



I'll go up and find out what they need and the rest of you start coding!

*Agile – Adoption*

# AGILE MYTHS

Colm O'hEocha – AgileInnovation  
[colm@agileinnovation.eu](mailto:colm@agileinnovation.eu)  
[www.agileireland.org](http://www.agileireland.org)

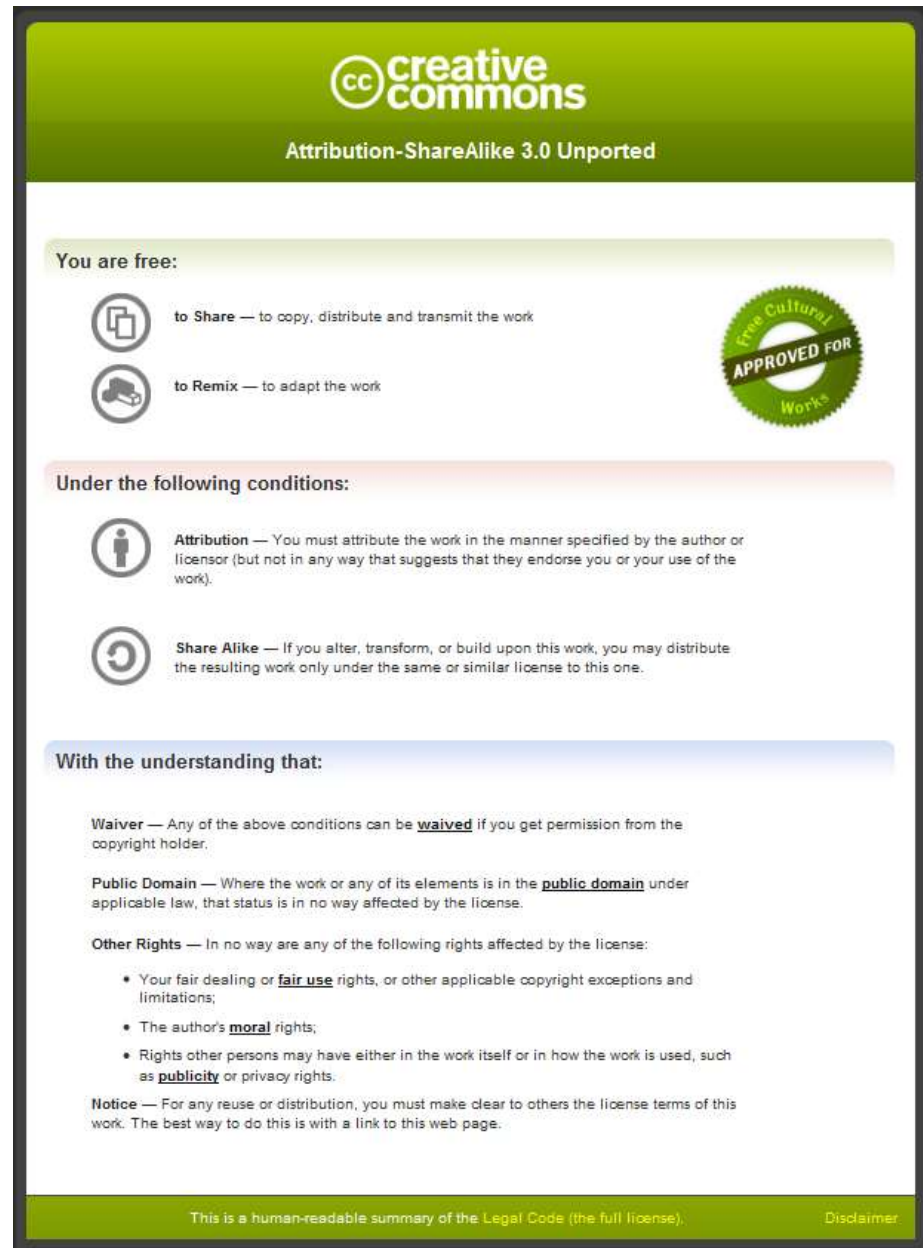
This material is Copyright Protected under the following Creative Commons License:

## Attribution-ShareAlike 3.0

Please use this material freely and derive further work from it. BUT, attribution must be given to the original author, and these same rights must govern all derived works.

Please use the following attribution:



“Colm O’hEocha,  
AgileInnovation Ltd. 2010:  
www.agileinnovation.eu”




The image shows a Creative Commons Attribution-ShareAlike 3.0 Unported license card. It features the Creative Commons logo and the text "Attribution-ShareAlike 3.0 Unported". The card is divided into several sections: "You are free:" which lists "to Share" (copy, distribute, and transmit) and "to Remix" (adapt); "Under the following conditions:" which lists "Attribution" (must attribute the author) and "Share Alike" (must use the same license); "With the understanding that:" which includes "Waiver" (conditions can be waived), "Public Domain" (status is unaffected), and "Other Rights" (fair dealing, moral rights, publicity/privacy rights); and "Notice" (must make clear the license terms). A circular seal on the right says "Free Cultural Works APPROVED FOR". The footer contains a link to the full license and a disclaimer.



**CC creative commons**  
Attribution-ShareAlike 3.0 Unported

**You are free:**

-  **to Share** — to copy, distribute and transmit the work
-  **to Remix** — to adapt the work



**Under the following conditions:**

-  **Attribution** — You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).
-  **Share Alike** — If you alter, transform, or build upon this work, you may distribute the resulting work only under the same or similar license to this one.

**With the understanding that:**

**Waiver** — Any of the above conditions can be waived if you get permission from the copyright holder.

**Public Domain** — Where the work or any of its elements is in the public domain under applicable law, that status is in no way affected by the license.

**Other Rights** — In no way are any of the following rights affected by the license:

- Your fair dealing or fair use rights, or other applicable copyright exceptions and limitations;
- The author's moral rights;
- Rights other persons may have either in the work itself or in how the work is used, such as publicity or privacy rights.

**Notice** — For any reuse or distribution, you must make clear to others the license terms of this work. The best way to do this is with a link to this web page.

This is a human-readable summary of the Legal Code (the full license). [Disclaimer](#)

Like any new idea,  
misinformation  
abounds...





# Agile is eXtreme...

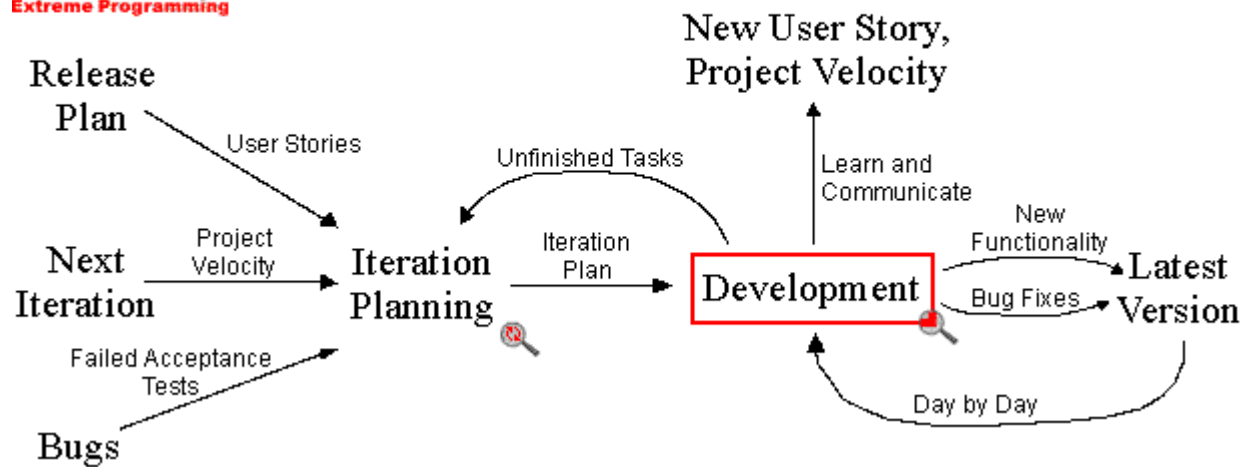
# No process – Ad-hoc Development



Of all software projects nominally using a process, only 6% adhere to it.  
(Fitzgerald et al. 1998)



## Iteration



# Cowboy Programming

The Myth: Agile teams develop code all in a rush, with no time or the discipline to carefully design and implement code



- Shared Code Base
- Code Standards
- Continuous Integration
- Test Driven Development
- Refactoring to Simplify
- Peer review
- Pair programming

# Agile Doesn't Do Documentation

Bring Idea & Implementation  
Closer Together

Late Elaboration

Stories  
vs.  
Specification



Document the most important,  
Stable stuff  
- Discuss the shifting detail

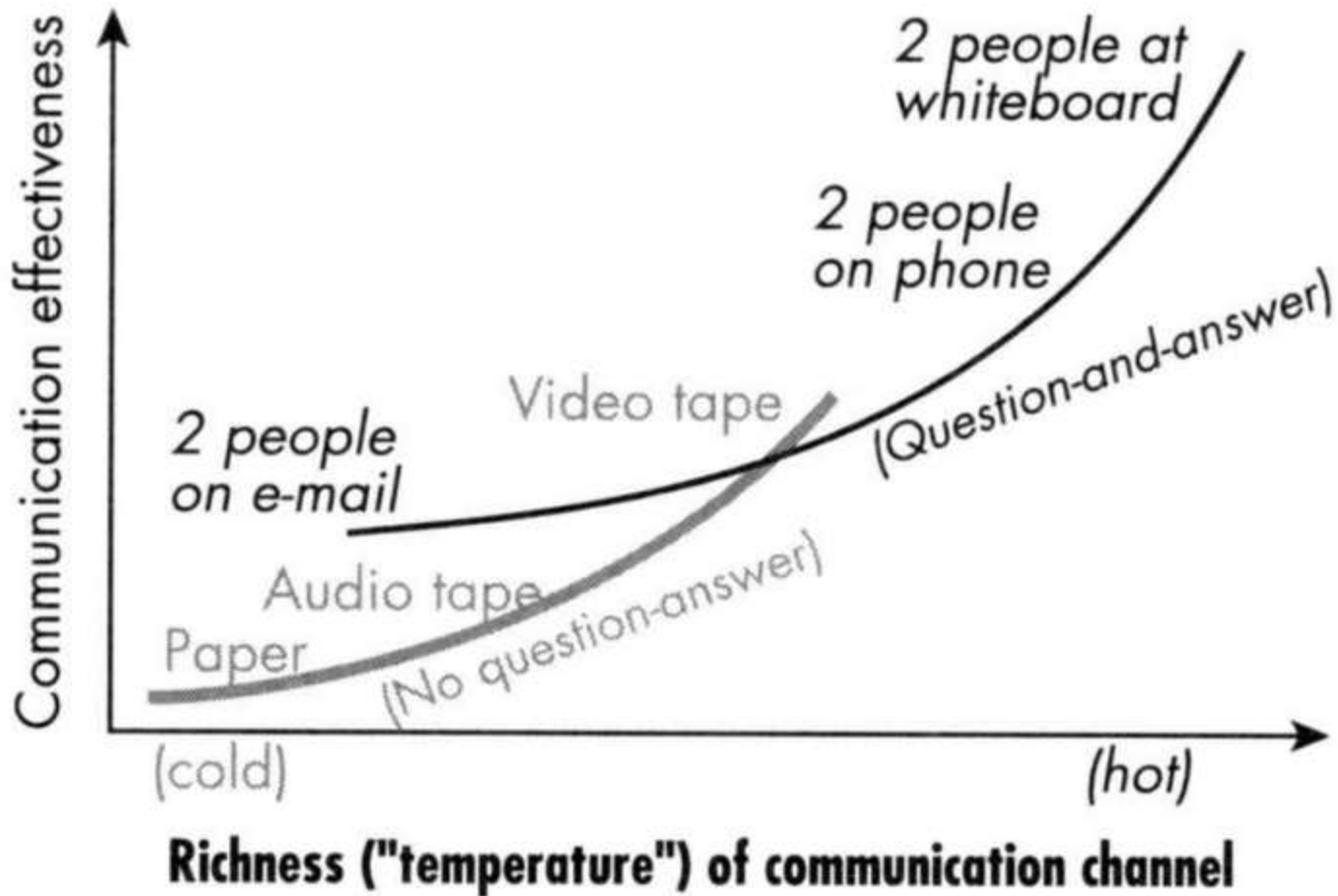
Face To Face

Shared Mental  
Model

Document the Product,  
Not the Project



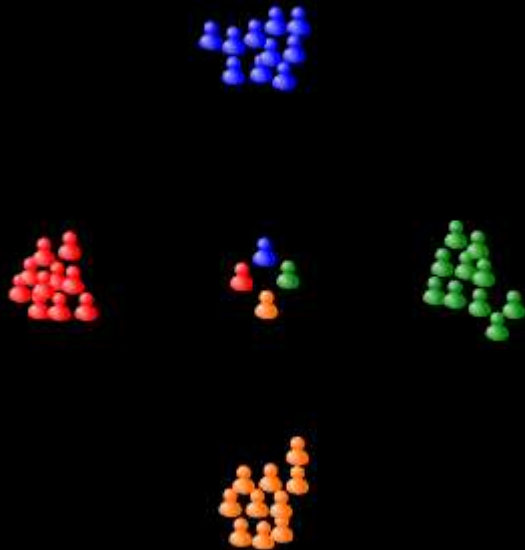
# Face to Face Communication



Source: Alistair Cockburn



# Small, Experienced, Co-Located Teams



# No Control - No Discipline

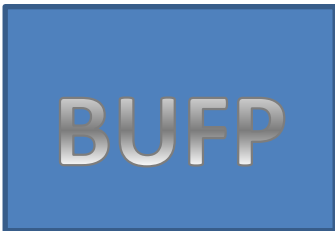
## Discipline is NOT:

- paper-pushing
- ticking boxes
- compliance
- Bureaucracy
- Stifling innovation
- Hiding in the process



- Discipline is:  
Always working on the most important thing
- Always refactoring, reflecting, inspecting, adapting, (re)planning, collaborating, delivering

# Just Do It!



Continuous (Re)-Planning



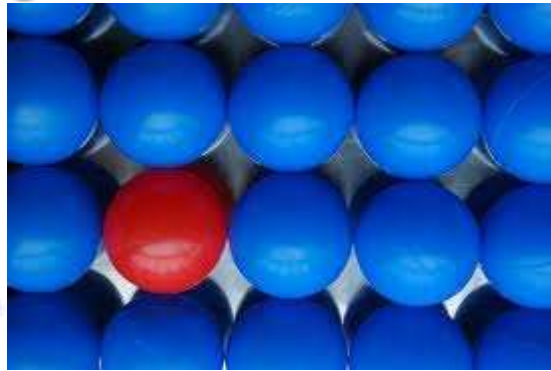
*Planning is everything, the plan is nothing*

# Won't work here...

We have no  
End User

Our teams are  
too big

Our features take  
months to develop



We need to walk  
before we can run

We're just part of a  
bigger waterfall project

Agile doesn't work for  
batch/real-time/embedded/etc

We're too busy!



# Not for Mission/Life Critical Systems

No place for  
defects to hide

Compress  
Build-Test Cycles

Build Quality In

**MISSION CRITICAL**

Test ALL the Time  
(not just before release)

Build it, Test it, Use It  
From the beginning

Quality  
Non-Negotiable

# Architecture is Compromised



*Refactoring is for implementing new learning!*

# When done correctly...

- Agile plans all the time
- Agile tests all the time
- Agile makes it more visible
- Agile leads to stronger commitment
- Agile relies on discipline
- Agile uses the most efficient communication tool
- Agile nurtures learning & innovation
- Agile invests in the total product lifecycle
- Agile keeps it simple - Minimum critical specification
  - *Not too much - just enough*



Colm O'hEocha – AgileInnovation

[colm@agileinnovation.eu](mailto:colm@agileinnovation.eu)

[www.agileireland.org](http://www.agileireland.org)



# About the Presenter

Colm O'hEocha,  
AgileInnovation Ltd.  
Colm@agileinnovation.eu



Colm is an IT industry veteran, with over 20 years design and development experience. His interest in Lean Thinking began when automating Just-In-Time production lines in the 1980s. Implementing early agile software methods in 2001, Colm has extensive hands-on experiences across teams, organisations and technologies. In varied roles of developer, architect and Director of R&D, he has seen Lean and Agile from a variety of perspectives. More recently Colm has partnered with LERO, the Irish software engineering research institute, investigating how agile and lean methods influence innovation in software development.

AgileInnovation offers agile advisory and coaching services. Pre and post adoption assessments, coaching services to help teams get the most from agile, developing a agile business case, and targeted workshops on specific agile practices. Customers include Intel, HMM, Omnipay, Trigraph and Sogeti



# Agile Assessments, Training and Coaching



## Training Includes:

- *Executive Briefing on Agile and Lean (½ day)*
- *Introduction to Agile Methods (½ day)*
- *Succeeding with Agile (2 days)*
- *ScrumMaster Training (2 days)*
- *Product Owner Training (1 day)*

## Coaching Includes:

- *Structuring and facilitating*
  - *Daily StandUps*
  - *Sprint Planning*
  - *Sprint Reviews*
- *Developing User Stories*
- *Estimation*
- *Release Planning*

## Consulting Includes:

- *Project and Organisational Assessments*
- *Developing an Agile Adoption Strategy*
- *Getting the best from Agile and Waterfall*
- *Agile in Complex Contexts (firmware, B2B, batch, etc)*





# AgileInnovation: Embracing the Inevitability of Change