

CC51A – Ingeniería de Software

Introducción a la Ingeniería de Software

Sergio Ochoa D.

Parte de este material ha sido cedido por el Prof. Jaime Navón, PhD.

(www.ing.puc.cl/~jnavon)

Estructura de la Presentación

- 1- Generalidades de la Ingeniería de Software.
- 2- Problemas con la Ejecución de Proyectos Grandes.
- 3- ¿Qué se Requiere?.
- 4- Características de la Ingeniería de Software.
- 5- Conclusiones.

Ingeniería de Software

¿Qué es la Ingeniería de Software?

“La Ingeniería de Software es el área de las ciencias de la computación que trata con la construcción de sistemas de software, los cuales son tan grandes y complejos que se construyen con equipos de ingenieros [Ghezzi 91].”

“Construcción multi-persona de software multi-versiones [Parnas 87].”

Ingeniería de Software (cont...)

“ Es un proceso definido paso a paso, que facilita la especificación, el diseño, la implementación y las pruebas de una solución de software, para un conjunto de requisitos explícitos, de modo eficiente y eficaz”

Esto requiere que antes de empezar se tenga:

- Objetivos claros.
- Planes para lograr los objetivos.
- Procedimientos que implementan los planes.
- Procedimiento de monitoreo y control de los planes.
- Un ambiente conducente al logro de los objetivos.

Ingeniería de Software (cont...)

- ◆ Analice un problema antes de plantear una solución
- ◆ No se debe plantear una solución en busca de un problema
- ◆ Separe los problemas complejos en sub-problemas más simples
 - Asegúrese que las relaciones entre los sub-problemas están acotadas.
- ◆ No evite el cambio, adminístrelo
- ◆ Especifique la calidad
- ◆ Hay que desarrollar una visión compartida del proyecto.

Ingeniería de Software (cont...)

- ◆ Nunca pierda de vista al usuario y al cliente.. Identifícalos y modélalos (casos de uso). Eso te dará una comprensión general del problema.
- ◆ Ten en cuenta:
 - Los objetivos y necesidades del *usuario* y del *cliente*.
 - Temores del usuario.
 - Costo de aprender.
 - Costo de usar (incluya las actividades sin computador)
 - Agrado/Desagrado de usar el software

Ingeniería de Software (cont...)

- ◆ Diseña algo implementable
 - Con tecnología accesible
 - Con costos aceptables
 - Con plazos adecuados
 - Con una buena relación “costo/beneficio”

Ingeniería de Software (cont...)

La Ingeniería de Software tiene que ver con la concepción, diseño, implementación y mantenimiento de sistemas GRANDES basados en computador.

- **software debe ser terminado a tiempo y dentro del presupuesto**
- **software debe tener niveles de desempeño y usabilidad aceptables**
- **software debe ser correcto, confiable y robusto**

La parte difícil es lograr todo esto en proyectos grandes

Ingeniería de Software (cont...)

¿Qué es un Proyecto grande?

- Windows NT 4.0 tiene entre 6 y 10 millones de líneas de código
- El Software del bombardero B2 tiene 3.5 millones
- Un switch telefónico típico tiene 2 millones

Para tener una idea de lo que es 1 millón de líneas de código (1MLOC):

- 13.000 páginas (75 líneas por página)
- 26 tomos de 500 páginas c/u
- 22 horas para imprimir (10 pp/min)
- Medio día para recompilar (... depende de la máquina)

¿Cuál es el Problema ?

A pesar de que la industria del software es hoy en día uno de los motores de la economía mundial, numerosos problemas aún persisten...

- el desarrollo del hardware siempre aventaja al software asociado a él
- la demanda por nuevos productos y soluciones no puede ser satisfecha
- el software no es confiable (y somos cada vez más dependientes de él)
- los proyectos raramente terminan en las fechas previstas
- los proyectos raramente terminan costando lo previsto
- proyectos terminados que jamás se usan
- proyectos que se usan pero usuarios altamente insatisfechos
- costo exorbitante de mantenimiento

¿Cuál es el Problema ? (cont...)

En casos extremos ...

- quiebra del productor del software
- quiebra de los clientes que dependen del producto
- pérdida de vidas humanas

Historias de Horror....

- El Bank of America proyectó US\$23 M para un proyecto a 5 años. Se terminó gastando mas de US\$60 M para finalmente abandonar el proyecto. Pérdidas totales estimadas en más de US\$1000 M
- El bombardero B1 requirió US\$ 1000 M más de lo proyectado para mejorar sus sistemas de defensa

Historias de horror ...

- AllState (seguros) comenzó en 1982 un proyecto de automatización integral de sus operaciones de 5 años de duración y US\$8 M de presupuesto. Fue abandonado en 1993 después de gastar US\$100 M
- Blue Cross (isapre norteamericana) perdió más de US\$60 M en pagos incorrectos debido a un error en el software por el que habían pagado mas de US\$200 M
- El 4 de Julio de 1996 un cohete Ariadne se desvió de su curso y explotó a 3700 m de altura. Causa: error de software. Parte de las conclusiones de la comisión investigadora afirma ...
"... In the failure scenario, the primary technical causes are the Operand Error when converting the horizontal bias variable BH, and the lack of protection of this conversion which caused the SRI computer to stop"

Historias de horror ... (cont...)

- Entre 1987 y 1995 a lo menos 6 pacientes fueron severamente heridos o muertos por un error en los sistemas del Therac-25 (acelerador lineal usado para administración de radioterapia)

Particularidades de Sistemas Grandes

Escala: una sola persona no puede entenderlo todo

Complejidad: Requiere trabajo de equipo:

- problemas de manejo de gente
- coordinación, egos, motivación, recambios, etc.

Cambio: durante y después del desarrollo inicial

Vida: años y décadas

Especificaciones imprecisas: ambigüedad, contradicciones, requisitos cambiantes

¿ Qué se requiere ?

- **Burocracia (útil y efectiva, tedioso pero vital)**
 - manejo formal del proceso de desarrollo
 - documentación formal detallada tanto interna como externa
 - puede pensarse en términos de contratos cliente/productor, etc.
 - trazabilidad
 - ¿de quién es este código?
 - ¿cuándo se agregó esta parte?
 - manejo de configuración y control de versiones

¿ Qué se requiere ? (cont...)

- **Análisis cuidadoso del problema**
 - interactuando con el usuario
- **Diseño cuidadoso usando principios que han demostrado ser útiles**
 - abstracción, ocultamiento, modularización, etc.
- **Implementación cuidadosa**
- **Pruebas rigurosas**
 - procedimientos bien definidos de antemano
- **Planificación de largo plazo (mantenimiento)**

¿Qué se Requiere ? (cont...)

Más que código: *En un proyecto de software se genera además del código, muchos otros documentos ...*

- requerimientos formales
- diseño de alto nivel
- diseño detallado
- plan y baterías de pruebas (tests)
- documentación de usuario
- documentación de desarrolladores
- estudios de factibilidad
- informes de marketing
- planes de mantenimiento
- informes de errores y correcciones
- etc.

¿Qué se Requiere? (cont...)

◆ Ingeniero de Procesos

◆ Ingeniero de Calidad

◆ Analista de Requisitos

◆ Diseñador de Software

◆ Programador

◆ Verificador-Téster

◆ Gerente de Proyecto

◆ Gestor de Configuración del Software



Característ. de la Ingeniería

- **Ataca problemas prácticos reales**
 - la gente realmente quiere o necesita resolver estos problemas
- **Genera soluciones razonables**
 - en términos de costos, uso de recursos, etc.
- **Tiene su base en la ciencia**
 - sus resultados son repetibles
 - usa modelos matemáticos
 - es área técnica bien entendida

Característ. de la Ingeniería (cont...)

- **Codificación del conocimiento**

- la experiencia de generaciones se escribe en enormes manuales (ej. Perry's Handbook) y se organiza para ser reutilizada

- **Responsabilidad profesional**

- Código de conducta
- Ética profesional
- Acreditación y monitoreo por parte de una sociedad profesional

Caract. de la Ing. de Software

No es posible aplicar directamente metodologías usadas en la ingeniería de otros productos, porque hay varias diferencias importantes ...

- **El software no se manufactura sino que se desarrolla o crea**
 - Control de calidad de distinta naturaleza
 - Recursos humanos de distinta naturaleza
 - Estructura de costos radicalmente distinta
- **El software no se desgasta (idealmente)...**
... pero sufre de envejecimiento.

Caract. de la Ing. de Software (cont...)

- **La mayoría de los proyectos implican partir casi de cero (no hay catálogos de componentes a disposición de los ingenieros)**
- **Falencias más importantes:**
 - Procesos no son repetibles
 - Modelos aportan descripciones precisas
 - Existe poca confiabilidad, tanto en el proceso como en el producto
 - No se comparte la experiencia (codificación y reuso)
 - Hay una falta de entrenamiento continuo de los profesionales

Principios de la I.S.



Principio: Propositiones o verdades que rigen el desarrollo.

Método: Guía general que gobierna la ejecución de una actividad.

Técnica: Mecanismos y recursos del desarrollo.

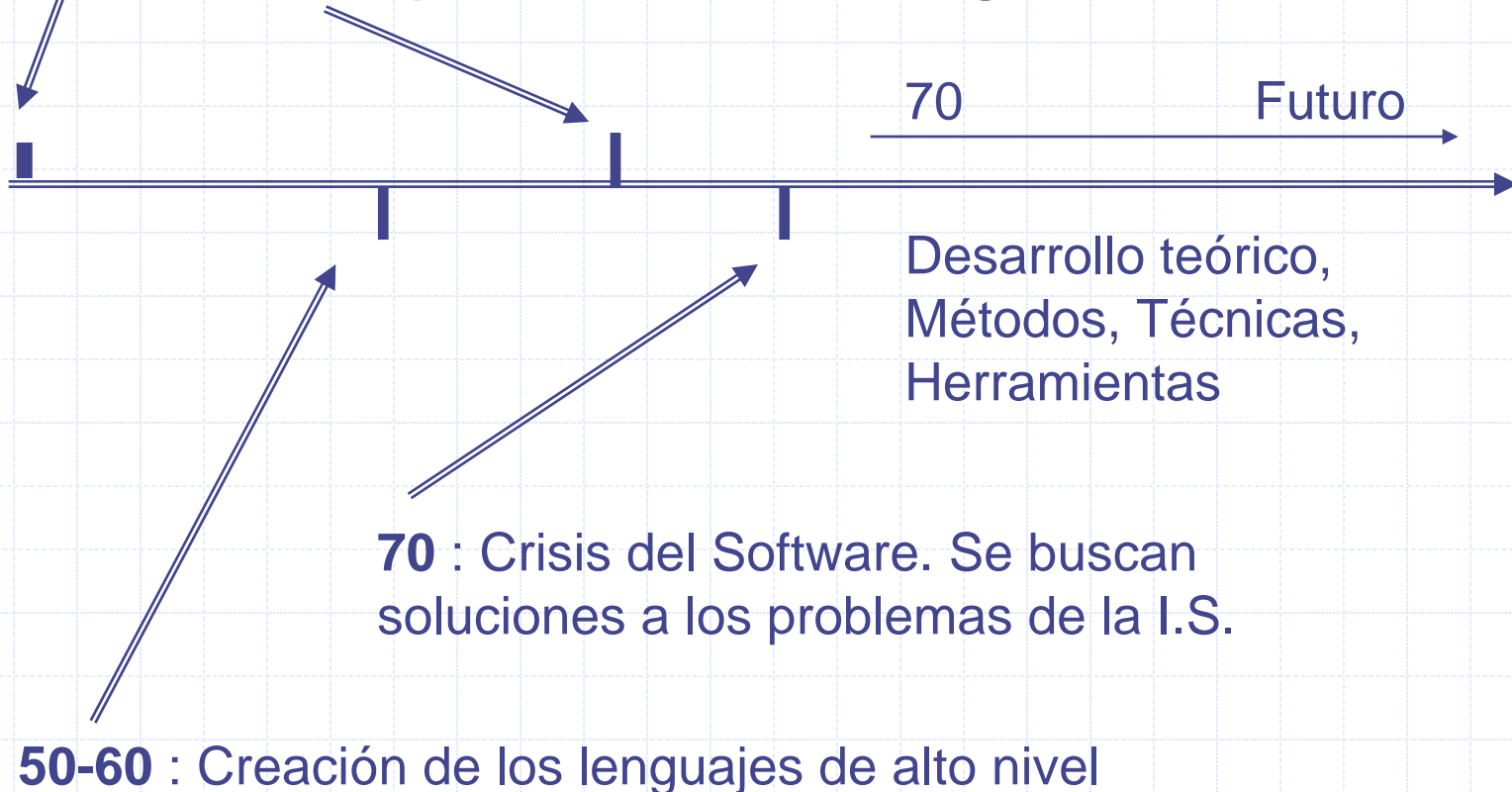
Metodología: Conjunto de métodos y técnicas .

Herramientas: Soportan la aplicación de metodologías.

Evolución de la I.S.

OS 360 IBM : Sistema “grande” , se crea el término INGENIERIA DE SOFTWARE

Comienzos : **Poner un conjunto de instrucciones para que las computadoras hicieran algo útil**



Conclusiones

- **Proceso de desarrollo de software**
 - para mejorar la situación es necesario mejorar el proceso
 - para mejorar el proceso es necesario hacerlo visible, definirlo y medirlo.
- **Administración y control de proyectos de software**
 - uso de métricas, recursos y tiempos
 - manejo del riesgo
 - manejo de los cambios (Software Configuration Management)
 - control de la calidad

Conclusiones (cont...)

- **Análisis y especificación de requisitos**
 - no queremos software que no se use
 - no queremos usuarios descontentos
- **Diseño de software**
 - queremos diseños que respeten los requisitos
 - queremos diseños realistas
- **Verificación y validación**
 - queremos productos confiables
 - queremos productos que satisfagan las especificac.
- **Apoyo del computador**
 - queremos hacer todo esto de la manera más eficiente posible