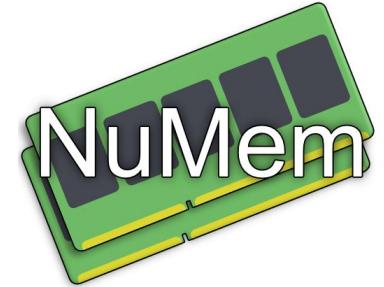

Moving to numem



What's new in Dplug?



Meeting
Dec 18th 2025

Executive Summary



- Dplug v16 now depends upon `numem` package
 - End goal is to allow an alternative D runtime (nurt)
- For now on using Dplug 16+ you need to move to LDC 1.41 (works in macOS 10.15+, Ubuntu 22.04+, Windows 7+)
- You don't have to change anything else (not yet).

Current situation



Your plugin
here

Dplug

Phobos
+druntime
(disabled!)

OS functions

C runtime

Current situation

A Dplug plug-in depends on:

- the C runtime
- the disabled D runtime (no GC, no thread registration)

Your plugin here

Dplug

Phobos
+druntime
(disabled!)

OS functions

C runtime

Current situation

A Dplug plug-in depends on:

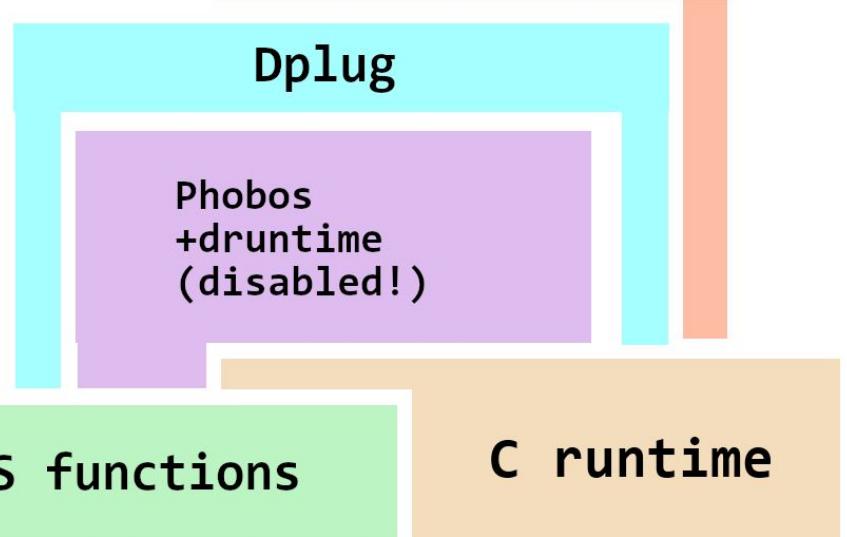
- the C runtime
- the disabled D runtime (no GC, no thread registration)

Consequence

Can only use libraries made that depends on

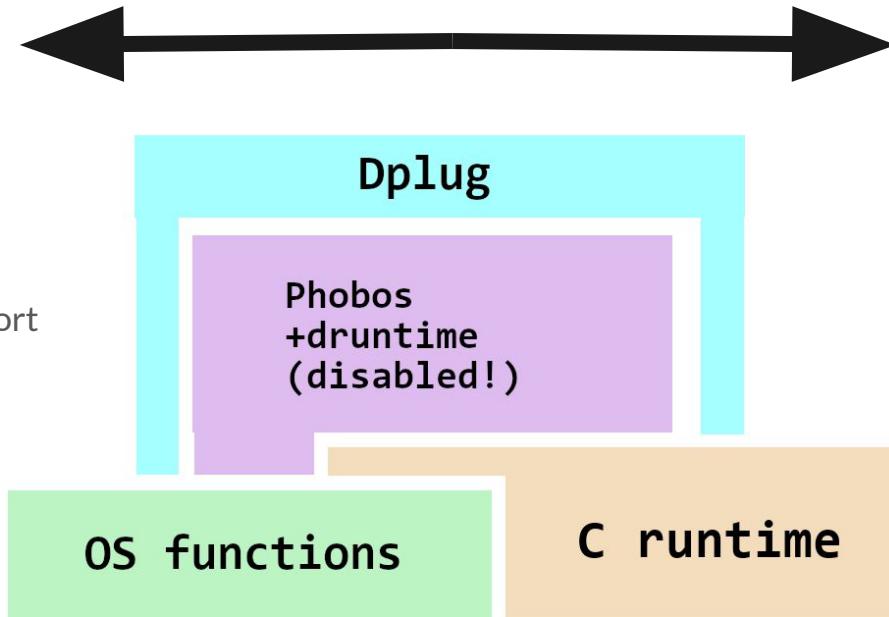
- the C runtime
- A disabled D runtime
- and/or `dplug:core`

And are `@nogc`.



This is our “runtime surface”

Larger = More difficult to port



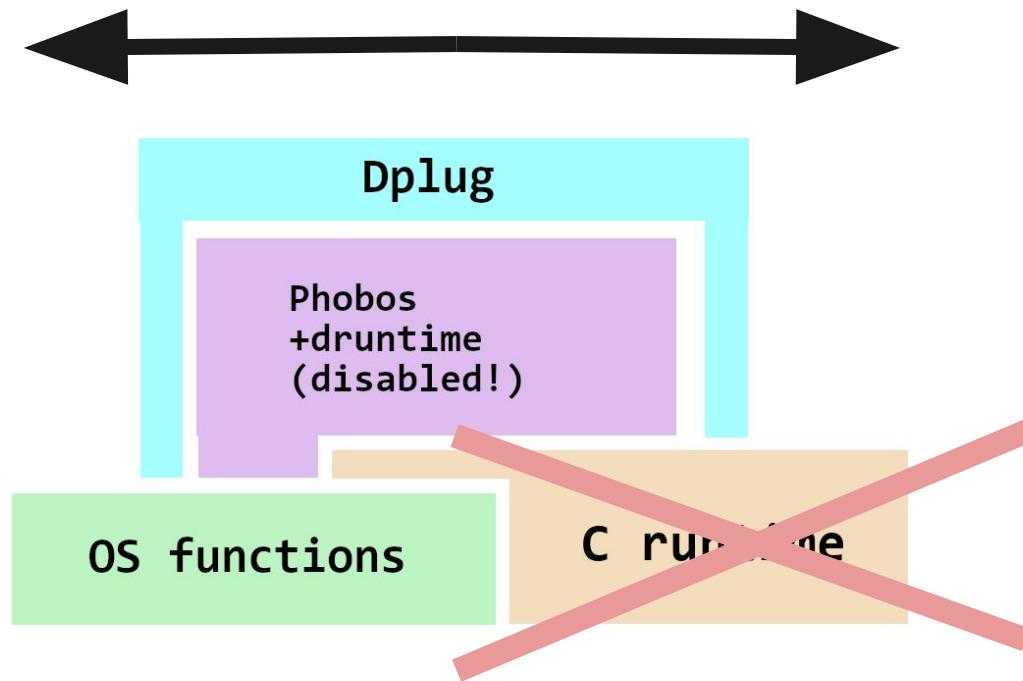
Risks and opportunities : Web

FL Studio is now on the web: <https://fl.studio>

And we have no support for WASM



There is no libc in WASM



Risks and opportunities: disabled Druntime is DANGEROUS

Phobos
+
druntime (disabled)

- In a Dplug plugin, inadvertently using the Druntime (GC) enables it; which is a crash at runtime one *some* platforms.

Risks and opportunities: disabled Druntime is DANGEROUS

Phobos
+
druntime (disabled)

- In a Dplug plugin, inadvertently using the D runtime (GC) enables it; which is a crash at runtime one *some* platforms.
- Might as well not link with it, and get a link error

Risks and opportunities: disabled Druntime is DANGEROUS

Phobos
+
druntime (disabled)

- In a Dplug plugin, inadvertently using the Druntime (GC) enables it; which is a crash at runtime one *some* platforms.
- Might as well not link with it, and get a link error
- Right now we *do* depend on some stuff being linked, like some std.math functions and more. UNKNOWN

Risks and opportunities: disabled Druntime is DANGEROUS

Phobos
+
druntime (disabled)

Issue #912 was a crash for every macOS user of every Dplug plugin.

Sequoia 15.4 breaks everything URGENT

#912 · by p0nce was closed on Apr 3

Still getting support about it, and for years to come.

More future risks with Phobos

Phobos
+
druntime (disabled)

newGC: an initiative to bring better GC to D, unfortunately invasive and preempt every call to `pthread_create`

More future risks with Phobos

Phobos
+
druntime (disabled)

newGC: an initiative to bring better GC to D, unfortunately invasive and preempt every call to `pthread_create`

PhobosV3: an initiative to break Phobos in baseD and a GC-using Phobos.

More future risks with Phobos

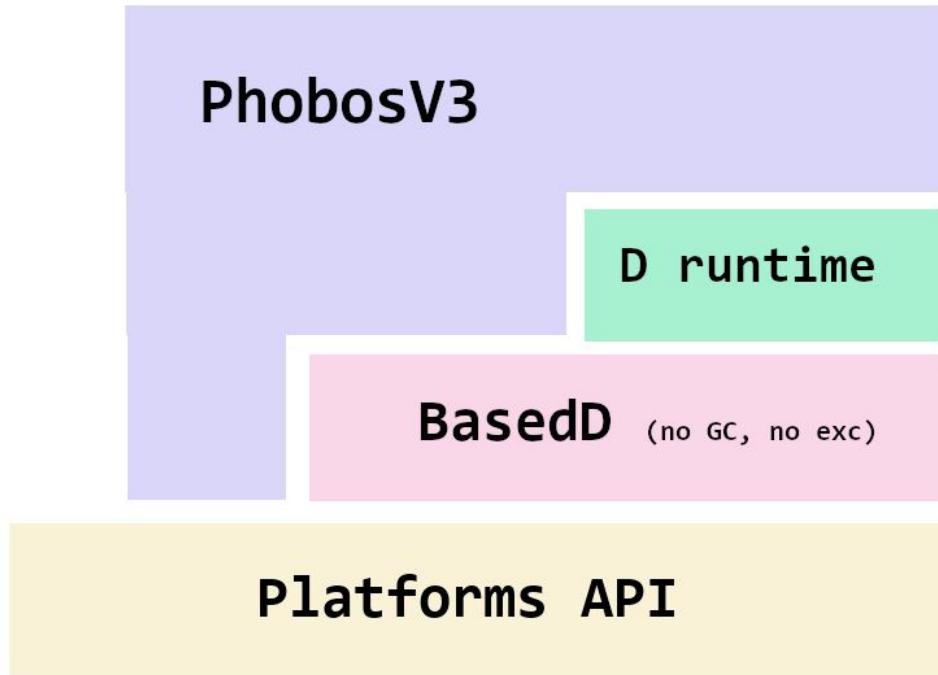
Phobos
+
druntime (disabled)

newGC: an initiative to bring better GC to D, unfortunately invasive and preempt every call to `pthread_create`

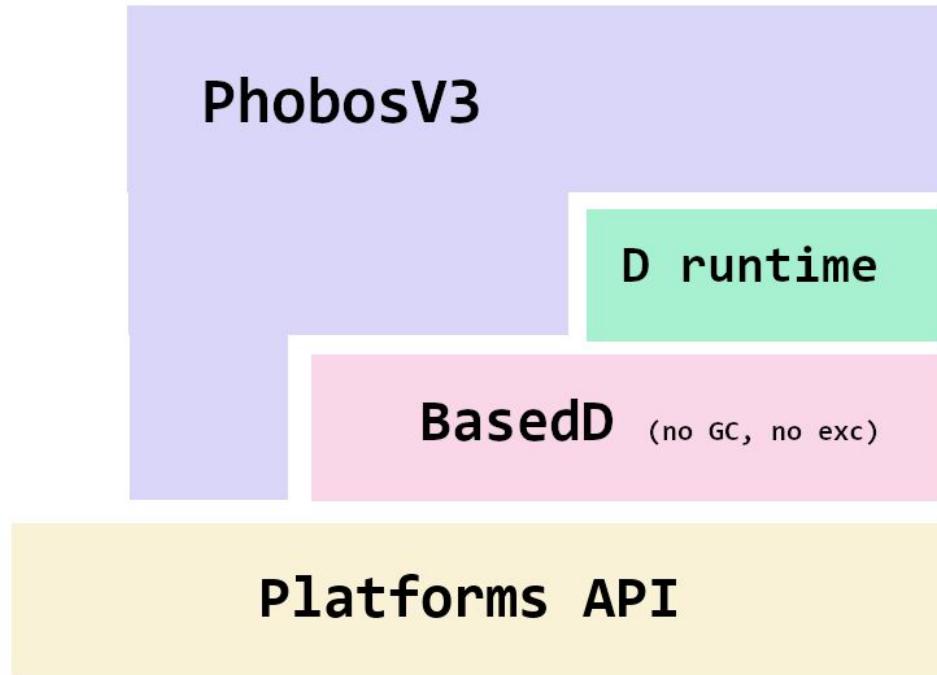
PhobosV3: an initiative to break Phobos in baseD and a GC-using Phobos.

Large instabilities incoming. I don't want us to be at the mercy of whatever Phobos does.

Future shape of Phobos/druntime



Future shape of Phobos/druntime



Kind of what we need?

- Except not sure if we'll have class
 - virtual dispatch is a NEED
- Except it doesn't exist yet!

What if I told you...



...other people are making consumer software in D?

Inochi2D



Inochi2D

Open Source Real-time 2D Animation

Verified

466 followers

Denmark

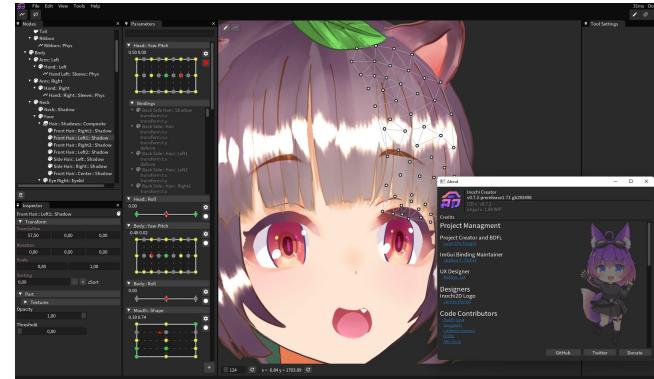
<https://inochi2d.com>

@Inochi2D

<https://bsky.app/profile/inochi2d.com>

dev@inochi2d.com

Inochi2D is a framework for realtime 2D puppet animation – by creating 2D meshes and layering creating the illusion of depth and movement from using 2D artwork.



Some of the libraries built for Inochi2D by Luna



Niobium => 3D graphics API abstraction



Hairetsu <= Font library

and: many Apple bindings using the new Objective-C binding
method, also by Luna...

Some of the libraries built for Inochi2D by Luna



Niobium => 3D graphics API abstraction

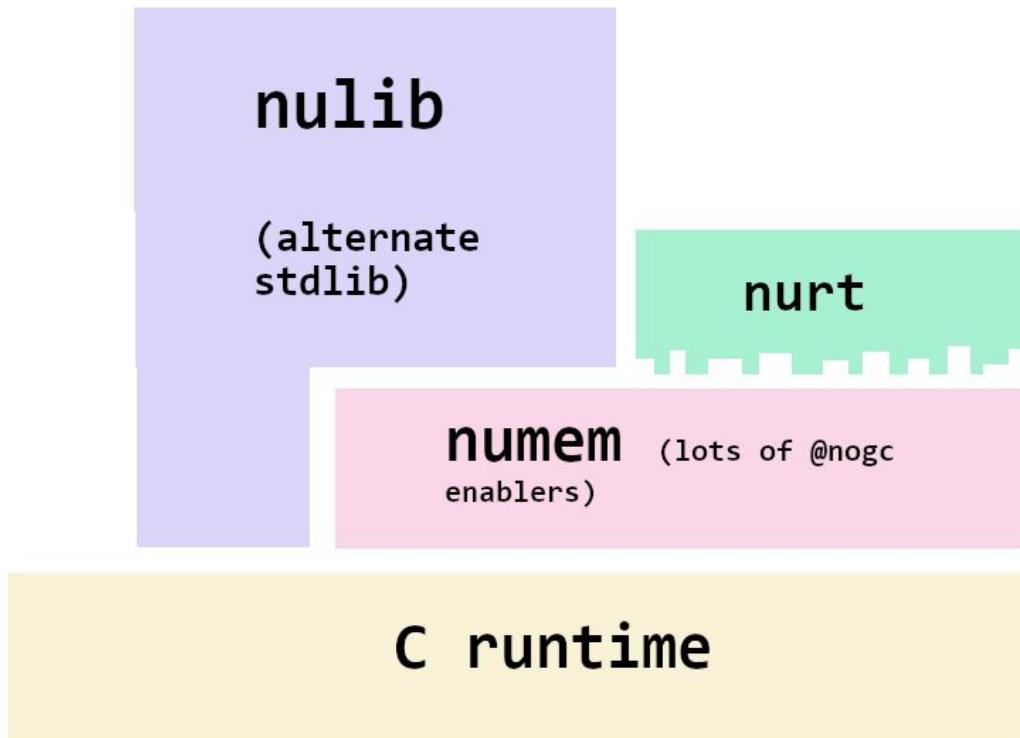
Hairetsu <= Font library

and: many Apple bindings using the new Objective-C binding method, also by Luna... **Goal of this stack is serve the needs of Inochi2D, not ours.**

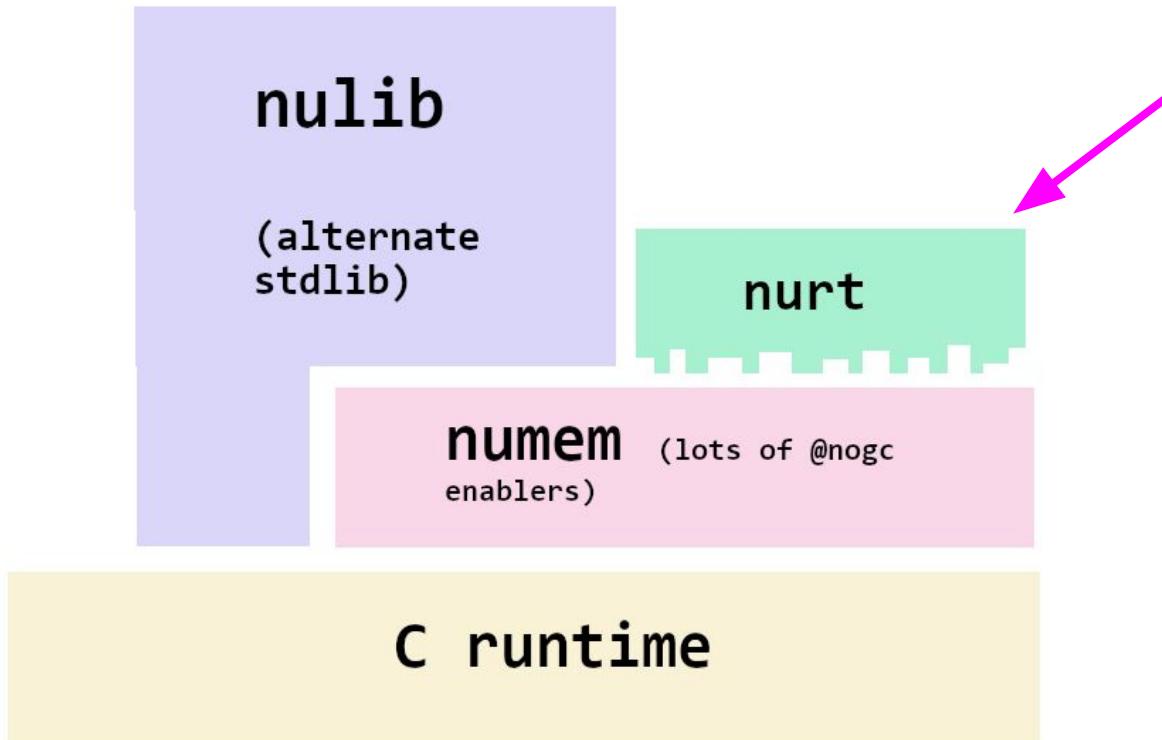


Let's be RESPECTFUL of the maintainers

The Inochi2D “nu” stack

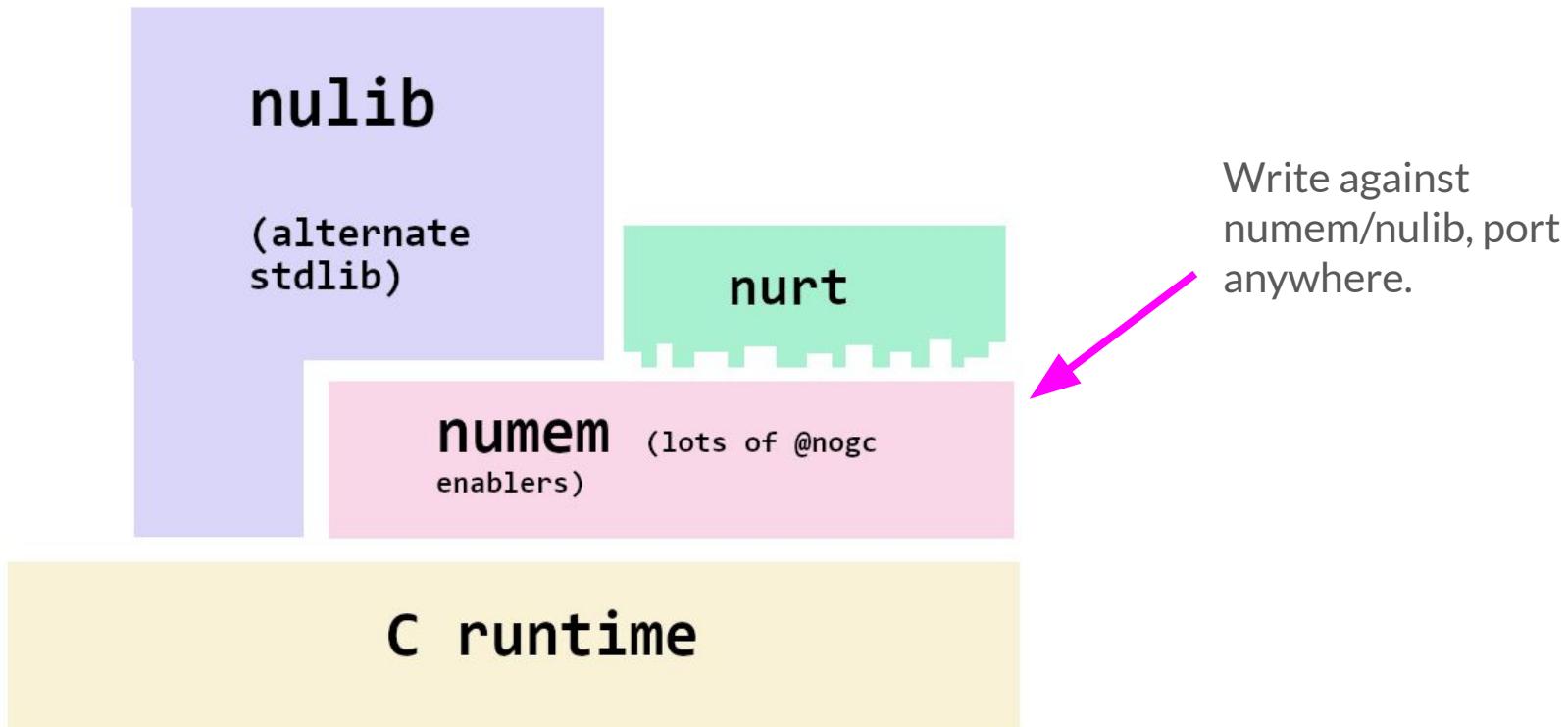


The Inochi2D “nu” stack

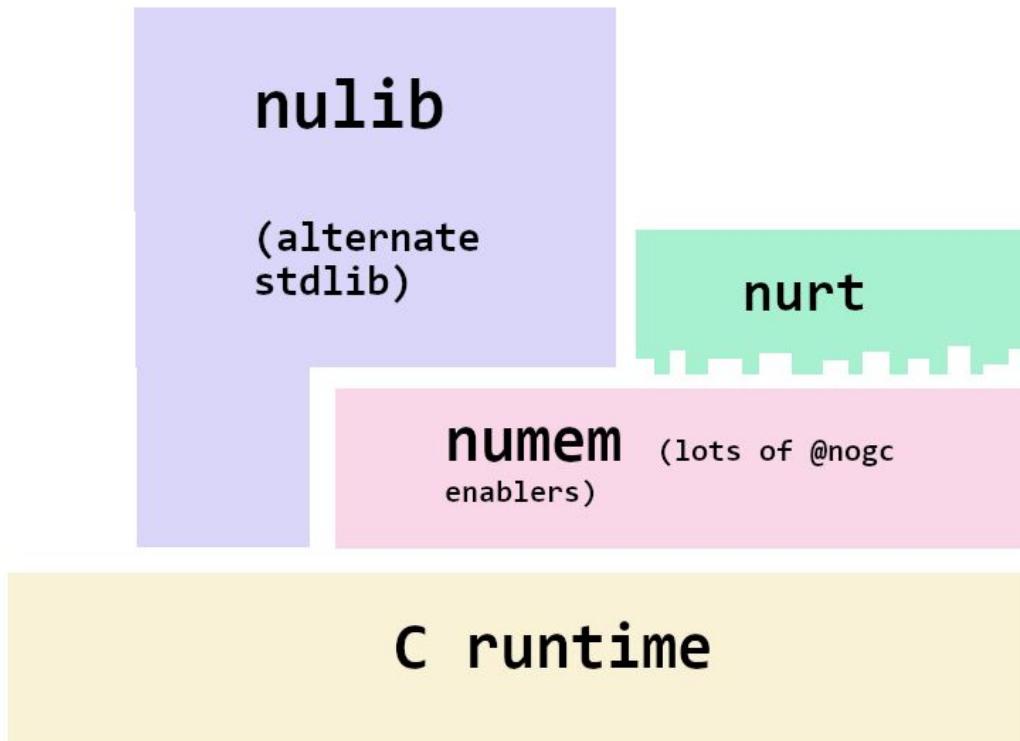


Pluggable D runtime.
nurt dependency is a late
decision.

The Inochi2D “nu” stack

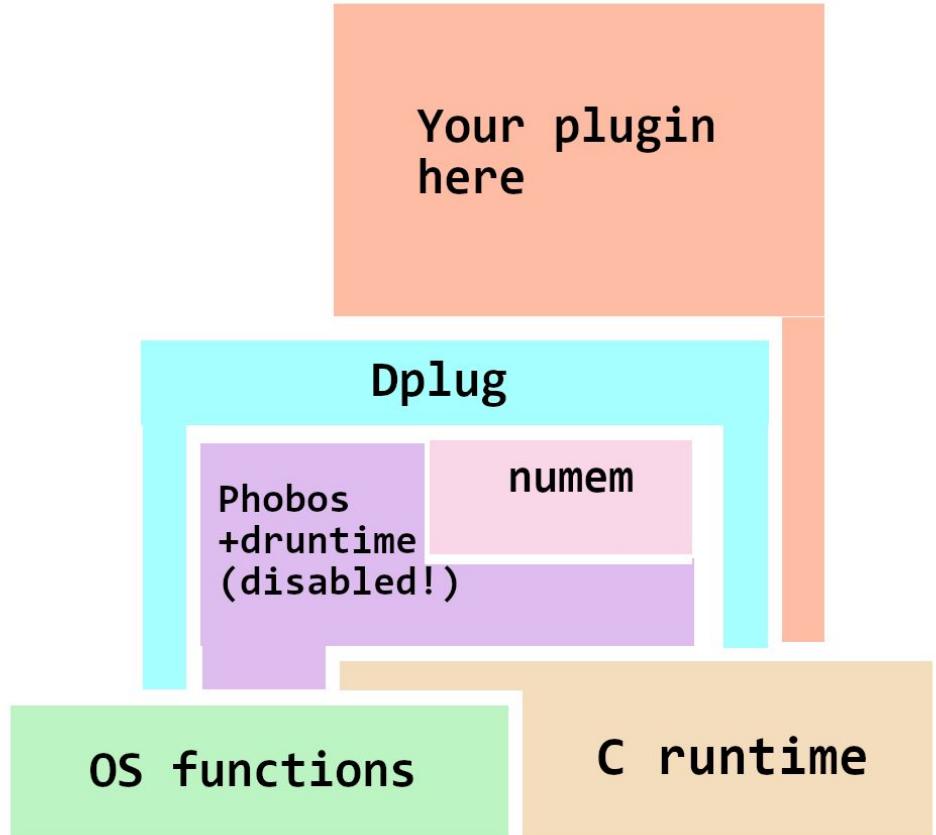


The Inochi2D “nu” stack



Also kinda pluggable
with numem “hooks”
(weak linking)

Today's transition situation

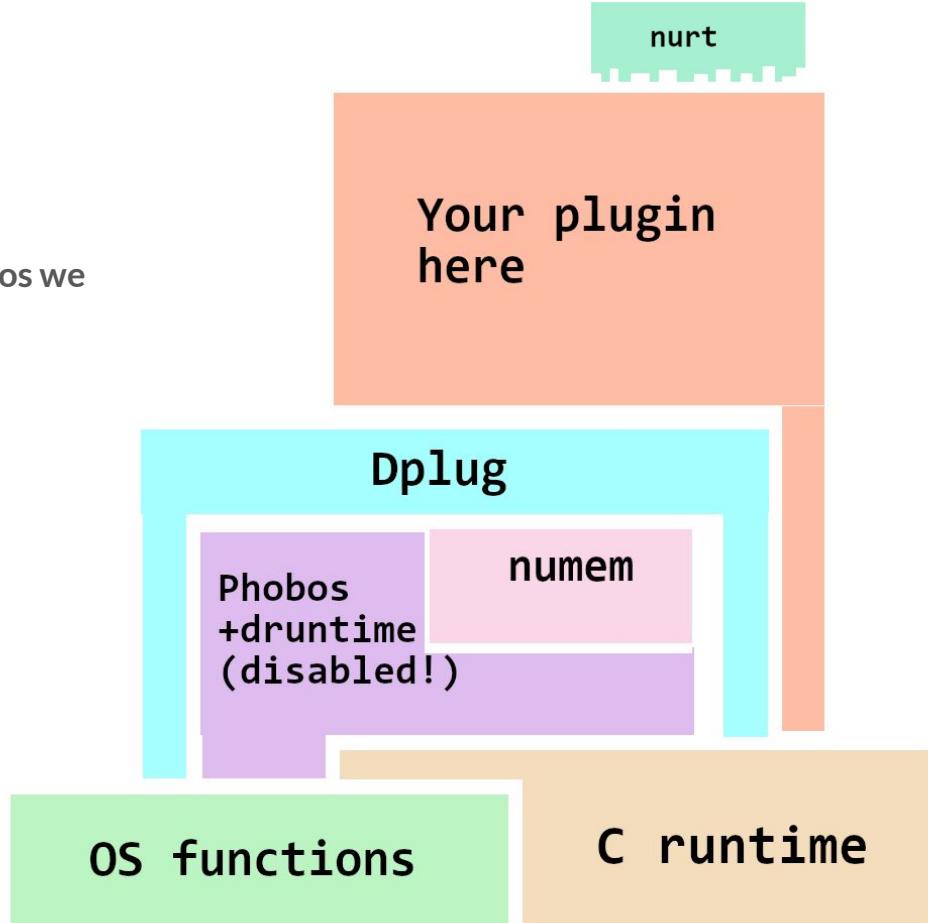


Today's transition situation

nurt

- 1. We'll need to find every part of Druntime/Phobos we actually use. Dplug shall use nulib.

Example: std.math.exp std.math.pow

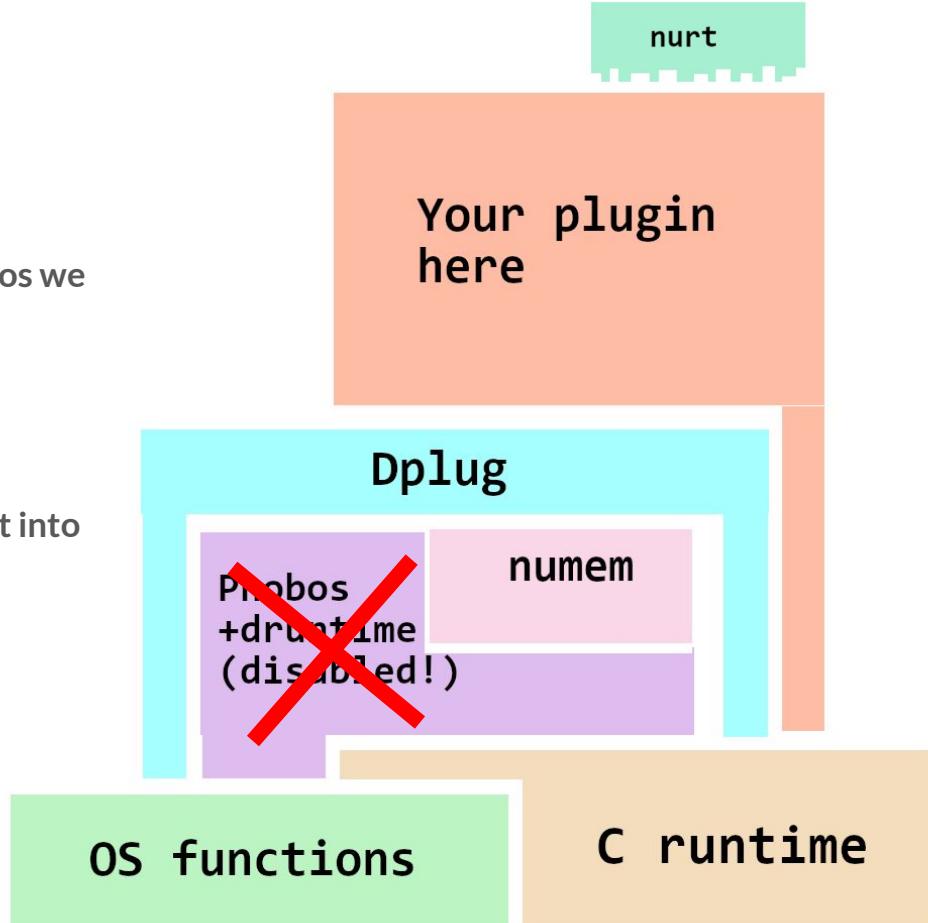


Today's transition situation

1. We'll need to find every part of Druntime/Phobos we actually use. Dplug shall use nulib.

Example: std.math.exp std.math.pow

- 2. Plugin developer decides if nurt enabled and get into the slightly restricted subset.



Today's transition situation

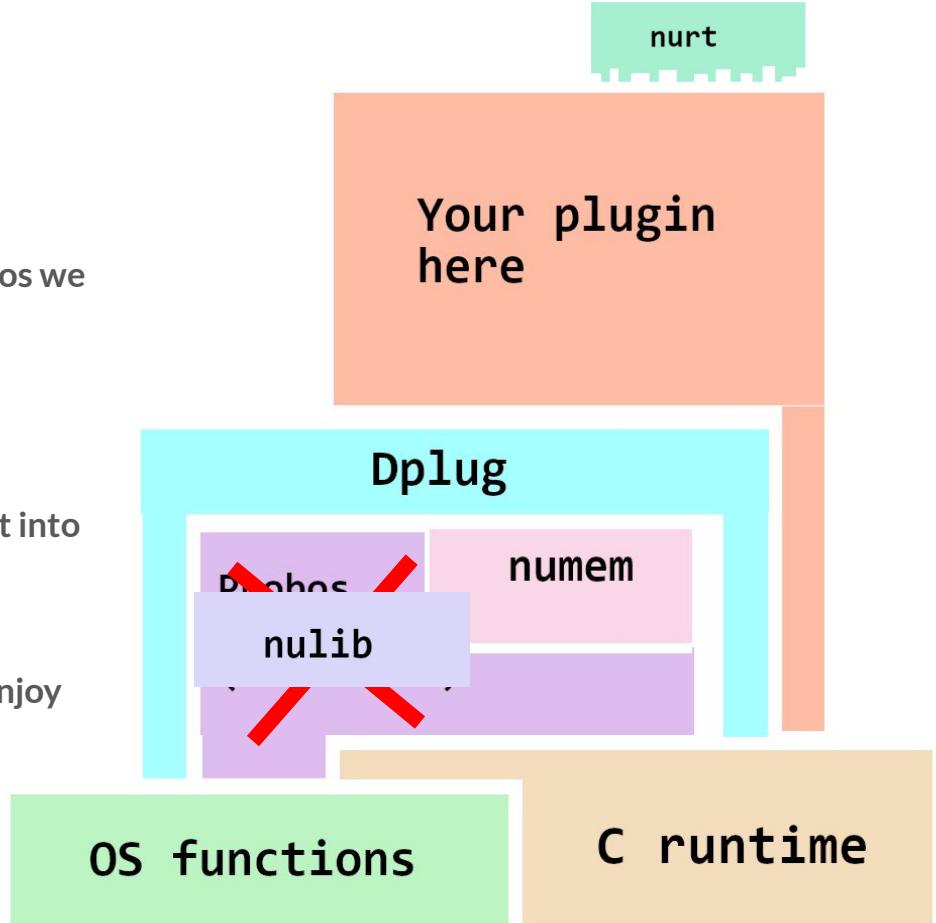
1. We'll need to find every part of Druntime/Phobos we actually use. Dplug shall use nulib.

Example: std.math.exp std.math.pow

2. Plugin developer decides if nurt enabled and get into the slightly restricted subset.



3. Enforces that, drop Phobos and druntime and enjoy increased portability and lower uncertainty, and D runtime as a late binding decision.



By building new code on top of numem/nulib

You allow people from Dplug and Inochi2D verses to reuse your code.

You do not constrain D code to be used with a particular D runtime (of which, 6 or 7 exist).

Normal D applications and scripts can use them.

Not limited to a (hastily designed) dplug:core + just core/stdc



Questions?