



SALT LAB

SOFTWARE ANALYSIS AND TESTING

Hybrid DOM-Sensitive Change Impact Analysis for JavaScript

Saba Alimadadi, Ali Mesbah and Karthik Pattabiraman

ECOOP 2015

saba@ece.ubc.ca

ece

Electrical and
Computer
Engineering



a place of mind

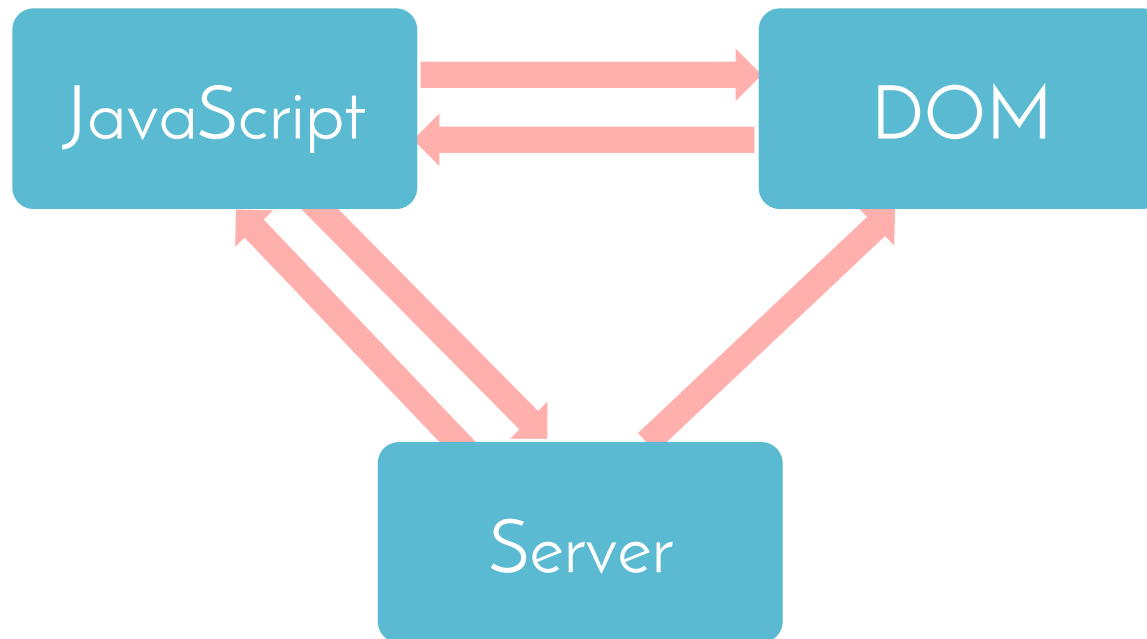
THE UNIVERSITY OF BRITISH COLUMBIA

Change Impact Analysis (CIA)

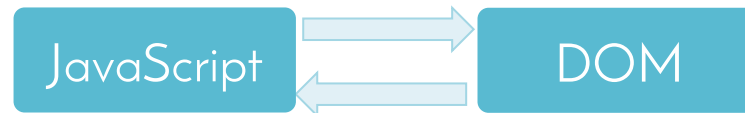
- Software must continually change to adapt to the changing environment.
- **Goal:** identifying parts of the program that are potentially affected by a change.



Challenges of CIA for JavaScript

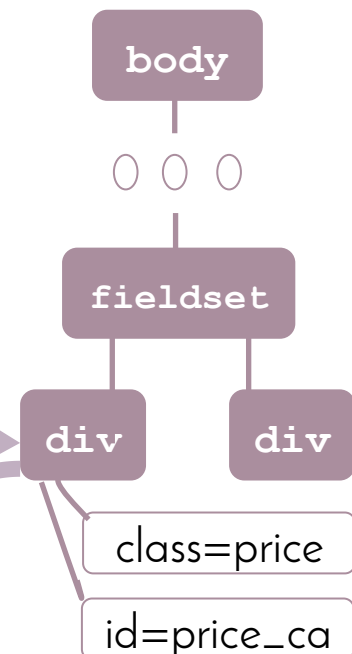


Challenge 1: Impact through the DOM

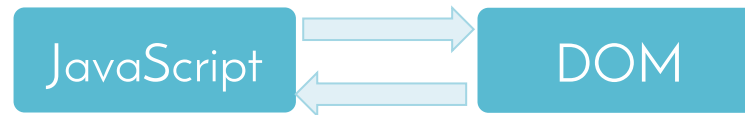


```
function calculateTax() {  
  $('price').each(function(index) {  
    $(this).text(addTaxToPrice(  
      $(this).text());  
    });  
}
```

```
function checkPrice() {  
  . . .  
  var cad_price = $('#price_ca').innerText();  
  . . .  
}
```



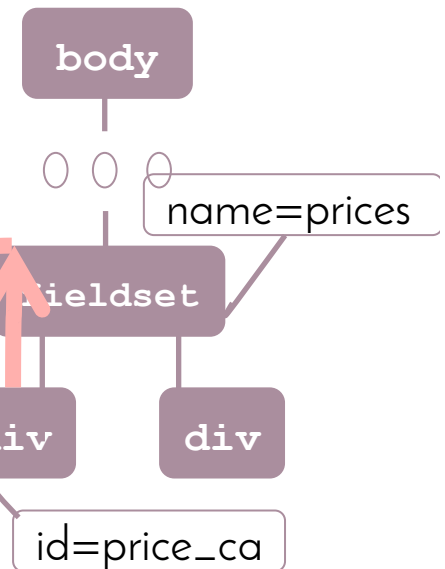
Challenge 2: Impact through Event Propagation



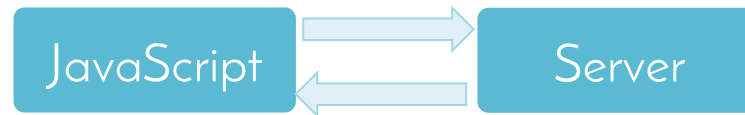
```
$('#price_ca').bind('click', checkPrice);  
$('#prices').bind('click', calculateTax);
```

```
function calculateTax() {  
    . . .  
}
```

```
function checkPrice() {  
    . . .  
}
```



Challenge 3: Impact through XHRs



```
function checkPrice() {  
  var itemName = extractName($('item231')  
  $.ajax({  
    url : 'prices/latest.php',  
    type : 'POST',  
    data : itemName,  
    success : eval(getAction() + 'item')  
  });  
}
```

XHR

```
function updateItem(xhr) {  
  var updatedInfo = getUpdatedPrice(xhr.responseText)  
  suggestItem.apply(this, updatedInfo,  
}
```

Challenges of CIA for Client-Side JavaScript

1. JavaScript and Document Object Model (DOM)
 2. Events and event propagation
 3. JavaScript and XMLHttpRequests (XHRs)
- + High dynamism of JavaScript

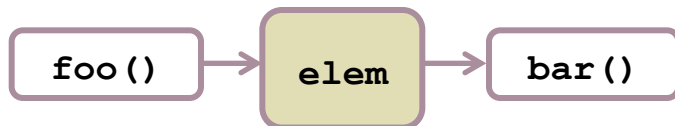
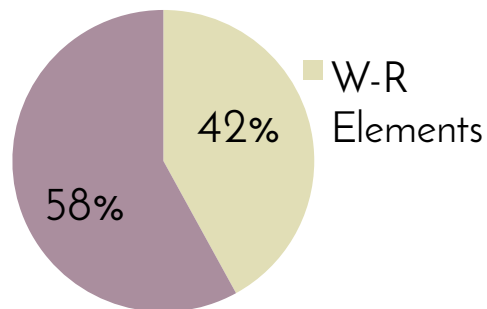
Exploratory Study: DOM-related and Event-based Impacts

- Subject: 10 web apps (online contests, GitHub trending, etc.)
- Gathered data:
 - JavaScript-DOM interactions (write-read pairs)
 - Event propagation
- Further analysis of the structure of graphs.
 - Measured metrics: fan-in and fan-out of functions and DOM elements, and average path lengths

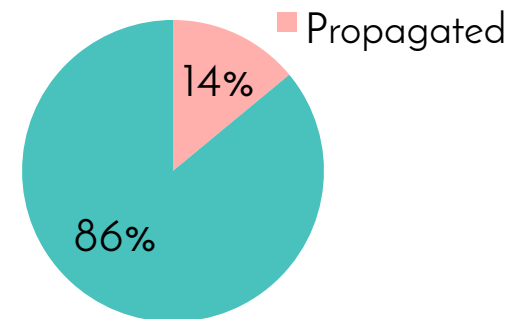
Exploratory Study: Results

- W-R DOM elements: 42%
- Propagated handlers: 14%

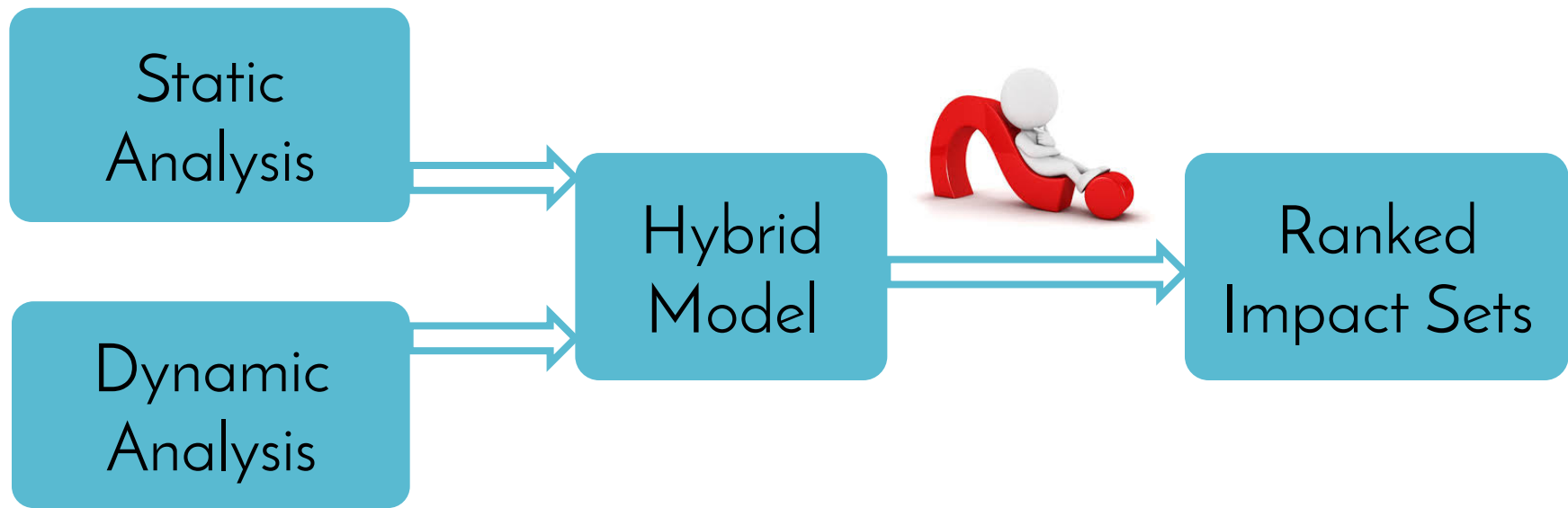
DOM Elements



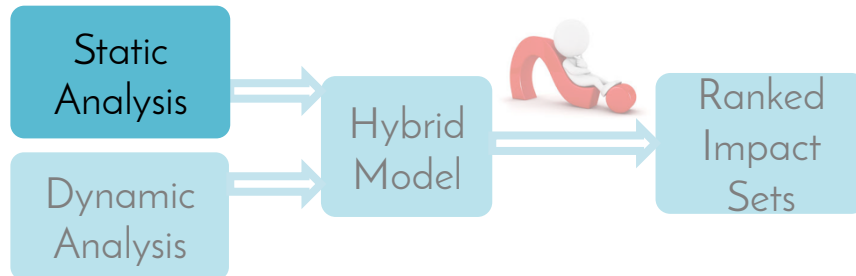
Event Handlers



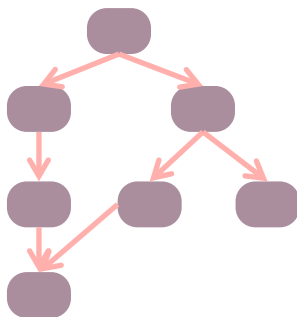
Hybrid Analysis



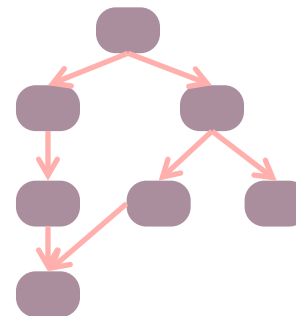
Static Analysis



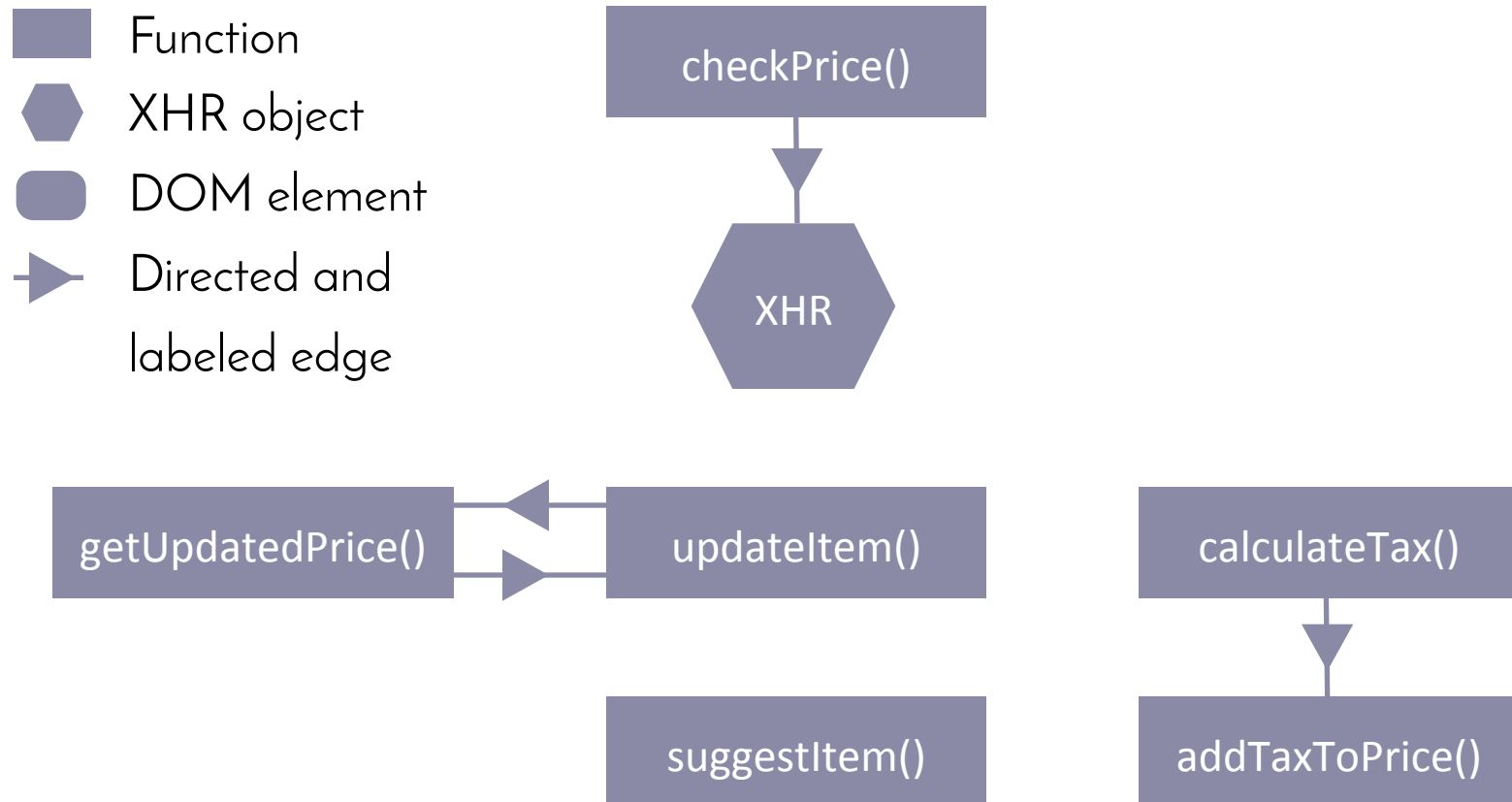
Control (and data)
dependencies



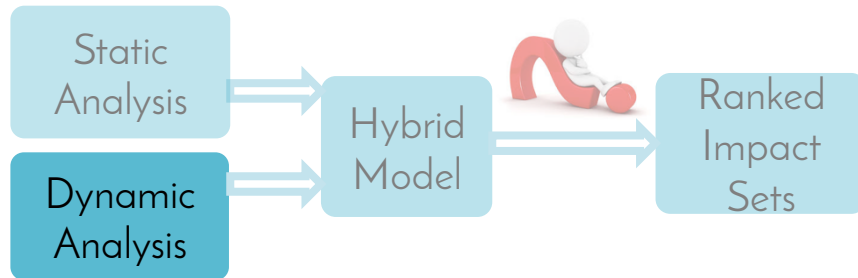
Partial data-flow
analysis



Example: Static Model



Dynamic Analysis



Impact through DOM



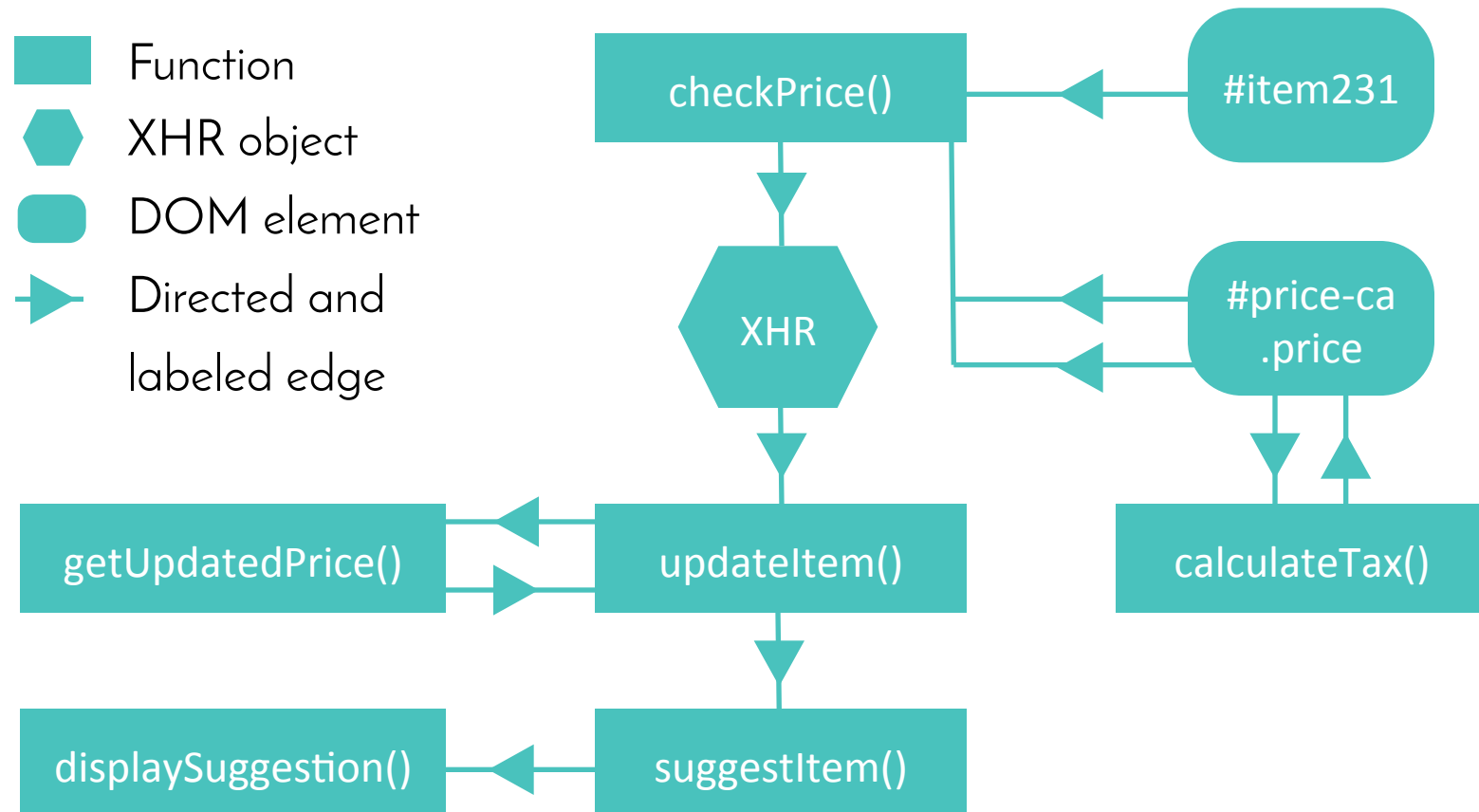
Impact through events

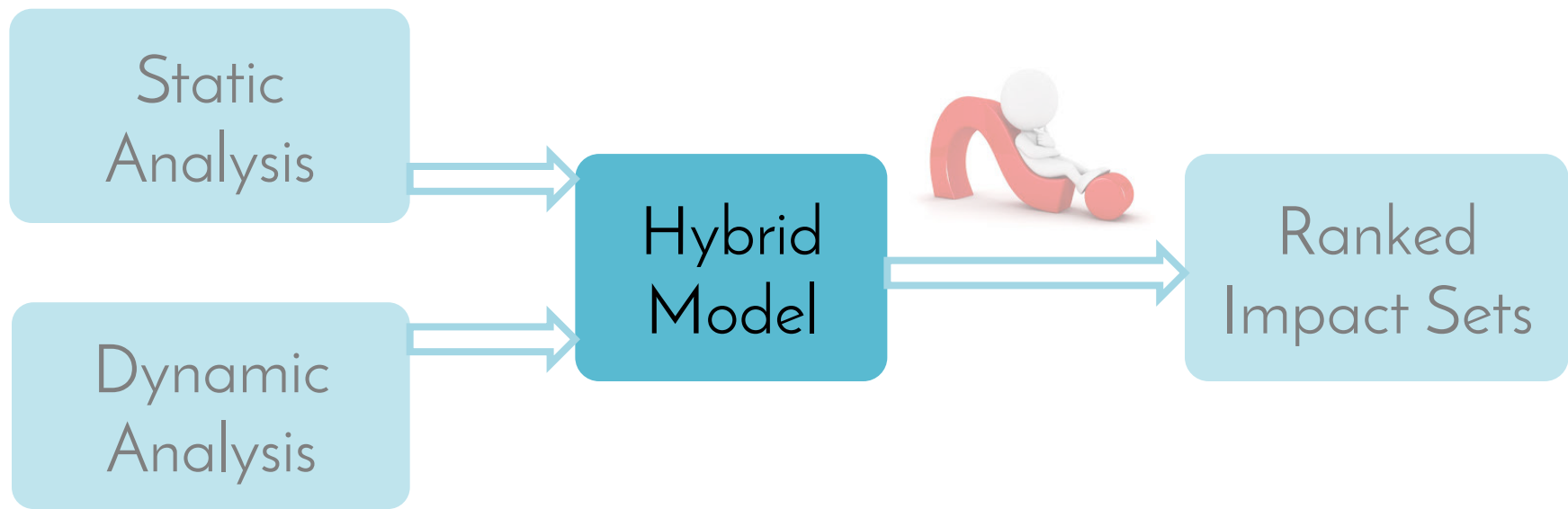
Dynamic call graph

XHR relations
(open, send, response)

JavaScript dynamism
(eval(), function variadicity)

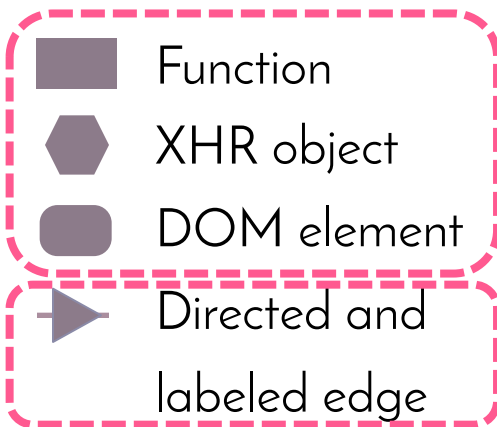
Example: Dynamic Model



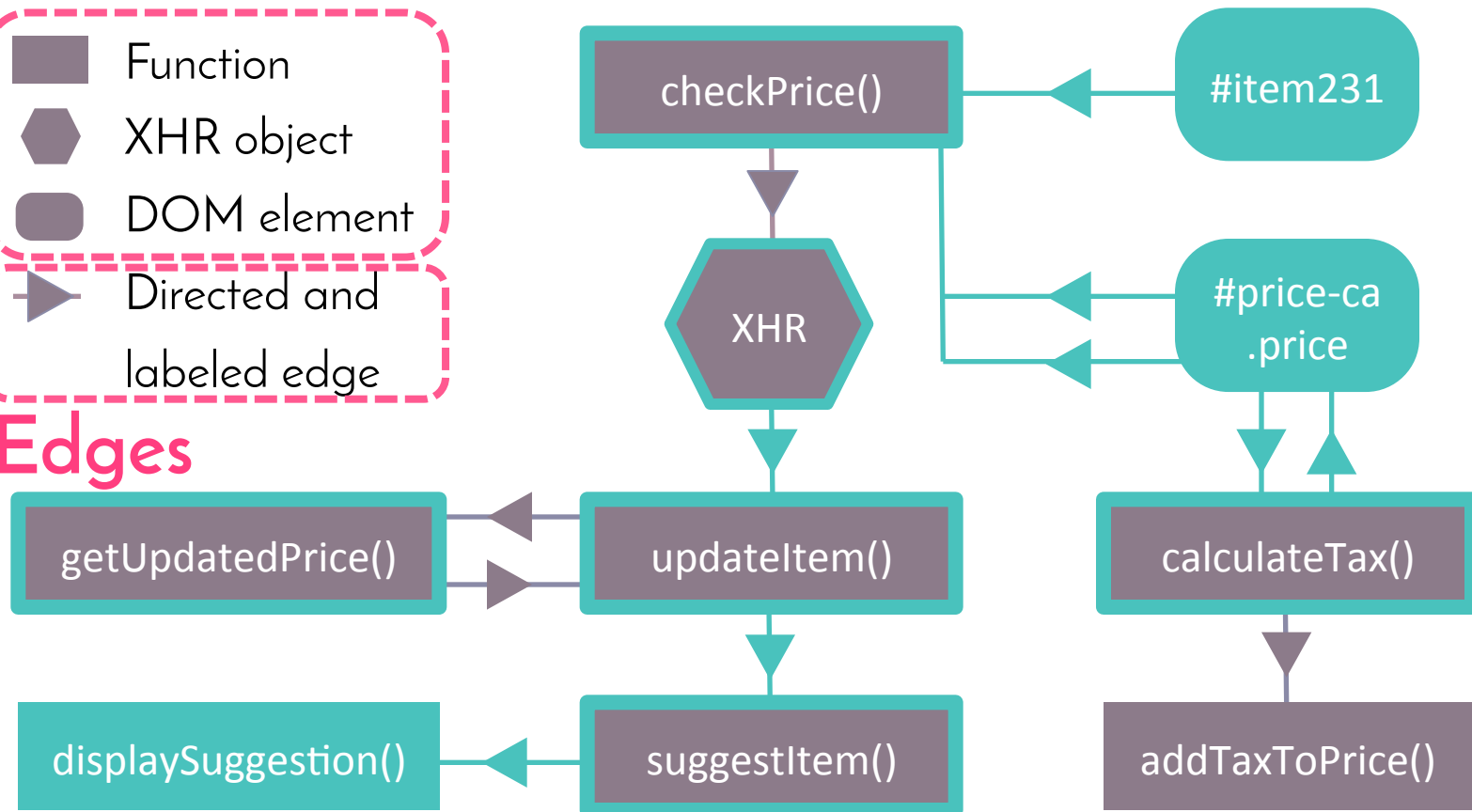


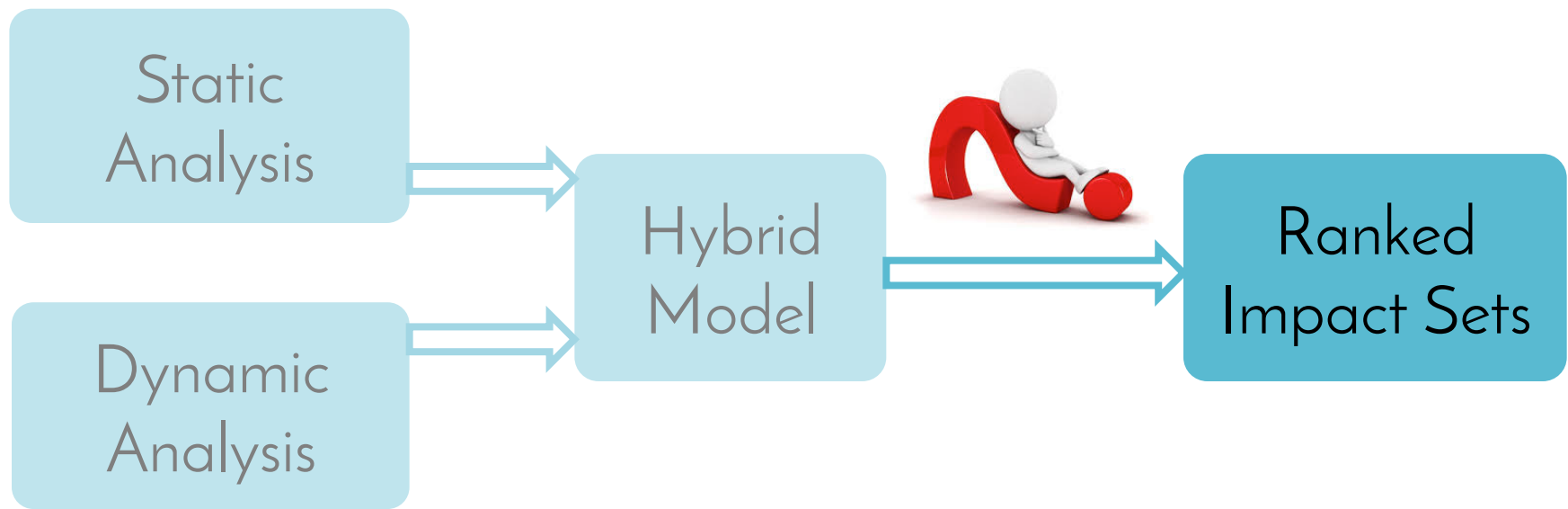
Example: Hybrid Analysis

Vertices



Edges





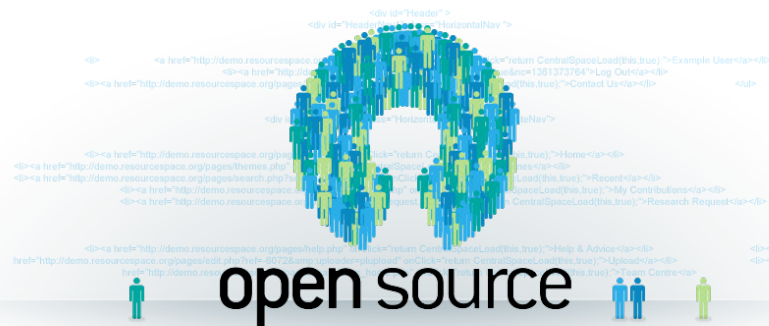
Impact Metrics and Impact Set Ranking

- Problem: size of impact sets
- Solutions: impact ranks, based on impact metrics
 - $f_{in}(d)$: Number of functions f such that $f W d$
 - $f_{in}(f)$: Number of elements d such that $f R d$
 - $f_{out}(f)$: Number of elements d such that $f W d$
 - $L_{avg}(P)$: Average length of impact paths in the app
 - $D_m(e)$: Minimum distance of e from the change set
 - $IR_{pr}(e)$: Impact of previous entity



Tool Implementation: Tochal

- Tochal: open source
 - <https://github.com/saltlab/tochal>
- Proxy (Java, JavaScript)
 - Esprima, Estraverse, Escodegen, Mutation Summary, WALA
- Client-side (Google Chrome extension)
 - Chrome DevTools



Research Question 1

Does Tochal outperform static and dynamic analysis methods in terms of the completeness of the results obtained?



Study: Static vs. Dynamic vs. Tochal

- 10 web applications
- 3 random functions as change sets
- Comparing:
 - Size of impact sets
 - Number of functions in dependency graphs

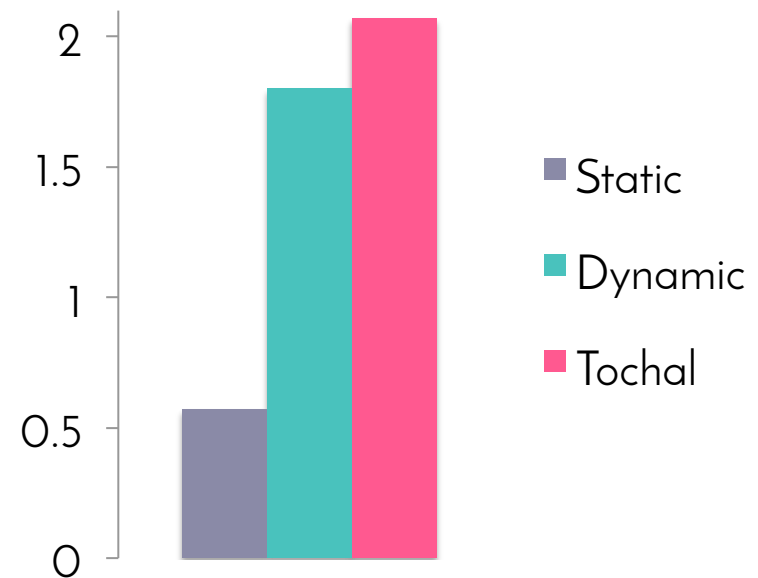


Results: Impact Sets

- Comparing size of impact sets

$\frac{\text{Static}}{\text{Hybrid}}$: **26%**

$\frac{\text{Dynamic}}{\text{Hybrid}}$: **80%**

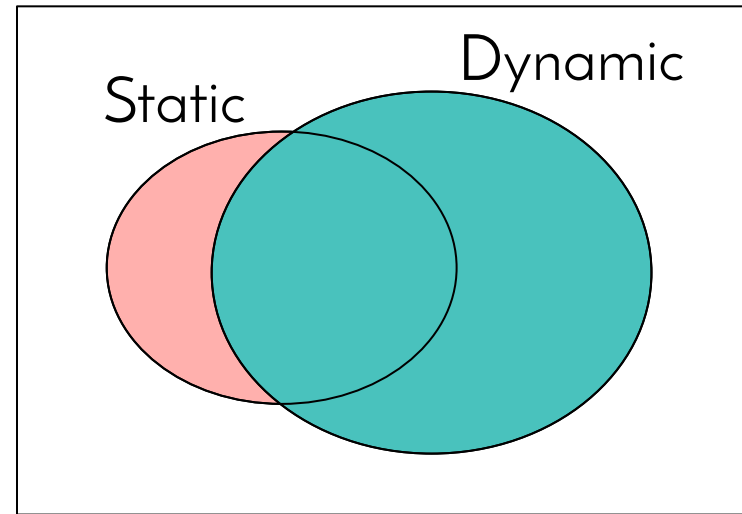


Results: Graphs

- Comparing size of model graphs

$\frac{\text{Static}}{\text{Hybrid}} : 59\%$

$\frac{\text{Dynamic}}{\text{Hybrid}} : 84\%$

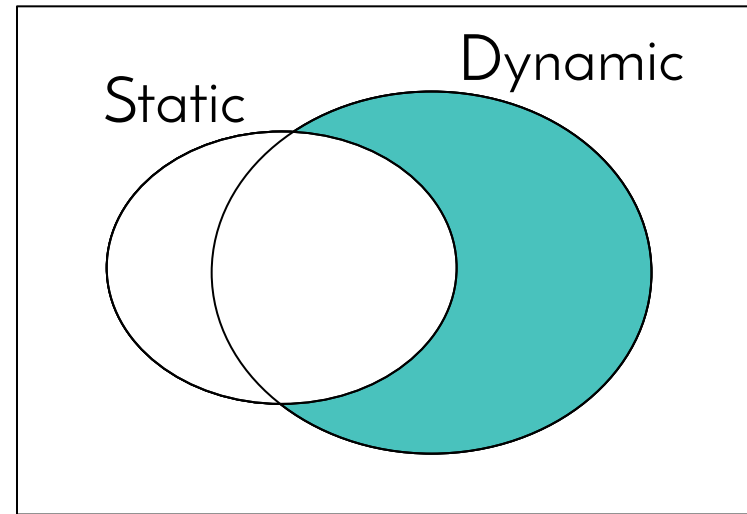


Results: Graphs

- Comparing size of model graphs

$\frac{\text{Pure Static}}{\text{Hybrid}} : 15\%$

$\frac{\text{Pure Dynamic}}{\text{Hybrid}} : 42\%$



Research Question 2

Does Tochal help developers in practice to perform change impact analysis?



Experiment: Design

- 12 participants from industry

- Performed 4 tasks

Task	Description
T1	Finding the potential impact of a DOM element
T2	Finding the potential impact of a JavaScript function
T3	Finding a conflict after making a new change (<u>no ranking</u>)
T4	Finding a bug in JavaScript code

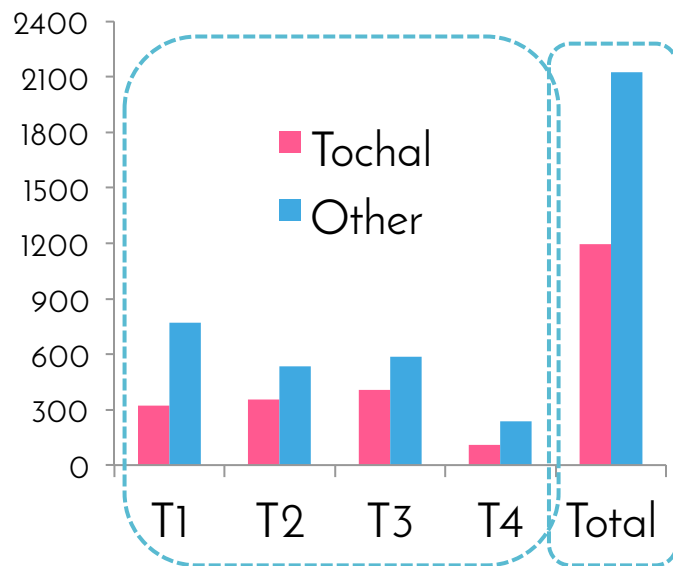
- We measured task completion duration and accuracy



User Experiment: Results



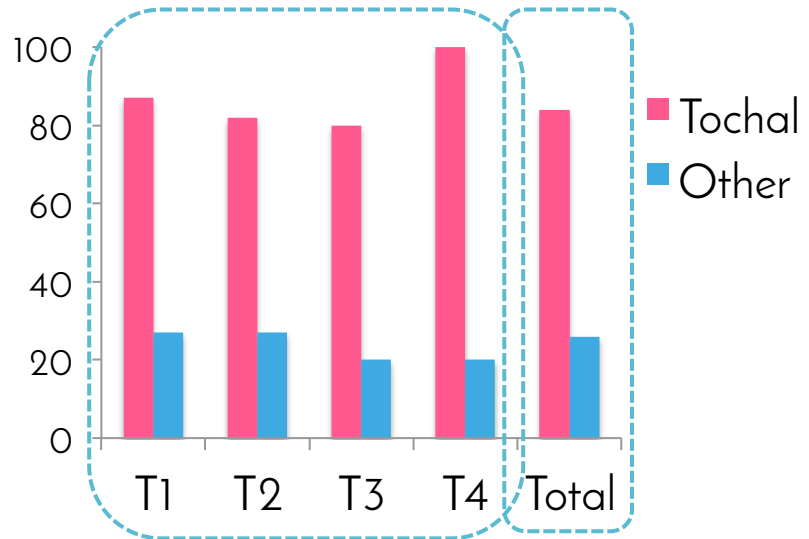
Duration



80% faster



Accuracy

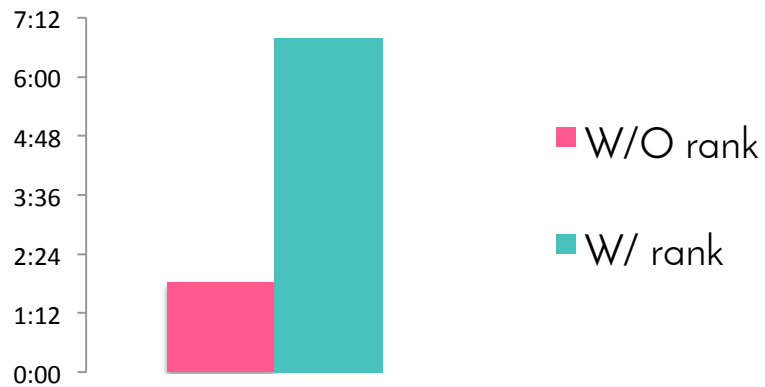


2 times more accurate

Results: Ranking



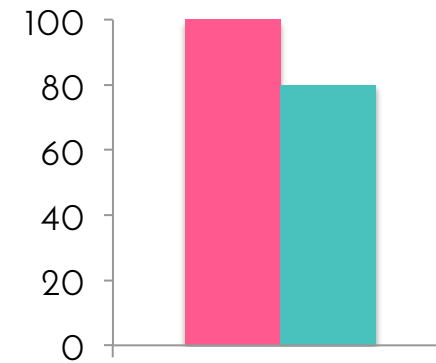
Duration



2~3 times faster



Accuracy



25% more accurate

Challenges of CIA for Client-Side JavaScript

1. JavaScript and Document Object Model (DOM)
 2. Events and event propagation
 3. JavaScript and XMLHttpRequests (XHRs)
- + High dynamism of JavaScript

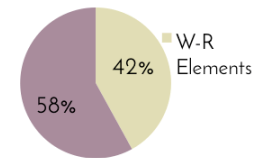


6

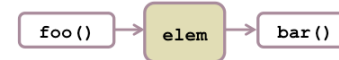
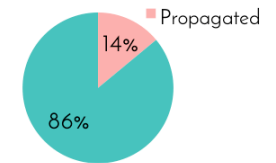
Exploratory Study: Results

- W-R DOM elements: 42%
- Propagated handlers: 14%

DOM Elements

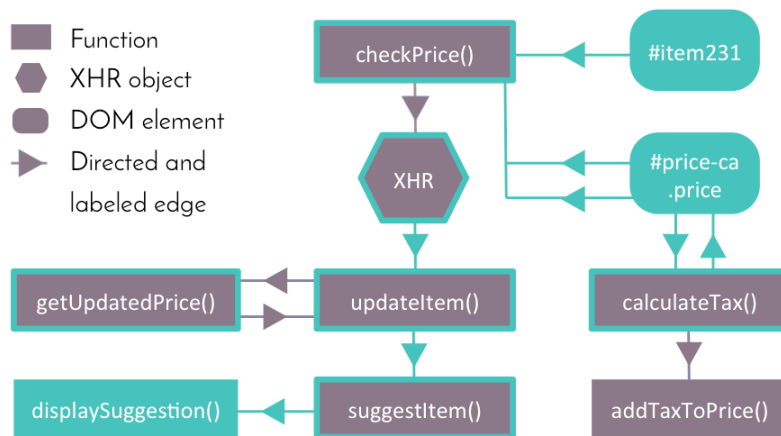


Event Handlers



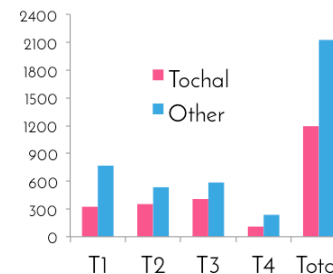
8

Example: Hybrid Analysis

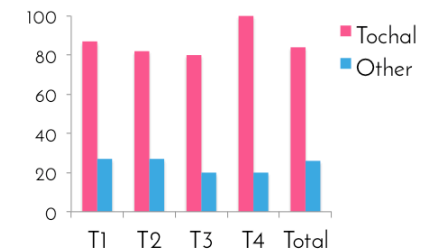


15

User Experiment: Results



2 times more accurate



80% faster



26