

# **El Proceso Unificado de Desarrollo**

Unified Process

# Componentes de un Método

## Elementos de modelado:

- Conjunto fundamental de conceptos de modelado para capturar el conocimiento semántico sobre un problema y su solución.

## Notación:

- Conjunto de vistas y notaciones para presentar la información de modelado subyacente que permite a las personas examinarlos y modificarlos.



# Componentes de un Método

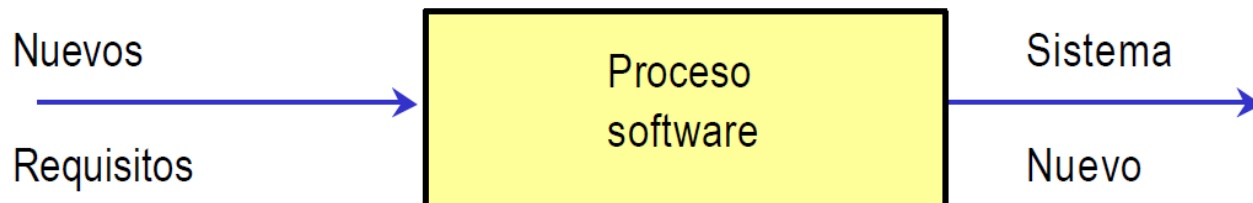
- Proceso:
  - Tiene como cometido la formalización de las actividades relacionadas con la elaboración de sistemas software.
- Experiencia
  - Colección de reglas y heurísticas para llevar a cabo el desarrollo.

# UML no es un método



# ¿Qué es un Proceso?

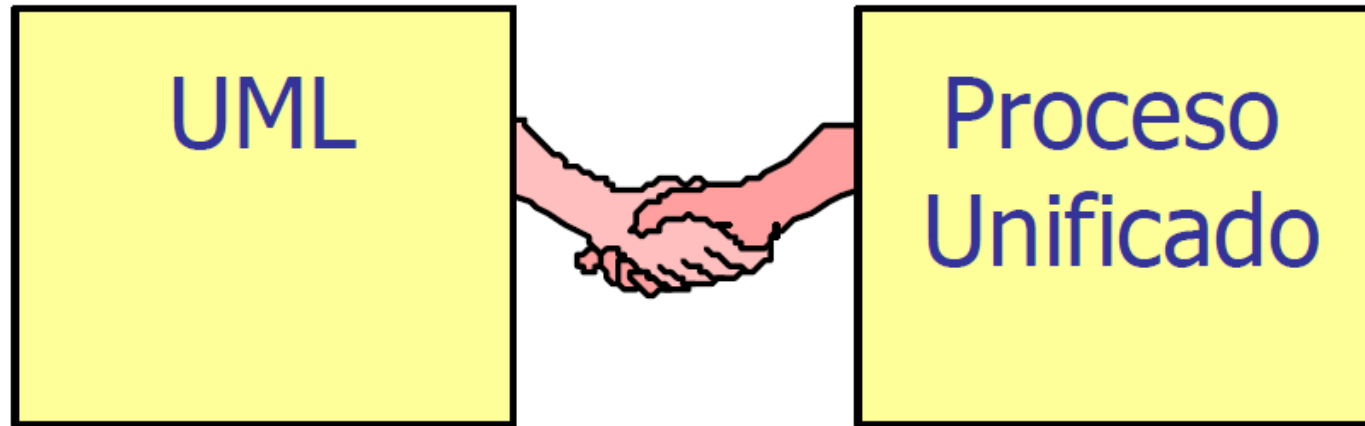
- Describe un conjunto de actividades que deben realizarse en un determinado orden:
  - Define **quien** está haciendo, **qué**, **cuándo**, y **cómo** alcanzar determinado **objetivo**.
- Debe ser:
  - Reproducible
  - Definido
  - Medible en cuanto a rendimiento
  - Optimizable....



# ¿Qué es un Proceso?

- Debe:
  - capturar las mejores prácticas
  - para reducir el riesgo
  - y hacer el proyecto más predecible
  - al mismo tiempo lograr una visión y cultura comunes
  - logrando que cada interesado comprenda su papel en el desarrollo

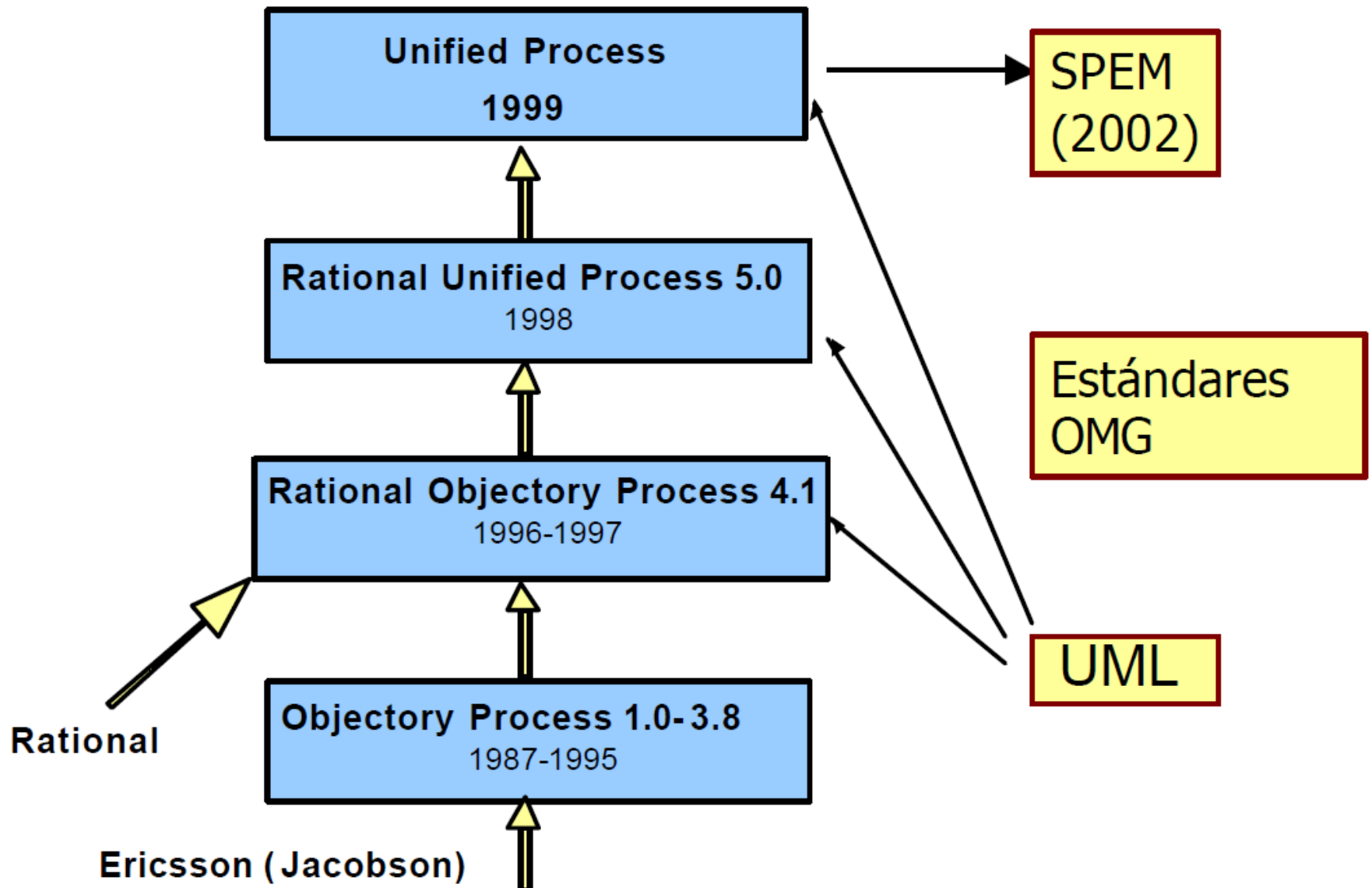
# Dos elementos complementarios



- **Estándar  
OMG**

- **Proceso  
marco  
adaptable**
- **Estándar en  
fase de  
propuesta**

# Antecedentes del Proceso Unificado





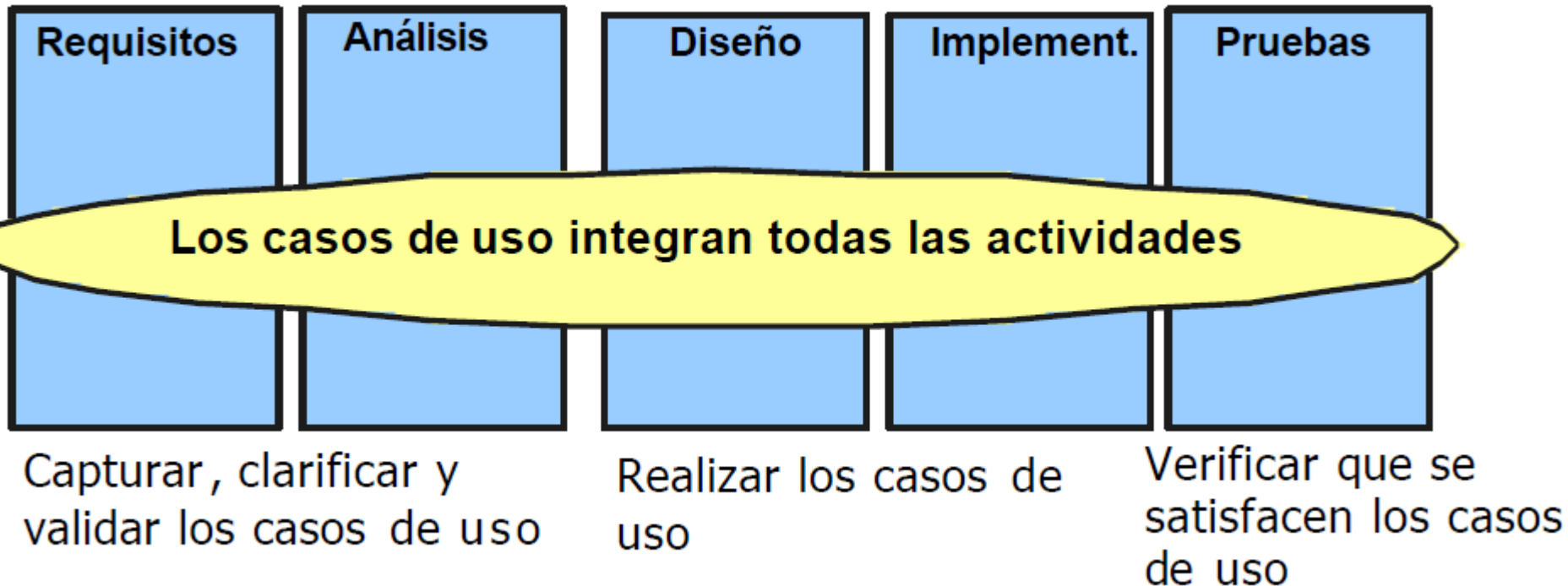
# UP es un Proceso “marco”

- No existe un proceso Universal.
- UP es flexible y extensible:
  - Permite varias estrategias de desarrollo.
  - Se pueden definir diferentes conjuntos de productos.
  - Se pueden definir actividades y encargados de las mismas.

# Características del Proceso Unificado

- Está dirigido por los casos de uso:
  - Desde la especificación hasta el mantenimiento.
- Se centra en la arquitectura:
  - La arquitectura es prioritaria desde el principio hasta el final.
  - Se facilita el refinamiento progresivo de la arquitectura.
- Iterativo e incremental:
  - El trabajo se divide en iteraciones pequeñas en función de la importancia de los casos de uso y el análisis de riesgos.

# Conducido por Casos de uso

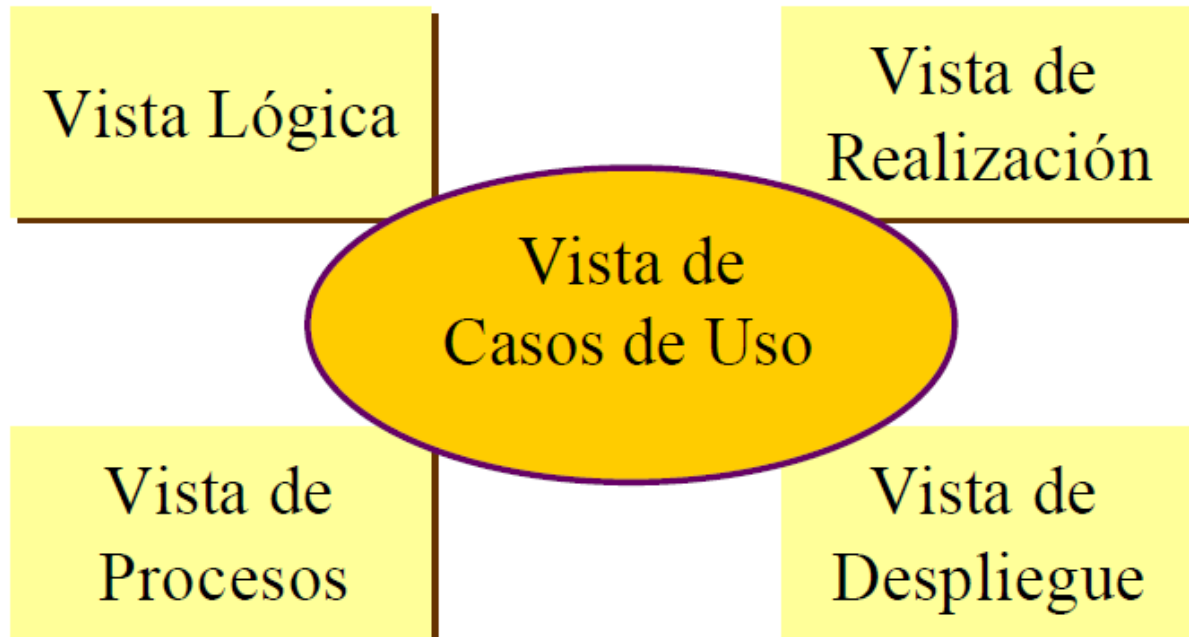


# Centrado en la Arquitectura

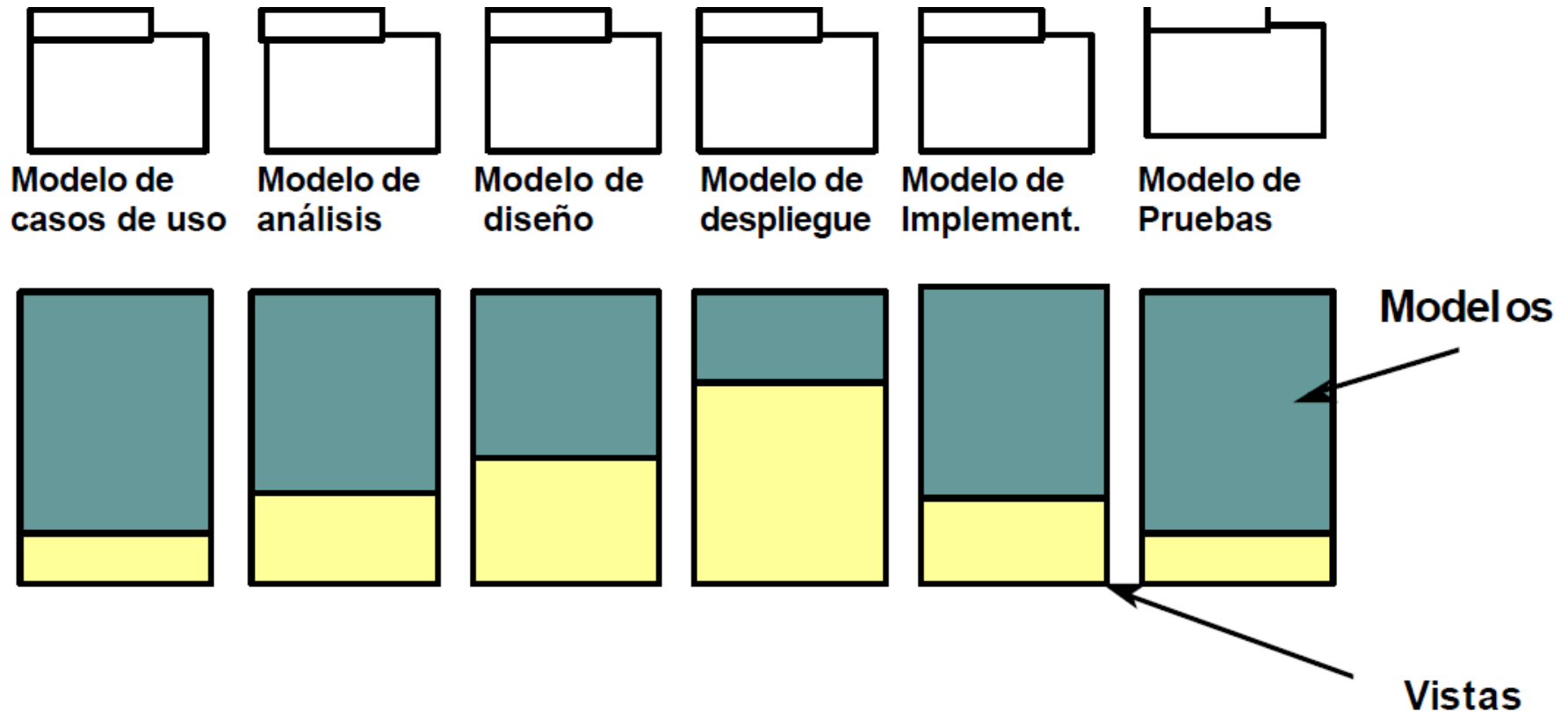
- La arquitectura describe los elementos fundamentales del sistema:
  - Subsistemas
  - Dependencias
  - Interfaces
  - Colaboraciones
  - Nodos
  - Clases activas...
- Incluye decisiones importantes sobre:
  - Organización del sistema.
  - Elementos estructurales, interfaces y su comportamiento.
  - Composición y comportamiento de los subsistemas.
  - El estilo de la arquitectura que guía esta organización.

# Modelo de Arquitectura: 4 + 1 vistas

- Los modelos son instrumentos para visualizar, especificar, construir y documentar la arquitectura del sistema.
- Cada vista es una parte de un modelo.



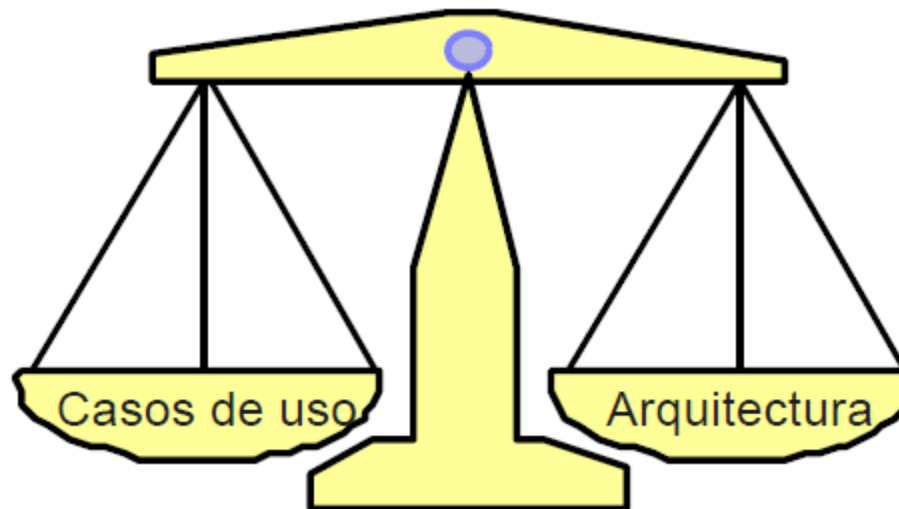
# Arquitectura y Modelos



La arquitectura incorpora una colección de vistas de los modelos

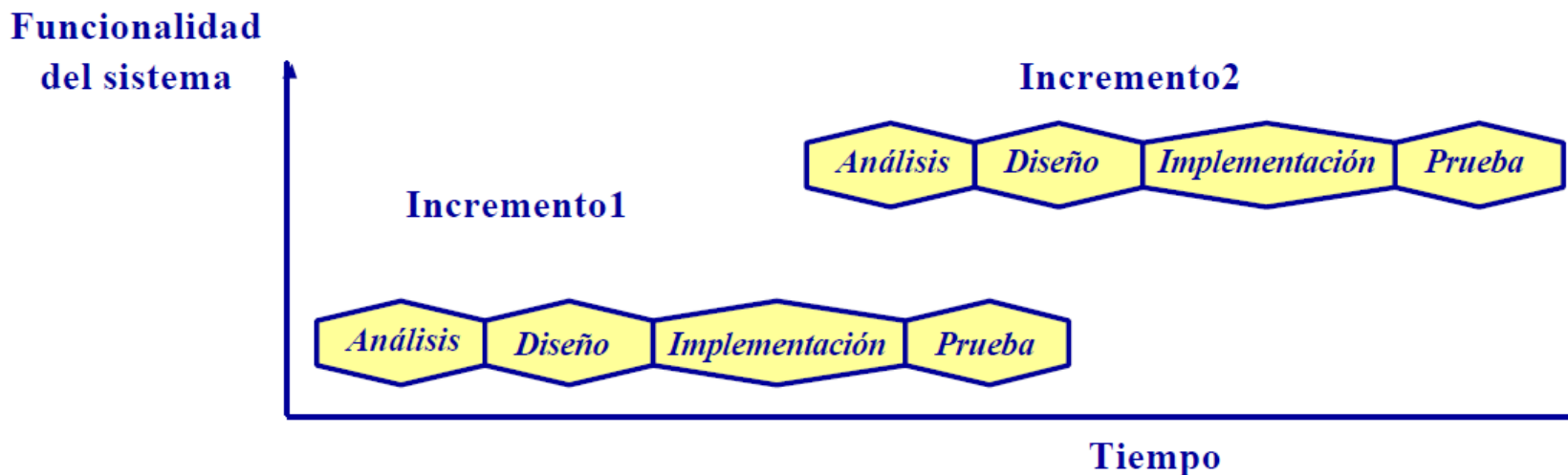
# Estructura y función

- Los casos de uso especifican las funciones
- La arquitectura especifica la estructura
- Los casos de uso y la arquitectura deben estar en equilibrio.



# Proceso iterativo e incremental

- ...Pero la característica fundamental de UP:
  - Es un proceso iterativo.
  - Se basa en la ampliación y el refinamiento del sistema.
  - Una serie de desarrollos cortos (mini proyectos de 2 a 6 semanas, cada iteración reproduce el ciclo de vida a menor escala).
  - No sólo se mejora sino que el sistema también crece:  
Proceso iterativo e incremental.

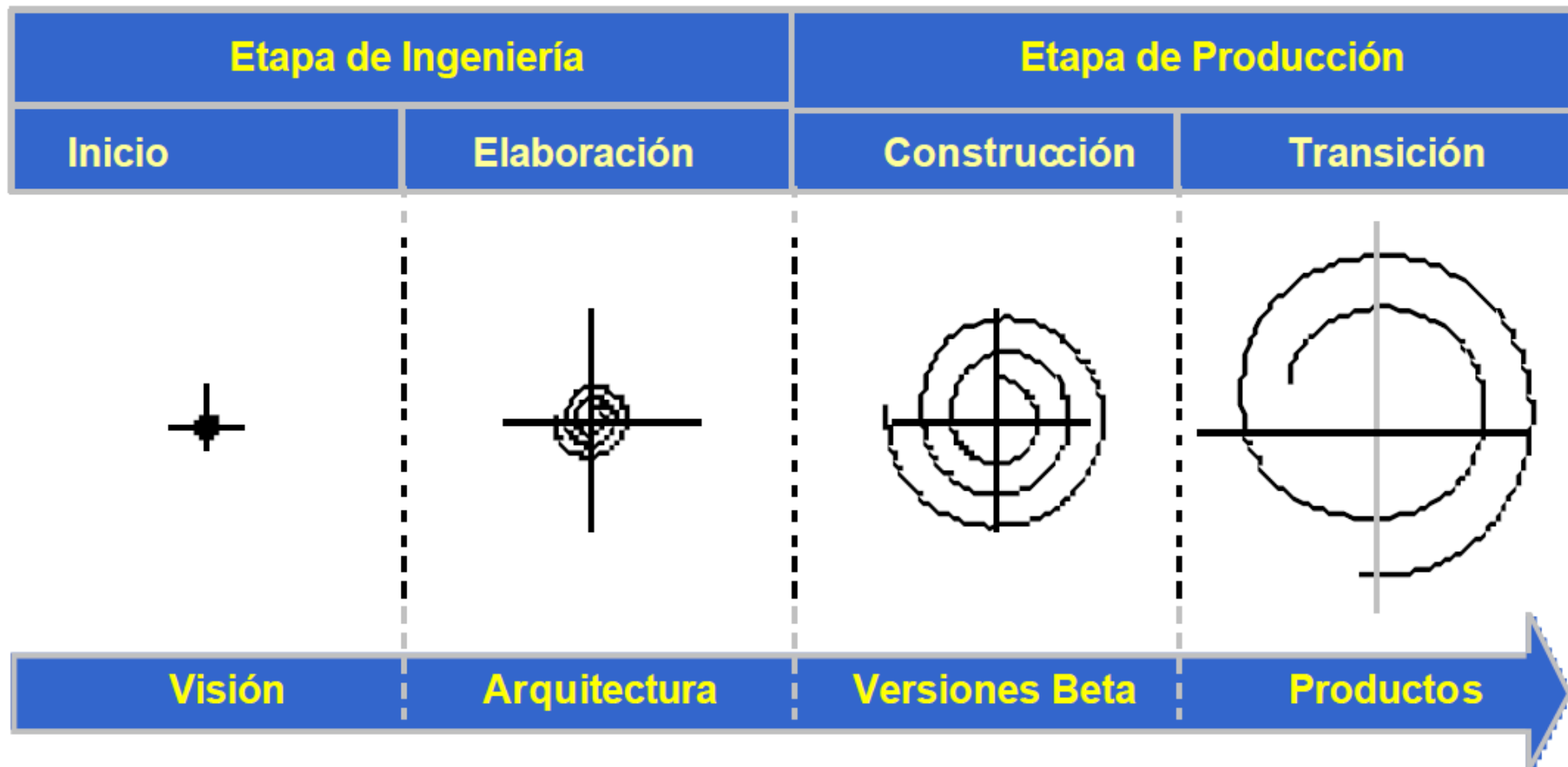




# Proceso iterativo e incremental

- El resultado de cada iteración es un sistema ejecutable (aunque sea incompleto y no esté listo para su instalación)
- Un sistema instalable requiere varias iteraciones:
  - Evolución de prototipos ejecutables
  - Los objetivos de una iteración se establecen en función de la evaluación de las iteraciones precedentes
  - Concepto de “time-boxing”: cada iteración debe tener una duración fija (el máximo, 6 meses)
    - En lugar de retrasar el final de una iteración se recomienda eliminar algunos de los requisitos (se dejan para la siguiente iteración)
  - La realimentación del usuario es fundamental en este proceso
  - El progreso es visible

# Iterativo: varias espirales

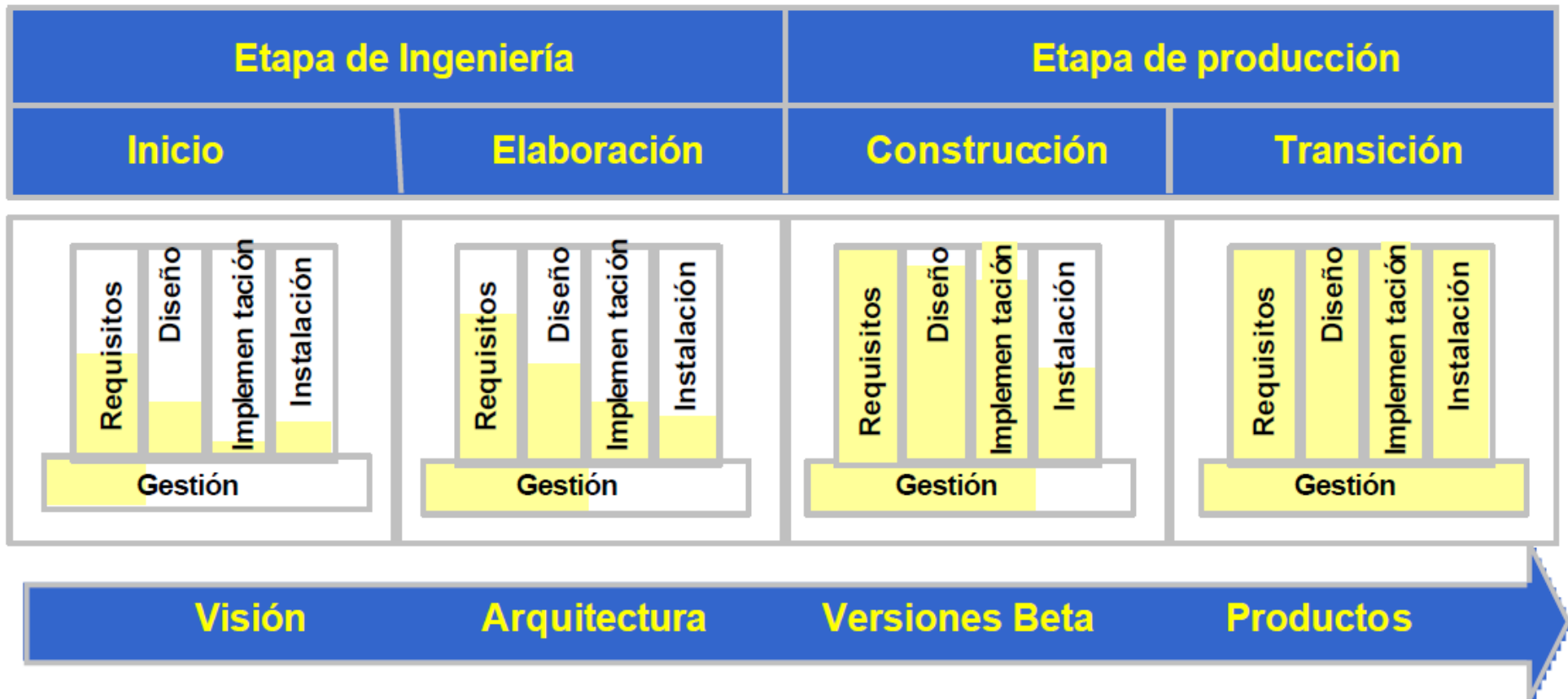


# Cada iteración comprende:

- Planificación de la iteración (estudio de riesgos).
- Análisis de Casos de uso y escenarios.
- Diseño de opciones arquitectónicas.
- Codificación y pruebas. La integración del nuevo código con el existente de iteraciones anteriores se hace gradualmente durante la construcción.
- Evaluación de la entrega ejecutable (evaluación del prototipo en función de las pruebas y de los criterios definidos).
- Preparación de la entrega (documentación e instalación del prototipo).

# Incremental

- Primero, la arquitectura,
- Después, se van añadiendo los detalles según avanza el desarrollo.



# Gestión del riesgo

- El análisis de riesgos consiste en evaluar el proyecto, la tecnología y los recursos con el fin determinar y comprender la naturaleza y el origen de los riesgos.
- Posibles riesgos:
  - Comerciales (competencia, etc.)
    - Financieros (económicos, etc.)
    - Técnicos (¿base tecnológica sólida y probada?)
    - De desarrollo (¿equipo experimentado?)
  - Cada iteración se centra en los riesgos más importantes.

# Las cuatro “P”

- Personas
  - Todos los interesados
- Proyecto
  - Elemento organizativo a través del cual se gestiona el desarrollo del software.
- Producto
  - Artefactos que se crean durante la vida del proyecto.
- Proceso
  - Definición del conjunto completo de actividades necesarias para transformar requisitos en un producto.

# Fases y disciplinas

- Fases y puntos de control.
- Disciplinas (Flujos de trabajo)
- Artefactos

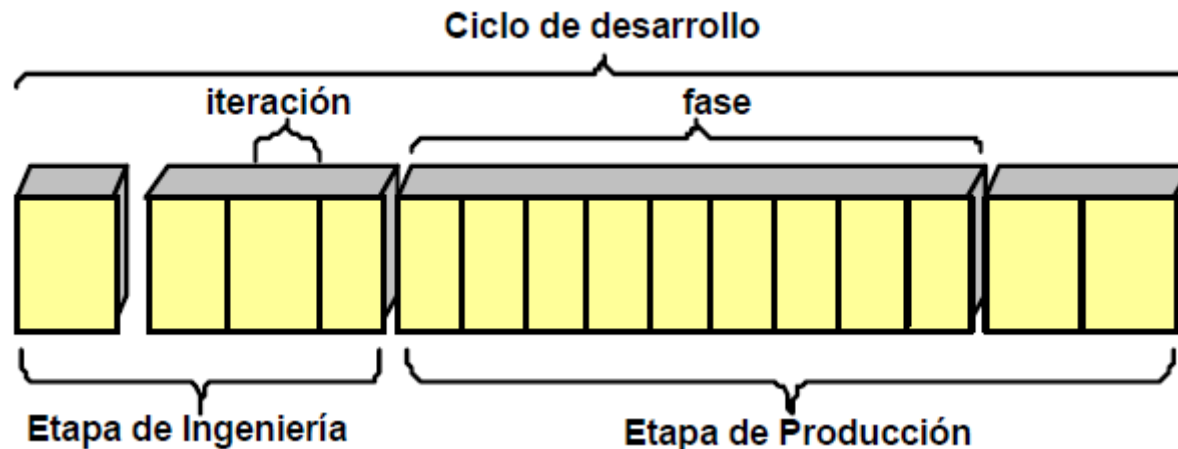
# Elementos del Proceso Unificado

- Fases:
  - Es preciso diferenciar temporalmente las fases del ciclo de vida
  - La división temporal necesita...
- Puntos de control o hitos:
  - Separan las etapas, las fases, las iteraciones
- Disciplinas o Flujos de trabajo:
  - Organizan las actividades fundamentales de gestión y desarrollo
  - Se pueden solapar en el tiempo.
  - El resultado de las actividades de los flujos de trabajo son...
- Artefactos:
  - Cualquier tipo de información producida por los desarrolladores de un sistema (diagramas UML, código, ejecutables, casos de prueba...)
  - Se construyen de forma incremental



# Planificación temporal del proyecto

- UP propone una serie de ciclos de desarrollo:
  - Hay que separar claramente la etapa de Ingeniería de la etapa de Producción.
  - Cada una de las dos grandes etapas se dividen en fases.
  - Las fases se dividen en iteraciones.



# Etapas y fases del ciclo de vida

- Etapa de Ingeniería: equipos pequeños, actividades poco predecibles (análisis, viabilidad, planificación). Las fases son:
  - Inicio
  - Elaboración
- Etapa de Producción: equipos grandes, actividades predecibles, menos riesgos (programación, pruebas). Las fases son:
  - Construcción
  - Transición.



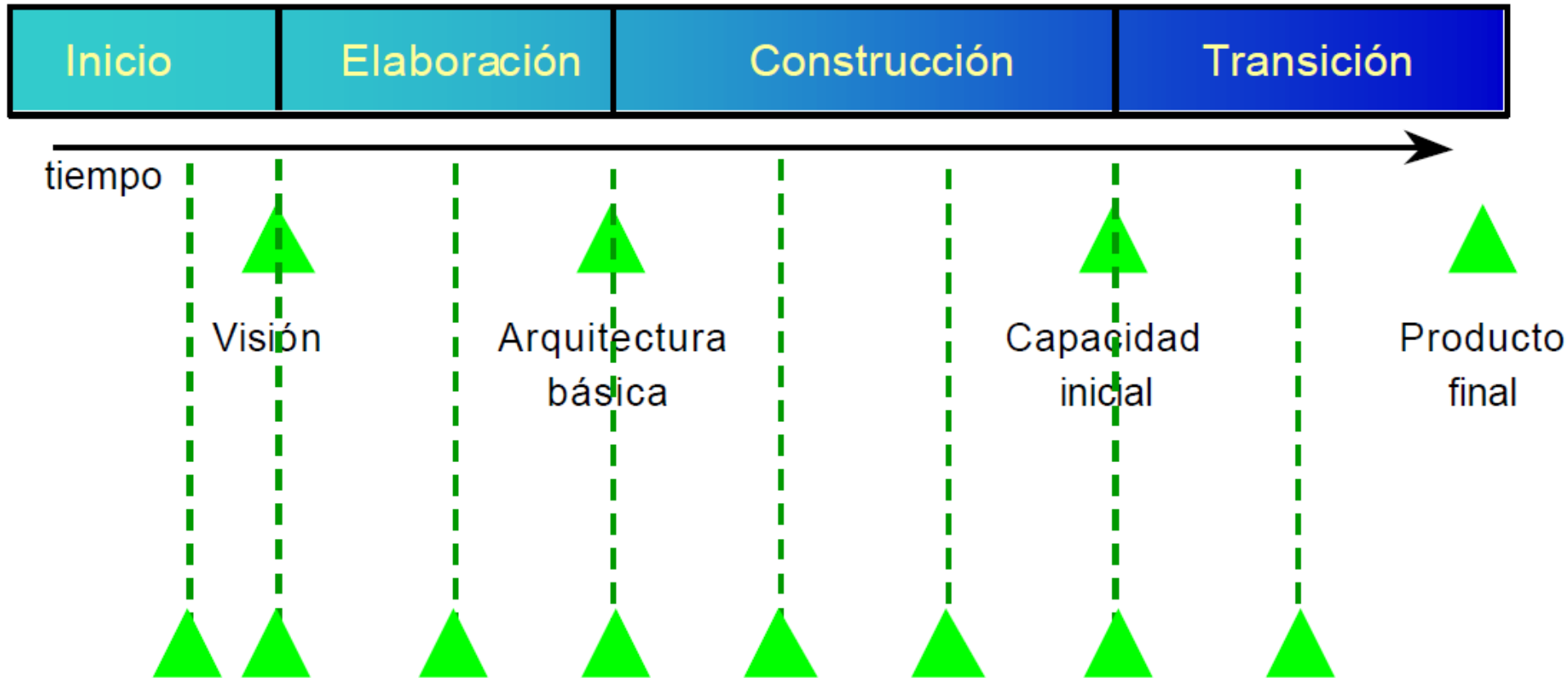
# Objetivos de las fases

- Inicio del proyecto (inception)
  - Define el ámbito y objetivos del proyecto
- Elaboración
  - Define la funcionalidad y una arquitectura básica
- Construcción
  - El producto se desarrolla a través de iteraciones
- Transición
  - Se libera el producto y se entrega al usuario para su uso real

# Hitos

- Los hitos son puntos de control en los cuales los participantes en el proyecto revisan el progreso del proyecto.
- Se pretende:
  - Sincronizar las expectativas y la realidad
  - Identificar los riesgos
  - Se evalúa la situación global del proyecto
- Se necesitan:
  - Resultados tangibles para comparar con las expectativas
- Varios niveles:
  - Hitos principales al final de cada fase
  - Hitos secundarios final de cada iteración

# Hitos principales y secundarios



Una iteración es una secuencia de actividades con un plan establecido y unos criterios de evaluación, cuyo resultado es una versión ejecutable (hito secundario)

# Disciplinas o flujos de trabajo

- Organizan las actividades fundamentales de gestión y desarrollo del proyecto
  - Disciplinas de desarrollo: requisitos, análisis, diseño, implementación, pruebas, etc.
  - Disciplinas de gestión o soporte: gestión de proyecto, gestión de configuraciones, entorno, evaluación, etc.
- Al contrario de lo que ocurre con las fases, las distintas actividades del equipo de desarrollo se pueden solapar en el tiempo.

# Fases, iteraciones y disciplinas

## Disciplinas:

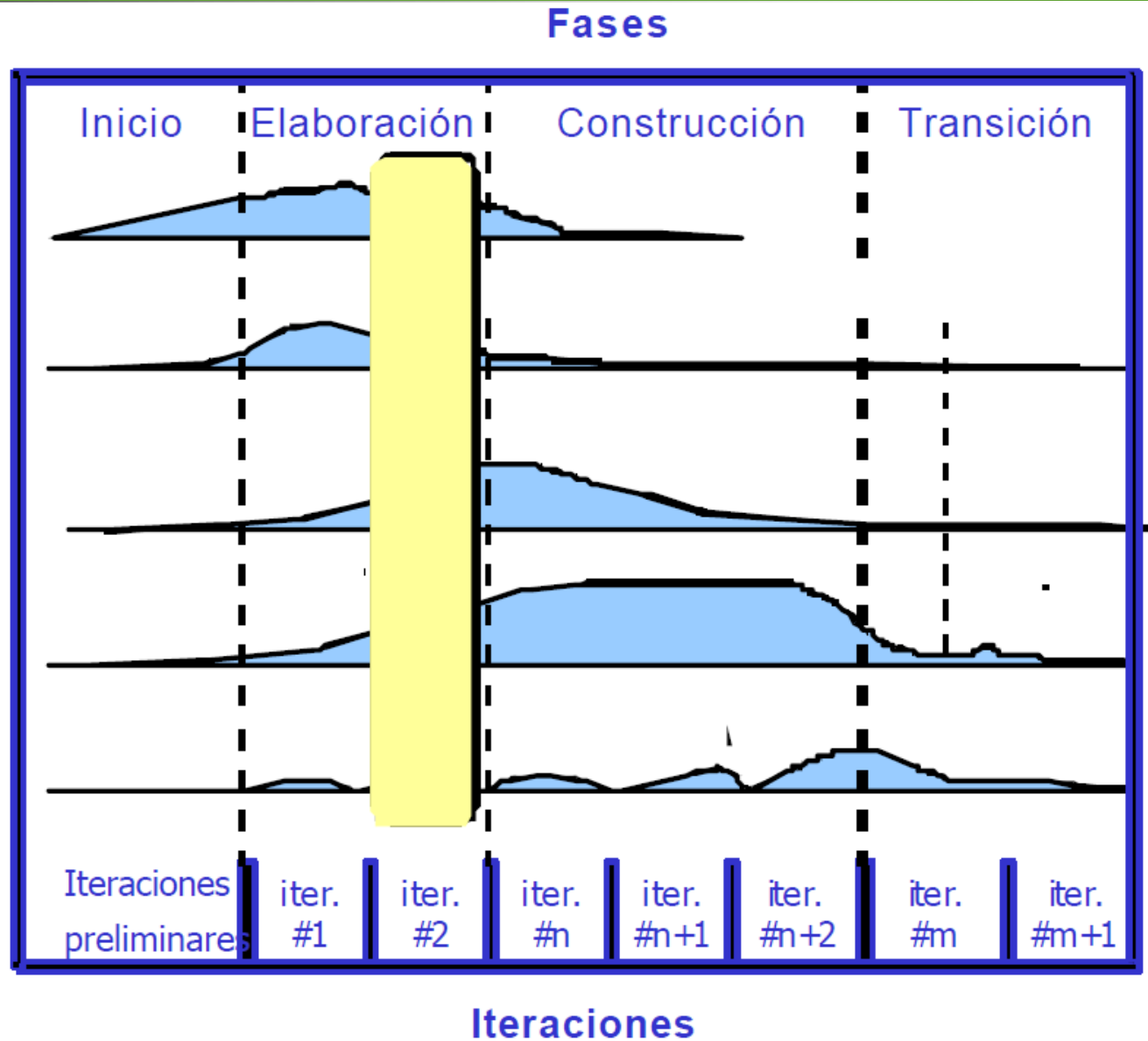
Requisitos

Análisis

Diseño

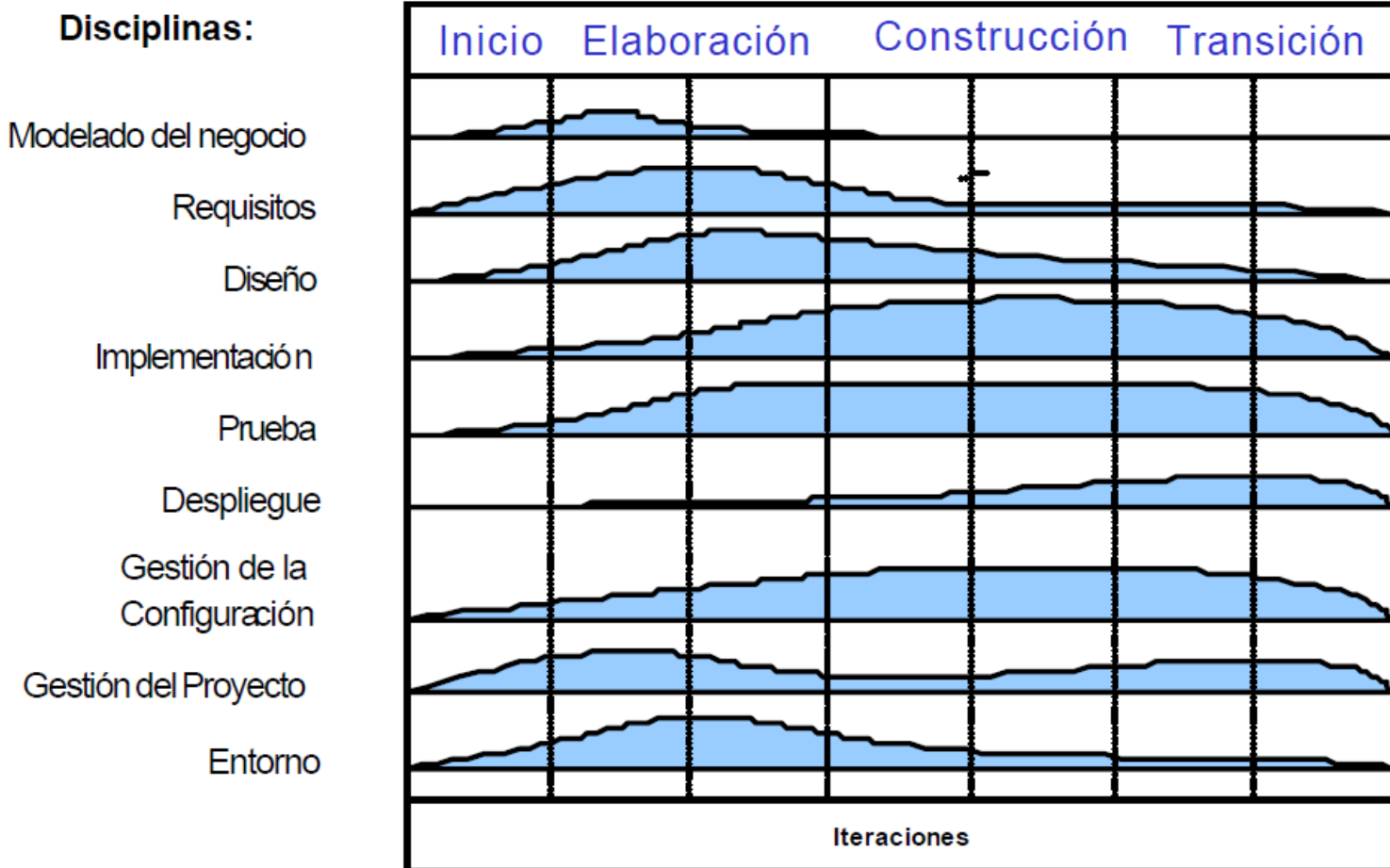
Implementación

Pruebas



# Fases y disciplinas: SPEM

- La propuesta de proceso estándar admite distintas combinaciones de disciplinas y fases.

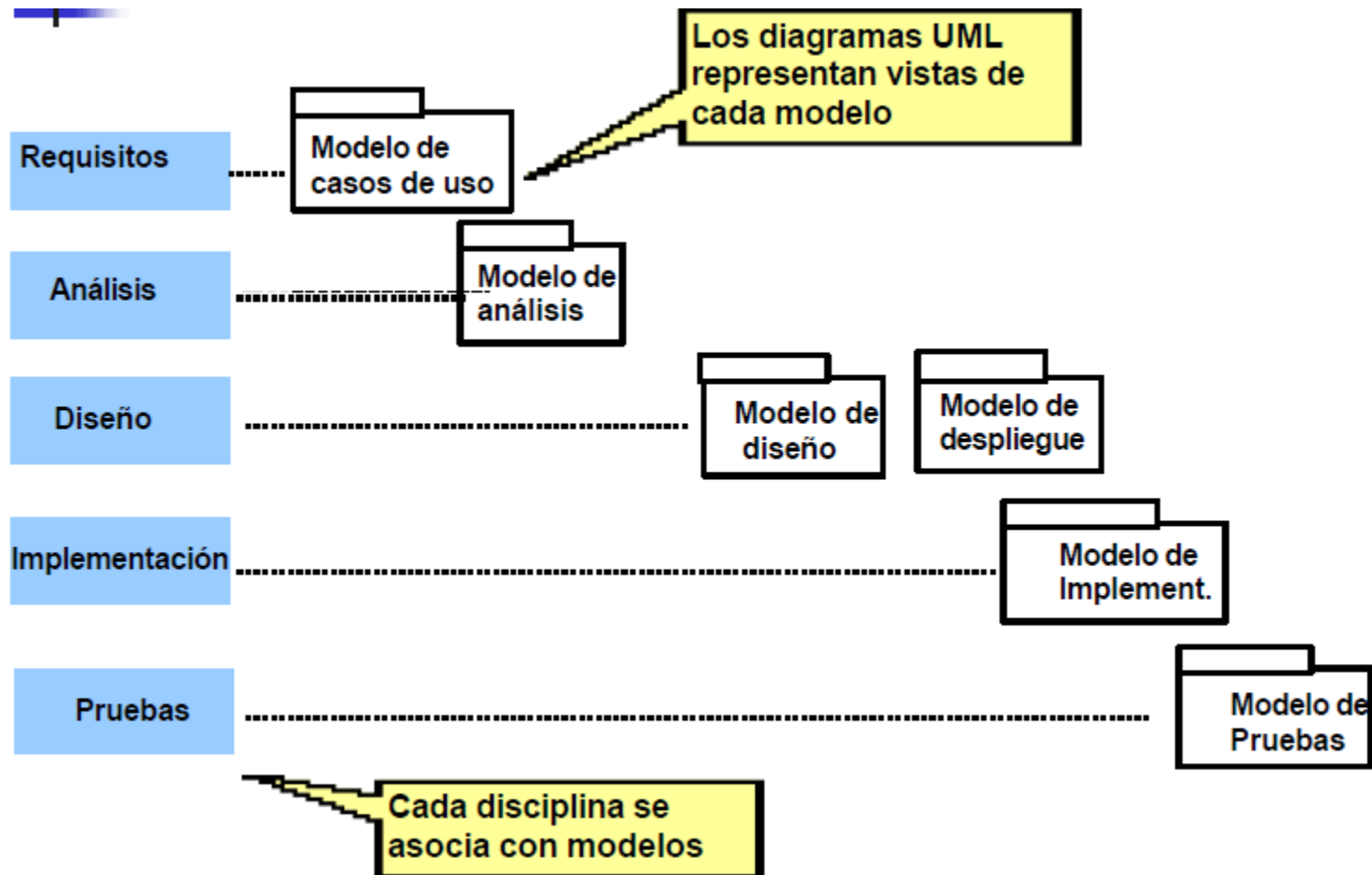




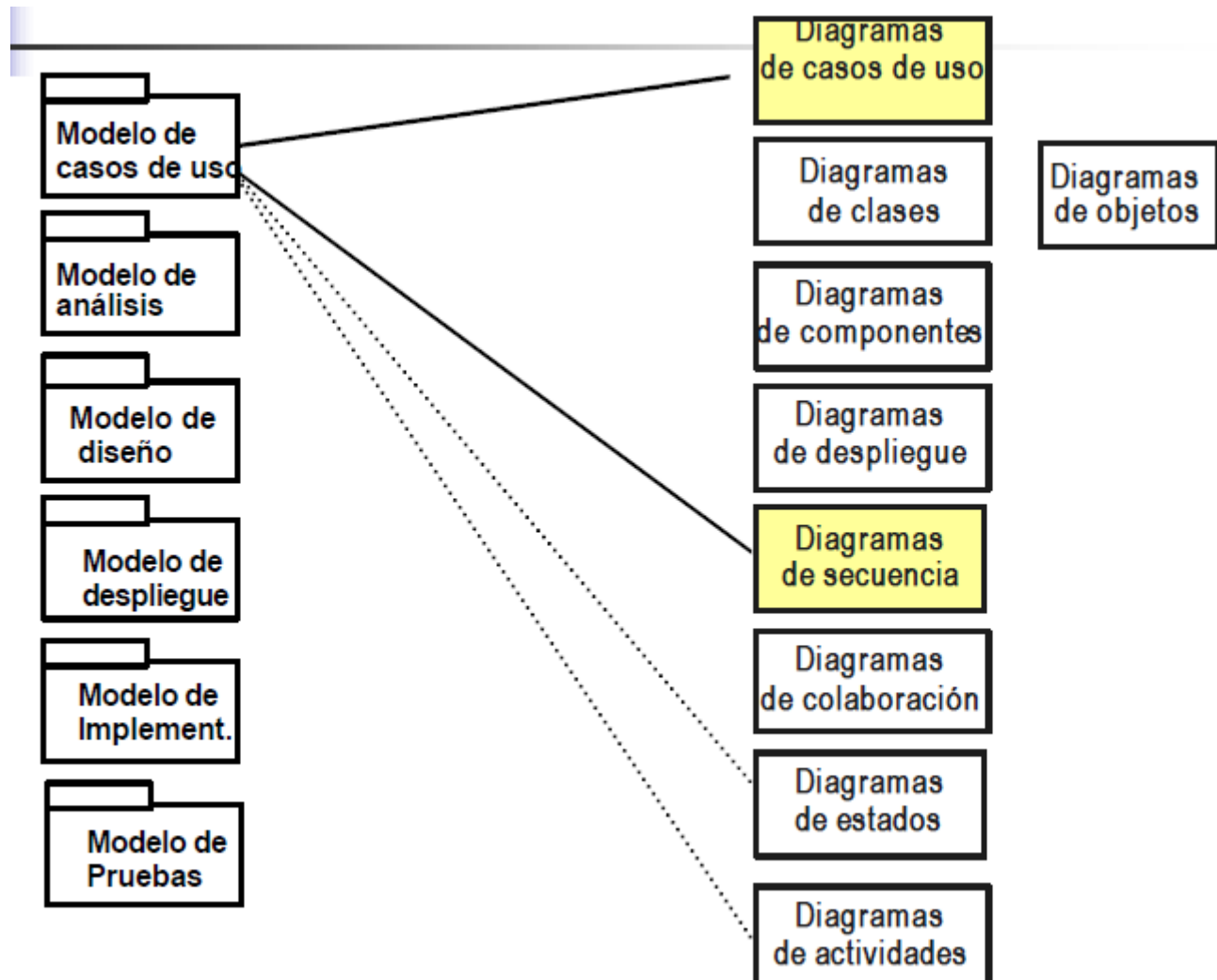
# Artefactos

- Definición de artefacto (o producto):
  - Cualquier tipo de información producida por los desarrolladores de un sistema.
- Se construyen de forma incremental.
- Tipos de artefactos
  - Diagramas UML
  - Código fuente
  - Ejecutables
  - Casos de prueba...
- Los modelos son los artefactos básicos que producen las disciplinas (incluyen otros artefactos).

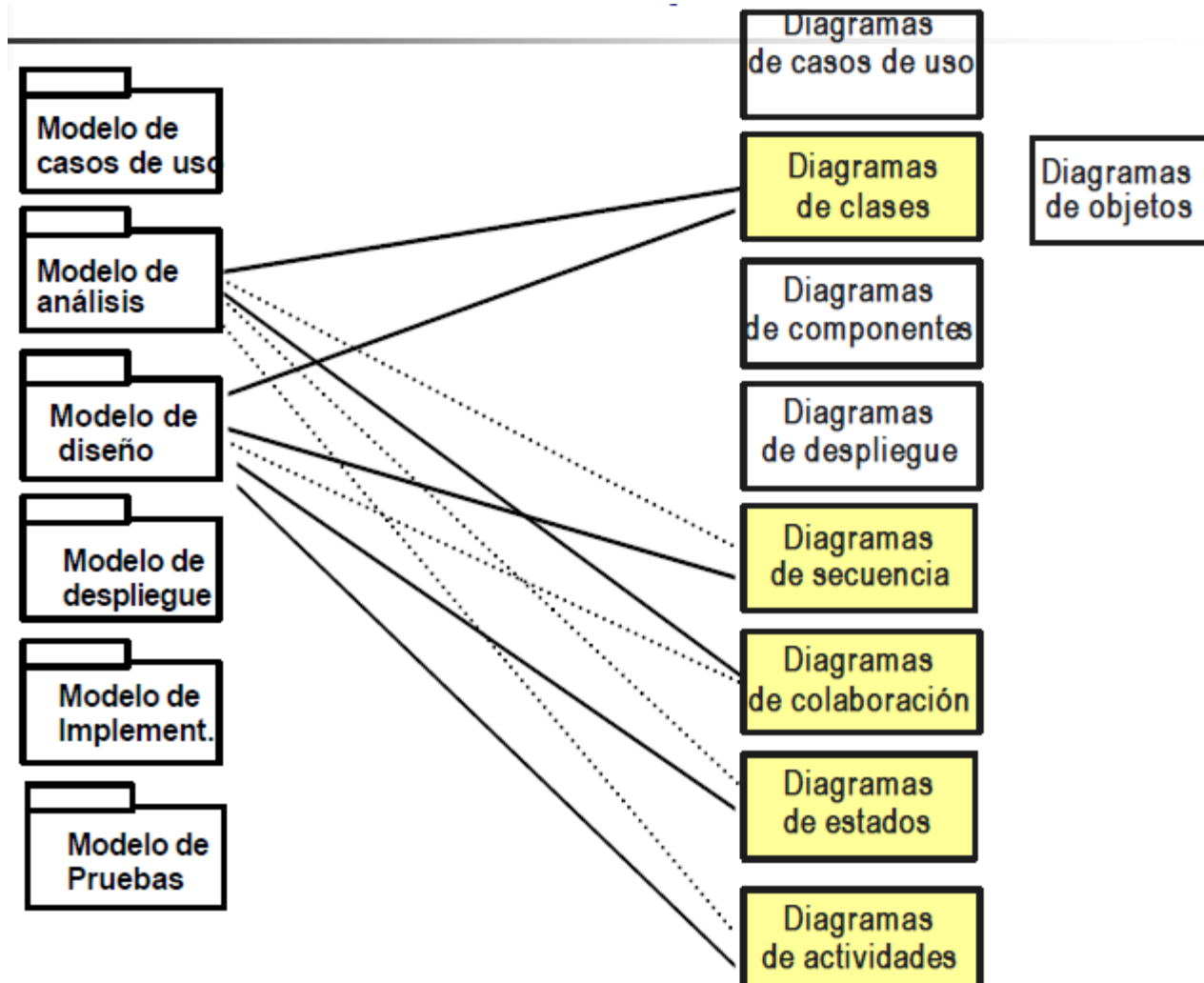
# Disciplinas y modelos principales



# Modelo de casos de uso



# Modelos de análisis y diseño



# El “Caso de desarrollo”

- El número de posibles diagramas, modelos, vistas, ficheros fuente, casos de pruebas, etc. es muy grande
- Es preciso definir los artefactos que son necesarios en cada desarrollo concreto
- Uno de los artefactos iniciales es el “Caso de desarrollo”:
  - Qué artefacto es necesario en cada disciplina
  - En qué fase se crea
  - En qué fases se actualiza
- Esta posibilidad permite tanto desarrollos “pesados” como “ágiles”

# El “Caso de desarrollo”

Disciplina	Artefacto	Inicio	Elabora- ción	Construc- ción	Tran- sición
Requisitos	Modelo de casos de uso Visión Glosario Modelo del dominio	I I I	R R R I		
Análisis	Modelo de análisis		I	R	
Diseño	Modelo de diseño Arquitectura Modelo de datos		I I I	R R R	
Implementación	Modelo de implementación		I	R	R
Pruebas	Modelo de pruebas		I	R	
Gestión del Proyecto	Plan de desarrollo	I	R	R	R
Entorno	Caso de desarrollo	I	R		

# La fase de inicio (Inception)

# La fase de Inicio (Inception)

- Al comenzar un proyecto hay que contestar algunas preguntas:
  - ¿Cuál es la visión del sistema?
  - ¿Es viable?
  - ¿Se puede comprar o hay que fabricar el sistema?
  - ¿Cuánto va a costar?
  - Y, finalmente ¿seguimos adelante o paramos?



# Objetivos de la fase de Inicio

- El objetivo es desarrollar el análisis de negocio hasta el punto necesario para la puesta en marcha del proyecto
- Para ello, es necesario:
  - Delimitar el alcance y objetivos del proyecto
  - Definir la funcionalidad y capacidades del producto
  - Tener una idea de la arquitectura (arquitectura candidata)
  - Reducir los riesgos cuanto antes
  - Hacer estimaciones iniciales de costes, agenda

# Criterios de evaluación de la fase

- Al comienzo de la fase de Inicio, se establecen:
  - Una planificación provisional
  - Los criterios de evaluación de la fase. Al final, tendremos que haber sido capaces de:
    - Fijar el ámbito del sistema
    - Resolver ambigüedades en los requisitos
    - Determinar una arquitectura candidata
    - Mitigar los riesgos críticos
    - Analizar las posibilidades de “negocio” (evaluar el “caso de negocio”)

# Disciplinas en la fase de Inicio

- Requisitos
  - Enumerar los requisitos iniciales (características del sistema)
  - Comprender el contexto del sistema
  - Representar los requisitos como casos de uso
  - Recoger los requisitos no funcionales
- Análisis
  - Análisis de la arquitectura
  - Análisis de los casos de uso (de algunos representativos)
- Diseño
  - Esbozo de la arquitectura
- Implementación
  - ¿Prototipo desechable?
- Pruebas
  - No

# Artefactos de la fase de Inicio

Artefacto	Descripción
Visión Lista de características Especificación adicional Modelo de casos de uso	Grandes objetivos y restricciones  Requisitos no funcionales Describe los requisitos funcionales
Glosario Modelo inicial de dominio	Terminología básica del dominio Define el contexto
Modelo de análisis Modelo de diseño Prototipos (desechables)	Esbozo inicial  Validar la tecnología
Plan de desarrollo Lista de riesgos	Recursos (incluye Plan de la 1ª iteración) Riesgos posibles y forma de abordarlos
Análisis de negocio Caso de desarrollo	¿Beneficios? Cómo vamos aplicar UP a este proyecto

# La fase de Elaboración

# Objetivos de la fase de Elaboración

- Tanto la funcionalidad como el dominio del problema se estudian en profundidad
- Se define la arquitectura básica
- Se planifica el proyecto considerando recursos disponibles

# Criterios de evaluación de la fase

- Al comienzo de la fase de Elaboración:
  - Se planifica la fase y se forma el equipo
  - Se establecen criterios de evaluación que habrá que cumplir al final:
    - Respecto a los requisitos:
      - ¿Se han identificado? ¿Se han detallado lo suficiente?
    - En cuanto a la arquitectura:
      - ¿Satisface los requisitos? ¿Es robusta?
    - Los riesgos:
      - ¿Se han eliminado los críticos? ¿Se ha completado la lista?
    - Evaluar el proyecto:
      - ¿Se puede fijar un precio y una fecha de entrega?

# Disciplinas en la fase de elaboración

- Requisitos
  - Encontrar los casos de uso y actores
  - Determinar la prioridad de los casos de uso
  - Detallar los casos de uso
  - Estructurar el modelo de casos de uso
  - Construir prototipos de las interfaces de usuario
- Análisis
  - Análisis de la arquitectura
  - Análisis de los casos de uso
  - Análisis de clases y paquetes
- Diseño
  - Diseño de la arquitectura (estilo, subsistemas)
  - Diseño de los casos de uso
- Implementación
  - Implementación de la arquitectura base (para una fracción de casos de uso)
  - Integración del sistema (con bibliotecas de servicios, frameworks)
- Pruebas
  - Planificar y diseñar las pruebas
  - Realizar pruebas de integración y de sistema



# Artefactos de la fase de Elaboración

Artefacto	Descripción
Modelo de casos de uso Modelo de dominio	La mayoría de los casos de uso Conceptos del dominio
Modelo de análisis	Diagramas de clases Diagramas de interacción
Modelo de diseño Arquitectura del sistema	Diagramas de paquetes y clases Ideas fundamentales del diseño que se utilizará en el sistema
Modelo de pruebas	Qué debe ser probado y cuándo
Modelo de implementación	Incluye diagramas de implementación y el código fuente disponible
Prototipos de la interfaz de usuario	Todo lo relacionado con la interfaz
Modelo de datos	Traducción a esquemas de bases de datos

# Las fases de Construcción y Transición

# Fase de Construcción

- El producto se desarrolla a través de iteraciones
  - Cada iteración involucra análisis, diseño e implementación
  - La arquitectura básica se refina de manera incremental conforme se construye
- Gran parte del trabajo es programación y pruebas
- Se documenta tanto el sistema construido como el manejo del mismo
- Esta fase proporciona un producto construido junto con la documentación
- Al comienzo de esta fase, se asigna personal y se fijan los criterios de evaluación:
  - Lista de casos de uso implementados
  - Documentación inicial para los usuarios

# Disciplinas en la fase de Construcción

- Requisitos
  - Completar los casos de uso y el detalle de los mismos
  - Desarrollar prototipos de interfaz de usuario
- Análisis
  - Análisis de los casos de uso añadidos
  - Análisis de clases
- Diseño
  - Diseño de los casos de uso añadidos
- Implementación
  - Implementación de la arquitectura
  - Implementación de clases y subsistemas
  - Realizar pruebas de unidad
  - Integración del sistema
- Pruebas
  - Planificar y diseñar las pruebas
  - Realizar pruebas de integración
  - Realizar pruebas de sistema
  - Evaluar las pruebas

# Control en la fase de Construcción

- Además de las disciplinas técnicas, es preciso llevar a cabo labores de gestión:
  - Control del análisis de negocio
  - Evaluación de la fase de Construcción
  - Planificación de la fase de Transición

# Artefactos de la fase de Construcción

Artefacto	Descripción
Modelo de casos de uso Modelo de análisis Modelo de diseño Modelo de pruebas Arquitectura del sistema	Conjunto de artefactos producidos hasta ahora    Arquitectura definitiva
Modelo de implementación Modelo de pruebas	Incluye el código fuente
Sistema ejecutable	Versión con capacidad operativa inicial (V. Beta)
Manual de usuario	Versión inicial
Análisis de negocio Plan de proyecto	Situación actual del proyecto Plan para la fase de Transición

# Fase de Transición

- Se libera el producto y se entrega al usuario para un uso real
- Se incluyen tareas de instalación, configuración, entrenamiento, soporte, mantenimiento, etc.
- Los manuales de usuario se completan y refinan con la información anterior
- Estas tareas se realizan también en iteraciones
- Al comienzo de la fase, se reasigna personal y se establecen los criterios de evaluación:
  - ¿Han sido capaces los usuarios de llevar a cabo todos los casos de usos?
  - ¿Ha superado el producto las pruebas de aceptación?
  - ¿Tiene el manual de usuario una calidad suficiente?
  - ¿Están listos los cursos de formación para los usuarios?
  - ¿Están los usuarios satisfechos?

# Disciplinas en la fase de Transición

- El esquema de actividades es distinto del resto de las fases:
  - Preparar la versión de pruebas de aceptación a partir de la versión inicial
  - Instalar la versión en los lugares elegidos
    - Incluirá la migración de datos
  - Reaccionar a los resultados de las pruebas
    - Fallos en un componente, un diseño, un caso de uso
    - Problemas de fondo
  - Adaptación del producto a entornos variados
- ¿Cuándo acaba el proyecto?
  - En un producto “a medida”, el punto clave son las pruebas de aceptación
  - En un producto de venta masiva, el proyecto no acaba nunca realmente



**Muchas gracias**