

ARCHITECTURE EVOLUTIONS

G. Molines
2018-2019



TIME EVOLUTION

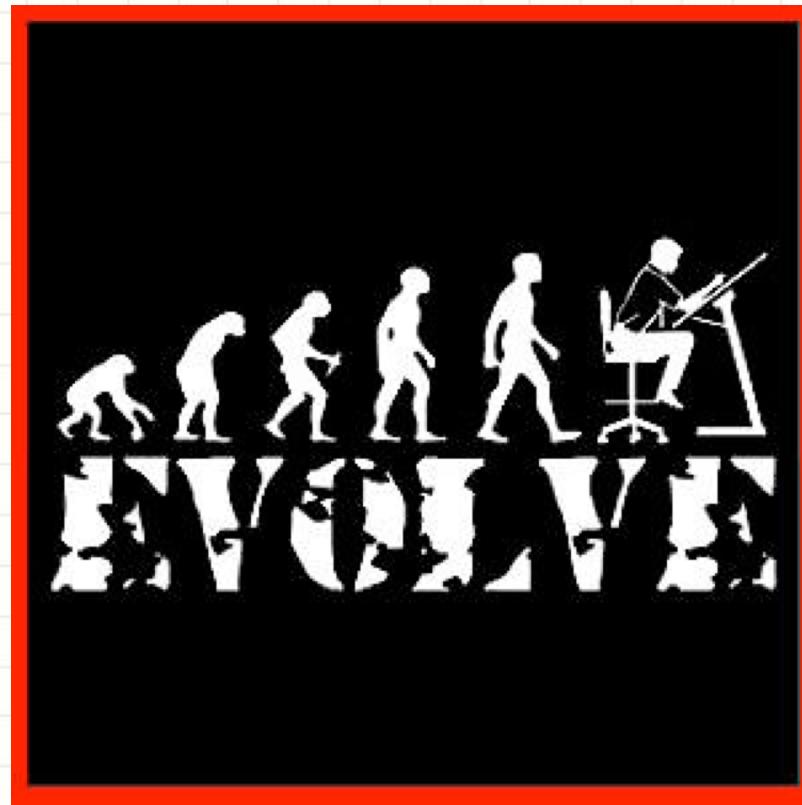


Architecture's life

Can you define the architecture at the beginning of the project and be done with it?

Yes? congratulations!





Why?

- Legal
- Finance
- Competition
- Technology aging
- Maintainability
- “Fix the mess” (aka pay now or pay later)

And of course...

- Evolving requirements

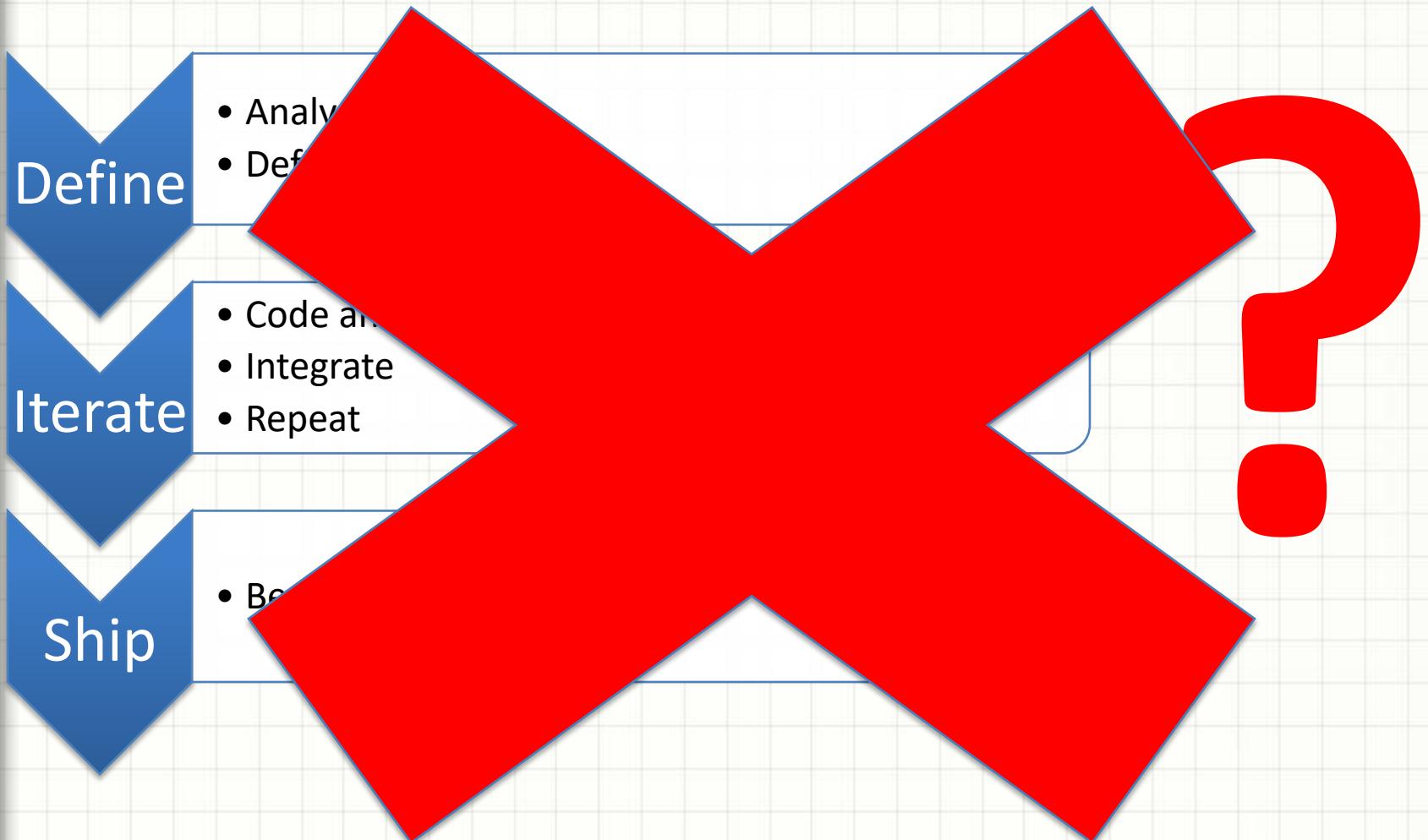
But why ???

- Because you have a **problem**....
 - Example: VTT move from DB2 to MongoDB
- Notion of architectural smell

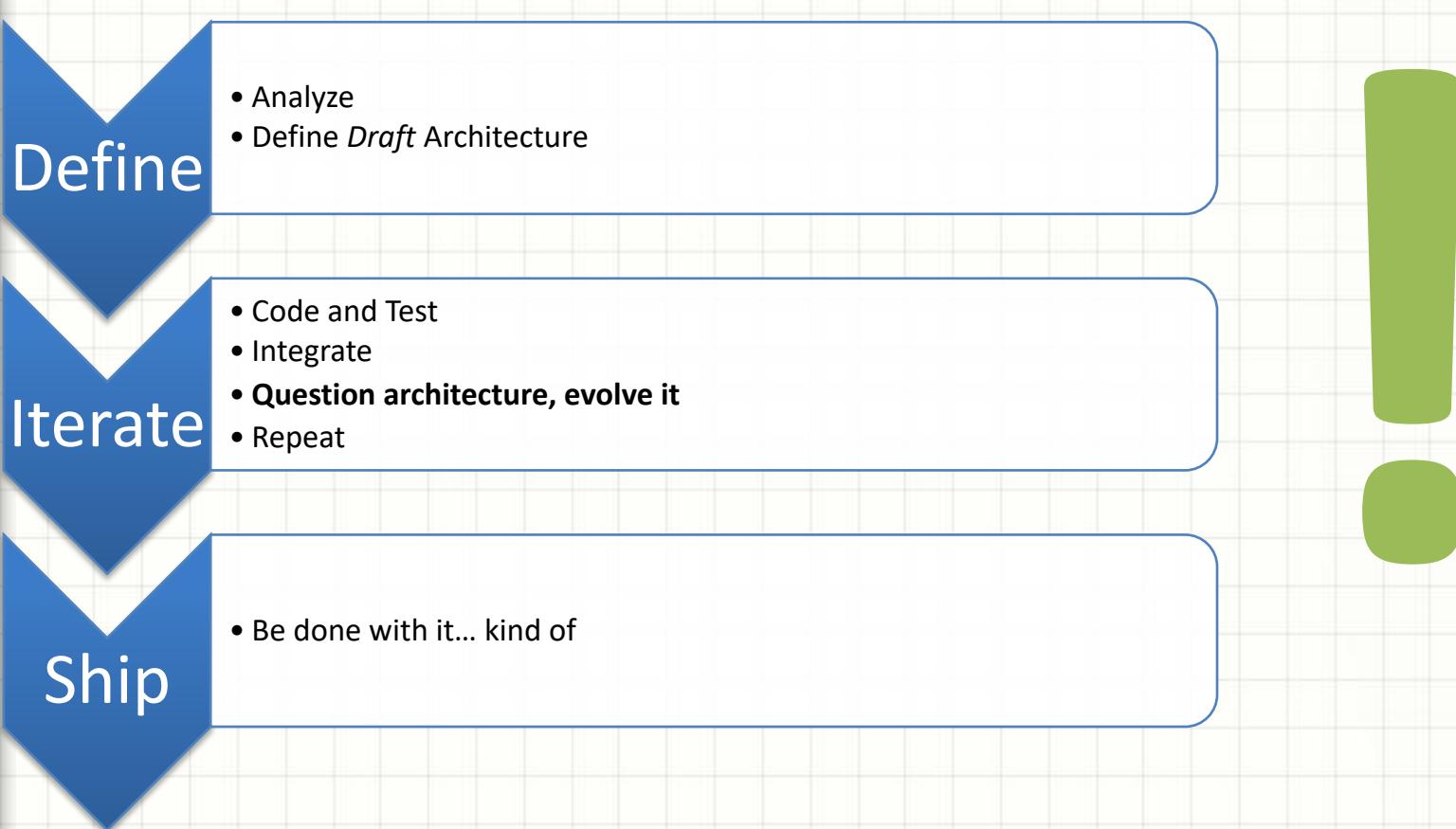
How

- No scientific method
- Avoid big-bang
- Validate each change
- → it has a cost!
- But keeping defects around also has a cost
 - Bad news doesn't age well

Iterative development



Iterative development



Techniques

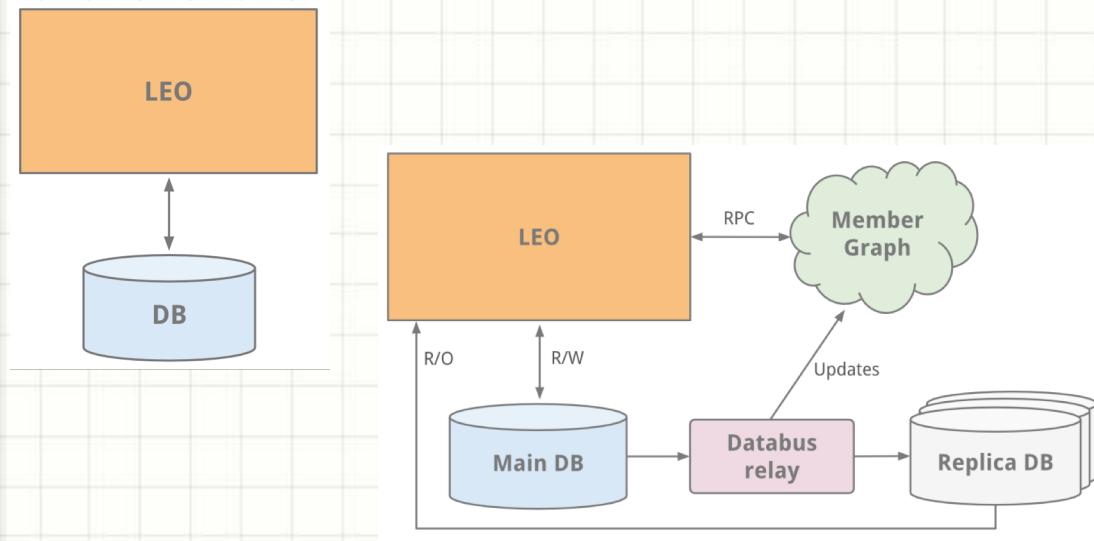
- Deprecation
- API resiliency over time
- Façade
- Rolling replace (or rolling upgrade)

EVOLUTION IN SPACE



Scaling

- Real life example: Linked In story



SOA

<https://engineering.linkedin.com/architecture/brief-history-scaling-linkedin>

Execution platform

- Desktop to embedded in device



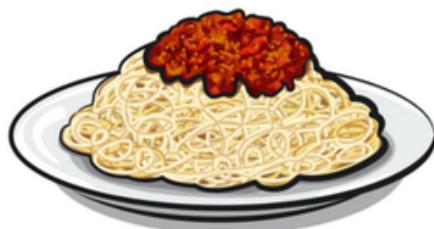
- Server to Cloud



SOFTWARE ARCHITECTURE

1990's

SPAGHETTI-ORIENTED
ARCHITECTURE
(aka Copy & Paste)



2000's

LASAGNA-ORIENTED
ARCHITECTURE
(aka Layered Monolith)



2010's

RAVIOLI-ORIENTED
ARCHITECTURE
(aka Microservices)



WHAT'S NEXT?

PROBABLY PIZZA-ORIENTED ARCHITECTURE



QUESTIONS?

Projects – deliverable #1

- Page numbers please...
- Scope: ok
- Coarse architecture: ok
- Technology choices: mostly ok
 - Often lacks reasoning
- Now is time to code!