

# INGENIERÍA DE SOFTWARE

## CICLOS DE VIDA Y METODOLOGIAS

Rubby Casallas, Andrés Yie

Departamento de Sistemas y Computación

Facultad de Ingeniería

Universidad de los Andes

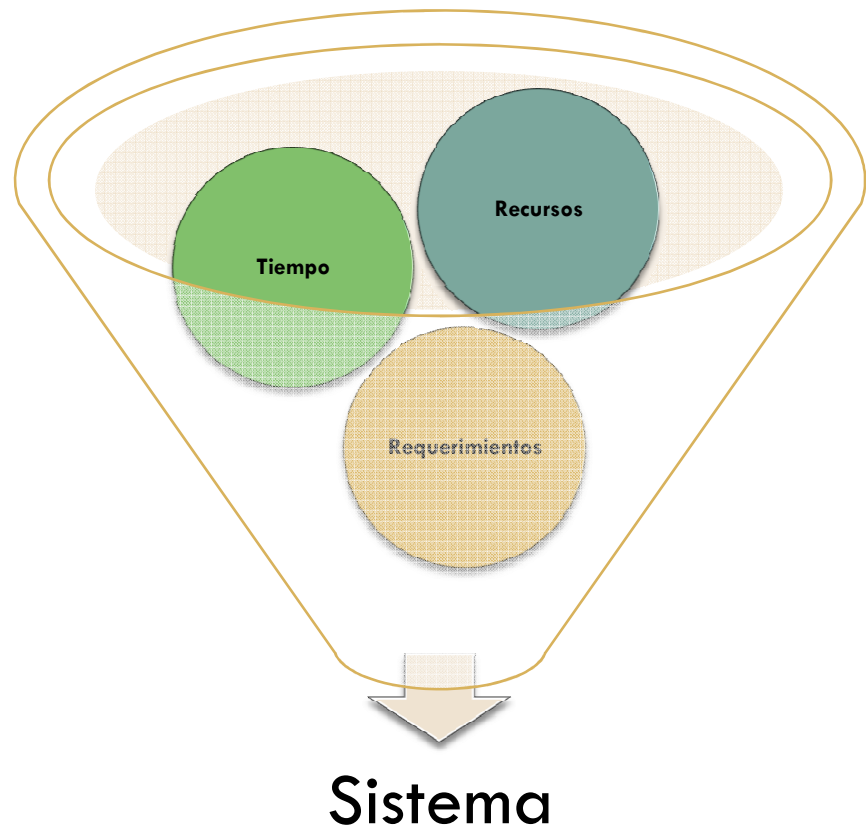
# Agenda



- Contexto
- Ciclos de vida:
  - ▣ Modelo en Cascada
  - ▣ Modelo en V
  - ▣ Modelo en Espiral
  - ▣ Modelo Unified Process
- IEEE 1074

# Contexto

- El desarrollo de un sistema de software está enmarcado por los **recursos**, el **tiempo** y un conjunto de **requerimientos**.



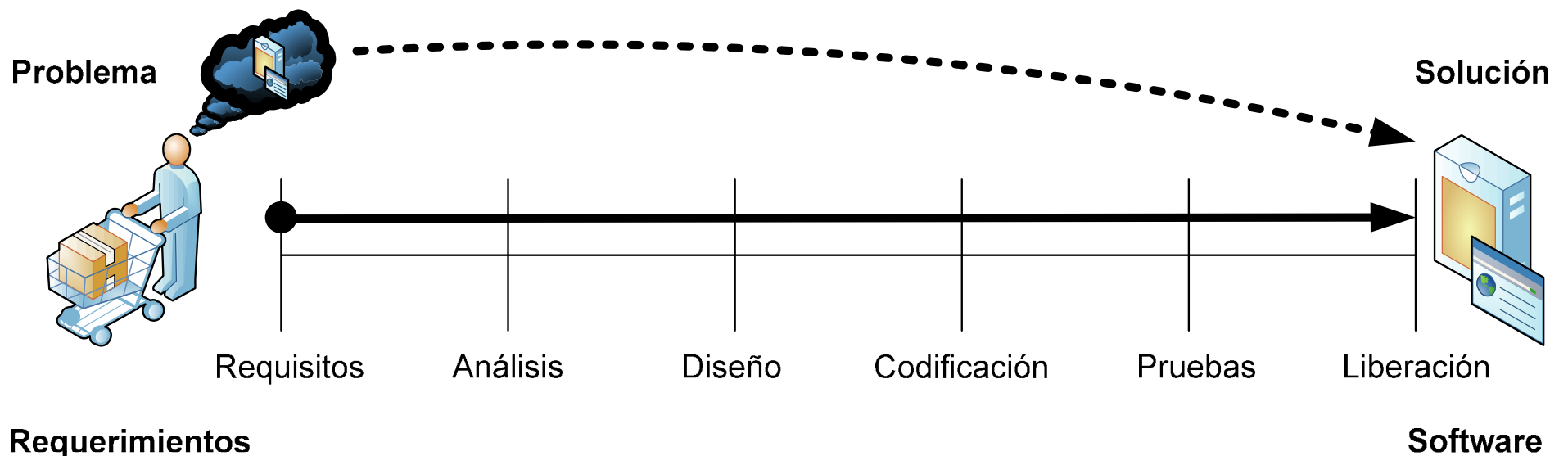
# Contexto

- Para lograrlo debe existir una **planeación** y un **seguimiento** a ésta.
- Una planeación está conformada por **actividades, recursos y tiempo**
- Esas actividades se llevan a cabo dentro de un **proceso definido**



# Proceso de construcción de software

- *“El conjunto completo de actividades de ingeniería de software necesarias para transformar los requerimientos del usuario en software.” [Humphrey]*



# Ciclos de Vida

- El **proceso** que se sigue para **construir, entregar** y hacer **evolucionar** el software, desde la concepción de una idea hasta la entrega y el retiro del sistema.
- Representa todas las **actividades** y **artefactos** (productos intermedios) necesarios para **desarrollar** una aplicación



# Ciclos de Vida

- Permiten manejar la complejidad de un proyecto de desarrollo
- Vistas complementarias:
  - ▣ Las actividades
  - ▣ Los entregables (artefactos)



# Actividades de un Ciclo de Vida

Implícita o Explícitamente todos los modelos de ciclo de vida cuentan por lo menos con las siguientes actividades



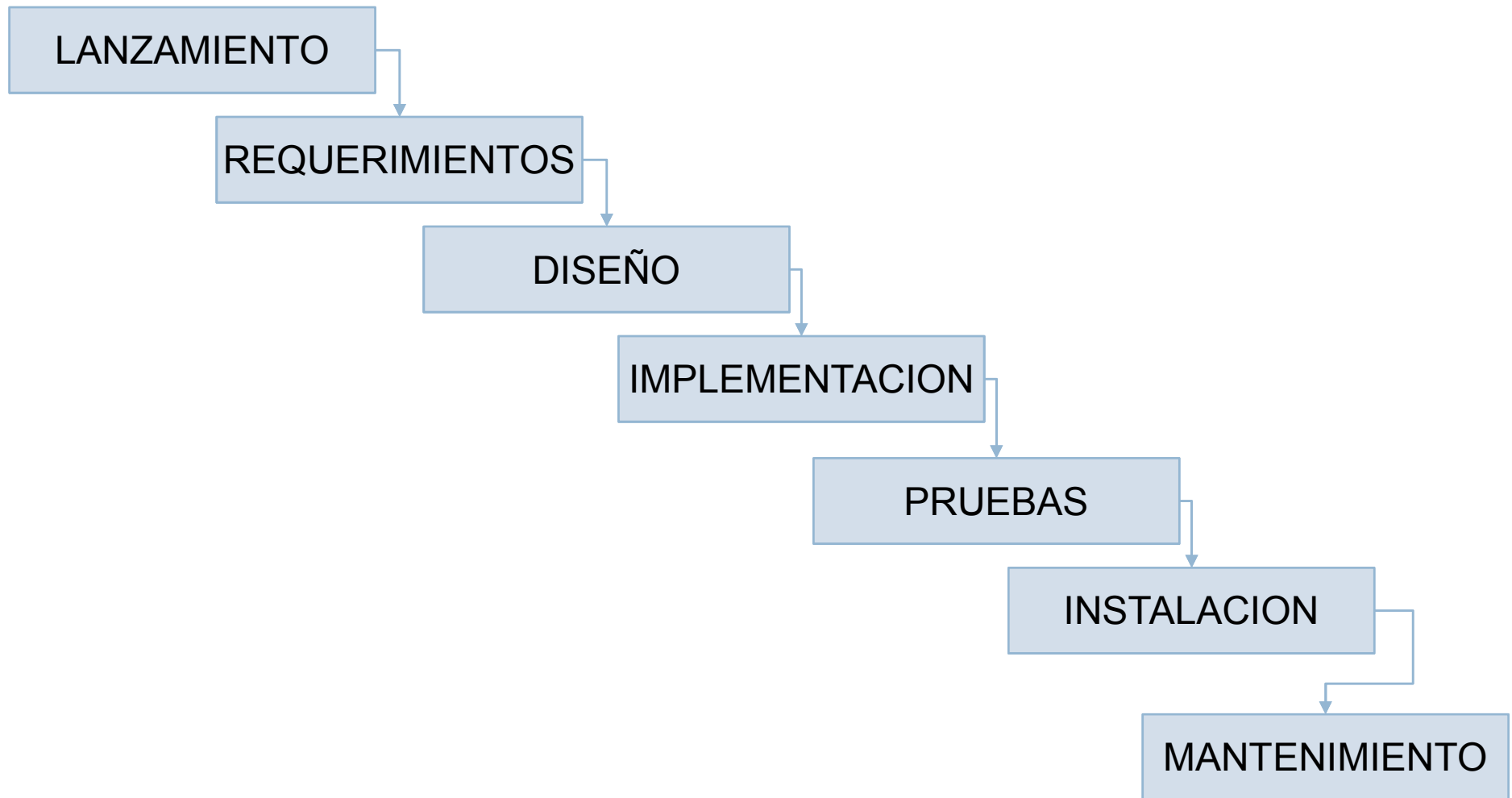


# Modelo en Cascada



- Modelo orientado en las actividades
- Prescribe una ejecución secuencial de un subconjunto de los procesos de desarrollo y de administración
- Es el modelo mas antiguo, propuesto por Winston Royce en 1970.

# Modelo en Cascada



# Modelo en Cascada - Fortalezas



- ❑ Fácil entendimiento e implementación
- ❑ Ampliamente utilizado y conocido ( En teoría )
- ❑ Refuerza buenos hábitos: definir antes que diseñar, diseñar antes que codificar
- ❑ Identifica entregables e hitos.
- ❑ Orientado a documentos.
- ❑ Funciona bien en productos maduros y equipos débiles

# Modelo en Cascada - Debilidades

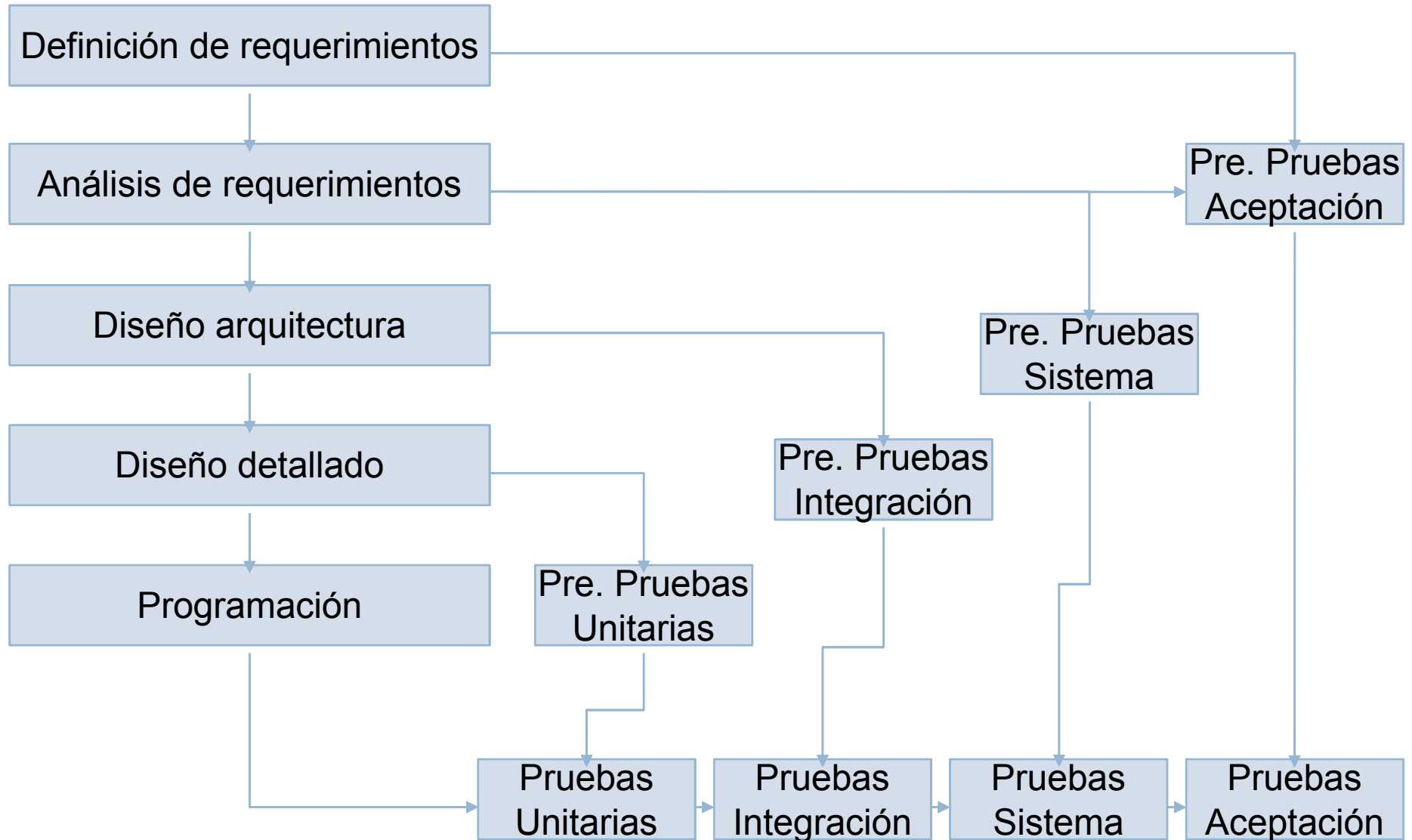
- ❑ No aprovecha la iteración, ni el desarrollo exploratorio
- ❑ Espera requerimientos definidos completamente al inicio del proyecto. (IREAL!!!)
- ❑ Dificultar para integrar administración del riesgo
- ❑ **El software es entregado tarde en el proyecto. Esto hace que se detecten errores graves muy tarde.**
- ❑ **Hacer cambios es difícil y costoso.**

# Modelo en V



- Busca hacer la actividad de pruebas más efectiva y productiva
- Los planes (y casos de prueba) se van elaborando a medida que se avanza en el desarrollo del proyecto

# Modelo en V



# Modelo en Espiral

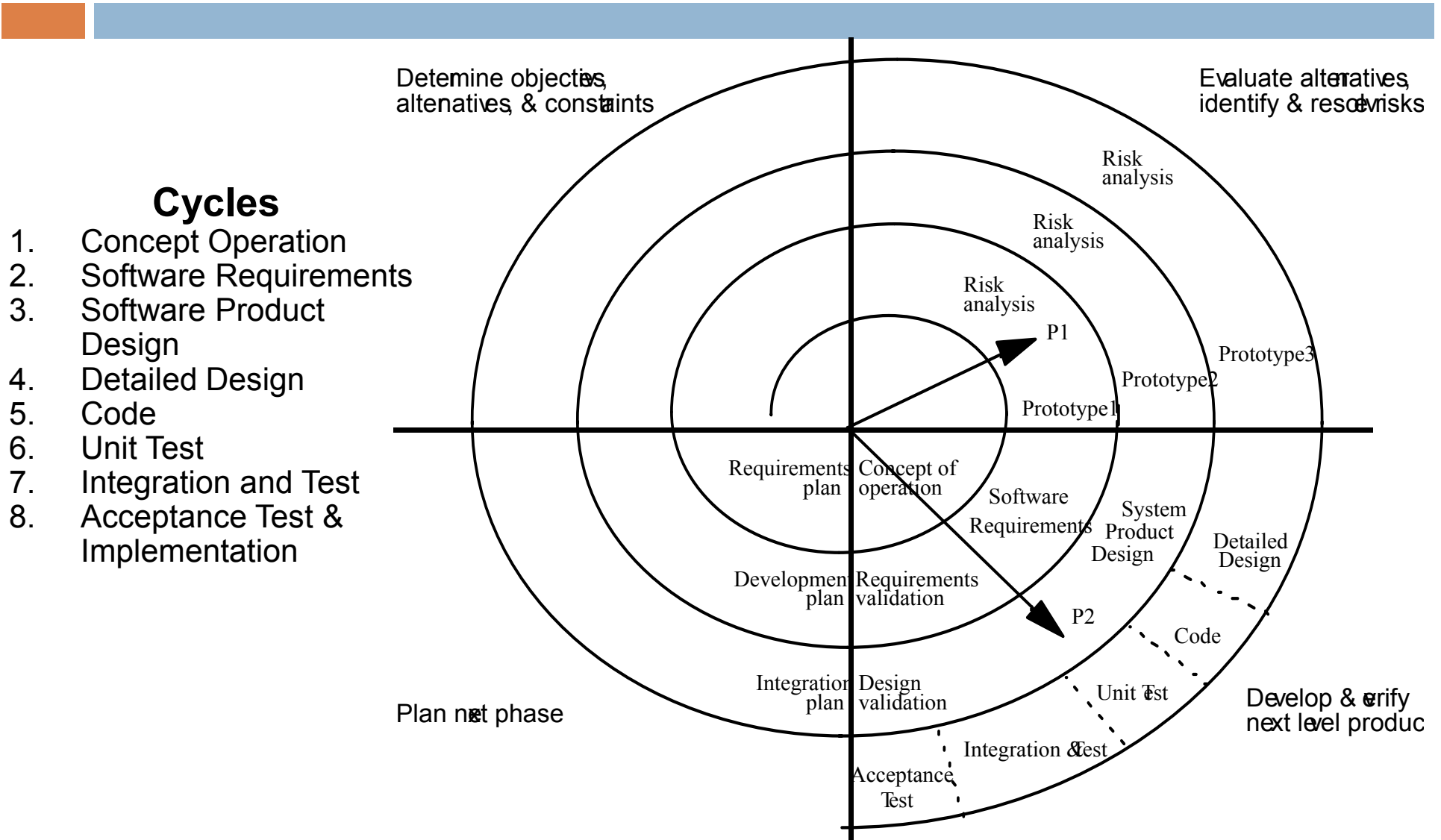
- Propuesto por Bohem en 1987
- Modelo centrado en la actividades
- Basado en las mismas actividades del modelo de cascada
- Introduce: manejo de riesgos y creación de prototipos
- Las actividades son organizadas en ciclos

# Modelo en Espiral

- Un ciclo corresponde a la construcción de un producto intermedio
- Las fases de cada ciclo son:
  - ▣ Identificar alternativas, restricciones y objetivos
  - ▣ Administrar Riesgos
  - ▣ Desarrollar y validar un prototipo
  - ▣ Planear la siguiente fase



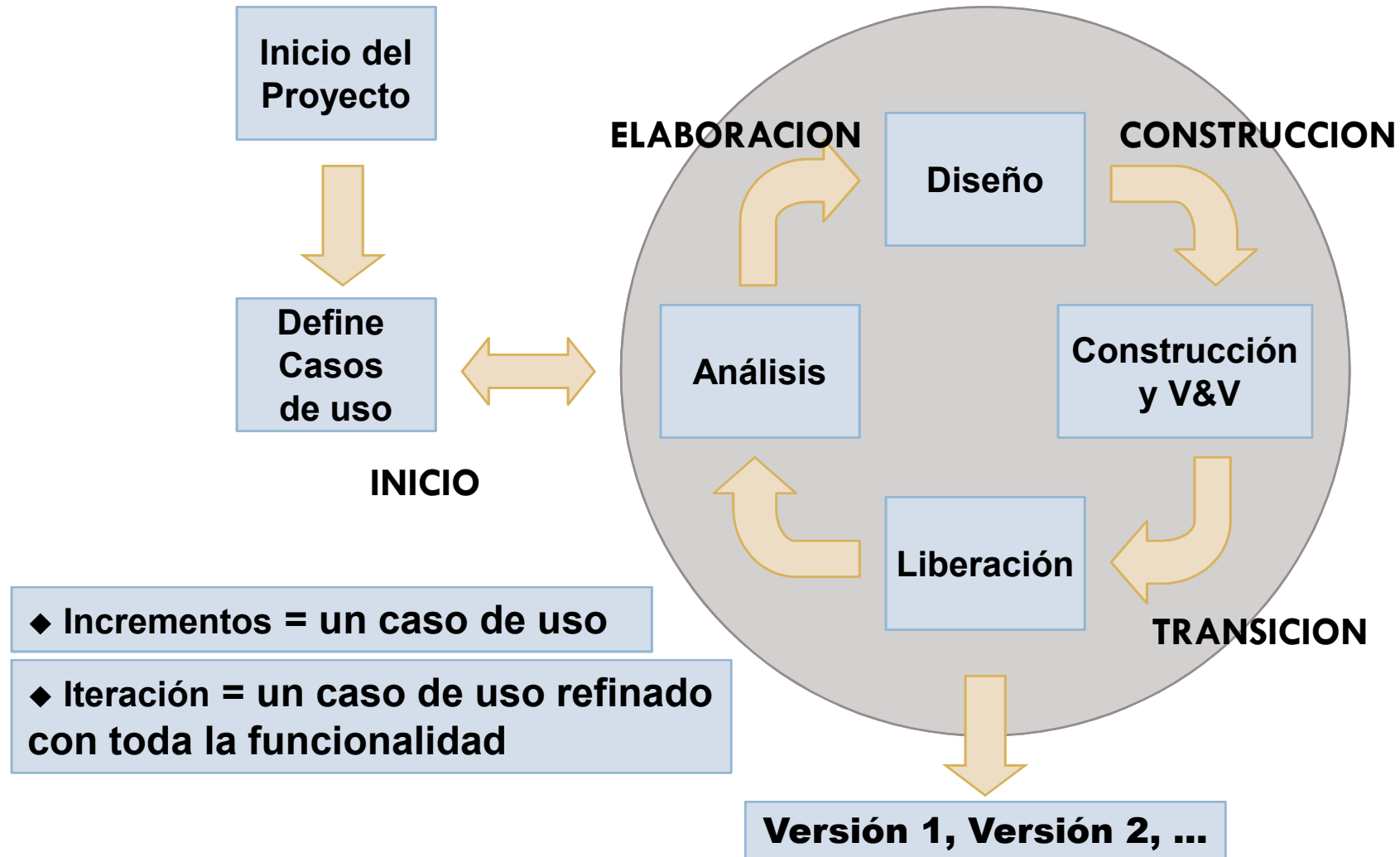
# Modelo en Espiral



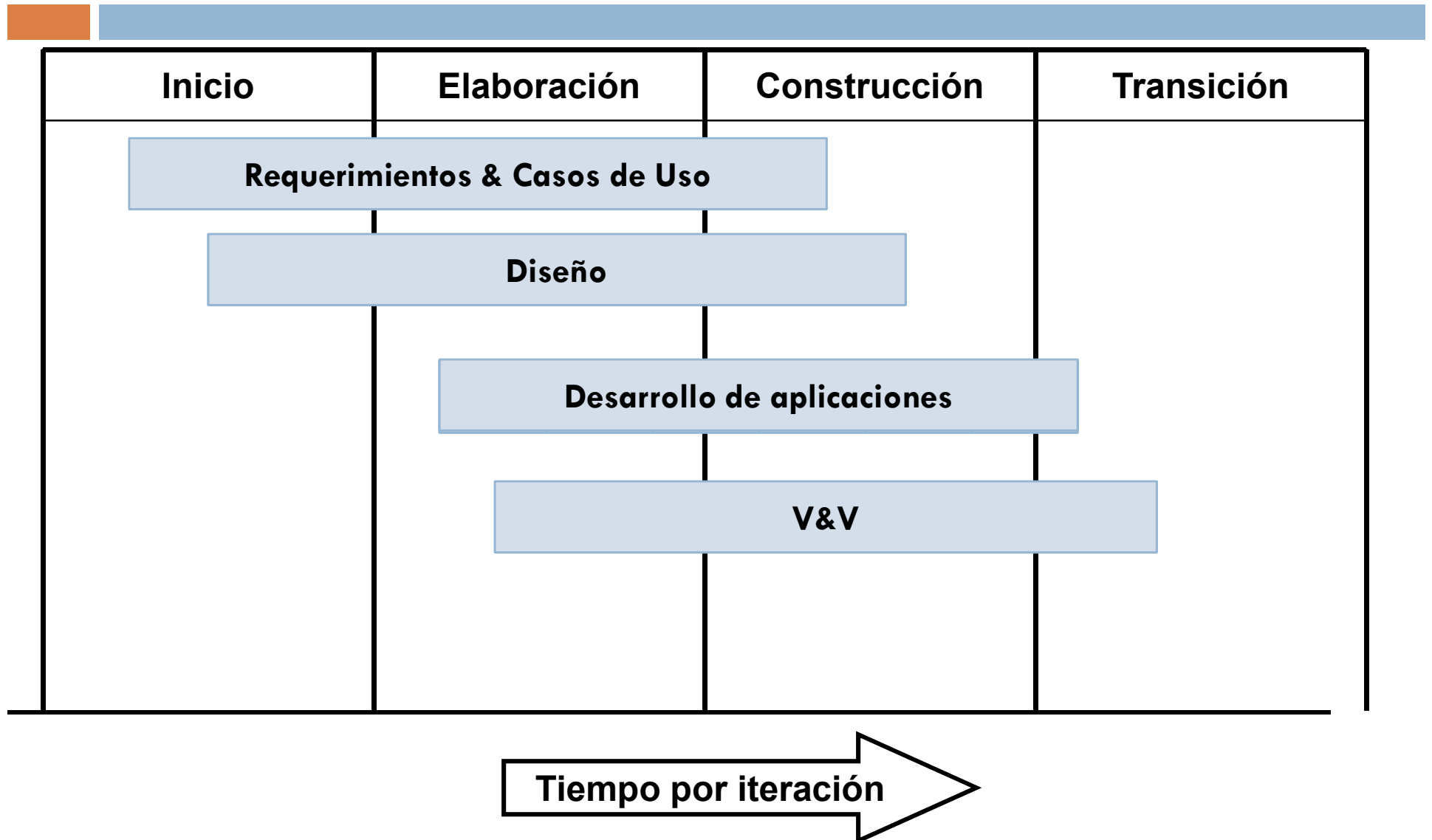
# Unified Process Model

- Presentado por Booch, Jacobson y Rumbaugh en 1999
- Consiste en varios ciclos.
- Al final de cada uno, un producto es entregado al cliente
- Cada ciclo consiste de cuatro fases:
  - ▣ Inception
  - ▣ Elaboration
  - ▣ Construction
  - ▣ Transition
- Cada fase puede tener varias iteraciones
  - ▣ Una iteración construye un conjunto de casos de uso relacionados o mitiga algún riesgo de los identificados

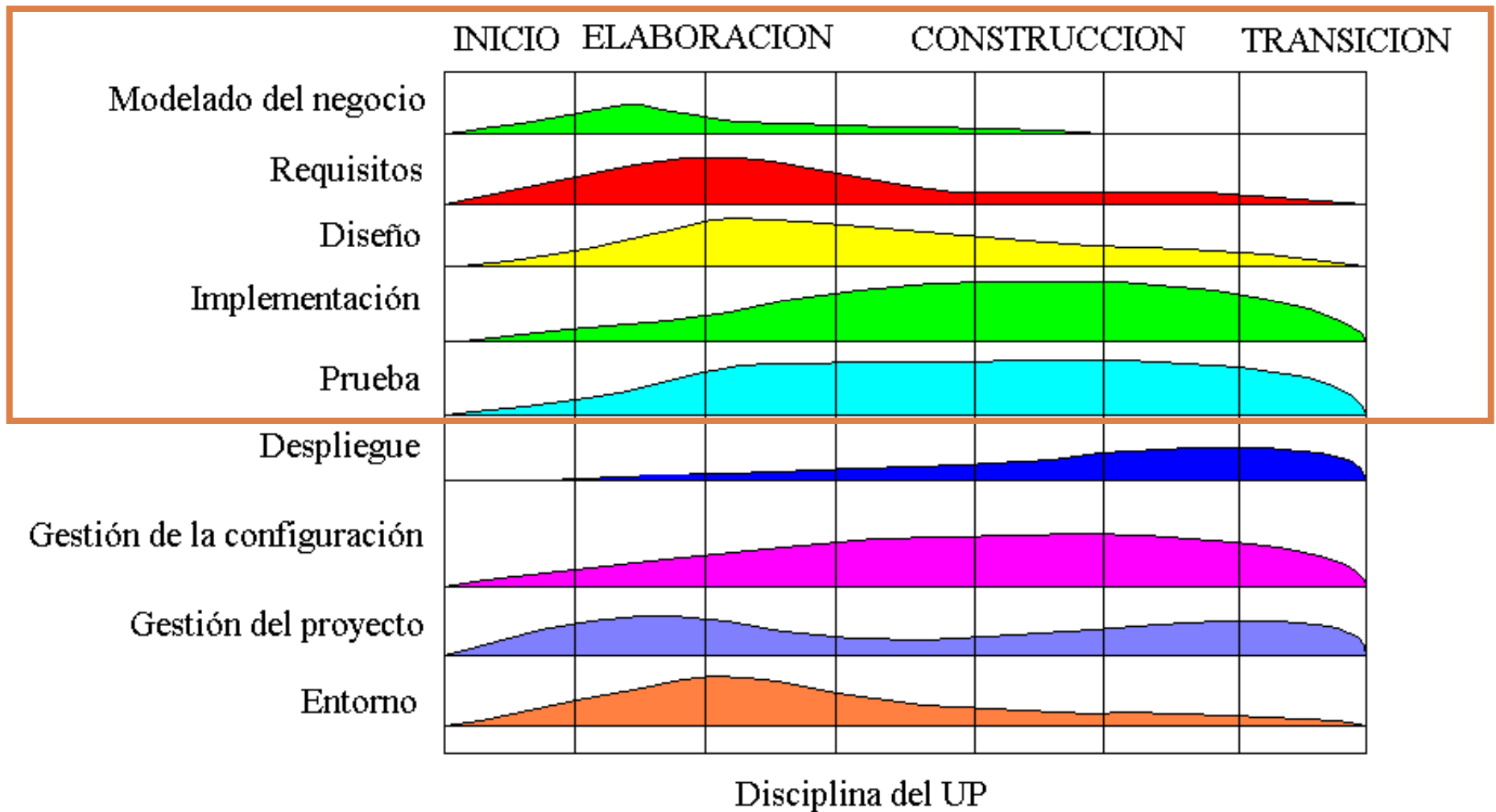
# Unified Process Model



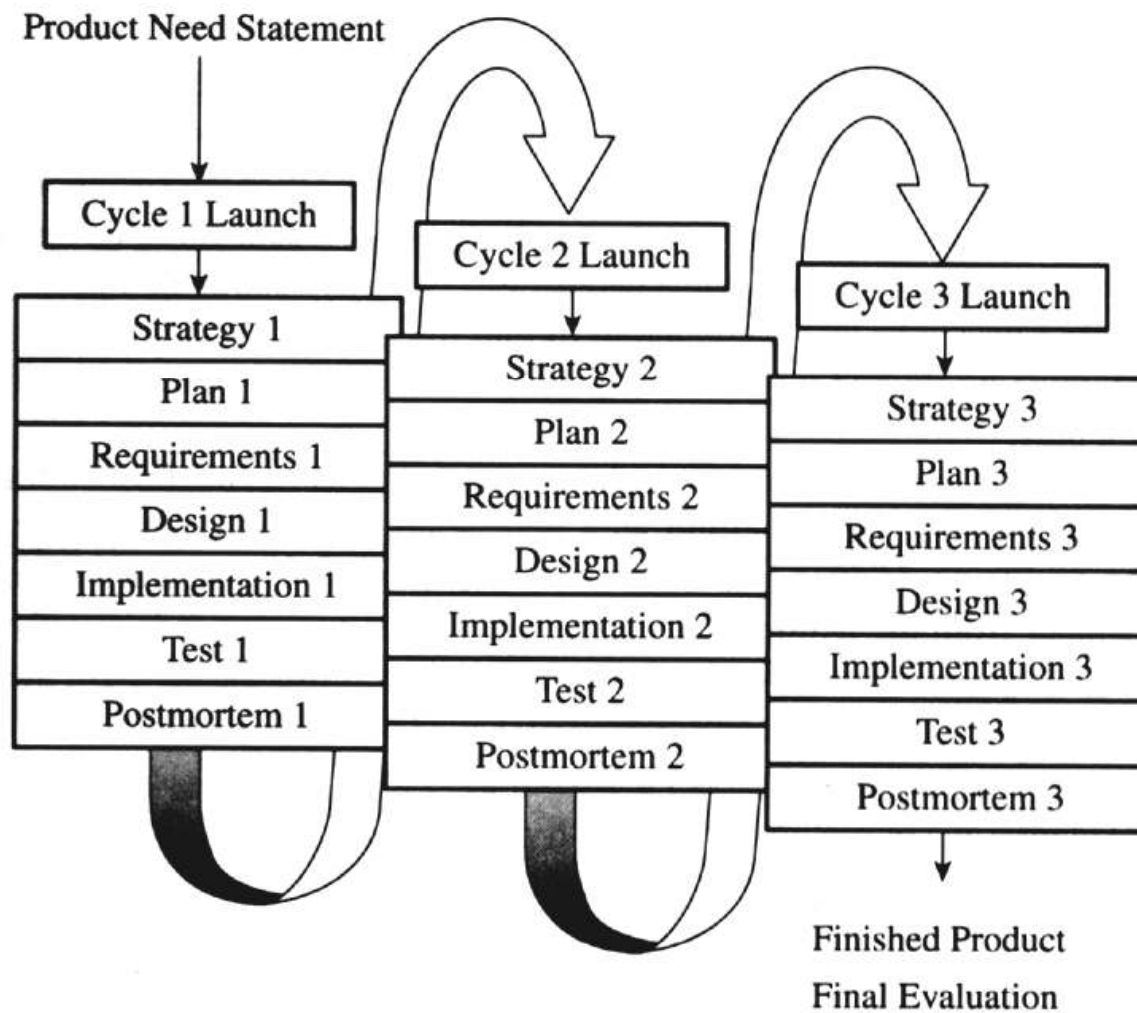
# Unified Process Model



# DESARROLLO ITERATIVO Y EL PROCESO UNIFICADO



# Team Software Process TSP



# IEEE 1074 Procesos de ciclo de vida de software



- ❑ Establecer un marco común para desarrollar modelos de ciclo de vida
- ❑ Proceso: conjunto de actividades para alcanzar un
- ❑ Propósito
- ❑ El estándar define 17 procesos organizados en grupos de procesos
- ❑ Cada proceso está compuesto de actividades

# Grupos de Proceso

1. Modelamiento del Ciclo de Vida	Selección de un ciclo de vida
2. Administración del Proyecto	Iniciación del Proyecto Monitoreo y Control del Proyecto Administración de la calidad del SW
3. Pre-desarrollo	Exploración del concepto Asignación de Recursos
4. Desarrollo	Requerimientos Diseño Implementación
5. Pos-desarrollo	Instalación Operación y Soporte Mantenimiento Retiro
6. Procesos Integrales	Verificación y Validación Administración de la configuración Documentación Entrenamiento



# 1. Modelamiento del ciclo de vida

- Responsable: Gerente del proyecto
- Personalizar las actividades de IEEE1074 a los requerimientos del proyecto y de la empresa
- Seleccionar el ciclo de vida:
  - ▣ Cascada
  - ▣ Modelo en V
  - ▣ Spiral
  - ▣ Rational Unified Process
  - ▣ Team Software Process
  - ▣ Extreme Programming
  - ▣ Sawtooth (Diente de tiburón) – Prototipos

## 2. Administración del proyecto



- Procesos

- ▣ Iniciar el proyecto
  - ▣ Supervisar y controlar el proyecto
  - ▣ Administración de la calidad

## 2. Administración del proyecto – Iniciar el Proyecto

- Responsable: Gerente del proyecto
- Entregable: Plan de Administración del proyecto de software (SPMP)
- Propósito:
  - ▣ Establecer la correspondencia entre las actividades y el modelo del ciclo de vida del software
  - ▣ Asignar recursos al proyecto
  - ▣ Establecer del ambiente del proceso
  - ▣ Realizar la planeación de la administración del proyecto

## 2. Administración del proyecto – Iniciar el Proyecto

- Inicio del proyecto en general es:
  - ▣ Definir el plan de tareas
  - ▣ Definir el cronograma
  - ▣ Definir el presupuesto
  - ▣ Definir la organización del proyecto
  - ▣ Definir el ambiente del proyecto
    - Estándares
    - Comunicación
    - Procedimientos de reunión y reporte
    - Metodología de desarrollo
    - Herramientas de desarrollo

## 2. Administración del proyecto - Supervisión y control del proyecto

- Responsable: Gerente del proyecto
- Entregable: Plan de Administración del proyecto de software (SPMP) actualizado
- Propósito:
  - ▣ Analizar riesgos
  - ▣ Realizar planeación de contingencias
  - ▣ Administrar el proyecto
  - ▣ Conservar registros
  - ▣ Implementar el modelo de reporte de problemas

## 2. Administración del proyecto – Admin. de la calidad

- Responsable: Equipo de administración de calidad
- Entregable: Plan de Administración de la calidad
- Propósito:
  - ▣ Planear la administración de la calidad del software
  - ▣ Definir métricas
  - ▣ Administrar la calidad del software
  - ▣ Identificar las necesidad de mejora de calidad

# Grupos de Proceso

1. Modelamiento del Ciclo de Vida	Selección de un ciclo de vida
2. Administración del Proyecto	Iniciación del Proyecto Monitoreo y Control del Proyecto Administración de la calidad del SW
3. Pre-desarrollo	Exploración del concepto Identificación del sistema
4. Desarrollo	Requerimientos Diseño Implementación
5. Pos-desarrollo	Instalación Operación y Soporte Mantenimiento Retiro
6. Procesos Integrales	Verificación y Validación Administración de la configuración Documentación Entrenamiento

# 3. Predesarrollo



- Administración o marketing y el cliente identifican una idea o una necesidad
  - ▣ Nuevo esfuerzo de desarrollo
  - ▣ Cambio de interfaz de un sistema existente
  - ▣ Reemplazo de software de un proceso de negocios existente
- Procesos
  - ▣ Exploración del concepto
  - ▣ Asignación del sistema



### 3. Predesarrollo - Exploración del concepto

- Responsable: Administración y Cliente
- Entregable: Enunciado del problema
- Propósito:
  - ▣ Producir el enunciado del problema (“Statement of needs”)
  - ▣ Identificar ideas o necesidades
  - ▣ Formular enfoques potenciales
  - ▣ Realizar estudios de factibilidad
  - ▣ Planear la transición del sistemas
  - ▣ Refinar y finalizar la idea o necesidad

# 3. Predesarrollo – Identificación del sistema

- Responsable: Administración, Cliente, Arquitecto de software
- Entregable: Enunciado del problema y Documento de arquitectura
- Propósito:
  - ▣ Establecer la arquitectura inicial
    - Base para la comunicación entre los miembros
  - ▣ Identificar
    - Hardware
    - Software
    - Requerimientos funcionales

# Grupos de Proceso

1. Modelamiento del Ciclo de Vida	Selección de un ciclo de vida
2. Administración del Proyecto	Iniciación del Proyecto Monitoreo y Control del Proyecto Administración de la calidad del SW
3. Pre-desarrollo	Exploración del concepto Identificación del sistema
4. Desarrollo	Requerimientos Diseño Implementación
5. Pos-desarrollo	Instalación Operación y Soporte Mantenimiento Retiro
6. Procesos Integrales	Verificación y Validación Administración de la configuración Documentación Entrenamiento

## 4. Desarrollo - Requerimientos



- Responsable: Equipo de desarrollo y cliente
- Entregable: Documento de requerimientos
- Propósito:
  - ▣ Definir y desarrollar los requerimientos de software
  - ▣ Definir los requerimientos de la interfaz
  - ▣ Establecer la prioridad e integrar los requerimientos de software

# 4. Desarrollo - Requerimientos



- Responsable: Equipo de desarrollo
- Entregable: Documentos de diseño
- Propósito:
  - ▣ Producir una representación del sistema coherente y bien organizada
  - ▣ Descomponer en subsistemas
  - ▣ Asignar los requerimiento a los sistemas de hardware y software
  - ▣ Describir de las condiciones de frontera
  - ▣ Seleccionar de componentes pre- existentes
  - ▣ Definir los objetivos de diseño

## 4. Desarrollo - Implementación



- Responsable: Equipo de desarrollo
- Entregable: Código, Pruebas
- Propósito:
  - ▣ Construir la solución
  - ▣ Crear datos de pruebas
  - ▣ Crear código fuente
  - ▣ Crear código objeto
  - ▣ Crear la documentación operativa
  - ▣ Planear la integración
  - ▣ Realizar la integración

# Grupos de Proceso

1. Modelamiento del Ciclo de Vida	Selección de un ciclo de vida
2. Administración del Proyecto	Iniciación del Proyecto Monitoreo y Control del Proyecto Administración de la calidad del SW
3. Pre-desarrollo	Exploración del concepto Identificación del sistema
4. Desarrollo	Requerimientos Diseño Implementación
5. Pos-desarrollo	Instalación Operación y Soporte Mantenimiento Retiro
6. Procesos Integrales	Verificación y Validación Administración de la configuración Documentación Entrenamiento

## 5. Posdesarrollo - Instalación



- Responsable: Administración y cliente
- Entregable: Sistema
- Propósito:
  - ▣ Planear la instalación
  - ▣ Distribuir el software
  - ▣ Instalar el software
  - ▣ Aceptar el software en el ambiente operacional



## 5. Posdesarrollo - Operación y soporte



- Responsable: Administración y cliente
- Propósito:
  - ▣ Operar el sistema
  - ▣ Proporcionar asistencia técnica y consultoría
  - ▣ Mantener la bitácora de peticiones del clientes

## 5. Posdesarrollo - Mantenimiento



- Responsable: Administración y equipo de desarrollo
- Propósito:
  - ▣ Mantener el sistema adecuado a los cambios en los requerimientos, nuevas necesidades, nuevas tecnologías.

## 5. Posdesarrollo - Retiro



- ▣ Notificar a los usuarios
- ▣ Realizar operaciones paralelas
- ▣ Retirar el sistemas

# Grupos de Proceso

1. Modelamiento del Ciclo de Vida	Selección de un ciclo de vida
2. Administración del Proyecto	Iniciación del Proyecto Monitoreo y Control del Proyecto Administración de la calidad del SW
3. Pre-desarrollo	Exploración del concepto Identificación del sistema
4. Desarrollo	Requerimientos Diseño Implementación
5. Pos-desarrollo	Instalación Operación y Soporte Mantenimiento Retiro
6. Procesos Integrales	Verificación y Validación Administración de la configuración Documentación Entrenamiento

## 6. Procesos integrales - Verificación y validación

### □ Propósito:

- Planear la verificación y validación: inspecciones, revisiones, auditorias, pruebas
- Ejecutar las tareas de verificación y validación
- Recopilar y analizar datos de medidas
- Planear las pruebas
- Desarrollar los requerimientos de las pruebas
- Ejecutar las pruebas

## 6. Procesos integrales - Administración de la configuración del software

- Propósito:

- Planear la administración de la configuración
- Desarrollar la identificación de la configuración
- Realizar el control de cambios

## 6. Procesos integrales - Documentación



- Planear la documentación
- Implementar la documentación
- Producir y distribuir la documentación

## 6. Procesos integrales - Entrenamiento



- Planear el programa de entrenamiento
- Desarrollar los materiales de entrenamiento
- Validar el programa de entrenamiento
- Implementar el programa de entrenamiento



# Referencias

- [Bernd Bruegge, Dutoit Allen. *Object-Oriented Software Engineering: Using UML, Patterns, and Java*, 2004, Prentice Hall, segunda edición. Capítulo 15
- [http://standards.ieee.org/catalog/olis/arch\\_se.html](http://standards.ieee.org/catalog/olis/arch_se.html)