

Alpha Testing as a Catalyst for Organizational Change

*Conference of the Association for Software Testing,
August 8-10, 2016, Vancouver*

By Steven Woody

The Internet of Things

Internet of Things

Items embedded with electronics, software, sensors, controllers/actuators, and network connectivity

- Smart grids
- Smart homes
- Smart cities
- Intelligent transportation

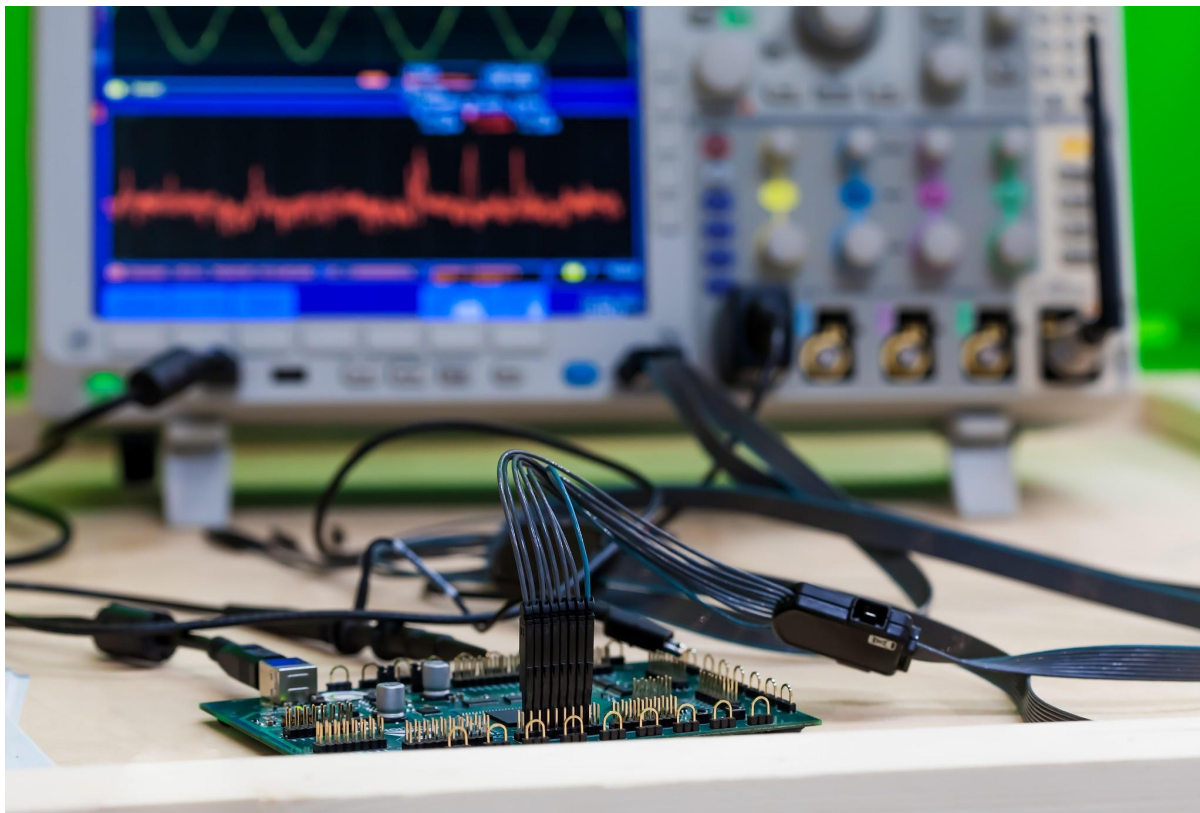


Key elements to alpha testing in the Internet of Things:

- Remote monitoring
- Remote troubleshooting
- Remote software updating

What is my definition of alpha testing?

First, lab testing...



My Definition of Alpha Testing:

- Field test of “the 80 percent solution”
- Friendly users (mostly employees) who were self-selected, expecting some issues, and willing to provide feedback
- Installation and operation in the end-user environment
- Actually using the product on a daily basis for its intended purpose
- Feedback report & weekly monitoring
- By definition an end-to-end systems test
- Long-running, continuous operational test of the product

Developing an Alpha Test Program

Developing an Alpha Test Program

- Be the Change: Start with yourself
- Widen the Circle: “Friends & Family”
- Watch Your Garden Grow: Crowdsourcing

Alpha Testing as a Catalyst for Organizational Change

A Catalyst for Organizational Change

How did alpha testing bridge the gap between makers and users?

Between engineers and the rest of the business?

How can alpha testing change an organization?

A Catalyst for Organizational Change

- Diminished Siloing
- Reinforced A System Viewpoint
- Built Intimate Product Knowledge
- Created Ownership
- Clarified the Use Cases
- Improved the Test Cases
- Demonstrated Actual Product Readiness

What We Learned

Where We Struggled

- Not “officially required testing”
- “Where are your test cases?”
- Participants understanding their role
- Adding more Usability Testing: Mock Installs
- Some hardware prototypes were just not ready
- Tricky to get the alpha timing right: not too early but not too late
- Too many alphas running at the same time - full time job
- More executive participants

Alpha Testing Drawbacks

- Alpha testing does not sufficiently exercise all features
- Alpha testing can still miss big bugs!
- Alpha testing can be seen as unnecessary, additional work
- Early alpha success may mislead management

Closing the Gap

How can software testers can get the benefits of alpha testing for other types of software products?

- Analyze field issues - how could have they been found in the lab?
- Continually strive to make your lab testing more like “the real thing”
- Keep regular contact with customers to understand use cases & challenges
- Create and maintain longevity test beds - six test beds with the oldest upgraded every 90 days gives a rolling 540 day longevity test
- Allow users to self-select themselves to participate in early access
- Consider expanding beta testing: earlier feedback from friendlier customers

Always Another Way

Always Another Way

How can alpha testing be used to offset limitations that commonly exist in test departments?

Always Another Way

Alpha testing to offset challenges that commonly exist in test departments

- Limited people resources?
- Limited test bed resources?
- Lots of testing, but not finding the important bugs?
- Poor testing?
- Bugs found in the lab are discredited?
- Beta testing is just too little, too late?

“Test as you Fly, Fly as you Test”
-- Jet Propulsion Lab (JPL) Mantra



References

- “Classic Testing Mistakes”, by Brian Marick, STARWEST 1997
- “Developing test cases from use cases”, by Ross Collard, STQE Magazine, July 1999
- "Ain't Nothing Like the Real Thing", by Jonathan Kohl, Better Software Magazine, February 2007
- “Software Longevity Testing”, by Steven Woody, Better Software Magazine, September 2009

Open Season

I want to hear about your challenges and successes with alpha testing. Thank you!

Steven Woody
stevenwoody1@gmail.com