

25 September 2024

# Deprecation is Hard to Do, and We Can Do it Better

# Agenda

01 Quick talk

02 Discussion

03 Next Steps

01

# Quick Talk

# Deprecation definition:

the classification of a software feature as obsolete and best avoided, typically because it has been superseded.

# Classes of Deprecation

## Unshipped

Not available

The feature is no longer in the browser and can not be used by a website.

## Disabled

Opt-in

A website can decide to use the feature by taking some action.

## Discouraged

Opt-out

A website automatically uses the feature, but may choose not to and is guided not to by best-practices or warnings

# Why should we deprecate things?

1

Accessibility

2

Site Security

3

User protections

4

Platform improvement

5

Probably more that I'm not thinking of here.

02

# Discussion



# Discussion Topics for the Session

1

## Harms

What happens if we rip it out?

Notes:

- A. Unknown use cases, can not reach zero, but must try (Hyrum's Law)
- B. Unknown unknowns!
- C. You break things, and in theory websites should work forever
- D. Outsized influence on smaller development ops
- E. Don't want to use threat of deprecation to leverage new web spaces that would not be - can feel like a bait and switch
  - a. Sometimes browsers have to be vague :(
- F. Ecosystem implications, economics and incentives

2

## Mitigations

What can we do to make it less bad?

Notes:

- A. Storage access heuristics
  - a. weren't standardized because we wanted to not make them permanent
  - b. Temporary site-specific workarounds are needed, and useful here
    - i. Impact reliability for web-devs and equity
    - ii. Takes agency away from web devs: well-known thing
- B. Off by default instead of hard removal (depends on threat model justifying deprecation), allows long timescale (document.domain knowledge)
  - a. Initially opt-in to removal

3

## Whether or not to Act

What are the criteria to go for it?

Notes:

- A. Mitigations are not impermanent
- B. Prioritization: harms
- C. Alignment
- D. Presence of an alternative
- E. User benefit/harm
- F. Platform benefit (perf, arch, UX)
  - a. v0 of a spec
    - i. Who is likely to be using it? Will they hear?
    - ii. How ossified is it?
- G. How does the feature affect other sites?
- H. What is backwards compatibility?
- I. Need Human Feedback.



03

# Next Steps

**WHATWG/com**  
**pat**

# Thank you