Ingeniería de Software

Procesos de Desarrollo de Software

Juan Pablo Sandoval

Actividad de Clase (opcional)

- *El objetivo es diseñar un proceso para ordenar un conjunto de cartas.*
- *Necesito 3 grupos, 5 personas.*
  - *Analizar/Diseñar/Implementar/Probar un proceso para ordenar cartas.*
  - *Tienen 5 minutos para analizar y diseñar un plan.*
    - *Todos los miembros del equipo tiene que hacer al menos una tarea.*
  - *El equipo que logre ordenar las cartas mas rápido ganará 1* ***décima****.*



Escribir código es relativamente simple



Desarrollar software de calidad es difícil

- *Trabajar con personas puede llegar a ser un problema…*
  - *Clientes*
  - *Desarrolladores*
  - *Managers*
- *Crear programas grandes toma mucho tiempo…*
- *Tan pronto se involucra más de dos personas en el proceso aparecen complicaciones … esto incluye al cliente!*

Seguir un proceso (cualquier proceso) es de suma importancia!

Desarrollar software de calidad es difícil

One of the planning documents for software research revealed --in a parenthetical remark only-- an unchallenged tacit assumption by referring to ***"the tradeoff between cost and quality"****. Now in all sorts of mechanical engineering it may make sense to talk about* ***"the tradeoff between cost and quality"****, in software development this is absolute nonsense, because poor quality is the major contributor to the soaring costs of software development.*

Dijkstra, EWD690

Un proceso?

- *Organizaciones y/o empresas quieren tener procesos de desarrollo bien definidos, entendibles y repetidles.*
- ***Porque?***

Un proceso?

- *Organizaciones y/o empresas quieren tener procesos de desarrollo bien definidos, entendibles y repetidles.*
- ***Porque?***
  - Encontrar y repetir buenas practicas
  - Administración
    - Como saber qué debemos hacer después?
    - Cómo saber en qué tarea vamos y cuál haremos después?
    - Como sabemos si estamos tarde?
    - Como medimos el progreso?
    - Necesitamos estimar el tiempo o el costo para terminar…
  - Como los nuevo miembros del equipo sabrán que tienen que hacer?

Proceso de Desarrollo de Software

- *Un proceso de desarrollo es un conjunto de actividades que toman lugar en secuencia con el fin de crear un software*
- *“La creación y traducción de un necesidad de las personas en requerimientos, requerimientos en diseño, el diseño en código, pruebas, instalación, etc ”*

Proceso de Desarrollo de Software

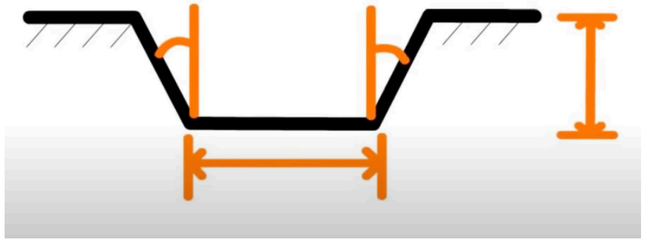
- *Todos los procesos tiene consideran en alguna medida:*
  - *El recopilado de requerimientos*
  - *La especificación de requerimientos*
  - *Diseño*
  - *Desarrollo (diseño, implementación, pruebas)*
  - *Validación*
  - *Evolución (mantenimiento)*
- *La diferencia, existen varios procesos que enfatizan cada uno de estas etapas.*

Procesos de Desarrollo

- *Proceso en Cascada*
- *Proceso Iterativo*
- *Proceso Iterativo e Incrementar*

Procesos de Desarrollo

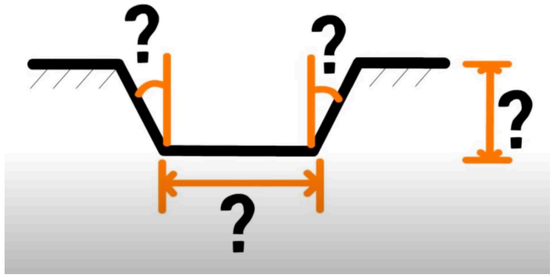
- ***Proceso en Cascada***
- *Proceso Iterativo*
- *Proceso Iterativo e Incrementar*

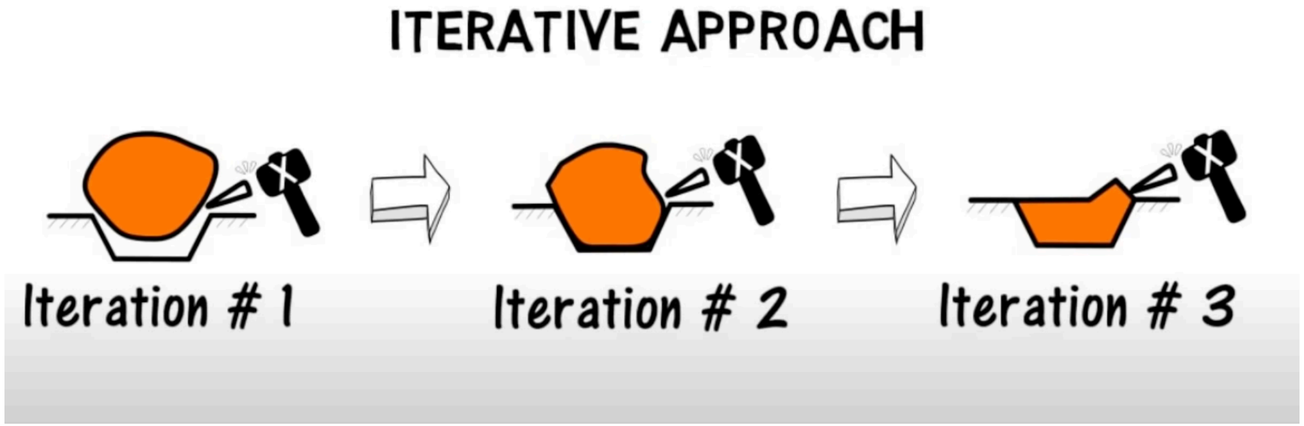




Procesos de Desarrollo

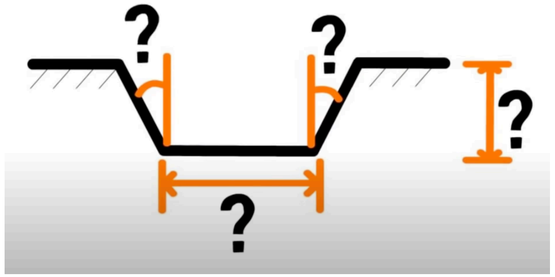
- *Proceso en Cascada*
- ***Proceso Iterativo***
- *Proceso Incremental*
- *Proceso Iterativo e Incrementar*

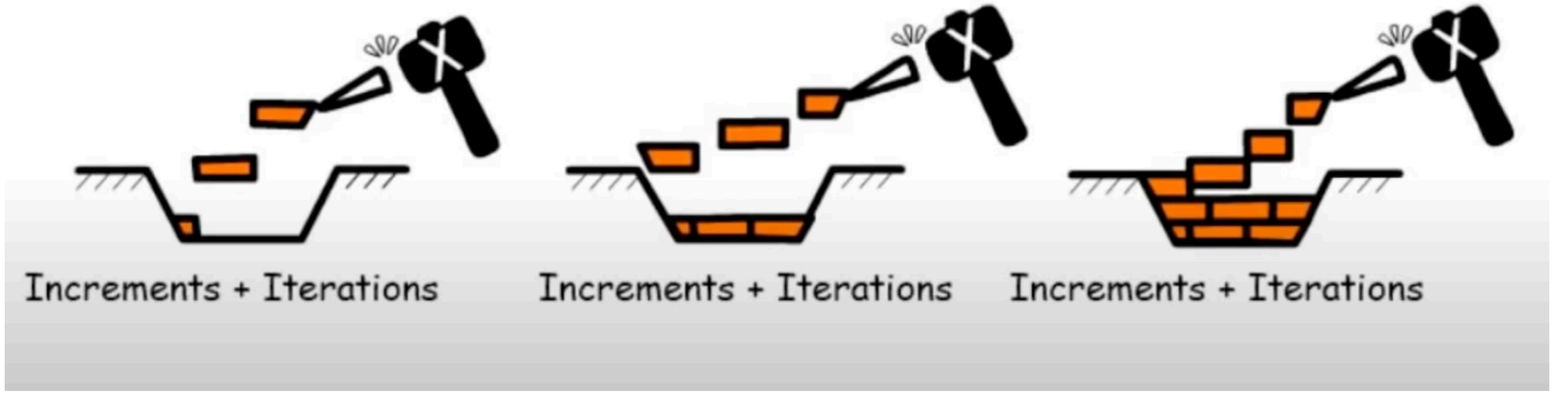




Procesos de Desarrollo

- *Proceso en Cascada*
- *Proceso Iterativo*
- ***Proceso Iterativo e Incremental***





Iterative AND Incremental

One way of doing things is building things piece by piece, **incrementally**.





Another way is sketching and refining, **iteratively**.

In reality we do **both** at the same time, with **feedback after every step.**





Jimmy Janlén - Agile Thingy #1

YouTube https://agilewithjimmy.com

© Jimmy Janlén 2021

Agile WITH Jimmy

Eligiendo un proceso

- *Factores organizacionales*
  - *El equipo esta en el mismo lugar o es remoto.*
  - *Como esta estructurado el equipo.*
    - *Existen project managers, señor devs, equipo pequeño, grande.*
- *Factores tecnológicos*
  - *Como le gusta trabajar al equipo (pizarras o post-its)*
  - *Tenemos la posibilidad de tener una comunicación efectiva*

Eligiendo un proceso

- *Factores del negocio*
  - *Cuan familiar eres con el producto?*
  - *Cuan a menudo te piden cambios o mejoras?*
  - *Cuan rápido el estado del arte esta cambiando?*
- *Factores regulatorios*
  - *Estas construyendo algo que necesita aprobación del gobierno?*
  - *Necesitas proporcionar documentación muy especifica?*

Eligiendo un proceso

- *Factores humanos*
  - *Como esta compuesto tu equipo?*
  - *Están en el mismo lugar?*
  - *Están uno alado del otro?*
  - *Confían uno del otro?*
  - *Como es la cultura de la compañia?*

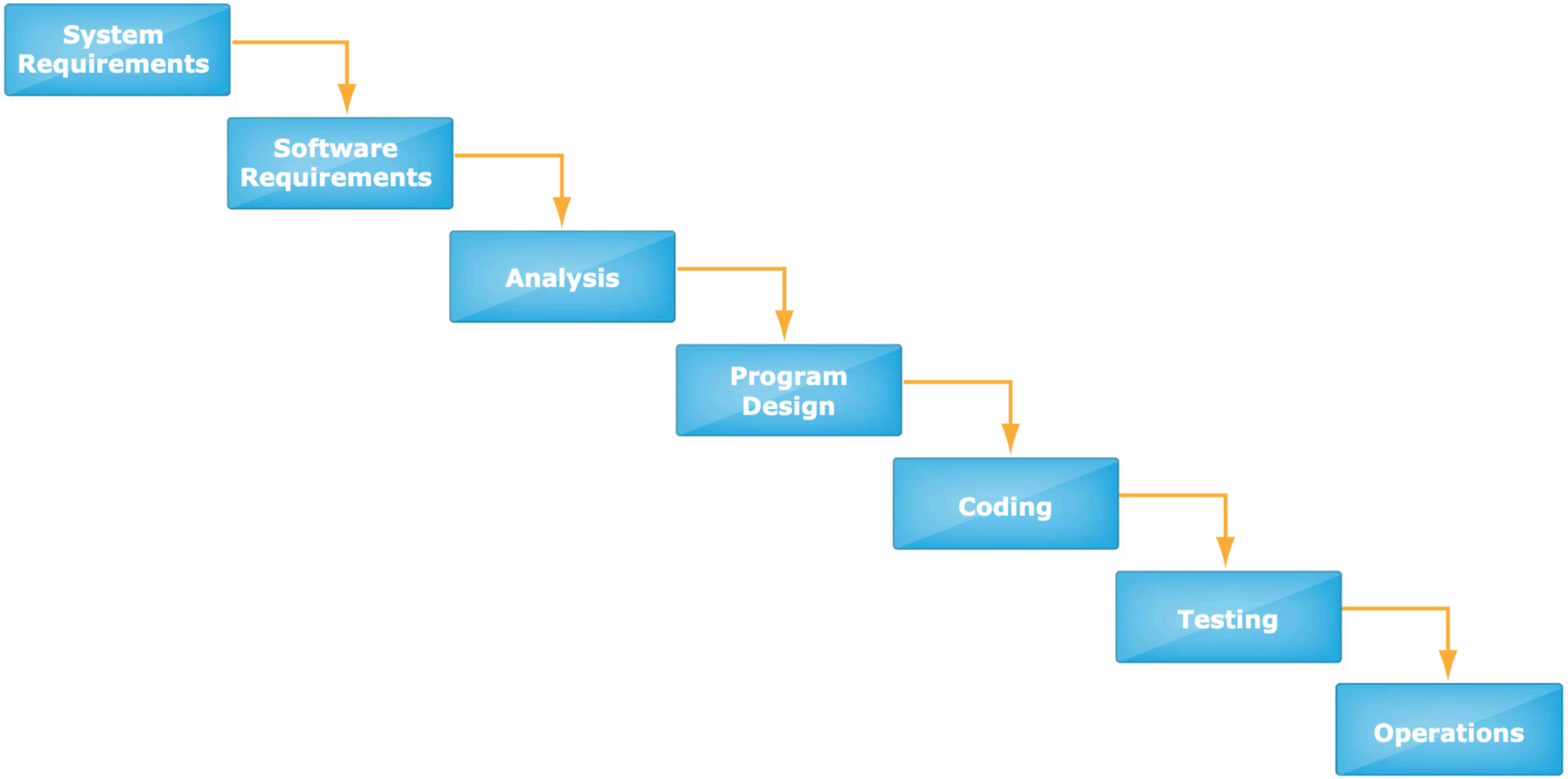
Raya para la suma

- *No existe un **buen** proceso que tenga todo cubierto.*
  - *Depende: del equipo, del cliente, de dominio, etc.*
  - *En realidad, la mayoría de las compañías no utilizan un proceso de forma estricta.*
  - *Todo es personalizado al proyecto, equipo y/o empresa.*
  - *Es posible tomar inspiración de varios procesos dependiendo a las necesidades.*

Ejemplos

- *Modelo en Cascada*
- *Modelo en Espiral*
- *Proceso Unificado*
- *Programación Extrema*
- *Scrum*

The waterfall model, 1970

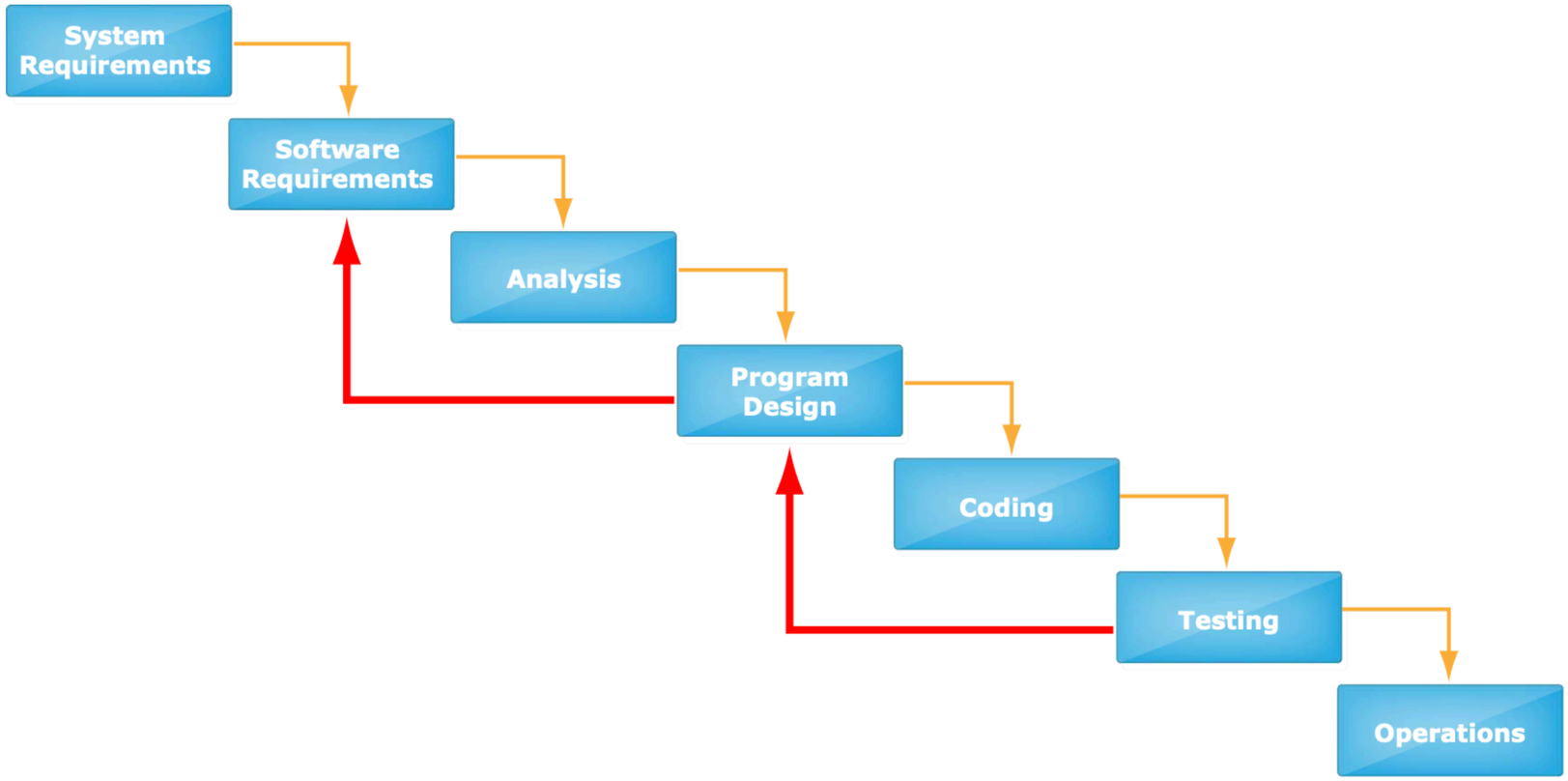


from: **Winston Royce**, “Managing the Development of Large Software Systems”, Proceedings of IEEE W

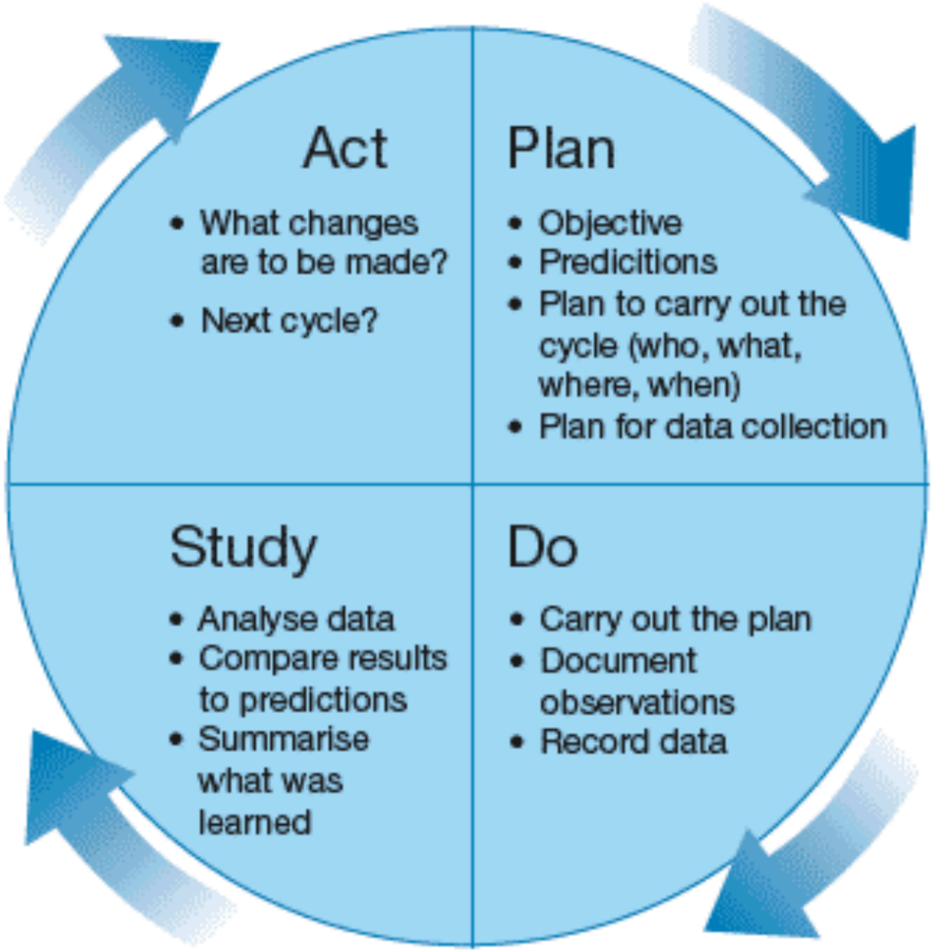
The waterfall model, 1970

- *La principal ventaja es que es fácil de entender.*
- *Desventaja:*
  - *No considera retroalimentación.*
  - *Es muy difícil incorporar cambios.*
  - *Es muy rígido, no es realista considerar los resultados de cada etapa como definitivos.*
  - *Dilata la reducción del riesgo.*

Lo que pasa en la practica



Origenes de los procesos iterativos

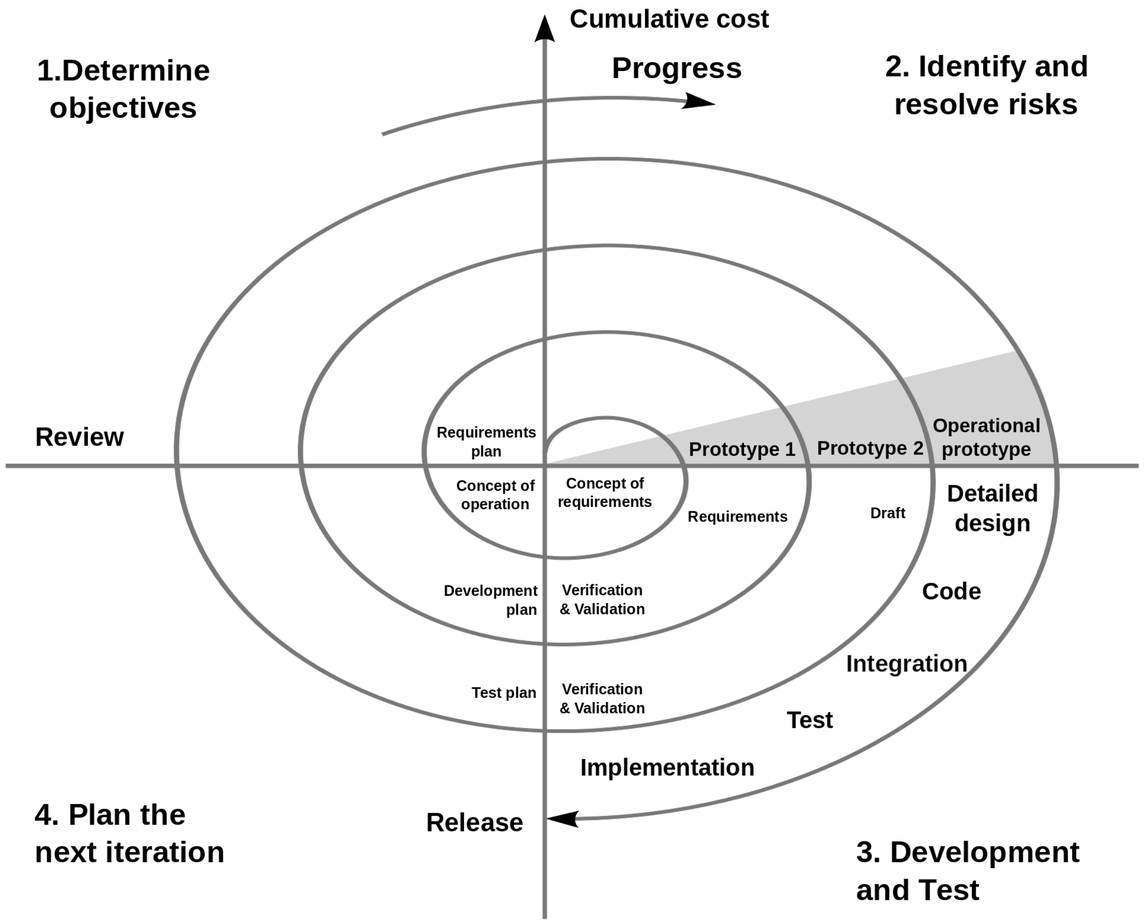


plan-do-study-act
- Shewart, 1930s
- Deming, 1940s

Project Mercury
- NASA, 1960s
- half-day iterations
- tests before each iter
- became IBM Federal Systems Division

Craig Larman, Victor R. Basili (June 2003). "Iterative and Incremental Development: A Brief History". IEEE Computer(IEEE Computer Society) 36 (6): 47–56.

Spiral Model, 1986



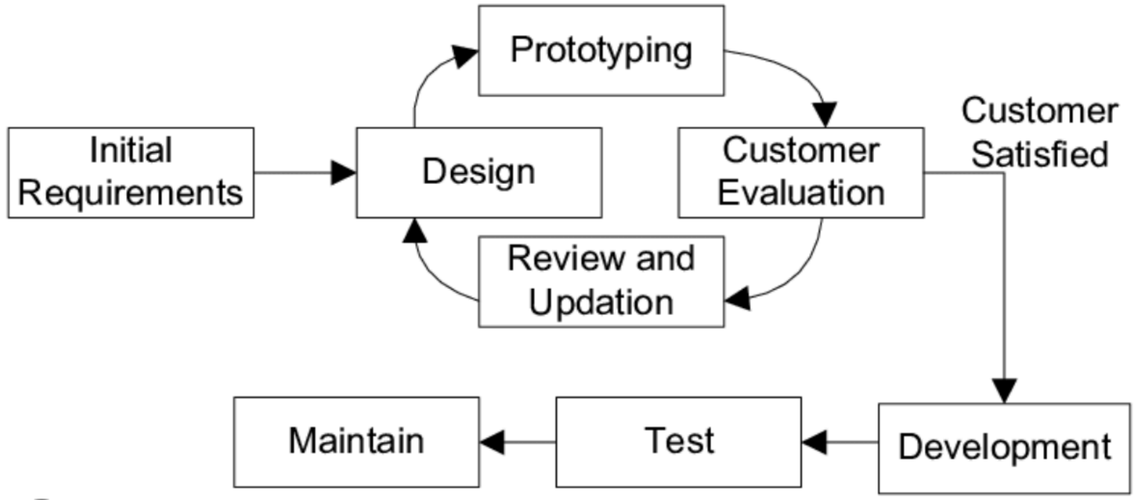
Boehm, B (July 2000). "Spiral Development: Experience, Principles, and Refinements" (PDF). *Special Report*. Software Engineering Institute. CMU/SEI-2000-SR-008.

Spiral Model, 1986

- *“Una mejora al modelo en cascada”*
- *El modelo en espiral tiene la habilidad de iterar en ciclos y fases.*
- *Las mejoras importantes:*
  - *Existen un análisis de riesgo que se hace explicito en cada etapa.*
  - *Introduce la noción de proceso iterativo y prototipos.*
  - *La repetición de actividades esta explicito en el modelo.*

Prototype Model

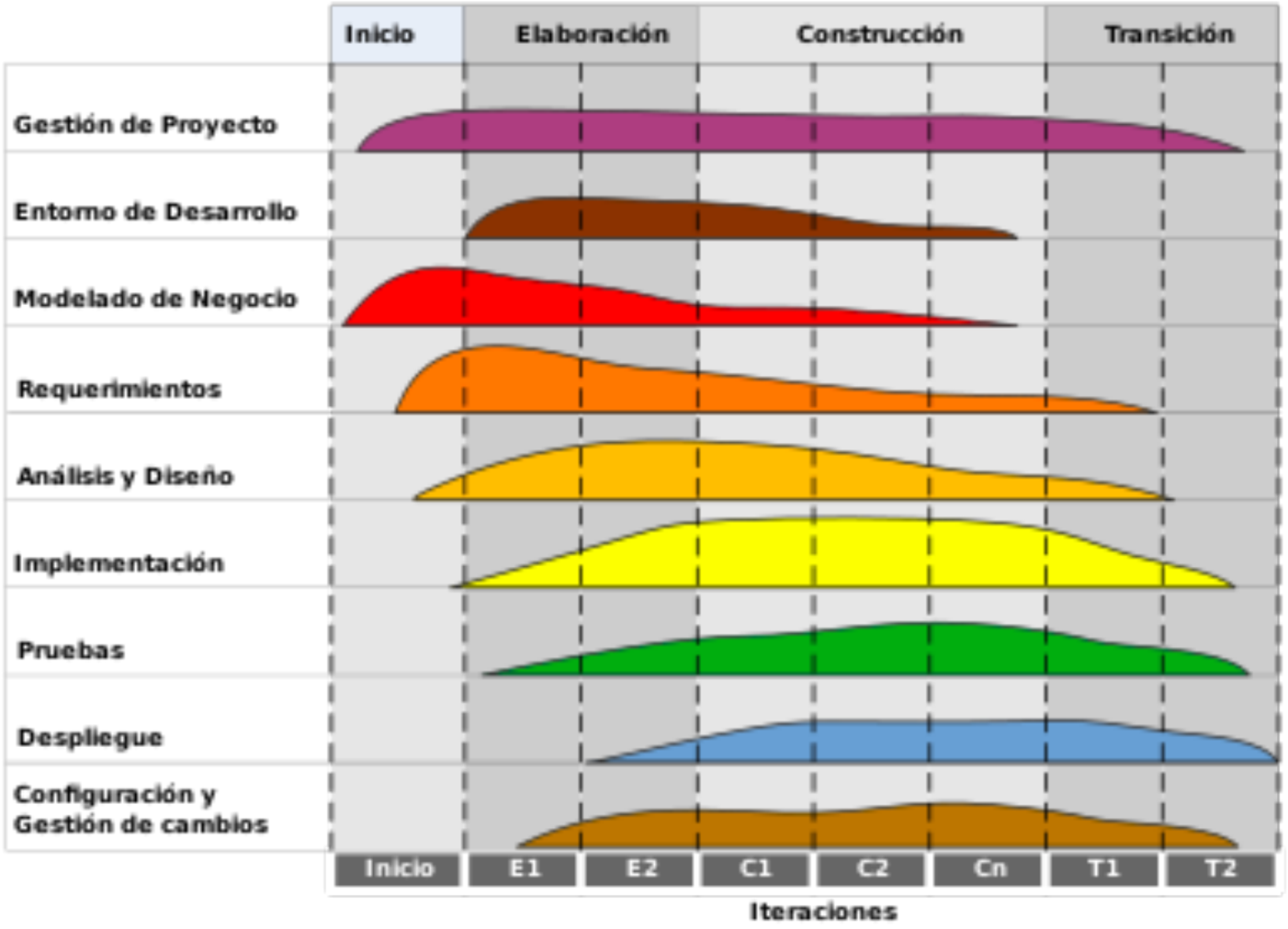
- *La idea es que el usuario pueda tener una mirada concreta del producto en forma temprana y resolver el dilema "el usuario solo sobrar lo que quería cuando le muestres el resultado"*
  - *Prototipos desechables.*
  - *Prototipos evolutivos.*

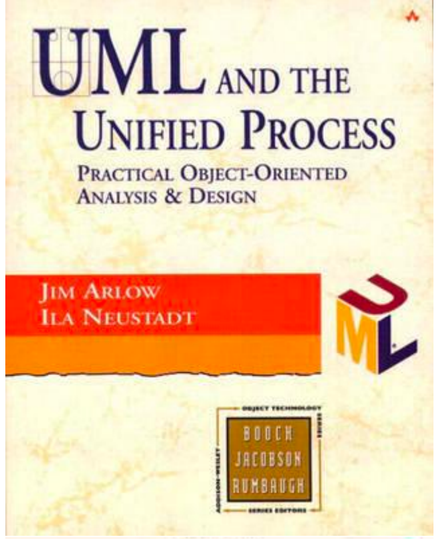


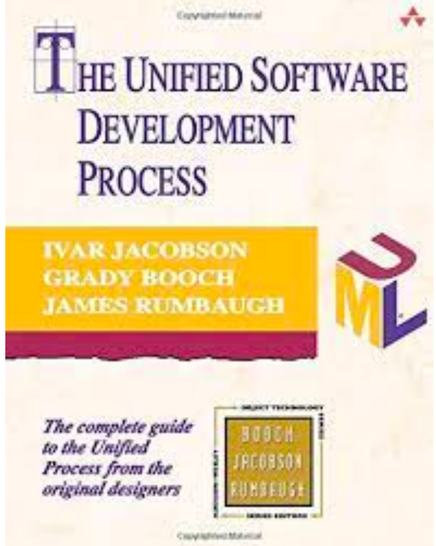
Prototype Model

- ***Ventajas:***
  - *Permite enfrentar falta de claridad en los requerimientos*
  - *No existen grandes sorpresas al final*
  - *Menor frustración y ansiedad durante el desarrollo*
  - *… entre otros*
- ***Desventajas:***
  - *Puede generar expectativas de que el software está “casi listo”.*
  - *Producto final no queda igual que prototipo que se desecha.*
  - *El usuario puede estar nunca satisfecho.*
  - *Análisis incompleto de los requisitos*
  - *… entre otros*

Proceso Unificado

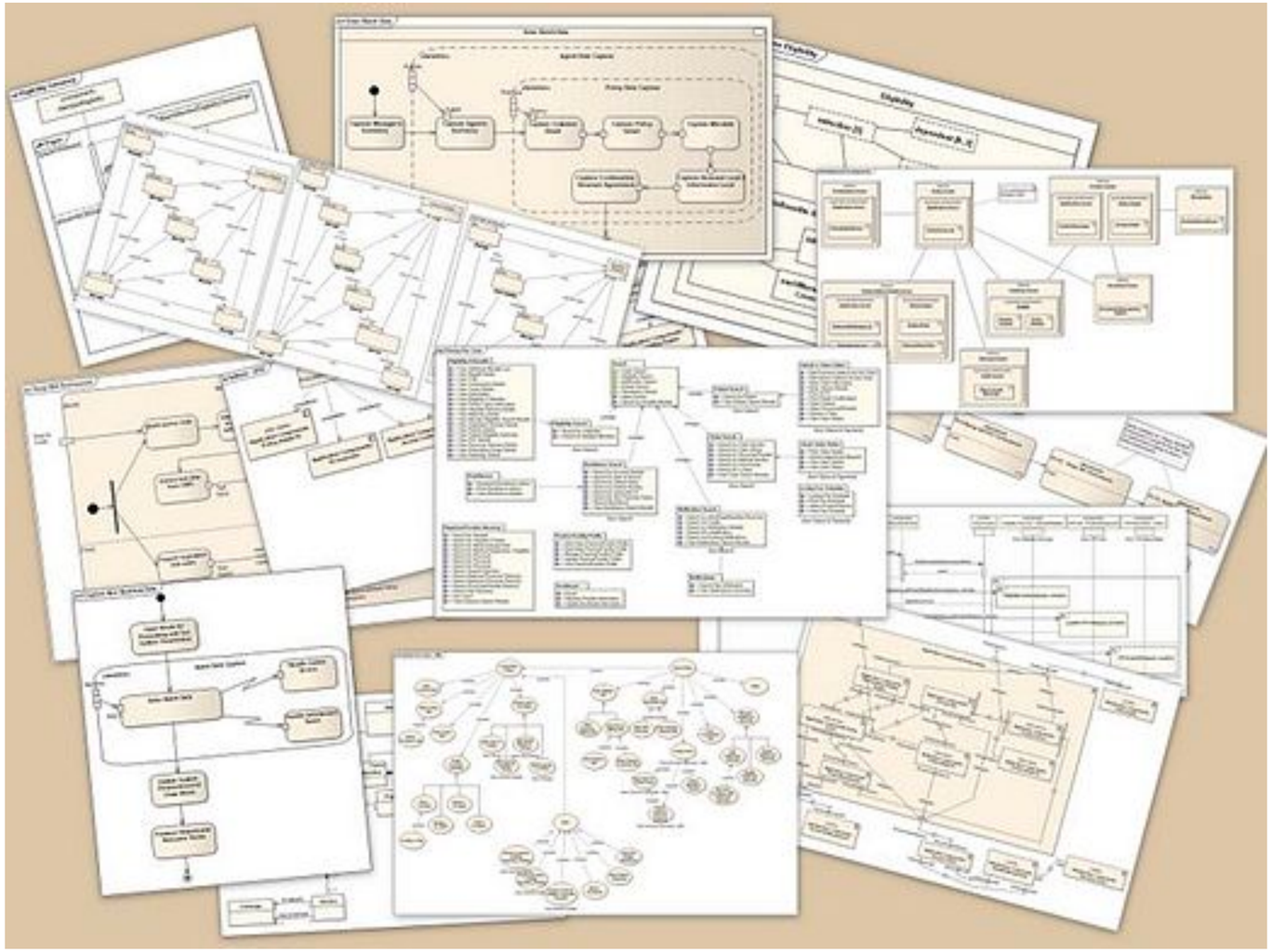








Lenguaje del Modelo Unificado



Xtreme Programming

- ***Kent Beck, 1999***
  - *take best practices to “extreme” levels*
  - *developed during C3 project with Ron Jeffries*
- ***a sample of XP practices***
  - *test first: acceptance and unit tests*
  - *continuous integration*
  - *pair programming*
  - *repeated refactoring*

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
Mike Beedle
Arie van 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

© the above authors. All rights reserved. This content is excluded from our Creative Commons license. For more information, see http://ocw.mit.edu/fairuse.

Pero, ¿qué es Agile?



- *Dar más valor a …*
  - *Individuos y interacciones que a procesos y herramientas.*
  - *Software funcional que a la documentación exhaustiva.*
  - *Colaboración con el cliente que a la negociación de un contrato.*
  - *Respuesta al cambio que a seguir con el plan*

https://agilemanifesto.org/principles.html

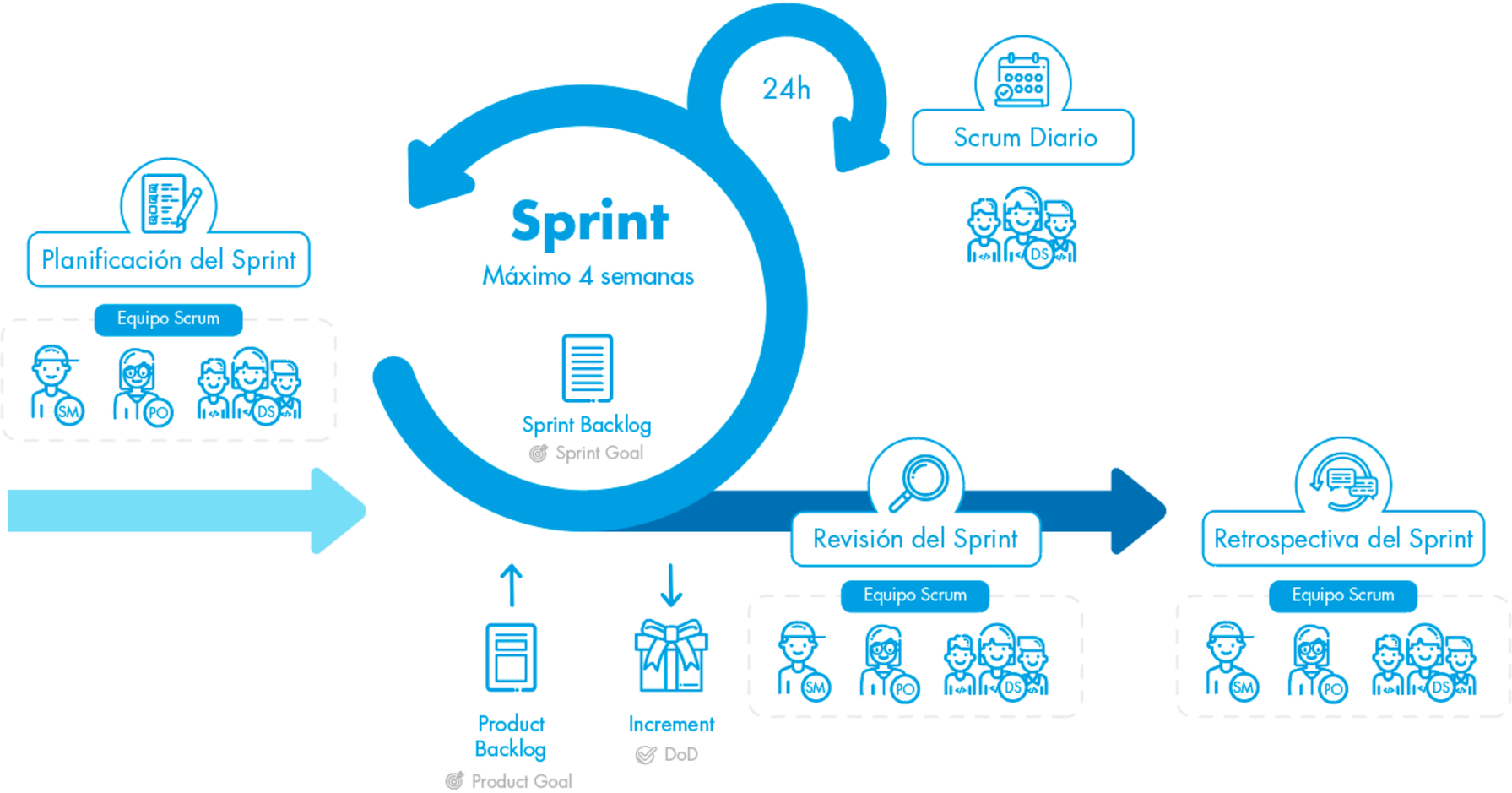
Pero, ¿qué es Agile?



- *Ademas el manifiesto ágil presenta 12 principios, como por ejemplo:*
  - *Nuestra mayor prioridad es satisfacer al cliente a través de una entrega temprana y continua de software valioso para él.*
  - *Abrazamos los cambios en los requerimientos, incluso si llegan tarde en el proceso.*

https://agilemanifesto.org/principles.html

Scrum


24h
Scrum Diario
Planificación del Sprint
Equipo Scrum
SM
PO
DS
Sprint
Máximo 4 semanas
Sprint Backlog
Sprint Goal
Revisión del Sprint
Equipo Scrum
Retrospectiva del Sprint
Equipo Scrum
Product Backlog
Product Goal
Increment
DoD