

Do's and Don'ts We Learn from Our Past

— Mary DeFoe, CSQA, Principal of [Integral Process Solutions](#)

In my early teenage years, my mother taught me to golf. She showed me how to hold the club, swing, count my strokes and be courteous to other players. As a typical teenager, I didn't want to be told too much. My mother's insistence on the basics wasn't always welcomed. I wanted to have fun with the game in my own way. When I struggled with the game, she would step in to help. To my surprise, my improvement came from this coaching, and measuring my progress in putts and total score.

As I grew older, I came to appreciate what my mother taught me. Because I understood the basics of golf I was able to compete in golf leagues and enjoy my weekend games. I was able to improve from coaching, practicing, taking additional lessons from golf pros and keeping score.

This is similar to building software. Software development basics and theory were subjects offered in college. In the real world, senior engineers mentored me to perform my job. Over time, I learned that these senior engineers were skilled at their craft but didn't rely on documentation to perform their jobs. Their improvement came from trial and error and most of their knowledge was in their heads.

I remember asking a couple of senior engineers for the software development methodology. To my surprise I received a smirk, a chuckle and was directed to a shelf with some manuals. Even though productive, their process wasn't repeatable for new engineers coming onto the scene. So what will happen when these skilled resources leave the job market and the incoming resources don't have the expertise to maintain or build new systems?

To achieve success, organizations need to tap into the experience of their resources and assess the shape of their software development process. The following are important questions to consider: (1) How much software development process is repeatable and documented? (2) What measures are in place to monitor progress and ensure success? (3) Is ongoing training available for new resources and advanced training for experienced resources? and (4) How are we transitioning knowledge to our resources?

There are plenty of opportunities to repeat good things from the past:

- ***Do understand the basics of software development but up your game.*** Included are learning the organization's approach to software development procedures, standards, expected deliverables, and quality check points.
- ***Do learn from mistakes.*** Establish quality review checkpoints. Review what was or wasn't successful. Understand the activities you want to leverage and adjust as the work proceeds.
- ***Do understand your handicap.*** Establish a baseline set of measures for projects and ongoing maintenance. The most common are effort, size, cost, schedule and quality.



© Integral Process Solutions. All Rights Reserved.

info@integralprocesssolutions.com — Mary: 651-260-9954 • Bobbi: 651-260-9979 — www.integralprocesssolutions.com/

There are plenty of opportunities to avoid mistakes from the past:

- ***Don't let pride get in the way.*** Many of us get set in our ways and don't even realize we resist change when it comes. Discomfort is ok when trying new changes. It usually brings us to a new understanding of a component within the whole process.
- ***Don't resist help when it's offered.*** New individuals will always bring new ideas. Listen to the coaching that is provided. To achieve lasting success, aid these subject matter experts to understand the past and the culture. Try new ways of doing things and work together for effective improvement. The change may not work the first time, but you can course correct.
- ***Don't forget to analyze the score.*** Once the data is collected from your process, it's important that you look at the data for themes and messages that can be found. Many times, teams hurry onto the next project without looking at the objective data that is readily available for future process improvement.

So whether you focus on your sport of choice or software development, consider what you know today and how you can learn from the past to improve in the future.



© Integral Process Solutions. All Rights Reserved.

info@integralprocesssolutions.com — Mary: 651-260-9954 • Bobbi: 651-260-9979 — www.integralprocesssolutions.com/