



UNIVERSIDAD TÉCNICA
FEDERICO SANTA MARÍA

Ingeniería de Software: Parte 2

Agustín J. González

ELO329: Diseño y Programación
Orientados a Objeto

Adaptado de:

<http://www.dsic.upv.es/~uml>

<http://inst.eecs.berkeley.edu/~cs169/>
entre otras fuentes.

Algunas Metodologías ...

- Personal Software Process y Team software Process
- XP (Programación Extrema)
- RUP (Rational Unified Process)
- La familia Cristal de Cockburn
- Código Abierto
- ASD (Desarrollo de Software Adaptable)
- SCRUM
- FFD (Desarrollo Manejado por Rasgos)
- DSDM (Método de desarrollo de sistema dinámico)

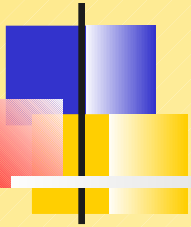


Yo no conozco todas, pero vale la pena conocer al menos una. Aquellas en **rojo** son populares.



UNIVERSIDAD TÉCNICA
FEDERICO SANTA MARÍA

RUP ...





Desarrollo de software: Características de RUP ...

Guiado y Manejado por Casos de Uso

Centrado en la Arquitectura

Iterativo e Incremental

Desarrollo Basado en Componentes

Utilización de UML

Proceso Integrado



Casos de uso: Idea general

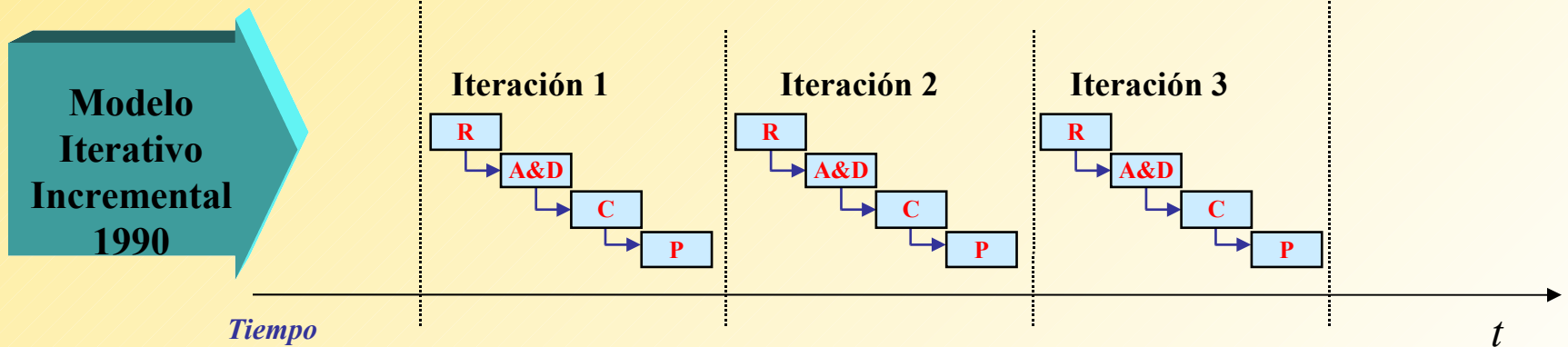
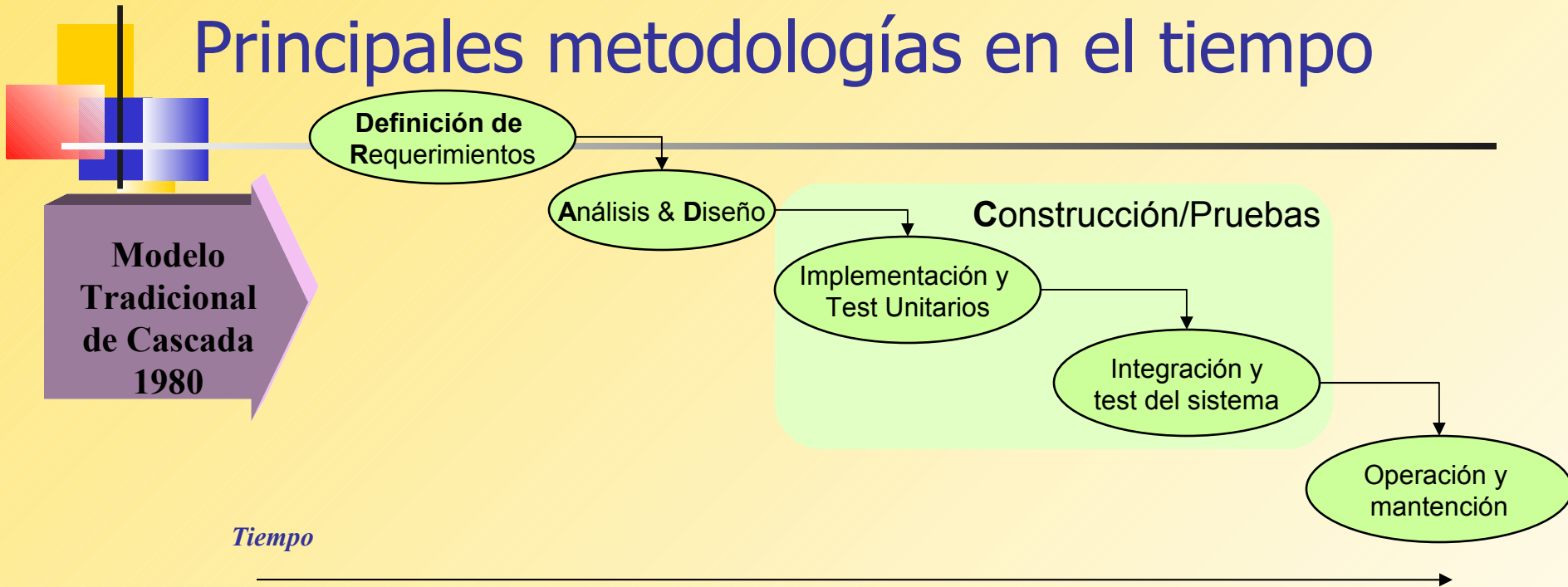
- Un caso de uso es una colección de **escenarios** de éxito y fallas relacionados que describen **actores** usando el sistema para alcanzar algún objetivo.
- Un **escenario** es una secuencia específica de acciones e interacciones entre actores y el sistema bajo discusión.
- Un **actor** es algo externo al sistema discusión que interactúa con éste. Por ejemplo: Una persona, otro sistema computacional, un sensor, etc.
- Los casos de uso muestran el comportamiento observable del sistema.

Recordar el proceso de desarrollo de software completo



Parte a ser considerada en adelante

Principales metodologías en el tiempo



Rational Unified Process (RUP)



Rational Unified Process
1998

Rational Objectory Process
1996-1997

Objectory Process
1987-1995

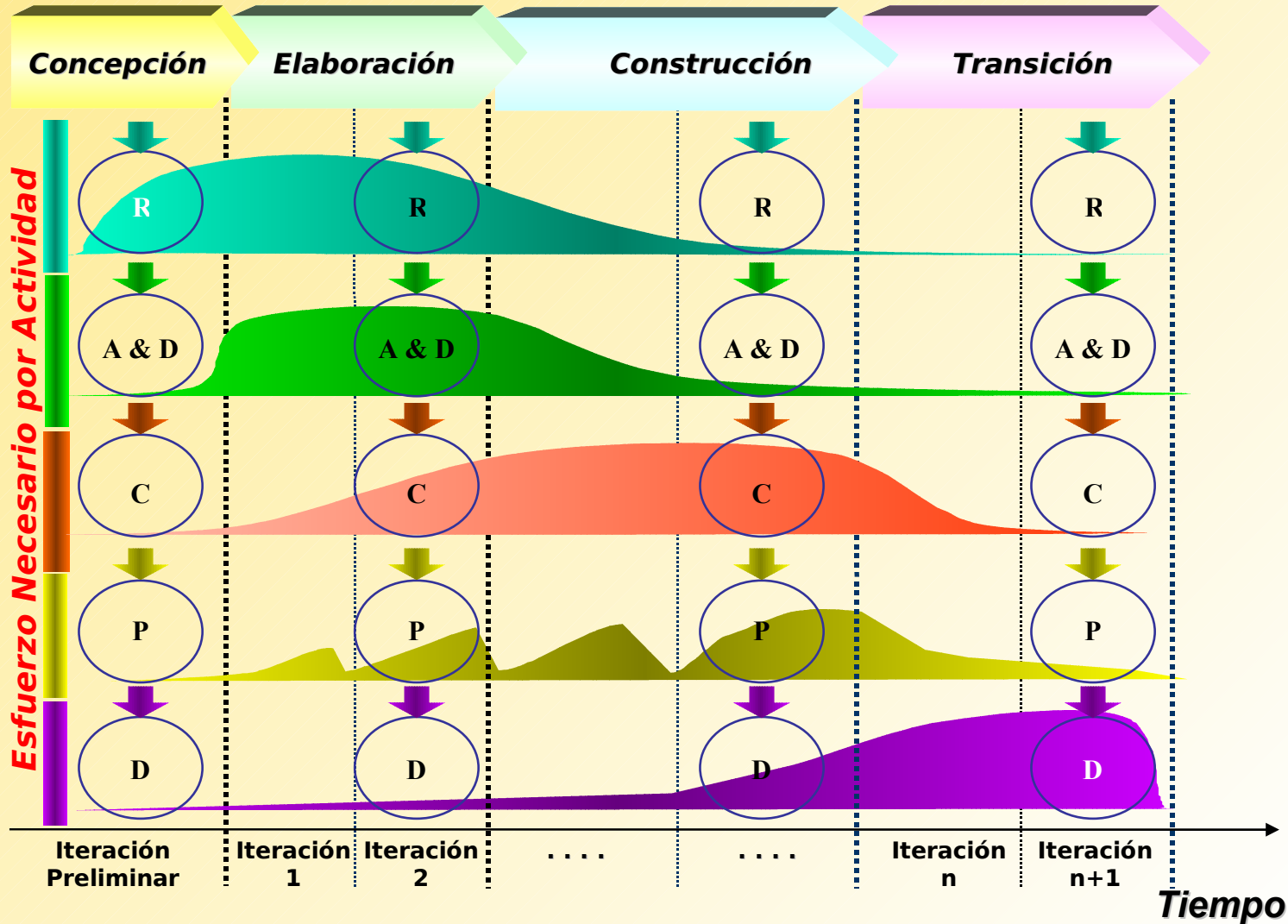
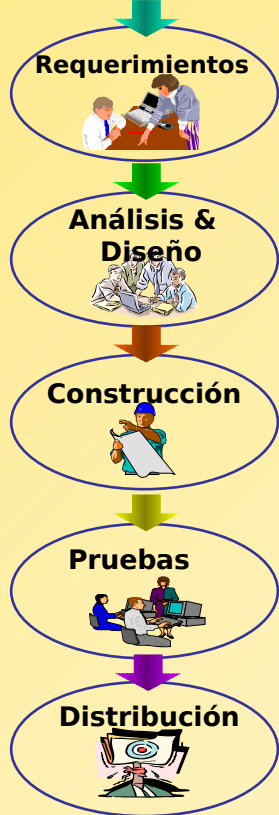
Enfoque Ericsson

- Pruebas funcionales
- Pruebas de desempeño
- Gestión de requisitos
- Gestión de cambios y configuración
- Ingeniería de Negocio
- Ingeniería de datos
- Diseño de interfaces

UML

RUP Define Fases de Desarrollo ...

Áreas de trabajo
(Workflow)





Fases del Ciclo de Vida

- El ciclo de vida consiste en una serie de ciclos, cada uno de los cuales produce una nueva versión del producto
- Cada ciclo está compuesto por fases y cada una de estas fases está compuesta por un número de iteraciones
- Las fases son 4:
 - Concepción, Inicio o Estudio de oportunidad
 - Elaboración
 - Construcción
 - Transición



..Fases del Ciclo de Desarrollo

- **Inicio o Estudio de oportunidad (*inception*)**
 - Define el ámbito y objetivos del proyecto
 - Se define la funcionalidad y capacidades del producto
- **Elaboración**
 - Tanto la funcionalidad como el dominio del problema se estudian en profundidad
 - Se define una arquitectura básica
 - Se planifica el proyecto considerando recursos disponibles



...Fases del Ciclo de Desarrollo

Construcción

- El producto se desarrolla a través de iteraciones donde cada iteración involucra tareas de análisis, diseño e implementación
- Las fases de estudio y análisis sólo dieron una arquitectura básica que es aquí refinada de manera incremental conforme se construye (se permiten cambios en la estructura)
- 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



...Fases del Ciclo de Desarrollo

- **Transición**

- Se libera el producto y se entrega al usuario para un uso real
- Se incluyen tareas de marketing, empaquetado atractivo, 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

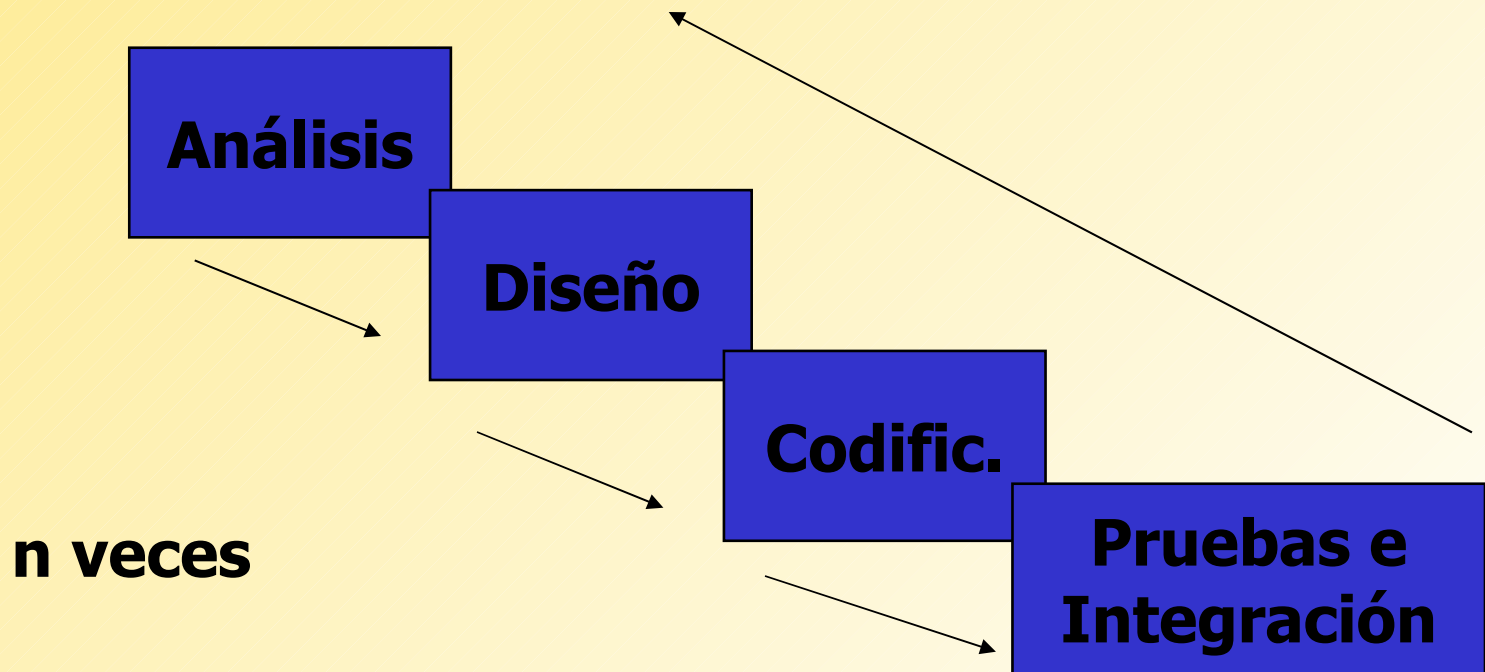


Proceso Iterativo e Incremental

- El ciclo de vida iterativo se basa en la evolución de prototipos ejecutables que se muestran a los usuarios y clientes
- En el ciclo de vida iterativo, en cada iteración se reproduce el ciclo de vida en cascada a menor escala
- Los objetivos de una iteración se establecen en función de la evaluación de las iteraciones precedentes

... Proceso Iterativo e Incremental

- Las actividades se encadenan en una minicascada con un alcance limitado por los objetivos de la iteración





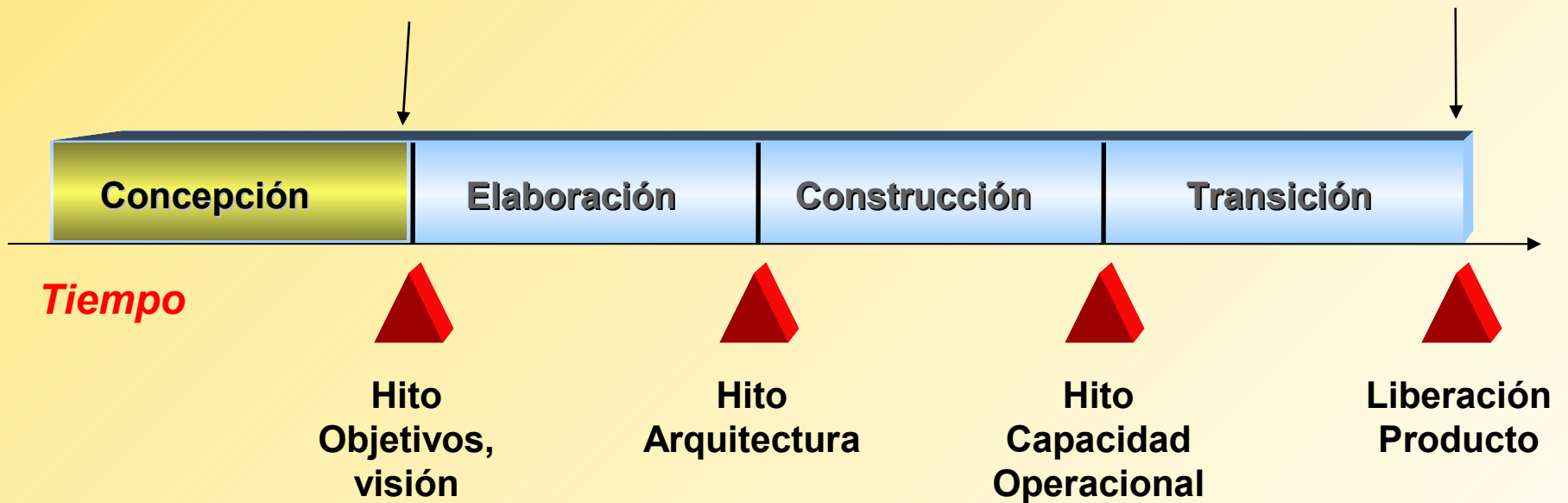
... Proceso Iterativo e Incremental

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

Importancia de los Hitos en RUP ...

Compromiso de recursos para fase elaboración

Aceptación del cliente





... Elementos en RUP

- **Artefactos**
- Es el Resultado parcial o final que es producido y usado durante el proyecto. Son las entradas y salidas de las actividades
- Un artefacto puede ser un documento, un modelo o un elemento de modelo
- **Conjuntos de Artefactos**
 - Business Modeling Set
 - Requirements Set
 - Analysis & Design Set
 - Implementation Set
 - Test Set
 - Deployment Set
 - Project Management Set
 - Configuration & Change Management Set
 - Environment Set



Mejores Prácticas de RUP ...

Desarrolle Iterativamente

**Administre los
Requerimientos**

**Use
Arquitectura
de
Componentes**

**Modele
Visualmente**

**Verifique
Calidad**

Controle los Cambios



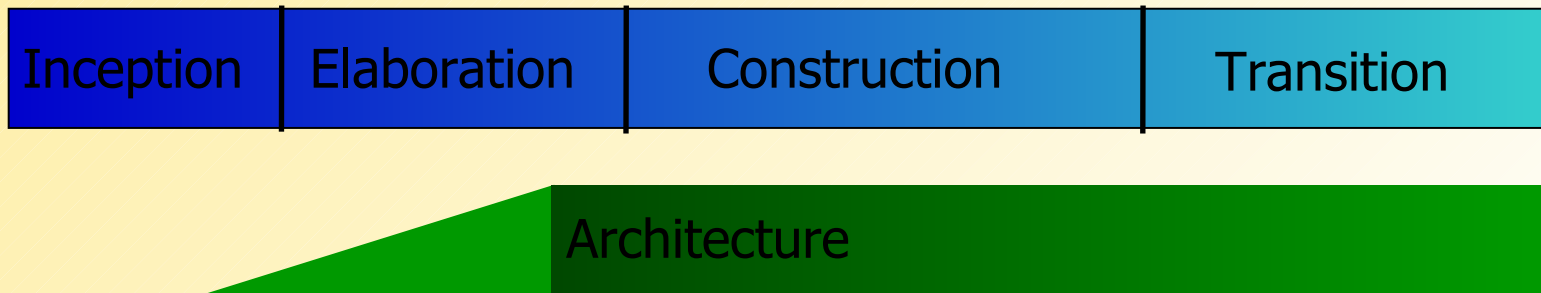
Características Esenciales de RUP

- Proceso Dirigido por los Casos de Uso
- Proceso Iterativo e Incremental
- Proceso Centrado en la Arquitectura

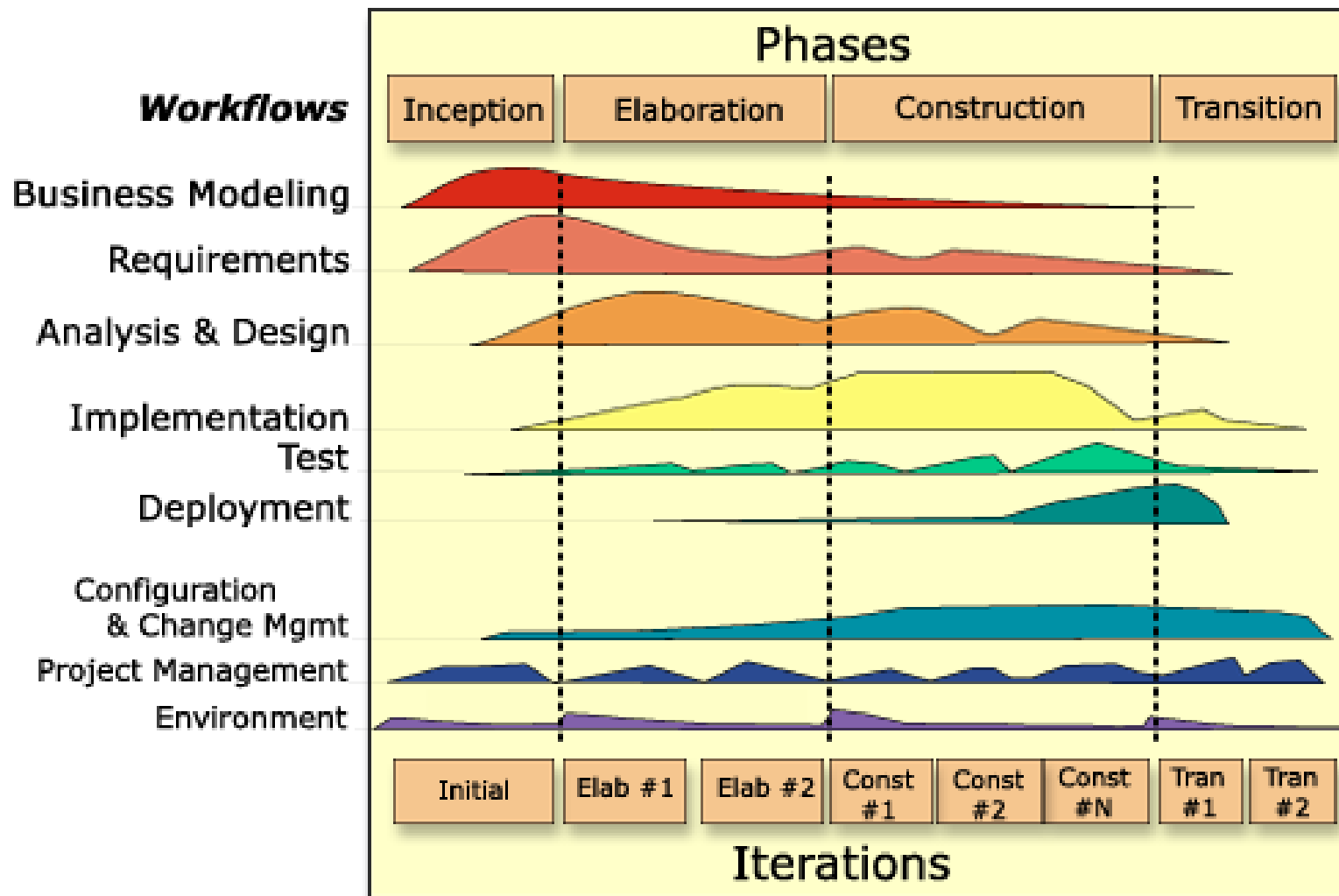


Proceso Centrado en la Arquitectura

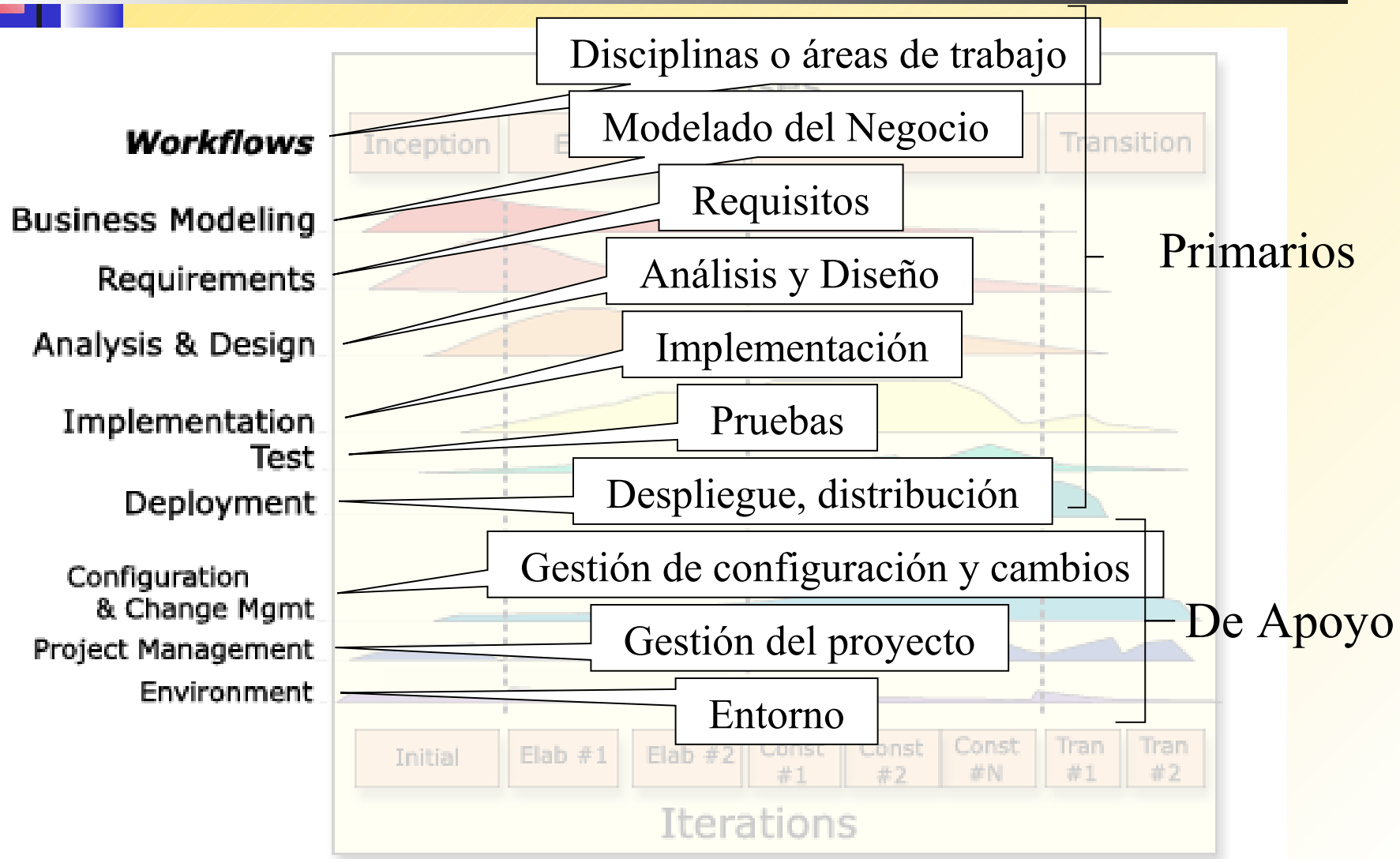
- La arquitectura de un sistema es la organización o estructura de sus partes más relevantes
- Una arquitectura ejecutable es una implementación parcial del sistema, construida para demostrar algunas funciones y propiedades
- RUP establece refinamientos sucesivos de una arquitectura ejecutable, construida como un prototipo evolutivo



Otra visión similar con más Actividades

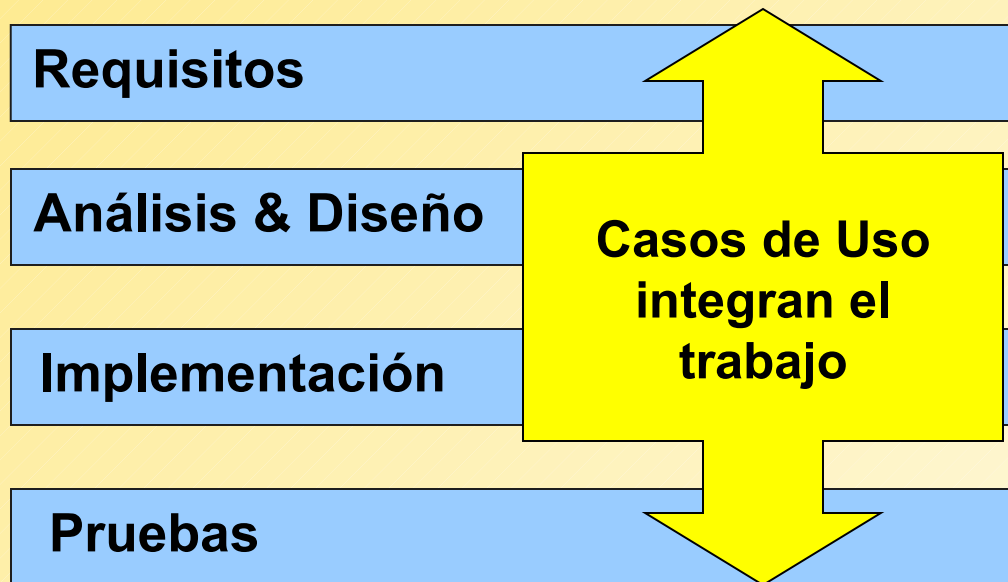


Otra visión similar con más Actividades





Proceso dirigido por los Casos de Uso

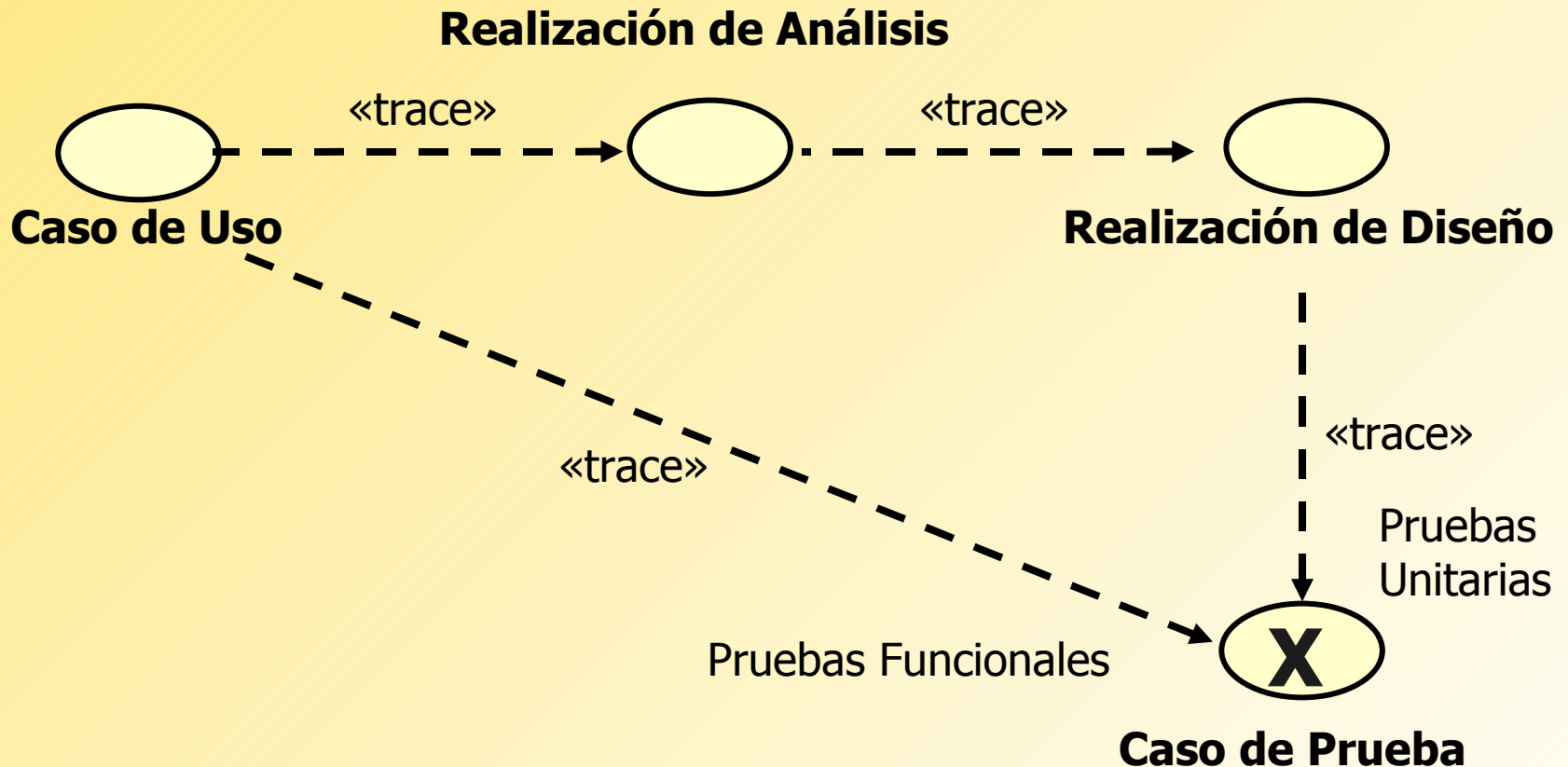


Capturar, definir y validar los casos de uso

Realizar los casos de uso

Verificar que se satisfacen los casos de uso

.. Proceso dirigido por los Casos de Uso



... Proceso dirigido por los Casos de Uso

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

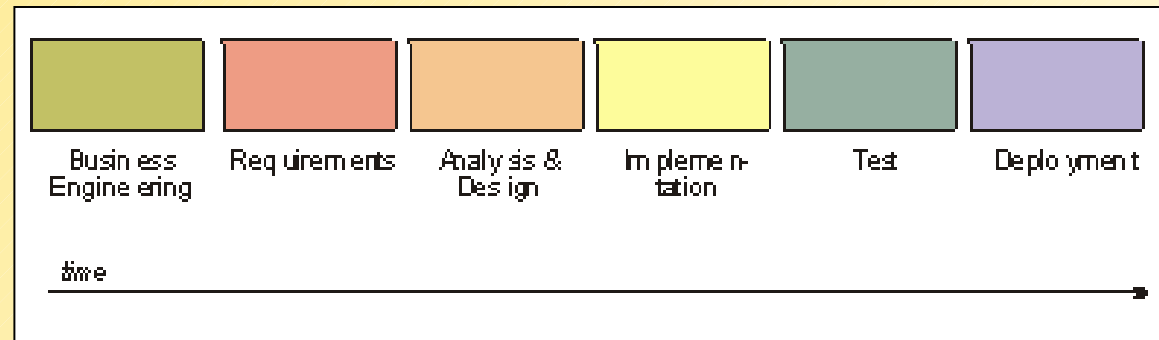
| | Modelo de Negocio Terminado | Casos de Uso Identificados | Casos de Uso Descritos | Casos de Uso Analizados | Casos de Uso Diseñados, Implementados y Probados |
|-----------------------------|------------------------------------|-----------------------------------|-------------------------------|--------------------------------|---|
| Fase de Concepción | 50% - 70% | 50% | 10% | 5% | Muy poco, puede que sólo algo relativo a un prototipo para 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 | | | | | |

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

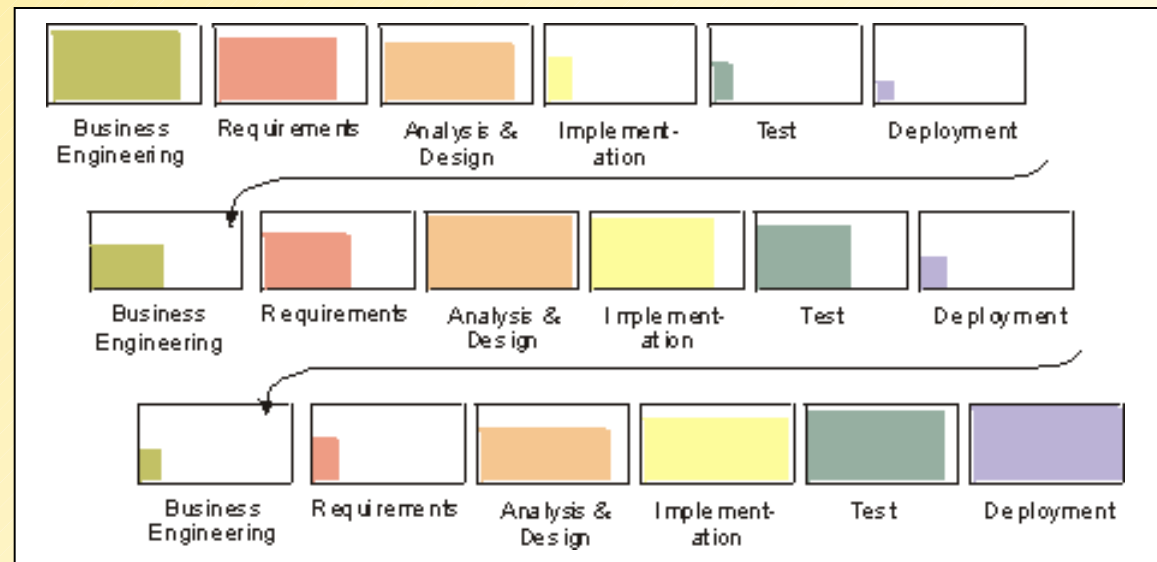
Proceso Iterativo e Incremental

Grado de completitud de cada área de trabajo

Enfoque
Cascada

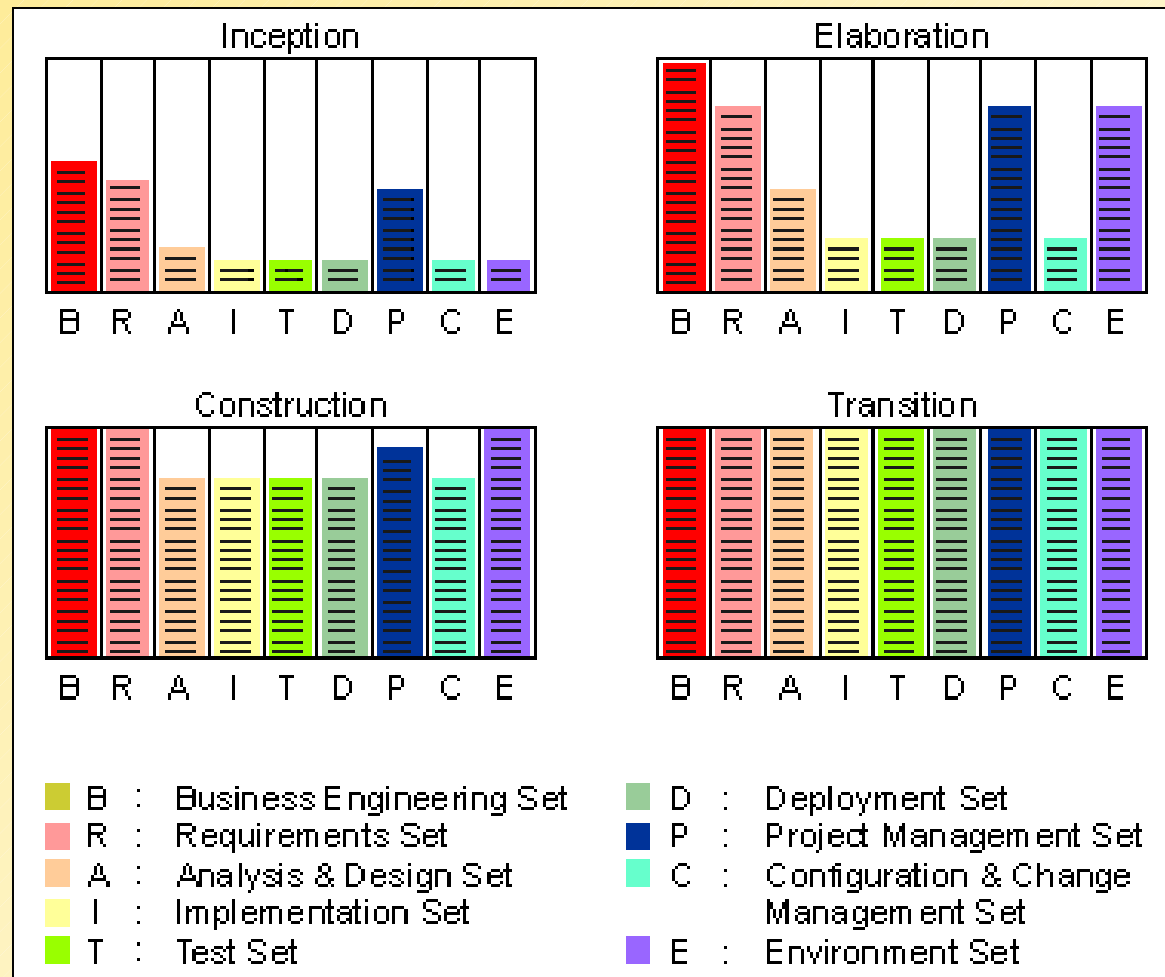


Enfoque
Iterativo e
Incremental

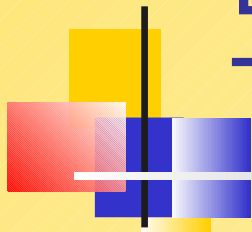


... Proceso Iterativo e Incremental

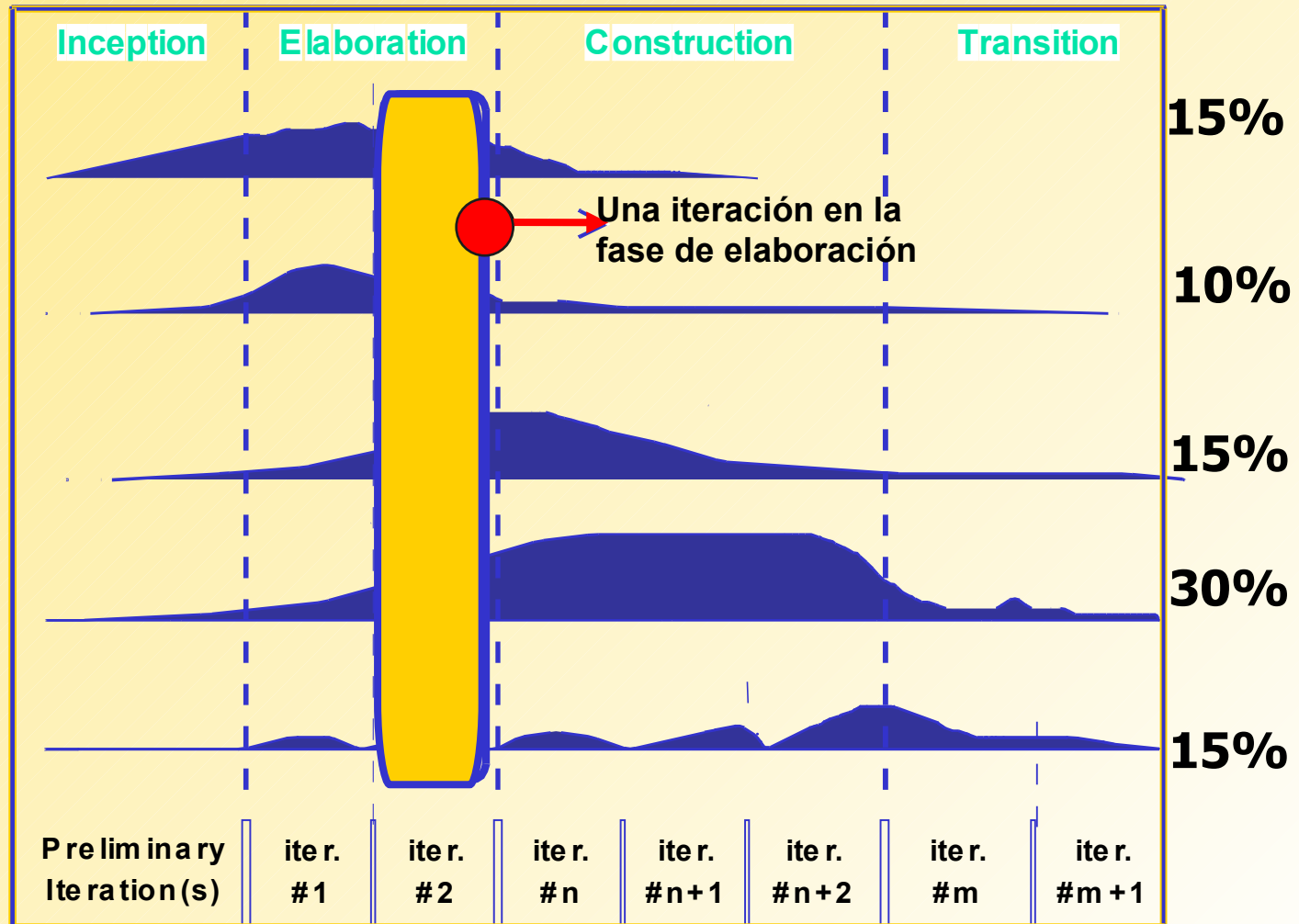
Grado de Finalización de Artefactos



Esfuerzo respecto de las Áreas de Trabajo (Workflows)

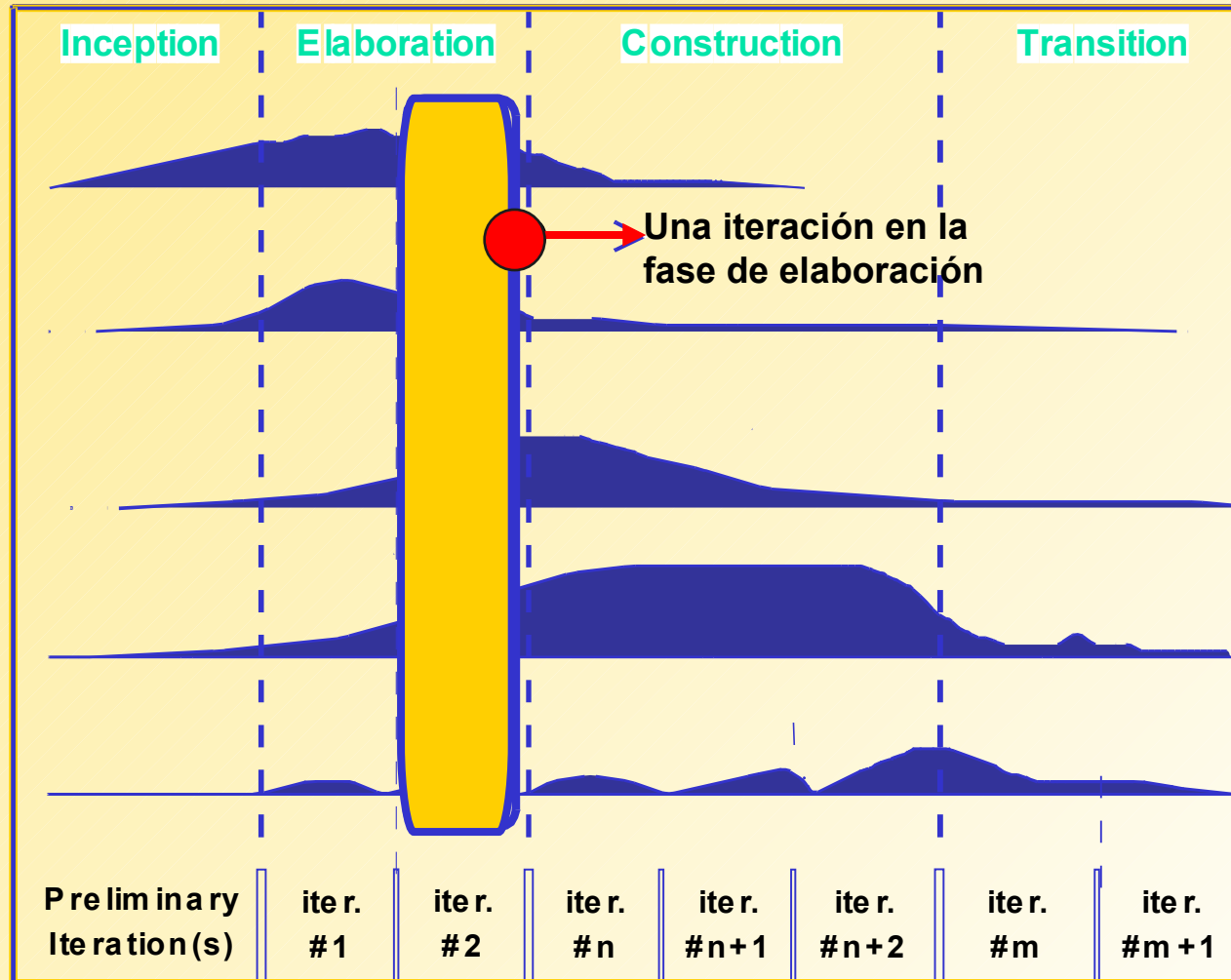


- Requisitos
- Análisis
- Diseño
- Implementación
- Pruebas



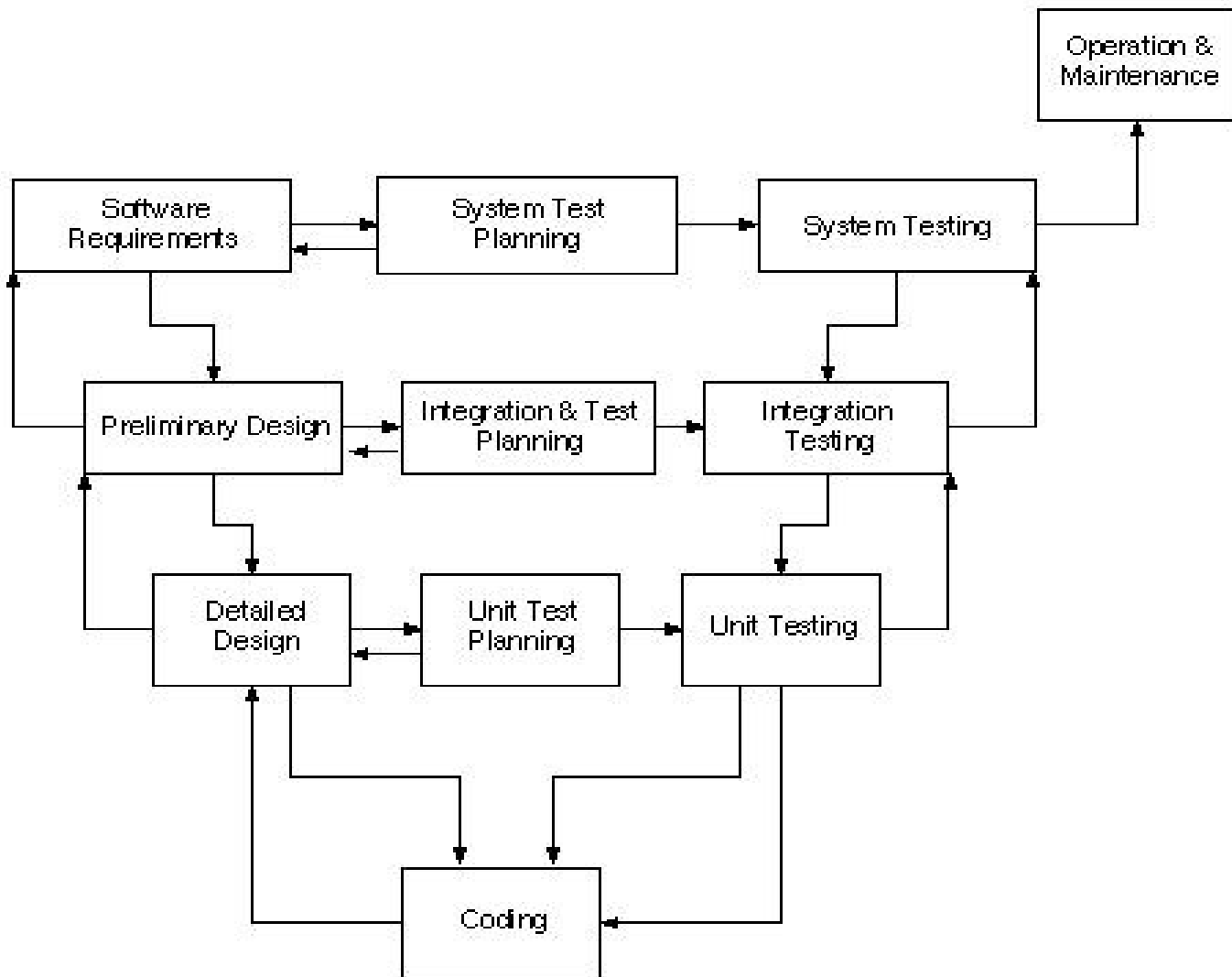
...Esfuerzo respecto de las Fases

- Requisitos
- Análisis
- Diseño
- Implementación
- Pruebas



| | | | | |
|------------------|------------|------------|------------|------------|
| Esfuerzo: | 5% | 20% | 65% | 10% |
| Duración: | 10% | 30% | 50% | 10% |

Un Ejemplo: Comparar con V-Model (Motorola)





Dos lecciones importantes

- El tiempo es independiente del contexto. Ahorrar una semana la comienzo de un proyecto es tan bueno como ahorrarla al final. Una semana es una semana.
- Es mucho más fácil ahorrar tiempo al inicio del proyecto (cuando los “entregables” son menos claros).
- Conclusión: Cómo se aplica esto al proyecto del ramo?