



RUP(Rational Unified Process)

Sierra Lynn Brown

06/03/19

Rational Unified Process:

- Es un proceso de ingeniería de software. Proporciona un enfoque disciplinado para asignar tareas y responsabilidades dentro de una organización de desarrollo. Su objetivo es garantizar la producción de alta calidad de software que satisface las necesidades de sus usuarios finales, dentro de un calendario y presupuesto predecibles.



Implementación efectiva de las 6 mejores prácticas

1. Desarrollar software de forma iterativa.
2. Gestionar los requisitos.
3. Utilizar arquitecturas basadas en componentes.
4. Modelar visualmente el software.
5. Verificar la calidad del software.
6. Control de cambios al software.



1. Desarrollar software de forma iterativa:

Se requiere un enfoque iterativo que permita una comprensión cada vez mayor del problema a través de sucesivas refinamientos y para aumentar de manera incremental una solución efectiva en múltiples iteraciones.

2. Gestionar los requisitos:

cómo obtener, organizar y documentar lo requerido funcionalidad y restricciones; rastrear y documentar las compensaciones y decisiones, capturar y comunicar fácilmente requisitos comerciales.

3. Utilizar arquitecturas basadas en componentes:

Describe cómo diseñar una arquitectura resistente que sea flexible, se adapte al cambio, sea intuitivamente comprensible y promueva una reutilización más efectiva del software.

4. Modelar visualmente el software:

Esto le permite ocultar los detalles y escribir el código usando "gráfico bloques de construcción".

Las abstracciones visuales lo ayudan a comunicar diferentes aspectos de su software, ver cómo los elementos del sistema encajan entre sí, asegura de que los bloques de construcción sean consistentes con su código mantener la consistencia entre un diseño y su implementación y promover la comunicación.

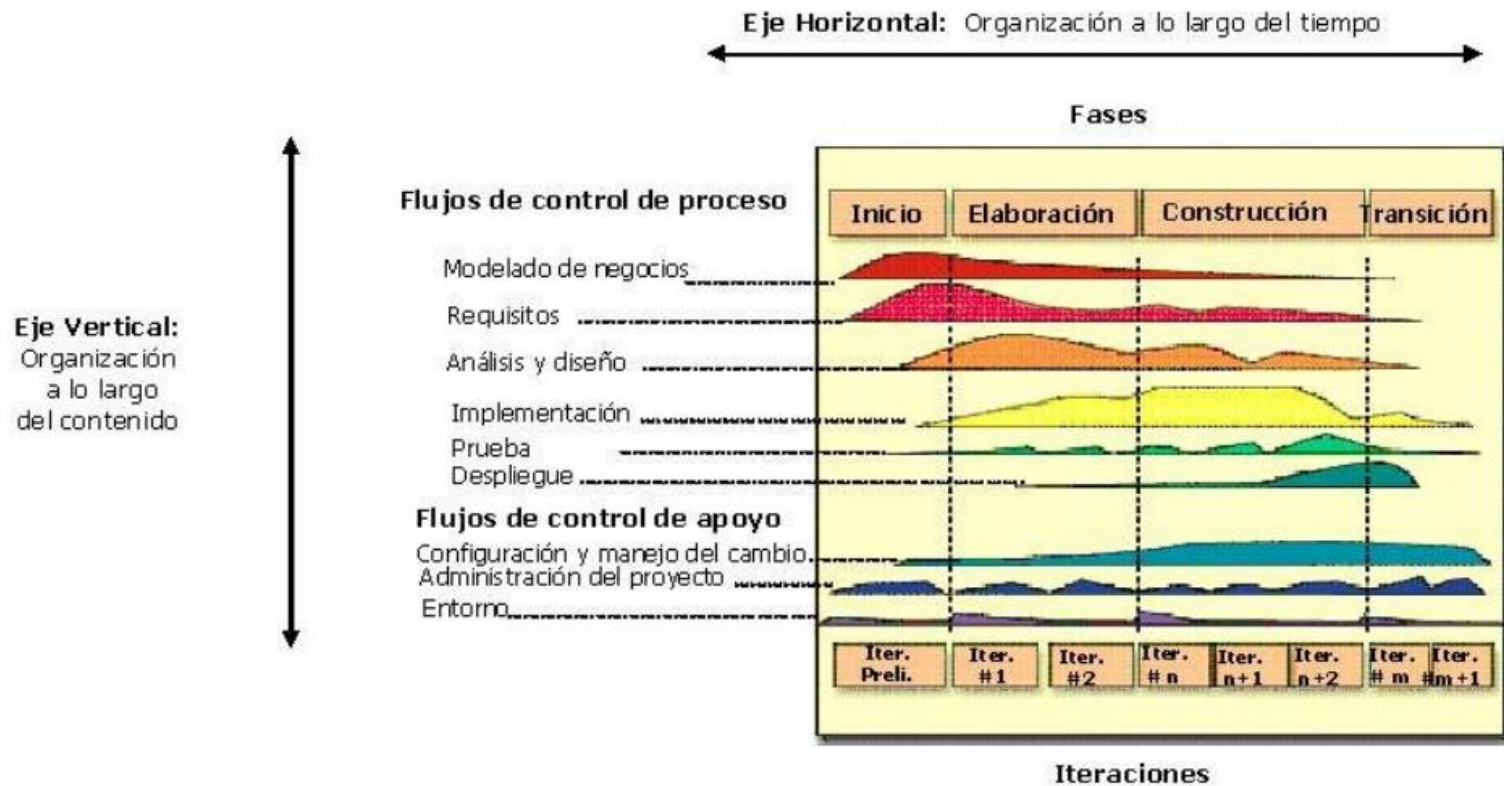
5. Verificar la calidad del software:

La calidad debe revisarse con respecto a los requisitos basados en confiabilidad, funcionalidad, rendimiento de la aplicación y rendimiento del sistema.

6. Control de cambios al software:

El proceso describe cómo controlar, seguir y monitorear los cambios para permitir un desarrollo iterativo exitoso. También le guía sobre cómo establecer espacios de trabajo seguros para cada desarrollador al proporcionar aislamiento de los cambios realizados en otros espacios de trabajo y al controlar los cambios de todos los artefactos de software (por ejemplo, modelos, códigos, documentos, etc.).

Estructura del proceso

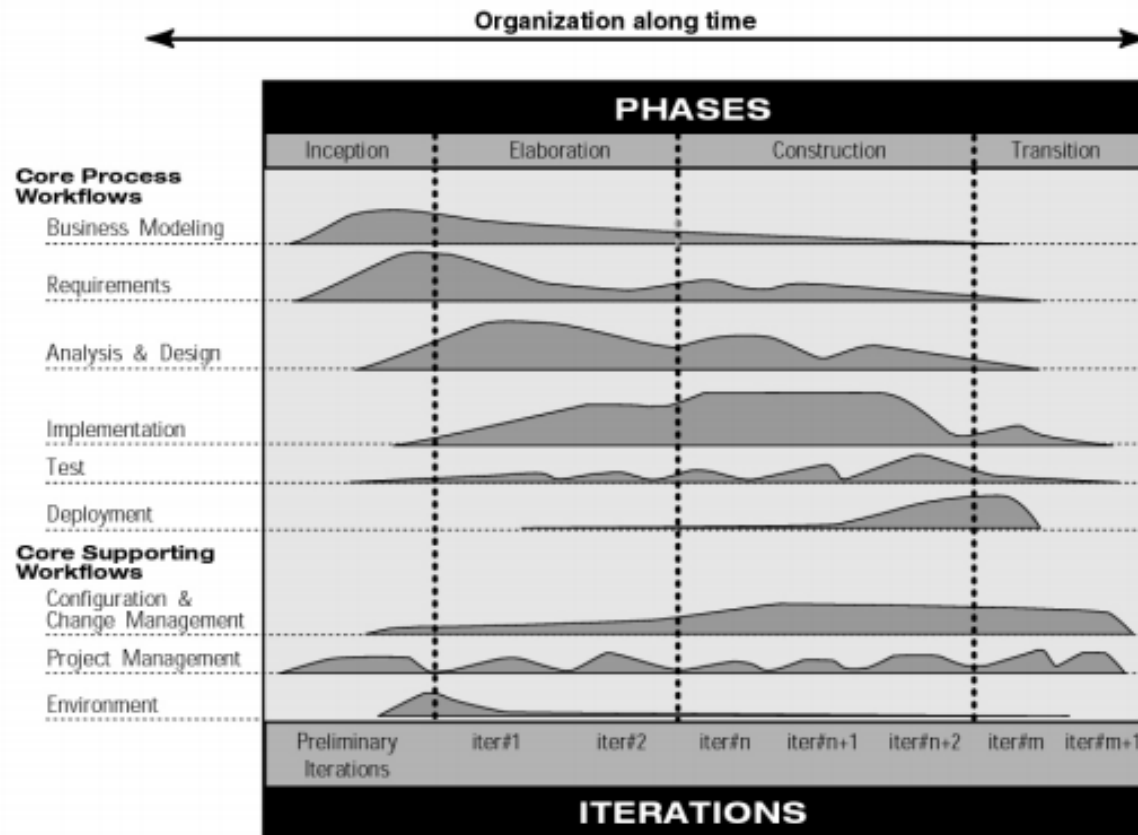


- 
- El proceso se puede describir en dos dimensiones o en dos ejes:

1. El eje horizontal representa el tiempo y muestra el aspecto dinámico del proceso a medida que se realiza, y es expresado en términos de ciclos, fases, iteraciones .

2. El eje vertical representa el aspecto estático del proceso: cómo se describe en términos de actividades, Objetos, trabajadores y flujos de trabajo.

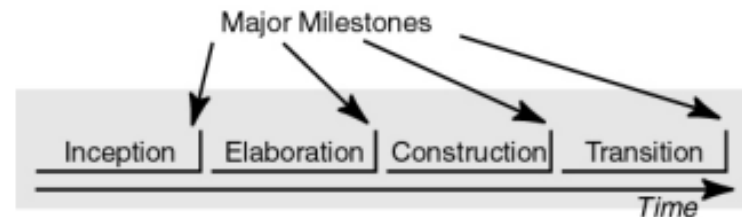
Organization
along content



Fases e iteraciones

El ciclo de vida del software se divide en ciclos, cada ciclo funciona en una nueva generación del producto. El Proceso Unificado de Rational divide un ciclo de desarrollo en cuatro fases consecutivas:

- Fase de inicio
- Fase de elaboración
- Fase de construcción
- Fase de transición



Cada fase concluye con un hito bien definido: un punto en el tiempo en el que se deben tomar ciertas decisiones críticas

Fase Inicio

Inception

Elaboration

Construction

Transition

Los objetivos de esta fase son:

- Establecer el ámbito del proyecto y sus límites.
- Encontrar los Casos de Uso críticos del sistema, los escenarios básicos que definen la funcionalidad.
- Mostrar al menos una arquitectura candidata para los escenarios principales.
- Estimar el coste en recursos y tiempo de todo el proyecto.
- Estimar los riesgos, las fuentes de incertidumbre.



Los resultados de la fase de inicio deben ser:

- Un documento de visión: Una visión general de los requerimientos del proyecto, características clave y restricciones principales.
- Modelo inicial de Casos de Uso (10-20% completado).
- Un glosario inicial: Terminología clave del dominio.
- El caso de negocio.
- Lista de riesgos y plan de contingencia.
- Plan del proyecto, mostrando fases e iteraciones.
- Modelo de negocio, si es necesario
- Prototipos exploratorios para probar conceptos o la arquitectura candidata.



El resultado de la fase de inicio es:

- Un documento de visión: una visión general de los requisitos del proyecto central, características clave y limitaciones principales.
- Un modelo de caso de uso inicial (10% -20%) completo.
- Un glosario inicial del proyecto (opcionalmente puede expresarse parcialmente como un modelo de dominio).
- Un caso de negocios inicial, que incluye el contexto de negocios, criterios de éxito (proyección de ingresos, mercado reconocimiento, y así sucesivamente), y previsión financiera.
- Una evaluación inicial de riesgos.
- Un plan de proyecto, mostrando fases e iteraciones.
- Un modelo de negocio, si es necesario.
- Uno o varios prototipos.

Hito: Objetivos del ciclo de vida

Milestone : Lifecycle Objectives



- Al final de la fase inicial se encuentra el primer hito importante del proyecto: el hito Objetivos del ciclo de vida.
- Los criterios de evaluación para la fase de inicio son:
- Concordancia de los interesados en la definición del alcance y las estimaciones de costo / programa.
- Comprensión de los requisitos como lo demuestra la fidelidad de los casos de uso principales.
- Credibilidad de las estimaciones de costo / cronograma, prioridades, riesgos y proceso de desarrollo.
- Profundidad y amplitud de cualquier prototipo arquitectónico.

Fase de elaboración

Inception

Elaboration

Construction

Transition

Los objetivos de esta fase son :

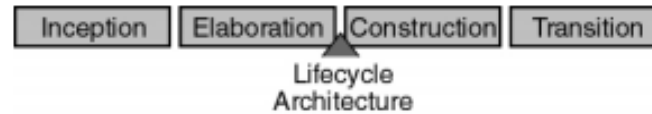
- Definir, validar y cimentar la arquitectura.
- Completar la visión.
- Crear un plan fiable para la fase de construcción. Este plan puede evolucionar en sucesivas iteraciones. Debe incluir los costes si procede.
- Demostrar que la arquitectura propuesta soportará la visión con un coste razonable y en un tiempo razonable.



Al terminar deben obtenerse los siguientes resultados :

- Un modelo de Casos de Uso completa al menos hasta el 80%: todos los casos y actores identificados, la mayoría de los casos desarrollados.
- Requisitos adicionales que capturan los requisitos no funcionales y cualquier requisito no asociado con un Caso de Uso específico.
- Descripción de la arquitectura software.
- Un prototipo ejecutable de la arquitectura.
- Lista de riesgos y caso de negocio revisados.
- Plan de desarrollo para el proyecto.
- Un caso de desarrollo actualizado que especifica el proceso a seguir.
- Un manual de usuario preliminar (opcional).

Hito: Arquitectura del ciclo de vida



Los principales criterios de evaluación para la fase de elaboración incluyen las respuestas a estas preguntas:

- ¿Es estable la visión del producto?
- ¿Es estable la arquitectura?
- ¿La demostración ejecutable muestra que los principales elementos de riesgo se han abordado de manera creíble?
- ¿resuelto?
- ¿Es el plan para la fase de construcción suficientemente detallado y exacto? ¿Está respaldado con un crediente?
- base de estimaciones?
- ¿Todas las partes interesadas están de acuerdo en que la visión actual se puede lograr si el plan actual se ejecuta para desarrollarlo?
- ¿El sistema completo, en el contexto de la arquitectura actual?
- ¿Es aceptable el gasto real de recursos versus el gasto planificado?

Fase de construcción



Los objetivos concretos incluyen:

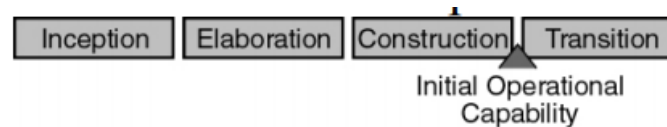
- Minimizar los costes de desarrollo mediante la optimización de recursos y evitando el tener que rehacer un trabajo o incluso desecharlo.
- Conseguir una calidad adecuada tan rápido como sea práctico.
- Conseguir versiones funcionales (alfa, beta, y otras versiones de prueba) tan rápido como sea práctico.



Los resultados de la fase de construcción deben ser:

- Modelos Completos (Casos de Uso, Análisis, Diseño, Despliegue e Implementación)
- Arquitectura íntegra (mantenida y mínimamente actualizada)
- Riesgos Presentados Mitigados
- Plan del Proyecto para la fase de Transición.
- Manual Inicial de Usuario (con suficiente detalle)
- Prototipo Operacional – beta
- Caso del Negocio Actualizado

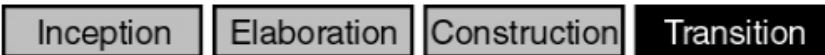
Hito: Capacidad operacional inicial



Los criterios de evaluación para la fase de construcción implican responder a estas preguntas:

- ¿Es esta versión del producto estable y lo suficientemente madura como para ser implementada en la comunidad de usuarios?
- ¿Están todas las partes interesadas listas para la transición a la comunidad de usuarios?
- ¿Los gastos de recursos reales en comparación con los gastos planificados siguen siendo aceptables?

Fase de transición



Los principales objetivos de esta fase son:

- Conseguir que el usuario se valga por si mismo.
- Un producto final que cumpla los requisitos esperados, que funcione y satisfaga suficientemente al usuario.



Los resultados de la fase de transición son :

- Prototipo Operacional
- Documentos Legales
- Caso del Negocio Completo
- Línea de Base del Producto completa y corregida que incluye todos los modelos del sistema
- Descripción de la Arquitectura completa y corregida
- Las iteraciones de esta fase irán dirigidas normalmente a conseguir una nueva versión.



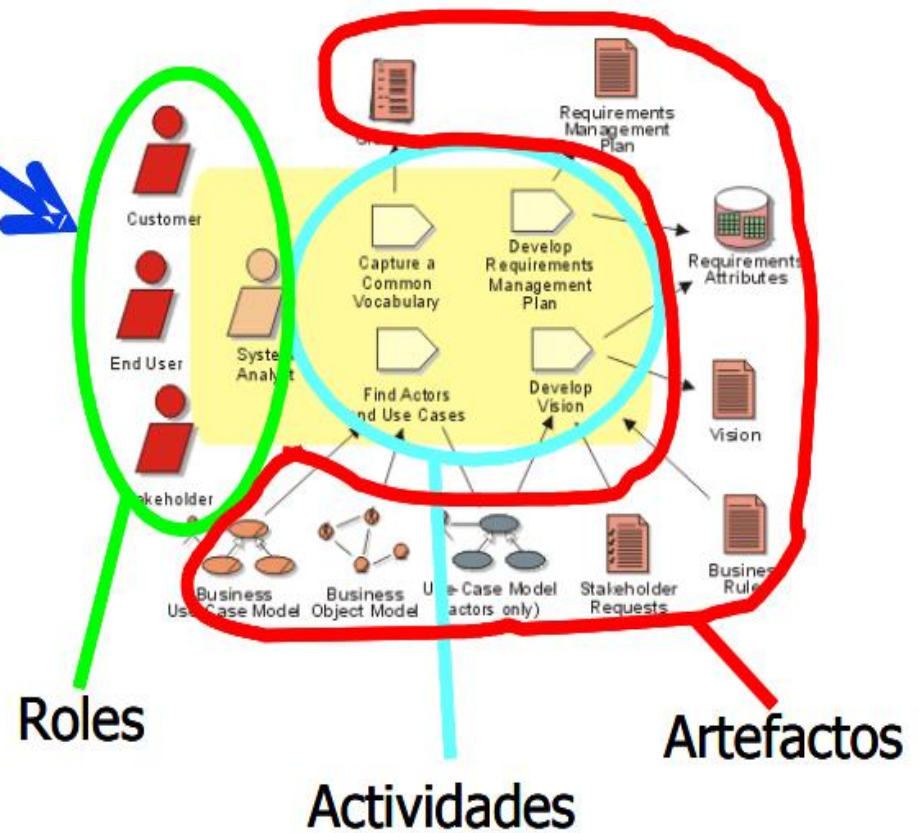
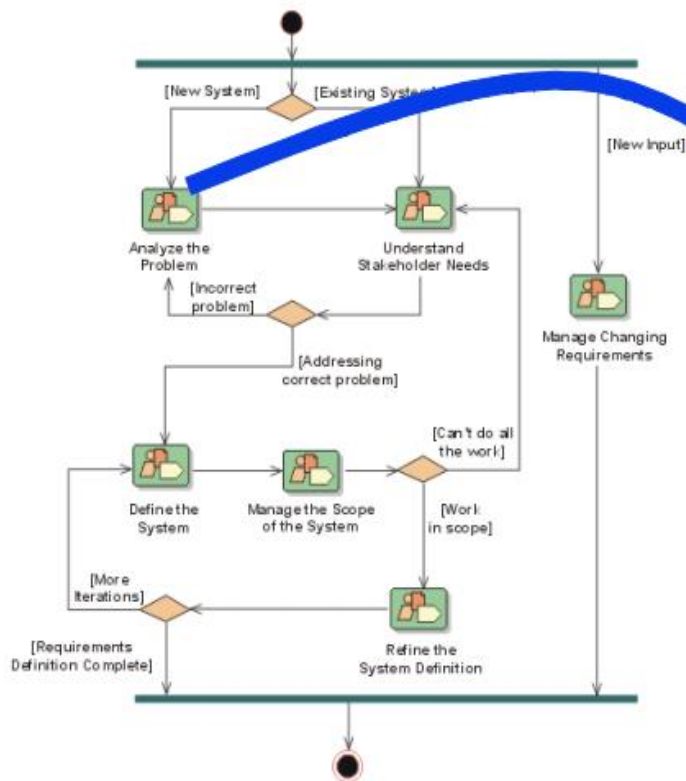
Estructura Estática del proceso. Roles, actividades, artefactos y flujos de trabajo

Un proceso de desarrollo de software define quién hace qué, cómo y cuándo.

RUP define cuatro elementos los roles, que responden a la pregunta ¿Quién?, las actividades que responden a la pregunta ¿Cómo?, los productos, que responden a la pregunta ¿Qué? y los flujos de trabajo de las disciplinas que responde a la pregunta ¿Cuándo?.



Relación entre roles, actividades, artefactos



Detalle de un workflow mediante roles, actividades y artefactos

Referencias

- <http://ima.udg.edu/~sellares/EINF-ES2/Present1011/MetodoPesadesRUP.pdf>
- https://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251_bestpractices_TP026B.pdf