



Università  
Ca' Foscari  
Venezia

# INGEGNERIA DEL SOFTWARE

## AA. 2014/'15

**Tino Cortesi**

Dipartimento di Informatica  
Università Ca' Foscari  
Venezia



Università  
Ca' Foscari  
Venezia

# Obiettivi del corso

- Il corso di Ingegneria del Software presenta i metodi, le tecniche e gli strumenti fondamentali di documentazione della specifica, analisi e progetto di sistemi software **complessi** da un punto di vista architetturale
- A lezione daremo particolare spazio alla parte di analisi/specifica dei requisiti e progettazione usando UML.
- Il corso richiederà la collaborazione attiva in un progetto unico, strutturato in gruppi, nel quale ci “giocheremo la faccia” in termini di professionalità.



Università  
Ca' Foscari  
Venezia

# Competenze in Uscita

- Riconoscere un modello di processo
- Avere un'idea dei problemi affrontati dalla ingegneria dei sistemi
- Conoscere problemi, attività e prodotti della ingegneria dei requisiti
- Conoscere le principali attività e metodi di gestione dei progetti di sviluppo del software
- Avere le nozioni di base della definizione dei requisiti del software e saper specificare semplici requisiti
- Aver acquisito le nozioni di base della progettazione del software e della documentazione relativa
- Conoscere le principali attività e metodi di verifica e validazione
- Conoscere le principali attività e metodi di stima della qualità del software, del processo e della gestione del progetto.



Università  
Ca' Foscari  
Venezia

# Strumenti di lavoro

## **Area wiki riservata:**

[http://blogs.unive.it/groups/software\\_engineering/](http://blogs.unive.it/groups/software_engineering/)

**Lezioni:** Lunedì e Martedì dalle 8.45 alle 10.15 (puntuali)

**Ricevimento:** su appuntamento (via e-mail)



Università  
Ca' Foscari  
Venezia

# Il progetto in breve...

- Obiettivo:  
Realizzazione di un sito web di tipo social a supporto della messa in rete e della diffusione delle diverse iniziative promosse a livello regionale e nazionale che hanno come finalità o effetto la riduzione del gender gap in ambito ICT (**“informatica sarà lei!”**)
- Modalità:
  - Progettazione, realizzazione e messa in esercizio del sito
  - Raccolta e messa in linea della documentazione e di strumenti di alimentazione semiautomatica dei contenuti
  - Realizzazione di una campagna virale avente come target le studentesse delle scuole superiori del Veneto
- Progetto in collaborazione con la Commissione pari opportunità della Regione Veneto
- **<http://www.score-contest.it/>**



Università  
Ca' Foscari  
Venezia

# Testi di riferimento

- Roger Pressmann,  
**Principi di Ingegneria del  
Software**, quinta ed.  
McGraw Hill  
ISBN 9788838662164

oppure

- Ian Sommerville,  
**Ingegneria del Software**,  
settima ed.  
Pearson Education,  
ISBN 9788871923543





Università  
Ca' Foscari  
Venezia

# Modalità d'esame

## Per chi frequenta

1. Essere presente alle lezioni e ai meeting di progetto
2. Rispettare le consegne intermedie:
  - Presentazione proposte (30/09/2014)
  - Piano di progetto (15/10/2014)
  - Documento di analisi e specifica (07/11/2013)
  - Documento di progettazione (30/11/2013)
3. Completare la realizzazione e l'integrazione (28/02/2015)

**Valutazione:** 50% qualità documentazione  
30% qualità del prodotto finale  
20% frequenza (valutata con test a sorpresa)

## Per chi non frequenta:

Prova orale: discussione della documentazione completa relativa alla realizzazione di uno strumento specifico che realizzi un servizio da integrare nel progetto.

**Valutazione:** 60% qualità documentazione  
40% qualità del prodotto finale



Università  
Ca' Foscari  
Venezia

# Piano delle lezioni

- Introduzione all'IS
- Ingegneria di Sistema
- Project Management
- Struttura Organizzativa
- Il Piano di Business
- Il Piano di Progetto
- Modelli di processo
- Analisi dei Requisiti
- Documento dei Requisiti
- Modelli UML
- Principi di Progettazione del Software
- Progettazione Architettuale
- Modelli di progettazione Orientata ad Oggetti
- Progettazione di Interfacce Utenti
- Metodi di Verifica e Validazione
- Defect Testing - Tecniche di Inspection e Walkthrough
- Analisi Statica





Università  
Ca' Foscari  
Venezia

# Perché un corso di Ingegneria del Software?

- E' importante distinguere:
  - i sistemi semplici (uno sviluppatore, un utente, prodotto sperimentale)
  - dai sistemi “hard” (molti sviluppatori, molti utenti, “prodotto vero”)
- L'esperienza acquisita nello sviluppo di sistemi semplici non è scalabile! Analogia con la costruzione di un ponte:
  - Su un ruscello = facile, basta una persona
  - Sul ponte di Messina ... ?
- Il problema è la complessità
  - Un telefonino contiene 5+ MLOC (fonte Nokia)
  - Windows XP contiene 40+ MLOC (Windows 95: 11 MLOC)
- Focus sul lavoro di team



Università  
Ca' Foscari  
Venezia

# (nuove?) Categorie di software

- Cloud computing software
- Social software
- Web Services
- Software “mobile” e agenti (es. applet)
- Data mining
- Open source



Università  
Ca' Foscari  
Venezia

# Esempio: sw per ideogiochi

- Sforzo tipico: 100÷500 anni/persona
- Team: di solito 50 ÷ 100 persone (Assassin Creeds 2009: 450 persone)
- Vendere un milione di copie è ok ma non eccellente



Università  
Ca' Foscari  
Venezia

# Perchè preoccuparsi?

Robert Cringely (giornalista scientifico):

“...se l'automobile avesse seguito lo stesso sviluppo del computer,  
una Rolls-Royce costerebbe oggi 100 \$,  
farebbe un milione di chilometri con 5 litri...

**ed esploderebbe una volta all'anno  
causando la morte di tutti i passeggeri**





Università  
Ca' Foscari  
Venezia

#	Company	Software Revenues <i>mln US\$</i>	Software Revenue growth	Total Revenues <i>mln US\$</i>	Software Revenue share
1	<b>Microsoft &gt;</b>	54,270	11%	67,383	81%
2	<b>IBM &gt;</b>	22,485	5%	99,870	23%
3	<b>Oracle &gt;</b>	20,958	13%	30,180	69%
4	<b>SAP &gt;</b>	12,558	11%	16,654	75%
5	<b>Ericsson &gt;</b>	7,274	-4%	30,307	24%
6	<b>HP &gt;</b>	6,669	8%	126,562	5%
7	<b>Symantec &gt;</b>	5,636	1%	6,013	94%
8	<b>Nintendo &gt;</b>	5,456	-20%	13,766	40%
9	<b>Activision Blizzard &gt;</b>	4,279	-7%	4,279	100%
10	<b>EMC &gt;</b>	4,356	10%	17,015	26%
11	<b>Nokia Siemens Networks&gt;</b>	4,229	-7%	16,918	25%
12	<b>CA &gt;</b>	4,136	3%	4,454	93%
13	<b>Electronic Arts &gt;</b>	3,413	-8%	3,413	100%
14	<b>Adobe &gt;</b>	3,177	14%	3,826	83%
15	<b>Alcatel-Lucent &gt;</b>	2,561	-5%	21,374	12%
16	<b>Cisco &gt;</b>	2,383	12%	41,045	6%
17	<b>Sony &gt;</b>	2,083	9%	83,039	3%
18	<b>Hitachi &gt;</b>	1,939	22%	113,500	2%
19	<b>Dassault &gt;</b>	1,885	19%	2,090	90%
20	<b>BMC &gt;</b>	1,843	5%	1,981	93%
21	<b>SunGard &gt;</b>	1,762	-12%	4,992	35%
22	<b>Autodesk &gt;</b>	1,701	9%	1,932	88%
23	<b>Konami &gt;</b>	1,643	3%	3,122	53%
24	<b>Salesforce.com &gt;</b>	1,523	28%	1,628	94%
25	<b>Sage &gt;</b>	1,485	-5%	2,228	67%

Dati in milioni di dollari, 2011. fonte: <http://www.softwaretop100.org/>

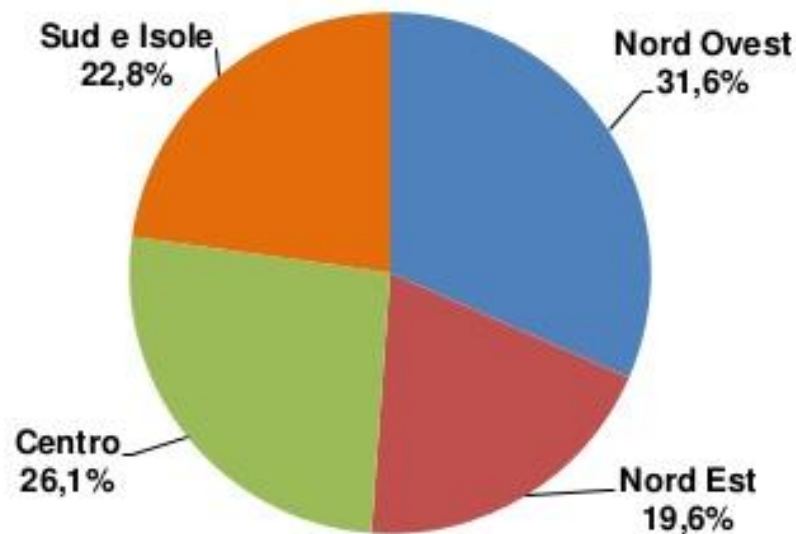
# Il contesto

## Caratteristiche strutturali del settore ICT in Italia (2012)

### Numerosità

<b>N° imprese attive</b>	<b>89.042</b>
<b>Addetti</b>	<b>464.562</b>
<b>Dimensione media (N° addetti per impresa)</b>	<b>5,2</b>

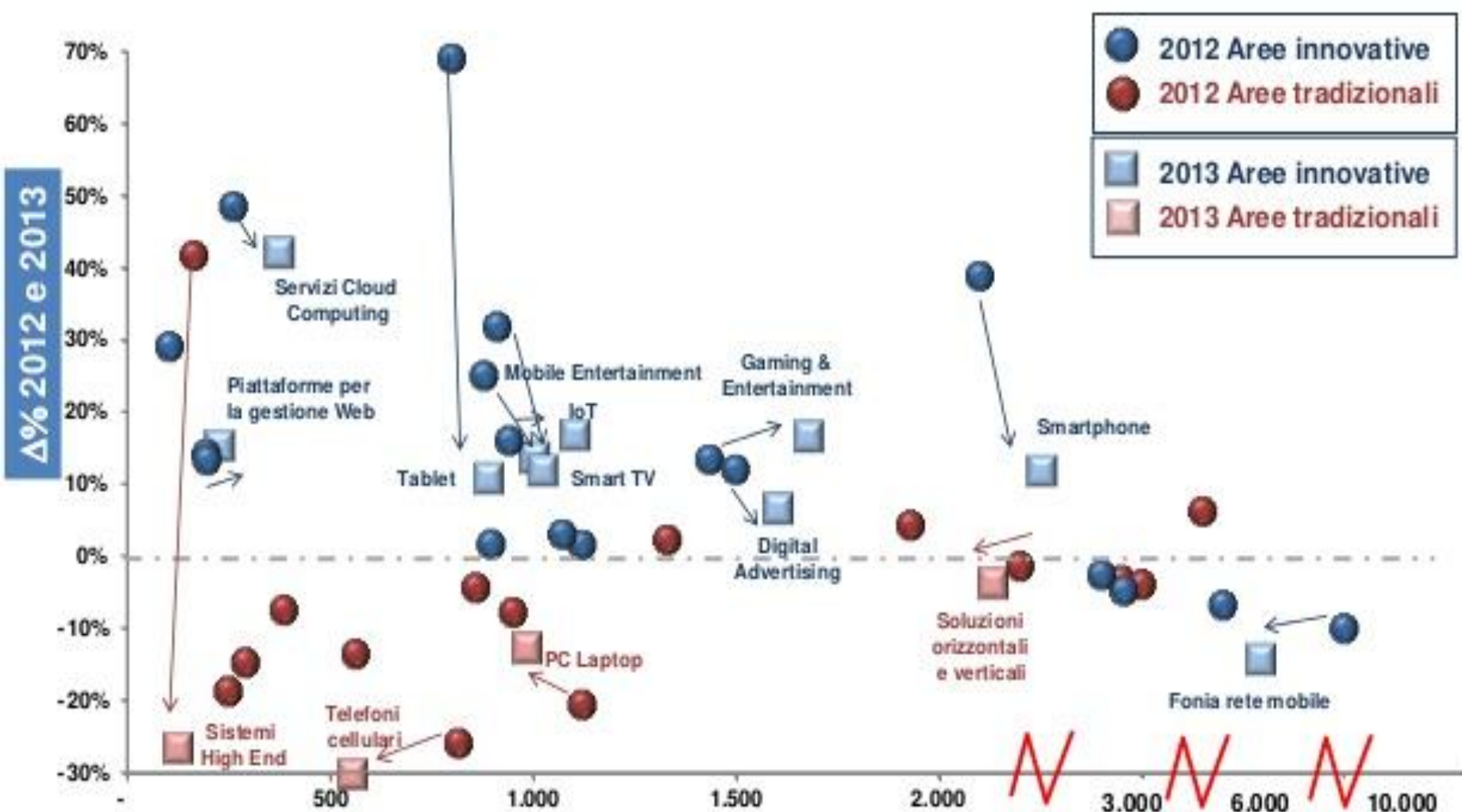
### Distribuzione geografica



Fonte: Elaborazioni NetConsuting su fonti varie



# L'andamento dei principali prodotti e servizi del Global Digital Market (2012/2013)



Valore del mercato 2012/2013 M€

Milano, 13 giugno 2013

5



Assinform

# L'Universo delle Apps

N° Apps scaricate (mld)

Valore mercato Apps nel mondo (Mld\$)



\$ 15,5  
2012



Ore spese in media / gg per utente



63%  
Tasso  
refresh Apps

N° medio Apps usate  
8

Categorie  
più sviluppate



A quali device si rivolgono



11 Mln utenti

Fonte: Elaborazioni NetConsulting su Gartner e GIGAOM Pro





Università  
Ca' Foscari  
Venezia

# Ingegneria del Software

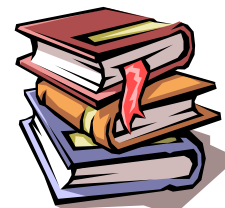
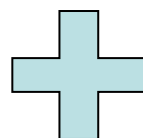
**Necessità di  
ingegnerizzazione**



Categoria	Numero tipico di programmatori	Durata tipica	Dimensione del prodotto in righe di codice	Esempi	Analogia con l'edilizia
Banale	1	1–2 settimane	< 500	Compiti a casa per studenti	Piccola ristrutturazione della casa
Piccolo	1–3	Poche settimane o mesi	500–2 000	Progetti di gruppo per studenti, compiti avanzati	Aggiunta di un locale
Medio	2–5	Da qualche mese a un anno	2 000–10 000	Progetti di ricerca, semplice software di produzione come assembleri, editor, applicazioni ricreative ed educative	Casa monofamiliare
Grande	5–25	1–3 anni	10 000–100 000	La maggior parte delle applicazioni attuali – elaboratori di testi, fogli elettronici, sistemi operativi per piccoli computer, compilatori	Piccolo supermercato
Molto grande	25–100	3–5 anni	100 000–1 milione	Sistemi di prenotazione per aerei, sistemi di controllo del magazzino per multinazionali	Grande palazzo di uffici
Grandissimo	>100	> 5 anni	> 1 milione	Sistemi operativi real-time su larga scala, sistemi militari avanzati, reti di telecomunicazioni internazionali	Grattacielo imponente

# Cos'è un prodotto software?

- Qualcosa di più di un insieme di linee di codice...



- Un insieme di linee di codice, strutturato in packages
- Tutta la documentazione che descrive la struttura del sistema
- I dati di configurazione, che permettono di installarlo
- Il manuale utente



# Prodotti software: una prima classificazione

1. System software – operating systems, drivers, compilers, etc.
2. Application software – Custom business apps.
3. Engineering/Scientific software – Mentor Graphics, ANSYS.
4. Embedded software – Cell phones, PDAs.
5. Product software – Word, Excel.
6. Web Applications
7. Artificial Intelligence software



Università  
Ca' Foscari  
Venezia

# Prodotti Software: un'altra classificazione

- **Prodotti Generici**
  - Sistemi stand-alone prodotti da un'organizzazione di sviluppo e venduti sul mercato ad ogni cliente
- **Prodotti Dedicati**
  - Sistemi che sono commissionati da un cliente specifico e sviluppati appositamente
- La maggior spesa di software riguarda sistemi generici, ma il maggior sforzo di sviluppo è su prodotti dedicati
- La differenza principale? Chi dà la specifica del prodotto (il produttore o il consumatore).

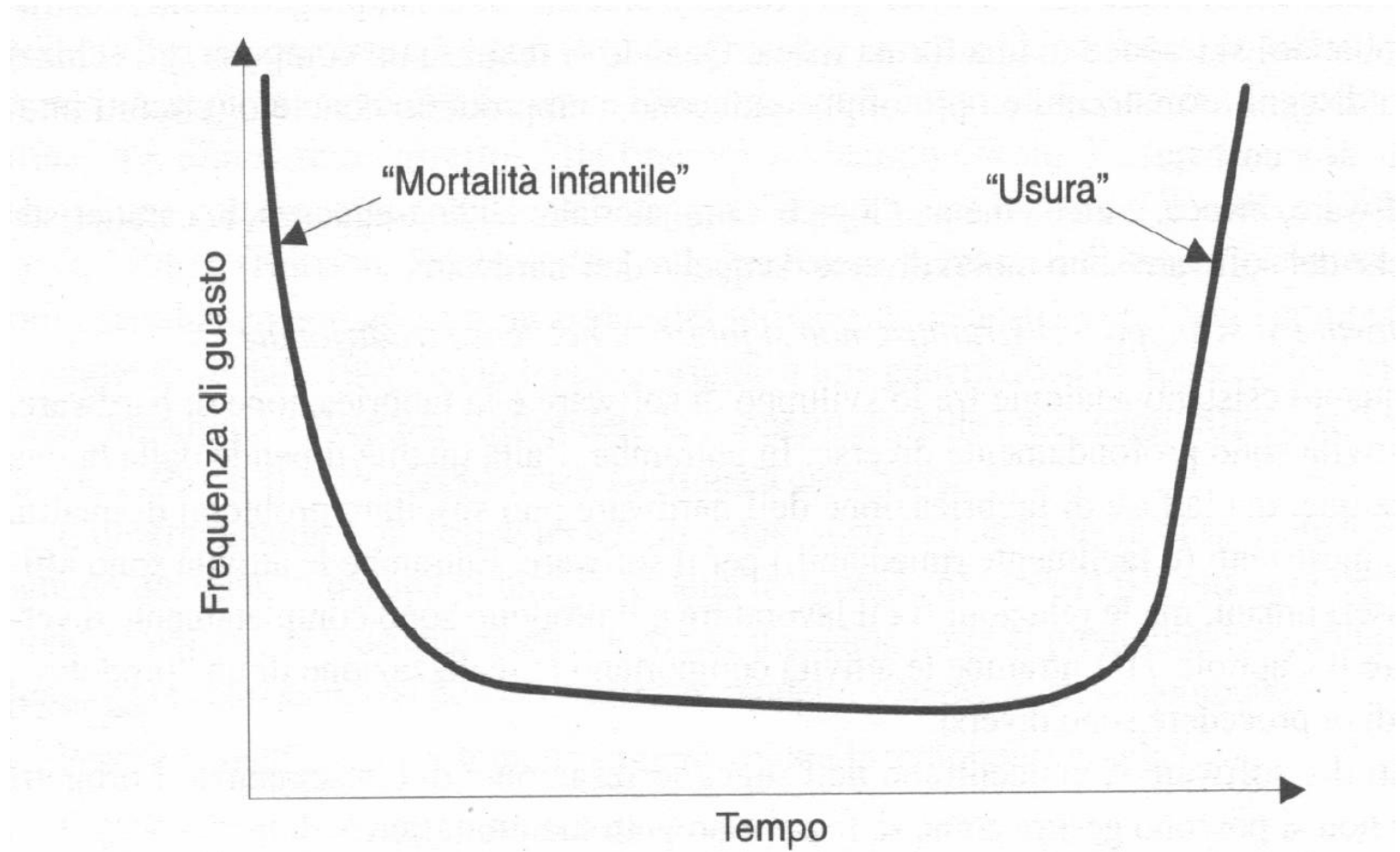


Università  
Ca' Foscari  
Venezia

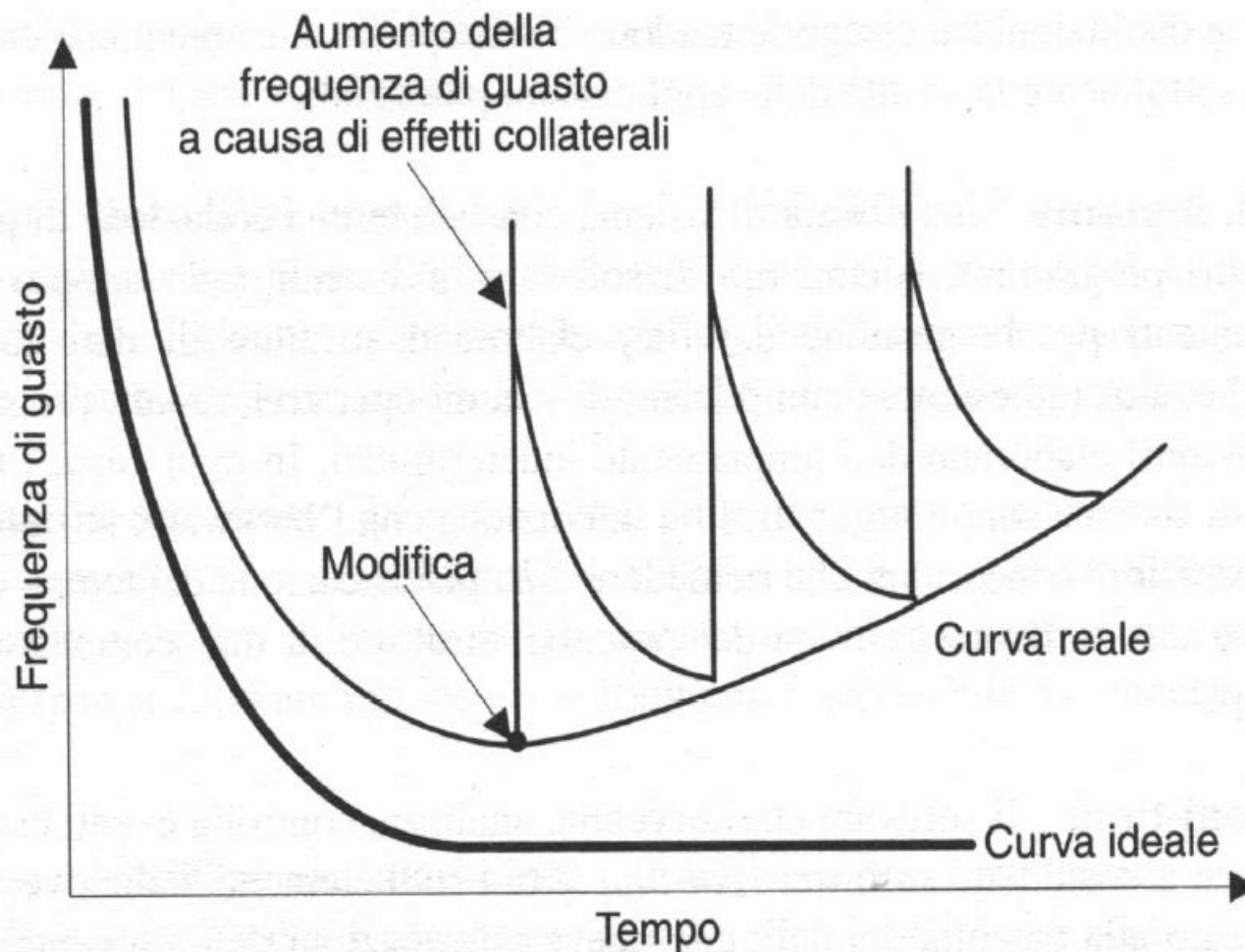
# Le caratteristiche del prodotto software

- Il software è un prodotto speciale...
- È invisibile e intangibile
- È facilmente duplicabile e distribuibile su rete
- In Europa non è brevettabile (ma protetto)
- Il software di consumo non è garantito
- Viene acquisito su licenza
  - Proprietaria (normale, shareware, freeware)
  - Public domain
  - Open source

# Curva dei guasti per l'hardware



# Curva dei guasti per il software





Università  
Ca' Foscari  
Venezia

# Alcuni dati

Numero di difetti (fault) rilevati durante l'esercizio

- I peggiori sistemi militari: 55 faults/KLoC
- I migliori sistemi militari: 5 faults/KLoC
- Prodotti ottenuti con sviluppo agile (XP): 1.4 faults/KLoC
- Apache web server (open source): 0.5 faults/KLoC
- NASA Space shuttle: 0.1 faults/KLoC

Fonte: [www.easterbrook.ca/steve/?p=1366](http://www.easterbrook.ca/steve/?p=1366)





Università  
Ca' Foscari  
Venezia

# Cos'è l'Ingegneria del Software?

- Definizione IEEE:

“The application of systematic, disciplined,  
quantifiable approach to the development,  
operation, and maintenance of software;  
that is,  
the application of engineering to software.”

# Cos'è l'ingegneria del software?

- “Software engineering” è una disciplina che cerca di fornire le **regole** per il processo di produzione del software
- Un ingegnere del software dovrebbe:
  - adottare un approccio sistematico e organizzato al proprio lavoro
  - usare strumenti e tecniche appropriate, che dipendono dal problema che deve essere risolto, dai vincoli presenti e dalle risorse disponibili.



# Attributi di qualità di un prodotto software

Le qualità su cui si basa la valutazione di un sistema software possono essere

- *interne*, se riguardano le caratteristiche legate alle scelte implementative e non sono visibili agli utenti;
- *esterne*, se riguardano le funzionalità fornite dal sistema e sono visibili agli utenti.

Le due categorie sono legate, infatti non è possibile ottenere qualità esterne se il sistema non gode di qualità interne.



Università  
Ca' Foscari  
Venezia

# Attributi di qualità di un prodotto software

**Correttezza** - un sistema è corretto se rispetta le specifiche.

**Affidabilità (reliability)** - un sistema è affidabile se l'utente può dipendere da esso.

**Robustezza** - un sistema è robusto se si comporta in modo ragionevole anche in circostanze non previste dalle specifiche.

**Efficienza** - un sistema è efficiente se usa bene le risorse di calcolo.

**Facilità d'uso** - un sistema è facile da usare se l'interfaccia che presenta all'utente gli permette di operare in modo naturale.



Università  
Ca' Foscari  
Venezia

## Qualità del software (ctd.)

**Verificabilità** - un sistema è verificabile se le sue caratteristiche sono verificabili.

**Riusabilità** - un sistema è riusabile se può essere usato, in tutto o in parte, per costruire nuovi sistemi.

**Portabilità** - un software è portabile se può funzionare su più piattaforme hardware/software.

**Facilità di manutenzione** - un sistema è facile da mantenere se

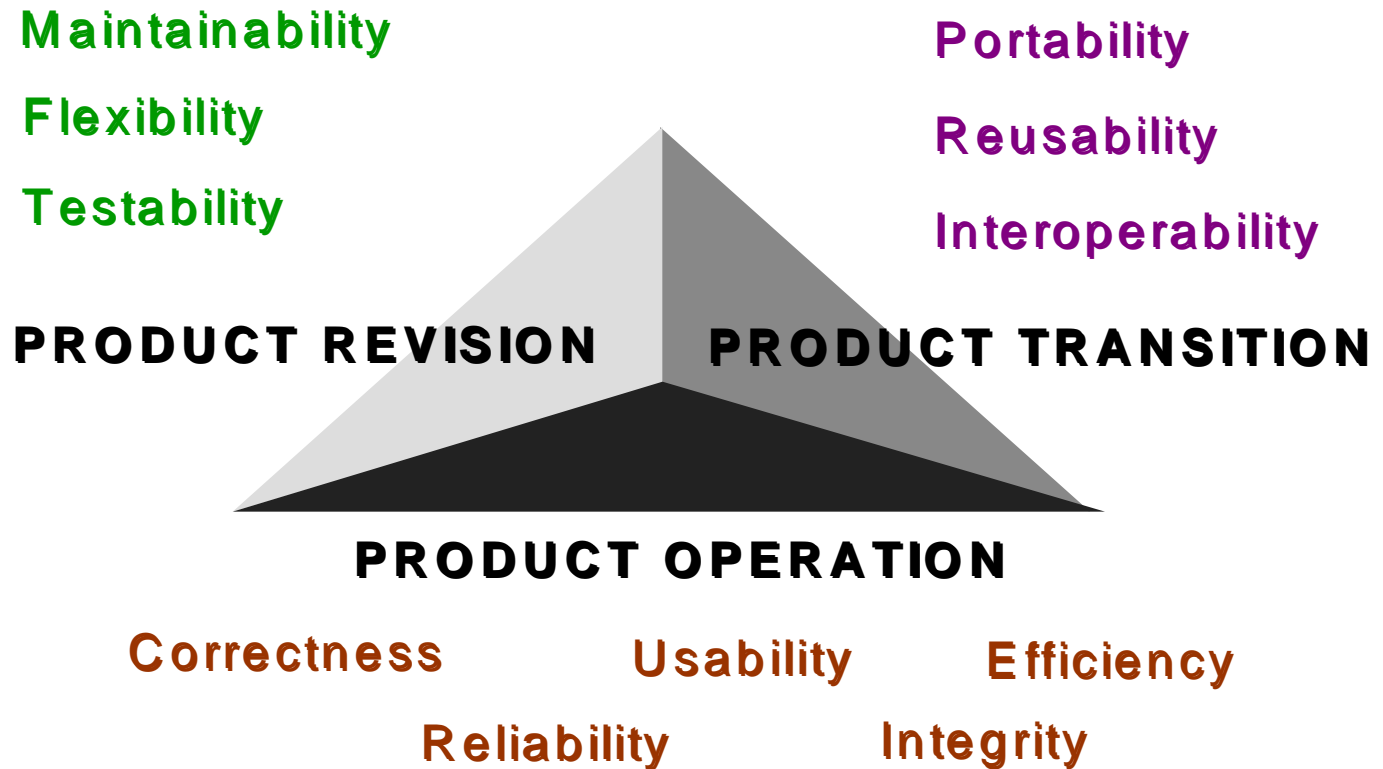
- è strutturato in modo tale da facilitare la ricerca degli errori,
- la sua struttura permette di aggiungere nuove funzionalità al sistema,
- la sua struttura permette di adattarlo ai cambiamenti del dominio applicativo.

**Interoperabilità** - abilità di un sistema di cooperare con altri sistemi, anche di altri produttori.



Università  
Ca' Foscari  
Venezia

# Il triangolo di McCall





Università  
Ca' Foscari  
Venezia

# Ingegneria del software e informatica

- L'informatica è una scienza: il “cuore” sono i fondamenti teorici: linguaggi – algoritmi – complessità – formalismi ecc.
- L'ingegneria del software ha a che fare con aspetti più “pratici”: come pianificare e sviluppare la produzione di software di qualità.
- Ad un ingegnere del software le conoscenze di base dell'informatica servono quanto la fisica ad un ingegnere elettrico



Università  
Ca' Foscari  
Venezia

# Processo di produzione software

- Il processo di produzione software è un insieme di attività il cui fine complessivo è
  - lo **sviluppo** di un prodotto software oppure
  - la **modifica** di un prodotto software





Università  
Ca' Foscari  
Venezia

# Attributi di qualità del processo di produzione software

- Comprensibilità
- Visibilità
- Supportabilità (CASE )
- Accettabilità
- Robustezza
- Mantenibilità
- Rapidità



Università  
Ca' Foscari  
Venezia

# Problemi nel processo di sviluppo del software

- Specifiche incomplete/incoerenti
- Mancanza di distinzione tra specifica, progettazione e implementazione
- Assenza di un sistema di validazione
- Il software non si consuma: la manutenzione non significa riparare alcune componenti “rotte”, ma modificare il prodotto rispetto a nuove esigenze

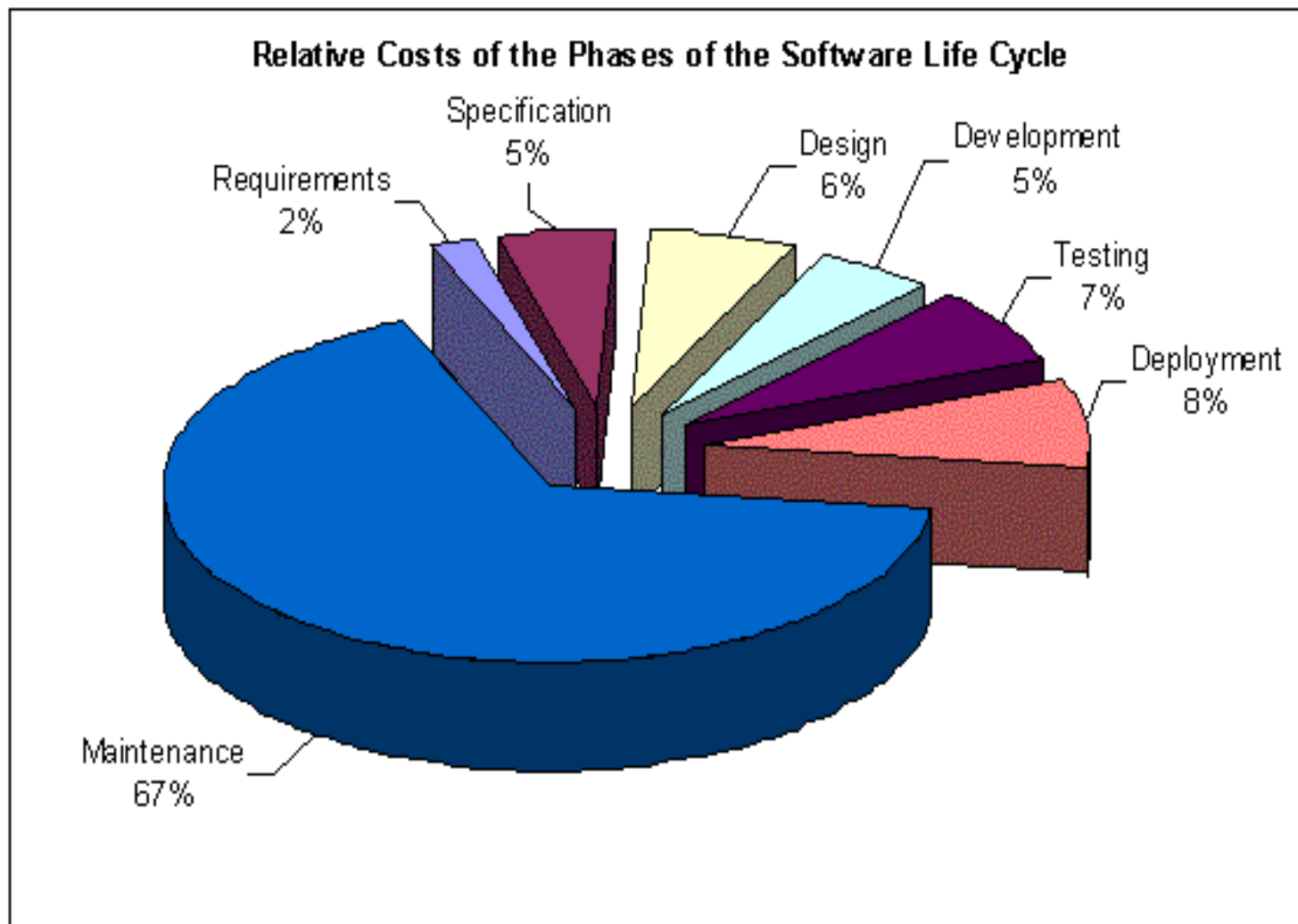


Università  
Ca' Foscari  
Venezia

# I costi di un prodotto software

- All'incirca il 60% dei costi è legato allo sviluppo, il 40% sono costi per la verifica e validazione (testing).
- I costi variano a seconda del tipo di sistema che deve essere sviluppato e da requisiti quali la performance o l'affidabilità del sistema.
- La distribuzione di costi nelle varie fasi del processo di produzione del software dipende dal modello di processo.

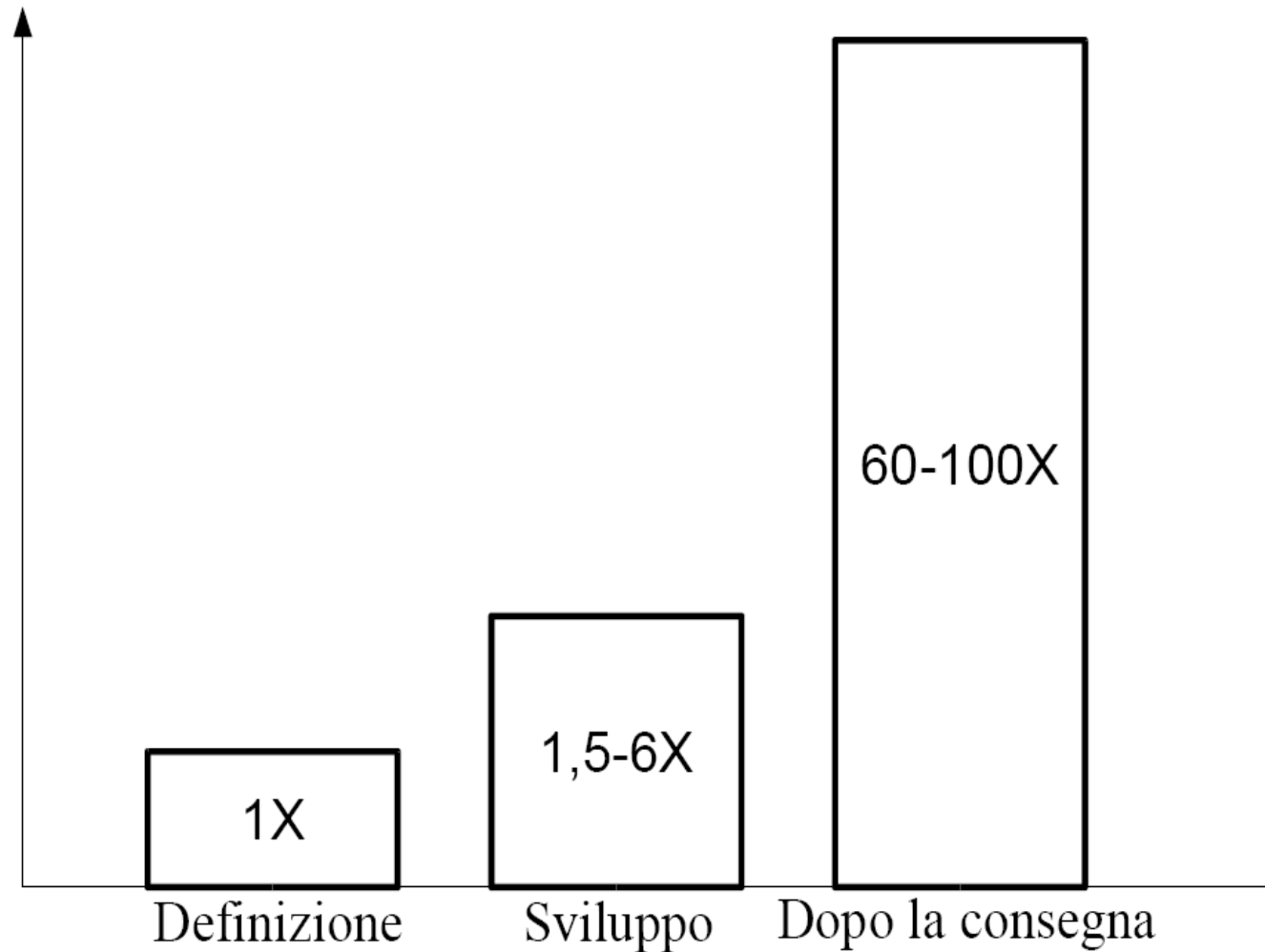
# Costi relativi nel ciclo di vita del sw





Università  
Ca' Foscari  
Venezia

# Costo di una modifica





Università  
Ca' Foscari  
Venezia

# Le sfide ed i problemi

- **Skill shortage**

Secondo una ricerca condotta da EITO nel 2003, in Europa ci sono 1 milione e 700 mila posti nell'Information & Communication Technology che non riescono ad essere coperti perché mancano le giuste risorse.

In Italia la carenza di risorse è calcolata in 167 mila unità.

- **Skill gap**

Problema ancor più grave è lo Skill gap, ossia la distanza fra le competenze tecniche richieste dal mercato ICT e la professionalità degli operatori del settore.

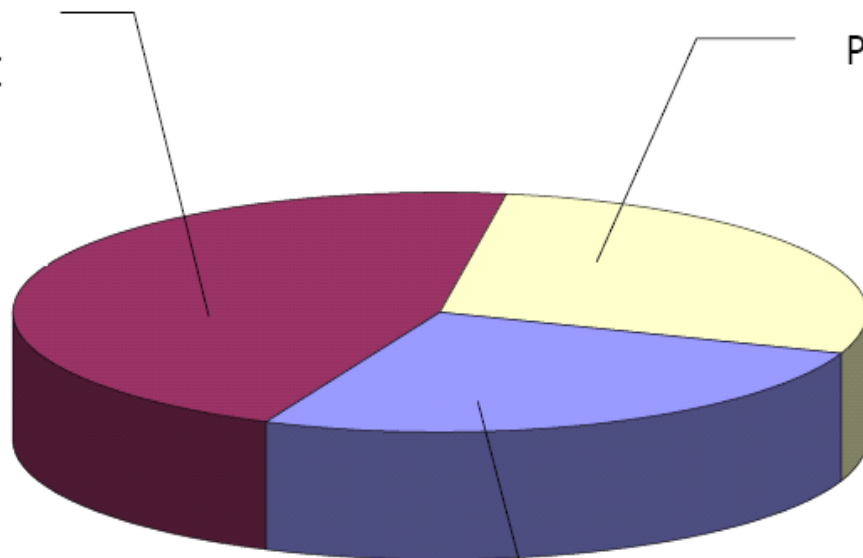
- **Alta percentuale di fallimento**

Indagine dello Standish Group, basata su un campione di 28.000 progetti e pubblicata da Computer Weekly il 9 luglio 1998:



Università  
Ca' Foscari  
Venezia

PROGETTI IN RITARDO  
CON COSTI IMPREVISTI  
CON FUNZIONALITA'  
INADEGUATE  
46%



PROGETTI FALLITI  
28%

PROGETTI RIUSCITI  
26%



Università  
Ca' Foscari  
Venezia

# Testi integrativi

- Sulle metodologie object-oriented in generale:
  - L. Maciaszek, Sviluppo di Sistemi Informativi con UML, Addison Wesley.
- Sull'ingegneria dei requisiti:
  - Wieringa, Design Methods for Reactive Systems, Elsevier.
- Sulla verifica e la validazione e stima dei costi:
  - Ghezzi, Jazayeri, Mandrioli, Ingegneria del Software, 2a Edizione, Pearson;
  - Pfleeger, Atlee, Software Engineering: International Edition:3/e Prentice Hall, 2006.
- Sulla notazione UML: Fowler, UML Distilled, Pearson, 2004
- Esempi commentati su UML:
  - Binato, Fuggetta, Sfardini, Ingegneria del Software: Creatività e metodo, Pearson, 2006;
  - Baresi, Lavazza, Pianciamore, Dall'idea al codice con UML 2 Pearson, 2006; L. Maciaszek, Sviluppo di Sistemi Informativi con UML, Addison Wesley;
  - Collana SHAUM - UML.





Università  
Ca' Foscari  
Venezia

# Riviste scientifiche

- *IEEE Transactions on Software Engineering*
- *ACM Trans. on Sw Engineering and Methodology*
- *IEEE Software*
- *ACM Software Engineering notes*
- *Software Practice and Experience*
- *Journal of Systems and Software*
- *IEEE Proceedings – Software*
- *Empirical Software Engineering*
- *Automated Software Engineering*
- *Science of Computer Programming*
- *Computer Languages, Systems and Software*
- ...



Università  
Ca' Foscari  
Venezia

# Siti

- Newsgroup: <nntp://comp.software-eng>
- Siti
  - [www.swebok.org](http://www.swebok.org) (SWE Body of Knowledge)
  - [www.ieee.org](http://www.ieee.org) (per gli standard IEEE del software)
  - [www.w3c.org](http://www.w3c.org) (per gli standard del software Web)
  - [www.omg.org](http://www.omg.org) (per gli standard del software a oggetti)
  - [www.oasis-open.org](http://www.oasis-open.org) (per gli standard del software business)
  - [www.softwarehistory.org](http://www.softwarehistory.org)



Università  
Ca' Foscari  
Venezia

# Concludendo

L'**ingegneria del software** tratta della realizzazione di sistemi software di dimensioni e complessità tali da richiedere uno o più team di persone.

L'ingegneria del software è la disciplina tecnologica e manageriale che riguarda la *produzione sistematica* e la *manutenzione* dei prodotti software che vengono sviluppati e modificati entro tempi e costi preventivati.

L'ingegneria del software è un insieme di teorie, metodi e strumenti, sia di tipo tecnologico che organizzativo, che consentono di produrre applicazioni con le desiderate caratteristiche di *qualità*.