

Golang dependency management tools

Sergey Lanzman | 13.12.2017



The Saga of Go Dependency Management

GOROOT

The root of the Go tree, often `$HOME/go1.X`. Its value is built into the tree when it is compiled, and defaults to the parent of the directory where `all.bash` was run. There is no need to set this unless you want to switch between multiple local copies of the repository.

GOPATH

The `GOPATH` environment variable is used to specify directories outside of `$GOROOT` that contain the source for Go projects and their binaries.

No versions

Only default branch

Install:

```
go get -u github.com/go-redis/redis
```

Import:

```
import "github.com/go-redis/redis"
```

Vendor

GOPATH locale per project

Order for find dependency

1. Vendor
2. GOPATH
3. GOROOT

```
import (  
    "time"  
  
    "gopkg.in/redis.v5"  
  
    "github.com/newrelic/go-agent"  
    "github.com/tylerb/gls"
```



Solutions

Single most important point

Support Point

- [manul](#) - Vendor packages using git submodules.
- [Godep](#)
- [Govendor](#)
- [godm](#)
- [vexp](#)
- [gv](#)
- [gvt](#) - Recursively retrieve and vendor packages.
- [govend](#) - Manage dependencies like go get but for /vendor.
- [Glide](#) - Manage packages like composer, npm, bundler, or other languages.
- [Vendetta](#)
- [trash](#)
- [gsv](#)
- [gom](#)
- [Rubigo](#) - Golang vendor utility and package manager

<https://github.com/golang/go/wiki/PackageManagementTools>

Glide

Are you used to tools such as Cargo, npm, Composer, Nuget, Pip, Maven, Bundler, or other modern package managers? If so, Glide is the comparable Go tool.

Example <https://github.com/kubernetes-incubator/external-dns>



Dep

THANK YOU