16 AUGUST 2018 / GOLANG

Statically compile GOLANG programs

One of the reasons I love working with `GO` is because I can compile go code into a `statically linked binary`. Why bother? Well, it's hard to answer but simply putting it everything is needs is right inside the binary as opposed to a `dynamically linked` binary. A dynamically linked binary is usually smaller in size as it `dynamically` links to it's dependencies installed on an `OS`. Lets take `LibC` for example, `LibC` contain certain library functions required by the application code, when I run a `dynamically` linked binary, it will link to whatever `LibC` found on the system whereas a statically linked binary has `LibC` (and other dependent libraries) embeded inside the binary itself hence `statically` linked binary is larger in size but does not use any of the system libraries. It has many advantages but those are out of scope of this article. This article answers `How` and assumes you know `Why`.

To compile a `source code` is required, let's use the code that we produced during restful article which can be found at github.

Now lets get started. It's easy to compile `golang` code,

```
go build .
```

does the job. It would produce a binary (dynamically linked) with the same name as the `directory`. If you want to name the binary in a different name, you can do

```
go build -o app .
```

`-o` tells `go build` to set the `output` binary name, the `.` in the end tells the compiler to compile `package` represented by `current directory`. Go compiler compiles `packages` as a smallest distributable unit is a `package`.

Linking statically

Now lets compile statically, in order to do so, we need to tell compiler for what OS to compile for and what system platform. In my case the OS is `darwin` (os-x), you might be having `linux` (any linux distribution), it also needs `CGO` settings, to statically compile we have to tell compiler to disable `CGO` by setting `CGO_ENABLED=0`. You can optionally provide `linker arguments`. There are many other configurations that you can pass during compiling or even writing code, but for this article we are only going to cover basic.

Following code will compile the binary statically.

for OS-X

```
env CGO_ENABLED=0 GOOS=darwin GOARCH=amd64 go build -a -o app
```

for Linux

```
env CGO_ENABLED=0 GOOS=linux GOARCH=amd64 go build -a -o app
```

As I mentioned, go compiler looks for settings from environment variables as well. In above command line, we specify following environment variables:

- `CGO_ENABLED=0` This tells compiler to disable `CGO` and statically link `c` bindings as well.

- `GOOS=darwin` / `GOOS=linux` This tells compiler for which OS it needs to compile.

- `GOARCH=amd64` This tells compiler to compile for an `amd64` compatible system architecture (64 bit processor).

> *The* `OS` *and* `Platform` *list gets bigger and bigger every day, you can use* `go tool dist list` *to see current latest OS/ARCH combination that you can target. At time of writing this article, this is what I got*
>
> *android/386*
> *android/amd64*
> *android/arm*
> *android/arm64*
> *darwin/386*
> *darwin/amd64*

darwin/arm64

dragonfly/amd64

freebsd/386

freebsd/amd64

freebsd/arm

linux/386

linux/amd64

linux/arm

linux/arm64

linux/mips

linux/mips64

linux/mips64le

linux/mipsle

linux/ppc64

linux/ppc64le

linux/s390x

nacl/386

nacl/amd64p32

nacl/arm

netbsd/386

netbsd/amd64

netbsd/arm

openbsd/386

openbsd/amd64

openbsd/arm

plan9/386

plan9/amd64

plan9/arm

solaris/amd64

windows/386

windows/amd64

Go ahead and invoke ... / ... On a terminal you will not see any

```
images .
```

That's it!

Cheers and Happy coding!

Subscribe to Odd Code

Get the latest posts delivered right to your inbox

| youremail@example.com | Subscribe |

Dave Amit

Howdy folks! I am Dave Amit, an accidental programmer, father to a lab puppy, hubby to a beautiful wife, addicted to puzzles & a noob blogger. This is my effort to simplify odd codes from the wild.

Read More

ALSO ON ODD CODE

Build RESTful api with express-js and …

6 years ago • 2 comments

Step by Step guide on how to build RESTful apis in node-js using express-js …
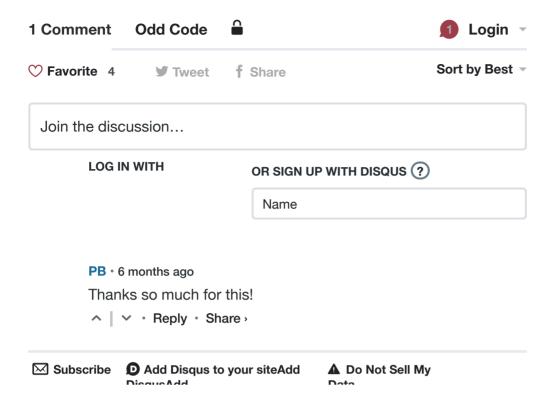
EntityFramework code first migration …
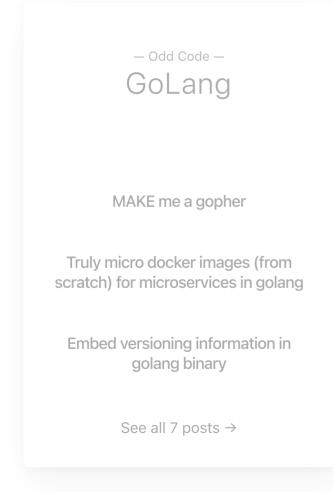
7 years ago • 2 comments

All you need to know about EntityFramework code first migrations, Automatic, …

The Go (visual :

4 years ag

Setup go covers h configure

OC **Odd Code**

Share this 👉

1 Comment   Odd Code 🔒   1 Login ⌄

♡ Favorite 4   🐦 Tweet   f Share   Sort by Best ⌄

> Join the discussion…

LOG IN WITH   OR SIGN UP WITH DISQUS ?

                     [Name   ]

PB • 6 months ago
Thanks so much for this!
∧ | ∨ • **Reply** • **Share ›**

✉ **Subscribe**   🅓 **Add Disqus to your siteAdd DisqusAdd**   ⚠ **Do Not Sell My Data**

— Odd Code —
GoLang

MAKE me a gopher

Truly micro docker images (from scratch) for microservices in golang

Embed versioning information in golang binary

See all 7 posts →

GOLANG
Embed versioning information in golang binary

Embed version related information while compiling a statically linked using a utility called govvv

👤 DAVE AMIT

OC **Odd Code**

Share this ☞

GETTING STARTED

REST a little bit with Golang.

Basic RESTful Service with GOLANG using mux.

DAVE AMIT

Odd Code © 2022

Latest Posts　　Facebook　　Twitter　　Ghost