 times out of 1000 opportunities
  - Common Causes – variation
  - Outcomes: 1) move the center line up or down, 2) reduce variation.





Once people understand the patterns they are looking for and the outcomes to be achieved, they quickly orient themselves to figuring out how to *win the game*.

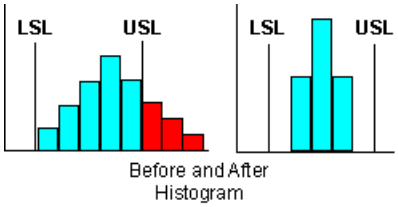
How to Collapse Capability Analysis

Problem: Capability Analysis training that focuses on manually drawing histograms and calculating Cp, Cpk, Pp and Ppk.

Solution: Use the QI Macros Histogram or Capability Suite

- One minute to create a histogram or Capability Suite consisting of a control chart, histogram, normal probability plot and so on.
- Two hours of *understanding the patterns* in histograms:
  - Capable/Not Capable (Cp>1, Cpk>1)

- Outcomes: 1) center the data, 2) reduce variation.



Before and After Histogram

Again, once people understand the desired transformation, they will figure out how to *win the game.*

- Consider using Weibull histogram for non-normal data if needed.

How to Collapse Hypothesis Testing

Problem: Hypothesis testing mainly focuses on comparing central tendency and variation. This takes a day or more. Honestly, I've never used hypothesis testing because I don't work in manufacturing or research. I work on achieving business results: speed, quality and profitability.

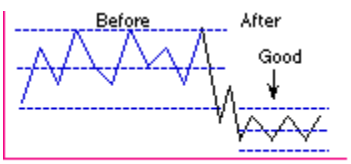
Solution: Eliminate Hypothesis Testing all together. Stop trying to turn people into statisticians. If you must teach hypothesis testing, use the QI Macros Stat Wizard.

- One minute of Stat Wizard to do t-tests, F-tests, ANOVA, or Levenes test on sample data.
- One hour of understanding the results:
  - Forget the statistical jargon (e.g., "cannot reject the null hypothesis"), focus on results:
    - Means are the same or different
    - Variation is the same or different
    - What to do in either case

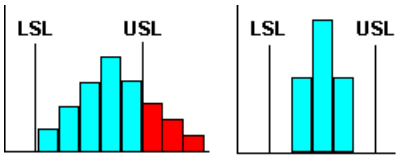
Non-Normal Data

Problem: People agonize over what kind of data they have and what to do with it.

Solution: Don't cover non-normal data. Control charts are robust to non-normal data. Service turnaround times are often non-normal, but the goal is to reduce variation and move the average.



Comparing Control Charts



Before and After Histogram

Team Dynamics

Problem: Most Six Sigma training includes topics like brainstorming, multivoting, team dynamics (form, storm, norm, perform), etc. I have found that given a meaningful project to work on, teams will jump into action.

Solution: *Let the data pick the project*. Teams should *never brainstorm what problem to solve, because they usually want to fix someone else—customers, suppliers, management or subordinates*. Instead, use the Data Mining Wizard to analyze raw data. The team will self-organize to solve the problems revealed. Only teach specific methods when they are needed for a real project.

Measurement Systems Analysis (MSA),

Problem: Most Six Sigma courses teach MSA, GageR&R, Design of Experiments (DOE) and other complex tools only used on manufacturing factory floors.

Solution: Don't teach these methods to service companies. It's a waste of time and causes confusion. It's a form of *overproduction*, which violates a basic rule of Lean.

Six Sigma History

Problem: Most Six Sigma courses teach the history of Six Sigma from Shewhart, Juran and Deming to the present.

Solution: Point people to excellent online references about this *if they want to know the history*.

Comparing Lean Six Sigma Bodies of Knowledge

I took the ASQ LSS Yellow Belt Body of Knowledge (BOK) and stripped out everything *not* required to solve problems in a service environment. The word count dropped from 1281 words to 144, an 89% reduction. Think of it as *Leaning* Lean Six Sigma.

I believe that LSS Yellow Belt training is all that is required to solve the vast majority of problems facing American business. Anything else is overkill and problematic.

I did the same thing with the LSS Green Belt BOK. The word count fell from 3133 to 2178; a 30% reduction. I think GB and BB trainings are ideal for people who work on a manufacturing factory floor, but they represent less than 2% of the workforce. About 2% work in agriculture. Everyone else works in services.

When I went to college, I learned operations research, differential equations and all kinds of stuff that I might need. *I have never used any of it.* Similarly, teaching service employees how to improve a factory is a foolish waste of time, money and resources.

If we can get people using software to solve problems immediately using real data, they will embrace Lean Six Sigma and accelerate the quality transformation. So far, I have seen only limited signs of real improvement across many industries. Why not?

I believe it's because we're still trying to teach Six Sigma the same way it has been done for the last two decades to prepare people for non-existent jobs in manufacturing. Most Six Sigma software is cumbersome and doesn't help analyze data, select the right charts or statistics automatically. The QI Macros simplifies, streamlines and optimizes the learning experience. I routinely take people from zero to hero in one day using QI Macros.

We can cling to our traditions or embrace a new and improved way to engage people in problem solving that gets results. Isn't it time for *Agile Lean Six Sigma?*