18 Jul 2024 | OCaml Windows WG

Attendees: pierre.boutillier@laposte.net benjamin@canou.fr Sudha Parimala David Allsopp Nicolas Ojeda Bar, Manas, Loic, Antonin

Agenda

- ocaml-windows org on GitHub, papercuts as central issue tracker: https://github.com/ocaml-windows/papercuts/issues
- Opam 2.2 is out!
- Debugging on Windows
- Profiling
- Winget manifests with portal-executables for opam: https://github.com/ocaml/opam/issues/6095
- Language survey

Notes

- Create all windows related issues on papercuts. Any issue can be opened and dispatched.
- Opam 2.2 is out!: Oldest opam issue closed!
    - It should be possible to install lsp-server and platform tools without needing lots of hacks. If something's not working it's an issue.
    - Pierre: if we launch vscode from start menu, lsp-server can't find the switch
    - Lsp-server: install lsp server recompiles ocaml. 5.2 compatible version is not released yet.
    - Manas: started to use opam 2.2. Trying to reproduce issues on Windows. Run an online meetup
    - Loic: Tried and no problem so far. Using an existing install of msys2 with msvc.
    - Sudha: Also started using it. Going to check status of platform tools
- Debugging
    - What's the LexiFi story?
    - Nicolas: We use printf :)
    - David: been able to debug the runtime using visual studio - it's disassembly of OCaml code.
    - Benjamin: We started to study what other languages to do, there are interesting topics to add. One would be how other languages do relocatability and debugging.
    - Manas: We need better dwarf 5 support. DWARF dumps from flambda2 were very rich, on Linux, same couldn't be done on Mac.
    - Work going on to upstream DWARF flambda2 work. When flambda2 arrives DWARF support will come in soon. The patches were complicated and have invasive changes. There are tools to convert DWARF to PDB.
    - WinDBG hasn't adopted dwarf yet? VS still uses PDB
    - There's no arm64 support for dwarf on flamda2
    - We probably need to wait for flambda2
    - Haskell has interesting Windows debugging support

    - Experience F# provides could be the goal.
    - We should look at what rust does - they're the cool kids, and have llvm.
    - Languages get msvc support for free by opting into llvm.
    - 5.3 will have clang support
    - It's rare that you hit a problem where it's hard to solve with print. Ocaml-debugger is useful to debug types. Once the tool is there, more people will use it. LexiFI published landmarks for profiling.
    - Nicolas: In general OCaml is a very high-level language, profilers and debuggers are geared towards lower-level uses. These tools are less useful for these kind of problems. For really tricky bugs, they're typically logical bugs where you would observe the bug in deployment, debugger may not be very useful. Not against the idea of debugger, but may not be as useful for regular programming.
    - Manas: trying to debug a statically linked binary, a systems programming problem - a different libc implementation. Could not reproduce the issue. Stepping through gdb, it showed wrong linking. It helps when you hit a wall.
    - Gdb in cygwin is usable
    - Tim and Miod are working on stock OCaml debugging.
    - Statmemprof works, no other profiling
- The issue is solved with 2.2 release. It installs in a place that's unexpected. Jonah mentioned that's because it uses DkML installer. Raised a PR to the winget repo to use the new binary type.
    - Is it a good time to create a manifest under ocaml?
- Language survey
    - Sdl2 bindings need heavy patching before it's usable.
    - Ctypes hacks
    - Is it hard to write the demonstrator in C? Was able to install and write in one hour.
- Compiler developments
    - Relocatable OCaml expected to land soon. For Windows it'd improve the experience of creating second switch.
    - Cross-compilation: lots of talks are going on, timeline unclear. Maybe prototypes in 2025.

Action items

- [ ]

10 Jun 2024 | OCaml Windows WG kick-off

Attendees: David Allsopp KC Sivaramakrishnan Sudha Parimala Vincent Balat Anirudh S Raja Boujbel, Manas, lwagner, nojb, oandrieu, Benjamin Canou, Pierre Boutillier, Kento

Agenda

- Welcome! Introductions
- Opam 2.2
- Debugging
- User survey
- Survey of other languages
- Scope? - Deeper integration with Windows package managers like scoop, chocolatey

Notes

- Introductions
    - Sudha: Tarides, lead of the OCaml on Windows project. Got motivated to work on this at OCaml workshop at IndiaFOSS
    - Vincent: Engineering manager at Tarides
    - Pierre: Worked at Nomadic. Worked on cross-compilation.
    - Benjamin: Worked at Nomadic. Plan was to try and make games in OCaml and launch a studio. Tried to build executables for small examples, but didn't manage to.
    - KC: Teaches OCaml at IIT Madras. Similar issues faced at IndiaFOSS.
    - Raja: OCamlPro, maintain opam since 6 years, recently worked with David and Kate to release opam 2.2
    - David: using OCaml on Windows for 20 years! Past
    - Nicolas: works at LexiFi, one of the earliest industry users of OCaml. Uses Windows applications since then.
    - Anirudh: Beginner to OCaml, Elixir developer. Alternate b/w Windows, Ubuntu and macOS. Ran into difficulties during the OCaml workshop.
    - Manas: prometheansacrifice on GitHub. Maintainer of esy package manager. Previously member of ligo team. Freelance programmer now. Onivim builds on Windows. Esy as a tool has been working well on Windows, not necessarily the best tool right now.
    - Kento: works at University of Vienna - mediate and support meeting staff. Have been maintaining this markup language for scientific hypertext.
    - Olivier: work for Ansys, been shipping a product for Windows for 15 years. It's a compiler for a programming language. Previously using OCaml compiler without deps, recently started using third party libraries, with Windows it has become more difficult
    - Loic: works at Ansys with Olivier
- David: beta3 will be released soon. Uncompromising Windows support, focusing on correctness rather than speed. On any Win11 system: winget install opam, opam init simply works!
    - Opam-repository being updated
    - Manas: low hanging fruits: can we get a list?
    - The major problem is extracting files. Reading repo by extracting a tar-ball. Msys2 is not a first-class citizen (no official support).
    - Manas: compiler build time improvement?
    - Pierre: testing opam betas, troubles were as soon as we touch bindings and external dependencies. It will be the job of each individual package to deal with external dependencies? It's a long term problem, have you thought about how to tackle that?

    - David: there is a plan, the main PR on opam-repo merged last week. That's a green signal for any package to go through it. Whole depext. Longer term. Opam-repo-ci have a Windows job.
- Debugging: been in touch with Tim, related to MacOS. Windbg, dwarf how complete is the support?
- Survey of other languages:
    - Devices a methodology - centred on two main pain points: setup environment and install packages. How do they deal with external deps?
    - Started from a fresh VM install, there's a free Windows install available for edge testing.
    - Demo code - first a hello world. Then code that downloads OCaml from GitHub and shows via SDL. Requires some png library and an SSL stack. It's short but also telling. Experimented with Go and Python. Both of them easy to install. Python getting an exe is not straightforward. Didn't manage to get SDL to work with Go. They don't have depext system, they have a nice ctypes like system. Easily ship Go code with external C, but installing is a manual process. On Linux it was really easy.
    - Manas: Was it msvc or gnu libc?
    - Benjamin: Tried both. C Go internal tools does not work with msvc, required mingw. README tells to install msys and to put it on a specific PATH. It did not work. Was not able to see which PATH var was long. Tried to build openSSL and deps with vcpkg.
    - Python: super easy to install, all dev tools worked well. Their SSL library is a bit broken, they don't recognise system certificates. SDL worked very easily. In the native package manager, they have DLLs. In Python you don't link, you'll dynamically at the launch time load with ctypes.
    - Next try C++, how do people deal with bindings?
- Winget: fetches from windows app store. A single command to have opam up and running.
    - OCaml compiler is not relocatable, makes it harder to have a prebuilt compiler.
    - Longer term plan is to add opam to scoop, chocolatey, etc.
- Prebuilt compiler
    - Getting started with OCaml takes time. DkML did not install. It is a necessity to have everything built from source. We need relocatable compiler. Experimenting with prebuilt Windows compiler.
        - What can we do about relocatable compiler
        - Making binary relocatable by replacing the prefix - esy does this.
        - Benjamin: when you install Go with prebuilt binaries there's no choice of location.
        - https://icfp22.sigplan.org/details/ocaml-2022-papers/12
        - Relocatable compiler PR will be up by the time 5.3 is branched and available in 5.4. Having a toolchain for user still sounds interesting.
        - Esy has had some success building on CI.
        - Opam-bin and ocp-win from OCamlPro

Updates (from Manas)

- Prebuilt Windows compiler

Communication channels

- OCaml discuss – https://discuss.ocaml.org/
- OCaml discord – https://discord.gg/Rn6nY5ca
- OCaml Windows mailing list – https://groups.google.com/g/ocaml-windows-wg

Action items

- [ ]