

Modelando el Ciclo de Vida de Desarrollo del Software

Módulo 1: Introducción al RUP

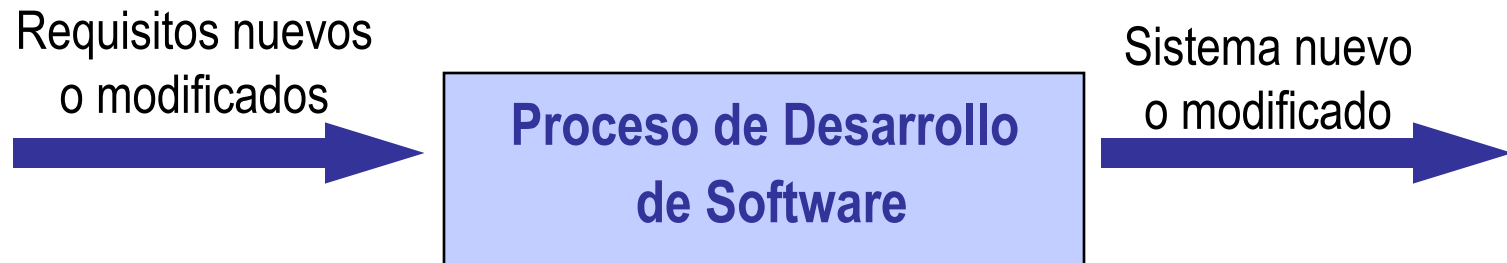


Tópicos

- El Proceso de Desarrollo del Software
- ¿Qué es RUP?
- Conceptos Generales
- ¿Por qué RUP?

¿Qué es un Proceso de Desarrollo del SW?

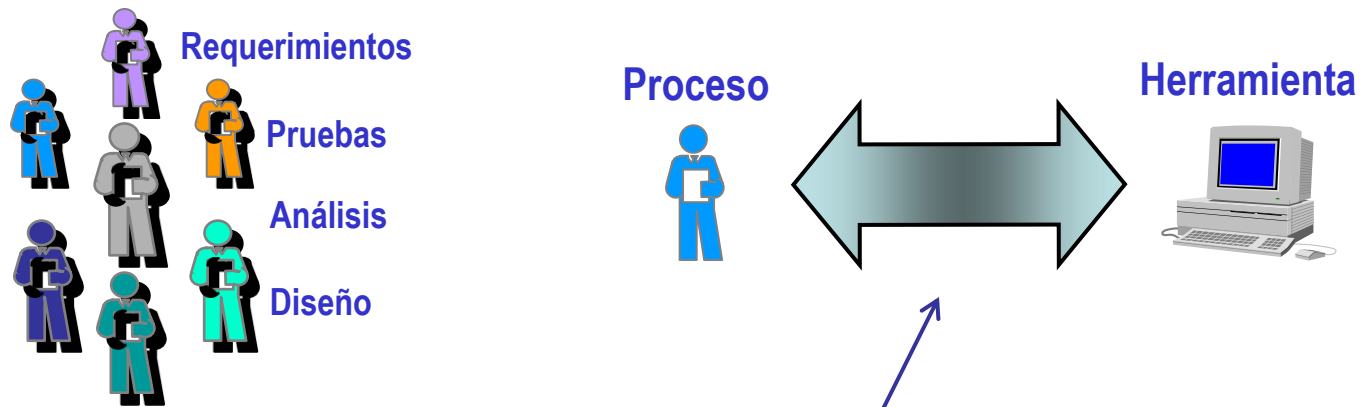
- Define **Quién** debe hacer **Qué**, **Cuándo** y **Cómo** debe hacerlo.



- El objetivo es construir un producto de software ó mejorar alguno existente.

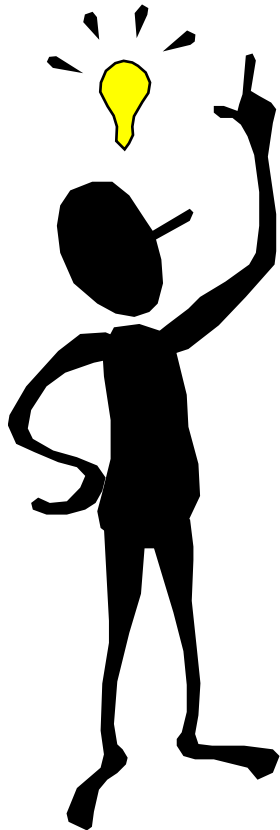
¿Por qué es necesaria una Metodología para el proceso de Desarrollo del SW?

- Para evitar que equipos funcionales diferentes utilicen procesos y lenguajes de modelamiento inconsistentes.



Los procesos deben estar apropiadamente relacionados con herramientas, ó estar propiamente automatizados.

¿Qué es RUP?



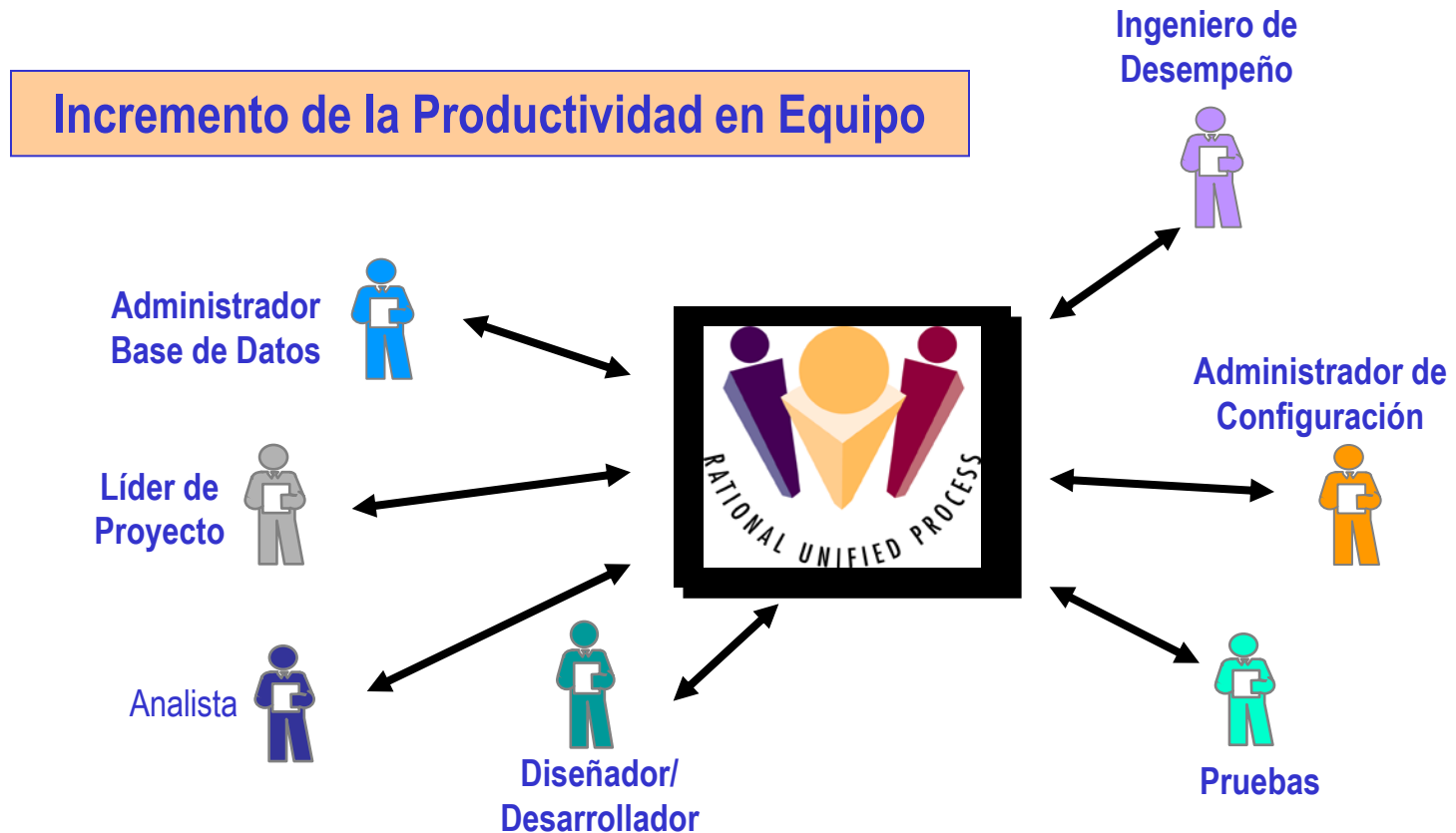
- Es un **proceso de desarrollo de software** y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.
- Sus siglas provienen del inglés ***Rational Unified Process***.
- ***RUP*** provee una forma disciplinada de asignar tareas y responsabilidades (quién hace qué, cuándo y cómo).

Rational Unified Process - RUP

- RUP provee a cada uno de los miembros del equipo de desarrollo un fácil acceso a una **base de conocimiento con lineamientos y guías sobre qué herramientas usar para todas las actividades críticas del desarrollo.**

Objetivos del RUP

- Asegurar la producción de software de calidad dentro de plazos y presupuestos predecibles.

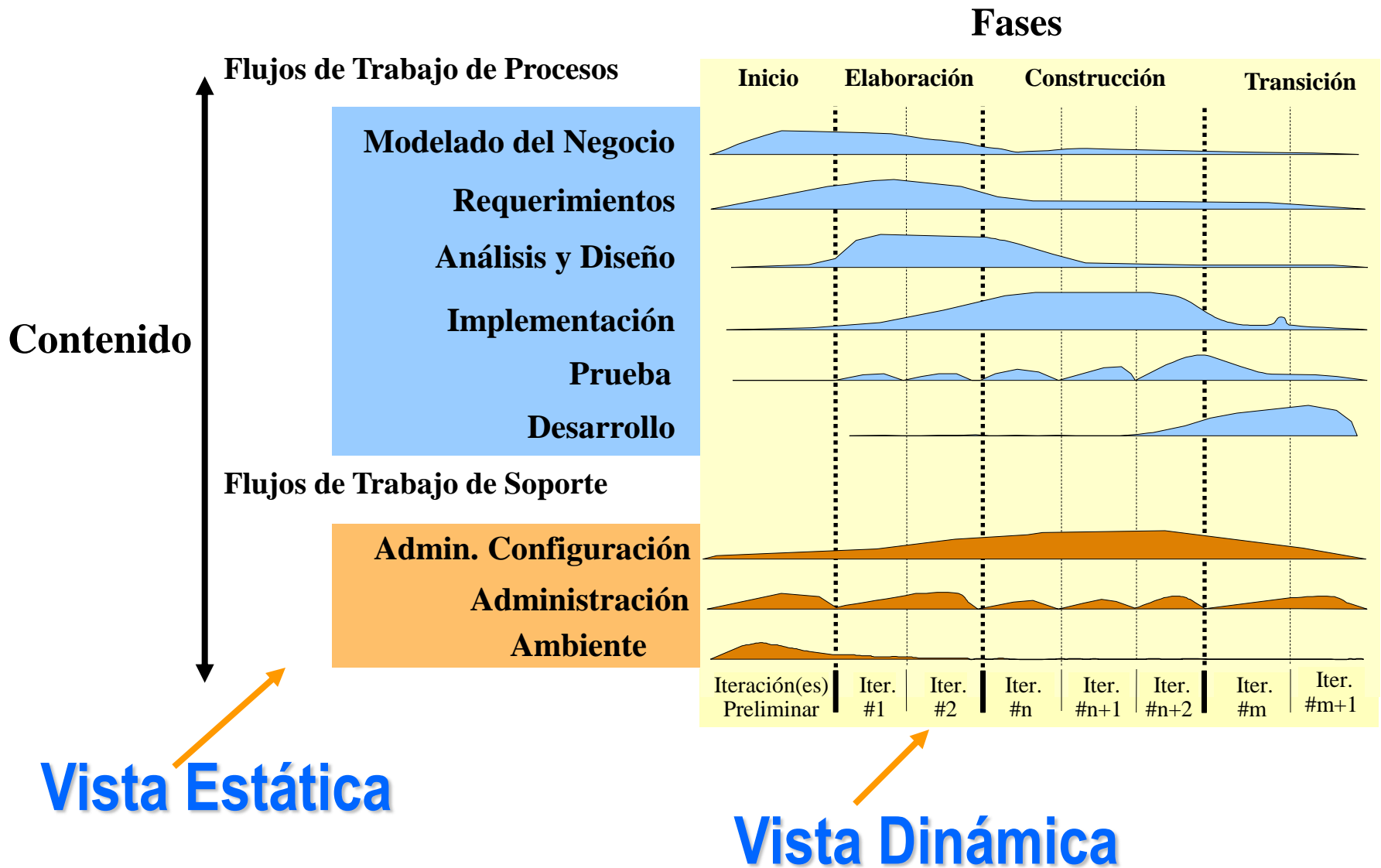


Conceptos Generales

Estructura del RUP

- El RUP puede ser descrito en dos vistas:
 - Vista Estática: Describe el proceso en términos de **flujos de trabajo, actividades, artefactos y trabajadores**.
 - Vista Dinámica: Indica las características del ciclo de vida del proceso expresado en términos de **fases, iteraciones e hitos**.

Estructura del RUP

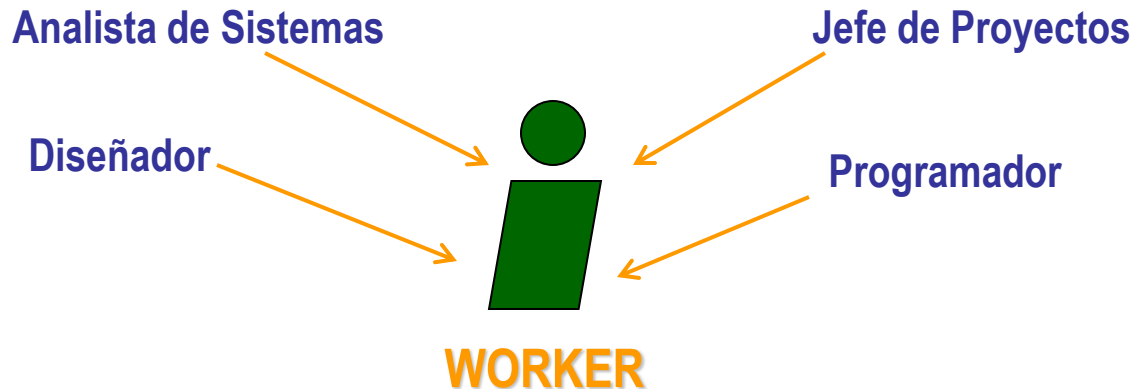


Conceptos Generales

Vista Estática

- Trabajador

Define el comportamiento y responsabilidades de un individuo o un grupo de individuos trabajando juntos como equipo. También se conoce como **Rol**.



Conceptos Generales

Vista Estática

- **Actividades**

RUP determina el trabajo de cada rol a través de actividades. Cada actividad del proyecto debe tener un propósito claro, y se asigna a un trabajador específico.

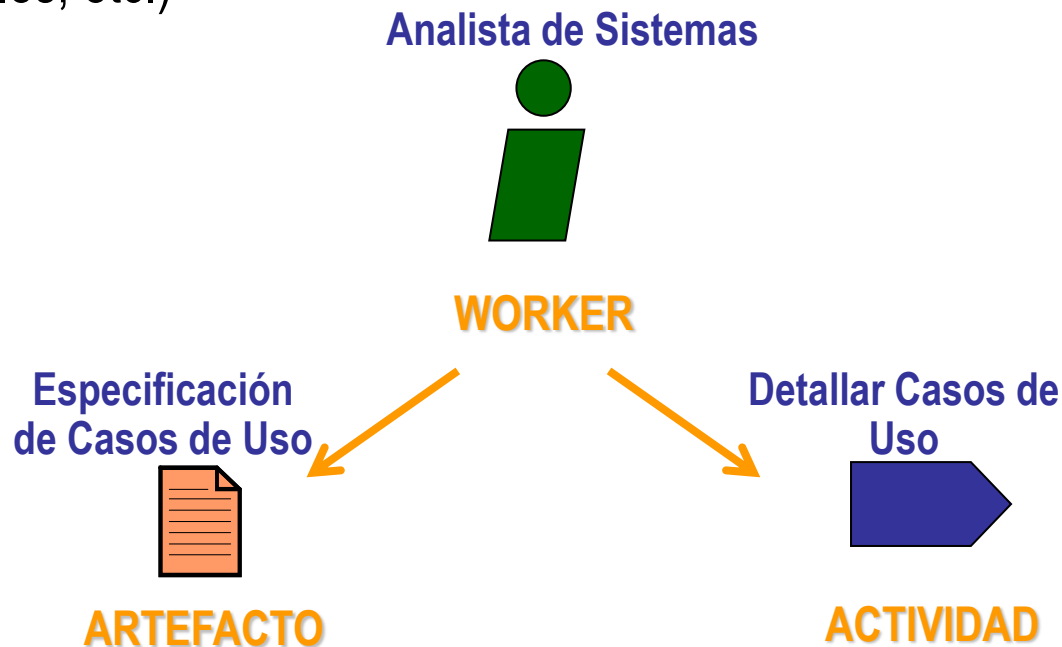


Conceptos Generales

Vista Estática

- **Artefactos**

Son productos tangibles del proyecto. Las cosas que el proyecto produce o usa para componer el producto final (modelos, documentos, código, ejecutables, etc.)



Conceptos Generales

Vista Estática

- **Flujos de Trabajo**

RUP, además, define nueve actividades a realizar en cada fase del proyecto:

- **Flujos de Trabajo Primarios**

- Modelamiento de Negocios
- Requerimientos
- Análisis y Diseño
- Implementación
- Pruebas
- Despliegue

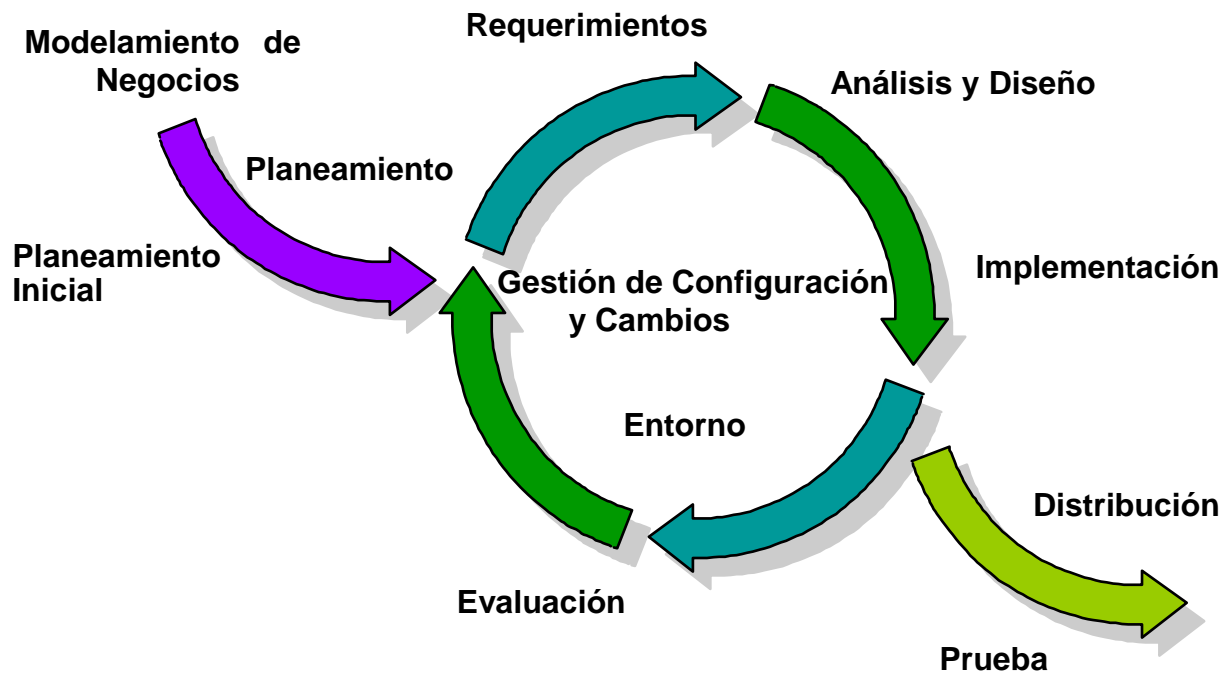
- **Flujos de Trabajo de Apoyo**

- Entorno
- Gestión del Proyecto
- Gestión de Configuración y Cambios

Conceptos Generales

Vista Estática

- Flujos de Trabajo



Este proceso define roles que se asignan entre los miembros del equipo del proyecto, asignando las tareas y el 'artefacto' resultante de cada uno.

Vista Dinámica

- En su visión dinámica, la visión de la estructura del ciclo de vida RUP se basa en un desarrollo iterativo, por hitos bien definidos para revisar el avance y planear la continuidad o posibles cambios.

Vista Dinámica

- Cuatro son las fases que dividen el ciclo de vida de un proyecto RUP:

1. Inicio

Es la fase de la idea, de la visión inicial de producto, su alcance. El esbozo de una arquitectura posible y las primeras estimaciones. Concluye con el “hito de objetivo”.

2. Elaboración

Comprende la planificación de las actividades y del equipo necesario. La especificación de las necesidades y el diseño de la arquitectura. Termina con el “hito de Arquitectura”.

Vista Dinámica

3. Construcción

Desarrollo del producto hasta que se encuentra disponible para su entrega a los usuarios. Termina con el “hito del inicio de la capacidad operativa”.

4. Transición

Traspaso del producto a los usuarios. Incluye: manufactura, envío, formación, asistencia y el mantenimiento hasta lograr la satisfacción de los usuarios. Termina con el “hito de entrega del producto”.

Las 6 Mejores Prácticas

- RUP integra un conjunto de “buenas prácticas” para el desarrollo de software.

RUP: 6 Mejores Prácticas

- ✓ **Desarrollo iterativo.**
- ✓ **Administración de Requerimientos.**
- ✓ **Uso de arquitecturas basadas en componentes.**
- ✓ **Modelamiento Visual.**
- ✓ **Verificación continua de la calidad.**
- ✓ **Gestión y control de cambios.**

¿Por qué RUP?

- La complejidad de los proyectos de Sw hoy en día, el constante cambio de requerimientos y la falta de una documentación durante el proceso de desarrollo provoca que los proyectos se retrasen en tiempo y se incrementen en costo.
- La solución a esta problemática es implantar una metodología de desarrollo que permita hacer seguimiento a los proyectos desde su etapa de requerimientos, hasta su implantación.
- **RUP ofrece un Proceso Unificado** para el desarrollo de los proyectos de Sw, desde la etapa de Ingeniería de Requerimientos hasta la etapa de pruebas.



Resumen

- RUP provee a cada uno de los miembros del equipo de desarrollo un fácil acceso a una base de conocimiento con las guías necesarias para todas las actividades críticas del desarrollo.
- Estructura del RUP:
 - Vista Dinámica: *fases, iteraciones e hitos.*
 - Vista Estática: *flujos de trabajo, actividades, artefactos y trabajadores.*
- El RUP esta basado en las 6 Mejores Prácticas para el desarrollo de Sw:
 - Desarrollo iterativo.
 - Administración de Requerimientos.
 - Uso de arquitecturas basadas en componentes.
 - Modelamiento Visual.
 - Verificación continua de la calidad.
 - Gestión y control de cambios.

