

# DOM Transactions for Testing JavaScript

Phillip Heidegger,

Annette Bieniusa, Peter Thiemann

University of Freiburg

2010/09/05

# Task: Maintenance

- Finding bugs in JavaScript programs
- Understanding JavaScript programs

# JavaScript – Features

Some important features:

- Weak, dynamic typing
- Object-based language (no classes, but prototypes)
- Functions are first class values
- Side effects
- Automatic conversion of values
- `eval` expression
- `with` statement

# Example

```
1 var s = "A nice string";
2 var x = 8;
3 function f(o) {
4   with (o) {
5     return x;
6   };
7 };
8 s.x = 5;
9 f(s)
```

# Example

```
1 var s = "A nice string";
2 var x = 8;
3 function f(o) {
4   with (o) {
5     return x;
6   };
7 };
8 s.x = 5;
9 f(s)
```

→ 5 or 8?

# Example

```
1 var s = "A nice string";
2 var x = 8;
3 function f(o) {
4   with (o) {
5     return x;
6   };
7 };
8 s.x = 5;
9 f(s)
```

```
1 var s = "A nice string";
2 var x = 8;
3 function f(o) {
4   if (hasProperty(o,x)) {
5     return o.x
6   } else {
7     return x;
8   }
9 };
10 s.x = 5;
11 f(s)
```

→ 5 or 8?

# Example

```
1 var s = "A nice string";  
2 var x = 8;  
3 function f(o) {  
4   return o.x;  
5 };  
6 s.x = 5;  
7 f(s)
```

→ ...

# Unit Testing vs. Side Effects

## Unit Testing:

- Isolated test cases
- Simple test case specification
- Modular testing of the system



# Unit Testing vs. Side Effects

## Unit Testing:

- Isolated test cases
- Simple test case specification
- Modular testing of the system

## Side Effects:



# AJAX Code

Typical AJAX Code manipulate:

- Add/Delete HTML nodes
- Hide/Show parts of the webpage
- Add/Delete event listeners

→ AJAX Code performs a lot of side effects

# Unit Testing of AJAX Code

## **Transactions** on HTML Pages

- Begin the transaction if the test case starts
- Check the result environment is as expected
- Reverts all changes after the test case

# Implementation

- Integrated into **JSConTest**
    - Contracts for Specification
    - Random Testing
    - Guided Random Testing
  - Log based transactions
  - Transformation of JS Code
  - Browser independent
- `setUp` and `tearDown` rarely necessary

# Speed?

Slowdown:  $\approx 4.3$

# Speed?

Slowdown:  $\approx 4.3$

→ much better than restart the browser

# Conclusion

## JSConTest

- Unit Testing Framework
- Transactions for functions with side effects
- Contract as Specification
- Browser independent
- `http://proglang.informatik.uni-freiburg.de/JSConTest/`