

Universiteit Utrecht

[Faculty of Science Information and Computing Sciences]

Functional Instrumentation of ActionScript Programs Arie Middelkoop, Alexander Elyasov, Jurriaan Hage,

Wishnu Prastya

Department of Information and Computing Sciences, Universiteit Utrecht J.Hage@uu.nl

October 4, 2011

FITTEST project

- EU financed FP7 project, 4 mln euros
- Future Internet
 - highly dynamic
 - extremely complex
 - autonomic, and self-modifying
 - very configurable
- Needs continuous testing
- Testing is log-based





Universiteit Utrecht

[Faculty of Science Information and Computing Sciences]

(日)

Case study: Habbo Hotel

- Social Networking site for teenagers
- Developed by Sulake Oy, Finland
- > 230 mln people registered, 10 mln unique users per month
- Habbo is very dynamic, also at a technical level
- Currently built in Flash/Flex
- Testing the system is very hard and costly
- The Habbo client is large:
 - 3,000 classes, 25,000 methods and 800,000 instructions



Universiteit Utrecht

[Faculty of Science Information and Computing Sciences]

*ロト * 得 * * ミト * ミト ・ ミー ・ の へ ()

Cross-cutting concerns

Logging is a cross-cutting concern

- scattered functionality is spread out
- tangled in between other functionality
- Aspect-orientation helps implement the logging functionality unintrusively



Universiteit Utrecht

Aspect-oriented programming

Support for weaving in advice at particular point cuts, a selection of join points.

- Each AOP technology chooses
 - What are the join points?
 - Method call, entry, exit and return, block entry and exit, coercion call, return and failure
 - What is advice like?
 - Typically (source) code that can be woven in directly
 - How can point cuts be specified?
 - Typically some pattern language
- Advice (code) is woven in by an aspect weaver



Universiteit Utrecht

[Faculty of Science Information and Computing Sciences]

・ロト・日本・日本・日本・日本・日本

Upon matching

- There is direct access to static information
- and symbolically to dynamic/run-time information
- For example, if you match on a method name
 - static information: the name, number of arguments
 - dynamic information: types and values of arguments, depth of the call stack
- If you match on a method name pattern, then the name is also dynamic
 - All methods that contain "move" in their name



Universiteit Utrecht

Our contributions

Development of a

- Instrumentation language Asil for Flash
- Binary weaver Asic, which also computes reflection information,
- expressed as a deeply embedded DSL in Haskell,
- treat aspects as first-class citizens
 - Instrumentations can be easily composed, through monadic and alternative combinators
- Side effect: debugged the ActionScript specification



Universiteit Utrecht

[Faculty of Science Information and Computing Sciences]

・ロト・日本・日本・日本・日本・日本

Alternatives

- Change the Flash run-time: not allowed or outside our control
- Change the source code: not all code is available or accessible
- Manual intervention is too costly and very dangerous
 - Sulake already considers binary weaving to be dangerous
- Note: even with binary instrumentation certain privacy issues remain



Universiteit Utrecht

An overview of the pipeline





Universiteit Utrecht

[Faculty of Science Information and Computing Sciences]

◆□ → ◆□ → ◆ 三 → ◆ 三 → ○ へ ()

Manually instrumented code

. . . .

```
public function MyGame() : void {
  addEventListener("click", clicked); }
function clicked(event:MouseEvent) : void {
  var x : int = event.localX;
  var y : int = event.localY;
  Log.clicked(x,y);
  . . . .
  if (!this.selSquare && taken) {
    this.selSquare = target;
  } else if (this.selSquare && !taken) {
    Log.move(this.selSquare, target);
    this.move(this.selSquare, target);
    this.selSquare = null; } } }
                                              Faculty of Science
```



Universiteit Utrecht

Information and Computing Sciences]

*ロ * * 母 * * 目 * * 目 * * の < や

Calling Log.clicked(x,y) with an aspect

 $instrLogClick = \mathbf{do}$ $m \leftarrow matchEnter \ k_code_MyGame_clicked$ $let \ evt = param_1 \ m$ $eX = evt \ \# \ k_flash_events_MouseEvent_localX$ $eY = evt \ \# \ k_flash_events_MouseEvent_localY$ $call \ (static \ \# \ k_code_Log_clicked) \ (eX, eY)$ $return \ ()$

matchEnter matches one method name, almost statically:

- Match is usually resolved statically
 - if the name does not match, we statically know not to weave
 - if the name matches, we may have to check dynamically for it being in the right class
- m contains information about the match, e.g., argument



values ^{Universiteit Utrecht}

[Faculty of Science Information and Computing Sciences]

*ロ * * 母 * * 目 * * 目 * * の < や

Instrumenting this.move

- Only for this.move called within the method clicked!
 - By virtue of *onPrevious*



Universiteit Utrecht

Combining instrumentations

```
 \begin{array}{l} myInstr :: I \ () \\ myInstr = foldr \ (\otimes) \ (fail "initial") \\ [instrLogClick, instrLogMove] \end{array}
```

- \blacktriangleright \otimes is sequential composition
- Various alternative combinators can be used to orchstrate instrumentations
- Because general Haskell code would be hard to translate we partially evaluate monadic code and functions away
- This gives us AsilCore code which can be mapped to AVM2 byte code



Universiteit Utrecht

[Faculty of Science Information and Computing Sciences]

・ロト・日本・日本・日本・日本・日本

The weaver, Asic

- Takes AsilCore and applies it to all join points in the SUT
- Optimisation: weave time evaluation of primitive functions
 - non-primitives may have side effects
- Asic is implemented with Attribute Grammar technology
 - Courtesy of Doaitse Swierstra and many others



Universiteit Utrecht

[Faculty of Science Information and Computing Sciences]

Our experiences

- Haskell provides Asil's abstraction facilities for free
- Syntactic sugar courtesy of GADTs and type classes
 - Wasn't always easy
- Flash/ActionScript specifications available, but incomplete and wrong
 - Employed Swierstra's uuparsing-lib for error-correction and discovering bugs
- Performance: GC takes a lot of time.
 - AST is long-lived and large: preferably consign to older memory generation
- GHC crashes on symbol file generated for Habbo



Universiteit Utrecht

[Faculty of Science Information and Computing Sciences]

*ロ * * 母 * * 目 * * 目 * * の < や

Conclusions and Future Work

- Asil is an AOP for binary instrumentation of ActionScript
- Asil is heavily inspired by the functional programming paradigm, particularly higher-order functions
- Asil/Asic await serious experimentation
 - so very much under development and subject to change
- Currently: combining built-in support for event logging with Asil providing the necessary hooks.



Universiteit Utrecht