



# Modelos de proceso de software

# Temario

- Introducción
- Modelos de proceso
- Procesos guiados por planes vs. procesos ágiles

# Proceso

- Serie de pasos con actividades, restricciones y recursos, que producen una salida de cierto tipo (Pfleeger).
- Un proceso es un conjunto de acciones y actividades, relacionadas entre sí, que se realizan para crear un producto, resultado o servicio predefinido (PMI).
- Un proceso de software es una serie de actividades relacionadas que conduce a la elaboración de un producto de software (Sommerville).
- Cuando el proceso involucra la construcción de un producto, a veces se menciona como **ciclo de vida** (del producto).

# Modelo de proceso

- Modelo de proceso de software:
  - representación abstracta de un proceso
  - descripción a partir de una perspectiva en particular
- Pueden usarse como:
  - Prescripciones de la forma en que el desarrollo de software **debería llevarse a cabo**
  - Descripciones de la forma en que el desarrollo **se lleva a cabo realmente**
- Deben ser **adaptados** a la realidad particular de cada proyecto.
- Pueden ser **combinados** para lograr mejor adopción.

# Modelo de proceso y ciclo de vida

- La preocupación por el «proceso» (fin de los 80) es más reciente que la definición del «ciclo de vida» (fin de los 60).
- En general se asocia a la noción de modelo de proceso un mayor detalle y precisión.
- Los modelos previos presentan en general poco nivel de detalle y fueron propuestos originalmente como modelos de ciclo de vida.

# Modelos de proceso de sw

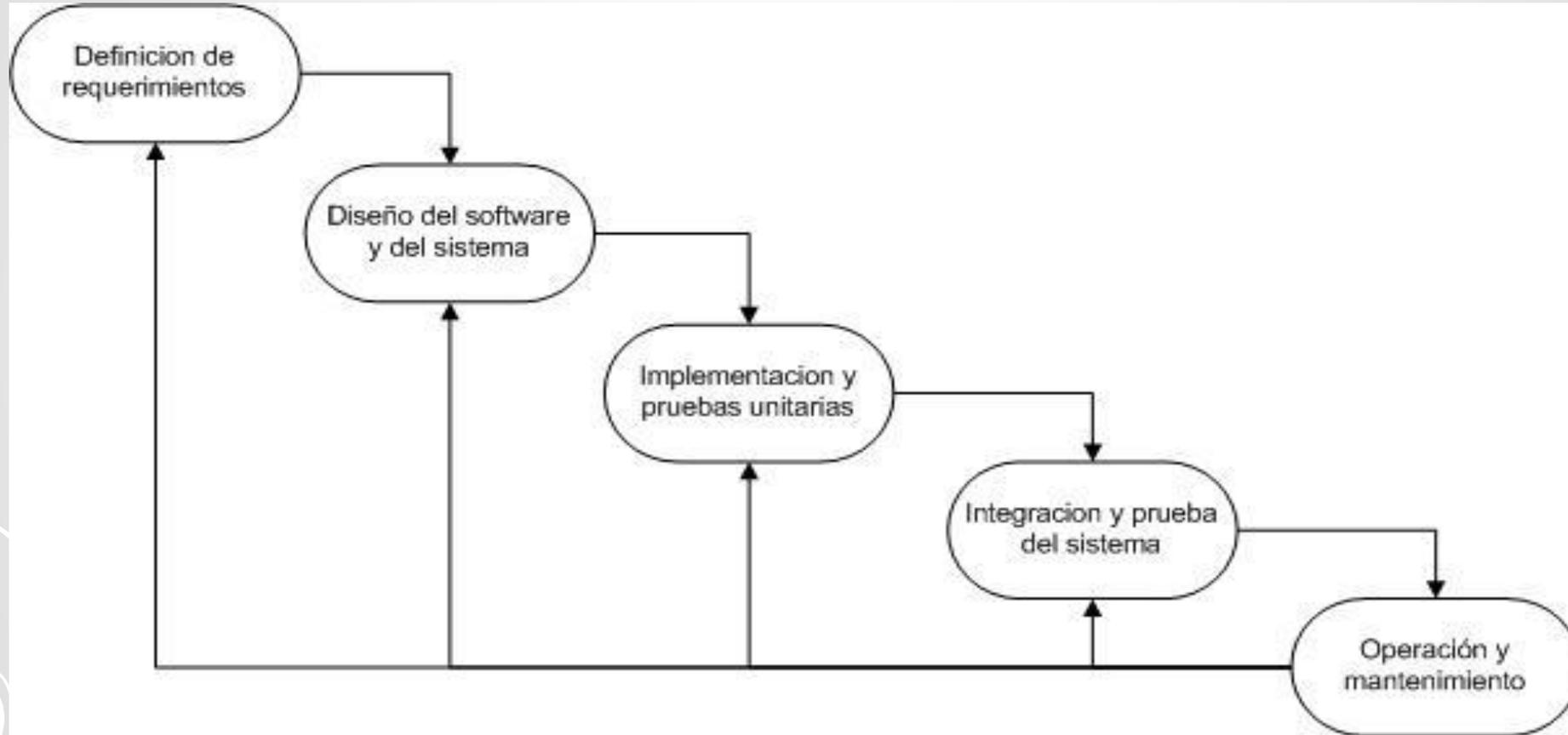
- Modelo en cascada
- Modelo en cascada con prototipado
- Modelo en V
- Modelo en fases:
  - con liberaciones parciales
  - con evaluaciones parciales
- **Modelo en espiral**

# Modelos de proceso de sw

- **Modelo en cascada**
- Modelo en cascada con prototipado
- Modelo en V
- Modelo en fases:
  - con liberaciones parciales
  - con evaluaciones parciales
- **Modelo en espiral**

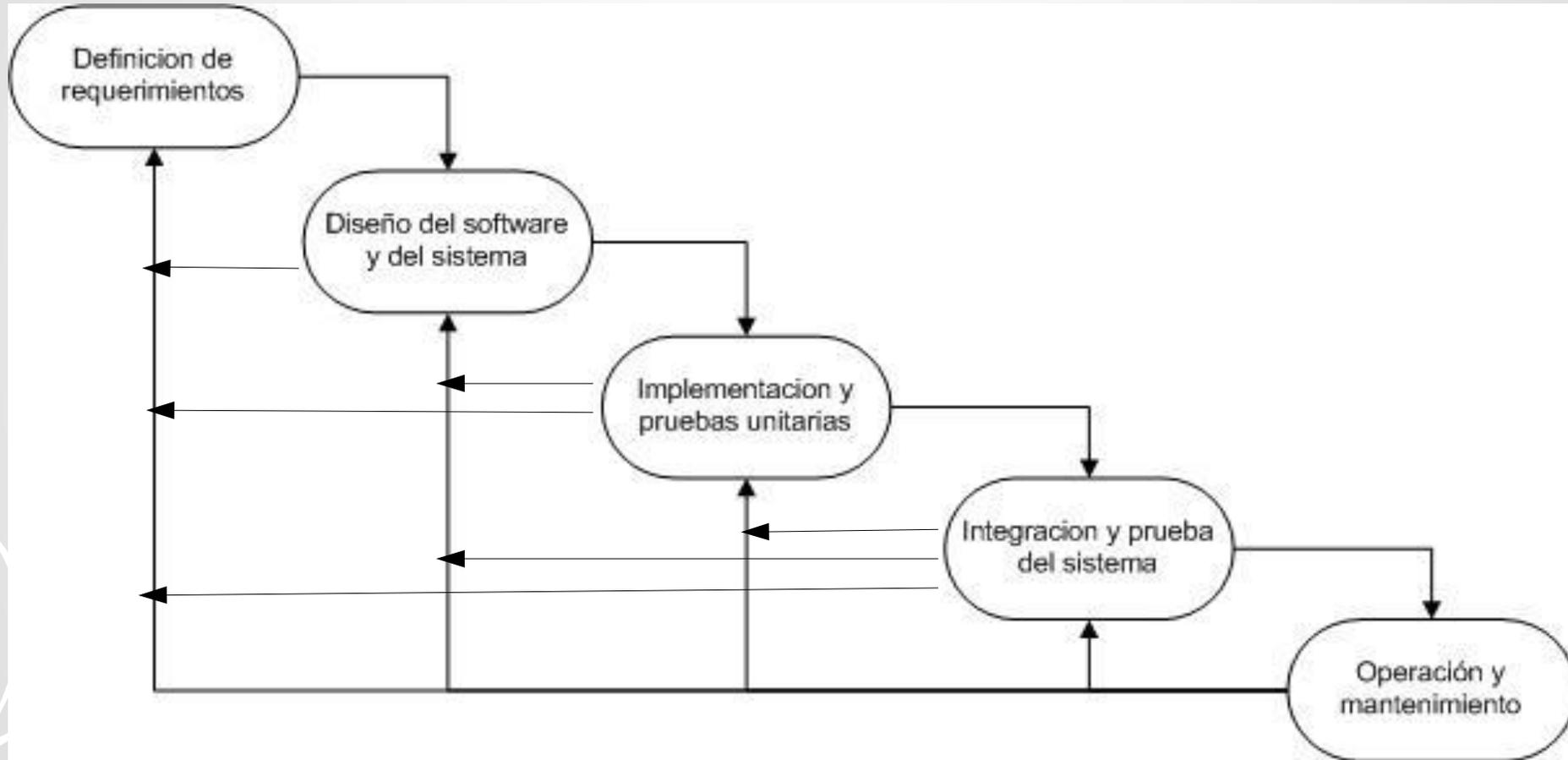


# Modelo en cascada

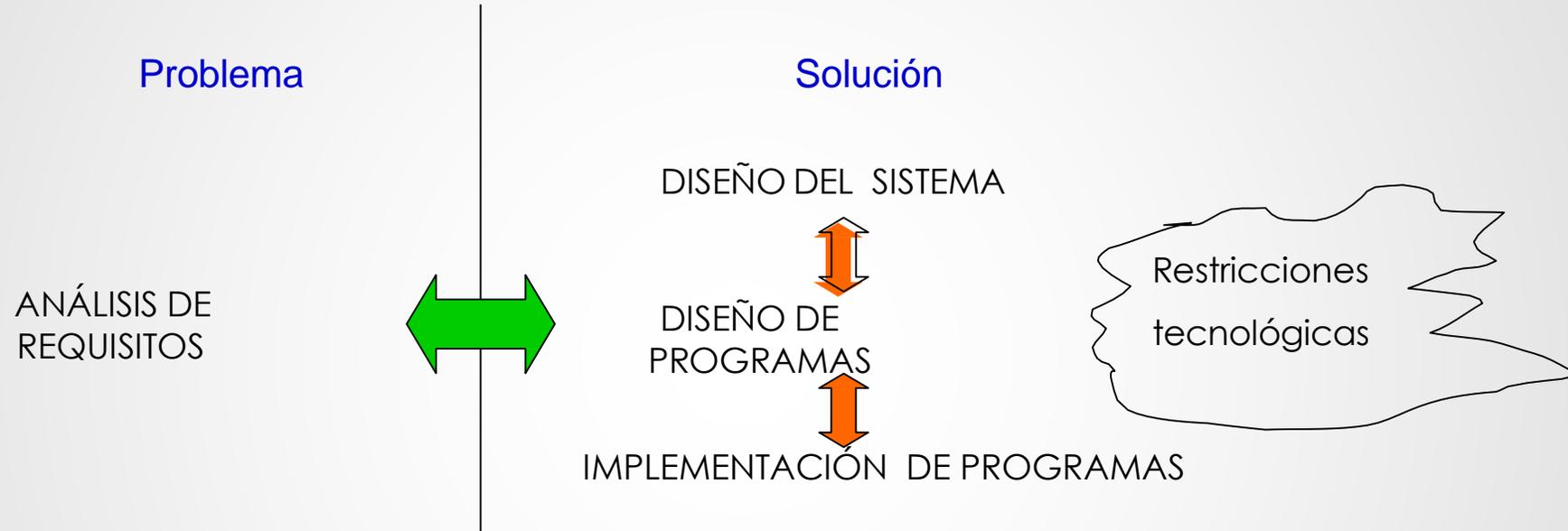


\* Ejemplo tomado de [Sommerville, 2011].

# Lo que realmente ocurre...



# Espacios del problema y de la solución



- Intentar solucionar un problema a menudo aclara el problema
- Evaluar los atributos de calidad externos del software (solución) requiere que exista «comportamiento»

# Problemas del modelo de proceso en cascada

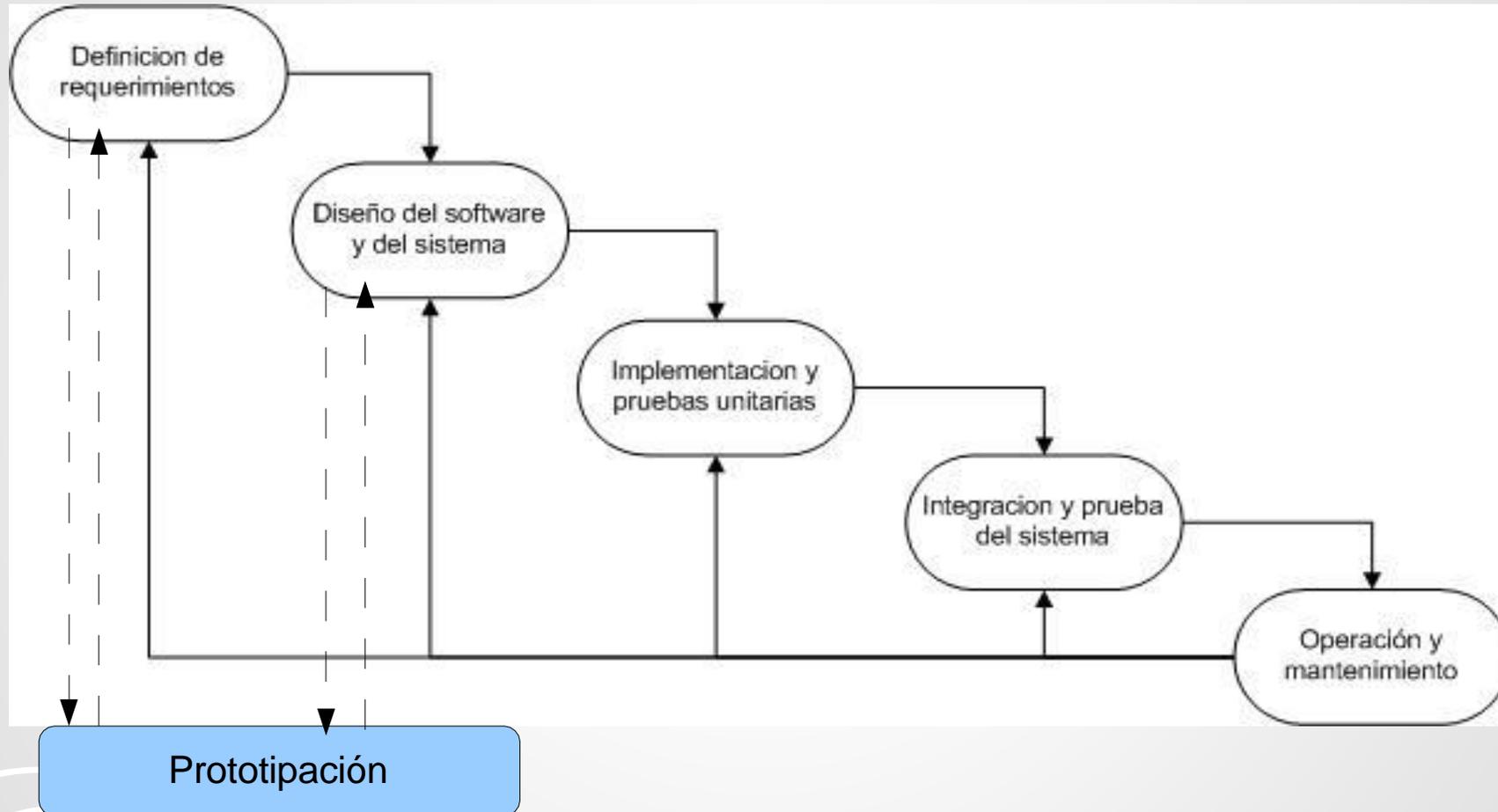
- Riesgo de que al ver el producto funcionando, el cliente diga: «No es lo que preciso».
- Requisitos
  - ¿Cuándo están completos los requisitos?
  - ¿Serán consistentes?
- No permite manejar bien cambios en los requisitos
  - ¿por qué pueden cambiar?
  - El costo del cambio es mayor a medida que avanza el proyecto
- Entrega tardía de valor

# Modelos de proceso de sw

- Modelo en cascada
- **Modelo en cascada con prototipación**
- Modelo en V
- Modelo en fases:
  - con liberaciones parciales
  - con evaluaciones parciales
- Modelo en espiral



# Modelo en cascada con prototipación

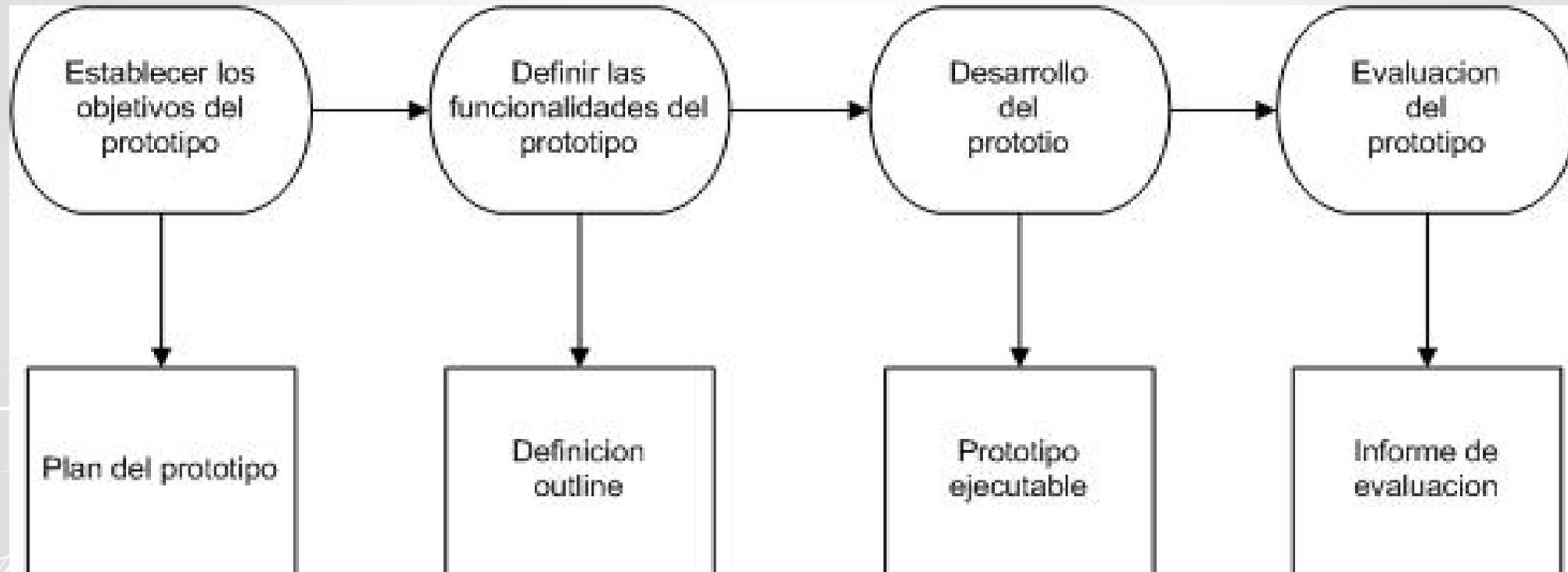


\* Figura tomada de [Pleeger y Atlee, 2010].

# Prototipos

- Un prototipo es una **versión inicial/parcial** del sistema usada para **demostrar conceptos** y probar **opciones** de diseño.
- Un prototipo puede ser usado en:
  - el proceso de ingeniería de requisitos para ayudar en la obtención de requisitos y en la validación de los mismos;
  - en los procesos de diseño para explorar opciones y desarrollar un diseño de interfaz de usuario.
- Puede realizarse en lenguajes de prototipado rápido, en papel, etc.
- Pueden ser:
  - **desechables** (se descartan luego de su evaluación)
  - **o evolutivos** (evolucionan hacia el sistema final)

# Modelo de construcción de prototipos



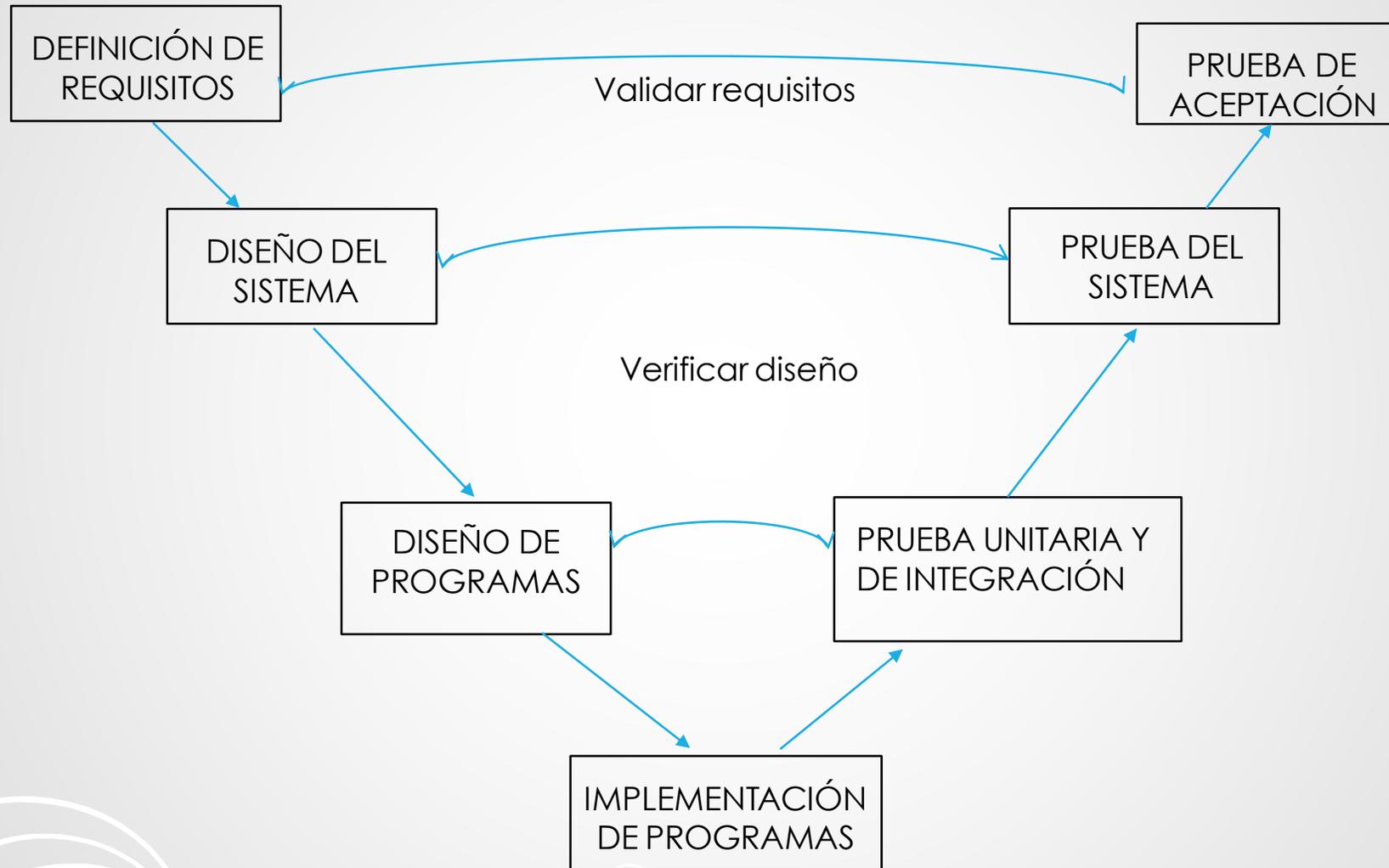
\* Figura tomada de [Sommerville, 2011].

# Modelos de proceso de sw

- Modelo en cascada
- Modelo en cascada con prototipado
- **Modelo en V**
- Modelo en fases:
  - con liberaciones parciales
  - con evaluaciones parciales
- Modelo en espiral

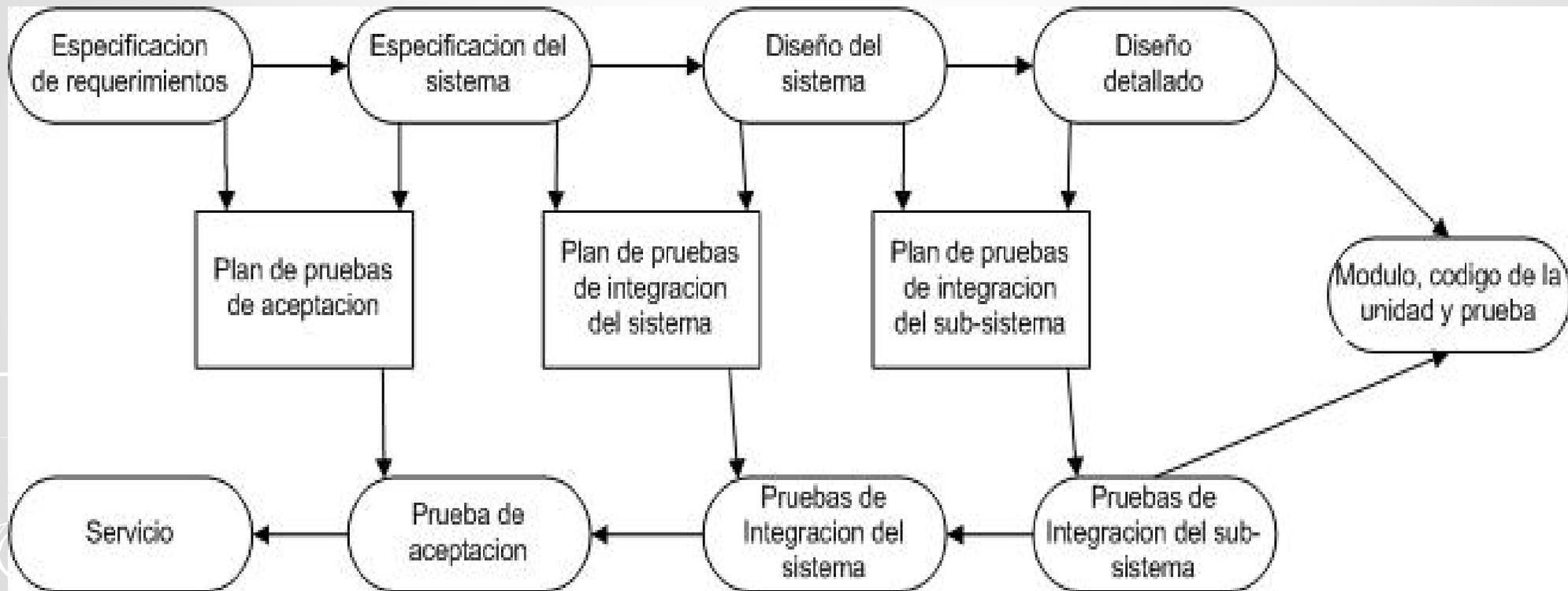


# Modelo en V



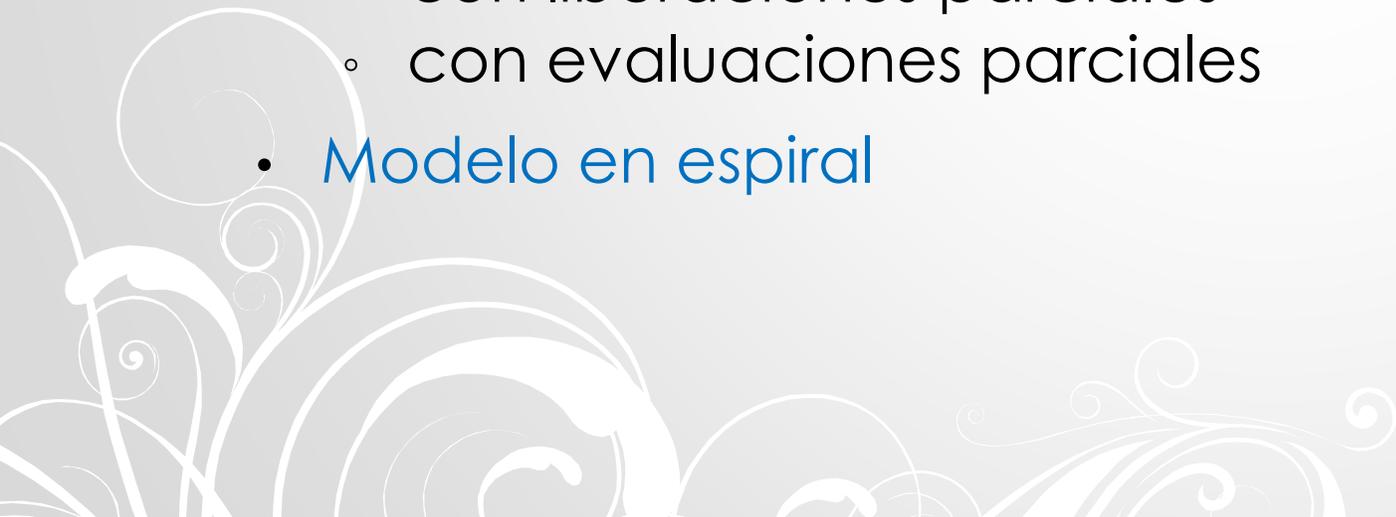
\* Figura tomada de [Pfleeger y Atlee, 2010].

# Un modelo para las pruebas consistente con el modelo en V



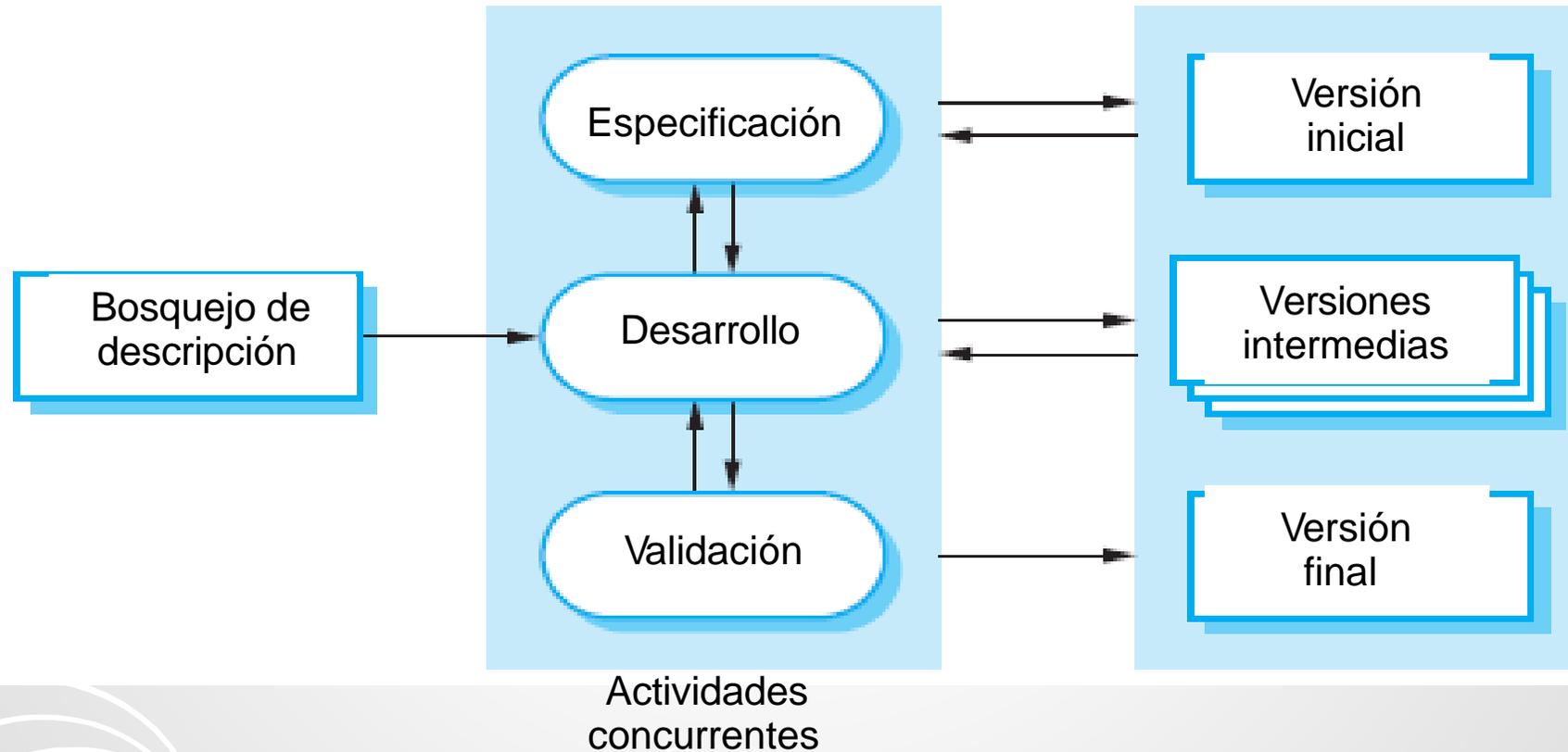
# Modelos de proceso de sw

- Modelo en cascada
- Modelo en cascada con prototipación
- Modelo en V
- **Modelo en fases:**
  - con liberaciones parciales
  - con evaluaciones parciales
- Modelo en espiral



# Modelos de desarrollo en fases

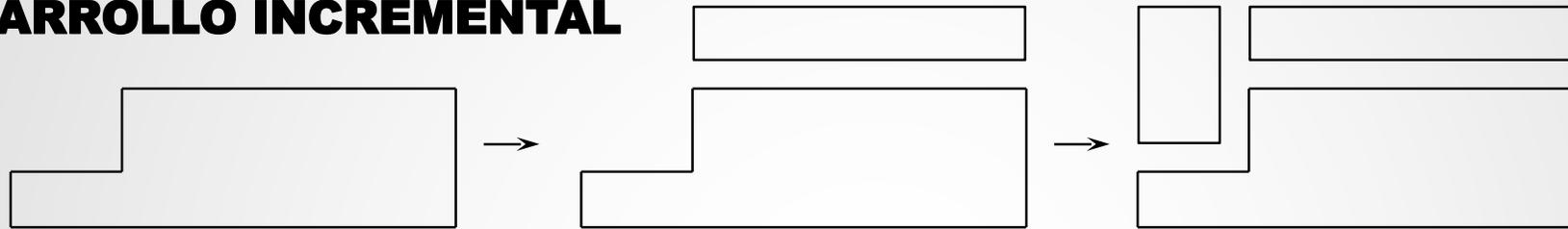
- Desarrollar una implementación inicial, que evolucione a través de diferentes versiones hasta el sistema final.



\* Figura tomada de [Sommerville, 2011].

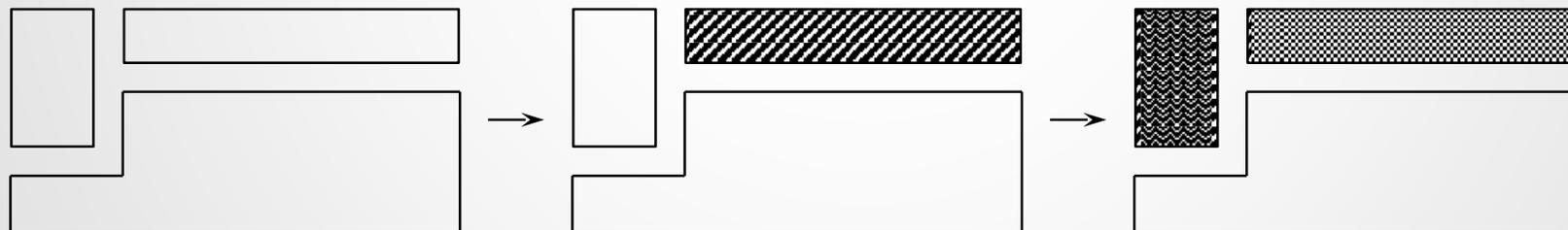
# Incrementos e iteraciones

## DESARROLLO INCREMENTAL



Cada incremento incorpora nueva funcionalidad.

## DESARROLLO ITERATIVO



En cada iteración se cambia funcionalidad de algún subsistema.

\* Figura tomada de [Pleeger y Atlee, 2010].

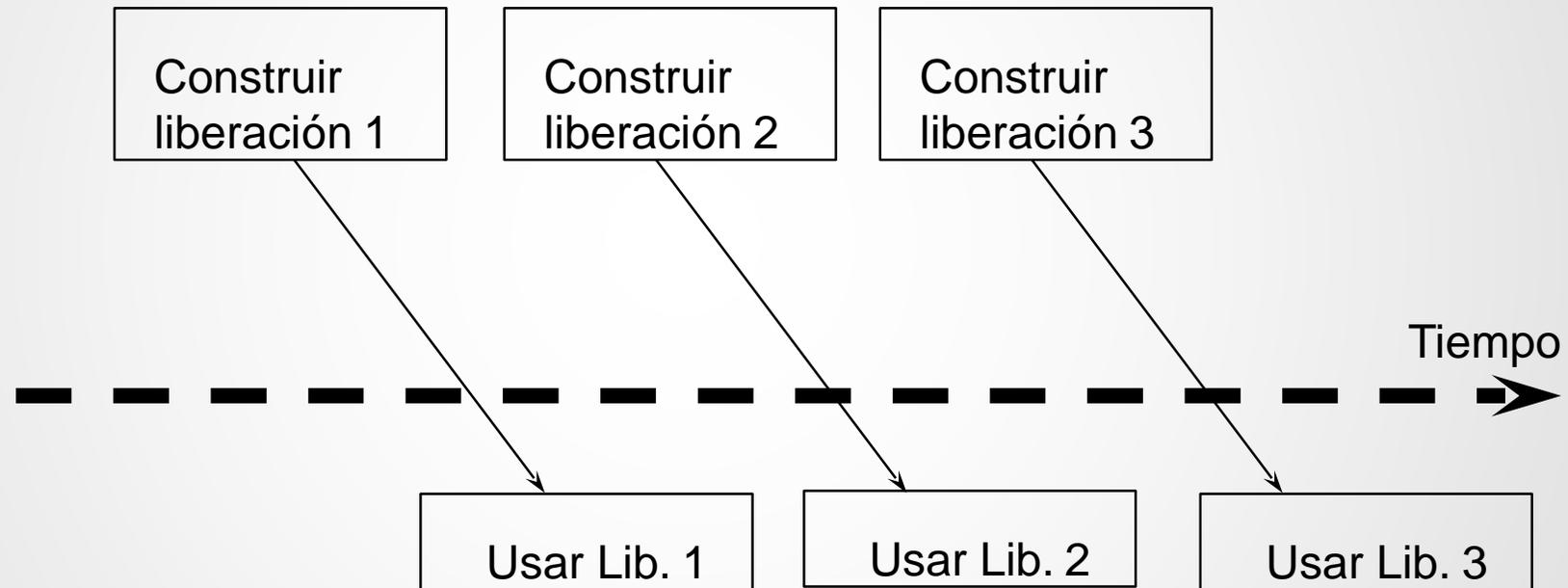
# Beneficios del modelo en fases

- Beneficios en comparación al modelo en cascada:
  - Tiempo total de desarrollo más corto.
  - El costo de implementar cambios de los requisitos se reduce.
  - Es mas fácil obtener **retroalimentación** de los clientes sobre el desarrollo que se esta realizando.
  - Es posible realizar una implementación y liberación de **software útil** para el cliente con **mayor rapidez**.

# Desarrollo en fases con liberaciones parciales

DESARROLLADORES

Sistemas en desarrollo



Sistemas en producción

USUARIOS

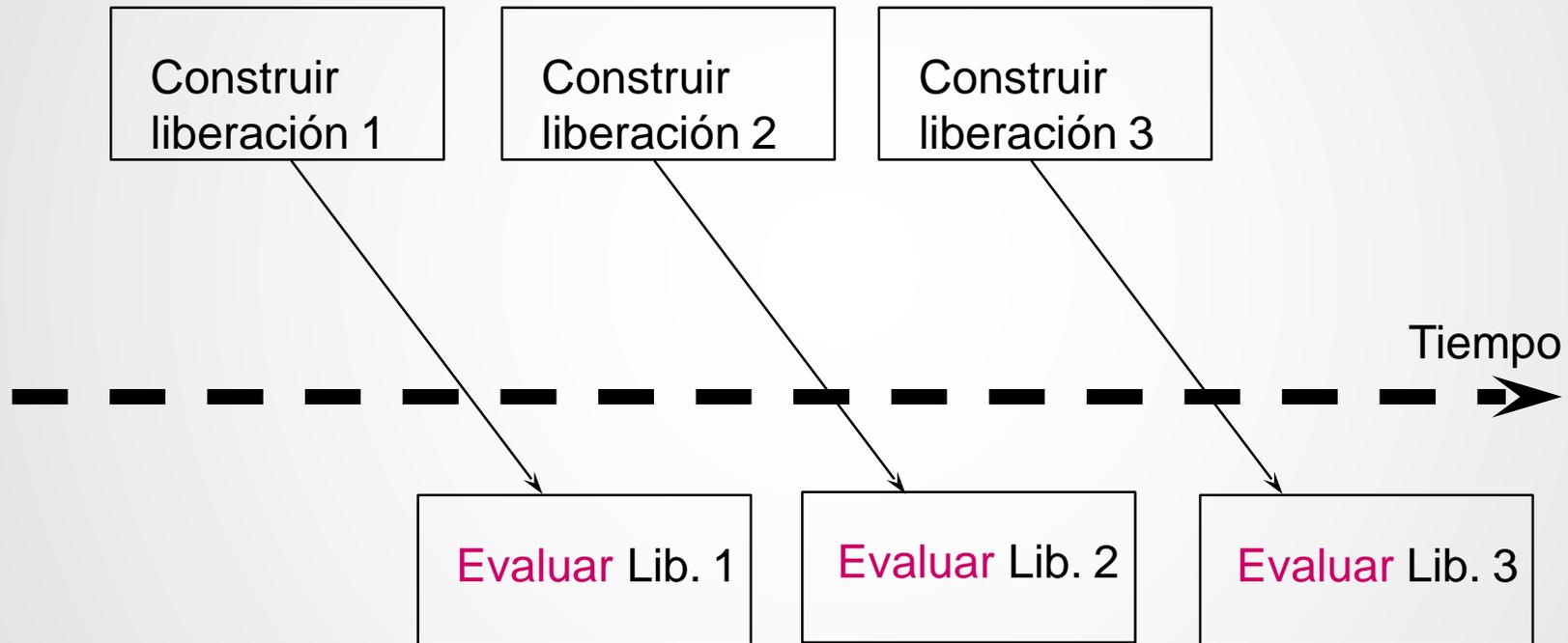
# Modelo en fases con liberaciones parciales

- El sistema se puede entregar en pedazos.
- Las versiones más tempranas incluyen la funcionalidad más importante o que se requiere con más urgencia.
- Así el cliente puede evaluar el sistema de forma temprana.
- Permite que los clientes tengan algo de funcionalidad mientras que se está desarrollando el resto.
- Permite que funcionen dos sistemas en paralelo:
  - el sistema en producción (liberación n): actualmente en uso
  - el sistema en desarrollo (liberación n+1): la siguiente versión

# Desarrollo en fases con evaluaciones parciales

DESARROLLADORES  
USUARIOS

## Sistemas en desarrollo



## Sistemas en evaluación

# Desarrollo incremental

- **Se usa tanto en procesos ágiles como en procesos dirigidos por planes, o en procesos que mezclen ambos enfoques:**
  - En un enfoque dirigido por planes, los incrementos del sistema se identifican con antelación;
  - En un enfoque ágil se identifican los primeros incrementos; los siguientes dependerán del avance y de las prioridades del cliente.

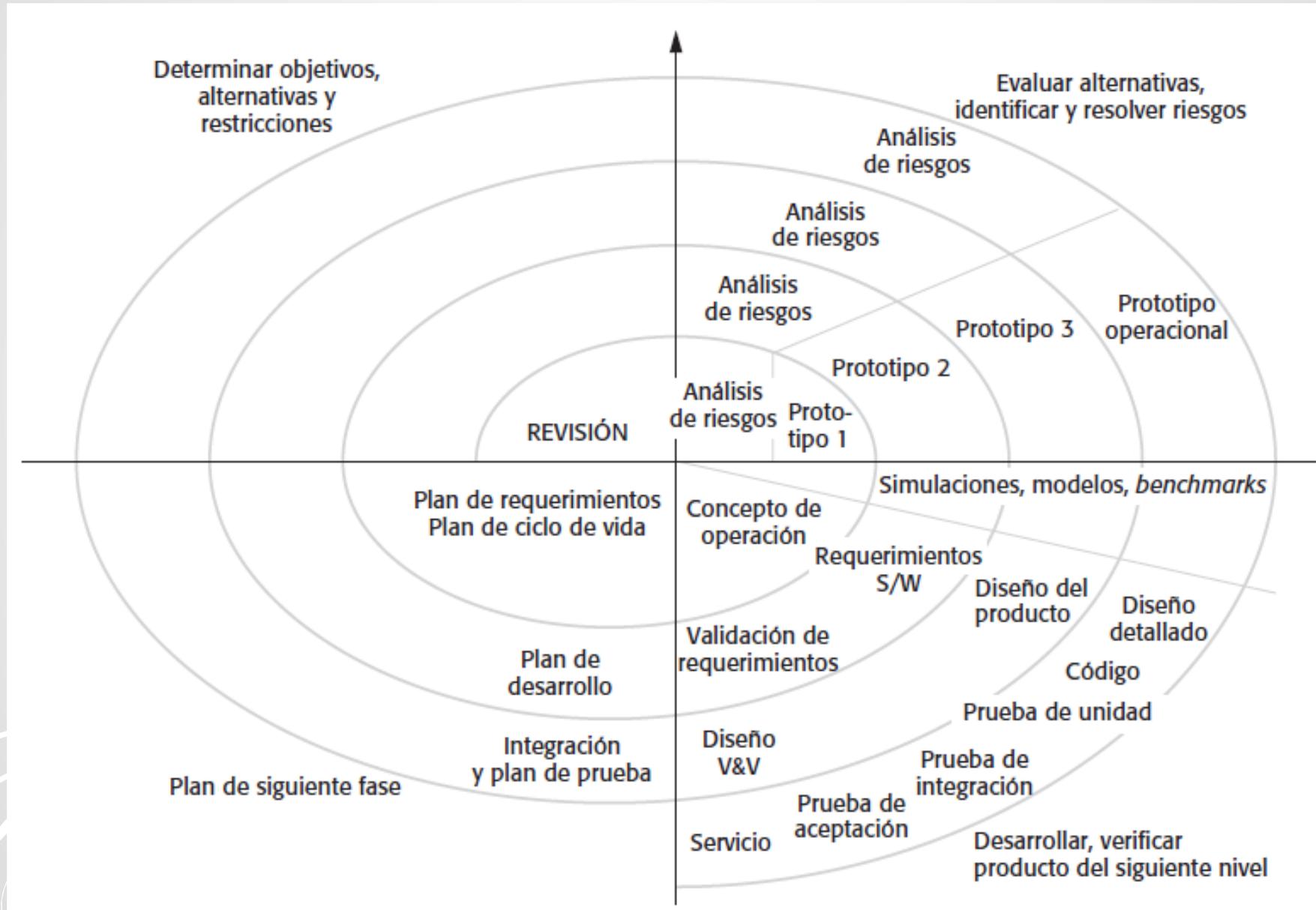
# Desarrollo incremental

- Problemas posibles:
  - La estructura del sistema puede degradarse a medida que se agregan nuevos incrementos como resultado de cambios inesperados. E incorporar nuevos cambios se torna difícil y costoso.
- Soluciones: ¿Cómo se puede mitigar este problema?
  - Invertir en **refactorización** para mejorar el software
  - Definir un marco estable de arquitectura con antelación.

# Modelo de proceso en espiral

- El proceso es representado como un espiral en lugar de como una secuencia de actividades con retroceso.
- Cada bucle en el espiral representa una fase en el proceso y cada cuadrante, una subfase.
- No hay fases fijas como las especificaciones o el diseño; los bucles en el espiral son seleccionados dependiendo de lo que se requiere.
- Los riesgos son evaluados y resueltos de manera explícita en todo el proceso.

# Modelo en espiral



# Planificación

Sommerville

# Planificación

- La planificación de proyectos es una de las labores más importantes de un administrador de proyectos de software.
- Como administrador, debe
  - dividir el trabajo en partes y asignar éstas a los miembros del equipo del proyecto,
  - anticipar los problemas que pudieran surgir y preparar posibles soluciones a tales inconvenientes.
- El plan creado al comienzo de un proyecto se usa para comunicar al equipo y los clientes cómo se realizará el trabajo, así como para ayudar a valorar el avance del proyecto.

# Desarrollo dirigido por un plan

- El proceso de desarrollo se planea a detalle.
- Se elabora un plan de proyecto que registra
  - el trabajo que se va a realizar,
  - quién lo efectuará,
  - el calendario de desarrollo y
  - los productos de trabajo.
- El plan se usa para:
  - apoyar la toma de decisiones del proyecto
  - medir el progreso.
- Manera tradicional de administrar grandes proyectos de desarrollo de software.
- Ventajas: la planeación temprana permite
  - que los asuntos de la organización (disponibilidad de personal, otros proyectos, etc.) se tomen en cuenta, y
  - que los problemas potenciales y dependencias se descubran antes de que se inicie el proyecto, y no cuando ya esté en marcha.

# Desarrollo ágil o adaptativo

- Muchas decisiones que afectan el desarrollo se retrasan y se hacen posteriormente, según se requiera, durante el proceso de desarrollo.
- Ventajas:
  - En el desarrollo dirigido por un plan, muchas decisiones tempranas deben revisarse debido a cambios al entorno en los que se desarrollará y usará el software.
  - Retrasar las decisiones evita tener que volver a trabajar.

# Enfoque mixto

- Una mezcla juiciosa de desarrollo basado en un plan y ágil.
- El equilibrio depende del tipo de proyecto y de las habilidades del personal que estén disponibles.
- los grandes sistemas críticos de seguridad y protección requieren un amplio análisis previo y quizá certificarse antes de utilizarse. Esto debe estar dirigido mediante un plan.
- los sistemas de información, de pequeños a medianos, que se usan en un entorno competitivo vertiginosamente cambiante, deben ser ágiles.
- Cuando muchas compañías están implicadas en un proyecto de desarrollo, con frecuencia se usa un enfoque basado en un plan para coordinar el trabajo a través de cada sitio de desarrollo.

# Procesos guiados por planes y procesos ágiles

- Procesos guiados por planes:
  - P. ej. cascada, RUP, MUM
  - El plan se construye fundamentalmente al comienzo.
  - La comunicación se basa en documentos.
- Procesos ágiles:
  - P. ej. XP, SCRUM, FDD, DSDM
  - El plan se completa a medida que avanza el proyecto.
  - La comunicación está basada en conversaciones cara a cara.

# Procesos guiados por planes

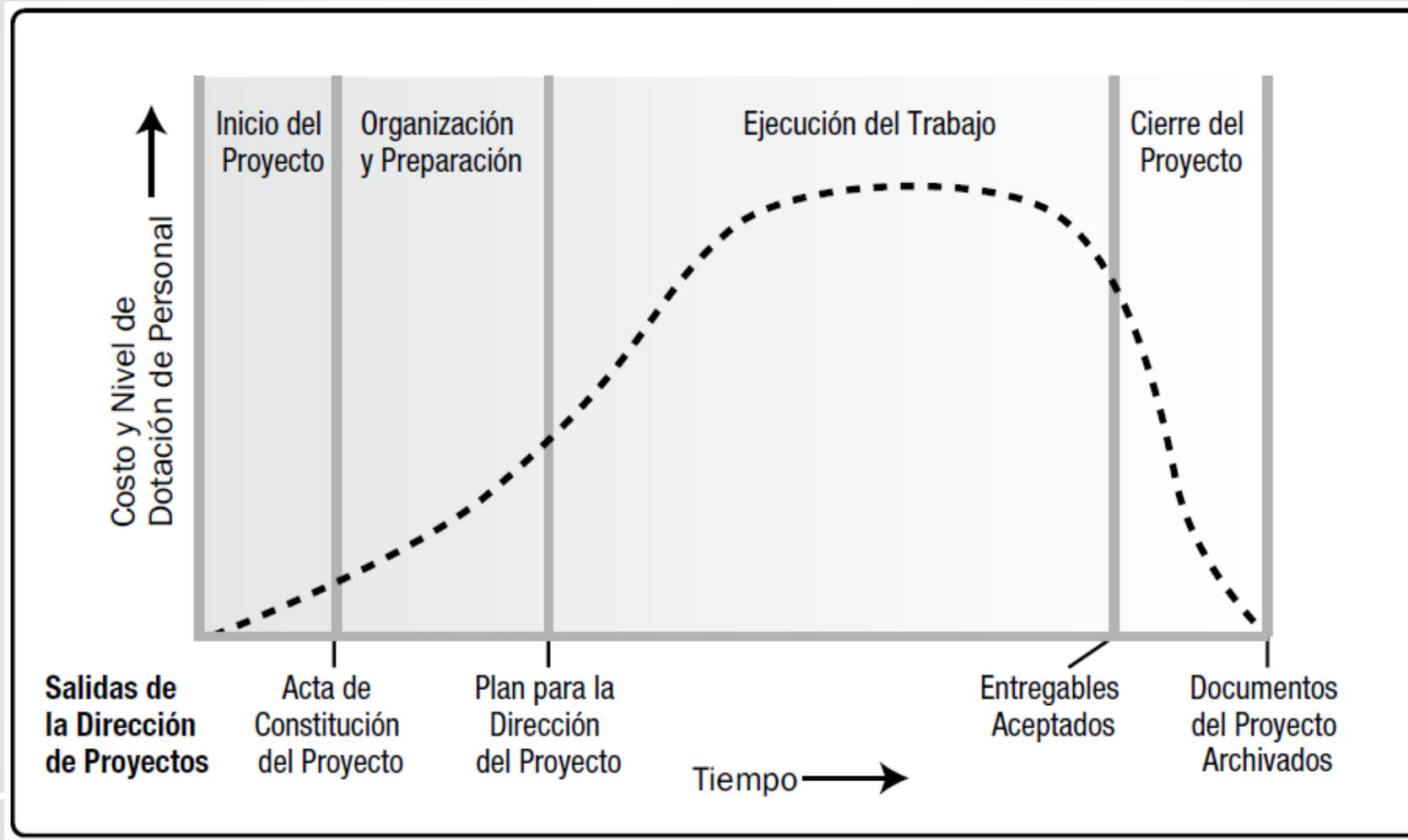
- Cuándo usar un proceso guiado por planes:
  - Cuando se trata de una aplicación crítica
  - En un proyecto muy grande con distintos equipos de desarrollo distribuidos.
  - Cuando los que lo van a mantener no son los mismos que los desarrollaron.



# Ciclo de vida del proyecto

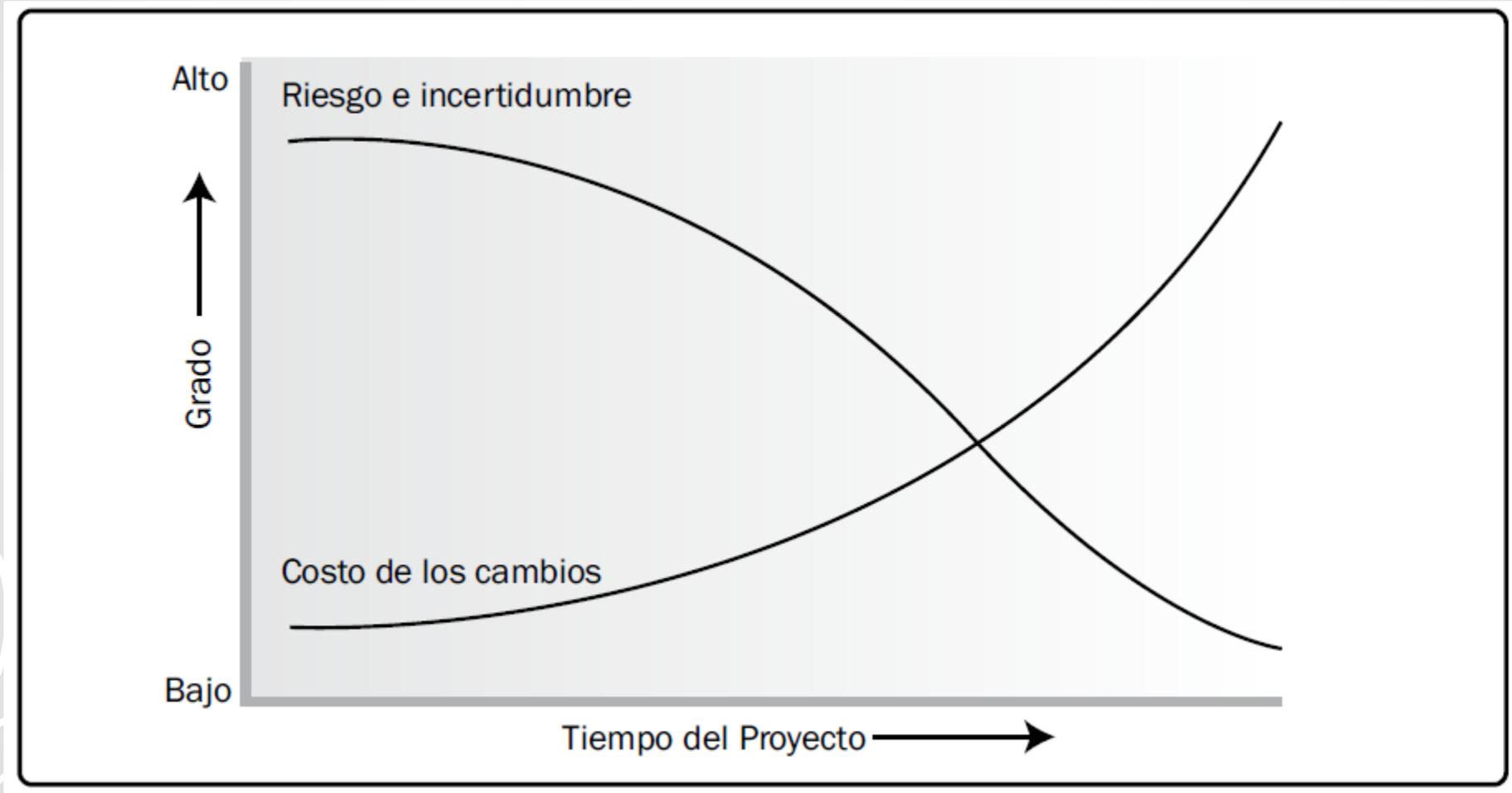
PMI

# Estructura genérica de ciclo de vida



Niveles de costo y dotación de personal

# Costos e incertidumbre

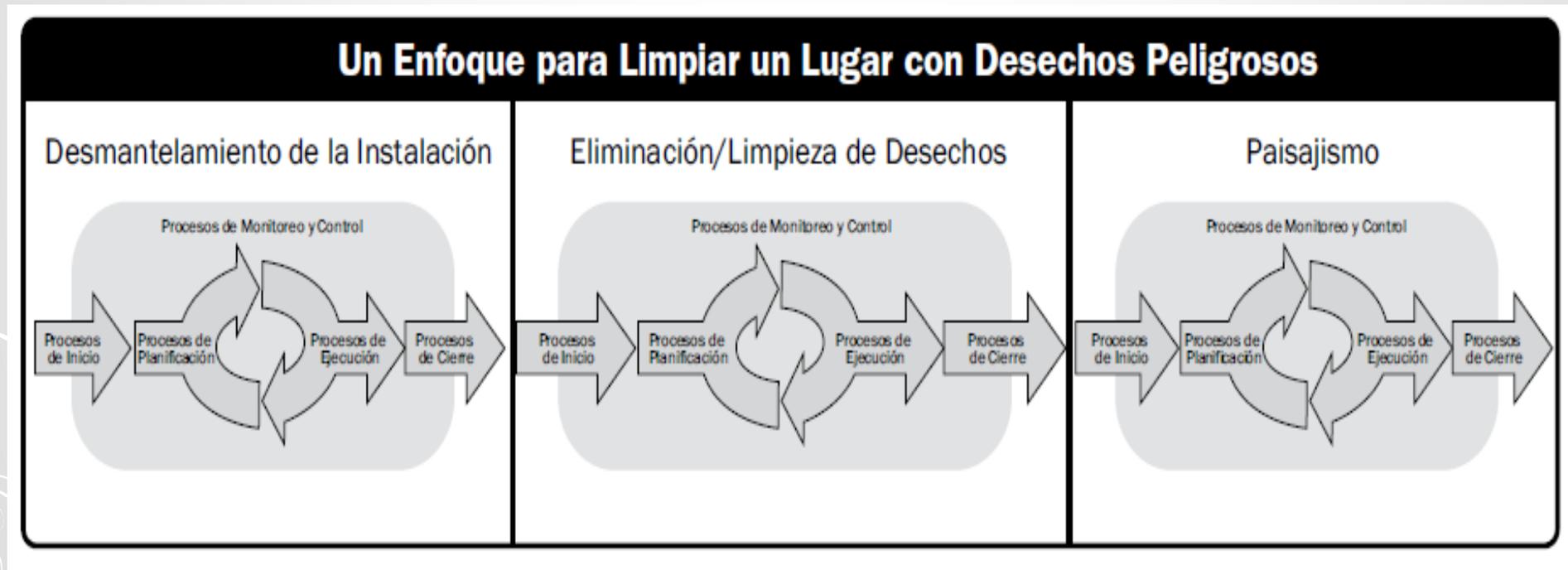


# Fases

- Fases: conjunto de actividades del proyecto, relacionadas de manera lógica, que culmina con la finalización de uno o más entregables.
- Los procesos son actividades que pueden realizarse y repetirse dentro de cada fase de un proyecto.
- Cada fase tiene **enfoque único** (distinto al de otras).
- El logro del objetivo de la fase requiere controles o procesos que son exclusivos de esa fase.
- El cierre de una fase termina con transferencia o entrega del trabajo producido (entregable de la fase).

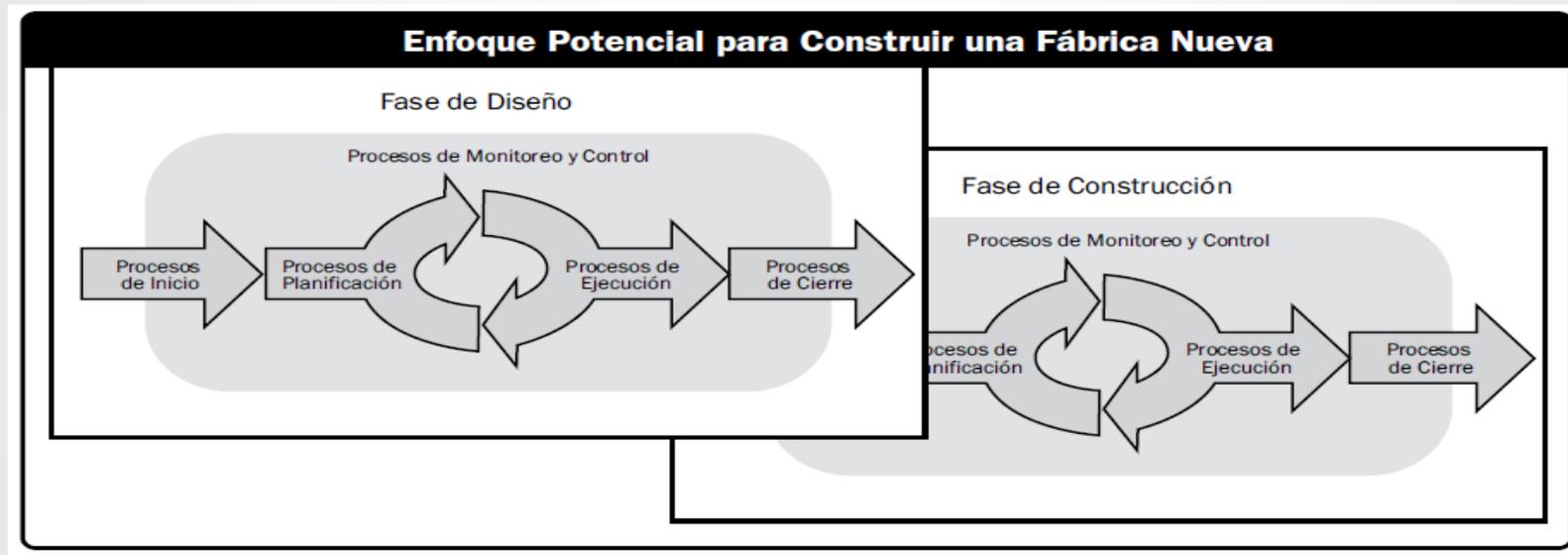
# Relaciones entre fases

- Secuencial



# Relaciones entre fases

- Superposición



- Técnica de compresión del cronograma (ejecución rápida).
- Puede:
  - requerir recursos adicionales
  - aumentar el riesgo
  - retrabajo

# Enfoques de ciclos de vida

- Enfoques predictivos u orientados a plan:
  - el producto y los entregables se definen al comienzo del proyecto y cualquier cambio en el alcance es cuidadosamente gestionado.
- Enfoques adaptativos u orientados al cambio:
  - el producto se desarrolla tras múltiples iteraciones y el alcance detallado para cada iteración se define en el comienzo de esta.

# Ciclos de vida predictivos

- Se opta por ellos cuando
  - el producto a entregar se comprende bien,
  - existe una base práctica significativa en la industria, o
  - cuando un producto debe ser entregado en su totalidad para que tenga valor para los grupos de interesados.
- Planificación progresiva.
- Pueden ser en cascada, iterativos e incrementales, etc.

# Ciclos de vida adaptativos

- Responder a niveles altos de cambio
- Participación continua de los interesados.
- Son métodos iterativos e incrementales, pero con iteraciones muy rápidas (2 a 4 semanas) y de duración y costo fijos.