



Facultad de Ciencias de la Computación

INTRODUCCION A LA DISCIPLINA COMPUTACIONAL

Unidad 3

Ingenieria de Software

Objetivos



- Definir la Ingeniería de Software y explicar su importancia.
- Discutir los conceptos de producto de software y proceso de software.
- Explicar la importancia de la visibilidad de los procesos.
- Introducir la noción de responsabilidad profesional.

Ingeniería de Software



- Las economías de los países desarrollados dependen en gran parte del software.
- Mas y más sistemas son actualmente controlados por software.
- La Ingeniería de Software concierne a teorías, métodos y herramientas para el desarrollo profesional de software.
- El gasto en La Ingeniería de Software, representa un alto porcentaje del PIB de los países desarrollados.

Costos del Software



- Los costos del software a menudo dominan al costo del sistema. El costo del software en un PC es a menudo mas caro que la PC.
- Cuesta mas mantener el software que desarrollarlo. Para sistemas con una larga vida, este costo se multiplica.
- La Ingeniería de Software concierne a un desarrollo efectivo en cuanto a costes del software.

Principios de la Ingeniería del Software



- Haz de la calidad la razón de trabajar.
- Una buena gestión es más importante que una buena tecnología.
- Las personas y el tiempo no son intercambiables.
- Seleccionar el modelo de ciclo de vida adecuado.
- Entregar productos al usuario lo más pronto posible.
- Determinar y acotar el problema antes de escribir los requisitos.
- Realizar un diseño.
- Minimizar la distancia intelectual.

Principios de la Ingeniería del Software

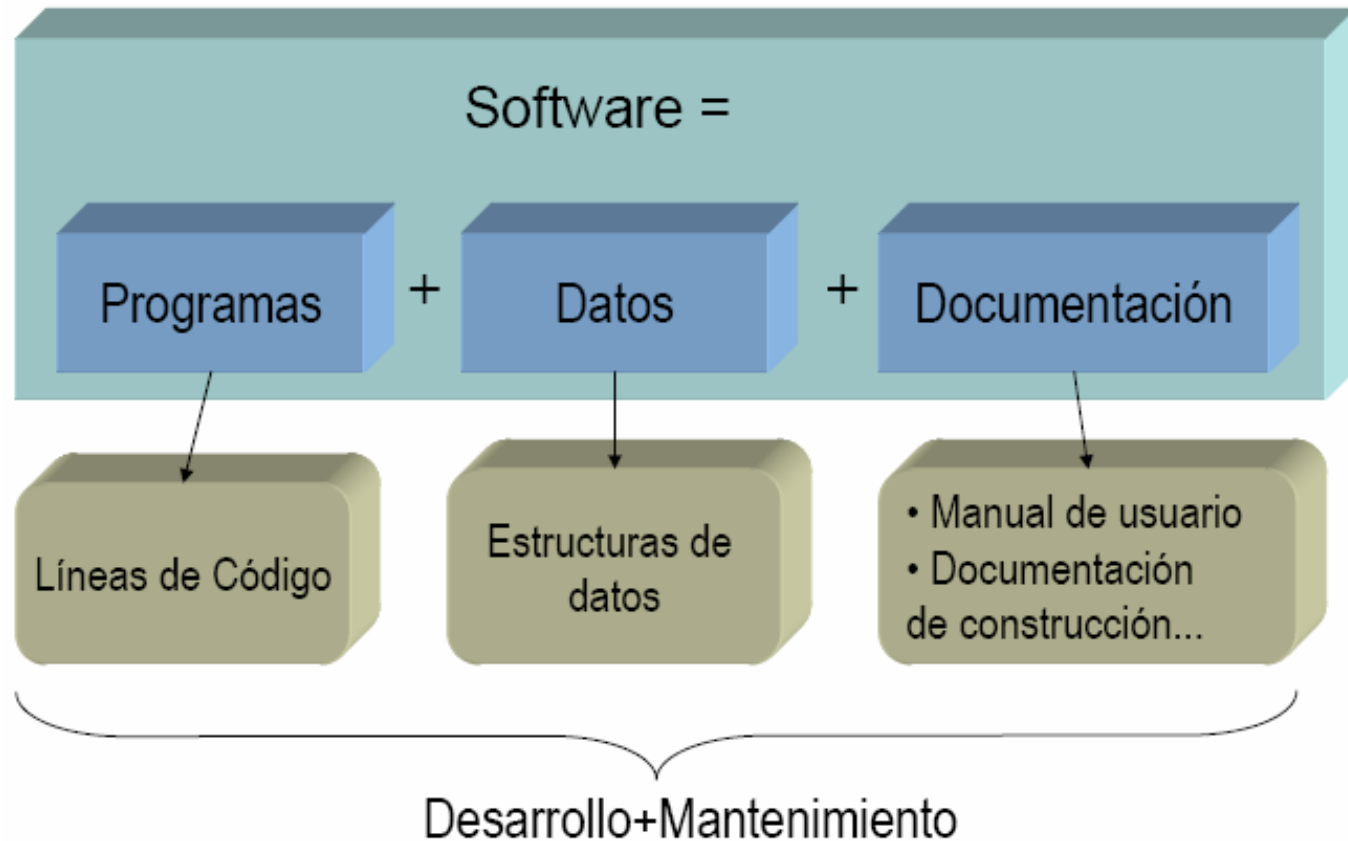


- Documentar.
- Las técnicas son anteriores a las herramientas.
- Primero hazlo correcto, luego hazlo rápido.
- Probar, probar y probar (** incluye inspecciones **).
- Introducir las mejoras y modificaciones con cuidado.
- Asunción de responsabilidades.
- La entropía del Software es creciente.
- La gente es la clave del éxito.



Pero que es el
Software??

Software: Definición



Características del Software

- Es un elemento lógico y no físico
- Es desarrollado, fabricado
- Se deteriora
- No hay piezas de repuesto
- Se construye a la medida



Ingeniería de Software

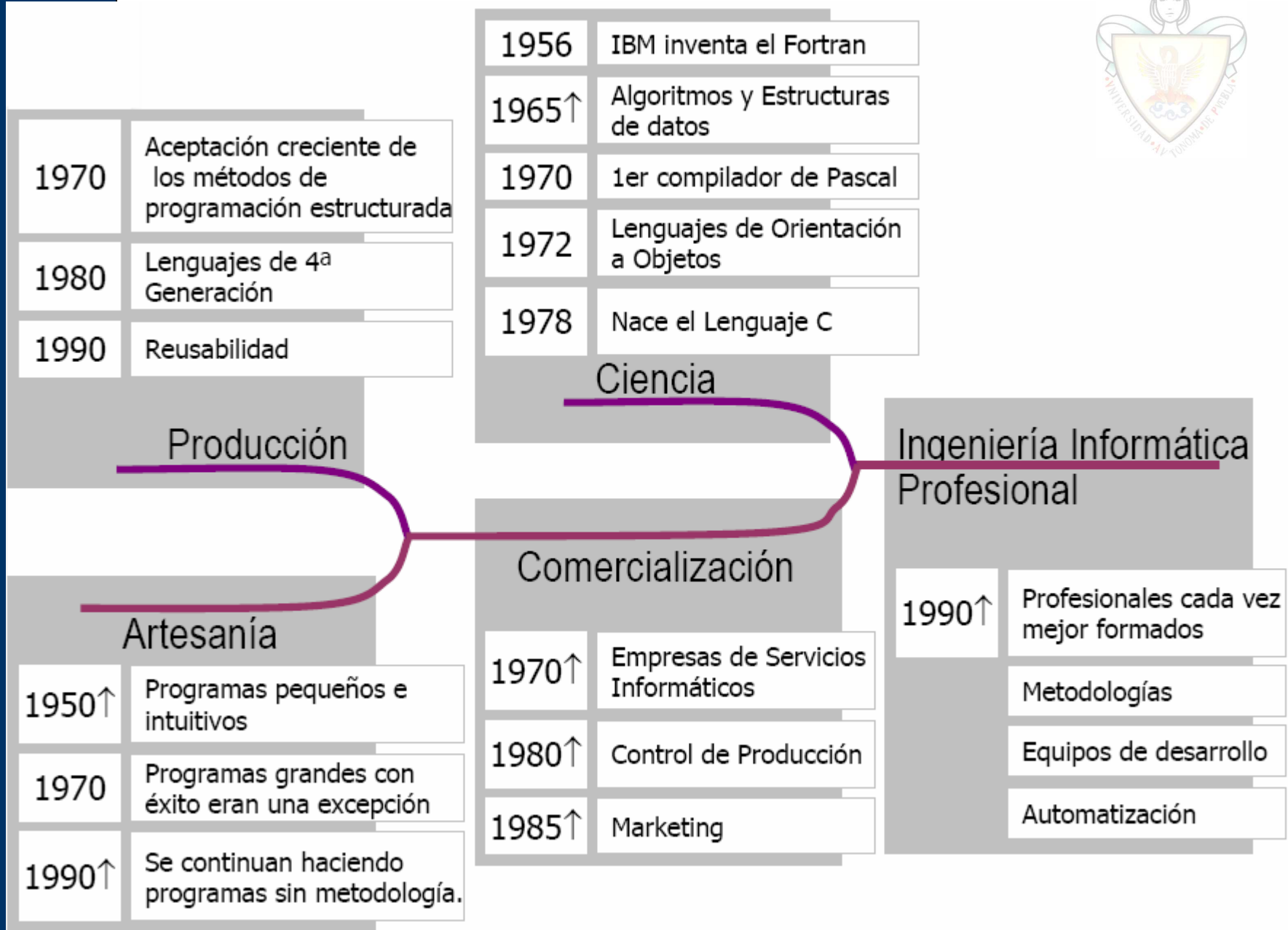


Características del Software

Un sistema software debe ser...

- Fácil de mantener.
 - Construido y documentado para permitir cambios.
- Fiable.
 - Debe hacer lo que se espera de él, no debe fallar más a menudo de lo que se acordó en la especificación.
- Eficiente.
 - No debe hacer uso innecesario de recursos.
- Fácil de usar.
 - Interfaz de usuario adecuada.

Evolución de la Ingeniería



¿Que es la Crisis del Software ?



- Causas.
- Síntomas.
- Consecuencias.
- Solución.

Crisis del Software



- **Causas:**
 - Hardware más potente.
 - Mayor demanda.
 - Falta de metodologías y técnicas (hasta hace poco).
 - Uso inadecuado de recursos.
 - Sistemas más complejos.
 - Poca información de los desarrolladores.

Crisis del Software



- **Síntomas:**

- Productividad de los desarrolladores: baja en relación a la demanda.
- Expectativas: los sistemas no responden a las expectativas de los usuarios.
- Fiabilidad: Los programas fallan a menudo.
- Calidad: No es adecuada.
- Costes: Difíciles de predecir, a menudo sobrepasan lo esperado.
- Mantenimiento: Modificación del sw costosa y compleja.
- Plazos: No se cumplen.
- Portabilidad: Difícil cambiar de plataforma.
- Eficiencia: No hay aprovechamiento óptimo de recursos.

Crisis del Software



- **Consecuencias:**
 - Baja productividad.
 - Baja calidad.

Crisis del Software



- **Solución:**

- Aplicar la Ingeniería del Software en la construcción de Sistemas Informáticos.
- La necesidad de un enfoque de ingeniería en el desarrollo del software fue propuesta en una conferencia de la OTAN en 1968.

Desafíos de la Ingeniería de Software



- Reducir el coste y mejorar la calidad del software.
- Explotar y aprovechar el potencial proporcionado por el hardware

Desarrollar y mantener software asegurando:

- Calidad.
- Fiabilidad.
- Facilidad de uso
- Imposibilidad de mal uso

...de tal manera que el humano dirija la computadora y no al revés.

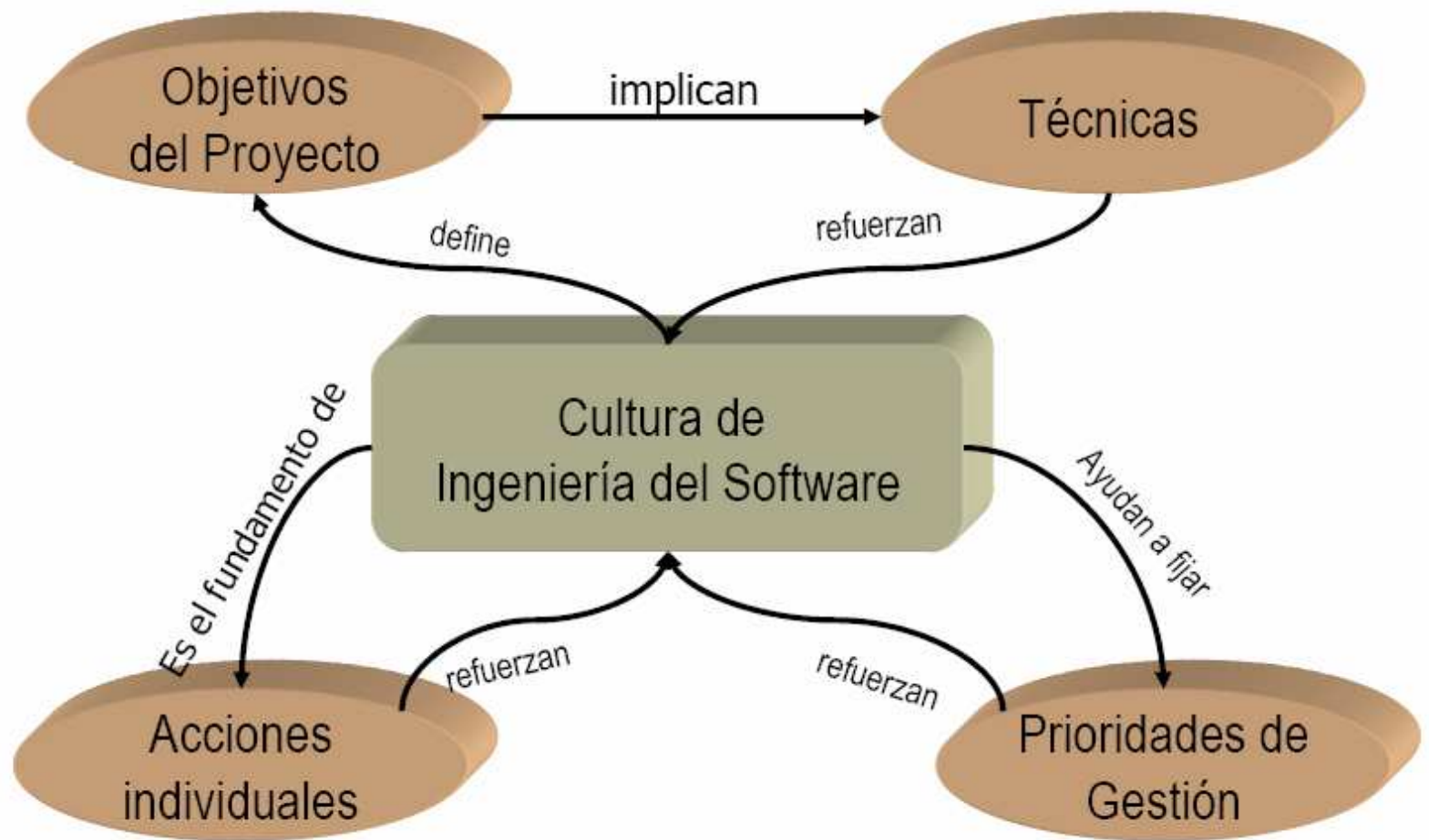
- Existen muchas clases de software:

- De sistemas.
- De tiempo real.
- De gestión.
- Científico.
- De Inteligencia Artificial.
- Empotrado.
- ...

...pero existen métodos y procedimientos comunes para construir **buen** software



Ingeniería de Software





¿Qué es la Ingeniería de Software?

- *“El establecimiento y uso de principios de ingeniería robustos, orientados a obtener económicamente software que sea fiable y funcione eficientemente sobre máquinas reales”*

Fritz Bauer.

- *“La aproximación sistemática al desarrollo, operación y mantenimiento del software.”*
- *“Software: programas de ordenador, procedimientos, reglas, documentación y datos asociados a un sistema de ordenador”.*

**IEEE Standard Glossary of Software Engineering
(IEE83).**

Objetivo : Ingeniería de Software

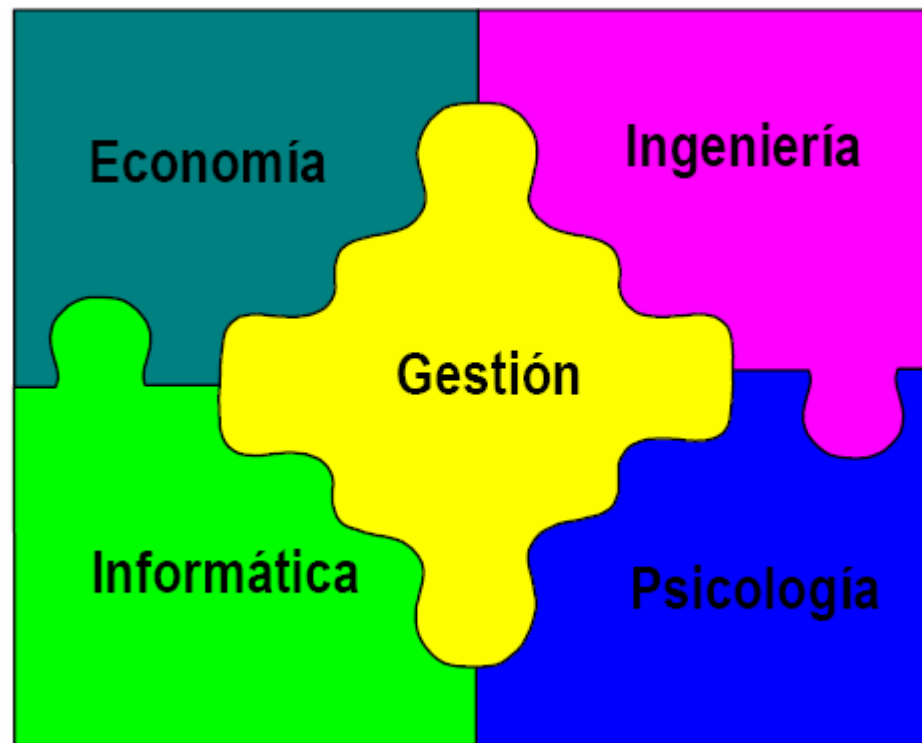


- El principal (común a todas las ingenierías)
- *“Construir instrumentos (HW, SW) que ayuden o faciliten al ser humano la realización de alguna tarea”*
- Conseguir un producto Software fiable, de alta calidad y bajo coste.
- Conducir un proceso de desarrollo y mantenimiento software de manera eficiente y con éxito.

Realizar un proyecto software no es (sólo) programar

Ingeniería de Software

Disciplinas que la componen:



Ingeniería del Software:

Fundamentos y técnicas



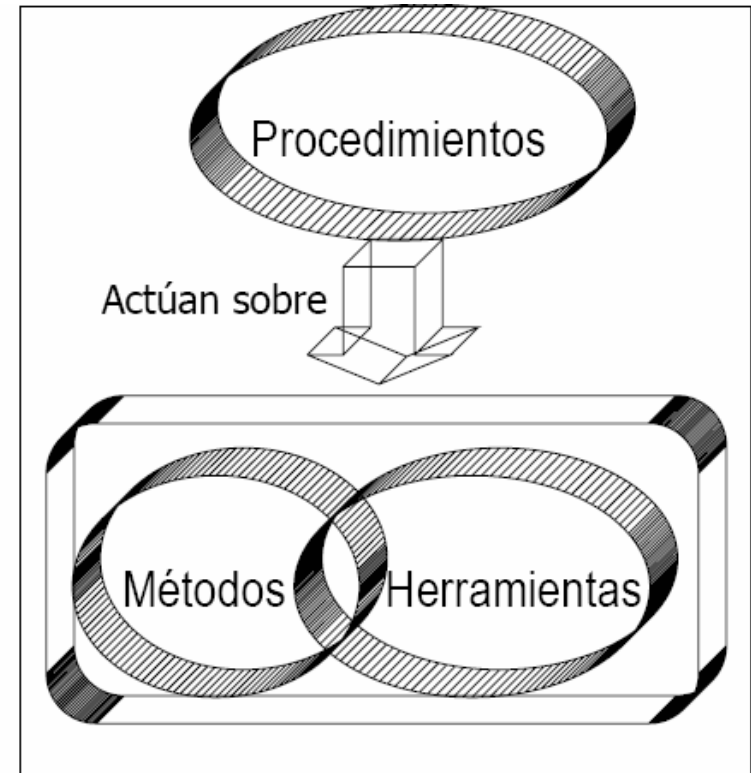
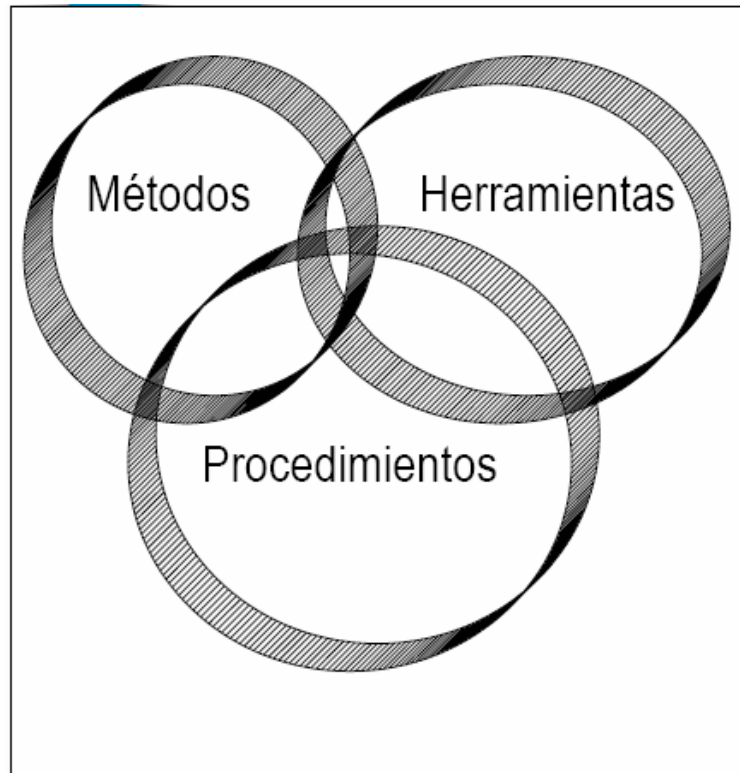
- Métodos:
 - Definen cómo construir el software desde el punto de vista técnico.
- Herramientas:
 - Proporcionan un soporte automático o semi-automático para los métodos.
- Procedimientos.
 - Punto de unión entre métodos y herramientas. Definen la secuencia en la que se aplican los métodos, cómo usar las herramientas, las entregas que se requieren, controles de seguimiento y calidad, guías para facilitar la labor de gestores y desarrolladores, etc.

Ingeniería del Software: Fundamentos y técnicas



- Métodos:
 - Planificación y estimación de proyectos.
 - Análisis de requisitos.
 - Diseño.
 - Codificación.
 - Pruebas.
 - Mantenimiento.
- Herramientas:
 - CASE.
 - CAD, ...
- Procedimientos.
 - PSP
 - TSP
 - CMMi.....

Ingeniería del Software: Fundamentos y técnicas



Otras técnicas de Apoyo



- Abstracciones:
 - Modelos de ciclo de vida.
 - Principios de las distintas fases, ...
- Representaciones:
 - Notaciones y lenguajes.
 - Diagramas de GANTT
- Evaluaciones:
 - Mediciones (sobre proceso y producto).

Actividades del Ingeniero del SW



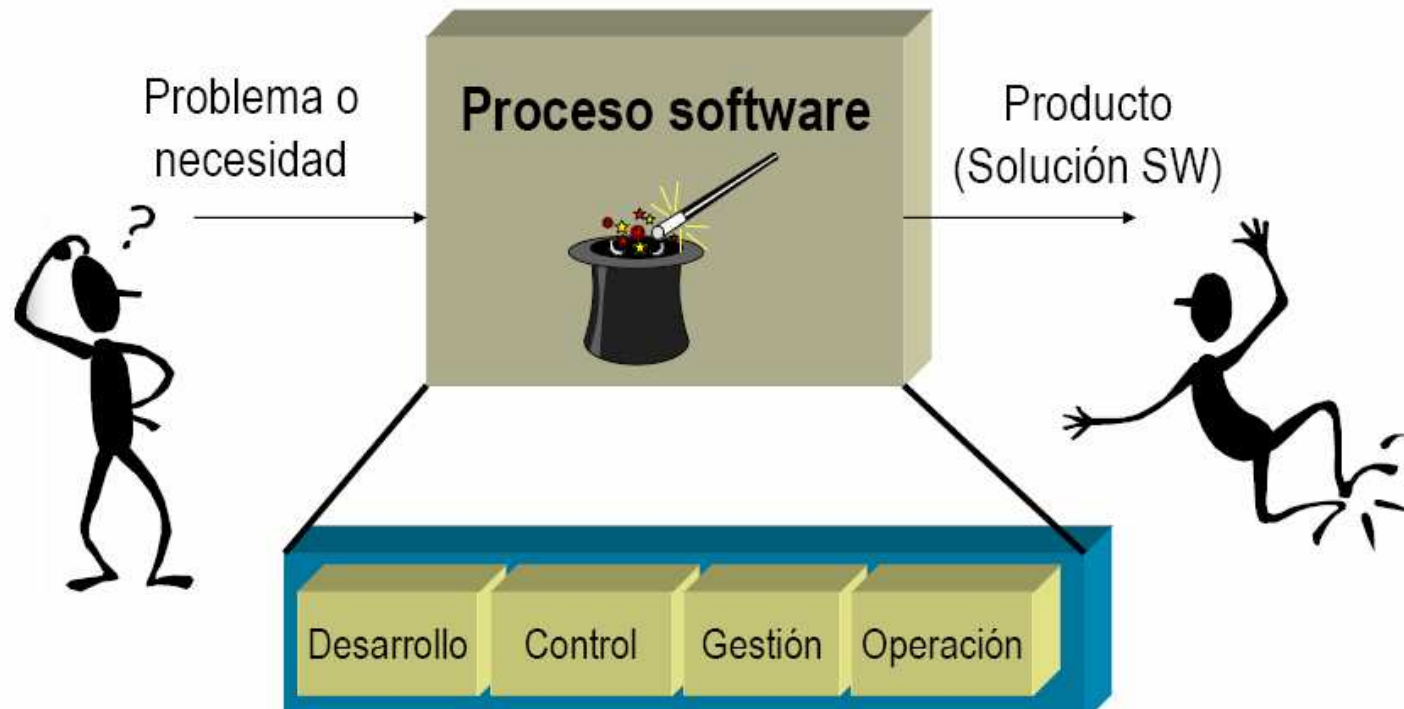
- **El Ingeniero del Software debe:**
 - Trabajar en equipo.
 - Analizar y estudiar problemas.
 - Trabajar bajo restricciones de tiempo, costes y recursos.
 - Interactuar con clientes y usuarios.
 - Tomar decisiones.
 - ...

Actividades del Ingeniero del SW



- Actividades de desarrollo:
 - Decidir qué hacer.(Análisis)
 - Decidir cómo hacerlo.(Diseño)
 - Hacerlo.(Codificación)
 - Probar el producto.(Pruebas)
 - Usar el producto.(Entrega/Instalación)
 - Mantener el producto.(Mantenimiento)

Producto y Proceso





Actividades del Ingeniero del SW

Análisis

- Estudio de Viabilidad.
- Educación de requisitos.
- Análisis de requisitos.
- Modelado del sistema.
- Maquetaje.
- ...

Diseño

- Arquitectónico.
- Detallado.
- Interfaz de usuario.
- Datos.
- ...

Construcción

- Codificación.
- Documentación.
- Debug
- ...

Mantenimiento

- Correctivo.
- Perfectivo.
- Adaptativo.
- ...

Validación y Verificación

- Inspecciones y revisiones.
- Planificación de pruebas.
- Pruebas de unidad.
- Pruebas de integración.
- Pruebas de regresión.
- Pruebas del sistema.
- Pruebas de aceptación.
- ...

Actividades del Ingeniero del SW



- Actividades de Gestión:
 - Planificación y estimación.
 - Seguimiento de los proyectos.
 - Administración de proyectos.
 - Dirección de proyectos.
 - ...
- Actividades de Operación:
 - Entrega (e instalación).
 - Puesta en marcha.
 - Formación a los usuarios.
 - ...

Actividades del Ingeniero del SW



- **El proceso de construir software es una actividad de resolución de problemas**

Realizar un proyecto software no es (sólo) programar

Actividad Colaborativa



1. Reúnete con tu equipo y asignen entre ustedes los siguientes roles:
 - **Secretario:** toma notas claras y organizadas sobre la discusión
 - **Moderador:** conduce la discusión, asegurándose de que todos participen activamente y que no se desvíen del tema. En caso de haber diferencias de opinión, se encargará de estabilizar la discusión.
 - **Informador:** presenta el resumen de lo discutido en el equipo, reflejando las ideas con exactitud.
 - **Observador:** observa y toma nota del comportamiento del resto de los integrantes del equipo. Posteriormente compartirá sus observaciones con los demás equipos.



2. Utilizando los temas ya vistos que cada uno de ustedes a estudiado, comenten sobre:

- Las diversas características que debe tener un software
- La importancia que tiene un producto de software
- Los principios que debe tener un producto de Software
- Las actividades que se deben desarrollar para generar una producto de software

y elijan lo mas representativo para presentar al resto de sus compañeros.

3. Todos deberán asegurarse de que todos los integrantes del equipo conozcan bien las características y la importancia que debe tener un buen desarrollo de software así como los riesgos que pueden suceder.



4. El informador de cada equipo presentará, en 2 minutos, el trabajo realizado.
5. Reúnanse de nuevo con sus grupos y discutan brevemente sobre las siguientes preguntas:
 - ¿Cómo les ayudó otro miembro del equipo a cumplir con el objetivo de la actividad?
 - ¿Cómo pueden mejorar como equipo para la próxima ocasión? En este punto, las notas del observador son de gran utilidad.