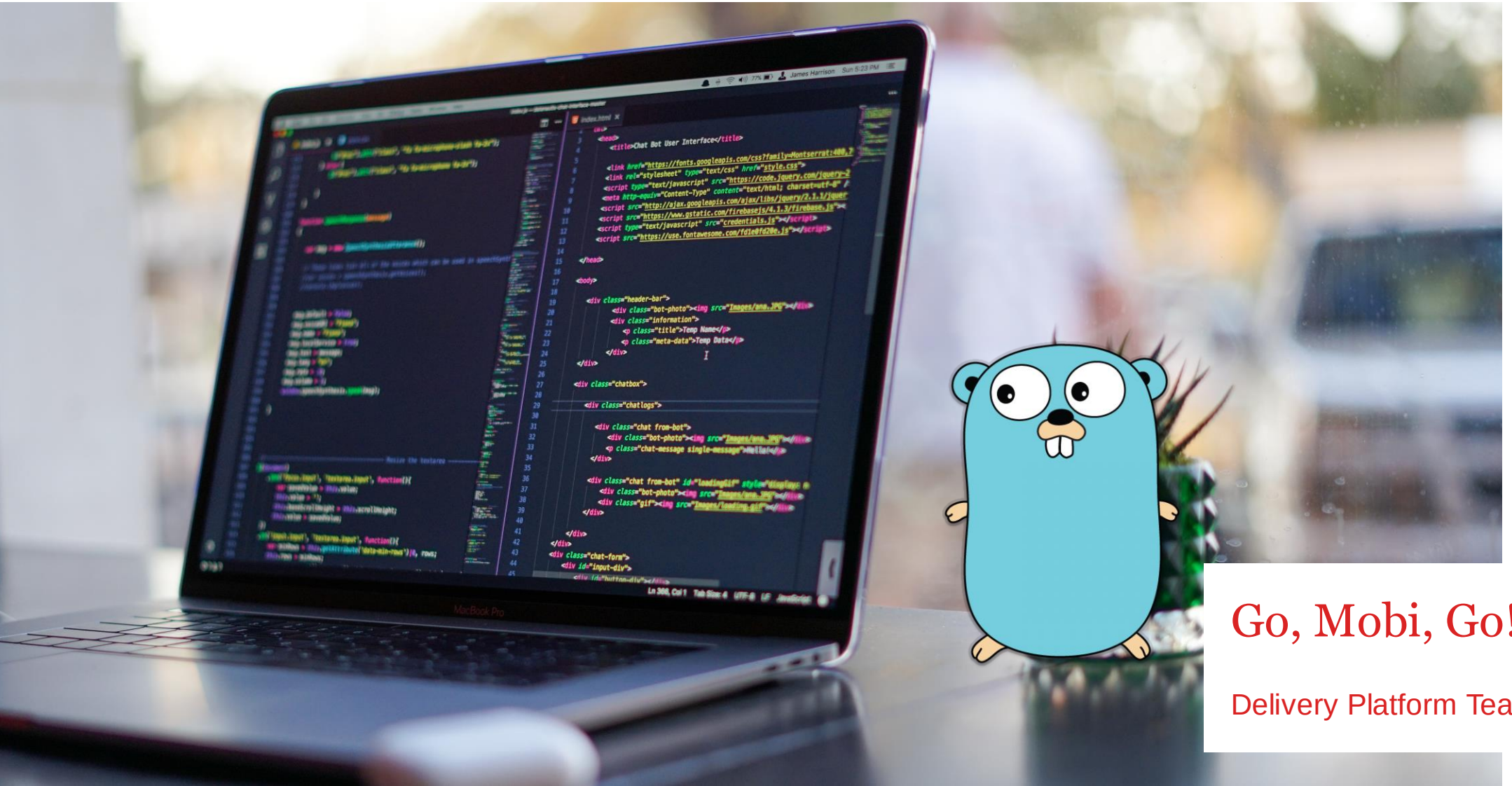
die Mobiliar



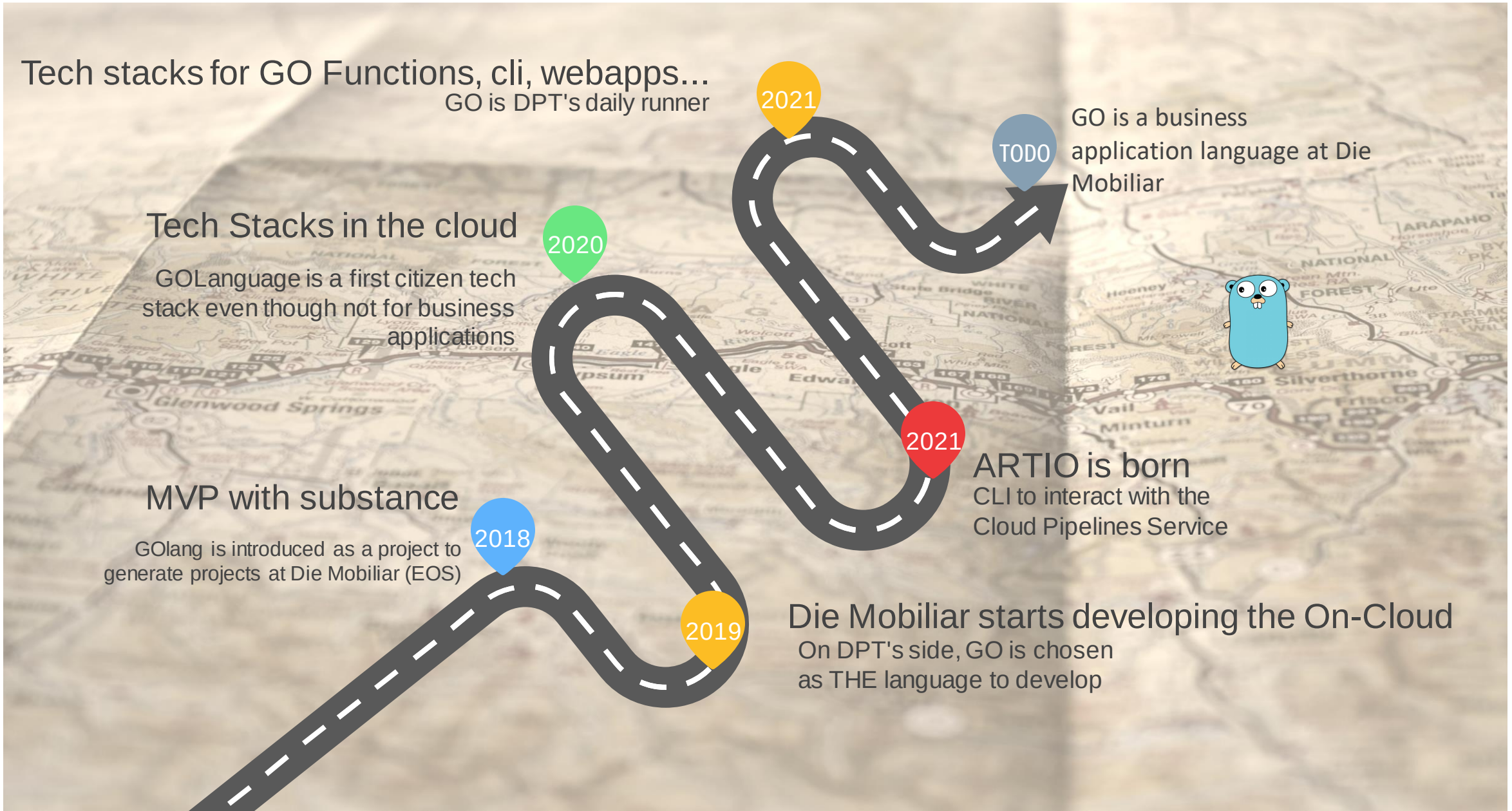
Go, Mobi, Go!

Delivery Platform Team

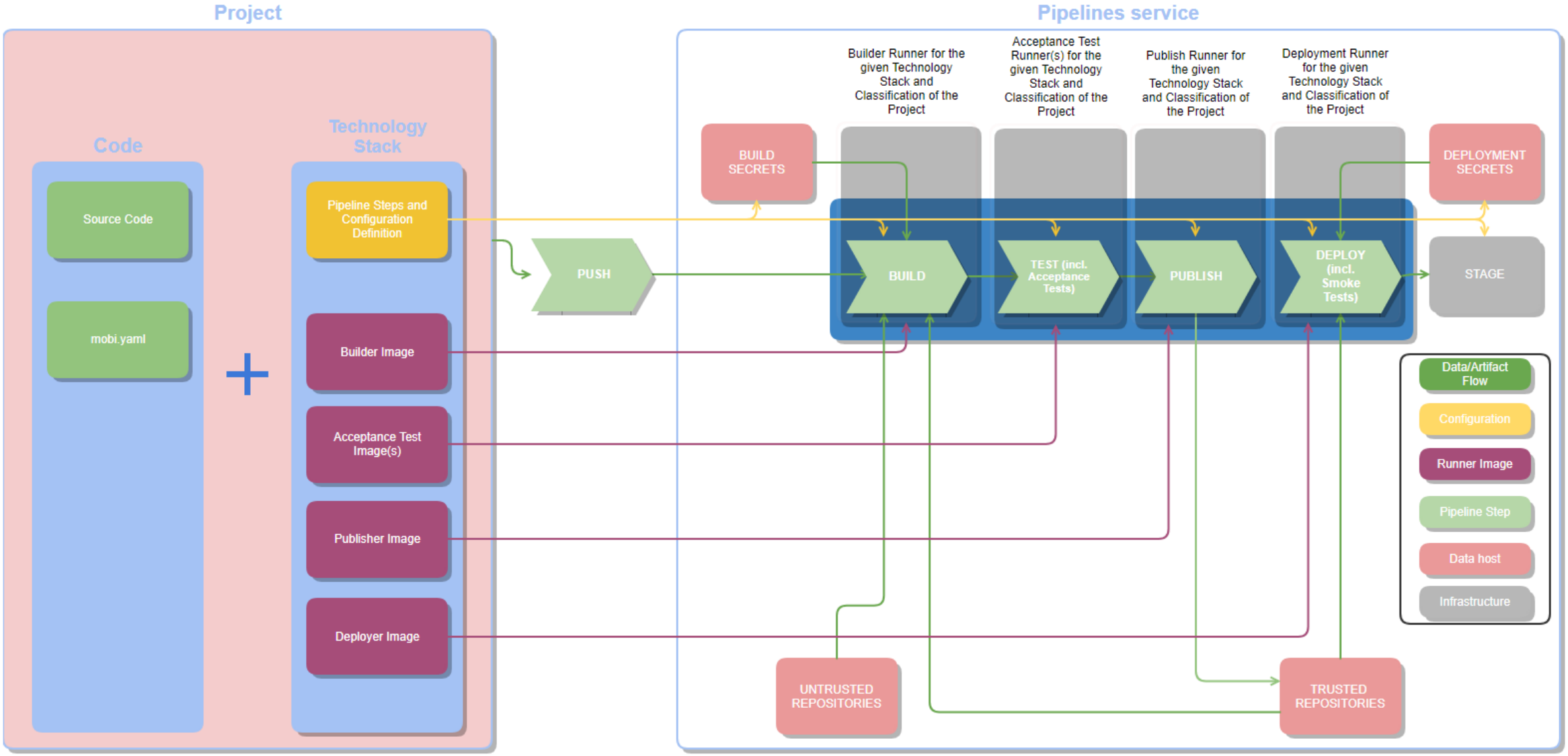
Agenda

1. Die Mobiliar's IT GO journey so far
2. Go Tech Stack(s)
3. Lessons learned
4. Questions

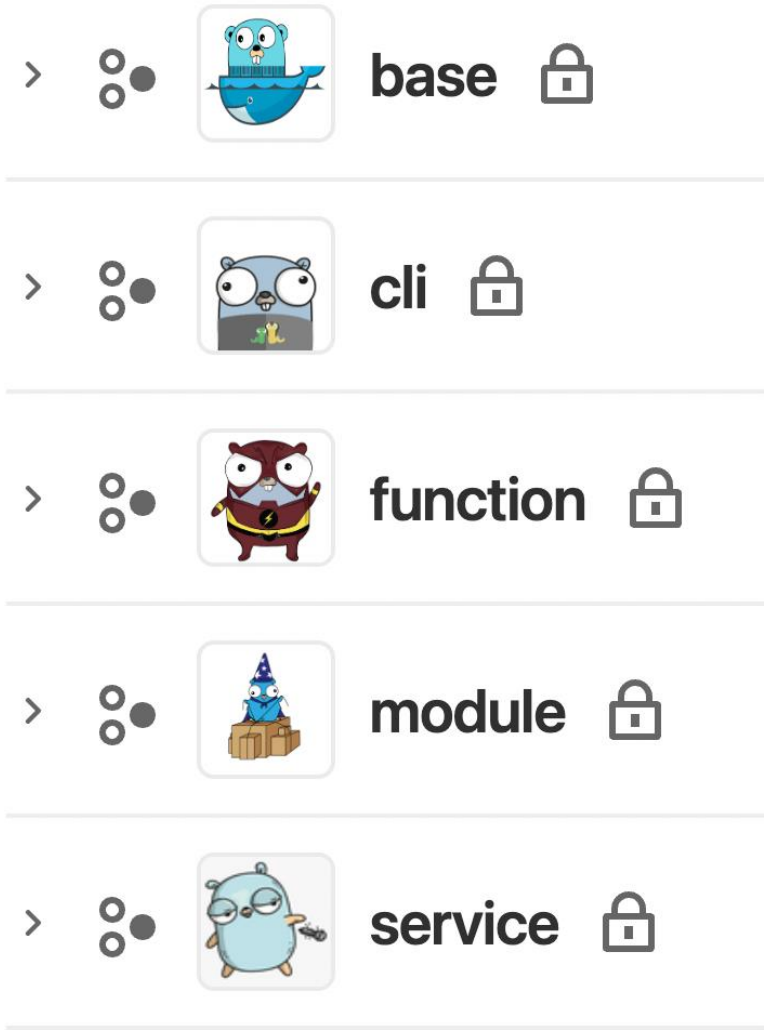
Die Mobiliar's GO journey so far...



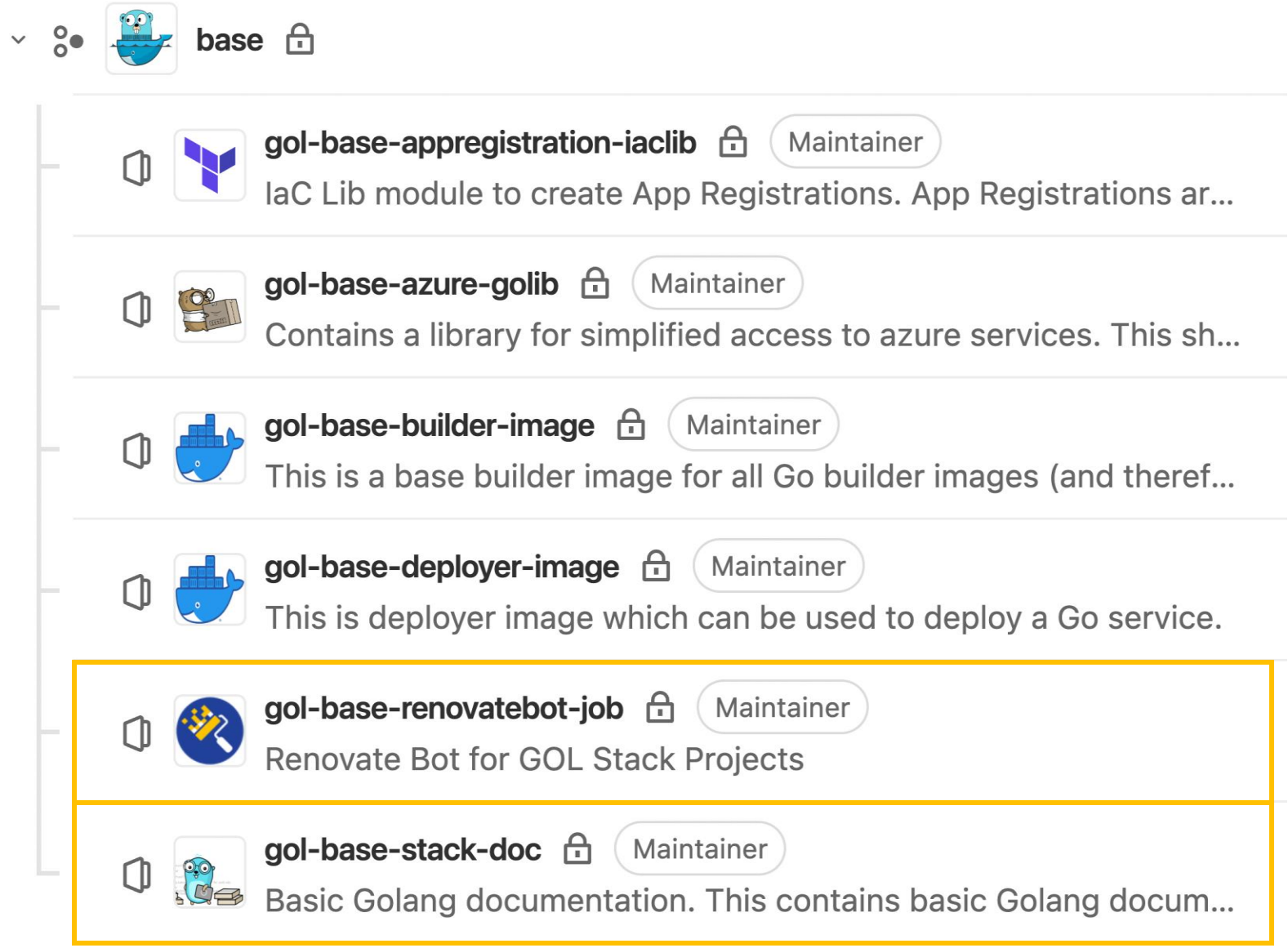
Technology Stacks in the Cloud Pipelines service



Go stacks



Go base projects



gol-base-stack-doc

Project ID: 23961308 Leave project

ENGINEERING HANDBOOK DOCS GOLANG STACK LATEST

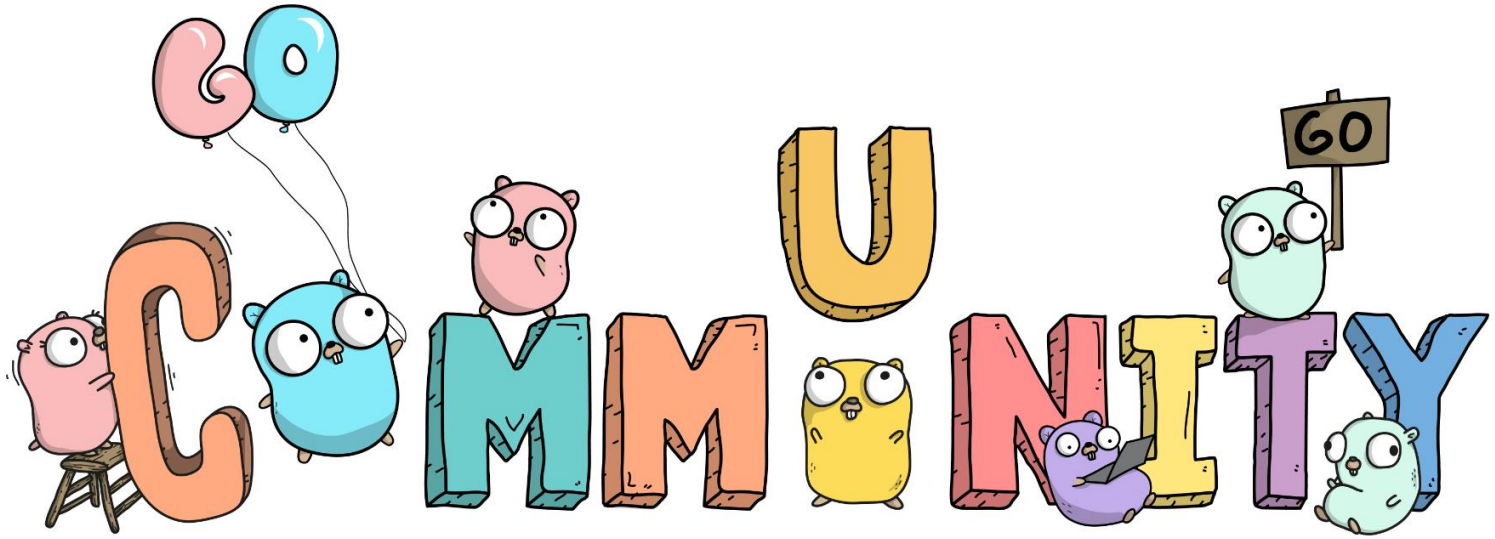
GOLANG STACK

Lifecycle

Build caching

- FUNCTIONS
  - Introduction
  - Getting started
  - Logging & Tracing
  - Local dev guide
- SERVICES
  - Introduction
  - Getting started
- MODULES
  - Introduction
  - Getting started
- CLI
  - Introduction
  - Getting started
- HELP AND SUPPORT
  - Contact

Introduction



Welcome to the Go Mobiliar community! The following chapters will help you with the setup of your Go environment on the Mobiliar context and also explain what we have prepared to run services and functions in the Mobiliar Cloud environment.

Warning: Golang is only allowed to be used for infrastructure services, but not for any kind of business logic. Do not use Go for a microservice/function used in a Mobiliar business context.

gol-base-renovatebot-job

Project ID: 33558498 Leave project

chore(renovate): Update Go modules (by gol-base-renovatebot-job)

Edit Code

Merged **Z101074** requested to merge `renovate/gomod` into `master` 2 hours ago

Overview 0 | Commits 1 | Pipelines 1 | Changes 2

This MR contains the following updates:

Package	Type	Update	Change
gitlab.com/diemobiliar/it/gol/function/gol-function-golib	require	patch	`v0.1.651` -> `v0.1.653`
gopkg.in/yaml.v2	require	major	`v2.4.0` -> `v3.0.1`

Release Notes

▶ diemobiliar/it/gol/function/gol-function-golib

▶ go-yaml/yaml

Configuration

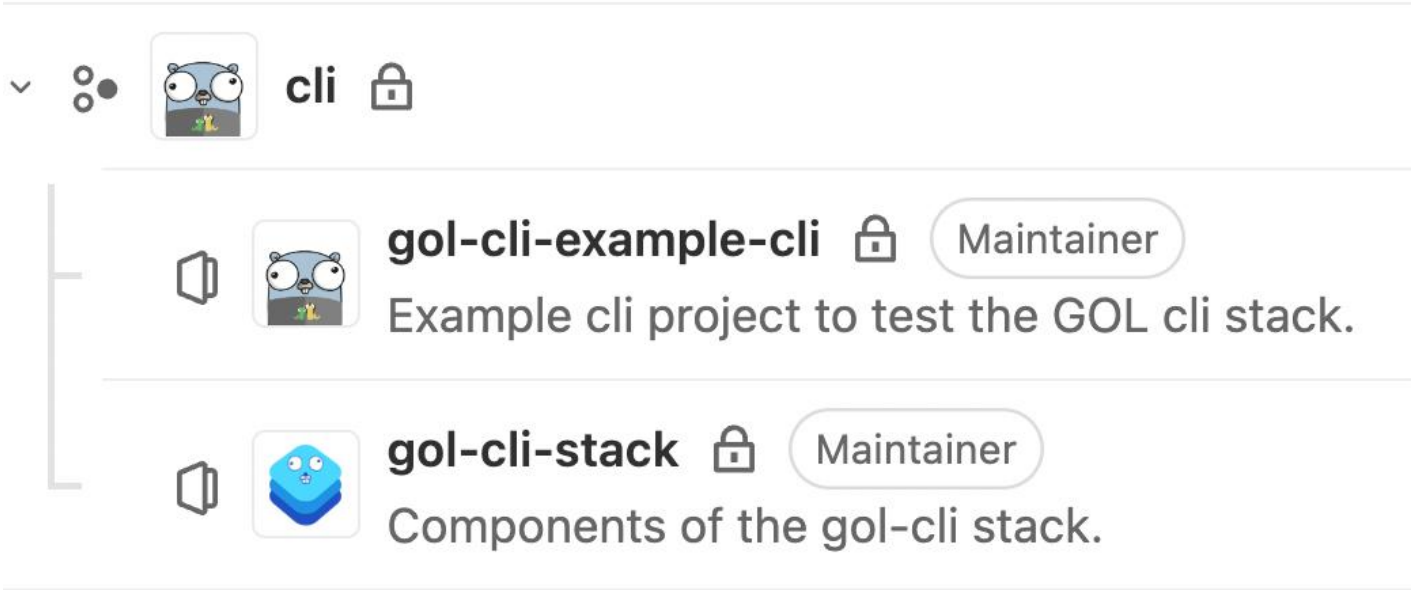
📅 **Schedule**: Branch creation - At any time (no schedule defined), Automerge - At any time (no schedule defined).

🚦 **Automerge**: Enabled.

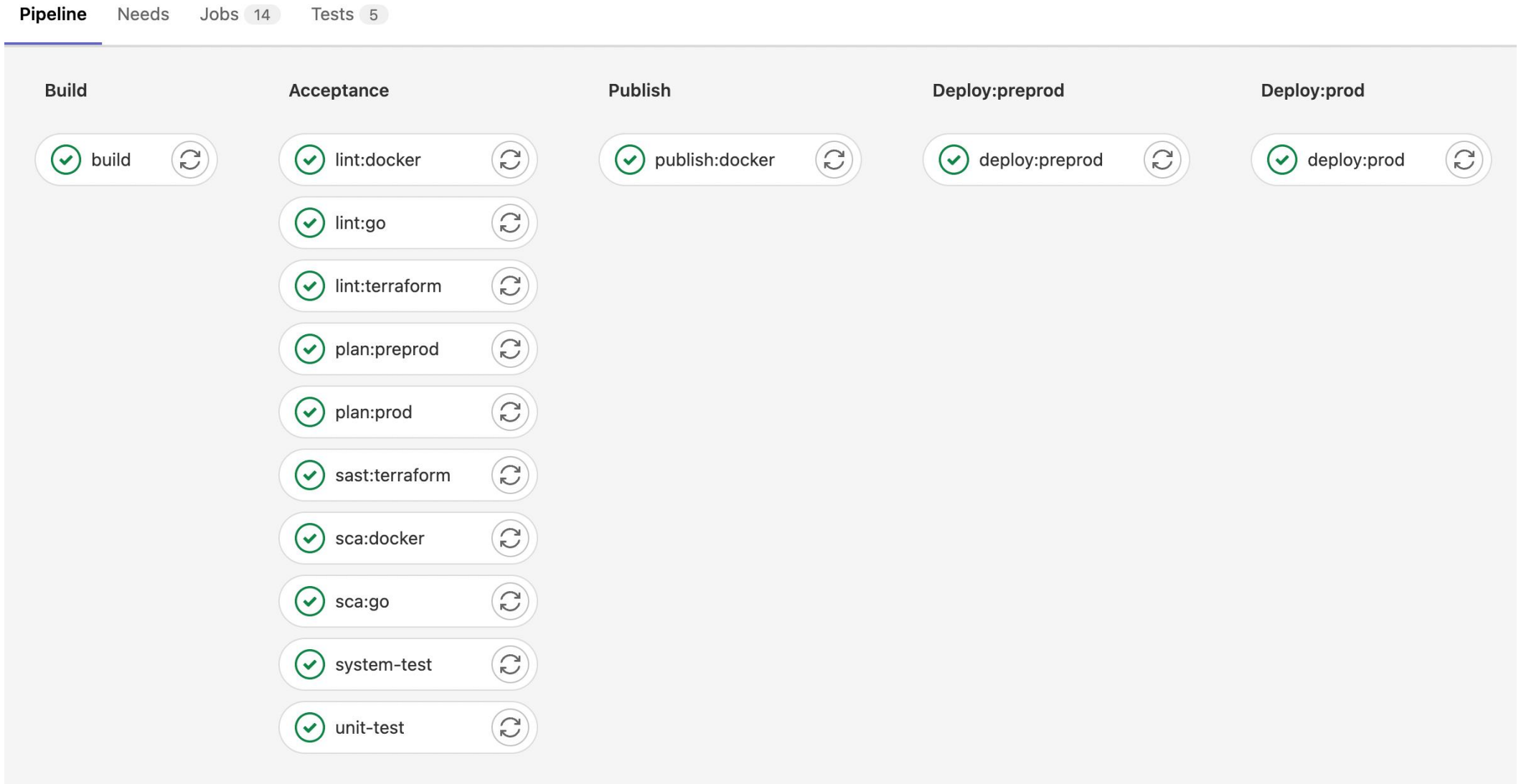
♻ **Rebasing**: Whenever MR is behind base branch, or you tick the rebase/retry checkbox.

👻 **Immortal**: This MR will be recreated if closed unmerged. Get config help if that's undesired.

Go CLI stack


cli
gol-cli-example-cli
Maintainer
Example cli project to test the GOL cli stack.
gol-cli-stack
Maintainer
Components of the gol-cli stack.

A typical deployment pipeline



dlp-devprocess-artio-cli

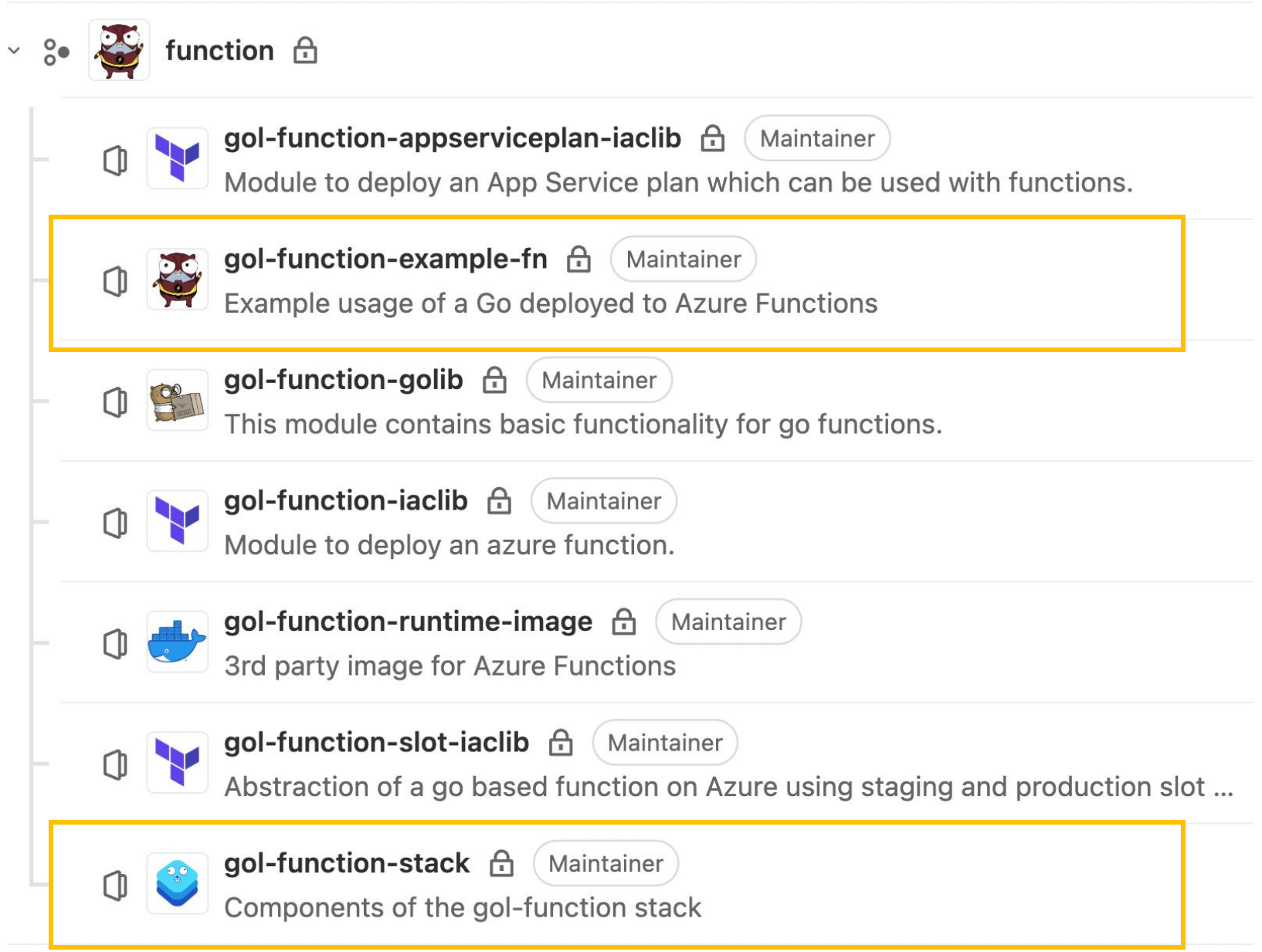
Project ID: 16457290 Leave project

```
Restored session: Fri Sep  2 14:56:27 CEST 2022
~ ❯
```

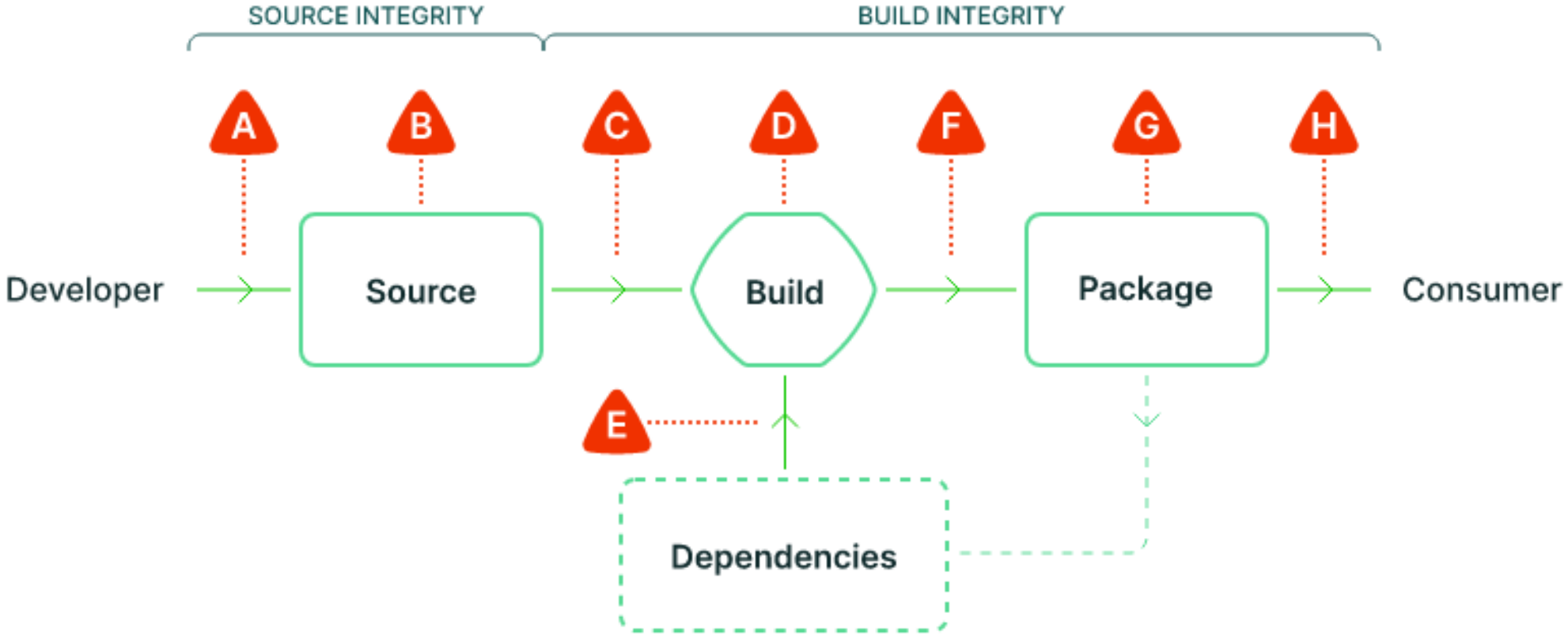
Go CLI projects

Project	Tech Debt	SAST	SCA	Test	Pipeline History (14 days)	Pipeline Overview
cin-terraform-codemods-cli				5%	no data	
cin-terraform-xterrafile-cli					no data	
dlp-cicd-commonscripts-image	1			87%	Build Acceptance Publish	
dlp-cicd-piedpiper-cli				84%	Build Acceptance Publish .post	
dlp-devprocess-artio-cli				86%	Build Acceptance Publish	
dlp-runnerpool-kube-cli	1			90%	Build Acceptance Publish	
gol-cli-example-cli				92%	Build Acceptance Publish	
gol-module-example-cli				100%	Build Acceptance Publish	
had-lab-privatelink-cli				21%	Build Acceptance Publish	

Go function stack



SLSA in a nutshell



A Submit unauthorized change

B Compromise source repo

C Build from modified source

D Compromise build process

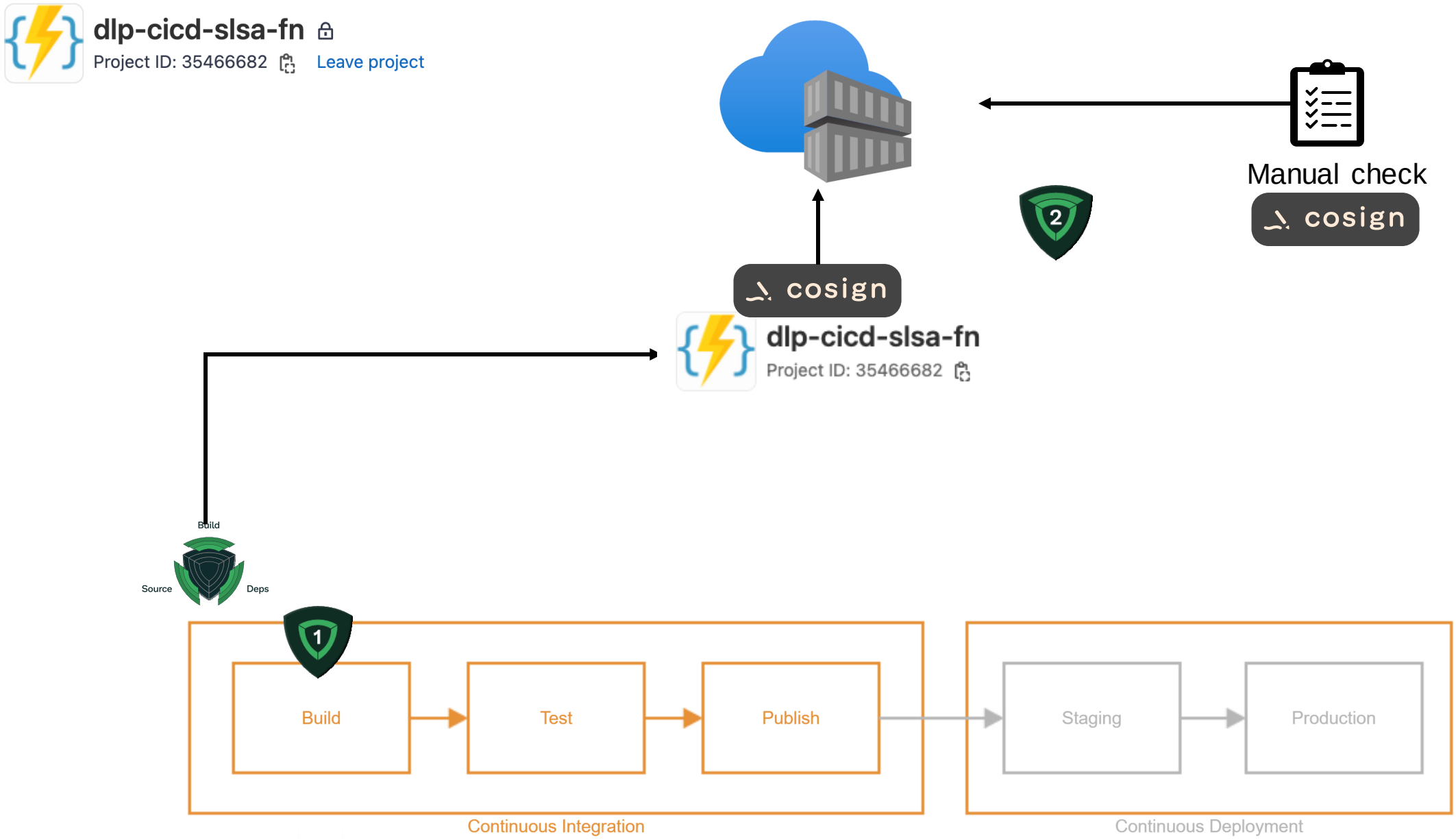
E Use compromised dependency

F Upload modified package

G Compromise package repo

H Use compromised package

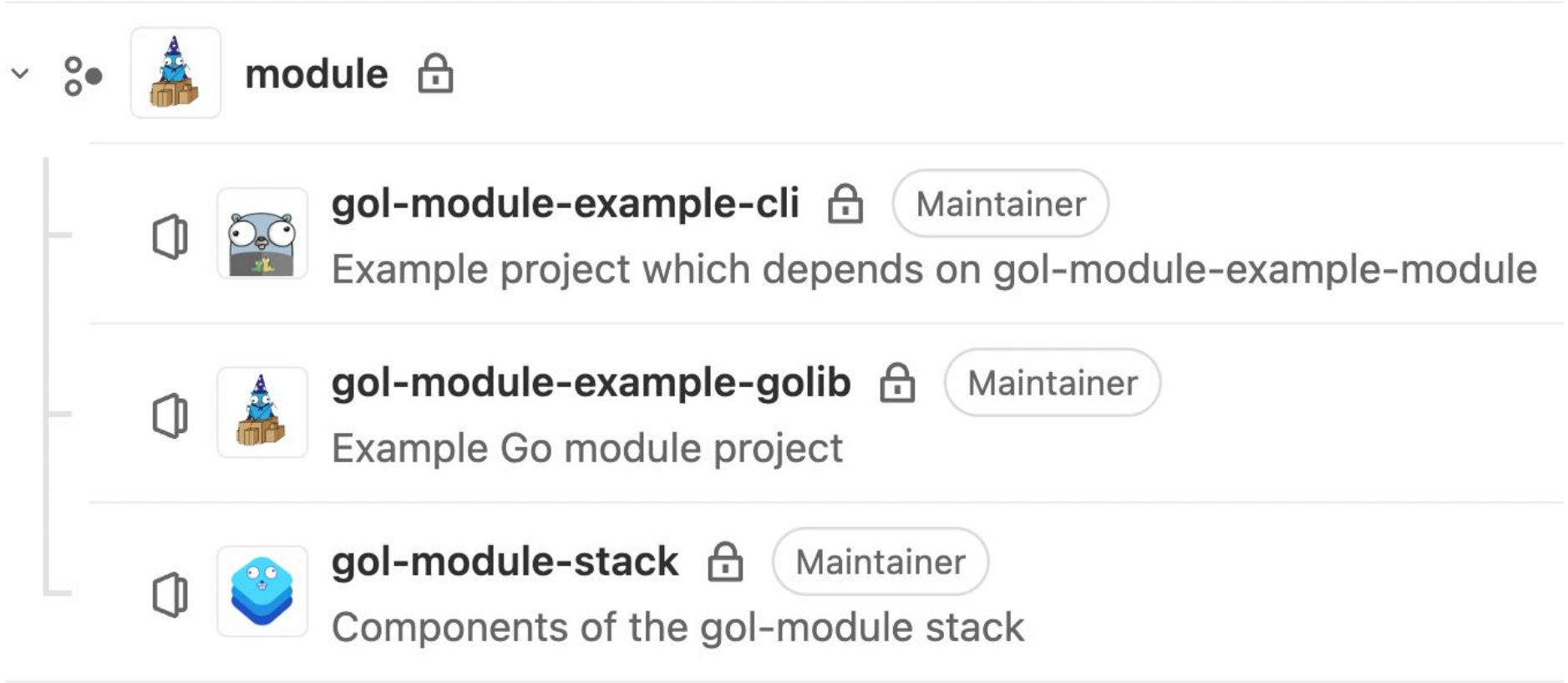
https://slsa.dev/spec/v0.1/threats


dlp-cicd-slsa-fn
Project ID: 35466682
Leave project
Manual check
cosign
2
cosign
dlp-cicd-slsa-fn
Project ID: 35466682
Build
Source
Deps
1
Build
Test
Publish
Staging
Production
Continuous Integration
Continuous Deployment

Go function projects

Project	Tech Debt	SAST	SCA	Test	
cos-alerts-sending-fn	2			83%	Build Acceptance Publish Deploy:preprod Deploy:prod
cos-availability-be-fn	4			96%	Build Acceptance Publish Deploy:preprod Deploy:prod
cos-availability-fe-fn	5			92%	Build Acceptance Publish Deploy:preprod Deploy:prod
cos-availability-fn				46%	no data
dlp-cicd-azureadtest-fn				92%	Build Acceptance Publish Deploy:preprod Deploy:prod
dlp-cicd-slsa-fn	2			83%	Build Acceptance Publish Deploy:preprod Deploy:prod
dlp-cicd-whitesourcecleanup-fn	3			88%	Build Acceptance Publish Deploy:preprod Deploy:prod
dlp-gitlabgov-backup-fn	2			75%	Build Acceptance Publish Deploy:preprod Deploy:prod
dlp-gitlabgov-permission-fn	2			86%	Build Acceptance Publish Deploy:preprod Deploy:prod
dlp-gitlabgov-runner-fn				92%	Build Acceptance Publish Deploy:preprod Deploy:prod
dlp-gitlabgov-settings-fn	1			94%	Build Acceptance Publish Deploy:preprod Deploy:prod
dlp-gitlabgov-validator-fn	1			89%	Build Acceptance Publish Deploy:preprod Deploy:prod
dlp-information-azurelogpersister-fn	1			86%	Build Acceptance Publish Deploy:preprod Deploy:prod
dlp-information-bmcchange-fn	3			87%	Build Acceptance Publish Deploy:preprod Deploy:prod
dlp-information-gitlabcrawler-fn	3			86%	Build Acceptance Publish Deploy:preprod Deploy:prod
dlp-information-gitlabeventlistener-fn	4			88%	Build Acceptance Publish Deploy:preprod Smoke:preprod Deploy:prod Smoke:prod
dlp-information-projectmetadatagen-fn	2			83%	Build Acceptance Publish Deploy:preprod Deploy:prod
dlp-information-runnerlogger-fn	2			98%	Build Acceptance Publish Deploy:preprod Deploy:prod
dlp-runnerpool-vmscaler-fn	2			86%	Build Acceptance Publish Deploy Deploy:preprod Deploy:prod
gol-function-example-fn	1			93%	Build Acceptance Publish Deploy:preprod Deploy:prod
mcl-backup-marketingcloud-fn				84%	Build Acceptance Publish Deploy:preprod Deploy:prod

Go module stack



Authentication with Azure in Go

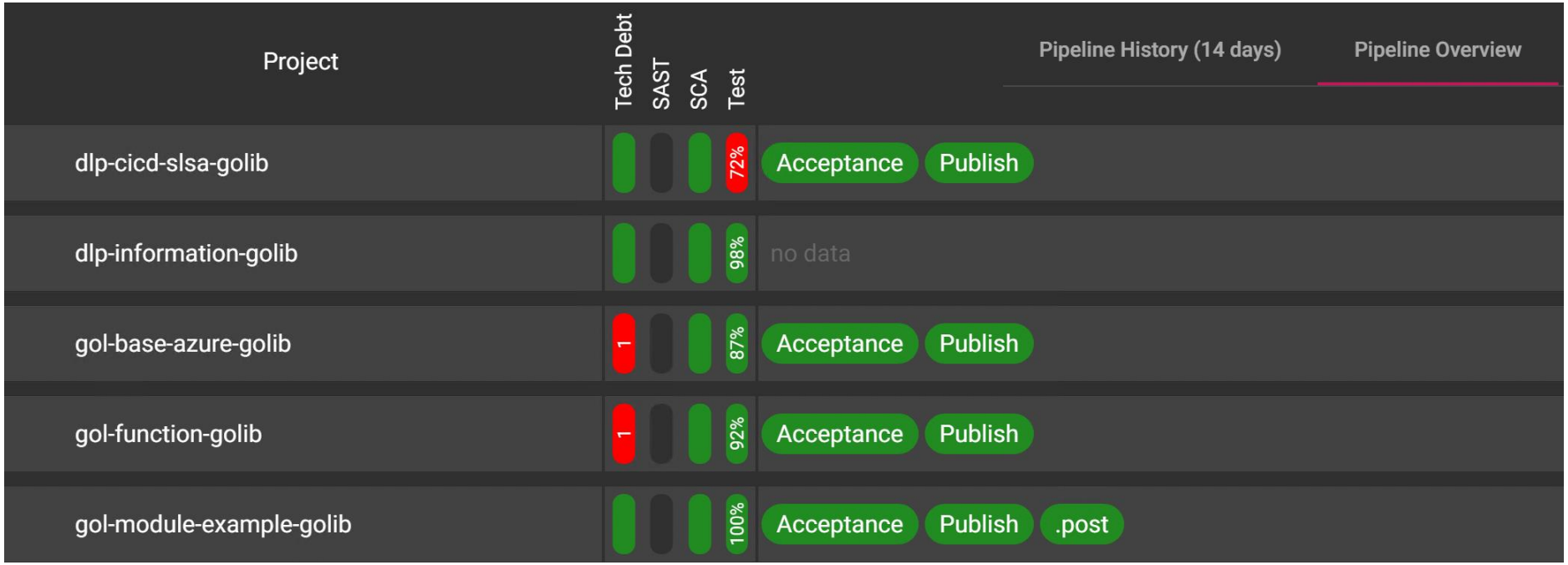
```go
// GetAuthorizer creates an Authorizer configured from environment variables in the order:
// 1. Client credentials
// 2. Client certificate
// 3. Username password
// 4. MSI
func (settings EnvironmentSettings) GetAuthorizer() (autorest.Authorizer, error) {
    //1.Client Credentials
    if c, e := settings.GetClientCredentials(); e == nil {
        logger.Instance.Writeln(logger.LogInfo, message: "EnvironmentSettings.GetAuthorizer() using client secret credentials")
        return c.Authorizer()
    }

    //2. Client Certificate
    if c, e := settings.GetClientCertificate(); e == nil {
        logger.Instance.Writeln(logger.LogInfo, message: "EnvironmentSettings.GetAuthorizer() using client certificate credentials")
        return c.Authorizer()
    }

    //3. Username Password
    if c, e := settings.GetUsernamePassword(); e == nil {
        logger.Instance.Writeln(logger.LogInfo, message: "EnvironmentSettings.GetAuthorizer() using user name/password credentials")
        return c.Authorizer()
    }

    // 4. MSI
    logger.Instance.Writeln(logger.LogInfo, message: "EnvironmentSettings.GetAuthorizer() using MSI authentication")
    return settings.GetMSI().Authorizer()
}
```

gol-base-azure-golib

Project ID: 27165768 Leave project

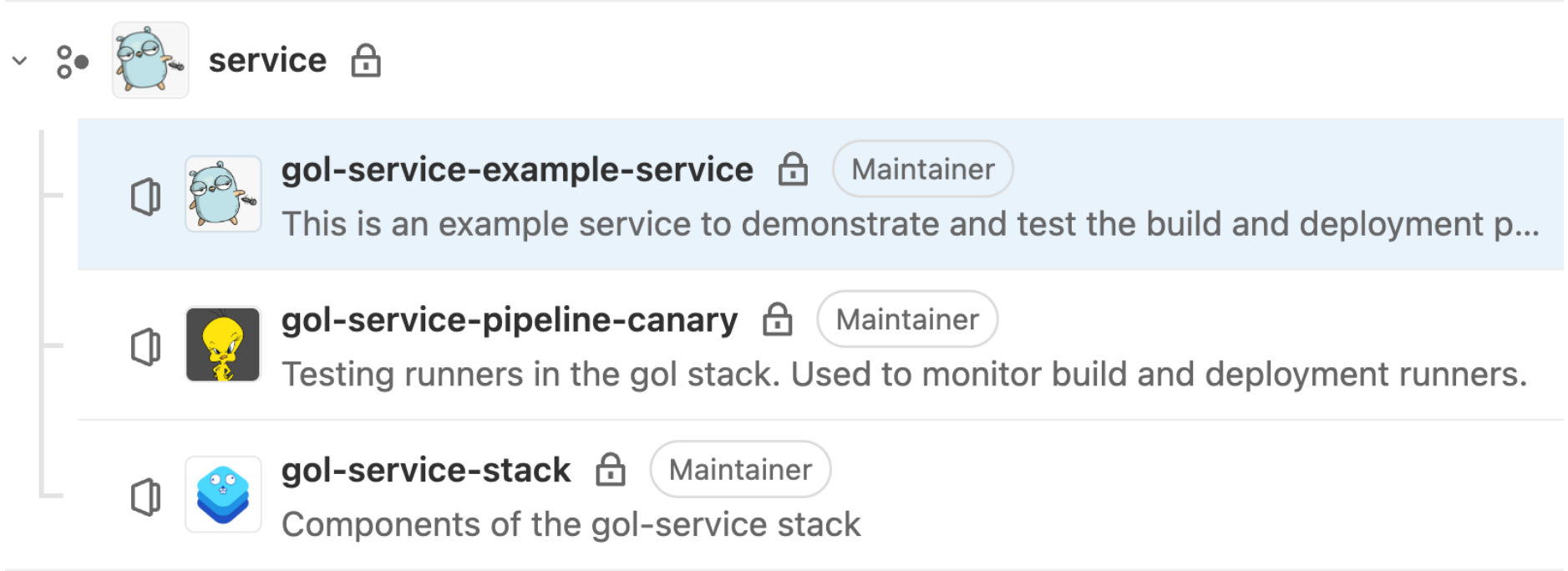
```go
func (c *mobiAuthConfig) GetCredentials() (azcore.TokenCredential, error) {
	if c.cachedCredential == nil {
		err := c.validateConfig()
		if err != nil {
			return nil, err
		}

		if c.UseMsi {
			miOptions := &azidentity.ManagedIdentityCredentialOptions{
				ID: azidentity.ClientID(c.ClientID),
			}
			c.cachedCredential, err = c.adapter.NewManagedIdentityCredential(miOptions)
			return c.cachedCredential, err
		}

		if c.hasClientSecretCredentials() {
			c.cachedCredential, err = c.adapter.NewClientSecretCredential(c.TenantID, c.ClientID, c.ClientSecret, nil)
			return c.cachedCredential, err
		}

		// Here we assume that we are using azure cli credentials or browser interactive login
		azCLI, err := c.adapter.NewAzureCLICredential(nil)
		if err != nil {
			return nil, err
		}

		var fallbackCred azcore.TokenCredential
		if c.allowBrowserLogin {
			fallbackCred, err = c.adapter.NewInteractiveBrowserCredential(nil)
		} else {
			fallbackCred, err = c.adapter.NewDeviceCodeCredential(nil)
		}

		if err != nil {
			return nil, err
		}

		c.cachedCredential, err = c.adapter.NewChainedTokenCredential([]azcore.TokenCredential{azCLI, fallbackCred}, nil)
		if err != nil {
			return nil, err
		}
	}
	return c.cachedCredential, nil
}
```

Go module projects



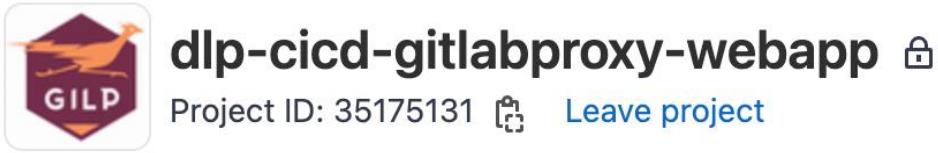
Project	Tech Debt	SAST	SCA	Test	Pipeline History (14 days)	Pipeline Overview
dlp-cicd-slsa-golib				72%	Acceptance Publish	
dlp-information-golib				98%	no data	
gol-base-azure-golib	1			87%	Acceptance Publish	
gol-function-golib	1			92%	Acceptance Publish	
gol-module-example-golib				100%	Acceptance Publish .post	

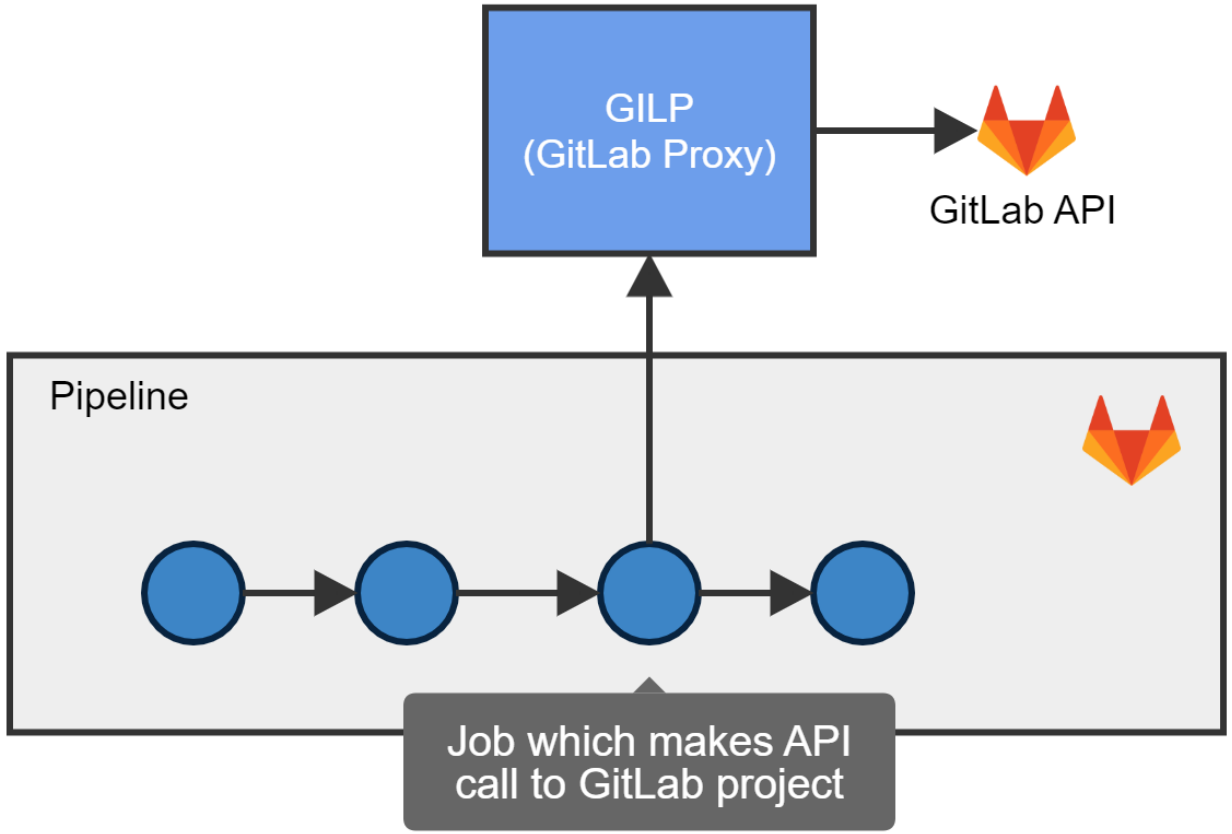
Go service (webapp) stack



go-containerregistry

ORAS


GILP
dlp-cicd-gitlabproxy-webapp
Project ID: 35175131
Leave project


GILP
(GitLab Proxy)
GitLab API
Pipeline
Job which makes API
call to GitLab project

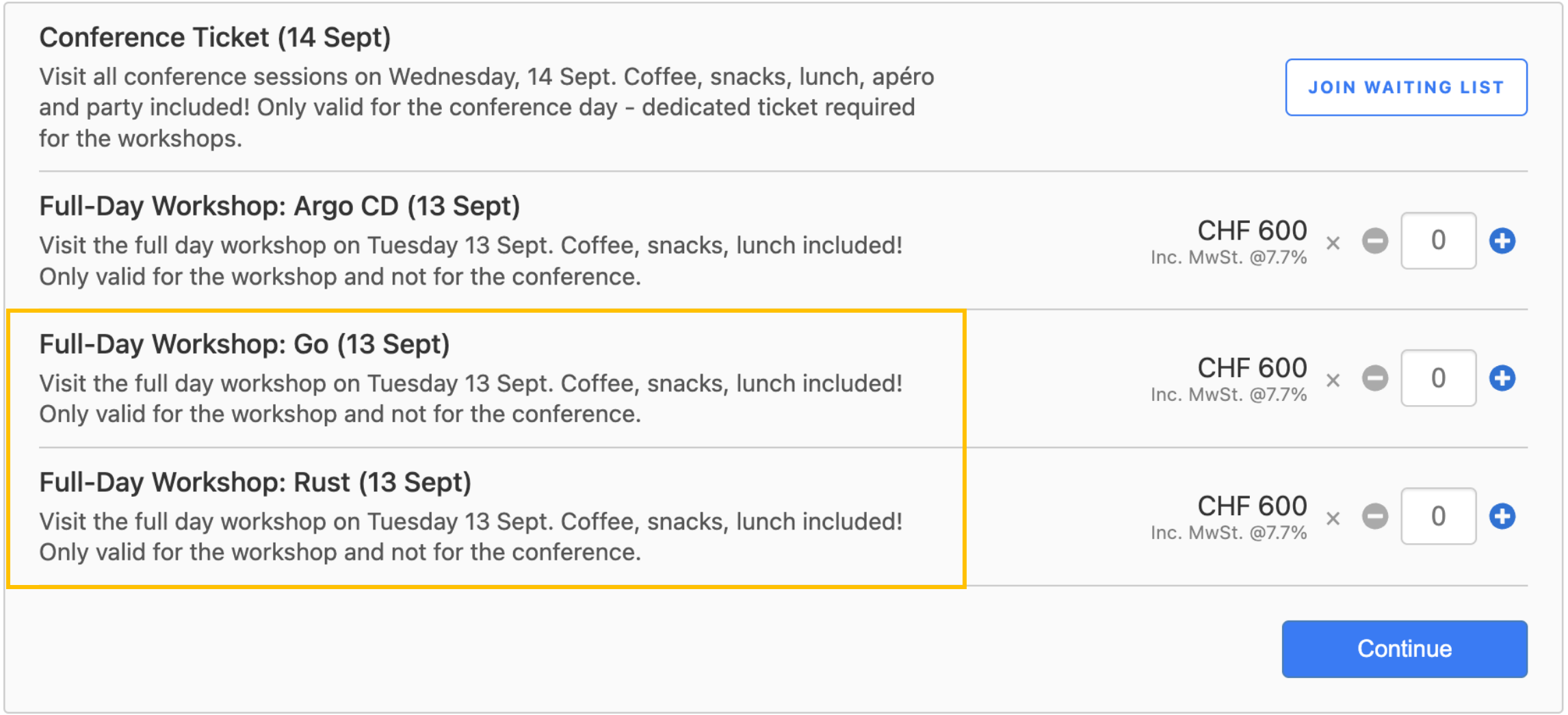
Lessons learned

The stack concept helped introduce Go in a controlled manner

Go is an excellent choice for infrastructure automation

It takes time and practice to become proficient (as for every language)

Questions?

https://www.mobiliar.ch/die-mobiliar/karriere

Go (and Rust) workshops at cloudnativeday.ch :)