

### Contenido:

- Introducción
- Flujos de trabajo fundamentales
- Desarrollo iterativo e incremental
- Fuentes Bibliográficas

### **Algunos modelos de procesos software:**

- SW-CMMI
  - uno semejante es SPICE (ISO/IEC 15504)
- PSP, TSP
  - Personal Software Process, Team Software Process
- Procesos de ciclo de vida
  - ISO/IEC 12207, contempla tres tipos de procesos: principales, de soporte y de la organización
- Procesos comerciales
  - RUP: Rational Unified Process

## Proceso Unificado.



Dependencias entre Procesos, Actividades y Tareas en el ISO/IEC 12207

## Proceso Unificado.

### Introducción.

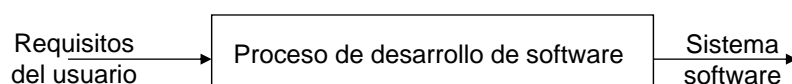
El Proceso Unificado (RUP) es un proceso\* de desarrollo de software. Según sus desarrolladores es más que un proceso, es un marco de trabajo genérico que puede especializarse para gran cantidad de sistemas de software, diferentes áreas de aplicación, diferentes tipos de organizaciones, etc.

Está basado en *componentes*, lo que indica que el sistema a construir estará formado por componentes software interconectados a través de *interfaces*.

Utiliza UML como lenguaje de modelado.

Los tres elementos claves que resumen el proceso unificado son:

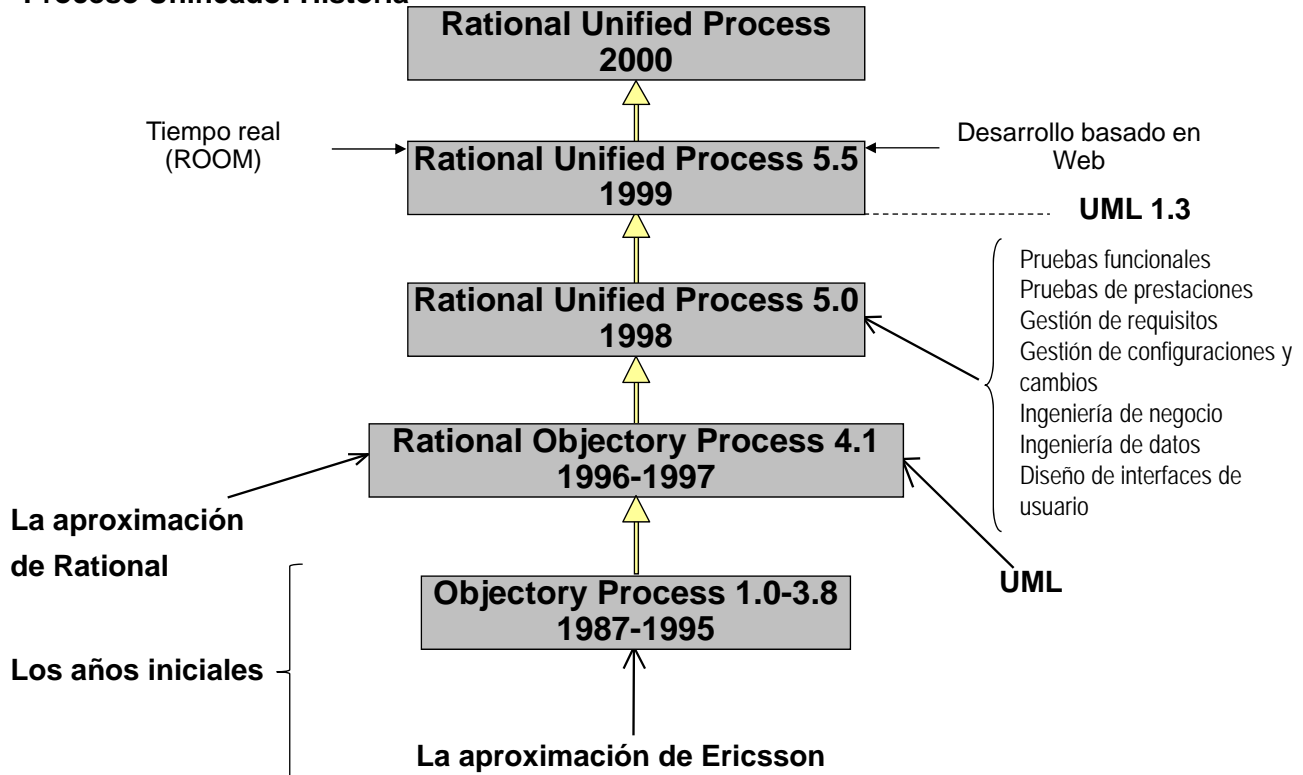
- ***dirigido por los casos de uso***
- ***centrado en la arquitectura***
- ***iterativo e incremental***



\*Un proceso define quién debe hacer qué, cuándo hacerlo y cómo alcanzar un cierto objetivo.

# Proceso Unificado.

## Proceso Unificado: Historia



# Proceso Unificado.

## Objetivos de RUP

- Proporcionar un marco de trabajo para gestionar proyectos software.
- Proporcionar guías prácticas para planificar, gestión de personal, ejecutar, realizar y seguir proyectos.
- Proporcionar un marco para gestionar riesgos

## No cubre cosas como:

- Gestión de personal: reclutamiento, formación, etc.
- Gestión de presupuestos: definir, reservar.
- Gestión de contratos con proveedores y usuarios.

## Enfoca fundamentalmente en los aspectos específicos de un proceso de desarrollo iterativo:

- Gestión de riesgos.
- Planificar un proyecto iterativo a través del ciclo de vida y planificar una iteración particular.
- Monitorizar el progreso de un proyecto iterativo y obtener métricas.

## Proceso Unificado.

---

Tanto RUP como UPEDU son aproximaciones al proceso de desarrollo de software. Un proceso debe estar adaptado:

- Al entorno
- A las personas
- A los objetivos de las organizaciones que lo utilizan

Ningún proceso puede ser adecuado a todas las organizaciones, por tanto un proceso debe ser adaptable y configurable.

Los factores que se deben tener en cuenta para la adaptación del proceso son:

- Dominio de la aplicación. Por ejemplo en las aplicaciones de tiempo real la parte de pruebas debe ser más elaborada.
- Las personas y los factores organizativos.
- Factores relacionados con el ciclo de vida del producto.
- Factores de proceso, por ejemplo, cumplimiento de estándares como ISO 15504 o CMMI

## Proceso Unificado.

---

- Estructura estática

personas	actividades	artefactos	flujos de trabajo
<i>quién</i>	<i>cómo</i>	<i>qué</i>	<i>cuándo</i>

- Estructura dinámica

- ciclos
  - fases
  - iteraciones
  - hitos
- } relaciones

## Proceso Unificado.

---

» descripción del proceso en base a personas, actividades, artefactos y flujos de trabajo. Ejemplos:

- **Personas**
  - Analista de sistemas, diseñador, diseñador de pruebas
- **Actividades**
  - Planificar, describir un caso de uso, revisar un diseño
- **Artefactos**
  - Modelos, documentos, código
- **Flujos de trabajo**
  - Análisis de requisitos, diseño, implementación, pruebas

## Proceso Unificado.

---

### Actividad

- Unidad de trabajo que puede ejecutar un individuo (o varios) en un rol específico
- Tiene un propósito claro y se expresa en términos de actualizar artefactos
- La granularidad de la actividad es generalmente de horas o pocos días
- Ejemplos de actividades:
  - Planear una iteración (administrador del proyecto)
  - Encontrar casos de uso y actores (analista del dominio)
  - Revisión del diseño (probador)

Nota: El proceso Unificado (Unified Process ) utiliza actividad para lo que a veces se ha definido como **tarea**.

## Proceso Unificado.

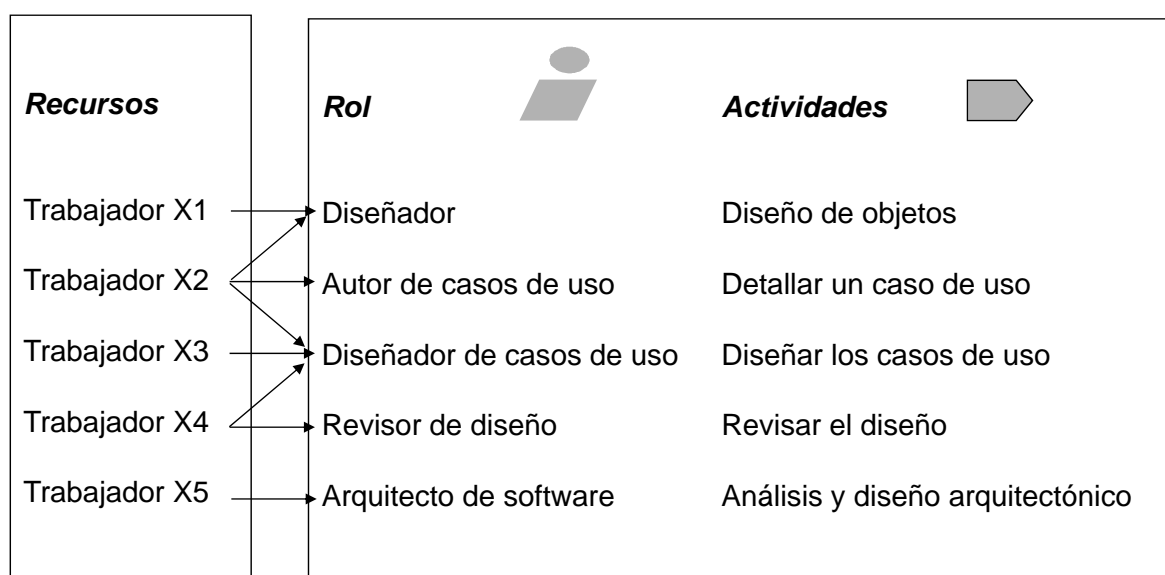
### Artefacto

- Pieza de información producida, modificada y utilizada en un proceso
- Producto tangible del proyecto
- Utilizado por los roles como entrada para la realización de sus actividades
- Resultado de las actividades realizadas por los roles
- En el metamodelo la clase rol tiene como métodos las actividades y como parámetros los artefactos

#### Conjuntos de Artefactos en el proceso unificado

- Business Modeling Set
- Requirements Set
- Analysis & Design Set
- Implementation Set
- Test Set
- Deployment Set
- Project Management Set
- Configuration & Change Management Set
- Environment Set

## Proceso Unificado.



### Recursos humanos y roles

# Proceso Unificado.

## Roles

### Analyst

- Business-Process Analyst
- Business Designer
- Business-Model Reviewer
- Requirements Reviewer
- System Analyst
- Use-Case Specifier
- User-Interface Designer

### Developer

- Architect
- Architecture Reviewer
- Capsule Designer
- Code Reviewer
- Database Designer
- Design Reviewer
- Designer
- Implementer
- Integrator

### Testing professional

- Test Designer
- Tester

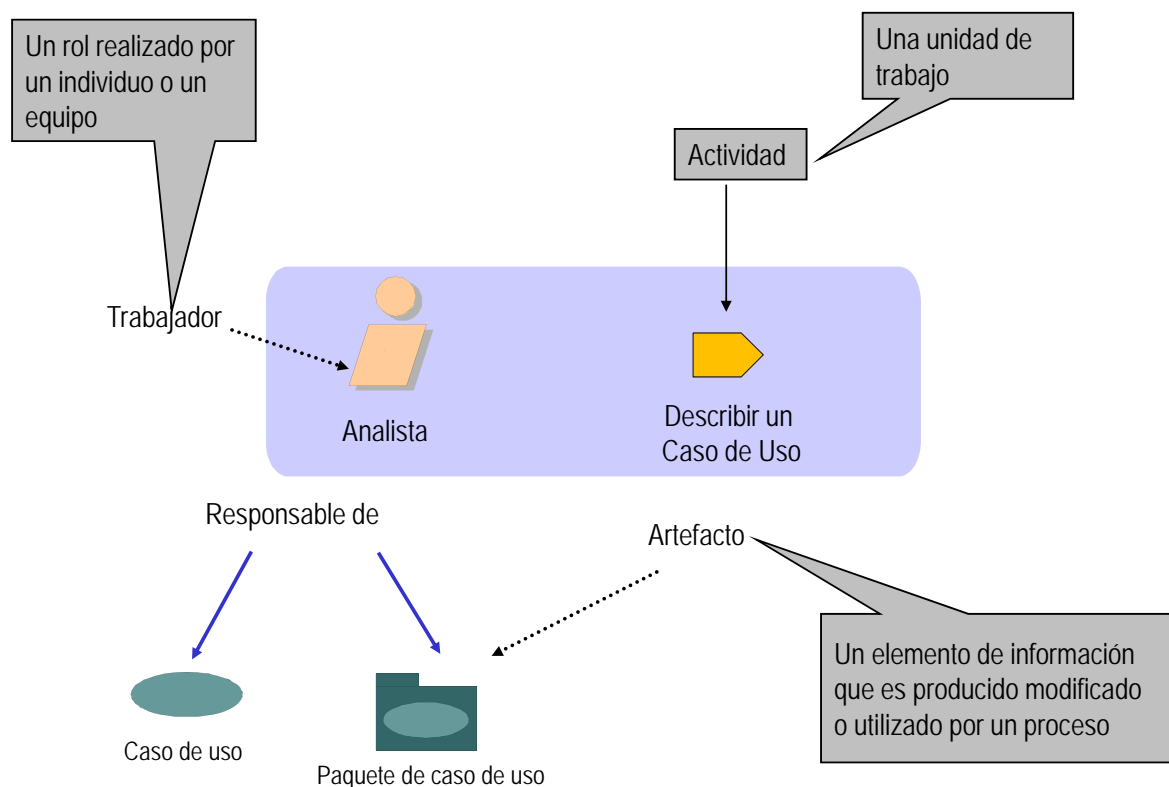
### Manager

- Change Control Manager
- Configuration Manager
- Deployment Manager
- Process Engineer
- Project Manager
- Project Reviewer

### Other

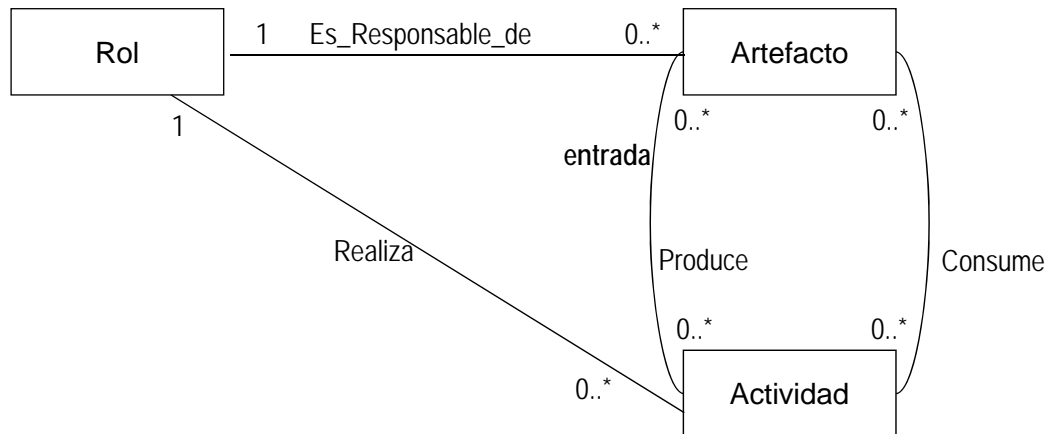
- Course Developer
- Graphic Artist
- Stakeholder
- System Administrator
- Technical Writer
- Tool Specialist

# Proceso Unificado.



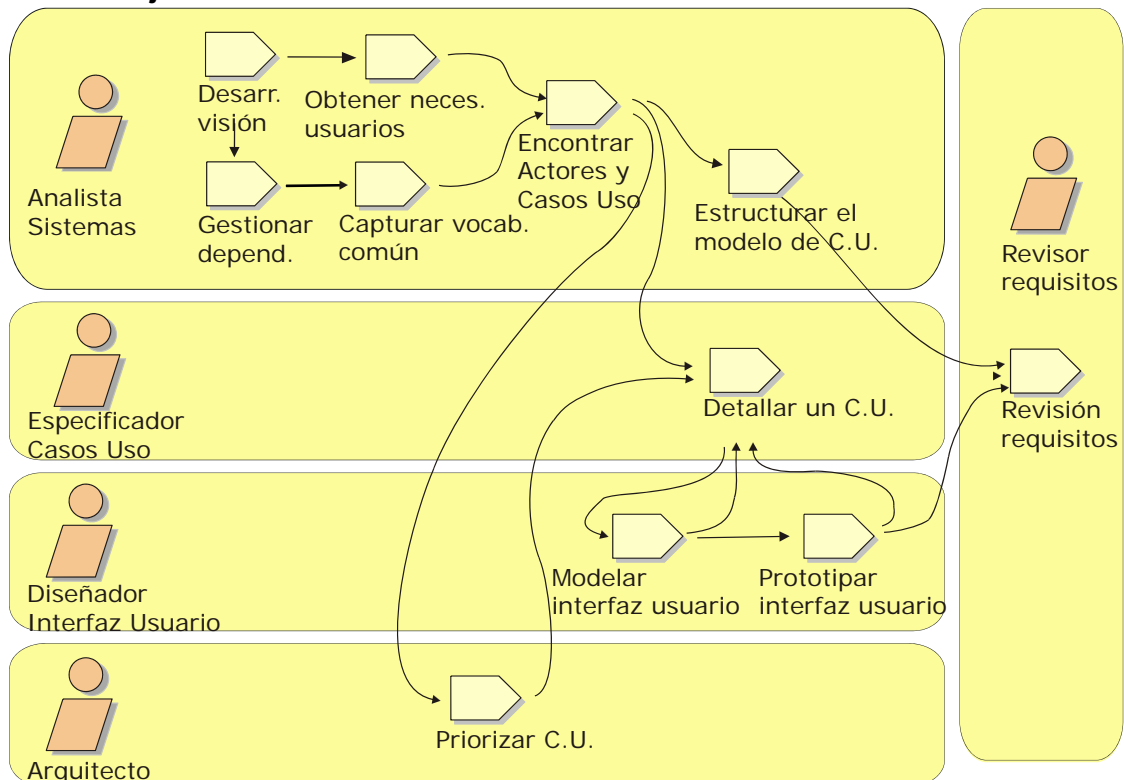
## Proceso Unificado.

### El modelo conceptual fundamental



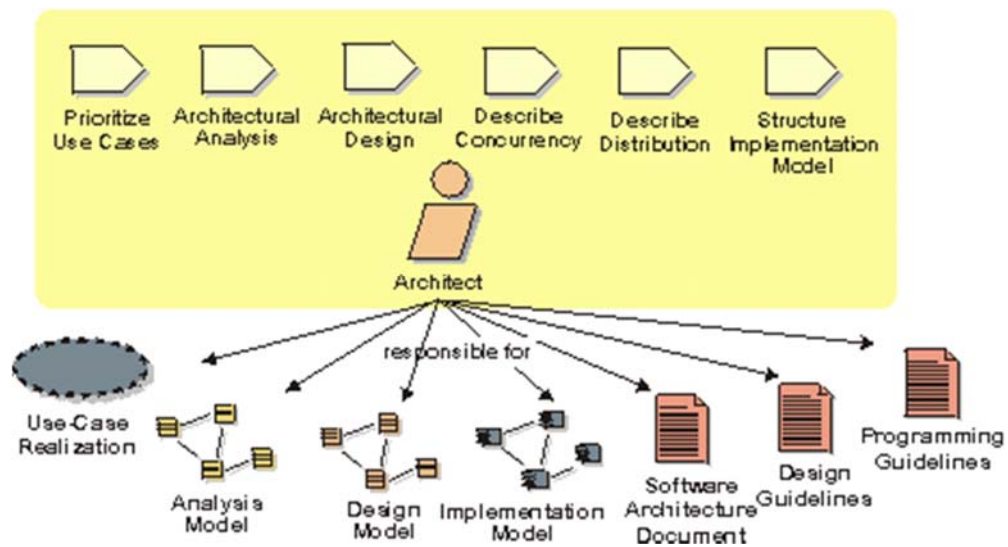
## Proceso Unificado.

### Flujos de trabajo





## Proceso Unificado.



### Actividades por trabajador (rol)

## Proceso Unificado.

### Ciclo de desarrollo

#### Flujos de Trabajo de Ingeniería

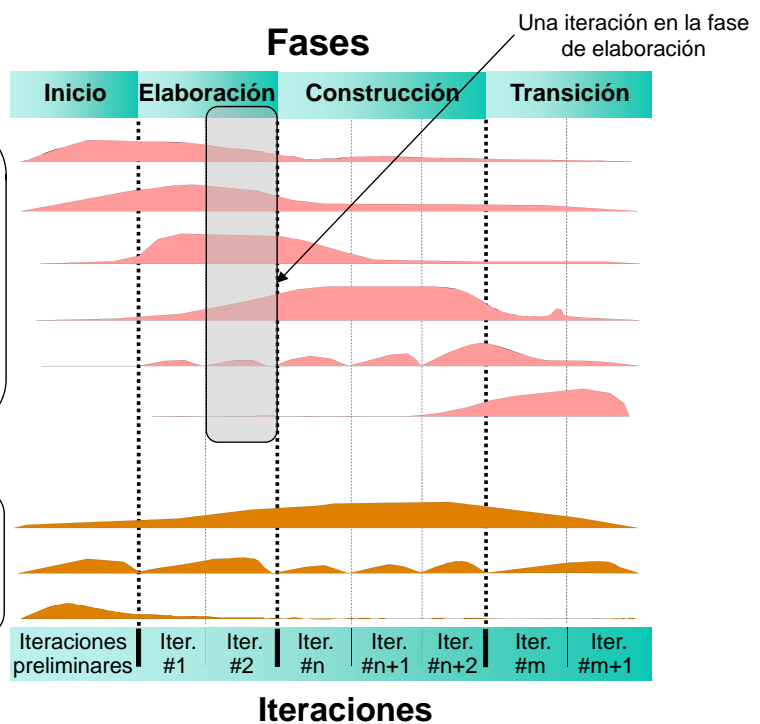
#### Flujos de proceso

**Modelado de negocio**  
**Requisitos**  
**Análisis & Diseño**  
**Implementación**  
**Prueba**  
**Lanzamiento**

#### Flujos accesorios

**Gestión de Configuración**  
**Gestión del proyecto**  
**Entorno**

#### Flujos de Trabajo de Apoyo



## Proceso Unificado.

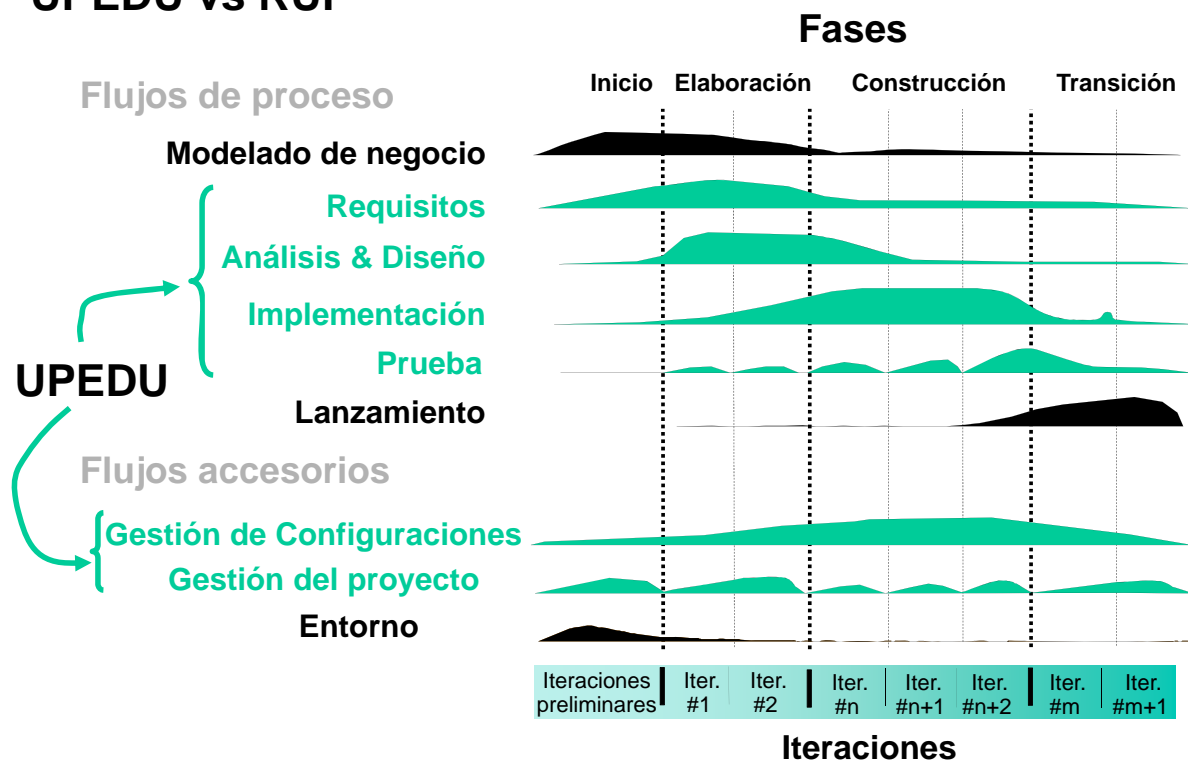
Estado de los casos de uso al finalizar cada fase

	Modelo de negocio terminado	Casos de uso identificados	Casos de uso descritos	Casos de uso analizados	Casos de uso diseñados, implementados y probados
Fase de Inicio	50% - 70%	50%	10%	5%	Muy poco, puede que solo algo relativo a probar conceptos
Fase de Elaboración	Casi el 100%	80% o más	40% - 80%	20% - 40%	Menos del 10%
Fase de Construcción	100%	100%	100%	100%	100%
Fase de Transición					

Fuente: *El Proceso Unificado de Desarrollo Software*. Ivar Jacobson, Grady Booch, James Rumbaugh. Addison Wesley, 1999

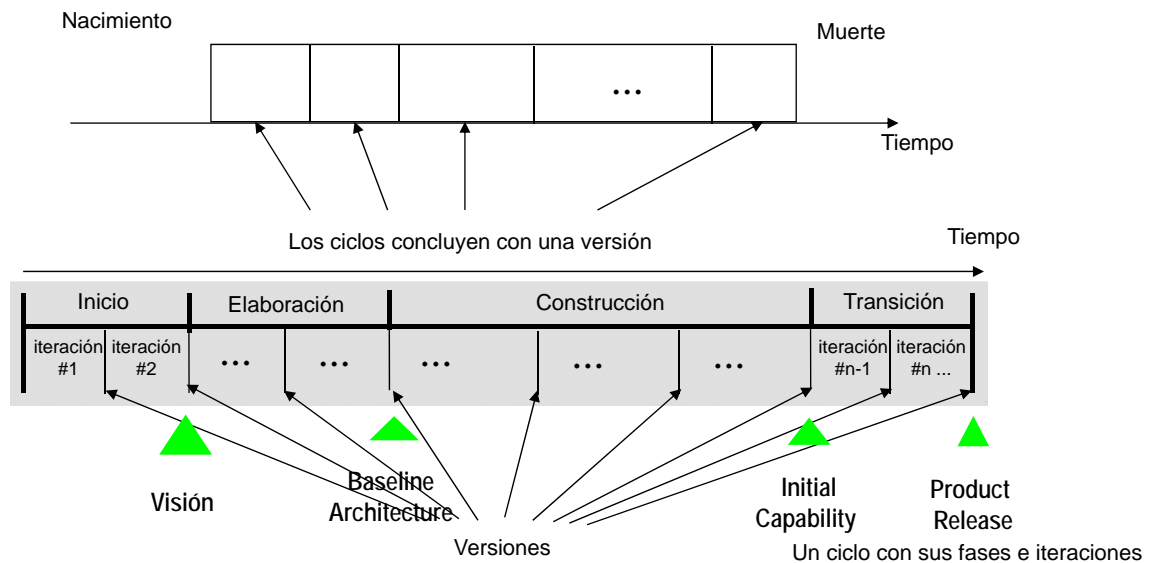
## Proceso Unificado.

### UPEDU vs RUP



# Proceso Unificado.

## Proceso Unificado: el ciclo de vida



- **Inicio** (inception): Define el ámbito del proyecto y desarrolla los casos de negocio.
- **Elaboración**: Se elabora el plan del proyecto, se especifican las características y el marco de referencia de la arquitectura.
- **Construcción**: se construye el proyecto.
- **Transición**: se pasa el producto a los usuarios.

# Proceso Unificado.

## Beneficios del desarrollo iterativo:

- Se produce una mitigación, en las etapas iniciales, de los riesgos más críticos ( técnicos, requisitos, objetivos, facilidad de uso, etc.).
- El progreso es visible desde muy pronto.
- La realimentación temprana, la implicación del usuario y la adaptación conducen a un sistema refinado que satisface más estrechamente las necesidades reales de los "usuarios".
- La complejidad se gestiona, el equipo no se abruma por la "parálisis del análisis" o por etapas muy grandes y complejas.
- El aprendizaje que se va obteniendo en cada iteración se puede utilizar metódicamente para mejorar el proceso de desarrollo, iteración a iteración.

## Proceso Unificado.

---

A medida que se avanza en el desarrollo de la ingeniería de software se debe definir un modelo de proceso para soportar lo siguiente:

- Evolución de los planes, los requisitos y arquitectura junto con puntos de sincronización bien definidos.
- Gestión de requisitos y medidas objetivas de progreso y calidad.
- Evolución de las capacidades del sistema a través de demostraciones de funcionalidad incremental.

La división del ciclo de vida en dos etapas:

- La *etapa de ingeniería*, caracterizada por equipos más pequeños, menos previsibles realizando actividades de diseño y síntesis.
- La *etapa de producción*, caracterizada por equipos más grandes pero más previsibles realizando actividades de construcción, prueba y lanzamiento.

## Proceso Unificado.

---

Las dos etapas del ciclo de vida: ingeniería y producción

Aspecto del ciclo de vida	Énfasis durante etapa de ingeniería	Énfasis durante etapa de producción
Reducción del riesgo	Planificación, factibilidad técnica	Coste
Productos	Línea base de la arquitectura	Líneas base de entregas de producto
Actividades (tareas)	Análisis, diseño, planificación	Implementación, pruebas
Valoración	Demostración, inspección, análisis	Pruebas
Economía	Resolución de la <i>diseconomía</i> de escala	Explotación de la economía de escala
Gestión	Planificación	Operaciones

La transición entre ingeniería y producción es un evento crucial para los “*stakeholders*”.

## Proceso Unificado.

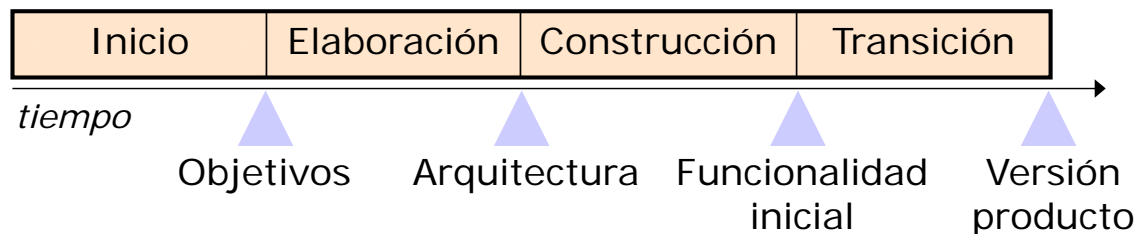
### Los diez puntos fundamentales de RUP

- Desarrollar una visión
- Gestionar el plan
- Identificar y mitigar los riesgos
- Asignar y seguir características
- Examinar los casos de negocio
- Diseñar una arquitectura de componentes
- Construir de forma iterativa e incremental y probar el producto
- Verificar y evaluar resultados
- Gestionar y controlar los cambios
- Proporcionar apoyo a los usuarios

Fuente: [http://www.therationaledge.com/content/dec\\_00/f\\_rup.html](http://www.therationaledge.com/content/dec_00/f_rup.html)

## Proceso Unificado.

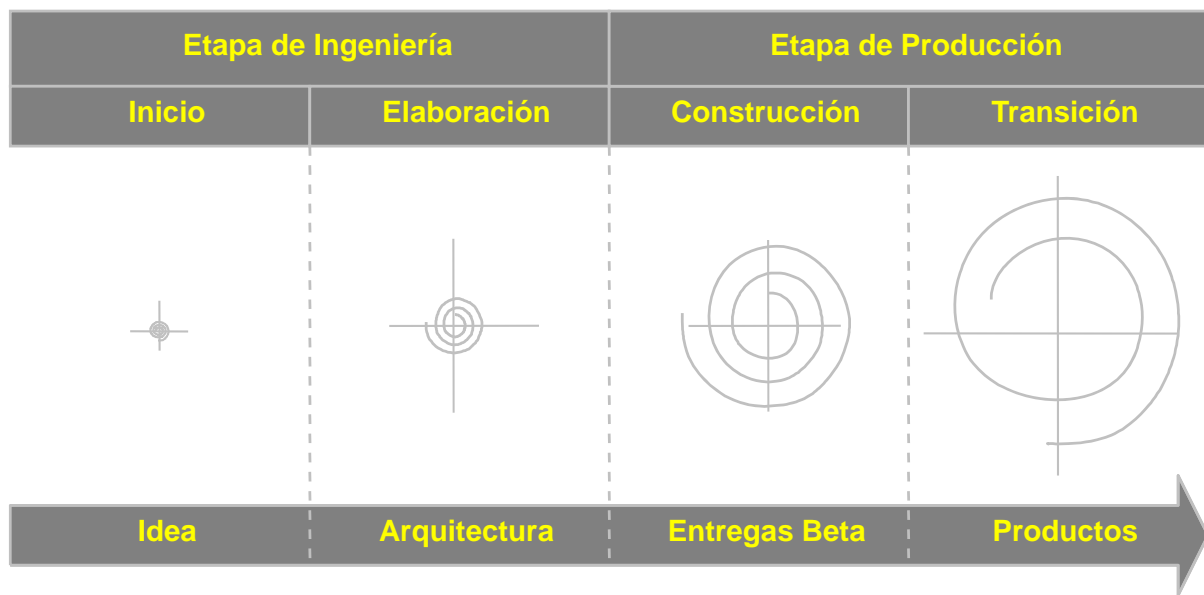
» cada fase define un conjunto de objetivos, actividades, productos y evaluación de hitos



- Inicio / objetivos
- Elaboración / arquitectura
- Construcción / producto inicial
- Transición / producto final

## Proceso Unificado.

### El ciclo de vida iterativo de RUP

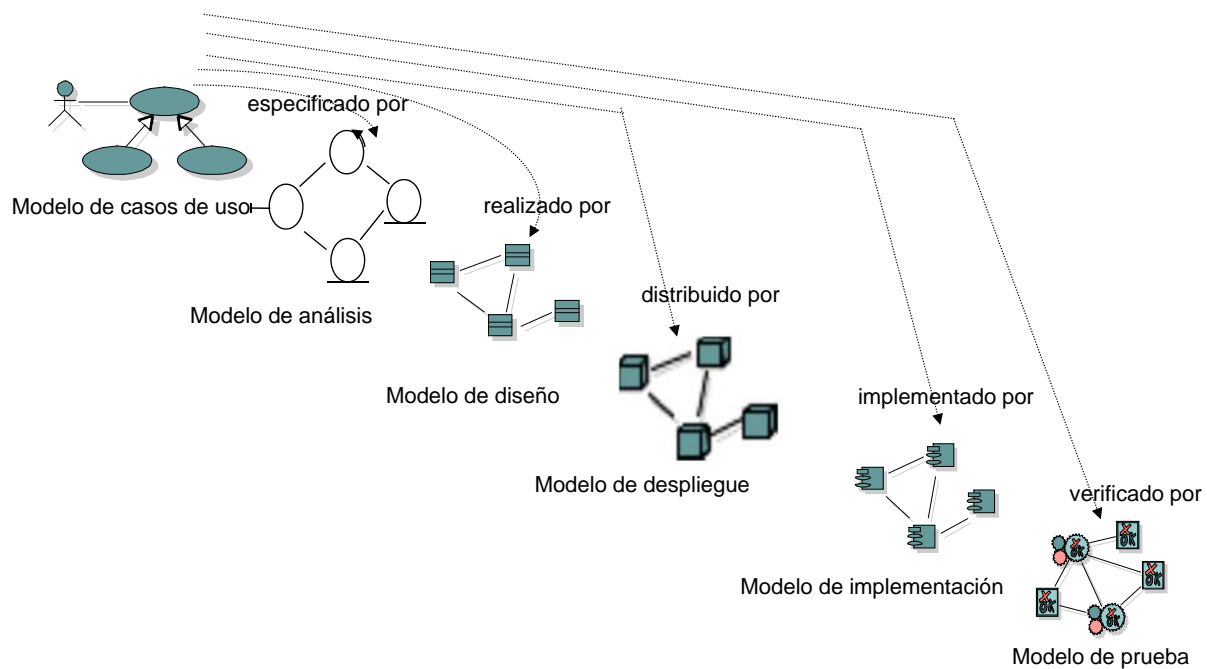


## Proceso Unificado.

### Modelos:

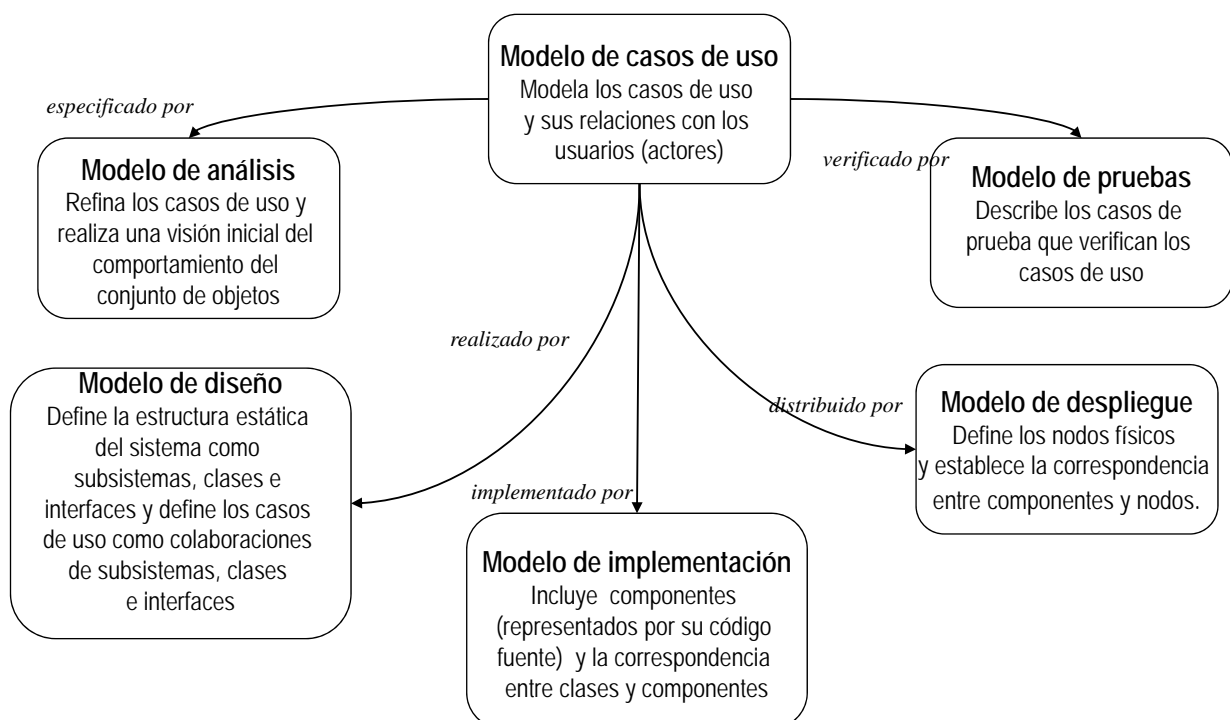
- Un modelo es una simplificación de la realidad con el objetivo de representar un sistema grande y complejo que sería difícil de comprender al considerarlo completamente.
- Un modelo no es la realidad sino una representación de ella. (Hayakawa, un lingüista, decía que el mapa no es el territorio).
- Un modelo único no puede cubrir todos los aspectos del desarrollo de software. Se necesitan múltiples modelos para cubrir todos ellos. Los modelos deben estar coordinados de forma adecuada.
- Los modelos son representaciones completas y consistentes del sistema a construir.

## Proceso Unificado.

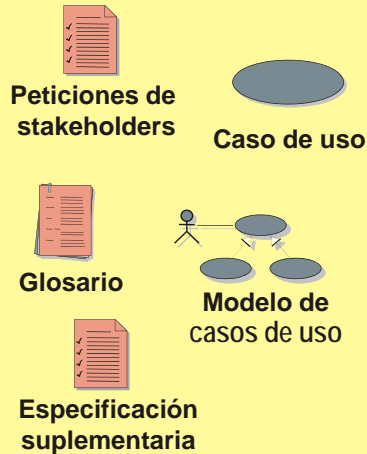


RATIONAL  
SOFTWARE

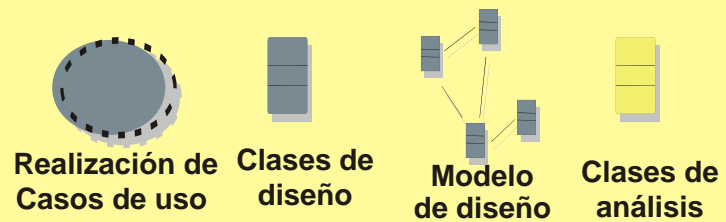
## Proceso Unificado.



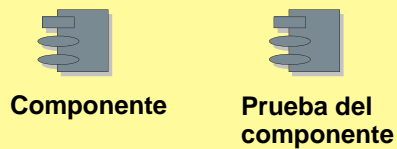
## Requisitos



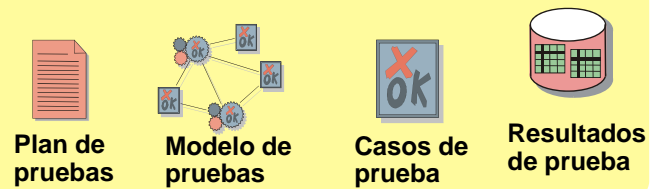
## Análisis and Diseño



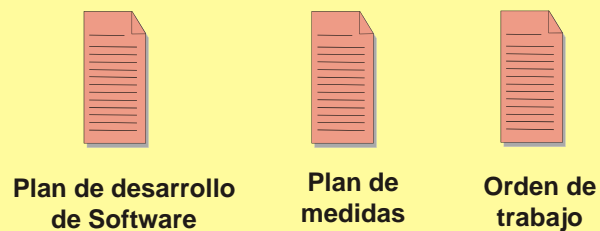
## Implementación



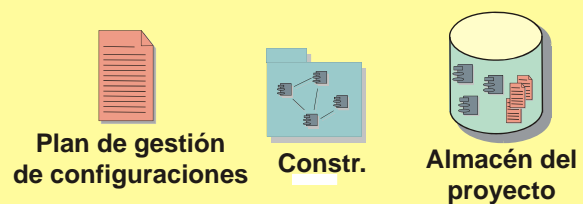
## Pruebas



## Gestión del Proyecto

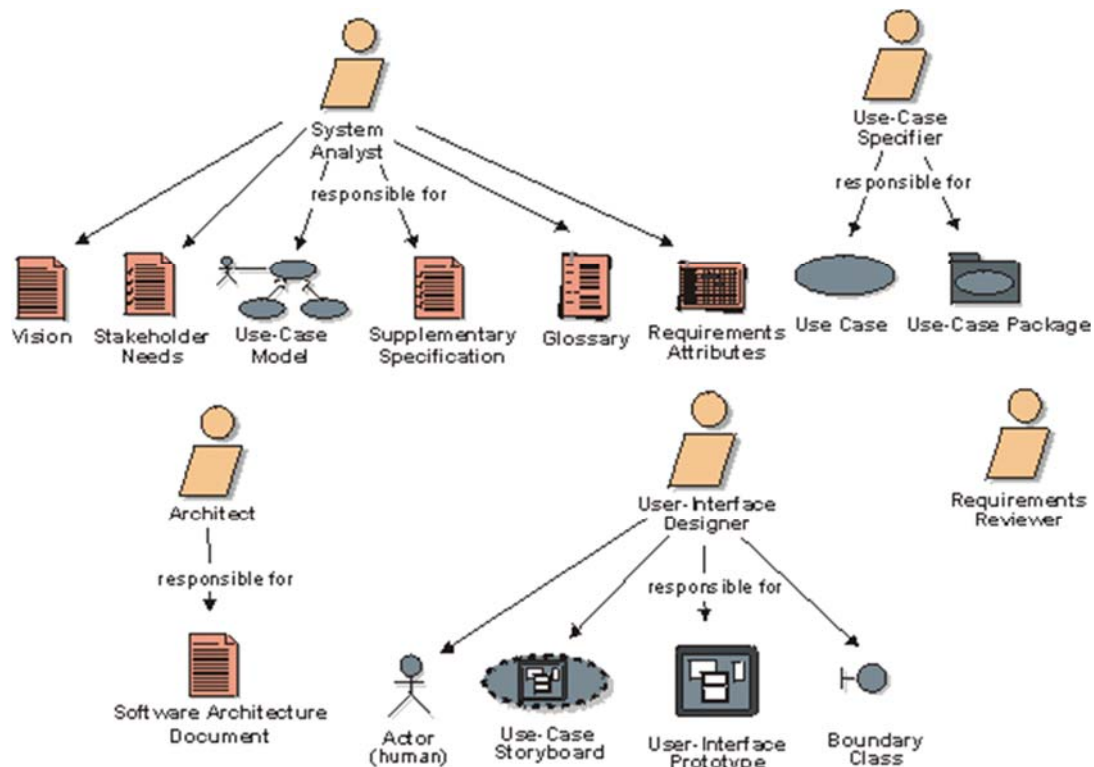


## Gestión de Config &amp; Cambios





## Proceso Unificado.



### Trabajadores y artefactos

Fuente: RUP

## Proceso Unificado.

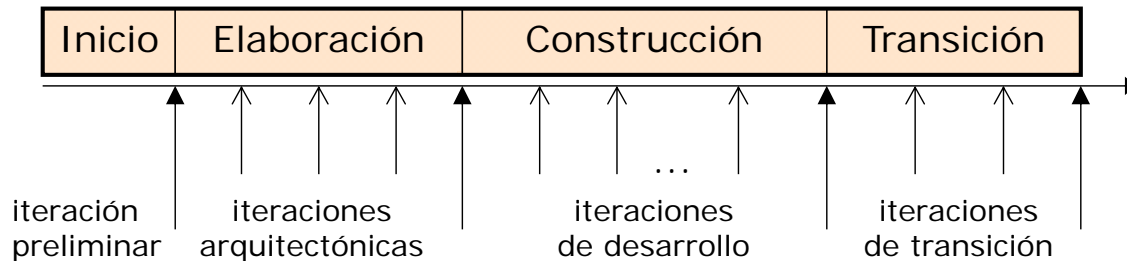


Fuente: Software project management. Walter Royce. 1998

## Proceso Unificado.

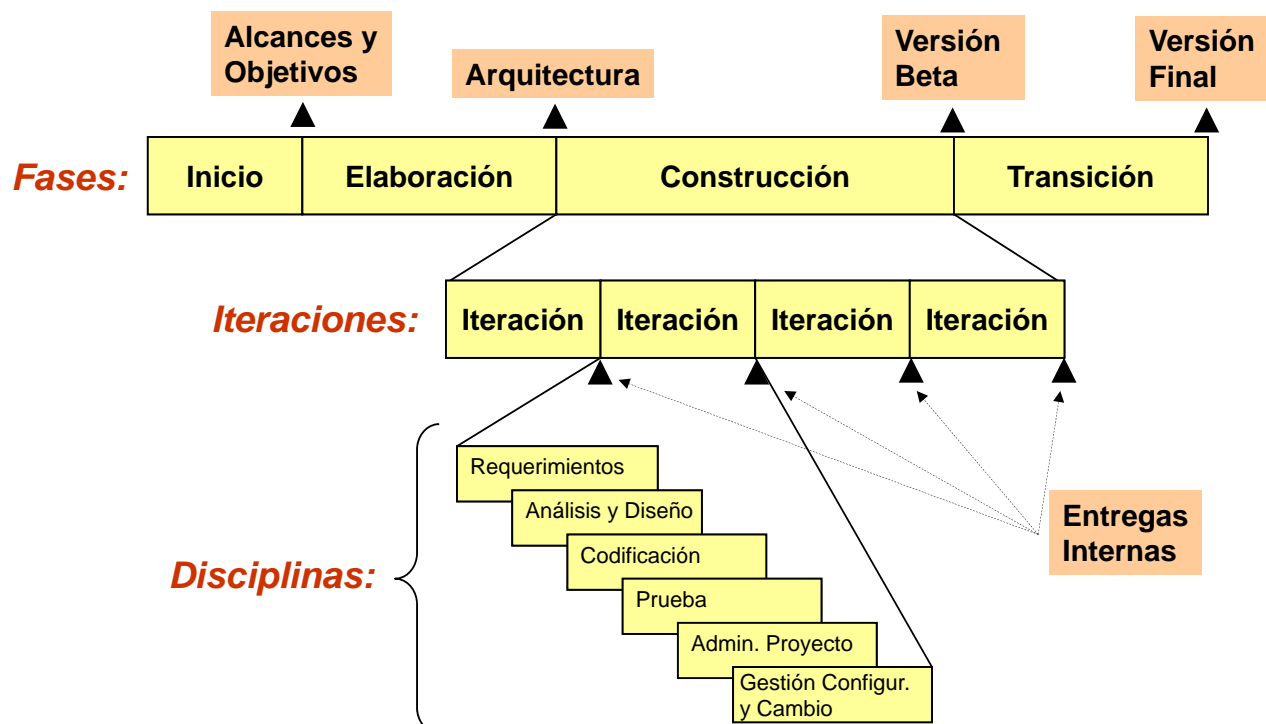
### Fases e iteraciones

» el software se desarrolla en *iteraciones*, cada una de las cuales *evoluciona incrementalmente*



- iteración es una secuencia planificada de actividades (en cascada) que finaliza en una versión (interna/externa)
- evoluciona incrementalmente hasta ser el producto final
- guiado por factores de riesgo

## Proceso Unificado.



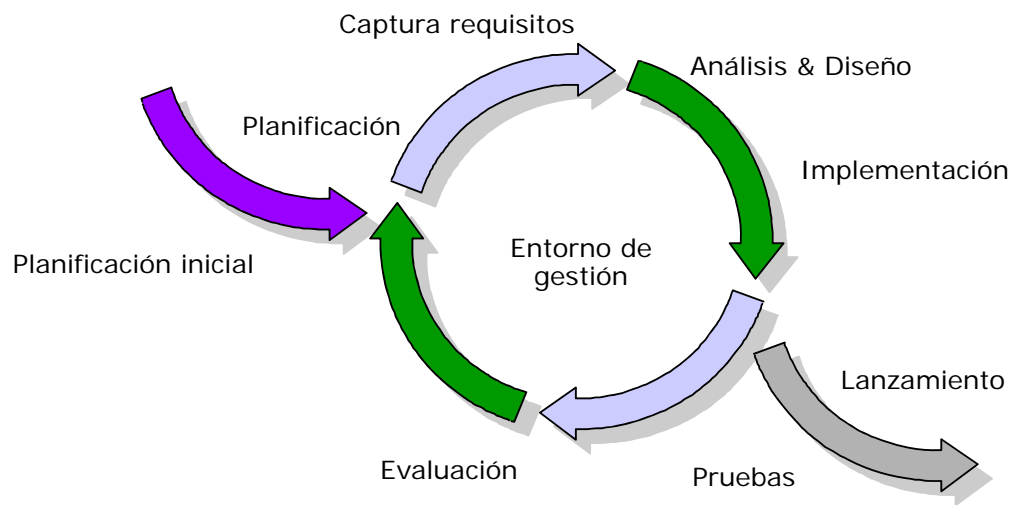
### Fases, Iteraciones y Disciplinas

Fuente: Baufest. Software Engineering

## Proceso Unificado.

---

### *Visión iterativa*



## Proceso Unificado.

---

### **Fase de inicio:**

Objetivos principales:

- Establecer el ámbito y los límites del proyecto, incluyendo conceptos operativos y criterios de aceptación.
- Localizar los casos de uso críticos y los escenarios principales que dirigirán los objetivos fundamentales del diseño.
- Comprobar al menos una arquitectura candidata frente a algunos de los escenarios principales.
- Estimar el coste y la planificación temporal del proyecto completo
- Estimar los riesgos potenciales.

## Proceso Unificado.

---

Actividades esenciales:

- Formular el ámbito del proyecto.
- Sintetizar la arquitectura.
- Planificar y preparar un caso de negocio.

Algunos criterios de evaluación:

- ¿Coinciden los “usuarios” con la definición del ámbito y las estimaciones de coste y tiempo?
- ¿Se han entendido totalmente los requisitos?
- ¿Son creíbles dichas estimaciones, así como las prioridades, los riesgos y el proceso de desarrollo?
- ¿La profundidad y amplitud del prototipo arquitectónico demuestran el criterio precedente?

Si el proyecto no satisface estos criterios habrá que replantear aquellos elementos que no han conseguido satisfacer alguno de los puntos anteriores.

## Proceso Unificado.

---

### ***Fase de inicio:***

- Se establece la oportunidad y alcance el proyecto, incluyendo conceptos operativos y criterios de aceptación.
- Se identifican todas las entidades externas con las que se trata (actores) y se define la interacción a un alto nivel de abstracción:
  - Identificar todos los casos de uso
  - Localizar los casos de uso críticos y los escenarios principales que dirigirán los objetivos fundamentales del diseño.
- La oportunidad del negocio incluye:
  - Criterios de éxito
  - Identificación de riesgos
  - Estimación de recursos necesarios
  - Plan de las fases incluyendo hitos

## Proceso Unificado.

---

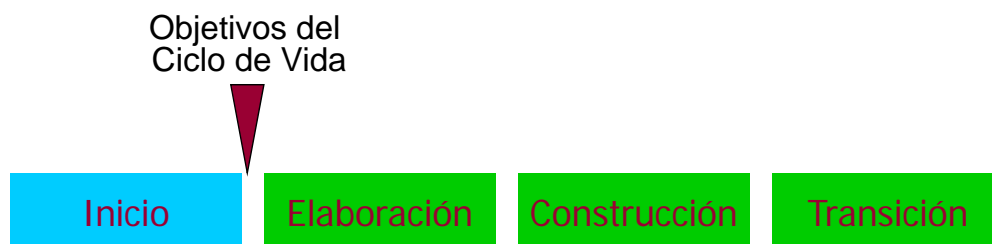
### ***Fase de inicio. Productos resultantes:***

- Un documento de visión general:
  - Requisitos generales del proyecto
  - Características principales
  - Restricciones
- Modelo inicial de casos de uso (10% a 20 % listos).
- Glosario inicial: terminología clave del dominio.
- Caso de negocio:
  - Contexto
  - Criterios de éxito
  - Pronóstico financiero
- Identificación inicial de riesgos.
- Plan de proyecto, mostrando las fases e iteraciones.
- Modelo de negocio si es necesario.
- Uno o más prototipos exploratorios para probar conceptos o la arquitectura candidata.

## Proceso Unificado.

---

### ***Fase de inicio. Hito:***



- Las partes interesadas deben acordar el alcance y la estimación de tiempo y costo.
- Comprensión de los requisitos plasmados en casos de uso.

## Proceso Unificado.

---

### Elaboración

- **Objetivos:**
  - Analizar el dominio del problema
  - Establecer una arquitectura base sólida
  - Desarrollar un plan de proyecto ajustado para la fase de construcción.
  - Eliminar los elementos de mayor riesgo para el desarrollo exitoso del proyecto
- Las decisiones de arquitectura requieren una visión global del sistema.
  - Actividades esenciales:
    - Elaborar la visión. Desarrollar los casos de uso a un nivel que dirija las decisiones de la arquitectura o de planificación.
    - Elaborar el proceso y la infraestructura.
    - Elaborar la arquitectura y los componentes seleccionados.

## Proceso Unificado.

---

### Elaboración. Productos:

- Es la parte más crítica del proceso:
  - Al final toda la ingeniería “dura” está hecha
  - Se puede decidir si vale la pena seguir adelante
- A partir de aquí la arquitectura, los requisitos y los planes de desarrollo son estables.
- Ya hay menos riesgos y se puede planificar el resto del proyecto con menor incertidumbre.
- Se construye una arquitectura ejecutable que contemple:
  - Los casos de uso críticos
    - Los riesgos identificados

## Proceso Unificado.

---

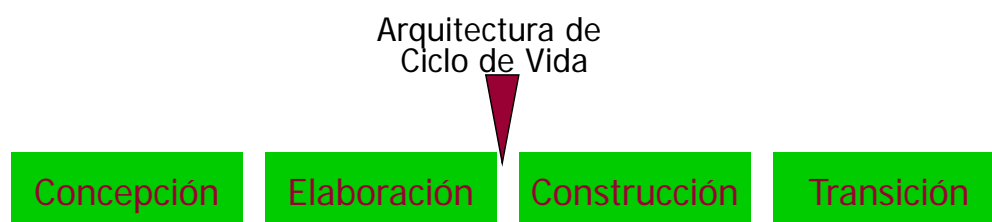
### Elaboración. Productos:

- Modelo de casos de uso (80% completo) con descripciones detalladas.
- Otros requisitos no funcionales o no asociados a casos de uso.
- Descripción de la Arquitectura del Software.
- Un prototipo ejecutable de la arquitectura.
- Lista revisada de riesgos y del caso de negocio.
- Plan de desarrollo para el resto del proyecto.
- Un manual de usuario preliminar.

## Proceso Unificado.

---

### Elaboración. Hito:



- Criterios de evaluación:
  - ¿Es estable la visión del producto?
  - ¿Es estable la arquitectura?
  - ¿Las pruebas de ejecución demuestran que los riesgos han sido abordados y resueltos?
  - ¿El plan del proyecto es algo realista?
  - Las personas involucradas ¿están de acuerdo con el plan?

## Proceso Unificado.

---

### **Fase de Construcción:**

Objetivos principales:

- Minimizar los costes de desarrollo optimizando los recursos e impidiendo la realización de trabajo “dos veces”.
- Conseguir una calidad adecuada.
- Conseguir versiones utilizables (alfa, beta, de prueba).

Actividades esenciales:

- Gestión y control de los recursos y optimización del proceso.
- Desarrollo de componentes completos y prueba de los mismos frente a criterios de evaluación.
- Valoración de las versiones de producto frente a los criterios de aceptación

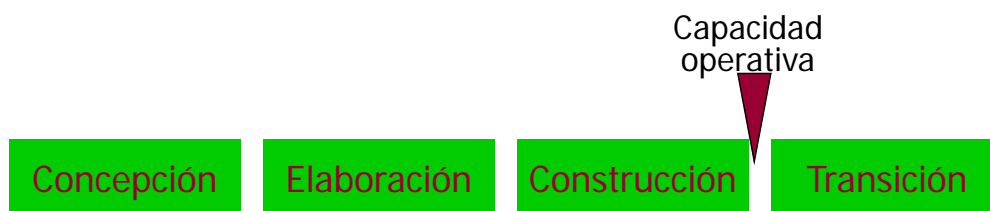
Algunos criterios de evaluación:

- ¿El producto es lo suficientemente estable y maduro como para ser entregado al usuario? (la existencia de defectos o cambios pendientes no son un obstáculo para conseguir el propósito en la entrega siguiente).
- ¿Están los usuarios dispuestos para la transición?
- ¿Los gastos actuales de los recursos son aceptables todavía frente a los planeados?

## Proceso Unificado.

---

### **Construcción. Hito:**



- Se obtiene un producto Beta y se debe decidir si puede ponerse en ejecución sin mayores riesgos.
- Condiciones de éxito:
  - ¿El producto está maduro y estable para instalarlo en el entorno del cliente?
  - ¿Están los interesados listos para recibirlo?



# Proceso Unificado.

---

## **Fase de Transición:**

Objetivos principales:

- Conseguir que el usuario sea capaz de mantener el producto.
- Conseguir la aceptación por el usuario (*stakeholder*) que lo entregado es completo y consistente con el criterio de evaluación fijado en la visión inicial del proyecto.
- Conseguir un producto final tan rápido y eficiente respecto al coste como práctico.

Actividades esenciales:

- Sincronización e integración de los incrementos de construcción concurrentes en líneas base de entrega consistente.
- Realizar una ingeniería específica de entrega (producción y empaquetamiento comercial, plan de formación del personal, etc.).
- Valoración de las líneas base de entrega frente a la visión global y los criterios de aceptación en el conjunto de requisitos.

Algunos criterios de evaluación:

- ¿El usuario está satisfecho?
- ¿La utilización de recursos actual es aceptable frente a la planificada?

# Proceso Unificado.

---

## **Arquitectura**

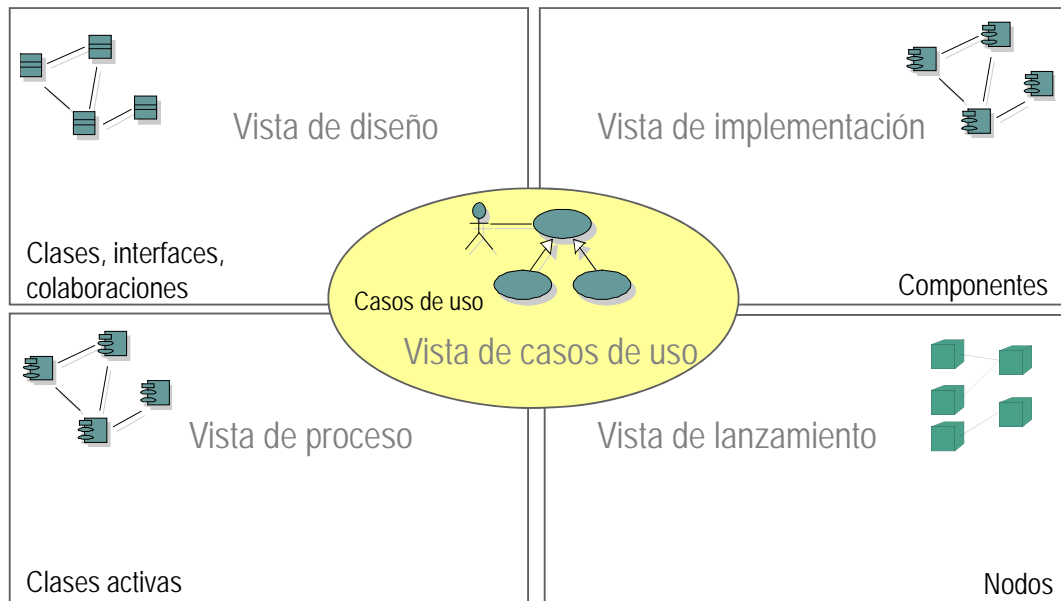
Visualizar, especificar, construir y documentar un sistema software precisa una visión del sistema desde distintos puntos de vista.

Cada actor implicado en el proceso mira al sistema desde una perspectiva diferente a lo largo de la vida del proyecto. La arquitectura del sistema es uno de los productos más importantes que se pueden utilizar para gestionar estos puntos de vista y, por tanto, para controlar el desarrollo del sistema a través de su ciclo de vida.

Una arquitectura del sistema abarca el conjunto de decisiones importantes sobre:

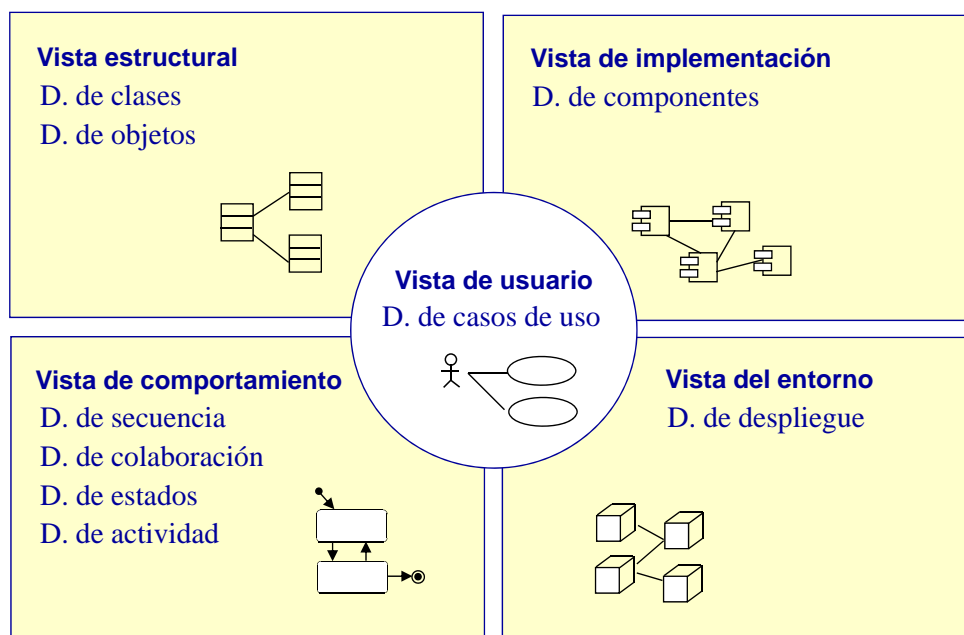
- La organización del sistema software.
- La selección de los elementos estructurales y sus interfaces, de los que está compuesto el sistema.
- Su comportamiento, especificado como colaboración entre los distintos componentes.
- La composición de estos elementos (estructurales y de comportamiento) en subsistemas cada vez más grandes.
- El estilo arquitectónico que guía las organizaciones.

## Proceso Unificado.



- Los casos de uso especifican la función, la arquitectura la forma
- Debe existir un balance entre casos de uso y arquitectura

## Proceso Unificado.



Diagramas ligados a cada vista

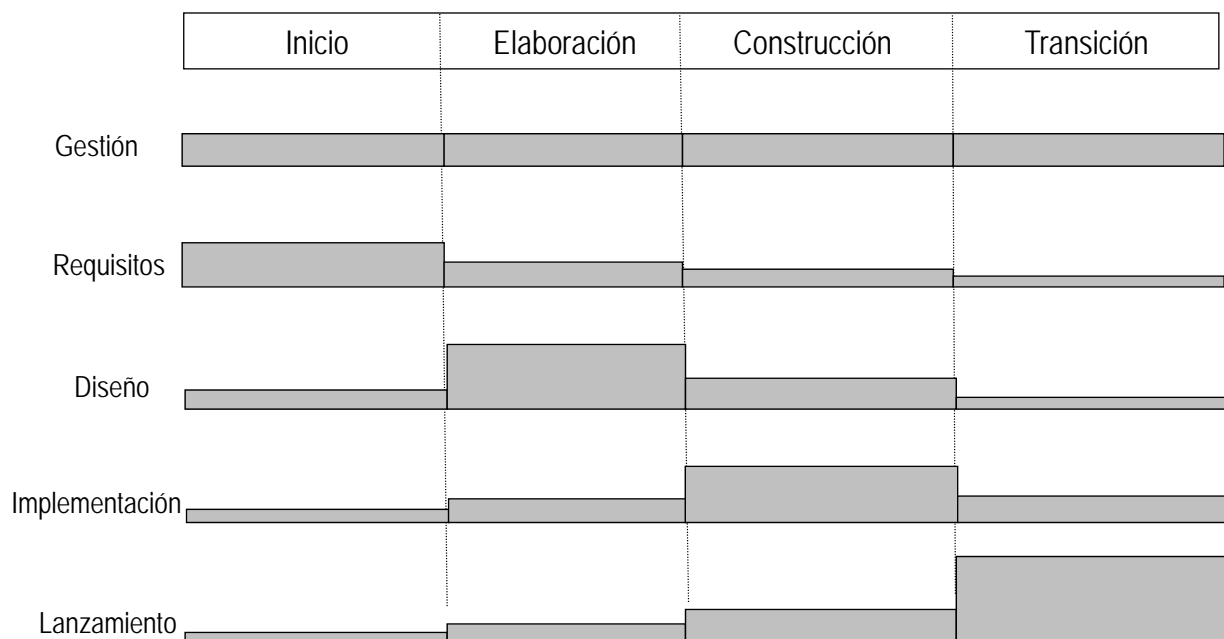
## Proceso Unificado.

Conjunto de requisitos	Conjunto de diseño	Conjunto de implementación	Conjunto de lanzamiento
1.- Documento de visión 2.- Modelo de requisitos	1.- Modelo(s) de diseño 2.- Modelo de pruebas 3.- Descripción de la arquitectura del software	1.- Reglas básicas para código fuente 2.- Ficheros en tiempo de compilación asociados 3.- Componentes ejecutables	1.- Reglas básicas para producto ejecutable 2.- Ficheros run-time asociados 3.- Manual de usuario
Conjunto de gestión			
<b>Artefactos de planificación</b> 1.- Estructura de descomposición del trabajo 2.- Casos de negocio 3.- Especificaciones de revisión 4.- Plan de desarrollo de software		<b>Artefactos operativos</b> 5.- Descripciones de versión 6.- Valoraciones de estado 7.- Base de datos de órdenes de cambio de software 8.- Documentos de lanzamiento 9.- Entorno	

### Visión general de los conjuntos de artefactos

Fuente: Software project management. Walter Royce. 1998

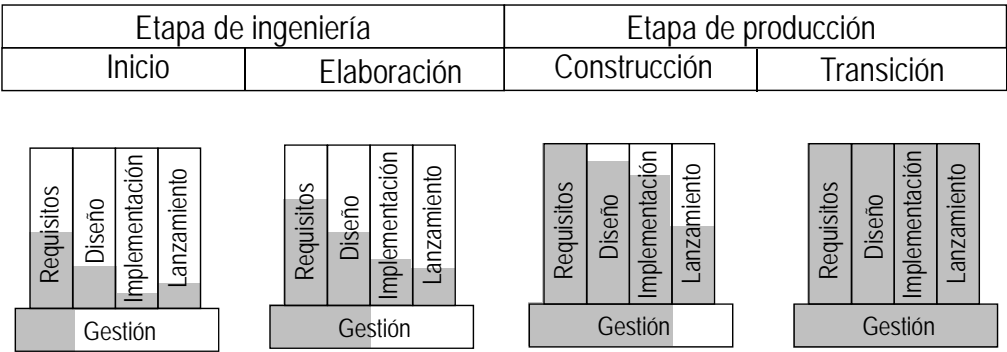
## Proceso Unificado.



### Distribución de los conjuntos de artefactos a lo largo del ciclo de vida

Fuente: Software project management. Walter Royce. 1998

# Proceso Unificado.

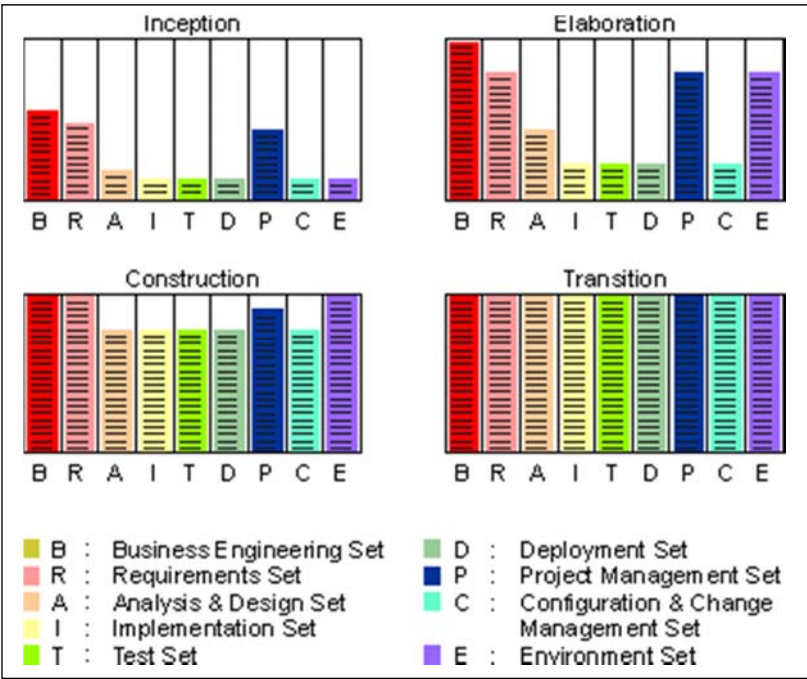


## Evolución de los conjuntos de artefactos a lo largo del ciclo de vida

Fuente: Software project management. Walter Royce. 1998

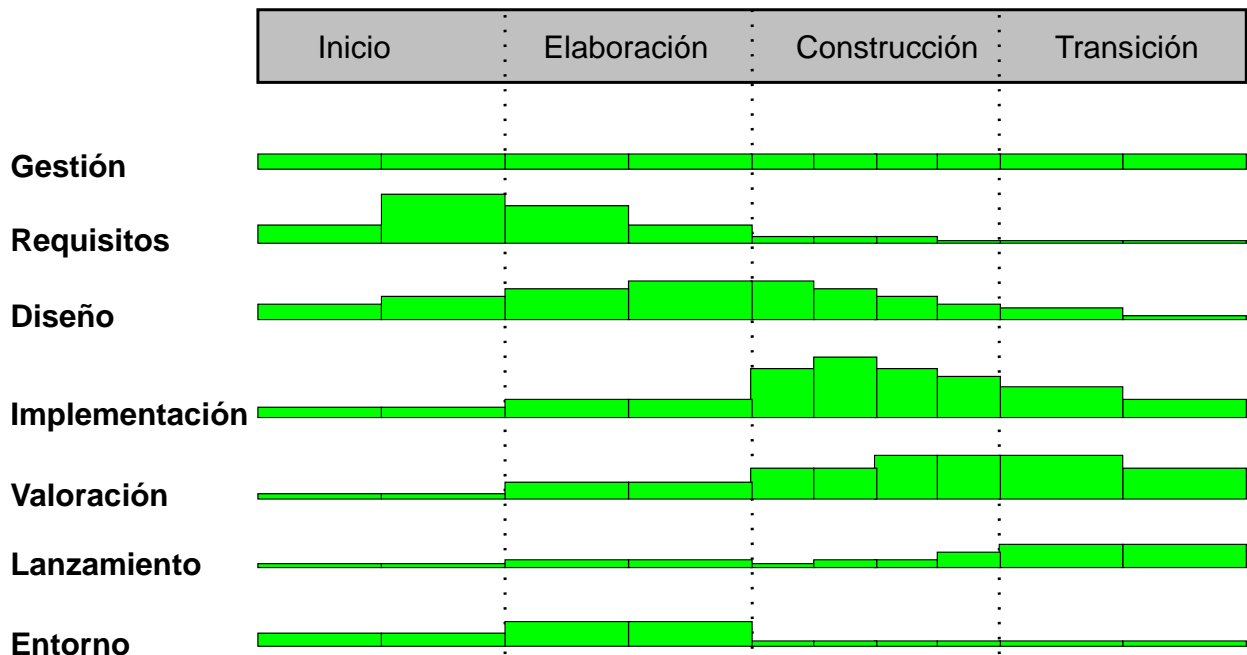
# Proceso Unificado.

## Grado de Finalización de los conjuntos de Artefactos en las distintas fases



## Proceso Unificado.

### Actividades del ciclo de vida iterativo (con iteracciones incluidas)



## Proceso Unificado.

### *Iterativo e incremental*

Basado en casos de uso significa ajustarse a las necesidades reales del usuario.

Centrado en la arquitectura significa que el trabajo de desarrollo se centra en obtener un patrón que dirigirá la construcción del sistema en las primeras fases.

Conseguir un equilibrio entre casos de uso y arquitectura es algo parecido al equilibrado entre forma y función en el desarrollo de cualquier producto. La mejor forma de conseguirlo es mediante un desarrollo iterativo e incremental.

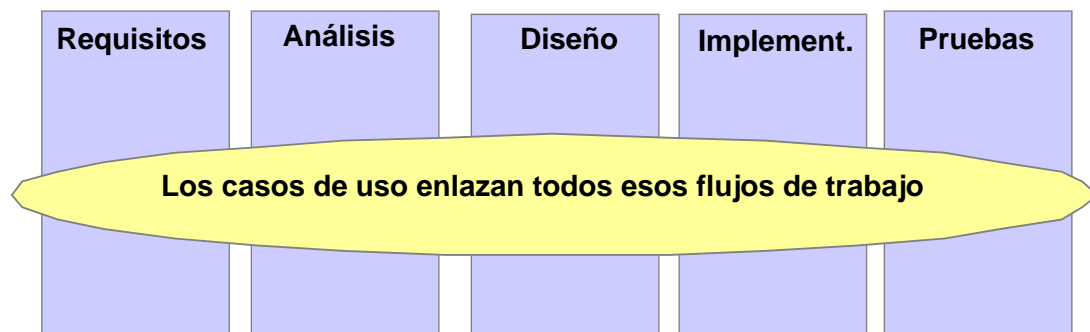
Esto plantea una estrategia de desarrollo para productos software basada en pequeños pasos manejables:

- Planificar poco.
- Especificar, diseñar e implementar un poco.
- Integrar, probar y ejecutar un poco en cada iteración.

Si en cada paso el resultado es satisfactorio se avanza. Además esto nos permite ajustar los objetivos concretos en cada paso. Cada iteración tiene todo lo que debe tener un proyecto de software: planificación, desarrollo de una serie de flujos y preparación para la entrega. Pero una iteración no es algo totalmente independiente, es una etapa dentro de un proyecto

Cada iteración nos permite avanzar en la comprensión de cuestiones relativas al proyecto. Por ejemplo las primeras nos ayudan a comprender los riesgos, determinar la viabilidad, etc.

## Proceso Unificado.



- Los casos de uso dirigen unas determinadas actividades de desarrollo.
  - Creación y validación de la arquitectura del sistema
  - Definición de los casos y procedimientos de prueba.
  - Planificación de las iteraciones
  - Creación de la documentación del usuario
  - Lanzamiento del sistema
- Sincronizan el contenido de los diferentes modelos.

RATIONAL  
SOFTWARE

## Proceso Unificado.

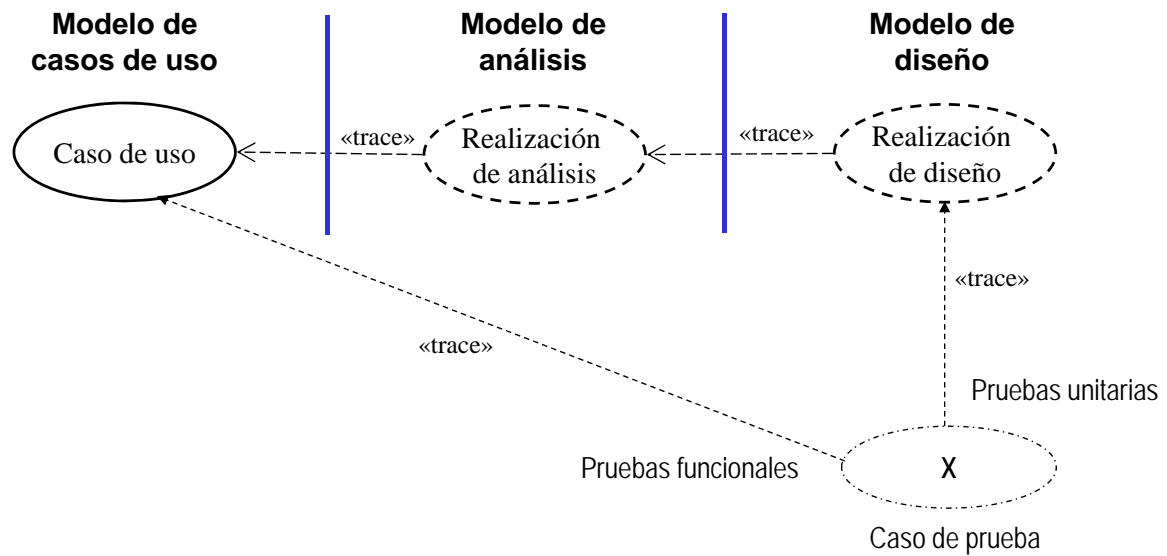
### Evolución de los casos de uso a lo largo de las fases

#### Estado de aspectos de los Casos de Uso al finalizar cada fase

	Modelo de Negocio Terminado	Casos de Uso Identificados	Casos de Uso Descritos	Casos de Uso Analizados	Casos de Uso Diseñados, Implementados y Probados
<b>Fase de Concepción</b>	50% - 70%	50%	10%	5%	Muy poco, puede que sólo algo relativo a un prototipo para probar conceptos
<b>Fase de Elaboración</b>	Casi el 100%	80% o más	40% - 80%	20% - 40%	Menos del 10%
<b>Fase de Construcción</b>	100%	100%	100%	100%	100%
<b>Fase de Transición</b>					

The Unified Software Development Process. I. Jacobson, G. Booch y J. Rumbaugh. página 358. Addison-Wesley, 1999.

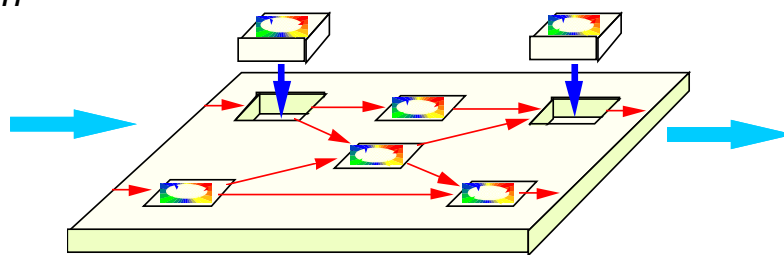
## Proceso Unificado.



Ejemplo de seguimiento por los casos de uso

## Proceso Unificado.

### Resumen



No existe un proceso universal

El proceso unificado está diseñado para flexibilidad y extensibilidad

- » permite una variedad de estrategias de ciclos de vida
- » selecciona qué artefactos producir
- » define las actividades y trabajadores
- » considera los conceptos de los modelos

## Proceso Unificado.

---

En un informe (*white paper*) revisado en 2011 por personas de Rational consideran que RUP facilita las siguientes mejores prácticas para el desarrollo de software:

1. Desarrollar el 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. Controlar los cambios del software

Fuente: *Rational Unified Process Best Practices for Software Development Teams*. Rational Software White Paper TP026B, Rev 11/01

## Proceso Unificado.

---

Hay actividades que los autores reconocen que no están tratadas en RUP pero que se podrían tratar de la misma forma que los flujos de trabajo. Por ejemplo:

- La interacción con los clientes para nuevos requisitos.
- La preparación de una oferta para los clientes.
- La comprensión del contexto de un sistema creando un modelo de negocio.
- La planificación y administración de un proyecto.
- El establecimiento y administración del entorno de desarrollo: proceso y herramientas.
- La administración de los riesgos.
- La instalación del producto en el lugar de destino.



## Proceso Unificado.

---

### Beneficios:

- » Reducción temprana de los errores
  - » Gestión de los cambios
  - » Mayor nivel de reutilización
  - » Evaluaciones y mejoras periódicas
  - » Mejor calidad final
- 
- » ¿Inconvenientes?

## Proceso Unificado.

---

### Una visión de la evolución del proceso de Software

RUP Workflow	Proceso convencional (Waterfall)	Proceso moderno (Iterativo)	Proceso futuro
Gestión	5%	10%	12%
Requisitos	5%	10%	12%
Diseño	10%	15%	20%
Implementación	30%	25%	14%
Pruebas & Valoración	40%	25%	18%
Lanzamiento	5%	5%	12%
Entorno	<u>5%</u>	<u>10%</u>	<u>12%</u>
	100%	100%	100%

Más balance, se consume menos en integración y pruebas

## Proceso Unificado.

Reparto del presupuesto de acuerdo con la descomposición de tareas por defecto:

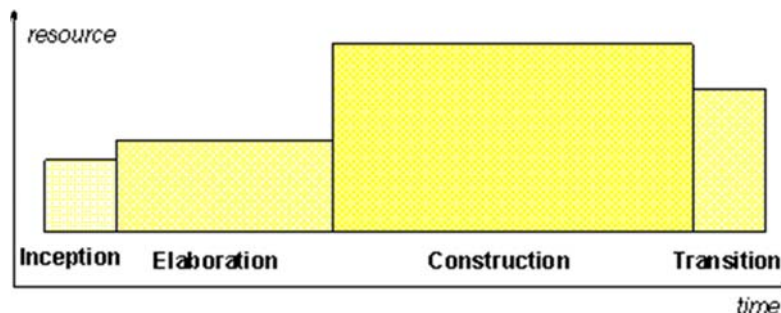
Elementos del primer nivel de WBS	Presupuesto por defecto
Gestión	10%
Entorno	10%
Requisitos	10%
Diseño	15%
Implementación	25%
Pruebas	25%
Lanzamiento	5%
<b>Total</b>	<b>100%</b>

Fuente: *Philippe Krutchen. A Rational Development Process*

## Proceso Unificado.

Distribución por defecto del esfuerzo y el tiempo por fase en un proyecto normal:

Dominio	Inicio	Elaboración	Construcción	Transición
Esfuerzo	5%	20%	65%	10%
Tiempo	10%	30%	50%	10%

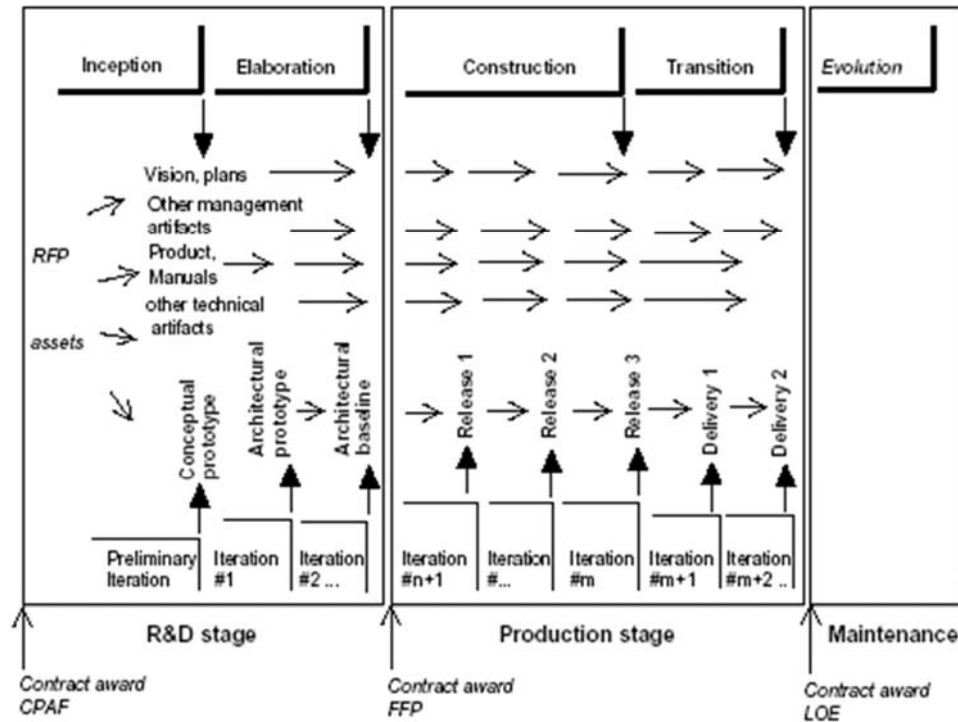


Distribución por defecto del esfuerzo y el tiempo por fase en un proyecto difícil:

Dominio	Inicio	Elaboración	Construcción	Transición
Esfuerzo	8%	24%	60%	8%
Tiempo	20%	33%	40%	7%

Fuente: Ivar Jacobson, Grady Booch, James Rumbaugh. *"El Proceso Unificado de Desarrollo Software"*. Addison Wesley, 1999

## Proceso Unificado.



## Proceso Unificado.

El ejemplo 1 ilustra los detalles de un proyecto muy pequeño que consiste de 5.000 líneas de código Java nuevo y que precisa unas 12 personas-mes en 7 meses de tiempo.

	Inception	Elaboration	Construction	Transition
Staff	1	1.5	2	2
Duration in weeks	3	6	18	3
Number of iterations (and duration of each in weeks)	1 (3)	1 (6)	3 (6)	1 (3)

En el ejemplo 2 se ven los detalles de un proyecto pequeño de 10.000 líneas de código Java nuevo, que precisa unas 27 personas-mes en 8 meses de tiempo.

	Inception	Elaboration	Construction	Transition
Staff	1.5	2.5	4	3
Duration in weeks	4	7	20	4
Number of iterations (and duration of each in weeks)	1 (4)	1 (7)	3 (7)	1 (4)

## Proceso Unificado.

El ejemplo 3 ilustra los detalles de un proyecto medio que consiste de 40.000 líneas de código Java nuevo, que precisa unas 115 personas-mes en 15 meses de tiempo.

	Inception	Elaboration	Construction	Transition
Staff	3	5	10	8
Duration in weeks	6	16	36	6
Number of iterations (and duration of each in weeks)	1 (6)	2 (8)	4 (9)	1 (6)

El ejemplo 4 ilustra los detalles de un proyecto grande que consiste de 1.500.000 líneas de código Java nuevo, que precisa unas 4.600 personas-mes en 45 meses de tiempo.

	Inception	Elaboration	Construction	Transition
Staff	35	70	140	100
Duration in weeks	20	50	100	20
Number of iterations (and duration of each in weeks)	2 (10)	2 (25)	3 (33)	2 (10)

Fuente: *John Smith*. A Comparison of the IBM Rational Unified Process and eXtreme Programming. 2001

## Proceso Unificado.

### Fuentes Bibliográficas:

Ivar Jacobson, Grady Booch, James Rumbaugh. **“El Proceso Unificado de Desarrollo Software”**. Addison Wesley, 1999. El capítulo 12 describe el flujo de trabajo de iteración genérico. Los cuatro siguientes describen cada una de las etapas del proceso.

Philippe Krutchen. **“The Rational Unified Process. An Introduction”**. Addison Wesley, 2000.

Walter Royce. **Software Project Management**. Addison Wesley, 1998

Pierre N. Robillard, Philippe Kruchten, Patrick d'Astous, **Software Engineering Processes: With the UPEDU**. Addison Wesley, 2002

Eclipse Process Framework Project (EPF). <http://www.eclipse.org/epf/general/description.php> página de acceso al proyecto sobre el marco de proceso de ECLIPSE. En él se accede a OpenUP.

<http://www.dsic.upv.es/assignaturas/facultad/lsi/ejemplorup/> En esta página hay un ejemplo, realizado en un trabajo práctico, sobre desarrollo de una aplicación utilizando RUP.