

Ingeniería de Software

Procesos de Software *Hacer frente al cambio*

Sommerville capítulo 2
Sección 2.3 Coping with change

Hacer frente al cambio...

- El cambio es inevitable en todos los grandes proyectos de software
 - Los cambios en el negocio conducen a cambios en los requisitos de los sistemas
 - Emergen nuevas tecnologías las cuales posibilitan mejoras en la implementación
 - Cambios en las plataformas requieren cambios en las aplicaciones
- Los cambios conducen al **retrabajo**, por tanto el costo de un cambio incluye tanto el costo del retrabajo (ej. re-analizar los requisitos) como el costo de la implementación de esa nueva funcionalidad



Reduciendo el costo del retrabajo

- **Anticipación al cambio:** incluye actividades que puedan anticipar los posibles cambios antes de que el re-trabajo requerido sea muy grande
 - Por ejemplo: se puede desarrollar un prototipo del sistema que muestre algunas funcionalidades claves a los clientes para su validación
- **Tolerancia al cambio:** el proceso es diseñado de forma tal que los cambios pueden ser acomodados con un costo relativamente bajo
 - Esto normalmente involucra desarrollo incremental

Prototipado de software

- Un prototipo es una versión inicial de un sistema, el cual se utiliza para demostrar conceptos y probar opciones de diseño
- Un prototipo puede ser usado en:
 - El proceso de ingeniería de requisitos, para relevar y validar requisitos
 - En el proceso de diseño, para explorar opciones y desarrollar la interfaz de usuario (UI)
 - En el proceso de pruebas, para ejecutar pruebas “back-to-back”

Beneficios del prototipado

- Mejora la usabilidad del sistema
- Obtiene una mirada más certera de las reales necesidades del cliente
- Mejora la calidad del diseño
- Mejora la mantenibilidad
- Reduce el esfuerzo en retrabajo

Desarrollo del Prototipo

- Puede ser basado en lenguajes o herramientas de prototipado rápido
- Puede involucrar dejar de lado alguna funcionalidad
 - Debe enfocarse en las áreas del producto que aún no se entienden por completo
 - No es necesario que incluya recuperación y chequeo de errores
 - Generalmente se enfocan más en los requisitos funcionales que en los no-funcionales, como ser confiabilidad o seguridad (aunque pueden haber prototipos técnicos que sí evalúen requisitos no-funcionales)

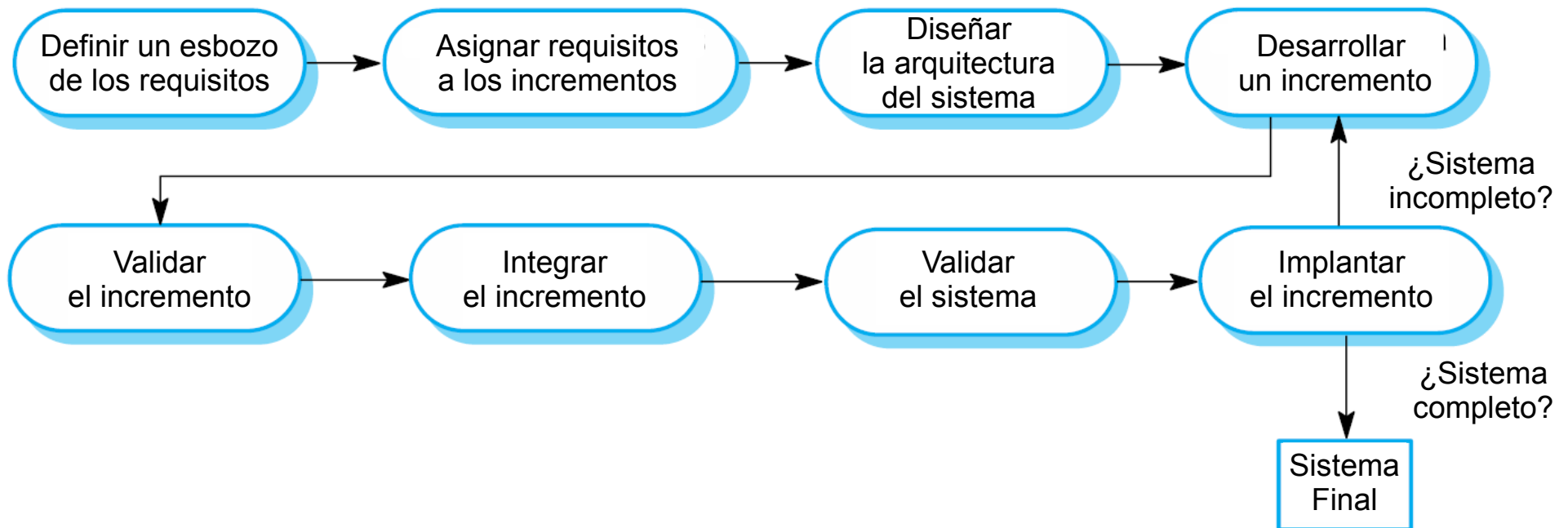
Prototipos descartables

- Los prototipos deben ser descartados si no conforman una buena base para el sistema en producción
 - Puede que sea imposible (o muy costoso) ajustarlo para que cumpla con requerimientos no funcionales
 - Muchas veces no se documentan
 - Si los cambios son muy rápidos y frecuentes, pueden degradar la estructura del prototipo fácilmente
 - Los prototipos generalmente no cumplen con los estándares de calidad organizacionales

Entregas incrementales

- Sería el mismo concepto de “**desarrollo en fases con liberaciones parciales**” de años anteriores.
- El desarrollo se particiona en incrementos, en donde cada incremento entrega parte de la funcionalidad requerida.
- Los requerimientos de usuario son priorizados y los más importantes se incluyen en incrementos tempranamente.
- Es difícil de implementar cuando se debe reemplazar un sistema por otro, ya que se tiene menos funcionalidad que el sistema que es reemplazado.

Entregas incrementales



Ventajas de Entregas Incrementales

- Disponibilidad temprana de funcionalidades de valor para el cliente.
- Las entregas tempranas actúan de “prototipos” que ayudan a validar y relevar requisitos para próximos incrementos.
- Las funcionalidades/servicios de más alta prioridad son las que reciben más testing.

Desventajas de Entregas Incrementales

- La mayoría de los sistemas requieren un conjunto básico de facilidades (core) que son usadas por las diferentes partes del sistema.
 - Si los requerimientos no están definidos en detalle, es difícil identificar ese conjunto de facilidades que serán necesarias para todos los incrementos.
- La esencia de los procesos iterativos es que la especificación es desarrollada en conjunto con el software.
 - Sin embargo, esto muchas veces genera conflicto con los modelos de adquisición de varias organizaciones, en donde la especificación completa del sistema es parte del contrato del desarrollo del mismo.