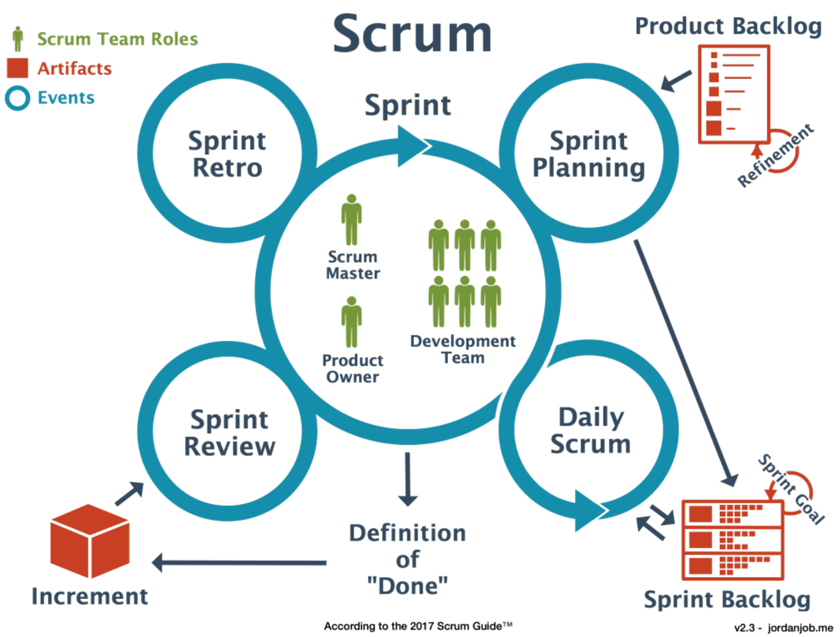
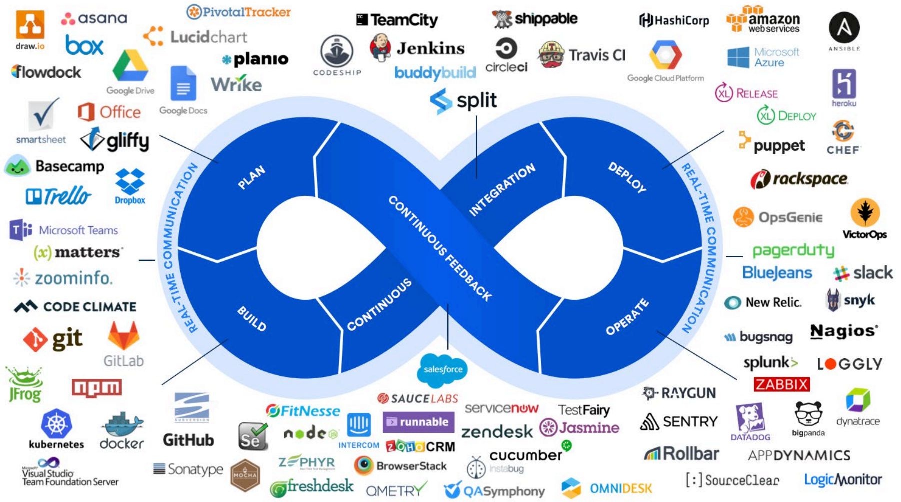
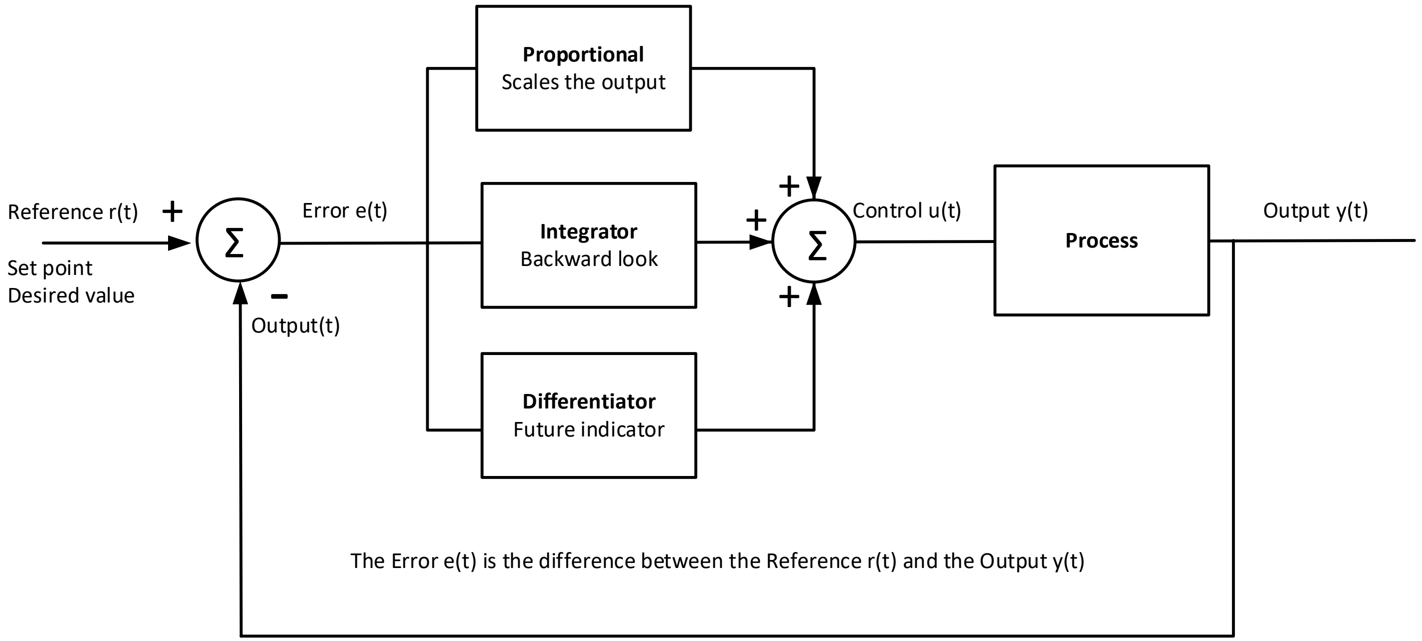
# How is your Agile Today?

## an
## Electrical Engineer’s Perspective







Reference r(t)

Set point
Desired value

Error e(t)

Output(t)

**Proportional**
Scales the output

**Integrator**
Backward look

**Differentiator**
Future indicator

Control u(t)

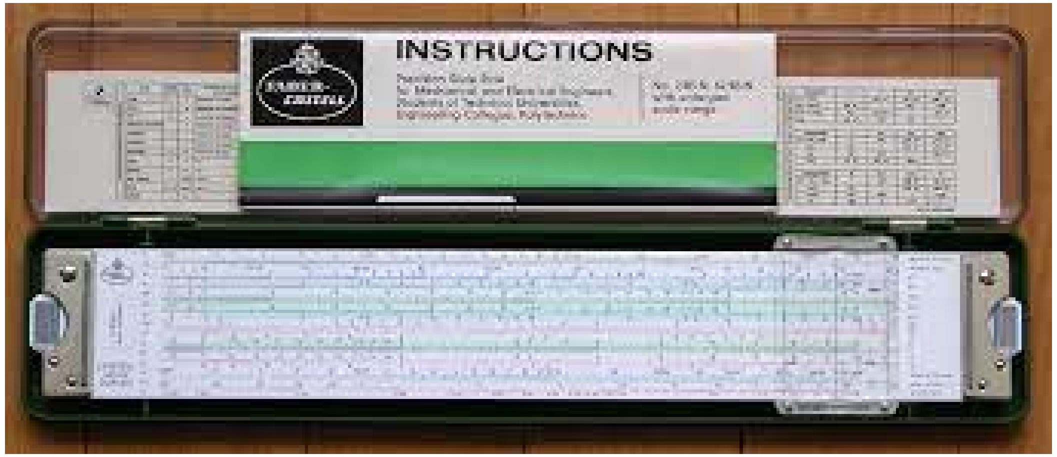
**Process**

Output y(t)

The Error e(t) is the difference between the Reference r(t) and the Output y(t)

Adriaan Vorster 31 March 2022

# My Background



http://www.stefanv.com/calculators/fc283n.html

I am an old fashioned Electric Systems Engineer, still grieving the fact that, in 1973, the Hewlett Packard HP35 replaced my trusty Faber Castell Novo Duplex slide rule.
Whereas the slide rule, with the astonishing 530mm W scales and the ability to represent four significant digits, required thinking about what you were doing, the HP35 allowed even incompetent first year students to confidently generate wrong answers with 12 digit precision. Fool + Tool ≠ Genius!

It has been amazing to see how the rapid adoption of technology, as solutions, has followed the same trajectory of the HP35 calculator and it's counterparts, allowing people to "do without thinking".

The outbreak of World Peace in 1989 saw me, reasonably good at complex systems analysis,  engaged in a research environment and soon to be unemployed.

Fortunately I found that the ICT domain presented opportunities for systems analysis, synthesis and realisation and I started writing software.

This led to assignments as CIO and a career in Enterprise Architecture.

Not being much of a conformist, I quite enjoy making holy cow hamburgers.

https://en.wikipedia.org/wiki/HP-35

# I am not an Agile Agnostic

- Having been around software development since the days of Royce and the Waterfall methodology when klocs ruled the roost and having experienced the utility of the V model, I was very interested in the rise of Extreme Programming and the resulting Agile movement.
- I have spent significant time studying and consulting on the Agile Software Development approach and the Scrum methodology in particular.
- To my mind these approaches potentially have two meritorious aspects, very short process cycles, continual stakeholder involvement.
- But, I am also aware that when you have a hammer, everything looks like a nail!

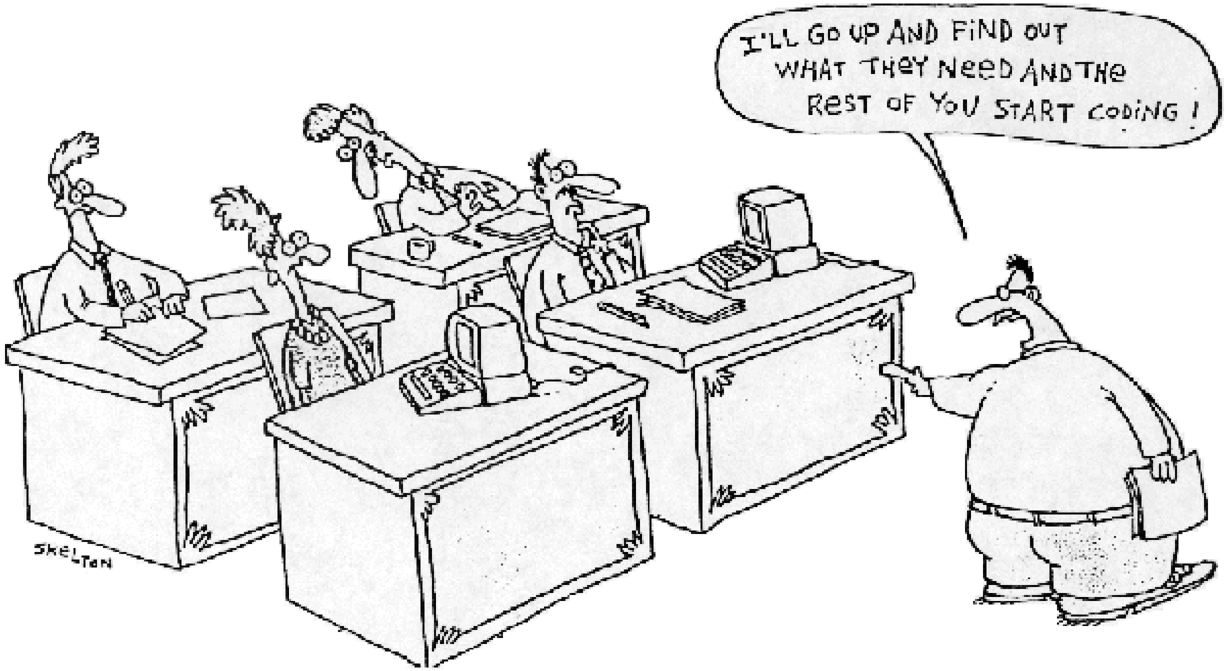
SCRUM
The Art of Doing Twice the Work in Half the Time
JEFF SUTHERLAND
Co-creator of Scrum
The Addison-Wesley Signature Series
ESSENTIAL SCRUM
A Practical Guide to the Most Popular Agile Process
Kenneth S. Rubin
Forewords by Mike Cohn and Ron Jeffries
Robert C. Martin Series
Agile Estimating and Planning
Mike Cohn
The SCRUM Fieldbook
Faster performance. Better results. Starting now.
J.J. Sutherland
The Addison-Wesley Signature Series
THE SCRUM FIELD GUIDE
Agile Advice for Your First Year and Beyond
Second Edition
Mitch Lacey
Forewords by Jeff Sutherland and Kenneth S. Rubin
ZEN & THE ART OF MOTORCYCLE MAINTENANCE
ROBERT M. PIRSIG

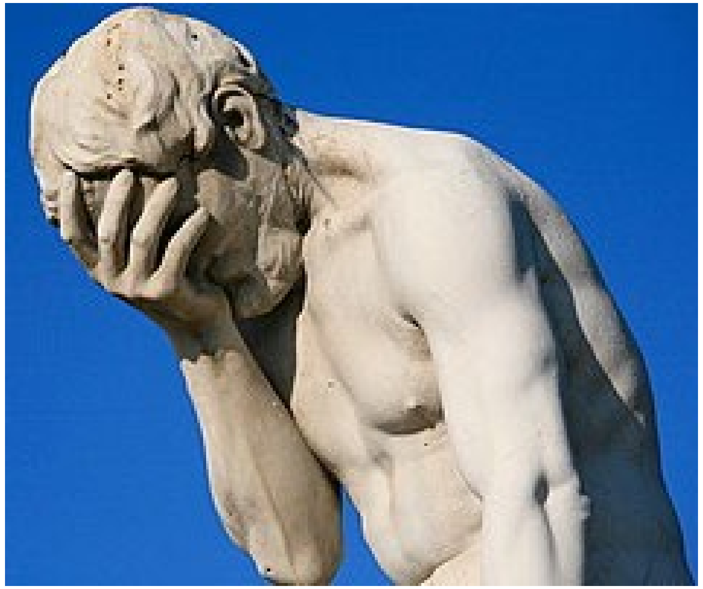
Zen and the Art of motorcycle maintenance has nothing to do with Scrum

# Software Development Used to be so Easy!

Software developer

User


I'LL GO UP AND FIND OUT WHAT THEY NEED AND THE REST OF YOU START CODING !
SKELTON



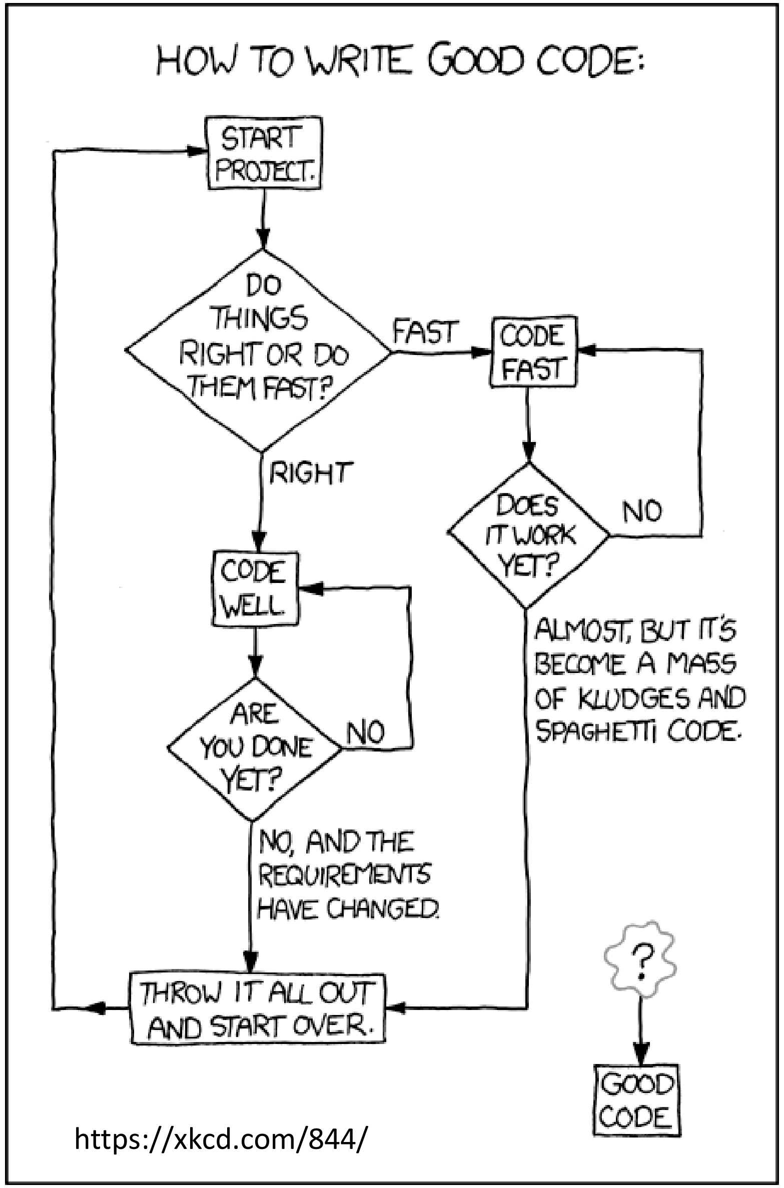
It's just what I asked for,
but NOT what I WANT!

https://bjohnson.lmu.build/cmsi185web/week02.html

Cain by Henri Vidal, Tuileries Garden. Paris, 1896.
https://en.wikipedia.org/wiki/Facepalm#/media/File:Paris_Tuileries_Garden_Facepalm_statue.jpg

# The Law of the Conservation of Misery.

- The accelerating rate of change has forced us to continually evolve our capacity and capability to rapidly deliver good-enough software.
- This has led to a host of new approaches in order to thwart what seems to be a basic, but largely unknown Law of the Universe; the Law of the Conservation of Misery.
- This law, well known in the Energy Laboratory of the Rand Afrikaans University during the latter years of the previous century, states that Misery is a strictly rising tendency that only in the Limit will remain Constant.
- The fact that Microservices often lands you in Merge Hell is an example of this universal law.


HOW TO WRITE GOOD CODE:
START PROJECT.
DO THINGS RIGHT OR DO THEM FAST?
FAST
CODE FAST
RIGHT
DOES IT WORK YET?
NO
CODE WELL.
ARE YOU DONE YET?
NO
ALMOST, BUT IT'S BECOME A MASS OF KLUDGES AND SPAGHETTI CODE.
NO, AND THE REQUIREMENTS HAVE CHANGED.
THROW IT ALL OUT AND START OVER.
?
GOOD CODE

https://xkcd.com/844/

# I Like the Manifesto for Agile Software Development

- As a dyslexic kid who grew up rebelling against the tyranny of spelling rules whilst not being able to get writing to fit between the lines, I resonated with the Founding Fathers of the Manifesto for Agile Software Development who described themselves as  “a gathering of organizational anarchists”.
- To my mind, the image used on the web site imbibed the Manifesto with this mystical feeling of the persecuted, gathering to launch a revolution and free us all from the Dibertesque Corporate Culture .
- All very good stuff and the simple rules of the Manifesto for Agile Software Development harkened back to my childhood where good clean living was basically condensed to: Love thy neighbour like thyself.
- How could I go wrong?

https://agilemanifesto.org/

Manifesto for Agile Software Development

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.

| | | |
|---|---|---|
| Kent Beck | James Grenning | Robert C. Martin |
| Mike Beedle | Jim Highsmith | Steve Mellor |
| Arie van Bennekum | Andrew Hunt | Ken Schwaber |
| Alistair Cockburn | Ron Jeffries | Jeff Sutherland |
| Ward Cunningham | Jon Kern | Dave Thomas |
| Martin Fowler | Brian Marick | |

© 2001, the above authors
this declaration may be freely copied in any form,
but only in its entirety through this notice.

Twelve Principles of Agile Software

# Values of the Manifesto for Agile Software Development

Having been deployed in a combat situation as a member of the Technical Support team of an Airborne Unit; a meritocracy of different competent elements that were able to solve problems on the fly, without the structures and strictures of Military Hierarchies, the Values of the Manifesto resonated with me.

- **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**.
- Could the Manifesto for Agile Software Development present the new Silver Bullet?

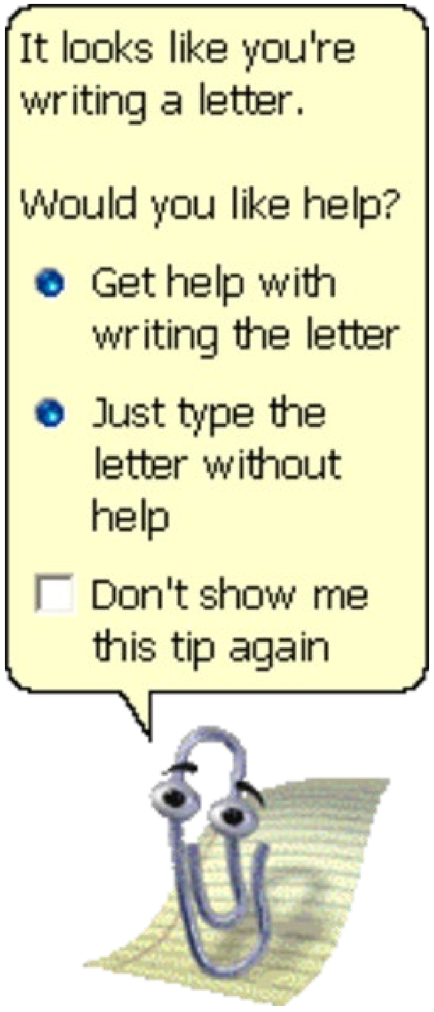
# No Silver Bullet? The Essence of Software Engineering

- Unfortunately, I had to get past Frederick Brooks and his contention that there are no silver bullets due to the Essence of Software Engineering which were defined as:
- **Complexity** The complexity of software is an essential property, not an accidental one. Hence, descriptions of a software entity that abstract away its complexity often abstract away its essence.
- **Conformity.** Much of the complexity that the software engineer must master is arbitrary complexity, forced, without rhyme or reason, by the many human institutions and systems to which interfaces must conform.
- **Changeability**. The software entity is constantly subject to pressures for change.
- **Invisibility.** Software is invisible and unvisualizable.

Brooks, Frederick P., "No Silver Bullet: Essence and Accidents of Software Engineering,“ *Computer*, Vol. 20, No. 4 (April 1987) pp. 10-19.

# Changeability, Conformity and Software Evolution

- Changeability and conformity go hand-in-hand.
- Not only are software systems under constant pressure to change due to users requiring continual enhancements, the environment within which software systems are deployed are also subject to continual change.
- Lehman, a colleague of Brook during the development of the IBM System 360 developed some Laws of Software Evolution, with Evolution defined as the “cumulative inherited change in a population of organisms through time leading to the appearance of new forms and the death of the useless”.
- The essence of Lehman’s Laws is that all software that is used, will get changed. This will add complexity, leading to decreased maintainability and the end of the useful life of the software.


It looks like you're writing a letter.
Would you like help?
● Get help with writing the letter
● Just type the letter without help
□ Don't show me this tip again

https://www.nicepng.com/maxp/u2q8w7y3r5w7t4a9/

# Complexity and Invisibility - the Zachman Framework

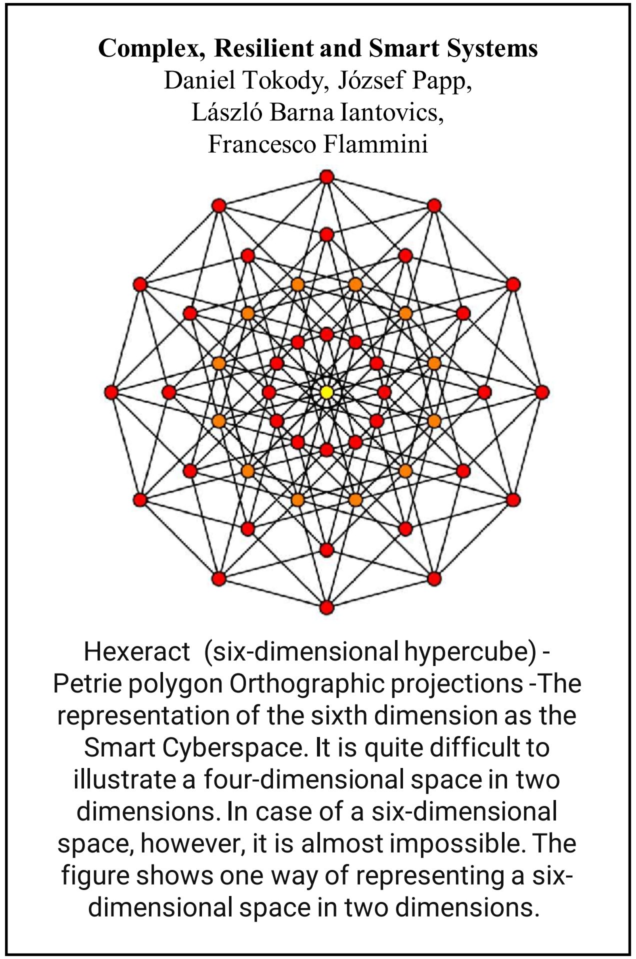
At virtually the same time that Brooks published No Silver Bullet, John Zachman published his paper: A framework for Information Systems Architecture.

His Framework is based on the six primitive interrogatives: What, How, Where, When, Who and Why.

These interrogatives have been around since time immemorial, but John Zachman had the insight to understand that the same questions, asked at different levels of abstraction, could yield different results and hence he defined six levels of abstraction: Planner, Owner, Designer, Builder, Implementer, Operator.

As the six interrogatives are all unique and non overlapping, they represent six orthogonal domains, resulting in a six dimensional hypercube!

This illustrates the **Complexity** inherent to Information Systems as well as the **Invisibility** - ie our inability to represent all of an information system on a sheet of paper, but Architecture tools have the ability to allow for the establishment and maintenance of the linkages.



**Complex, Resilient and Smart Systems**
Daniel Tokody, József Papp,
László Barna Iantovics,
Francesco Flammini

Hexeract (six-dimensional hypercube) - Petrie polygon Orthographic projections -The representation of the sixth dimension as the Smart Cyberspace. It is quite difficult to illustrate a four-dimensional space in two dimensions. In case of a six-dimensional space, however, it is almost impossible. The figure shows one way of representing a six-dimensional space in two dimensions.

https://www.researchgate.net/figure/Hexeract-41-six-dimensional-hypercube-Petrie-polygon-Orthographic-projections-The_fig3_330643330/download

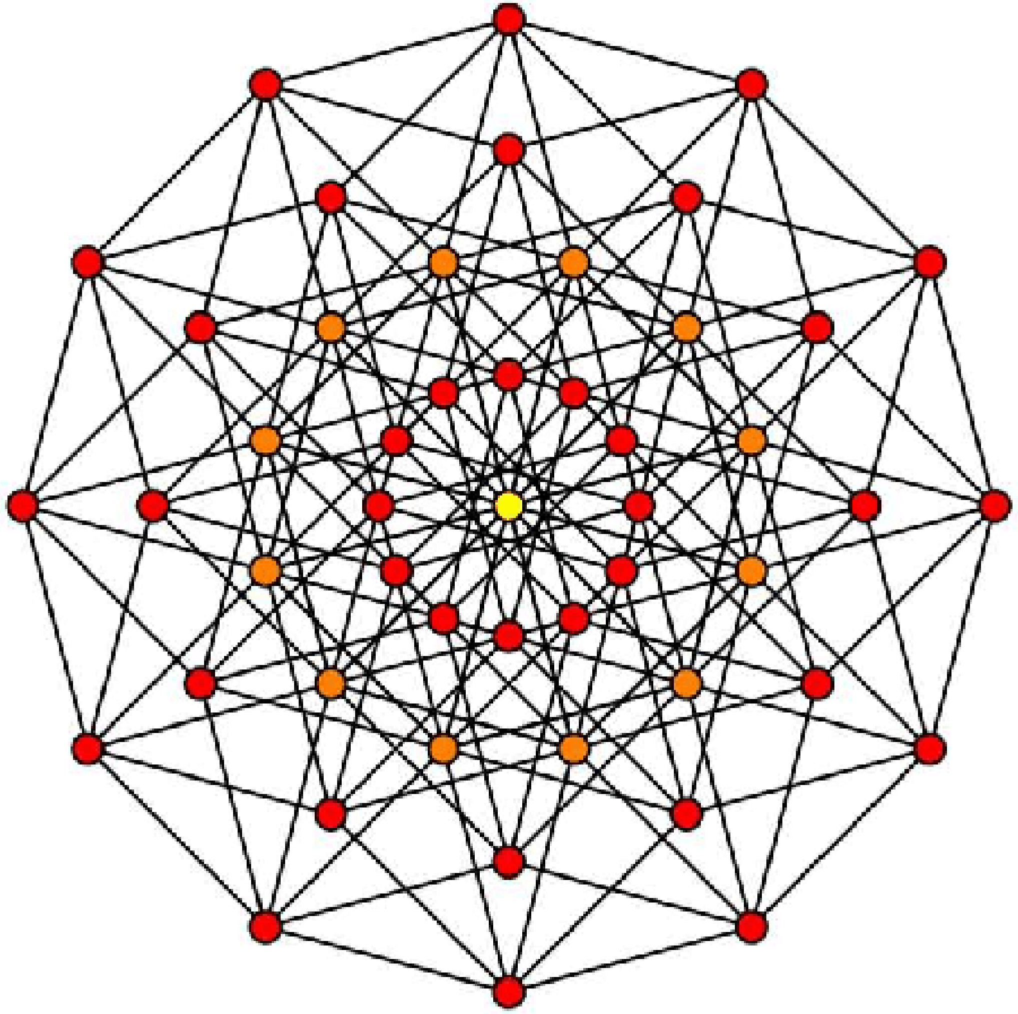
J.A. Zachman: IBM Systems Journal: Volume 26, Number 3, Page 276 (1987)

# The Zachman Framework - Managing Complexity

This is how we use the Zachman Framework when thinking about a situation

| The Zachman Framework (1999 version) | What | How | Where | Who | When | Why |
|---|---|---|---|---|---|---|
| Planner Model | List of Things important to the Business | List of Processes the Business performs | List of Locations where the Business operates | List of Users /Groups /Units of the Business | List of Business Events/Cycles | List of Business Goals/ Strategies |
| Owner Model | Entity relationship Model | Business Processes | Logistics network | Organisation chart | Business Events/ Triggers | Business Plan |
| Designer Model | Data Architecture | Application Architecture | Distributed Systems Architecture | User Interface Architecture | State Transition Diagrams | Business Rules |
| Builder Model | Data Design | Application Design | Hardware Systems design | User Interface Design | State Transition design | Knowledge Design |
| Sub-contractor Model | Data Definition | Program | Network Architecture | Access Architecture | Timings/ Interrupts | Rule Definition |
| Product | Data | Application Code | Communication | Users | Schedules | Rules |

This is what the six dimensional problem actually looks like from one perspective



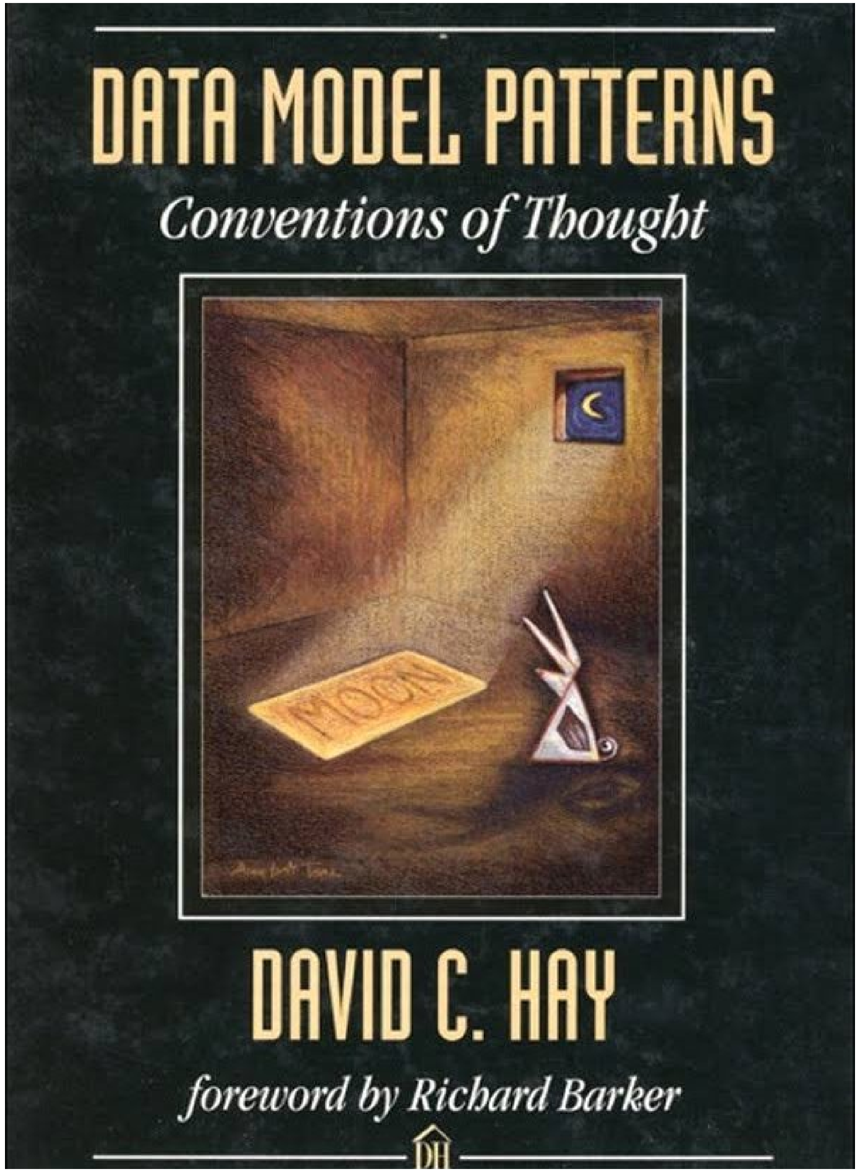
Consider that other Stakeholders may have different, equally valid, perspectives

# Let’s Cut Some Complexity

- Your Software Project always has all of the Zachman Framework models.
    - You have a business rule model – determines why things should be done.
    - You have a process model - a business process is the execution of the business rules.
    - You have a data model – the data model are the “things” upon which the business rules and process impacts.
    - You have a geographic dispersion model – different functionality in different clouds?
    - You have a temporal model – sequences of events.
    - You have a people model – user-interfaces allow users to execute business rules and processes on the data
- All these models are always present at every level of abstraction.
- Even if you chose not to make them explicit, or ignore them, they do not go away. You simply wind up not knowing where to look when disaster strikes.
- Your task is to develop, deliver and maintain the code that reconciles and integrates these orthogonal models of the Hexaract.

# Everything Always Remains Constant, Until it Changes

- David Hay, both a data and origami guru, selected this intriguing image for the cover of one of his books.
- The image depicts David Hay’s origami dog staring at an area of light that has been defined by a script on the floor as “MOON”.
- But, we know that the dog is staring at a patch of light that represents the light from the sun, reflected from the moon, that had been filtered by the opening in the wall.
- We also know that the movement of the earth will cause the patch of light to move and possibly distort the shape, but the inscription on the surface will remain stationary.
- Understand what you really may be looking at, and that shifts in your current reality could happen at any time, possibly invalidating your current models,  requiring frequent interaction and collaboration with the Customer.

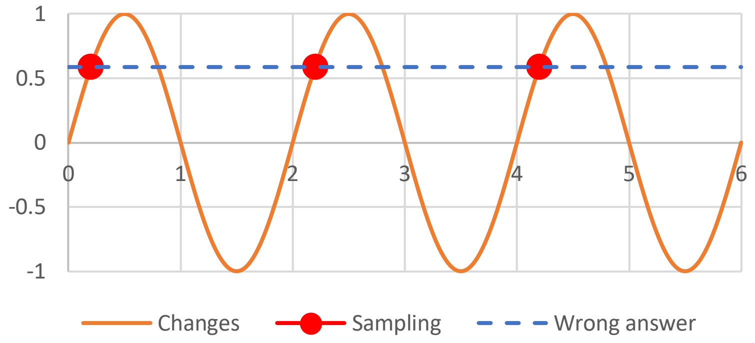


David C Hay, data Model Patterns; Dorset House 1995; ISBN-10  :  0932633749

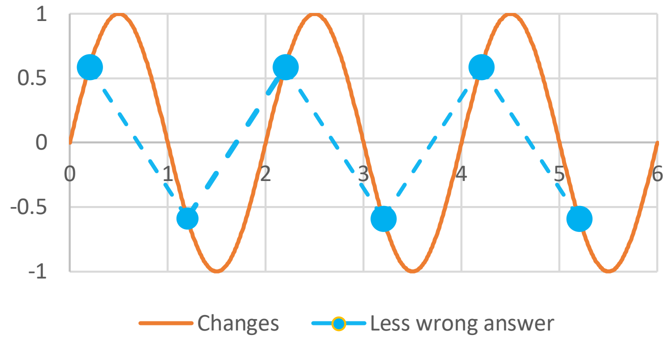
# Detecting Change, Nyquist and Shannon

- The Nyquist theory states that your frequency of measurement, how often you look at something, must be at least twice as high as the phenomena that is changing.
- Shannon states that only that which changes, conveys information.
- Two observations:
  - The frequency of change, both from the User and the Environment, could have a significant impact on the outcome of the project. This requires frequent interaction and customer collaboration.
  - This could explain why most people prefer staring at a log fire to staring at an electrical lamp. (Your retinal response under samples the frequency of the electrical lamp to the extent that it appears not to change.)

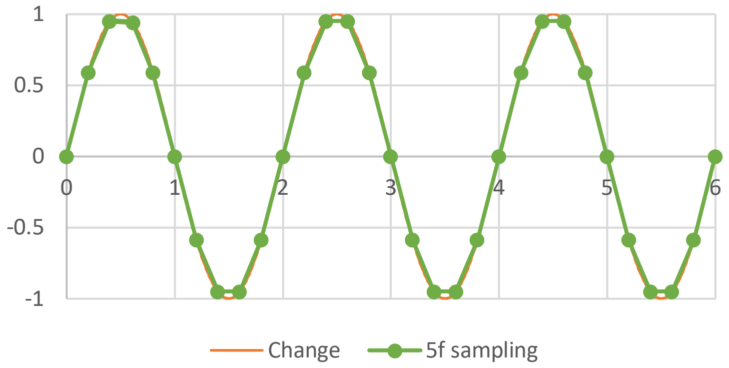
Sampling at the frequency of change.
I do not see any change.



Sampling at twice the frequency of change.
I see something is changing.



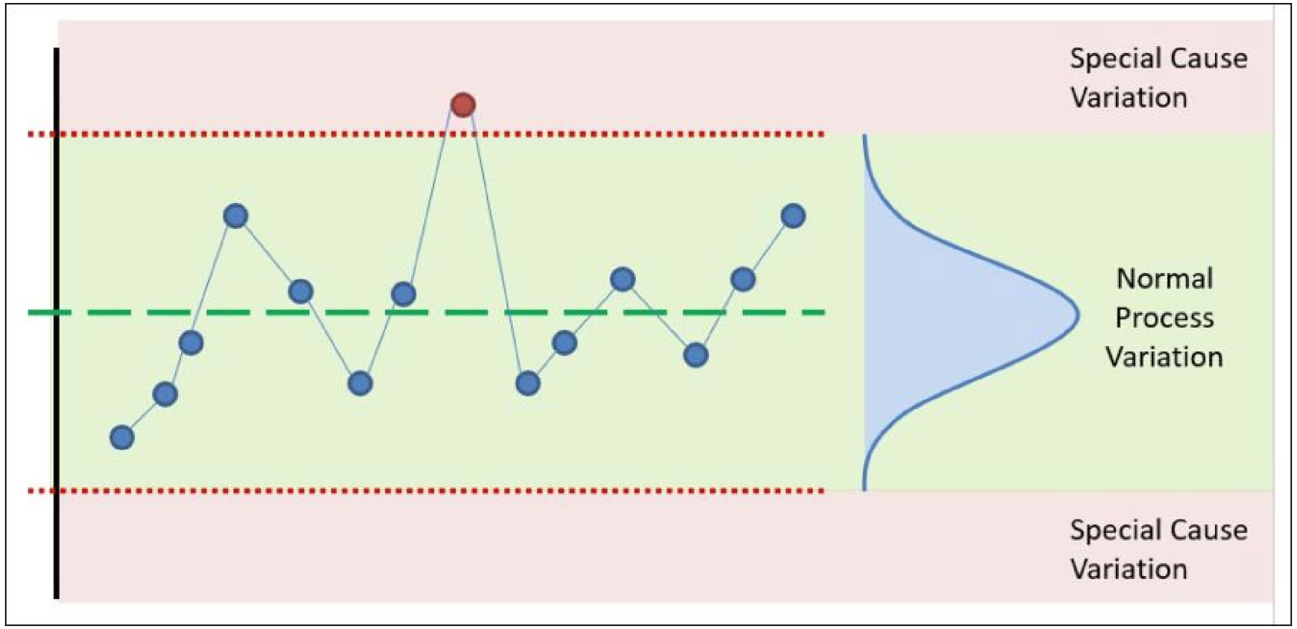
Sampling at more than the frequency of change.
I see more about the change.



https://en.wikipedia.org/wiki/Nyquist%E2%80%93Shannon_sampling_theorem

https://en.wikipedia.org/wiki/Entropy_(information_theory)

# Try Not to be a Control Freak; Just Enough Measurement.

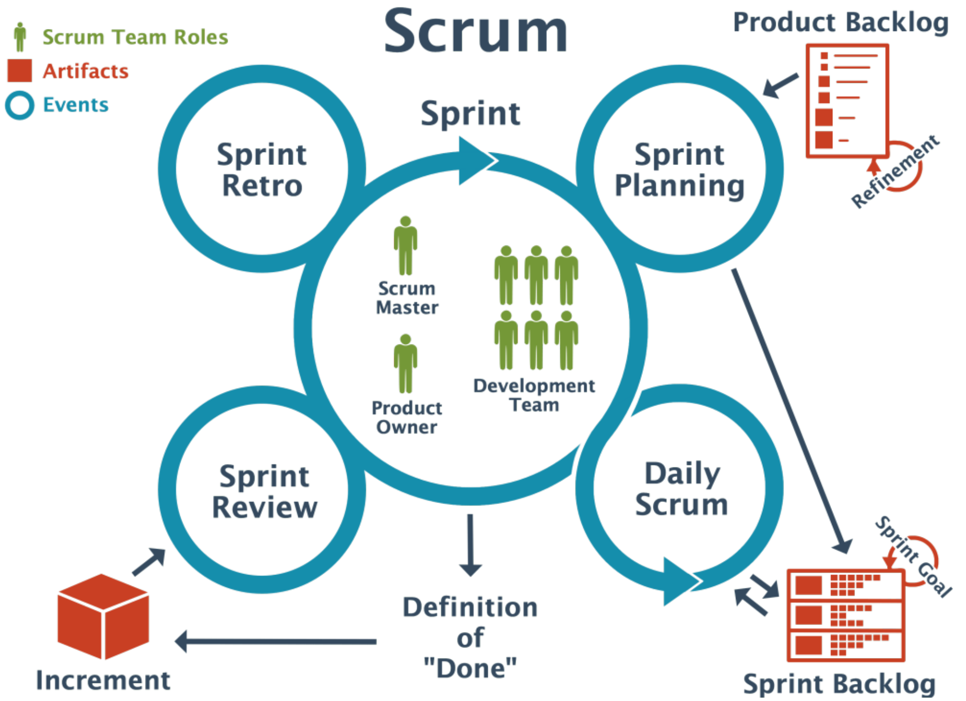


- Peter Drucker famously said: What gets measured gets managed.
- Having previously looked at frequency of measurement, we must also allow for natural process variation. Here Demming comes into play.
- Overcontrolling a process by imposing control actions whilst the process is within the statistical band, is called tampering. This wastes resources and could lead to unwanted process behaviour.
- Process control band excursions provide information about process stability.
- Warning: Normalised processes could exclude the outliers. This could be really important.

https://cqeacademy.com/cqe-body-of-knowledge/quantitative-methods-tools/statistical-process-control-spc/

# Are These Methodologies Comparable?

Well defined processes that allow even incompetents like me to deliver reasonable performance?



Scrum Team Roles
Artifacts
Events

Scrum

Sprint

Sprint Retro

Sprint Planning

Product Backlog

Refinement

Scrum Master

Development Team

Product Owner

Sprint Review

Daily Scrum

Sprint Goal

Sprint Backlog

Definition of "Done"

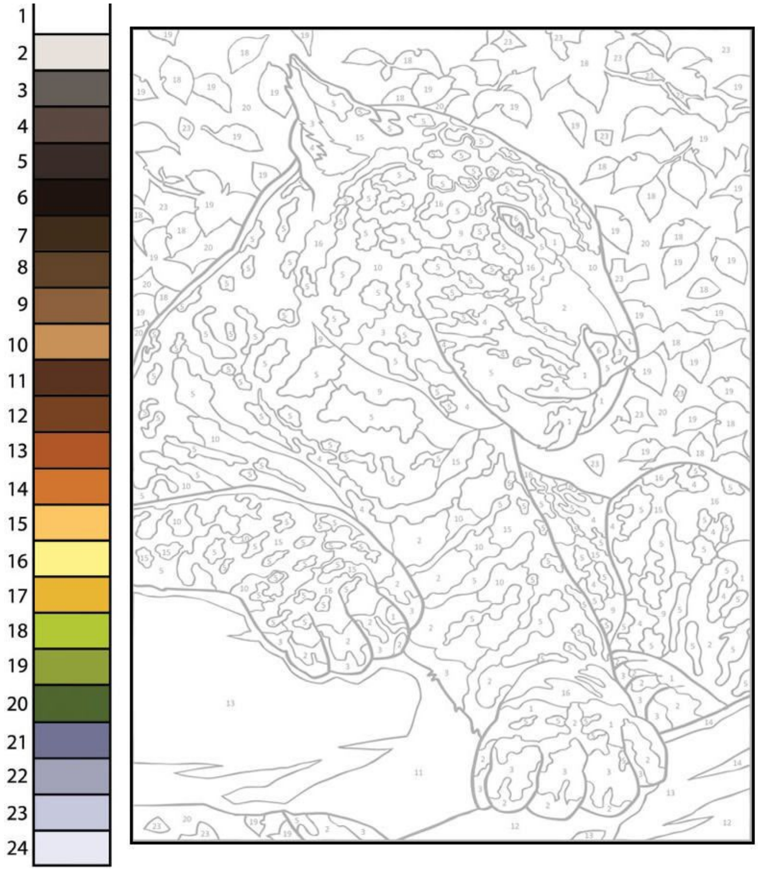
Increment

According to the 2017 Scrum Guide™

v2.3 - jordanjob.me

https://jordanjob.me/blog/scrum-diagram

Paint by Numbers



https://za.pinterest.com/pin/579275570816531687/

# Looking at a Popular Agile Methodology

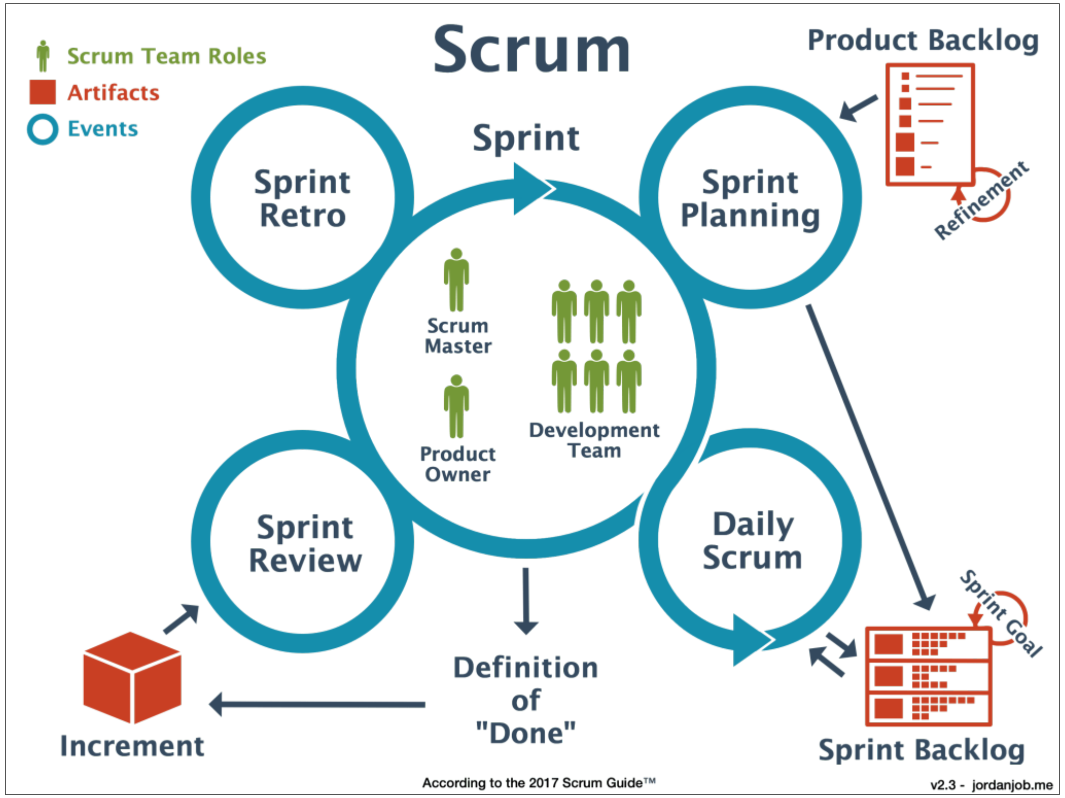
## Agile Manifesto Values

**Individuals and interactions** over processes and tools

**Working software** over comprehensive documentation

**Customer collaboration** over contract negotiation

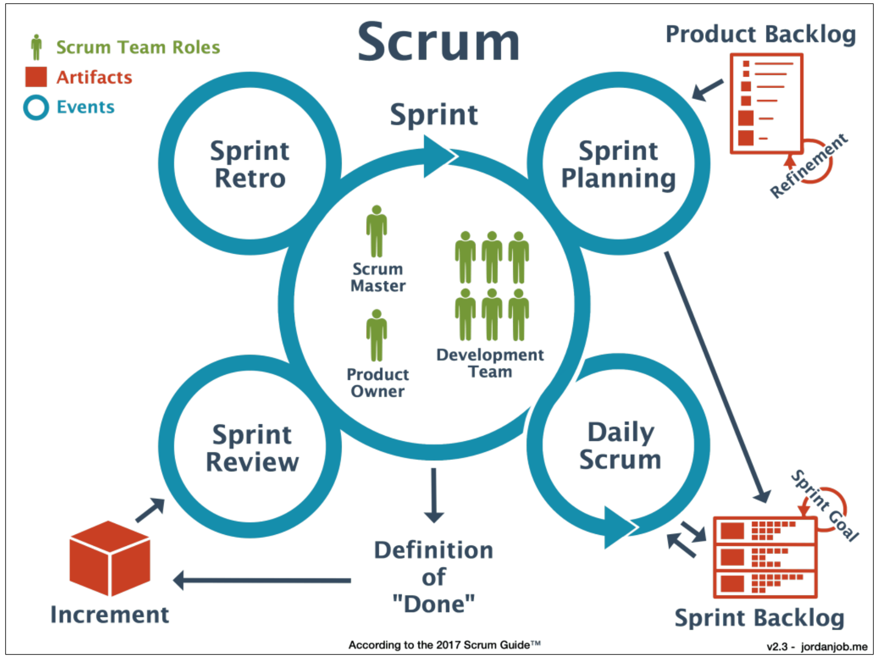
**Responding to change** over following a plan



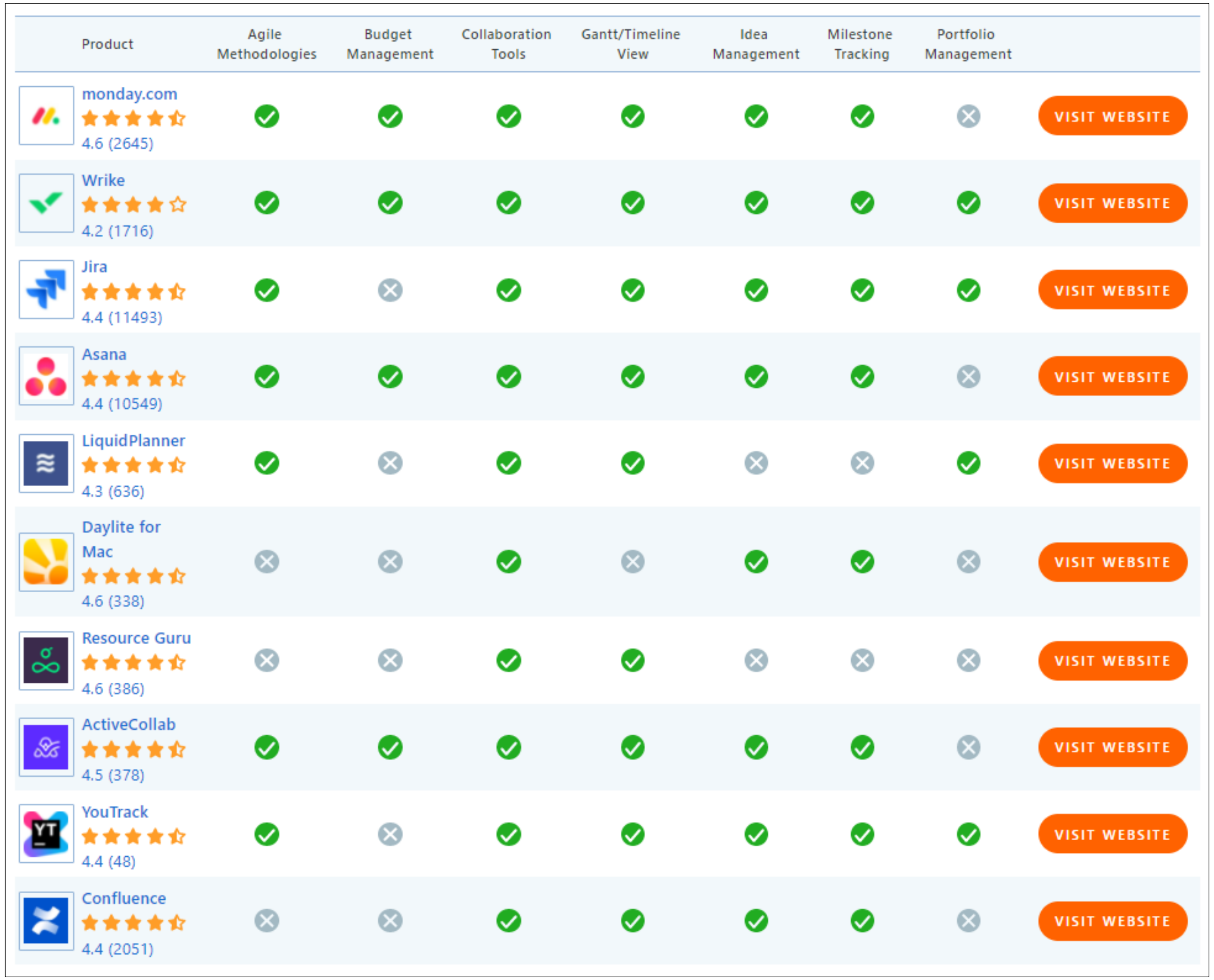
https://jordanjob.me/blog/scrum-diagram

# **Individuals and Interactions** over Processes and Tools

## Process, supported by tools



## Tools, having processes, used to support processes



| Product | Agile Methodologies | Budget Management | Collaboration Tools | Gantt/Timeline View | Idea Management | Milestone Tracking | Portfolio Management | |
|---|---|---|---|---|---|---|---|---|
| monday.com<br>4.6 (2645) | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✗ | VISIT WEBSITE |
| Wrike<br>4.2 (1716) | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | VISIT WEBSITE |
| Jira<br>4.4 (11493) | ✓ | ✗ | ✓ | ✓ | ✓ | ✓ | ✓ | VISIT WEBSITE |
| Asana<br>4.4 (10549) | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✗ | VISIT WEBSITE |
| LiquidPlanner<br>4.3 (636) | ✓ | ✗ | ✓ | ✓ | ✗ | ✗ | ✓ | VISIT WEBSITE |
| Daylite for Mac<br>4.6 (338) | ✗ | ✗ | ✓ | ✗ | ✓ | ✓ | ✗ | VISIT WEBSITE |
| Resource Guru<br>4.6 (386) | ✗ | ✗ | ✓ | ✓ | ✗ | ✗ | ✗ | VISIT WEBSITE |
| ActiveCollab<br>4.5 (378) | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✗ | VISIT WEBSITE |
| YouTrack<br>4.4 (48) | ✓ | ✗ | ✓ | ✓ | ✓ | ✓ | ✓ | VISIT WEBSITE |
| Confluence<br>4.4 (2051) | ✗ | ✗ | ✓ | ✓ | ✓ | ✓ | ✗ | VISIT WEBSITE |

# Scrum Teams as Agile Warriors

- The networked war fighter, with constant access to developing scenarios, has revolutionised the concept of Military Small Team Tactics.
- These Special Forces represent small numbers, typically less than ten, of highly competent, cross discipline, members that operate as one, achieving the Commander's Intent by their own means.
- Key terms are Competence, Shared Values, Trust, Respect, Discipline.
- The Commander facilitates the mission by removing impediments.
- The teams adapt and adopt command and control structures as required and do not need detailed planning and approval forced upon them from on high.
- This could be an acceptable model for the Development Team members

**Agile Team**

Outcomes focussed

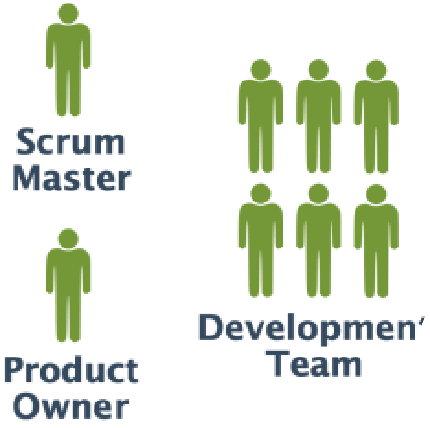
Operational autonomy

Shared values

Trust

Respect

Discipline



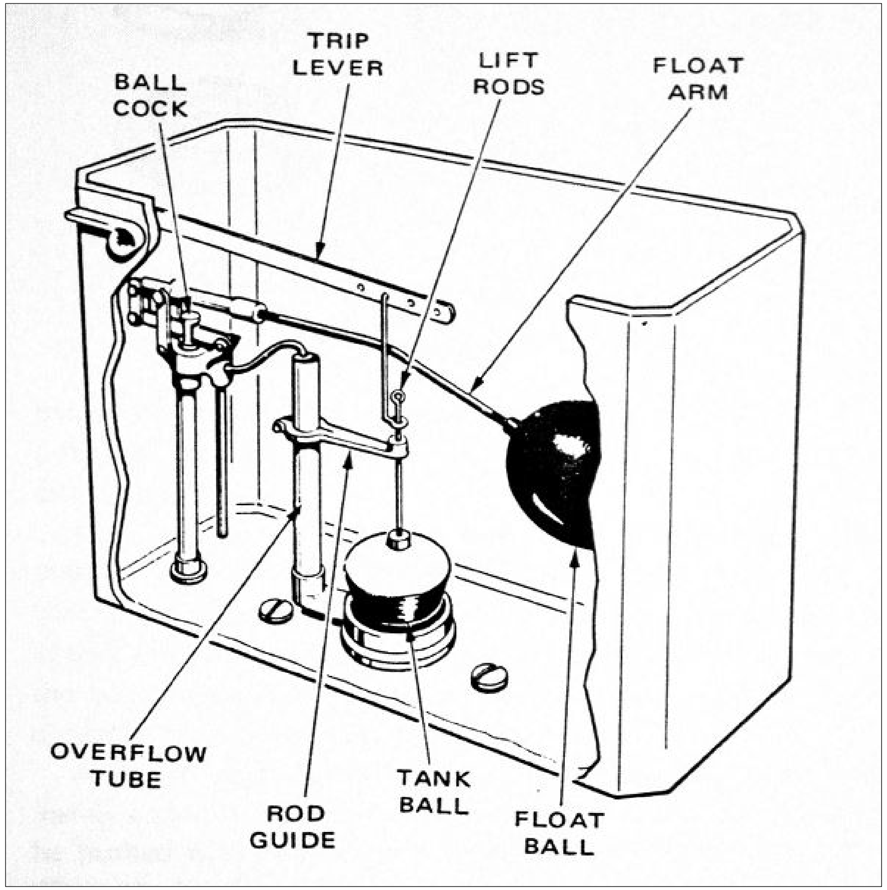
Constant practice

Innovative

Fail safely

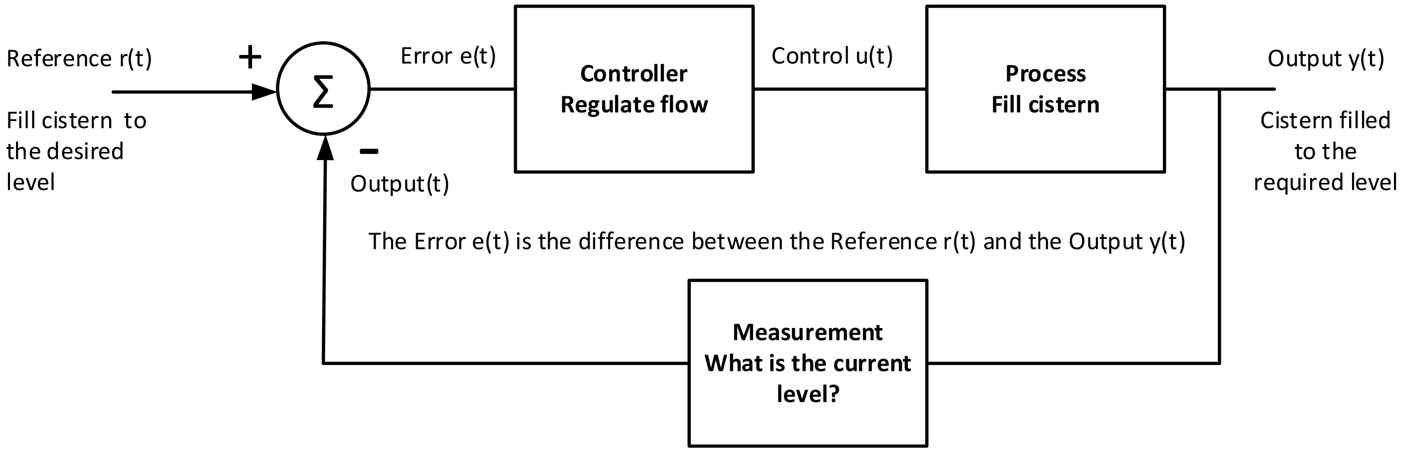
# Control Theory 101

- Control theory is about being able to automatically manage a process to achieve optimal results.
- It typically involves a required Output that is a determined by a Reference Input, set point or desired value.
- The Input is applied to a Process that contains a means to measure the Output, determine the Error, or deviation, from the Reference and uses a Controller to provide a modified Input to the Process in order to eliminate the Error.



https://inspectapedia.com/plumbing/Toilet_Tank_Operation.php

# Example – Plumbing 101 – Proportional Control



Reference r(t)

Fill cistern to the desired level

+

Σ

−

Error e(t)

**Controller**
**Regulate flow**

Control u(t)

**Process**
**Fill cistern**

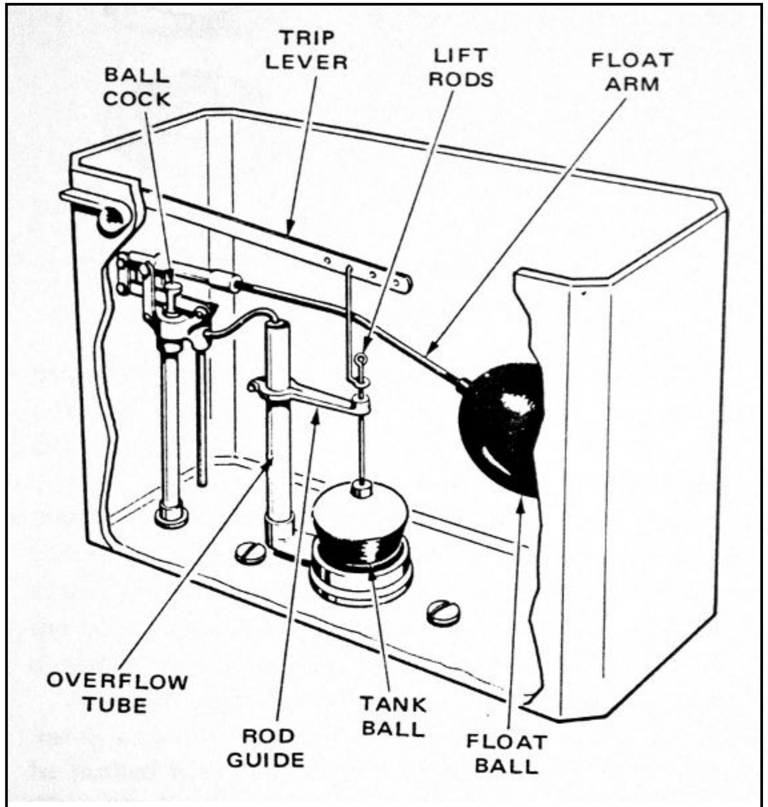
Output y(t)

Cistern filled to the required level

Output(t)

The Error e(t) is the difference between the Reference r(t) and the Output y(t)

**Measurement**
**What is the current level?**



- The water supply to the cistern is regulated by opening and closing the ball cock.
- The ball cock is connected to the float arm and the float which measures the current value of the water level.
- The float arm is connected to the ball cock and  and shuts off the water flow when the set point is reached.
- In this way the ball cock only responds to the difference between the reference level and the current water level.
- Note that the system has proportional control in as much as a low water level will result in the ball cock being maximally opened and a rising water level will cause the flow to be gradually reduced.

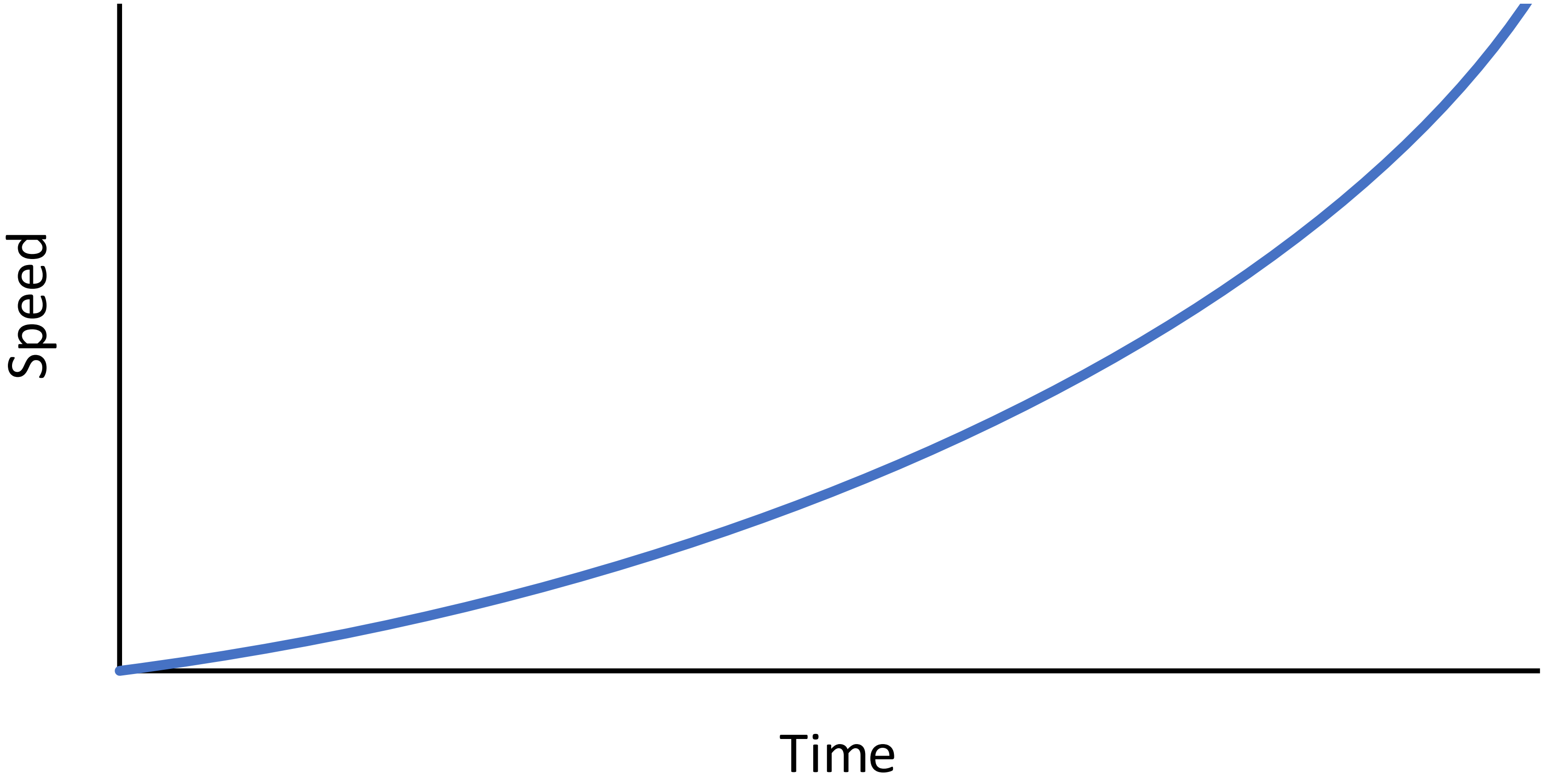
**Question:**

What would happen if you only measured the water level every two weeks.
Would this count as a waterfall model?
You have to be Agile; continuous measurement and adaptation.

https://inspectapedia.com/plumbing/Toilet_Tank_Operation.php

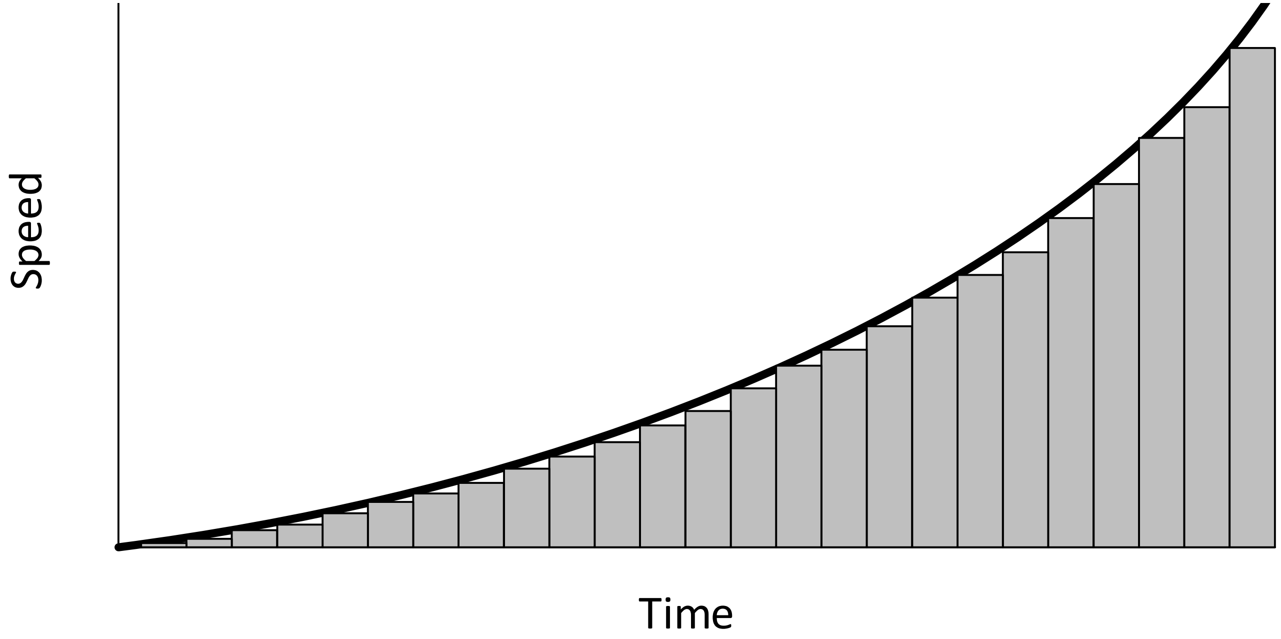
# Integration and Differentiation

- Consider the graph that depicts the Speed of a vehicle, V, versus the elapsed Time from the point of departure.
- How do we figure out where we have been and what could happen next?

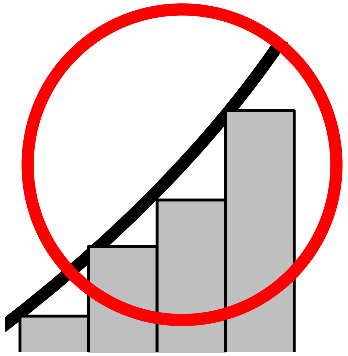


# Integration, Finding Distance, a Backwards Glance

- Let us determine how far we have travelled by defining five minute intervals, and for each of these, get the product of time and velocity. ( Note that speed = $\frac{distance}{time}$. Hence $\frac{distance}{time} * time = distance$.)
- The final distance is simply the sum of all of these *distance* rectangles.



What about this error?

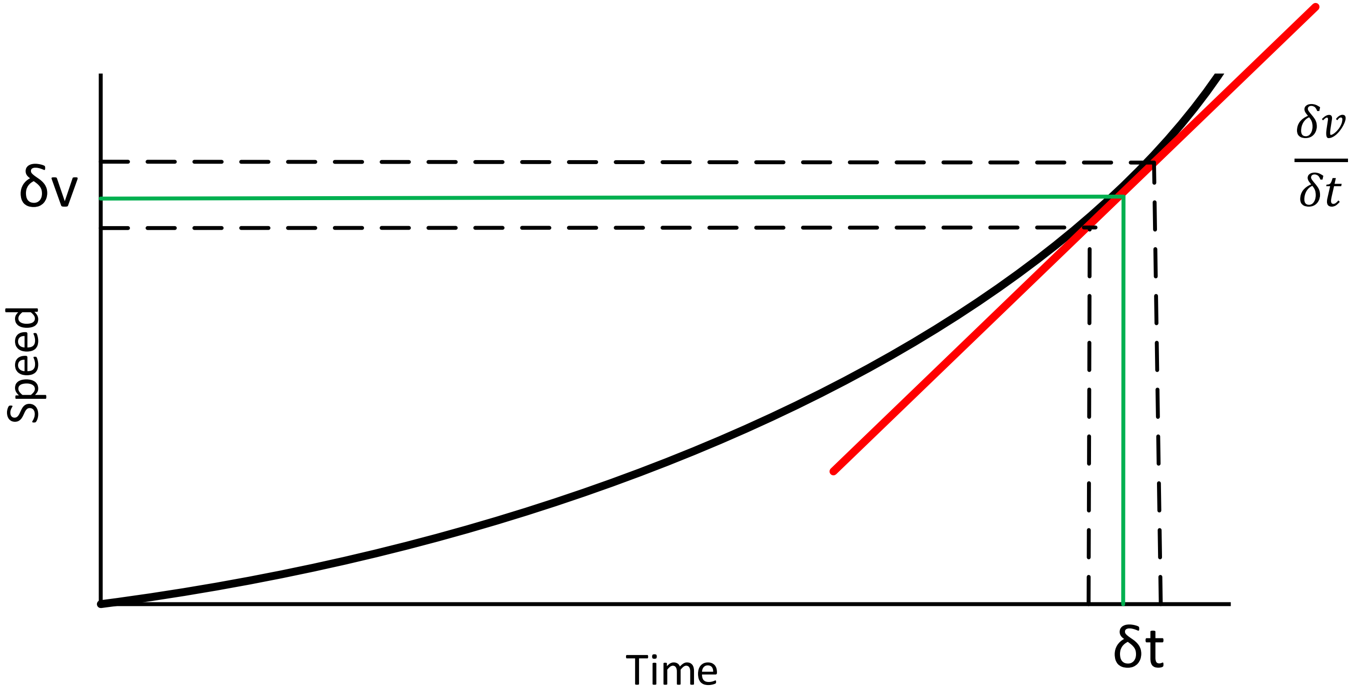


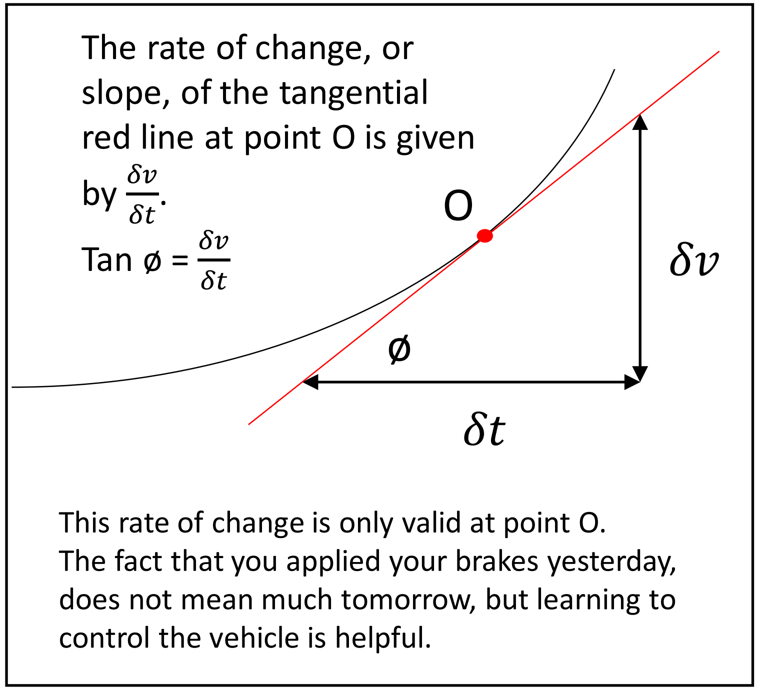
This is a quantification error due to our 5 minute time intervals.
By making the intervals infinitesimally small we will eliminate this error.

- By summing all of the Speed/Time rectangles we have Integrated the area under the curve to yield the total distance from the origin.
- Integration effectively gives us a backward glance; a historical view.

# Differentiation, Rate of Change, What is Coming?

- The change in the speed graph with respect to time, $\frac{\partial v}{\partial t}$, yields the acceleration at the specific point in time where the measurements were done.
- The instantaneous rate of change of the velocity is given by the slope of the curve at the specific point, the change in speed divided by the in time it took for the change to happen.
- This yields an indication of what could be expected to happen in the future, based on the events of that instant.
- It is similar to timeously applying the brakes of your vehicle to avoid hitting the vehicle in front of you.

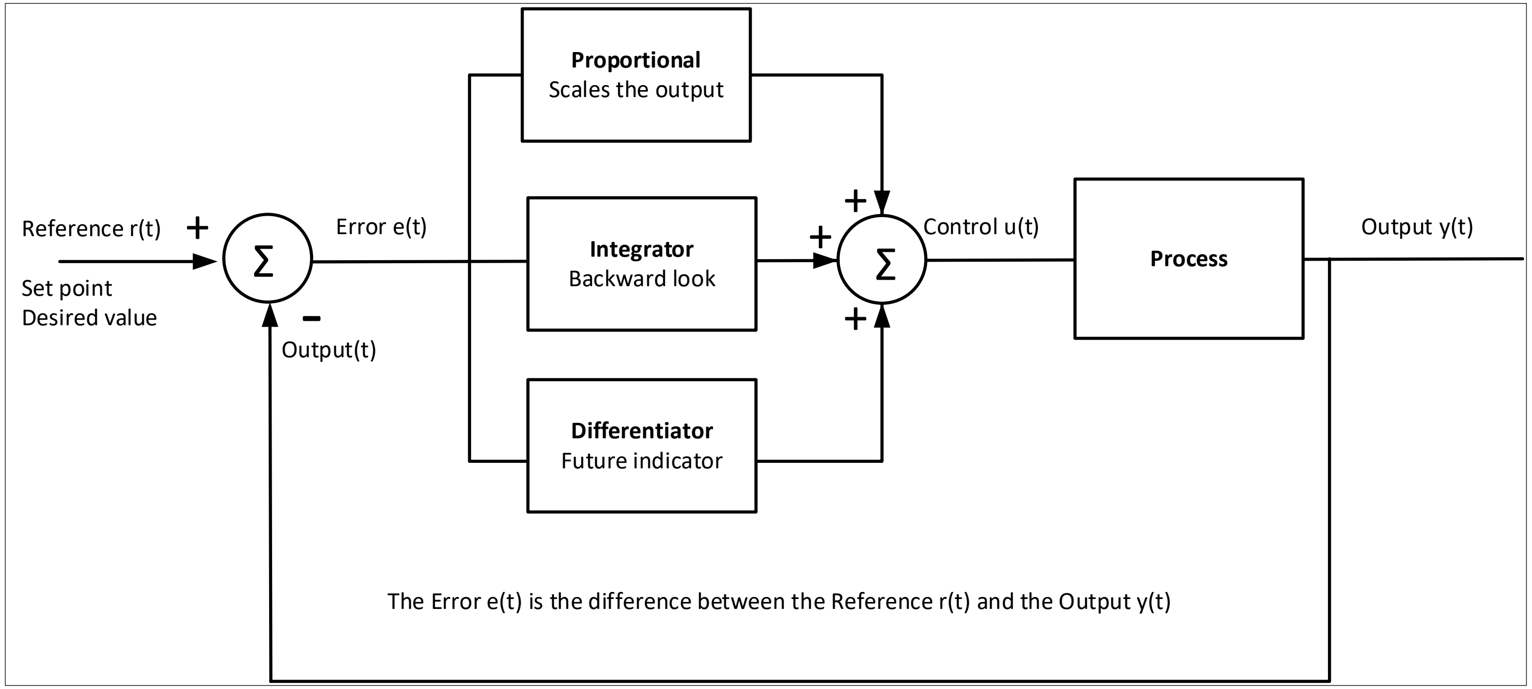



The rate of change, or slope, of the tangential red line at point O is given by $\frac{\delta v}{\delta t}$.

Tan ø = $\frac{\delta v}{\delta t}$

This rate of change is only valid at point O.
The fact that you applied your brakes yesterday, does not mean much tomorrow, but learning to control the vehicle is helpful.

# Enter the Proportional, Integral, Differential (PID) Controller!

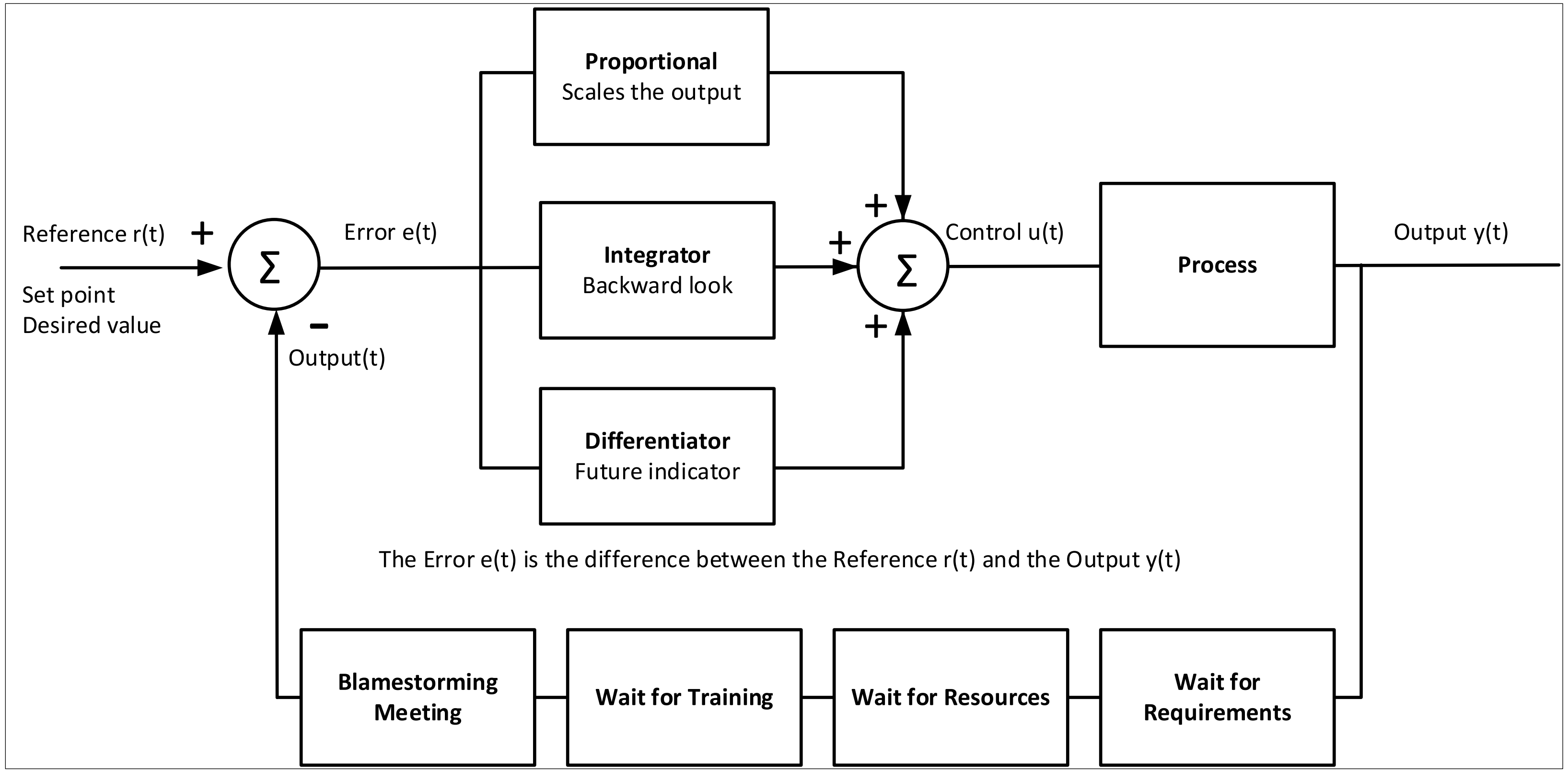


The PID controller has three paralleled capabilities that provides the Control input to the Process.

- The Proportional function that has the ability to scale the error value
- The Integrator function, as we have seen, provides a backwards look.
- The Differentiator function yields the rate of change and hence provides an indication of future conduct.

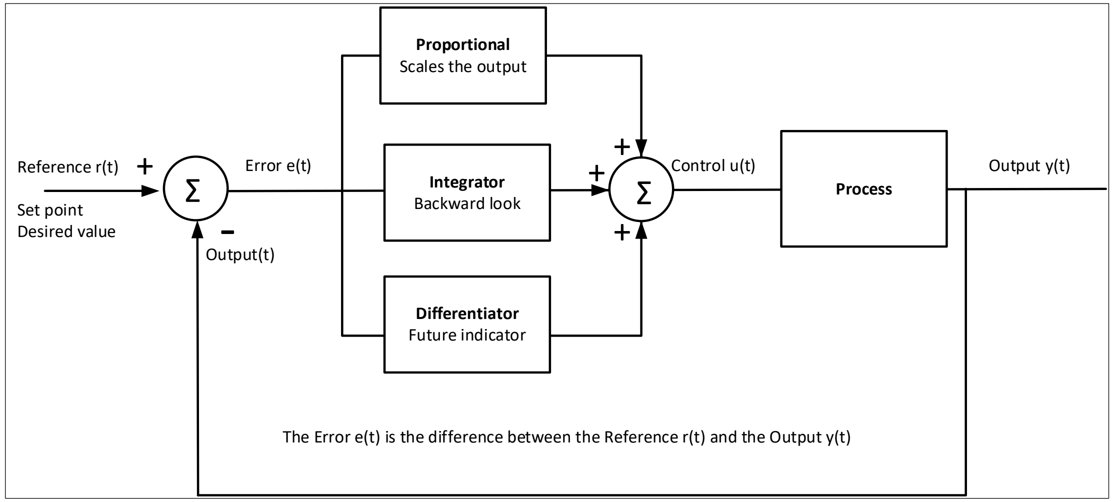
All of these control strategies are continuously applied and updated as fast as the feedback loop can manage.

# Delays in the Feedback Loop Could be Fatal



- Delays in the feedback loop could result in the process control strategy not being updated and implemented.
- This could result in novel process behaviour which could include total failure of the project, disillusionment of the team members and the loss of clients.

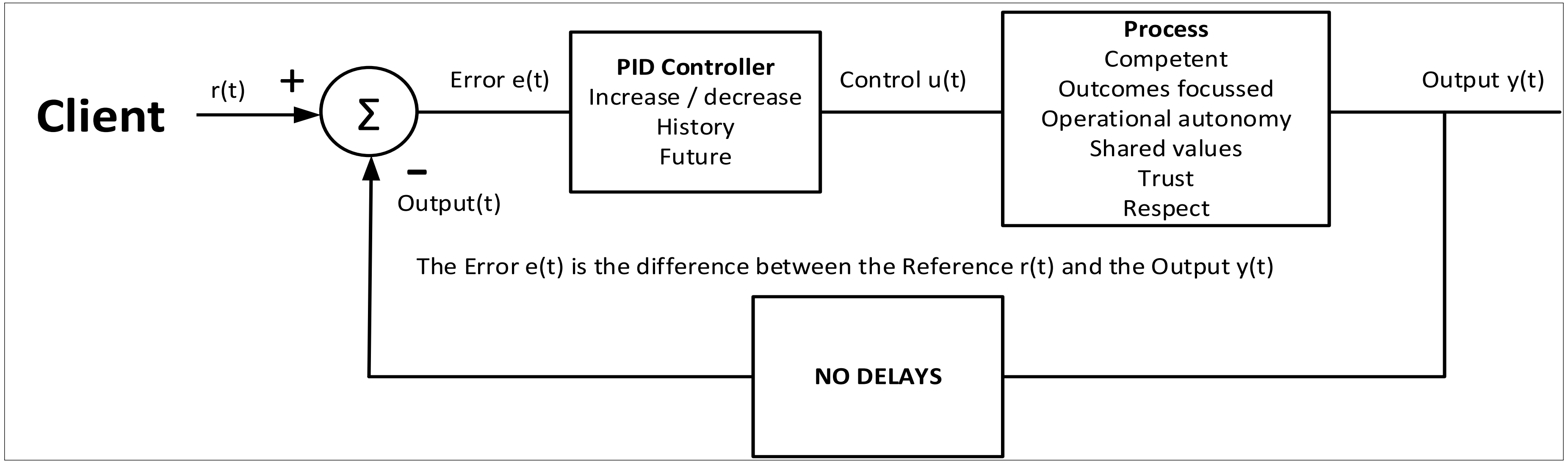
# Applying PID Control Principles to a Process



The Error e(t) is the difference between the Reference r(t) and the Output y(t)

- This process is very similar to the natural way of walking into a dark room.
- Determine where you want to be, determine the Goal or Purpose
  - It could be a very good idea to talk to the Client, not the Product Owner
    - Client: requirements focus: I need this feature
    - Product Owner: internal focus, burn down chart focus: Let me explain how you can cope without it.
- Take a small step in the direction of least peril; the easiest way to recover from a mistake.
- See where you are now, compared to where you wanted to be.
- Based on how you got there and the effect of the change in position, consider altering your stride length and/or direction.
- Repeat from: Take a small step…..
- Now stub your toe against an unseen object. What is the time delay for the pain to his you and the implementation of remedial action? Fast feedback loop?

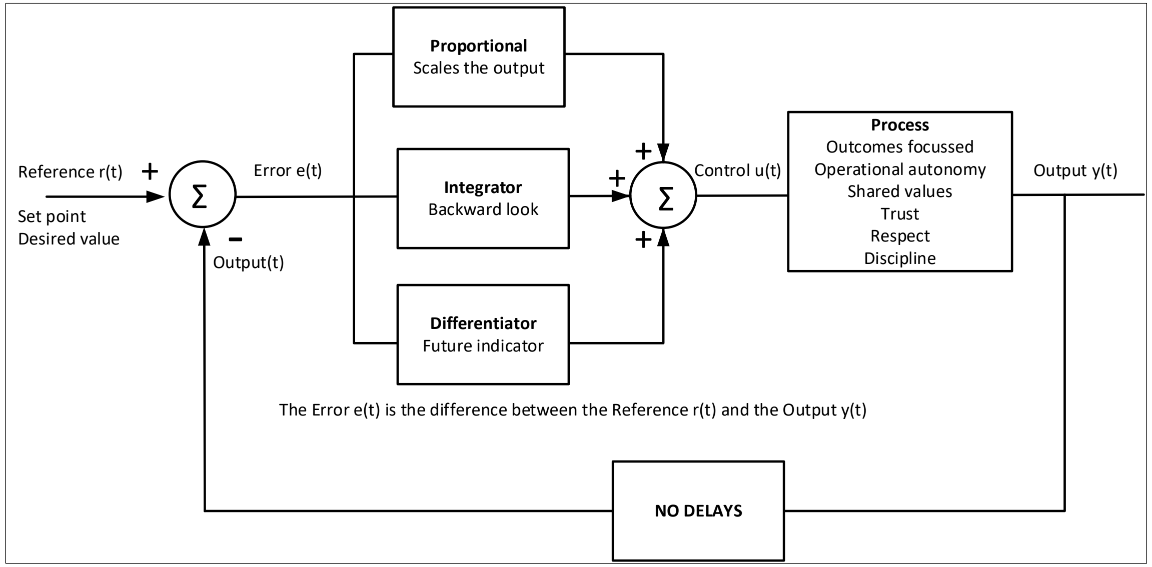
# The New!! Improved!! Agile!! PID Project Management Process



- Remarks
  - Goal clarity is important. If you aim at nothing, you will hit it.
  - Continually speak to the client.
  - Small steps in the direction of least peril to eliminate the error term
  - Continual measurement against the Client requirements and adaptation to eliminate errors.
    - What have we tried in the past?
    - Can we scale that which works and moderate that which does not work?
    - Where do we go next?
  - Immediate communication, error value instantaneously available, hence eliminate delays and distractions in the feedback loop.
  - Manage the residual error and drive it towards zero, there is no blame or shame if, at the end of a cycle, there still is a residual error, all it means is that a different approach could be beneficial.

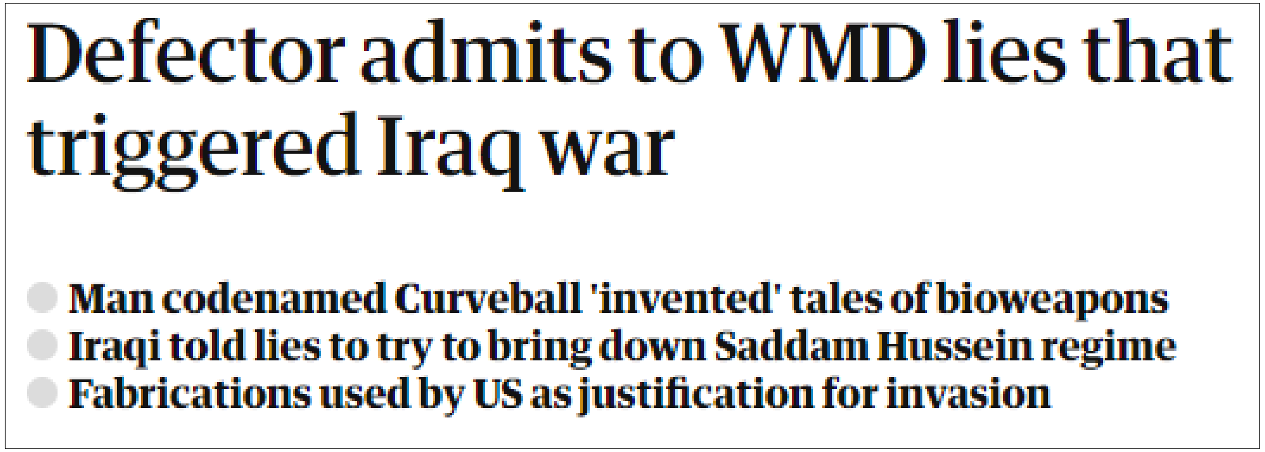
# What a Wonderful, Lightweight, Self Directed Solution!

- Before rushing off to patent the PID approach, which has only been around since 1911, then writing a Sacred Manual, raising a Certification Body, opening Seminaries to educate and Certify the Faithful, developing a Subscription Based Community and erecting a Paywall to monetise access, a word of caution may be in order.
- Returning to the Military concept of Small Team Tactics and the Commander’s Intent.
  - Clearly stated goal
  - Good intelligence, understand the environment
  - Logistics, supply and evacuation, backup
  - Tools and tactics
  - Small steps in the best direction
  - Short execution cycle
  - Continual measurement and feedback
  - Mentoring and risk management
  - Well defined success measures
  - Discipline



For interest: [https://www.presby.edu/doc/military/ROTC-AC-TACSOP.pdf](https://www.presby.edu/doc/military/ROTC-AC-TACSOP.pdf)

# User Stories May Require Vetting



Defector admits to WMD lies that triggered Iraq war

- Man codenamed Curveball 'invented' tales of bioweapons
- Iraqi told lies to try to bring down Saddam Hussein regime
- Fabrications used by US as justification for invasion

https://www.theguardian.com/world/2011/feb/15/defector-admits-wmd-lies-iraq-war

# My Thinking Framework: a Personal View

There is no Agile methodology that reduces the complexities of the Universe to a two dimensional, flat, surface.

The six dimensional Zachman Framework has some very good features:

- It is fractal and infinitely adaptable.
- It provides a complete thinking tool.
- It allows me to think about detail aspects without losing contextual awareness and integration requirements.
- Security is required at the cell boundaries, require protocols.
- I can even get it to dance to the Agile tune.

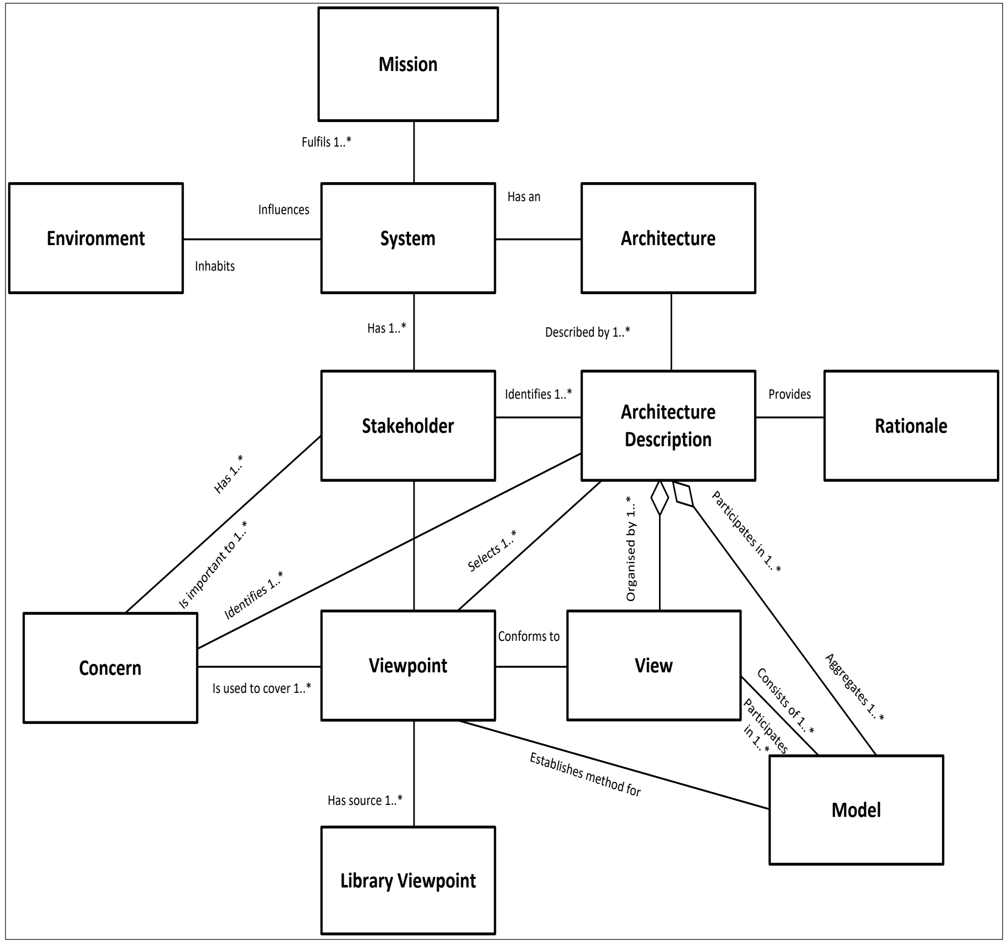
Reducing real world situations to the two dimensional Consulting space of Time and Money may not be appropriate.

Please consider using the Zachman Framework as a checklist.

| | | Six dimensions | | | | | |
|---|---|---|---|---|---|---|---|
| | An example of the Zachman Framework (1999) with SCRUM Roles | **What** | **How** | **Where** | **Who** | **When** | **Why** |
| Stakeholder concerns | **Client Model** | List of Things important to the Business | List of Processes the Business performs | List of Locations where the Business operates | List of Users /Groups /Units of the Business | List of Business Events/Cycles | List of Business Goals/ Strategies |
| | **Product Owner Model** | Entity relationship Model | Business Processes | Logistics network | Organisation chart | Business Events/ Triggers | Business Plan |
| | **Theme Model** | Data Architecture | Application Architecture | Distributed Systems Architecture | User Interface Architecture | State Transition Diagrams | Business Rules |
| | **Epic Model** | Data Design | Application Design | Hardware Systems design | User Interface Design | State Transition design | Knowledge Design |
| | **Story Model** | Data Definition | Program | Network Architecture | Access Architecture | Timings/ Interrupts | Rule Definition |
| | **Sprint** | Data | Application Code | Communication | Users | Schedules | Rules |

SCRUM Roles: See Rubin KS; Essential Scrum: A Practical Guide to the Most Popular Agile Process (Addison-Wesley Signature Series (Cohn)) 2013; ISBN-13: 978-0137043293

# My Thinking Process IEEE 1471: 2000



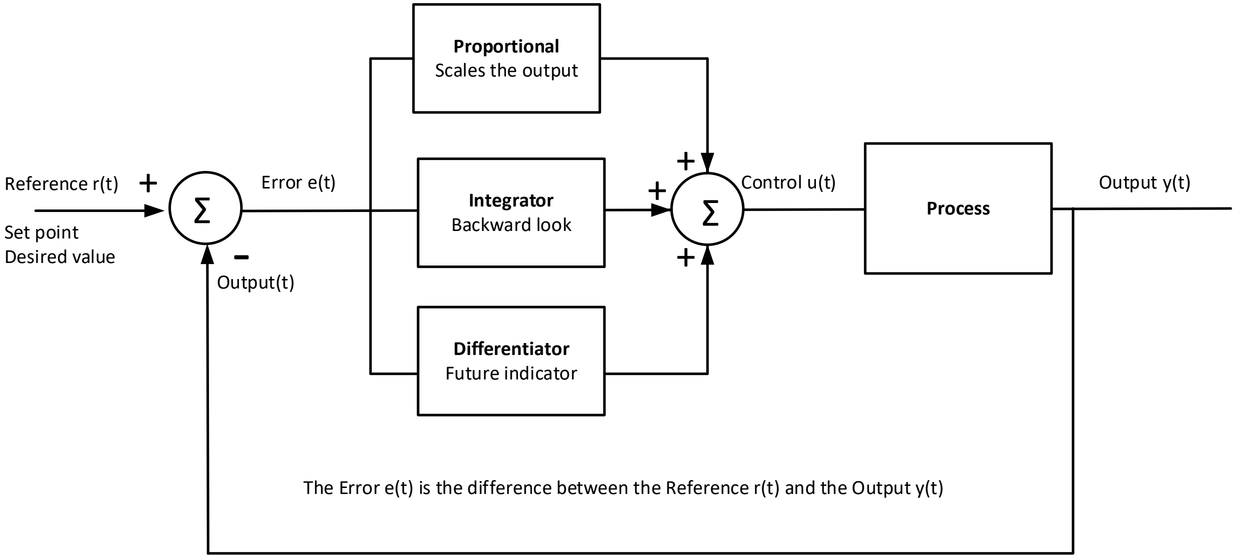
https://en.wikipedia.org/wiki/IEEE_1471

http://www.mit.edu/~richh/writings/hilliard-role-of-AFs.pdf

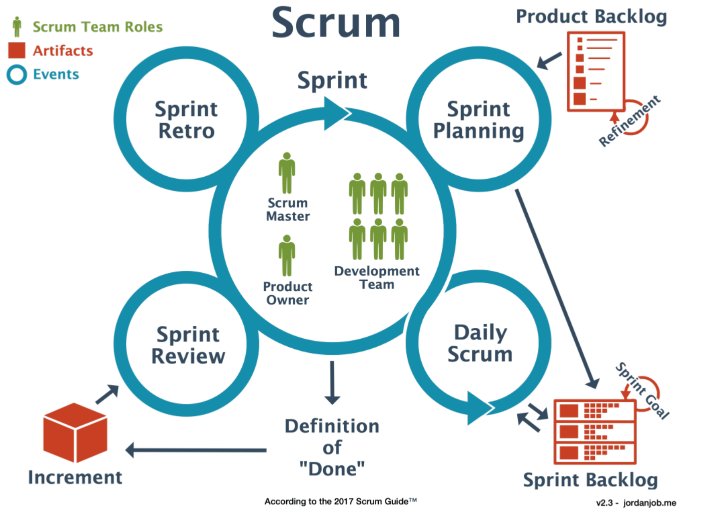
- This has only been around since 1995, formalised in 2000.
- Expression of the system, in terms of goals, and its evolution within its environment
- Communication among the system stakeholders
- Evaluation and comparison of architectures in a consistent manner
- Planning, managing, and executing the activities of system development
- Expression of the persistent characteristics and supporting principles of a system to guide acceptable change
- Verification of a system implementation’s compliance with an architectural description
- Recording contributions to the body of knowledge of software-intensive systems architecture
- Encourages the appropriate re-use of existing architecture artefacts

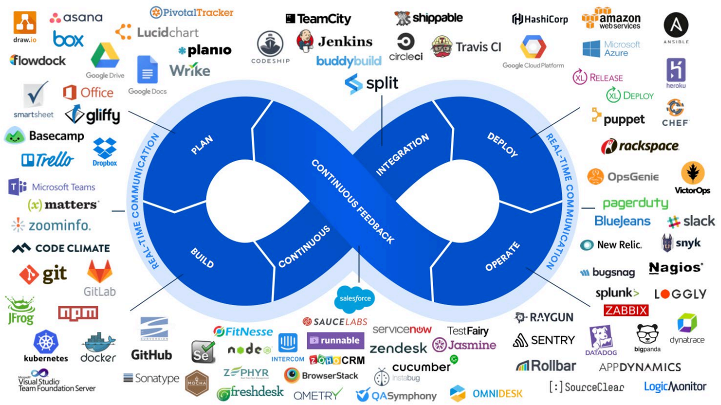
# The Natural Fate of Lightweight Approaches?

## How do we get from here → To here



The Error e(t) is the difference between the Reference r(t) and the Output y(t)



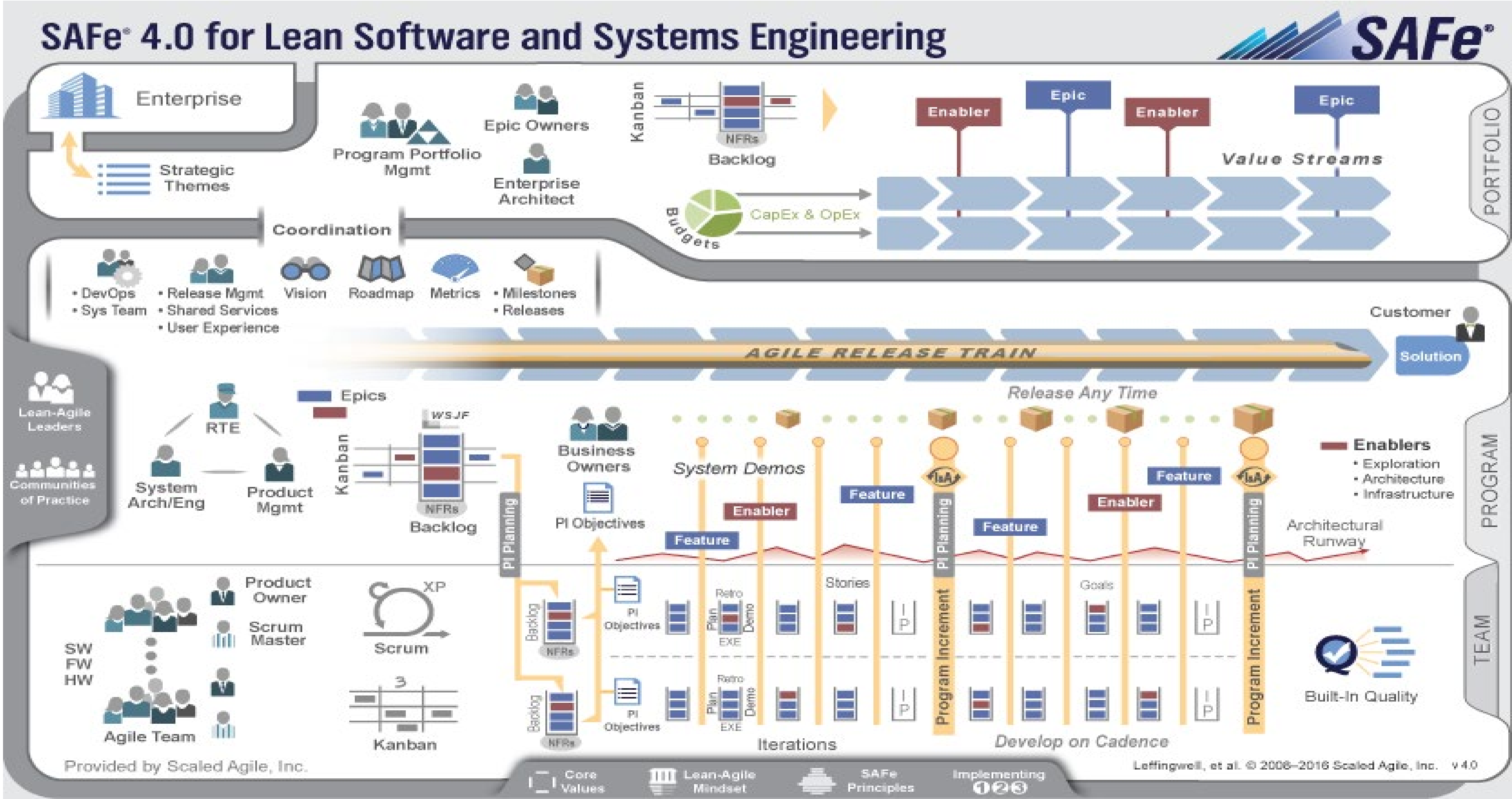


# Administratium - the Existential Threat

- The heaviest element known to science was recently discovered by investigators at a major U.S. research university.
- The element, tentatively named administratium, has no protons or electrons and thus has an atomic number of 0.
- However, it does have one neutron, 125 assistant neutrons, 75 vice neutrons and 111 assistant vice neutrons, which gives it an atomic mass of 312.
- These 312 particles are held together by a force that involves the continuous exchange of meson-like particles called morons.
- A minute amount of administratium causes one reaction to take over four days to complete when it would have normally occurred in less than a second.
- It tends to concentrate at certain points such as government agencies, large corporations, and universities.
- Scientists point out that administratium is known to be toxic at any level of concentration and can easily destroy any productive reaction where it is allowed to accumulate.
- Attempts are being made to determine how administratium can be controlled to prevent irreversible damage, but results to date are not promising.

https://www.mit.edu/people/dmredish/wwwMLRF/links/Humor/Administratium.html

# I rest my case!



https://manifesto.co.uk/scaling-agile-with-safe/

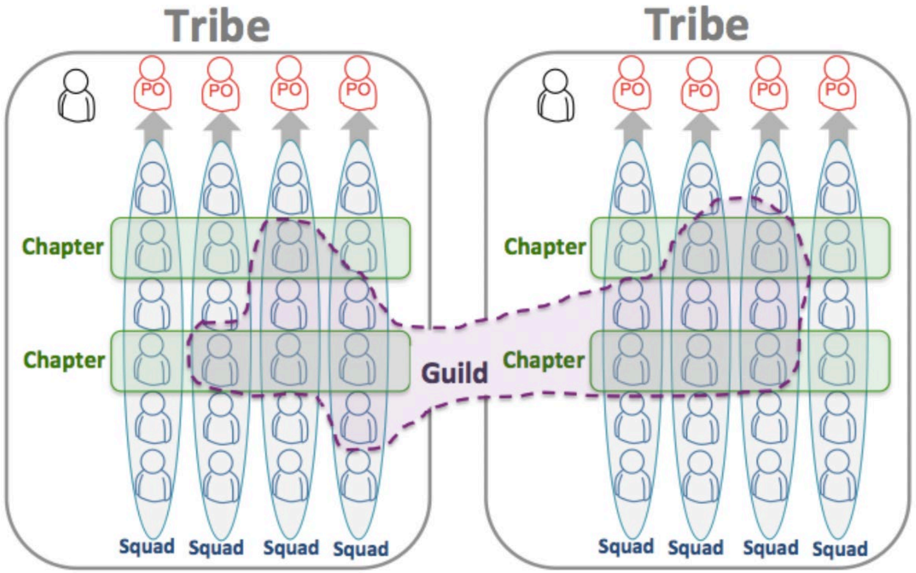
# Small Team Tactics Really Work! Ask Spotify

### The Spotify Way

- A Squad is similar to a Scrum team and is designed to feel like a mini-startup.
- They sit together, and they have all the skills and tools needed to design, develop, test, and release to production.
- They are a self-organizing team and decide their own way of working - some use Scrum sprints, some use Kanban, some use a mix of these approaches.
- The Development Team is the Release team.
- Hack days encourage experimentation and innovation, and provide a fail-safe environment.
- Open and honest communication, collaboration and sharing.
- Shared values and discipline.

## The Agile Software Development Manifesto

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



**Please do read**

Scaling Agile @ Spotify with Tribes, Squads, Chapters & Guilds

Henrik Kniberg & Anders Ivarsson

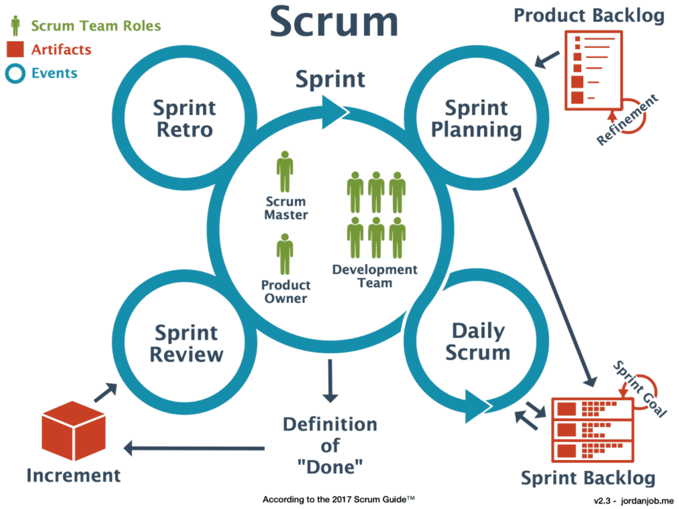
Oct 2012

https://blog.crisp.se/wp-content/uploads/2012/11/SpotifyScaling.pdf

# The Fallacy of Teaching Methodologies

or

Why I still am not a World Famous Chef!



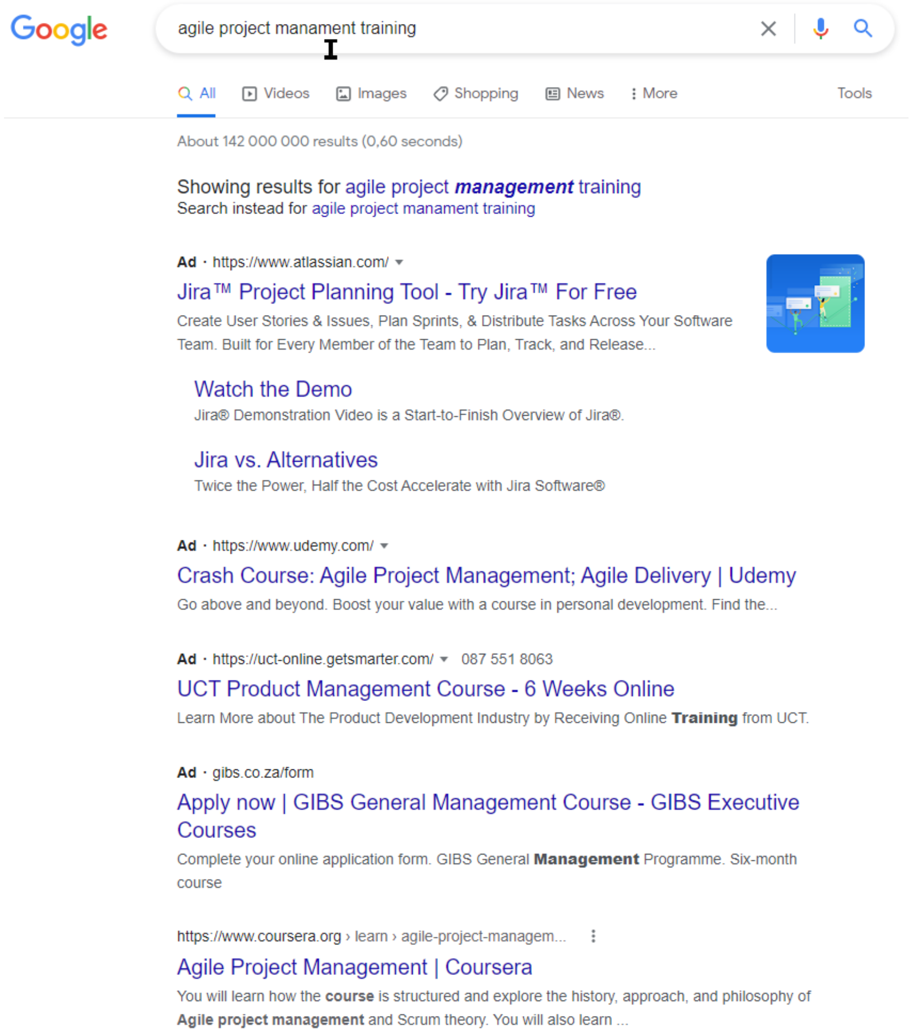
- In spite of having several meters worth of rack space consumed by guidance and methodologies on proper ways to prepare sumptuous meals, and having a certificate for attending a cooking course, I still struggle to become even a mediocre cook!
- A friend of mine once remarked that my Butter Chicken tastes exactly like the dish his mother used to make, and that he now is starting to have heartburn exactly like his father used to have!
- Methodologies are excellent for teaching, highlighting essential components and the sequence of events, like shell the eggs before adding to the flour, but you cannot teach experience, mentoring in a fail safe environment is crucial!
- Going through the motions, following instructions, religiously performing the rituals and ceremonies and hoping for the best, qualifies you to become a member of a modern methodology oriented Cargo Cult!!

https://en.wikipedia.org/wiki/Cargo_cult#:~:text=A%20cargo%20cult%20is%20an,during%20the%20Second%20World%20War.

# Study is Good, but Learning is Better

- The medical profession has a mantra in respect of Learning:
  - Study one
  - Do one
  - Teach one
- Mastery is confirmed by being able to understand, execute and teach each and every aspect of a subject or topic.
- Competence is established and maintained by continual practice and immersion in innovation
- Establish multi-disciplinary teams through collaboration and mentoring, encourage innovation, honesty, fail safely, have hack days.
- A certificate does not necessarily mean competence
  - A fool with a tool, still is a fool.

**142 000 000 Agile Project Management Study Opportunities**



agile project manament training

About 142 000 000 results (0,60 seconds)

Showing results for agile project ***management*** training
Search instead for agile project manament training

**Ad** · https://www.atlassian.com/
Jira™ Project Planning Tool - Try Jira™ For Free
Create User Stories & Issues, Plan Sprints, & Distribute Tasks Across Your Software Team. Built for Every Member of the Team to Plan, Track, and Release...

Watch the Demo
Jira® Demonstration Video is a Start-to-Finish Overview of Jira®.

Jira vs. Alternatives
Twice the Power, Half the Cost Accelerate with Jira Software®

**Ad** · https://www.udemy.com/
Crash Course: Agile Project Management; Agile Delivery | Udemy
Go above and beyond. Boost your value with a course in personal development. Find the...

**Ad** · https://uct-online.getsmarter.com/ 087 551 8063
UCT Product Management Course - 6 Weeks Online
Learn More about The Product Development Industry by Receiving Online **Training** from UCT.

**Ad** · gibs.co.za/form
Apply now | GIBS General Management Course - GIBS Executive Courses
Complete your online application form. GIBS General **Management** Programme. Six-month course

https://www.coursera.org › learn › agile-project-managem...
Agile Project Management | Coursera
You will learn how the **course** is structured and explore the history, approach, and philosophy of **Agile project management** and Scrum theory. You will also learn ...

# Rethinking Agile

A stands for Anarchism. A philosophy that is skeptical of authority and rejects all involuntary, coercive forms of hierarchy. Zero adminstratium!

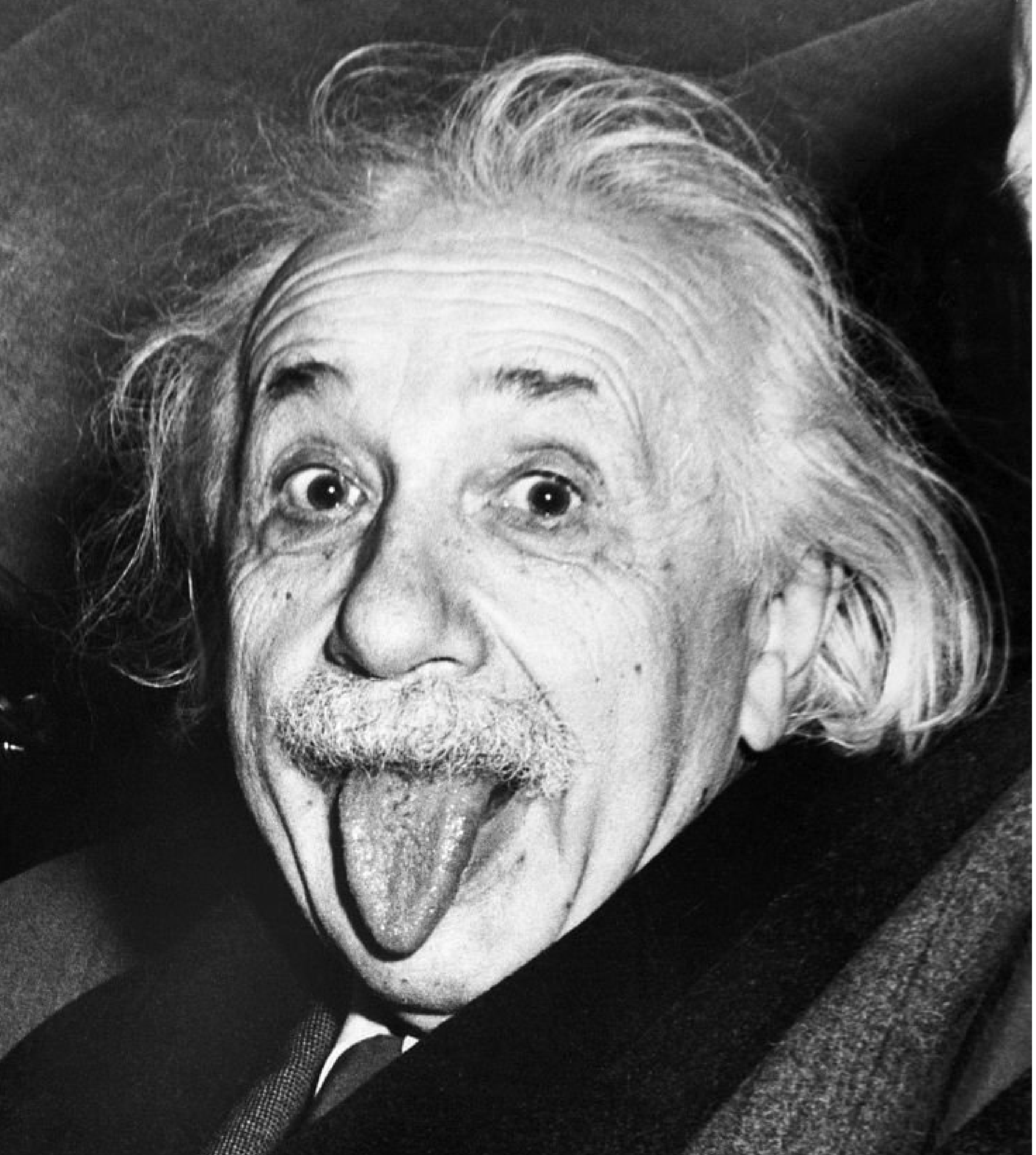
G stands for Goal-oriented. Never loose sight of the Customer’s requirements or the bigger picture

I stands for Iteration, Increment and Integration. The integration of repeated, small steps in the right direction incrementally delivers the product

L stands for Lean processes and thinking. Use the Age of Abundancy concept as enabler

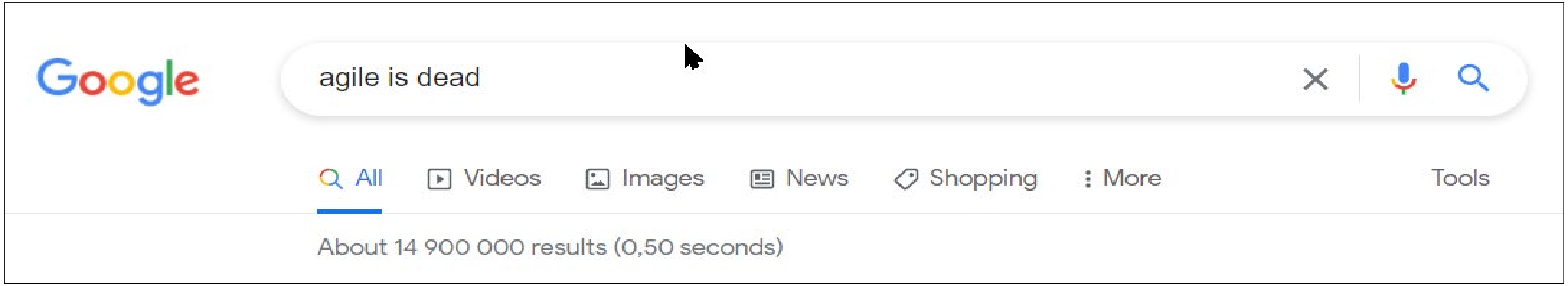
E stands for Exceptional performance, competent team members, shared values, trust, respect, autonomy, discipline

https://en.wikipedia.org/wiki/Anarchism



**The significant problems we face will not be solved by the same level of thinking that created them.**

# Thank-you, especially to Dave Thomas, Stuart and Carla



Google

agile is dead

About 14 900 000 results (0,50 seconds)

No, the Agile approach is not dead!

Agile is an adjective, in practice it is a mindset that is based on an evolutionary approach of small steps, fast feedback, continual measurement and control.

It is not a rigid methodology that sees the Universe as reducible to a set of rituals and incantations, used by “the faithful” to cast fertility spells over the Problem Space and software development efforts in particular.

(Ask the TQM, Six Sigma and Balanced Scorecard survivors who are still in therapy.)

Agile – the final frontier?

COSMOPOLIT
ONLY £2
YOUR SUMMER STYLE GUIDE*
SQL Injection: just say NO!
*Flame emojis: incoming
Mouse pads that really work
AND WON'T SLIDE OFF YOUR FACE BY 5PM (DAMN YOU, HUMIDITY!)
BOSS YOUR BREAK-UP
10 tips to ease your move from Office 365 to Google Workspace
Molly-Mae
Kanban or Scrum: Is KanRum the answer?
THE TRUTH ABOUT AGILE ORGASMS
YOUR SECRET SEX LIVES REVEALED
I only code in COBOL
YES, YOU CAN STILL GET A JOB ABROAD
Scrum
H [redacted]
This image bears no resemblance to any person or publication

# Discussion?

1N73LL1G3NC3 15 7H3 4B1L17Y
70 4D4P7 70 CH4NG3
–
573PH3N H4WK1NG