R2Z2: Detecting Rendering Regressions in Web Browsers through Differential Fuzz Testing

Suhwan Song, Jaewon Hur, Sunwoo Kim*, Philip Rogers^, Byoungyoung Lee

서울대학교 SEOUL NATIONAL UNIVERSITY

*Samsung Research

^Google

: sshkeb96@snu.ac.kr

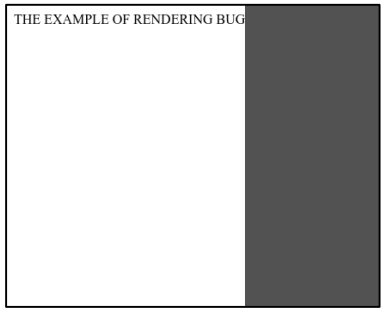
: https://suhwansong.github.io/

Rendering bug

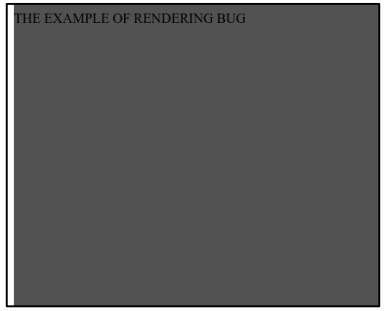
- A rendering bug is a bug when browser fails to correctly render an HTML.

```
<style>
  span {
    backdrop-filter:
          hue-rotate(1deg);
    filter:
          brightness(0.3);
    padding: 60%;
  }
</style>
<body>THE EXAMPLE OF
RENDERING BUG <span></span>
</body>
```

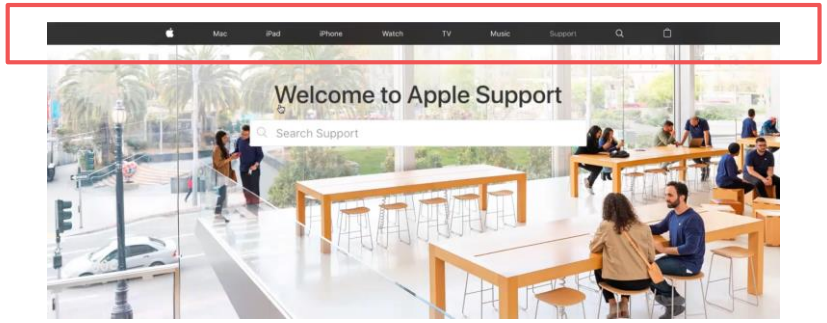
Example code



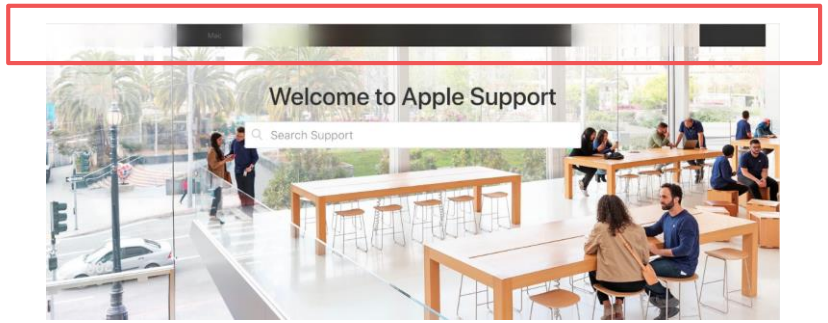
Correct (Chrome 85)



Incorrect (Chrome 87)

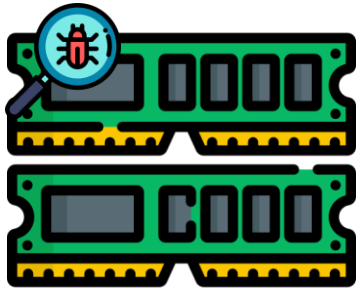


Correct (Chrome 79.0.3944)



Incorrect (Chrome 79.0.3945)

Identifying rendering bugs is challenging

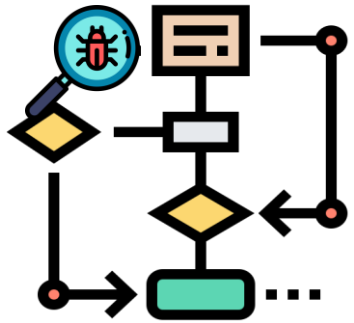


Memory Corruption

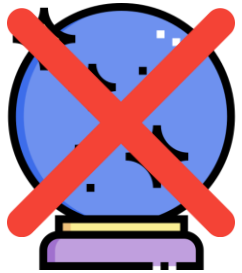
Memory Corruption Oracle

Why? Can detect valid & invalid memory region

Rendering bug is semantic bug!

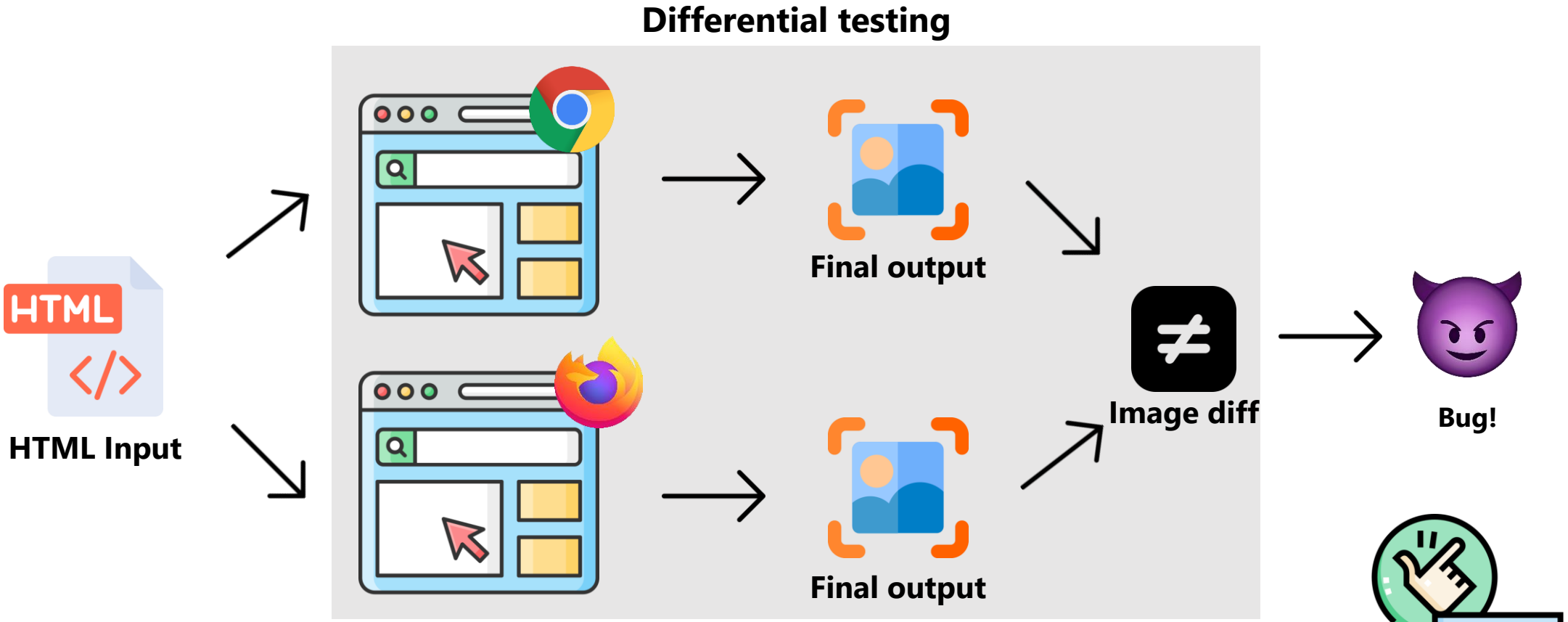


Semantic Bug



Rendering Oracle

Naïve approach



Rendering bug if two browsers produce different results!

“Image results” are meant to be different

Cross-browser testing alone can generate many false positives due to **benign browser incompatibilities**

1. Different features

CSS contain: strict is supported by **Chrome** but not **Safari**.

```
<style>
 div { contain: strict; }
</style>
<div>invisible</div>
```

Example code

Chrome

invisible

Safari

2. Different benign design

<input type=“file”> design in **Chrome** and **Firefox** are different

```
<body>
  <input type="file">
</body>
```

Example code

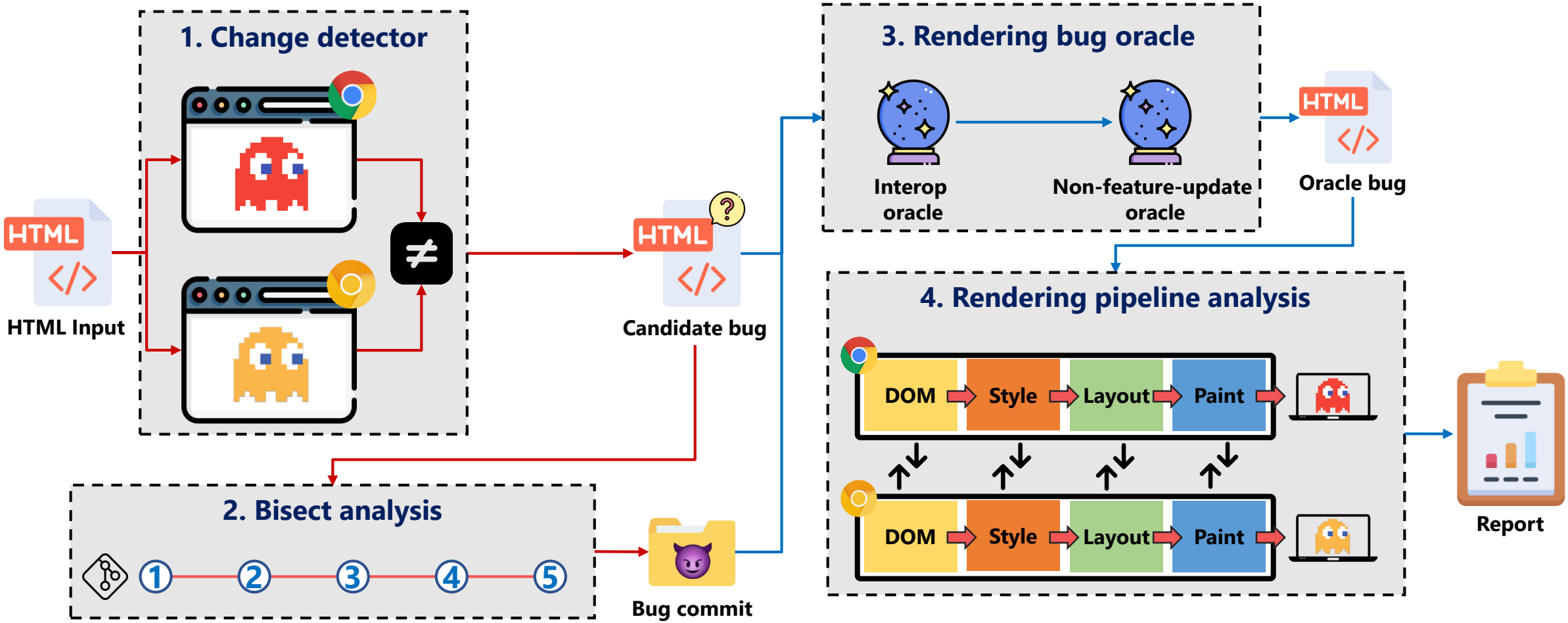
Choose File No file chosen

Chrome

Browse... No file selected.

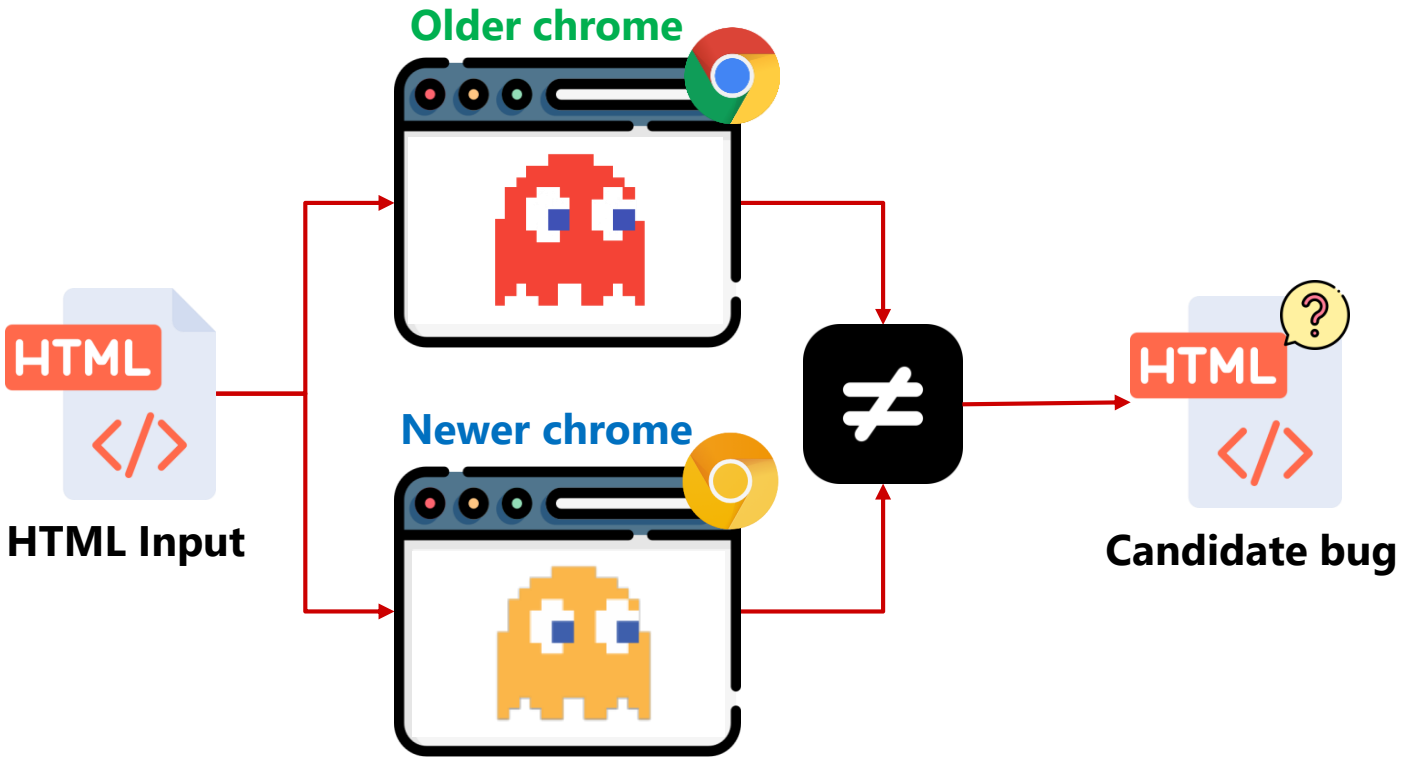
Firefox

R2Z2 overview



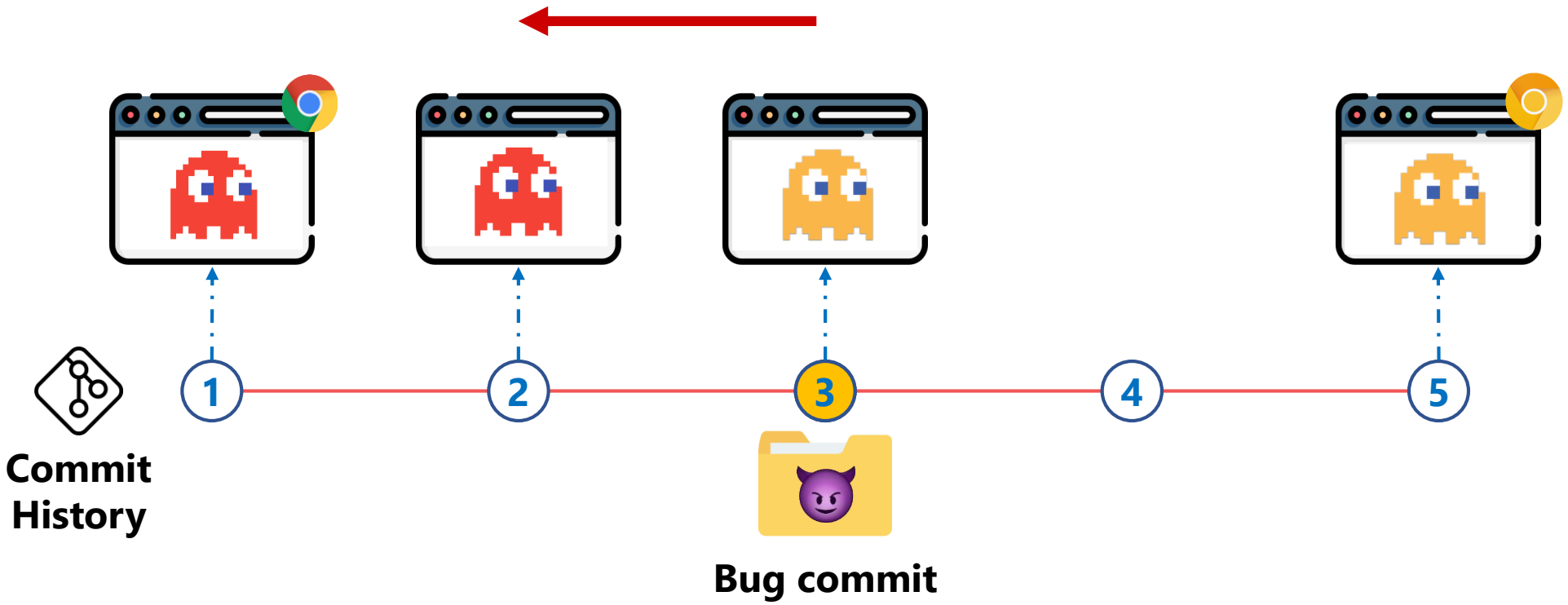
Change detector

Cross-version differential testing to detect rendering changes **between two browser versions**



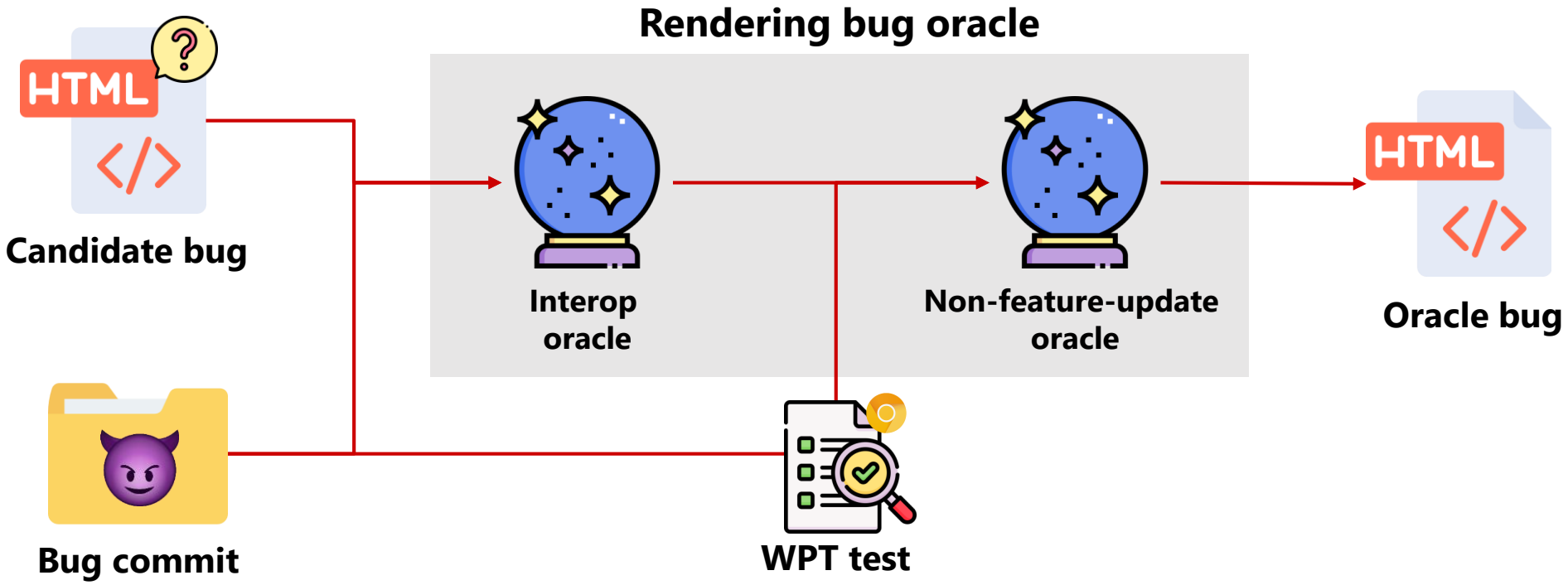
Bisect analysis

Binary search to find the bug commit,
which first introduces the rendering change.



Rendering bug oracle

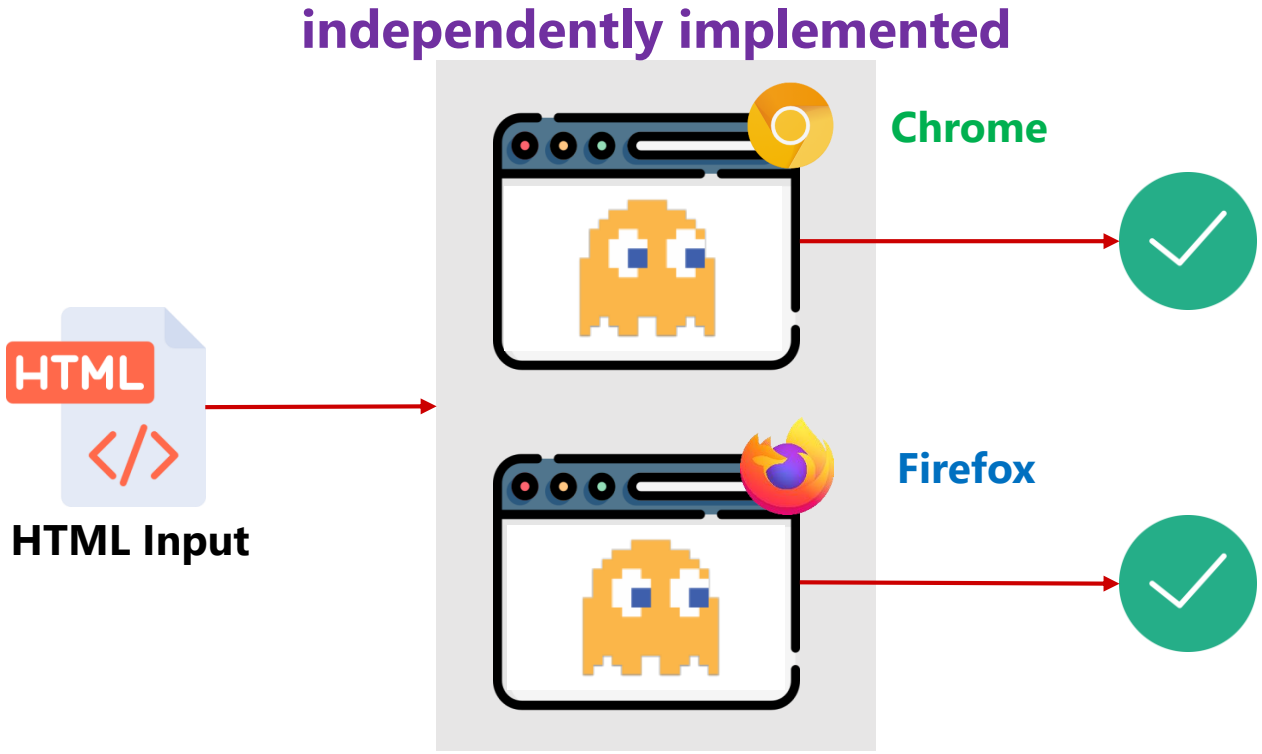
Filter out false positives by using
the candidate bug and its bisected browser version.



Our assumption

- If **two independently-implemented** browsers generate the same rendered results

  ⇒ **both produce the correct rendering**

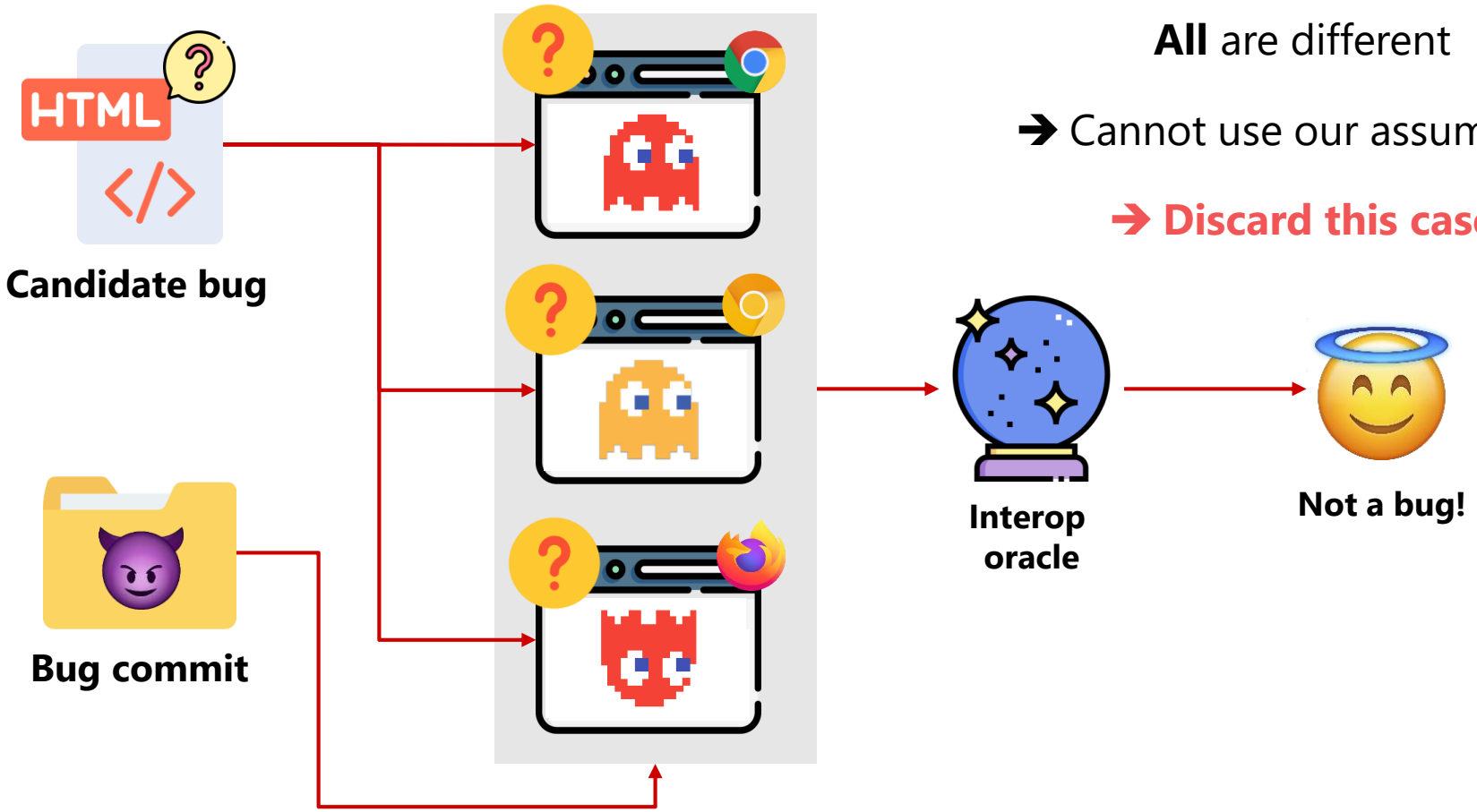


Interop oracle

Case 1: All of three are different.

Assumption:
If two independently-implemented browsers generate the same rendered result

⇒ both produce the correct rendering



All are different

➔ Cannot use our assumption

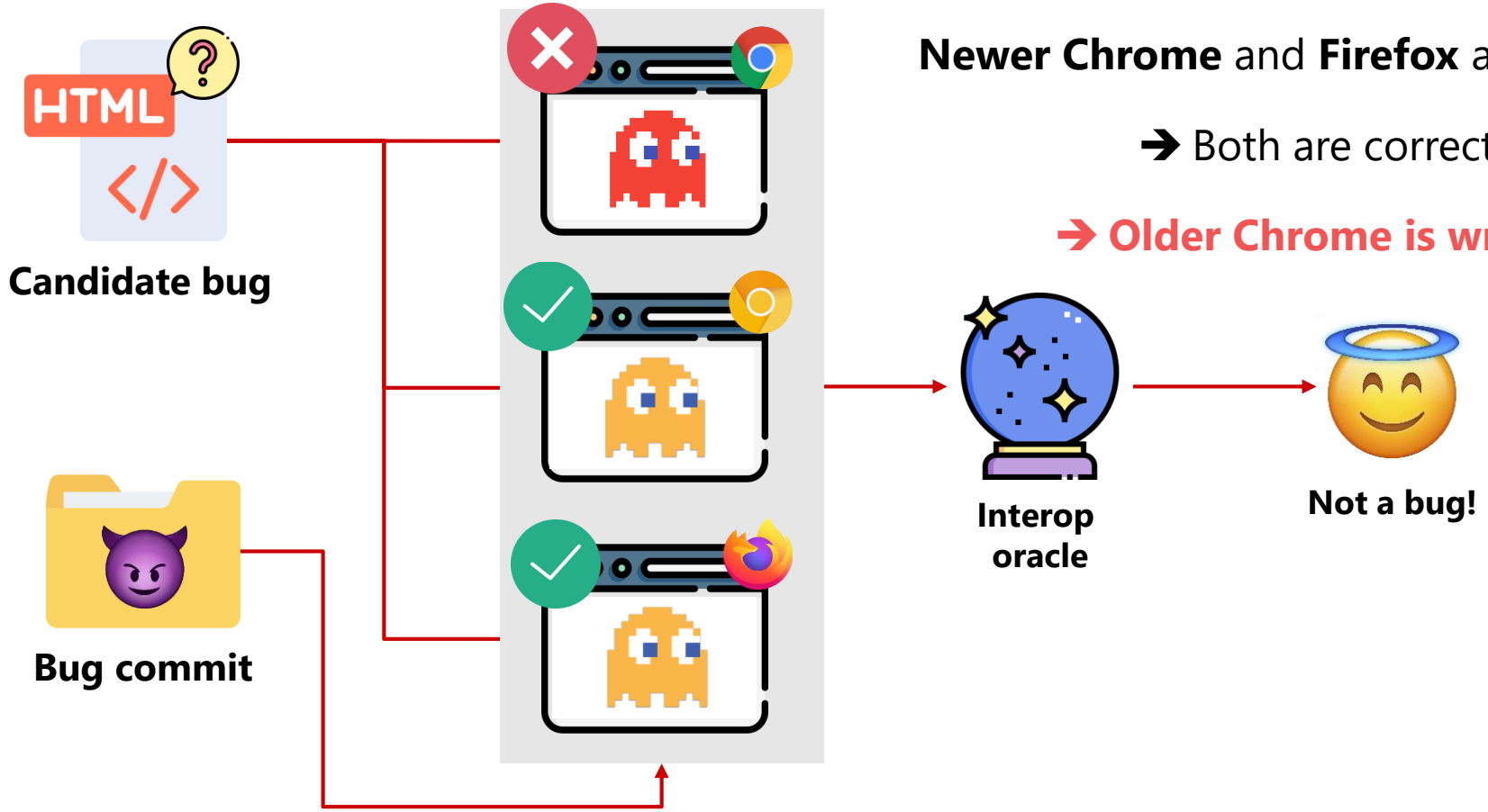
➔ **Discard this case**

Interop oracle

Case 2: Only is different.

Assumption:
If two independently-implemented browsers generate the same rendered result

⇒ both produce the correct rendering



Newer Chrome and **Firefox** are the same

➔ Both are correct

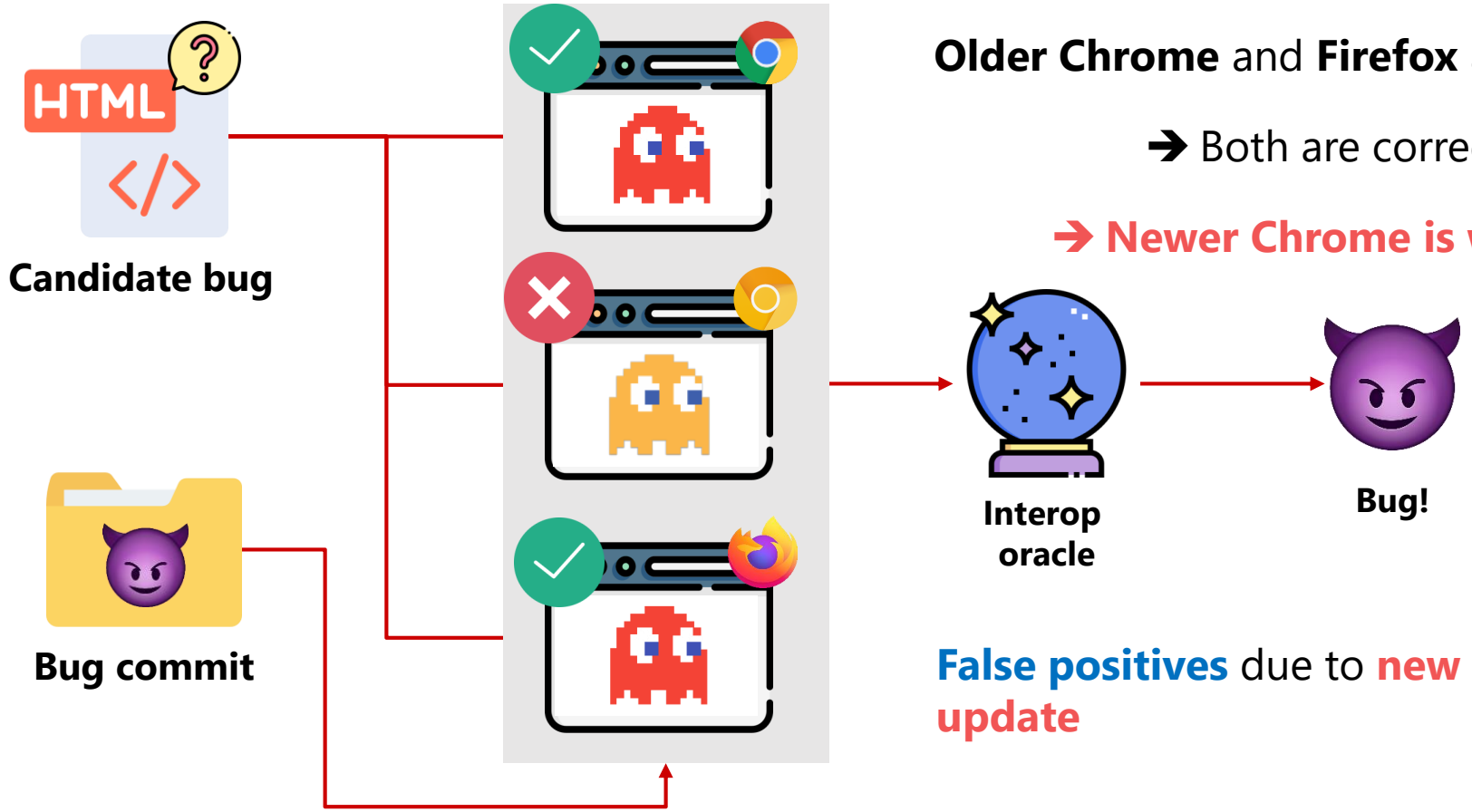
➔ **Older Chrome is wrong**

Interop oracle

Case 3: Only is different.

Assumption:
If two independently-implemented browsers generate the same rendered result

⇒ both produce the correct rendering



Older Chrome and **Firefox** are the same

➔ Both are correct

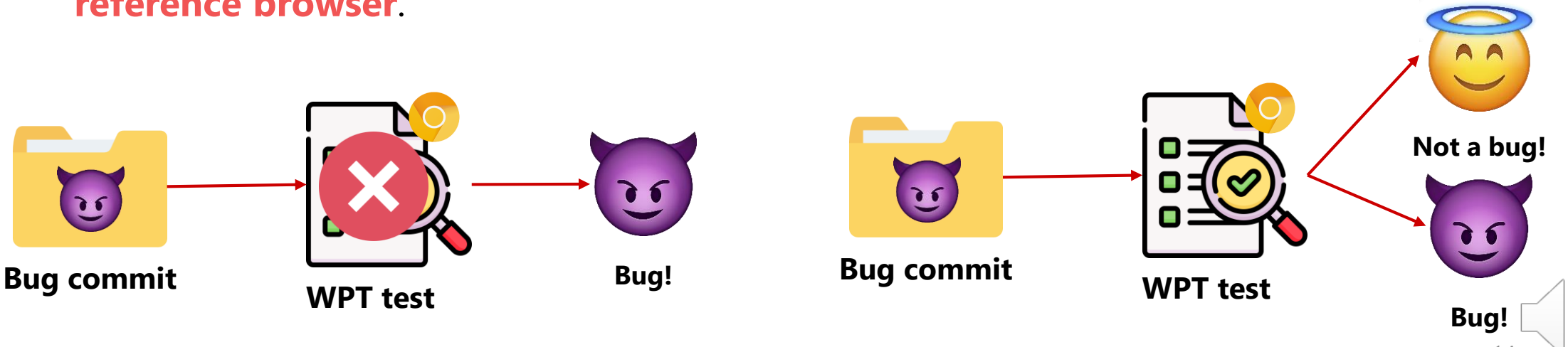
➔ **Newer Chrome is wrong**

False positives due to **new feature update**

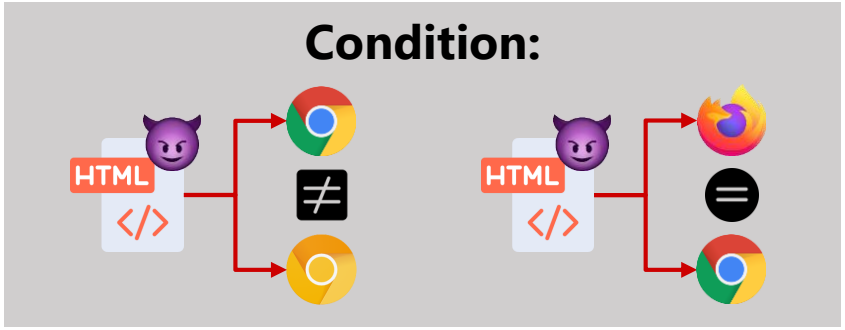
Non-feature-update oracle

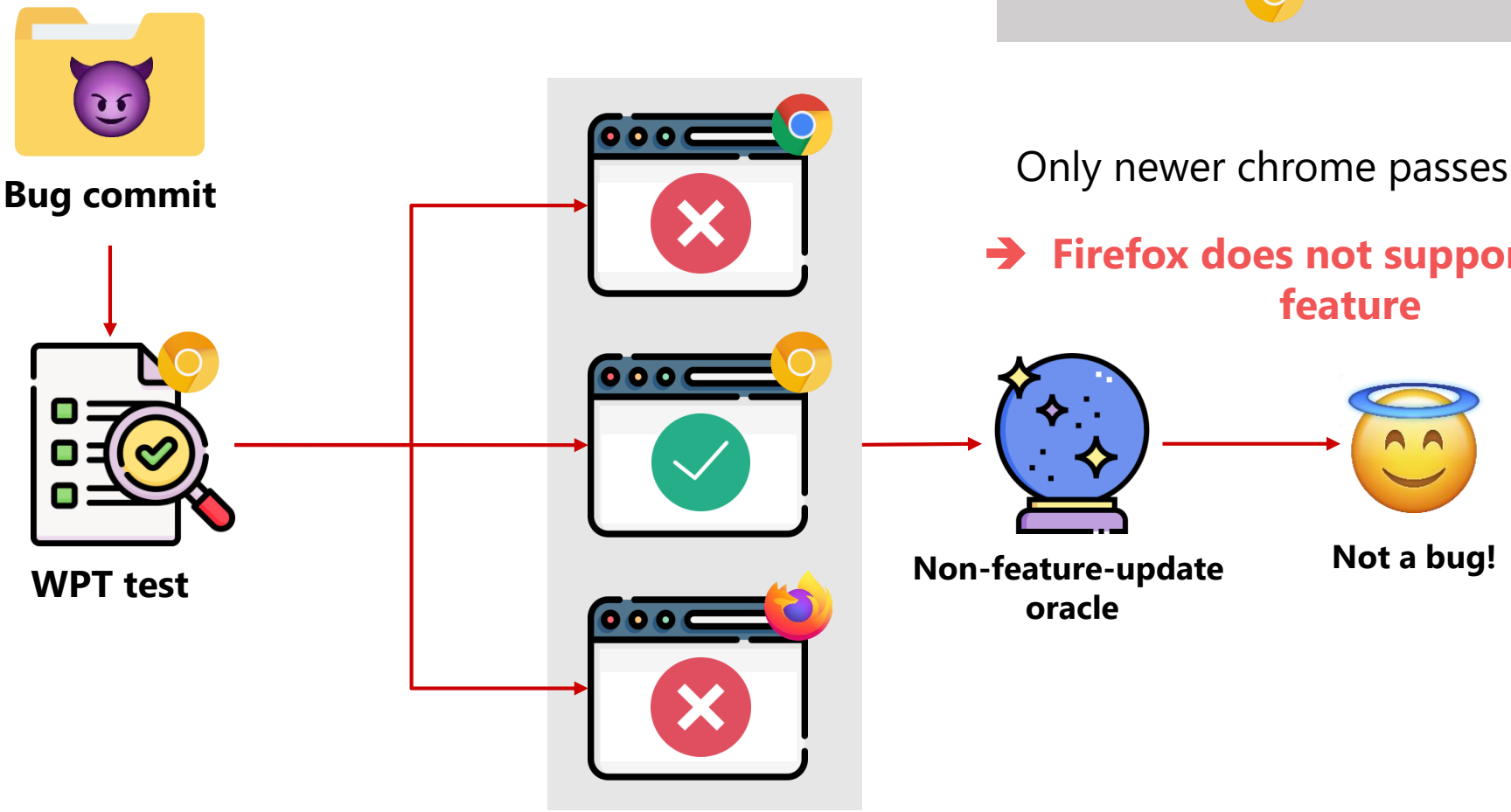
- Remove false positives by **using web-platform-test (WPT) tests**.
  - **WPT test** is a test file that web browser developers may add for code commit to **validate the new feature**.

- Use WPT tests to check whether
  - **bad commit introduces** a **new rendering feature** that is **not supported by reference browser**.



Non-feature-update oracle

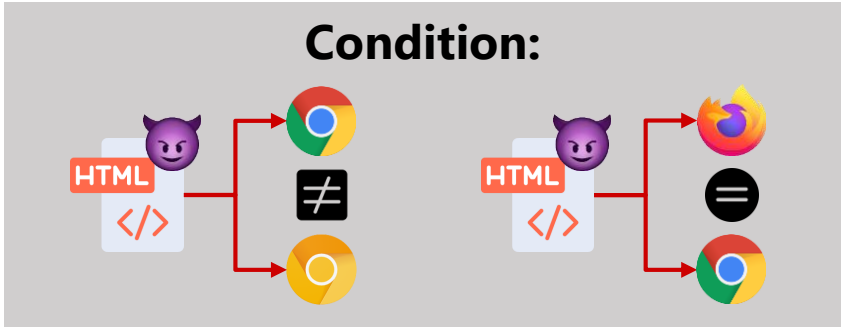


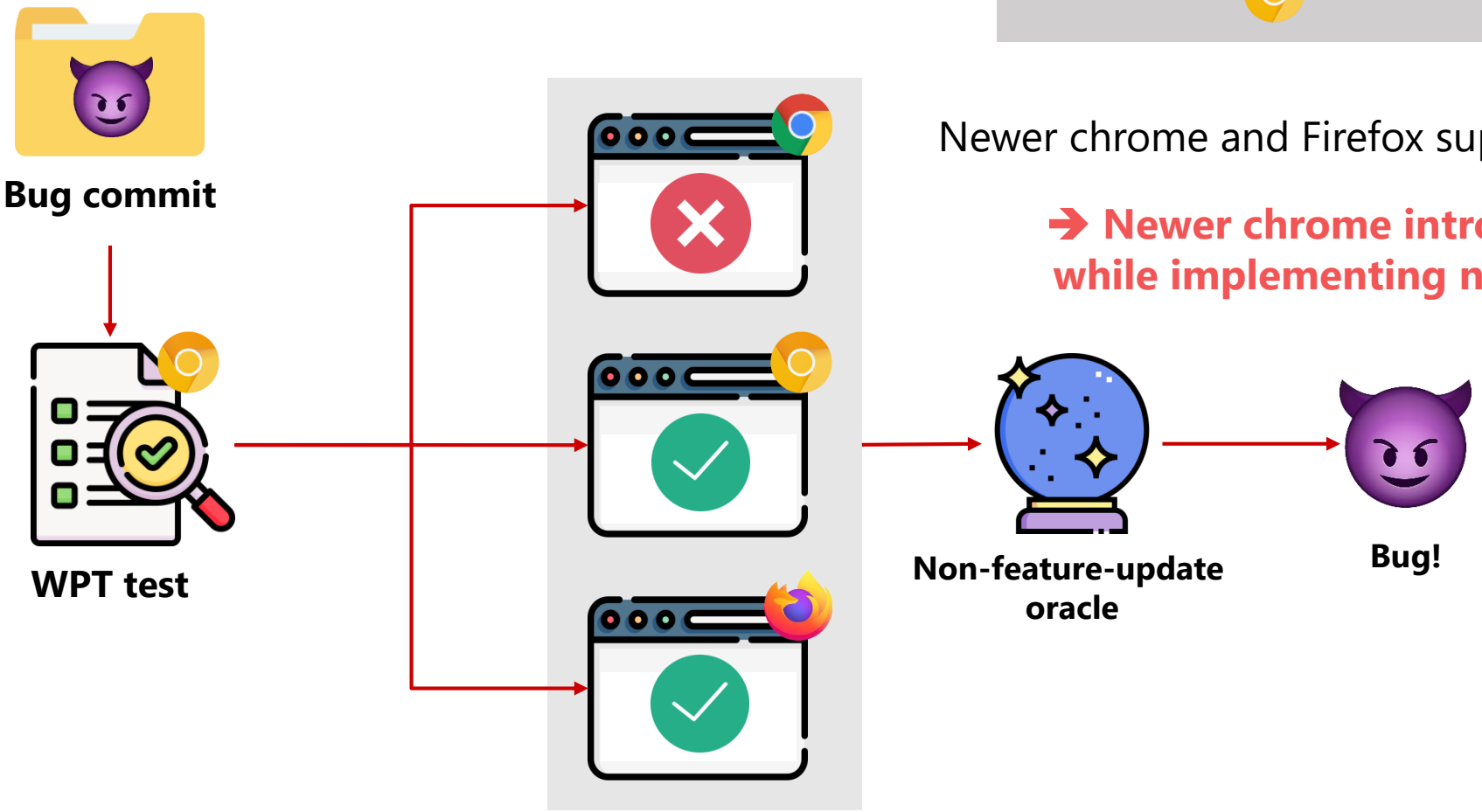


Only newer chrome passes WPT test

➔ **Firefox does not support this new feature**

Non-feature-update oracle

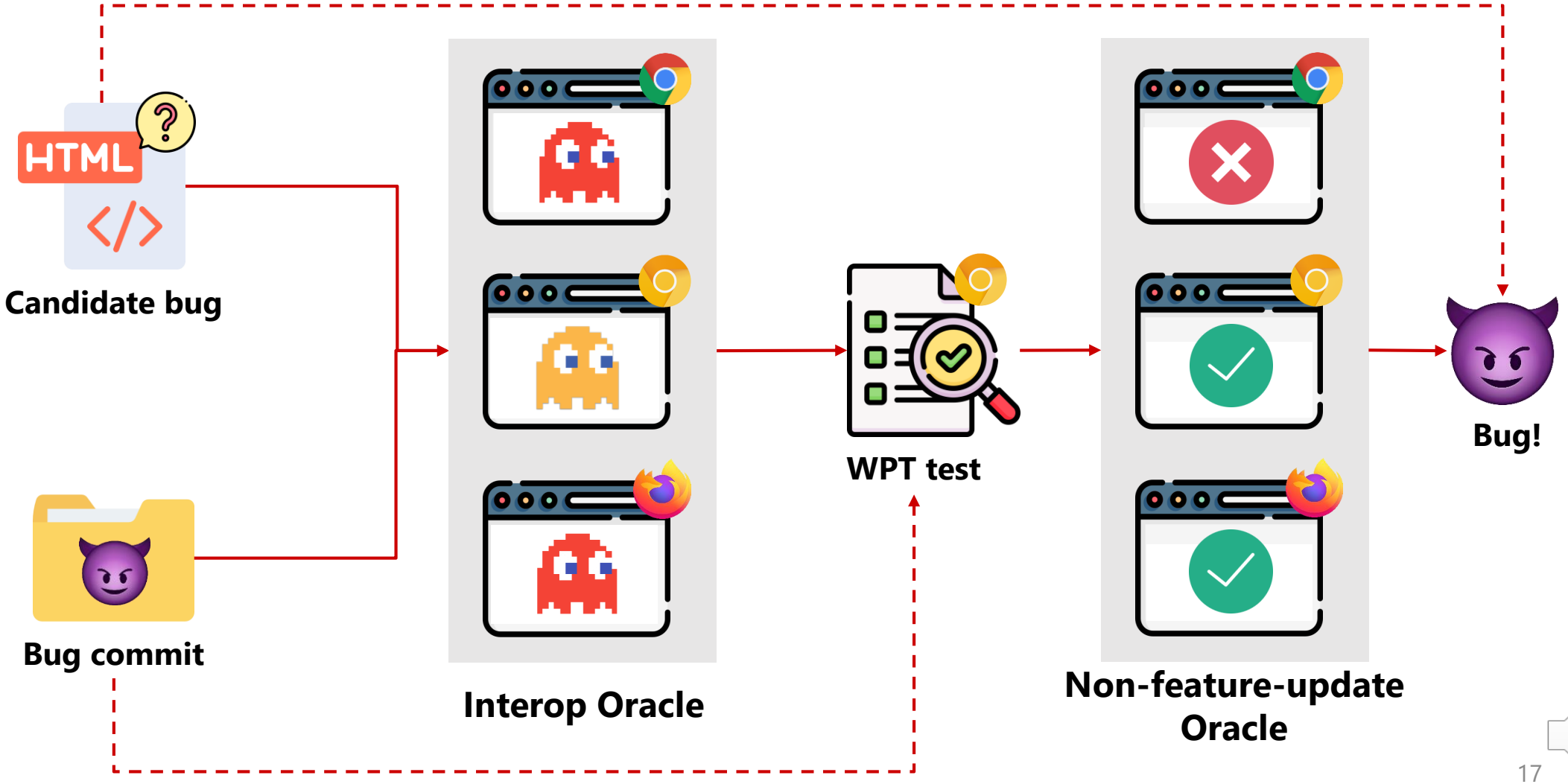




Newer chrome and Firefox support new feature

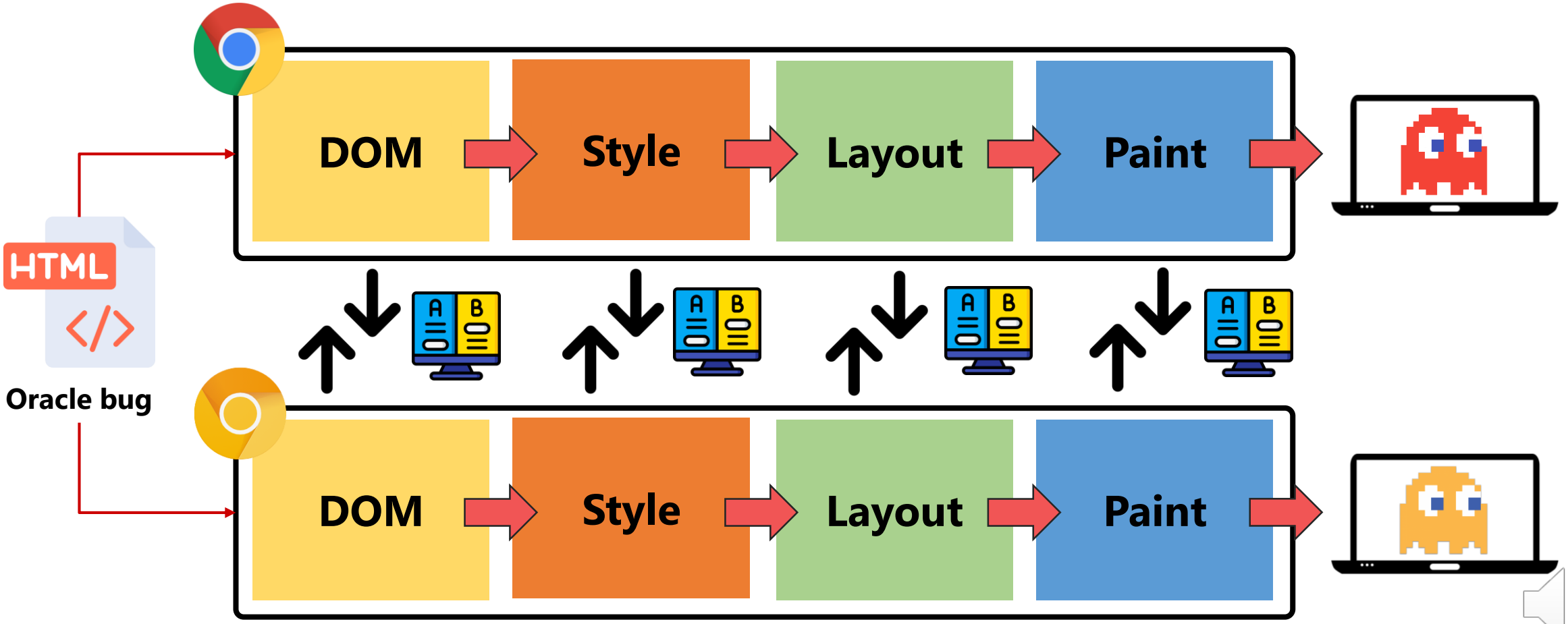
➔ **Newer chrome introduces bug while implementing new feature**

Workflow of rendering bug oracle



Rendering pipeline analysis

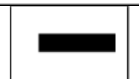
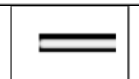
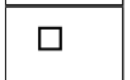
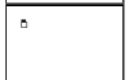
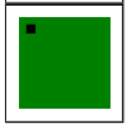
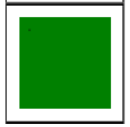
The stage that first introduces the differences is a bug stage!



New rendering bugs

We found **11** new rendering bugs and **six** of them were fixed.

Issue ID	Culprit Commit	Culprit Stage	Correct	Incorrect	Confirmed	Fixed
#1121082	775116 (✓)	Paint (✓)			✓	✓
#1164652	779663 (✓)	Layout (✓)			✓	
#1226558	780992 (✓)	Layout (✓)			✓	✓
#1231397	770064 (✓)	Paint (✓)			✓	
#1237352	885372 (✓)	Paint (✓)			✓	✓

Issue ID	Culprit Commit	Culprit Stage	Correct	Incorrect	Confirmed	Fixed
#1240854	885961 (✓)	Paint (✓)			✓	
#1240856	890916 (✓)	Layout (✓)			✓	✓
#1241345	889344 (✓)	Undecided			✓	
#1241356	888805 (✓)	Layout (✓)			✓	✓
#1241436	885635 (✓)	Paint (✓)			✓	✓
#1242851	887727 (✓)	Layout (✓)			✓	

Conclusion

- This paper proposed R2Z2, a differential fuzz testing technique to find rendering bugs with low false positive rate (i.e., ~20%).
- R2Z2 features the bisect analysis and the rendering pipeline analysis, allowing R2Z2 to spot the bug commit and pipeline stage.
- R2Z2 identified 11 new rendering bugs in Chrome.