

Package: anvlcc (via r-universe)

July 20, 2026

Title TinyCC Backend Bridge from 'anvl' Traces to 'tccquickr' Kernels

Version 0.0.0.9000

Description Compiles 'anvl' traced array programs to C kernels through 'tccquickr' and the TinyCC just-in-time compiler from 'Rtinycc'. Traced graphs are lowered to declared R using anvl's own quickr lowering rules wherever those emit tccquickr's declared subset, with bridge-owned tinycc rules for primitives that have a native tccquickr surface form (matrix products as '%*%' contractions, aval-checked broadcast passthrough). The result is a millisecond-latency CPU compile path for anvl programs with no XLA dependency.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Imports anvl (>= 0.3.0), S7, tccquickr, tengen

Suggests Rtinycc, rmarkdown, tinytest

Additional_repositories <https://r-xla.r-universe.dev>,
<https://sounkou-bioinfo.r-universe.dev>

Repository <https://sounkou-bioinfo.r-universe.dev>

Date/Publication 2026-07-20 19:36:49 UTC

RemoteUrl <https://github.com/sounkou-bioinfo/anvlcc>

RemoteRef HEAD

RemoteSha 5cd2e89b6faaa0a1aee629e8a8238b2729444fcf

Contents

graph_to_tinycc_r_function	2
tinycc_jit	2
tinycc_source	3

Index	4
--------------	----------

graph_to_tinycc_r_function

Lower an anvl graph to a declared R function

Description

Walks an `anvl::AnvlGraph`'s calls in order and emits one declared R function in tccquikr's input dialect. Each primitive is lowered by its tinycc bridge rule when one claims the configuration, and by anvl's own quickr rule otherwise, so the bridge only owns the primitives whose quickr code would leave tccquikr's declared subset.

Usage

```
graph_to_tinycc_r_function(graph)
```

Arguments

graph (AnvlGraph)
 A graph traced by `anvl::trace_fn()`.

Details

Graph constants become trailing formals; their traced values are attached as the `anvl_tcc_constants` attribute so callers can complete the call.

Value

A function whose body starts with `declare(type(...))`, suitable for `tccquikr::tccq_analyze()` and plain R evaluation alike.

tinycc_jit

Compile an anvl-traceable function to a TinyCC kernel

Description

Traces `f` with `anvl::trace_fn()`, lowers the graph through `graph_to_tinycc_r_function()`, and JIT-compiles the result with tccquikr's Rtinycc backend. The returned function takes plain R scalars/vectors/arrays in the traced argument order and returns plain R values; traced constants are bound automatically.

Usage

```
tinycc_jit(f, ...)
```

Arguments

f	(function) Function over anvl arrays to trace.
...	Named example arguments for tracing: AnvlArrays, or plain R values which are wrapped with the "plain" anvl backend.

Value

A compiled function over plain R values.

tinycc_source	<i>Show the C source for an anvl-traceable function</i>
---------------	---

Description

Same pipeline as `tinycc_jit()` but stops at source emission.

Usage

```
tinycc_source(f, ...)
```

Arguments

f	(function) Function over anvl arrays to trace.
...	Named example arguments for tracing: AnvlArrays, or plain R values which are wrapped with the "plain" anvl backend.

Value

(character(1)) the generated C kernel source.

Index

`anvl::AnvlGraph`, [2](#)
`anvl::trace_fn()`, [2](#)

`graph_to_tinycc_r_function`, [2](#)
`graph_to_tinycc_r_function()`, [2](#)

`tccquickr::tccq_analyze()`, [2](#)
`tinycc_jit`, [2](#)
`tinycc_jit()`, [3](#)
`tinycc_source`, [3](#)