

Polyglot *BSD

Porting and Packaging programming languages

How to get new language implementations building

How to communicate with upstream to get any needed fixes/additions in

How to be a good BSD representative in language communities

How to write a good port

Maintaining language ports into the future

Difficulties with all of these

Rewards of all these

What this talk is not

A programming tutorial. We just don't have the time to talk about the nitty gritty syntax and semantics of each language, plus I don't think that's as useful to us tonight as the call to get involved in porting languages.

About me

bcallah@openbsd.org

Been doing mostly ports since
2013

According to Repology.org, 160
ports

Wide variety of stuff

#PlayOnBSD

Faculty in ITWS@RPI

Been around NYC*BUG for a
while

You may have heard my other
NYC*BUG talks

My never-ending quest

- I want to have a package of every (open source) language implementation there is!
- **The big takeaway:** It's as much for their benefit as it is ours.
 - They get more users.
 - We get more users.
 - And it legitimately fun and exciting to get a new programming language up and running!

Agenda

What is your favorite programming language?

Out of the box

What is supported today?

Highlights of my experiences with some language communities

Looking ahead, how you can help



What is your favorite
programming language?

Lots to choose from!

What made you choose that language?

Best tool for the job?

Familiarity?

Involved in language development?

How good is the platform support for that language?

Some, this is easy: C runs everywhere

Others, less clear

Others still, might have a single platform of support!

What does it mean to have language support?

- Is it:
 - Able to compile a compiler/interpreter?
 - Having a package of a compiler/interpreter?
 - Having full upstream knowledge and support?
 - Active involvement in the language community?
 - Historical connection to a language?
 - *Something else?*

Out of the Box

What languages do you get?

On a default install of OpenBSD...

- C, C++
- Assembly
- Perl
- Shell (Bourne/Korn/C)
- Awk
- (Probably others I've forgotten)

What more could you possibly need? :)

What is supported today?

Augmenting through packages!

Can you believe it???

It turns out that not everyone is satisfied with those out of the box languages.

Good thing we have ports then!

A (potentially infinite) treasure trove of really interesting programming languages to explore.

What the BSDs support, today

Since 2017, I have maintained a running list of all the languages I know about that work on at least one *BSD:

<http://daemonforums.org/showthread.php?t=10478>

I'm sure it's incomplete since I only update as I'm made aware of new languages in ports or I port them myself.

Compilers I've (helped) port to OpenBSD

- Tiny C Compiler
- a60 (Algol 60)
- Algol68g
- Seed7
- GNU APL
- GNU Cim (Simula)
- CParser+LibFIRM
- Nvidia Flang
- LLVM Flang
- Lily
- voc (Oberon)
- FreeBASIC
- GNU Pascal
- Kona (K)
- PC-BASIC
- Pony
- QuickJS
- Regina REXX
- GNU D Compiler
- Digital Mars D Compiler
- Cyclone (Scheme)
- LLVM D Compiler
- GNU Modula-2
- Cowgol
- MIT/GNU Scheme
- CSNOBOL4
- lacc (C)
- Gravity
- Emojicode
- Hobbes
- J
- And more...



Let's talk about some of them

And the lessons I learned



Seed7

My first language port

Seed7

- One man operation
 - Based off his dissertation work

Gains

It was one of my first ports, and it helped me learn about potential intricacies of porting languages and how to work with upstream language providers to incorporate OpenBSD patches



Nvidia Flang

A bridge for alternative Fortran support

Nvidia Flang

Fortran compiler, based off of PGI Fortran

Requires CLA to participate

Uses the LLVM backend, but not all LLVM platforms are supported...

Gain

I wanted a Fortran compiler for arm64, and at the time, gfortran was not working (this has been since remedied).

Served its purpose, but probably want to retire in favor of new LLVM Flang



tcc

Sometimes your hardware is as good as your skills!

tcc

Was on my radar for a number of years, but there were always blockers (aka other things I was working on).

In late 2020, I noticed that BSD support became a priority for the tcc devs.

They were **so** close...

Gains

Another C compiler, sure

People actually use tcc, and have thanked me for this port

Additional platform support (armv7, arm64) for tcc



How to represent the BSDs well



Inexplicably popular blog post
Had been on my radar for years
but like tcc, always had
something else to work on
Some members of the D
community began to prioritize
OpenBSD support
I swooped in and finished the
very last mile (and not well...)

Gains

D has a very dedicated
community. We are on their
radar now!
A particularly good case study
for us on how to represent the
BSDs well in a language
community



GNU Modula-2

Why do I keep ending up on Hacker News?

GNU Modula-2

Another small team

Was supposed to be incorporated into gcc something like 20 years ago

Finally seems to be happening

Gains

Feeding my nostalgia :)

Going to be a part of gcc

Eventually we will probably update gcc...

Learned some interesting bits of the C++ standard

Unofficial GDC and GM2 packages

If you want, I made available **unofficial and unsupported** packages for the GNU D Compiler and GNU Modula-2. Both are available here:

<https://github.com/ibara/misc/releases/tag/v6.9>

Requires bleeding edge (as of May 5, 2021) -current

Pay very close attention to the MESSAGES that appear after install!



The positives

Why I do this, and you should too.

Gives our users the impetus for interaction

OpenBSD users can connect with the language community

Maybe this is the right tool for someone trying to get their work done

The language community can connect with OpenBSD

Maybe OpenBSD provides a unique perspective into helping make the language better

Learn some new languages

Did you know...?

You don't need to know anything about the target language to successfully port it!

Personal examples:

Seed7

GNU Cim

Kona

You can always learn the language after you have the tools to do so!

Get your name in the GCC source tree!

Yes, for real:

<https://gcc.gnu.org/git/?p=gcc.git;a=blob;f=libphobos/libdruntime/core/sys/openbsd/unistd.d>

Is it significant? Not at all.

Am I amused? *Absolutely!*



The negatives

What goes missing, and how you can help

Loss of mitigations, potentially

It depends on a lot of factors.
You don't lose everything, of course...

Unless you're tcc, then you lose (almost) everything!

How you can (easily) check:

If the compiler compiles target language to something base clang then compiles, that's the best scenario. Clang will compile it like any other C/C++ code, and you get all the mitigations built into clang.

Loss of support, potentially

Many upstreams are really good at incorporating OpenBSD support patches.

YMMV, every upstream is different. But my experiences have been mostly positive.

Even so, it's on us to ensure continued support. Many upstreams aren't going to install OpenBSD to check for issues. But they are happy to receive patches that fix issues discovered only on/by OpenBSD :)

Takeaways

What the future holds

More languages, of course

I am **always** on the lookout for more languages to port to OpenBSD. If you know of any, let me know!

More users, perhaps

If you are someone who started using BSD, and discovering that your language of choice was one of the factors that pushed you over the edge to start using BSD, I want to hear from you!

I also want you to bring the BSDs with you to your language community, and be a good steward of the BSDs there.

More porters, please

I have occasionally collaborated with devs from other BSDs when working on bringing up support for a new compiler (Nvidia Flang, notably). But oftentimes it is a solo operation. Having others to share the work, both initial and long-term, would be great.

Get involved!

Yes, port the languages you want to OpenBSD

But also document it

I blog. But I think videos are a good addition, I should do those too

Make a splash in the language community

Maybe they'll end up making a splash in ours, too!

WARNING: PLEASE TRY THIS AT HOME!

- I can't (and fortunately, don't have to) do it all myself!
- It would be great to form a *BSD language team of porters across all the BSDs to tackle and share the work of porting and maintaining all these languages and their compilers.
- There are some big ones I need help with...

Thank you!

bcallah@openbsd.org

[@bcallah@bsd.network](https://mstdn.social/@bcallah@bsd.network) (Mastodon)