

Ingeniería de Software

Procesos de Software

Actividades del Proceso de Desarrollo de Software

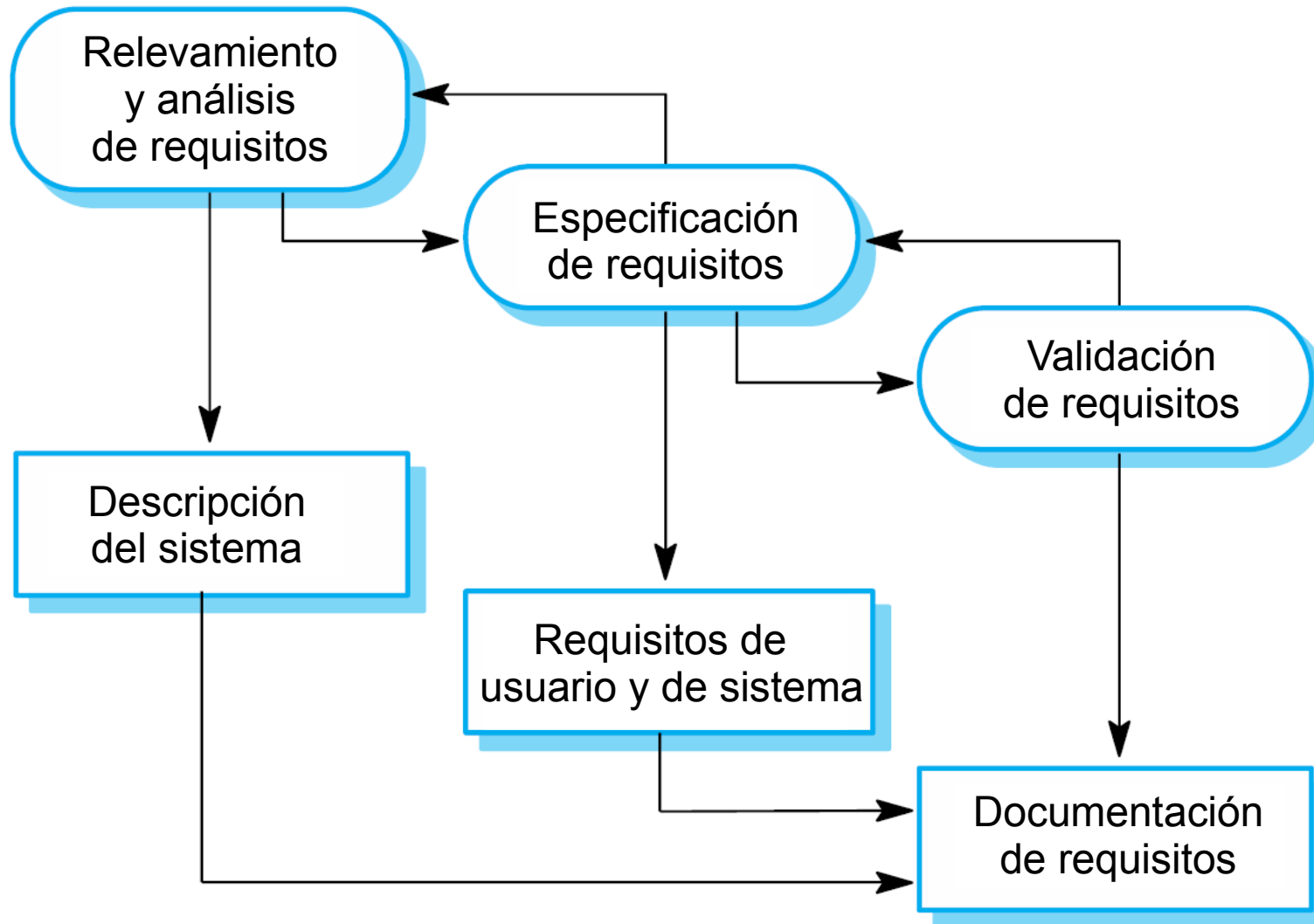
Sommerville capítulo 2
Sección 2.2 Process activities

Actividades del proceso de software

- Los procesos “reales” de software son secuencias intercaladas de actividades técnicas, colaborativas y de gestión, con el objetivo general de especificar, diseñar, implementar y probar el sistema de software.
- Las cuatro actividades de proceso básicas (**especificación, desarrollo, validación y evolución**) son organizadas de forma diferente en los diferentes procesos de desarrollo.
- Por ejemplo: en el modelo en cascada se organizan de forma secuencial y en el incremental de forma intercalada.



Proceso de Ingeniería de Requisitos



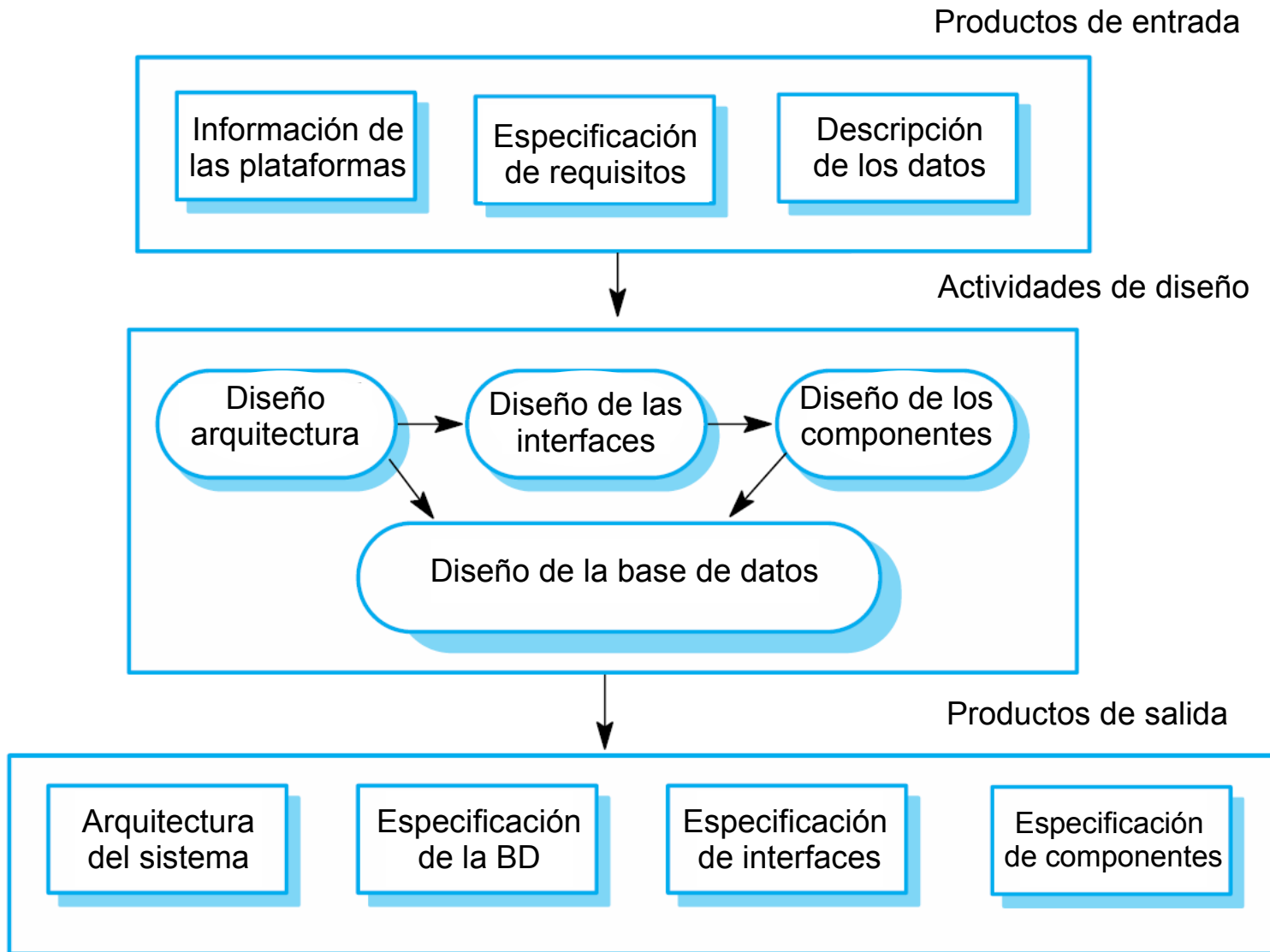
Especificación de Software

- Es el proceso de establecer qué servicios (o funcionalidades) son requeridos y las restricciones acerca de la operación y el desarrollo.
- Proceso de ingeniería de requisitos
 - **Relevamiento y análisis de requisitos:** ¿Qué es lo que el cliente y los interesados requieren y esperan del sistema?
 - **Especificación de requisitos:** definir los requisitos en detalle.
 - **Validación de requisitos:** comprobar la validez de los requisitos.

Diseño e implementación de software

- Es el proceso de convertir la especificación del sistema en un sistema ejecutable
 - **Diseño de software:** diseña la estructura del software que satisface su especificación.
 - **Implementación:** traducir dicha estructura en un programa ejecutable.
- Las actividades de diseño e implementación están estrechamente relacionadas y pueden realizarse de forma intercalada.

Proceso general de diseño



Actividades de diseño

- **Diseño de arquitectura:** identificar todas las estructuras del sistema, los principales componentes (subsystemas o módulos), sus relaciones y cómo están distribuidos.
- **Diseño de la base de datos:** diseñar las estructuras del sistema de tados y cómo éstos serán representados en la base de datos.
- **Diseño de interfaz de usuario:** se definen las interfaces entre los componentes del sistema.
- **Selección y diseño de componentes:** se buscan componentes reutilizables. En caso de que no estén disponibles, se diseña cómo deberían operar (para luego ser implementados).

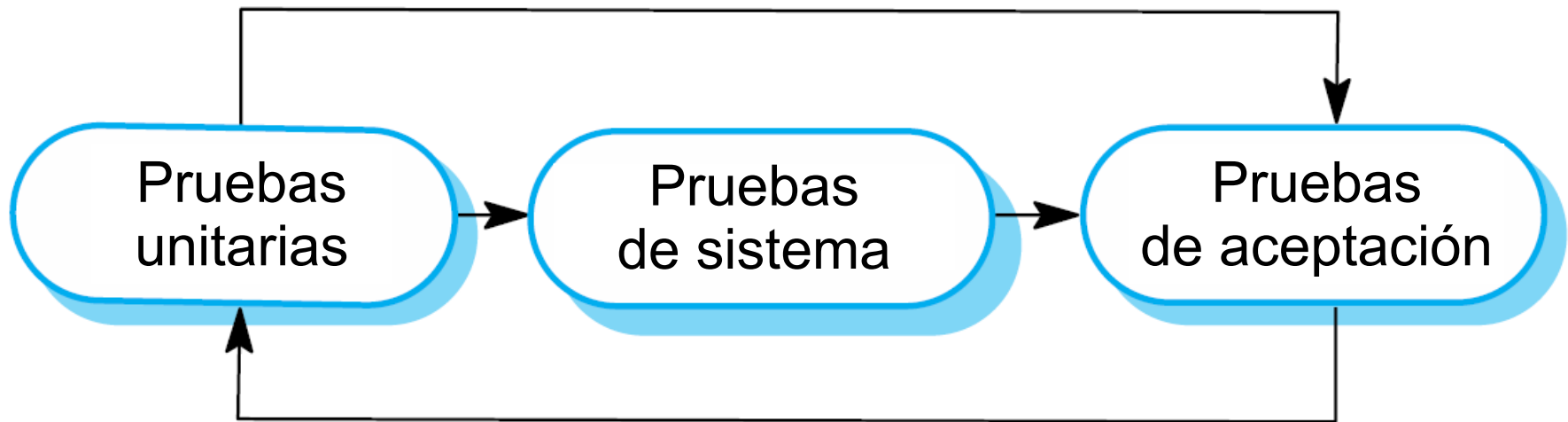
Implementación del sistema

- El programa es desarrollado mediante la implementación del programa (o programas), o mediante la configuración de una aplicación de sistema.
- Diseño e implementación son actividades intercaladas en la mayoría de los sistemas de software.
- La programación es una actividad individual sin un proceso estándar.
- La actividad de “debugging” refiere a encontrar las faltas en los programas y corregirlas.

Validación de software

- **La verificación y la validación (V & V)** pretende mostrar que el sistema se ajusta a su especificación y cumple con los requisitos del cliente.
 - Involucra los procesos de comprobación, revisión y pruebas de sistema.
 - Las pruebas del sistema involucran la ejecución del sistema con casos de prueba, los cuales se derivan de la especificación del mismo e intentan simular un procesamiento real del sistema.
- Las pruebas de software son unas de las actividades más comunmente usadas en las actividades de V & V.

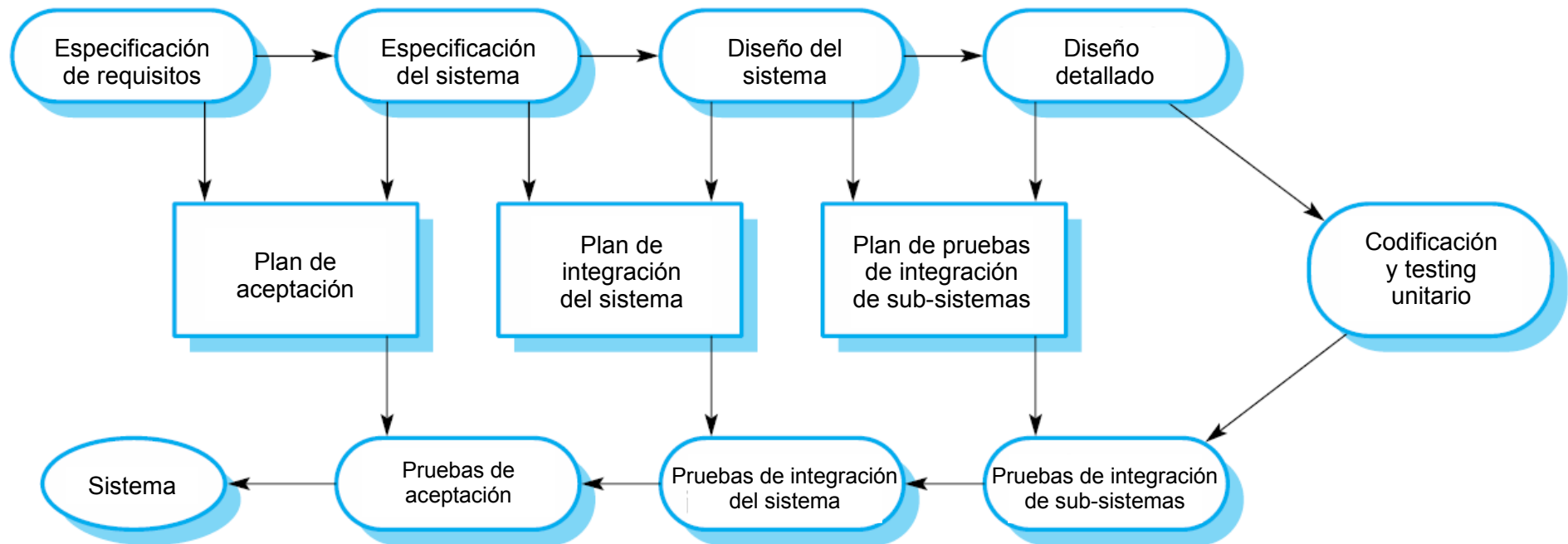
Etapas del testing



Etapas del testing

- **Pruebas unitarias:** unidades (o componentes) de software son probados de forma independiente.
 - Pueden ser funciones, objetos, o grupos (de funciones y objetos) agrupados de forma coherente.
- **Pruebas de sistema:** implica probar el sistema como un todo.
- **Pruebas de aceptación:** se prueba el sistema con datos del cliente para comprobar que el sistema cumple con sus necesidades.

Fases del testing en un proceso dirigido por planes (modelo en V)



Evolución de Software

- El software puede cambiar...
- Cuando cambian las circunstancias de negocio, cambian los requisitos del software. Por tanto el software debe evolucionar y cambiar (o se vuelve inútil y obsoleto)
- Aunque ha habido una demarcación entre “desarrollo” y “evolución” (mantenimiento), esto es cada vez más irrelevante a medida de que ya casi no existen sistemas “completamente nuevos”

Evolución del sistema

