

Anyone Can Performance Test?!?

Hello There!

@3r1c p/ericp@freeradical.zone

- Staff Test Infrastructure Engineer II,
Credit Karma, Oakland, CA
- Workshop on Performance and Reliability

Formerly...

- President, AST
- Senior Director, Product Quality
- Load Test Tools Product Owner
- Performance Consulting Manager
- Performance Team Lead



LFG!!!

1

WE'RE GONNA PERFORMANCE TEST!

1. Choose your handiest Internet-accessible device
2. Consider your Internet connection - The WiFi? Cellular?
3. Prepare a timing mechanism - analog/electronic
4. Open the Browser of your choice
5. I will give you a URL in a moment
6. Time how long it takes to load the site and record it
7. Please add your data to this sheet:

<https://tinyurl.com/cast22any1>



performance-workshop.org

Performance: Tacit Requirements

2

SOME REQUIREMENTS ARE TACIT

NFRs are Non-Functional Requirements. They need help!

- Security: “Meets Anti-Gremlin Standard 2.3”

Which includes?

- Reliability: “99.9% Uptime”

24x7x365? Business Hours?

Which functionalities failing count as Downtime?

- Meets Regulatory Data Governance Standards

In which countries? For which data?

Based on whose interpretation?

HOW FAST: TACIT REQUIREMENT

Does the app *feel* fast?

- How long do Users wait?
- How long do they expect to wait?
- How long have they waited before?
- What do they see while they are waiting?
- Once Users think software is Slow...
- ...but Users know Slow when they see it

EXPLICIT REQUIREMENT?

“Make Sure it’s (Performant/Responsive/Fast Enough)!”

- How fast is fast enough?
- “Whatever Industry Standard is”
 - If you’re Amazon, under 1 second
 - E-Commerce under 2 seconds
 - You have plenty of support to quote “3 seconds”
 - 5 seconds might be an eyeroll. 10 seconds is a sigh
 - 15 seconds is probably a bye!
 - Not entirely in our control (Last Mile, Device)

HOW MANY: TACIT REQUIREMENT

Will the App/Site Scale?

- If it gets really busy, will it slow down? Or crash?
- Can I handle going viral?
- Can I handle 3x organic growth?
- Can I rely on this Cloud/Container stuff?

TACIT REQUIREMENTS - THE REAL ONES

- Will it work? (*Will we/I succeed?*)
- How can I know? (*Help me find/devise an Oracle*)
- What is due diligence? (*What is a reasonable amount of time/money/risk tradeoff to invest?*)
- Who can help? (*Do we have/have access to the knowledge/focus we need to investigate?*)
- Will it keep working? (*Is it Reliable Enough?*)

HARD TRUTHS ABOUT PERF (AND LIFE)

Much of this is out of your control

- Can't control which devices connect
- Can't control Last Mile network conditions
- Organic Load is Clumpy/Variable; it's not overall load that gets you, it's the arrival rate
- Cloud/DNS go down sometimes
- Some bugs get out

**YKTV,
LFG!!!**

3

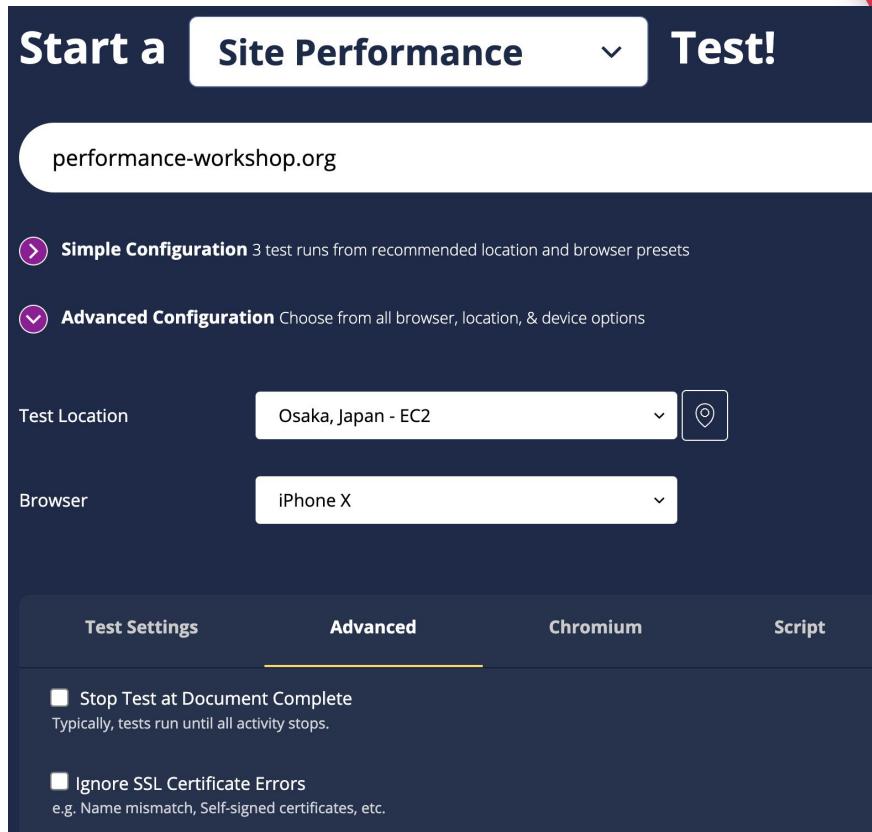


TESTING FROM GLOBAL LOCATIONS

webpagetest.org

- Click Advanced
- Pick a Location
- Pick a Browser

More settings on next slide...



CLIENT SETTINGS

- Connection
- Browser Dimensions
- Set These:
 - 3 Tests
 - First and Repeat
- Skip others for now

Test Settings	Advanced	Chromium	Script
Connection		3G Fast (1.6 Mbps/768 Kbps 150ms RTT)	
Desktop Browser Dimensions		default (1366x768)	
Number of Tests to Run		3	
Repeat View		<input checked="" type="radio"/> First View and Repeat View	
<input checked="" type="checkbox"/> Capture Video			

COOKING SHOW: WHILE WE WAIT

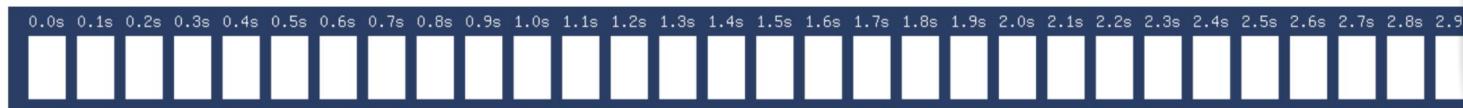
Observed Metrics (Based on Median Run by: ▶ Speed Index)

 **Note:** Metrics offered will vary.

FIRST VIEW (RUN 2)

First Byte	Start Render	FCP	Speed Index	LCP	CLS	TBT	Total Bytes
2.416 s	3.600 s	3.603 s	4.224 s	5.109 s	.093	.000 s	508 KB

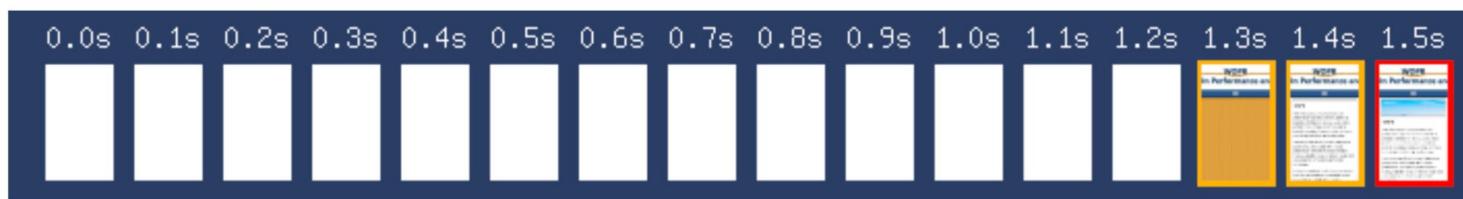
Visual Page Loading Process (Explore)



REPEAT VIEW (RUN 2)

First Byte	Start Render	FCP	Speed Index	LCP	CLS	TBT	Total Bytes
1.089 s	1.300 s	1.316 s	1.395 s	1.407 s	.093	.000 s	13 KB

Visual Page Loading Process (Explore)



LET'S ADD TO OUR DATASET

1. We'll use LCP (Last Contentful Paint) as our measure
2. Please add your data to the second sheet here:

<https://tinyurl.com/cast22any1>

A (Quick!) History of Performance and Load Testing

4

FORGIVE SOME GENERALIZATIONS...

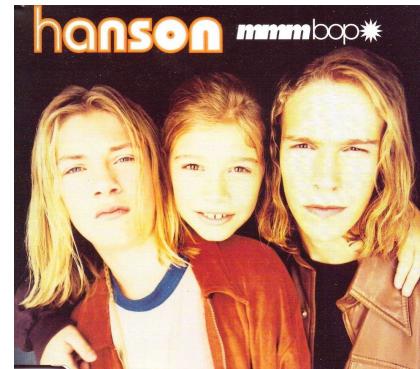
It's always dangerous to speak to some hypothetical mean

- No one can inspect all the contexts
- There are no two contexts that are identical
- There are contexts similar enough to group
- There are always exceptions/special cases
- Narrative requires some loss of resolution/granularity
- Narrators choose which context to include

199x: Client - Server

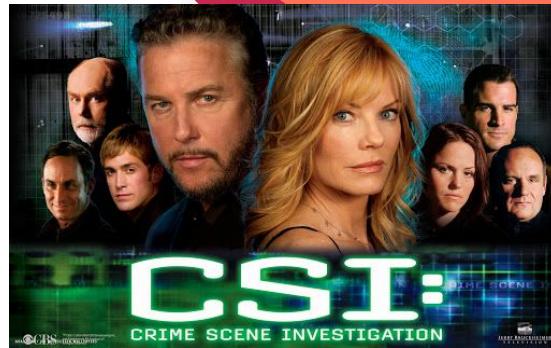
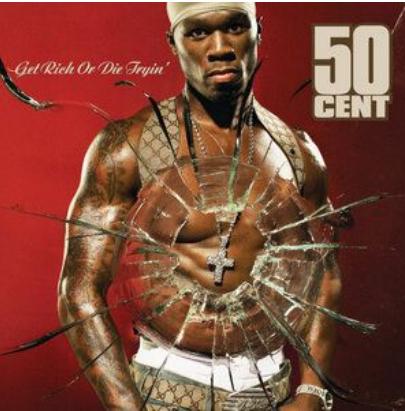


MERCURY INTERACTIVE



- Dozens/hundreds of connections from clients to multiple servers
- Physical Servers on LANs, Software installed on-prem
- Kilobits of home dialup bandwidth
- 32 bit processors, Pentium II MMX 266mhz, 66mhz bus
- Most response time is back end; Performance = Load Testing

2003: In-Browser Apps



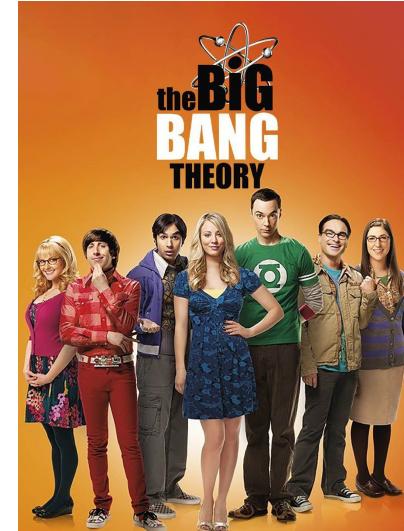
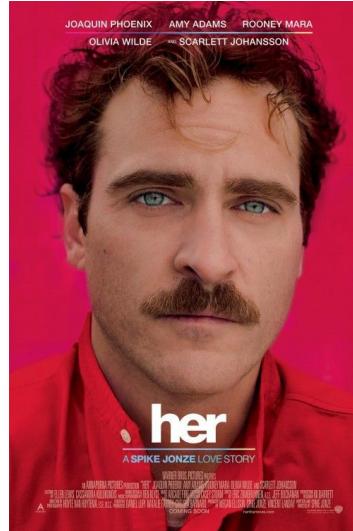
- Hundreds of browsers connecting to Web/App Server stacks
- Virtualized Servers on LANs
- 1-2 megabits of early “broadband” bandwidth
- First 64-bit processors: Athlon 3200 (2ghz), 400mhz bus
- Response time still back end; Performance = Load Testing, maybe supplemented with some Prod monitoring

2008: Browser Apps



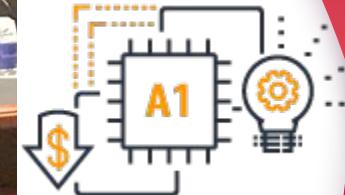
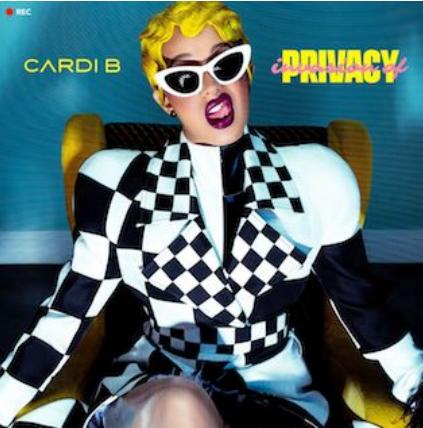
- Thousands of browsers connecting to SaaS apps
- Virtualized fleets, moving towards AWS, automated deployments
- Several megabits of bandwidth for many offices/homes
- Response time is getting fuzzier; Load Testing is missing more
- Iterative Development making inroads

2013: The Cloud



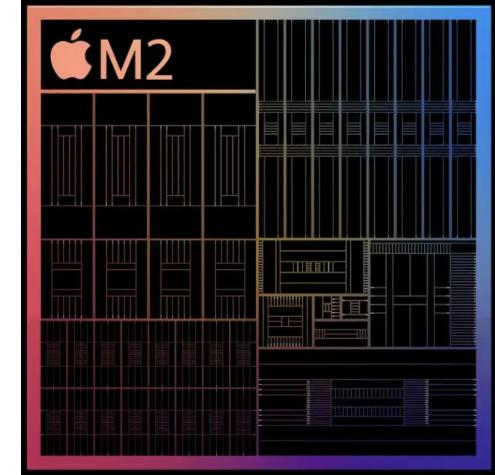
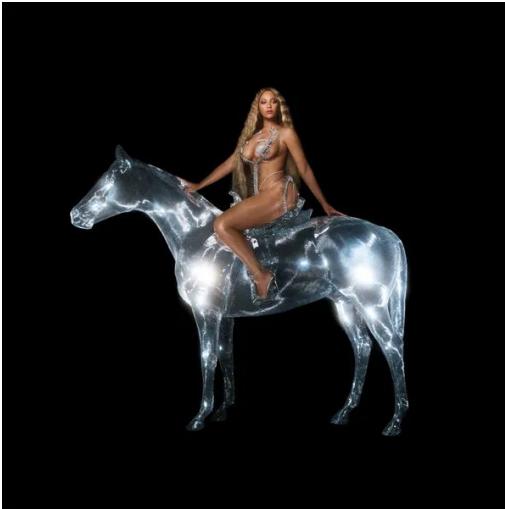
- Browsers connecting to multiple services, Native coming
- Cloud Architecture first
- Tens of megabits of bandwidth
- First 64 bit processors, Pentium II MMX 266mhz, 66mhz bus
- Response time is now mostly front end; APM is displacing tests
- Iterative development -> more frequent, incremental releases

2018: Cloud Native



- Hundreds of megabits of bandwidth to many US homes
- Competing clouds, “serverless” (kinda didn’t happen)
- AWS Graviton: Custom ARM Silicon. We’ll see this again
- Iterative is more or less a standard
- Better observability/traffic shaping - less pre-release testing

2022: “Phones” First



- In my context, desktop browsers are < 10% of traffic
- > 1 gigabits of fibre available, 5G in our veins
- Apple M2: Shared architecture at pocket and desk
- Response time is likely native + federated/stand alone
- A bit of a...Renaissance in Load Testing (as part of a toolbox)

WHAT HAVE WE LEARNED?

Most of what “Everybody Knows” about Load Testing is based on largely obsolete context. Now, we can:

1. Monitor Capacity and Observe Systems in Real Time
2. Increase Compute Resources in Minutes
3. Deploy multiple versions, change out/rollback code and use soft flags to change parameters in seconds
4. Ship Code Updates in...not a lot of time
5. Get obscene bandwidth on pocket supercomputers

WHAT HASN'T CHANGED (MUCH)

We Still:

1. Don't know what the future holds - but it will be cool!
2. Try to guess what we'll need, learn as much as we can
3. Seek to understand how our systems are being used - what's changed is how good our Observability is
4. Focus on Efficiency: one can't be faster than many

**LET'S DO A
LOAD TEST!!!**

5

WE'RE GONNA LOAD TEST!

1. Open your browser
2. Hit performance-workshop.org
3. Repeat - feel free to refresh before loading completes
4. Will 50+ requests/sec be too much for the site?
5. I'll help a little...

WHILE THAT'S HAPPENING...

A quick word about load testing:

1. JMeter is a fine load tool, with an active community. Not hearing a lot about commercial tooling these days
2. Doesn't have to be "accurate" in modeling to learn something from a test
3. Give it a spike and see what happens
4. There is a rich body of knowledge - and people who can help

Thanks!

Open Season?





Free templates for all your presentation needs



For PowerPoint and
Google Slides



100% free for personal
or commercial use



Ready to use,
professional and
customizable



Blow your audience
away with attractive
visuals