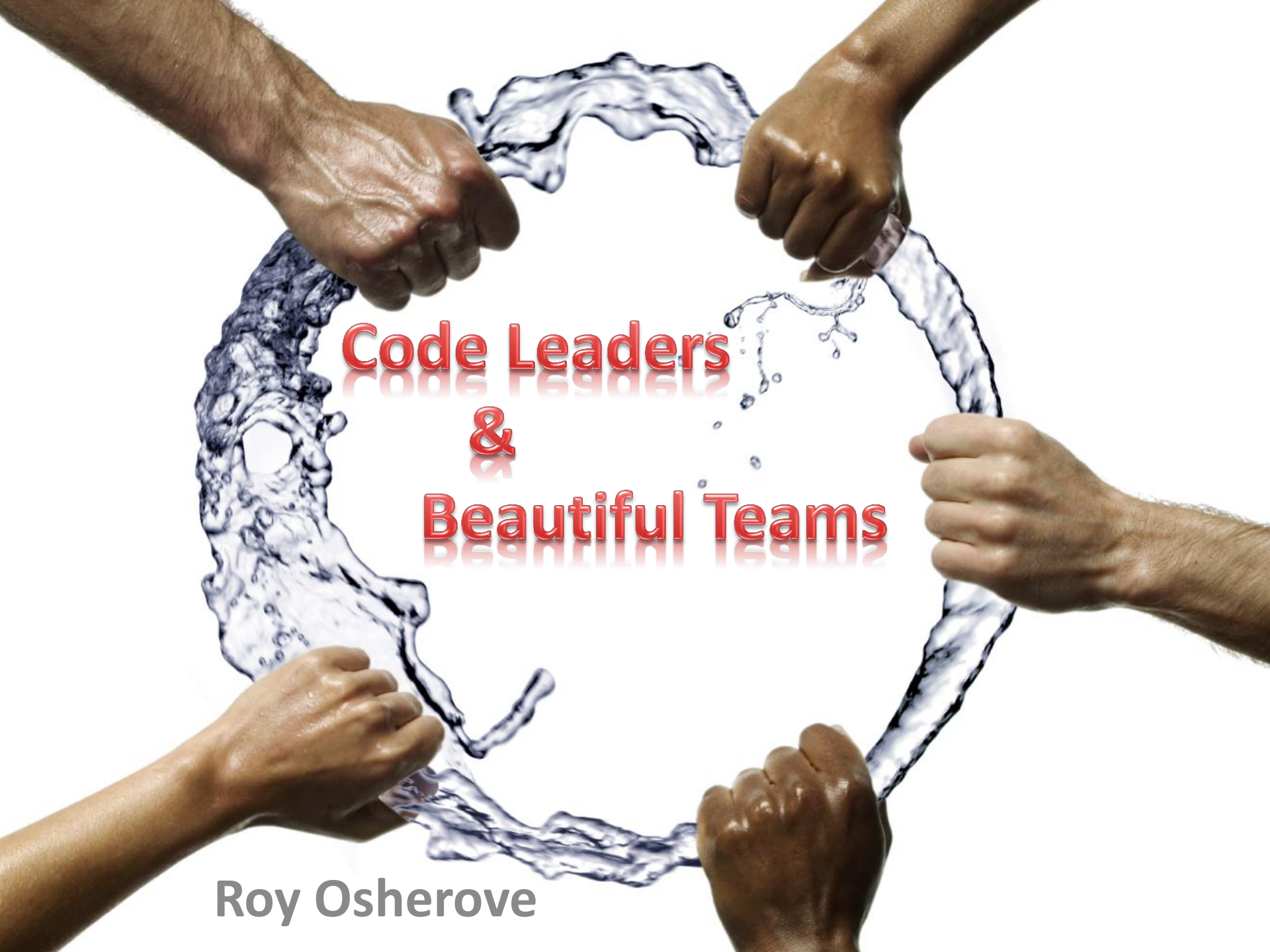
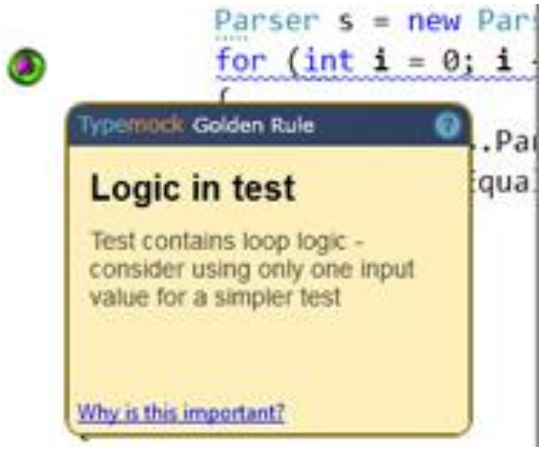

Code Leaders
&
Beautiful Teams
Roy Osherove

@RoyOsherove

5Whys.com – blog

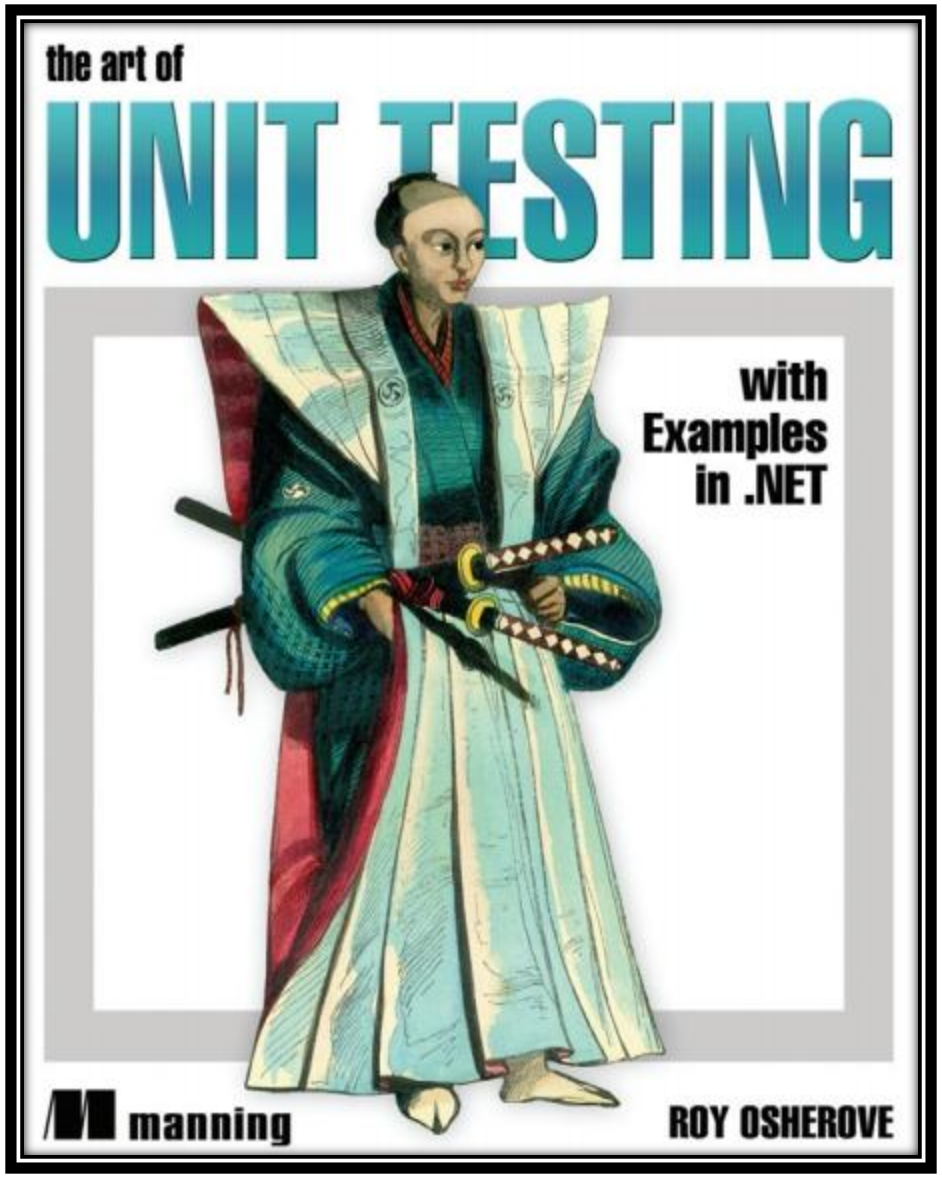
- Course: **Essential Skills for team leads** : *osherove.com*

Typemock™
Test Lint



Typemock™
Easy Unit Testing

www.ArtOfUnitTesting.com



Upcoming TDD Course @ iMeta July 19th

Agenda

- Teams
  - Buy tools, Master your tools, get rid of tools
  - Automate
  - Get Quick Feedback
  - Communicate without meetings
  - Build by feature, not layer
  - Work close
  - Code and test reviews
- Leads:
  - Be a Bottleneck Ninja
  - Grow people with Integrity
  - Remove obstacles, create a great env.

My worst team leaders

Configuration

Deployment

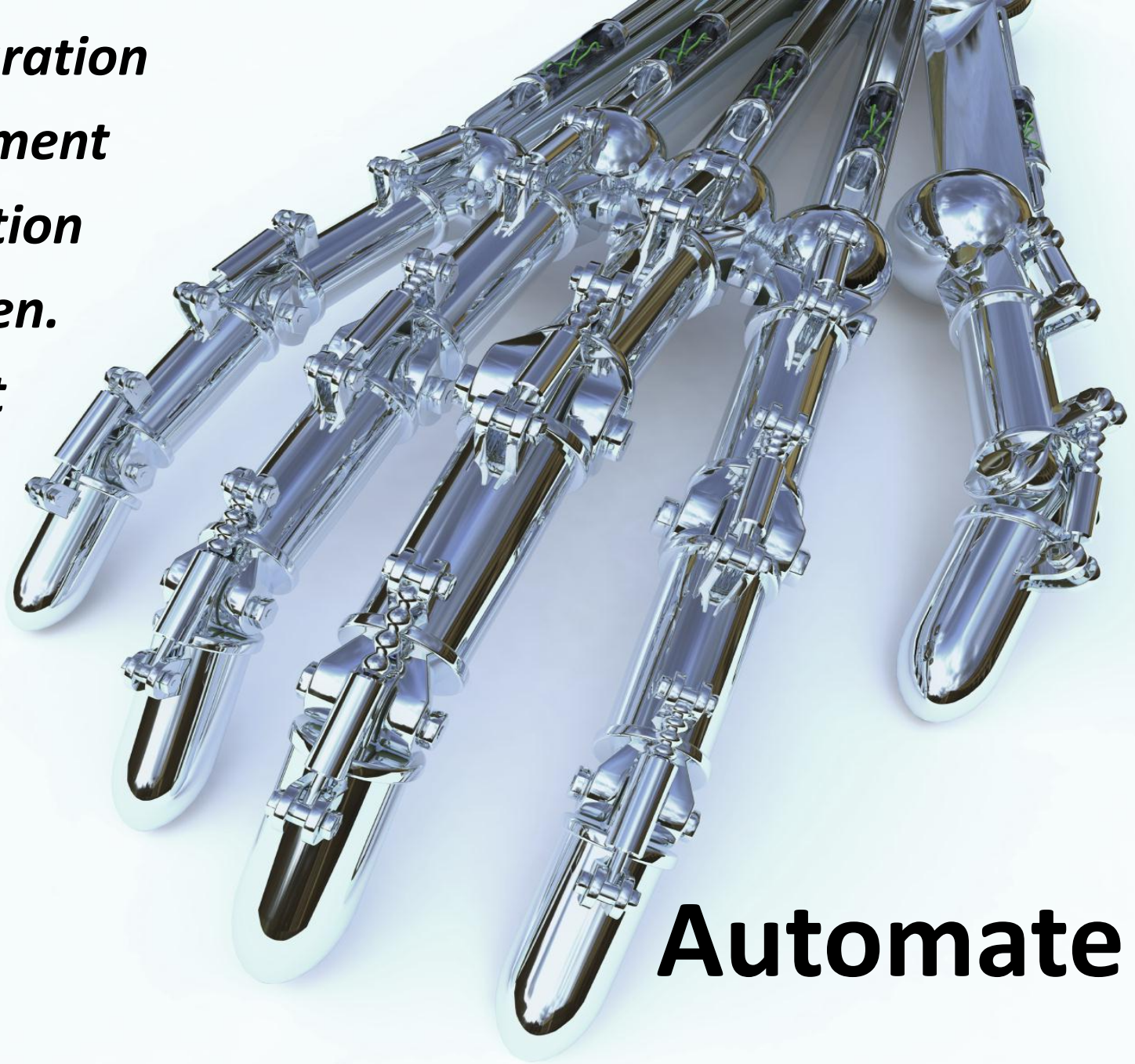
Navigation

Code Gen.

Support

Builds

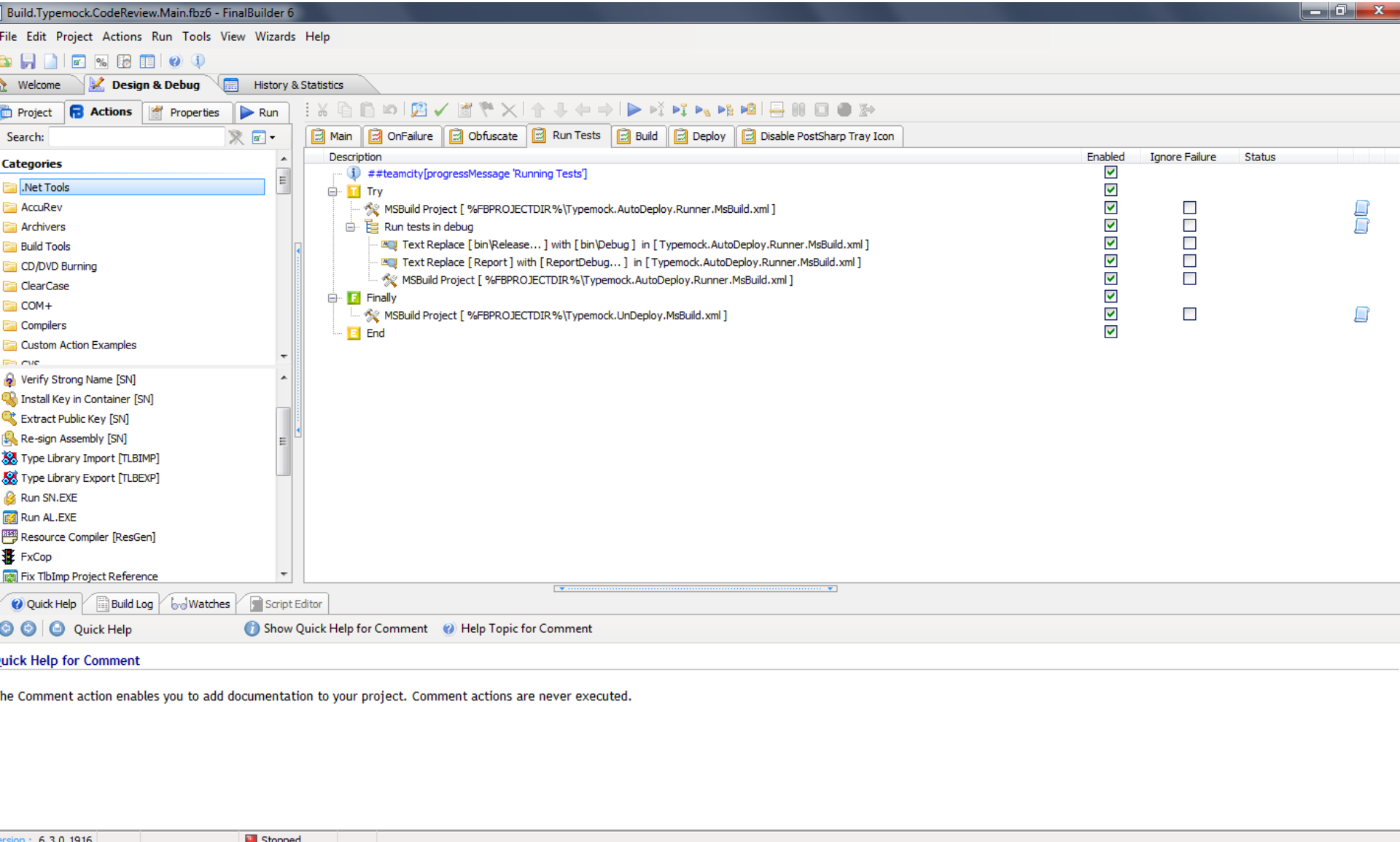
Tests



Automate

Automation tools

- FinalBuilder\ Automize
- Batch files
- Nant files
- AutoHotKey


Build.Typemock.CodeReview.Main.fbz6 - FinalBuilder 6
File Edit Project Actions Run Tools View Wizards Help
Welcome
Design & Debug
History & Statistics
Project
Actions
Properties
Run
Search:
Categories
.Net Tools
AccuRev
Archivers
Build Tools
CD/DVD Burning
ClearCase
COM+
Compilers
Custom Action Examples
Verify Strong Name [SN]
Install Key in Container [SN]
Extract Public Key [SN]
Re-sign Assembly [SN]
Type Library Import [TLBIMP]
Type Library Export [TLBEXP]
Run SN.EXE
Run AL.EXE
Resource Compiler [ResGen]
FxCop
Fix TlbImp Project Reference
Main
OnFailure
Obfuscate
Run Tests
Build
Deploy
Disable PostSharp Tray Icon
Description
Enabled
Ignore Failure
Status
##teamcity[progressMessage 'Running Tests']
Try
MSBuild Project [%FBPROJECTDIR%\Typemock.AutoDeploy.Runner.MsBuild.xml]
Run tests in debug
Text Replace [bin\Release...] with [bin\Debug] in [Typemock.AutoDeploy.Runner.MsBuild.xml]
Text Replace [Report] with [ReportDebug...] in [Typemock.AutoDeploy.Runner.MsBuild.xml]
MSBuild Project [%FBPROJECTDIR%\Typemock.AutoDeploy.Runner.MsBuild.xml]
Finally
MSBuild Project [%FBPROJECTDIR%\Typemock.UnDeploy.MsBuild.xml]
End
Quick Help
Build Log
Watches
Script Editor
Quick Help
Show Quick Help for Comment
Help Topic for Comment
uick Help for Comment
he Comment action enables you to add documentation to your project. Comment actions are never executed.

Controls
Automation tools
Editors
Diff helpers
Bug mgmt.
Source Control

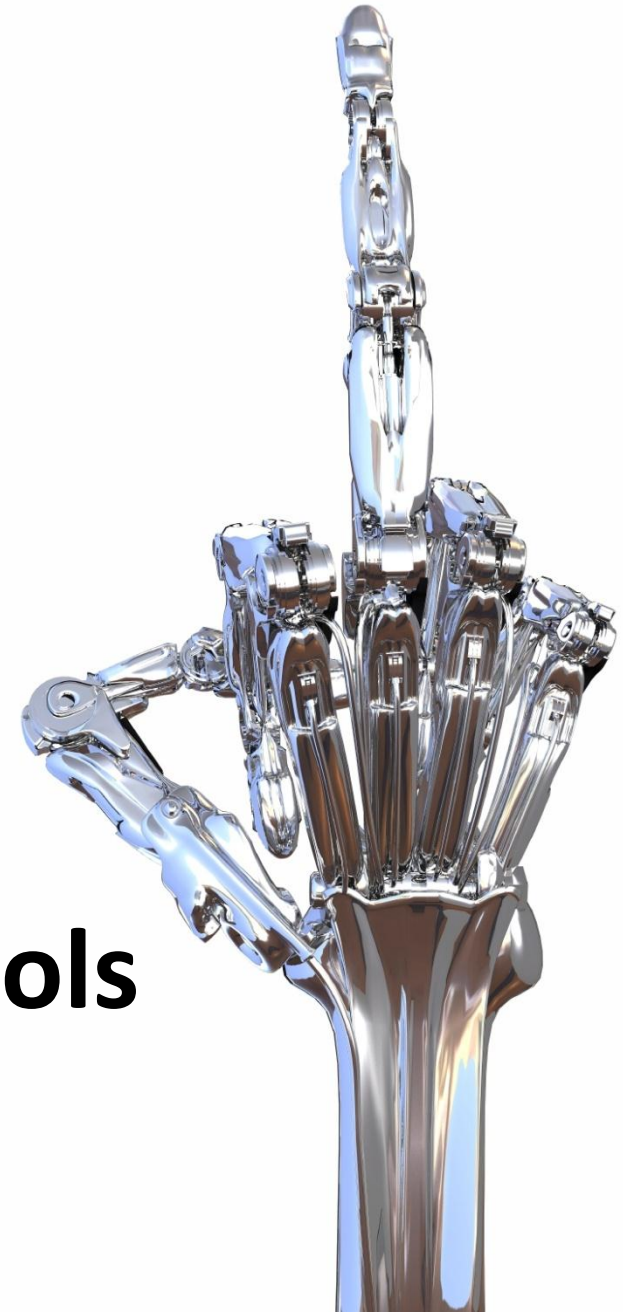
Test & Buy the right tools

Beyond Compare and lack of testing

Throw out
the wrong tools

- IDE Shortcuts
- Refactoring tools
- Outlook..?
- Macros

Master your tools



Get Quick, Continuous Feedback

- Your code works
- From customers
- Feedback from peers

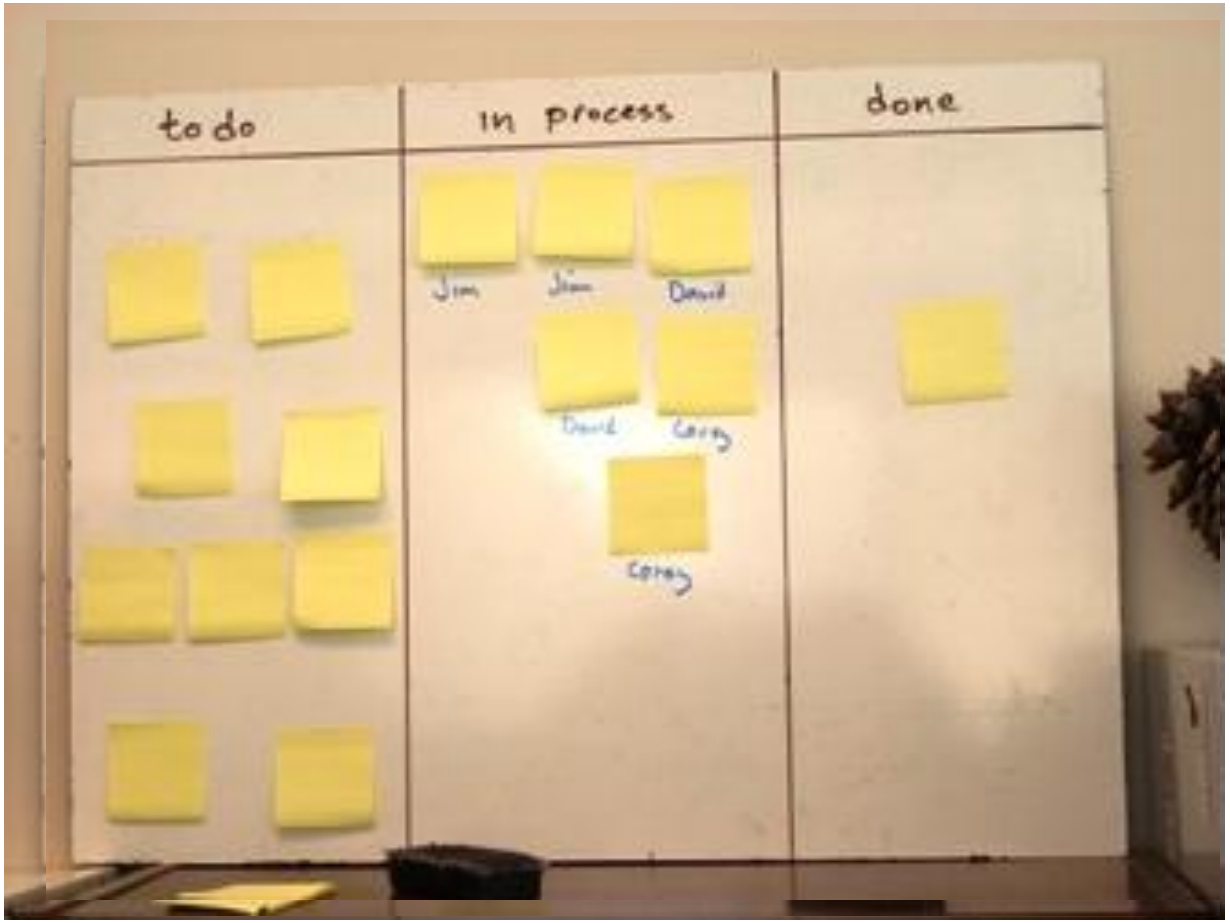
Does my code work?

- Automated tests
- Automated builds running tests
- Automated scheduled builds
- Continuous Feedback via
  - Continuous Integration
  - TeamCity
  - CruiseControl
  - Cruise
  - Hudson

Did I build the right thing?

- Customer feedback
- Every iteration
  - (1-2 weeks?)

Showing Big Visible Progress



http://leansoftwareengineering.com/ksse/scrum-ban/

Scrumban

Essays on Kanban Systems for Lean Software Development

Corey Ladas

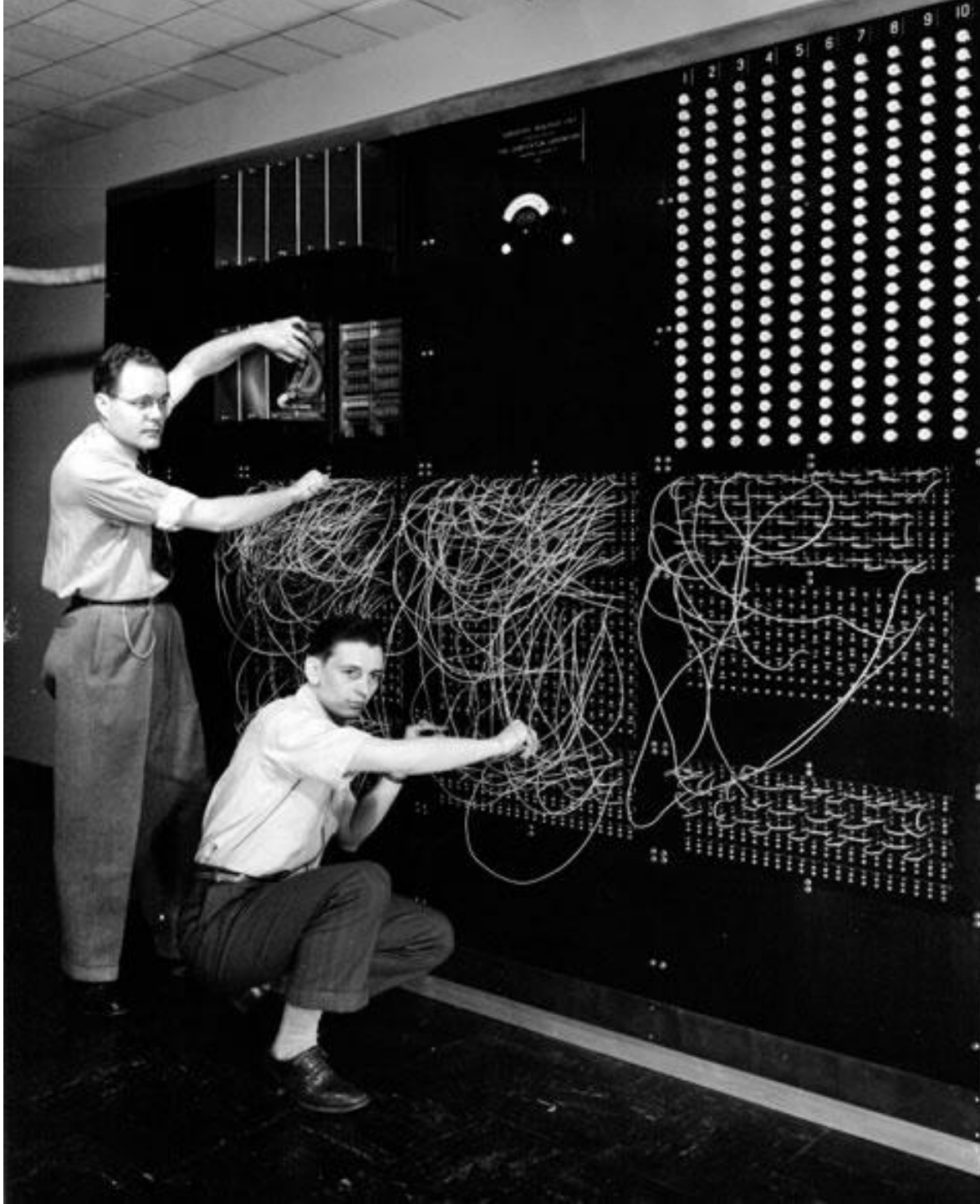
Modus Cooperandi Lean Series

Communicate **without** “meetings”

No Meetings?

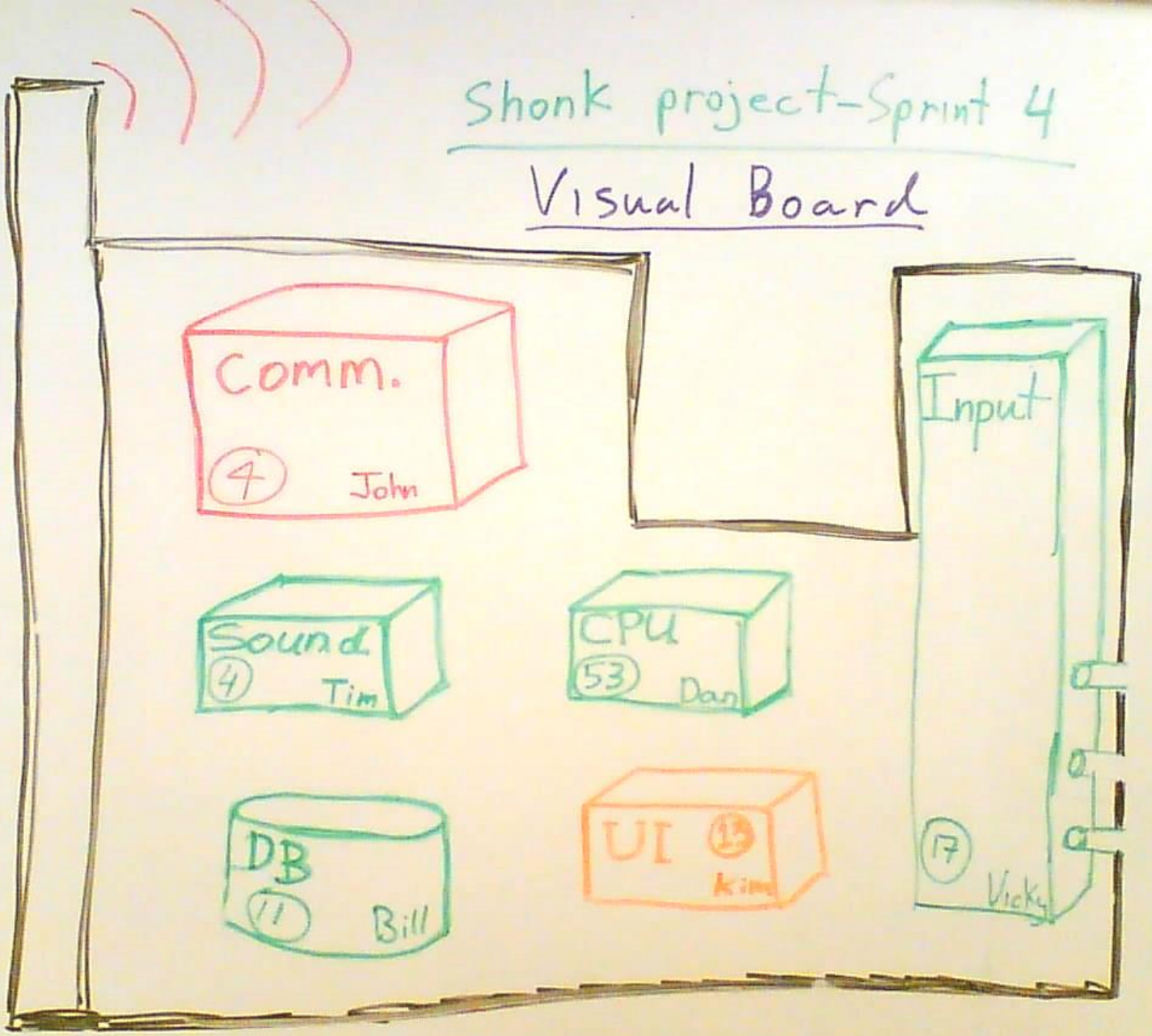
- Daily standup
  - What have you done yesterday?
  - What will you do today?
  - What's holding you back?
- Pair Programming
- Peer code review
- Same room
- Big Visual Progress\Notifications

Work in Pairs



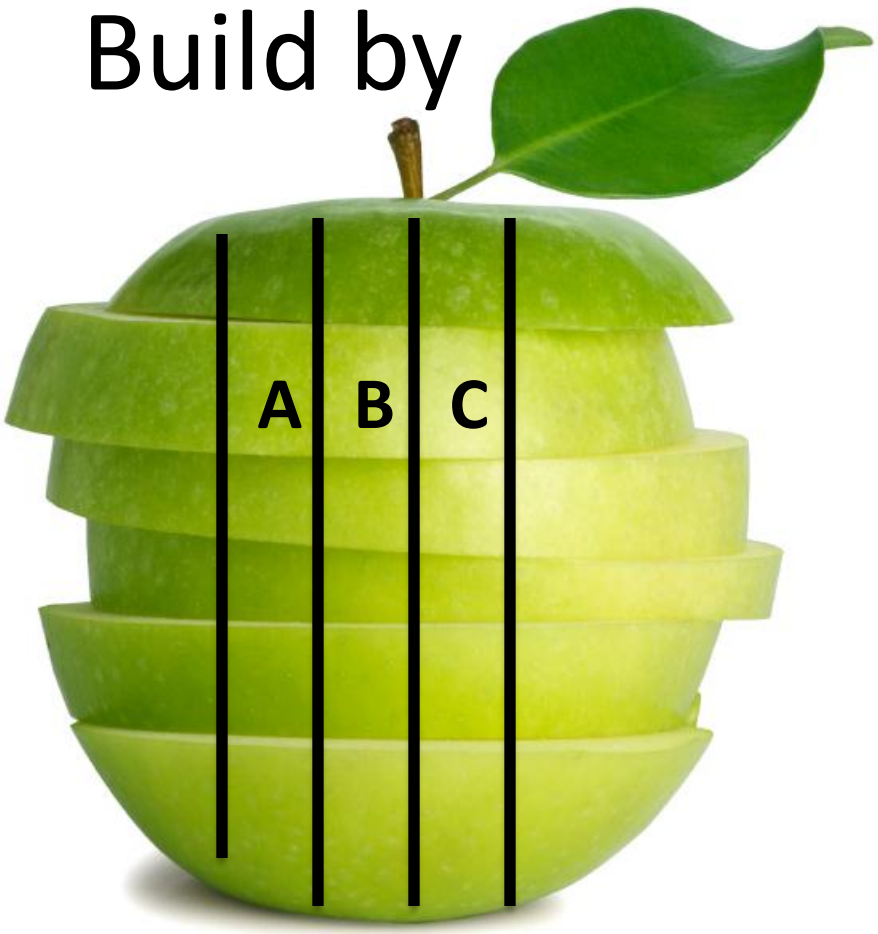
Shonk project-Sprint 4

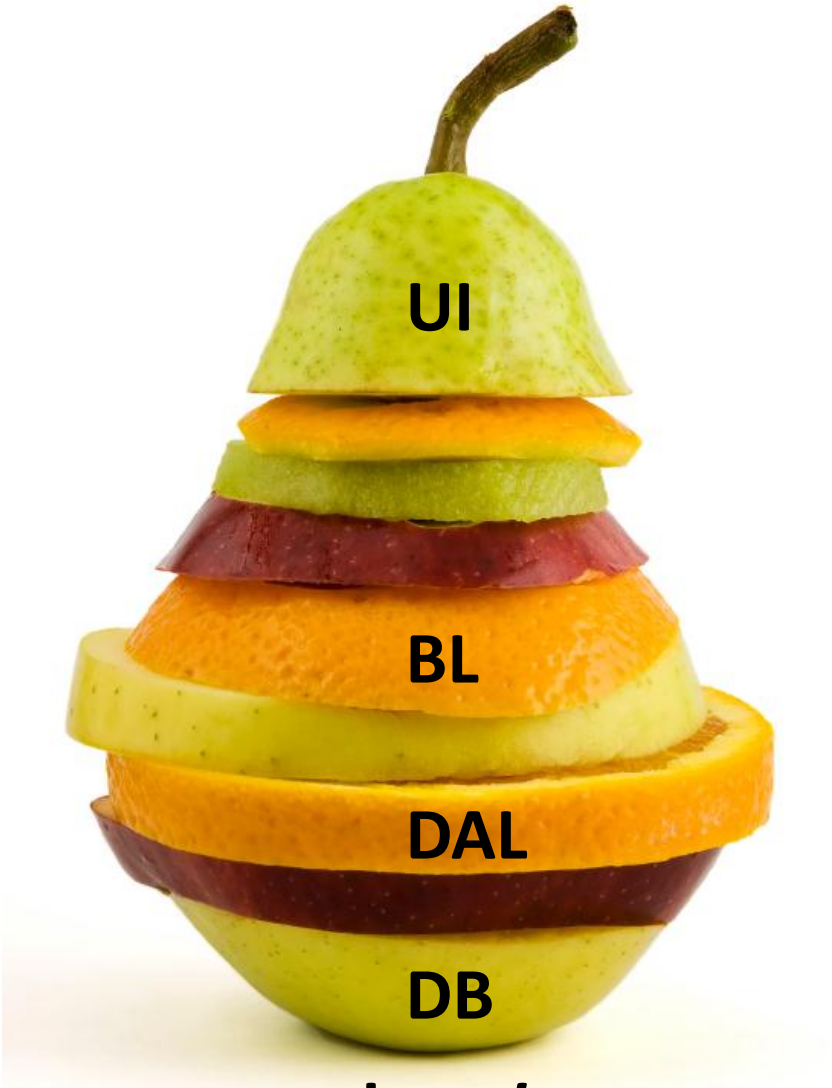
Visual Board



Code Review,
Test Review

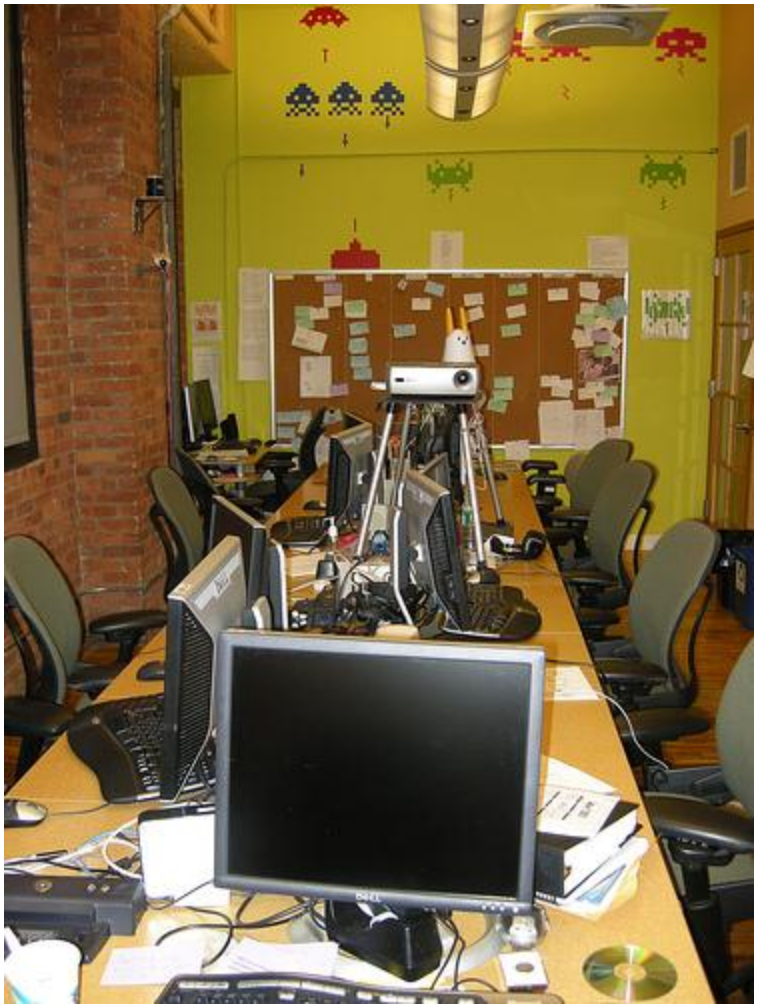
Build by ***feature***





Not by *layer*

Small team. same room.
BIG Feedback



Leading
people

My worst team leaders

Unable to take design decisions
Getting stuck on a problem
“swallowed” by a task
“pinged” from outside the team

Be a Bottleneck Ninja

Grow
people
with
integrity

"Where would you like to get better?"

"Here's where I <u>know</u> you can be 10X better than today.."

"are you <u>fully committed</u> to doing this?"

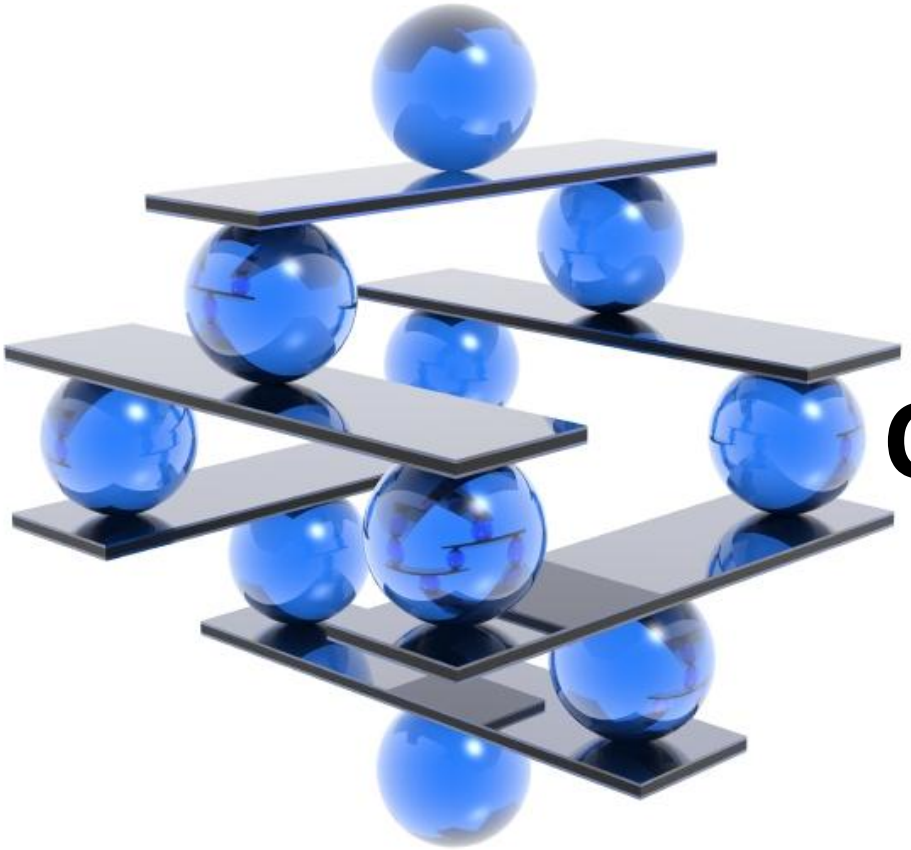
Remove

Obstacles

Das Keyboard
1
A

Growing – Asking for solutions

- Teach how to find solutions
- YOU are a bottleneck



Connecting it all

Daily\weekly questions

- **Lead:**
- daily: what bottlenecks exist in the team? what have I solved?
- will my devs be better in a month or two than they were before?
- what prevents my devs from working? what am I doing about this?
- **Team:**
- What can we **automate**?
- where are we **"Reinventing the wheel"**?
- what are the **tools that slow us down**?
- what **tools** can we use **better**?
- are there **bugs** that I could have found **earlier**? how do I make sure I find them earlier?
- **when** do we find out we built **the right thing**?
- **when** do we find out our **code\design sucks**? how can we make that earlier?
- How do we show **progress at the team level**? at **the management level**?
- How many **meetings** does each dev have every week? how can we remove them?
- Are we building **by feature or by layer**?
- can we make all our team **sit in the same place**?

If you'd like to know more

- Developer anti patterns
- Team lead anti patterns
- Influencing others and management
- More tips and practices

- **5Whys.com**
- **Course:** Essential skills for team leads

  http://***Osherove.com***

Resources

- **5Whys.com** – My Team Lead Blog
- **Osherove.com : TDD and Team Lead Course**
- Typemock™ .com for **test lint**