

# Ciclo de Vida y Modelos de Desarrollo

---

Teoría de Sistemas

## Temario

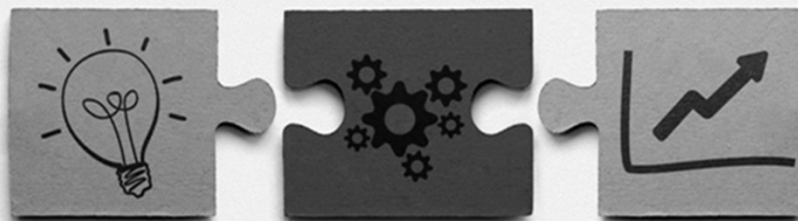
---

Bases metodológicas para el estudio y diseño de sistemas.

Ciclo de vida y modelos de desarrollo.

## Ciclo de vida y modelos de desarrollo

En la incorporación y puesta en marcha de sistemas de información y comunicación, intervienen diferentes tecnologías. Las principales son:

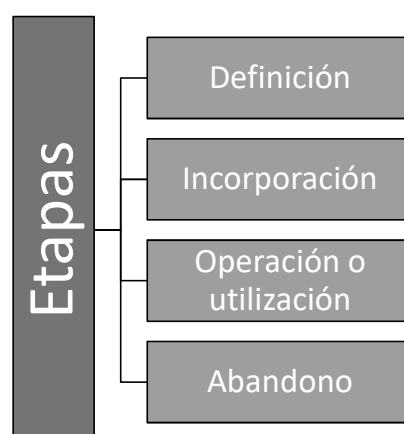


## CICLO DE VIDA DE UNA APLICACIÓN

## Ciclo de vida de una aplicación

Ciclo de vida: etapas por las que pasa un sistema a lo largo de su vida, desde su concepción hasta el abandono en su uso.

## Ciclo de vida de una aplicación



## Definición

La etapa de Definición incluye:

- el establecimiento de la **visión externa del sistema**,
- el establecimiento de **límites y alcances del sistema**,
- la **estimación del costo y esfuerzo** requerido y,
- la **decisión** de incorporarlo.

## Definición

Esta etapa permite:

- Identificar las necesidades del usuario.
- Determinar el alcance del proyecto, enunciando sus funciones y límites.
- Identificar alternativas de realización.
- Realizar el cálculo de costo-beneficio y el plan global de trabajo.

En esta etapa la participación del usuario es trascendente.

## Incorporación

Incluye todas las actividades necesarias para la **adquisición y/o construcción y, puesta en marcha.**

Comprende las siguientes fases o etapas:

Organización y planeamiento.

Ejecución y control:

- análisis y diseño
- adquisición/construcción y prueba
- puesta en marcha

Finalización.

## Incorporación

La **puesta en marcha** incluye principalmente las tareas de:

- Entrenamiento a usuarios
- Conversión y/o vuelco de datos
- Instalación de hardware y relacionados
- Prueba operativa, seguimiento y ajustes
- Operación inicial del sistema

## Operación o utilización

La **utilización** corresponde a la **vida útil del sistema**, durante la cual estará sometido a *mantenimiento*.

Ampliaciones

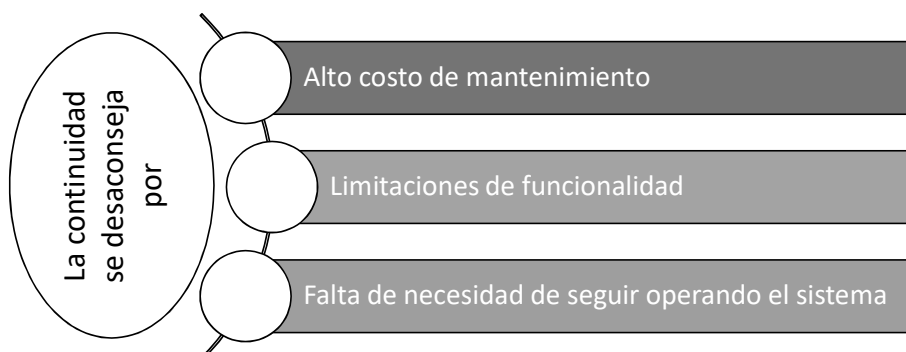
Correcciones



El mantenimiento, en particular cuando requiere aplicación de significativa cantidad de recursos, debe tratarse como un proyecto en sí mismo.

## Operación o utilización

Durante la etapa de operación del sistema una de las actividades distintivas es la de **resolver la continuidad o el abandono**.



# Abandono

Por último, **el sistema es dejado de lado**, siendo o no reemplazadas sus funcionalidades por otro.

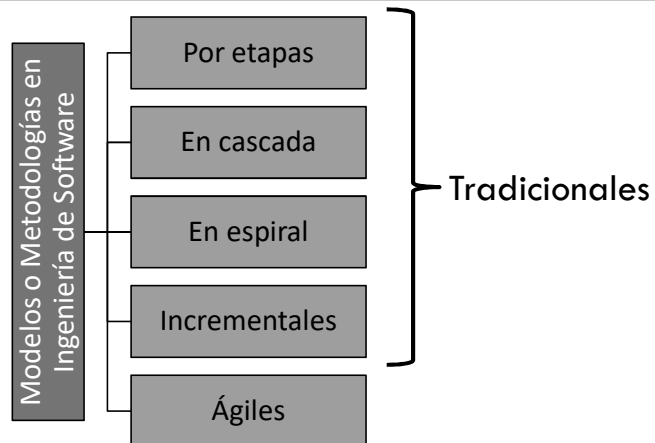
En caso de reemplazo, las actividades de transición correspondientes al abandono se deben considerar en el plan del nuevo proyecto.



# METODOLOGÍAS DE DESARROLLO

## Metodologías o modelos de desarrollo

Un modelo o metodología para el desarrollo de software es una **representación abstracta de un proceso**.



## Modelo por etapas



La característica distintiva de este modelo es la secuencialidad.



## Modelo por etapas

### Ventaja

- Resulta sumamente fácil conceptualizar qué tareas atender en cada etapa.

### Desventajas

- Requiere la **documentación previa y completa de los requerimientos**.
- El **usuario** deja de tener contacto con el desarrollo en la etapa inicial y sólo la recupera en la implementación.

## Modelo por etapas

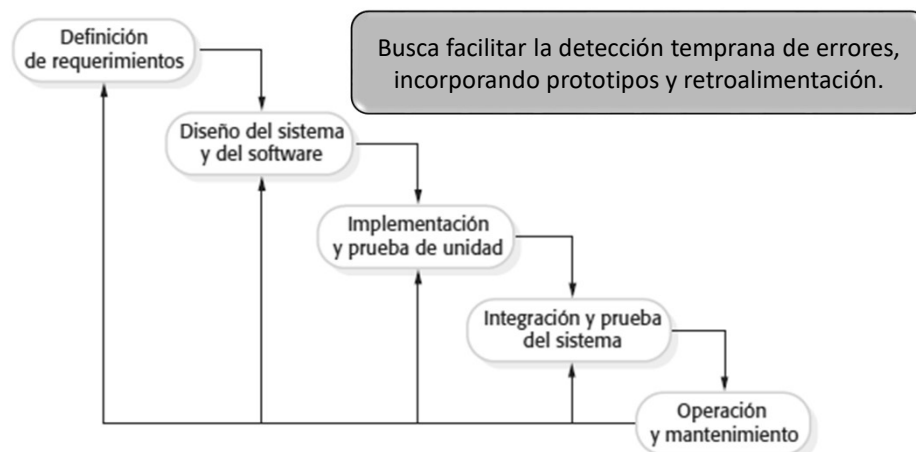
### Mayor debilidad

- Mantenimiento previo a la operación

La **detección tardía de necesidades** de modificación provoca un significativo aumento de costos y una demora en la implementación.



## Modelo en Cascada



## Modelo evolutivo o en espiral

El modelo en espiral del proceso del software que originalmente fue propuesto por Boehm (1988).

El modelo en espiral consta de **etapas o fases** que se van realizando en manera **continua y cíclica**.



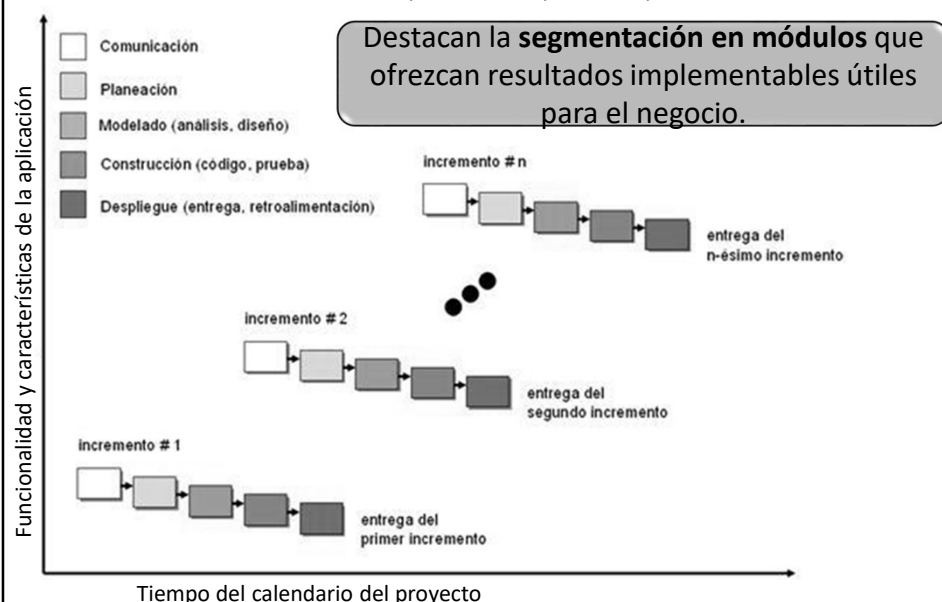
Barry Boehm  
(n. 1935)

## Modelo evolutivo o en espiral

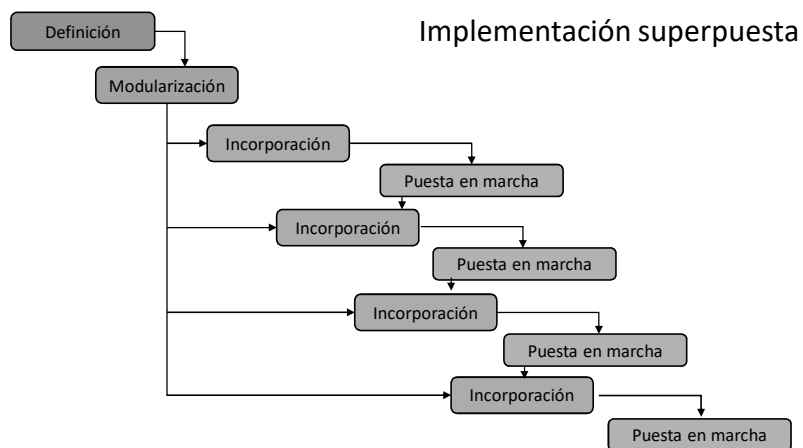


Busca reducción de riesgos de modificaciones, enfatizando prototipos e incorporando **paralelismo** y **modularidad**.

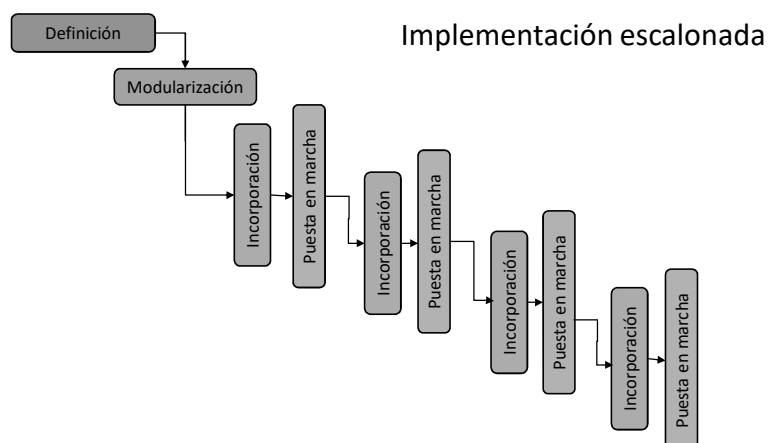
## Modelos incrementales impulsados por un producto utilizable



## Modelos incrementales impulsados por un producto utilizable



## Modelos incrementales impulsados por un producto utilizable



## Modelos ágiles

Si bien muchos de los **métodos tradicionales** incluyen iteraciones y paralelismos, **en su esencia son prescriptivos y formales.**

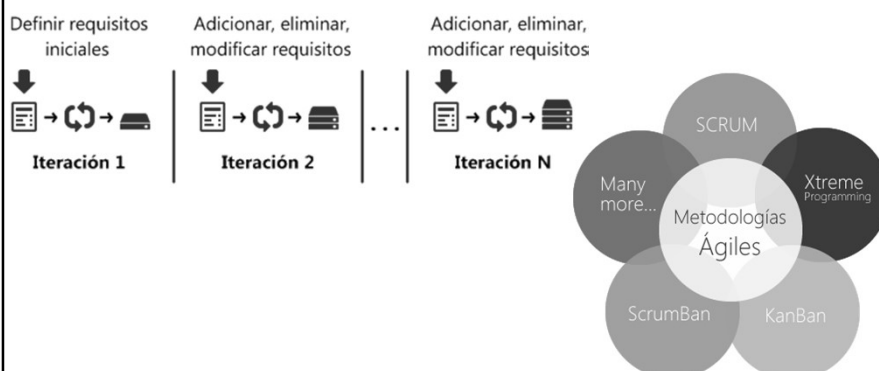
Todos ellos consideran una definición previa de límites y alcances, el “relevamiento” de las necesidades del usuario y que el “especialista” diseñe el sistema para que luego sea construido e implementado.



En cambio, los **procesos ágiles de desarrollo de software** intentan evitar los tortuosos y burocráticos caminos de las metodologías tradicionales **enfocándose en la gente y los resultados.**

## Modelos ágiles

Se entiende como desarrollo ágil de software a un paradigma de desarrollo de software basado en procesos ágiles.



## Principios de los Métodos Ágiles

Principio	Descripción
Participación del cliente	Los clientes deben intervenir estrechamente durante el proceso de desarrollo. Su función consiste en ofrecer y priorizar nuevos requerimientos del sistema y evaluar las iteraciones del mismo.
Entrega incremental	El software se desarrolla en incrementos y el cliente especifica los requerimientos que se van a incluir en cada incremento.
Personas, no procesos	Tienen que reconocerse y aprovecharse las habilidades del equipo de desarrollo. Debe permitirse a los miembros del equipo desarrollar sus propias formas de trabajar sin procesos establecidos.
Adoptar el cambio	Esperar a que cambien los requerimientos del sistema y, de este modo, diseñar el sistema para adaptar dichos cambios.
Mantener simplicidad	Enfocarse en la simplicidad tanto en el software a desarrollar como en el proceso de desarrollo. Siempre que sea posible, trabajar de manera activa para eliminar la complejidad del sistema.

## RAD

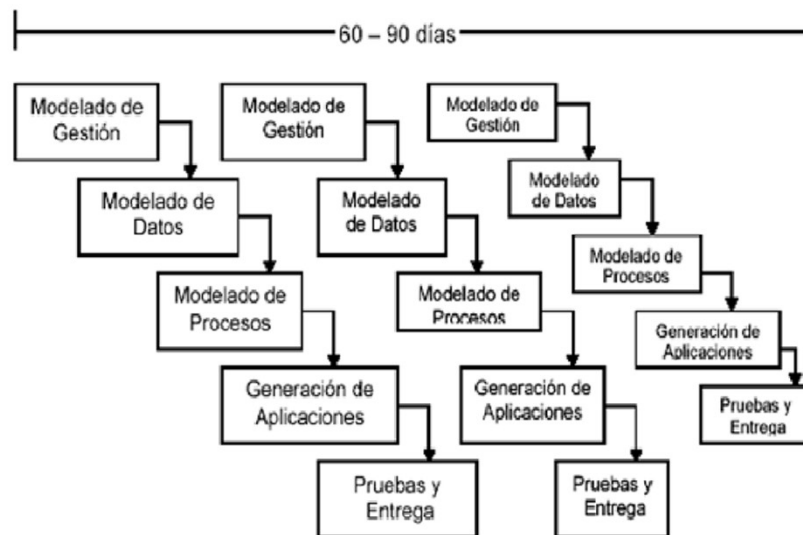
El desarrollo rápido de aplicaciones o RAD (Rapid Application Development) es un proceso de desarrollo de software, iniciado por James Martin en 1980.

El método comprende el **desarrollo iterativo**, la **construcción de prototipos** y el **uso de utilidades CASE** (*Computer Aided Software Engineering, Ingeniería de Software Asistida por Computadora*).

### Herramientas CASE:

Son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el costo de las mismas en términos de tiempo y de dinero.

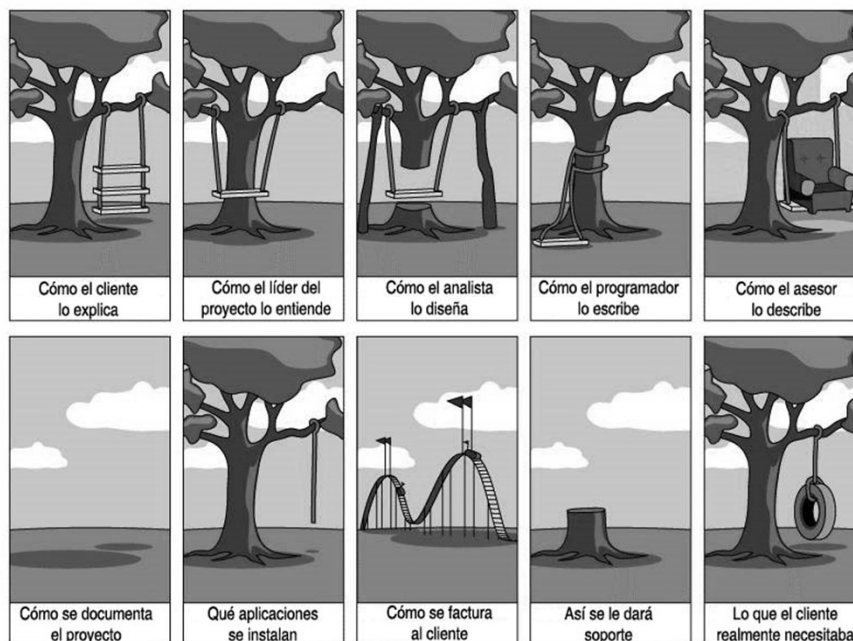
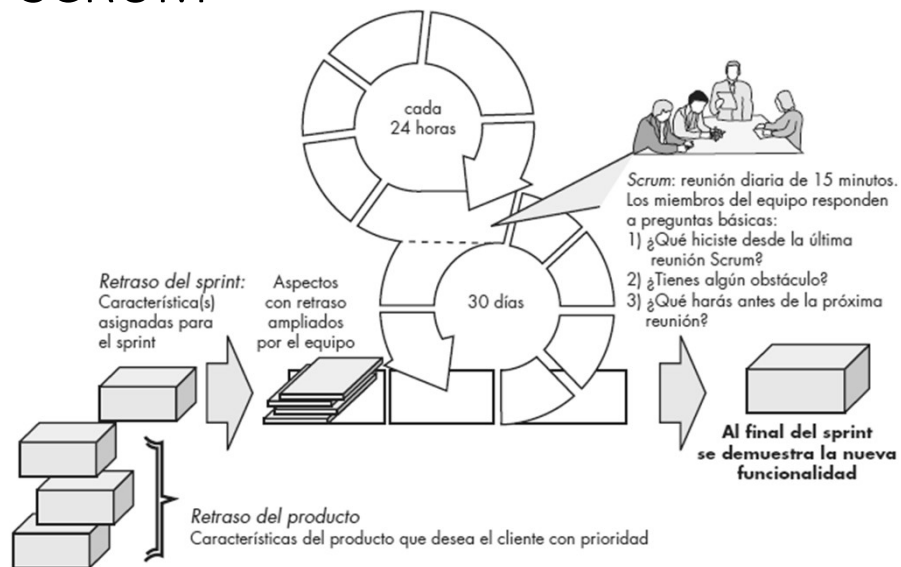
## RAD



## Programación Extrema (XP)



# SCRUM





## Bibliografía

---

BRIANO, Juan Carlos y otros. Sistemas de información gerencial: tecnologías para agregar valor a las organizaciones. 1ª Edición. Editorial Prentice-Hall, 2011.

PRESSMAN, Roger S. Ingeniería del software. Un enfoque práctico. Madrid: Mc Graw Hill. 7ª Edición, 2010.

## Actividad

---

**Caso 1:** Considere el desarrollo de un sistema cuyo dominio de aplicación es conocido, sus objetivos y requerimientos funcionales son estables y simples de comprender desde un principio, la tecnología a utilizar ya está determinada y es bien conocida por el equipo de desarrollo. ¿Qué metodología elegiría para el desarrollo de dicho sistema?

**Caso 2:** Considere ahora el desarrollo de un sistema cuyo dominio de aplicación no es muy conocido por el equipo de desarrollo. En este caso, el cliente tampoco tiene muy claro qué es lo que quiere, de manera que los objetivos y requerimientos funcionales del sistema son inestables y difíciles de comprender. Además, el equipo de desarrollo va a utilizar una tecnología que le resulta completamente nueva. Discuta qué modelo de ciclo de vida es más apropiado y qué etapas se deberían utilizar para desarrollar este sistema.