O'REILLY

J. Beningo, *Embedded Software Design*

Next Steps

Jacob Beningo[1]

(1) Linden, MI, USA

Where Are We Now?

Throughout this book, we've explored the three areas that must be balanced for an embedded software team to deliver their software successfully:

- Software Architecture and Design (Part 1)
- Agile, DevOps, and Processes (Part 2)
- Development and Coding Skills (Part 3)

I call these three areas the embedded software triad because if you don't successfully balance them, the development cycle will be a boondoggle. Success might come, but it will be through painful and overexerted effort.

In each part of the book, we looked at the fundamentals necessary for you to start understanding what each area entails and some of the techniques required to help you be successful. Obviously, given the time constraints and the size that a typical book can be, we couldn't cover everything. However, we have covered enough for you to successfully figure out where you are today and what you need to do to start to balance your development cycle.

You may be a single developer or perhaps the leader of a team of developers. No matter what the case may be, completing this book is just the start of your journey. I've shared some of my insights, skills, and experiences working with teams worldwide and in many industries successfully developing embedded software. While I hope this has been helpful to you, we're now at a point where you need to decide how you will take what we've discussed and apply it.

Defining Your Next Steps

You can take several approaches to decide where you go from here. But, no matter which direction you take, there are several questions that we need to answer before you race off into the sunset.

First, you need to understand where you are today. The embedded software triad is meant to help you understand how to break up everything that we must know and deal with and visualize it in a simple chart. As we have seen, balanced teams are more likely to succeed, so a good first step is to evaluate where you currently are. Is there an area that you are ignoring? Are there skillsets that you are missing? If you haven't already done so, you might consider taking my survey, **www.surveymonkey.com/r /7GP8ZJ8**, which will give you a high-level overview of where you are. You may find that taking a quick 50-question survey doesn't give you the level of detail you need. If that is the case, and you'd like professional assistance, you can always reach out to me, and I can help point you in the direction that may make the most sense for you and your team.

Next, once you know where you are, it's time to decide where you want to go. You can look at your personalized survey results and determine where you may need to add more focus. You can also survey your team or just look back over the chapters in this book and see where your passions lay. Perhaps improving testing is exciting, and the idea of improving your software quality is at the top of your list.

My recommendation is not to try to bite off too much at once. Instead, start by defining no more than three goals that you would like to accomplish in the next three months. Once you know what those improvements are, put the time and budget necessary to make incremental improvements. Once you achieve those initial goals, review where you are, define where you want to be next, and then repeat.

The one truth that I have seen so far in my career is that the developers, managers, and teams that focus on consistent and incremental improvements are the ones that succeed the most. I've often seen them grow beyond many of their own wildest expectations. Of course, discipline is re-

quired, but the potential results can be staggering.

Choosing How You Grow

When I was a kid, I loved hockey. I watched every Red Wing game I could and read everything I could about hockey. I read books about hockey skills and how to improve my hockey skills. Consuming all these materials helped me to understand the game and how to play it. But unfortunately, they didn't do anything to help me improve my hockey skills because I wasn't applying what I was reading!

I had to get skates, a hockey stick, and a puck to improve my hockey skills and go out and play. Playing hockey and getting the experience tremendously helped me take my book knowledge and apply it to the real world of hockey. Practicing and playing hockey on my own helped to improve my skills, but I did discover that in some areas, no matter what I did, I couldn't get better! For example, try as I might, I could not figure out how to hockey stop! My version of stopping was to just plow into another player or the boards. Not very skillful!

I spent years on my own trying to learn to stop. (Seriously, no joke. It was pretty embarrassing.) Then, when I got to college, I decided to take a hockey class for fun and additional exercise. Lo and behold, the teacher discovered I couldn't stop. For the first few weeks of the course, he coached me one on one how to properly shift my momentum and better utilize the edges on my skates, and, finally, I was able to stop! I had been attempting to figure out how to stop for years, using advice from friends and family, and it just took a coach a few weeks to get me there.

I know you're wondering what my hockey skill woes do with embedded software or getting a product to market. My story demonstrates the different ways you can learn and improve your team and the rate at which you can expect results. You can engage your team and go it alone. You will improve and gain the processes and skills you need to succeed. However, you may sometimes struggle or find that you are progressing at the rate you want.

The best athletes worldwide have coaches who watch what they do and give them expert feedback. I know companies whose policy is that every manager has an internal or external coach. I'm a solo consultant and business owner. I have mentors and coaches who are colleagues and even paid business coaches that help give me feedback and whom I can bounce ideas off. I've learned from my hockey experience years ago that I sometimes need external assistance and can't do it all alone for rapid, continued progress. (Or if I can, I can do it faster with some guidance.)

I suggest that you figure out the best route for you and your team to meet your company's goals. For example, you might be able to set up a good, disciplined process and skill enhancement program on your own. Or, you may know you have issues but aren't sure how to resolve them. I've often seen that the teams making waves in the industry are the ones that are willing to admit they're drinking their own Kool-Aid and need some external advice and feedback to keep them on the right path. I've sometimes even seen where a team internally says they need A, but management doesn't agree until a third party comes in and says that the company needs A.

I'm not trying to pitch you anything. I've personally benefited and seen my clients benefit from mentoring and advice. I suggest you carefully examine what it will take for you and your team to succeed. There are many ways that you can ensure you and your team grow and are on the right path. For example, you can pair junior and senior engineers together. You can pair managers in cross-function teams. There are many online communities as well that engineers can join to get questions asked and interact with experts all over the world[1] (and yes, even consultants).

We often look at our embedded software through the lens of developers. At the end of the day, we are contributing to the success or failure of the business we work for. Do what needs to be done and is in the company's best interest. When you do that, you'll find that you end up with higher-quality software, controllable software budgets, and predictable times to market. Your end customers will love it, and management will too!

Final Thoughts

There are a lot of resources, both free and commercial, that you can use to help take your embedded software development cycle and skills to the next level. However, as we close the book, I have a few recommendations.

First, don't overlook the Appendix in this book. There are several tutorials that I believe you'll find exciting and will help you apply some of the topics discussed in this book. So make sure that you take full advantage of it!

Next, there are many great resources that you can find to continue your embedded software journey. For example, you may find the following sites to be helpful to you:

- **www.beningo.com**
- **www.embeddedonlineconference.com**
- **www.embeddedrelated.com**
- **www.embedded.com**
- **www.designnews.com**

I may be a little biased toward these sites since I regularly contribute content. However, I feel that these resources have some of the best free and paid materials in the embedded software industry. So don't hesitate to take advantage of them.

Finally, if you ever discover that you are unsure where to go from here or get stuck, you can always reach out to me at jacob@beningo.com. I'll do what I can to direct you to any additional resources or discuss how I can be of assistance.

I've appreciated your time and attention throughout this book. I hope you have found some new insights, acquired new ideas, and will drive your embedded software development cycle and skills to the next level. Until next time, happy coding, and good luck!

Action Items
Our journey through this book has ended, but your embedded software journey may just be beginning. After that, it's up to you to choose your next adventure. The

following are a few more summary thoughts for you to consider for what may be next:

- Where are you today? Take time to figure out where you and your team are at.
  - Are you lacking skills in any area of the embedded triad?
  - Is your development cycle out of balance?
  - What challenges continue to trip you up, cause you stress, or deliver late?
- Where are you going?
  - What changes do you need to make to be more successful?
  - What does a balanced development cycle look like to you?
  - What improvement can you make to improve the impact of your software on your customers?
- How are you going to get there?
  - Are you going to go it alone?
  - Engage with your team?
  - Leverage an external expert to coach and advise you.

Footnotes

1 Think LinkedIn groups, embeddedrelated.com forum, etc.