OCamlPro

OCaml Workshop 2012

Promoting OCaml
in the Industry

Fabrice LE FESSANT

http://www.ocamlpro.com/

OCamlPro

- Spin-off of INRIA

- Founded on April 2011

- Located on the "Campus de Saclay",
  in the "Ecole Polytechnique", near Paris

- Member of the Caml Consortium since 2011

- Work funded by a few sponsors/customers

  - Jane Street, Lexifi, Esterel Tech, Citrix

  - … we are looking for more !

Mission Statement

OCamlPro promotes the use of the OCaml functionnal programming language in the industry, as a solution to increase development productivity and software quality.

OCamlPro provides all the services and tools to help companies use OCaml to develop their products, with a particular emphasis on accessibility and perenity.

The Team

- A team of experts in OCaml development:
  - Thomas Gazagnaire (CTO, R&D)
  - Çagdas Bozman (PhD student)
  - Pierre Chambart (R&D)
  - Vincent Bernardoff & Guillem Rieu (short mission)
- In close collaboration with an INRIA group:
  - Frédéric Tuong (DORM project)
  - Michael Laporte (intern)
  - Luca Saiu (post-doc)
  - Fabrice Le Fessant (founder, consultant)

OCamlPro Services

- Training of developers
  - Courses for beginners, and for experts (the type system, writing efficient code, the GC, etc.)
- Consulting by OCaml experts
  - Profiling and Optimizing code
  - Providing OCaml specific solutions
- Development in OCaml
  - Generic open-source libraries and tools
  - Specific components for customer

OCamlPro Products & Tools

- Development tools
  - TypeRex, development studio
  - Build Management tools (opam, ocp-build)
  - Memory Profiling tools
- OCaml 4 Windows
  - Easy to install and to use
  - Better integration
- OCaml with Long Term Support
  - Hot fixes, fast release cycle
  - Customer specific extensions

Serving the Community

- OCaml Cheat Sheets
    - www.ocamlpro.com/blog/2011/06/03/cheatsheets.html
- Try-OCaml and Try-js_of_ocaml
    - OCaml tutorial in a browser... even disconnected !
    - http://try.ocamlpro.com/ (4.00.0)
- OPAM package manager
    - http://opam.ocamlpro.com/
- Contribution to the ocamlweb project
    - http://www.ocaml-lang.org/ (→ github)

Contributions to 4.00.0

- New option -bin-annot for ocamlc/ocamlopt

    - Save the typedtree in .cmt files

        → auxiliary tools (dead code detect, ocaml-spotter)

- New options for ocamldep (-all, -sort)

    ocamlopt -o my_program `ocamldep -sort *.ml *.mli`

- CFI directives in x86/amd64 backends

        → correct backtraces in debugger ('bt' in gdb)

- Misc optimizations (partial funs, pipe operators)

    external (|>) : 'a -> ('a -> 'b) -> 'b = "%revapply"

    external (@@) : ('a -> 'b) -> 'a -> 'b = "%apply"

TypeRex

- Towards a Development Studio for OCaml
- v1 released in Feb 2012
  - Emacs, Tuareg-mode, 3.12.1
  - color, refactoring, go-to-definition, auto-completion
  - monolithic + needs its own compiler
- working now on v2
  - Emacs/VI, tuareg-mode / caml-mode, 4.00.0
  - less powerful, but better accessibility
  - indep. tools + use -bin-annot of 4.00.0

TypeRex
Project Manager

```
begin library "mylib"
    link = ["-cclib"; "-lxycomp"]
    files = [ "x.ml" "y.ml" "stubs.c"]
    requires = [ "unix" ]
end
```

```
begin library "sndlib"
    files = [ "lex.mll" "a.ml"]
    requires = [ "mylib"]
    sort = true
end
```

```
begin program "myprog"
    files = [ "cc.ml" (asmcomp = [ "-inline"; "30"])
             "main.ml"]
    requires = [ "sndlib"]
end
```

Simple DSL to describe packages and dependencies in multiple dirs, enabling parallel incremental build (ocp-build)

What's next ?

- Big functors: functors packing compilation units

- Big blocks: more than 16MB on 32 bits

- Namespaces Proposal

- Frame pointers for Linux perf tools

- Builtins for binary operations (bswap)

- What do you want ?

  → become a sponsor or customer of OCamlPro !

Contact

- Email: contact@ocamlpro.com

- Web: http://www.ocamlpro.com/

We are hiring !

- We are looking for OCaml experts for:
  - Post-docs
  - R&D Engineers

  (positions at INRIA or OCamlPro, Saclay)

- With experience in compilation, typing, static analysis, etc.

- And different systems (Unix, Windows)