


2.2 MODELOS DE PROCESO

Prof. Andrés Rice Mora Fundamentos de Ingeniería de Software Departamento Ingeniería Informática 1



MODELOS DE PROCESO

¿ QUE ES UN MODELO ?

- Es una conceptualización que se representa como un esquema simplificado de una realidad.
- Contiene las características y relaciones más importantes.
- Se elabora para facilitar la comprensión y sirve como punto de referencia para imitarlo o reproducirlo.

Ejemplos

Modelo Cliente-Servidor, se apoya en terminales (clientes) conectados a un computador que los provee de recursos (servidor).

Modelo relacional, representan los datos y las relaciones entre éstos, a través de una colección de tablas.

Modelo matemático.

....

Modelos de procesos de software.

Prof. Andrés Rice Mora Fundamentos de Ingeniería de Software Departamento Ingeniería Informática 2

MODELOS DE PROCESO

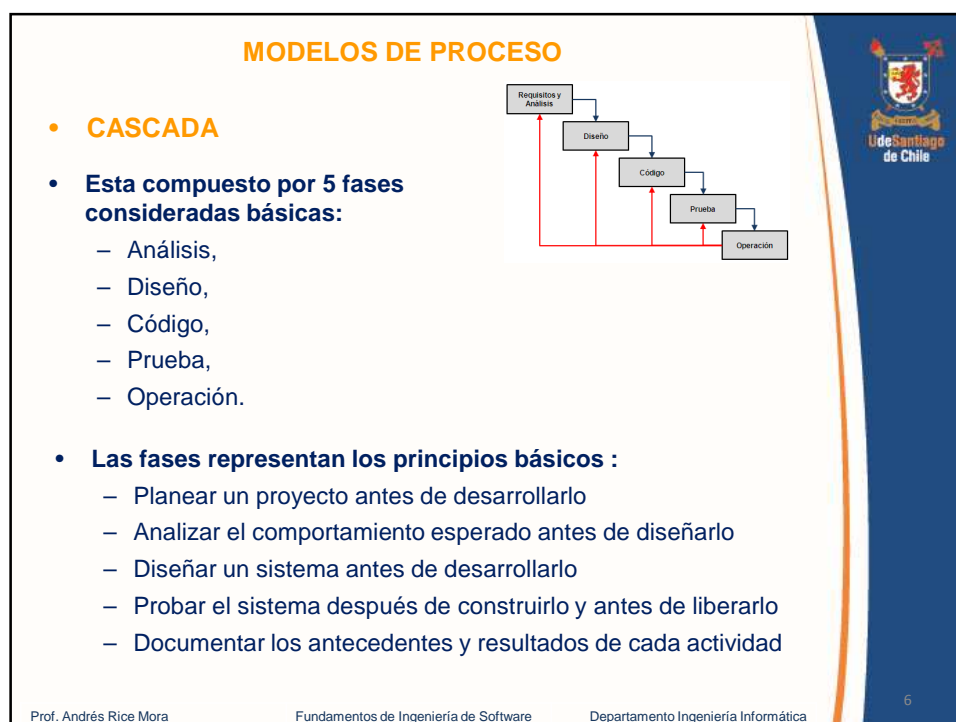
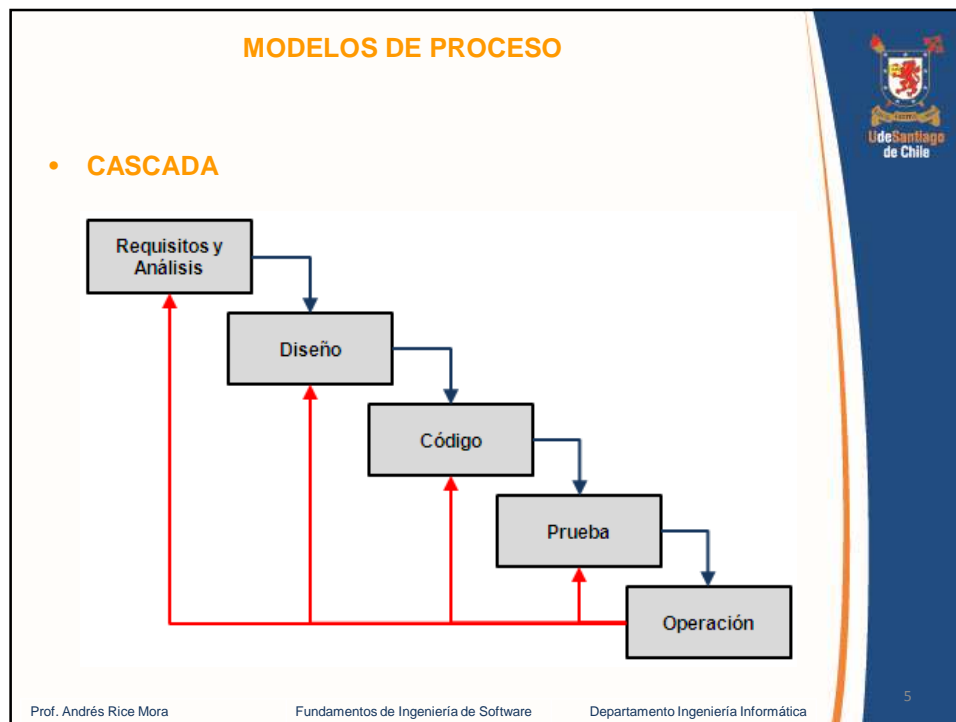
Para resolver los problemas reales de una industria, un ingeniero del software o un equipo de ingenieros debe incorporar **una estrategia de desarrollo que acompañe al proceso**. Esta estrategia se conoce como **modelo de proceso o paradigma de ingeniería del software**.

MODELOS DE PROCESO

- Los modelos están constituidos por un conjunto de fases que agrupan las actividades y marcan los principales hitos del proyecto
- Las fases se realizan según las necesidades tanto de la organización del proyecto como del tipo de producto

¿ PARA QUÉ SIRVE?

- Definir el modelo de trabajo.
- Estimar los riesgos.
- Confeccionar la Gantt de proyecto.
- Gestionar el proyecto de software.
- Elemento de comunicación con los involucrados.



MODELOS DE PROCESO

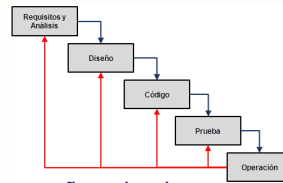
• CASCADA

• Ventajas

- Fácil de implementar y entender.
- Ampliamente usado en proyectos pequeños, donde los riesgos sean mínimos o al menos controlables.
- Fácil de gestionar.
- La documentación se genera conforme avanza su ejecución.

• Desventajas

- Permite iteraciones entre fases pero retardan la liberación del producto.
- Se espera que la totalidad de los requisitos sean declarados, en su totalidad, al inicio del proyecto.
- El cliente no obtiene avances para ir revisando el producto. El Software se libera al final del ciclo lo que puede generar frustración en el cliente si no obtiene lo esperado.



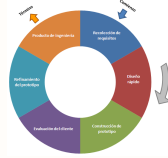
MODELOS DE PROCESO

• MODELO PROTOTIPO (ITERATIVO)



MODELOS DE PROCESO

MODELO PROTOTIPO (BROOKS, 1975)




- **Producto en tres pasos:**
 - Definir objetivos globales e identificar los requisitos con el cliente
 - Construcción de la maqueta (prototipo)
 - El cliente prueba la maqueta y depura los requisitos del software
- **Características:**
 - Sirve como base para presentar requisitos recolectados
 - Exige contar con herramientas ágiles de diseño y que permitan experimentar
 - Requiere de un equipo de desarrollo reducido
 - Puede ser cerrado o abierto (desechable o evolutivo)
 - Si el enfoque es cerrado, después de aceptado por el cliente el prototipo es desechado y se selecciona otro paradigma para su construcción definitiva
 - Si el enfoque es abierto, el producto es evolucionado hasta alcanzar su estado final deseado

Prof. Andrés Rice Mora Fundamentos de Ingeniería de Software Departamento Ingeniería Informática

MODELOS DE PROCESO

MODELO PROTOTIPO (BROOKS, 1975)

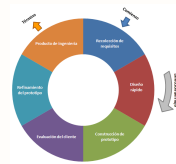


- **Ventajas:**
 - El producto obtenido puede progresar hasta un producto final
 - Al aumentar la probabilidad de éxito, reduce el riesgo de construir productos que no satisfagan las necesidades del usuario
- **Desventajas:**
 - No es apropiado para proyectos grandes porque elevaría los costos
 - Pone énfasis en la interfaz de usuario, pero no en la calidad del producto
 - Descuida la documentación y los aspectos de sus futuras mantenciones
 - El cliente al ver que es muy fácil modelar, puede caer en la tentación de expandir el proyecto original
 - El cliente puede frustrarse al saber que el prototipo ahora debe ser desarrollado

Prof. Andrés Rice Mora Fundamentos de Ingeniería de Software Departamento Ingeniería Informática

MODELOS DE PROCESO

• MODELO PROTOTIPO (BROOKS, 1975)

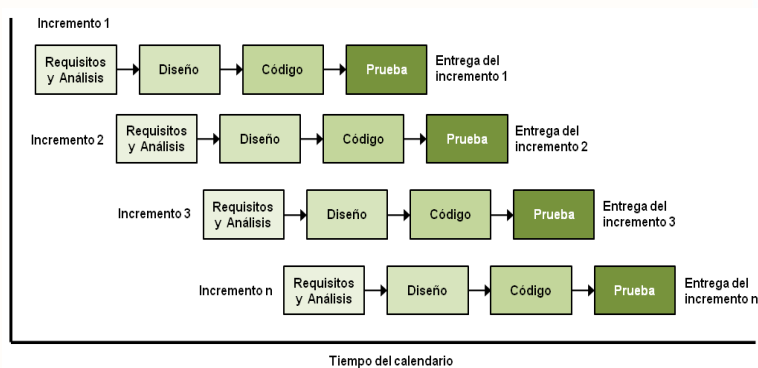


¿Desechamos o no el prototipo?

Según Fred Brooks, en la mayoría de los proyectos, el primer sistema construido apenas se puede utilizar. Puede ser demasiado lento, demasiado grande o torpe en su uso, o las tres a la vez. No hay otra alternativa que comenzar de nuevo,.....por lo tanto la pregunta no es si tirarlo o no, tendremos que hacerlo.

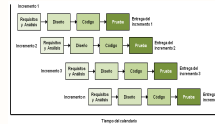
MODELOS DE PROCESO

• MODELO INCREMENTAL



MODELOS DE PROCESO

• MODELO INCREMENTAL

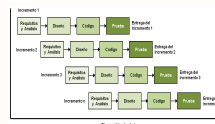


• Características:

- Combina elementos del modelo cascada más la filosofía iterativa del modelo prototipo
- Se centra en la entrega de un producto operacional con cada incremento
- Se suele ocupar en desarrollos en los que los requisitos no están claros
- Es particularmente útil cuando la dotación de personal no está disponible para una implementación completa. Los primeros incrementos se pueden implementar con menos personas

MODELOS DE PROCESO

• MODELO INCREMENTAL



• Ventajas:

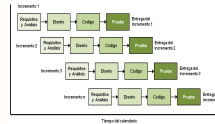
- Se puede financiar el proyecto por partes
- No requiere de una dotación de personal completa como para el proyecto completo
- Es apropiado para proyectos grandes de larga duración
- El usuario se ve obligado a una mayor participación, lo que ayuda a disminuir las desviaciones
- Existe un bajo riesgo de fallo total del proyecto. Los incrementos se van entregando de forma satisfactoria para el cliente
- Los clientes no tienen que esperar hasta que el sistema completo se entregue para sacar provecho de el

MODELOS DE PROCESO

• MODELO INCREMENTAL

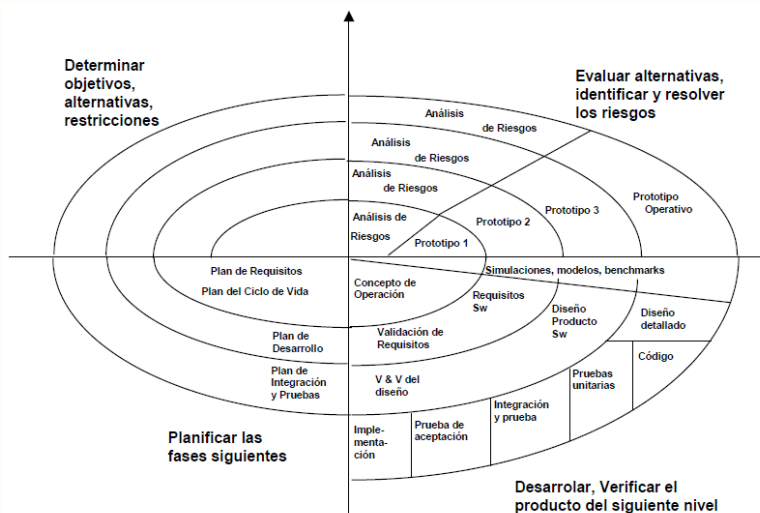
• Desventajas:

- Es difícil estimar el costo total
- Puede aumentar el costo total debido a las reiteradas pruebas
- Dificulta su aplicación en sistemas transaccionales ya que operan como un todo
- Requiere gestores experimentados
- Cada incremento debe aumentar la funcionalidad



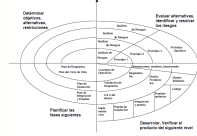
MODELOS DE PROCESO

• MODELO ESPIRAL (BOEHM, 1988)



MODELOS DE PROCESO

• MODELO ESPIRAL (BOEHM, 1988)

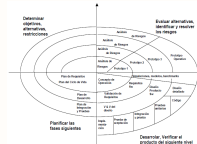


• Características:

- Representa el proceso de software como una secuencia de actividades
- Cada avance en 360° representa un ciclo y cada ciclo equivale a una fase del proceso. Así, el ciclo más interno podría referirse a la viabilidad del sistema, el siguiente ciclo a la definición de requisitos, el siguiente ciclo al diseño del sistema, etc.
- Una diferencia respecto a otros modelos de proceso de software es la consideración explícita del riesgo en el modelo
- Incorpora las cualidades del modelo cascada y de prototipo

MODELOS DE PROCESO

• MODELO ESPIRAL (BOEHM, 1988)

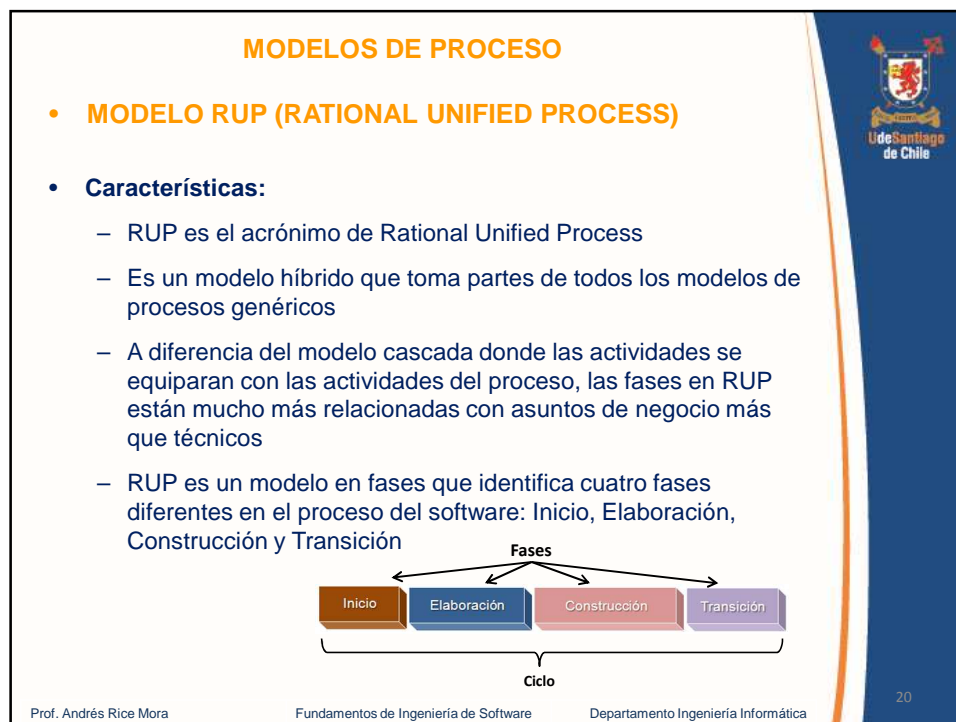
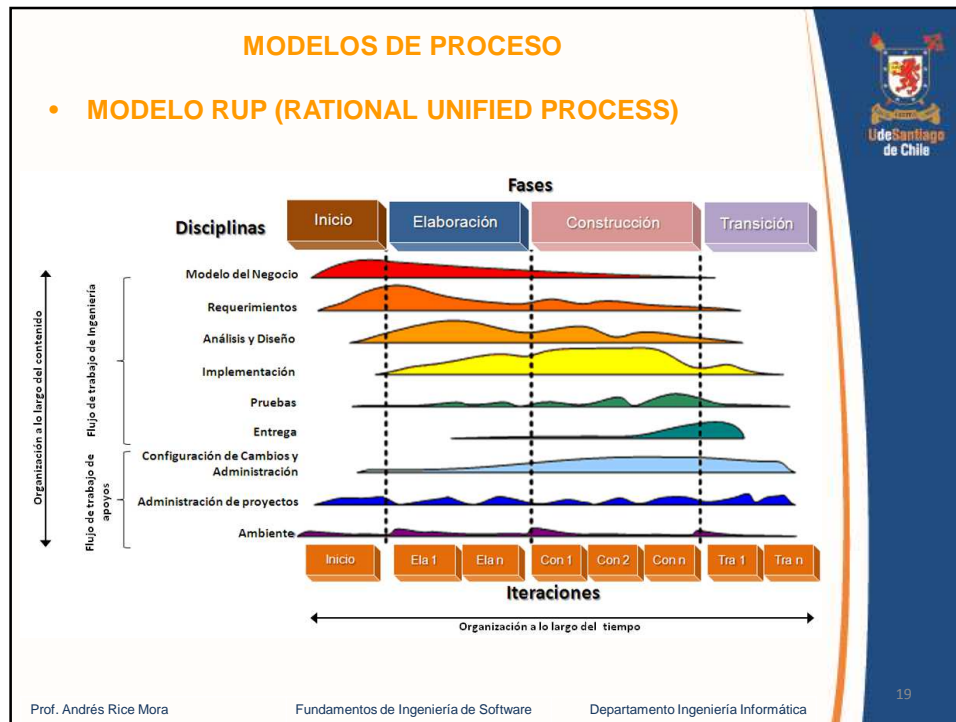


• Ventajas:

- Refleja la naturaleza iterativa de los proyectos con requisitos poco nítidos
- Disminuye los riesgos ya que estos se analizan en cada ciclo
- Cada ciclo puede ser adaptado a las necesidades inmediatas del proyecto

• Desventajas:

- Requiere gestores experimentados
- No siempre es aceptado por los clientes debido a su naturaleza adaptativa o cambiante
- Exige un buen control del proyecto de lo contrario, se puede transformar en un fracaso inminente
- Cada paso por la región de planificación produce ajustes en el plan del proyecto
- El costo se ajusta según la reacción ante la evaluación del proyecto

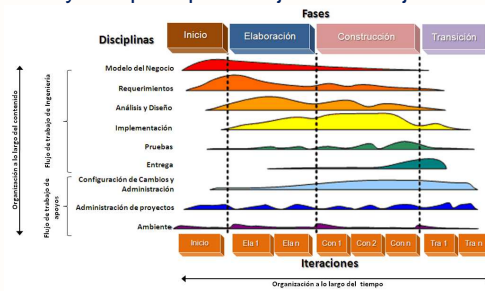


MODELOS DE PROCESO

• MODELO RUP (RATIONAL UNIFIED PROCESS)

• Características:

- La vista estática de RUP se centra en las actividades que tiene lugar durante el proceso de desarrollo. Éstas se denominan *flujos de trabajo*. Existen seis principales flujos de trabajo del proceso y tres principales flujos de trabajo de soporte.



- RUP utiliza UML para describir los flujos de trabajo asociados.
- En estricto rigor, todos los flujos de trabajo de RUP pueden estar activos en todas las etapas del proceso

Prof. Andrés Rice Mora

Fundamentos de Ingeniería de Software

Departamento Ingeniería Informática

21

MODELOS DE PROCESO

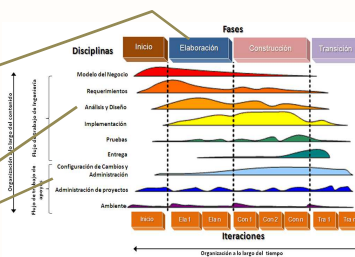
• MODELO RUP (RATIONAL UNIFIED PROCESS)

RUP se describe normalmente desde tres perspectivas:

Perspectiva dinámica que muestra las fases del modelo sobre el tiempo

Perspectiva estática que muestra las actividades del proceso que se representan

Perspectiva práctica que sugiere buenas prácticas a utilizar durante el proceso



Prof. Andrés Rice Mora

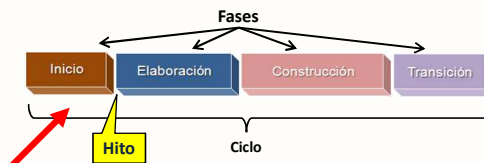
Fundamentos de Ingeniería de Software

Departamento Ingeniería Informática

22

MODELOS DE PROCESO

• MODELO RUP (RATIONAL UNIFIED PROCESS)

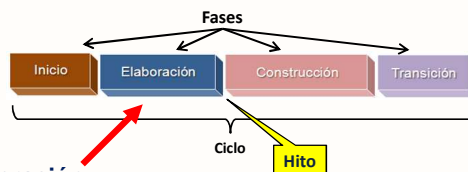


• Fases inicio

- El objetivo es establecer un caso de negocio para el sistema. Se debe identificar todas las actividades externas (Personas y sistemas) que interactúan con el sistema y definir estas interacciones. Esto se utiliza para evaluar el aporte que el sistema hace al negocio. Si el aporte es de poca importancia, se puede cancelar el proyecto después de esta fase

MODELOS DE PROCESO

• MODELO RUP (RATIONAL UNIFIED PROCESS)

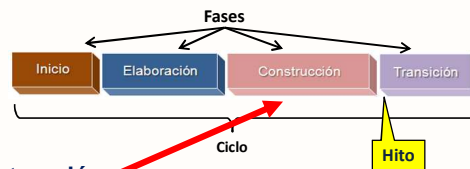


• Fase Elaboración

- El objetivo es desarrollar una comprensión del dominio del problema, establecer un marco de trabajo arquitectónico para el sistema, desarrollar el plan del proyecto e identificar los riesgos clave del proyecto. Al terminar esta fase se debe tener un modelo de los requerimientos del sistema (se especifican los casos de uso UML), una descripción arquitectónica y un plan de desarrollo del software.

MODELOS DE PROCESO

• MODELO RUP (RATIONAL UNIFIED PROCESS)

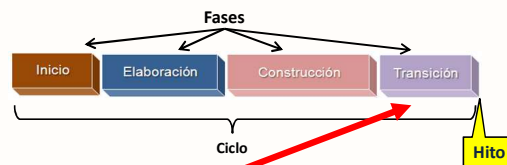


• Fase Construcción

- El objetivo es el diseño del sistema, la programación y las pruebas. Durante esta fase se desarrollan e integran las partes del sistema. Al terminar esta fase, se debe tener un sistema de software operativo y la documentación correspondiente lista para entregarla a los usuarios.

MODELOS DE PROCESO

• MODELO RUP (RATIONAL UNIFIED PROCESS)



• Fase Transición

- Esta última fase se ocupa de mover el sistema de la comunidad de desarrollo a la comunidad del usuario y hacerlo trabajar en un entorno real. Esto se deja de lado en la mayor parte de los modelos de proceso del software pero es, en realidad, una actividad de alto costo y a veces problemática. Al terminar esta fase se debe tener un sistema software documentado que funciona correctamente en su entorno operativo.

MODELOS DE PROCESO

• MODELO RUP (RATIONAL UNIFIED PROCESS)

• Flujo de trabajo estático :

Modelado del negocio	Los modelos del negocio se modelan utilizando casos de uso de negocio.
Requerimientos	Se definen los actores que interactúan con el sistema y se desarrollan casos de uso para modelar los requerimientos del sistema.
Análisis y Diseño	Se crea y documenta un modelo del diseño utilizando modelos arquitectónicos, modelos de componentes, modelos de objetos y modelos de secuencias.
Implementación	Se implementan y estructuran en subsistemas los componentes del sistema. La generación automática de código de los modelos del diseño ayuda a acelerar este proceso.
Pruebas	Las pruebas son un proceso iterativo que se llevan a cabo conjuntamente con la implementación. A la finalización de la implementación tienen lugar las pruebas del sistema.
Entrega	Se crea un <u>release</u> del producto, se distribuye a los usuarios y se instala en su lugar de trabajo.
Configuración de cambios y Administración	Este flujo de trabajo de soporte gestiona los cambios del sistema.
Administración del proyecto	Este flujo de trabajo de soporte gestiona el desarrollo del sistema.
Entorno	Este flujo de trabajo se refiere a hacer herramientas software apropiadas y disponibles para los equipos de desarrollo de software.

Prof. Andrés Rice Mora

Fundamentos de Ingeniería de Software

Departamento Ingeniería Informática

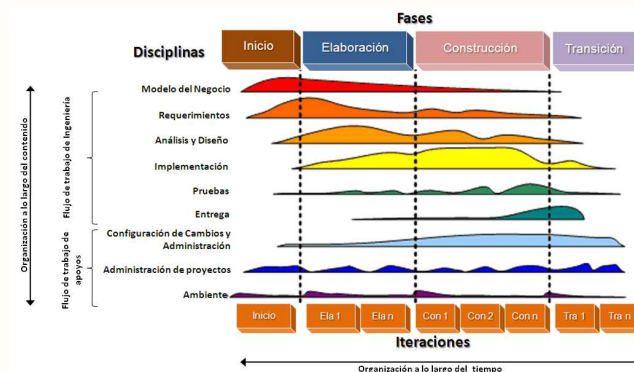
27

MODELOS DE PROCESO

• MODELO RUP (RATIONAL UNIFIED PROCESS)

• Desventajas:

- RUP no es un proceso apropiado para todo tipo de desarrollo. Para proyectos pequeños, el costo de gestionarlo con RUP, puede superar el costo del propio software.
- Requiere gestores experimentados.



Prof. Andrés Rice Mora

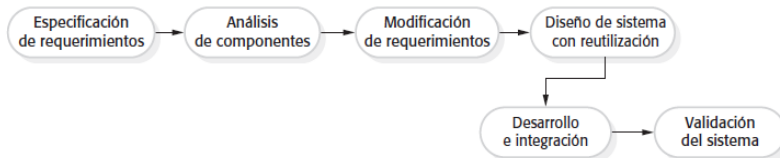
Fundamentos de Ingeniería de Software

Departamento Ingeniería Informática

28

MODELOS DE PROCESO

• INGENIERÍA DE SOFTWARE ORIENTADA A LA REUTILIZACIÓN

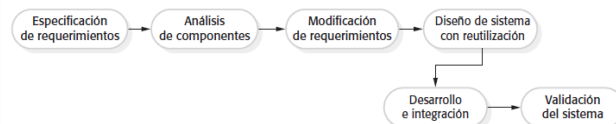


• Características:

- Los enfoques orientados a la reutilización se apoyan en una gran base de componentes de software reutilizable y en la integración de marcos para la composición de dichos componentes.

MODELOS DE PROCESO

• INGENIERÍA DE SOFTWARE ORIENTADA A LA REUTILIZACIÓN

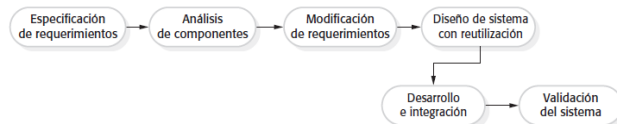


Análisis de componentes Dada la especificación de requerimientos, se realiza una búsqueda de componentes para implementar dicha especificación. Por lo general, no hay coincidencia exacta y los componentes que se usan proporcionan sólo parte de la funcionalidad requerida.

Modificación de requerimientos Durante esta etapa se analizan los requerimientos usando información de los componentes descubiertos. Luego se modifican para reflejar los componentes disponibles. Donde las modificaciones son imposibles, puede regresarse a la actividad de análisis de componentes para buscar soluciones alternativas.

MODELOS DE PROCESO

• INGENIERÍA DE SOFTWARE ORIENTADA A LA REUTILIZACIÓN



Diseño de sistema con reutilización Durante esta fase se diseña el marco conceptual del sistema o se reutiliza un marco conceptual existente. Los creadores toman en cuenta los componentes que se reutilizan y organizan el marco de referencia para atenderlo. Es posible que deba diseñarse algo de software nuevo, si no están disponibles los componentes reutilizables.

Desarrollo e integración Se diseña el software que no puede procurarse de manera externa, y se integran los componentes y los sistemas para crear el nuevo sistema. La integración del sistema, en este modelo, puede ser parte del proceso de desarrollo, en vez de una actividad independiente.

MODELOS DE PROCESO

• INGENIERÍA DE SOFTWARE ORIENTADA A LA REUTILIZACIÓN

• Características:

Existen tres tipos de componentes de software que pueden usarse en un proceso orientado a la reutilización:

1. Servicios Web que se desarrollan en concordancia para atender servicios estándares y que están disponibles para la invocación remota
2. Colecciones de objetos que se desarrollan como un paquete para su integración con un marco de componentes como .NET o J2EE
3. Sistemas de software independientes que se configuran para usar en un entorno particular.

MODELOS DE PROCESO

• INGENIERÍA DE SOFTWARE ORIENTADA A LA REUTILIZACIÓN

• Ventajas

- Reduce la cantidad de software a desarrollar
- Disminuye los costos de producción de software
- Disminuye los riesgos inherentes al proceso productivo de software
- Conduce a entregas más rápidas del software

• Desventajas

- Puede suceder que los compromisos de requerimientos no estén completamente cubiertos en el/los componente(s) reutilizable(s)
- Se pierde algo de control sobre la evolución del sistema, conforme las nuevas versiones de los componentes reutilizables no estén bajo el control de la organización que los usa



MODELOS DE PROCESO

El modelo de desarrollo que usted seleccione puede resultar ser un problema adicional si no se elige adecuadamente. El modelo puede marcar la diferencia entre el éxito y el fracaso de un proyecto de software.



MODELOS DE PROCESO

Para resolver los problemas reales de una industria, un ingeniero de software debe incorporar una estrategia de desarrollo que acompañe al proceso, métodos , herramientas y fases genéricas. Esta estrategia se conoce como **modelo de proceso** de la **ingeniería de software**.

La selección de un modelo se hace considerando los siguientes aspectos:

- Alcance, tamaño y complejidad del proyecto.
- Métodos y herramientas a utilizar.
- Plazos intermedios y total de término
- Recursos disponibles.
- Calidad requerida.
- Riesgos involucrados.

