Informática I

2019

Ciclo de vida del Software

Es el proceso por el que pasa el software en su desarrollo, desde que se concibe la idea hasta que el software deja de utilizarse.

Un proceso está compuesto por un conjunto de actividades y tareas que se deben realizar, y una serie de documentos asociados a ellas.

- ¿Qué es un proceso?

Un proceso es un conjunto de actividades que se suceden siguiendo una ordenación temporal determinada.

- ¿Qué es una actividad?

Una actividad es un conjunto de tareas.

- ¿Qué es una tarea?

Una acción que transforma unas entradas en unas salidas.

Según la **Norma *ISO/IEC Standard 12207:2008: Software life-Cycle processes*** propuesta por la ISO (International Organization for Standardization):

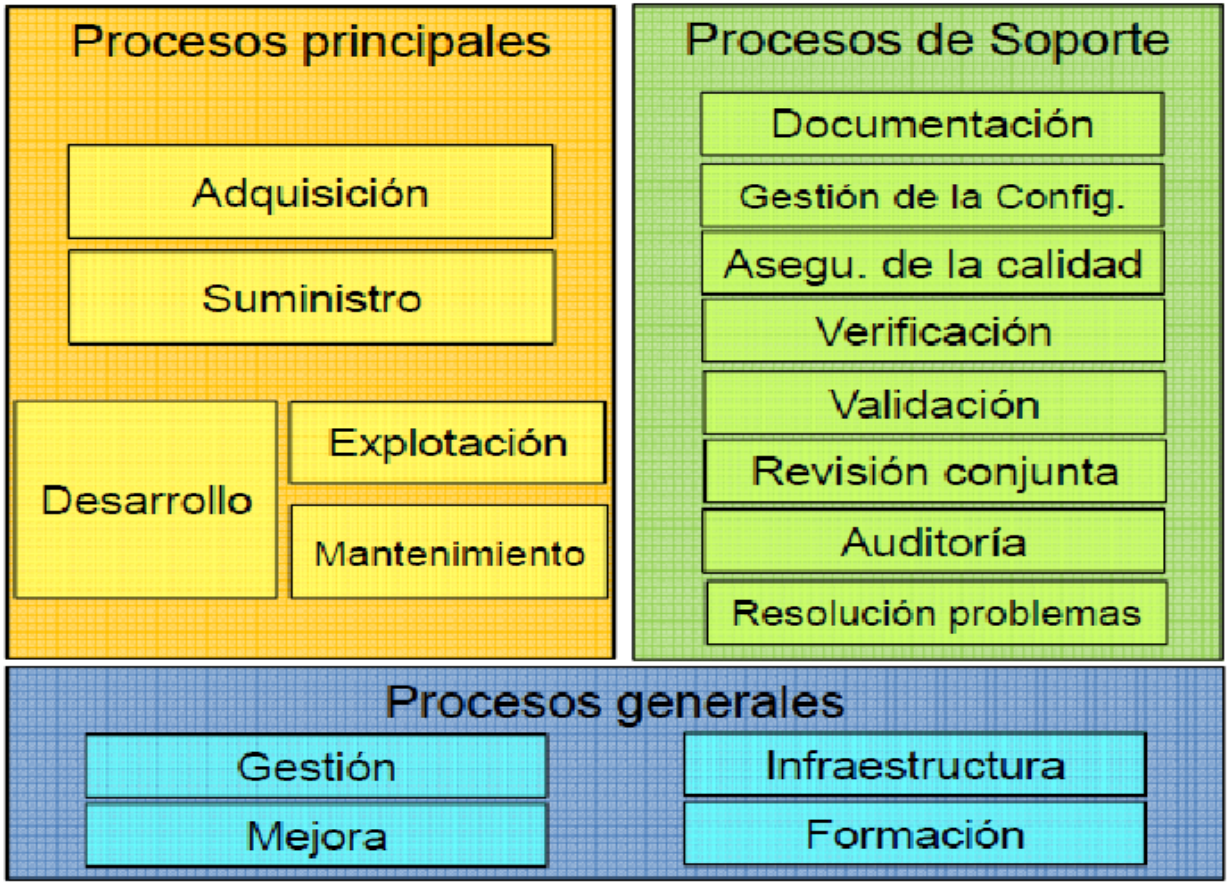
"Es un marco de referencia que contiene los procesos, las actividades y las tareas involucradas en el desarrollo, explotación y mantenimiento de un producto software, abarcando la vida del sistema desde la definición de requisitos hasta que se deja de utilizar"

Los procesos del Ciclo de Vida del Software:

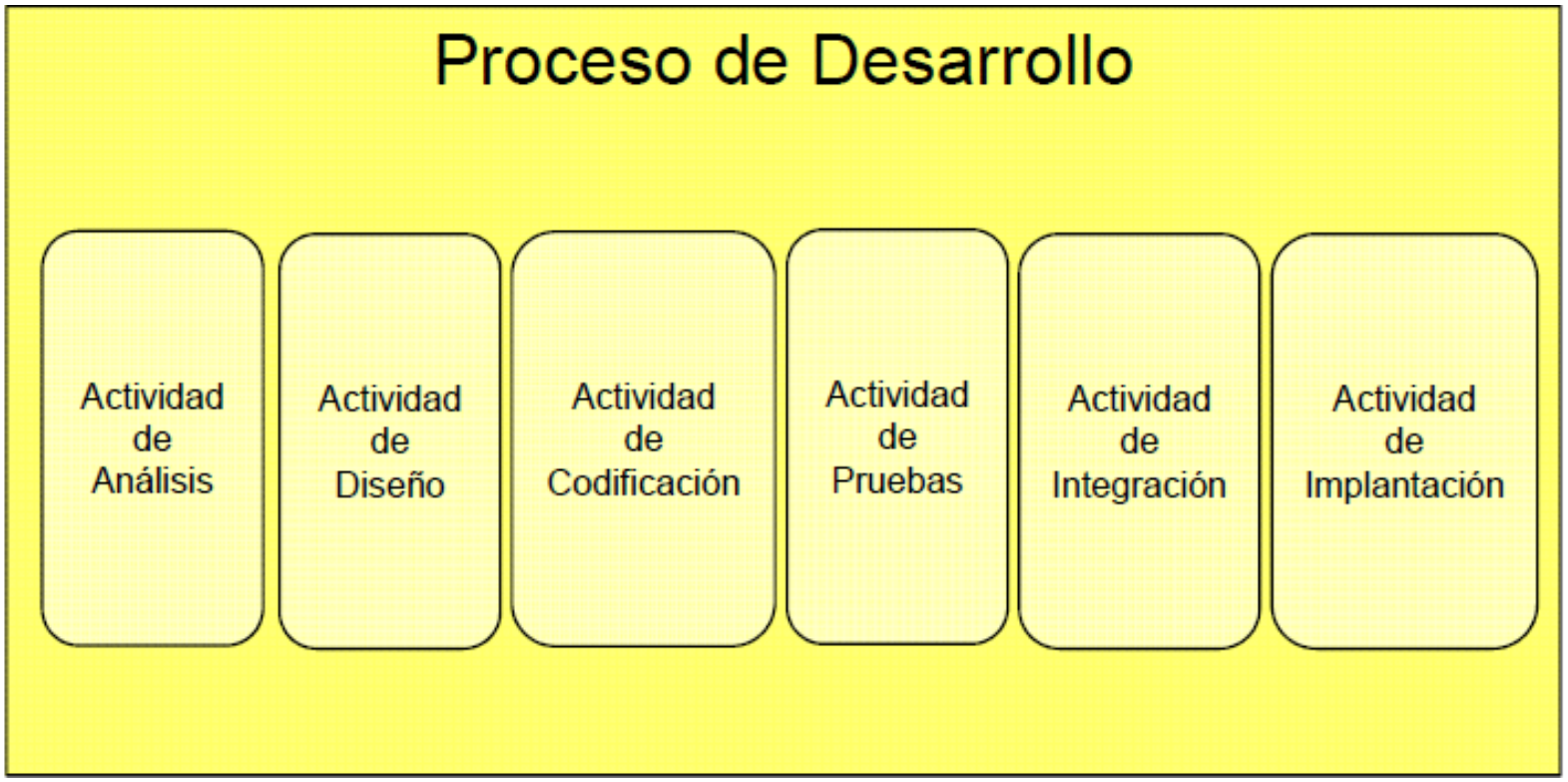
Según esta norma las actividades que se pueden llevar a cabo durante el ciclo de vida del Software se pueden agrupar en:

- 5 procesos principales.
- 8 procesos de soporte.
- 4 procesos de organización o generales.

Los procesos del Ciclo de Vida del Software:



Los procesos del Ciclo de Vida del Software:



Cada una de estas actividades está compuesta por diferentes tareas.

Procesos Principales

- **Adquisición**: Actividades y tareas que el comprador, el cliente o el usuario realizan para adquirir un sistema, un servicio o un producto software:
- ✓ Preparación y publicación de ofertas.
- ✓ Selección del suministrador de Software.

- **Suministro**: Actividades y tareas del suministrador:
- ✓ Preparar contratos como respuesta a una petición de un comprador de un producto Software.
- ✓ Identificar los recursos necesarios para llevar a cabo con éxito el desarrollo del producto Software.

Procesos Principales

Desarrollo: Actividades y tareas enfocadas a la obtención de un producto Software.

- Análisis
- Diseño
- Codificación
- Pruebas
- Integración
- Implantación

Explotación: Explotación del Software y soporte operativo a los usuarios.

Procesos Principales

Mantenimiento: Actividades que incluyen modificaciones del producto, tanto del código como de la documentación, debido a errores o a la necesidad de mejora o/y adaptación.

- ✓ Migración hacia un nuevo entorno operativo
- ✓ Retirada del producto

Procesos de Soporte

- ***Procesos de soporte****: dan soporte al resto de procesos y se aplican durante cualquier momento del ciclo de vida del Software.*

- **Documentación:** Registrar la información producida por un proceso o actividad del ciclo de vida:

✓ Diseñar, editar, distribuir y mantener los documentos producidos durante el desarrollo del SW

- **Gestión de la Configuración**: Actividades que controlan las modificaciones y versiones de los elementos.

✓ Registrar las peticiones de cambios e informar de los estados de éstos.

Procesos de Soporte

- **Aseguramiento de la calidad:** Actividades para asegurar que los productos cumplen los requisitos especificados y se ajustan a los planes establecidos.

- **Verificación:** Actividades para determinar el buen funcionamiento de un producto software.

- **Validación:** Actividades para determinar si el producto cumple los requisitos previstos.

Procesos de Soporte

- **Revisión conjunta:** Actividades que permiten determinar el estado de los productos en una determinada actividad del ciclo de vida o en una cierta fase del proyecto. Puede ser una reunión conjunta con el cliente, el grupo de desarrollo y los clientes potenciales para revisar el trabajo hecho.

- **Auditorías:** Actividades que permiten determinar en unos momentos determinados si se han conseguido los objetivos propuestos: requisitos, cumplimiento del contrato.

Procesos de Soporte

- **Resolución de problemas:** Actividades que permiten analizar y resolver los problemas o disconformidades con los requisitos o con el contrato, que hayan surgido durante el desarrollo, la explotación, el mantenimiento, o en cualquier otro momento.

- ✓ Disponer de un medio documental que permita asegurar que todos los problemas se han tratado.

Procesos Generales

***Procesos que dan soporte a la organización**: gestión, formación del personal, mejora de los procesos.*

- **Gestión:** Actividades de planificación, seguimiento, control, revisión y evaluación.
- **Infraestructura**: Actividades para determinar la infraestructura necesaria para un proceso. Incluye HW, Software, instalaciones…
- **Mejora**: Valorar, medir, controlar, evaluar y mejorar todos los procesos del ciclo de vida.
- **Formación**: Plan de formación para los empleados.

Modelos de procesos

¿Qué es un modelo de proceso?

- *Representación abstracta de un proceso del software.*
- Son ***estrategias de desarrollo*** que ayudan a organizar las diferentes actividades del ciclo de vida del software. →*Modelos de ciclo de vida del software*

- Estos modelos ayudan al control y a la coordinación del proyecto.

- El modelo a utilizar depende del tipo de proyecto.

Modelos de procesos Variantes

Modelos de procesos generales

- ✓ Modelo en cascada
- ✓ Modelo evolutivo
- ✓ Desarrollo formal de sistemas
- ✓ Desarrollo basado en la reutilización

Modelos de procesos híbridos

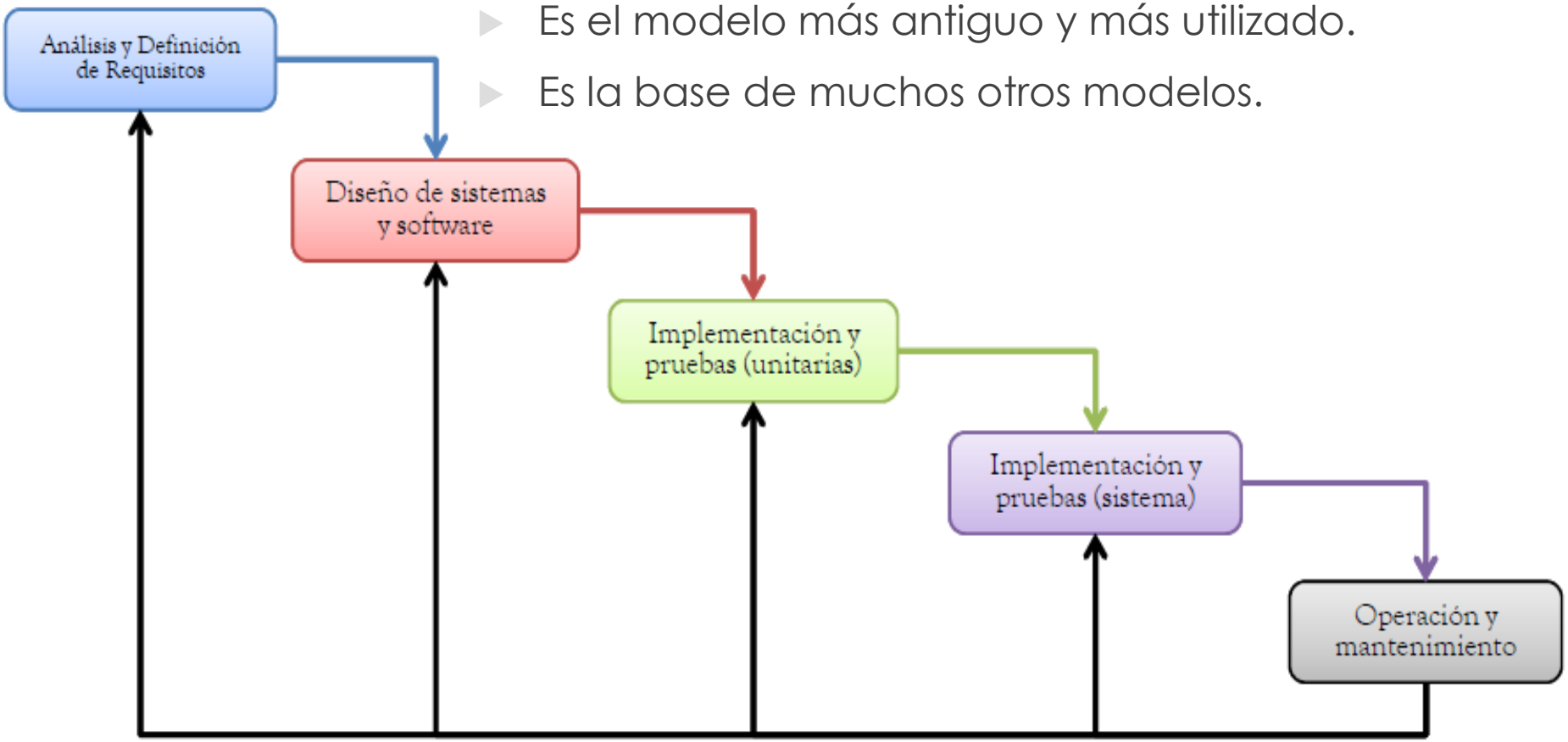
- ✓ Modelo en espiral
- ✓ Modelo incremental

Modelos de procesos generales
Modelo en Cascada (Waterfall)

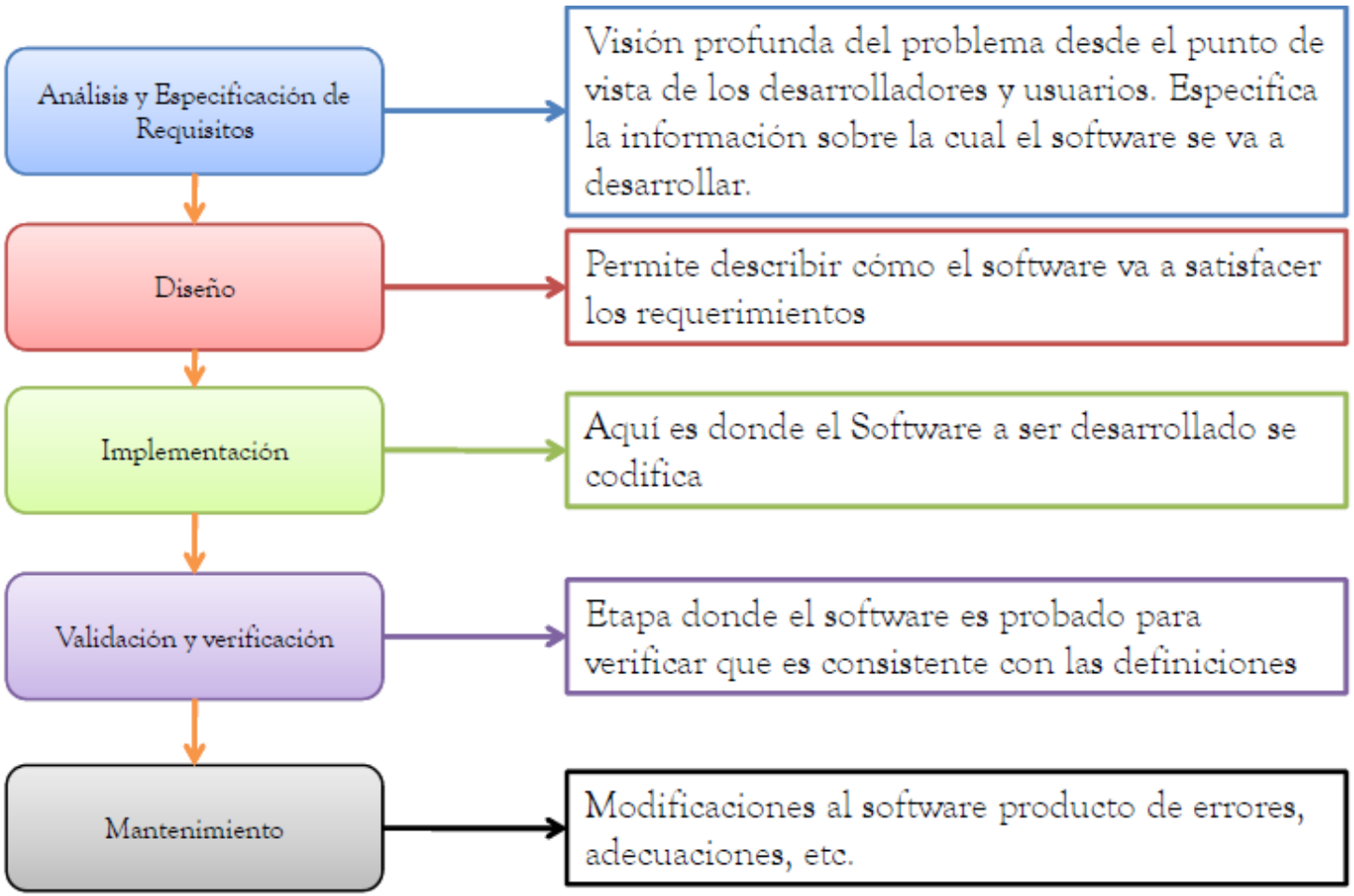
- También llamado *ciclo de vida básico* o *modelo lineal-secuencial.*
- Divide el proceso de desarrollo en un conjunto de etapas secuenciales.
- Una etapa no puede empezar hasta que no ha terminado la anterior.
- Al final de cada fase, el personal de desarrollo y los usuarios revisan el progreso del proyecto.
- En cada fase se genera todo un conjunto de documentos. Es un *modelo dirigido por documentos.*
- Son los productos principales en cada etapa.

Modelos de procesos generales
Modelo en Cascada (Waterfall)

- Es el modelo más antiguo y más utilizado.
- Es la base de muchos otros modelos.



Modelos de procesos generales
Modelo en Cascada (Waterfall)



Modelos de procesos generales
Modelo en Cascada (Waterfall)

Ventajas

- Cuando tenemos proyectos complejos y grandes pero que se entienden y quedan bien definidos desde un comienzo.
- Cuando el equipo de desarrollo no está muy cualificado o es aún inexperto porque la estructura de trabajo que propone es muy ordenada y ayuda a minimizar esfuerzo.
- Cuando realizamos una migración de software desde un entorno tecnológico obsoleto.

Modelos de procesos generales
Modelo en Cascada (Waterfall)

Desventajas

- Definir todos los requisitos al inicio del proceso no es práctico ya que el cliente añade y modifica según le van surgiendo necesidades durante el proceso de desarrollo.
- **¡El cliente nunca tiene claro lo que quiere!**
- El cliente no ve el producto en funcionamiento hasta el final del proceso.
- La validación de los requisitos iniciales no se hace hasta el final.
- Poca o nula flexibilidad a cambios.

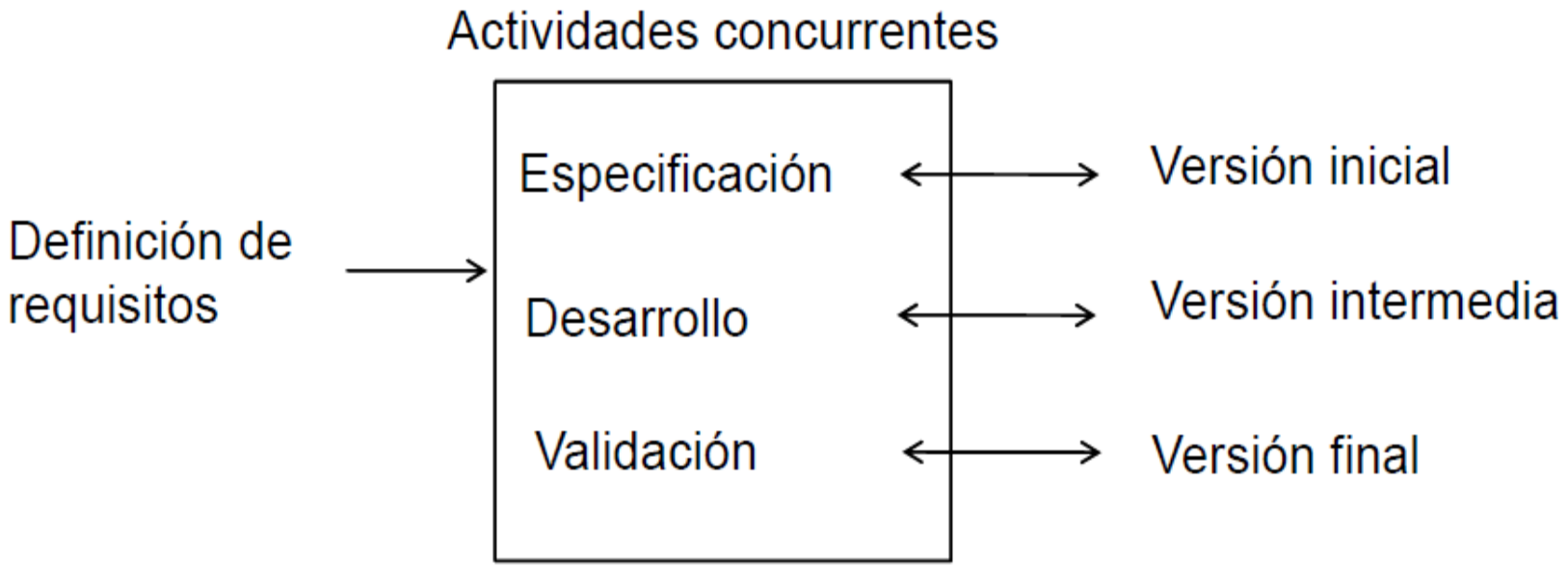
Modelos de procesos generales
Modelo Evolutivo

Un **prototipo** es una versión limitada del producto que permite a las partes responsables de su creación probarlo en situaciones reales y explorar su uso.

- ✓ Con este modelo hay un acercamiento al cliente.
- ✓ Gracias al prototipo, el cliente puede hacerse una idea de cómo está evolucionando el producto y esto ayuda a refinar los requisitos del sistema
- ✓ Con los prototipos definimos de forma clara y concreta qué quiere el cliente *los requisitos del sistema*.

Modelos de procesos generales
Modelo Evolutivo

La idea es desarrollar una implementación inicial que se expone a los comentarios del cliente/usuario. Esta implementación inicial evoluciona con las diferentes versiones que se crean hasta llegar a la solución final.

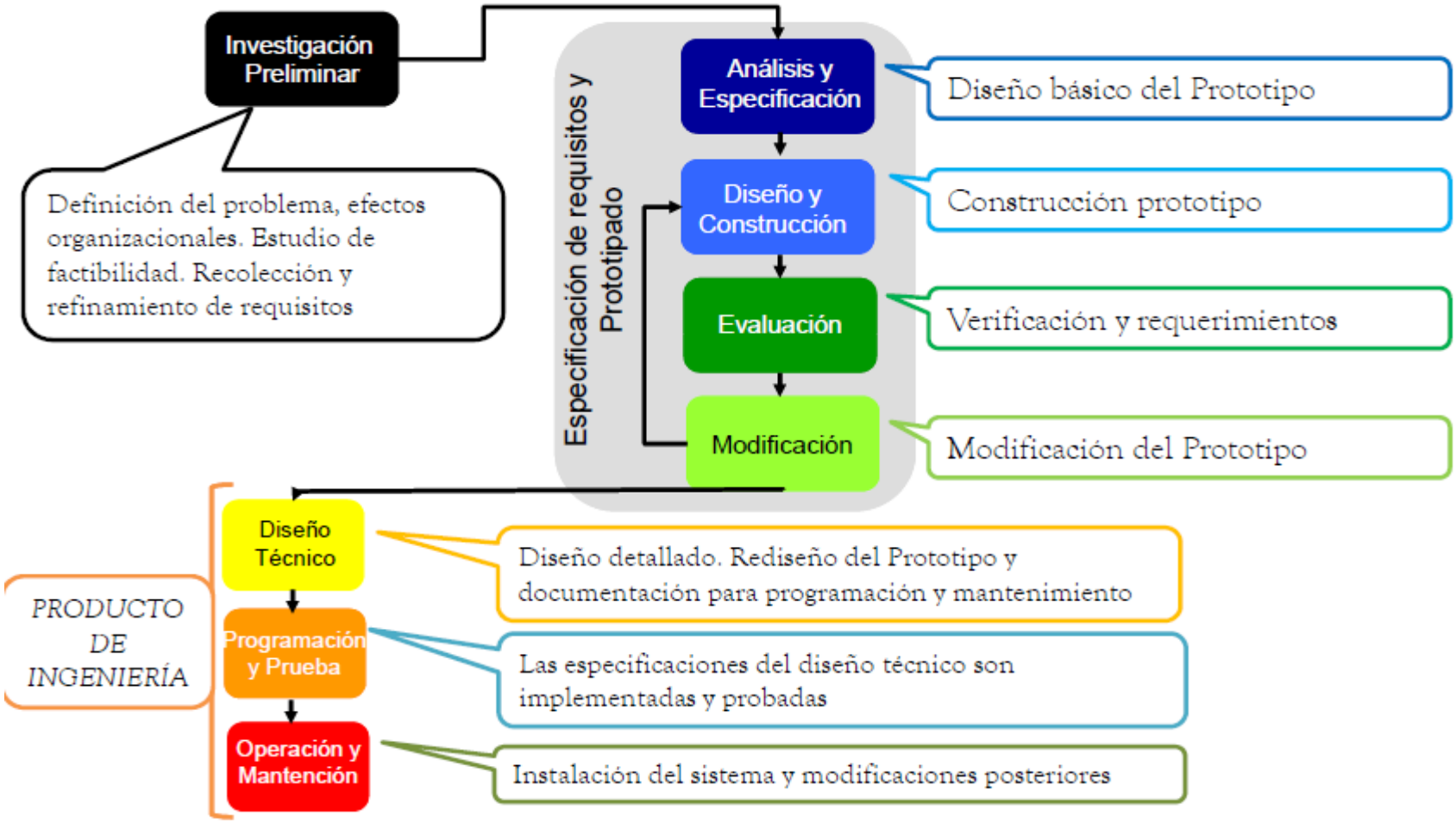


Modelos de procesos generales
Modelo Evolutivo

Ventajas

- Se recomienda para clientes que quieren ver resultados a corto plazo.
- Reduce costos y aumenta la probabilidad de éxito.
- Cuando el cliente no sabe lo que quiere y los requisitos no están bien definidos desde el principio Reduce el riesgo de construir productos que no satisfagan las necesidades de los usuarios.
- Cuando los requisitos evolucionan muy rápidamente.
- Conveniente para proyectos pequeños o medianos.
- Para sistemas on-line donde es más importante la parte de la interfaz con el usuario que las funcionalidades del sistema.
- Se recomienda utilizar este *modelo sólo para la especificación de requisitos*. Mejor continuar el desarrollo utilizando otro modelo.

Modelos de procesos generales
Modelo Evolutivo



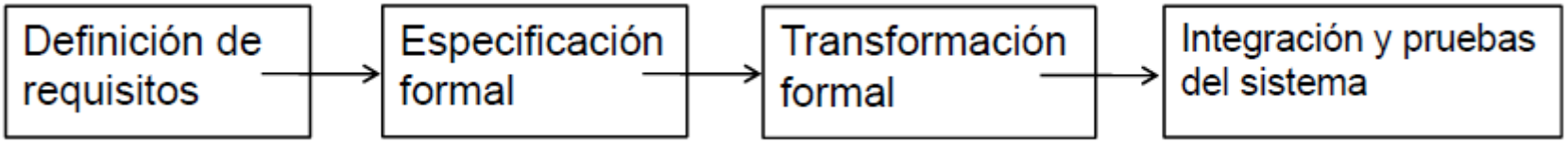
Modelos de procesos generales
Modelo Evolutivo

Desventajas

- El diseño rápido del prototipo hace que los desarrolladores utilicen herramientas que faciliten la rápida generación de código, dejando a un lado aspectos de calidad (eficiencia, fiabilidad, mantenibilidad del código, etc.).
- Probablemente no se tendrá un código óptimo.
- Exige disponer de las herramientas adecuadas.
- Descoordinación entre el desarrollo y la documentación. El desarrollo es tan rápido que a veces no se tiene tiempo suficiente para producirla.
- Podemos obtener una arquitectura débil y difícil de mantener.

Modelos de procesos generales
Desarrollo formal de sistemas

- Tiene puntos en común con el desarrollo en cascada pero éste se basa en la transformación matemática formal de una especificación del sistema a un programa ejecutable.
- La especificación de requisitos de software se desarrolla mediante una notación matemática.
- Las actividades de diseño, implementación y pruebas de unidades se reemplazan por un proceso de transformación formal.



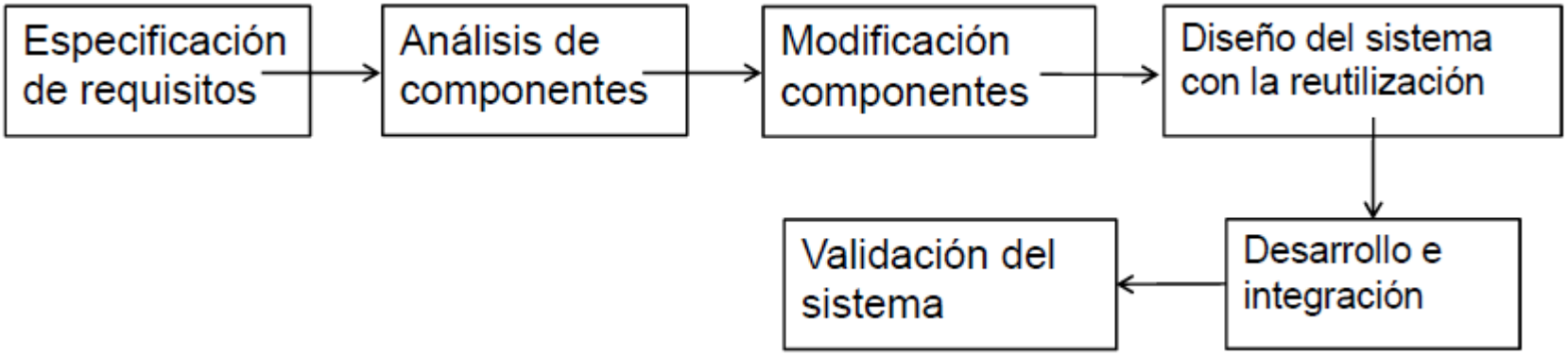
Modelos de procesos generales
Desarrollo formal de sistemas

Desventajas

- Las pruebas suelen ser largas y poco prácticas.
- Se utiliza principalmente en dominios muy especializados.
- Requiere la supervisión de un experto.
- No ofrece ventajas significativas si se compara con el modelo en cascada y el evolutivo.

Modelos de procesos generales
Desarrollo basado en la reutilización

- Se trata de utilizar diseños o código (componentes de software reutilizables) ya existentes que sean similares al que se necesita para el nuevo proyecto.
- Estos componentes se buscan y se modifican acorde a lo requerido y se incorporan en el sistema.



Modelos de procesos generales
Desarrollo basado en la reutilización

Ventajas

- Reduce la cantidad de software a desarrollar.
- Reduce costes y riesgos.
- Por lo general, conduce a una entrega más rápida del software.

Modelos de procesos híbridos
Modelo en Espiral

- Es una combinación del modelo en cascada y evolutivo (Boehm'88)
- Es un modelo evolutivo del desarrollo, formado por un conjunto de vueltas de espiral.
- ✓ En las primeras vueltas el SW es un modelo en papel, la especificación de un producto. Aún no funciona.
- ✓ En las sucesivas vueltas, se desarrolla un prototipo.
- ✓ En la últimas iteraciones se obtienen versiones completas del producto.

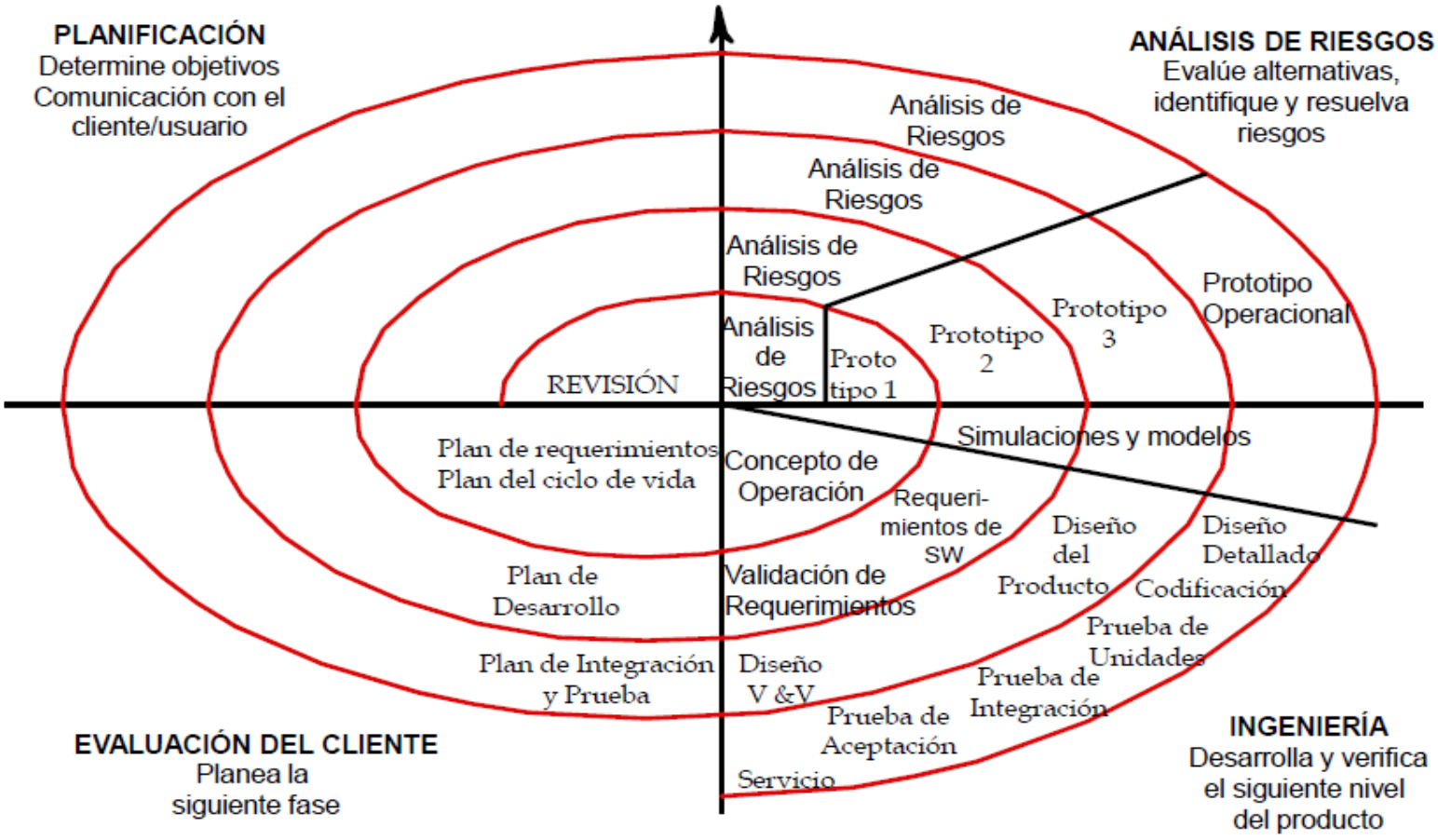
Modelos de procesos híbridos
Modelo en Espiral

- Cada ciclo del espiral representa una fase del proyecto software.
- 4 sectores por ciclo:
  - ✓ Definición de Objetivos
  - ✓ Evaluación y reducción de riesgos
  - ✓ Desarrollo y Validación
  - ✓ Planificación

Modelos de procesos híbridos
Modelo en Espiral

- Con este modelo obtenemos el producto final a partir de piezas más pequeñas.
- El número de actividades a realizar se incrementa notablemente a medida que nos alejamos del centro de la espiral. Las primeras son menos costosas.
- La evaluación después de cada fase permite cambios.
- Incorpora el factor Riesgo →es un modelo orientado a Riesgos.
- ✓ Tiene como objetivo vital pensar en las cosas que pueden ir mal en el desarrollo del software y saber cómo resolverlas.

Modelos de procesos híbridos
Modelo en Espiral



Modelos de procesos híbridos
Modelo en Espiral - Fases

Planificación

- Comunicación con el cliente.
- Determinar los objetivos.
- Determinar las alternativas de desarrollo.
- Analizar las restricciones de cada alternativa.

Análisis de riesgos

- Ver todos los puntos que pueden fallar.
- Evaluar las diferentes alternativas.
- Determinar y resolver o minimizar los riesgos.
- Identificación de los riesgos para cada alternativa, así como la manera de resolverlos.

Modelos de procesos híbridos
Modelo en Espiral - Fases

Ingeniería

- ✓ Desarrollo del producto.
- ✓ En cada iteración el proyecto se va completando.

Evaluación del cliente

- ✓ Revisión para ver si está de acuerdo, o no, con los resultados obtenidos. Si todo va bien, se pasa a la siguiente fase.
- ✓ En la revisión participan todas las personas y organizaciones que tienen relación con el producto.
- ✓ Se planifica la siguiente vuelta. Previsión de los recursos necesarios.

Modelos de procesos híbridos
Modelo en Espiral

Desventajas

- Es difícil establecer los hitos para determinar si podemos pasar a la siguiente vuelta de espiral.
- La evaluación de riesgos es una tarea compleja Se necesitan expertos en evaluación de riesgos.
- Es difícil convencer a los clientes que un modelo evolutivo es controlable.
- No se aconseja para proyectos que tienen pocos riesgos. Demasiado coste.
- En definitiva, no es un modelo muy real ni claro.

Modelos de procesos híbridos
Modelo en Espiral

Ventajas

- Cuando tenemos proyectos complejos, donde el problema no está muy bien definido y conlleva una serie de riesgos.
- Proyectos dinámicos.
- Proyectos innovadores y ambiciosos.

Modelos de procesos híbridos
Modelo Incremental

- Es un tipo de modelo evolutivo →es iterativo: permite a los ingenieros desarrollar versiones cada vez más completas.
- Combina elementos del modelo en cascada (aplicados repetidamente) con la filosofía interactiva de la construcción de prototipos.
- Cada secuencia lineal produce un incremento → las entregas de los incrementos se definen al principio del proceso software.
- Cada entrega constituye un **producto operacional**.
- Es útil cuando el personal o los recursos no están disponibles hasta cierto tiempo dentro del proceso de desarrollo Se adapta a entornos de alta incertidumbre.

Modelos de procesos híbridos
Modelo Incremental

- El diseño e implementación del software se dividen en una serie de incrementos los cuales se desarrollan uno a uno y se van integrando hasta llegar al producto final.
- Cada incremento es un subconjunto de funcionalidades del sistema.
- Las funcionalidades se agrupan por grado de importancia.
- Los servicios de prioridad más alta son los que se entregan antes al cliente.
- Una vez se han identificado los incrementos, se desarrollan con el modelo más adecuado. Dependiendo de cómo estén de bien definidos sus requisitos funcionales se va a utilizar un modelo en cascada o un modelo evolutivo.

Modelos de procesos híbridos
Modelo Incremental

Ventajas:

- El hecho de que el primer incremento recoja los requisitos más importantes y críticos hace que el cliente obtenga un producto más o menos útil en las primeras iteraciones.
- Existe bajo riesgo de fallar por la participación continua del usuario.

Desventajas:

- Conviene que los incrementos no sean muy grandes.
- Es difícil obtener incrementos equilibrados.

Modelos de procesos
Otros

- DRA (*Desarrollo Rápido de Aplicaciones*)
- Proceso Unificado
- Espiral WINWIN
- Desarrollo concurrente
- Técnicas de 4ª generación
  - ✓ Modelos para Desarrollo OO:
  - ✓ Modelo de Agrupamiento
  - ✓ Modelo fuente
  - ✓ Modelo remolino
  - ✓ Modelo Pinball, etc.

¿Preguntas?