

Filosofía Agil Lean Empíricos Procesos

Definido vs. Empírico



El principio (sin Ingeniería)

1era. fila (de izq a der): Bill Gates, Andrea Lewis, Marla Wood, y Paul Allen. En el medio: Bob O'Rear, Bob Greenberg, Marc McDonald, and Gordon Letwin. Atrás: Steve Wood, Bob Wallace, and Jim Lane.



Desarrollo ágil de Software (Agile)

- Un compromiso útil entre nada de proceso y demasiado proceso (Fowler, 2001)

Manifiesto Ágil

Declaración de principios Ágiles, los cuatro valores :

- **Individuos e interacciones** por sobre procesos y herramientas
- **Software funcionando** por sobre documentación detallada
- **Colaboración** por sobre negociación con el cliente
- **Responder a cambios** por sobre seguir un plan

©2001, Kent Beck, Mike Beedle, Arievan Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland, Dave Thomas, 850+ signers

Los 12 principios del Manifiesto(I)

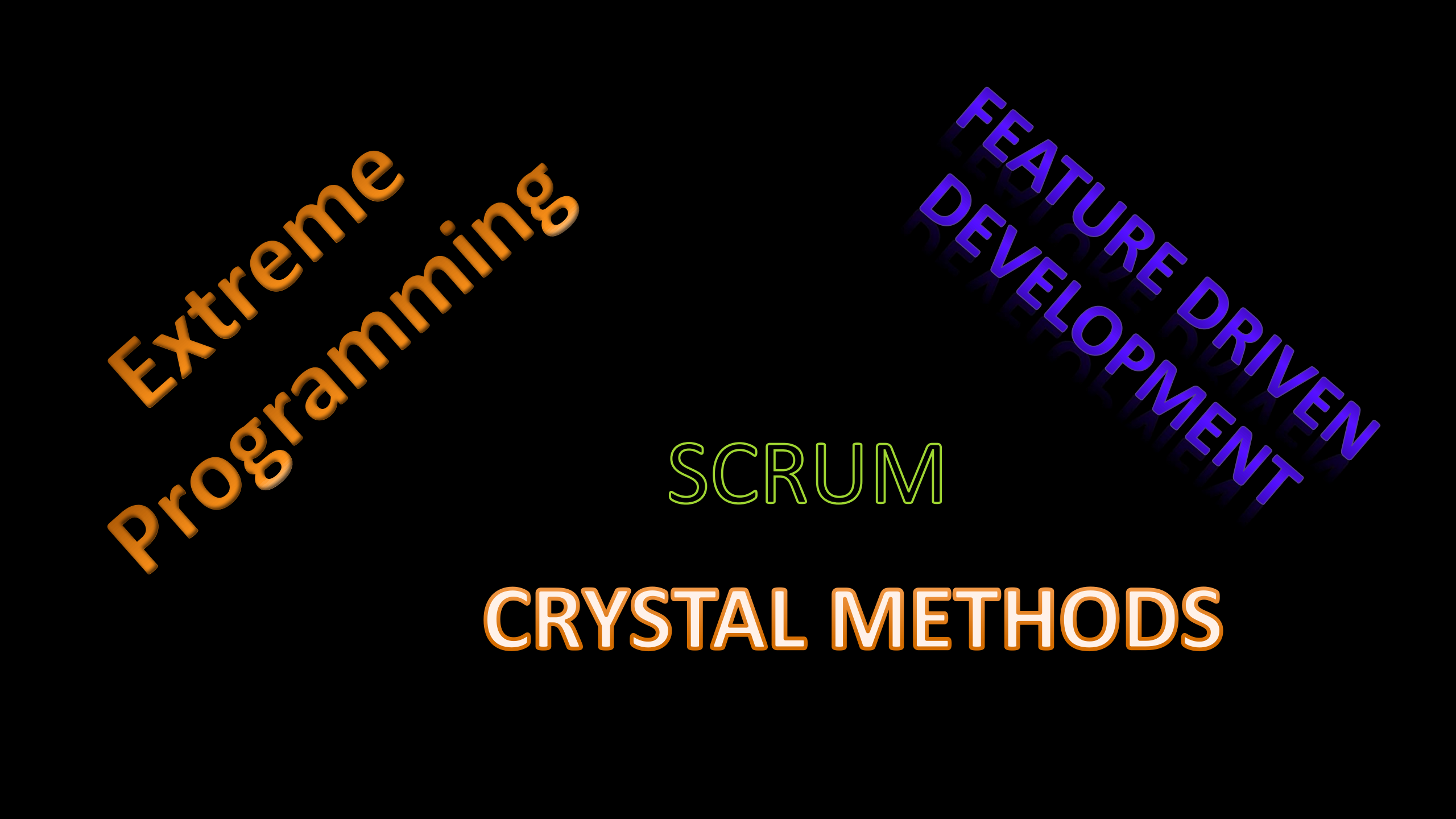
- 1 - La prioridad es satisfacer al cliente a través de releases tempranos y frecuentes (2 semanas a un mes)
- 2 - Recibir cambios de requerimientos, aun en etapas finales
- 3 – Release frecuentes (2 semanas a un mes)
- 4 - Técnicos y no técnicos trabajando juntos TODO el proyecto
- 5 - Hacer proyectos con individuos motivados
- 6 - El medio de comunicación por excelencia es cara a cara

Los 12 principios del Manifiesto(II)

- 7 - La mejor métrica de progreso es la cantidad de software funcionando
- 8 - El ritmo de desarrollo es sostenible en el tiempo
- 9 - Atención continua a la excelencia técnica
- 10 - Simplicidad - Maximización del trabajo no hecho
- 11 - Las mejores arquitecturas, diseños y requerimientos emergen de equipos auto-organizados
- 12 - A intervalos regulares, el equipo evalúa su desempeño y ajusta la manera de trabajar

¿Pero que significa Ágil?

- Balance entre ningún proceso y demasiado proceso. La diferencia inmediata es la exigencia de una menor cantidad de documentación, sin embargo no es eso lo más importante:
 - Los métodos ágiles son adaptables en lugar de predictivos.
 - Los métodos ágiles son orientados a la gente en lugar de orientados al proceso.



Extreme
Programming

FEATURE DRIVEN
DEVELOPMENT

SCRUM

CRYSTAL METHODS

¿Por qué ir a Agile?



87%

Ability to manage changing priorities



85%

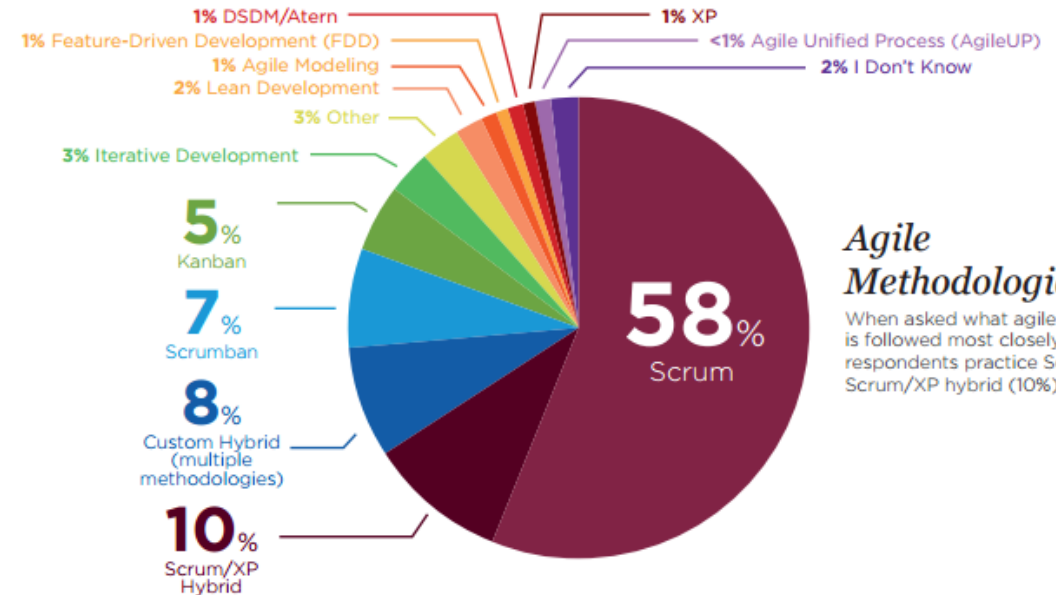
Increased team productivity



84%

Improved project visibility

AGILE METHODS AND PRACTICES



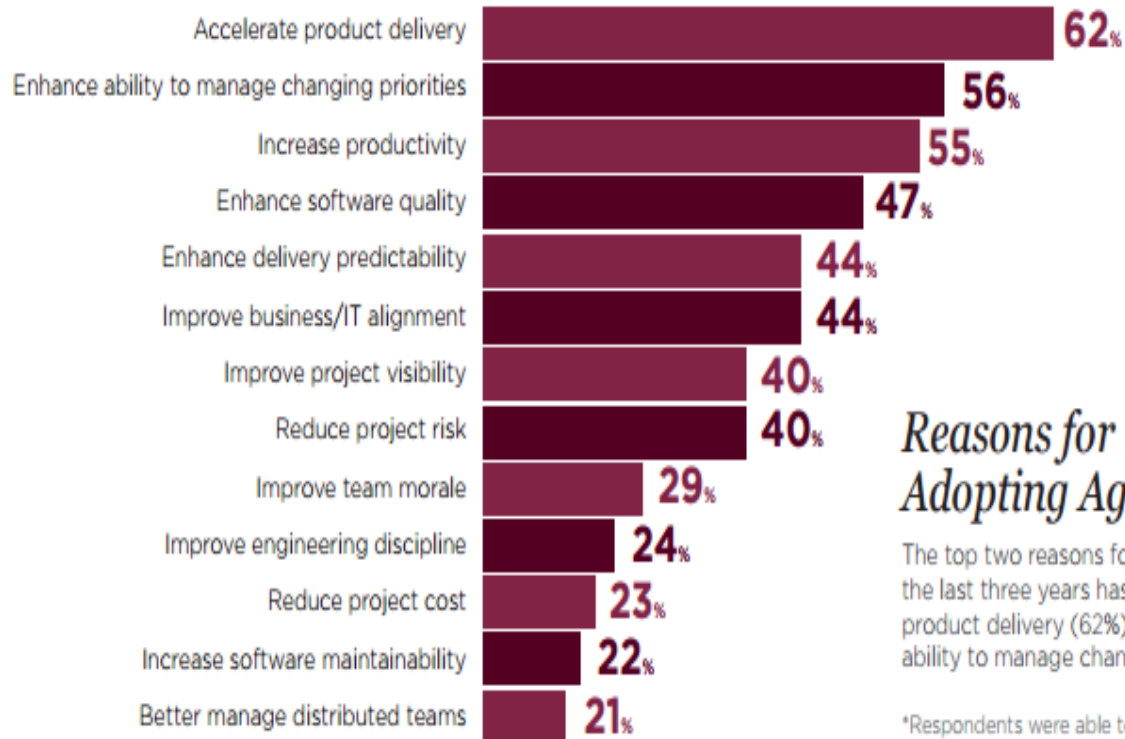
Agile Methodologies Used

When asked what agile methodology is followed most closely, nearly 70% of respondents practice Scrum (58%) or Scrum/XP hybrid (10%).

Reasons for Adopting Agile

The top two reasons for adopting the last three years has been to accelerate product delivery (62%) and enhance ability to manage changing priorities (56%).

*Respondents were able to make multiple selections.



Técnicas efectivas

Agile Techniques Employed

More than 39% of the respondents practiced Kanban within their organizations, up from 31% in 2014. Conversely, iteration planning dropped slightly from 71% in 2014 to 69% in 2015, likely indicating a transition to more flow-based methods such as Lean and Kanban.

TOP 5 AGILE TECHNIQUES



83%

DAILY
STANDUP



82%

PRIORITIZED
BACKLOGS



79%

SHORT
ITERATIONS



74%

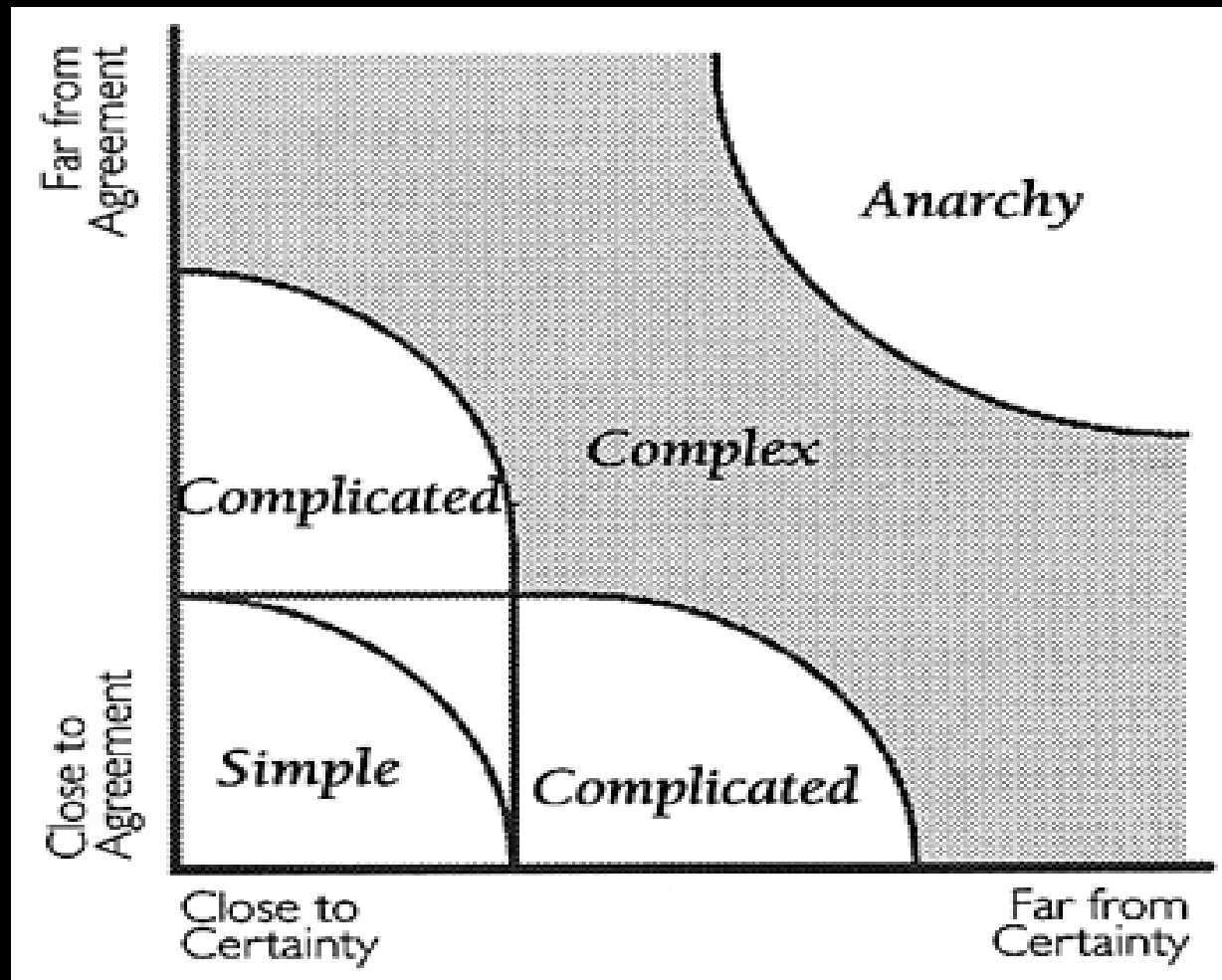
RETROSPECTIVES



69%

ITERATION
PLANNING

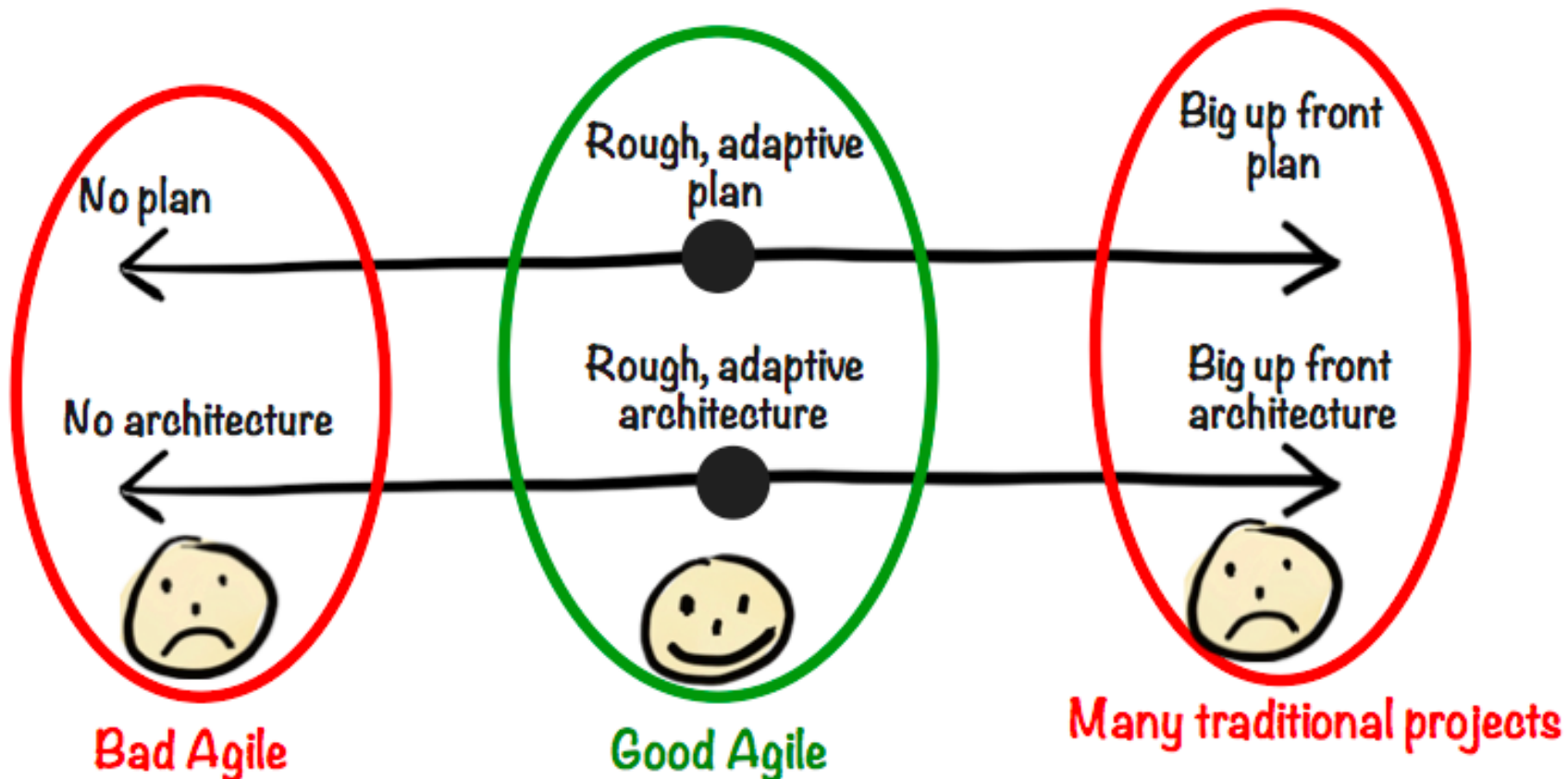
¿Cuándo Agile es aplicable?



- Agile da mejores resultados cuando los problemas a ser resueltos caen dentro del espacio “Complex”.
- El desarrollo de nuevos productos y Knowledge Work tienden a estar en el espacio Complex.
- Investigación esta dentro de Anarchy
- Mantenimiento cae en Simple (siempre???)

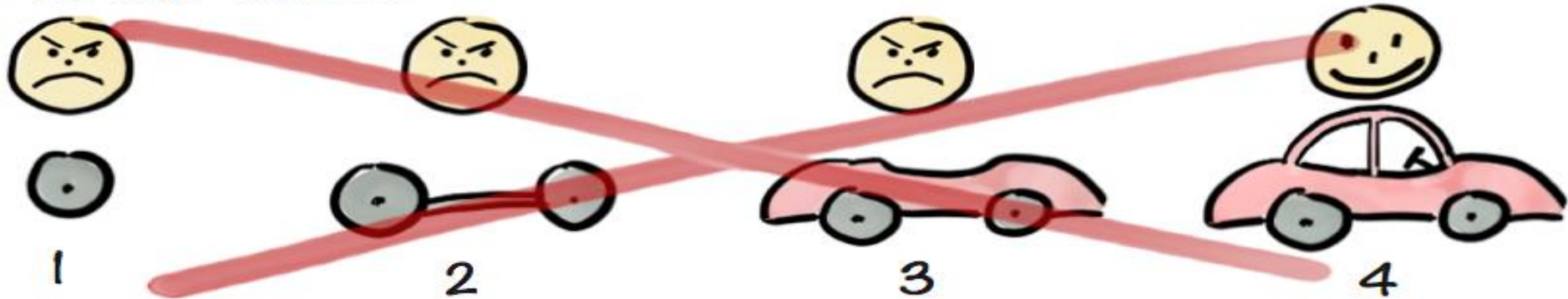
Ser Ágil no es ser indisciplinado.

Don't go overboard with Agile!

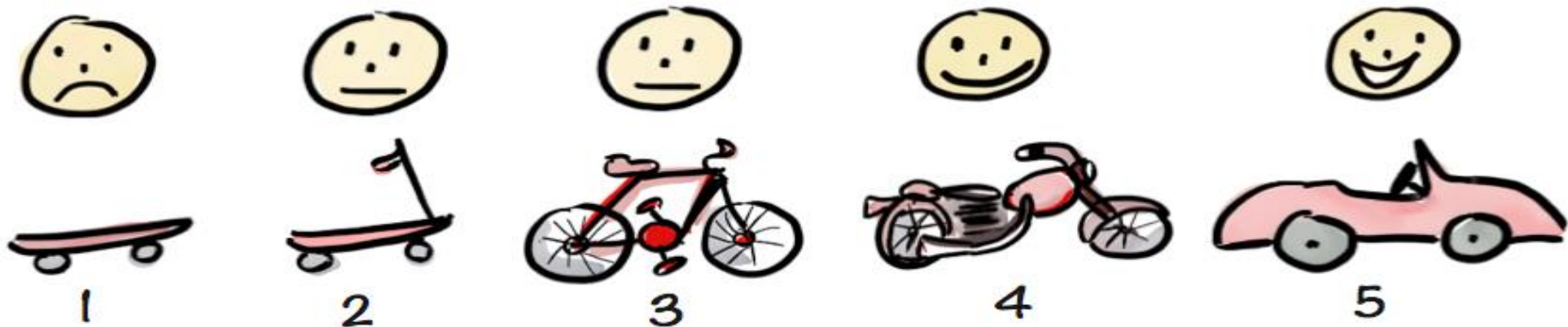


Entonces hacemos todo por pedacitos y
somos agiles!!!?? NO!

Not like this....



Like this!



Filosofía Lean



Principios Lean



Principios Lean

- Eliminar el desperdicio: evitar que las cosas se pongan viejas antes de terminarlas o evitar re-trabajo.
 - Tiene que ver con el principio ágil de Software funcionando y el de simplicidad (arte de maximizar lo que no hacemos)

Pensamiento Lean: Siete Principios

- Eliminar Desperdicios: reducir el tiempo removiendo lo que no agrega valor.
 - Desperdicio es cualquier cosa que interfiere con darle al cliente lo que el valora en tiempo y lugar donde le provea más valor.
 - En manufactura: el inventario
 - En Software: es el trabajo parcialmente hecho y las características extra!
 - El 20 % del software que entregamos contiene el 80% del valor

Pensamiento Lean: Siete Principios

- Amplificar el aprendizaje: crear y mantener una cultura de mejoramiento continuo y solución de problemas.
 - Un proceso focalizado en crear conocimiento esperará que el diseño evolucione durante la codificación y no perderá tiempo definiéndolo en forma completa, prematuramente.
 - Se debe generar nuevo conocimiento y codificarlo de manera tal que sea accesible a toda la organización.
 - Muchas veces los procesos “estándares” hacen difícil introducir en ellos mejoras.

Principios Lean

- Embeber la integridad conceptual. Encastrar todas las partes del producto o servicio, que tenga coherencia y consistencia (tiene que ver con los Requerimientos No Funcionales). La integración entre las personas hace el producto más integro.

PENSAMIENTO LEAN: SIETE PRINCIPIOS

- Embeber la integridad conceptual: se necesita más disciplina no menos!
 - Integridad Percibida: el producto total tiene un balance entre función, uso, confiabilidad y economía que le gusta a la gente.
 - Integridad Conceptual: todos los componentes del sistema trabajan en forma coherente en conjunto.
 - El objetivo es construir con calidad desde el principio, no probar después.
 - Dos clases de inspecciones:
 - Inspecciones luego de que los defectos ocurren.
 - Inspecciones para prevenir defectos.
 - Si se quiere calidad no inspeccione después de los hechos!
 - Si no es posible, inspeccione luego de pasos pequeños.

Principios Lean

- Diferir Compromisos. El último momento responsable para tomar decisiones (en el cual todavía estamos a tiempo). Si nos anticipamos tenemos información parcial.
 - Se relaciona con el principio ágil: decidir lo más tarde posible pero responsablemente. No hacer trabajo que no va a ser utilizado. Enlaza con el principio anterior de aprendizaje continuo, mientras más tarde decidimos más conocimiento tenemos.

Pensamiento Lean

- Diferir compromisos: las decisiones deben tomarse en el último momento que sea posible.
 - No significa que todas las decisiones deben diferirse.
 - Se debe tratar de tomar decisiones reversibles, de forma tal que pueda ser fácilmente modificable.
 - Vencer la “parálisis del análisis” para obtener algo concreto terminado.
 - Las mejores estrategias de diseño de software están basadas en dejar abiertas opciones de forma tal que las decisiones irreversibles se tomen lo más tarde posible.

Principios Lean

- **Entregar rápido:** estabilizar ambientes de trabajo a su capacidad más eficiente y acotar los ciclos de desarrollo.
- **Entregar rápidamente** esto hace que se vayan transformando “n” veces en cada iteración. Incrementos pequeños de valor. Llegar al producto mínimo que sea valioso. Salir pronto al mercado.
 - Relacionado con el principio Ágil de entrega frecuente.



Principios Lean

- Dar poder al equipo: ejemplo, vamos a comer a un restaurante y no nos metemos en la cocina del restaurante. Nos fijamos en el precio, pedimos y esperamos. Hay mucho micro management, el dueño no decide cuánta sal poner a la comida.
- Respetar a la gente
 - Entrenar líderes
 - Fomentar buena ética laboral
 - Delegar decisiones y responsabilidades del producto en desarrollo al nivel más bajo posible
- Ágil: El propio equipo pueda estimar el trabajo.

Principios Lean

- Ver el todo: tener una visión holística, de conjunto (el producto, el valor agregado que hay detrás, el servicio que tiene los productos como complemento).

Desperdicios

En Manufactura

- Producción en exceso
- Stock
- Pasos extra en el proceso
- Búsqueda de información
- Defectos
- Esperas
- Transportes

En software

- Características extras
- Trabajo a medias
- Proceso extra
- Movimiento
- Defectos
- Esperas
- Task Switching

En desarrollo de software el referente principal es...

KANBAN