AGILE IN A NON-AGILE WORLD: APPLYING AGILE PRINCIPLES TO SAFETY-CRITICAL SOFTWARE

Jeffery Payne
jeff.payne@coveros.com
www.linkedin.com/in/jeffery-payne-21373
@jefferyepayne
Jeffery Payne is CEO and founder of Coveros, Inc., a company that helps organizations accelerate the delivery of secure, reliable software using agile methods. Coveros has been recognized as one of the Top 10 Emerging Agile Companies, Hottest Startups in DC, and one of the fastest growing private companies in the country (Inc. Magazine)

Prior to founding Coveros, he was the co-founder of software assurance company Cigital, where he served as Chairman of the Board and CEO for 16 years.

Jeffery is a recognized software expert and popular keynote speaker at both business and technology conferences on a variety of software quality, security, DevOps, and agile topics. He has testified in front of congress on issues such as digital rights management, software quality, and software research.

Jeffery is also the technical editor of the AgileConnection community (www.agileconnection.com)
About Coveros

• Agile & DevOps Engineering to help organizations accelerate the delivery of secure, reliable software

• Services
  • Agile transformations & coaching
  • Agile software development
  • Agile testing & automation
  • DevOps engineering
  • DevSecOps implementations

• Agile, DevOps, Testing, Security Training

• Open source products
  • SecureCI – Secure CI/CD stack
  • Selenified – Agile test framework
“Agile Doesn’t Work For …”

- Safety-critical systems
- Regulated software
- Large-scale applications
- Security-critical systems
- Legacy applications
- Embedded applications
Agile Is Misunderstood

The Agile Manifesto

<table>
<thead>
<tr>
<th>Individuals and Interactions</th>
<th>over</th>
<th>Processes and Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>Working Product</td>
<td>over</td>
<td>Comprehensive Documentation</td>
</tr>
<tr>
<td>Customer Collaboration</td>
<td>over</td>
<td>Contract Negotiation</td>
</tr>
<tr>
<td>Responding to Change</td>
<td>over</td>
<td>Following a Plan</td>
</tr>
</tbody>
</table>

*That is, while there is value in the items on the right, we value the items on the left more.*

www.agilemanifesto.org
When Most Regulations Were Written

The Smart and Easy Way to Keep in Touch

- It's a great telephone.
- It's an easy way to send and receive e-mail.
- It's one-touch access to the Internet.

Designed with simplicity in mind, the iPhone combines a telephone with web browsing, e-mail, speakerphone, and a digital answering machine—all in a single, easy-to-use appliance.

advantage of all that the Internet has to offer. Simply put, the iPhone makes using the Internet as easy as using a telephone.
• Coveros hired by medical device company to transition their software division to an agile process

• *Business driver:* accelerate software delivery while assuring patient safety
  • Software releases were taking 2 years to complete
  • Organization concerned about emerging competition that was more nimble and quicker to market
  • Software team had been promising improvements for years with no success
• Transformation took nearly 2 years
  • Assessment – 3 months
  • Build / Pilot – 9 months
  • Scale – 9 months

• Not everybody made it through the journey

• Investment of $1M - $2M
Assessment and Planning Phase

• Evaluate current agility capabilities across Agile Principles
• Perform gap analysis against industry best practices
• Make recommendations for improving people skills, agile practices, tooling, team structure
• Create transformation backlog and team
• Educate business stakeholders
Build and Pilot

- Implement necessary agile practices, provide initial training, build paved paths for practices & tools
- **Select an appropriate project** to apply agile principles to
- Measure productivity and quality gains
- Iterate on approach as necessary
Scale Agile

• Spread agile capabilities across organization
• Spin up internal agile coaches to help other teams
• Provide fundamental training
• Help hire to fill the gaps
• Identify incremental scaling approach
• Re-assess agile maturity and identify next set of improvements

Coaches drive consistency and best practices

“Train and drop” model across all locations
Case Study Results

- Reduced software release delivery from 2 years to 6 months to 3 months
- Software changes can go through the delivery process in less than 1 week
- Repeatable, consistent software process with customers in the loop
- Integration with hardware design, development, testing, and verification
Lessons Learned Applying Agile

Regulations don’t restrict software process

• While many interpret regulations as mandating a waterfall approach, a closer read shows this is not the case

• Meeting regulatory requirements DOES require additional documentation but this does not conflict with agile

• Compliance activities and personnel must be integrated into day-to-day activities for agile to work (shift left)
Culture change is critical to making agile work and never more so than when dealing with regulated software.

Focus initially on team and organizational culture using agile principles to guide you.

Get specification, design, coding, testing, and verification all on the same cross-functional team.

But don’t forget the importance of sound engineering practices or you will ultimately fail.
Before transformation you often see:

• Heavyweight docs produced during process followed by significant rework and approval process

• Results in significant reductions in productivity for the team
Lessons Learned Applying Agile
Automate Your Documentation Delivery

After Transformation:

- Lightweight docs maintained in software during dev/test process
- Auto generation of docs needed for safety case and compliance
- Significant increase in productivity AND accuracy
Before transformation

• Device access constraints
• No continuous integration
• Manual device setup
• No test automation
Lessons Learned Applying Agile
Automate Your Software Delivery (cont.)

After Transformation

• Git supports distributed versioning
• Windows slaves provide emulators for dev and regression testing
• Jenkins to orchestrate delivery process
• Check-in tests performed in emulators and …
After Transformation

- ... in device farm
- Device access controlled with consoles to support all types of CI and CD automated testing
- Automated testing supplemented with exploratory testing
- Device communication with expect, legacy ‘sh’ shell, and IP locks
There is nothing stopping you from adopting agile principles other than inertia.

Focus on cultural change / building cross-functional team to get started.

Leverage engineering and automation to take your process to the next level and accelerate productivity.

Effectively balance speed with quality (and safety / security).

Agile has been used successfully to deliver safety-critical systems and passed regulatory scrutiny … you are not the first so go for it!
Thank You!

Contact Information:

Jeffery Payne
jeff.payne@coveros.com
@jefferyepayne