

株式会社 ITS MORE

2020年4月設立

ITS more

2020年10月21日 投稿者: SATOXITS

GShell 0.7.2 – go/chan/select入門

開発：ふああ… ああ…

(^_^) 基盤：寝過ぎて脳みそが溶けそうです。

社長：Wikiで吉田拓郎とか読みふけてしまいましたからね。

* * *

開発：ということで、方針を180度転換。CGoは必要でない限り使わない、syscallに関わる部分はパッケージを分けるという方向で行きたいと思います。

開発：その前にちょっとコードの掃除を… GShellの吹き出しの位置とか変ですし。あー、これを書いた時はまだ、addEventListener というのを知らなかったんですね。window.onresize とか書いてます。

社長：それでまとめているんな事を書いてるんで整合性が取れなくなってたんですね。

開発：その後中身は移して空っぽに近かったんですが、吹き出しのとこだけ取り残されてました。

玄関：ぴんぽーん。

開発：こんな時間に何でしょう…

基盤：はいはい。がちゃ。あ、はい。どうもご苦労さまです。がちゃ。

基盤：ゆうパックでした。

開発：えー、こんな時間に。ブラックですねえ。気の毒というか。

社長：まあ霞が関に比べれば。たぶん最近も変わってないんじゃないですかね。

基盤：そういうちはブラックじゃないんでしょうか。

社長：少し赤方偏移してるだけですよ。

開発：あーそれから、9/23に作ったLive HTML Snapshot ってセクションは、どんがらだけ作って details を開いても中身が入ってないように見えたんです。ですが、コードを読むと、ページの先頭にある Save / Load / Vers の事でした。

社長：一ヶ月前の事とか、全く忘れちゃいますね。

基盤：DOMダンプをロードした時に event listener を回復するところが未実装みたいです。

開発：まあCSSOMのダンプとリカバリも必要ですしね。そのほかもなんか怪しいです。たぶん翌日にやってるフレーム間のイベント転送に興味が移ってしまったのでしよう。

基盤：9/27に WirtualDesktopを始めて、10月の最初の1週間それ関係ですね。

開発：あれはこれまでで一番長時間続けたワークだったと思います。

社長：今日はお掃除で終了ですかね。

開発：いや、CGo部分を切除するところまではやりたいかなと。

* * *

社長：で、そもそもこのCGo部分はなんであるかということですが。

開発：もとは、8/27のGshell 0.2.3で、IMEの入力をリプレイ表示しようということで、入力を実際のキーボードから取るかリプレイ用のキューから取るかを選択するためでした。でもその時はsyscall.Selectで実装している。で、パッケージのドキュメントと実装が整合してないとかぶうぶう書いています。

社長：だから、実装依存なビットマップを使うSelectじゃなくてPollのほうがいいと。

開発：実際この関係で、macOSでは動くけどLinuxでは動かないという状態だったと思います。で、9/6のGShell 0.3.3で、CGo版のpollを使って実装して解決した。けれど、10/18のGShell 0.7.1で、Windows対応をやってみたところ、MinGWでpollのサポートが不明ということになりました。

開発：そのへんは適当にやり過ぎて、他のsyscall関係の関数をCGoに移し始めたのですが、まてよこれはやはり違うなど。

社長：CGoもsyscallも使わない解決法…

開発：要するにユーザプログラムでpollするより、Goルーチンで並列待ちするとか、chanelをswitchで選択するのが、Go的には正しい道なのでは無いかと思うんです。そっちのほうがプログラム構造が明示的になるし、たぶん性能もそちらの方が出るんじゃないかと。

* * *

開発：ふうう… できました。

社長：けっこう手こずってましたね。

開発：いや、Goroutineの間の並列実行のスケジューリングの意味がわからなくて。

社長：普通のスレッドと同じでは無いんですか？

開発：いえ、極めて普通であることがわかりました。私の単なる思い違いです。これまでの実装では、コマンドのリプレイをタイミング付きでpipeに送る処理 go a => pipe を起動した後に、その入力を pipe => b という形で取り出すという形で逐次実行していたのですが、これを goroutine と chanel に置き換えた際に、 go a -> chan & chan -

> b と起動した (つमりの) 後に a の終了を待つようにしたら b が始まらない。なぜだと思ったら、b を起動してなかった。つまりもともと b は a が送り込む入力を消費する普通のコマンド行処理で実行するように作ってあったからです。ただ、a の終了を待つようにしたのは、go a -> chan の状態が渡りきらずに chan -> b が実行されてしまうように見えたからです。結局、実際 go a -> chan が動き始める前に戻って来てしまっていたという問題があって、ここに runtime.Gosched() を入れたら解決しました。

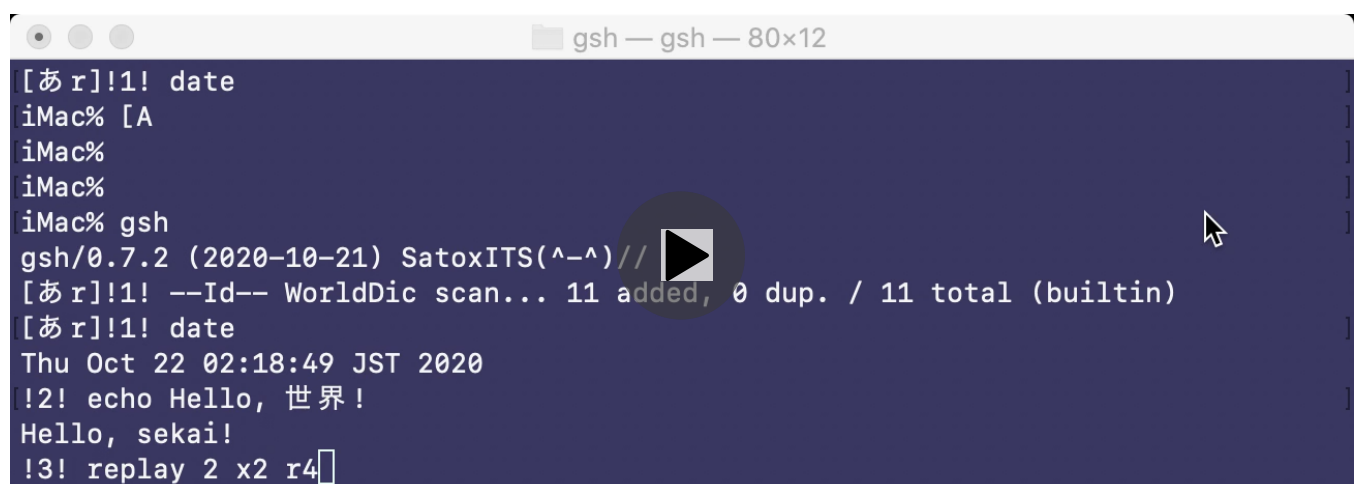
開発：これで syscall.Select/Poll だけでなく、syscall.Read/Write の使用も除外できましたので、一石二鳥でした。

社長：やはり言語がプログラムの並列構造を知ってて管理してくれるのはうれしいですね。

開発：それもありますし、型の規定されたデータをそのまま、OSの介在無しに送れるチャンネルは記述も処理も軽い。受信側での逐次的処理と並列処理も、selectかswitchかで簡単に選べる。やはり魅力的だなと思いました。Goを使うのにこれを使わないのは、宝の持ち腐れです。

社長：過ちを改むるに遅すぎる事無しでしたね。

開発：では、本日のめに、2ヶ月ぶりの Hello, 世界を。



```
gsh — gsh — 80x12
[あ r]!1! date
iMac% [A
iMac%
iMac%
iMac%
iMac% gsh
gsh/0.7.2 (2020-10-21) SatoxITS(^-^)//
[あ r]!1! --Id-- WorldDic scan... 11 added, 0 dup. / 11 total (builtin)
[あ r]!1! date
Thu Oct 22 02:18:49 JST 2020
!2! echo Hello, 世界!
Hello, sekai!
!3! replay 2 x2 r4
```

基盤：あれ？リプレイする時に[あr]が出てないですね。

開発：ありゃま。明日直します。

-- 2020-1021 SatoxITS

gsh-0.7.2.go [ダウンロード](#)

GShell-0.7.2 by SatoxITS 

UA:Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/14.0 Safari/605.1.15

2020/09/22 02:33:45

2020/09/22 02:33:59

Elapsed: 14.143 s

Skew: 1 ms

GShell version 0.7.2 // 2020-10-21 // SatoxITS



*/ ** /*

▶ Topbar

*/ ** /*

▶ Indexer

*/ ** /*

GShell // a General purpose Shell built on the top of Golang

It is a shell for myself, by myself, of myself. --SatoxITS(^-^)
[prev.](#)

[Edit](#) [Save](#) [Load](#) [Vers](#) 0 [Fork](#) [Stop](#) [Unfold](#)
[Digest](#) [Source](#)

*/ ** /*

▶ Statement

*/ ** /*

- ▶ Features
- */ /*
- ▶ Index
- */ //
- ▶ Go Source
- //
- ▶ Considerations
- // /*
- ▶ References
- */ /*
- ▶ Raw Source



- */ /*
- ▶ GJ Console
- */ /*
- ▶ Form Auto. Filling
- */ /*
- ▶ BlenderText
- */ /* *// //
- ▶ Golang / JavaScript Link
- /*
- ▶ GJ Link
- */ /* *// /*
- ▶ Live HTML Snapshot
- */ /*
- ▶ Event sharing
- */ /* // /*
- ▶ Wirtual Desktop
- */ // // // /*
- ▶ SBSidebar
- */ // // // // /*
- ▶ Affiliate
- */ // // // // /*
- ▶ Font Selector
- */ // // // // // /*
- ▶ Shading Canvas
- */ // // // // // /*
- ▶ Character Map
- */ // // // // // /*
- ▶ Collaborated Pointillism
- */ // // // // // /*

▼ StatCounter

StatCounter

000729 (counter as image tag)

(counter by inline script)

/ // // // // /

▶ Work Template

/ // // // // /

▼ Original Source

Original Source of GShell

**/ // // // //*

//

//