

# Ingeniería del Software

3º de I.T.I.S.

Curso 2008-2009

Dr. Francisco José García Peñalvo

([fgarcia@usal.es](mailto:fgarcia@usal.es))

Miguel Ángel Conde González

([mconde@usal.es](mailto:mconde@usal.es))

Sergio Bravo Martín

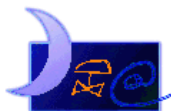
([ser@usal.es](mailto:ser@usal.es))

<http://moodle.usal.es>

<http://zarza.fis.usal.es/~fgarcia/docencia/isoftware/08-09/curso08-09.html>

<http://diaweb.usal.es>

Universidad de Salamanca – Departamento de Informática y Automática

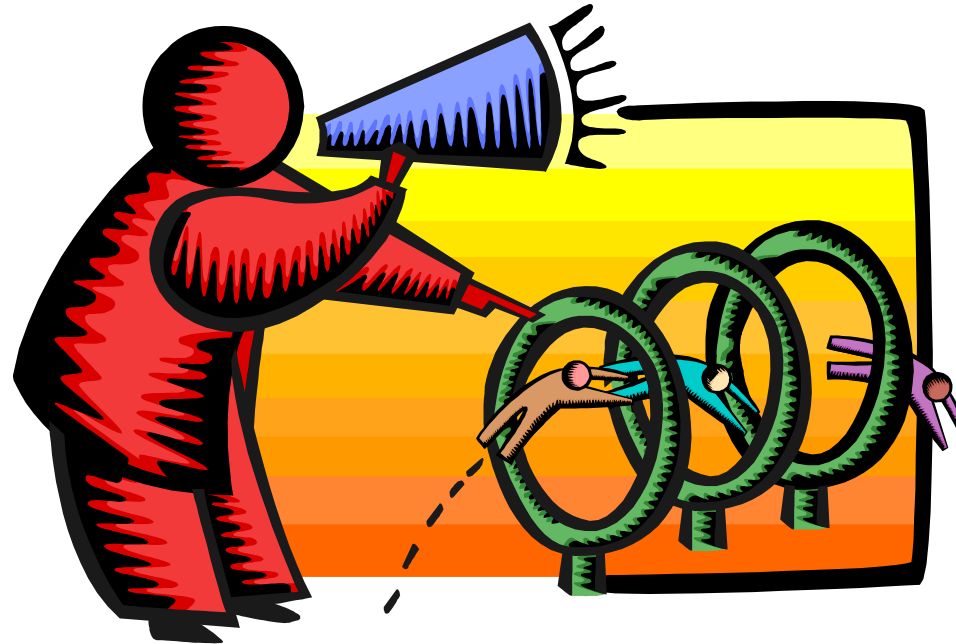


# Esquema

- Motivación
- Enfoque de ingeniería en el desarrollo del software
- Docencia de la Ingeniería del Software
- Asignatura Ingeniería del Software en ITIS
  - Objetivos
  - Temario
  - Evaluación
  - Bibliografía básica recomendada
  - Otras fuentes
  - Enlaces
  - Herramientas CASE
  - Tutorías
  - Campus virtual



# 1. Motivación

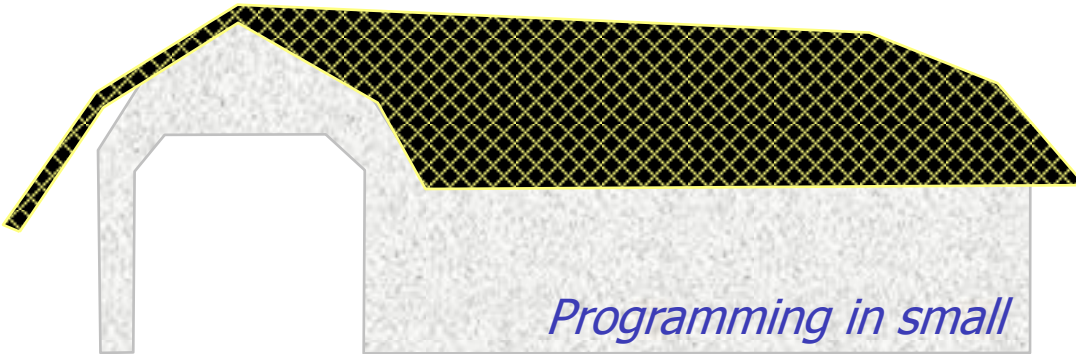


## Prácticas de software

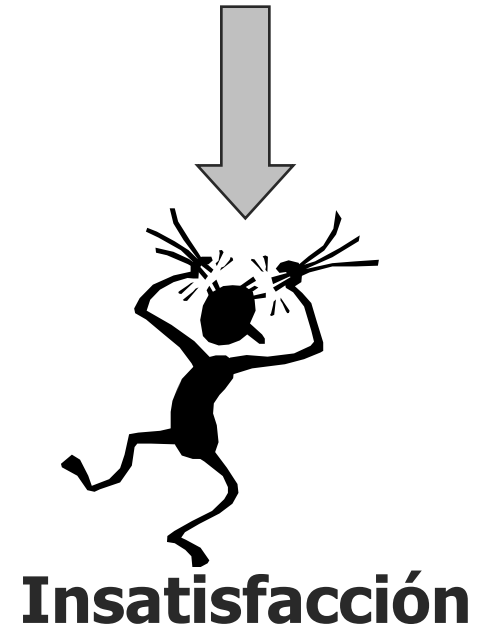
- La Ingeniería del Software dentro del currículo de los ingenieros en informática aporta la primera aproximación a la práctica **real** del desarrollo de software
  - Proyectos realizados por equipos de desarrollo
  - Programación a gran escala (*programming in large*)
  - Obtención (elicitación) de los requisitos
  - Modelos de ciclo de vida
  - Gestión de la configuración
  - Calidad del software
  - Mantenimiento
  - ...



# Aproximación personal al desarrollo del software

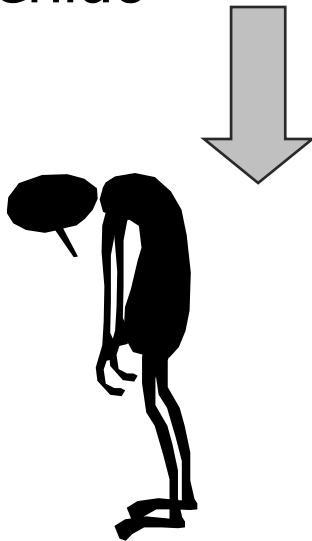


- Programa = Producto software final
- Carencia de documentación
- Imposibilidad de mantenimiento
- Dificultades en la evolución
- Dependencias personales insostenibles

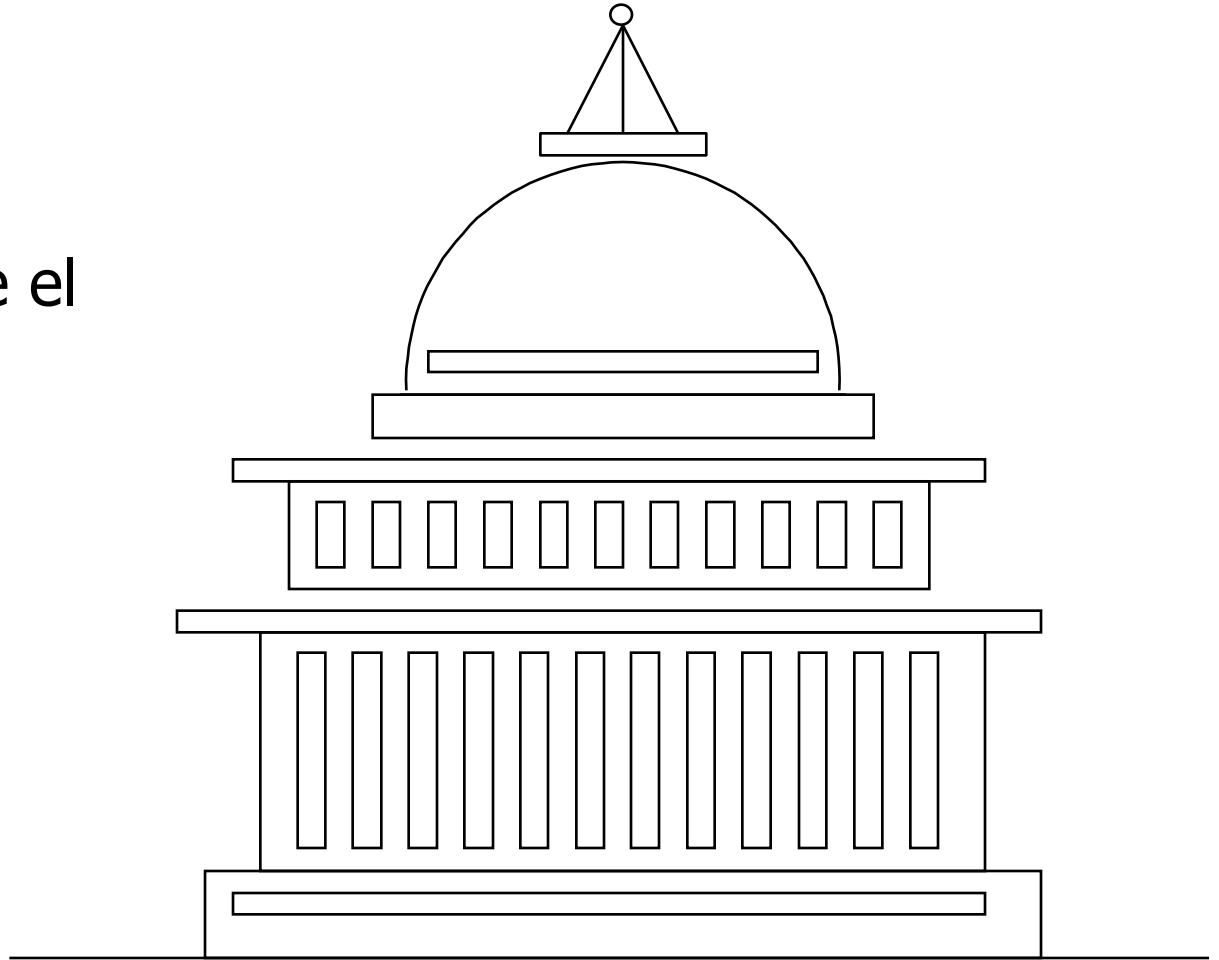


## Soluciones software desproporcionadas

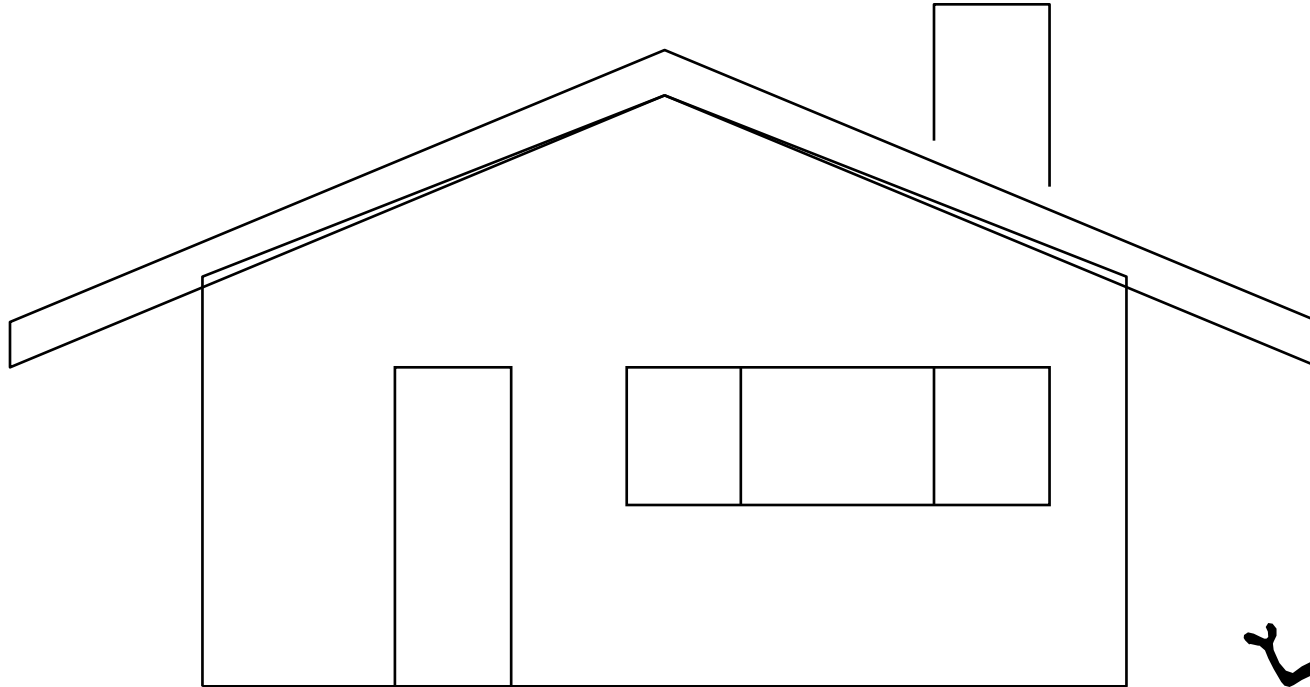
- Mala gestión
- Mala adquisición
- Desproporción entre el coste y el beneficio obtenido



**Insatisfacción**



# Desarrollo de software bajo las directrices de la Ingeniería del Software



"Se hacen planos para una casa tanto si ésta es grande como si es pequeña"

Barry W. Boehm





## Algunos apuntes económicos

- Los fallos del software en EEUU, tanto de funcionalidad como de seguridad, suponen 59.500 millones de dólares, esto es, un 0,6% del producto interior bruto de este país

NIST (*National Institute of Standards and Technology*)

- El 52,7% de los proyectos relacionados con las tecnologías de la información cuestan el 189% de su coste inicial estimado

*The Standish Group*

- El gobierno de EEUU estima que las empresas han gastado cerca de 3,3 trillones de dólares en tecnologías de la información en la última década





## Objetivos de la Ingeniería del Software (i)

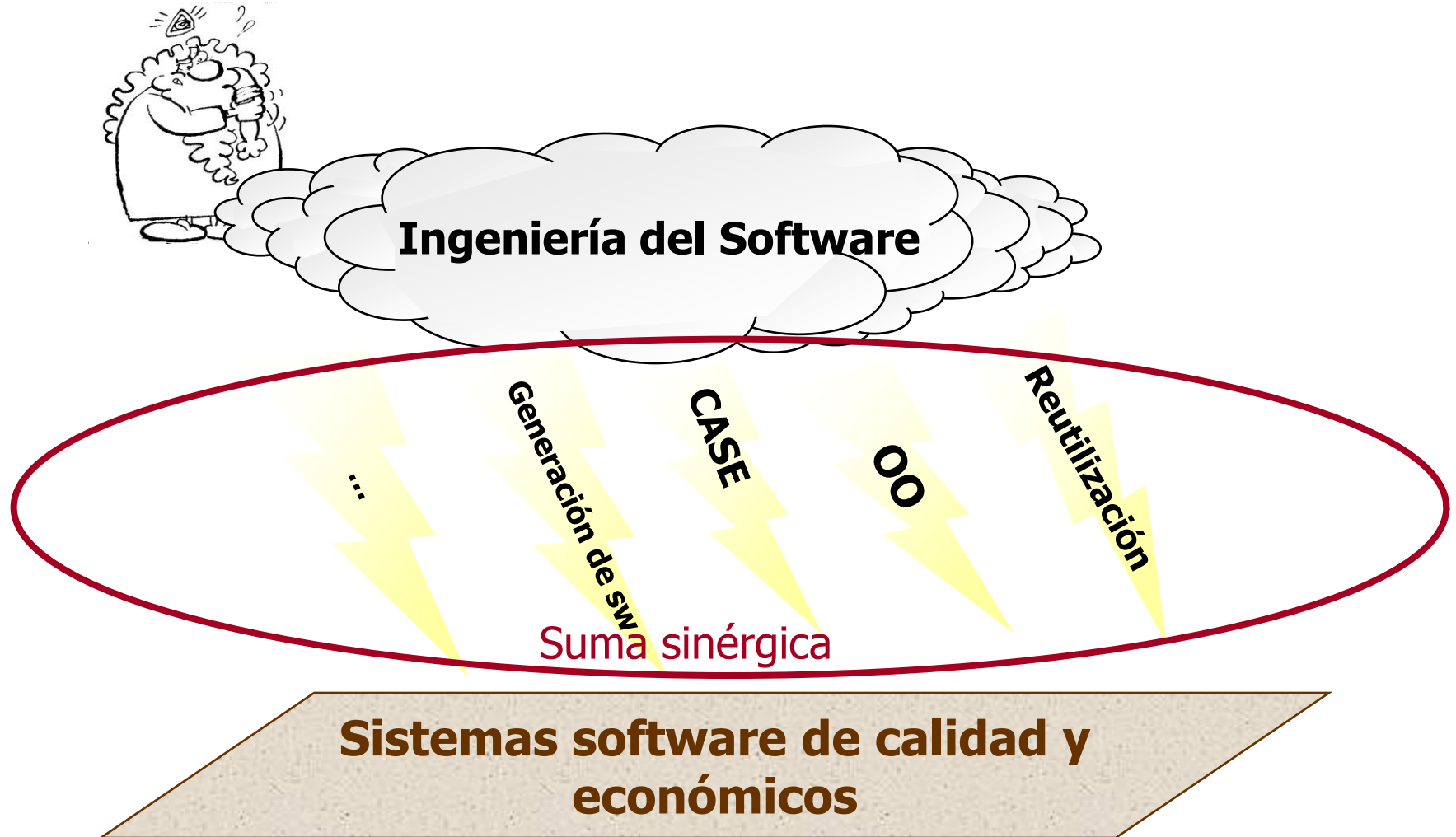
- Desarrollo de software de calidad
- Aumento de la productividad
- Desarrollo de software económico

“El desarrollo del software es un problema ingenieril ya que trata de crear soluciones efectivas y viables económicamente hablando a problemas reales”

Mary Shaw & James E. Tomayko



## Objetivos de la Ingeniería del Software (ii)



## 2. Enfoque de ingeniería en el desarrollo del software



## Ingeniería vs. métodos tradicionales (i)



VS



**El circo de la informática**

**La ingeniería informática**

## Ingeniería vs. métodos tradicionales (ii)



- Connotaciones sociales
  - Falta de una definición de la profesión de ingeniero informático
    - **Ética** en la ingeniería informática
    - ¿Colegios profesionales?
  - Intrusismo
  - Falta de **cultura** informática
  - Falta de **consideración** y **menosprecio** de los productos software



## Ingeniería vs. métodos tradicionales (iii)

### Ingeniería contra los mitos

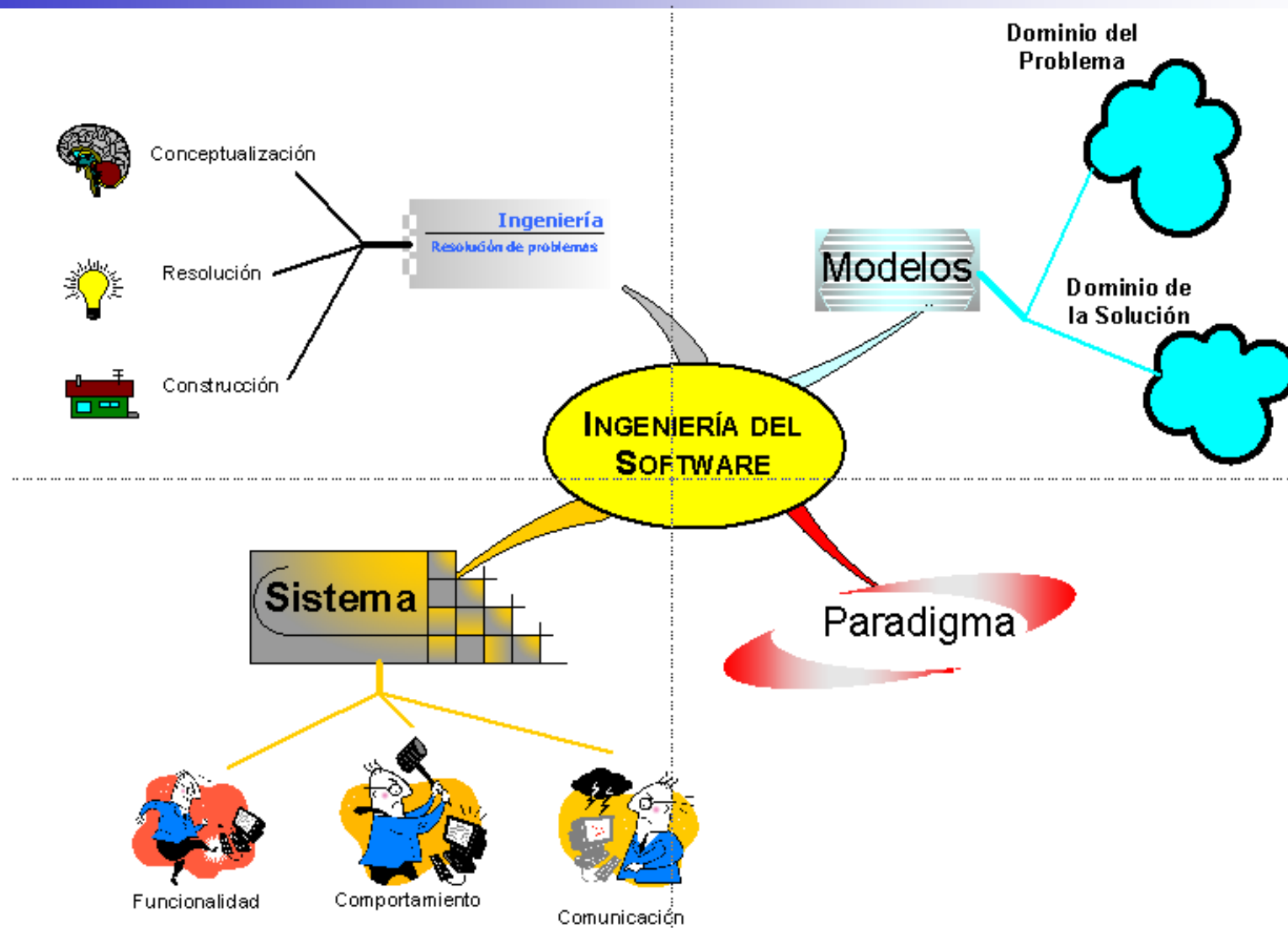


# Qué es y qué no es la Ingeniería del Software

- **La ingeniería** es la aplicación sistemática del conocimiento científico para construir soluciones, efectivas y viables en términos económicos, a problemas prácticos al servicio de la humanidad
- **La ingeniería del software** es la forma de ingeniería que aplica los principios de la ciencia de la computación y de las matemáticas para conseguir soluciones efectivas, y viables en términos económicos, a problemas relacionados con el software
- **La ingeniería del software** no es el diseño de programas que se implementan en otras áreas ingenieriles, ni es simplemente una forma de programar más organizada que la que prevalece entre aficionados, principiantes o personas con falta de educación y entrenamiento específico



# Contexto de la Ingeniería del Software





## 3. Docencia de la Ingeniería del Software



## Algunas consideraciones

- Dicotomía entre titulación y asignatura
- Contraposición al concepto de programador
- “Desarrollo de sistemas grandes para su uso en entornos de producción, durante un período largo, en el que posiblemente trabajan varias personas y, posiblemente, conlleve muchos cambios”
  - Desarrollo = gestión + mantenimiento + validación +  
+ documentación + ...
- La educación de un ingeniero del software debe tener unas raíces sólidas en la educación de la ingeniería tradicional (David Parnas)



## Mercado laboral

- Las instituciones que enseñan software son las **responsables** de producir profesionales del mundo del software que construyan y mantengan los sistemas para satisfacción de sus beneficiarios
- Gran mercado de trabajo para los “profesionales” del software
  - ¿Todos tienen que ser ingenieros informáticos?



## ¿Qué enseñar?

- Aproximación fácil – Herramientas de moda
  - No es la mejor opción para el futuro de los ingenieros
  - Atractivo a corto plazo, camino a la obsolescencia a medio/largo plazo
  - Evolución demasiado rápida de la disciplina
- Aproximación ingenieril – Fundamentos sólidos
  - Difícil de asimilar por los alumnos
  - Difícil de asimilar por las empresas
- Aproximación mixta – Fundamentos sólidos sobre herramientas adecuadas
  - Equilibrio entre teoría/práctica
  - Mayor capacidad para el éxito profesional para aquéllos que vayan más allá de las tendencias de moda y sean capaces de evolucionar con la disciplina



## 4. Asignatura Ingeniería del Software en ITIS



## Objetivos

- Descripción de las actividades técnicas e ingenieriles que se llevan a cabo en el ciclo de vida de un producto software
- Descripción de los problemas, principios, métodos y tecnologías asociadas con la Ingeniería del Software
- Presentación de la importancia de los requisitos en el ciclo de vida del software
- Introducción a las técnicas básicas de elicitación, documentación, especificación y prototipado de los requisitos de un sistema software
- Introducción a los métodos de análisis/diseño orientado a objetos
- Estudio y comprensión de los fundamentos del diseño de sistemas software
- Aplicar de forma práctica los conceptos teóricos sobre el desarrollo orientado a objetos
- Realización de un proyecto en grupo, aplicando los principios introducidos en la parte teórica de la asignatura



## Temario – Parte teórica

- **Unidad Didáctica I: Conceptos básicos (21 Horas)**
  - Tema 1. Introducción a la Ingeniería del Software (9 Horas)
  - Tema 2. Modelo Objeto. Una Descripción de UML (12 Horas)
- **Unidad Didáctica II: Requisitos (7 Horas)**
  - Tema 3. Introducción a la Ingeniería de Requisitos (7 Horas)
- **Unidad Didáctica III: Análisis y diseño orientado a objetos (17 Horas)**
  - Tema 4. Análisis Orientado a Objetos (5 Horas)
  - Tema 5. Principios del Diseño del Software (4 Horas)
  - Tema 6. Diseño Orientado a Objetos (8 Horas)

**Total: 4,5 Créditos**



## Temario – Parte práctica

### ■ Talleres

- Taller de modelado de datos
  - Repaso al modelo entidad-relación
- Taller de orientación a objeto
  - Diagramas de clase
- Taller de documentación de requisitos
  - Casos de uso

### ■ Práctica obligatoria

- Realización de una ERS (con algunos apartados de diseño)
- Se seguirán técnicas de ADOO
- Se realizará en grupos de máximo 3 personas (salvo excepciones justificadas)

**Total: 1,5 Créditos**





## Evaluación (i)

### ■ **Parte de Teoría (50% de la nota final)**

- Un examen final en el mes de enero/febrero (23-ene-09)
- Un examen final en el mes de septiembre (2-sep-09)
- Este examen constará de dos partes que hay que aprobar independientemente
  - Un test
  - Un conjunto de supuestos teórico/prácticos

### ■ **Parte Práctica (50% de la nota final)**

- Práctica realizada en grupos de trabajo
  - Se realizará una defensa de dicho trabajo
    - Todo grupo tiene derecho a dos defensas, siempre que la primera se haya hecho antes del mes de mayo
  - Esta práctica quedará superada para futuros cursos siempre y cuando se supere con una calificación mínima de un 6,5



## Evaluación (ii)

- Evaluación continua de la asignatura (modalidad opcional)
  - Se tendrá en cuenta la asistencia y la participación activa en clase
  - Todos los alumnos que se acojan a esta modalidad deberán haber rellenado su perfil (incluyendo una fotografía) en el campus virtual de la asignatura antes del 13/10/2008
  - Se realizarán 2 pruebas de test durante las clases de teoría
    - Prueba 1: Unidad I y II
    - Prueba 2: Unidad III
  - Se eliminará la parte de test del examen de teoría si se obtiene una calificación superior o igual a 5 en el conjunto de estas pruebas, obteniendo en todas ellas al menos una nota de 3
    - $\text{Nota test} = (\text{Prueba 1} * 0,5) + (\text{Prueba 2} * 0,5)$



## Evaluación (iii)

- Evaluación continua de la asignatura (modalidad opcional)
  - Talleres
    - Grupos de 3 personas
    - 15 minutos de puesta en común, 1h y 30 minutos de debate
    - Todos los grupos obligatoriamente entregan el modelo realizado previamente al comienzo de la sesión a través del campus virtual (en alguno de los formatos Open Document o PDF y realizado con alguna herramienta de modelado)
      - Si se detecta fraude se resta 1 punto
    - Grupos voluntarios defienden los trabajos y realizan informes con entrega en 15 días
      - Por la defensa se obtiene entre 0 y 0,5 puntos, por entregar el informe 0,5 puntos
    - Si no hay grupos voluntarios, se elegirá un grupo que defienda el trabajo
      - Por la defensa se obtiene entre -1 y 0,5 puntos, la entrega del informe es obligatoria y no aporta nota (pero puede restar si no se entrega en condiciones)
    - La participación activa, acertada y continuada puede aportar hasta 0,5 puntos
  - Entrega de ejercicios
    - Entregar 3 ejercicios de modelado, con enunciados sacados de libros (y no resueltos en ellos), tras su revisión en horario de tutorías aporta 0,75 puntos
  - Estas notas computarán en el apartado de supuestos teórico/prácticos del examen de teoría



## Evaluación (iv)

**Si** (Teoría  $\geq 5,0$ ) **y** (Práctica  $\geq 5,0$ )

Nota Final = (Teoría\*0,5) + (Práctica\*0,5)

**Si no**



**Fin si**

## Eventos

- Microsoft University Tour
  - Noviembre de 2008
- Curso extraordinario
  - Diseño de aplicaciones web con Patrones Java
    - 6, 7, 13, 14, 20 y 21 de marzo de 2009
    - 40 horas (30 h presenciales + 10 h on line)
    - 4 créditos de libre elección



# Bibliografía básica recomendada

## ■ Referencias principales

- **Booch, G., Rumbaugh, J., Jacobson, I.** *"El Lenguaje Unificado de Modelado"*. Addison Wesley, 1999
- **Jacobson, I., Booch, G., Rumbaugh, J.** *"El Proceso Unificado de Desarrollo de Software"*. Addison-Wesley, 2000
- **Larman, C.** *"UML y Patrones"*. 2ª Edición. Prentice-Hall, 2003
- **Pfleeger, S. L.** *"Ingeniería del Software. Teoría y Práctica"*. Prentice Hall, 2002
- **Piattini, M. G., Calvo-Manzano, J. A., Cervera, J., Fernández, L.** *"Análisis y Diseño de Aplicaciones Informáticas de Gestión. Una perspectiva de Ingeniería del Software"*. Ra-ma. 2004
- **Pressman, R. S.** *"Ingeniería del Software: Un Enfoque Práctico"*. 6ª Edición. McGraw-Hill. 2006
- **Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F., Lorensen, W.** *"Modelado y Diseño Orientados a Objetos. Metodología OMT"*. Prentice Hall, 2ª reimpresión, 1998
- **Rumbaugh, J., Jacobson, I., Booch, G.** *"El Lenguaje Unificado de Modelado. Manual de Referencia"*. 2ª Edición. Addison-Wesley. 2004
- **Sommerville, I.** *"Ingeniería del Software"*. 7ª Edición, Addison-Wesley. 2005

## ■ Otras referencias

- **Gamma, E., Helm, R., Johnson, R., Vlissides, J.** *"Patrones de Diseño"*. Addison-Wesley, 2003
- **Meyer, B.** *"Construcción de Software Orientado a Objetos"*. 2ª Edición. Prentice Hall, 1999
- **Schach, S. R.** *"Ingeniería de Software Clásica y Orientada a Objetos"*. 6ª Edición. McGraw-Hill. 2006
- **Yourdon, E.** *"Análisis Estructurado Moderno"*. Prentice-Hall Hispanoamericana. 1993



## Otras fuentes

- **Durán, A., Bernárdez, B.** "*Metodología para la Elicitación de Requisitos de Sistemas Software (versión 2.3)*". Informe Técnico LSI-2000-10, Universidad de Sevilla. <http://www.lsi.us.es/~amador>. [Última vez visitado, 24-9-2007]. Abril 2002
- **Durán, A., Bernárdez, B.** "*Metodología para el Análisis de Requisitos de Sistemas Software (versión 2.2)*". Universidad de Sevilla. <http://www.lsi.us.es/~amador>. [Última vez visitado, 24-9-2007]. Diciembre 2001
- **OMG.** "*OMG Unified Modeling Language Specification. Version 1.5*". Object Management Group Inc. Document formal/03-03-01. March 2003. <http://www.omg.org/docs/formal/03-03-01.pdf> [Última vez visitado, 24-9-2007]
- **OMG.** "*Unified Modeling Language: Superstructure. Version 2.0*". Object Management Group Inc. Document formal/05-07-04, August 2005. <http://www.omg.org/cgi-bin/doc?formal/05-07-04> [Última vez visitado, 24-9-2007]
- **Pohl, K.** "*Requirements Engineering: An Overview*". En M. Dekker (Ed.), Encyclopedia of Computer Science and Technology, 36. 1997. Disponible en <ftp://sunsite.informatik.rwth-aachen.de/pub/CREWS/CREWS-96-02.pdf> [Última vez visitado, 24-9-2007]



## Enlaces (i)

- **Association for Computing Machinery (ACM)**
  - <http://www.acm.org>
  - Fundada en 1947 fue la primera sociedad científica y de educación del mundo. El portal de información que presenta es impresionante, tanto en cuanto a enlaces de interés, grupos de trabajo, documentos electrónicos, conferencias como por su biblioteca digital conteniendo revistas y actas de congresos (<http://portal.acm.org>)
- **Cetus Links - Object-Orientation**
  - <http://www.cetus-links.org>
  - La colección de enlaces más completa sobre Orientación a Objetos. A fecha de 10 de febrero de 2006 se tenían contabilizados 18193 enlaces
- **Consejo Superior de Informática**
  - <http://www.csi.map.es/>
  - Web del Consejo Superior de Informática donde, entre otras cosas, se puede obtener la documentación y herramientas sobre Métrica 3 (<http://www.csi.map.es/csi/metrica3/index.html>)





## Enlaces (ii)

- **DACS (Data & Analysis Center for Software)**
  - <http://www.dacs.dtic.mil>
  - Sitio perteneciente al DoD de EEUU con numerosa información y enlaces a lugares relacionados con la Ingeniería del Software
- **European Software Institute (ESI)**
  - <http://www.esi.es>
  - El Instituto del Software Europeo tiene su sede en Bilbao (España). Dispone tanto de documentación privada para los miembros del instituto como documentación pública con los análisis de proyectos, necesidades de empresas y software europeo
- **Guide to the Software Engineering Body of Knowledge (SWEBOK)**
  - <http://www.swebok.org>
  - Proyecto para establecer un cuerpo de conocimiento común para la Ingeniería del Software



## Enlaces (iii)

### ■ Institute of Electrical and Electronics Engineers (IEEE)

- <http://www.ieee.org>
- Otra prestigiosa organización compuesta por diversas sociedades, donde la que más relación tiene con los temas abordados en el presente curso es la IEEE Computer Society (<http://computer.org>)
- A semejanza de ACM, ofrece información sobre conferencias, estándares, educación y mantiene otra biblioteca digital con revistas y actas de congresos

### ■ Object Management Group (OMG)

- <http://www.omg.org>
- Es un consorcio internacional de industrias con el fin de promover el uso de la Orientación a Objetos en la Ingeniería del Software. A diferencia de organizaciones como ISO o IEEE, OMG desarrolla estándares de “*facto*” como consenso entre las empresas que la forman
- Dicho servidor ofrece publicaciones electrónicas y enlaces a estándares y herramientas del sector relacionado con la tecnología de objetos
- En <http://www.omg.org/uml> se encuentran las especificaciones de la versión actual (así como de las anteriores) de UML



## Enlaces (iv)

### ■ IBM Rational Software

- <http://www-306.ibm.com/software/rational/>
- En esta dirección se tiene valiosa información sobre UML y RUP (documentos, informes, artículos, presentaciones, bibliografía recomendada...)
- Además, se pueden obtener versiones de demostración de diferentes herramientas que comercializan, siendo Rational Rose la más difundida

### ■ R. S. Pressman & Associates, Inc.

- <http://www.rspa.com>
- Bajo la dirección de Roger S. Pressman y la difusión internacional de su afamado libro sobre Ingeniería del Software, aparece una empresa de consultoría en Ingeniería del Software
- Lo más interesante que ofrece esta dirección es un portal que da entrada a otras fuentes de información relacionadas con cada uno de los capítulos tratados en su libro



## Enlaces (v)

### ■ Software Engineering Institute (SEI)

- <http://www.sei.cmu.edu>
- El Instituto de Ingeniería del Software en la Universidad Carnegie Mellon, es uno de los lugares más activos en pro de la Ingeniería del Software
- Se pueden encontrar documentos asociados a módulos curriculares en Ingeniería del Software, informes técnicos sobre diferentes áreas de la Ingeniería del Software, una revista en línea...
- Incluye además enlaces a otras organizaciones relacionadas con la Ingeniería del Software

### ■ The World Wide Web Consortium

- <http://www.w3.org/>
- Sitio central donde se recogen las especificaciones de los diversos lenguajes relacionados con la Web (HTML, XML, RDF, SOAP...)



## Enlaces (vi)

- **The World Wide Web Virtual Library Formal Methods**
  - <http://www.afm.sbu.ac.uk>
  - Sitio especializado en información sobre métodos formales
- **UML Resource Center**
  - <http://www-306.ibm.com/software/rational/uml/>
  - Sitio donde se centraliza gran parte de la información sobre UML
- **Unified Modeling Language (UML) Dictionary**
  - <http://www.well.com/~hirst/UML-dict.html>
  - Diccionario sobre términos de UML recopilados por Kendall Scott
- **Webliography Software Engineering**
  - [http://polaris.umuc.edu/~skerby/help/wbib\\_swe.htm](http://polaris.umuc.edu/~skerby/help/wbib_swe.htm)
  - Colección de enlaces sobre Ingeniería del Software
- **WWW Virtual Library - Software Engineering**
  - <http://mingo.info-science.uiowa.edu/soft-eng>
  - Colección de enlaces sobre Ingeniería del Software



# Herramientas CASE

## ■ Choosing a UML Modeling Tool

- [http://www.objectsbydesign.com/tools/modeling\\_tools.html](http://www.objectsbydesign.com/tools/modeling_tools.html)
- Elección de una herramienta para modelar con UML

## ■ ArgoUML

- <http://argouml.tigris.org>
- Java

## ■ Enterprise Architect

- <http://www.deiser.com>

## ■ Left CASE

- <http://zarza.fis.usal.es>
- GNOME/Linux

## ■ Microsoft Visio

- <http://www.microsoft.com/office/visio>

## ■ PowerDesigner

- <http://www.sybase.com/products/developmentintegration/powerdesigner>

## ■ Rational Rose

- <http://www-306.ibm.com/software/rational/>
- Windows/UNIX

## ■ REM

- [http://www.lsi.us.es/descargas/descarga\\_programas.php?id=3](http://www.lsi.us.es/descargas/descarga_programas.php?id=3)
- Windows

## ■ Studio CASE

- <http://zarza.fis.usal.es/~fgarcia>
- Windows

## ■ Visual Interaction

- <http://zarza.fis.usal.es/~fgarcia>
- Windows

## ■ Visual Paradigm

- <http://www.visual-paradigm.com/>
- Windows

## ■ Visual UML

- <http://www.visualobjectmodelers.com/products.htm>
- Windows



# Ingeniería del Software

3º de I.T.I.S.

Curso 2008-2009

Dr. Francisco José García Peñalvo

([fgarcia@usal.es](mailto:fgarcia@usal.es))

Miguel Ángel Conde González

([mconde@usal.es](mailto:mconde@usal.es))

Sergio Bravo Martín

([ser@usal.es](mailto:ser@usal.es))

<http://moodle.usal.es>

<http://zarza.fis.usal.es/~fgarcia/docencia/isoftware/08-09/curso08-09.html>

<http://diaweb.usal.es>

Universidad de Salamanca – Departamento de Informática y Automática

