



# The Pantheon Stack



# In This Lesson

- ▶ What an Application Container is and why Pantheon uses them
- ▶ What tools you have in the containers
- ▶ How caching works on Pantheon

# Application Containers

- ▶ Container
  - Security, resource allocation
  - Not a Virtual Machine
- ▶ Assigned internal TCP port, IP
  - Requests delivered by Styx
- ▶ Every environment gets at least one container
- ▶ Styx balances requests across containers, retrying failed requests

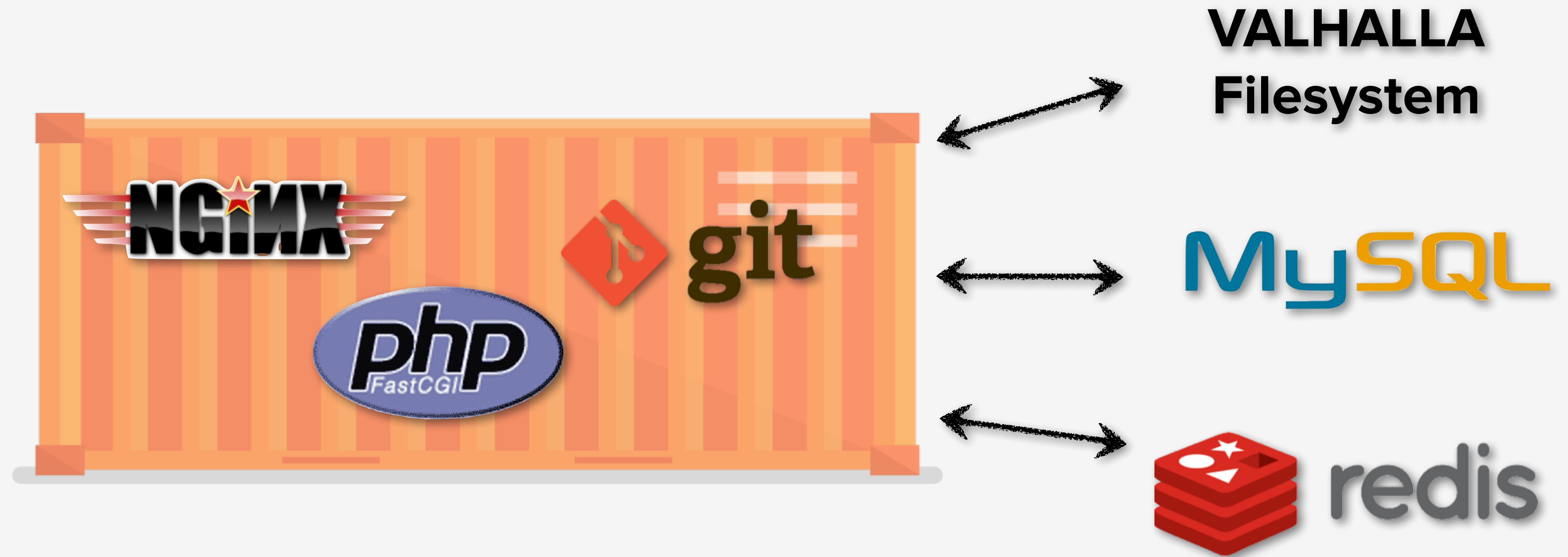
# Why Use Containers?

- ▶ Add, remove, redistribute in seconds
- ▶ Software scaling
- ▶ Provisioned automatically
- ▶ Configuration is always in sync
- ▶ Highly available
- ▶ Not limited like virtual machines

# What's In a Container



# What's In a Container

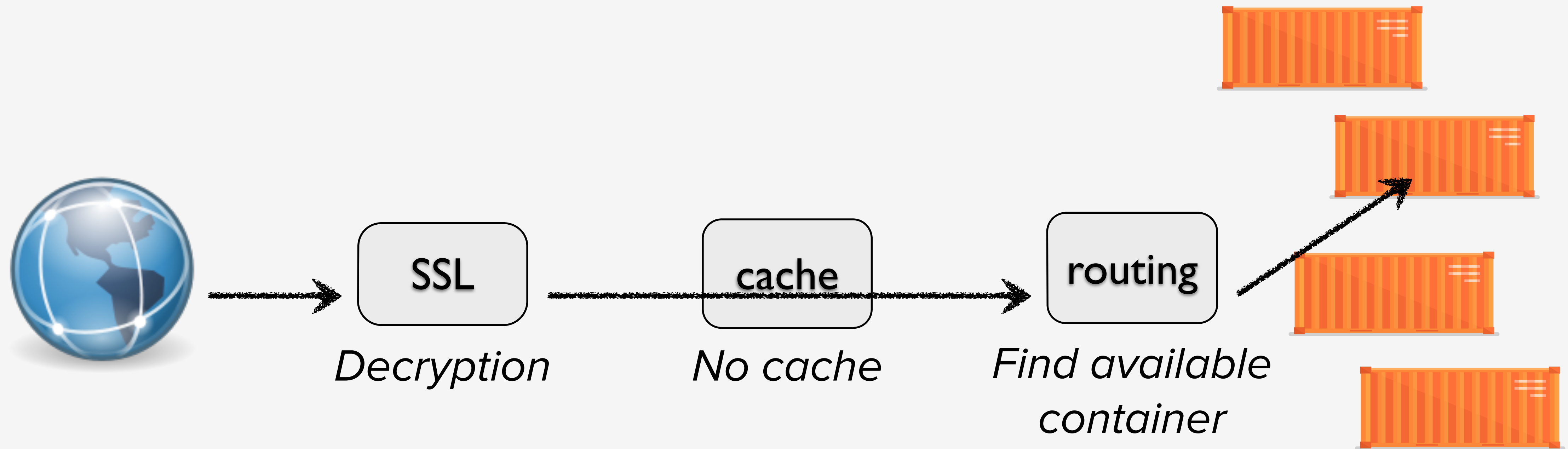




# Caching and Varnish

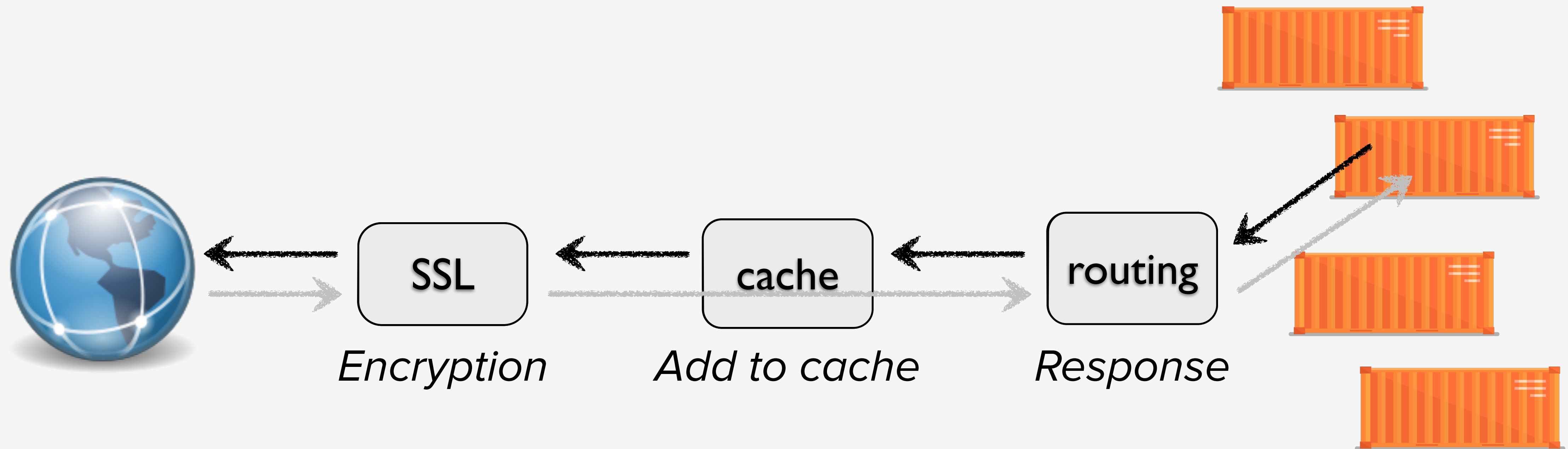
- ▶ Varnish: HTTP accelerator
- ▶ Serves static content, anonymous pages
- ▶ Every request goes through cache
- ▶ If uncached, served by container workers

# Uncached Request Lifecycle





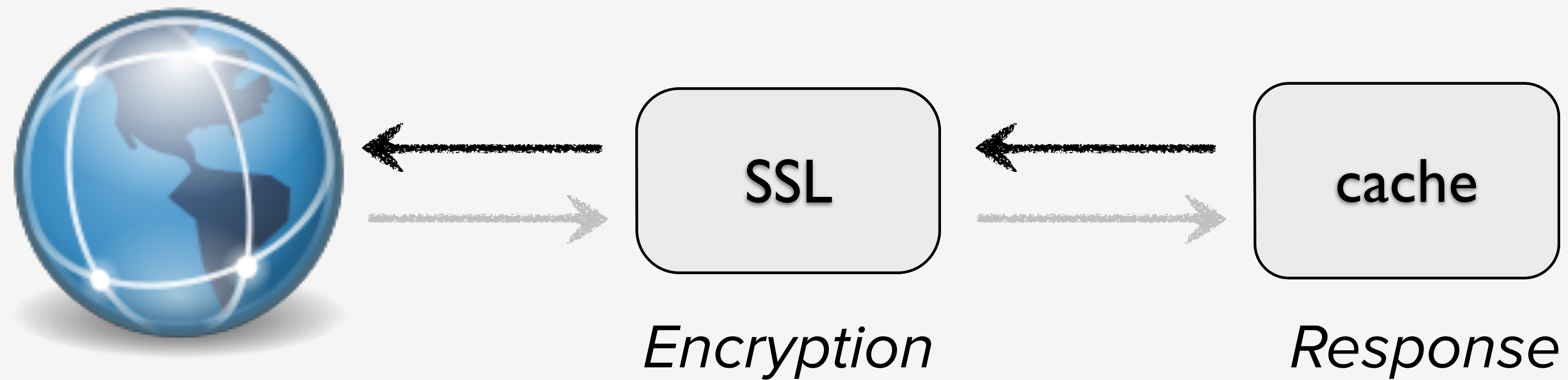
# Uncached Request Lifecycle



# Cached Request Lifecycle



# Cached Request Lifecycle



# Recap

- ▶ What an Application Container is and why Pantheon uses them
- ▶ What tools you have in the containers
- ▶ How caching works on Pantheon