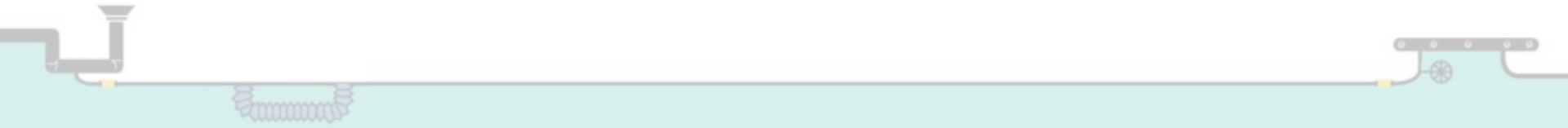# Technical Debt
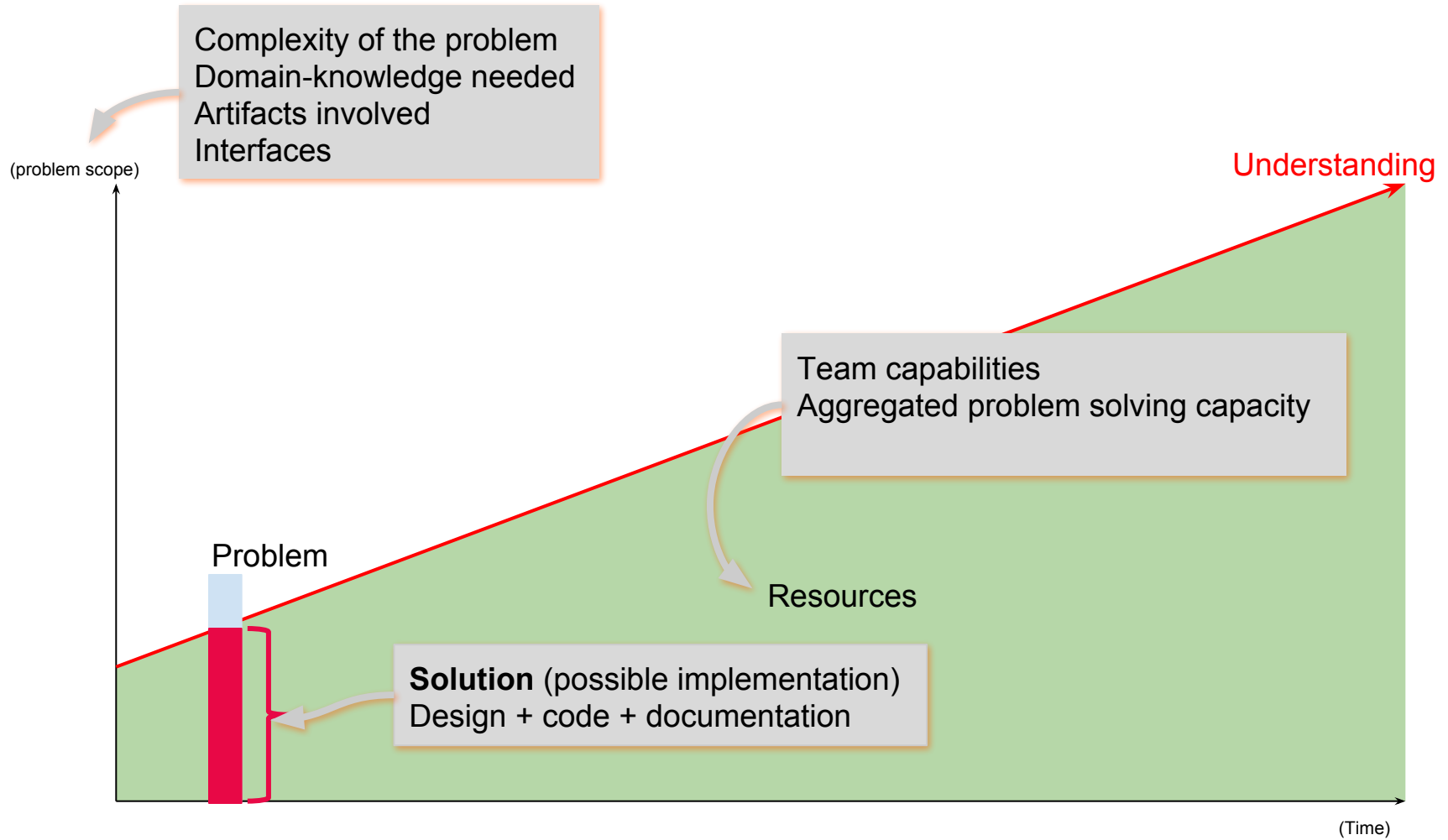
for dummies

The "Debt" Metaphor Corrupted

- *"A little debt speeds development so long as it is paid back promptly with a rewrite."*
  Term first introduced in 1992, by Ward Cunningham
  (OOPSLA '92 - Experience Report)

- A generally misused metaphor

- Another analogy: shovelling forward

*We acquire **debt** consciously knowing that it generates an **interest** that must be paid.*

Problem solving model

(problem scope)

Complexity of the problem
Domain-knowledge needed
Artifacts involved
Interfaces

Understanding

Team capabilities
Aggregated problem solving capacity

Problem

Resources

Solution (possible implementation)
Design + code + documentation

(Time)

Problem complexity over time all is good

(problem scope)

Understanding

Problem

Optimal solution

Problem

Resources

(Time)

Technical debt acquisition go get it!

(problem scope)

Understanding

Problem

Technical debt

Sub-optimal solution

(Time)
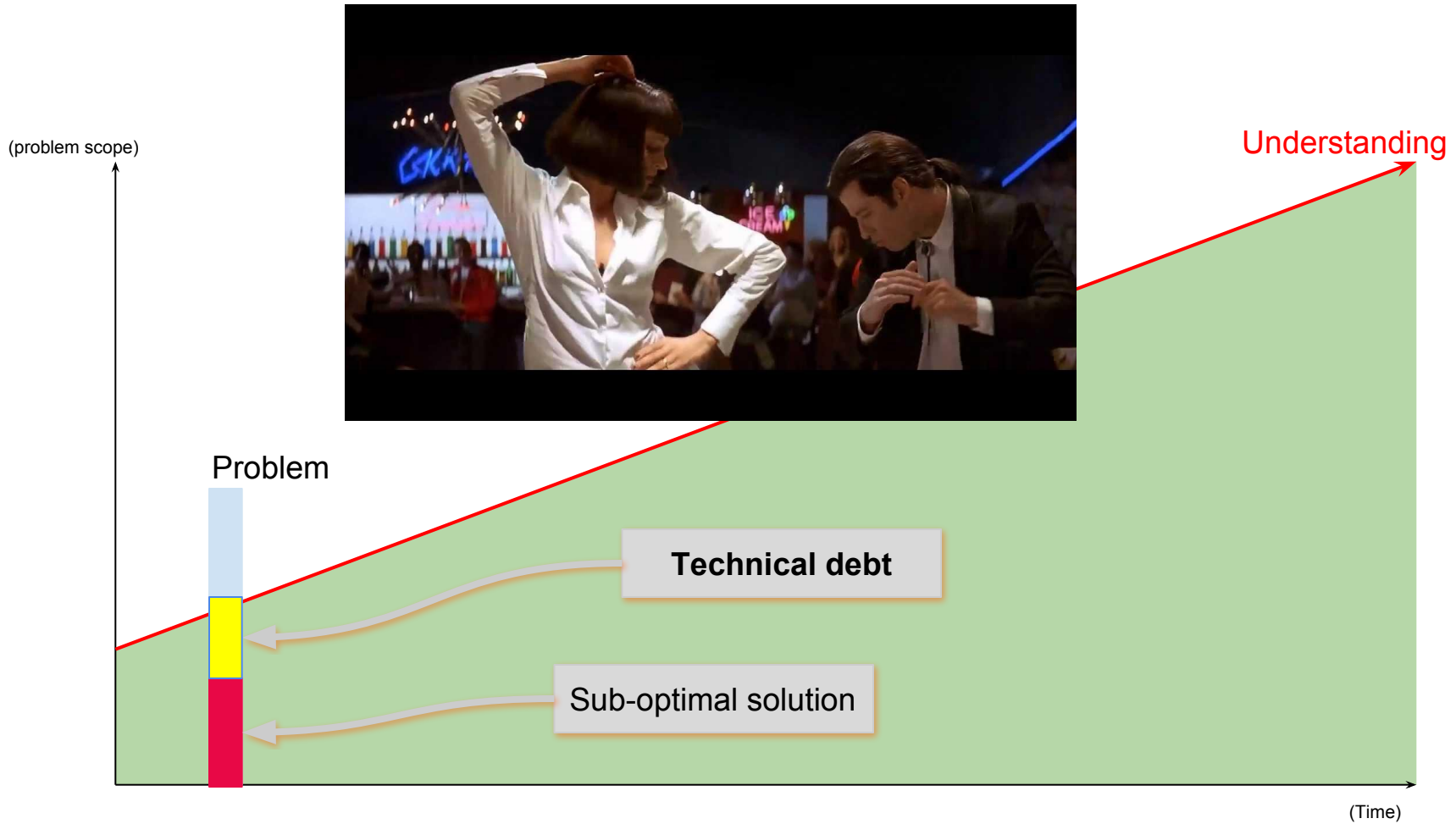
Technical debt growth out of control!
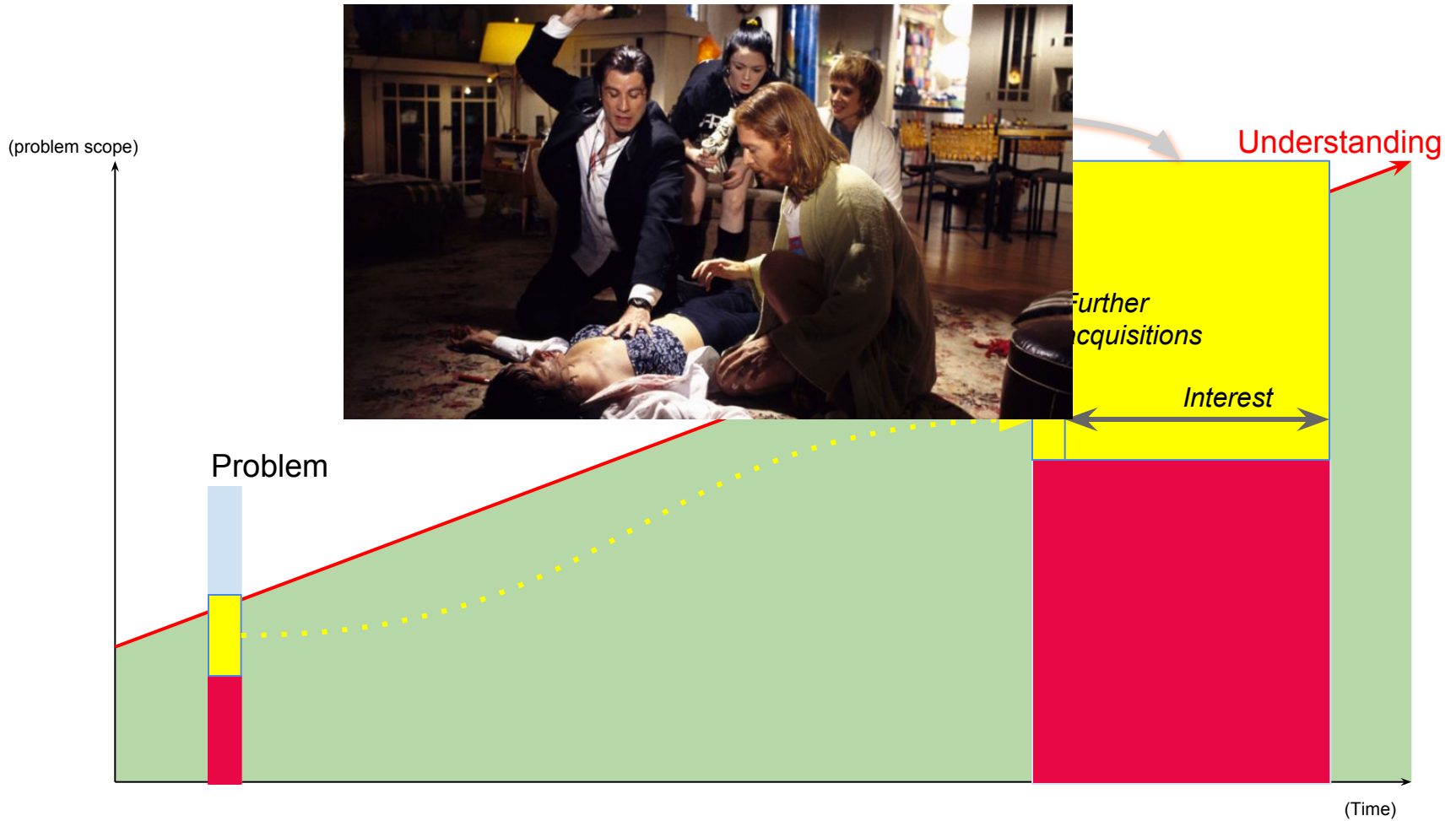
A tweet

Practicing Developer
@practicingdev

Seguir

It isn't technical debt if you aren't...

1) Leveraging it to get something valuable up front
2) Paying interest on it regularly
3) Treating it as a liability that you may eventually need to pay in full

Code that doesn't meet this criteria isn't debt, it's just low quality work.

🌐 Traducir Tweet

9:25 - 26 feb. 2018

Conclusion: management be a good boy

Technical and organizational implementation

Full-team **& *stakeholders*** responsibility

⬇

Must be managed

Tech-debt management pay your debts

- ## Limit by points
  (Assumes estimated issues) Don't allow accumulating more than X points of tech-debt (pay it!)

- ## Limit by count
  Don't allow accumulating more than X tech-debt issues

- ## Limit by age
  Don't allow tech-debt issues older than X days/sprints/releases

- ## Continuous payment
  (Assumes Scrum) Ensure that some tech-debt is paid on each Sprint

References

- http://c2.com/doc/oopsla92.html
- http://wiki.c2.com/?ComplexityAsDebt
- https://web.archive.org/web/20151111011323/http://ronjeffries.com/articles/015-11/tech-debt/
- https://martinfowler.com/bliki/TechnicalDebt.html
- http://www.ben-morris.com/technical-debt-is-an-overused-and-lazy-metaphor/

Thanks
¿Questions?

Carlos Matías de la Torre

cdelatorre@machinalis.com
@py_litox