

Ingegneria del Software

(2019-2020)

Laurea Triennale in Informatica

Prof. Paolo Ciancarini

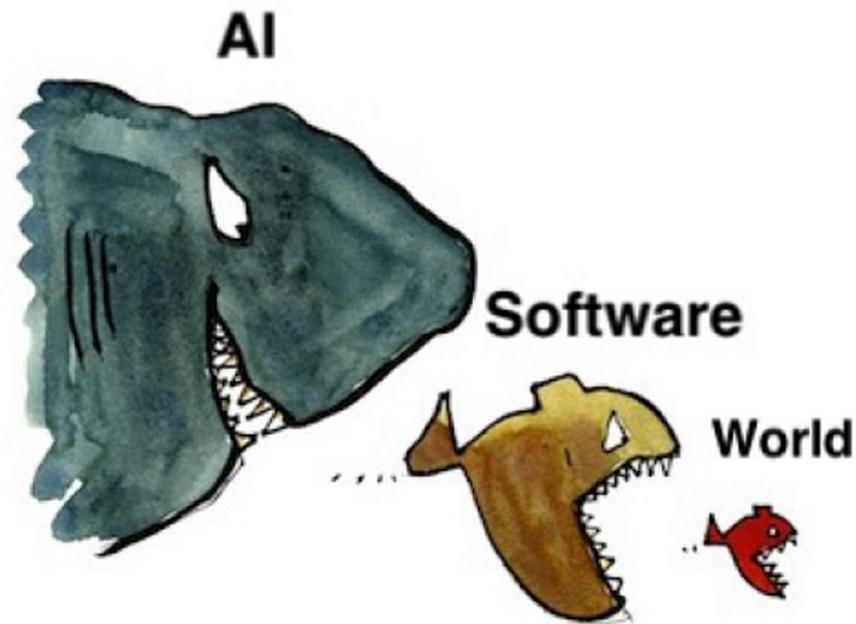
Email: `paolo.ciancarini@unibo.it`

Telegram: @PaLoCaPa

Software powers the world

(il software fa girare il mondo)

https://www.youtube.com/watch?v=F01JmJGJ9n8&fbclid=IwAR23HV3cn9kDY_Q8r4auAOfdBLtBz8xLBidbPXPfmG8FC0_EgUcYTtTBG64



Creare software è difficile

- È difficile progettarlo
- È difficile scriverlo
- È difficile leggerlo o descriverlo
- È difficile modificarlo
- È difficile misurarlo
- È difficile decidere se è fatto bene

Tutti i programmatori sono ottimisti

Frederick Brooks, The mythical man-month

Legge di Wirth: *«Il software diventa lento più rapidamente di quanto l'hardware diventi più veloce»*

Scopo del corso

Presentare e sperimentare *metodi e strumenti* di

- analisi,
- modellazione,
- progettazione, e
- misura

di prodotti, componenti e sistemi software

...con particolare riguardo ai metodi agili

Una storia vera

Il mio volo era in attesa di decollare quando il comandante fece un annuncio.

“Abbiamo un problema con l’aria condizionata. Su questo aereo il condizionatore controlla i livelli di ossigeno e dobbiamo essere sicuri che funzioni prima del decollo. Abbiamo provato a fare restart del condizionatore ma non funziona. Proviamo adesso a fare restart dell’intero aereo. Questi aerei moderni sono tutti controllati a software, e dunque non sono molto affidabili”

Il pilota spense tutto l’aereo e poi lo riaccese – ovvero fece reboot dell’aereo. Partimmo e tutto andò bene.

Fui molto contento di scendere da questo particolare volo

Il sistema MCAS degli aerei Boeing 737 è un software che corregge l'assetto dell'aereo e lo stabilizza.

Il suo (mal)funzionamento potrebbe aver causato gli incidenti in Etiopia (2019) e Indonesia (2018) – oltre 300 morti.

Sia le autorità federali statunitensi che il Congresso stanno indagando per capire se Boeing o la FAA avessero sottovalutato i rischi collegati all'uso di questo sistema.

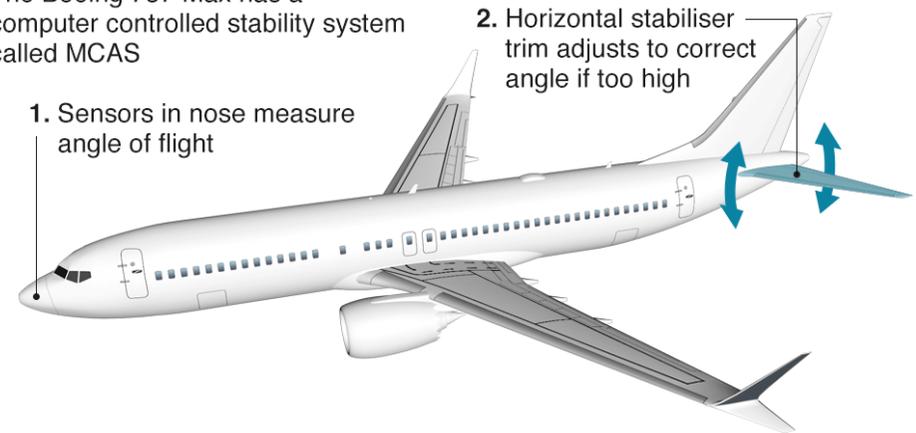
In particolare, vogliono capire se la FAA abbia valutato il MCAS rigorosamente, basandosi su comprovati standard ingegneristici e di design, oppure se, per favorire Boeing, abbia seguito scorciatoie che non hanno garantito adeguati livelli di sicurezza di volo o una sufficiente formazione ai piloti.

All'aereo Boeing di Ethiopian Airlines, così come a quello di Lion Air precipitato in Indonesia, mancava un sistema di sicurezza in grado di avvisare i piloti di eventuali problemi con il sistema MCAS.

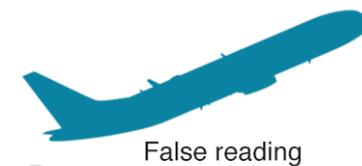
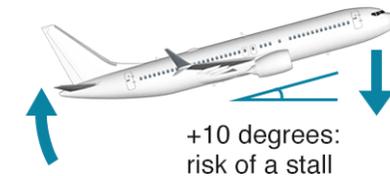
Il software di sicurezza, ritenuto non essenziale, [era venduto da Boeing come extra](#) e non era stato comprato né da Ethiopian Airlines né da Lion Air.

How the MCAS system works

The Boeing 737 Max has a computer controlled stability system called MCAS



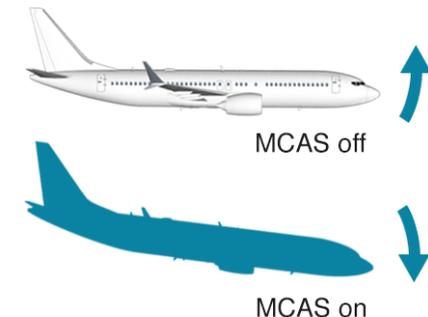
3. Nose pushed down to reduce risk of a stall



4. But if the sensor reading is wrong, MCAS may activate and push the nose down anyway

5. Pilots can temporarily switch off MCAS and pull up.

But system restarts if false readings continue, creating a tug of war between the aircraft and its crew



Come sviluppi il tuo sw?

- Come un falegname? Come un esploratore? Come uno scienziato? Come un archeologo? Come un architetto? Come un mercante?
- *"During software design, I'm an architect. When I'm designing the user interface, I'm an artist. During construction, I'm a craftsman. And during unit testing, I'm one mean son of a bitch!"*

O come ...

- [Roberto Bolle](#)
- [Uma Thurman and John Travolta](#)
- [The Rockettes](#)

Come ti piace programmare?

- Da solo?
- In coppia?
- In team?

Legge di Eagleson

*Qualsiasi codice che non hai rivisto
negli ultimi sei mesi o più
ti sembrerà estraneo,
come se l'avesse scritto qualcun altro*

Legge di Conway

Le organizzazioni che progettano sw sono indotte a generare design che sono copie delle strutture di comunicazione di tali organizzazioni

Alcune domande

- Come si *progetta* un prodotto software?
- Quali *strumenti* sono disponibili per chi costruisce prodotti software?
- Quanto *costa* costruire il software?
- Come si valuta la *qualità* del software?

Requisiti
del software

Design
del software

Costruzione
del software

Processo
di sviluppo

Testing
del software

Evoluzione
del software

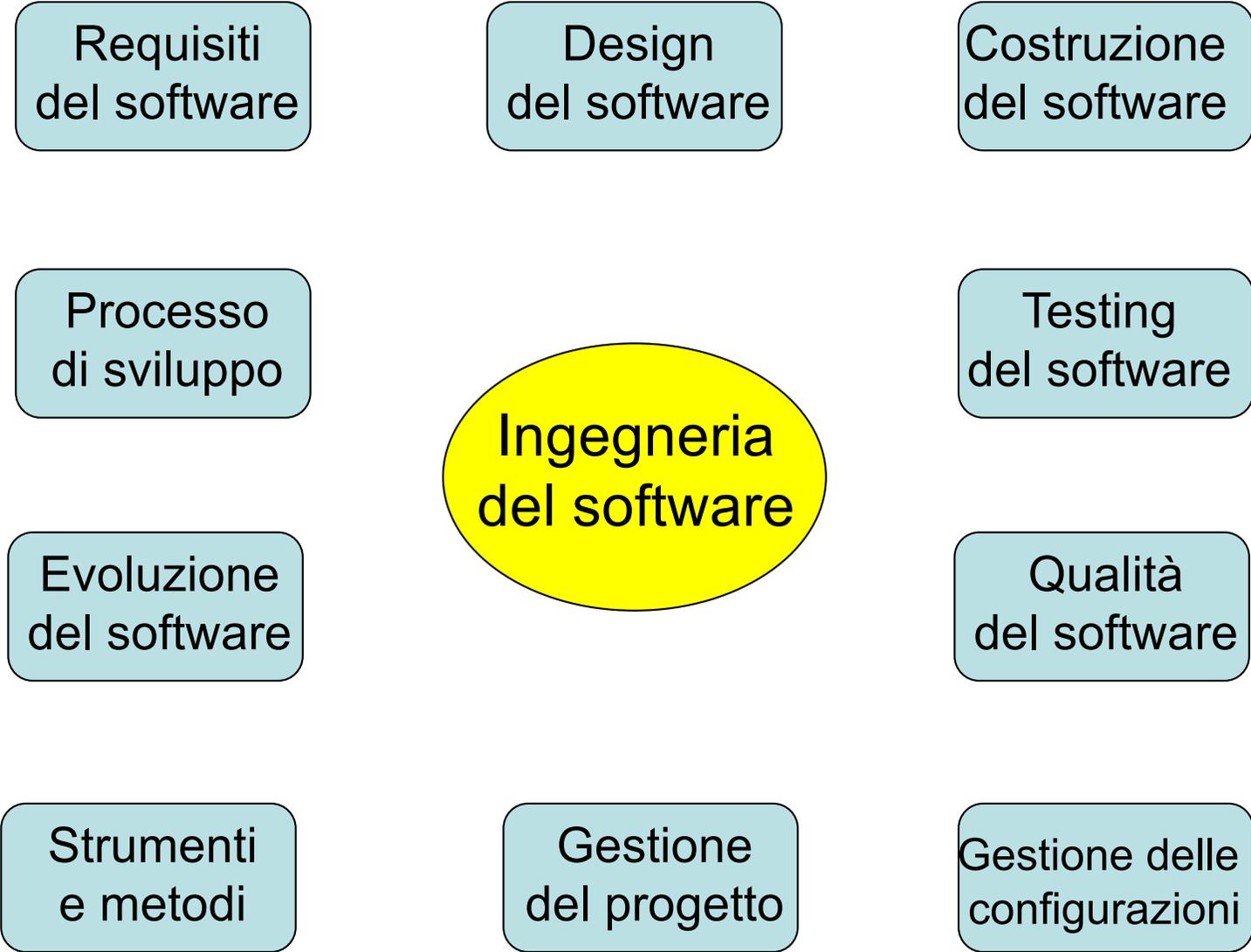
Qualità
del software

Strumenti
e metodi

Gestione
del progetto

Gestione delle
configurazioni

Ingegneria
del software



Corsi correlati

Prequel (prerequisiti):

- Programmazione, Sistemi operativi

Sequel (nella laurea magistrale):

- Architetture software

Agenda

- Gli standard di produzione del software
- Il ciclo di vita dei prodotti software
- I metodi agili
- L'analisi dei requisiti
- La progettazione del software
- La modellazione del software con UML
- Gestione di progetti software
- Controllare e misurare la qualità del software
- L'evoluzione del software

Tempi del corso

Da settembre a dicembre

Lezioni settimanali:

- Lunedì ore 13 $\times 2h$ aula Ercolani 2
- Martedì ore 15 $\times 3h$ aula Ercolani 2
- Mercoledì ore 14 $\times 3h$ aula Ercolani 2

Alcuni strumenti

- Telegram, facebook
- Strumenti per modellare con UML
- Powerpoint – o equivalente

Materiale didattico

Testi:

- Larman, *Applicare UML e i pattern*, 4ed., Pearson, 2016
- slide presentate a lezione

Testi aggiuntivi:

Sommerville, *Ingegneria del software*, Pearson, 2017

Canali di conversazione

Gruppo fb: [Corso Ingegneria del software UniBo](#)

Gruppo Telegram: [UniBoSWE](#)

Sito web:

www.cs.unibo.it/~cianca/wwwpages/LabIdSw.html

Mailing list: `swe at lists.cs.unibo.it`

Modalità di esame

Esame:

scritto (+ orale a richiesta)

presentazione su argomento concordato

Voto:

2/3 scritto o progetto,

1/3 presentazione (formato latex, pwp o equivalente),

Bonus per attività in classe

Valore dell' esame: 6 CFU

Argomento formato e consegna della presentazione

- **Argomento a scelta dello studente**, previa mia approvazione
- Scegliere un articolo “recente” (= in stampa dal 2010 a oggi) tratto dalle riviste internazionali di ricerca su Ingegneria del Software
 - IEEE Transactions on Software Engineering
 - <http://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=32>
 - ACM Transactions on Software Engineering and Methodology
<https://dl.acm.org/citation.cfm?id=J790>
- **Formato**: latex, powerpoint o equivalente
- **Consegna per email a me** con subject [presentazione IdSw] in formato pdf e titolo CognomeMatricola, due giorni prima dell’esame scritto

Attività in classe

- Test a risposte chiuse, senza voto ma con bonus finale
- Per ottenere il bonus bisogna partecipare a più della metà dei test (di solito ne preparo almeno sette, quindi occorre partecipare almeno a quattro)
- La partecipazione ai test dà diritto ad un “bonus” da 1 a 4 punti che viene aggiunto al voto finale
- Il bonus è concesso solo se si fa l’esame nei primi due appelli, a gennaio/febbraio

Importante

- Questo corso si supera **facilmente** seguendo le lezioni; è **difficile** da superare se non si frequentano le lezioni
- **Copiare** (i compiti scritti, le relazioni) è **vietato** e quando si viene scoperti l'esame viene annullato e diventa **impossibile** da superare

Domande?



Legge di Paolo:
*L'unica domanda stupida
è quella che non si fa*