

Learning from Open Source Software in commercial software development

How to change paradigms in software

2012-11-15

Mirko Kiefer - mail@mirkokiefer.com

Outline

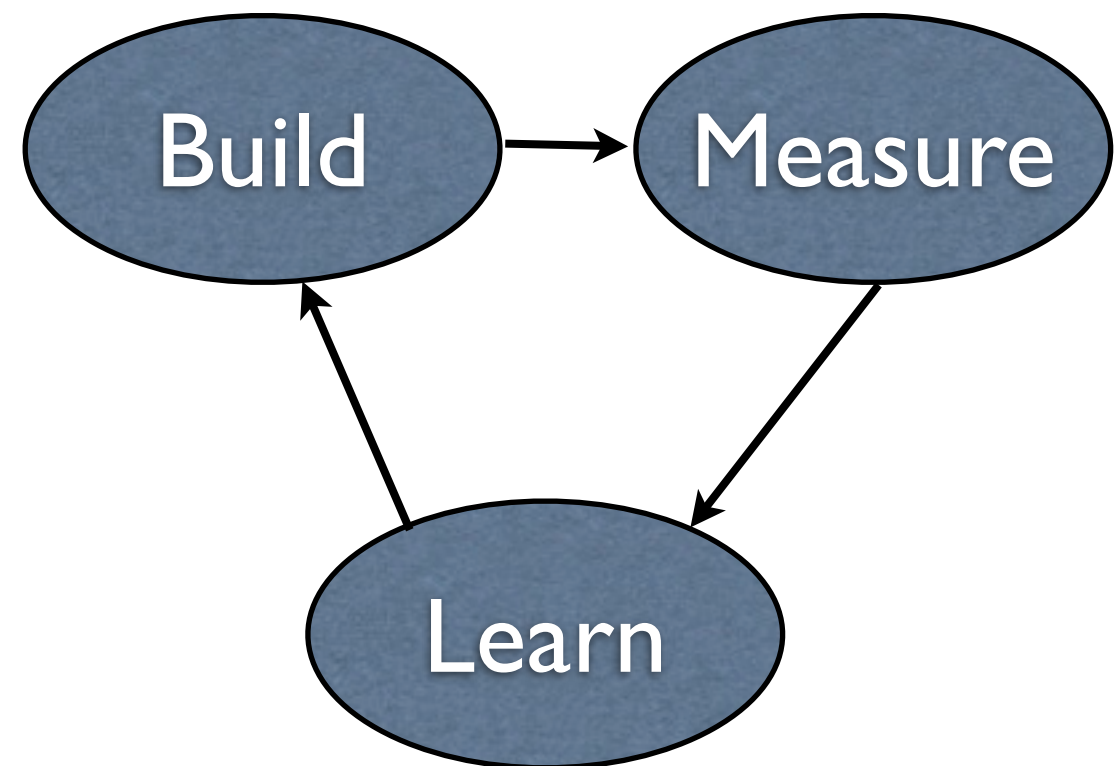
- What are attributes of change?
- Lean Startup principles
- How to apply on existing software?
- Example: Linux

What are attributes of change?

- Architecture
- Technology
- Testability
- Technical debt

Lean Startup principles

- Validated learning
- Minimum Viable Product



How to apply on existing software?

- Define your vision
- Identify the goal with the highest business value
- Create a testable hypothesis
- Validate hypothesis
- Pursue change if valid

Example - Linux

Managing contributions to the kernel

- 100+ contributors, 1000+ patches/month
- Problem: everybody pushes to Linus
--> Patches take long to merge and pile up
- replace patch system with git
- Result: a lean, pull-oriented system

Sources

- Eric Ries on Lean Startups:
<http://theleanstartup.com/principles>
<http://www.startuplessonslearned.com/2009/04/validated-learning-about-customers.html>
<http://www.startuplessonslearned.com/2010/09/good-enough-never-is-or-is-it.html>
- Linux and git:
git - a stupid content tracker, <http://download.hforge.org/doc/git-resources/200607-ols.pdf>
Linus Torvalds on Kernel SCM, <http://marc.info/?l=linux-kernel&m=111288700902396>
McDonald, Linux Kernel Development (2008)
- Raymond, E.: Cathedral & the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary. (2001)