

Producto y proceso de software

# Producto y proceso de software

¿Qué imagen se tiene?

# Objetivos

Definir la Ingeniería de Software y explicar su importancia.

Discutir los conceptos de producto de software y proceso de software.

Definir la importancia de la visibilidad de los procesos.

Introducir la noción de responsabilidad profesional.

# Agenda

- Productos de Software.
- El proceso de Software.
- El modelo de Espiral de Boehm.
- La visibilidad de los procesos.
- Responsabilidad profesional.

# Ingeniería de Software

Las economías de los países desarrollados dependen en gran parte del software.

Más y más sistemas son actualmente controlados por software.

La Ingeniería de Software concierne a teorías, métodos y herramientas para el desarrollo profesional de software.

# Costos del Software

Los costos del software a menudo dominan al costo del sistema. El costo del software es a menudo más caro que el hardware.

Cuesta más mantener el software que desarrollarlo. Para sistemas con una larga vida, este costo se multiplica.

La Ingeniería de Software concierne a un desarrollo efectivo en cuanto a costos del software.

# Productos de Software

- Productos genéricos.
  - Productos que son producidos por una organización para ser vendidos al mercado.
- Productos hechos a medida.
  - Sistemas que son desarrollados bajo pedido a un desarrollador específico.
- La mayor parte del gasto del software es en productos genéricos, pero hay más esfuerzo en el desarrollo de los sistemas hechos a medida.

# Características de los Productos de Software

- Mantenable.
  - Debe ser posible que el software evolucione y que siga cumpliendo con sus especificaciones.
- Confiable.
  - El software no debe causar daños físicos o económicos en el caso de fallos.
- Eficiente.
  - El software no debe desperdiciar los recursos del sistema.
- Fácil de usar.
  - El software debe contar con una interfaz de usuario adecuada y su documentación.



# Importancia de las características del producto

- La importancia relativa de las características depende del tipo de producto y el ambiente en el que será utilizado.
- En algunos casos, algunos atributos pueden dominar.
  - En sistemas de seguridad críticos de tiempo real, los atributos clave pueden ser la confiabilidad y la eficiencia.
- Los costos tienden a crecer exponencialmente si se requieren niveles estrictos de determinadas características.

# El Proceso de Software

- Conjunto estructurado de actividades requeridas para desarrollar un sistema de software.
  - Especificación.
  - Diseño.
  - Validación.
  - Evolución.
- Las actividades varían dependiendo de la organización y del tipo de sistema a desarrollar.
- Debe estar explícitamente modelado si va a ser bien administrado.

# Características del proceso

- Definido
  - ¿Se encuentra el proceso bien definido y es comprensible?
- Visible
  - ¿El proceso es visible?
- Asistido
  - ¿Puede el proceso ser soportado por herramientas CASE?
- Aceptable
  - ¿El proceso es aceptado por el personal involucrado?

# Características del proceso

- Confiable
  - ¿Los errores del proceso son descubiertos antes de que se conviertan en errores del producto?
- Robusto
  - ¿Puede continuar el proceso a pesar de problemas inesperados?
- Mantenible
  - ¿Puede el proceso evolucionar para cumplir con los objetivos organizacionales?
- Ágil
  - ¿Qué tan rápido puede producirse el sistema?

# Modelo de Ingeniería del Proceso

- Especificación: establecer los requerimientos y restricciones del sistema.
- Diseño: modelar la solución.
- Construcción: desarrollar el sistema.
- Prueba: verificar y validar que el sistema cumpla con las especificaciones requeridas.
- Instalación: entregar el sistema al usuario y asegurar su funcionamiento.
- Mantenimiento: ¿reparar fallos en el sistema cuando sea descubiertos?

# Problemas en el Modelo del Proceso

- Normalmente, las especificaciones son incompletas o anómalas
- No existe una distinción precisa entre la especificación, el diseño y la construcción
- No se puede probar el sistema hasta que se haya producido
- El software no se puede remplazar siempre durante el mantenimiento

# Modelos Genéricos de Desarrollo de Software

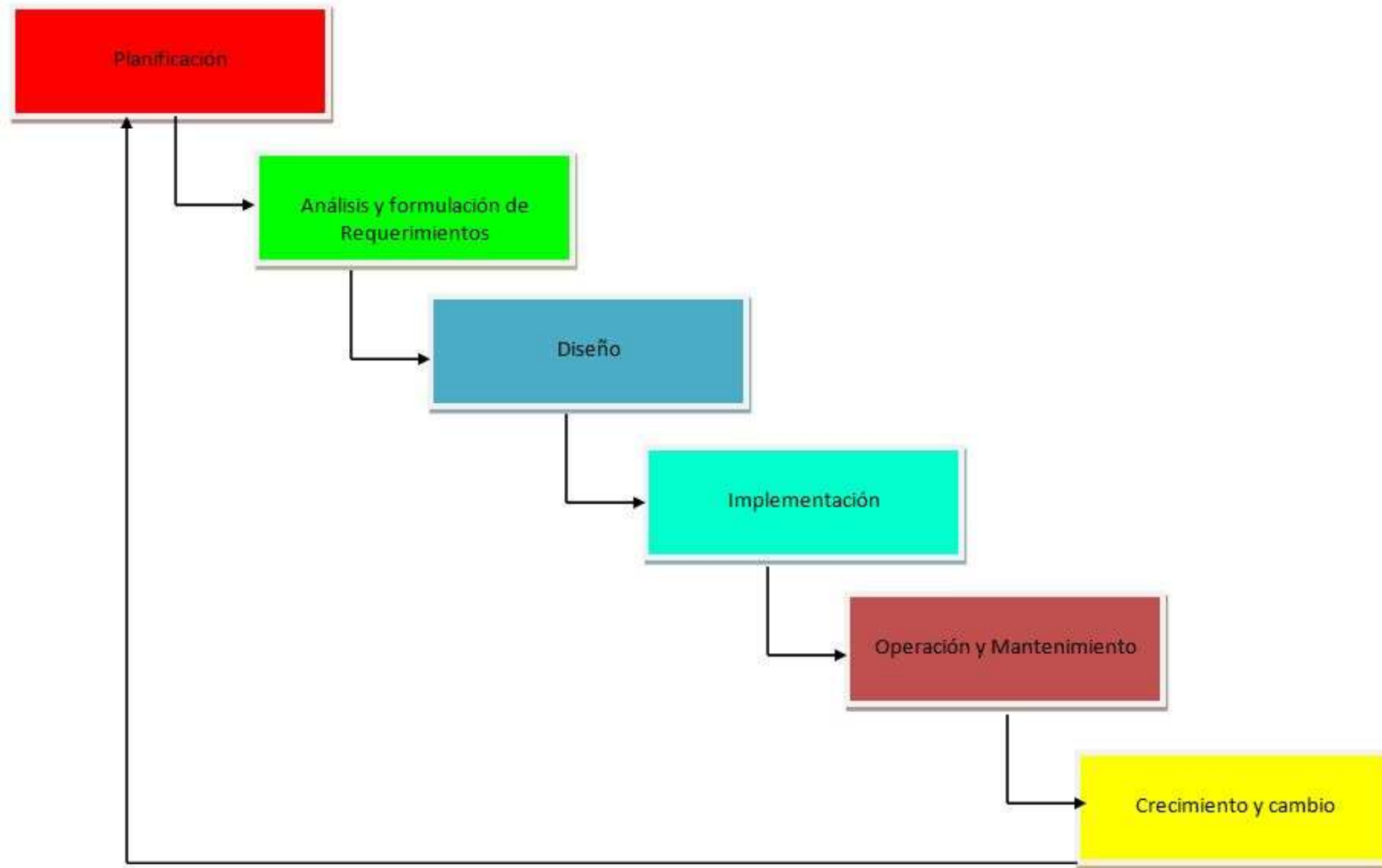
- Modelo de Cascada
  - Separa en distintas fases la especificación y el desarrollo.
- Desarrollo Evolutivo
  - La especificación y el desarrollo están intercalados.
- Prototipado
  - Un modelo sirve de prototipo para la construcción del sistema final.

# Modelos Genéricos de Desarrollo de Software

- Transformación Formal
  - Un modelo matemático del sistema se transforma formalmente en la implementación.
- Desarrollo basado en Reutilización
  - El sistema es ensamblado a partir de componentes existentes.



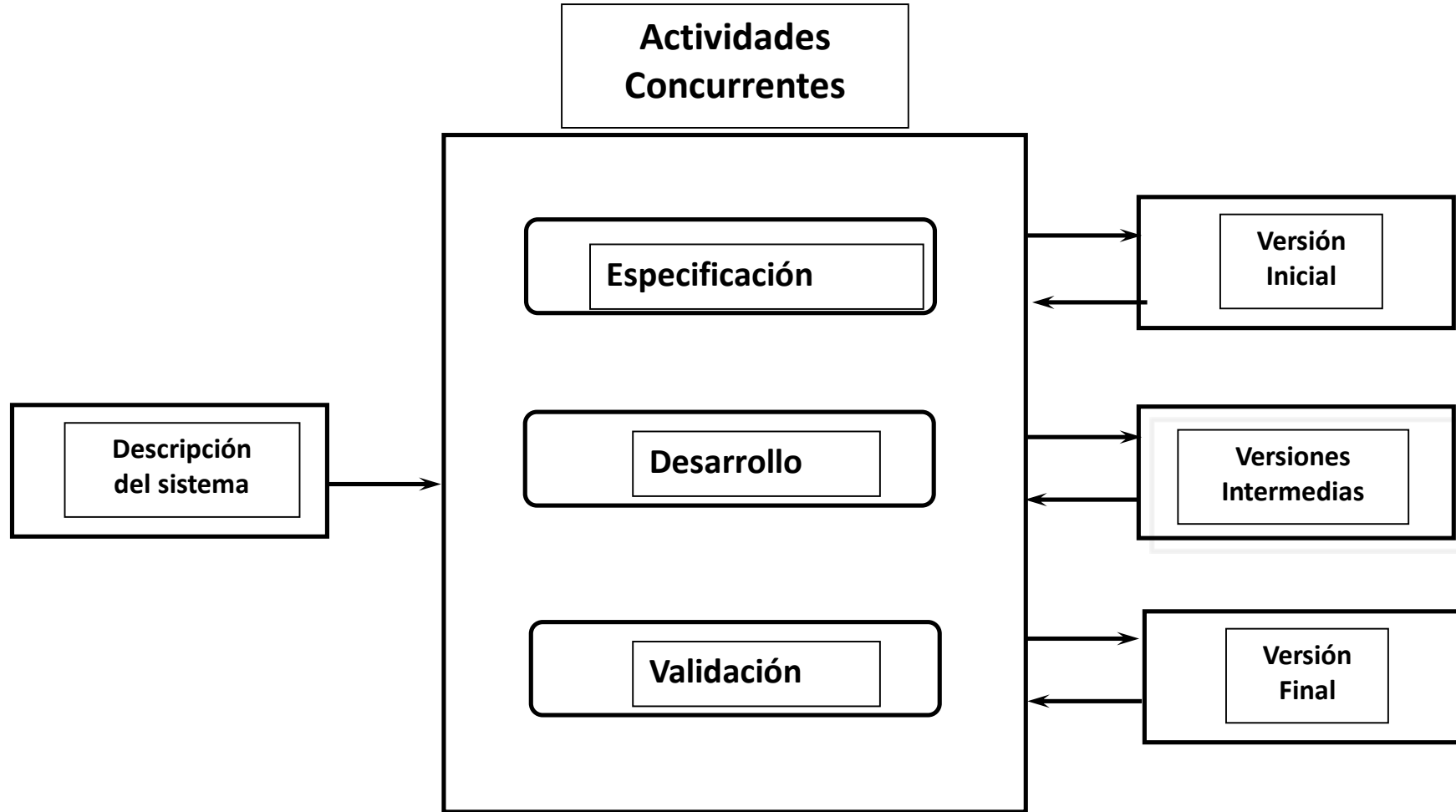
# Modelo de Cascada



# Fases del Modelo de Cascada

- Análisis de requerimientos y definición.
- Diseño del sistema y del software.
- Implementación y prueba de unidades
- Integración y prueba del sistema.
- Operación y mantenimiento.
- La dificultad en este modelo reside, en realizar cambios entre etapas.

# Desarrollo Evolutivo



# Desarrollo Evolutivo

- Problemas
  - Poca visibilidad en el proceso
  - Los sistemas están pobremente especificados
  - Se requieren habilidades especiales.
- Aplicabilidad
  - Para sistemas interactivos pequeños o medianos.
  - Para partes de sistemas grandes (p.ej. la interfaz de usuario).
  - Para sistemas de corta vida.

# Prototipado

- Prototipado exploratorio
  - El objetivo es trabajar con clientes hasta evolucionar a un sistema final, a partir de una especificación inicial. Se debe comenzar con unas especificaciones bien entendidas.
- Prototipado de “throw-away”.
  - El objetivo es entender los requerimientos del sistema. Se puede comenzar con especificaciones poco entendidas.

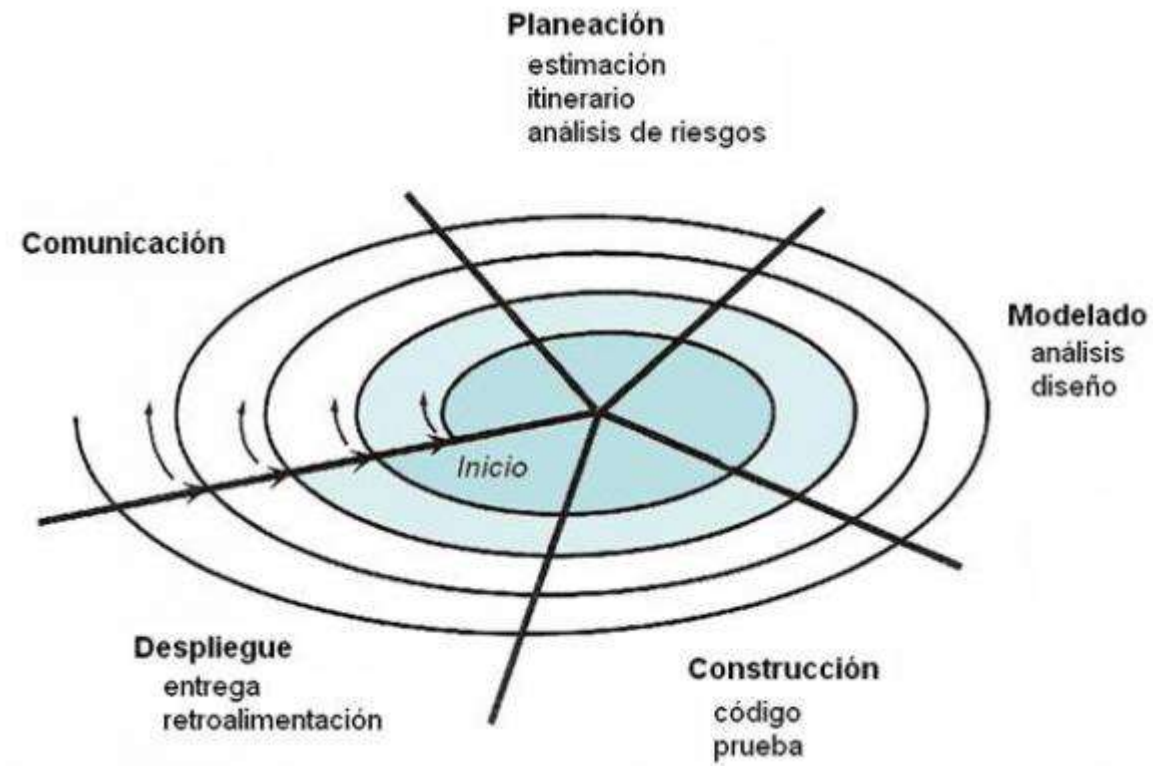
# Problemas y Riesgos con los Modelos.

- Cascada.
  - Alto riesgo en sistemas nuevos debido a problemas en las especificaciones y en el diseño.
  - Bajo riesgo para desarrollos bien comprendidos utilizando tecnología conocida.
- Prototipado.
  - Bajo riesgo para nuevas aplicaciones debido a que las especificaciones y el diseño se llevan a cabo paso a paso.
  - Alto riesgo debido a falta de visibilidad
- Evolutivo.
  - Alto riesgo debido a la necesidad de tecnología avanzada y habilidades del grupo desarrollador.

# Modelos de Procesos Híbridos

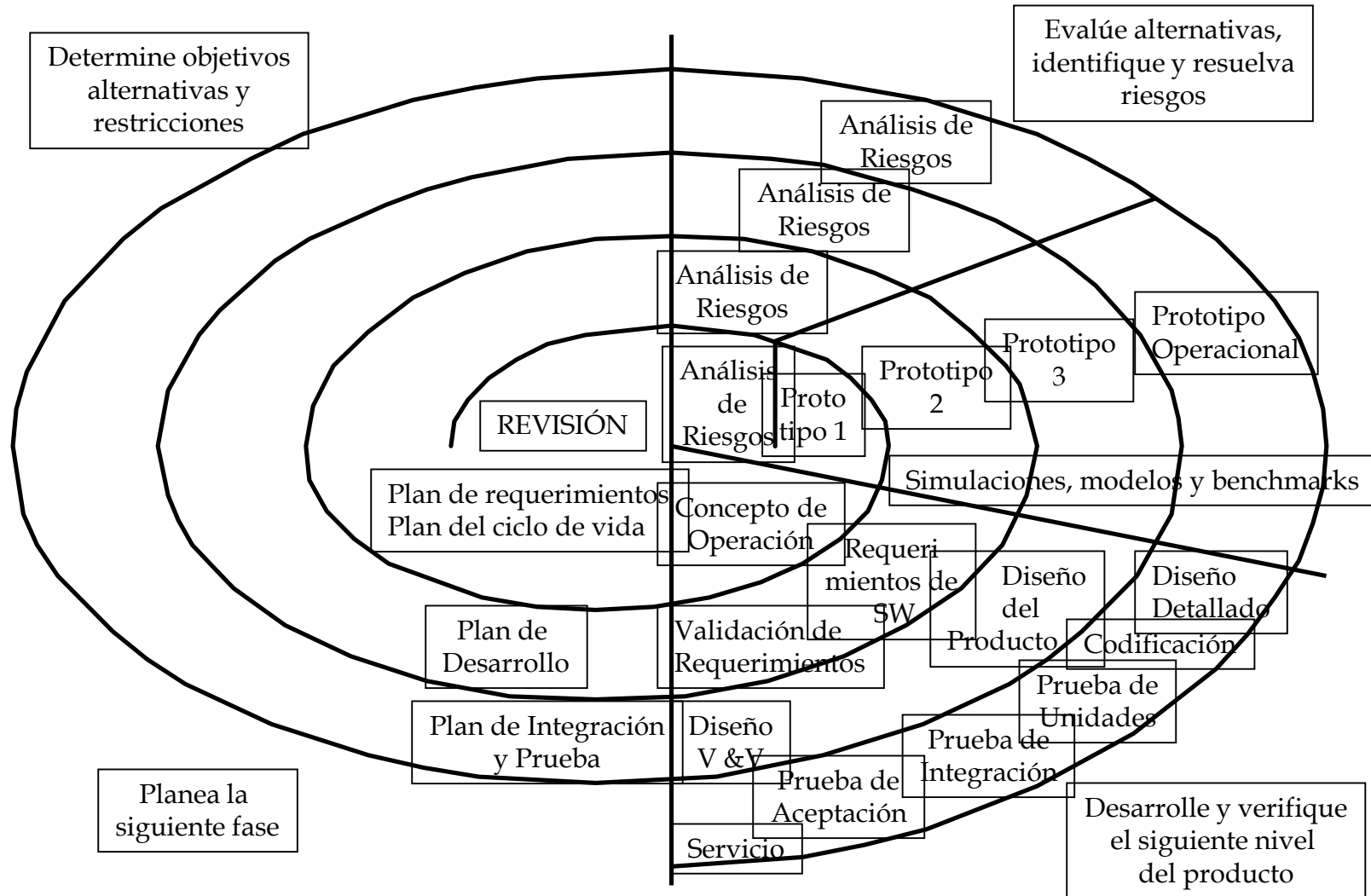
- Los sistemas grandes están hechos usualmente de varios subsistemas.
- No es necesario utilizar el mismo modelo de proceso para todos los subsistemas.
- El prototipado es recomendado cuando existen especificaciones de alto riesgo.
- El modelo de cascada es utilizado en desarrollos bien comprendidos.

# Modelo de Proceso de Espiral





# Modelo de Proceso de Espiral



# Modelo de Proceso de Espiral

- Planteamiento de Objetivos
  - Se identifican los objetivos específicos para cada fase del proyecto.
- Identificación y reducción de riesgos.
  - Los riesgos clave se identifican y analizan, y la información sirve para minimizar los riesgos.
- Desarrollo y Validación.
  - Se elige un modelo apropiado para la siguiente fase del desarrollo.
- Planeación.
  - Se revisa el proyecto y se trazan planes para la siguiente ronda del espiral.

# Plantilla para una ronda del espiral

- Objetivos.
- Restricciones.
- Alternativas.
- Riesgos.
- Resolución de riesgos.
- Resultados.
- Planes.
- Garantías (commitments).

# Ventajas del Modelo de Espiral

- Centra su atención en la reutilización de componentes y eliminación de errores en información descubierta en fases iniciales.
- Los objetivos de calidad son primordiales.
- Integra desarrollo con mantenimiento.
- Provee un marco de desarrollo de hardware/software.

# Problemas con el Modelo de Espiral

- El desarrollo contractual especifica el modelo del proceso y los resultados a entregar por adelantado.
- Requiere de experiencia en la identificación de riesgos.
- Requiere refinamiento para uso generalizado.

# Visibilidad del Modelo

Modelo de Proceso	Visibilidad del Proceso
Modelo de Cascada	Buena visibilidad, cada actividad produce un documento o resultado
Desarrollo Evolutivo	Visibilidad pobre, muy caro al producir documentos en cada iteración.
Modelos Formales	Buena visibilidad, en cada fase deben producirse documentos.
Desarrollo orientado a la reutilización	Visibilidad moderada. Importante contar con documentación de componentes reutilizables.
Modelo de Espiral	Buena visibilidad, cada segmento y cada anillo del espiral debe producir un documento.

# Resumen

- La Ingeniería de software concierne a las teorías, métodos y herramientas para el desarrollo, administración y evolución de productos de software.
- Los productos de software conformados por programas y documentación. Los atributos de los productos deben ser, mantenibilidad, confiabilidad, eficiencia y usabilidad.
- Un proceso de software consiste en aquellas actividades involucradas en su desarrollo.

# Resumen

- El modelo de cascada considera cada actividad del proceso como una actividad discreta.
- El modelo de desarrollo evolutivo considera actividades del proceso en forma concurrente.
- El modelo de espiral se basa en análisis de riesgos.
- La visibilidad del proceso involucra la creación de documentos o resultados de las actividades.
- Los Ingenieros de software deben tener responsabilidades éticas, sociales y profesionales.