

List of CI/CD Changes for Adopting Go Modules

We want to move away from glide and vendoring for dependency management. Utilizing Go modules will allow us to do that but will require changes to the CI/CD pipeline. This document will attempt to capture a list of those changes at a high level.

One thing to note – Where code changes are referenced, assume “edgex-go” as the example repository. These changes will need to be made to any repo that is to integrate with modules, but just be aware that in my mind as I’m writing this, I’m thinking of edgex-go and making it our first use case.

- 1.) Code changes
 - a. Developer needs to create a PR wherein a “go.mod” file containing all of the relevant dependencies, similar to the content in the glide.yaml, is added to the repo.
- 2.) CI Build Agent
 - a. Agent should have Go 1.11.2 minimum installed
 - b. Agent should perform a git clone of the repository to a location *outside of the GOPATH*.
 - i. If for some reason working in the GOPATH is preferable then the following environment variable must be set -- GO111MODULE=on
 - c. Eliminate `make prepare` from Jenkins verify job
 - i. Also in makefile
 - d. Agent executes `make test` as done today
 - i. The combination of the new `go` command plus the go.mod file will cause an import of all relevant dependencies at this time.
 - ii. <https://github.com/golang/go/wiki/Modules>
 1. Standard commands like `go build` or `go test` will automatically add new dependencies as needed to satisfy imports (updating `go.mod` and downloading the new dependencies).
 - e. Agent executes `make build && make docker` as done today
- 3.) Dockerfile changes
 - a. WORKDIR currently points to location in the GOPATH. Change this or else set the GO111MODULE environment variable.
 - b. CONCERN: With the removal of the vendor directory the line `COPY . .` will no longer copy dependencies from the build agent into the docker image. Therefore when a `go build` is performed inside the image, it will cause the module to download its dependencies. This will be done for every docker image in the repo, adding to build times
 - i. Review the following PR for possible mitigation
 - ii. <https://github.com/edgexfoundry/ci-management/pull/261>
- 4.)