



adaptTo()

EUROPE'S LEADING AEM DEVELOPER CONFERENCE
28th – 30th SEPTEMBER 2026

AEM Content Requests – Reduce, Reduce, Reduce!
Damian Langsweirdt, Adobe

Agenda

- What are content requests?
- Why are my content requests high?
- How do I know what's happening?
- How do I protect my content request budget?

What are content requests?

What are content requests?

- License metric for AEM Sites
 - Page views and API calls measured at CDN
 - Exclusions for certain paths, user agents, ...
- Intent is Usage → Value

The name of the game is to mind the budget!

Why are my content requests high?

The Six Stages of Content Request Grief

- Shock → *OMG that's a lot!*
- Denial → *Can't be right!*
- Anger → *&#^\$%^#!!!*
- Bargaining → *How to fix?*
- Reduce → *Yeah, feels good*
- Acceptance → *Better now*



Common Culprits - Categories

- Internal
 - Doing it to yourself!
- External
 - Somebody doing it to you!
- Preloading / Prefetching

Common Culprits - Internal

- Monitoring
- 404 as 200, or 404 as 301 to 200
- Security or Penetration Tests
- Anti-phishing email scans (check signatures!)
- Custom code or processes

First try to handle at the source, not the CDN

Common Culprits - External

- Crawlers
- Disguised Bots
- Bad Actors / Attacks
- Many types of bots...



Block at CDN using Traffic Filter Rules

Common Culprits – Preloading / Prefetching

- Google
 - Google on Chrome loads highly ranked pages before a user clicks
 - Virtuous cycle of low LCP improving search rank
- TikTok
 - Paid ads will preload a page before a user acts
- Meta (Facebook, Instagram)

How do I know what's happening?

Monitoring

- Log forwarding
- Log add'l properties using request transformations
- CDN Log Analysis Tooling (ELK dashboard)
- Cloud Manager
- Account team / Support ticket

Stay vigilant as things can change over time

Request Transformations - sample syntax

```
requestTransformations:
```

```
rules:
```

```
- name: log-on-request
```

```
  when: "*"
  actions:
```

```
    - type: set
```

```
      logProperty: req_cache_control
      value:
```

```
        reqHeader: cache-control
```

```
    - type: set
```

```
      logProperty: req_accept_language
      value:
```

```
        reqHeader: accept-language
```

```
    - type: set
```

```
      logProperty: req_ref
      value:
        reqHeader: referer
```

```
- type: set
```

```
  logProperty: bot_name
  value:
```

```
    reqProperty: botName
```

```
- type: set
```

```
  logProperty: client_as_name
  value:
```

```
    reqProperty: clientAsName
```

```
- type: set
```

```
  logProperty: client_ja4
  value:
```

```
    reqProperty: clientJa4
```

**Log
more
fields!**

How do I protect my content request budget?

Reduce, Reduce, Reduce!

- Focus on lowest hanging fruit
- Iterate
- Monitor over time



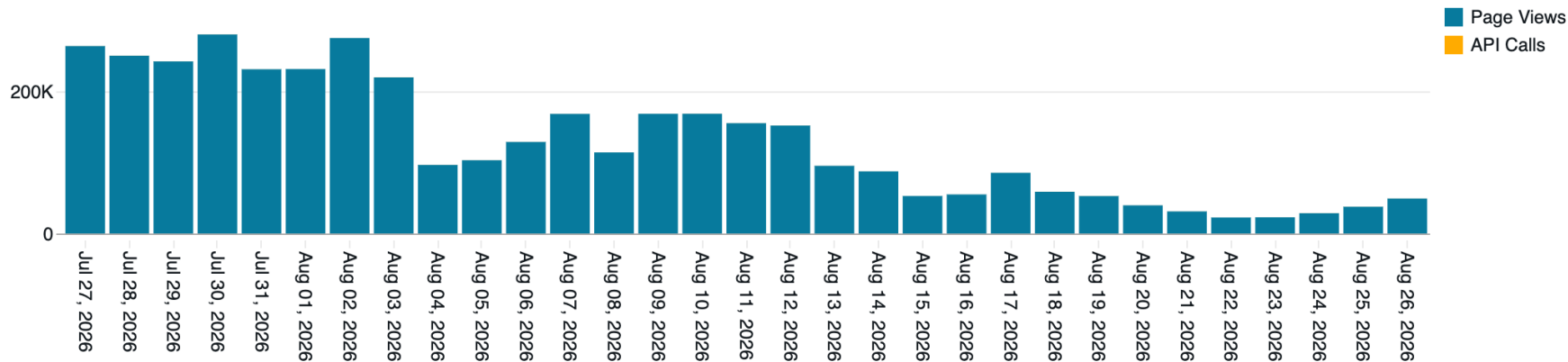
The fruit will get smaller with each iteration

Real results from a customer

- 80%+ reduction over 3 iterations

Daily Total

Breakdown of content requests by type (Page View or API Call) across the specified date range. API calls are weighted as 0.2 content requests in chart.



Make changes at the source

- Monitor using HEAD not GET requests
 - Or path /system/probes/health not homepage
- Respond with appropriate HTTP status
- Run security testing at STAGE not PROD
- Opt out of TikTok preloading

Traffic Filter Rules

- Block unwanted traffic if source is clear
 - By specific user agent, client IP, network, path, specific HTTP headers, or any combination
- Rate limit unwanted traffic when source is not as clear, or when it changes over time
 - groupBy allows you to be generic – any user agent, etc
- Do both – block and rate limit!

Traffic Filter Rules - sample syntax

```
trafficFilters:
  rules:
    - name: block-unwanted-traffic
      when:
        anyOf:
          - { reqHeader: user-agent, exists: false }
          - { reqHeader: user-agent, equals: "" }
          - allOf:
            - { reqProperty: path, equals: "/" }
            - { reqProperty: clientAsName, equals: "amazon.com inc." }
            - { reqHeader: cache-control, equals: "no-cache" }
      action: block
```

Appendix

References

- [Traffic Filter Rules](#)
- [Private Prefetch Proxy in Chrome](#)
- [Optimize your site for preloading content \[TikTok\]](#)
- [How to disable Facebook \(Meta\) link prefetch](#)
- [Understand Cloud Service Content Requests](#)
- [CDN Log Analysis Tooling \(ELK\)](#)