



Departamento de Informática
Universidad de Valladolid
Campus de Segovia

TEMA 2: EL CICLO DE VIDA DEL SOFTWARE

DEFINICIÓN DE CICLO DE VIDA DEL SOFTWARE

- **ISO/IEC 12207-1**

- Marco de referencia que contiene los **procesos**, las **actividades** y las **tareas** involucradas en el desarrollo, la explotación y el mantenimiento de un producto de software abarcando la vida del sistema desde la definición de requisitos hasta la finalización de su uso.

- **IEEE 1074**

- Una **aproximación lógica** a la adquisición, el suministro, el desarrollo, la explotación y el mantenimiento del software.

CARACTERÍSTICAS DEL CICLO DE VIDA DEL SOFTWARE

- Se considera ciclo de vida de un producto software (C.V.S.) a la evolución del Producto desde su concepción hasta su retirada.
- Las actividades del ciclo de vida deben ser separadas en Fases.
- El ciclo de vida software debe dar respuesta a las siguientes preguntas de gestión de un proyecto:
 - ¿Qué haré a continuación?
 - ¿Cuánto tiempo continuaré haciéndolo?

MODELOS DE PROCESOS

- **Los modelos de procesos** son modelos de referencia que determinan que procesos se han de realizar para desarrollar el software.
- **El ciclo de vida**, para los distintos modelos, determina en que orden y cuales son las actividades y tareas que se han de realizar. (método de desarrollo concreto).
- Una **actividad** es un conjunto de tareas.
- Una **tarea** es una acción que transforma entradas en salidas.

MODELOS DE CICLO DE VIDA

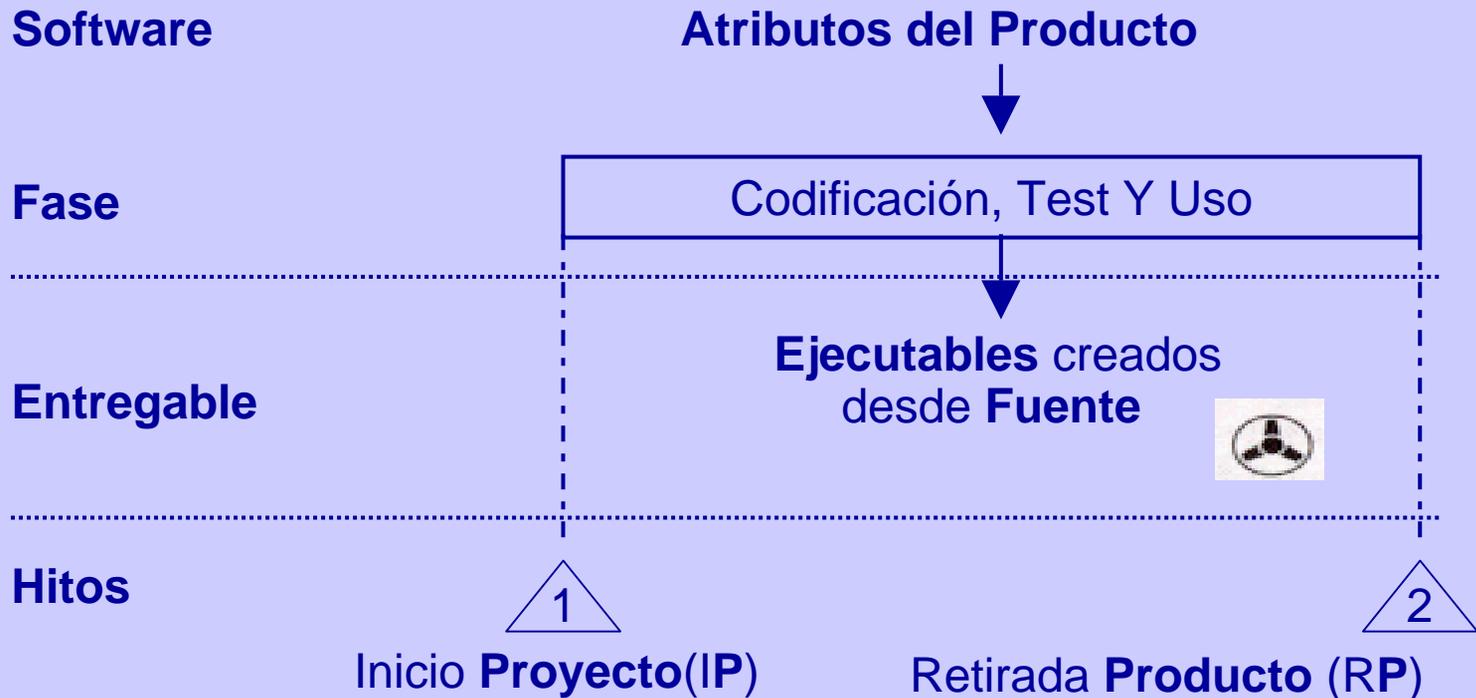
- Primeros ciclos de vida. Preingeniería del software hasta 1968.
 - CVS de una fase.
 - CVS de dos fases.
 - CVS de tres fases
- Desde 1968 hasta 1985.
 - Cascada(waterfall).
 - Prototipos.
 - Incremental.
- Ciclo de vida de procesos. Desde 1985 hasta el presente.
 - Espiral.

DEFINICIÓN DE HITO Y DE PRODUCTO A ENTREGAR

- **Hito:** Son puntos finales de una actividad en lo cuales se presentan informes cortos con los logros alcanzados.
 - Comienzo
 - Fases intermedias donde:
 - Se producen y se validan los subproductos obtenidos.
 - En el hito de viabilidad se toma la decisión sobre la continuidad del proyecto.
 - No se pasa al siguiente proceso hasta que se da el visto bueno
 - Finalización y entrega
- **Producto a entregar:** son resultados del proyecto que se entregan a los clientes.

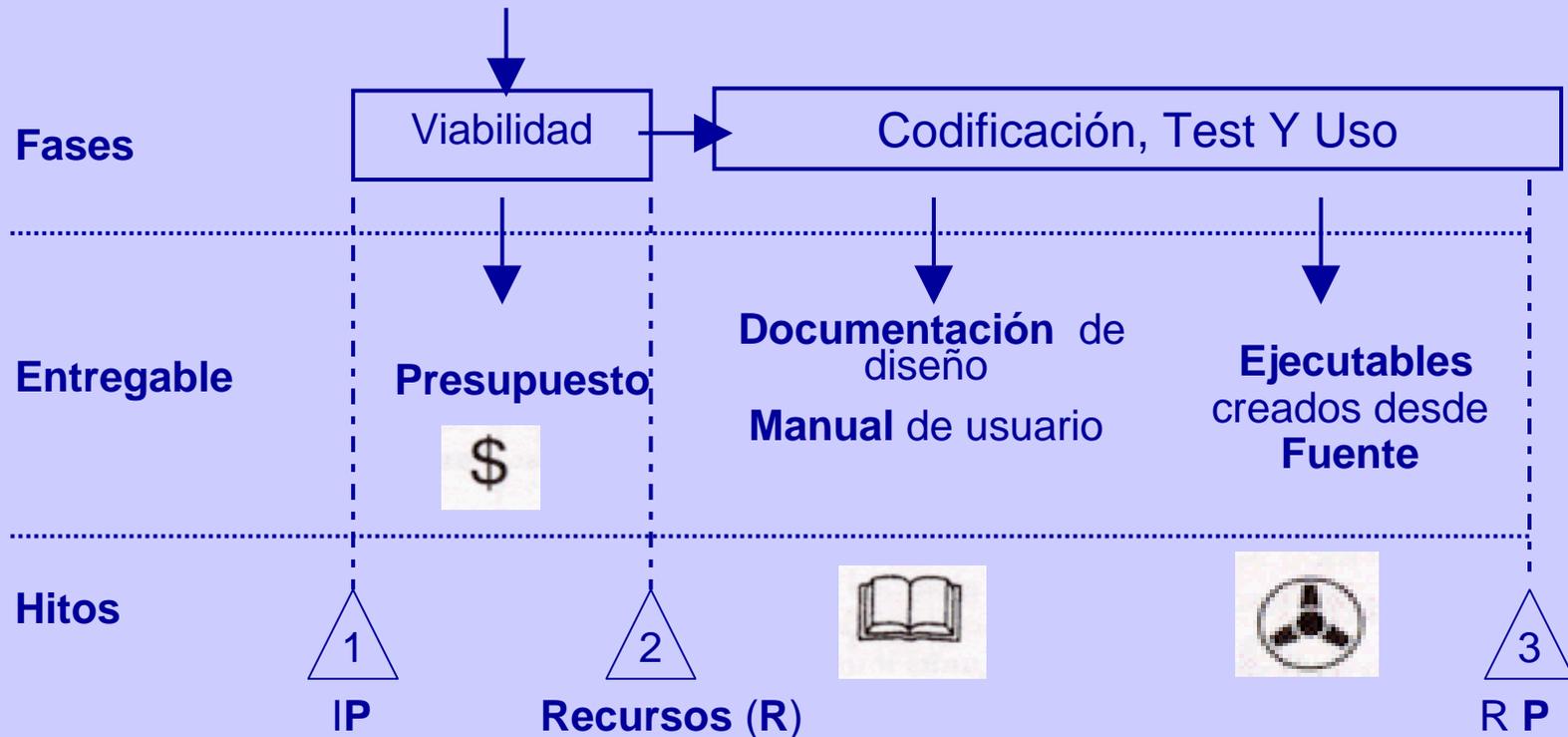
PRIMEROS CICLOS DE VIDA. PREINGENIERÍA DEL SOFTWARE A 1968

- CVS de una fase.



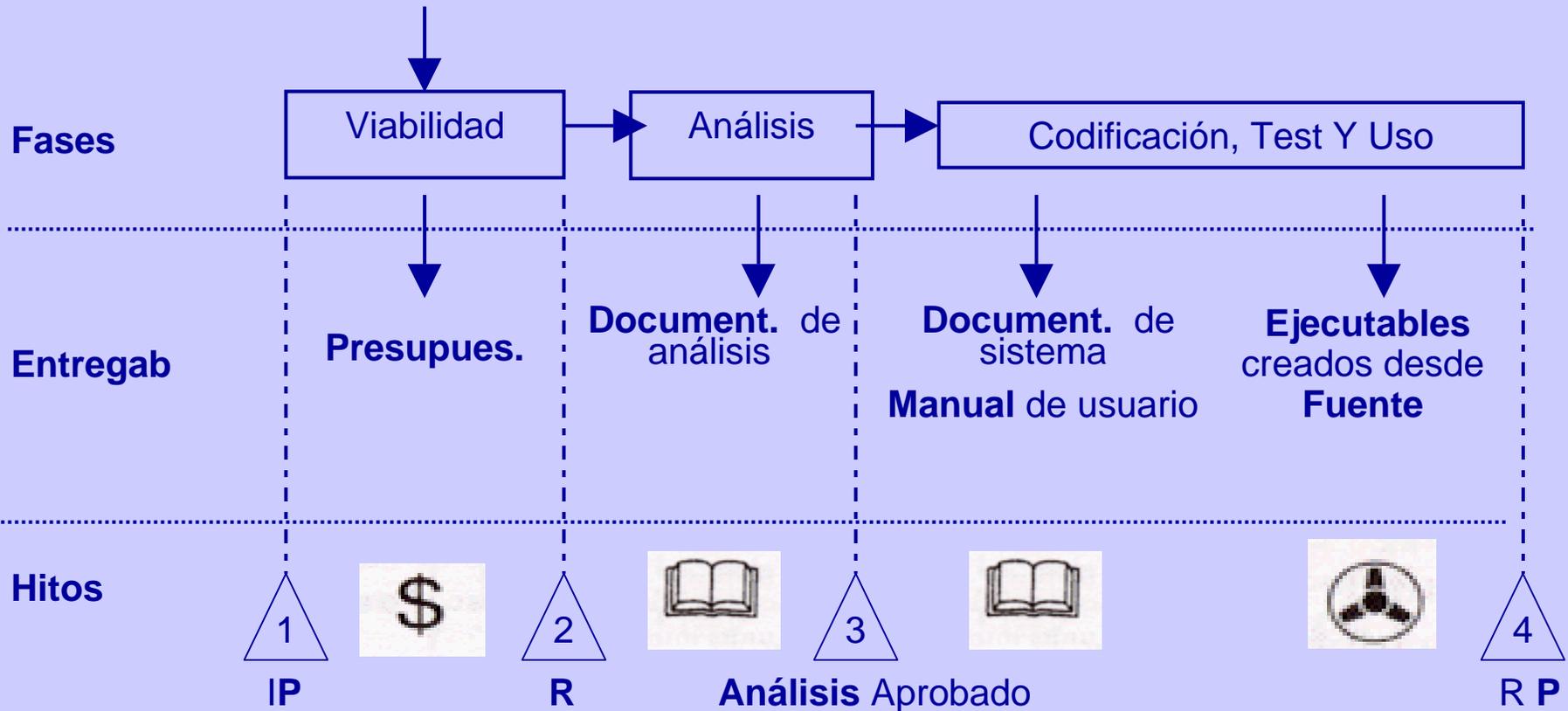
PRIMEROS CICLOS DE VIDA. PREINGENIERÍA DEL SOFTWARE A 1968

- CVS de dos fases.



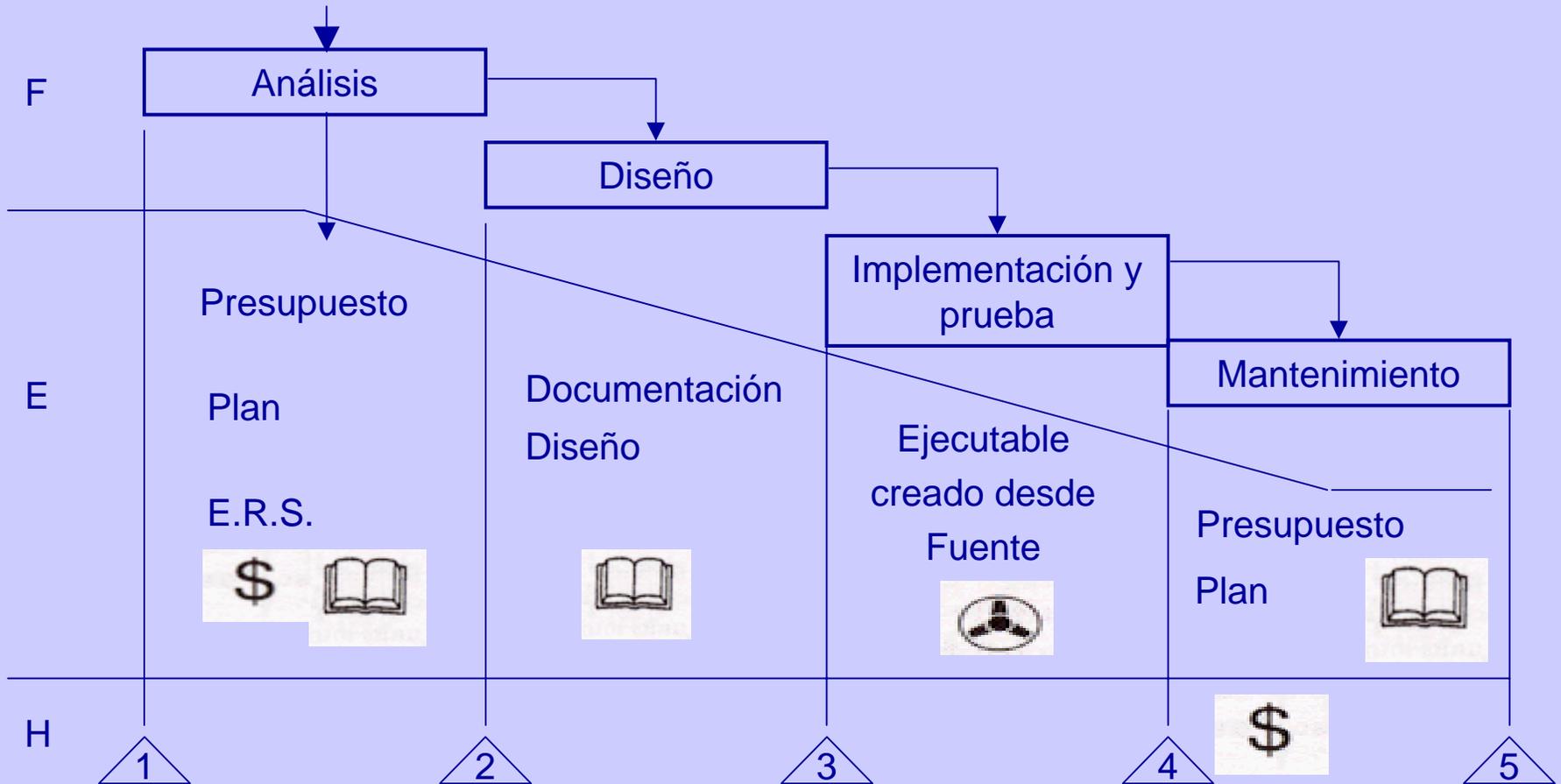
PRIMEROS CICLOS DE VIDA. PREINGENIERÍA DEL SOFTWARE A 1968

- CVS de tres fases.



DE 1968 A 1985.

- Waterfall (modelo lineal secuencial)



CARACTERÍSTICAS DE ESTE MODELO

- Cada fase empieza cuando se ha terminado la fase anterior.
- Para poder pasar a la siguiente fase se deben haber cumplido todos los objetivos marcados.
- Al final de cada fase tanto usuarios como personal técnico tienen la oportunidad de de revisar de forma conjunta el progreso del proyecto.

ESPECIFICACIÓN DEL SOFTWARE

- Etapa en la se establece que requisitos se requieren del sistema y cuales son las restricciones de operación y desarrollo del mismo.
- Se obtiene un documento de requisitos, con la especificación del sistema.
- Fases de la ingeniería de requisitos:
 - Estudio de factibilidad
 - Obtención y análisis de requisitos
 - Especificación de requisitos: los del usuario y los del sistema.
 - Validación de requisitos

DISEÑO DEL SISTEMA Y SU IMPLEMENTACIÓN.

- Etapa en la que se convierte la especificación del sistema en un sistema ejecutable.
- Diseño del software
 - Descripción de la estructura del software, los datos, las interfaces entre componentes.
- Implementación
 - Transformar la estructura anterior en un programa ejecutable.

VALIDACIÓN Y PRUEBA DEL SOFTWARE

- La verificación y la validación pretenden demostrar que un sistema es conforme con su especificación y que resuelve los requisitos del cliente.
- La prueba del sistema implica ejecutar el sistema con los casos de prueba que se obtuvieron en la especificación.

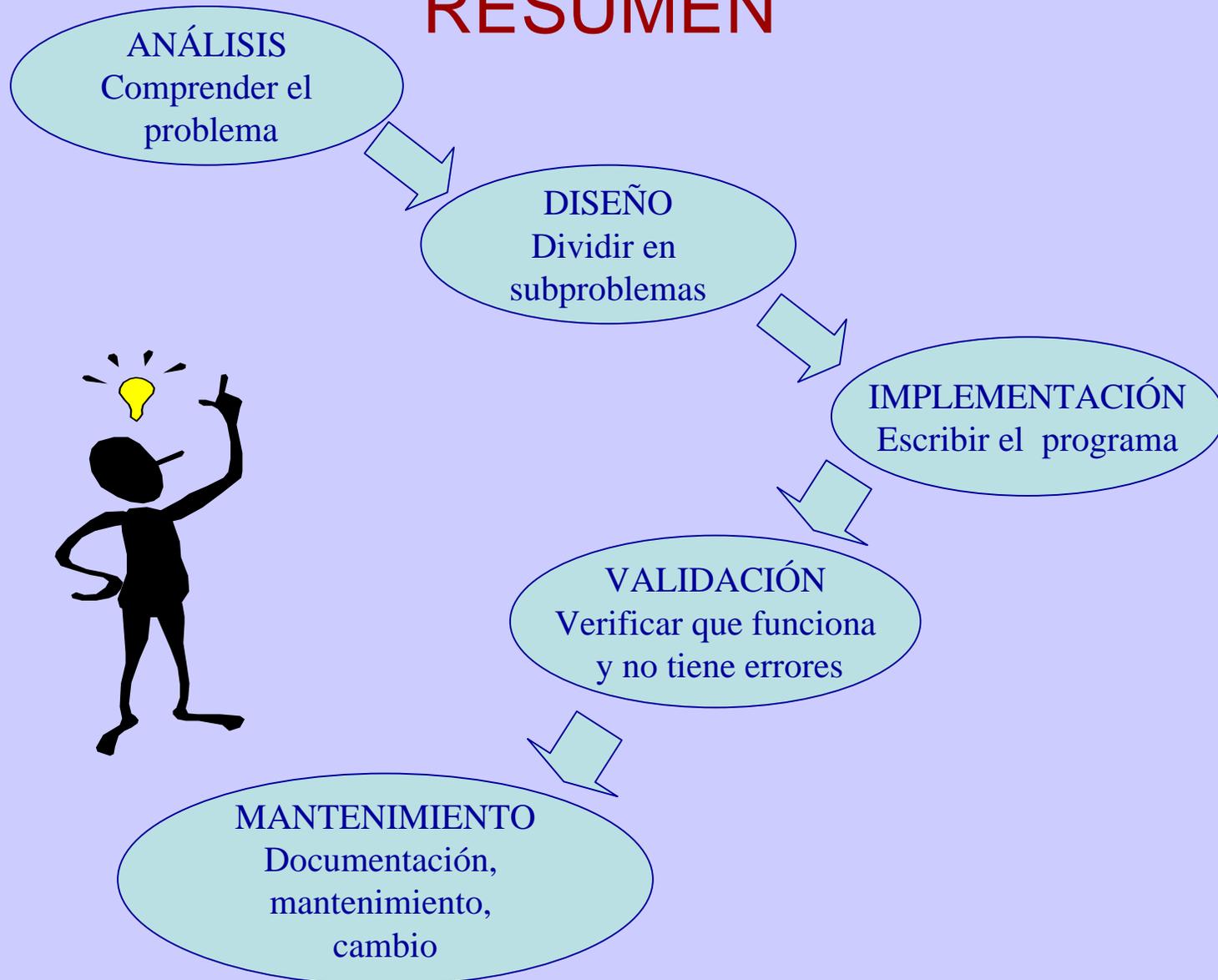
ETAPAS DEL PROCESO DE PRUEBAS

- Prueba de unidades
 - Se comprueban los componentes individuales.
- Prueba de módulos
 - Se prueban colecciones de componentes dependientes
- Prueba de subsistemas
 - Los módulos se integran en subsistemas y se prueban. Sobre todo se prueba el acoplamiento de las interfaces.
- Prueba de sistemas
 - Se prueban las interacciones entre los subsistemas y las propiedades emergentes.
- Prueba de aceptación
 - Se prueba con datos reales para comprobar que el sistema es aceptable por el cliente

MANTENIMIENTO.

- En esta fase se redacta la documentación tanto para el usuario como para su posterior mantenimiento.
- Se detectan y corrigen nuevos errores.
- Seguimiento de la aplicación de cara a posibles futuras modificaciones.
- Se puede acordar a priori un plan y un presupuesto de mantenimiento.

RESUMEN

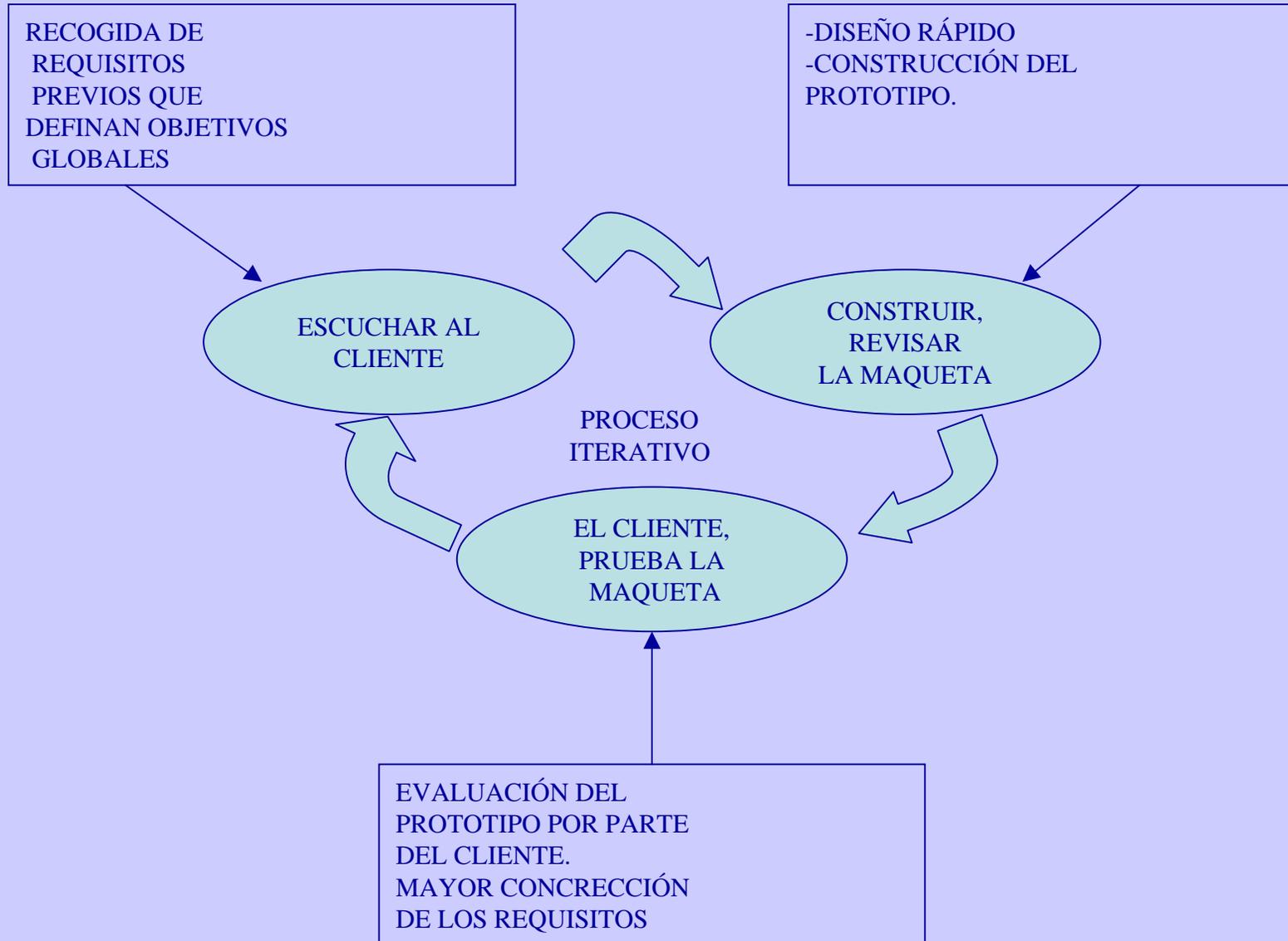


INCONVENIENTES DEL MODELO

- Los proyectos raras veces siguen una evolución secuencial.
- No todos los requisitos son expuestos, al principio, de forma explícita como requiere este modelo.
- El cliente debe tener paciencia, ya que la aplicación sólo estará disponible en un estado muy avanzado del proyecto.

DE 1968 A 1985.

- Prototipos desechables



INCONVENIENTES DEL MODELO

- El cliente puede pensar que el prototipo es una versión acabada.
- Esto implica que con las prisas no se han tenido en cuenta la calidad del software global o el mantenimiento a largo plazo.
- Las herramientas elegidas pueden ser inadecuadas.

APLICABILIDAD DEL MODELO

- Para sistemas interactivos pequeños o de tamaño pequeño.
- Para partes de sistemas grandes (e.g. el interfaz de usuario)
- Para sistemas con vida corta.

- **LA CLAVE DEL ÉXITO DE ESTE MODELO CONSISTE EN DEFINIR BIEN, DESDE EL PRINCIPIO, LAS REGLAS DEL JUEGO.**

(DIALOGO CLIENTE-PROGRAMADOR).

DE 1968 A 1985. MODELO INCREMENTAL

[Mills 71]

- El sistema no se entrega de una vez, sino que se divide y se entregan incrementos.
- Con cada incremento se entrega la parte de la funcionalidad que se ha establecido.
- Los requisitos son priorizados. Los requisitos con una más alta prioridad se incluyen en los incrementos más tempranos.
- Los requisitos de un incremento son inamovibles. Sin embargo estos puede verse modificados en incrementos posteriores.

DE 1968 A 1985. MODELO INCREMENTAL [Mills 71]

- Este proceso se repite hasta la obtención de un producto completo.
- Tanto el modelo incremental como el de prototipos comparten una naturaleza interactiva.
- Sin embargo el modelo incremental se centra en la entrega de un producto operativo en cada incremento.

VENTAJAS E INCONVENIENTES DEL MODELO INCREMENTAL

- Los clientes no tienen que esperar hasta tener el sistema completo. El primer incremento satisface los requisitos más críticos.
- Los primeros incrementos sirven como prototipo y ayudan en la tarea de detectar los posteriores requisitos.

VENTAJAS E INCONVENIENTES DEL MODELO INCREMENTAL

- Existe un riesgo bajo de fallar en el proyecto total.
- Los servicios del sistema con la prioridad más alta tienden a ser los más probados.
- Puede ser difícil ajustar los requisitos a los incrementos.

MODELOS EVOLUTIVOS. EL MODELO ESPIRAL

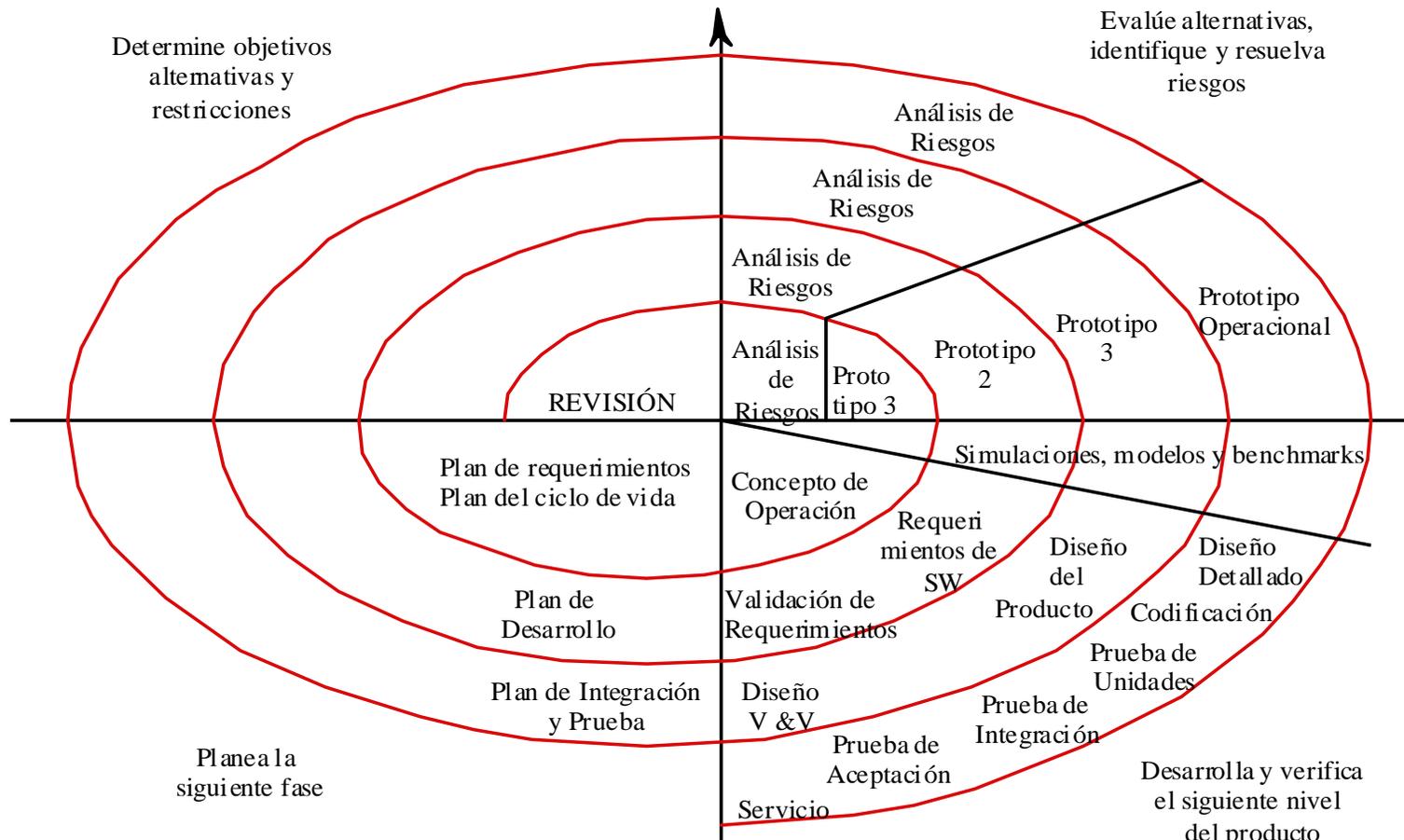
- Propuestos originalmente por Boehm. [BOE 88]*
- Cada vuelta en la espiral representa una fase del proceso.
- No hay fases fijas, cada vuelta en la espiral determina las actividades a realizar.

*Boehm, B. "A Spiral Model for Software Development and Enhancement", Computer, vol. 21, nº 5 Mayo 1988.

MODELOS EVOLUTIVOS. EL MODELO ESPIRAL

- La dimensión radial representa el coste acumulado en la financiación de las fases.
- La dimensión angular representa el progreso hecho en completar cada ciclo de la espiral.
- Un ciclo a través de la espiral es simular un paso a través de un modelo en cascada

Modelo de Proceso de Espiral



SECTORES DEL MODELO EN ESPIRAL

- Definición de objetivos
 - Se identifican los objetivos de cada fase, las alternativas y las restricciones
- Evaluación y reducción de riesgos
 - Se evalúan las distintas alternativas en función de los objetivos y las restricciones.
 - Se determinan los riesgos y se ponen en marcha las actividades que reduzcan dichos riesgos.
- Desarrollo y validación
 - Se elige el modelo de desarrollo más apropiado para el sistema de entre todos los modelos.
- Planificación
 - Se revisa el proyecto y se continua. Se planifica la siguiente fase (nueva vuelta de la espiral).