

Modelos de Proceso de Desarrollo del Software

MSc. Juan José Quesada Sánchez
ajuanjo@gmail.com



Agenda

- ▶ Concepto: modelo de proceso del software
- ▶ Factores que afectan los modelos de proceso
- ▶ Metodología
- ▶ Modelos de desarrollo
 - Cascada
 - Desarrollo Evolutivo
 - Desarrollo basado en componentes
 - Iterativo-incremental
 - Espiral
 - Prototipado



Modelo de Proceso del software

- ▶ “Un conjunto ordenado de actividades con resultados asociados que conducen la producción y evolución del software. Es una representación abstracta de un tipo de proceso de software” Kurbel (2008, p. 158)
- ▶ Conjunto formal de actividades y fases
 - ▶ Guían al equipo involucrado en el proyecto a través del desarrollo de un sistema



Modelo de Proceso de desarrollo del software

- ▶ **Actividades básicas en el desarrollo de un sistema de información:**
 - Requerimientos
 - Análisis
 - Diseño
 - Codificación
 - Pruebas
 - Implantación
 - Evolución
- ▶ **Existen una amplia variedad de modelos de proceso**
 - Cascada, Desarrollo Evolutivo, Desarrollo basado en Componentes, Iterativo-Incremental, etc.
 - Proponen perspectivas particulares y pueden seguir distintas rutas, pero comparten algunas características



Factores que afectan los Modelos de Proceso

► Necesidad de una definición clara del problema

- Los problemas que tiene el usuario deben ser bien comprendidos

► Las tareas requeridas para producir una solución bien-ingenieril

- El análisis-diseño-codificación-pruebas-mantenimiento aparece en la mayoría de modelos.
- Relaciones entre las tareas varía: secuencial/iterativo, funcionalmente independiente/relacionados, dinámico/estático



Factores que afectan los Modelos de Proceso

- ▶ El rol que juegan los afectados en el proceso de desarrollo
 - ▶ Incluyen: usuarios finales, patrocinadores, desarrolladores del sistema
 - ▶ Representa la dimensión de las personas en el proceso
 - ▶ Factor humano es preponderante
- ▶ Los entregables de documentación



Factores que afectan los Modelos de Proceso

- ▶ La salida esencial del proceso de desarrollo es un valor económico
 - ▶ Cuáles son los beneficios para el negocio?
 - ▶ Se produce alguna reducción de costos?
 - ▶ Satisfacción del cliente.



Metodología

- ▶ Se refiere a un marco de trabajo que se utiliza para estructurar, planificar y controlar el proceso de desarrollo de un sistema de información.
- ▶ Existe una gran variedad de estas metodologías, cada una con sus propias fortalezas y debilidades reconocidas.
- ▶ Cada una de las metodologías disponibles se adapta mejor a tipos específicos de proyectos, basado en diversas consideraciones técnicas, de proyectos de organización y equipo.



Conceptos Generales

- ▶ **Metodología:** Conjunto de procedimientos, técnicas, herramientas y un soporte documental que ayuda a los desarrolladores a realizar nuevo software.
- ▶ **Tarea:** Actividades elementales en que se dividen los procesos.
- ▶ **Procedimiento:** Definición de la forma de ejecutar la tarea.
- ▶ **Técnica:** Herramienta utilizada para aplicar un procedimiento. Se pueden utilizar una o varias.
- ▶ **Herramienta:** Para realizar una técnica, podemos apoyarnos en las herramientas software que automatizan su aplicación.
- ▶ **Producto:** Resultado de cada etapa.



Metodología vs Ciclo de Vida

- ▶ Una metodología puede seguir uno o varios modelos de ciclo de vida, es decir, el ciclo de vida indica qué es lo que hay que obtener a lo largo del desarrollo del proyecto pero no cómo hacerlo.
- ▶ La metodología indica cómo hay que obtener los distintos productos parciales y finales.
- ▶ Ciclo de Vida: “Periodo de tiempo que inicia cuando un producto de software es concebido y termina cuando este ya no está disponible para ser usado” (IEEE std610.12-1990)

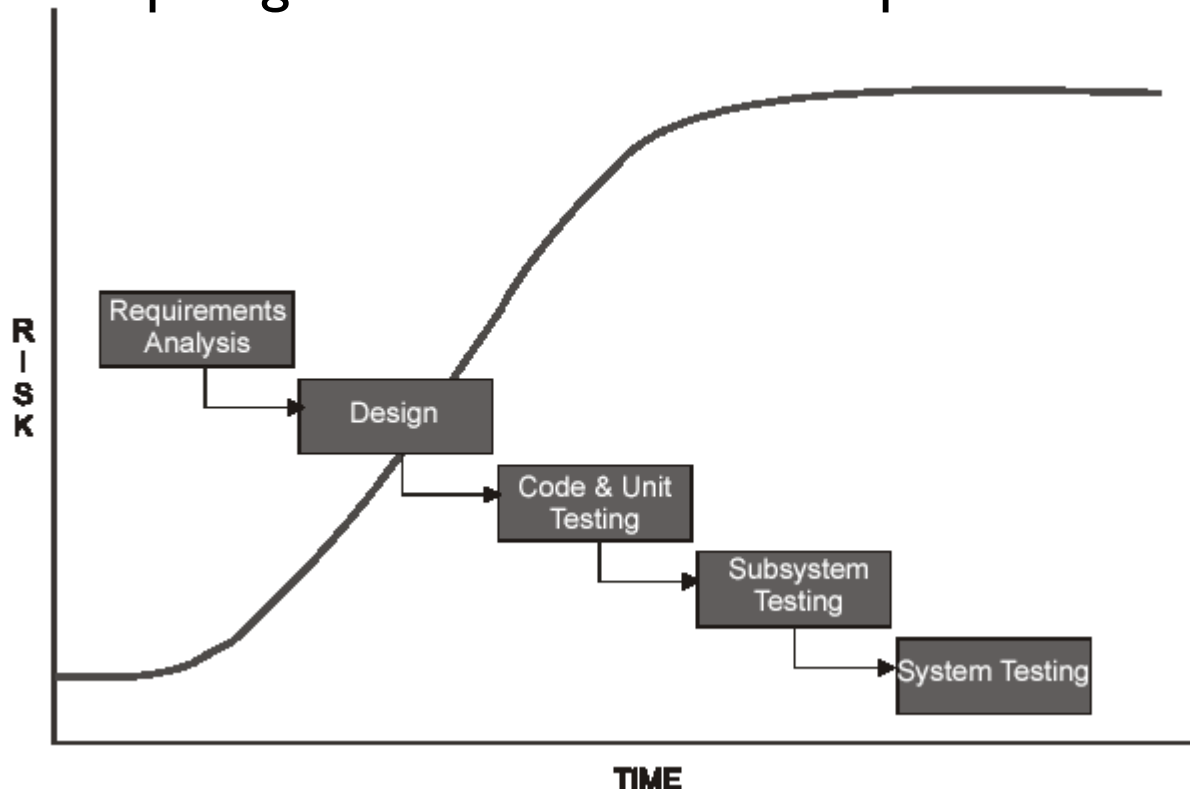


Características deseables de una Metodología

- ▶ Existencia de reglas predefinidas
- ▶ Cobertura total del ciclo de desarrollo
- ▶ Verificaciones intermedias
- ▶ Planificación y control
- ▶ Comunicación efectiva
- ▶ Utilización sobre un abanico amplio de proyectos
- ▶ Fácil formación
- ▶ Herramientas CASE
- ▶ Actividades que mejoren el proceso de desarrollo
- ▶ Soporte al mantenimiento
- ▶ Soporte de la reutilización de software

Desarrollo en Cascada

- ▶ Desarrollado por Winston Royce en 1970
- ▶ Proceso secuencial dividido en etapas con entradas y salidas
- ▶ La etapa siguiente inicia cuando la previa es completada





Desarrollo en Cascada

- ▶ Primer modelo de proceso de la Ingeniería del Software
- ▶ Conocido como: “Modelo del Ciclo de Vida del Software”
- ▶ Uno o más documentos deben ser aprobados antes del inicio de la siguiente etapa



Ventajas del desarrollo en cascada

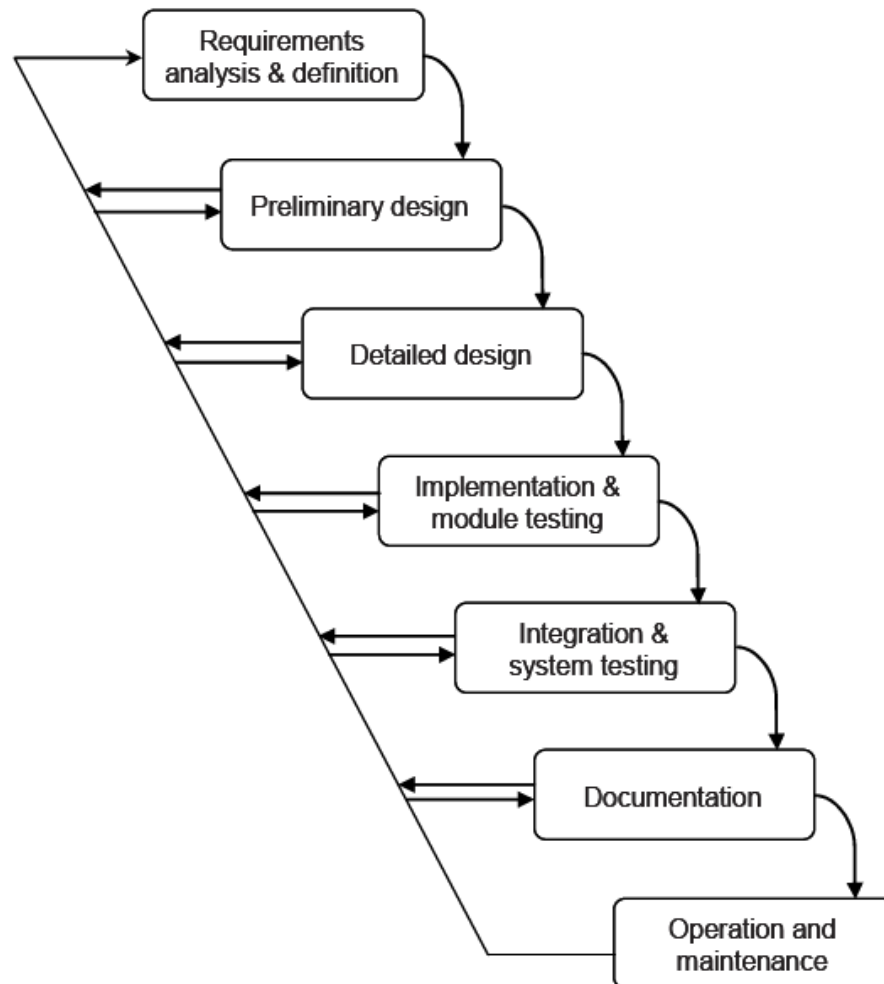
- ▶ Cada fase deriva en puntos de control y entregables
- ▶ Útil cuando
 - el dominio del problema es bien conocido
 - el producto estable y se conoce la tecnología
- ▶ Método estructurado y fácil de entender que funciona con gente de poca experiencia



Desventajas del desarrollo Cascada

- ▶ No es realístico esperar que una fase termine con resultados correctos
- ▶ El particionamiento inflexible del modelo hace difícil responder ante requerimientos nuevos o modificados
- ▶ Pocos sistemas tienen requerimientos estables
- ▶ Mucho retrabajo al final del ciclo

Desventajas del desarrollo Cascada



- Observar las implicaciones de los errores de diseño o fallas en la especificación de requerimientos detectados en la implementación

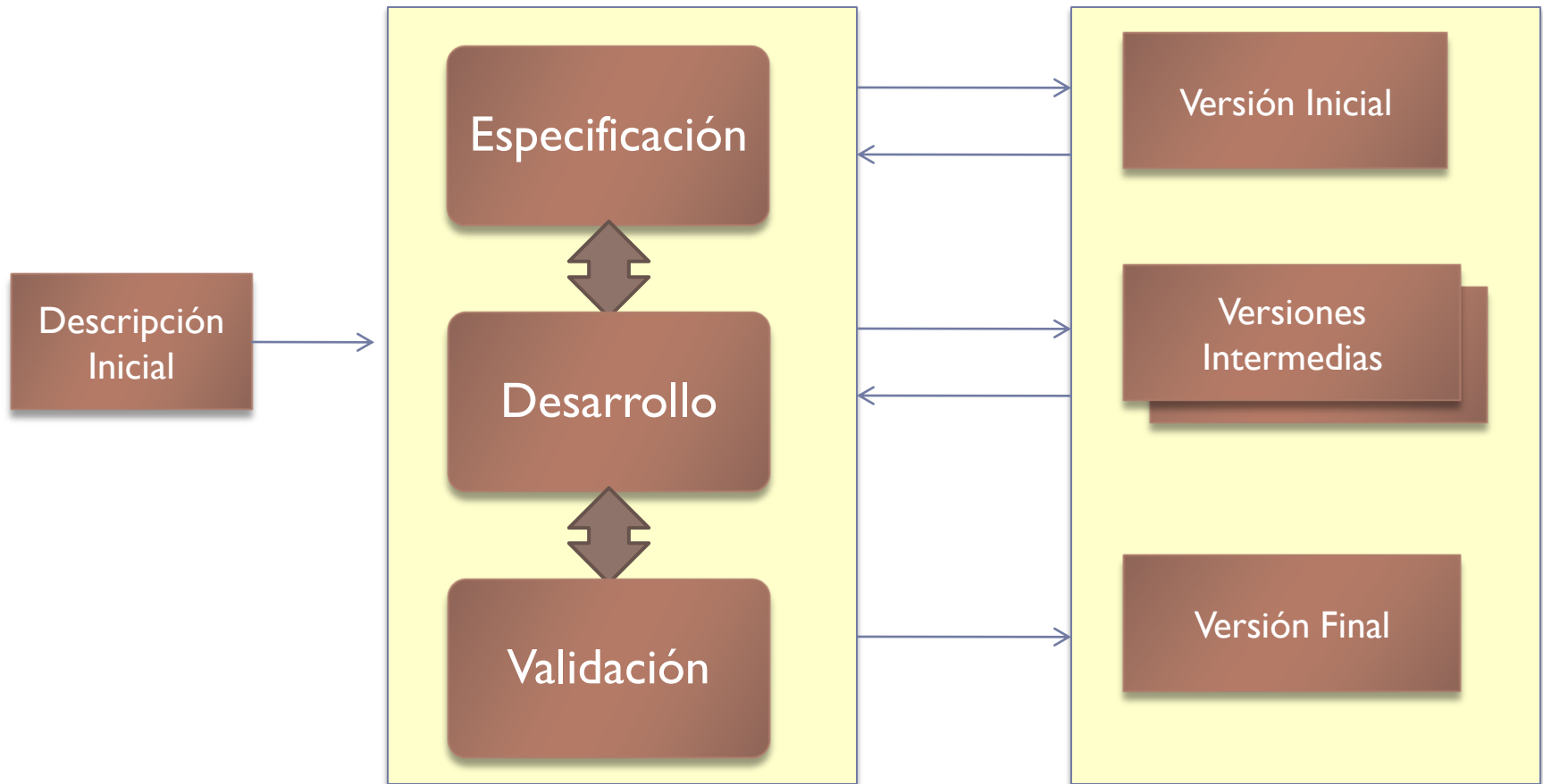


Desarrollo Evolutivo

- ▶ **Desarrollar una implementación inicial del sistema**
 - ▶ Exponerla a los comentarios del usuario
 - ▶ Refinar la versión inicial hacia nuevas versiones hasta llegar al sistema final
- ▶ **Dos tipos**
 - ▶ Desarrollo exploratorio
 - ▶ Prototipos desechables

Desarrollo Evolutivo

Actividades concurrentes





Desarrollo Evolutivo

▶ Desarrollo Exploratorio

- ▶ Explora los requerimientos del cliente para desarrollar un sistema final
- ▶ Inicia con partes del sistema mejor entendidas
- ▶ Se agregan nuevos atributos sugeridos por el cliente

▶ Prototipos desechables

- ▶ Experimentar con los requerimientos del cliente que no se comprenden del todo



Desarrollo Evolutivo

► Problemas

- Falta de visibilidad del proceso
- Los sistemas son pobremente estructurados
- Conocimientos especiales (lenguajes para prototipado rápido).

► Aplicabilidad

- Para sistemas interactivos pequeños o medianos
- Partes de sistemas grandes (GUI)
- Sistemas de corta vida



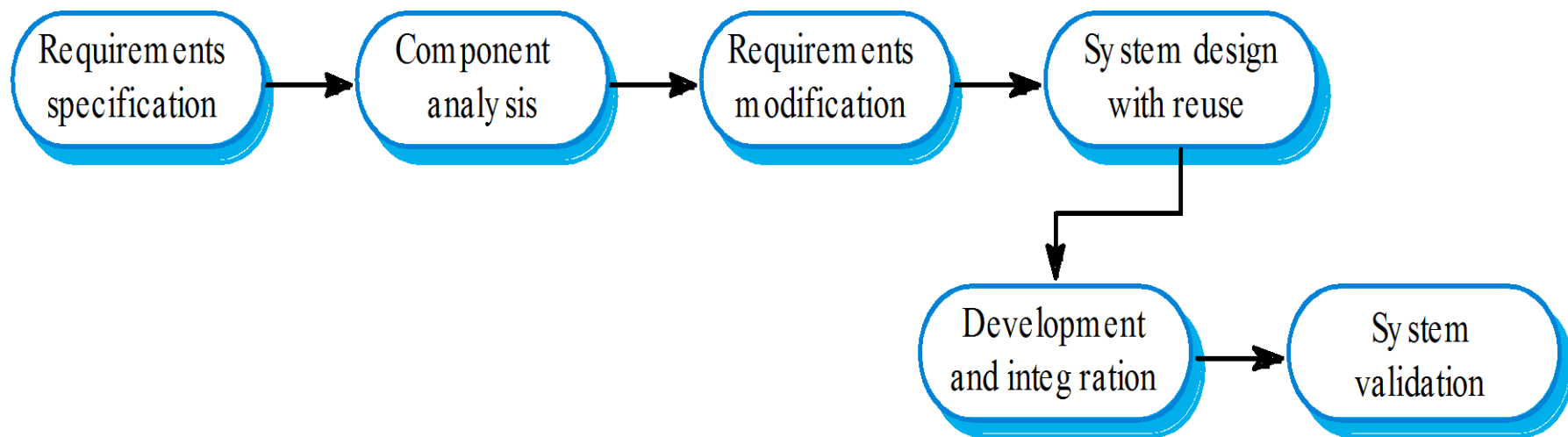
Desarrollo basado en Componentes

- ▶ Basado en la reutilización sistemática en donde los sistemas son integrados a partir de componentes existentes
- ▶ Etapas del proceso
 - Análisis de componentes
 - Modificación de requerimientos
 - Diseño del sistema con reutilización
 - Desarrollo e integración
- ▶ Se incrementa el uso conforme los estándares de componentes emergen



Desarrollo basado en Componentes

► Desarrollo orientado al reuso





Desarrollo basado en Componentes

▶ Ventajas

- ▶ Reducción de la cantidad de software a desarrollar → reducción de costos y riesgos
- ▶ Entrega más rápida

▶ Desventajas

- ▶ Se comprometen los requerimientos
- ▶ Se puede perder el control sobre la evolución del sistema

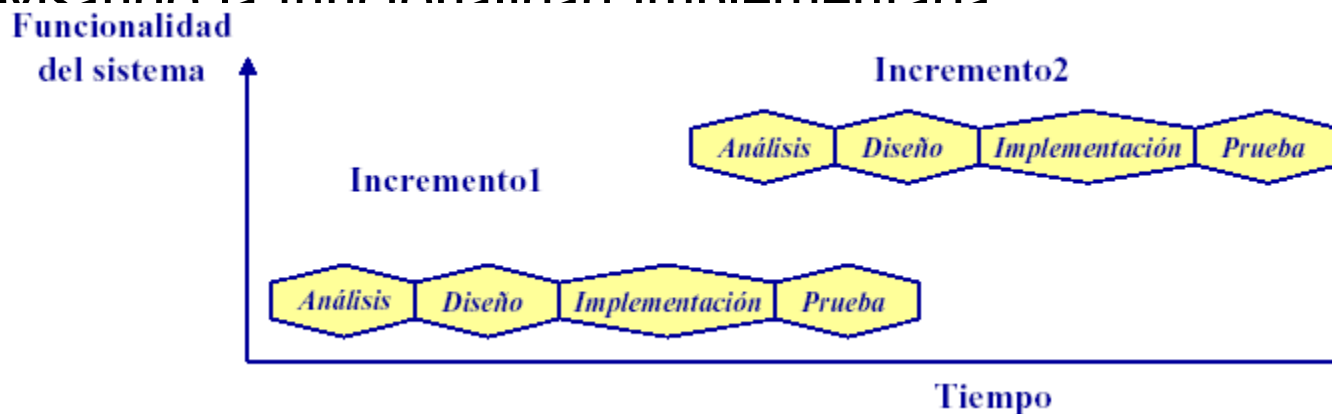


La iteración en el Proceso de Desarrollo

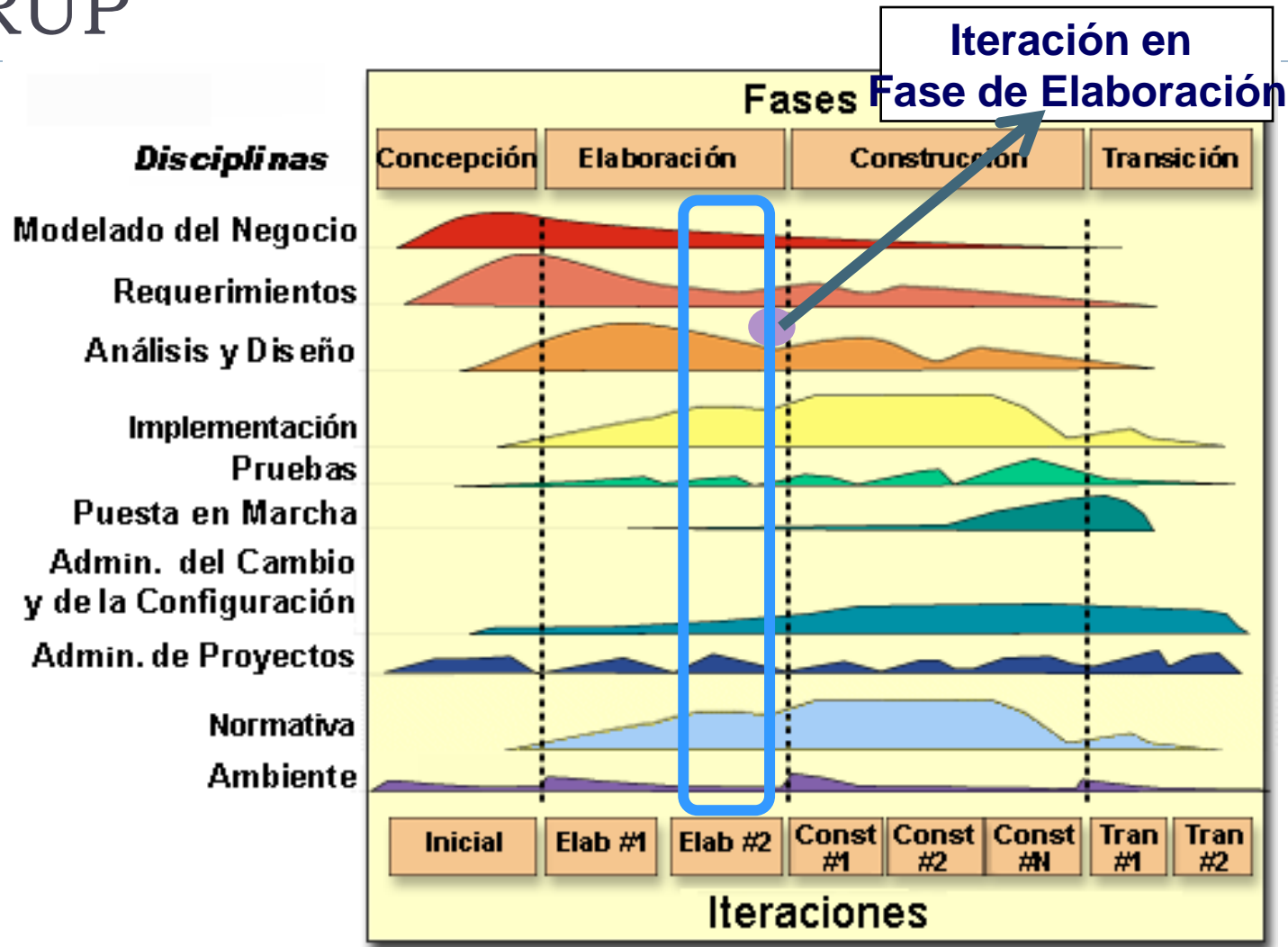
- ▶ Los requerimientos del sistema SIEMPRE cambian en el curso de un proyecto.
- ▶ La iteración del proceso en las etapas tempranas (implica retrabajo)
- ▶ Dos métodos:
 - ▶ Entrega incremental
 - ▶ Desarrollo en espiral

Desarrollo Iterativo-Incremental

- ▶ Divide el proyecto en varias partes o miniproyectos
- ▶ Cada miniproyecto es una **iteración** que resulta en un **incremento**
- ▶ Cada iteración es un ciclo de vida de desarrollo en miniatura
- ▶ Se basa en la ampliación y el refinamiento del sistema
- ▶ Cada iteración parte de la anterior incrementando o revisando la funcionalidad implementada

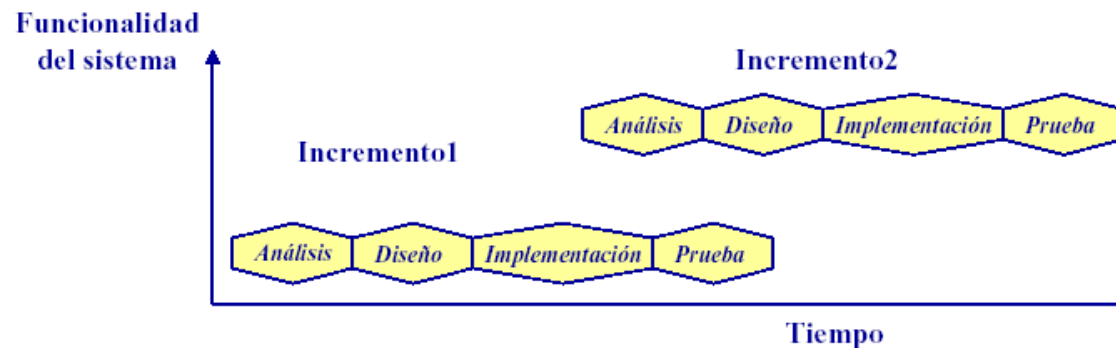


Desarrollo Iterativo-Incremental en RUP



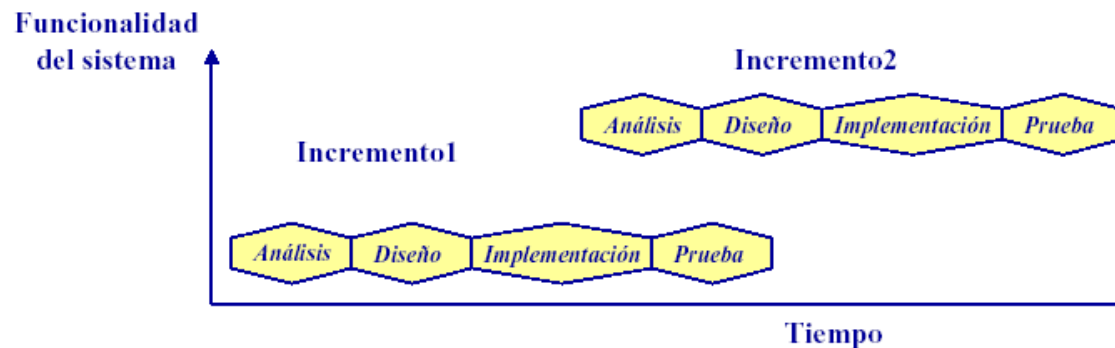
Ventajas desarrollo Iterativo-Incremental

- ▶ Omisiones serias son evidentes en las etapas tempranas del ciclo de vida, es posible reaccionar.
- ▶ Posibilita la retroalimentación del usuario en la elicitación de requerimientos.
- ▶ El equipo es forzado a enfocarse en los requerimientos más críticos para el proyecto
- ▶ Los stakeholders pueden tener evidencias claras del estado del proyecto.



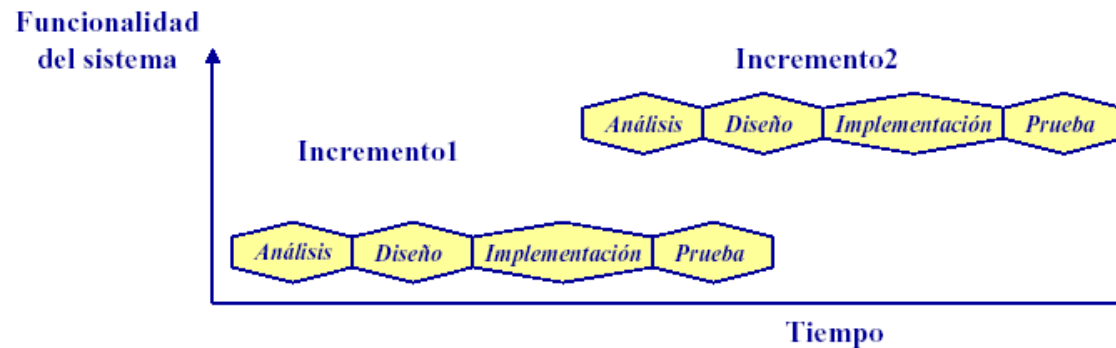
Ventajas desarrollo Iterativo-Incremental

- ▶ El cliente recibe entregas de la funcionalidad del sistema en cada incremento
- ▶ Las pruebas iterativas continuas dan una idea clara del estado del proyecto.
- ▶ Administración del riesgo
 - ▶ Bajo riesgo de falla total del proyecto



Dificultades Desarrollo Iterativo-Incremental

- ▶ Mayor planeación: General – Iteración
- ▶ Artefactos: modificados, revisados, aprobados
- ▶ Mucho retrabajo en las 1 eras. iteraciones.
- ▶ Pruebas e integración constantes

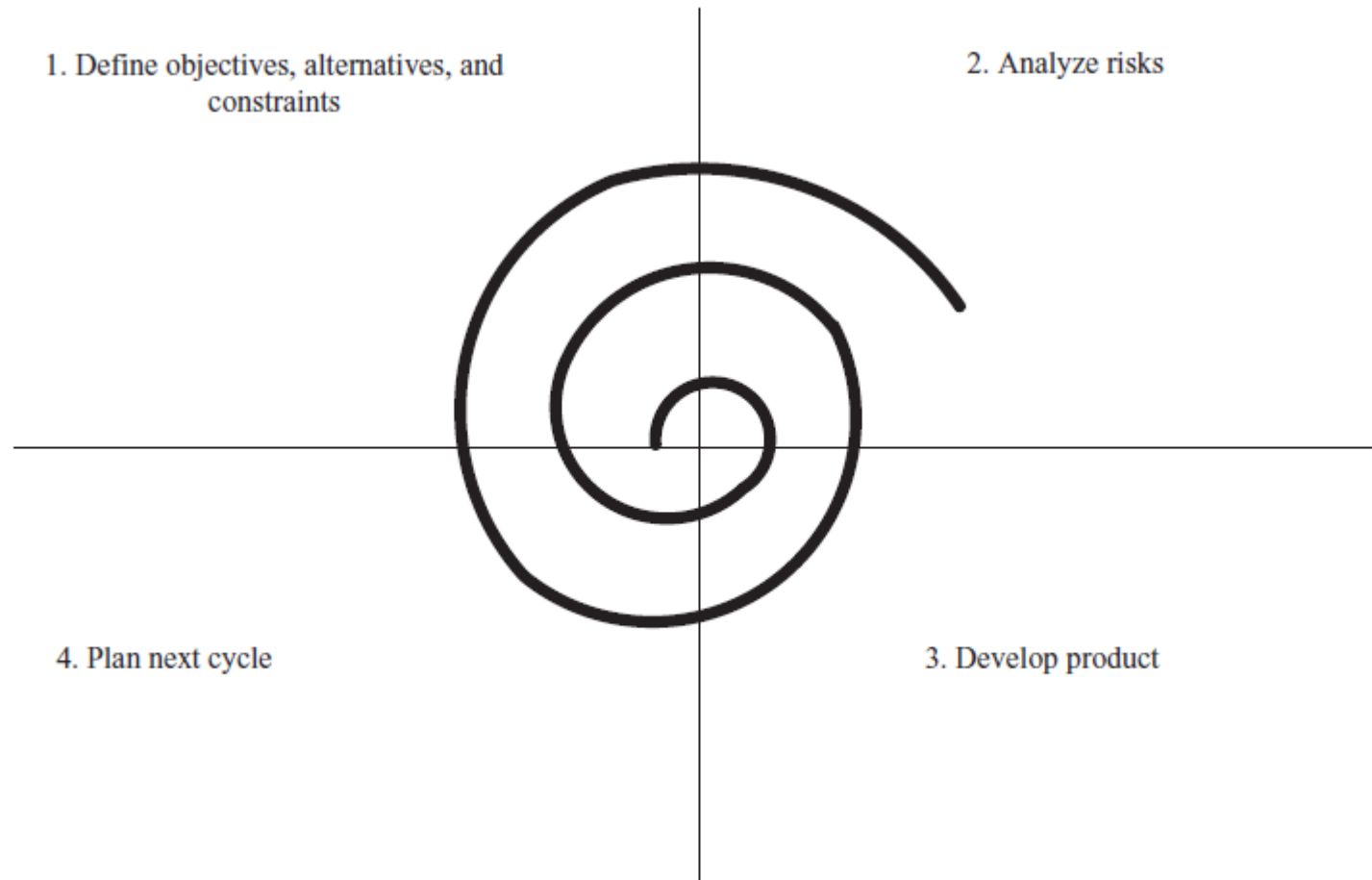




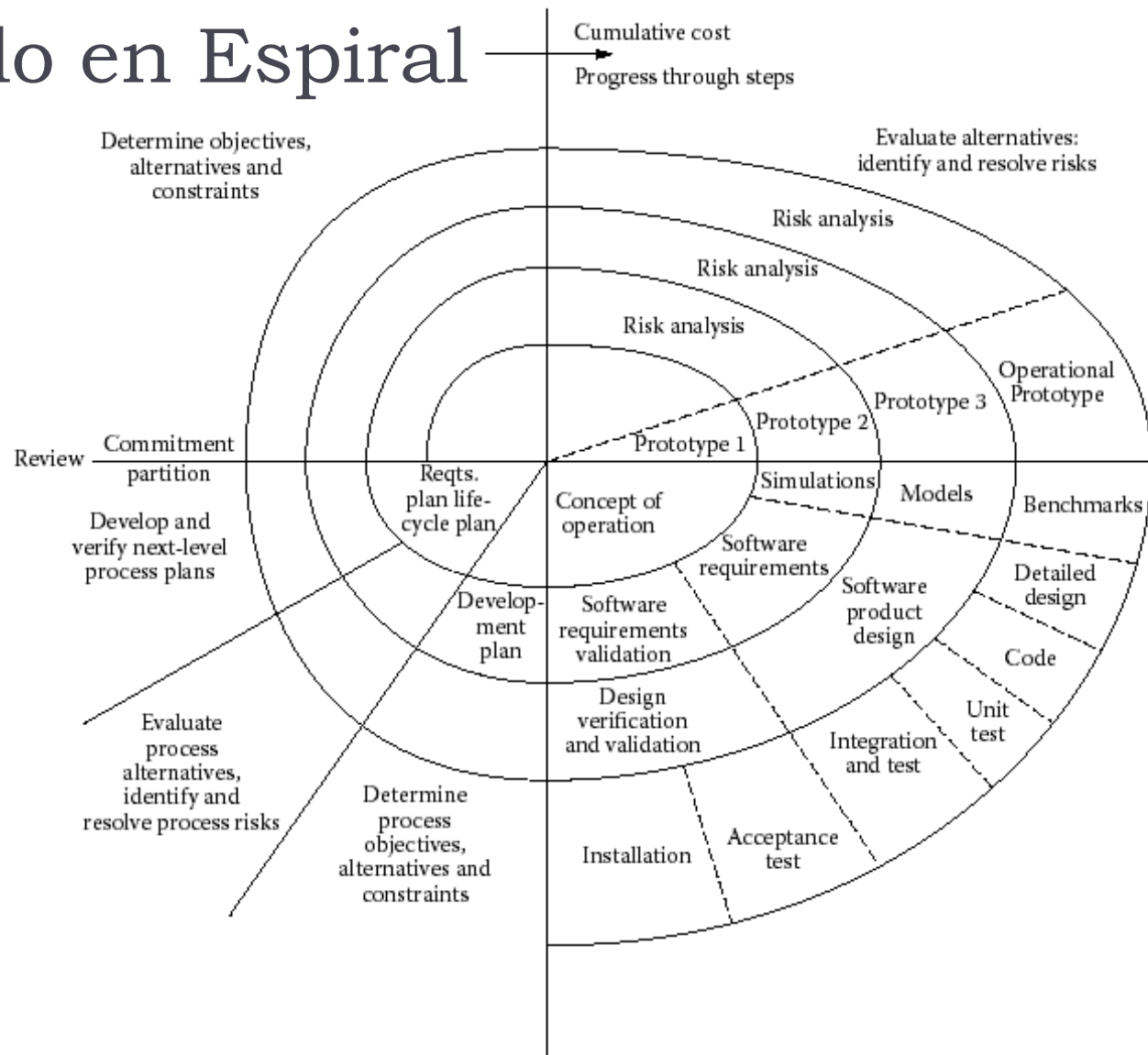
Desarrollo en Espiral

- ▶ Modelo de proceso dirigido por riesgos desarrollado por Boehm
- ▶ **Riesgo**: “estado o propiedad de un proyecto de desarrollo que si es ignorado o no resuelto incrementará la probabilidad de falla del proyecto” Deek et al. (2005)
- ▶ El proceso es representado como una espiral más que una secuencia de actividades
- ▶ Cada ciclo en la espiral representa un fase en el proceso: viabilidad, requerimientos, diseño
- ▶ Los riesgos son explícitamente estimados y resueltos a través del proceso

Desarrollo en Espiral

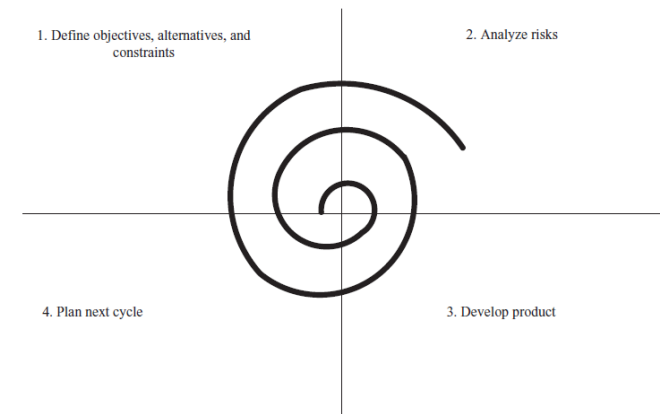


Desarrollo en Espiral



Desarrollo en Espiral: Sectores

- ▶ Configuración de objetivo
 - Objetivos específicos para la fase son identificados
 - Se identifican las restricciones
- ▶ Estimación y reducción del riesgo
 - Los riesgos son estimados y se desarrollan actividades para mitigarlos



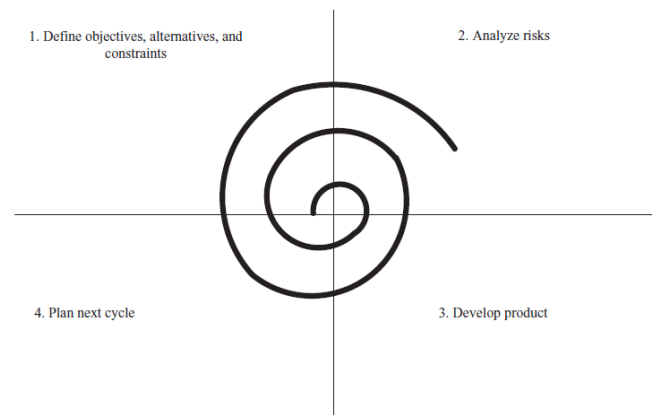
Desarrollo en Espiral: Sectores

► Desarrollo y validación

- Un modelo de desarrollo para el sistema es escogido el cual puede ser cualquiera de los modelos genéricos

► Planeación

- El proyecto es revisado y la siguiente fase es planeada





Beneficios Desarrollo en Espiral

- ▶ Eliminación de riesgos potenciales en etapas tempranas.
- ▶ Permite acomodar otros modelos en las iteraciones.
- ▶ Se tienen puntos de control en cada iteración.



Prototipado

- ▶ Empleado en muchas disciplinas del desarrollo
- ▶ Permite el desarrollo de pequeñas versiones del sistema objeto de construcción
- ▶ Un prototipo de sistemas de información debe cumplir tres características:
 - Ser un sistema temporal
 - Ser desarrollado de forma rápida
 - Proveer una expresión visual o tangible del sistema propuesto
- ▶ Se pueden emplear generadores de código o lenguajes 4GL



Prototipado

- ▶ El prototipo es descartado una vez que inicia el desarrollo del sistema
- ▶ Se puede obtener retroalimentación del usuario producto de la interacción con una aproximación del producto deseado

.



Prototipado: Clasificación

- ▶ **Prototipos exploratorios:** usados como una herramienta para elicitación o clarificación de requerimientos
 - El desarrollador entiende los problemas y necesidades del usuario
 - El usuario clarifica sus requerimientos
- ▶ **Prototipos experimentales:** usados para evaluar o probar si el sistema cumplirá las expectativas del usuario o explorar alternativas
- ▶ **Prototipos evolutivos:** explorar cambios en los requerimientos incrementalmente



Prototipado

► Ventajas

- Retroalimentación del usuario en etapas tempranas
- Línea base entre desarrolladores y usuarios para identificar oportunidades
- Motiva al usuario a involucrarse
- Previene malentendidos entre usuarios y desarrolladores

.

Referencias



- ▶ Deek F. P., McHugh J.A.M., Eljabiri Strategic O. M. (2005). **Strategic Software Engineering: An Interdisciplinary Approach**. USA: Auerbach Publications.
- ▶ Kurbel, K. E. (2008). **The Making of Information Systems: Software Engineering and Management in a Globalized World**. Germany: Springer-Verlag Berlin Heidelberg. {Cap. 4 - Developing Information Systems }
- ▶ Sommerville, I. (2005). **Ingeniería de Software**. 7ma. edición. Prentice-Hall.
- ▶ Mena, A. (2011). Modelos de Proceso de Desarrollo del Software. Curso Análisis y Diseño de Software, Universidad de Costa Rica