Agile at the Intersect of Mobile, Cloud, and the Internet of Things

Don MacIntyre
Don@ScrumEtc.com
www.ScrumEtc.com

Speaker Profile

- Agile Transformation Lead
- Agile Coach (CEC/CSC, SPC)
- Agile Leadership Coach (LA360)
- Former Executive
- Former Software Engineer
What are we going to talk about?

- Reflect on where Agile is being applied today
- Introduce the Internet of Things (IoT)
- Identify common patterns for success in applying Agile in the IoT and other places beyond traditional software development

Why is this important?

- The IoT is an explosive market
- Hardware centric companies are adding more and more software to their devices
- Hardware sales still drives their business
- Their traditional hardware based product development practices and lifecycle no longer suffice
- They need help applying agile to their hardware, embedded, and software environments
Where is Agile being applied in software today?

- Finance
- Healthcare
- Automotive
- Aerospace
- Government
- Virtually All Software Environments

How do we know?
Where is Agile NOT being applied in software today?

- In environments where Agile has failed
- In environments that are afraid of change
- In environments where there is no perceived need for change
- The use of Agile has little to do with the type of work being performed

Are some software projects TOO LARGE?

Let’s Assume

- Agile can be applied to **any kind of software**
- **No software project is too large** to benefit from some Scrum and/or Agile
- Is there a **sweet spot**?
Cloud Environments

- Massive Virtualization
- Frequent/Constant Updates
- Tools Galore
- Automated Test, Continuous Integration, Continuous Deployment, DevOps
- No excuse NOT to do be Agile

Mobile is Agile

“Samsung is using a faster method of developing smartphones called ‘Agile’ methodology. The adoption of the new method may bring forward the launch date of the Galaxy S7.”

Tech Times, Aug 2015

© Scrum, Etc.
### iPhone Refresh Cycle

<table>
<thead>
<tr>
<th>Days since last release</th>
<th>Sep 2015</th>
<th>311</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average</td>
<td></td>
<td>379</td>
</tr>
<tr>
<td>Recent releases</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sep 2014</td>
<td></td>
<td>365</td>
</tr>
<tr>
<td>Sep 2013</td>
<td></td>
<td>357</td>
</tr>
<tr>
<td>Sep 2012</td>
<td></td>
<td>364</td>
</tr>
<tr>
<td>Oct 2011</td>
<td></td>
<td>353</td>
</tr>
<tr>
<td>Jun 2010</td>
<td></td>
<td>467</td>
</tr>
<tr>
<td>Jun 2009</td>
<td></td>
<td>370</td>
</tr>
</tbody>
</table>

*Annual hardware refresh and major iOS update*

*Frequent minor iOS updates*

http://buyersguide.macrumors.com/#iPhone

---

### Much of Mobile is Embedded

- Years ago many people thought that **embedded** software was **different** and they could not do Agile
- **No-longer** the case
- Techniques have **always existed** and tooling continues to improve
- See **James Grenning's** work on Embedded TDD
Top 25 iOS Apps Update Frequency

Top iOS Apps Update Every 30 Days


The Internet Of Things

“Explosive Internet of Things Spending to Reach $1.7 Trillion in 2020” - IDC
Wikipedia Definition

“The Internet of Things (IoT) is the network of physical objects or “things” embedded with electronics, software, sensors, and network connectivity, which enables these objects to collect and exchange data. … Experts estimate that the IoT will consist of almost 50 billion objects by 2020.”


“Simply put, …

...this is the concept of basically connecting any device with an on/off switch to the Internet (and/or to each other).

This includes everything from cellphones, coffee makers, washing machines, headphones, lamps, wearable devices and almost anything else you can think of.” – Forbes May 13, 2014
Imagine for a moment...

You are designing and delivering software products...

- Is there value in...
  - Knowing where your software product is running?
  - Understanding what it is doing?
  - Verifying that it is working?
  - Allowing it to provide feedback?
  - Of course there is…
Now imagine for a moment...

You are designing and delivering **physical** products or ‘things’...

- Is there value in...
  - Knowing where your product is located?
  - Understanding what it is doing?
  - Verifying that it is working?
  - Allowing it to provide feedback?

- With software we take this for granted
The Internet of Things (IoT)

- The IoT allows ‘things’ to communicate
  - Where it is
  - What it is doing
  - How it is working
- But more importantly…
  - It allows ‘things’ to gather and share information about their environment

Where is your product?
Where is your product?

The other IOT: The Internet of Trucks

© Scrum, Etc.
Where is your product?

Forklift 2.0:
- Auto-Routing
- Auto-Lift-Height
- Crash Avoidance
- Worker Avoidance
- Maintenance Data
- Environmental Data

The Warehouse Of The Future Today!

© Scrum, Etc.
Interactive Retail Shopping

- **Beacons** to track customer movement
- Retailers will **know** what the **customer** was interested in
  - If he spent 5 minutes comparing coffee makers they could send him an email when the one he spent the most time with goes on sale
- **Real-time shelf inventory** (rather than POS)
- **Dynamic pricing** based on inventory or weather
- **Suggesting items** based on current or past purchases
Once your product is ‘home’

- WiFi
- App for Scheduling
- Firmware updates?
- New functionality?

Where is your product?
Wearables

© Scrum, Etc.

Patient Tracking
Doctor Tracking
Nurse Tracking
Heart Rate
Blood Pressure
Blood Sugar
Temperature
Activity Level
Sleep Quality
Hearing Aids
Vision Quality
Concussion - Detection
Infrared Pain - Therapy

Wearables in Healthcare

© Scrum, Etc.
Wearables in the NFL

Energy
Agriculture

“The 8R line of tractors has more lines of code than the space shuttle”
- Samuel Allen, CEO John Deere

Agriculture
Automotive

How many lines of code are in a toilet?
So what does this have to do with Agile?

- Hardware centric companies are adding more and more software to their devices
- Some traditional manufacturers are finding themselves shifting toward software and services
- Traditional development practices are not enough
- Agile can help in this non-traditional environment
- Agile is not just for software
  - Don’t be afraid to use it!

Patterns That Work Beyond Software
Leading Causes of Failed Agile Projects

- Company philosophy or culture at odds with core agile values: 46%
- Lack of experience with agile methods: 41%
- Lack of management support: 38%
- Lack of support for cultural transition: 38%
- Inconsistent agile practices and process: 38%
- External pressure to follow traditional waterfall processes: 36%
- Ineffective management collaboration: 34%
- A broader organizational or communications problem: 30%
- Unwillingness of team to follow agile: 30%
- Inability to continuously prioritize work: 28%
- Insufficient training: 27%
- Ineffective collaboration: 25%
- Don't know: 5%

Leaders Want ‘agile’ Organizations

- Business Performance
- Organizational Agility
- Leadership Agility

© Scrum, Etc.
Leaders Want Employee Engagement

**DRIVE** home how **Engagement** follows **Empowerment**

- Autonomy
- Mastery
- Purpose


Leaders Need Help Too

- **Provide Agile Training/Coaching** for Leadership teams
- Make sure those in charge understand **what Agile will mean** to their organization
- In **software** shops most people, even at the executive level, have **some** idea of what Agile is
- In a **non-software centric environment**, that may not be the case
Think Lean

- Hardware people likely know more about Lean than Agile
- Talk about Lean
- Create a Value Stream Map
- Build Quality In
- Minimize Waste
- Use Inventory Analogies
  - Not releasing software is just like not releasing inventory

Focus on Values and Principles

- Everyone needs to understand Agile values and principles
- Allow manifesto discussions to uncover concerns
  - Especially with leaders
- But don’t let the non-software folks get hung-up with the word “software” all over the manifesto
For Non-Software Teams, Considering Replacing ‘Software’ with ‘Value’ or ‘Product’

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

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

agilemanifesto.org

Agile Principles 1-4 with Value

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

2. Welcome changing requirements, even late in development. Agile processes harness change for the customer’s competitive advantage.

3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

4. Business people and developers must work together daily throughout the project.
Scrum Beyond Software

- Scrum can be used to manage any kind of work
- The Scrum framework and associated practices help teams learn how to work together as a team
- Scrum exposes all sorts of impediments
- Scrum teaches teams how to improve

Two Weeks!

- Hardware teams often initially panic when told they will need to deliver 'working product' to the 'customer' every 2 weeks
- Mechanical and electrical engineering ARE different than software engineering
- While it would be awesome to put production ready hardware in front of an actual customer every 2 weeks, that simply may not be possible at the present time
- Many hardware teams are used to delivering prototypes to other teams or special customers
  - Start there and then challenge teams to engage and improve
Scrum is not just for Software

If the time box of a Sprint is a real constraint, consider Kanban

Especially for ticket driven teams that are focused on working a queue rather than incrementally building a product

Awesome Scrum teams doing Continuous Deployment sometimes morph into Kanban

Caution: Kanban alone may not provide enough structure for new teams trying to become Agile
Understand Systems Engineering

- "Systems engineering is an interdisciplinary field of engineering that focuses on how to design and manage complex engineering systems over their life cycle"  

- Years ago Systems Engineering was common on large complex software projects

- Many companies building both hardware and software systems still rely heavily on Systems Engineering

- Systems Engineers sometimes make great Product Owners
  - Other times they cannot let go of the 'how'

Do Not Waterfall Your Agile Transition

- Non-software companies are often entrenched in a waterfall world

- Avoid 'waterfalling' your Agile transformation to fit their model

- Consider making your executive sponsor the PO for the transformation and Scrum your way to success

- Once you help them understand Agile they will understand why you cannot figure it all out up front
Focus on Incremental Improvement

- When helping **device manufacturers start with the application level**, then embedded, then hardware
- Avoid the **big bang** approach outside of pure software
- Allow a pilot project to **expose the pain points**
- Manage the pain
- Demonstrate, then duplicate, success
- **Don’t scale bad Agile**

Stepping Outside Your Comfort Zone

- **Do not pretend** to know it all
  - You may be new to non-software
  - The team may be new to Agile
- **Learn together**
- **Help** non-software teams **understand** Agile and then **ask them** how they can best apply it to their teams
- Help **guide them** to an Agile solution
Don’t Worry, Be Happy

- Other teams often see the software teams having fun and want to give it a try
- If you are not having fun then you’re doing it wrong
- If you have a high performing team having fun then you’re doing it right
- Happiness is a multiplier for productivity

The IKEA Effect

The IKEA effect is a cognitive bias in which consumers place a disproportionately high value on products that they partially created.

Bonus: These Patterns Work for Software Too!

Questions and Comments

Don MacIntyre
Certified Enterprise Coach
Don@ScrumEtc.com
www.ScrumEtc.com