



# INGENIERÍA DEL SOFTWARE I

## Tema 1

### *Introducción a la Ingeniería del Software*

*Univ. Cantabria – Fac. de Ciencias*  
*Francisco Ruiz*



### Objetivos

- Comprender qué es la **Ingeniería del Software** y su necesidad.
- Situarla en el contexto más amplio de la **Informática**.
- Conocer el significado e implicaciones de la palabra "ingeniería".
- Conocer las principales **áreas** que la forman.
- Conocer el cuerpo de conocimientos **SWEBOK**.



## Contenido

- El Problema del Desarrollo de Software
    - Evolución Histórica
    - Nuevos Paradigmas
    - Naturaleza del Problema
  - Contexto de la Ingeniería del Software
    - Perspectiva de Ingeniería
    - ¿Por qué Ingeniería del Software?
    - Definición
    - Cuerpo de Conocimientos - SWEBOK
  - Ingeniería del Software vs Informática
    - Currículos Internacionales
    - Mercado Profesional
    - Academia
  - ¿Es Hacer Software una Profesión?
  - Conclusiones
- CHARLA**
- Áreas de Conocimiento
    - Requisitos
    - Diseño
    - Construcción
    - Pruebas
    - Mantenimiento
    - Gestión de la Configuración
    - Gestión de la Ingeniería Soft.
    - Procesos
    - Herramientas y Métodos
    - Calidad
  - Disciplinas relacionadas



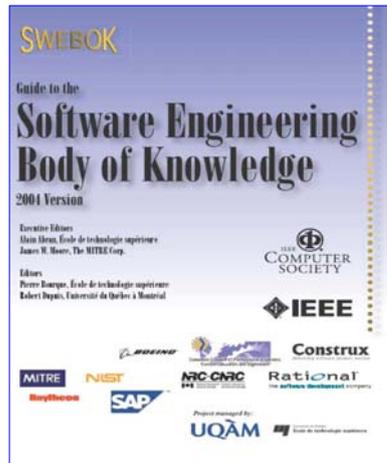
## Bibliografía

- Básica
  - IEEE Computer Society (2004)
    - SWEBOK - Guide to the Software Engineering Body of Knowledge, 2004 Version.
    - Capítulo 1.
    - <http://www.swebok.org/>
- Complementaria
  - Caps. 1, 2 y 4 del libro de Sommerville (2005).
    - Transparencias en
    - <http://www.comp.lancs.ac.uk/computing/resources/lanS/SE7/Presentations/index.html>
  - Cap. 1 del libro de Pressman (2005).



## Cuerpo de Conocimientos - SWEBOK

- **Software Engineering Body of Knowledge**
- <http://www.swebok.org/>



Francisco Ruiz - IS1

1.5



## Cuerpo de Conocimientos - SWEBOK

### *Software Engineering Body of Knowledge*

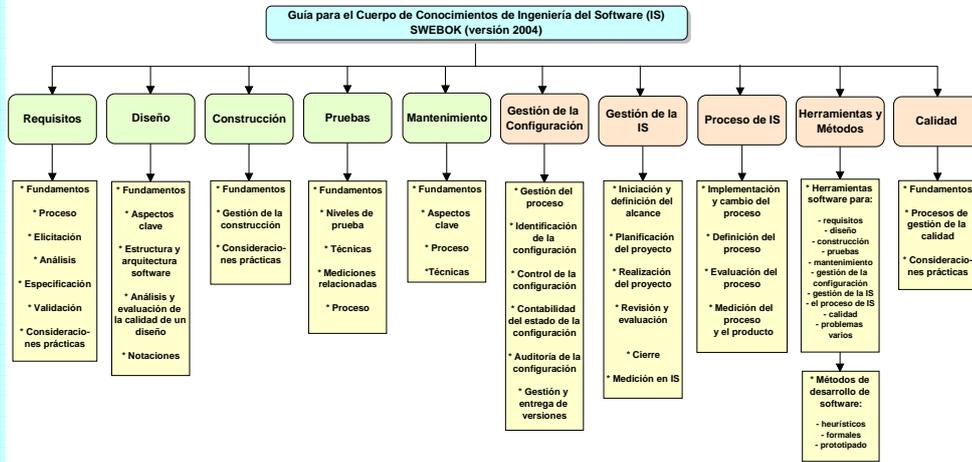
- Proyecto conjunto de **IEEE-CS** y **ACM**.
- Versión actual de 2004.
  - Aprobada oficialmente como ISO/IEC TR 19759:2005.
- Los objetivos principales de SWEBOK son cinco:
  - Promover una visión consistente del mundo de la IS.
  - Clarificar el papel –y delimitar las fronteras- de la IS con respecto a otras disciplinas asociadas: ciencia de la computación, gestión de proyectos, ingeniería de computadores, y matemáticas.
  - Caracterizar los contenidos de la disciplina.
  - Proveer acceso a los contenidos del cuerpo de conocimientos.
  - Proveer las bases para desarrollar planes de estudios o materiales para certificaciones individuales.

Francisco Ruiz - IS1

1.6



# Cuerpo de Conocimientos - SWEBOK

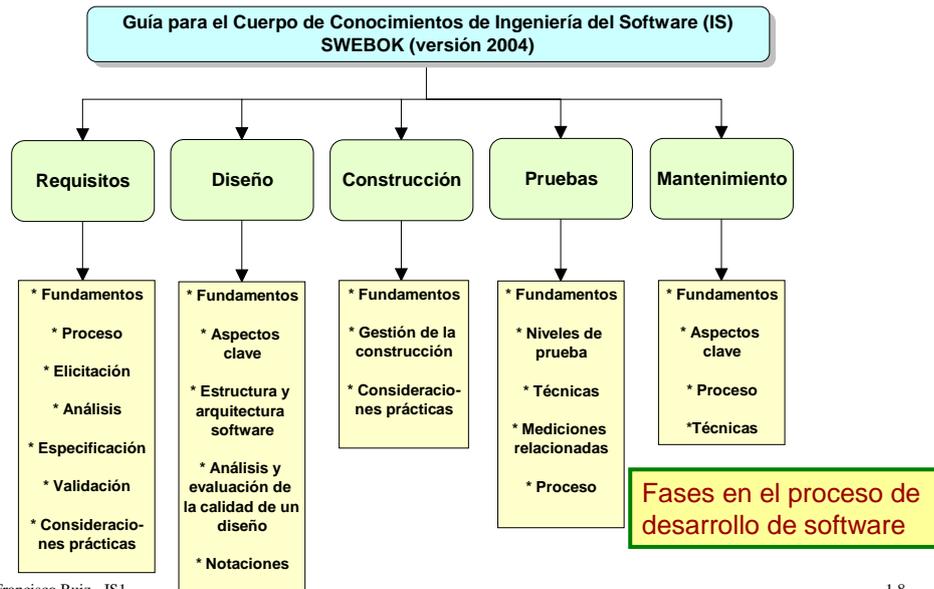


Francisco Ruiz - IS1

1.7



# Cuerpo de Conocimientos - SWEBOK



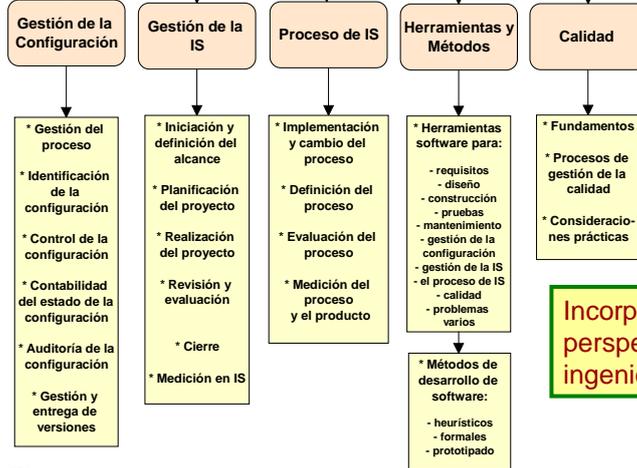
Francisco Ruiz - IS1

1.8



# Cuerpo de Conocimientos - SWEBOK

Guía para el Cuerpo de Conocimientos de Ingeniería del Software (IS)  
SWEBOK (versión 2004)

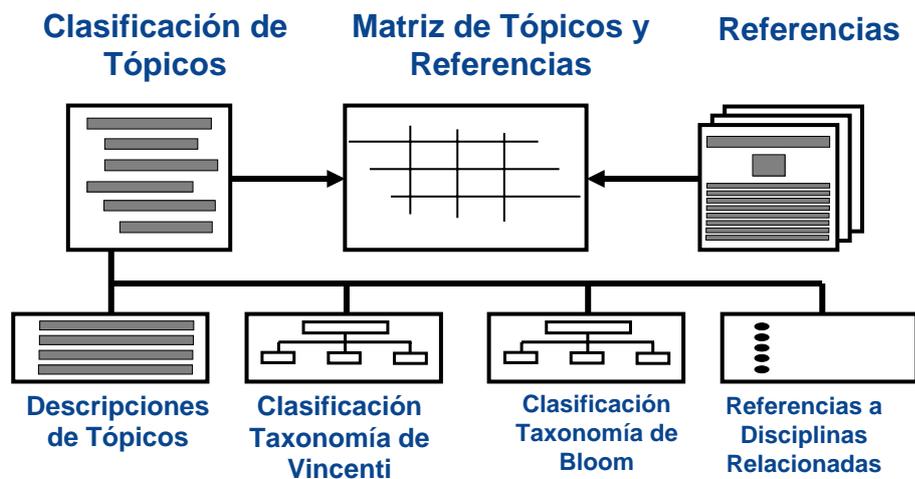


Incorporación de la perspectiva de ingeniería



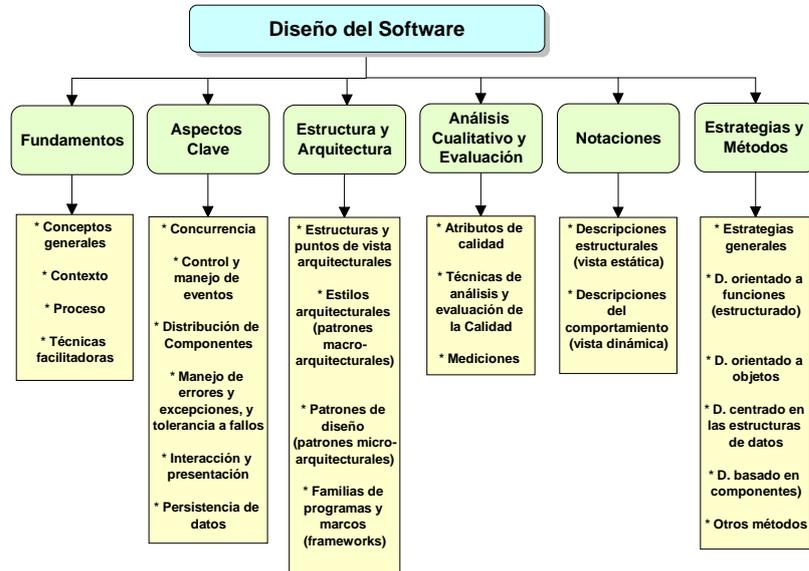
# Cuerpo de Conocimientos - SWEBOK

## Descripción de un Área de Conocimiento





## Cuerpo de Conocimientos - SWEBOK



Francisco Ruiz - IS1

1.11



## Áreas de Conocimiento

- Fases del Proceso de Desarrollo
  - Requisitos
  - Diseño
  - Construcción
  - Pruebas
  - Mantenimiento
- Perspectiva de Ingeniería
  - Gestión de la Configuración (gestión de productos)
  - Gestión de la Ingeniería (gestión de proyectos)
  - Proceso de Ingeniería (orientación a procesos)
  - Herramientas y Métodos (tecnología de soporte)
  - Calidad

Francisco Ruiz - IS1

1.12



## Áreas de Conocimiento

- **Requisitos**

- Se refiere a la elicitación, análisis, especificación y validación de los requisitos software.
- En la industria del software existe el consenso en que los proyectos de I.S. son muy vulnerables cuando estas actividades se realizan de forma pobre.
- Los requisitos software expresan las necesidades y restricciones que debe satisfacer un producto software para contribuir a la solución de un problema real.



## Áreas de Conocimiento

- **Diseño**

- El proceso de diseño de software consiste en analizar los requisitos con el fin de producir una descripción de la estructura interna del software que sirva como base para su construcción.
- Un diseño software (resultado) debe describir:
  - La arquitectura (cómo está descompuesto y organizado en componentes) y las interfaces entre dichos componentes; y
  - Los componentes con el nivel de detalle adecuado para poder construirlos.



## Áreas de Conocimiento

- **Construcción**
  - Se refiere a la creación detallada de software mediante la combinación de codificación, verificación, pruebas unitarias, pruebas de integración y depuración.



## Áreas de Conocimiento

- **Pruebas**
  - Sirve para evaluar la calidad de un producto software o para mejorarlo, mediante la identificación de sus defectos y problemas.
  - Consiste en la verificación dinámica del comportamiento real de un programa frente al comportamiento esperado, para un conjunto finito de casos de prueba (convenientemente seleccionados entre las usualmente infinitas posibilidades de ejecución) .



## Áreas de Conocimiento

- **Mantenimiento**
  - Todo producto software, después de su despliegue o entrega, "está destinado" a cambiar o evolucionar.
  - Algunas causas de ello son:
    - Defectos descubiertos durante su uso,
    - Cambios en el entorno operativo,
    - Nuevos requisitos del usuario, ..
- En SWEBOK este área se refiere a las actividades requeridas para proveer un adecuado soporte al software, sea antes o después del despliegue o entrega.



## Áreas de Conocimiento

- **Gestión de la Configuración**
  - Es la disciplina de identificar la configuración de un sistema en distintos momentos en el tiempo con el fin de controlar sistemáticamente los cambios y mantener la integridad y trazabilidad.
  - Una configuración de un sistema es una colección de versiones específicas de sus elementos (items de configuración) combinados de acuerdo a procedimientos de construcción adecuados a los propósitos buscados.



## Áreas de Conocimiento

- **Gestión de la Ingeniería**
  - Consiste en aplicar actividades de gestión (planificar, coordinar, medir, supervisar, controlar e informar) para asegurar que el desarrollo y mantenimiento de software se realizan de forma sistemática, disciplinada y cuantificable.
  - Básicamente, engloba dos clases de esfuerzos:
    - **Gestión de Proyectos** (Project Management)
    - **Medición** (futura nueva área en SWEBOK)



## Áreas de Conocimiento

- **Proceso de Ingeniería**
  - Se refiere a la definición, implementación, evaluación, medición, gestión, cambio y mejora de los propios procesos del ciclo de vida del software.
  - Engloba aspectos con fuerte impacto en la industria:
    - Madurez de las organizaciones (CMMI, SPICE)
    - Mejora de Procesos
  - Por ello, ha surgida la llamada *Ingeniería de Procesos Software*



## Áreas de Conocimiento

- **Herramientas y Métodos**

- Las herramientas (basadas en computador) ayudan a realizar los procesos del ciclo de vida del software.
- Los métodos imponen una manera o estructura para realizar las actividades de ingeniería del software, de forma que el trabajo sea más sistemático y más exitoso.



## Áreas de Conocimiento

- **Calidad**

- En este área se abordan las técnicas estáticas para alcanzar la calidad del software.
  - Las técnicas dinámicas (ejecutar el software) son parte de las Pruebas.
- Este campo también ha tenido un fuerte desarrollo en la industria:
  - Aseguramiento de la Calidad
  - Verificación y validación
  - Auditoría



## Disciplinas Relacionadas

---

- Ingeniería de Computadores
- Ciencia de la Computación
- Gestión
- Matemáticas
- Gestión de Proyectos
- Gestión de la Calidad
- Ergonomía del Software
- Ingeniería de Sistemas
  - incluye Sistemas de Información