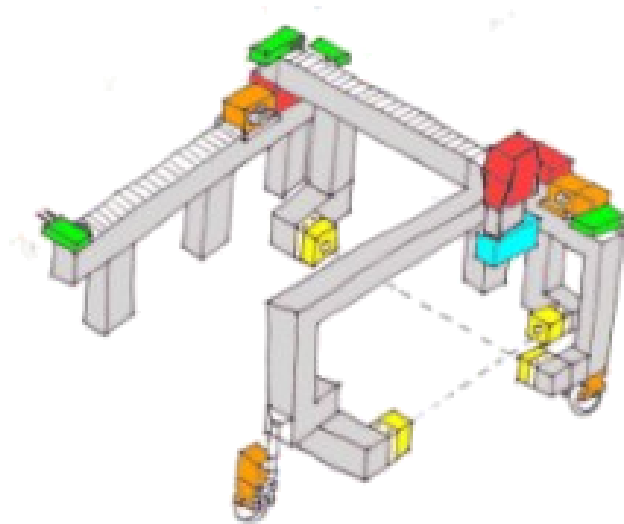

Tema 1: Introducción a la Ingeniería del Software



1. Introducción a la Ingeniería del Software

1.1. El producto Software.

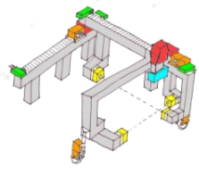
1.2. Concepto de Ingeniería del Software.

1.3. Proceso de desarrollo de Software.

1.1 El producto Software

- ✓ Definición y tipos de software.
- ✓ Características del software.
- ✓ Proceso de producción.
- ✓ Problemas en el proceso de producción.
- ✓ Otros temas de interés.

Bibliografía: [PRES13 (7ª edición) capítulo 1]
[SOMM11 (9ª edición) capítulo 1]



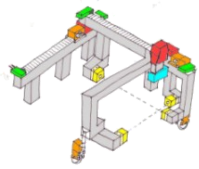
Definición de Software

- Programa o conjunto de programas de cómputo que incluye datos, procedimientos y pautas que permiten realizar distintas tareas en un sistema informático.

Software = Programa de computadora

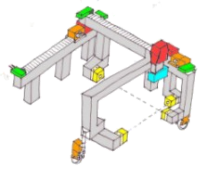
- El **software** es un transformador de información, para ello adquiere, gestiona, modifica, produce o transmite información.

¿Forma parte del software la **documentación** producida durante su desarrollo?



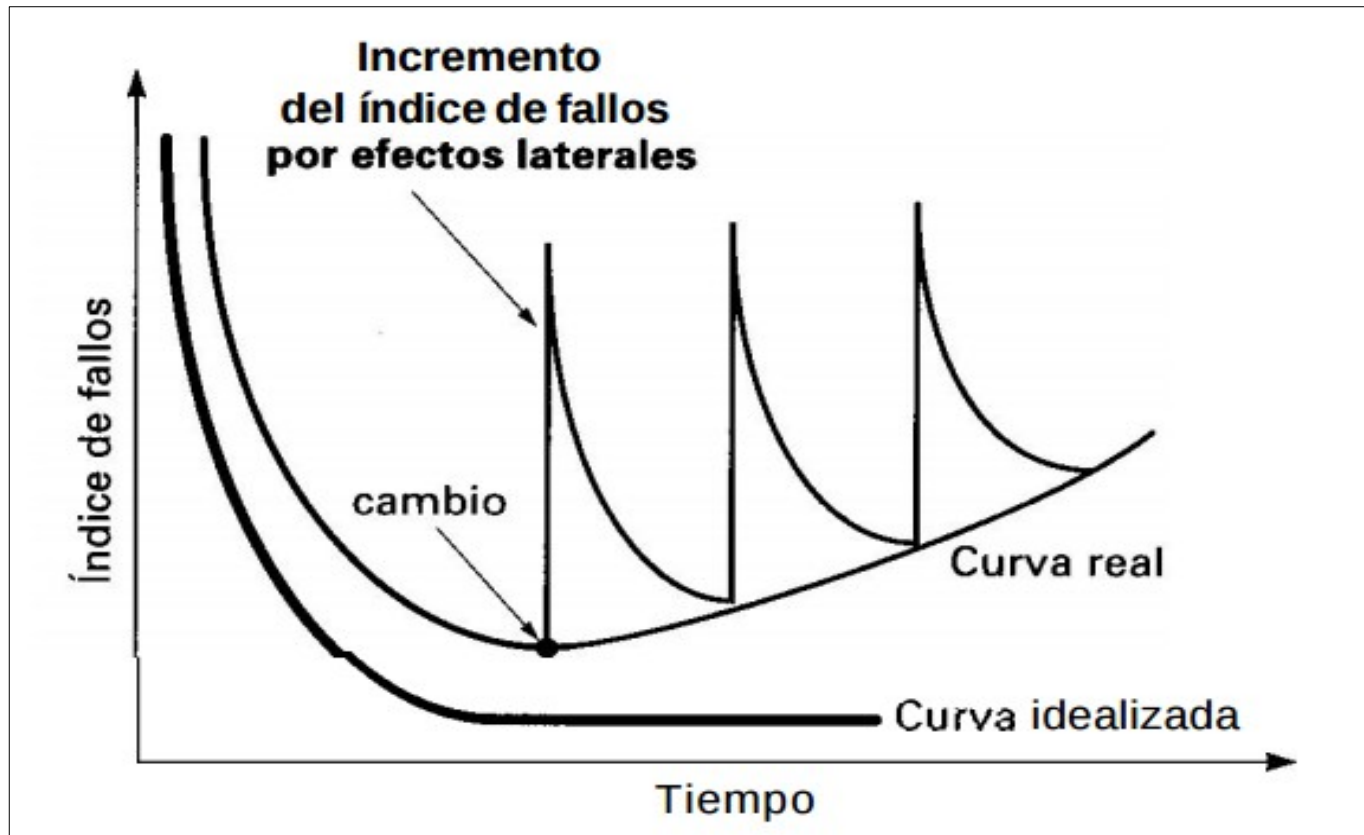
Tipos de software

- Por su **campo de aplicación**:
 - Software de **sistemas**
 - Software de **aplicaciones**
 - Software de **programación**
- Por el **tipo de licencia**:
 - Según derechos de autor:
 - Software de **código abierto**
 - Software de **código cerrado**
 - Software de **dominio público**
 - Según su destinatario:
 - Usuario final (software **hecho a medida**)
 - Para distribución (software **genérico**)

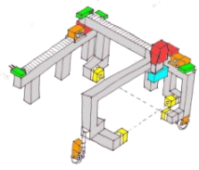


Características principales

- 1) El software es un **producto lógico**: se desarrolla, no se fabrica; **se deteriora**, no se estropea.



[PRESS13 página 5]



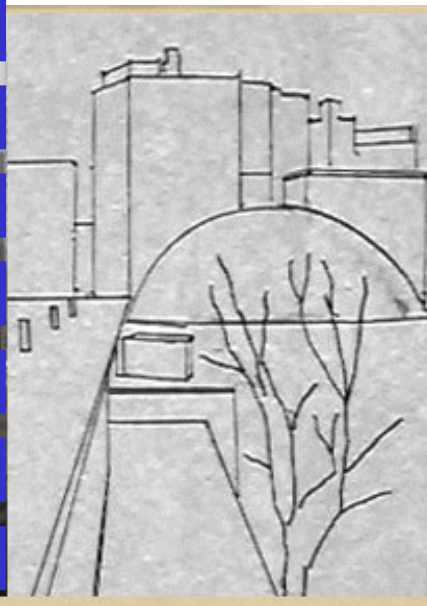
Características principales

2) El software **crea modelos** de la realidad.



La realidad

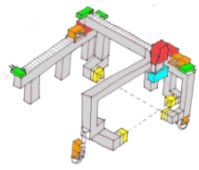
Dominio del
Problema



El modelo

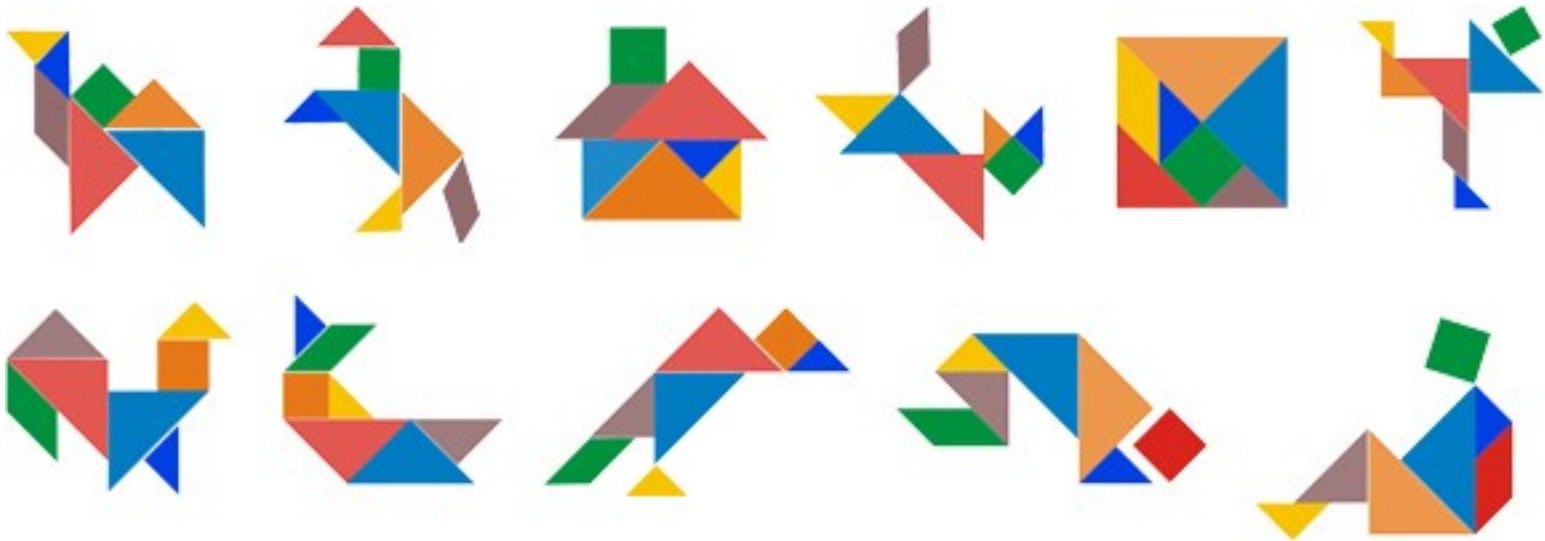
Dominio de la
Solución

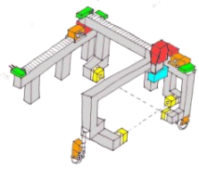
Modelo de funcionamiento
Modelo de la Información
...



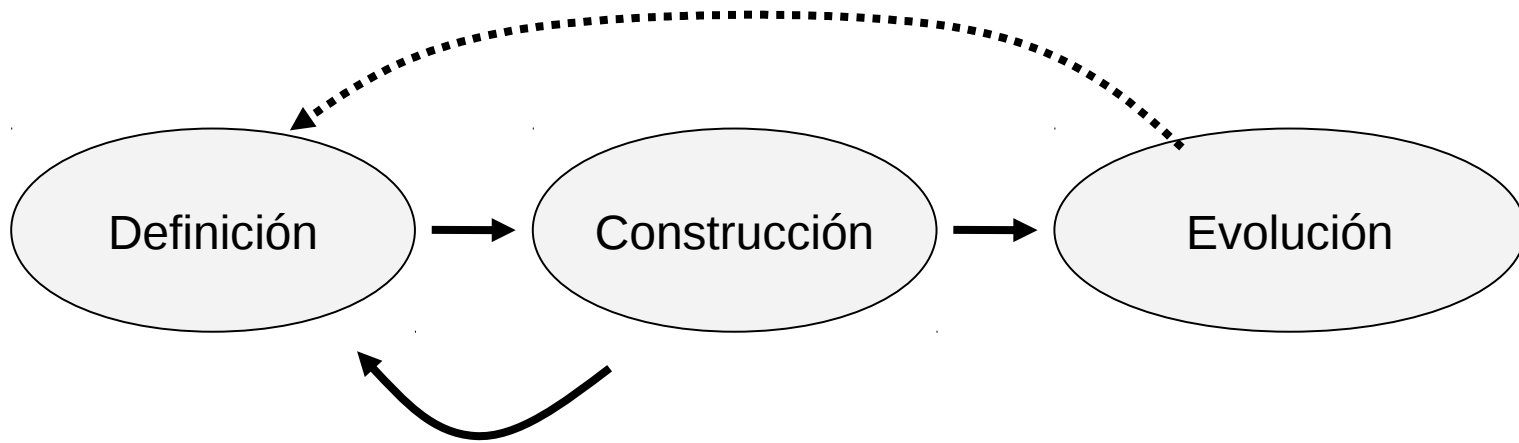
Características principales

3) El software está formado por **múltiples piezas** que deben **encajar perfectamente**.





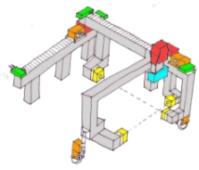
Proceso de producción



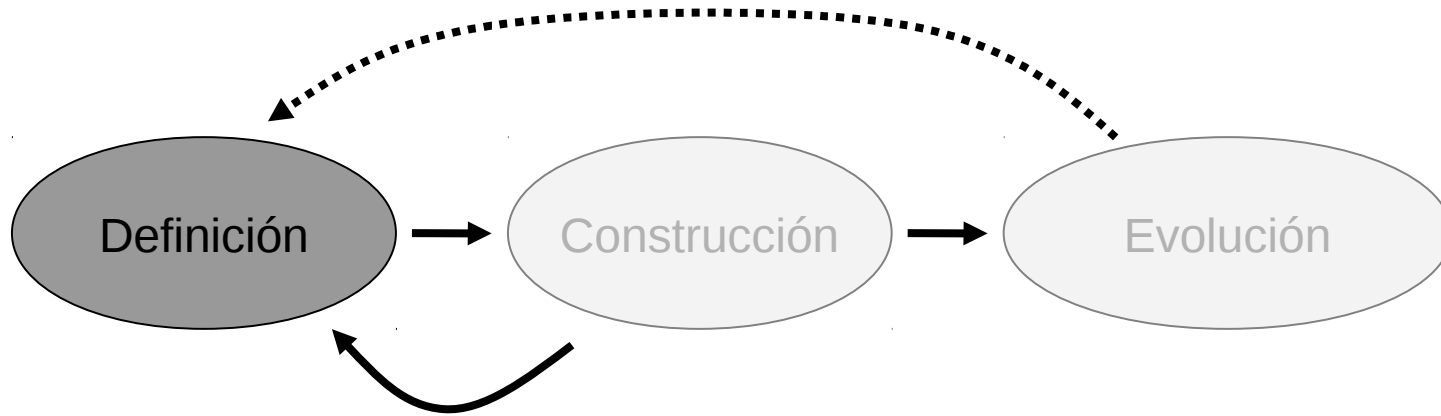
¿Qué?

¿Cómo?

¡Cambios!



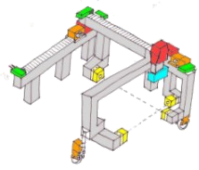
Proceso de producción



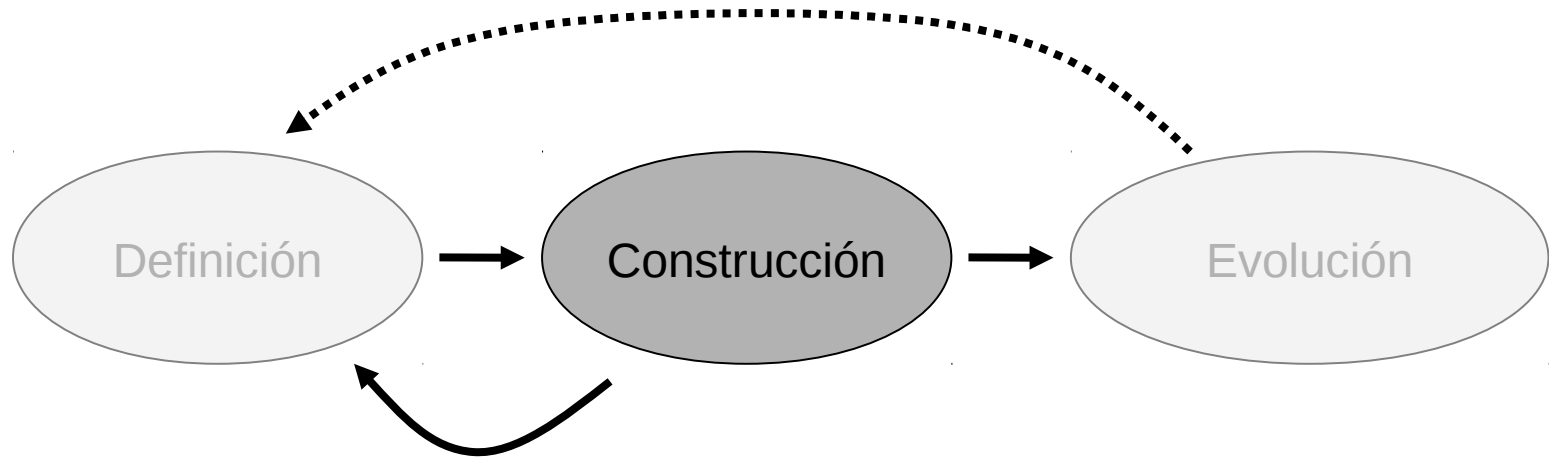
¿Qué vamos a desarrollar?

Tareas a realizar:

- Ingeniería de sistemas.
- Ingeniería de requisitos.
- Planificación de proyectos.

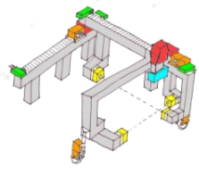


Proceso de producción

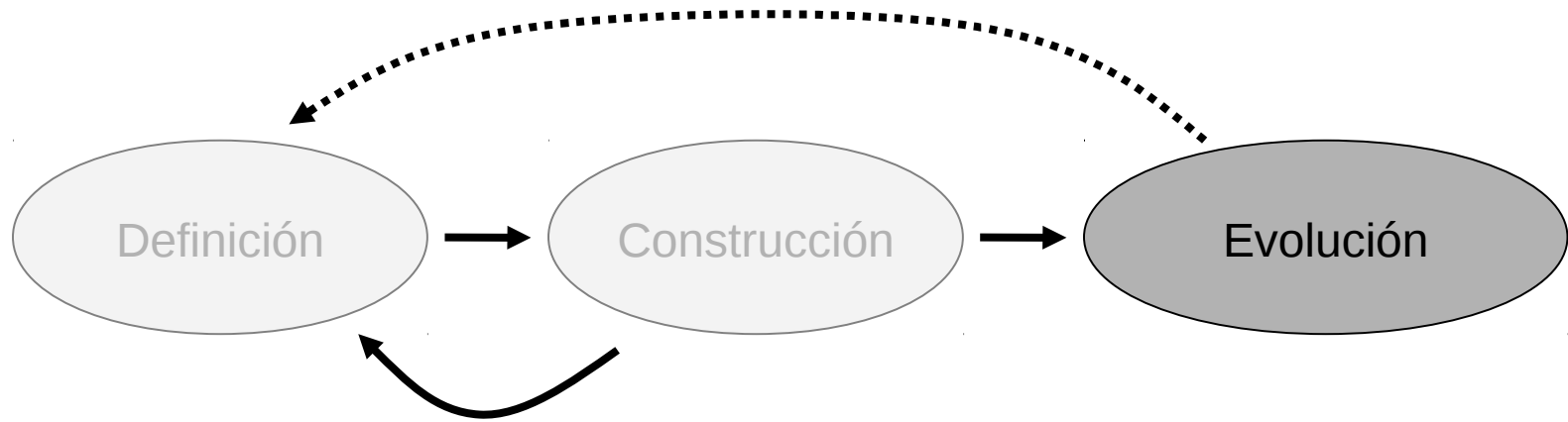


¿Cómo lo vamos a desarrollar?

Tareas a realizar: Diseño del software.
 Generación del código.
 Prueba del software.



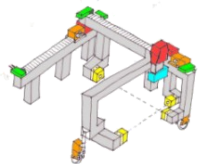
Proceso de producción



¿Qué va a cambiar en el software?

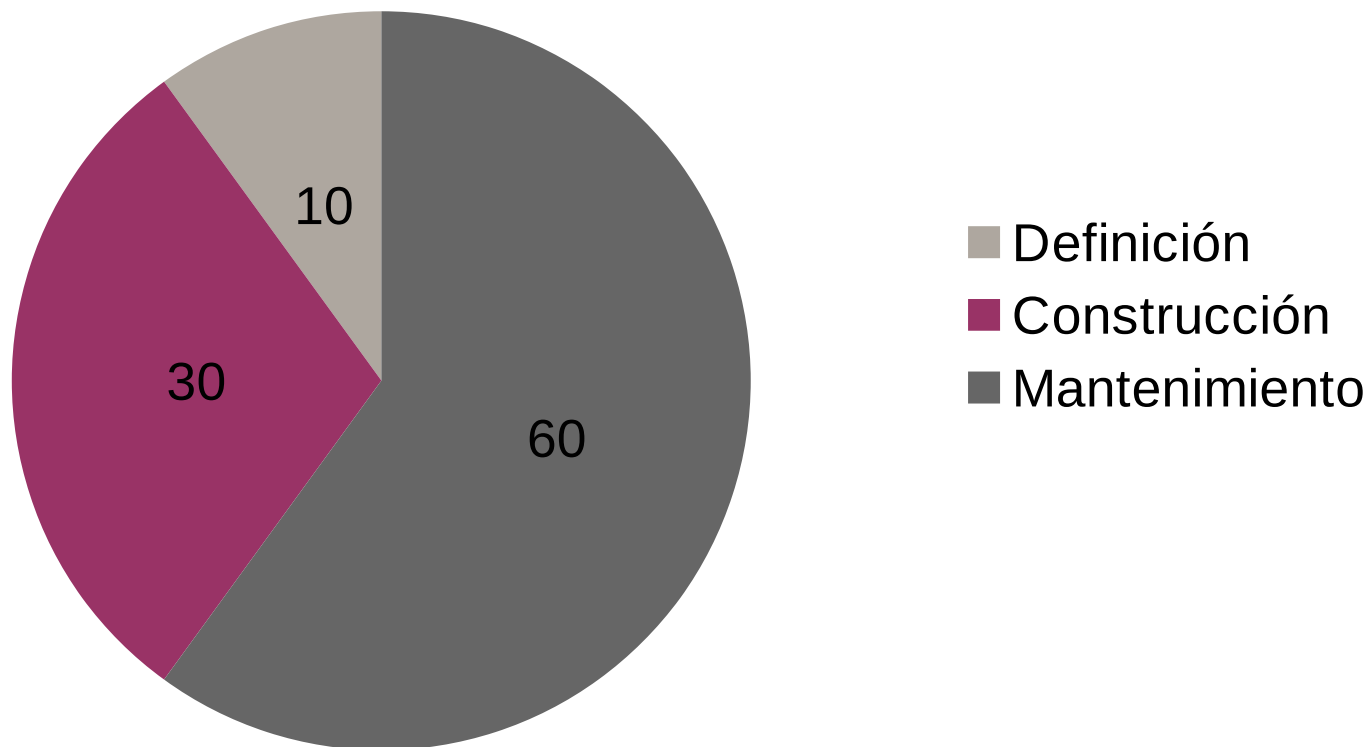
Tareas a realizar:

- Corrección.
- Adaptación.
- Mejora.
- Prevención.



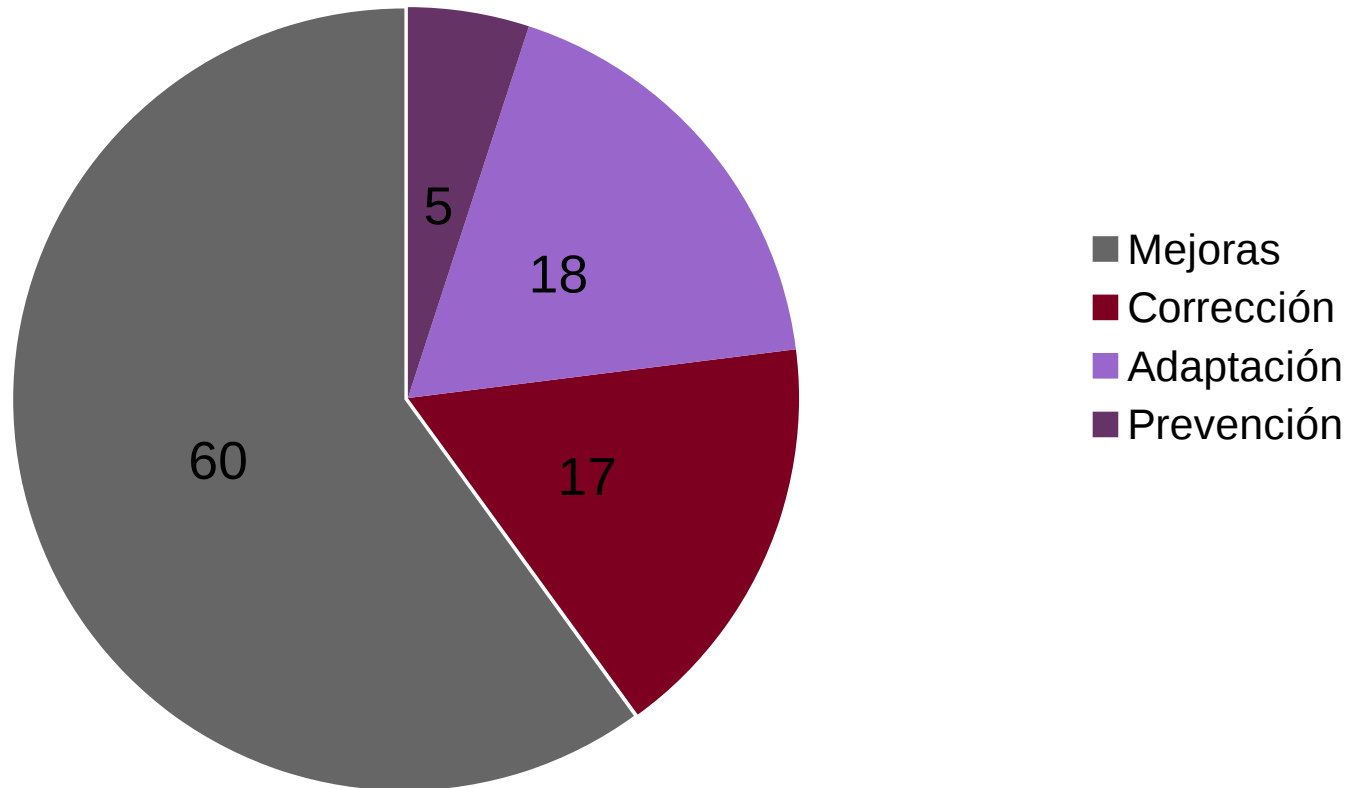
Proceso de producción

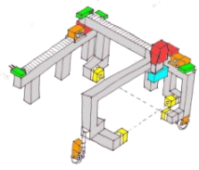
Esfuerzo requerido por cada etapa



Proceso de producción

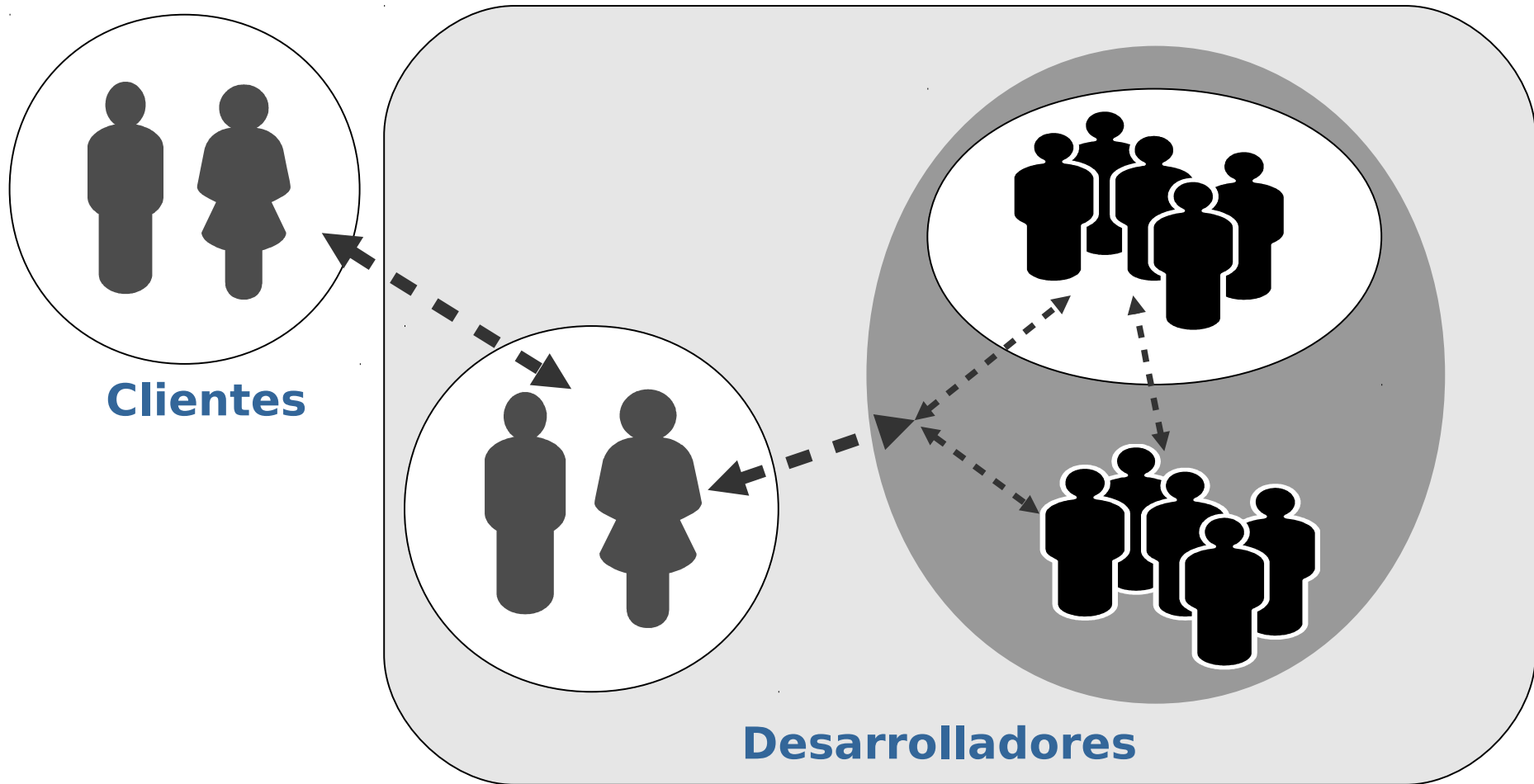
Distribución del esfuerzo en las tareas de mantenimiento

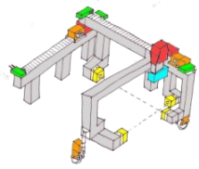




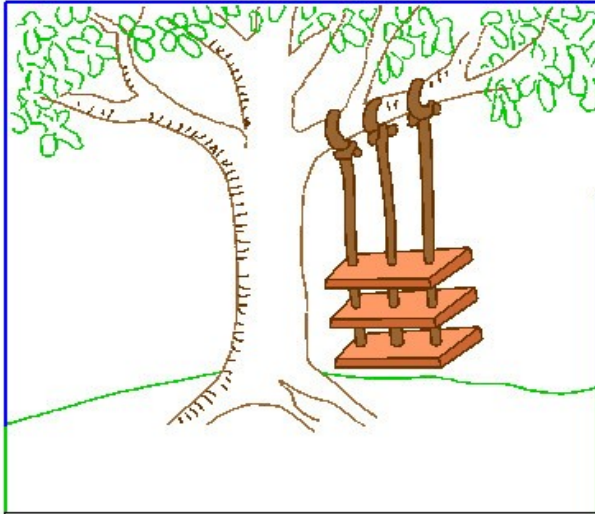
Problemas en el desarrollo

1) Comunicación entre personas.

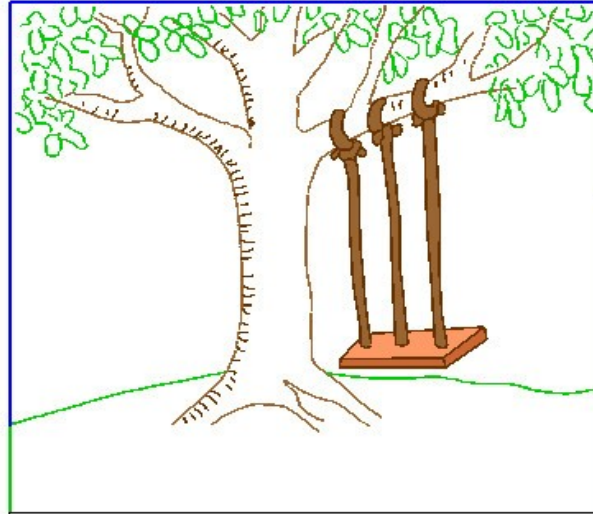




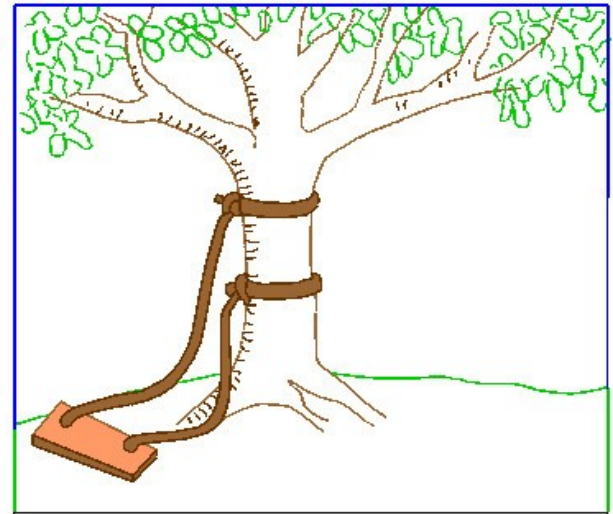
Problemas en el desarrollo



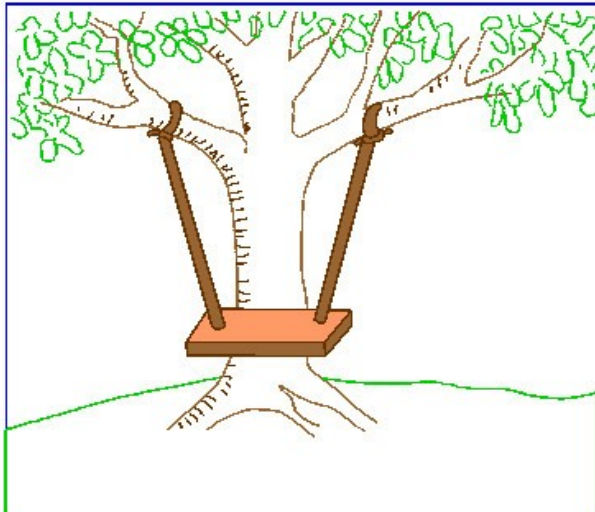
Como fue propuesto el proyecto



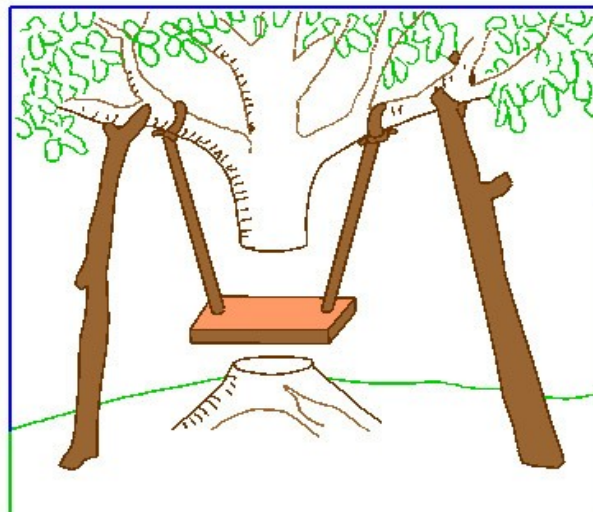
Como fue especificado



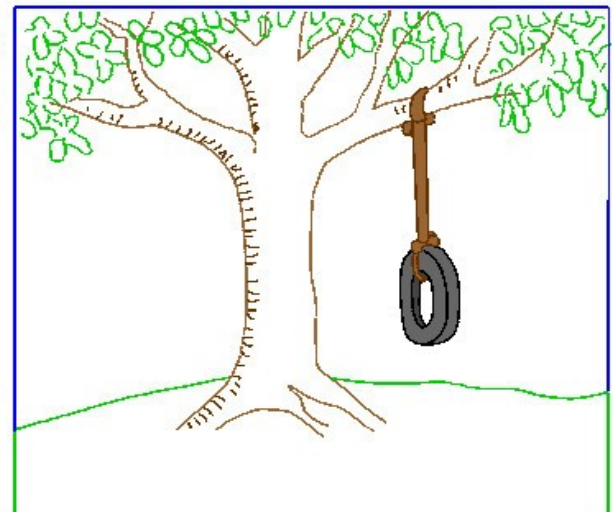
Como fue diseñado



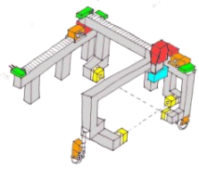
Asi fue programado



Asi quedo instalado



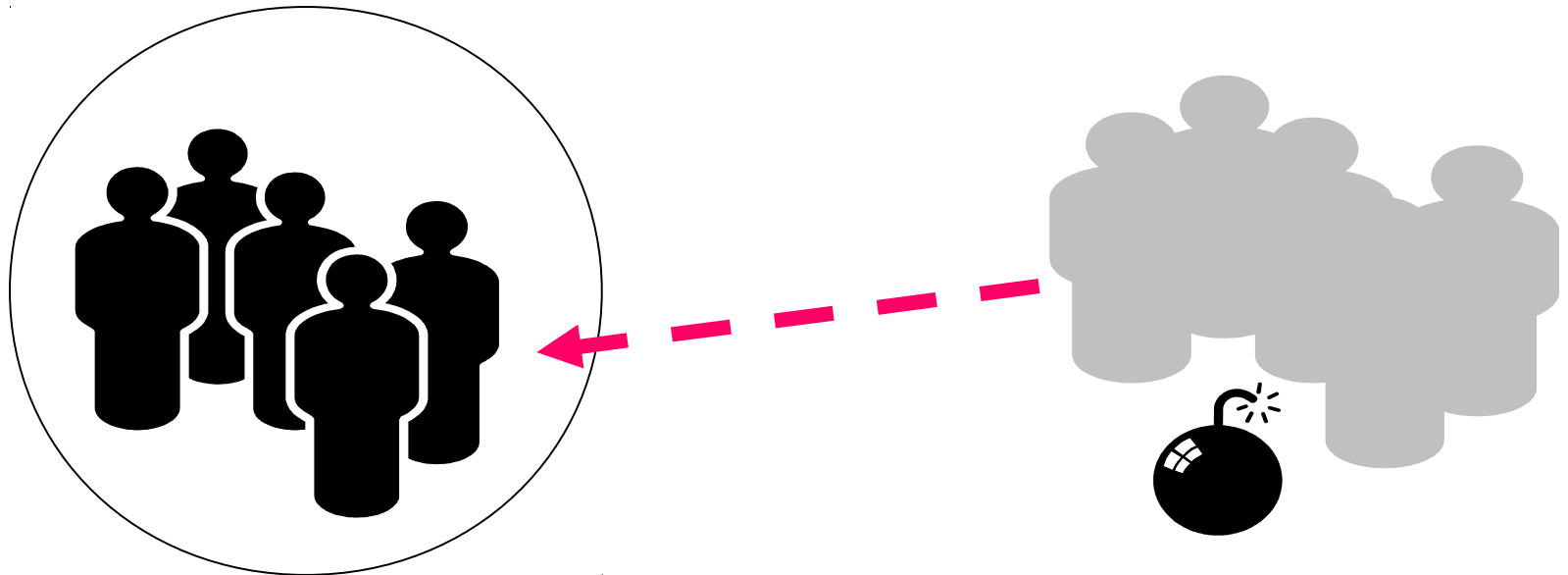
Lo que pidio el usuario



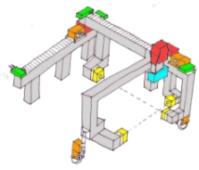
Problemas en el desarrollo

2) Incumplimiento de la **planificación**.

¿**Solución**: *horda mongoliana*?

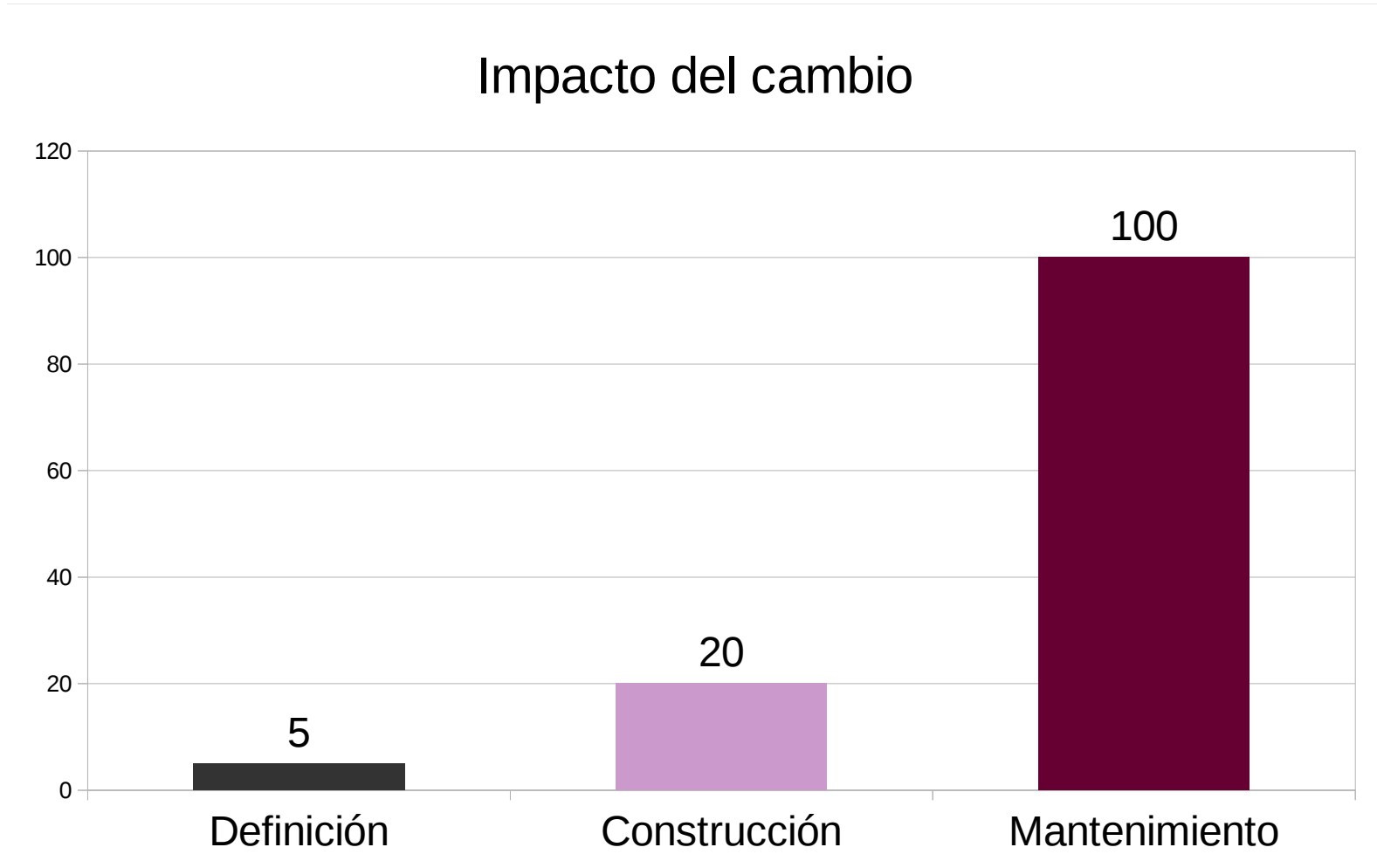


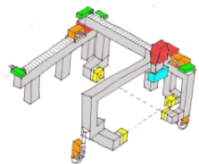
Equipo de desarrollo



Problemas en el desarrollo

3) Incorporar **cambios** en etapas avanzadas





Otros temas de interés

- **Desastres ocasionados por sistemas software**

La mayoría de ellos están ocasionados por pruebas deficientes, mala documentación, diseños pobres o inexistentes, estudio insuficiente del problema, deficientes análisis de riesgos..

- **Ética en el desarrollo de software:**

- **[SOOM11 páginas 14-17]**

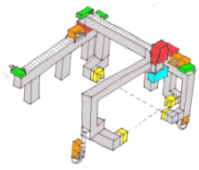
- **Mitos del software:**

- **[PRESS13 páginas 18-20]**

1.2. Concepto de Ingeniería del Software

- ✓ Historia y necesidad de la IS.
- ✓ Definición de IS.
- ✓ Terminología usada en IS.

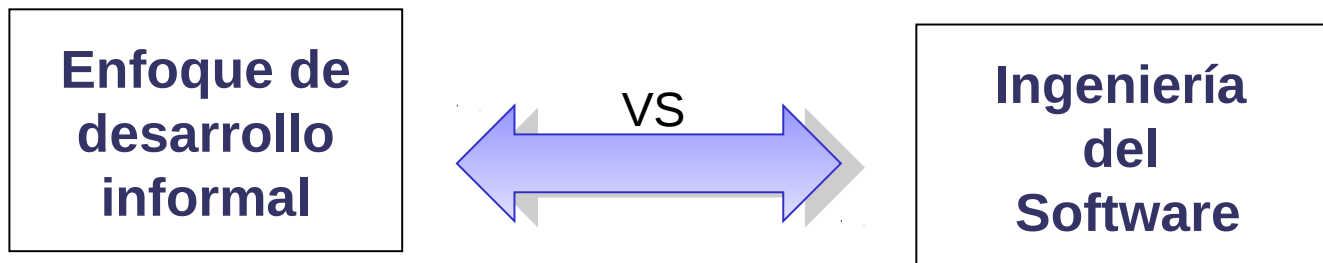
Bibliografía: [PRES13 capítulo 1]

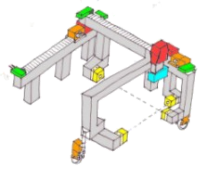


Historia y necesidad de la IS

- Breve historia de la Ingeniería del Software
- **Necesidad** de la Ingeniería del software:
 - Mal funcionamiento (Calidad).
 - Mantenimiento del software existente.
 - Demanda creciente de nuevo software.
 - Adaptación a las nuevas tecnologías.
 - Incremento de la complejidad.

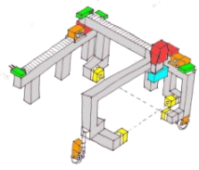
¿Cómo desarrollamos software?





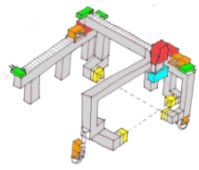
Definición de IS

- Establecimiento de los principios y métodos de la ingeniería a fin de obtener software de modo rentable que sea fiable, y trabaje en máquinas reales (F.L. Bauer, 1972).
- Aplicación práctica del conocimiento científico en el diseño y construcción de programas de computadora, y la documentación asociada y requerida para el desarrollo, operación y mantenimiento del programa (B. Bohem, 1976).
- Estudio de los principios y métodos para el desarrollo y mantenimiento de sistemas software (M.V. Zelkowitz, 1978).



Definición de IS

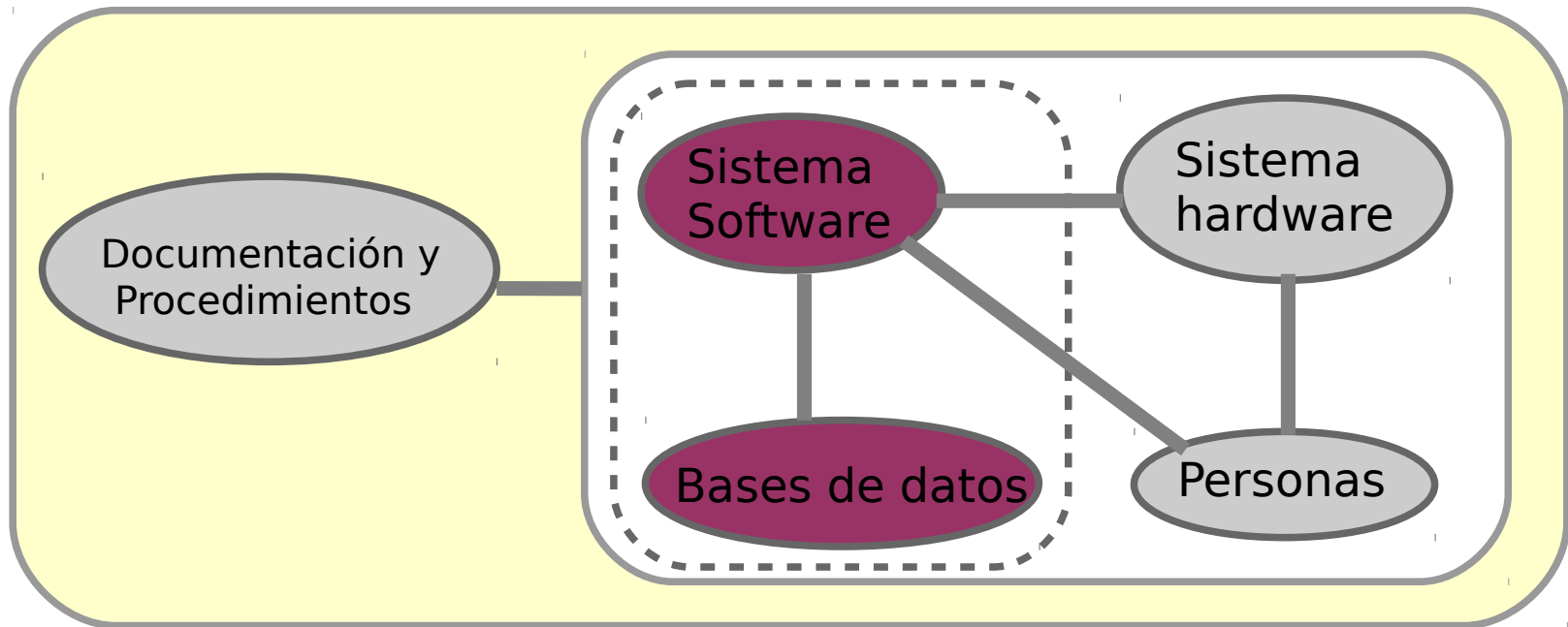
- Aplicación de un **enfoque sistémico**, disciplinado y cuantificable al desarrollo, operación y mantenimiento del software; es decir, aplicación de la ingeniería al software (standard - IEEE, 1993).
- Conjunto de **teorías, métodos e instrumentos** (tecnológicos y organizativos) que permitan construir sistemas software con las características de **calidad** deseadas” (M. Marré, 2002 apuntes de clase).
- Disciplina de ingeniería que se interesa **por todos los aspectos de la producción de software**, desde las primeras etapas de la especificación hasta el mantenimiento del sistema después de su puesta en operación (**I. Sommerville**, [SOMM11, página 7])

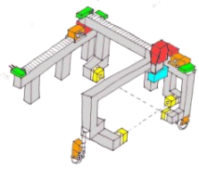


Terminología usada en IS

Sistema: Conjunto de elementos relacionados entre sí y con el medio, que forman una unidad o un todo organizativo.

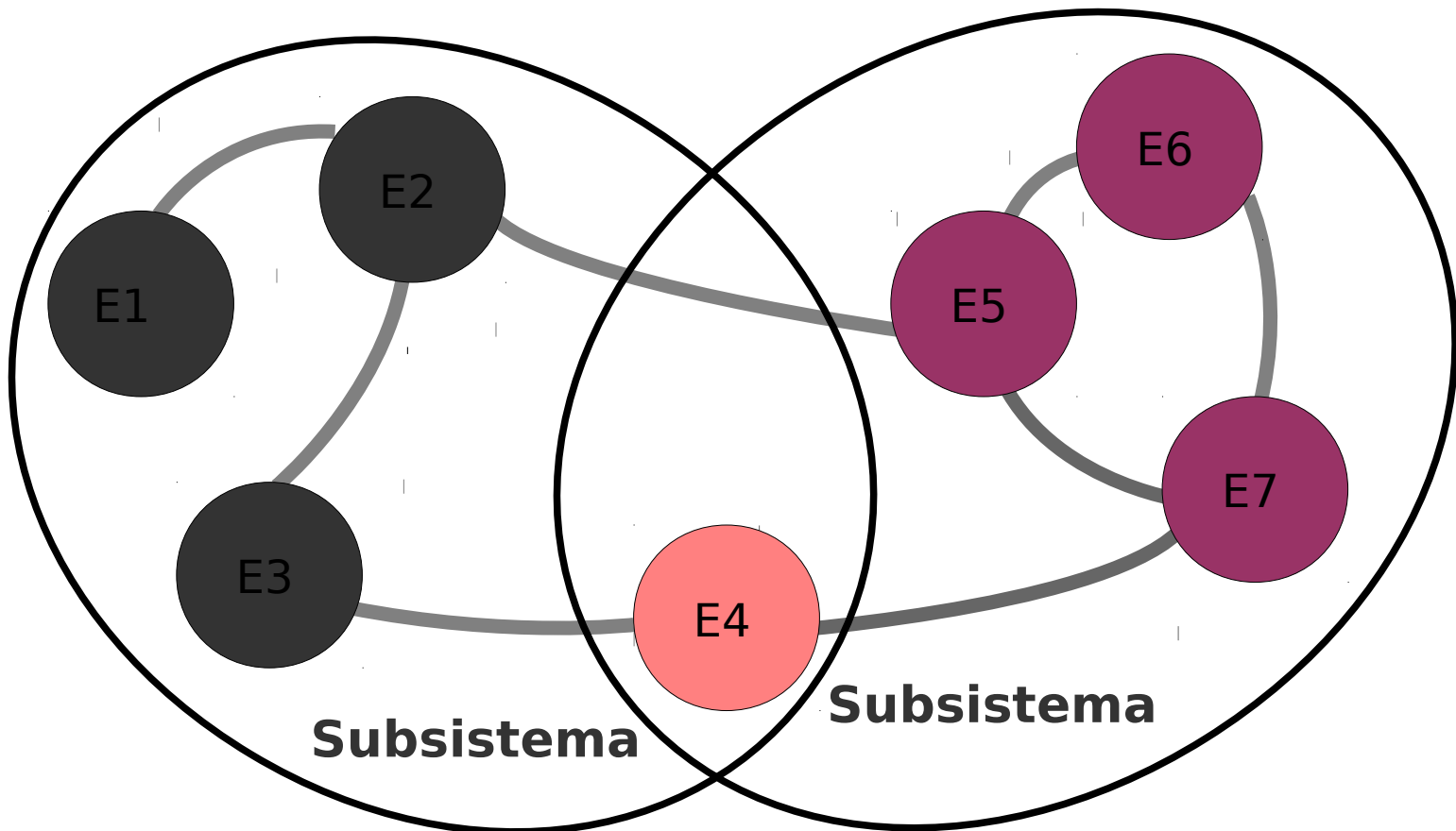
Sistema basado en computadora: Conjunto o disposición de elementos organizados para cumplir una meta predefinida al procesar información.

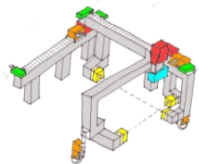




Terminología usada en IS

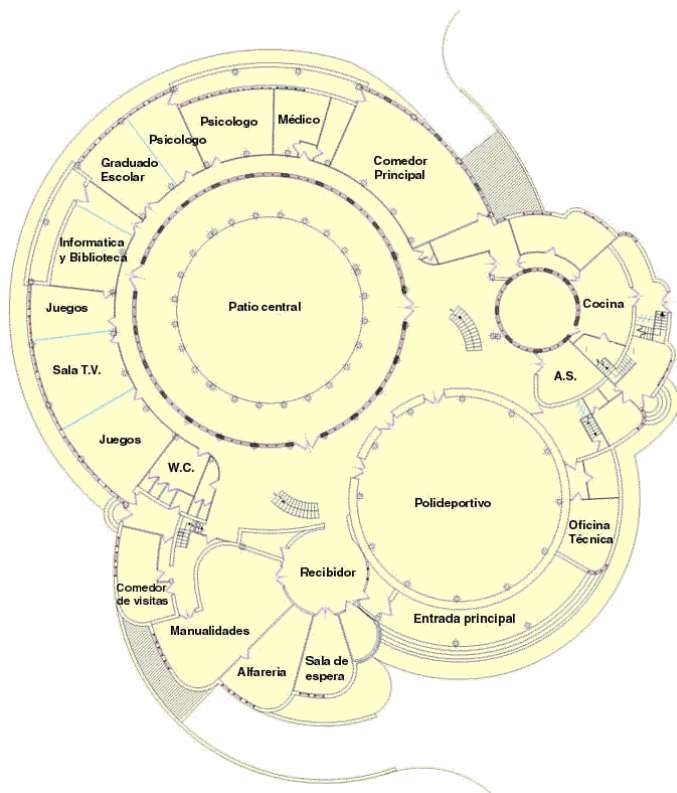
Sistema Software: Conjunto de piezas o elementos software relacionados entre sí y organizados en subsistemas.

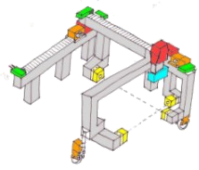




Terminología usada en IS

Modelo: Representación de un sistema en un determinado lenguaje. De un mismo sistema se pueden construir muchos modelos.

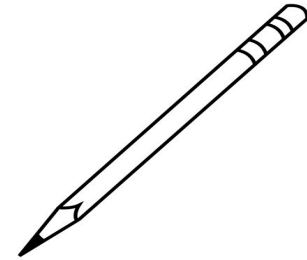




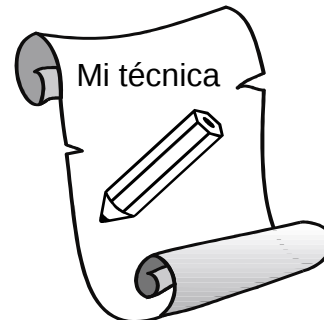
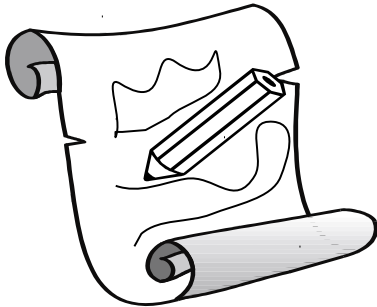
Terminología usada en IS

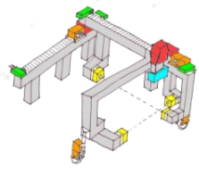
Principio: Elementos adquiridos mediante el conocimiento, que definen las características que debe poseer un modelo para ser una representación adecuada de un sistema.

Herramienta: Instrumentos que permiten la representación de modelos.



Técnica: Modo de utilización de las herramientas.





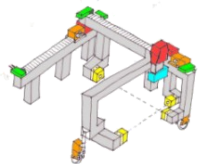
Terminología usada en IS

Heurísticas: Conjunto de reglas empíricas, que al ser aplicadas producen modelos que se adecuan a los principios.

“No usar materiales flexibles para representar la maqueta de un edificio.”

Proceso: Estructura que debe establecerse para la obtención eficaz de un producto de Ingeniería [PRES13 página 12].

Método: Proporcionan la experiencia técnica para elaborar el producto software, se basan en principios fundamentales e incluyen actividades de modelado [PRES13 página 12].



[PRES13 página 12]



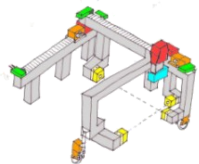
Definición de IS: Estudio de principios, metodologías y herramientas que forman parte de un proceso para facilitar el desarrollo y mantenimiento de sistemas software de calidad.

1.3. Proceso de desarrollo de software

- ✓ Concepto de proceso de desarrollo.
- ✓ Modelo de proceso
 - ✓ Un modelo general de proceso
 - ✓ Tipos de Modelo de proceso:
 - ✓ Modelo en cascada
 - ✓ Modelos de proceso incremental
 - ✓ Modelos de proceso evolutivos
- ✓ Proceso unificado.
- ✓ Desarrollo ágil.

Bibliografía:

[PRES13 capítulos 2 y 3]
[SOMM11 capítulos 2 y 3]
[ARLO05 capítulo 2]
[LARM03 capítulo 2]



Concepto de proceso de desarrollo

Proceso de desarrollo del software: Conjunto de **actividades, acciones y tareas** que se realizan cuando va a crearse un producto o sistema software.

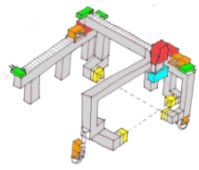
Actividad: busca el logro de objetivos amplios y e independiente del tipo de aplicación a desarrollar y de su complejidad, por ejemplo: planificación del proceso de desarrollo.

Acción: conjunto de tareas que producen un producto importante como resultado, por ejemplo: modelado de la arquitectura del sistema software.

Tarea: objetivo pequeño y bien definido que produce un resultado tangible, por ejemplo: realizar una prueba unitaria.

Este esquema permite enfoques adaptables, basados en un número mínimo de actividades y en que cada equipo de desarrollo defina el conjunto de acciones y tareas que forman parte de cada actividad.

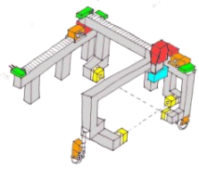
[PRES13 p.12-14]



Concepto de proceso de desarrollo

Tipo de actividades:

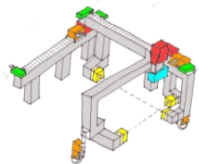
- **Estructurales**: dedicadas a obtener el producto y son:
 - **Comunicación**: Colaboración con el cliente para entender objetivos y requisitos del proyecto.
 - **Planificación**: Definir el plan del proyecto en el que se describen los riesgos probables, los recursos que se requieren, los productos que se obtienen y se programan las actividades, acciones y tareas.
 - **Modelado**: Representación mediante modelos del sistema propuesto junto con la solución o soluciones apropiadas.
 - **Construcción**: Generación de código y su prueba.
 - **Despliegue**: Entrega al consumidor y evaluación por parte de éste, la cual sirve de retroalimentación para el equipo de desarrollo.



Concepto de proceso de desarrollo

Tipo de actividades (continuación):

- **Sombrilla:** Aplicadas a lo largo de todo el proceso y se dedican a:
 - Seguimiento y control del proyecto.
 - Administración del riesgo.
 - Aseguramiento de la calidad.
 - Revisiones técnicas.
 - Mediciones de parámetros del proceso.
 - Administración de la configuración.
 - Administración de la reutilización.
 - Preparación y producción del producto de trabajo.



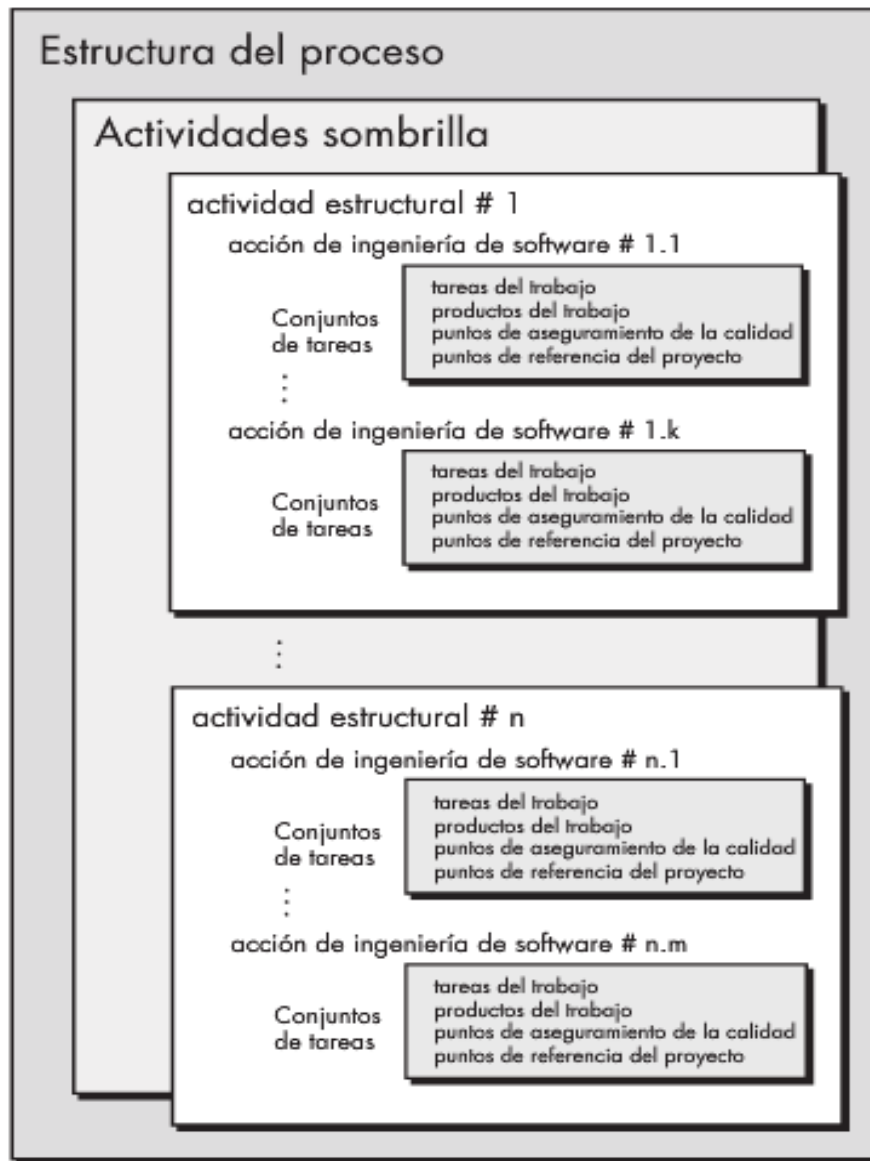
Modelo de proceso: Modelo general de proceso

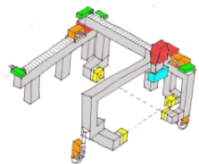
Estructura del proceso

Cada una de las actividades, acciones y tareas se encuadran dentro de una estructura que definen su relación con el proceso y entre ellas.

Un esquema general de la estructura de proceso sería el mostrado a la derecha

[PRES13 página 27]





Modelo de proceso: Modelo general de proceso

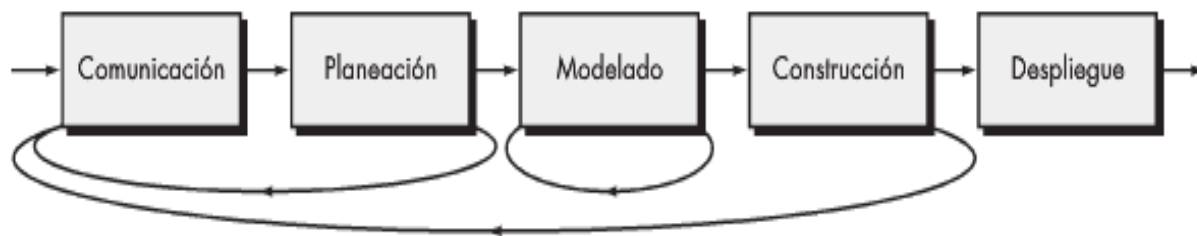
Flujo del proceso

Describe la manera en la que se organizan las actividades estructurales, acciones y tareas dentro de cada uno con respecto a la secuencia y el tiempo.

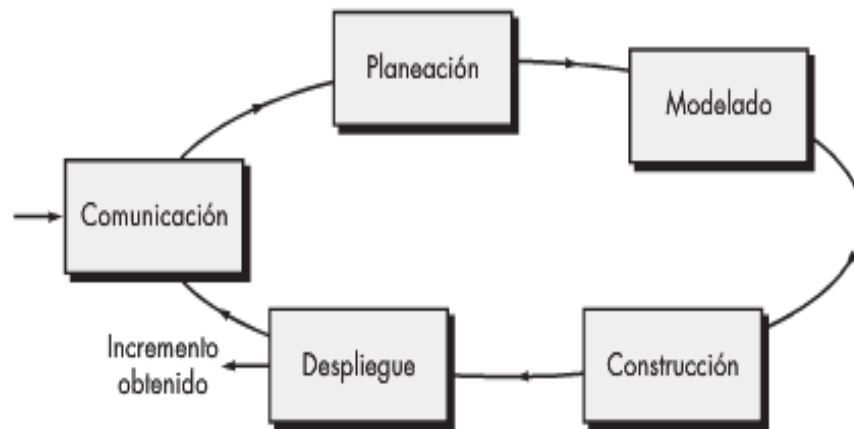
En el siguiente esquema se puede ver distintos flujos de proceso.



a) Flujo de proceso lineal

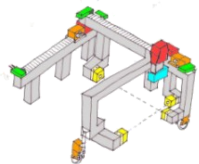


b) Flujo de proceso iterativo



c) Flujo de proceso evolutivo

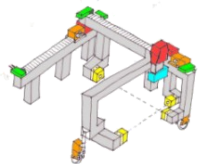
[PRES13 página 28]



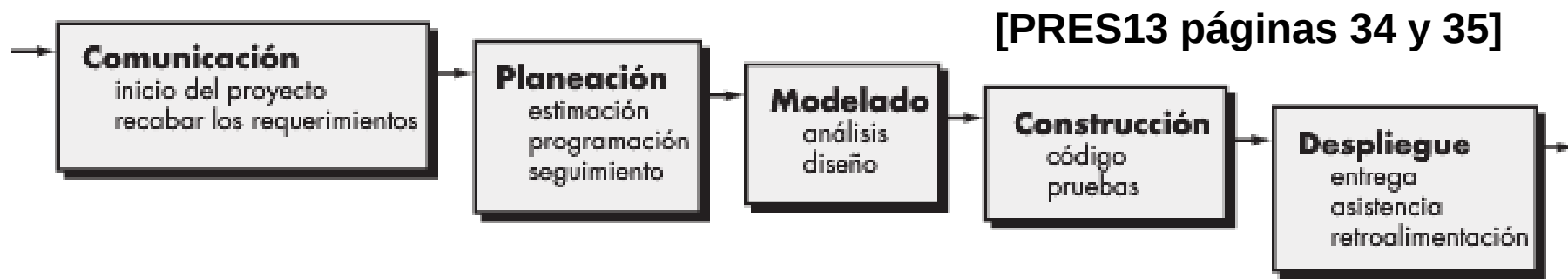
Modelo de proceso: Modelo general de proceso

Acciones y tareas de las actividades estructurales

- **Obtención de requisitos:** Indagación para obtener información respecto a qué es lo que se desea que realice el software.
- **Estimación y planificación del Proyecto:** Estimar el tiempo y los costes de desarrollo del software.
- **Análisis de requisitos:** Análisis del problema a resolver. Documento en el que se dice qué debe hacer el sistema software.
- **Diseño:** Búsqueda de la solución. Descripción de los componentes, sus relaciones y funciones que dan solución al problema.
- **Implementación:** Traducción del diseño a un lenguaje de programación entendible por una máquina.
- **Prueba del software:** Revisión y validación de todo el código que se va desarrollando.
- **Evaluación y aceptación:** Evaluación del producto y aceptación en su caso por parte de los interesados en el sistema software.
- **Entrega y asistencia:** Sistema operando y asistencia para su funcionamiento correcto.

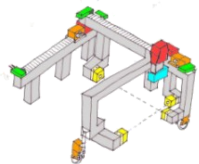


Tipo de modelos de proceso: Modelo en cascada

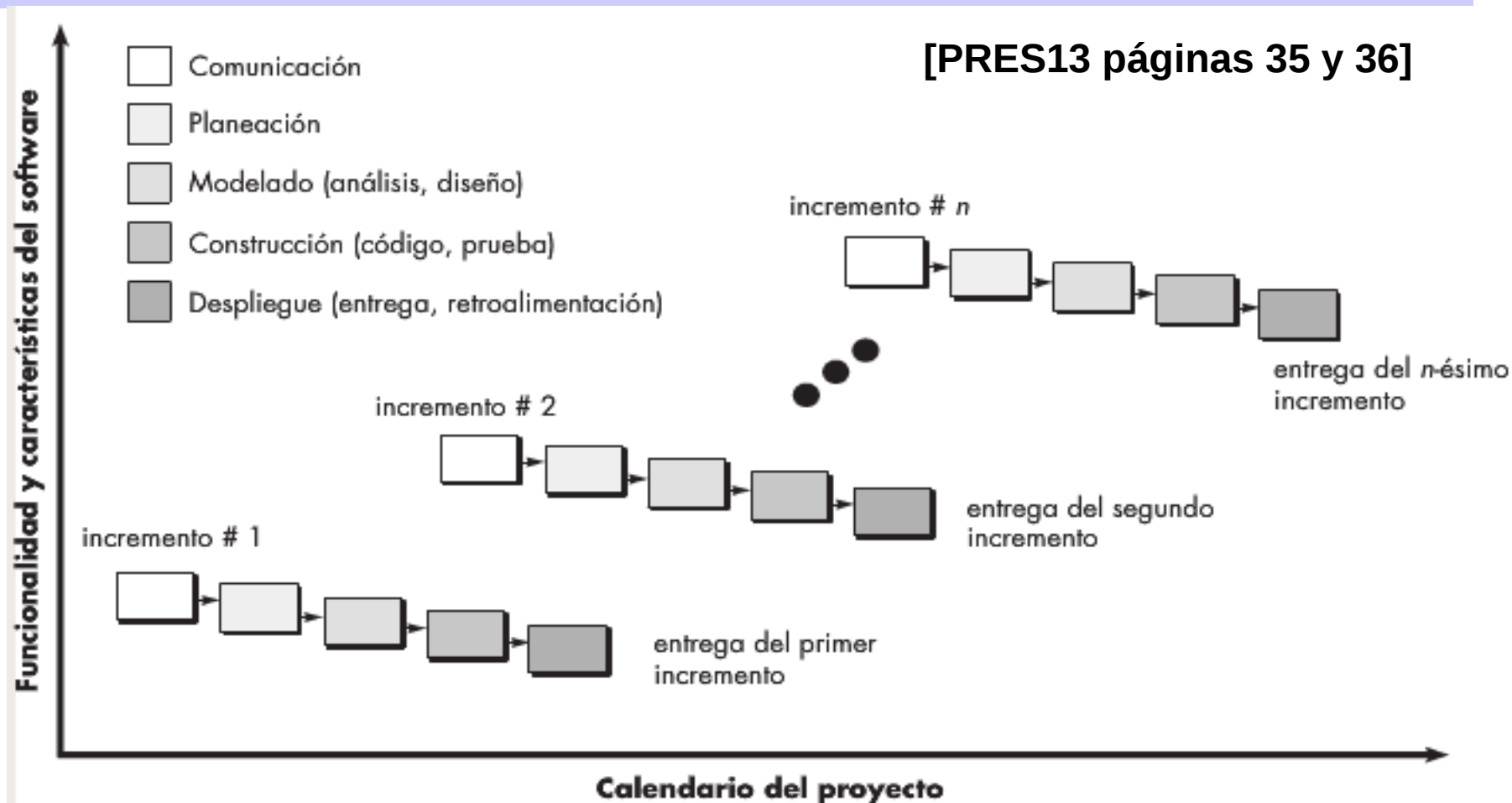


Características generales:

- Estructura secuencial y flujo de proceso lineal.
- Problemas que presenta:
 - Los proyecto reales difícilmente se adecuan a este modelo de proceso.
 - Dificultad para expresar por parte del cliente todos los requisitos al principio del proyecto.
 - Poca comunicación con cliente/usuario, hasta las etapas finales no hay un ejecutable que se pueda evaluar.

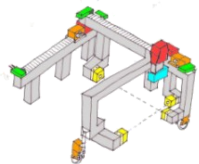


Tipo de modelos de proceso: Modelo incremental



Características generales:

- Estructura secuencial y flujo de proceso lineal y paralelo entre incrementos. (Ver en PRESS13 las demás características)



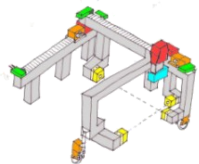
Tipo de modelos de proceso: Modelos Evolutivos

[PRES13 páginas 36-40]

Los **modelos evolutivos** son a su vez iterativos y nacen ante la exigencia de tiempo de entrega muy limitado, la necesidad de facilitar la incorporación de cambios y de satisfacer al usuario/cliente. En cada iteración del proceso se obtiene un producto terminado y operativo.

Características generales:

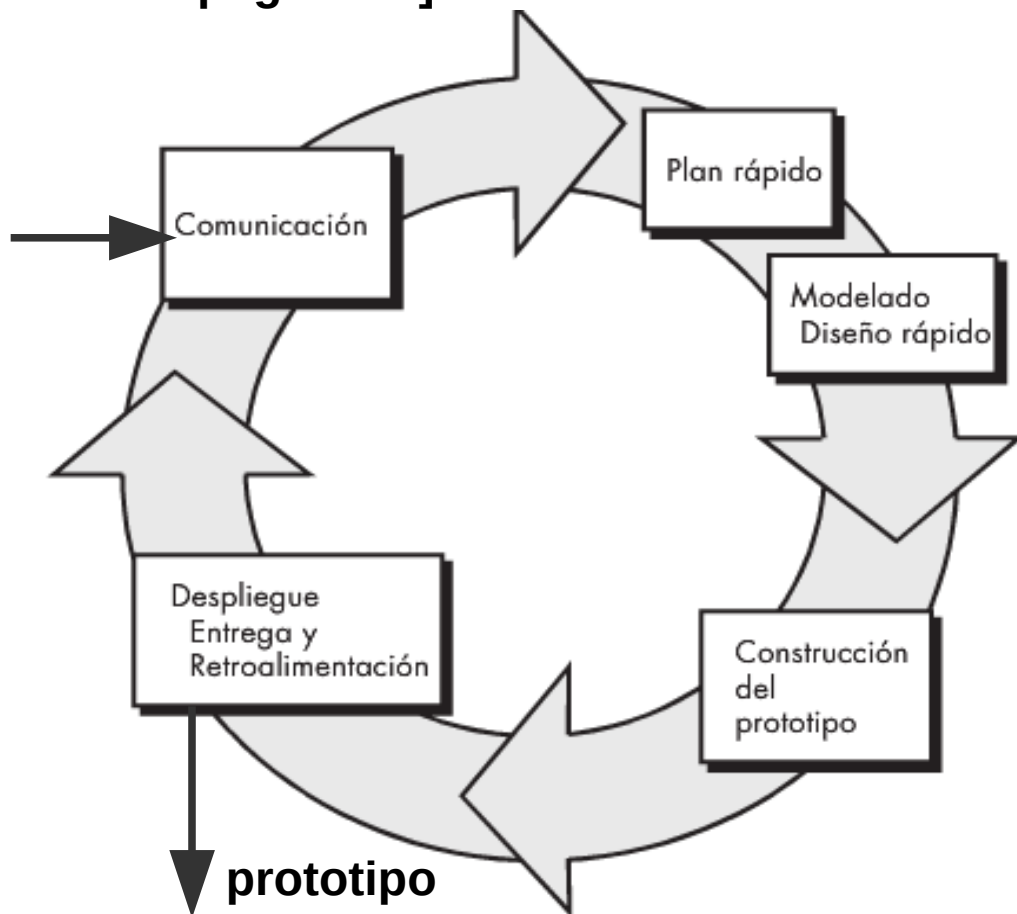
- Afrontan los riesgos altos (técnicos, de requisitos, de usabilidad...) tan pronto como sea posible.
- Retroalimentación temprana por parte del usuario.
- Manejo de la complejidad (pasos cortos y sencillos).
- El conocimiento adquirido durante una iteración de la evolución puede ser usado en el resto de iteraciones.
- Involucra continuamente al usuario (evaluación, retroalimentación y obtención y refinamiento de requisitos).



Tipo de modelos de proceso: Modelos Evolutivos

Modelo de prototipos

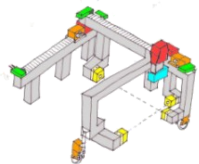
[PRES13 página 37]



Prototipo: Representación limitada de un producto, que se utiliza para probar las opciones de diseño y para comprender mejor el problema y sus posibles soluciones. Producto de funcionamiento limitado en cuanto a su capacidad, confiabilidad o eficiencia. (¿Calidad del producto?)

Tipos de prototipos:

- **Prototipo evolutivo:**
Como producto final
- **Prototipo desechable:**
usados dentro de otros modelos de proceso



Tipo de modelos de proceso: Modelos Evolutivos

Modelo de prototipos

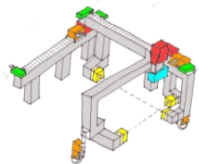
Se usa para:

- Facilitar la obtención y validación de requisitos (desechable).
- Estudios de viabilidad (desechable).
- Propuestas de soluciones (diseños) alternativas (desechable).
- En casos muy concretos como producto final (evolutivo).

Presentan todas las características de los modelos evolutivos, pero con los siguientes **inconvenientes**:

- Crea falsas expectativas por parte del cliente/usuario (desechable).
- Decisiones de diseño del prototipo que pasen a formar parte del producto final (evolutivo).

Su **uso vendrá determinado** por: tipo y complejidad de la aplicación, características del cliente y la disponibilidad de herramientas para su construcción.



Tipo de modelos de proceso: Modelos Evolutivos

Modelo en espiral de Boehm

[PRES13 páginas 39 y 40]

Planificación

Estimación
Búsqueda de alternativas
Análisis de riesgo

Modelado

Análisis y diseño

Construcción,

Código
Prueba

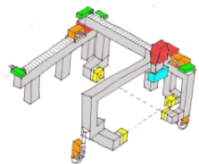
Despliegue

Entrega
Evaluación del cliente
Retroalimentación

Comunicación

Puntos de entrada

Inicio



Tipo de modelos de proceso: Modelos Evolutivos

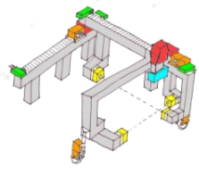
Modelo en espiral de Boehm

Característica (además de la de los procesos evolutivos):

- Centrado en el **análisis de riesgo**, haciendo uso de construcción de prototipos para su estudio.
- La espiral puede continuar una vez finalizado todo el proceso y entregado el producto para llevar a cabo la etapa de mantenimiento.
- Es un enfoque adecuado para el desarrollo de sistemas a **gran escala**.

Inconvenientes:

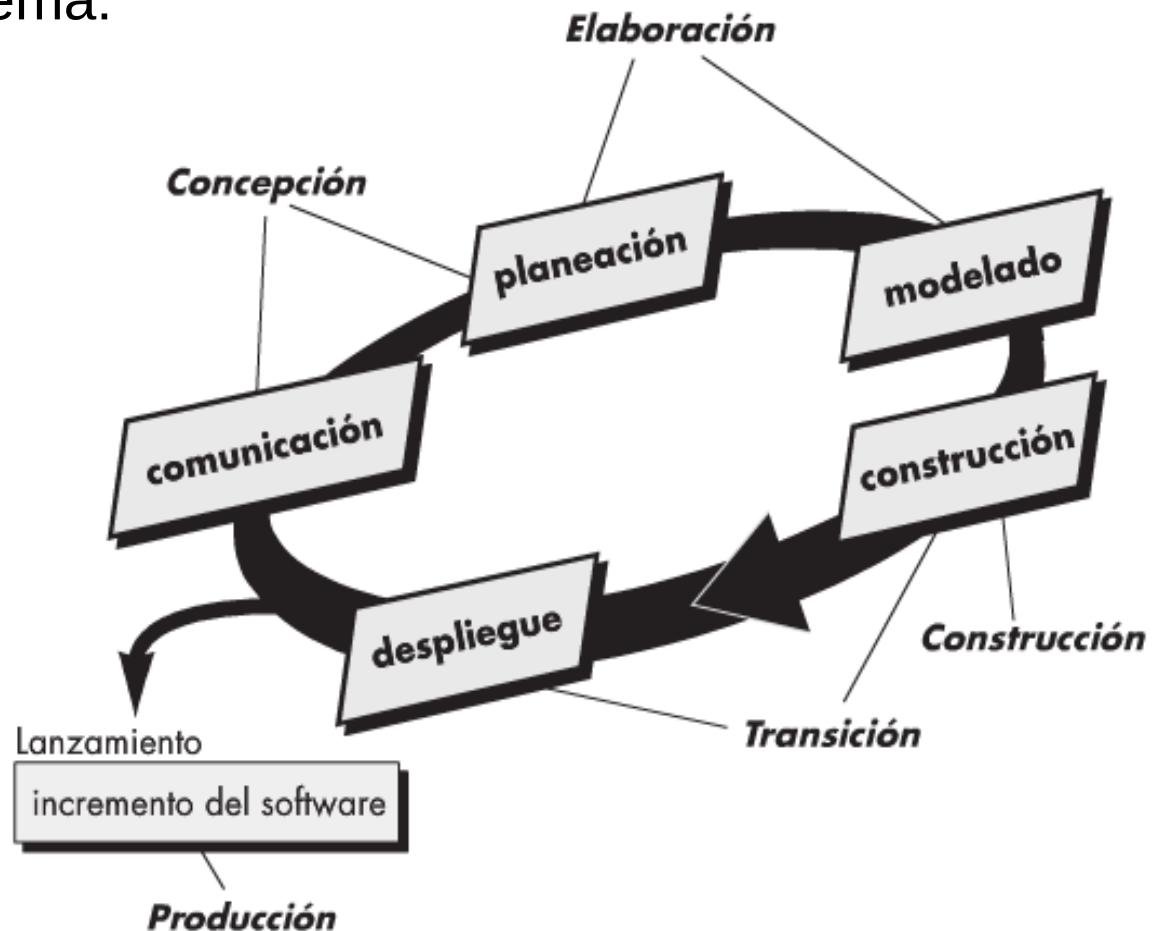
- Modelo de **proceso predictivo** no adaptable a la complejidad ni al tipo de sistema.
- Requiere un equipo de desarrollo con **gran experiencia en análisis de riesgo**.

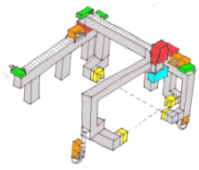


Proceso Unificado

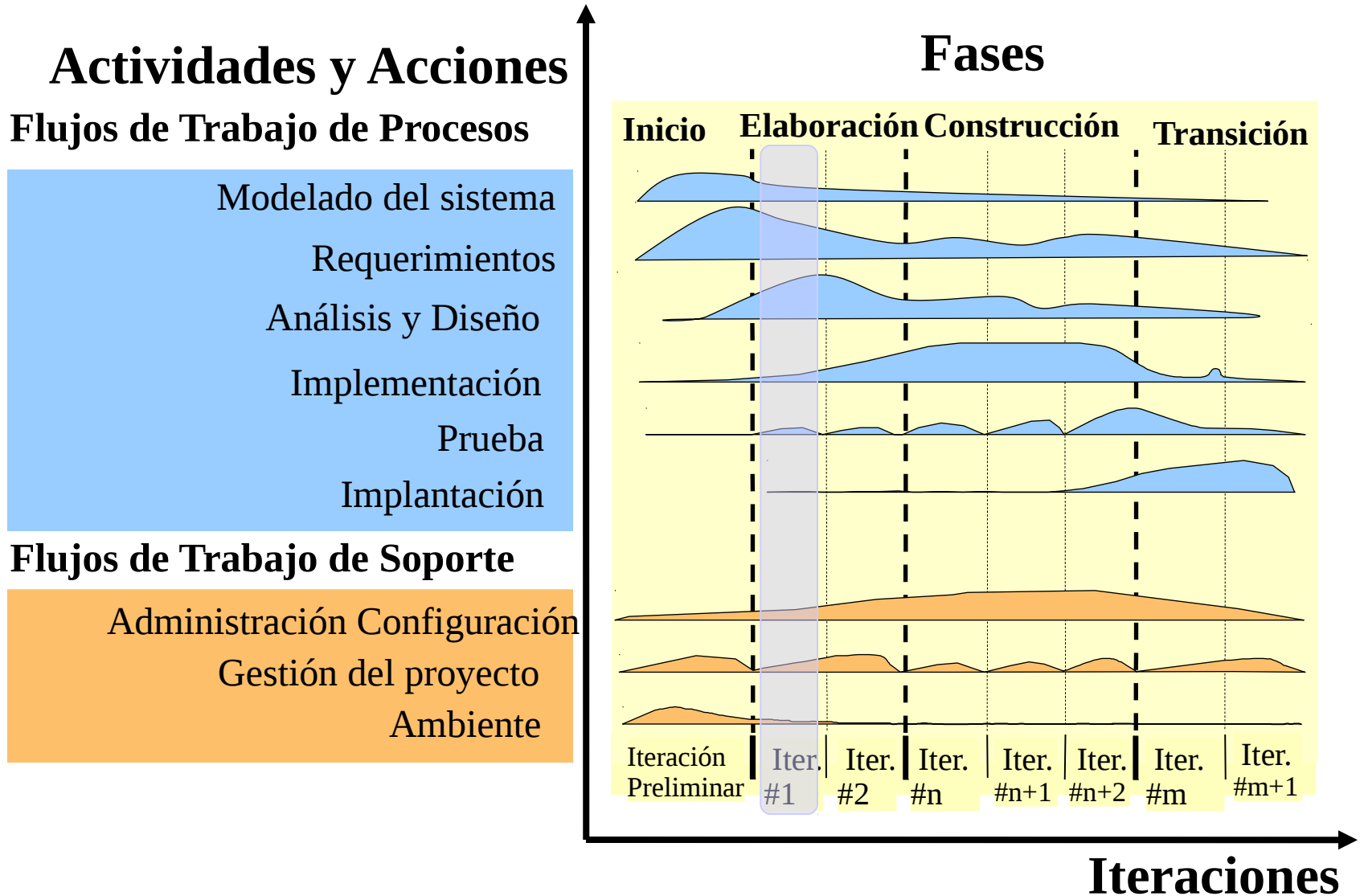
Modelo de proceso evolutivo y compuesto por cuatro fases o **etapas**: Inicio o Concepción, Elaboración, Construcción y Transición, que se reparten entre las actividades estructurales como puede verse en el siguiente esquema:

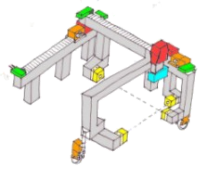
[ARLO05 capítulo 2]
[LARM03 capítulo 2]
[PRES13 páginas 47-49]





Proceso Unificado

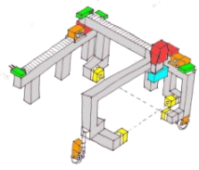




Proceso Unificado

Características (además de todas la de los modelos de proceso evolutivos):

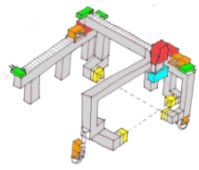
- En un modelo de **proceso adaptable** a la complejidad y al tipo de sistema [LARM03 páginas 22 y 23].
- Centrado en la **arquitectura**, mostrando y decidiendo los distintos aspectos arquitectónicos que presenta un sistema software en etapas tempranas, para que sirvan de base a las demás.
- Dirigido por **casos de uso**, desarrollándose en cada iteración determinados casos de uso. Eligiendo para iteraciones tempranas los casos de uso que determinan la arquitectura.



Proceso Unificado

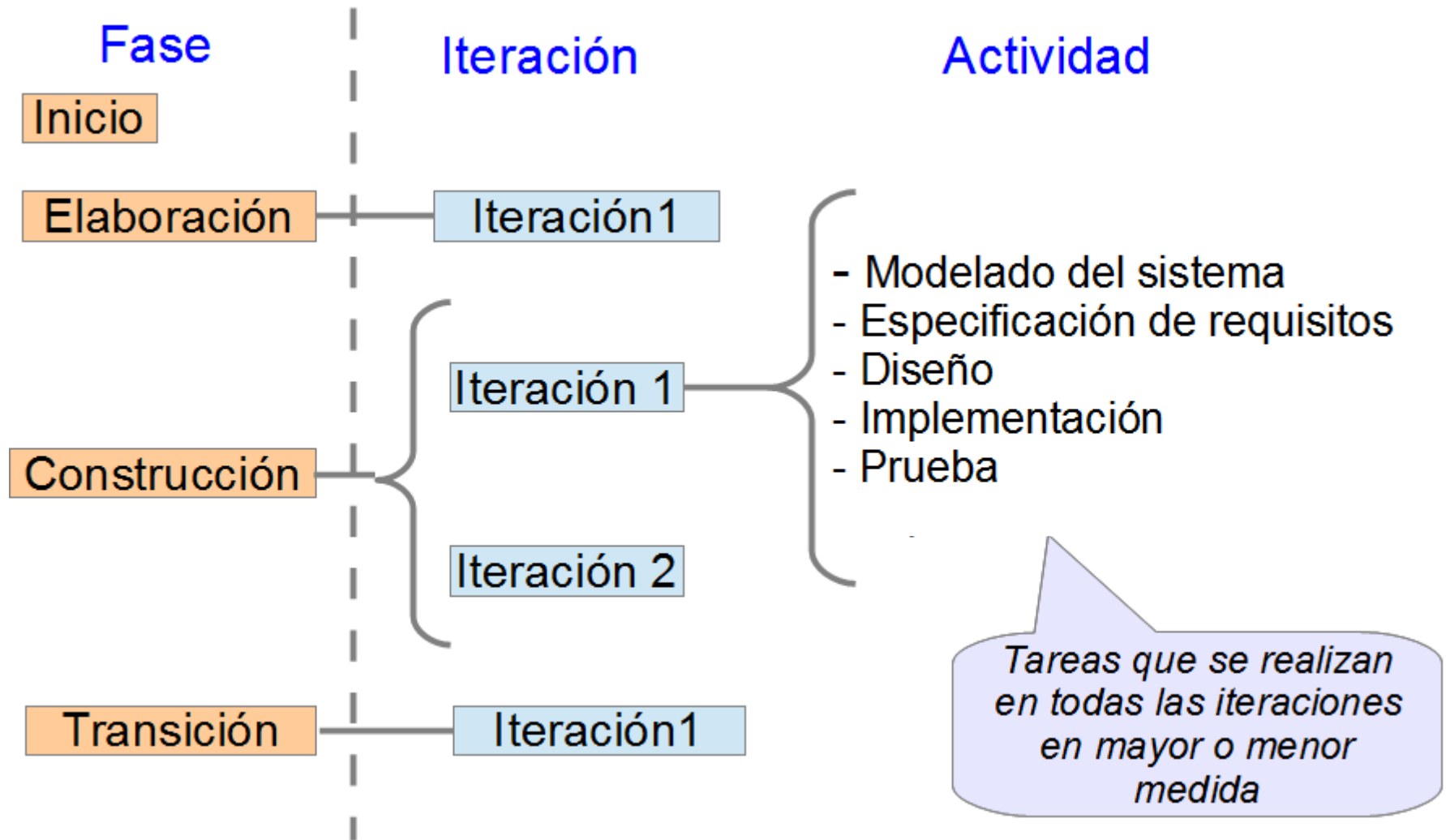
Acciones y tareas en cada fase:

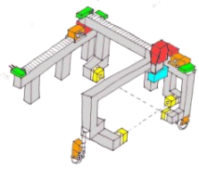
- **Inicio:** Estudio de viabilidad, alcance, objetivos y planificación del proyecto, análisis de riesgos, determinación de los requisitos fundamentales del sistema y propuesta de una arquitectura determinada.
- **Elaboración:** Ajustes de la planificación del proyecto y desarrollo completo de la arquitectura básica sobre la que se asentará la fase de construcción.
- **Construcción:** Se completa los modelos de requisitos y diseño de la elaboración, se implementan los elementos necesarios para completar el sistema y se realizan las pruebas de los distintos elementos que se van terminando, se integran y se hacen pruebas de aceptación por parte de usuario.
- **Transición:** Asegurarse que el sistema cumple con los requisitos especificados (pruebas por los usuarios) y tareas relacionadas con el despliegue de la estructura general de proceso, para preparar lo necesario para su lanzamiento al mercado.



Proceso unificado

Ejemplo concreto de PU para un **sistema simple**





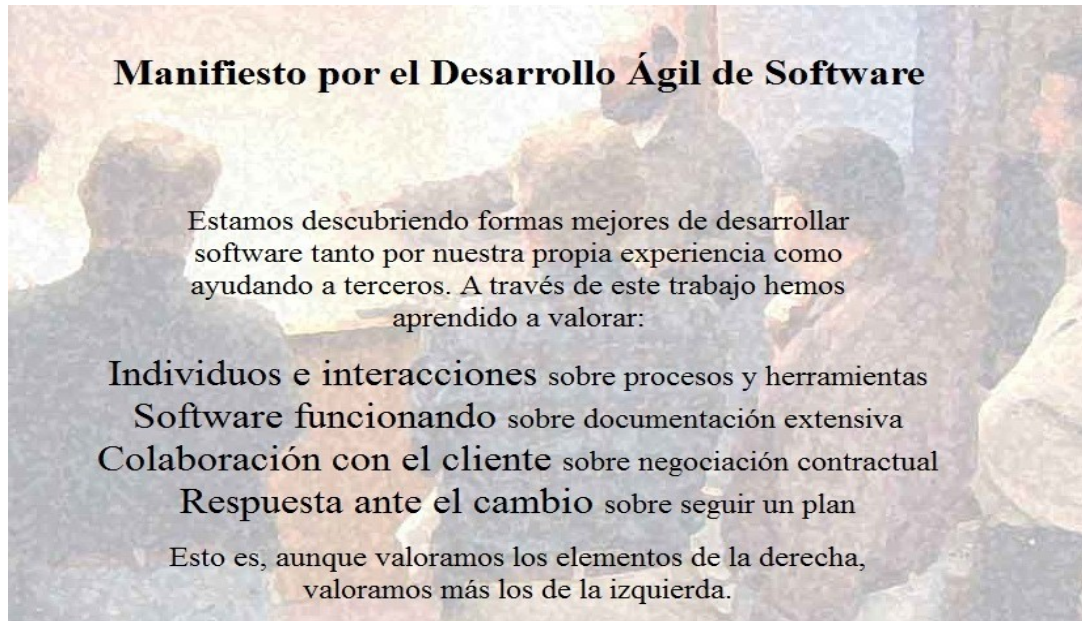
Desarrollo Ágil

[PRES13 capítulos 3] y [SOMM11 capítulos 3]

Muchos proyectos generan menos valor del esperado...

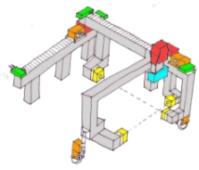
Snowbird, Utah (USA) feb 2001. ¿Por qué tantos proyectos de desarrollo de software no se terminan a tiempo, cuestan más que lo presupuestado originalmente, tienen problemas de calidad serios y generan menor valor que el esperado?

Se preguntaron 17 expertos, que después elaboraron el **Manifiesto Ágil**



Kent Beck Mike Beedle
Arie van Bennekum Alistair
Cockburn Ward
Cunningham Martin Fowler
James Grenning Jim
Highsmith Andrew Hunt
Ron Jeffries Jon Kern
Brian Marick Robert C.
Martin Steve Mellor Ken
Schwaber Jeff Sutherland
Dave Thomas

Ver el siguiente enlace: **Manifiesto**

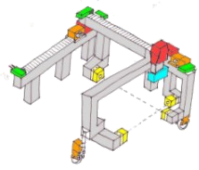


Características:

- Proceso iterativo e incremental: Evolutivo
- Entregas frecuentes
- Trabajo en equipo
- Autonomía del equipo de desarrollo
- Revisiones y reuniones retrospectivas frecuentes

Ver siguiente artículo: [Métodos Ágiles](#)





Desarrollo Ágil

Beneficios:

- Desarrollo guiado por valor
- Mejor manejo de riesgos e incertidumbres
- Mejora la productividad

Métodos y técnicas:

- Scrum ([Más Información](#))
- XP (Extreme Programming) ([Más Información](#))
- Programación en Parejas
- TDD (Test Driven Development) ([Más Información](#))