

2018-10-23 - golang-tools session - meeting notes

Previous session notes

[2018-10-09 - golang-tools session - meeting notes](#)

Recording

<https://youtu.be/sqvMq8e0yco> (unlisted)

Attendees

- Paul Jolly
- Ian Cottrell
- bhcleek
- Ramya Rao
- Kris Brandow
- bcmills
- Alan Donovan
- Rebecca Stambler
- Zac Bergquist
- Hana Kim

Agenda

- vim-go - welcome **Billie**
- VSCode release update - **Ramya, Rebecca**
 - **Ramya** - did you get answers on best practice for using go list?
- Update on tools: - all
 - Gocode
 - **Rebecca** - gocode issues cleanup
 - Godef
 - **Ian** - godef fixes - tests now merged?
 - Goimports
 - **Heschi** - initial version available
 - Gogetdoc
 - **Ramya/Joe/Zac/Billie** - VSCode switched over to use this?
 - Gotype
 - Ramya last mentioned this was needed in next version of VSCode
 - **Ian** - any update?
 -
- Atom updates
- Vim-go updates
 - Release of module-aware version of tools
 - Context discussion
 - LSP-integration plans
- go/packages - **Michael**
 - Appears to have been released?
- go list speed - **Bryan**
- Module boundaries in x/tools - **Bryan, Ian, Hana**
 - **Bryan** - can we just do tools?
 - **Hana** - any update on your doc?
- go/analysis - **Alan/Ian**
 - Daniel converting his tools as next step

- Paul still not converted his tools
 - Monzo (UK) have been following a similar approach
- LSP update - **Rebecca, Ian**
- Other tools
 - goblin
- AOB

Notes

- Welcome to Billie
 - Taken over from Fatih
 - Bille now up to speed on previous calls, notes etc
- VSCode trial with modules
 - No big news
 - ~900 people trying it out each day (opening a modules-based project)
 - General reports of slowness
 - Ramya has hadn't a chance to ask questions on go list best practice
- Tools update
 - Gocode
 - Rebecca has done gardening on <https://github.com/mdempsky/gocode>
 - No big issues, hence pretty stable
 - Godef
 - Tests now written using the new testing framework
 - Goimports
 - Horrifically slow
 - Using the new testing framework originally written for godef to convert goimports tests
 - Almost all tests passing
 - Particular use case with replace directives uncovered by tests that Heschi is working on
 - Testing framework will be rolled out as generally useful at some point for anyone writing tests that works with Go code
 - Gogetdoc
 - Didn't actually flip to using gogetdoc by default
 - Instead per-issue response on any issues is "use gogetdoc"; wiki status page reflects the same position
 - Speed is the issue the default hasn't been switched yet
 - As this gets more usage we might get more understanding on the speeds issues, but that's not a focus
 - Gotype
 - Ramya confirmed this is not a requirement; focus on LSP
- Atom
 - No update from Joe/Zac
- Vim-go
 - Context: an open question in Vim
 - Billie has ideas on how to solve this, but wants to write tests to help lock this down
 - Also thinking about the UX of modules in Vim - **action Billie**
 - In a Monday call, Billie and Paul discussed how Vim might integrate with an LSP
 - Anything we can learn from the TypeScript plugin (<https://github.com/Quammy/tsuquyomi>) here?
- go/packages
 - Michael away
 - go/packages out of beta into "release"
 - Hence only major things we will see change are:
 - Speed (indirectly via go list)
 - Overlays
- Go list speed

- Russ mentioned that we can make go list an order of magnitude faster by caching directory operations
- Bryan still working through bugs and file locking issues from Go 1.11 so not had a chance to focus on this yet
- Modules boundaries in x/tools
 - In the early days of this discussion, Brad suggested x repos should be versioned along with Go releases
 - Nobody is quite certain how this would work
 - Hence Russ and others hesitant to pull the trigger on creating modules, tagging etc
 - Hana has produced an analysis of the existing setup, problems, options etc - need to find time to review with Russ
 - Holding pattern for now
 - Ian: x/tools definitely not an exemplar at this point in time, despite this analysis
 - Alan:
 - One option was to put each command in its own module
 - Then leave everything else in the top-level module
 - Lots of breaking changes and transition to get to this point... no appetite for that right now
- go/analysis
 - Changes to break up vet into independent plugins
 - Alan in final stages of making the vet tool run in one of two modes:
 - Command line - invokes go/packages -> build system
 - Go vet the build system invoke the tool - run on a per-package basis (hence build system on top)
 - Will allow other binaries to use the tool in exactly the same way, on any build system
 - Change should be out later this week
 - At which point all of this stuff will be vendored in GOROOT
 - Paul: make connection between Monzo and Ian
- LSP
 - No specific update
- AOB
 - Meeting format
 - Still remains useful as a communication point
 - Keep frequency and format
 - People reading notes as a means of keeping up-to-date
 - Ramya will have an update on:
 - Use of go list in VSCode
 - Vendor mode
 - Paul will announce gobin to golang-tools over the coming days
 - Essentially to plug the gaps described in <https://github.com/golang/go/issues/24250> and <https://github.com/golang/go/issues/27653>