



# **Collapsed Forwarding**

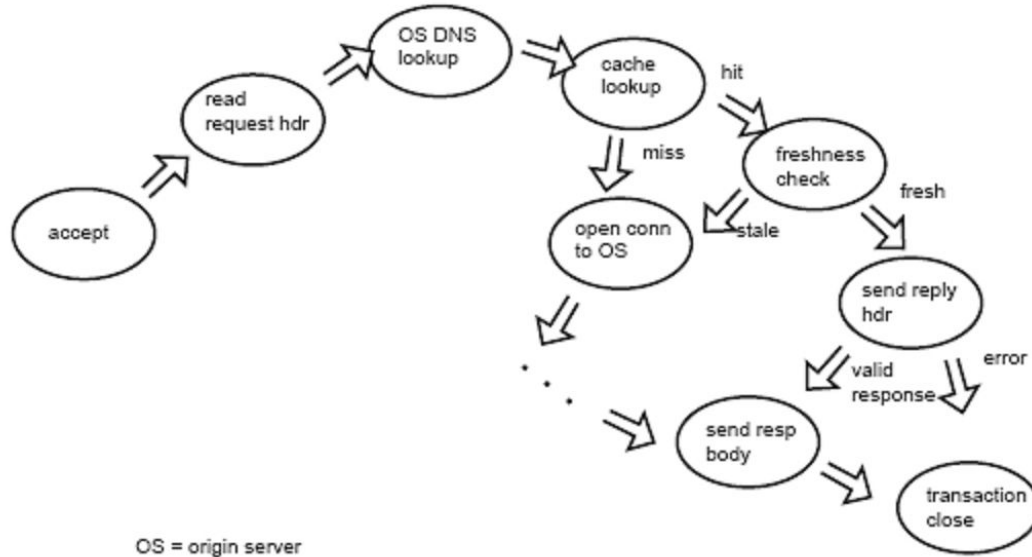
**Sudheer Vinukonda**

# Agenda

- ❖ Thundering Herd
- ❖ Current ATS Solutions
- ❖ New Proposal based on discussions with Leif

# ATS - HTTP State Machine

## Simplified HTTP Transaction



*Simplified HTTP Transaction*

# ATS - Cache State Machine

- ❖ Cache Lookup (lookup using the request URL as cache key).
  - Success, try to obtain a read lock, goto “Open Cache Read”
  - Else (cache miss), goto “Open Cache Write”
- ❖ Open Cache Read (try to obtain read lock)
  - Success, serve from cache, Done
  - Else goto “Open Cache Write”
- ❖ Open Cache Write (try to obtain write lock).
  - Write lock success, download the object into cache and to the client in parallel
  - Else, disable cache, and download to the client in a proxy-only mode.
- ❖ Done

# ATS - Thundering Herd Mitigations

- ❖ Read-While-Writer
  - Parallel concurrent streaming (background fill threshold, timeout - can be disabled (0))
  - Response headers need to be received to determine cacheability
    - No distinction between Cache refresh vs Cold Cache
  - Read-While-Writer-max-retries
  - Read-While-Writer-retry-delay
- ❖ Open Read Retry (triggered on open read failure)
  - max retries, retry delay
  - orthogonal to RWW, but, recommended to be configured together
- ❖ Open Write Retry (triggered on open write failure)
  - max retries, retry delay
- ❖ Open Write Fail Action (triggered on open write failure)
  - 5xx or stale cache copy
  - Adds an @-header for plugins to do custom treatment

# ATS - Cache State Machine

## ❖ Open Cache Read (try to obtain read lock)

- Success, serve from cache, Done
- Else “Open Cache Read Retry”
  - Success, serve from cache, Done
  - Else goto “Open Cache Write”

## ❖ Open Cache Write (try to obtain write lock).

- Write lock success, download the object into cache and to the client in parallel
- Else “Open Cache Write Retry”
  - Success, write/serve from cache
  - Else “Open Write Fail Action”
    - Return error/stale copy etc
    - disable cache, and download to the client in a proxy-only mode.

# ATS - Collapsed Forwarding

- ❖ Plugin solution to block concurrent requests
  - Solution built on leveraging open write fail action
  - Acts on the 5xx error responses with the @-header
  - 3xx redirect back to localhost after a small delay
    - Makes the client request circle back to cache states
  - delay, max retries
- ❖ Worked very well for Cache miss
  - Use cases : HLS live streaming, large video segment objects
- ❖ Cache Stale - Didn't quite work (bug?)
  - HLS Manifest refresh (smaller size, not quite as serious)
- ❖ Band-aid solution to a gap in the mitigations in the ATS core
- ❖ Inefficient, requires looping back request using 3xx

# Proposal

- ❖ Leif yelled at me and convinced that Open Write Fail Action isn't needed
- ❖ Open Read Retry handling in HttpCacheSM
  - Open Read Retry attempted only when Cache returns DOC\_BUSY
  - No read retry on a Cache Miss (not to impact a regular cache miss scenario)
- ❖ Solution : Jump back to attempt cache open read on cache open write fail
  - Built-in solution in the core filling the gaps in Thundering Herd mitigations
  - More efficient to schedule continuations and retry cache open than using 3xx redirect loop
  - Together with RWW, open read retry should plug the hole completely
    - Cache stale scenario should be addressed as well
- ❖ Wins
  - Deprecate the collapsed\_forwarding plugin (Yay!)
  - Stale-While-Revalidate (?) in the core (arguably better than RFC, fresh objects downloaded)



Thank you!

