

Exploring Go Modules

INTRODUCTION AND OVERVIEW



Michael Van Sickle

@vansimke



Course Overview

Introduction

Standard workflows

Managing dependency
versions

Advanced scenarios



Overview



Goals

History

Overview of modules

First look



Goals of Module System

Keep what works well with
workspaces

Address weaknesses



Goal: Keep What Works Well

**Retrieve
dependencies**

**Simplify build
process**

Share projects*



Goal: Address Weaknesses

Versioning and API stability

Vendoring and reproducible
builds



History

Versioning and API stability

Compatibility guidelines (2013)

gopkg.in (2014)

Semantic versioning proposal (2015)

dep (2016)

Vendoring and reproducible builds

goven (2012)

godep (2013)

gb (2015)

vendor experiment (2015)

dep (2016)

<https://research.swtch.com/vgo-intro#background>



Overview of Modules

One or more
related packages

Configured via
go.mod file

Version controlled

Strict semantic
versioning

Dependent
libraries kept in
cache

Integrity checks
via checksums



Overview of Modules – go.mod

```
module github.com/pluralsight/gomodules
```

```
go 1.12
```

```
require (  
    github.com/gorilla/mux v1.7.3 // indirect  
    github.com/gorilla/rpc v1.2.0 // indirect  
    golang.org/x/text v0.3.0  
)
```

```
// patching the way routing is done
```

```
replace github.com/gorilla/mux => ../mymods/github.com/vansimke/mux
```

```
exclude github.com/gorilla/rpc v1.1.0
```



Summary



Goals

History

Overview of module system

First look

