

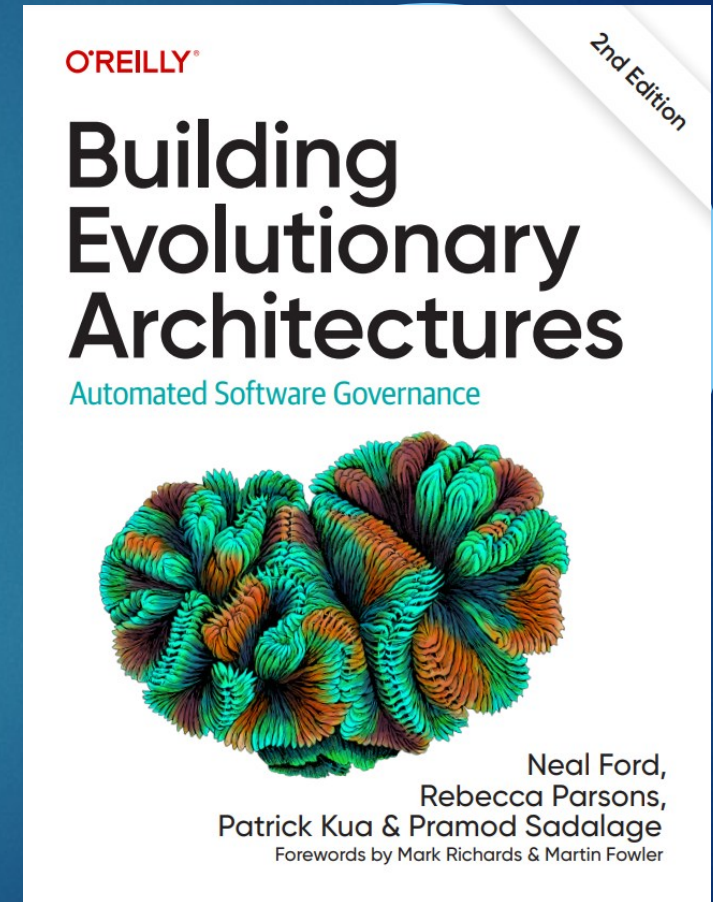


Architetture software: una corsa contro il tempo

MANUEL BONINI: SOFTWARE DEVELOPER @ INTRÉ S.R.L.

Building Evolutionary Architectures

- ▶ Neal Ford: direttore e software architect @ ThoughtWorks con anni di esperienza come CTO @ The DSW Group
- ▶ Rebecca Parsons: CTO @ ThoughtWorks con esperienza decennale nello sviluppo di applicazioni
- ▶ Patrick Kua: CTO coach con esperienza nell'industria informatica per più di vent'anni
- ▶ Pramod Sedalage: direttore di Data & DevOps @ ThoughtWorks



Perché architettura evolutiva?

*Building systems that **age** gracefully and effectively is one of the enduring challenges of software development generally and software architecture specifically.*

*Bit rot: ..., is either a slow deterioration of software quality **over time** or its diminishing responsiveness that will eventually lead to software becoming faulty.*

L'ecosistema tecnologico

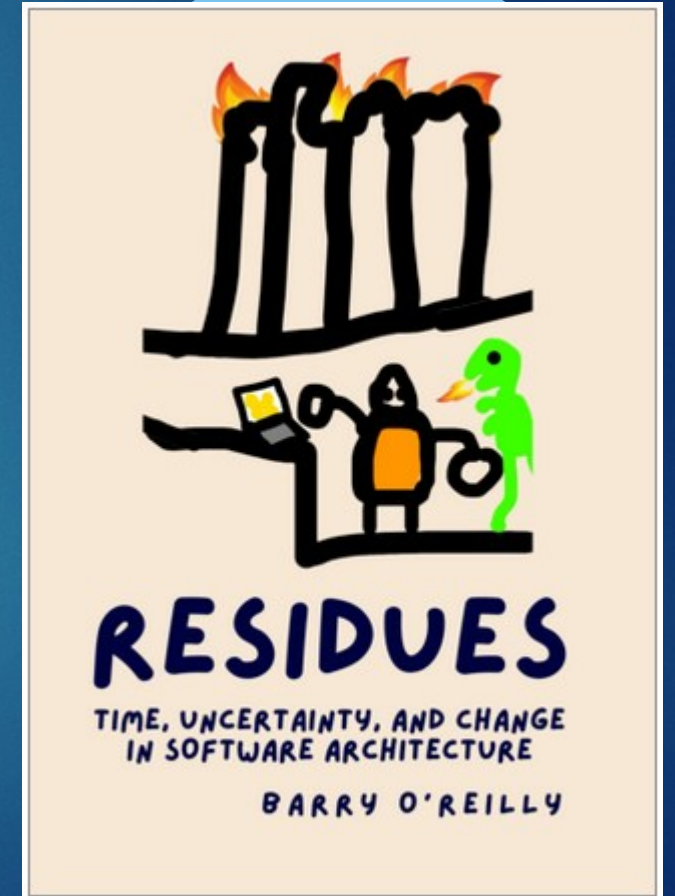
*The software development ecosystem **constantly evolves**, which leads to new architectural approaches.*

Caratteristiche architetturali

Software is rarely static; it *continues to evolve* as teams add new features, integration points, and a host of other common changes.

Residues: Time, Uncertainty and Change in Software Architecture

- ▶ Barry O'Reilly: Fondatore della Black Tulip Technology. Precedentemente è stato Chief Architect per Microsoft nella sezione West Europe e responsabile mondiale della community Microsoft's Solution Architecture.



Dipendenza dal business

Every architecture involves a software application which must live its life and execute inside a business environment.

...

*This **constant change** affects the architecture.*

Il problema dell'ingegneria classica

This paradigm - the engineering paradigm – has been the source of a great deal of the issues in software.

*Instead of learning to deal with an environment in **constant flux** we doubled down on prediction and control—attempting...*

Tutta una questione di tempo

Building systems that *age* gracefully

The software development ecosystem
constantly evolves

Software ...*continues to evolve*

constant change affects the architecture

an environment in *constant flux*

Ma perché il tempo?

- ▶ Il mercato cambia caratteristiche
- ▶ Le tecnologie cambiano
- ▶ Gli utilizzatori si abituano

Sviluppiamo software per persone

Anche l'architetto è una persona



Analysis paralysis:

- ▶ Impossibile enumerare tutte le possibilità
- ▶ L'analisi non può essere infinita perchè sarà vecchia
- ▶ Un progetto che non comincia non esiste

Facciamo il meno peggio



Ci resta che crescere...

- ▶ Essendo il tempo limitato per cominciare
- ▶ Accettando il fatto che il tempo invecchia il software

Stiamo sempre all'erta

Voi cosa ne pensate?

Anche secondo voi architettare software è
una corsa contro il tempo?