Analysis of JavaScript Web Applications Using SAFE 2.0

Jihyeok Park
KAIST
jhpark0223@kaist.ac.kr

Yeonhee Ryou
KAIST
ryou770@kaist.ac.kr

Joonyoung Park
KAIST
gmb55@kaist.ac.kr

Sukyoung Ryu
KAIST
sryu.cs@kaist.ac.kr

Background

< JavaScript >
- Standard language for **web programming**
- Extremely **functional** & **dynamic**
- Often vulnerable to **programmer errors**

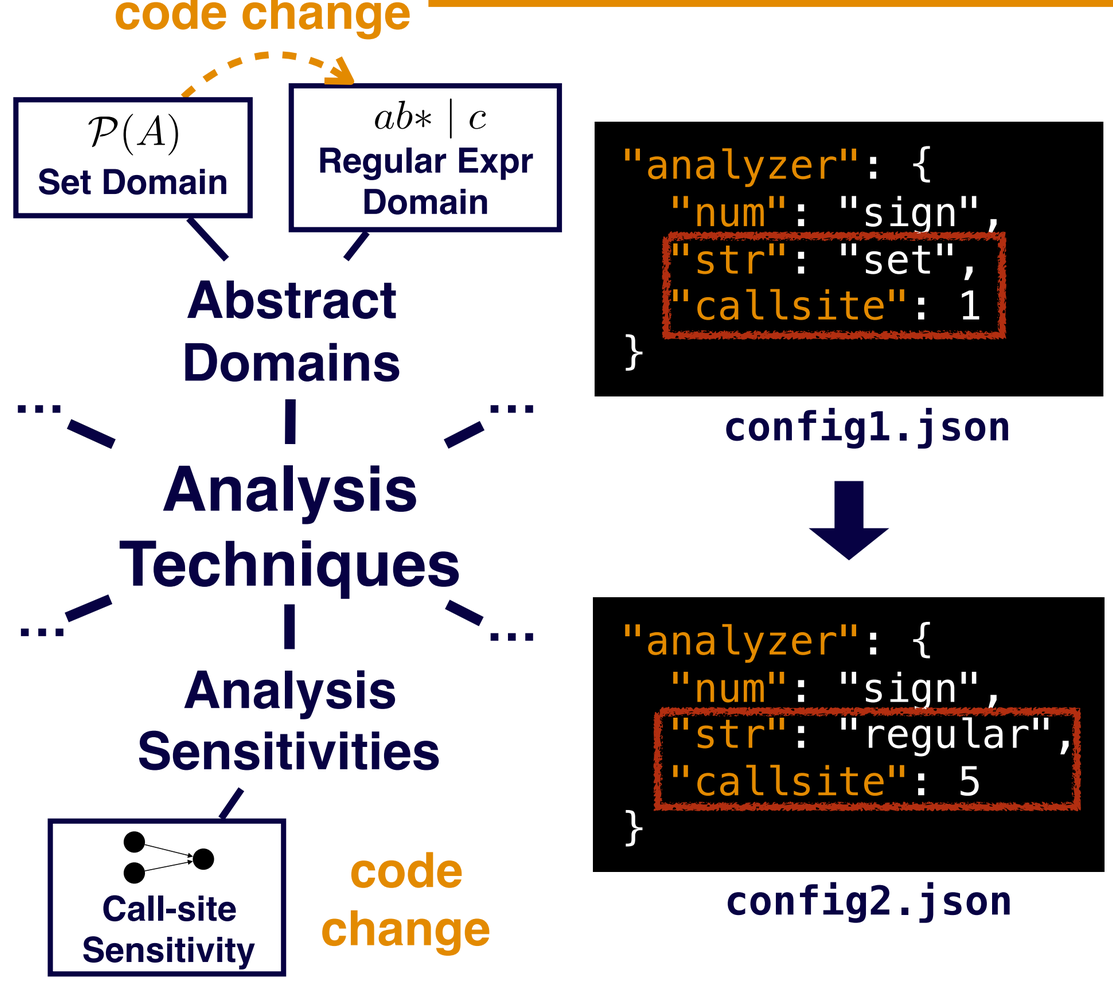
< Analysis Frameworks for JavaScript >
- SAFE / TAJS / WALA / ⋯
- Help **web-app developers** build **correct programs** by **detecting bugs**
- **BUT** poor **usability** for **analysis developers**

SAFE 2.0

A New Analysis Framework for JavaScript Web Applications

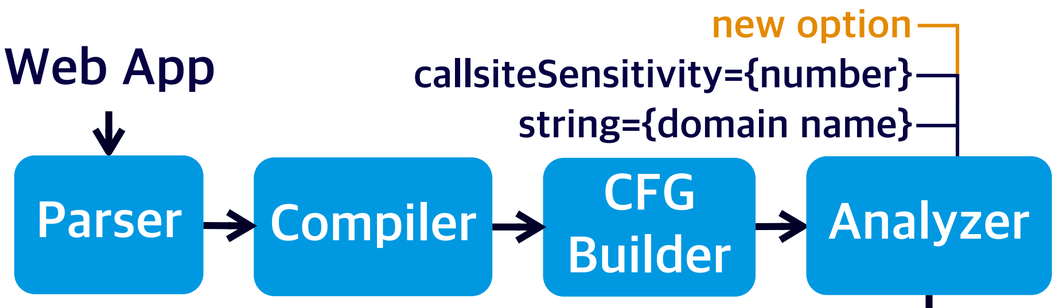
Pluggability

configure existing analysis techniques easily



Extensibility

add new techniques easily



<Commands>
- List of **phases** (with **options**)

<Example>
- **Bug Detector** - detect bugs **(new phase)**
- **Snapshot** - use dynamic info **(new option)**

<Testing using Test262>
- **Test262 :**
  The official ECMAScript conformance suite
- **Test** implementation of new techniques

```
> test262Test
[info] Run completed in 9 minutes, 32 seconds.
[info] Total number of tests run: 5552
[info] Suites: completed 1, aborted 0
[info] Tests: succeeded 5552, failed 0, canceled 0, ignored 0, pending 0
[info] All tests passed.
[success] Total time: 574 s, completed May 18, 2017 2:57:27 PM
```

Debuggability

debug analysis easily

<HTML Debugger>
- **Interactive debugging tool**
- **Instructions** for each node
- Control flow graph (**CFG**)
- Current abstract **state**