

Programming Languages and Formal Models

Anders Møller



Olivier Danvy
assoc. prof.



Klaus Ostermann
assoc. prof.



Michael Schwartzbach
professor



Anders Møller
assoc. prof.



Erik Ernst
assoc. prof.



Sebastian Erdweg
PhD student



Michael Achenbach
PhD student



Jakob G. Thomsen
PhD student



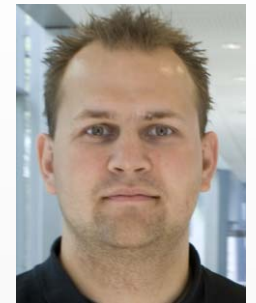
Johnni Winther
PhD student



Mathias Schwarz
PhD student



Simon Holm Jensen
PhD student



Anders Bach Nielsen
PhD student



Karl Klose
PhD student



Tillmann Rendel
PhD student



Christian Hofer
PhD student



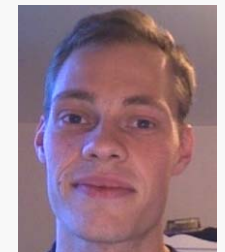
Casper W. Schmidt
stud. programmer



Asger Feldthaus
stud. programmer



Michael Dahl
stud. programmer



Thomas B.L. Jespersen
stud. programmer



Jan Midtgaard
post doc.



Ian Zerny
PhD student

An example of a current research project:

**static program analysis
for
JavaScript**

by Simon Holm Jensen, Peter Thiemann, and Anders Møller,
to appear at *the 16th International Static Analysis Symposium, SAS'09*

JavaScript: the lingua franca of Web 2.0

YAHOO! LOCAL Maps Welcome, erickfuture [Sign Out, My Account] [Maps Home](#) [Dial-Up Map \(Original\)](#) [Help](#)

GET MAP AND DIRECTIONS Clear
Address, City, State

FIND A BUSINESS ON THE MAP Clear
sushi **Search**

Search Results: sushi [Refine](#)

- 1 Taro Sushi** ★★★★★ (2)
(718) 398-0872
446 Dean St, Brooklyn, NY
[Directions To - From](#) | [More Info](#)
- 2 Sushi Garden**
(718) 222-8311
165 Joralemon St, Brooklyn, NY
[Directions To - From](#) | [More Info](#)
- 3 Sushi-Grill-Music-Russian Party-Black Vip Car**
(800) 424-5211
Brooklyn, NY
[Directions To - From](#) | [More Info](#)
- 4 Sushi D**
(718) 858-8058
207 De Kalb Ave, Brooklyn, NY
[Directions To - From](#) | [More Info](#)
- 5 Koodo Sushi**
(212) 425-2890
129 Front St, New York, NY
[Directions To - From](#) | [More Info](#)
- 6 One Greene Sushi Japanese** ★★★★★ (5)
(718) 422-1000
1 Greene Ave, Brooklyn, NY
[Directions To - From](#) | [More Info](#)
- 7 Faan** ★★★★★ (2)

Results 1-10 of 16 [Next](#) [View Local Results as a List](#)

Displaying 10 stories.

facebook Profile edit Friends

Search

Applications edit
Photos Groups Events Super Wall [more](#)

View Photos of Me (112)
View My Friends (103)
Get more Super Wall pics
Play Scrabulous with me
Edit My Profile

I am online now.

Lancaster Friends
37 friends at Lancaster. [See All](#)

Naomi Simmons
Corinth Dutton
Sana Intesar Siddiqui

Mark
Laura

April 5
Jenny commented on Helen Isley's photo. 10:29pm
"so you were how old when you gave birth to them?"

Jenny and David Threlkeld are now friends. 8:03pm

April 4
Jenny wrote on Helen Isley's wall. 8:19pm

April 3
Jenny is at home. 8:42pm
Jenny joined the group B&Q Drones. 8:39pm

March 28
Jenny wrote on Naomi Simmons's wall. 11:49pm

Gmail - Inbox

[http://gmail.google.c](#)

ahansen@gmail.com | [Feedback](#) | [Contacts](#) | [Se](#)

Search Mail

[Show search options](#) [Create a filter](#)

Compose Mail

Inbox
[Starred](#) [Sent Mail](#)
[All Mail](#)
[Spam](#)
[Trash](#)

Labels
[gmail stuff](#) [Edit labels](#)

Archive More actions... Refresh

Select: All, Read, Unread, Starred, Unstarred, None

<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	me, Ask, Robert (8)	» gmail test
<input type="checkbox"/>	<input checked="" type="checkbox"/>	Gmail Team	» Re: [#9130
<input type="checkbox"/>	<input checked="" type="checkbox"/>	Ask Bjørn Hansen	» test foo bar
<input type="checkbox"/>	<input checked="" type="checkbox"/>	Ask, me (2)	» pgp signed
<input type="checkbox"/>	<input checked="" type="checkbox"/>	Robert, me (2)	Re: Postfix
<input type="checkbox"/>	<input checked="" type="checkbox"/>	Gmail Team	» gmail stuff G

Select: All, Read, Unread, Starred, Unstarred, None

Archive More actions...

You are currently using 0 MB (0%) of your 1000 MB.
Shortcuts: o-open y-archive c-compose j-older k-newer

JavaScript is a *dynamic language*

- Object-based
- Prototype-based inheritance
- First-class functions, closures
- Runtime types
- ...

NO STATIC TYPE CHECKING

How JavaScript was designed

“I hacked the JS prototype in ~1 week. And it showed! Mistakes were frozen early.”



– Brendan Eich, inventor of JavaScript

An example

```
var x = {  
  a: 42,  
  b: function() {return a;}  
}
```

```
print(x.b())
```

uncaught JavaScript
runtime exception:
ReferenceError: "a"
is not defined.

```
var x = {  
  a: 42,  
  b: function() {return this.a;}  
}
```

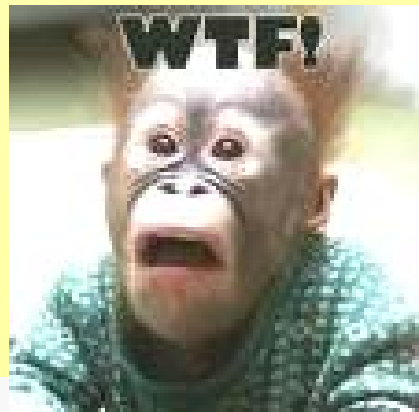
```
print(x.b())
```

42

Another example

```
function A() {}  
A.prototype.count = 0;  
A.prototype.inc = function() {A.prototype.count++;}  
function B() {}  
B.prototype = new A; // now B is a "subclass" of A
```

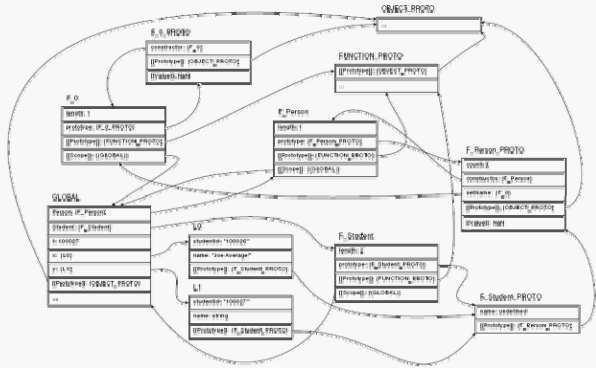
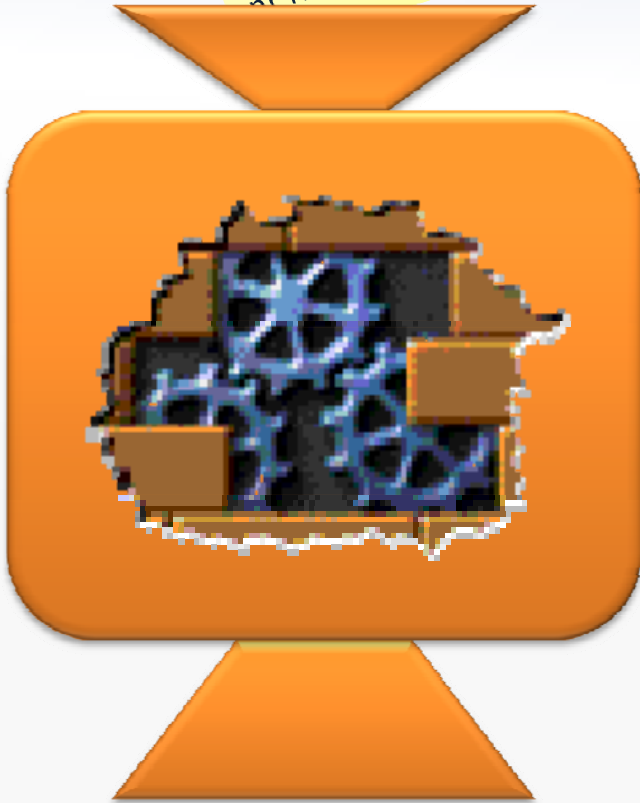
```
var x = new B;  
var y = new B;  
print(x.count); 0  
x.inc();  
x.inc();  
print(x.count); 2  
print(y.count); 2  
  
x.count++;  
print(x.count); 3  
print(y.count); 2
```



Our approach

- Abstract interpretation using “the monotone framework”
- The recipe:
 1. construct a **control flow graph** for the program to be analyzed
 2. define an appropriate **dataflow lattice** (abstraction of data)
 3. define **transfer functions** (abstraction of operations)

```
var x = {
  a: 42,
  b: function() {return a;}
}
print(x.b())
```



an abstract heap
for each program point

Preliminary results

- A conservative static analysis for the full JavaScript language (ECMA-262 3rd ed.)
- Useful analysis **precision**
 - can detect many kinds of likely programming errors
- Acceptable **performance**
 - but room for improvement 😊