

TIMEOUTS AND GRACEFUL DEGRADATION

TIAGO QUEIROZ



Onefootball

/ GitHub: belimawr

/ Twitter: @belimawr_

/ LinkedIn: <https://www.linkedin.com/in/queiroztiago/>

/ email: contato@tiago.eti.br

/ Example code: <https://github.com/belimawr/graceful-degradation>

What is Graceful Degradation?

"Graceful degradation is the ability of a computer, machine, electronic system or network to maintain limited functionality even when a large portion of it has been destroyed or rendered inoperative. The purpose of graceful degradation is to prevent catastrophic failure."

Amazon S3 outage: A guide to getting over cloud failures (<https://searchnetworking.techtarget.com/definition/graceful-degradation>)

The ability of maintaining functionality when portions of a system break down is referred to as graceful degradation."

Wikipedia: Fault tolerance (https://en.wikipedia.org/wiki/Fault_tolerance)

The problem

- / Given a list of team ids
- / Fetch name and country in a fixed set of languages
- / Return the list for each language
- / 404s must be ignored
- / Other errors must be informed/fail the request
- / Hard deadline/request timeout after a few seconds

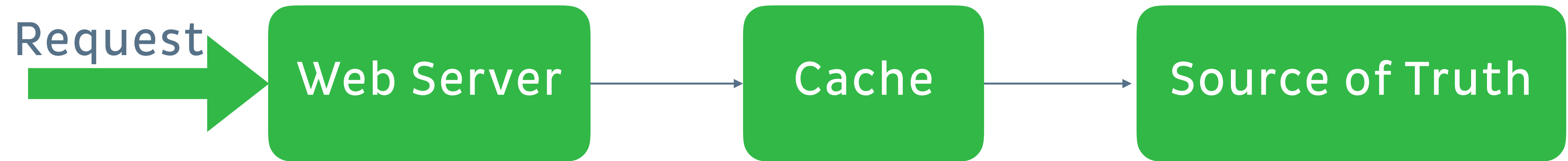
Naive

- / Parse the ids
- / Fetch one after another
- / Build and return the response

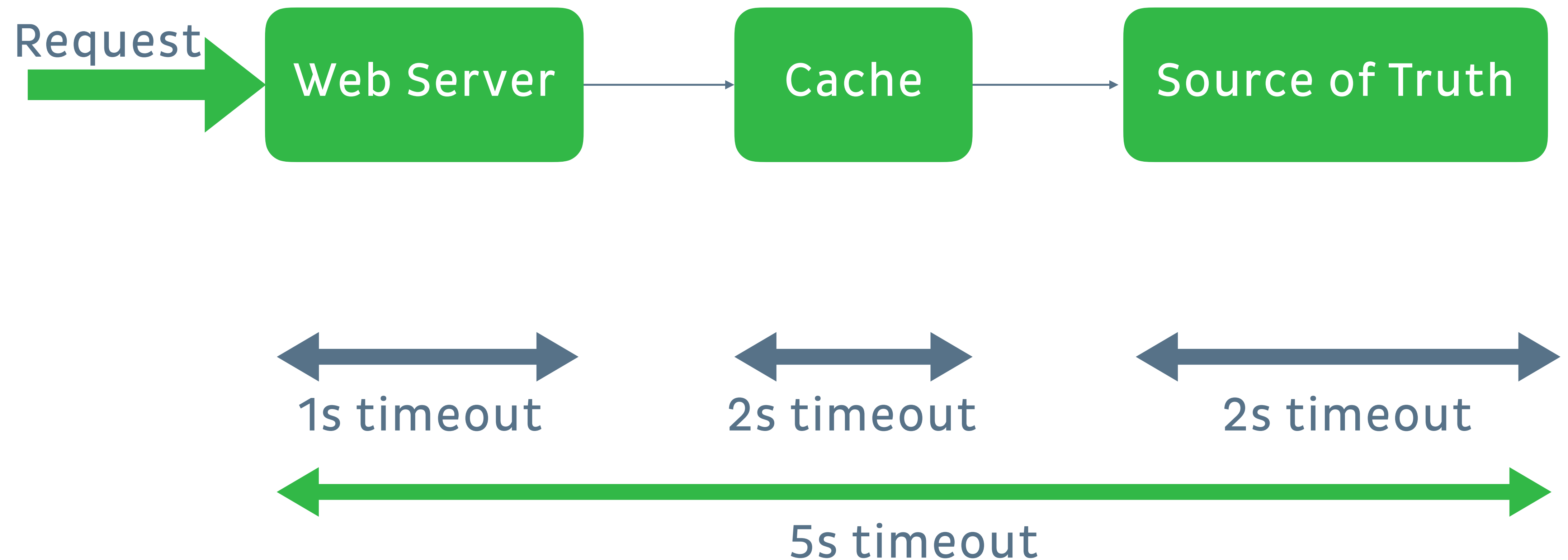
Cached

- / Parse the ids
- / Concurrently fetch from cache
 - If cache miss or error, go to source of truth
- / Concurrently fetch remaining data from source of truth
- / Save on cache

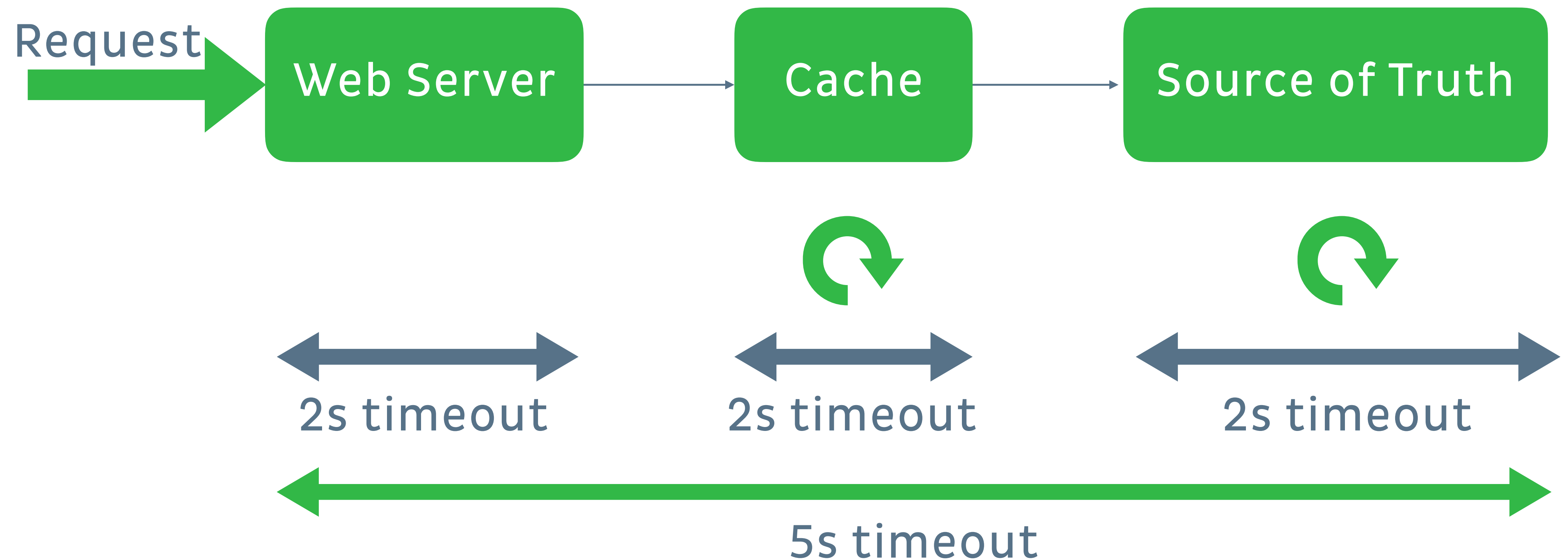
The problem



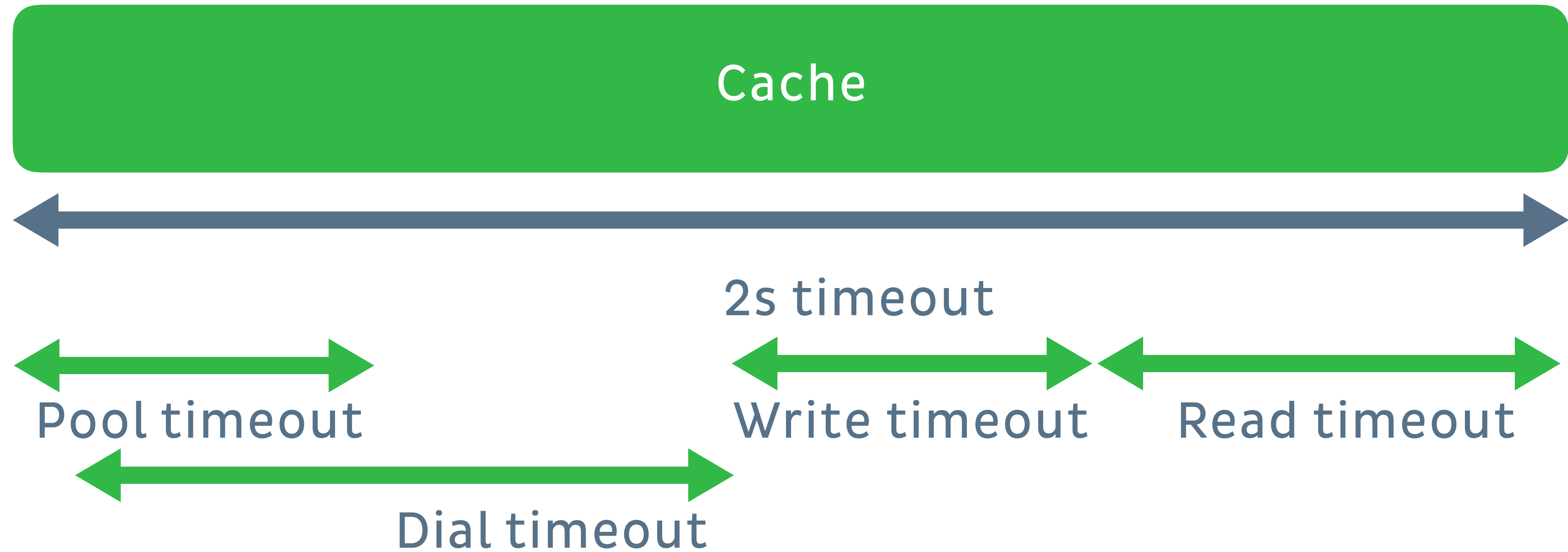
The problem



The problem



The problem



The problem



or

Graceful Degradation?

DEMO!