Scrum, un marco ágil de trabajo en equipo para la planificación y ejecución de proyectos.

Noviembre 2014

Enriclizondo@gmail.com
ScaledAgileCoach.com

Cosas de Scrum que quiero aprender

En 3 minutos

- Haced grupos de dos personas y comentad lo que sabéis de Scrum.
- Después, escribid en “post-its” un par de cosas que queréis aprender sobre Scrum/Agile
- Pegadlo en la pared en “Cosas de Scrum que quiero aprender”.

Cuanto sabes de Scrum?

5	Puedo ensenar esta clase, ahora mismo
4	Ya llevo al menos 5 años practicando Scrum
3	He practicado Scrum un par de años
2	He leído un par de libros
1	Que significa SCRUM?

Enriclizondo@gmail.com
ScaledAgileCoach.com

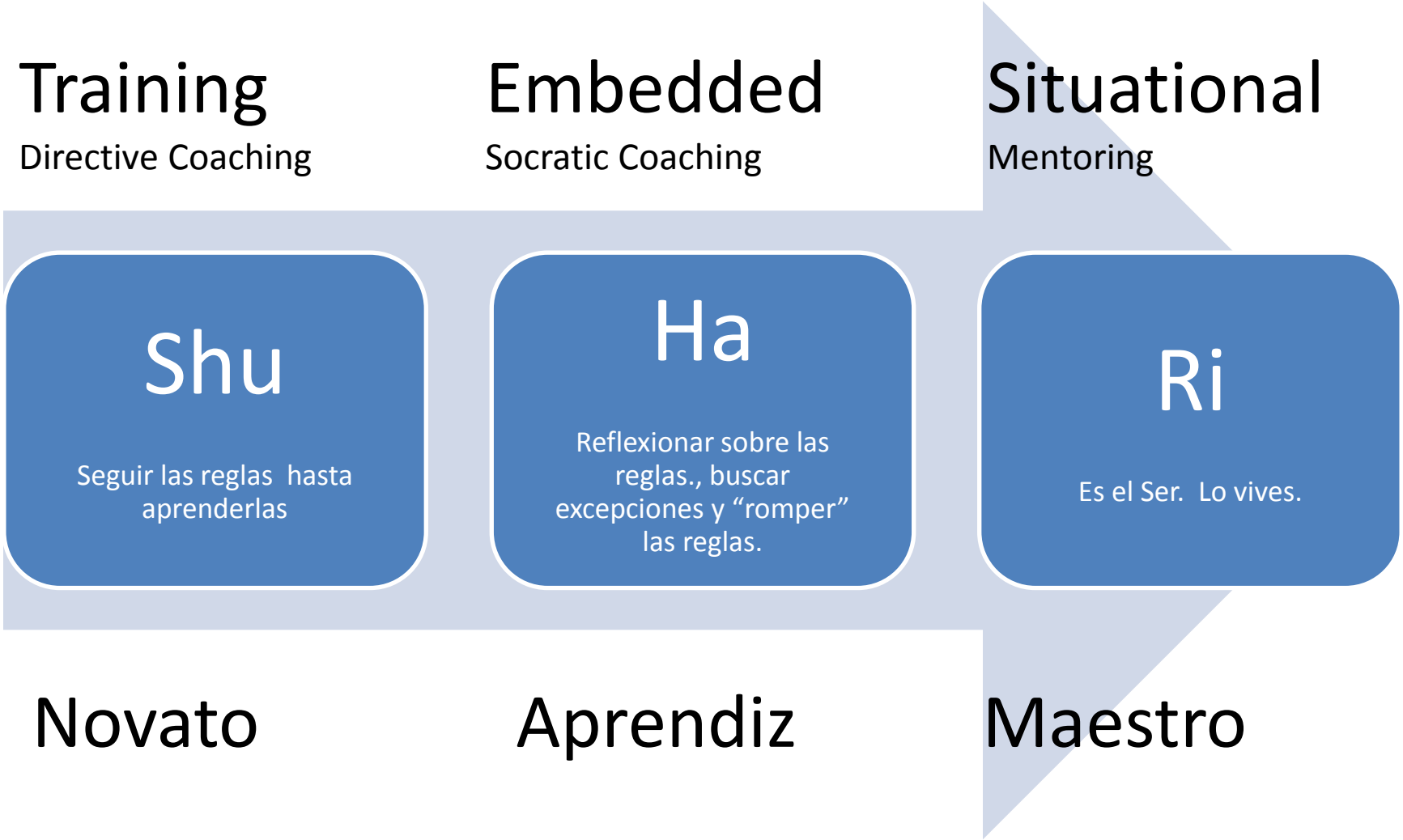
Trabajando juntos

- Teléfonos móviles y laptops
- Parking para otros temas o preguntas
- Que he aprendido
- Backlog de temas
- Intermedios
- Fotos/video
- Respeto por otros
- Mas ..?

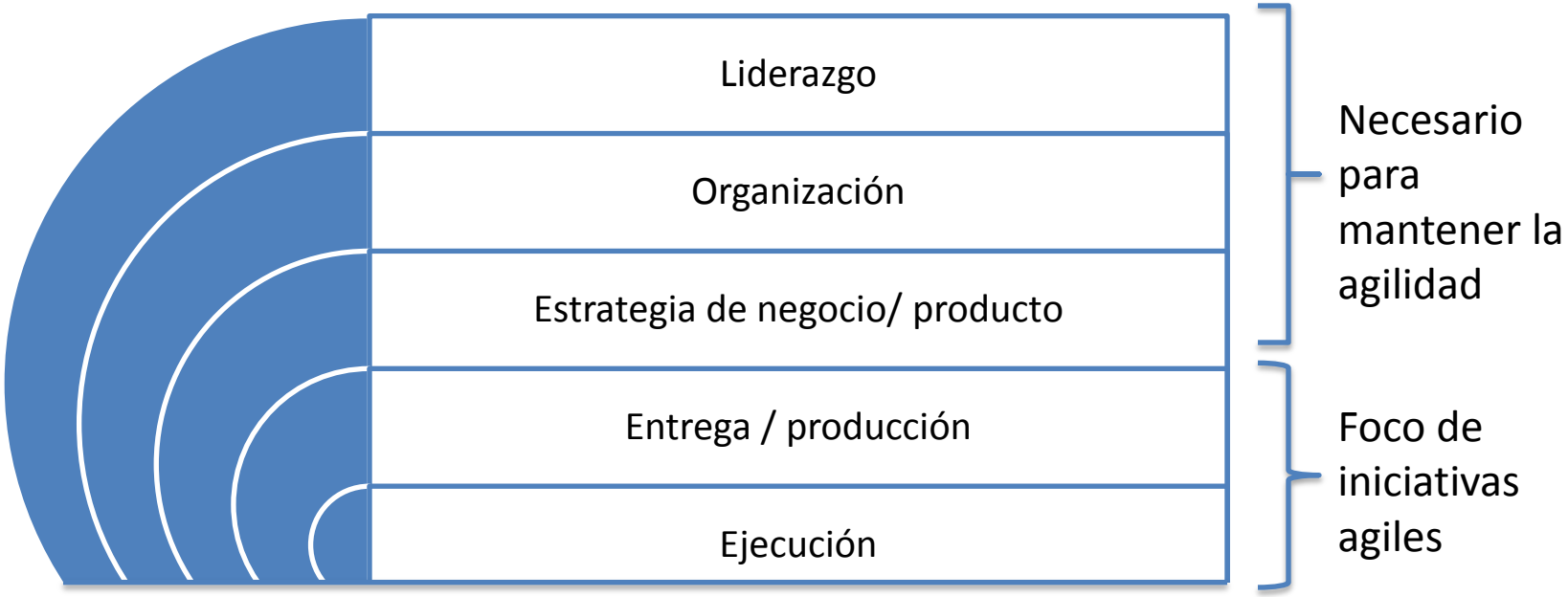
Objetivos

- Teoría de Scrum
- Filosofía Agile/Lean
- Técnicas Scrum
- Simulación de Scrum

Phases de aprendizaje



El viaje a Agile



El modelo Waterfall

Software Requirements: Are they really a problem?
--Bell, Thomas E., and T. A. Thayer. 1976

paragraphs. These requirements are subject to change due to external reasons, as the perceived threat changes, and due to internal reasons, as the developers learn more about the problems to be solved and the actual needs of the system. It is appropriate to

paragraphs. These requirements are subject to change due to external reasons, as the perceived threat changes, and due to internal reasons, as the developers learn more about the problems to be solved and the actual needs of the system. It is appropriate to

In summary, problems with requirements are frequent and important. Differences between types of requirement problems is quite small between projects. Improved techniques for developing and stating requirements are needed to deal with these problems.

35% incorrect
22% missing or incomplete
31% unclear
10% incompatible

(1956-1983) Herbert D. Benington pointing out that the process was not in fact performed in a strict top-down fashion, but depended on a prototype.

(1970) According to Winston W. Royce in the process model "the design iterations are never confined to the successive step", and for that model without iteration is "risky and invites failure". As alternative Royce proposed a more incremental development, where every next step links back to the step before.

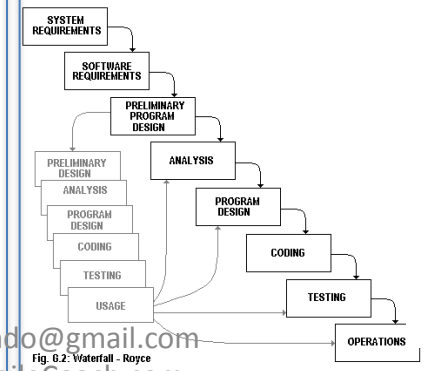


Fig. 6.2: Waterfall - Royce

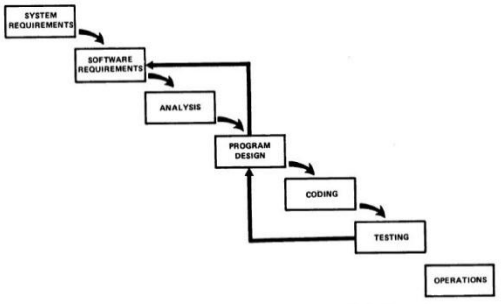


Figure 4. Unfortunately, for the process illustrated, the design iterations are never confined to the successive steps.

Enriclizondo@gmail.com
ScaledAgileCoach.com

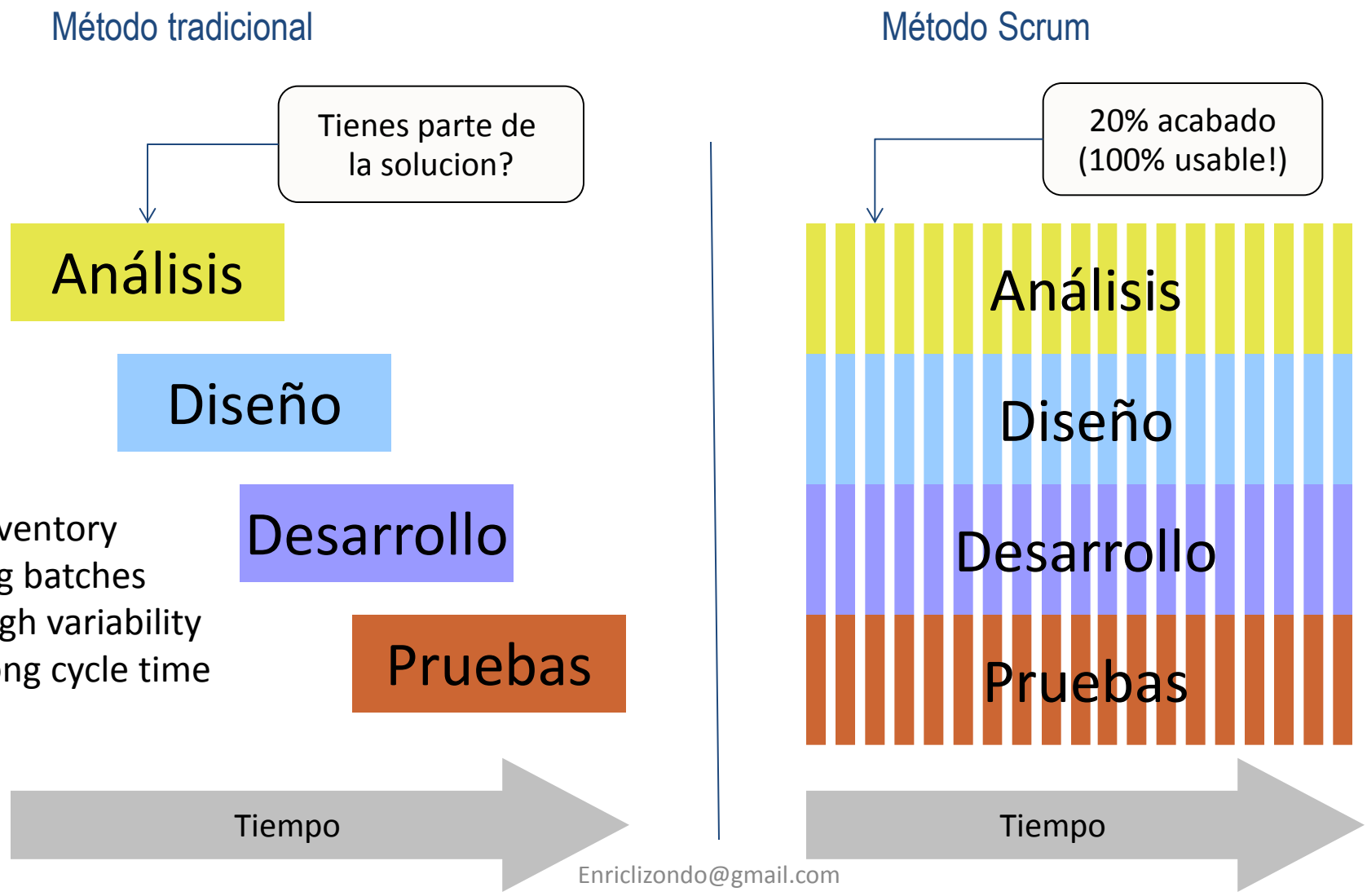
Predecible vs Adaptable

- Basado en plan / Agile
- Plan > “plan your work > work your plan”. Planea tu trabajo / Sigue el plan
- Modelo predecible
  - El éxito se mide por que tanto se ajusta la realidad al plan. En tiempo planeado y dentro del presupuesto. El proyecto tiene éxito porque hemos seguido el plan.
  - Requerimientos claros y estables
  - El plan depende de los requerimientos. El proyecto depende de los requerimientos.
  - Tener unos requerimientos estables no es la realidad.

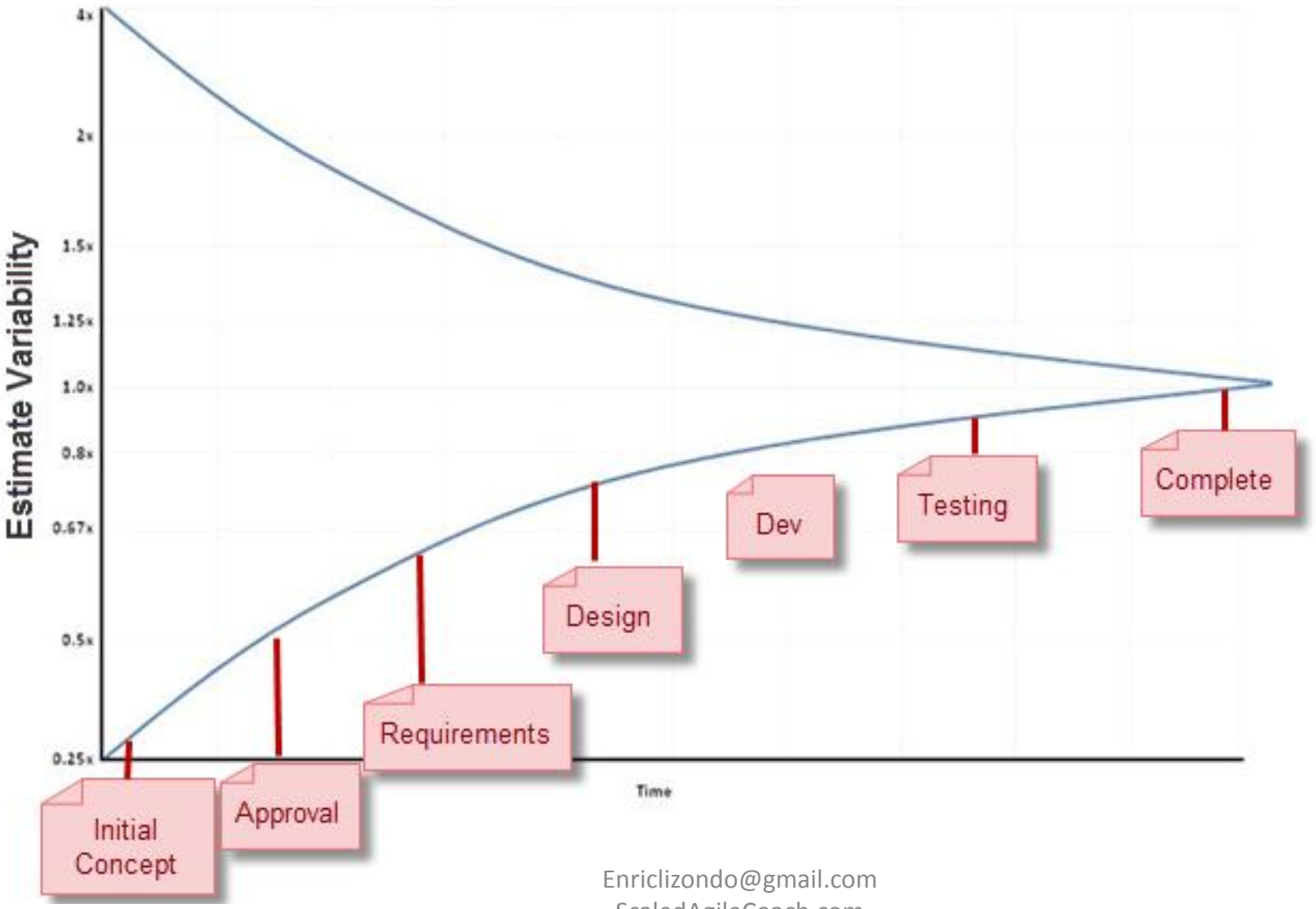
<< Rompemos la dependencia >>

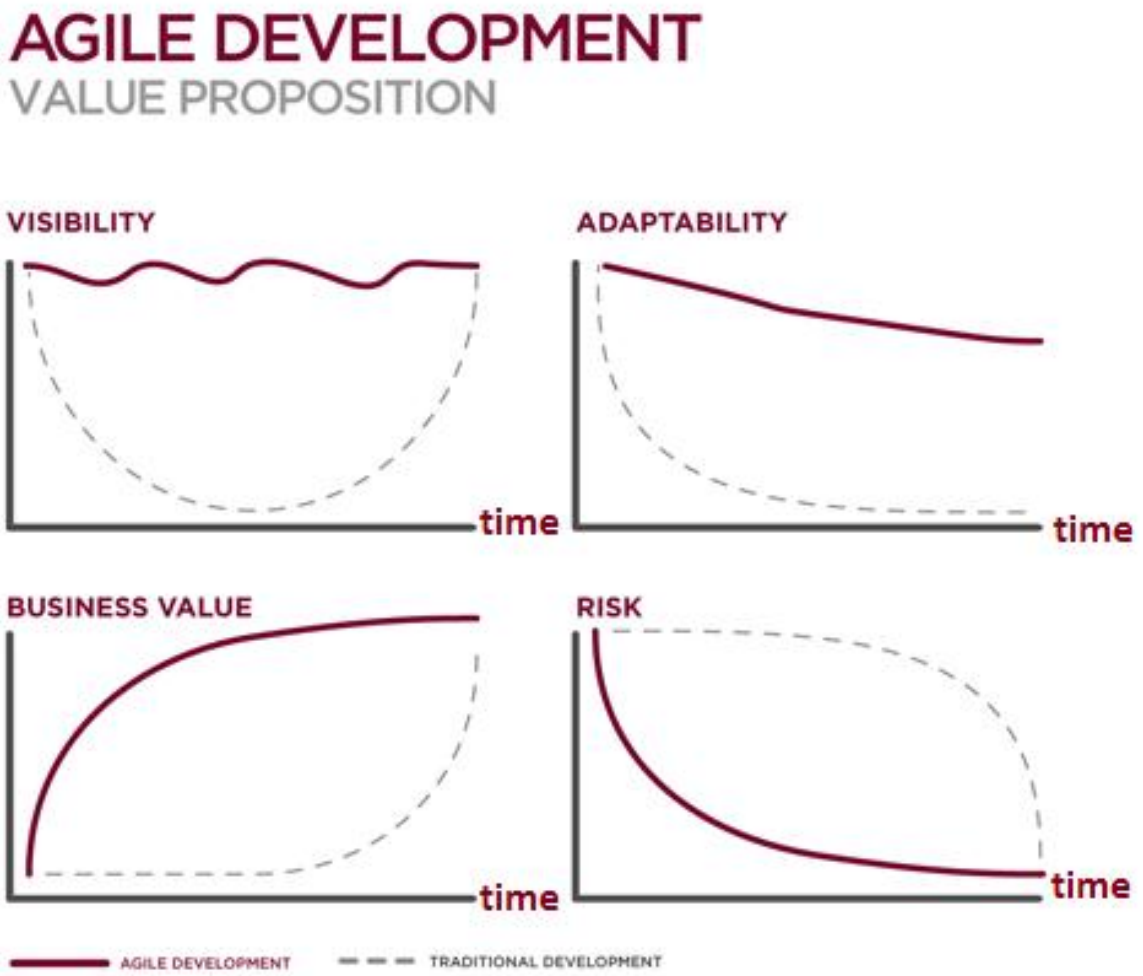
- Modelo adaptable
  - Ciclos de desarrollo y entrega
  - Cambio en el diseño de software (simple diseño, Refact, CI, XP)

Predecible (todo antes de nada) vs. Adaptable (feedback rápido)



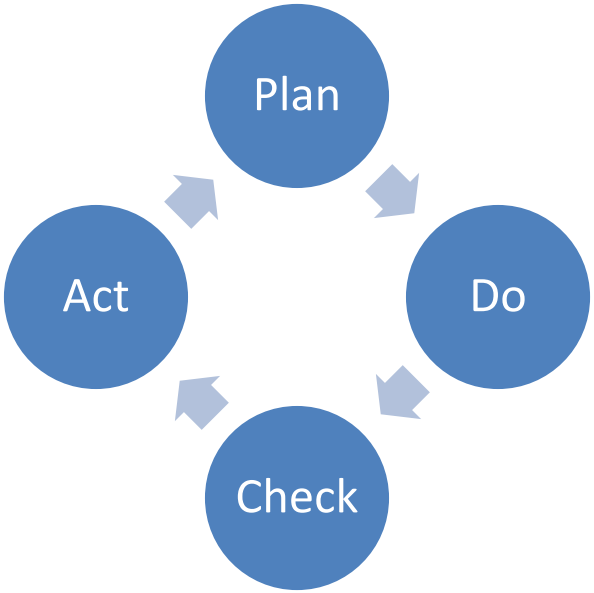
Cono de Incertidumbre

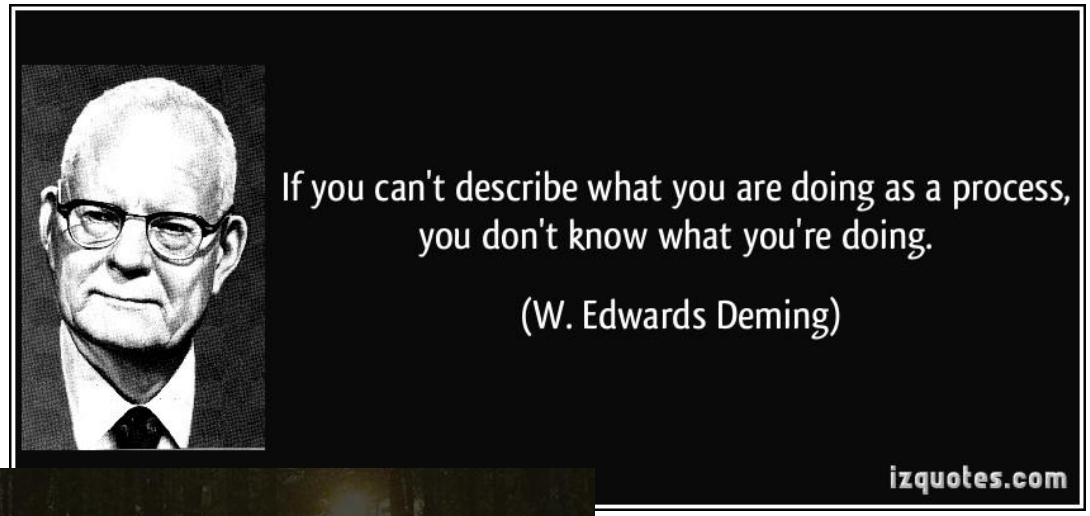


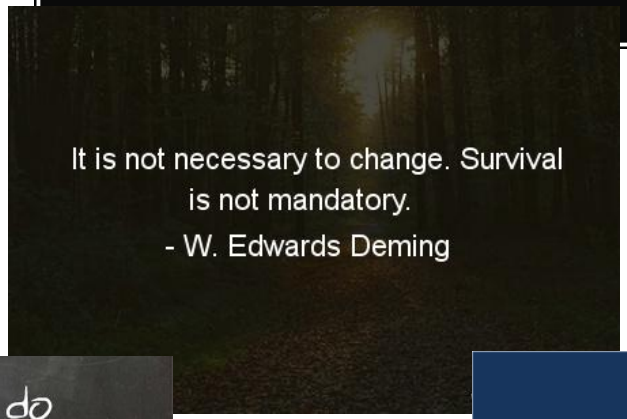


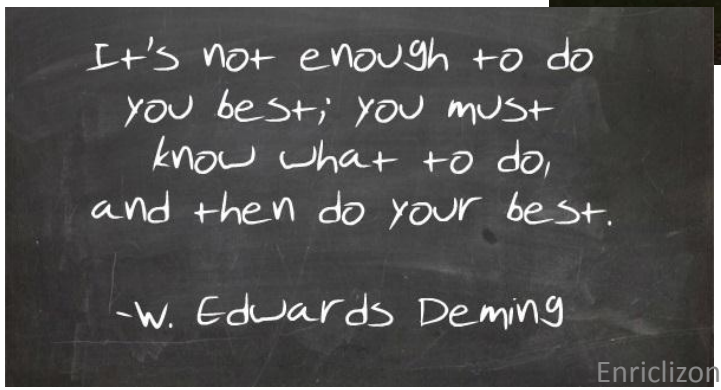
http://www.versionone.com/Agile101/Agile-Software-Development-Benefits/

Proceso empirico

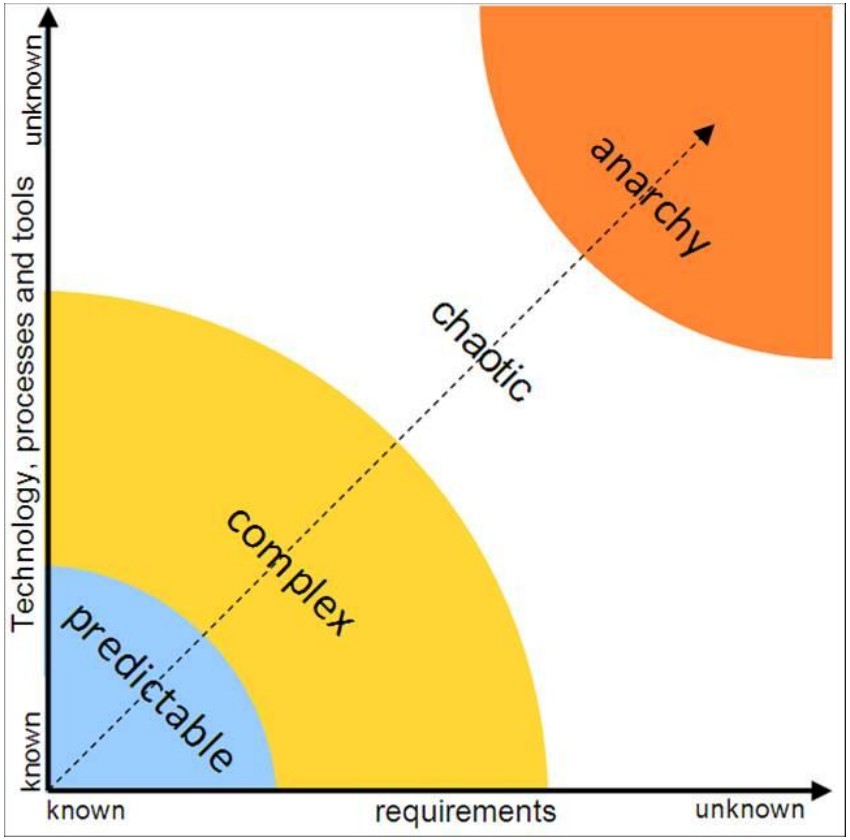






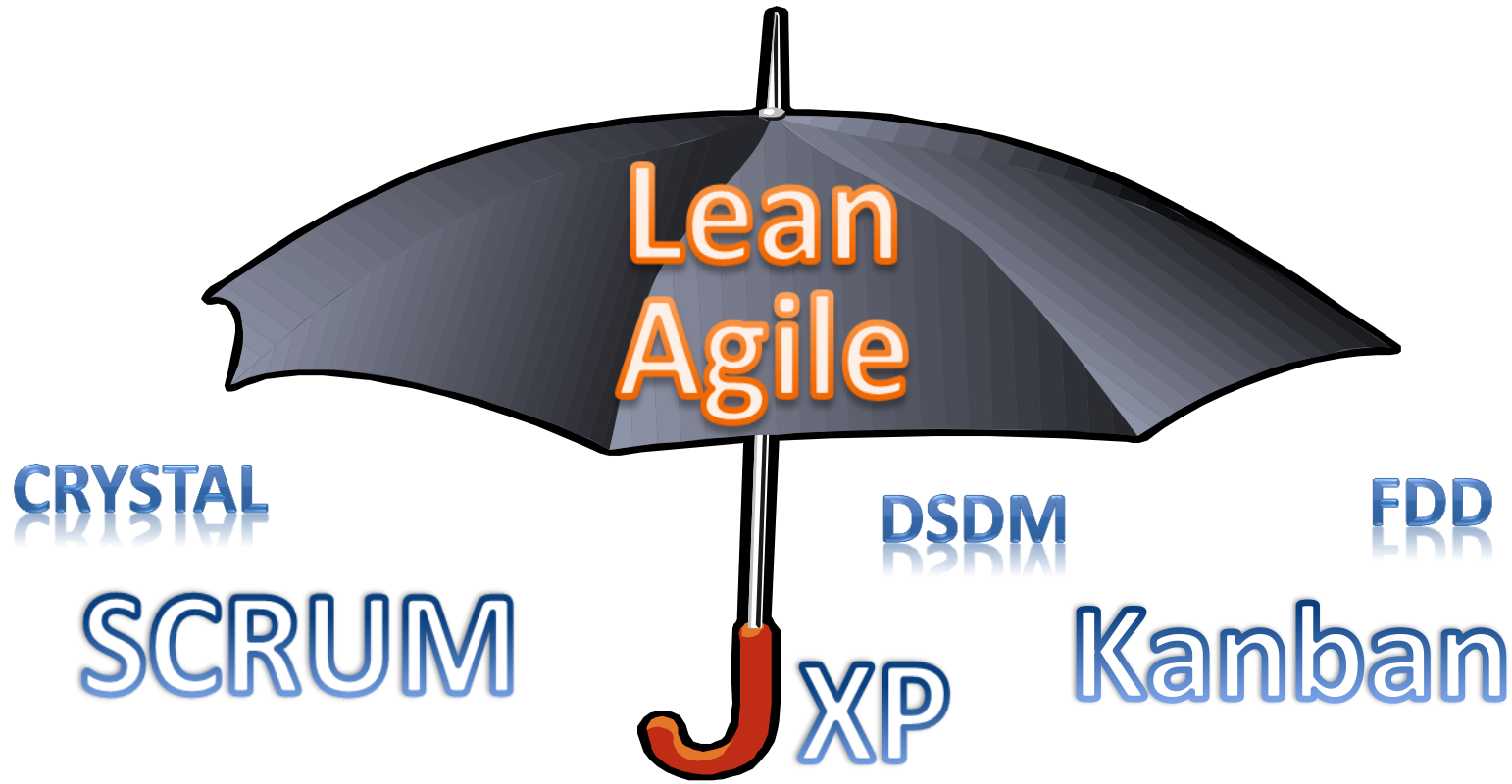


Cuando aplicar Scrum



El marco de trabajo Scrum se aplica en proyectos de alta incertidumbre por lo que se ajusta mas a proyectos complejos

Métodos Agile



Algo de historia

- < 1950 – Edwards Deming
- 1950 – TPS Taichi Ohno
- 1991 – Lean Jim Womack
- 2003 Mary Poppendieck (Lean Soft dev)
- 1986 – New new product development game (Takeuchi y Nonaka)
- 1990 - 2001 Scrum (Jeff Sutherland)
- 2001 – Agile Manifesto


Scrumpercalifragilesticexpia-leand ocious

Qué es Scrum

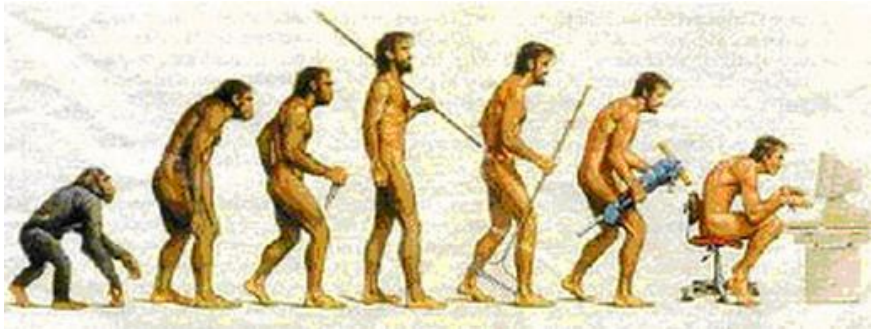
- Ligero
- Fácil de entender
- Extremadamente difícil de llegar a dominar

5 valores de Scrum

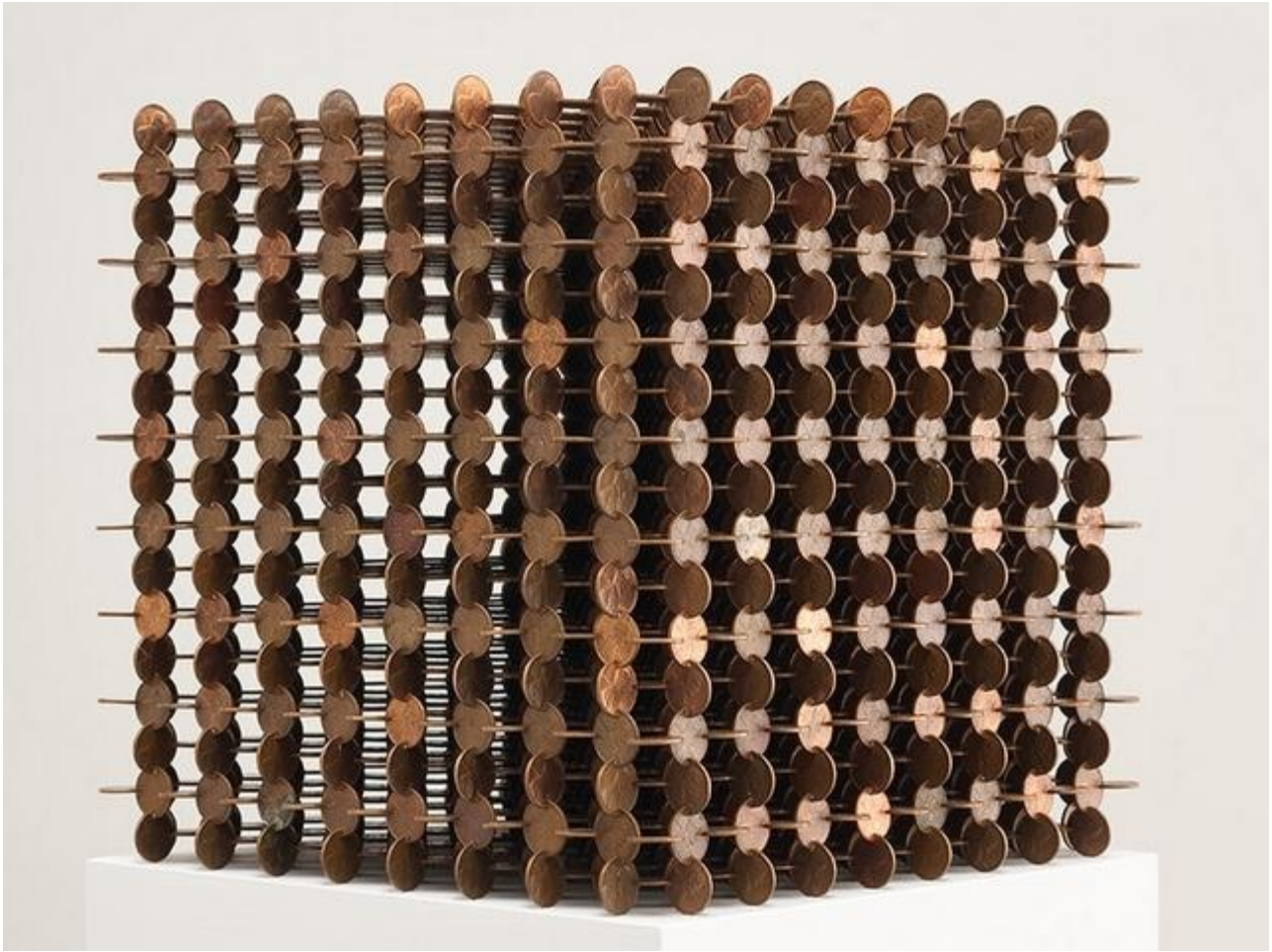
- Enfoque
- Tener valor (“Corage”)
- Transparencia (“openness”)
- Determinación (“Comittment”)
- Respeto

Los pilares de Scrum

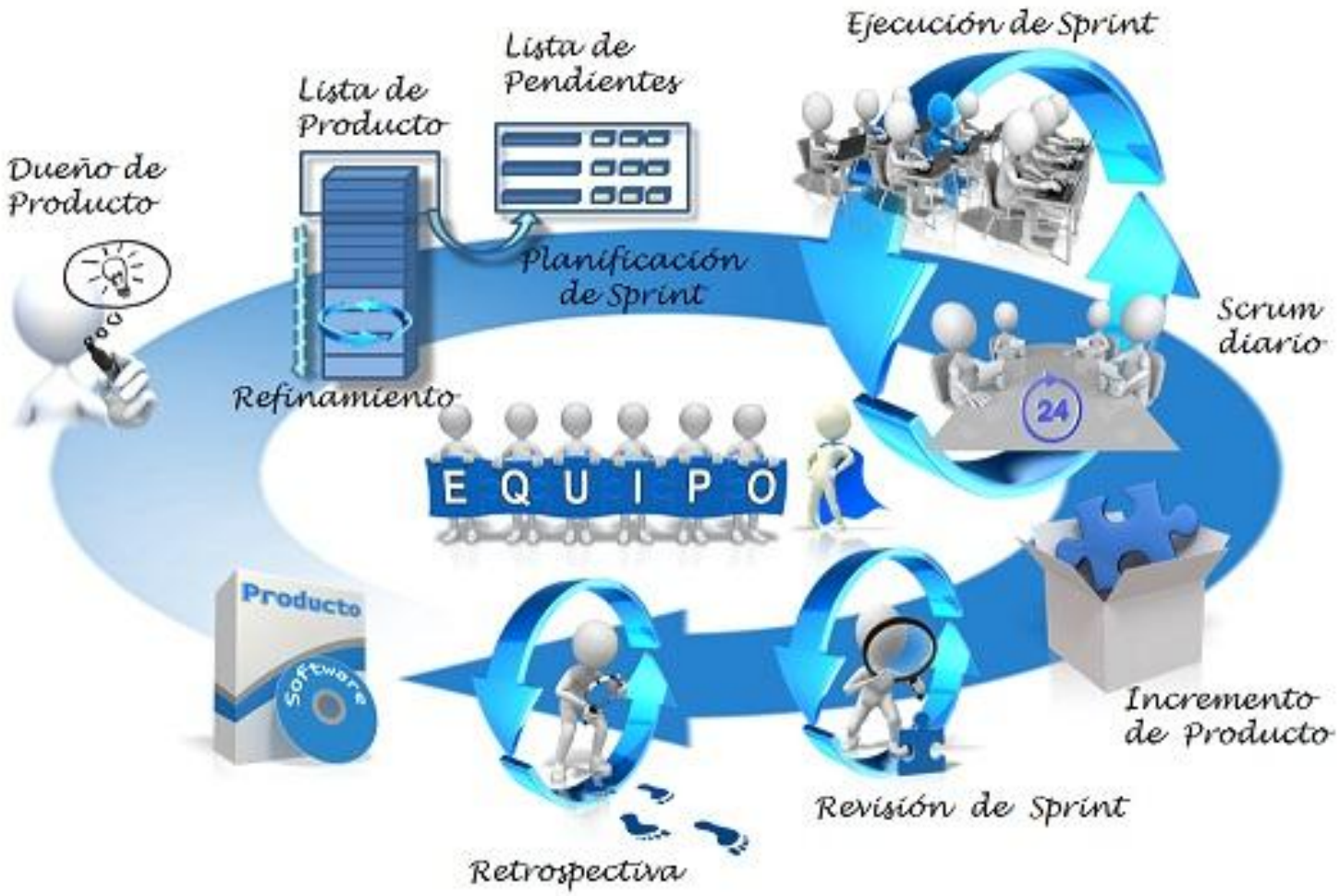
- Transparencia
- Inspección
- Adaptación



Actividad



Scrum Framework



Role: El Dueño de Producto (*Product Owner*)

- Dueño de producto
- Gestionar lista
- Expresar elementos
- Ordenar lista
- Optimizar valor
- Visibilidad
- Transparencia

Role: El Equipo de Desarrollo (*Development Team*)

- Auto organizados
- Multifuncionales
- Dedicados
- no títulos
- no sub-equipos
- responsabilidad en el Equipo

Actividad

Tuckman> Modelo de equipo

Role: El Scrum Master

- Eliminar impedimentos
- Facilitar los eventos
- Guiar al Equipo
- Entender y practicar la agilidad
- Motivar cambios
- Liderar y guiar
- Líder-servidor

Impact on Traditional Roles > Project Manager

The responsibilities of the traditional role of project management are shared between the 3 Scrum roles of PO, SM and Dev team.

- Team is self-organized team instead of control by command.
- Team self-assigns tasks and decides how to accomplish the sprint goal
- SM acts as a Servant/leader coaching, facilitating and tracking progress
- PO decides what to build and in what order

Artefacto: Lista de producto (Product Backlog)

- Requisitos
- Ordenada
- Piramidal
- Descripción
- Valor
- Orden
- Estimación

Que es un sprint?

- Bloque de tiempo
- Misma duracion
- No hay cambios
- Alcance clarificado
- No hay intervalos
- Crea un ritmo
- Reduce riesgo
- Da enfoque al equipo

Duración del sprint

- Duración del sprint: de 1 a 4 semanas
- Duración típica de un sprint: 2 semanas
- Contra las largo
  - Síndrome del estudiante
  - Ley de Parkinson
  - Tiende a ser waterfall
  - Aumento del riesgo
- Contra mas corto
  - Feedback mas rápido
  - Los problemas surgen mas más rápido
  - Disminuye el riesgo
  - Menos WIP
  - Promueve la gestión del backlog por el dueño de producto

Artefacto: Lista de pendientes del sprint (Sprint Backlog)

- Sub lista
- Objetivo sprint
- Estimación
- Cómo
- Asignación
- Capacidad vs determinación

Artefacto: Incremento (Product Increment)

- Suma elementos
- Listo para liberar
- Terminado
- Muestra progreso

Artefacto: Definición de “terminado” (DoD)

Evento: Refinamiento (Refinement or Grooming)

- Orden
- Detalles
- Estimación
- 10%
- Continuo

Evento: Planificación de Sprint (Sprint Planning)

- Bloque de tiempo (4h)
- Qué
- Cómo
- Compromiso

Evento: Scrum Diario (Daily Scrum)

- 15min cada 24horas
- Qué hice ayer
- Qué haré hoy
- Impedimentos para el objetivo
- I/A

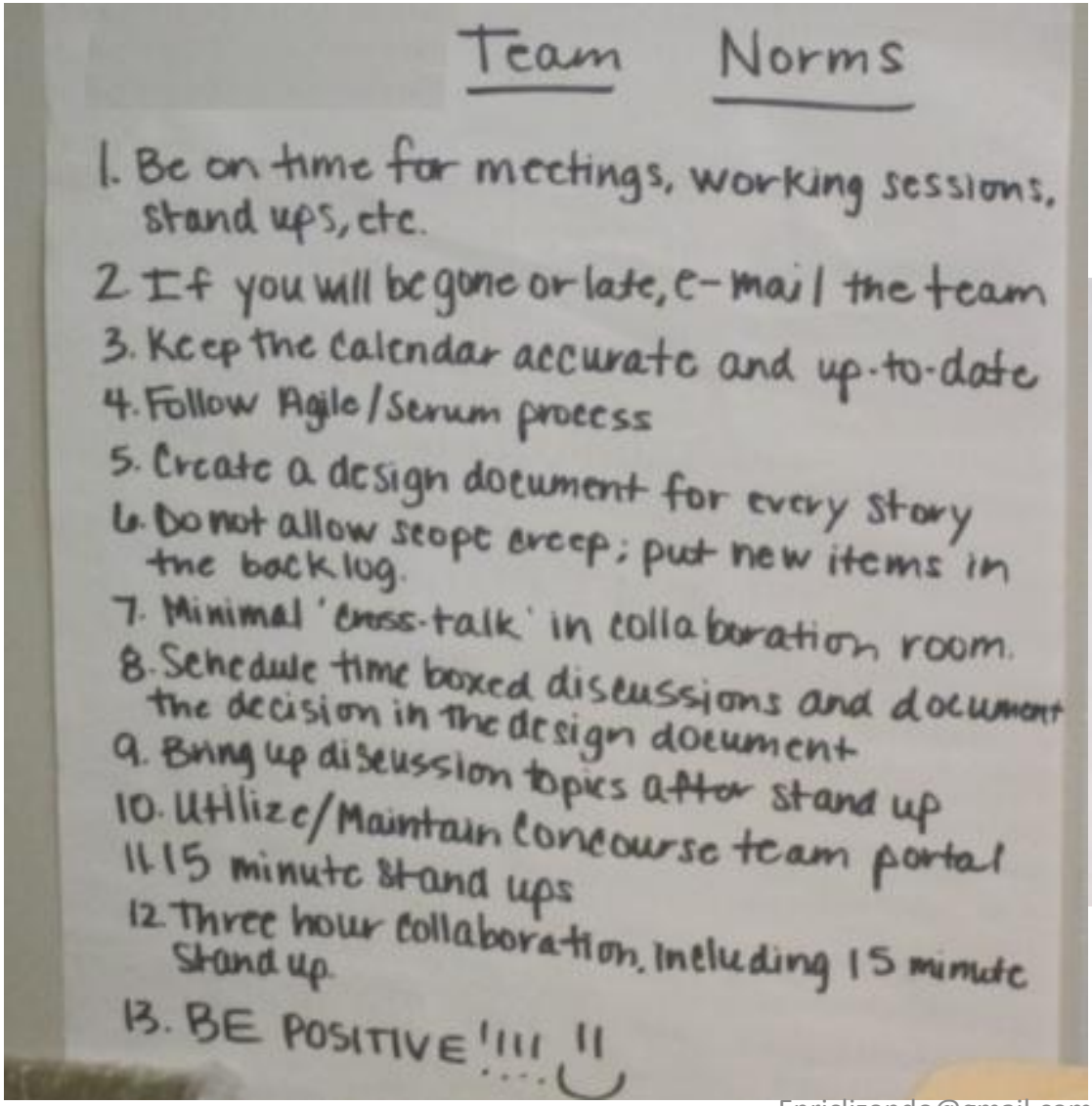
Evento: Revisión de Sprint (Review)

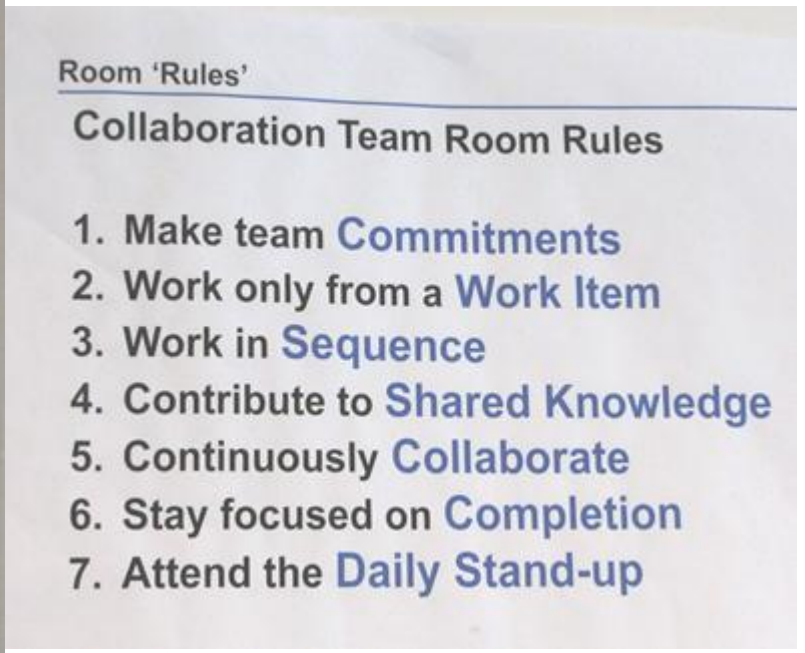
- Inspección del incremento
- Adaptación lista producto
- Demo

Evento: Retrospectiva de Sprint (Retro)

- Inspeccionar personas, relaciones, procesos y herramientas
- Identificar, ordenar, analizar y crear plan de acción

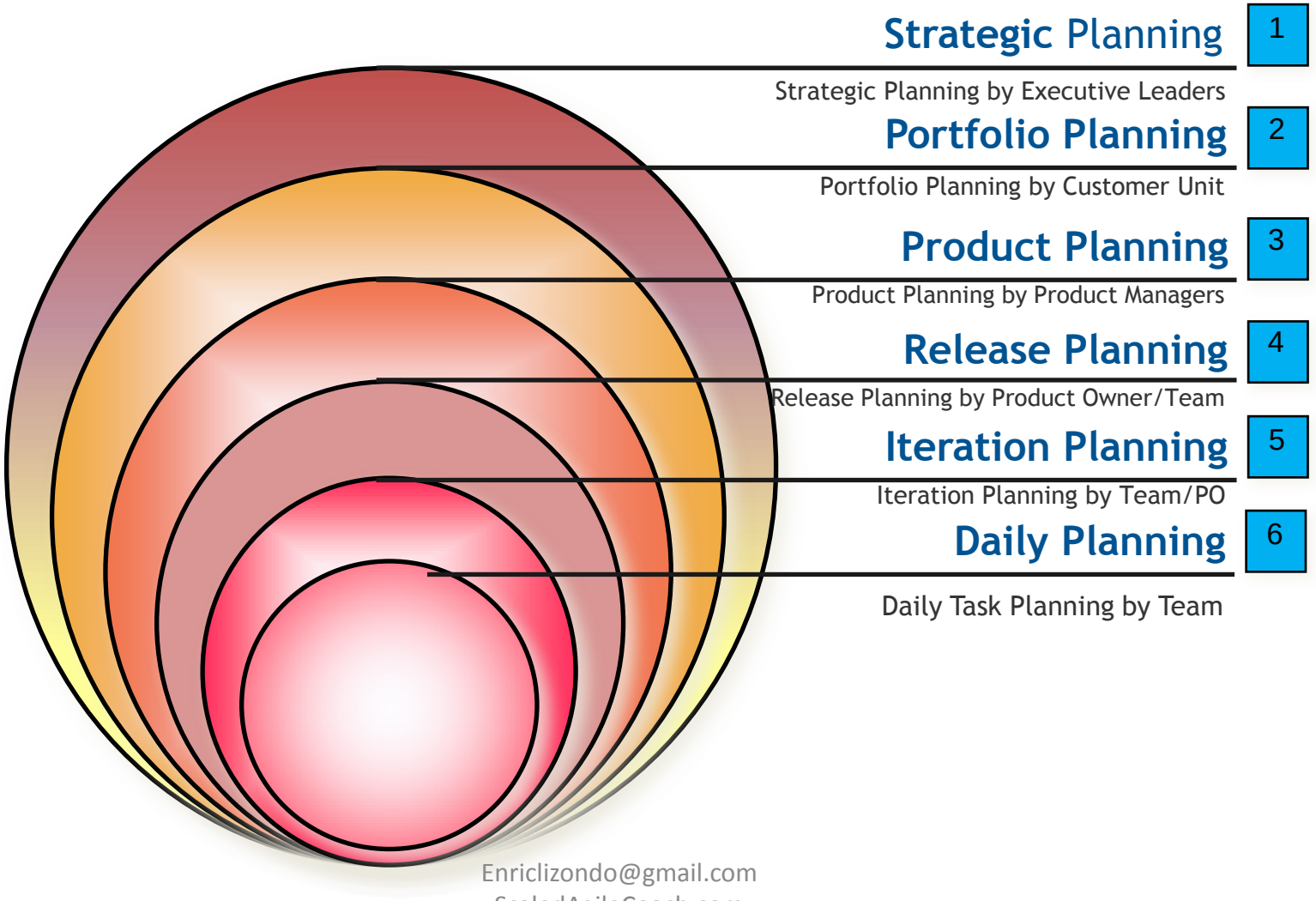
Team Norms


Team Norms
1. Be on time for meetings, working sessions, stand ups, etc.
2. If you will be gone or late, e-mail the team
3. Keep the calendar accurate and up-to-date
4. Follow Agile/Scrum process
5. Create a design document for every story
6. Do not allow scope creep; put new items in the backlog.
7. Minimal 'cross-talk' in collaboration room.
8. Schedule time boxed discussions and document the decision in the design document
9. Bring up discussion topics after stand up
10. Utilize/Maintain Concourse team portal
11. 15 minute stand ups
12. Three hour collaboration, including 15 minute stand up
13. BE POSITIVE!!!


Room 'Rules'
Collaboration Team Room Rules
1. Make team Commitments
2. Work only from a Work Item
3. Work in Sequence
4. Contribute to Shared Knowledge
5. Continuously Collaborate
6. Stay focused on Completion
7. Attend the Daily Stand-up

Planificación

Niveles de planificación



Product Visioning

Before you can develop the Product Backlog, you need to create a shared Vision that identifies a clear goal and objective.

The Vision should outline a summary of the Goal, Scope and Purpose behind it.

Product Vision Statement (Elevator Statement)

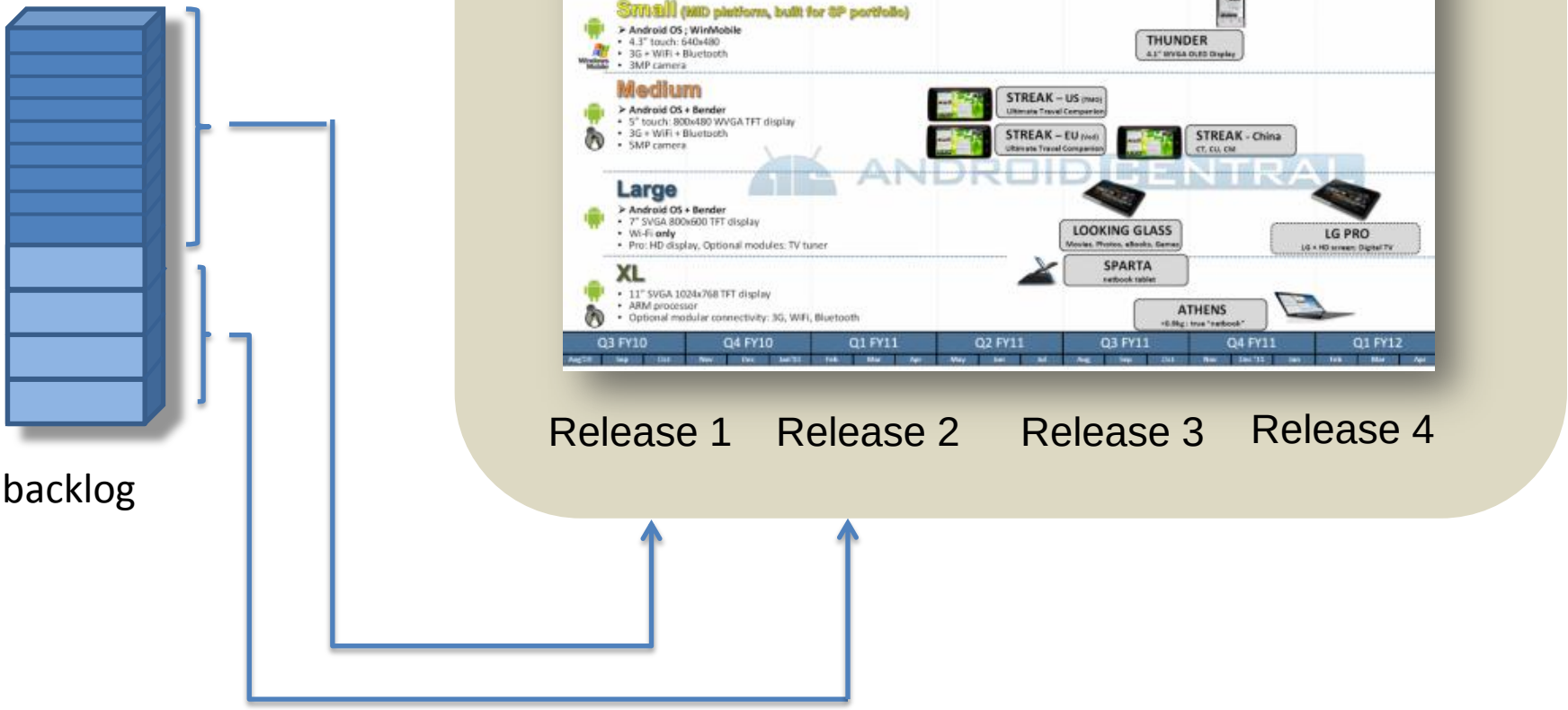
"For a mid-sized company's marketing and sales departments who need basic CRM functionality, the CRM-Innovator is a Web-based service that provides sales tracking, lead generation, and sales representative support features that improve customer relationships at critical touch points. Unlike other services or package software products, our product provides very capable services at a moderate cost."

- For (target customer)
- Who (statement of the need or opportunity)
- The (product name) is a (product category)
- That (key benefit, compelling reason to buy)
- Unlike (primary competitive alternative)
- Our product (statement of primary differentiation)

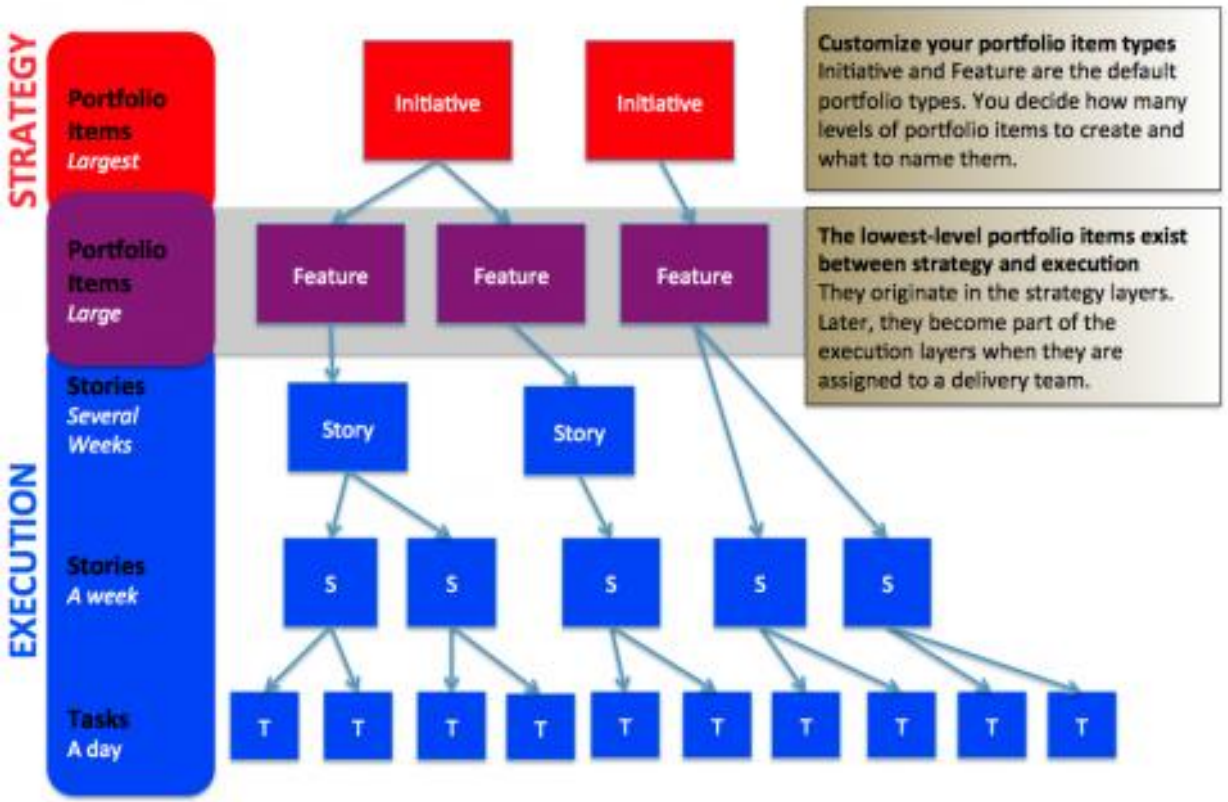
Geoffrey Moore's book Crossing the Chasm.

Cutter Summit Jim Highsmith

Release Roadmap Releases



From strategy to execution



https://help.rallydev.com/getting-started-rally-portfolio-manager

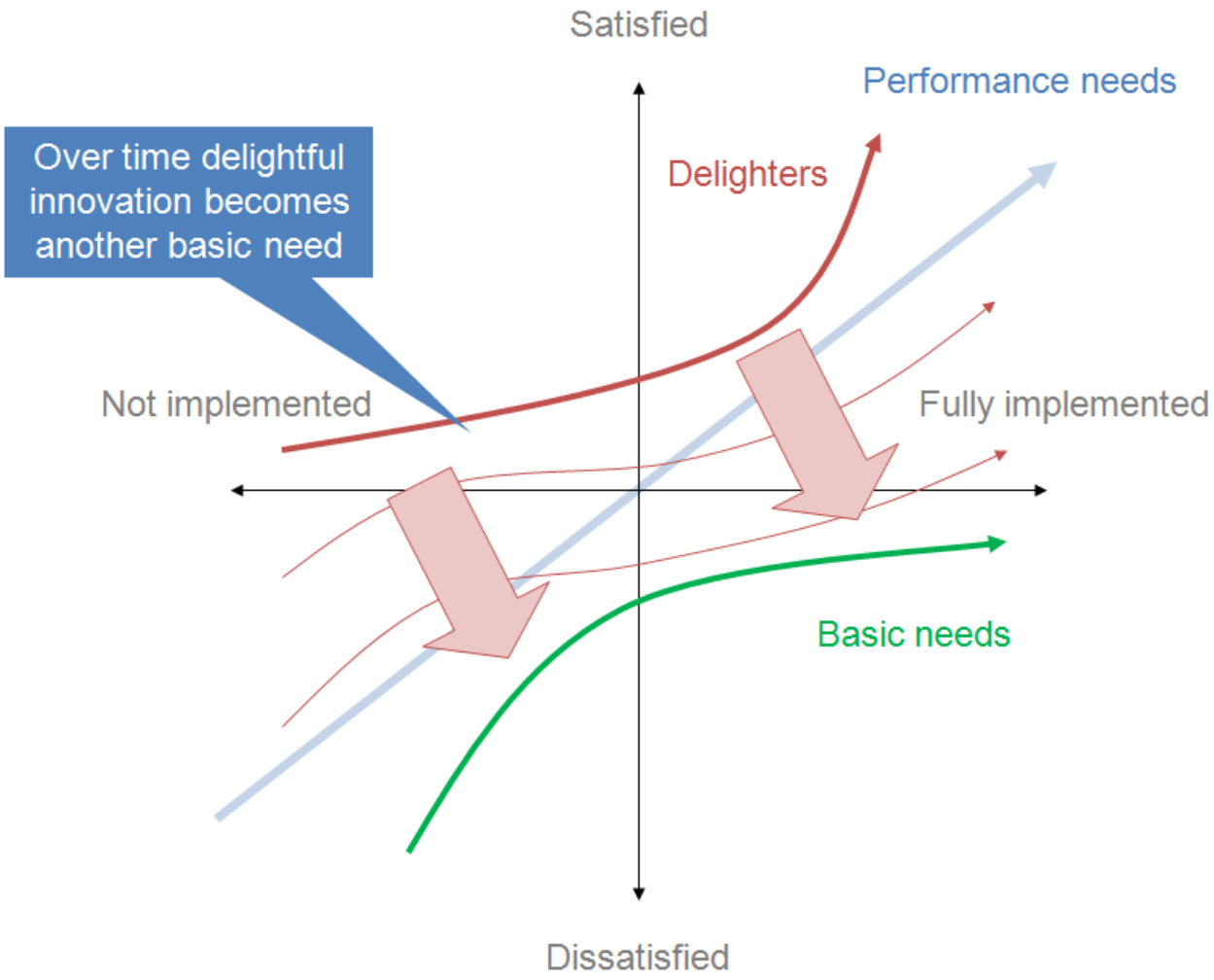
Priorizar lista producto

Minimum Viable Feature

- The smallest set of functionality that provides value to your market.
- Consists of one or more backlog work items
- Focus on Competitive differentiation,
- Earlier revenue generation / cost savings

"Perfection is achieved, not when there is nothing more to add, but when there is nothing left to take away."
Antoine de Saint-Exupery (1900 - 1944)

Kano model Prioritization Technique



http://www.infoq.com/presentations/prioritizing-your-product-backlog-mike-cohn

Kano model Prioritization Technique

Functional Question \ Dysfunctional Question	Like	Expect	Neutral	Live with	Dislike
Like	Q	E	E	E	L
Expect	R	I	I	I	M
Neutral	R	I	I	I	M
Live with	R	I	I	I	M
Dislike	R	R	R	R	Q

M = Mandatory
L = Linear
E = Exciter
Q = Questionable
R = Reverse
I = Indifferent

We ask to questions:

- How do you feel if a feature is present?
- How do you feel if that feature is absent?

Answers:
I like it that way
I expect it to be that way
I am neutral
I can live with it that way
I dislike it that way

Theme scoring: an example

Selection Criteria	Weight	Theme A		Theme B		Theme C	
		Rating	Weighted Score	Rating	Weighted Score	Rating	Weighted Score
Importance to existing cust.	.25	3	0.75	1	0.25	4	1.00
Competitive. with ABC	.10	2	0.20	3	0.30	3	0.30
Starts us integrating...	.15	3	0.45	4	0.60	4	0.60
Generates Q2 revenue	.50	5	2.50	2	1.00	3	1.50
Net score			3.90		2.15		3.40
Rank			1		3		2
Continue?			Yes		No		Yes

Cómo escribir requerimientos

Historias de usuario > User Stories

As a <user> I want to <goal> so that <reason/benefit>

Como <rol> necesito/deseo <funcionalidad> con la finalidad de <razón>

Cómo escribir historias?

- Card
- Conversation
- Confirmation

INVEST > INVEPP

- Independiente (Independent)
- Negociable (Negotiable)
- Valiosa (Valuable)
- Estimable (Estimable)
- Pequeño (Small)
- Testeable (Testable)

Cómo descomponer requerimientos

Cómo partir historias?

- CRUD (CLAB)
- Flujo
- Simple/compleja
- Reglas de negocio
- Volumen de datos

Cómo estimar: De proyectos > tareas

Estimación

Velocidad

Actividad

- Rinoceronte
- Mono
- Elefante
- León
- Tigre
- Oso

- Perro
- Jirafa
- Gorila
- Ardilla
- Ratón
- Caballo

Agile: Manifiesto y Principios

Manifiesto por el Desarrollo Ágil de Software

Estamos descubriendo formas mejores de desarrollar software tanto por nuestra propia experiencia como ayudando a terceros. A través de este trabajo hemos aprendido a valorar:

Individuos e interacciones sobre procesos y herramientas

Software funcionando sobre documentación extensiva

Colaboración con el cliente sobre negociación contractual

Respuesta ante el cambio sobre seguir un plan

Esto es, aunque valoramos los elementos de la derecha, valoramos más los de la izquierda.

Principios de Agile

1. Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor.
2. Aceptamos que los requisitos cambien, incluso en etapas tardías del desarrollo. Los procesos Ágiles aprovechan el cambio para proporcionar ventaja competitiva al cliente.
3. Entregamos software funcional frecuentemente, entre dos semanas y dos meses, con preferencia al periodo de tiempo más corto posible.
4. Los responsables de negocio y los desarrolladores trabajamos juntos de forma cotidiana durante todo el proyecto.
5. Los proyectos se desarrollan en torno a individuos motivados. Hay que darles el entorno y el apoyo que necesitan, y confiarles la ejecución del trabajo.
6. El método más eficiente y efectivo de comunicar información al equipo de desarrollo y entre sus miembros es la conversación cara a cara.
7. El software funcionando es la medida principal de progreso.
8. Los procesos Ágiles promueven el desarrollo sostenible. Los promotores, desarrolladores y usuarios debemos ser capaces de mantener un ritmo constante de forma indefinida.
9. La atención continua a la excelencia técnica y al buen diseño mejora la Agilidad.
10. La simplicidad, o el arte de maximizar la cantidad de trabajo no realizado, es esencial.
11. Las mejores arquitecturas, requisitos y diseños emergen de equipos auto-organizados.
12. A intervalos regulares el equipo reflexiona sobre cómo ser más efectivo para a continuación ajustar y perfeccionar su comportamiento en consecuencia.

Vision global
Systems view

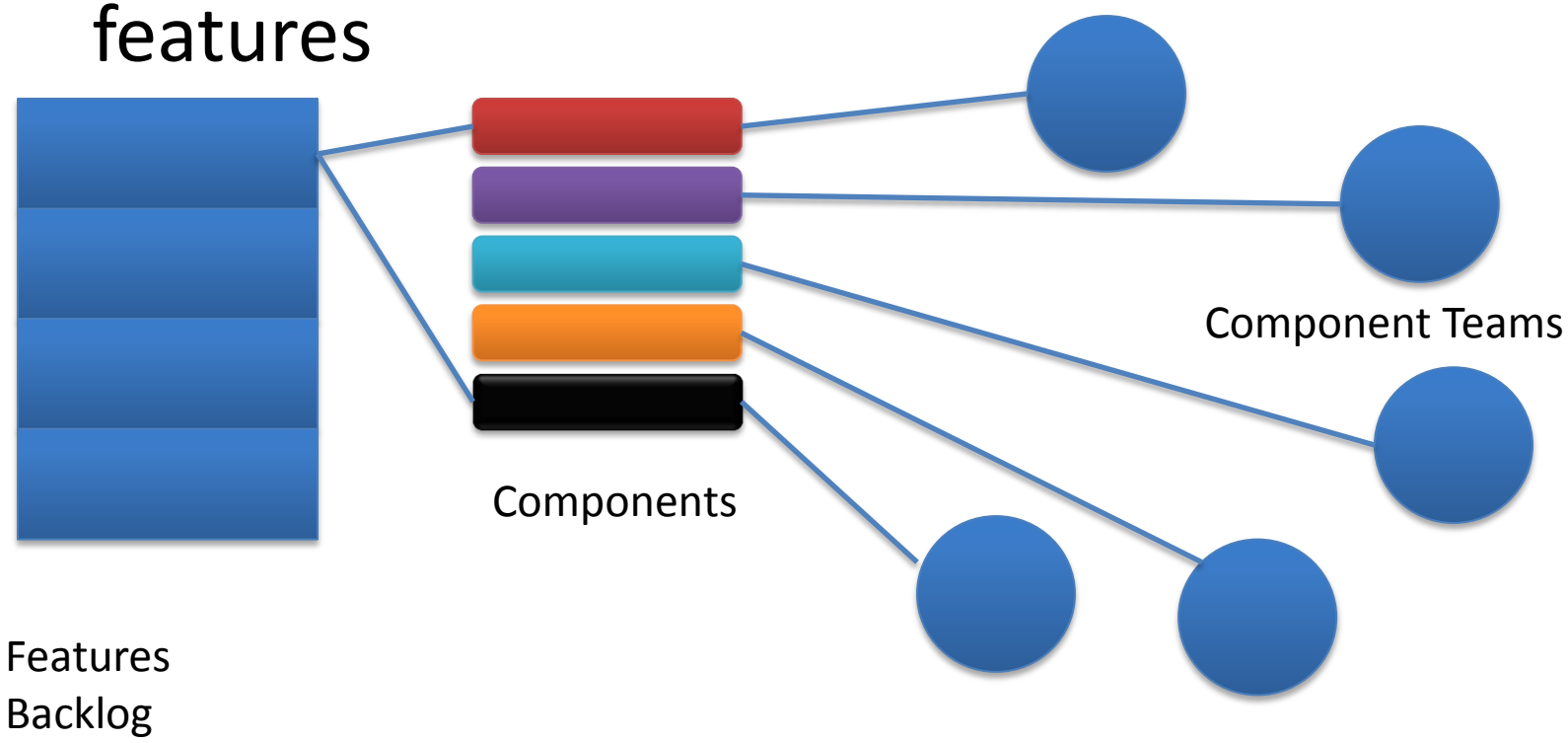
Lean Thinking

- Un modelo de gestión enfocado a la creación de flujo para poder entregar el máximo valor para los clientes, utilizando para ello los mínimos recursos necesarios: es decir *ajustados*
  - Máximo valor
  - Mínimo tiempo de espera
  - Alta calidad

Cómo formar equipos de desarrollo

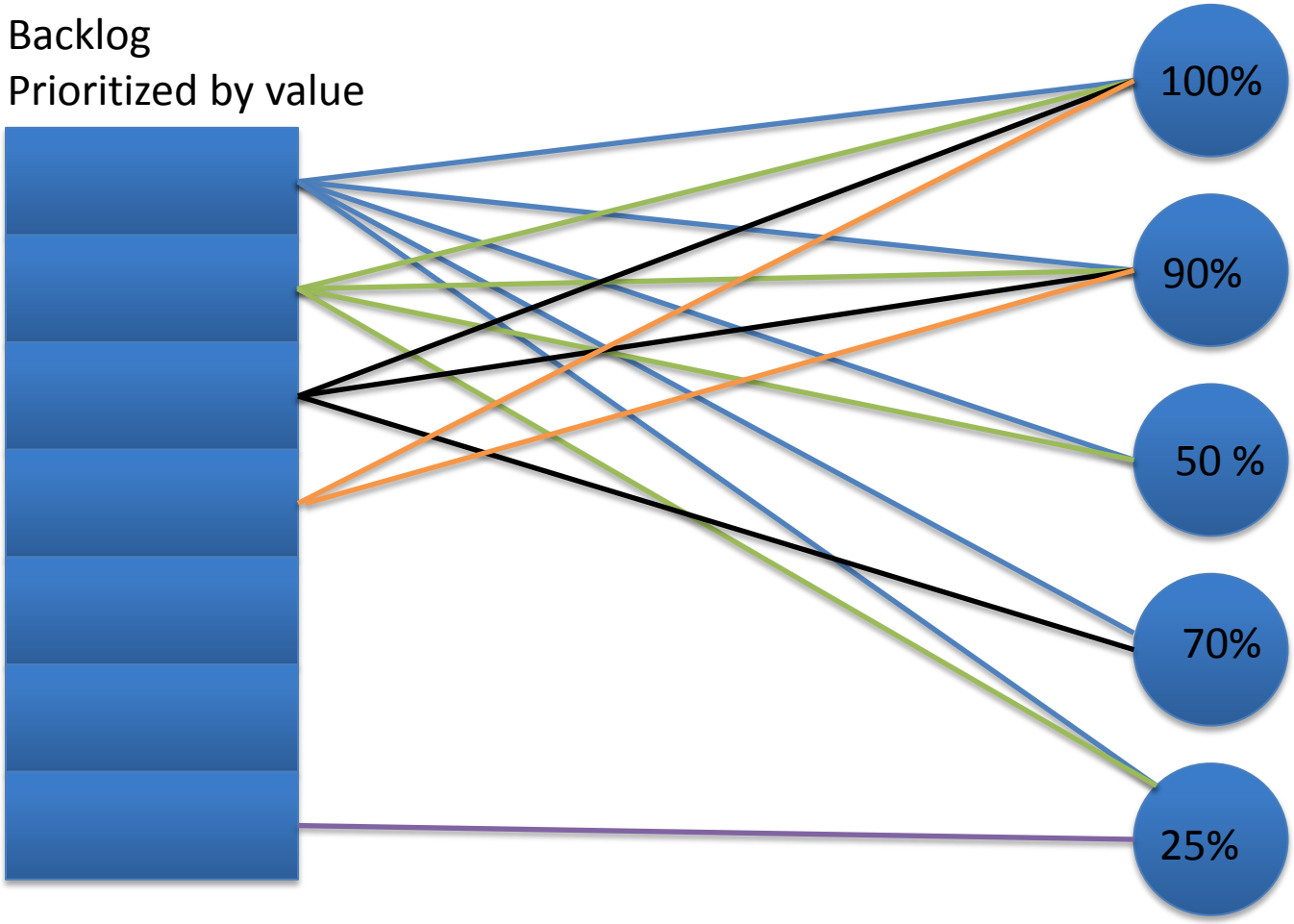
Component vs. feature teams

- Divide work by function, components or features



Component vs. feature teams

Features
Backlog
Prioritized by value

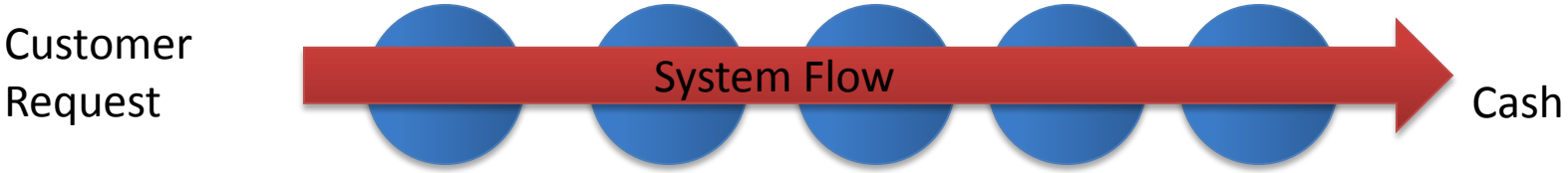


Team
Backlog
Prioritized by components

Creates inventory, long queues, increase CoD, slows down the system

Optimizing components sub optimizes the system

System View



- Value flows through the system
- Systems based on value streams

House of Lean

Actividad

Round 1: (WIP=10)

1. Fold all 10 sheets to fit envelops
2. Insert all 10 folded sheets in envelop (one per envelop)
3. Seal all 10 envelopes
4. Place 10 stamps, one per envelop

Round 2: (WIP=1)

1. Fold one sheet to fit envelop
2. Insert the folded sheet in envelope
3. Seal the envelope
4. Place a stamp
5. Repeat 10 times

Co-located Teams

- Richer communication
- Better decision making
- Sense of team
- More engagement
- Flow of information
- Knowledge sharing

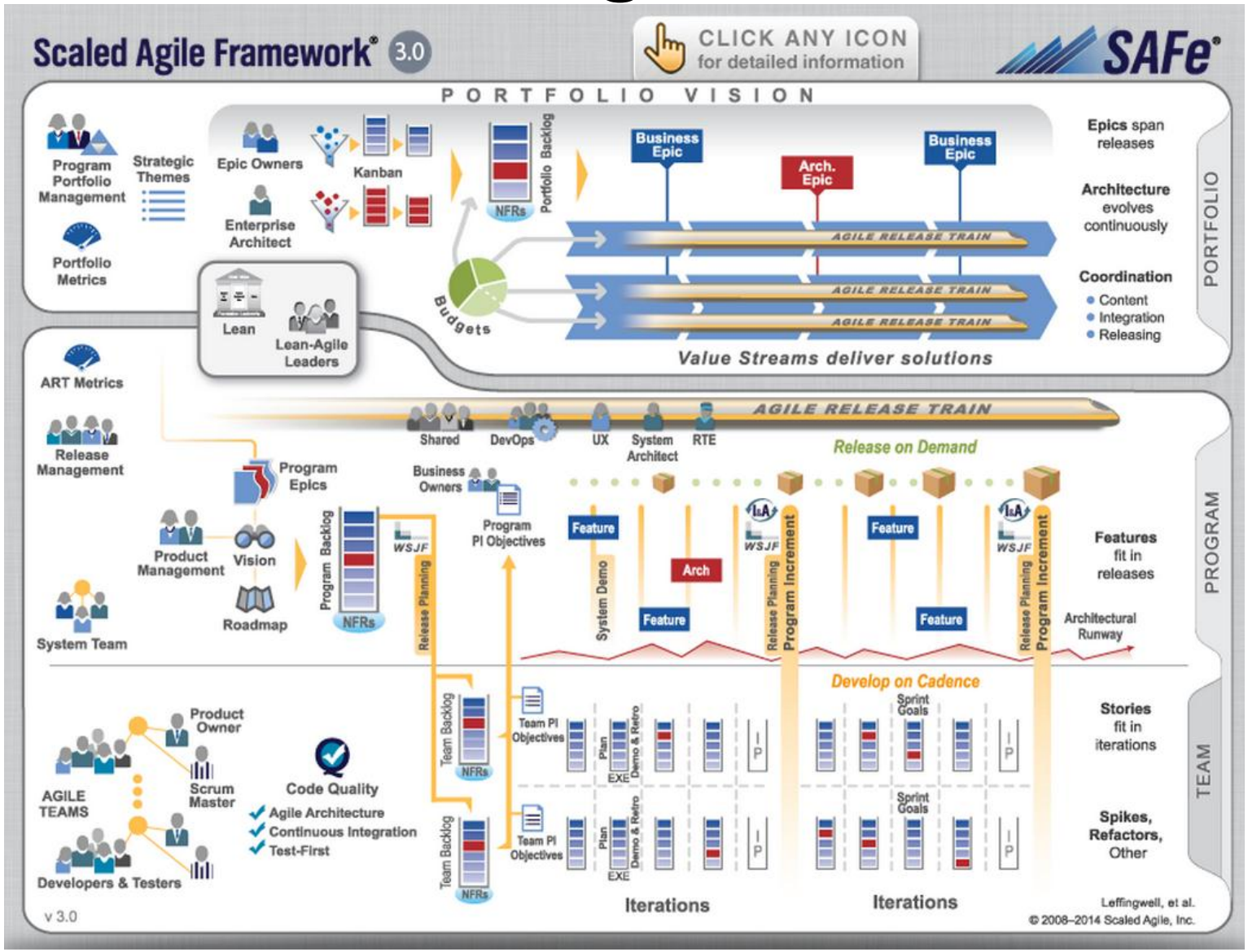
Osmotic Communication

- Osmotic communication – unconsciously picking up information around you.
- More knowledge sharing.
- Everyone on the team engaged subconsciously.

Information Radiators

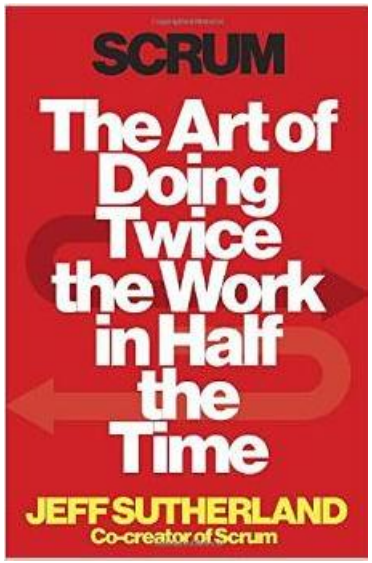
- Story Board – The stories for this iteration, what is complete, in-progress, and not started
- Purpose and Vision
- High level plan, Mid Level Plan – The overall project milestones, and the key iterations and release dates, with the anticipated objectives or deliverables for each
- Risks - The whiteboard list of risks, impact, and mitigation that needs to be taken (with owner)

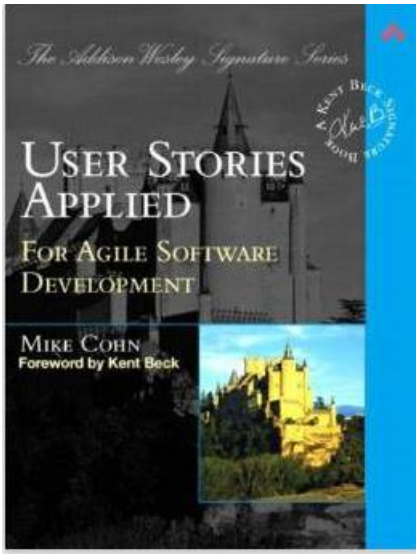
Scaling Scrum

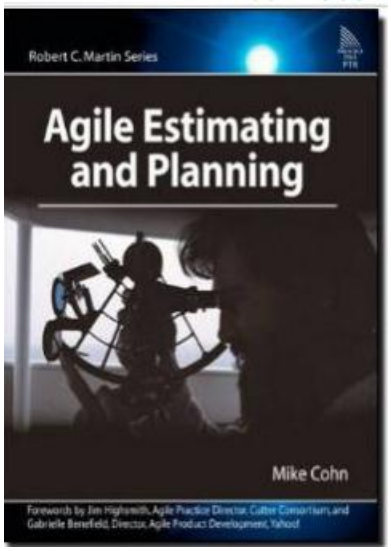


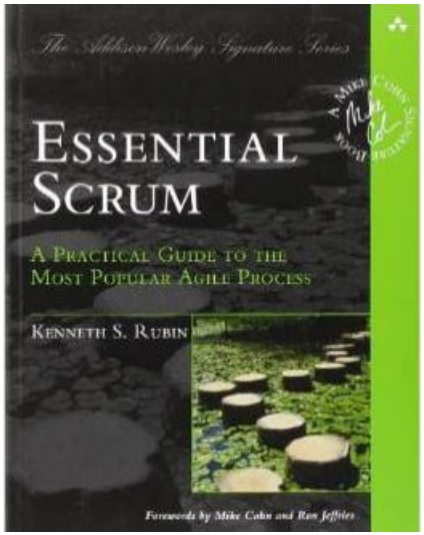
Simulación de Scrum

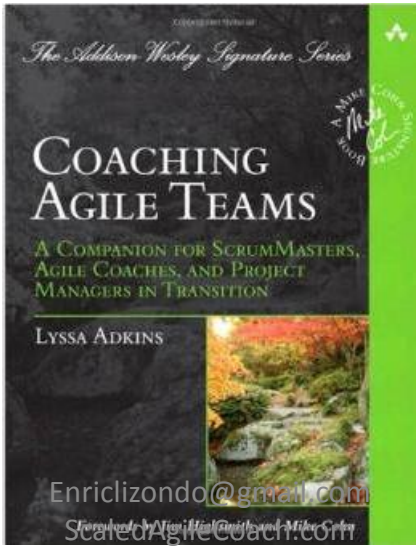
Scrum and coaching books

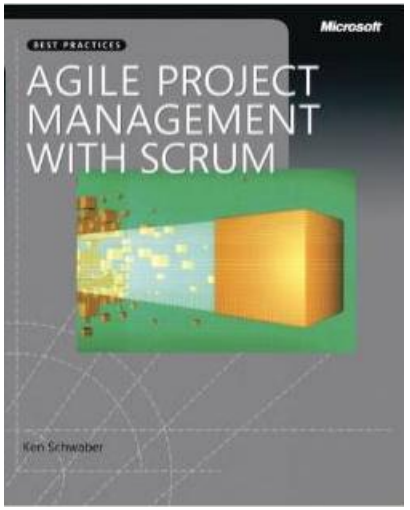




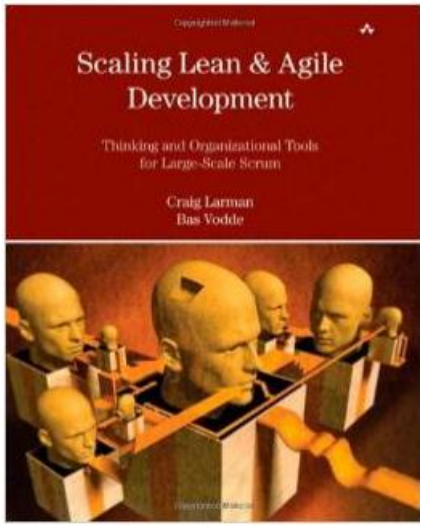


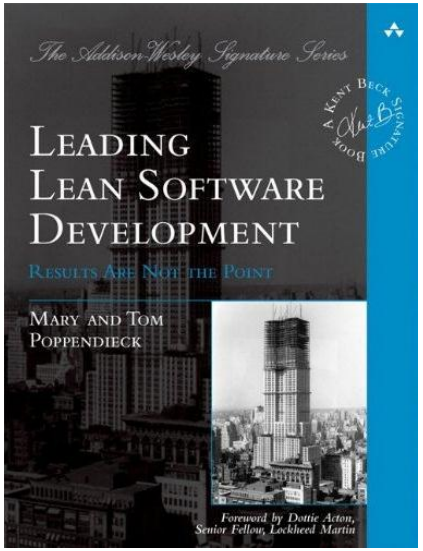


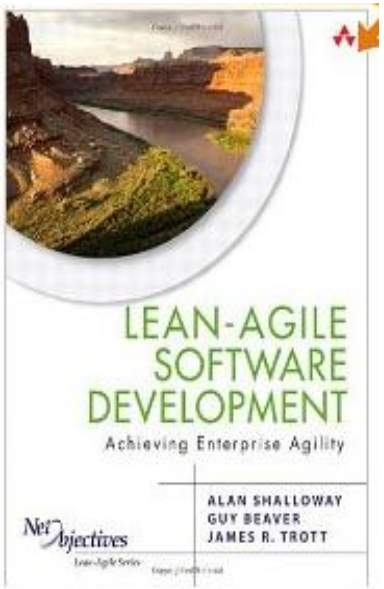


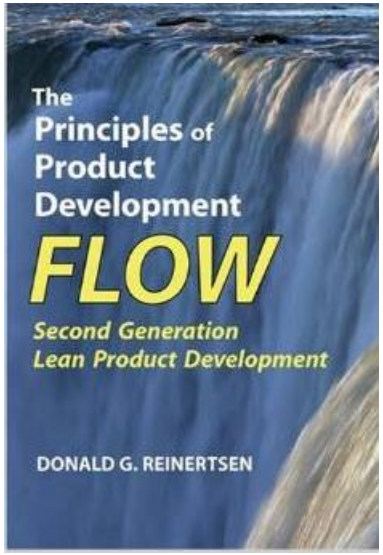


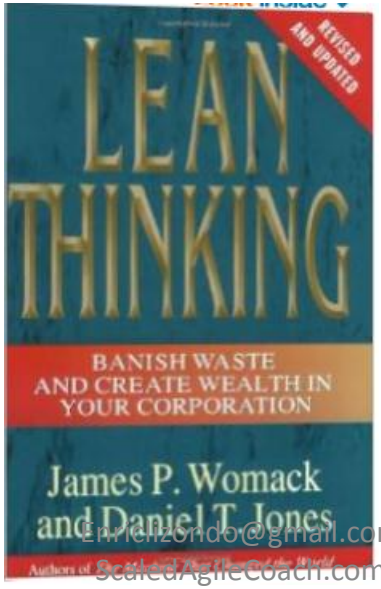
Lean and Agile Books

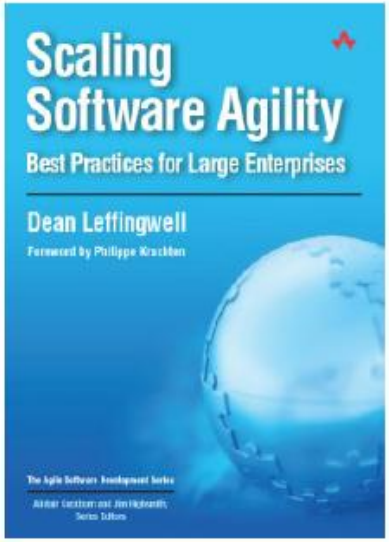




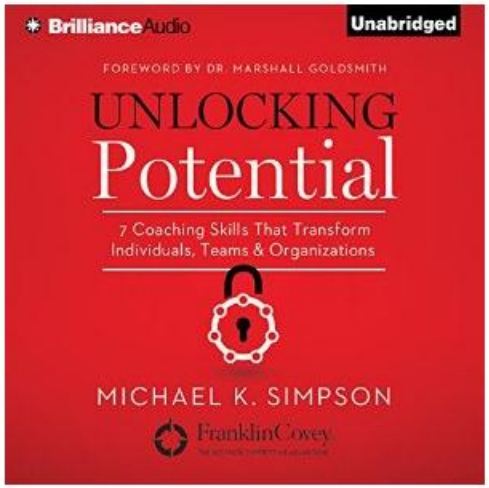


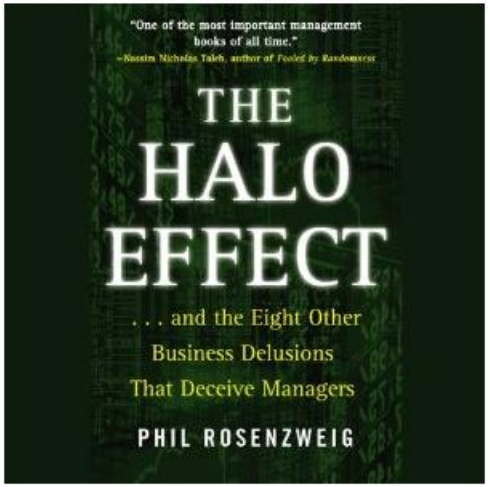


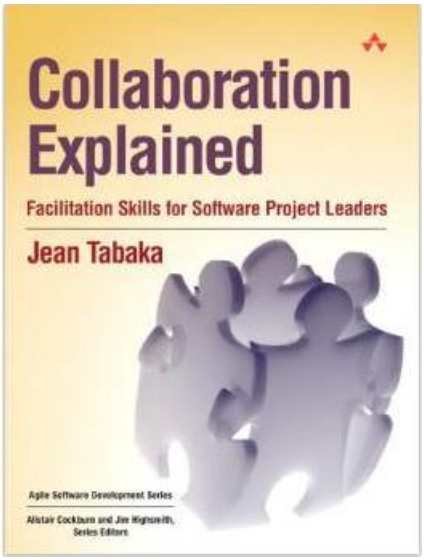




More books


Brilliance Audio
Unabridged
FOREWORD BY DR. MARSHALL GOLDSMITH
UNLOCKING
Potential
7 Coaching Skills That Transform
Individuals, Teams & Organizations
MICHAEL K. SIMPSON
FranklinCovey


"One of the most important management books of all time."
THE
HALO
EFFECT
. . . and the Eight Other
Business Delusions
That Deceive Managers
PHIL ROSENZWEIG


Collaboration
Explained
Facilitation Skills for Software Project Leaders
Jean Tabaka
Agile Software Development Series

Retro y cierre

GRACIAS