

“El Proceso de Ingeniería Web”



Rogelio Ferreira Escutia



Ingeniería de Software



Ingeniería del Software

- La Ingeniería del Software es el establecimiento y uso de firmes principios y métodos de Ingeniería para la obtención económica de software fiable y que funcione en máquinas reales.



Crisis del Software



- **A finales de los años 60, la mayoría de los proyectos tendían al fracaso.**

- **Al principio de los años 70 se presentó la “Crisis del Software” y ocasionó las siguientes preguntas:**
 - **¿Cómo estimar los costos y tiempos?**
 - **¿Por qué las desviaciones son tan grandes?**
 - **¿Cómo se debe detectar el error?**
 - **¿A qué se debe la alta tasa de fallas?**
 - **¿Cómo responder a los requerimientos del cliente?**
 - **¿Cómo manejar requisitos volátiles?**

Fases de la Crisis del Software

- **1) Albores (1945-1955): Se utiliza lenguaje máquina o ensamblador.**
- **2) Florecimiento (1955-1965): Aparecen los lenguajes de programación para facilitar la tarea de desarrollo.**
- **3) La Crisis (1965-1970): La mayoría de los desarrollos fracasan por distintos motivos.**
- **4) Innovación conceptual (1970-1980): Surgen las primeras metodologías estructuradas así como el concepto de pruebas.**
- **5) “El Diseño es el Problema” (1980-?): El peso de los proyectos comienza a caer en el diseño y aparecen los conceptos de fiabilidad, jerarquía, redundancia, reutilización, modularidad.**

Modelos Clásicos de Ingeniería de Software

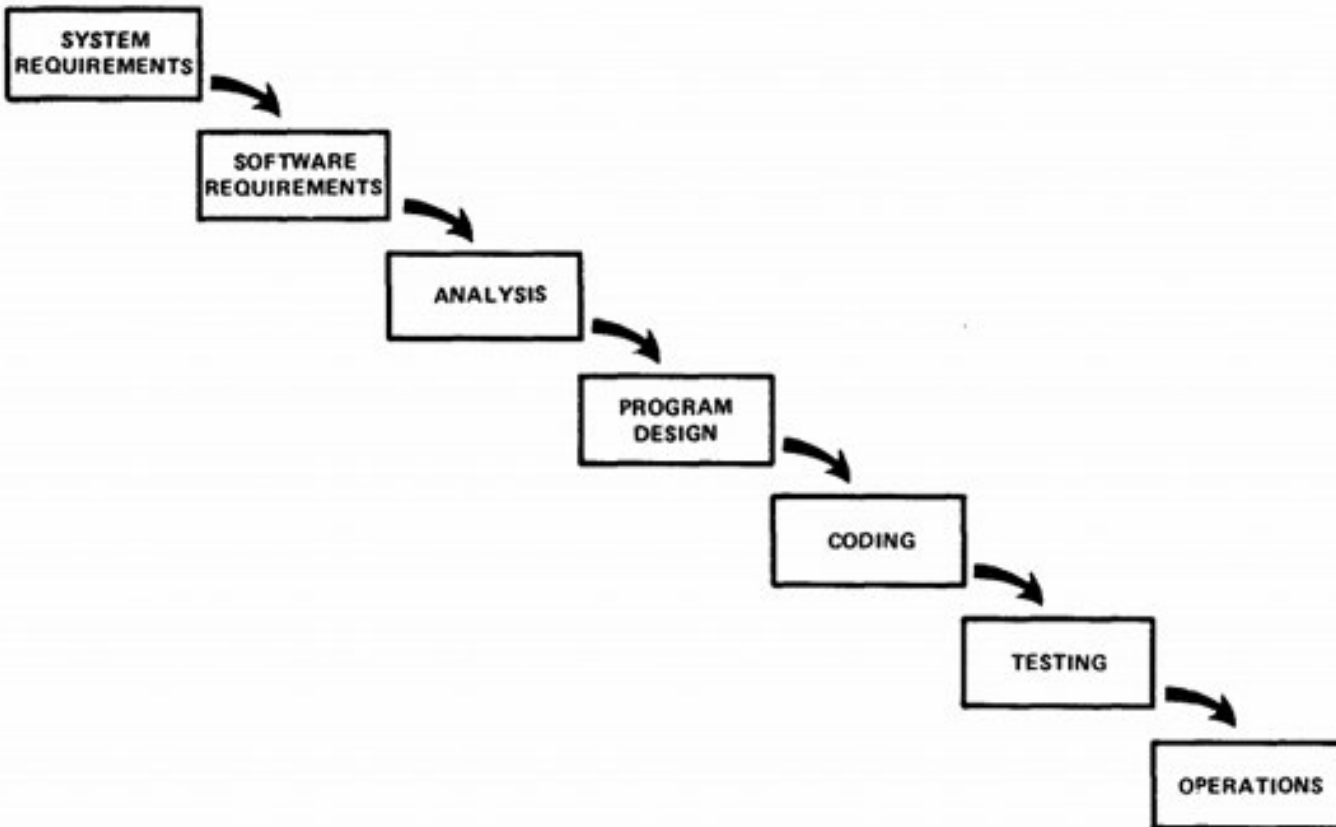


Clasificación de los Modelos Clásicos

- **Los modelos clásicos de la Ingeniería de Software los podemos clasificar en 3 grupos:**
 - **Lineales:** Cada etapa está determinada y sigue una secuencia definida.
 - **Evolutivos:** Se basa en obtener resultados rápidos y su confrontación con los deseos del cliente.
 - **Componentes:** Utiliza partes estandarizadas para lograr la construcción final.

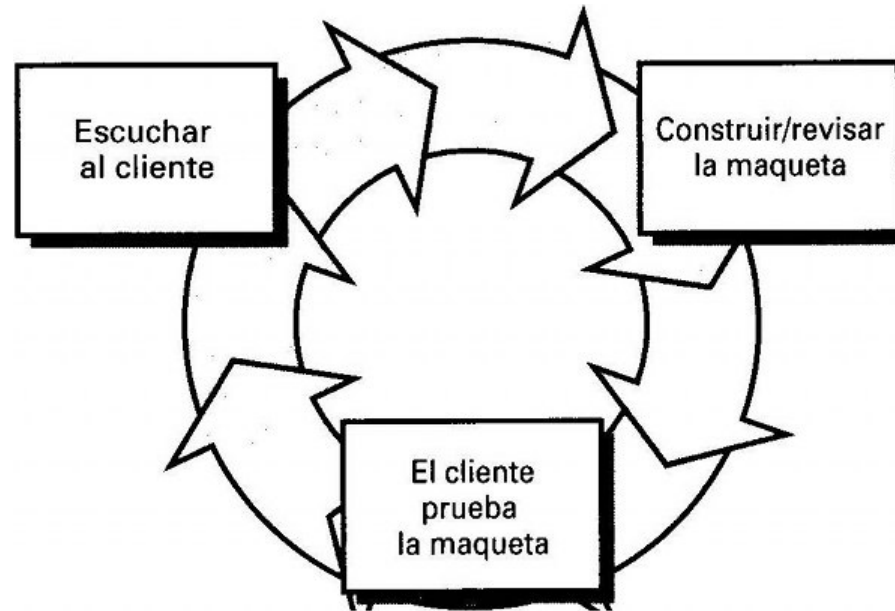
Modelo Lineal (en Cascada)

- Existe desde hace más de 35 años y se divide en fases secuenciales con metas bien definidas.



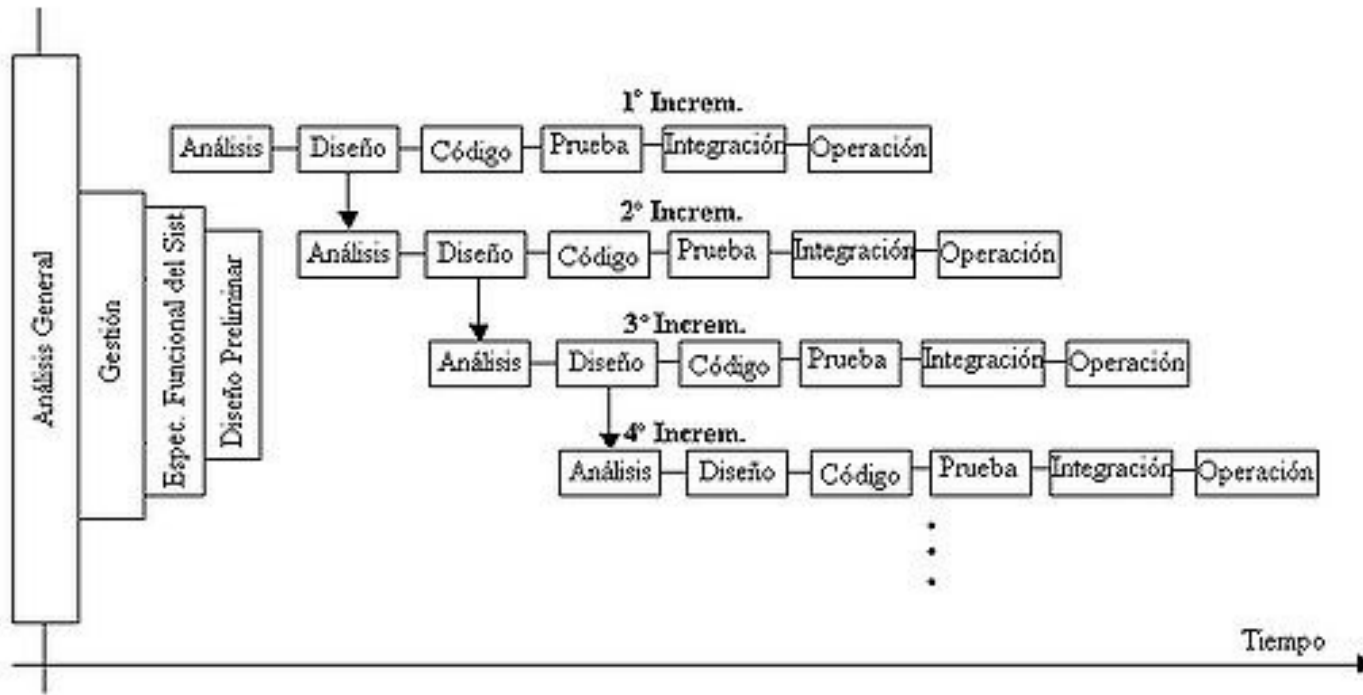
Modelo Evolutivo (Prototipo)

- Establece interacciones cortas de forma tal de mostrarle los avances al cliente.



Modelo Evolutivo (Incremental)

- El incremental es un modelo de tipo evolutivo que está basado en varios ciclos Cascada Realimentados aplicados repetidamente, con una filosofía iterativa.



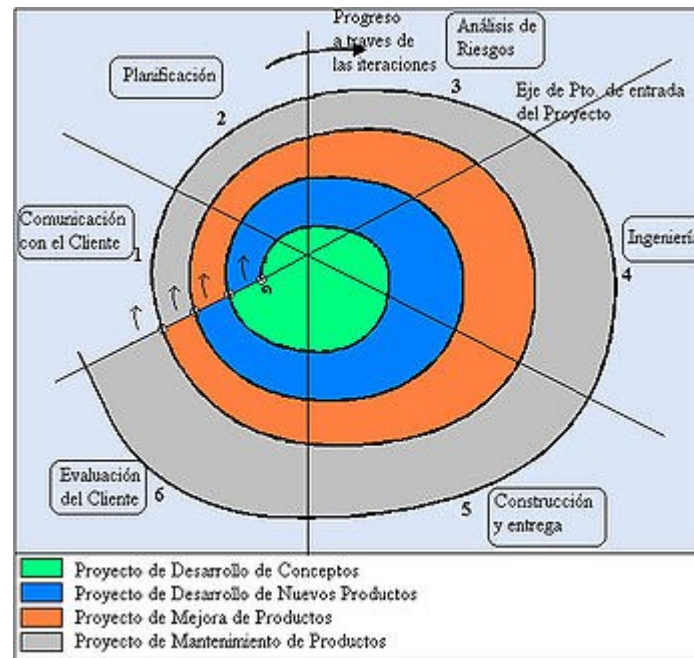
“Métodos Ágiles”, Sebastián Priolo, Editorial Alfaomega 2009

“Software”, http://es.wikipedia.org/wiki/Software#Modelo_iterativo_incremental, agosto 2013



Modelo Evolutivo (Espiral)

- Modelo mixto propuesto por Barry Bohem (lineal con evolutivo) donde se desarrollan versiones de software con mayor funcionalidad por iteración.



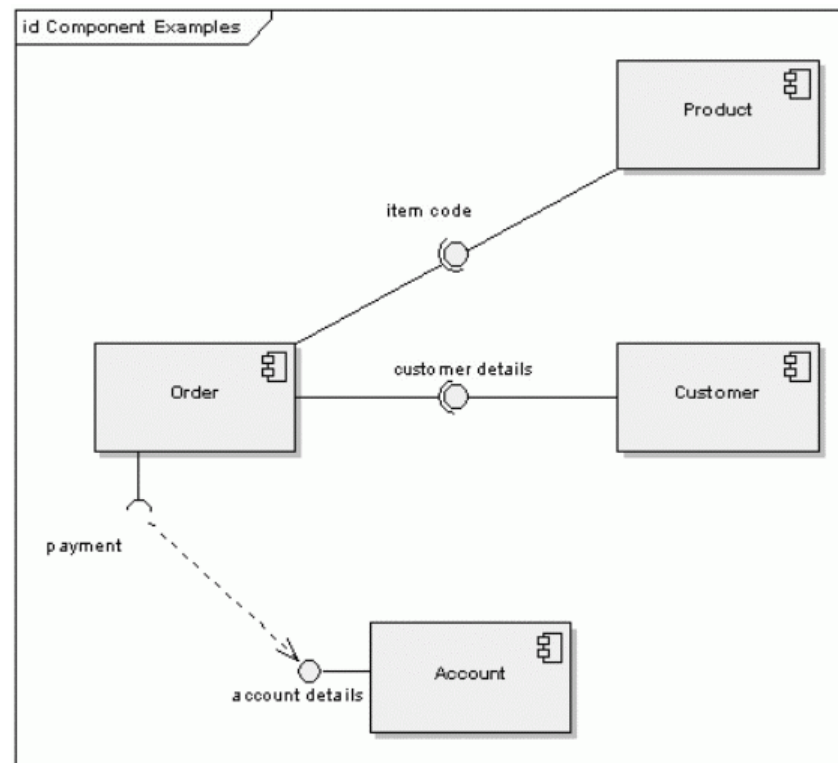
“Métodos Ágiles”, Sebastián Priolo, Editorial Alfaomega 2009

“Software”, http://es.wikipedia.org/wiki/Software#Modelo_iterativo_incremental, agosto 2013



Modelo por Desarrollo de Componentes

- Se enfoca en el desarrollo de componentes de acuerdo al Paradigma de la Programación Orientada a Objetos.



“Métodos Ágiles”, Sebastián Priolo, Editorial Alfaomega 2009

“Ingeniería del Software”, <http://www.monografias.com/trabajos34/ingenieria-software/ingenieria-software.shtml>, agosto 2013



Ingeniería Web



¿Qué es la Ingeniería Web?

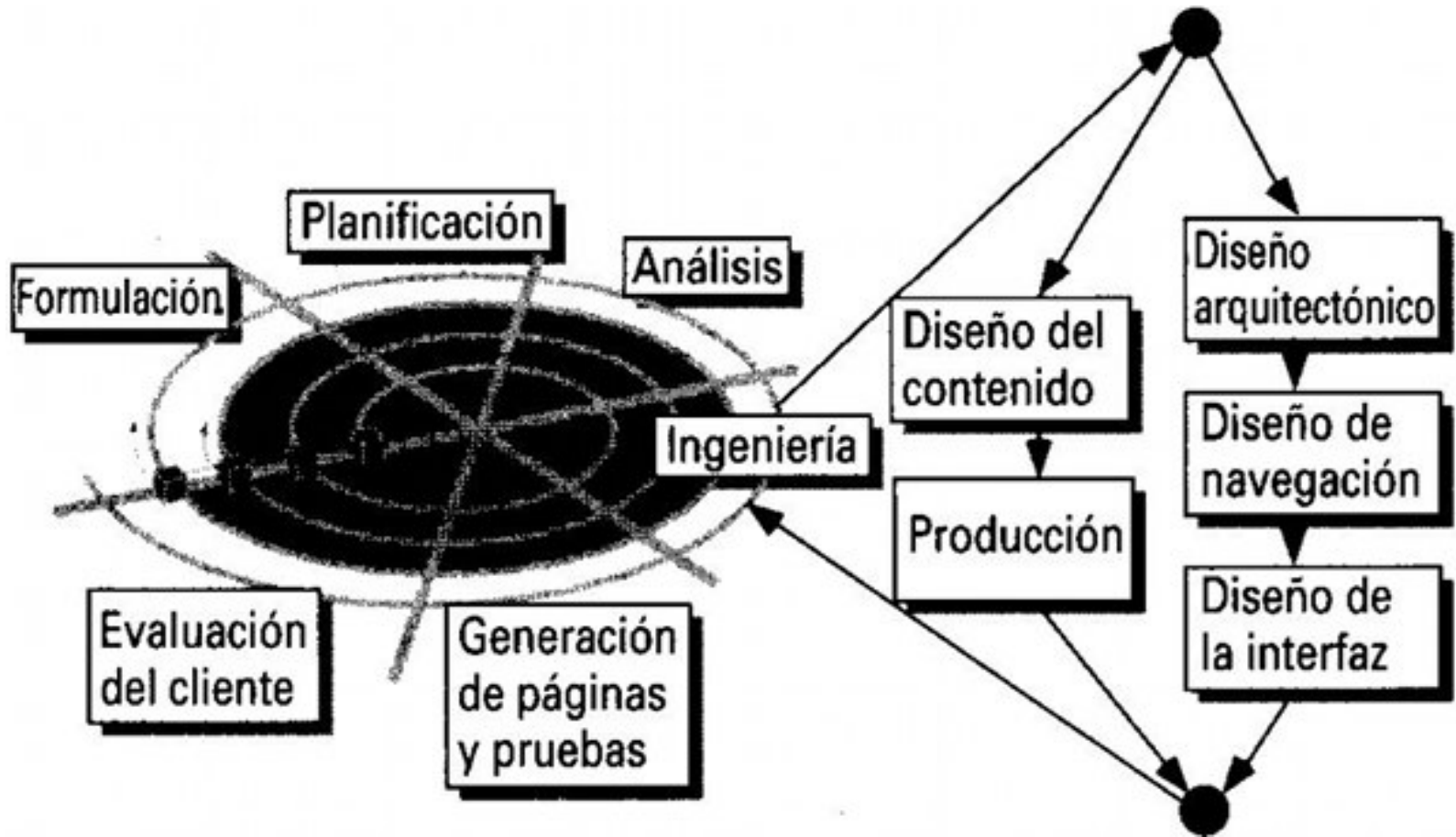
- “La ingeniería web es la aplicación de metodologías sistemáticas, disciplinadas y cuantificables al desarrollo eficiente, operación y evolución de aplicaciones de alta calidad en la World Wide Web”.



Ingeniería Web ***(Enfoque Tradicional)***



Proceso de Ingeniería Web



1) *Formulación*

- **Actividad que identifica las metas y los objetivos de la Aplicación Web.**
- **Establece el ámbito del primer incremento.**

2) Planificación

- **Estima el costo global del proyecto, evalúa los riesgos asociados con el esfuerzo del desarrollo, y define una planificación del desarrollo bien granulada para el incremento final de la aplicación Web.**

3) *Análisis*

- **Establece los requisitos técnicos para la Aplicación Web e identifica los elementos del contenido que se van a incorporar.**
- **También se definen los requisitos del diseño gráfico.**

4) *Ingeniería*

- **Incorpora 2 tareas paralelas: "Diseño de Contenido" y "Producción".**
- **El objetivo de estas tareas es diseñar producir, y/o adquirir todo el contenido de texto, gráfico y video que se vayan a integrar en las aplicaciones Web.**

5) *Generación de Páginas*

- **Actividad de construcción que hace uso de las herramientas automatizadas para la creación de las aplicaciones Web.**
- **El contenido definido en la actividad de Ingeniería se fusiona con los diseños arquitectónicos, de navegación y de la interfaz para elaborar páginas Web.**

6) *Evaluación del Cliente*

- **Se revisa cada incremento producido como parte del proceso de Ingeniería Web.**
- **En este punto es en donde se solicitan cambios que se integran en la siguiente flujo incremental del proceso.**

Ingeniería Web *(Enfoque Ágil)*



Tradicional vs Agiles

- Las Metodologías denominadas **“Tradicional”** centran esfuerzos en documentación y avances o procesos prefijados, mientras que las Metodologías **“Agiles”** perfieren un esquema mas realista, partiendo de que los imprevistos no pueden ser anulados, por lo que el cambio ocurrirá y es bueno adaptarse a éste.
- No importa la metodología, su objetivo sigue siendo la maximización de los recursos y el aseguramiento de la calidad.



TRADICIONAL



AGILE



Metodologías Ágiles

- Son un conjunto de metodologías, a veces denominadas livianas o ligeras, que utilizan prácticas similares, basadas en los resultados, la gente y la interacción.



Manifiesto Agil

- Debido a que existían distintos tipos de metodologías Agiles con características parecidas, los fundadores decidieron organizarse y unificar criterios.
- Con esta idea surgió el “Manifiesto Agil” que se basa en 4 postulados

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.

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

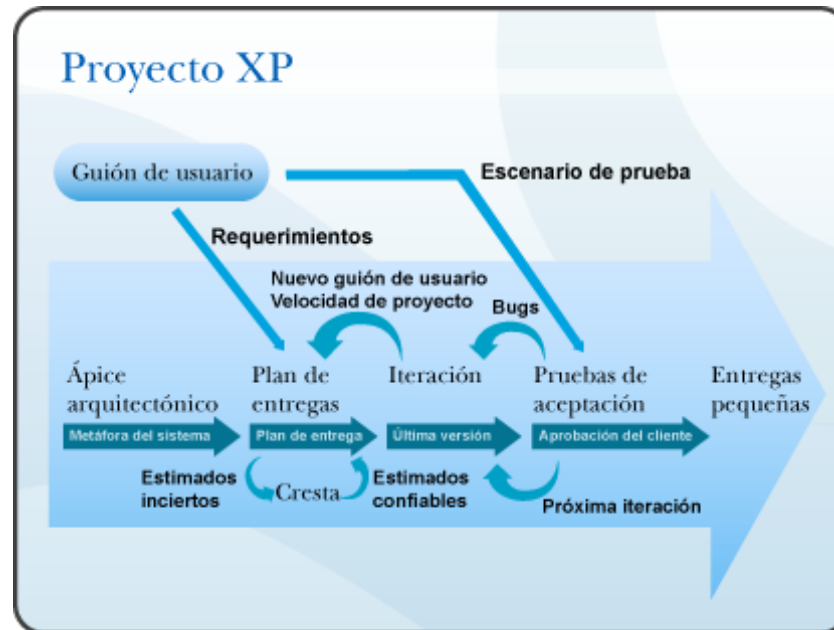
Metodologías Agiles

- Existen varias Metodologías Agiles, siendo las mas conocidas las siguientes:
 - Programación Extrema (XP)
 - Scrum
 - Crystal Clear
 - Feature Driven Development (FDD)
 - Adaptive Software Development (ASD)



Programación Extrema (XP)

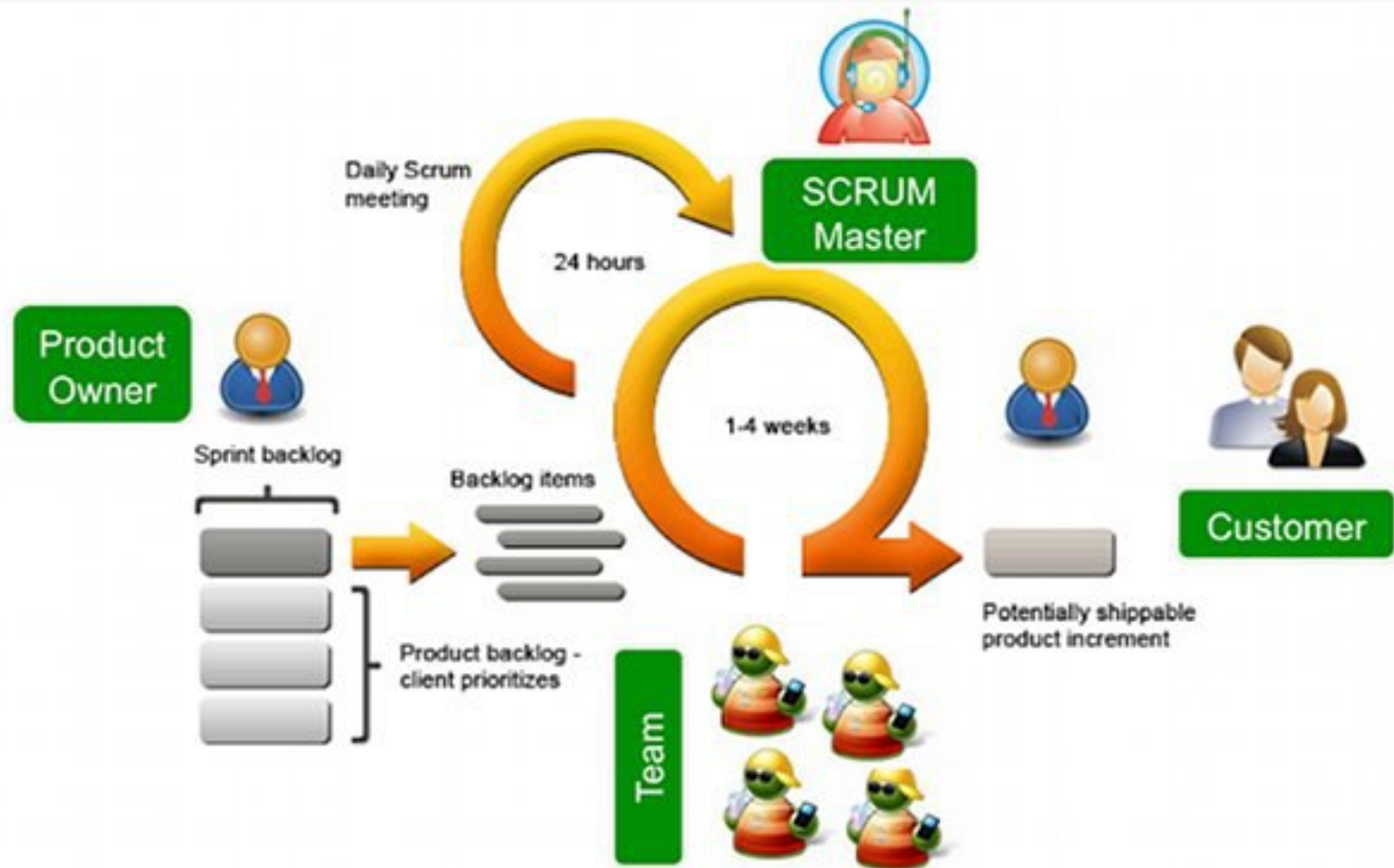
- Es una de las metodologías más usadas en la actualidad, integra el cambio continuamente, produciendo software con mejoras.
- Incorpora al cliente al equipo de trabajo y adopta la programación en pares.



Scrum

- **Scrum es un marco de trabajo para la gestión y desarrollo de software basada en un proceso iterativo e incremental utilizado comúnmente en entornos basados en el desarrollo ágil de software.**
- **Aunque Scrum estaba enfocado a la gestión de procesos de desarrollo de software, puede ser utilizado en equipos de mantenimiento de software, o en una aproximación de gestión de programas:**

Metodo Ágil: SCRUM



Crystal Clear

- **Creada por Alistair Cockburn consiste en 3 conceptos:**
 - **1) Habitabilidad: Implica que debemos buscar metodologías que nos agraden al igual que el equipo y el proyecto.**
 - **2) Eficiencia: Refinar y mejorar proyecto a proyecto.**
 - **3) Seguridad para el Proyecto: Cumplir las expectativas de los clientes.**

Feature Driven Development (FDD)

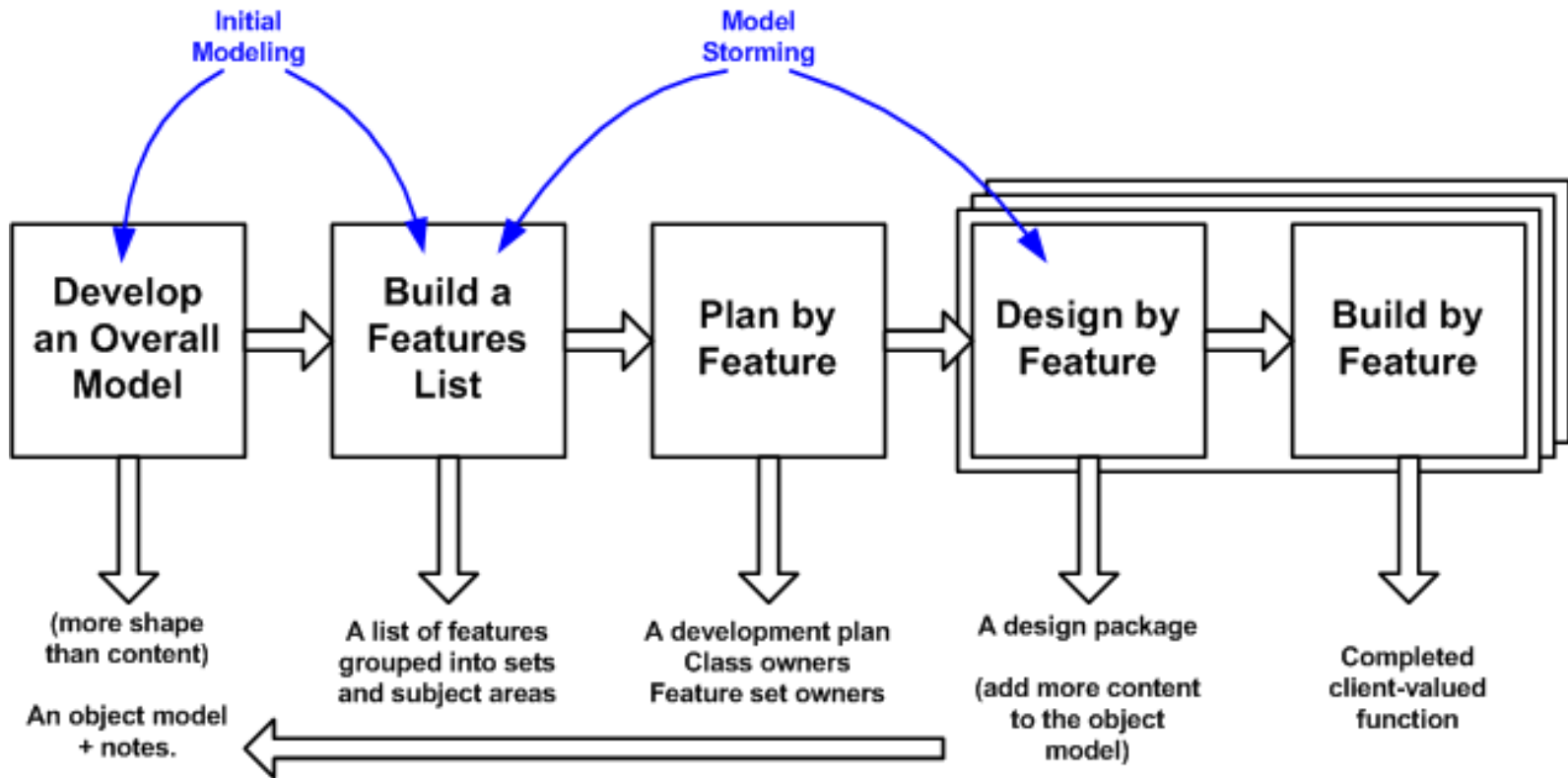
- **Enfatiza cuestiones de calidad y define claramente entregas tangibles y formas de evaluación del progreso.**
- **Consiste en 5 procesos secuenciales durante los cuales se diseña y construye el sistema.**

Feature Driven Development (FDD)

- **Los 5 pasos son los siguiente:**
 - **1) Desarrollo del modelo general.**
 - **2) Construcción de la lista de rasgos.**
 - **3) Planeamiento por rasgo.**
 - **4) Diseño por rasgo.**
 - **5) Construcción por rasgo.**

- **A diferencia de otros procesos Agiles, no cubre todo el ciclo de vida sino sólo las fases de diseño y construcción.**

Feature Driven Development (FDD)



Copyright 2002-2005 Scott W. Ambler
Original Copyright S. R. Palmer & J.M. Felsing

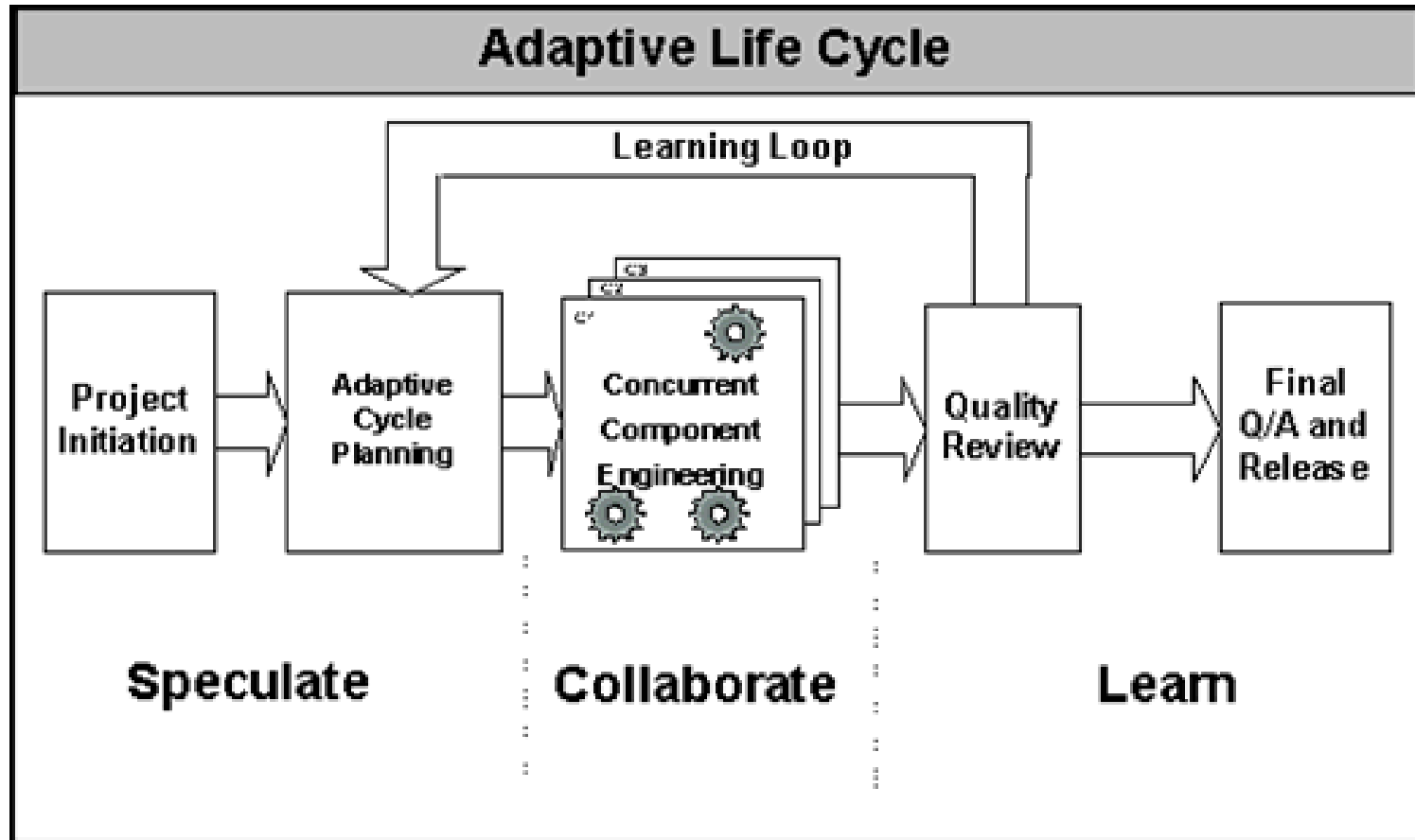
Adaptive Software Development (ASD)

- **Se basa en la adaptación continua a circunstancias cambiantes.**
- **No tiene un determinado ciclo de planificación – diseño – construcción del software, sino que se trata de especular – colaborar – aprender.**
- **Sus características se pueden resumir en:**
 - **Orientada a los componentes.**
 - **Tolerante a los cambios.**

Adaptive Software Development (ASD)

- **Su ciclo de vida tiene 3 etapas:**
 - **1) Especulación: Se inicia el proyecto y se planifican las características del software.**
 - **2) Colaboración: Se desarrollan las características.**
 - **3) Aprendizaje: Se revisa su calidad y se entrega al cliente.**

Adaptive Software Development (ASD)





Rogelio Ferreira Escutia

***Instituto Tecnológico de Morelia
Departamento de Sistemas y Computación***

Correo: rogeplus@gmail.com

rogelio@itmorelia.edu.mx

Página Web: <http://antares.itmorelia.edu.mx/~kaos/>

<http://www.xumarhu.net/>

Twitter: <http://twitter.com/rogeplus>

Facebook: <http://www.facebook.com/groups/xumarhu.net/>