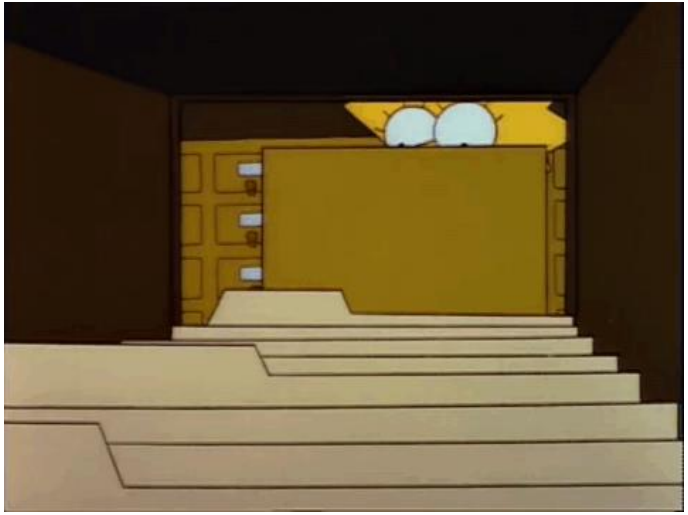
SLIs, SLAs, SL D'OHs

Learning About Service Uptime from Homer Simpson

Mason Egger - Developer Advocate, Digital Ocean
@masonegger
mason@digitalocean.com
twitch.tv/zelgiuscodes

Resources (for you and what I used)

- Google SRE Books https://landing.google.com/sre/
  - Most of this talk comes out of one of the Google SRE books and my personal experience
- Simpsons Meme Generator https://frinkiac.com/
  - Simpsons memes are fun and the world needs more of them



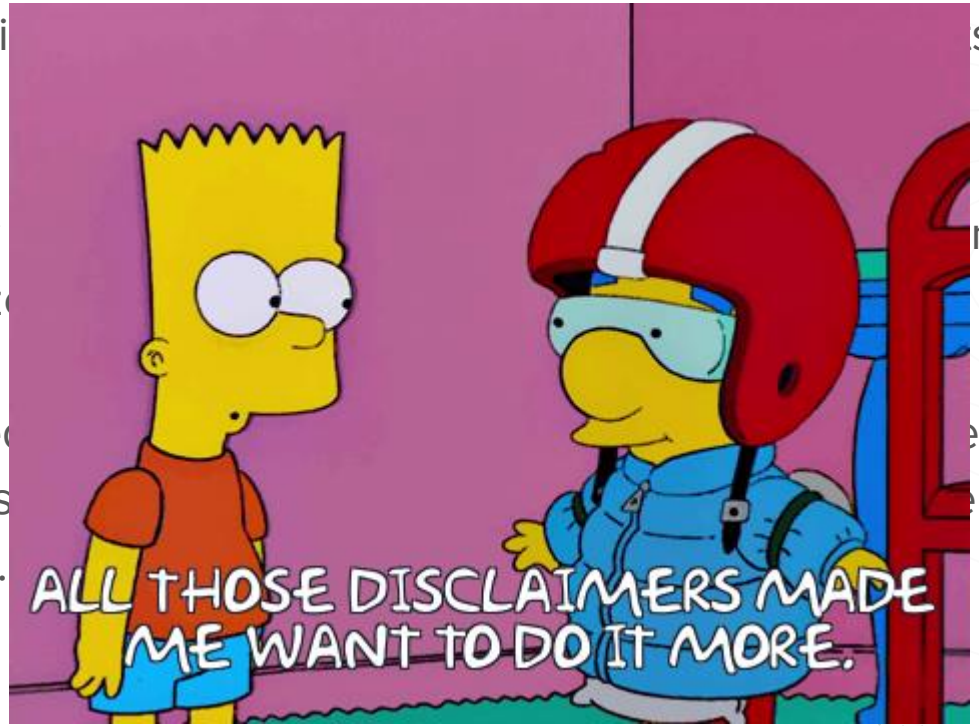
Disclaimer

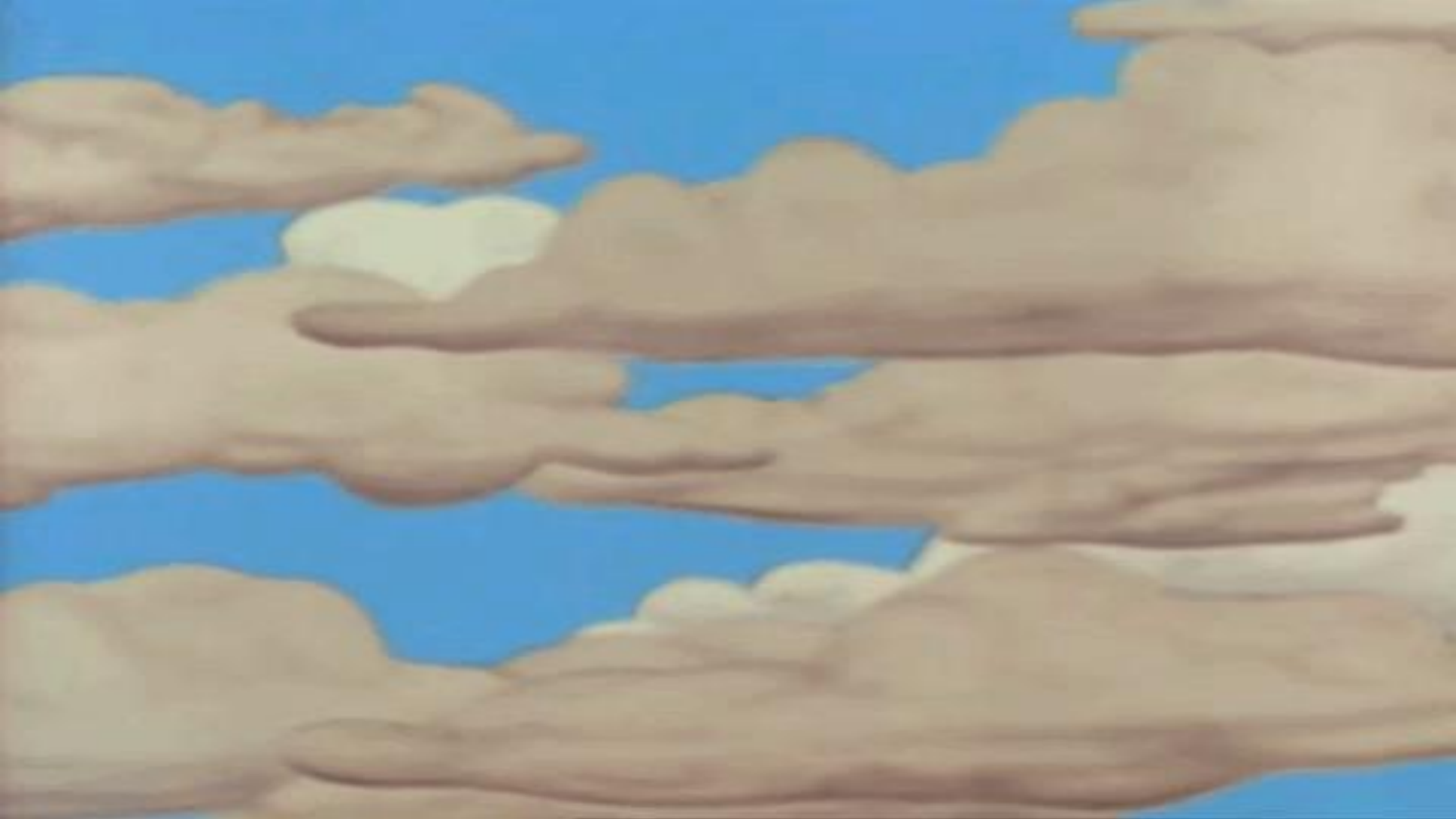
This talk is essenti s here and there (I hope).

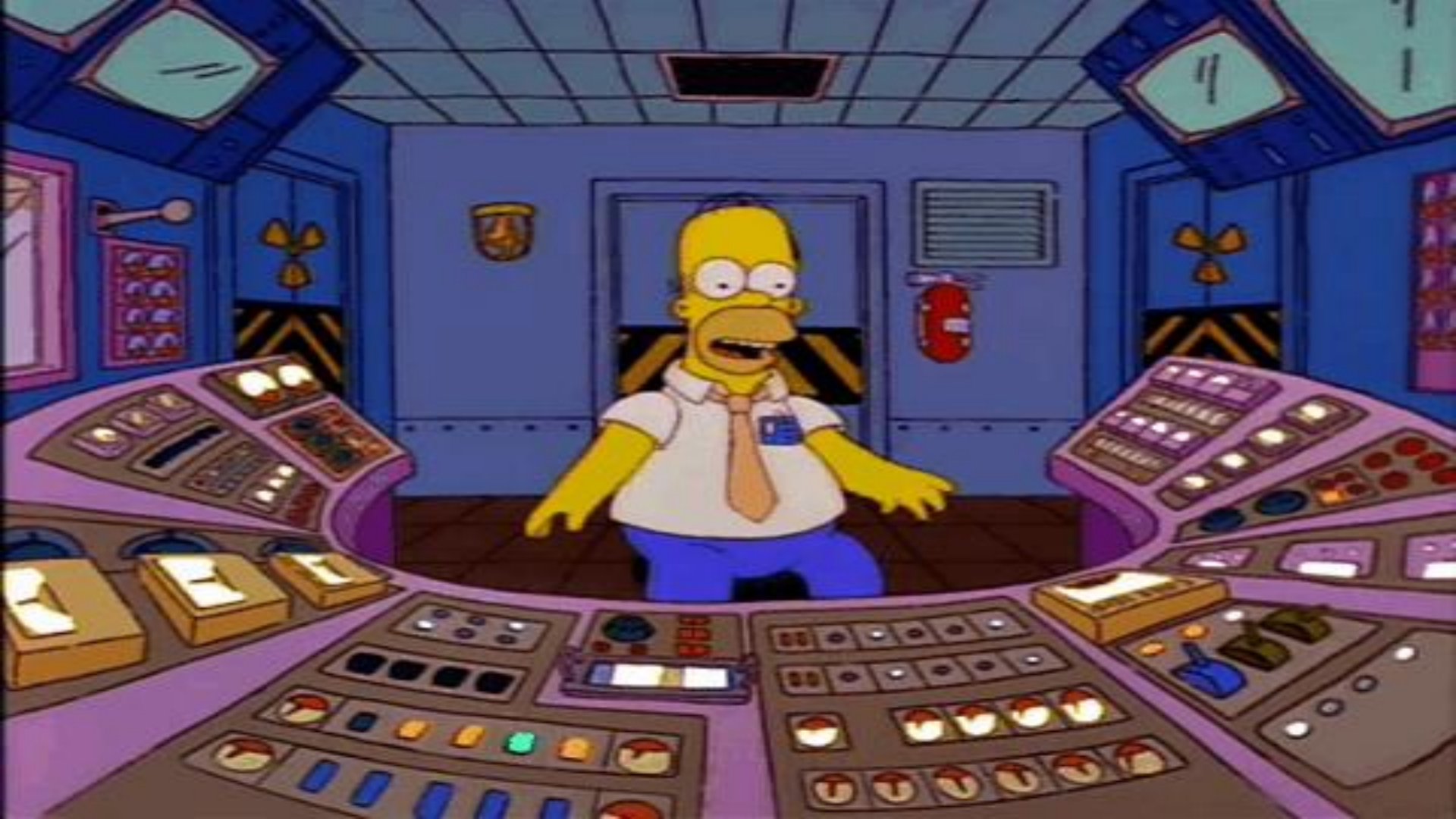
I was curious how ntation and get it past organiz

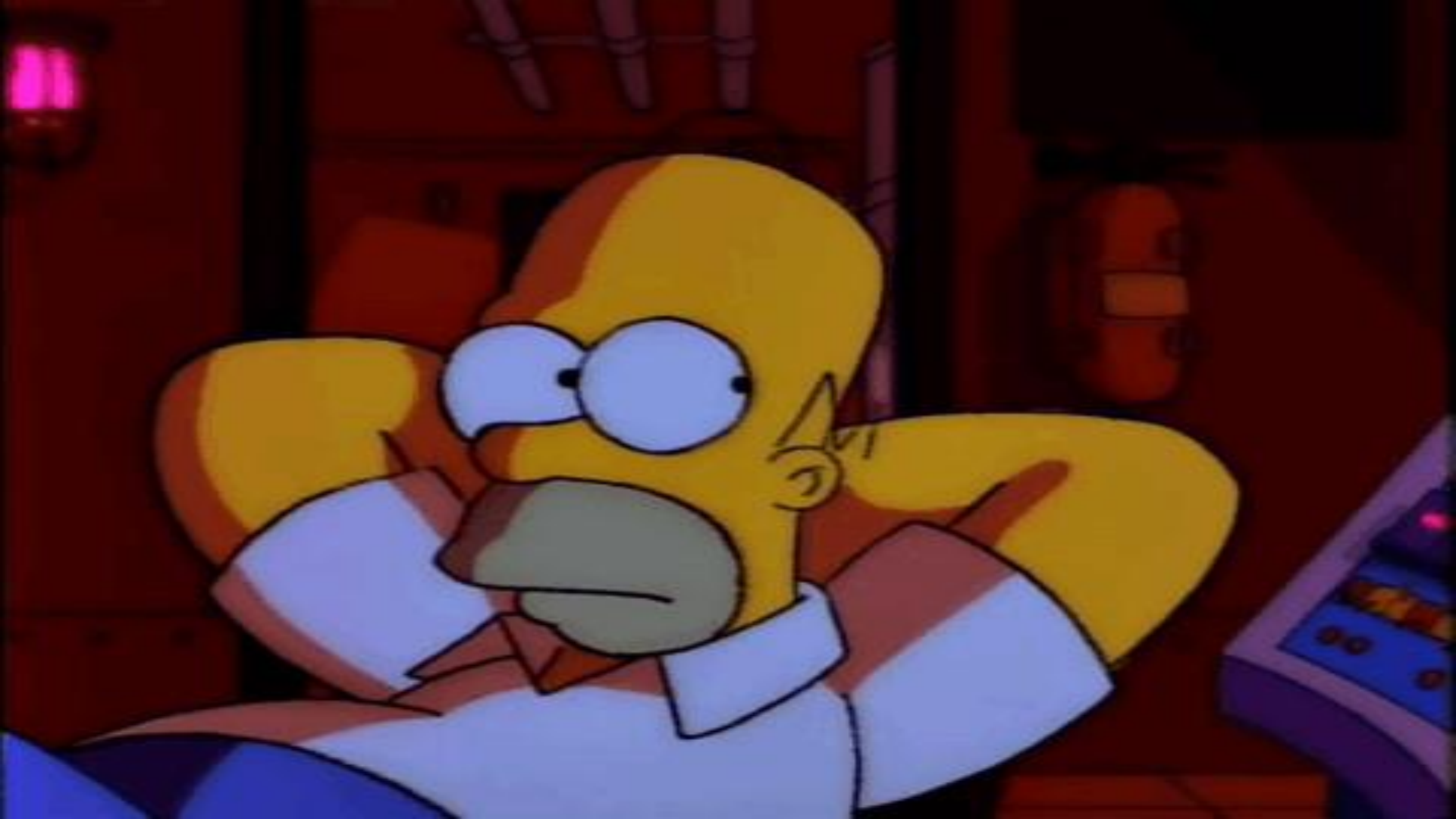
At a past job I use e clusters would have issues e job in Simpsons memes.

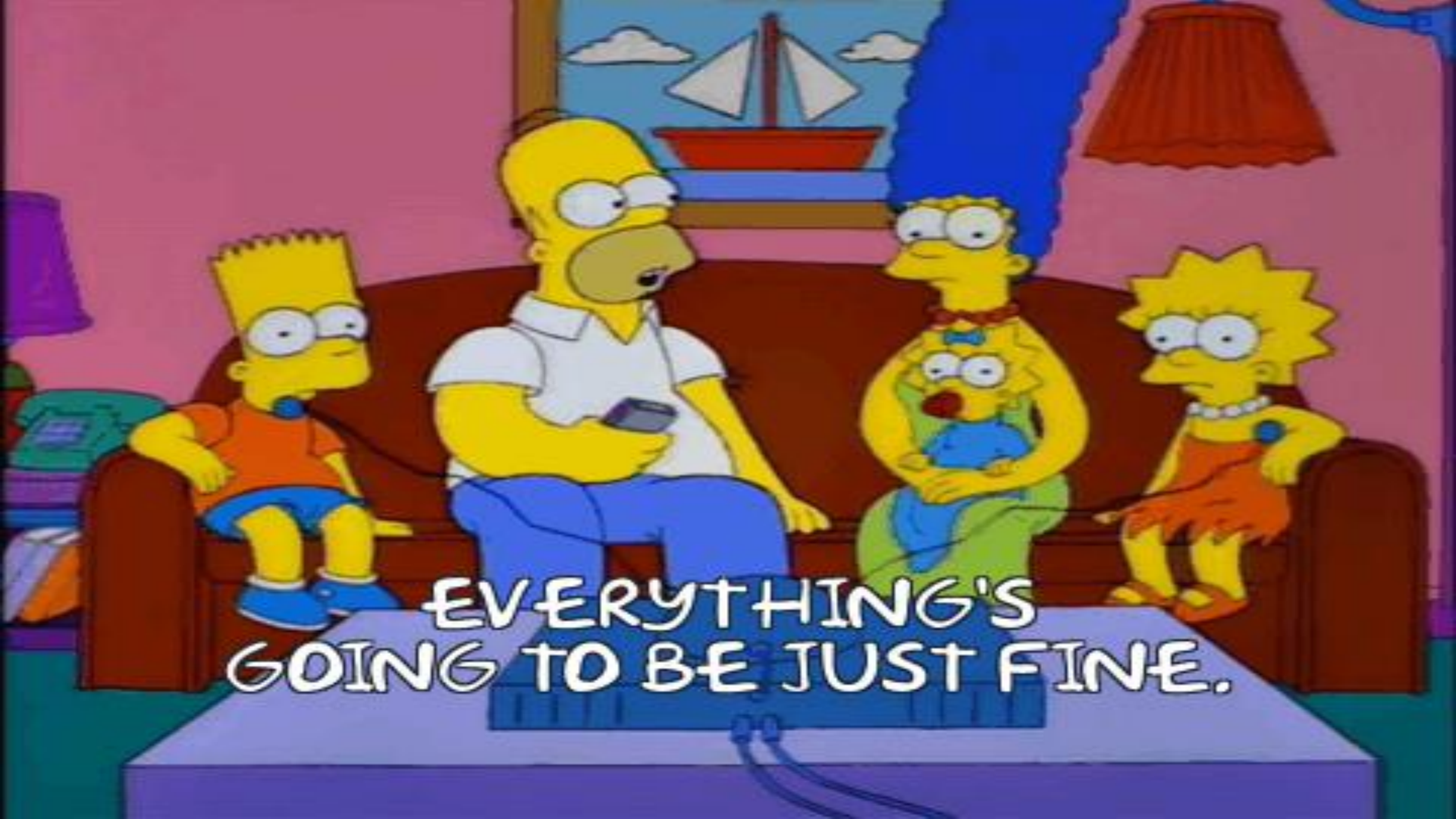
So here we go....

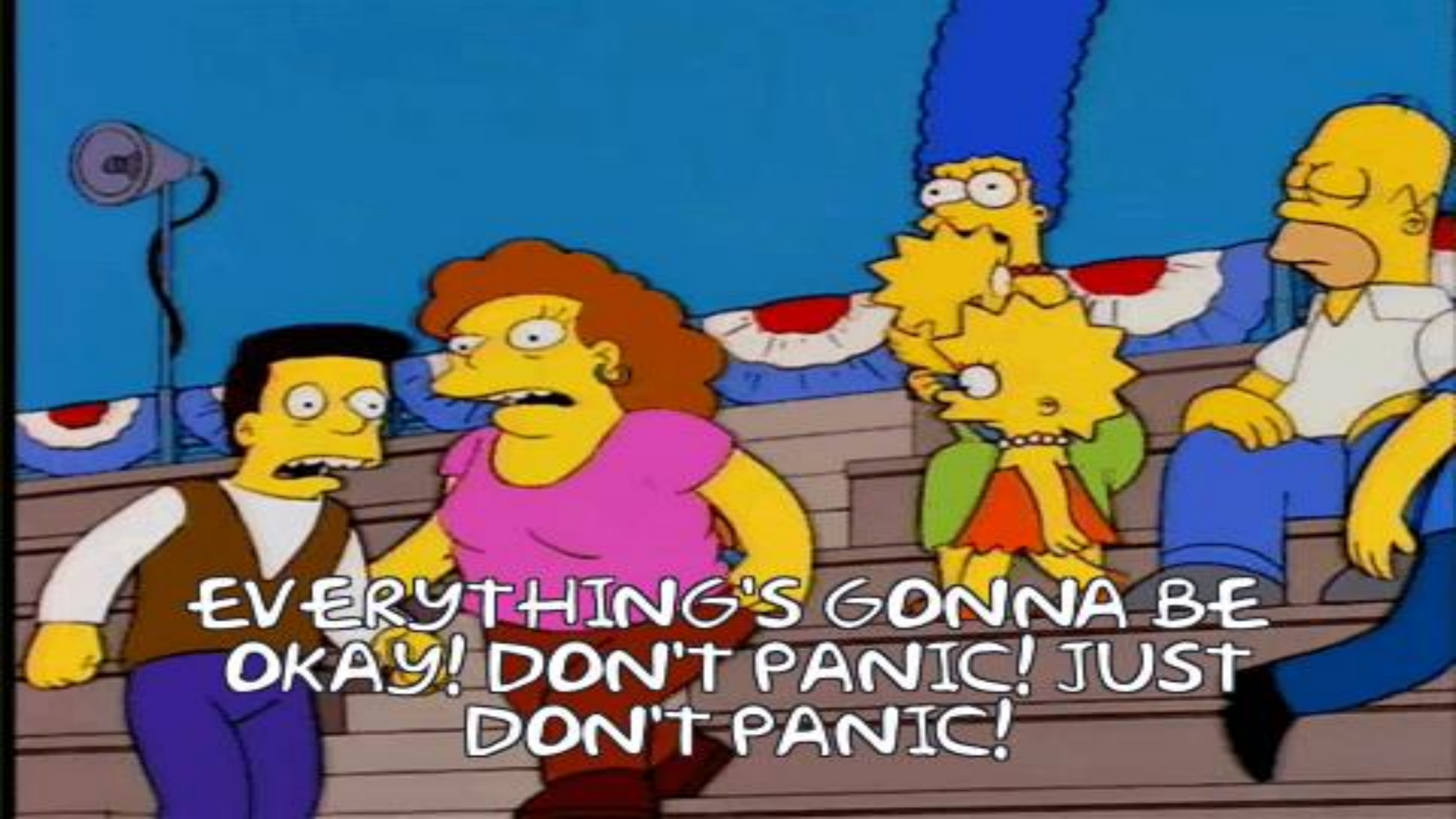


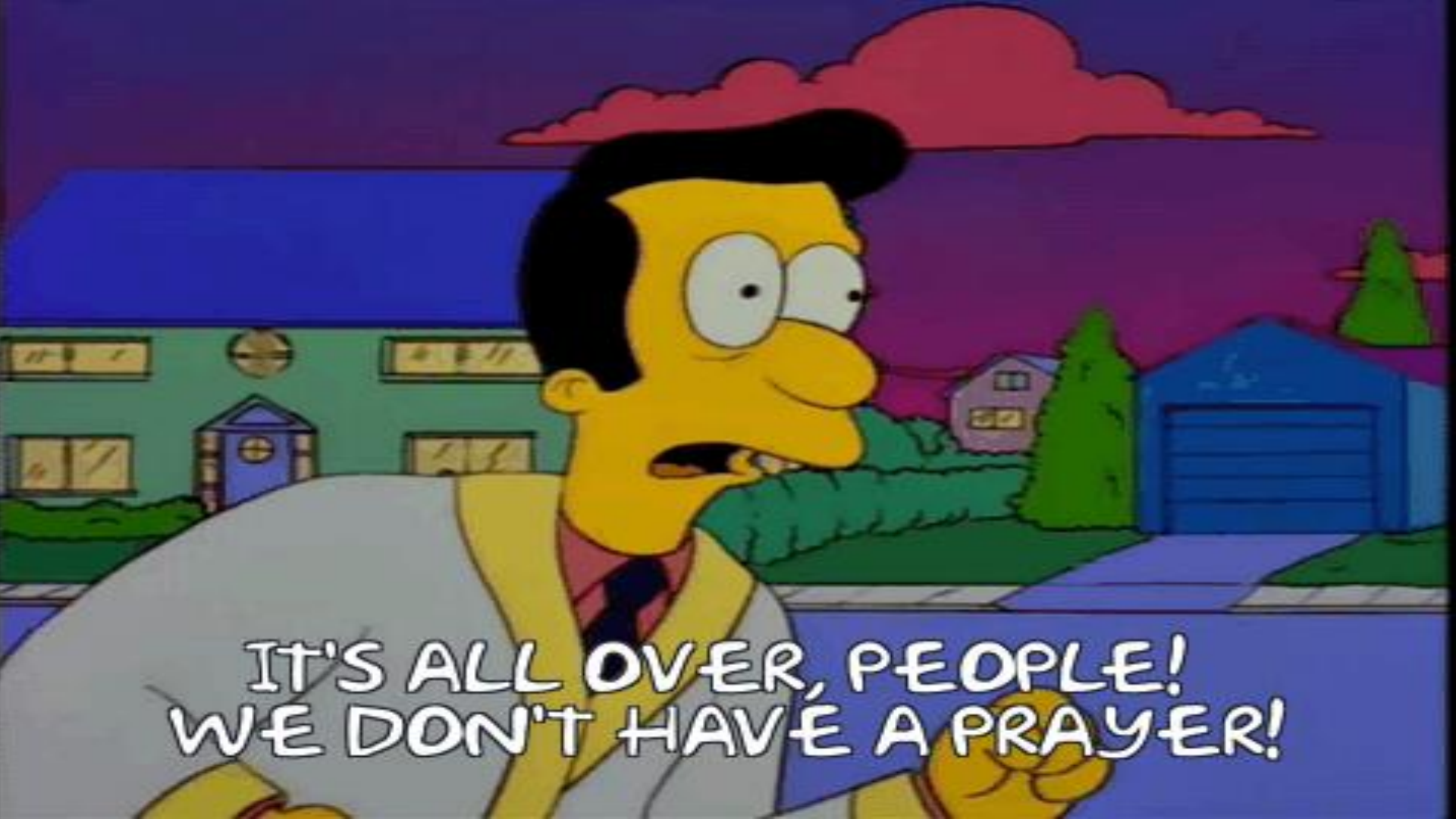


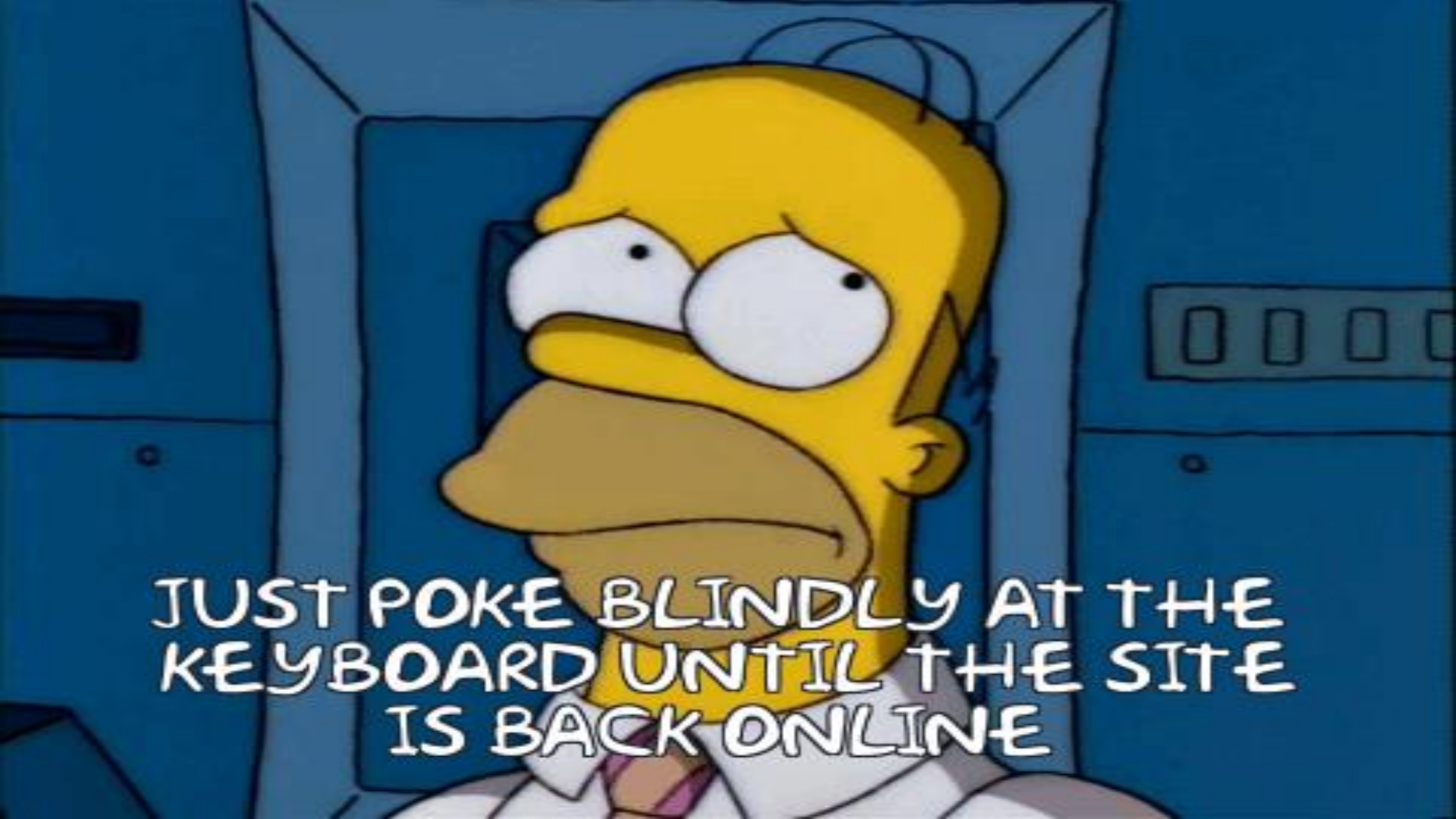


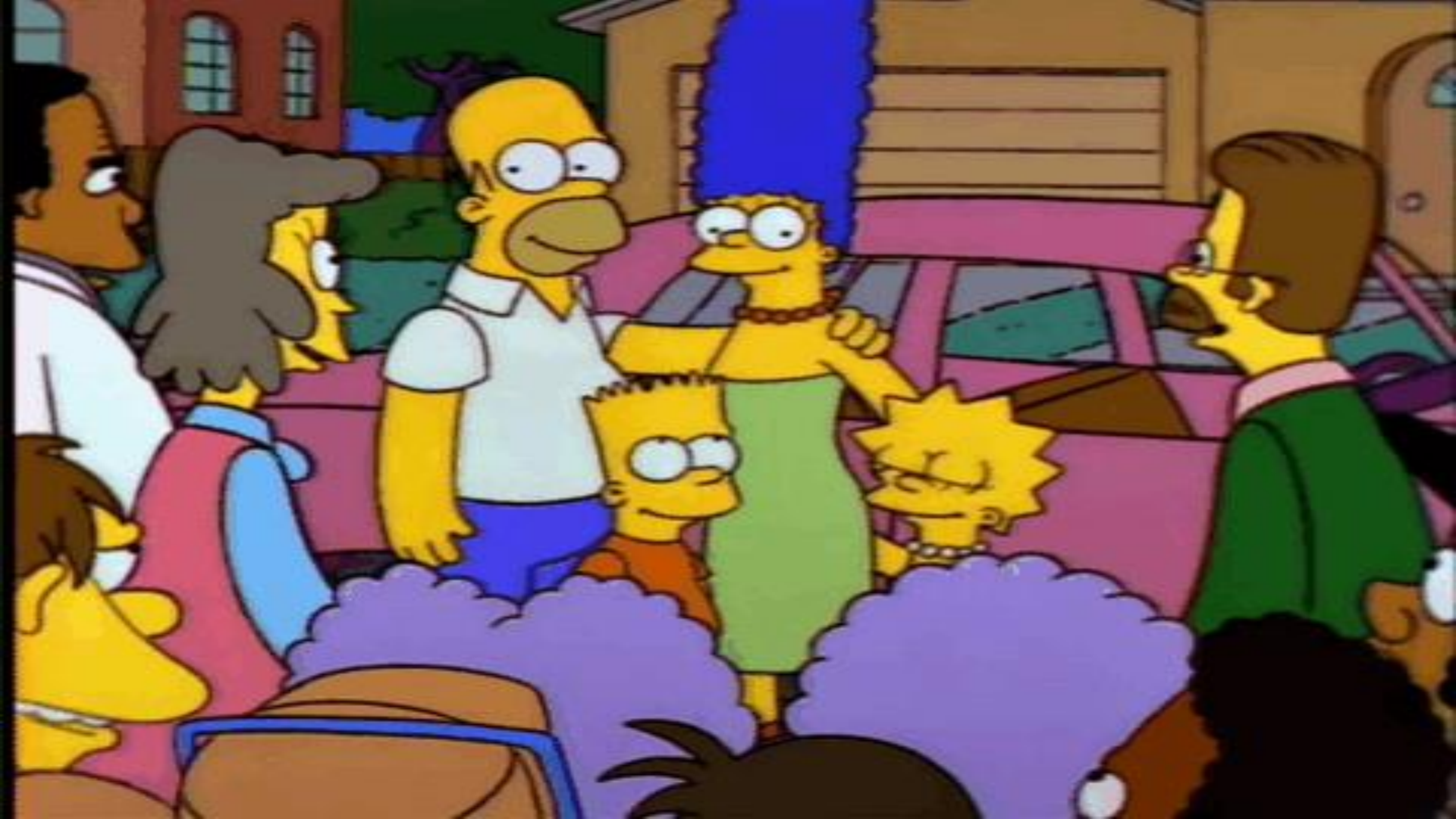


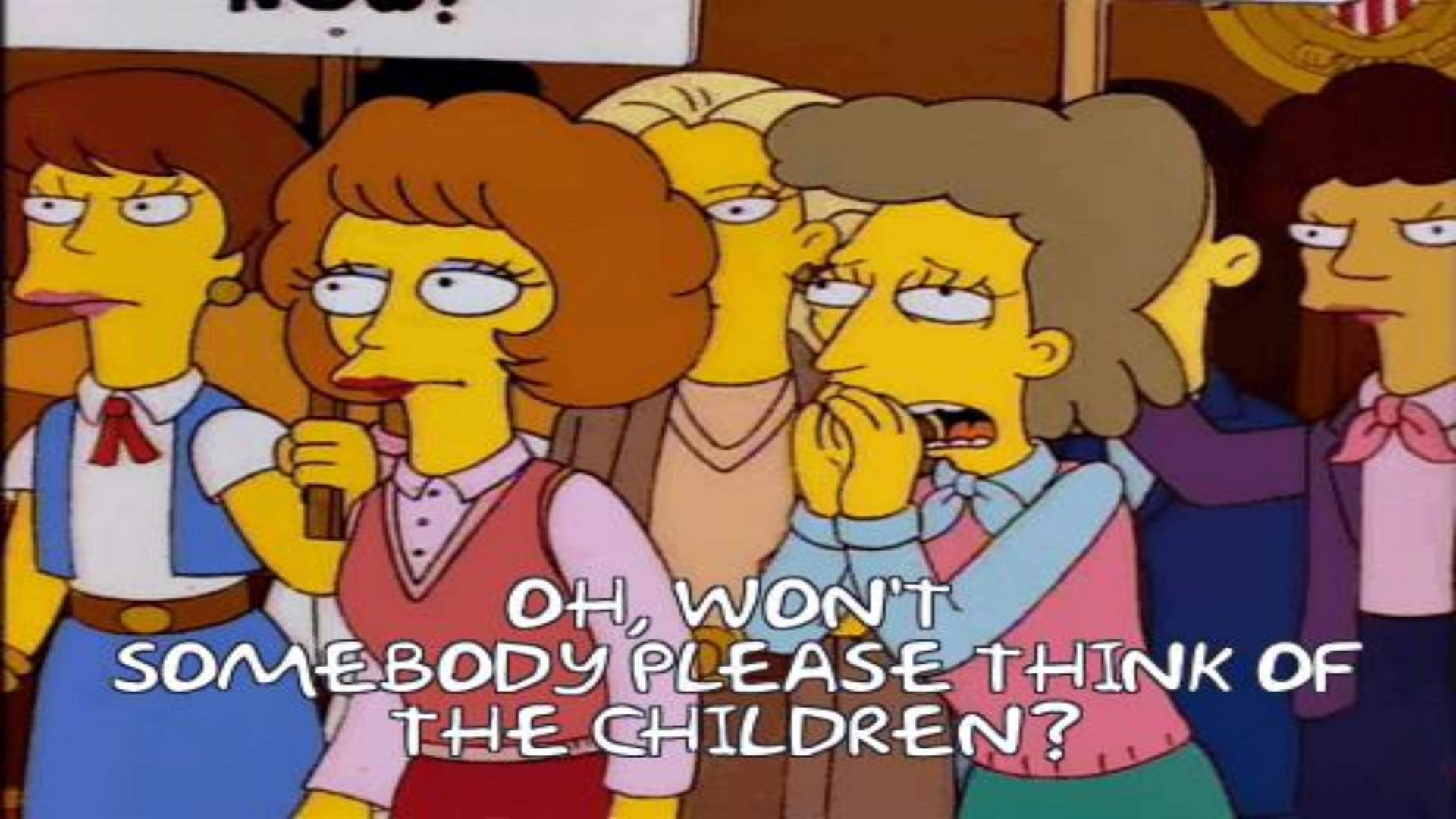

EVERYTHING'S
GOING TO BE JUST FINE.

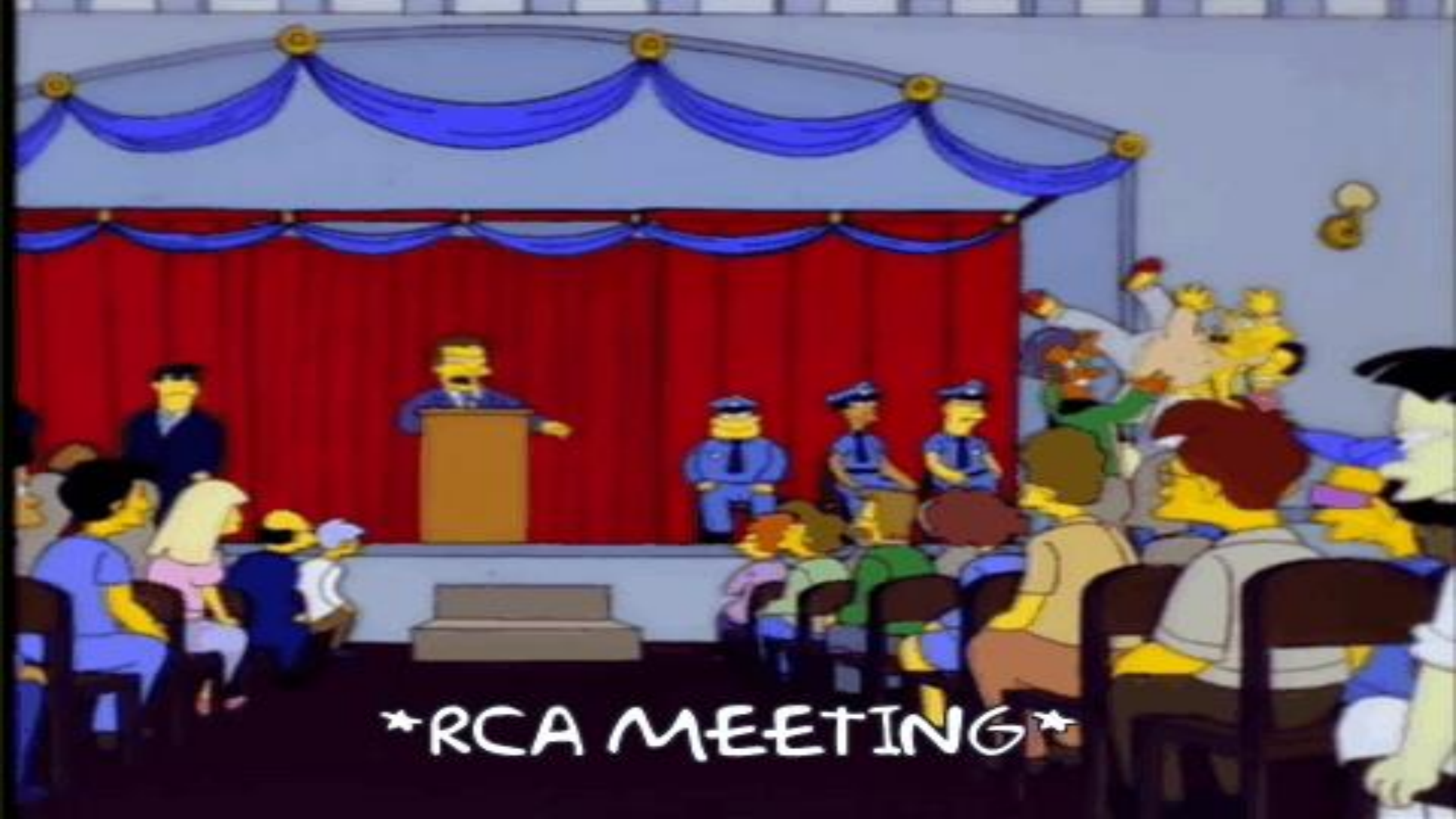

EVERYTHING'S GONNA BE OKAY! DON'T PANIC! JUST DON'T PANIC!


IT'S ALL OVER, PEOPLE!
WE DON'T HAVE A PRAYER!


JUST POKE BLINDLY AT THE KEYBOARD UNTIL THE SITE IS BACK ONLINE

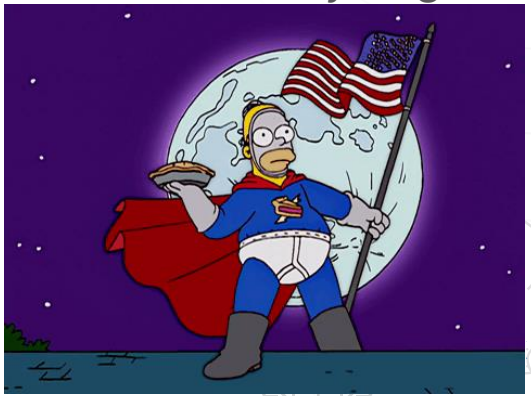


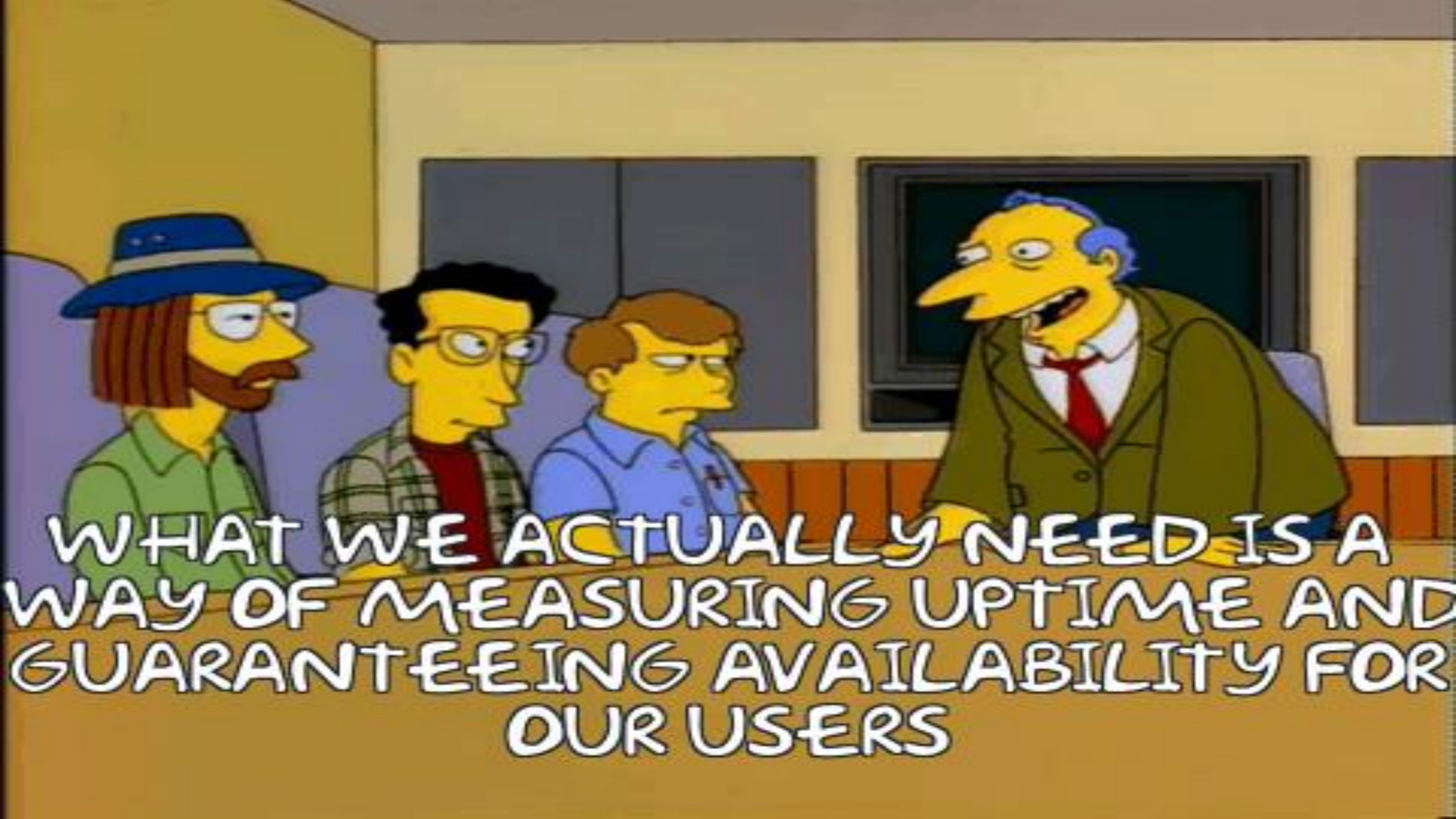

OH, WON'T
SOMEBODY PLEASE THINK OF
THE CHILDREN?


RCA MEETING

What is an SRE?

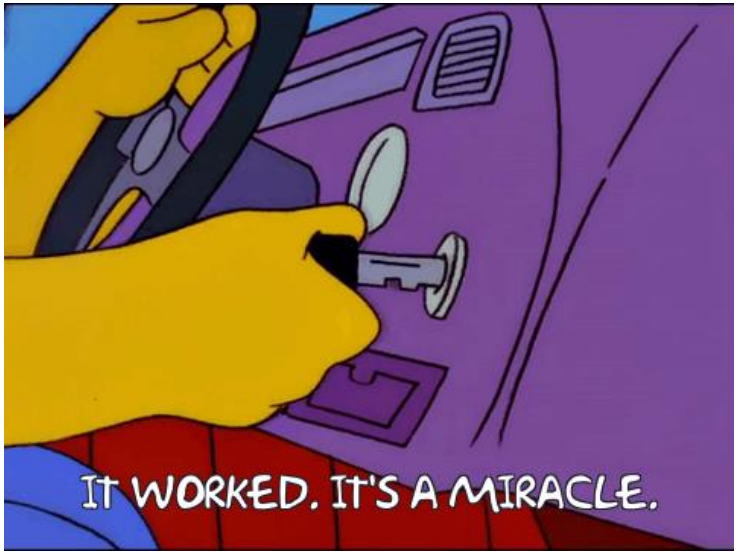
“Through this lens, then, we see that if software engineering tends to focus on designing and building software systems, there must be another discipline that focuses on the *whole* lifecycle of software objects, from inception, through deployment and operation, refinement, and eventual peaceful decommissioning. This discipline uses - and needs to use - a wide range of skills, but has separate concerns from other kinds of engineers. Today, our answer is the discipline Google calls Site Reliability Engineering”
- Google SRE book




WHAT WE ACTUALLY NEED IS A WAY OF MEASURING UPTIME AND GUARANTEEING AVAILABILITY FOR OUR USERS

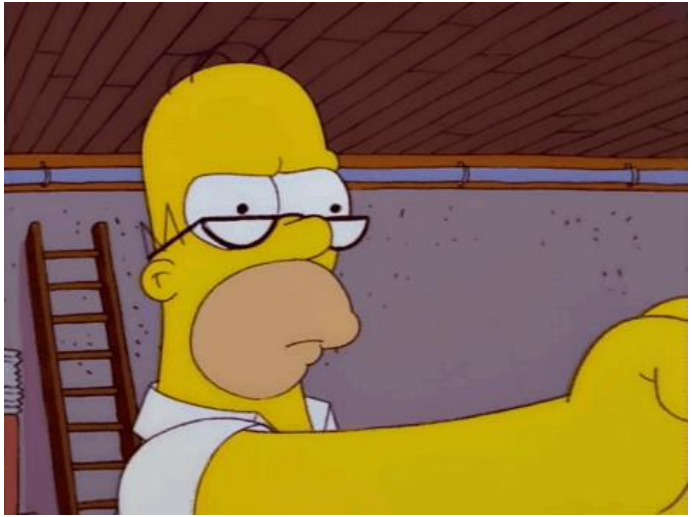
What is uptime?

- My service is running and I can ping it. That means it's up, right?
  - If the service is up but no one can access it, is it up?
  - If we have a highly distributed system and it's up in one of our 10 data centers, is it up?
- So what we really care about is availability



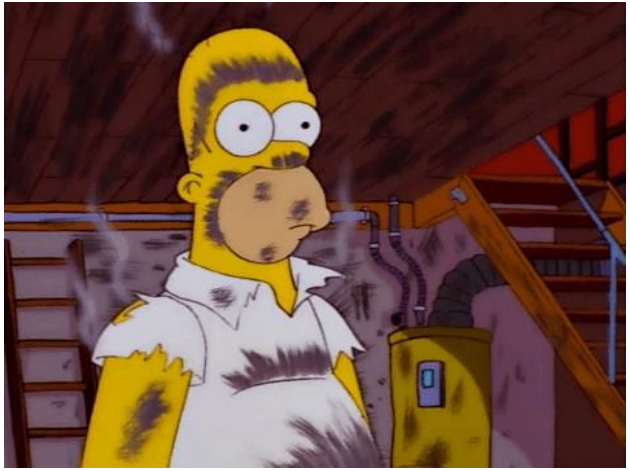
How Do We Measure Availability?

- Time-based availability $$\text{availability} = \frac{\text{uptime}}{(\text{uptime} + \text{downtime})}$$
- Aggregate availability $$\text{availability} = \frac{\text{successful requests}}{\text{total requests}}$$
  - Could also be negated for unsuccessful requests.
    - Error rate = unsuccessful requests / total requests



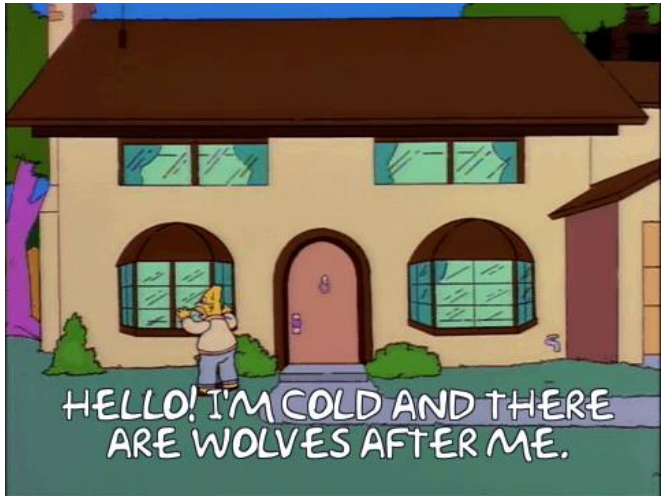
Not All Requests Are Created Equal

- Aggregate availability $$\text{availability} = \frac{\text{successful requests}}{\text{total requests}}$$
  - In complex systems not all requests are equal
    - A new user sign up request may not be as critical as a message send request
- Do different types of failures have different effects?
- What other service metrics are important to take into account?



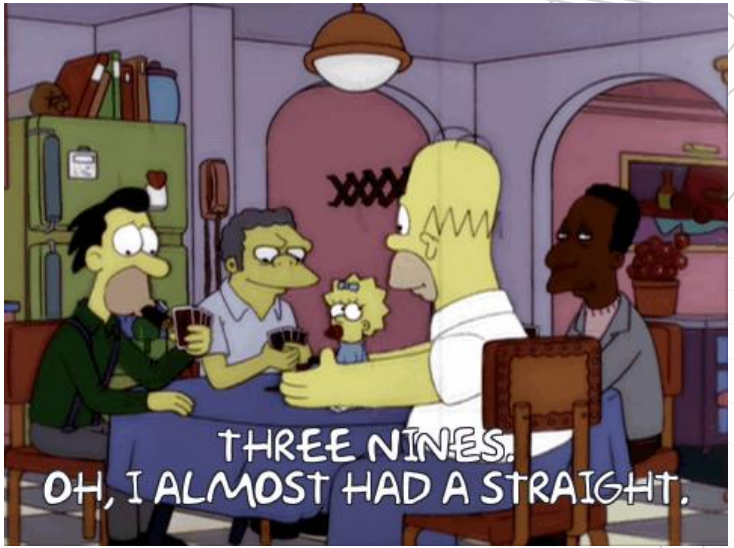
Defining Availability

- How do we know if we're available enough?
  - What level of service does the user expect?
  - Does this service tie directly to revenue (either yours or your customers)
  - Is this a free or paid service?
  - What does the competition look like?
  - Who is the target audience (consumers or enterprises)?



What about the 9s?

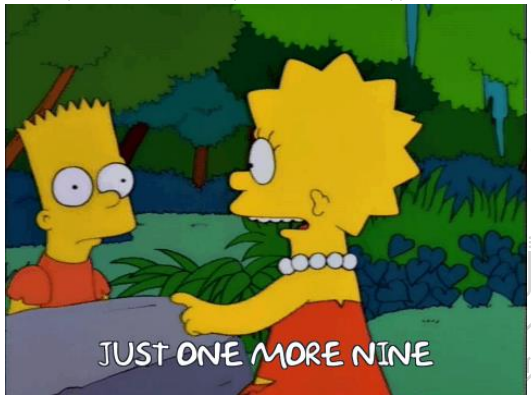
- Every DevOps, SRE, Sales Engineer, Used Car Salesman talks about the 9s.
- What is your uptime, measured in the number of 9s in your percentage.
  - 99%, 99.9%, 99.99%, 99.999999%



	Allowed unavailability window					
Availability Level	*per year*	*per quarter*	*per month*	*per week*	*per day*	*per hour*
90%	36.5 days	9 days	3 days	16.8 hours	2.4 hours	6 minutes
95%	18.25 days	4.5 days	1.5 days	8.4 hours	1.2 hours	3 minutes
99%	3.65 days	21.6 hours	7.2 hours	1.68 hours	14.4 minutes	36 seconds
99.50%	1.83 days	10.8 hours	3.6 hours	50.4 minutes	7.20 minutes	18 seconds
99.90%	8.76 hours	2.16 hours	43.2 minutes	10.1 minutes	1.44 minutes	3.6 seconds
99.95%	4.38 hours	1.08 hours	21.6 minutes	5.04 minutes	43.2 seconds	1.8 seconds
99.99%	52.6 minutes	12.96 minutes	4.32 minutes	60.5 seconds	8.64 seconds	0.36 seconds
99.999%	5.26 minutes	1.30 minutes	25.9 seconds	6.05 seconds	0.87 seconds	0.04 seconds

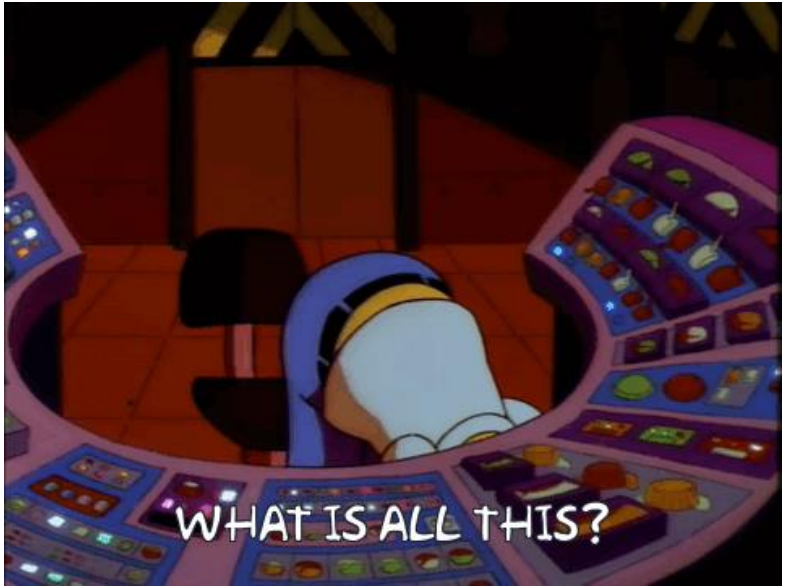
Do I Need Another 9?

- If we increase by a 9, what will the increase in revenue be?
- Does this extra revenue offset the cost of engineering?
  - Proposed improvement in availability target: 99.9% → 99.99%
  - Proposed increase in availability: 0.09%
  - Service revenue: $1M
  - Value of improved availability: $1M * 0.0009 = $900
- If you can build out the extra 9 for <= $900 then it's worth it. Otherwise you'll spend more money than you'll be making.
  - SRE @ $120,000 makes ~ $57 an hour (120000 / (8 * 5 * 52)) so a $900 budget would only allocate 15 and a half hours.



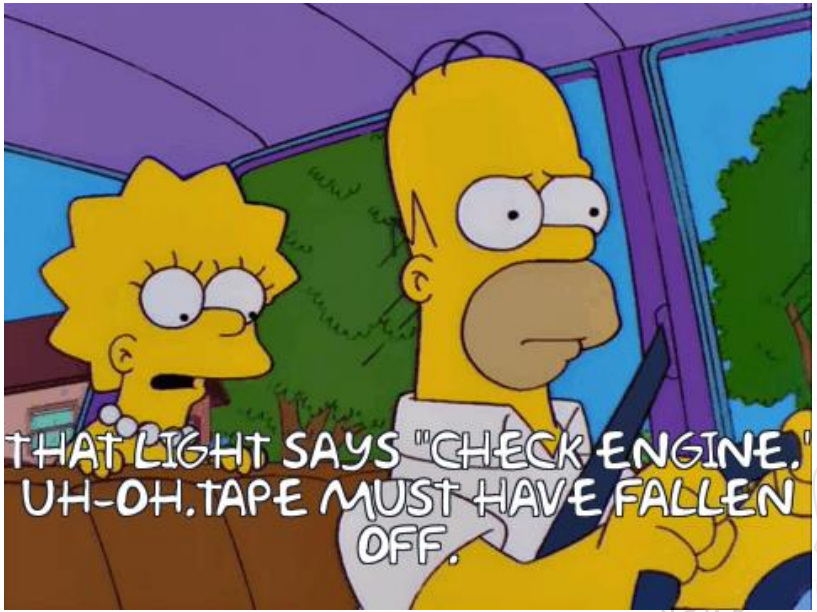
What are SLIs, SLOs, and SLAs?

- SL - Service Level
  - Indicators - Metrics that matter
  - Objectives – What ranges the metrics should be in
  - Agreements - How we'll react if something goes wrong



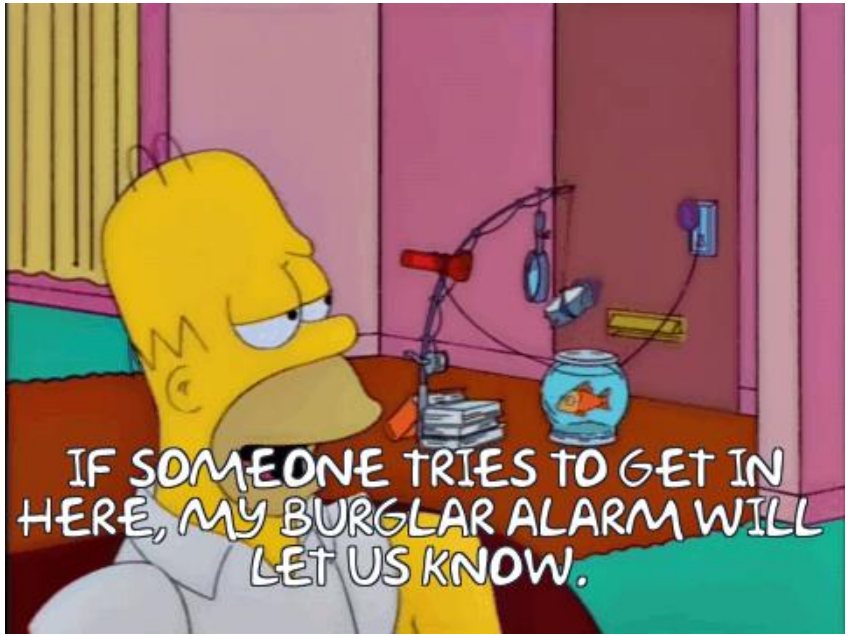
Service Level Indicator

- “A carefully defined quantitative measure of some aspect of the level of service that is provided.” - Google SRE Book
- Examples:
  - Request Latency
  - Error Rate
  - System Throughput



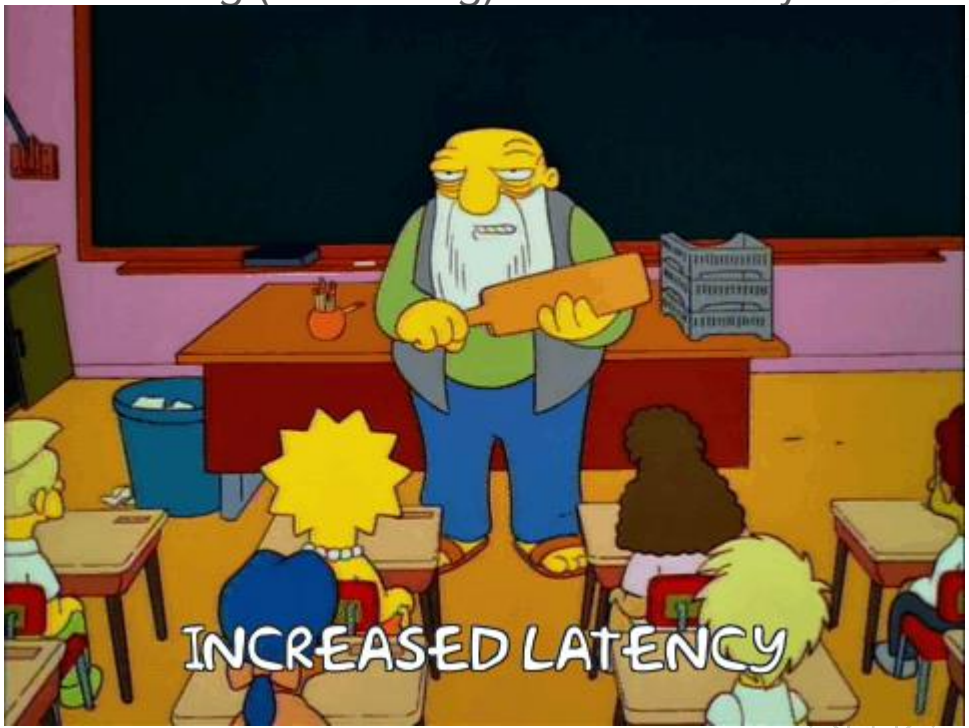
Service Level Objective

- “A target value or range of values for a service level that is measured by an SLI” - Google SRE Book
- Lower Bound <= SLI <= Upper Bound



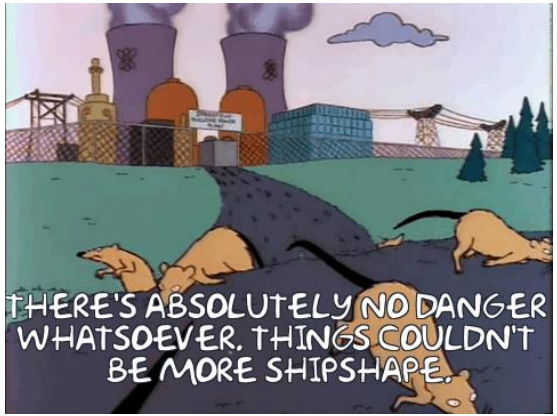
SLAs

- “An explicit or implicit contract with your users that includes consequences of meeting (or missing) the SLOs they contain” - Google SRE Book

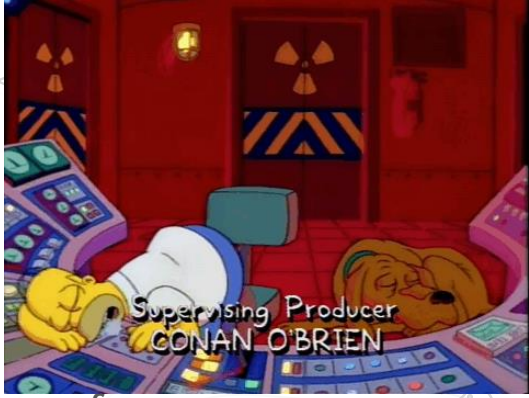


How Do We Define SLIs?

- What do you and your users care about?
- User-facing serving systems usually care about **availability, latency,** and **throughput.**
  - Could we respond to a request? How long did it take? How many requests could we handle?
- Storage systems usually care about **latency**, **availability**, and **accuracy**.
  - How long does it take to perform IO? Can I access my data when I need it? Is the data correct?



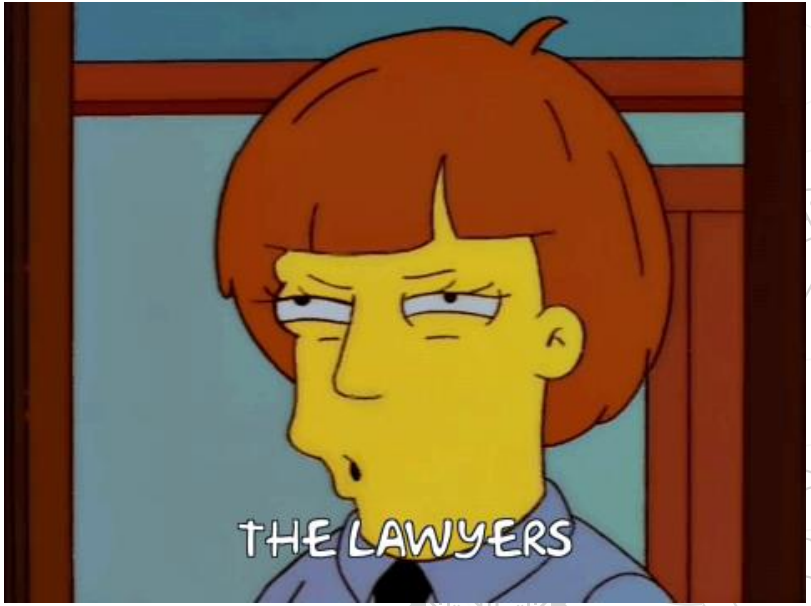
How Do We Define SLOs?

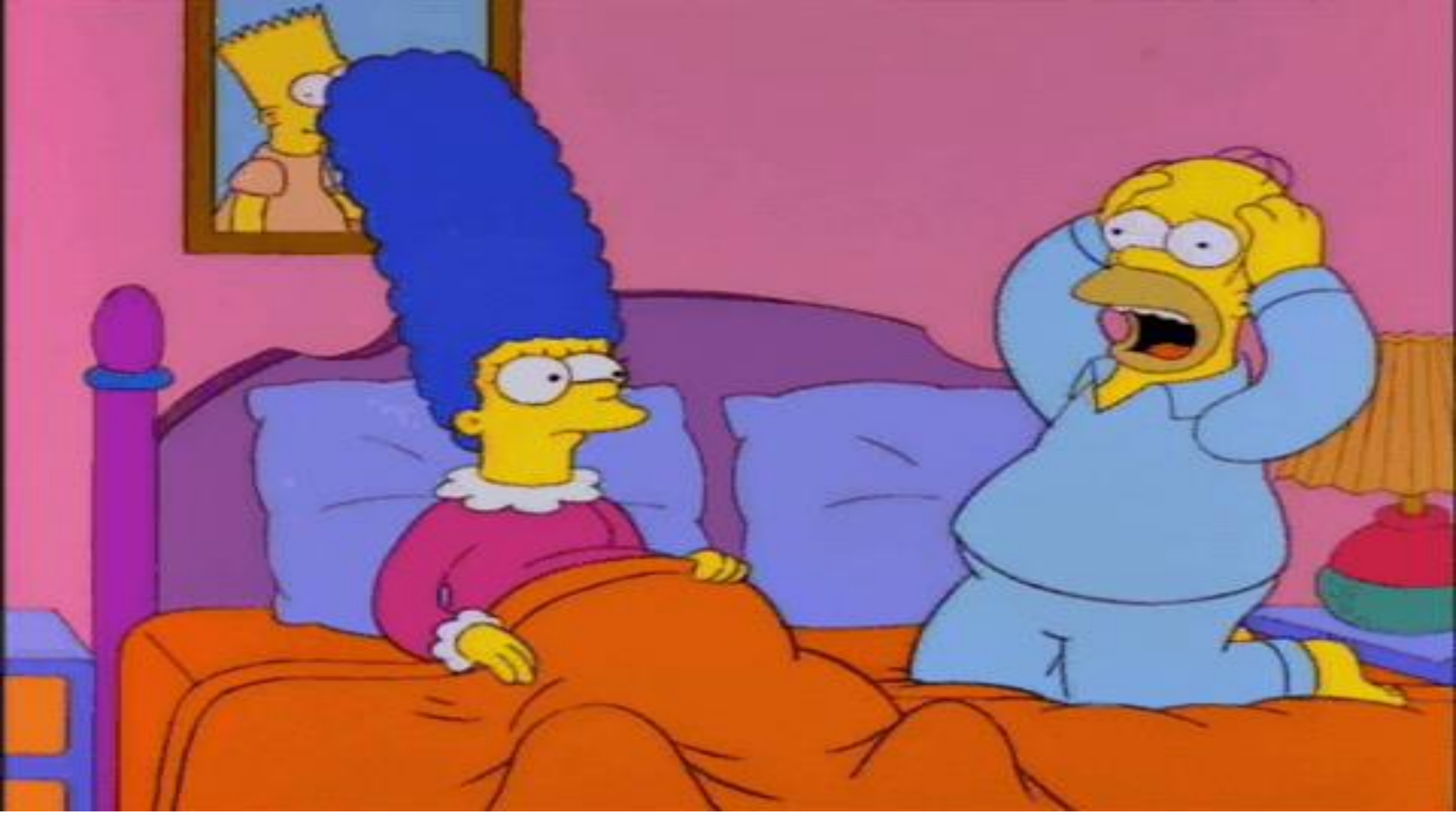


- Keep it Simple
  - To many data points can obscure changes in performance
- Avoid absolutes
- Be cautious when picking a target based on current performance.
  - Avoid getting locked into supporting something that could require herculean efforts to support in the future.
  - If you have no other metrics, it's a good place to start.
- Limit the number of SLOs
  - Chose enough SLOs to cover your system and allow you to win a conversation about priorities by quoting an SLO. If you can't win an argument with an SLO, it's probably unnecessary.
    - Note, some attributes such as "user satisfaction" aren't covered by SLOs, so there is a grey area.
- Perfection can wait
  - It's easier to add 9s than to take away 9s.

How Do We Define SLAs?

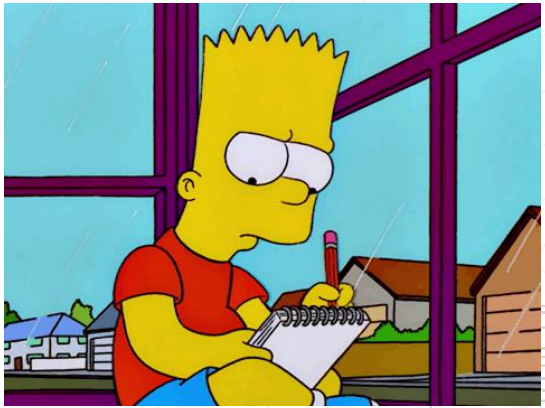
- SLAs are *usually* defined at a company level between business and legal involved since this usually involves reparations for customers.
- If your customers are internal developers then it may be within your team's ability to define SLAs.





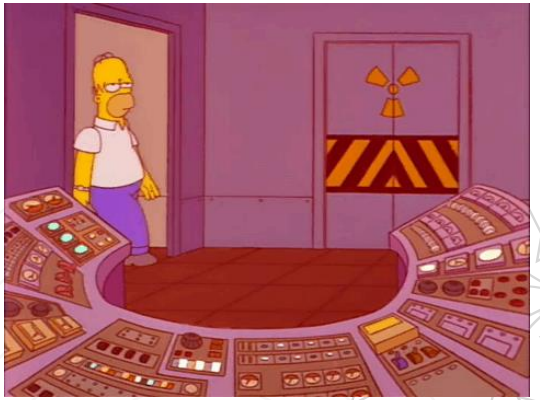
What Should I Actually Care About?

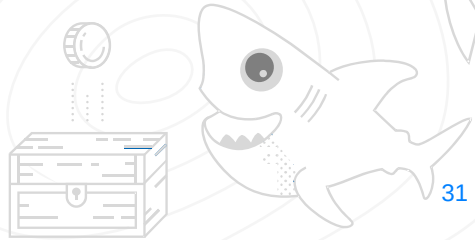
- SLIs are things that can be monitored and measured. SLOs are the ranges that are acceptable.
- SLOs set expectations for your service.
- Monitor and measure your system's SLIs.
- Compare the SLIs to your SLOs. If action needs to be taken, take it.
- Determine *what* action needs to be taken to meet a target. Creating runbooks or automation tasks are good for this.
- Take action when needed.



Tips

- Have internal SLOs to your team that are higher than your external facing SLOs.
- Users build based on the service you provide rather than what you say. If you provide way higher than what you say that's what they'll become accustomed to.
  - You can avoid this by rate-limiting requests, designing the system to produce similar performance on light and heavy loads, or planned outages.
  - Google's Chubby service introduced planned outages in response to being overly available
- Breaches of Agreements happen. Just breath and get the system back online. That's what RCAs are for.
- There is no room for blame in an SRE org. All post mortems should be blameless.





YEARDLEY SMITH

NO, BART IS NOT AVAILABLE
TOMORROW TO DELIVER A HUMAN
KIDNEY TO AMSTERDAM.