

# 株式会社 ITS MORE

2020年4月設立

ITS more

2020年8月22日 投稿者: SATOXITS

## GShell 0.1.8 – コマンド入力用IME

開発：さてリモートシェル第3日目行きましょうか。

社長：ちょっと待った。さっきから猛烈に、IMEがやりたくなりました。

開発：何故に。

社長：変な名前のファイルができちゃったんで消したくなかった時です。”??”とか。まあこれはエスケープすればいいんでしょうけど。日本語名のファイル名には昔から悩まされています。日本語部分を編集できなくても良いですが、日本語名のディレクトリに移動したいとか、ファイルにちょっとローマ字的にいじりたいとか。

基盤：それはよくありますね。

社長：それで、せっかくなら alias とか ファイル名の展開も、IME という枠組みで統合してやれば良いと思うんです。全部IMEの辞書に入れちゃうわけです。!3 とか入れると3番目のコマンドに置き換えられる。\*.html なんて入れると gsh-0.1.8.go.html になる。

基盤：当たり前のような。

社長：IMEの辞書として統合的に出来る点が新しいのです。

開発：alias とか history とか find の結果をがんがんIMEの辞書に入れる… まあ辞書を永続的な辞書と刹那的な辞書に分ける必要はありますね。

社長：優先度的な属性でも良いと思いますね。あと刹那というのは再来する文脈だと思うんです。消えてしまうものではなくて、時々そういう文脈になる。文脈毎に辞書を作るか、辞書項目に文脈の目印を入れると良いのかなと思います。

開発：ならいっそIMEで、ファイル名展開の構文で生成もできると良いかもしれませんね。あとふうIMEでは複数の候補から一つを選択して確定するわけですが、複数の候補をそのまま展開して確定ちゃうという変換方法がないといけません。

社長：うーん、面白そう。

開発：変換については、実際に置き換えるか、表示モードで切り替えるか、選べると良いかと思います。表示モード方式は、編集はコンパクトで、表示は展開してっていう形には利点があります。

社長：それ、プログラムのIDEでできるといいなあ。マクロを展開したらどうなるか、その場で見たいことって多いですね。

基盤：ついにIDEにも手を出すと。

社長：いや、ただのエディタでもビューアでも良いのですが… とにかく、複数の表示方法を切り替えられることが重要と思います。複数のビュー。HTMLで言えば最終的な表示とソースコードビュー。

開発：ただ、現状、入力用のラインエディタ `getline.c` は2週間前にGshellを始めた時にCで書いたもので、これをGoにしないといけないですね。まああの当時はGoのことがほとんどわからなかったわけですが、今ならすんなり書けるかな。ちっちゃなプログラムだし。stty は Unix のコマンドでやっちゃえばOKだし。

社長：そうしましょう。あー、でも今日は外食に行きたいです。飯食って飲んで鋭気を養ってきます。

\* \* \*

開発：あれ？目が醒めたらこんな時間に…

社長：よく寝ました。このところ飲んでなかったから酒に弱くなりましたかね。今日見かけた印象的な建造物。



基盤：なにか至近距離で花火が打ち上がってます。

経理：市から通知書が来てますね。

社長：どれどれログイン。おー、振込\*キユウフキンツクバシトクベツテイガ 入100,000円とな！

開発：口座名がカタカナでしかも大きいひらがなだけってなんとかならないんですかね？もともと濁点別文字だとすると20文字で打ち切り。もう何十年もこの状態。

基盤：キヤノンみたいな。

開発：ウエルシアなんていう表記になっちゃてるのもそのせいですかね。

社長：半角カナにすら小さいカタカナはありますから、それよりもっと前に決められた文字コードを使ってるんじゃないですかね。

基盤：でもWikiには銀行もこれを使ってるって書いてますけどね。

#### 半角カナ

金融機関のシステムにコンピュータが導入された当時は高速なコンピュータもネットワーク網もない時代で、日本のコンピュータ全体が、情報量が少なく済む1バイトの半角カタカナを使っていた。故に、銀行のシステムも半角カタカナで統一され、全国の金融機関に定着していった。後に、コンピュータの進化とともに全角文字が使えるようになったものの、一度莫大な資金を投入して出来上がったシステムは簡単に変更出来ず、現在もシステムが未熟な金融機関に合わせ、共通する形式は半角カタカナのままであるケースが多い。

開発：ハンコ文化が廃れないのと同じ理由ですかね。

社長：識別子は口座番号ですしね。名前の無い電話番号よりマシなのでは。

開発：そういえば法人の識別子として法人番号ができたのって割と最近みたいですね。

基盤：人間番号より後でしたっけ？

\* \* \*

開発：さてさて、ようやく思い腰を上げてラインエディタを Go で書き直しました。

社長：手間がかかりましたか？

開発：うーん、もとが500行程度のCのプログラムだったわけです。Goのプログラムとして通るようになるまでは、ほぼ機械的な書き換え作業ですので、1時間程度で終わりました。基本的にはCの標準的な関数、stdioとstring関係の関数、あとは system() ですが、これの互換関数的なものを定義してやりました。

```
func sscanf(str StrBuff, fmts string, params ...interface{}) (int) {
    fmt.Sscanf(string(str[0:strlen(str)]), fmts, params...)
    return 0
}
func fprintf(fp*os.File, fmts string, params ...interface{}) (int) {
    fmt.Fprintf(fp, fmts, params...)
    return 0
}
```

開発：で、もとのCのプログラムは最初、ほぼ変更無しでやってました。書き換えたのは func と struct と、for 文くらいですね。これは構文が違うので仕方ありません。ていうかこれ、人間の手作業でやる必要はなくて、機械的に変換できるはずですよ。

基盤：goto 文も残ってますね 😊

社長：Goto が通ったらすごいですね。外のモジュールに飛んじゃうとかw

開発：適切な goto は大変有効な機能ですからね。書きやすく、読みやすく、効率が良い。harmful だなんてとんでもありません。

社長：まああれも、魔女狩りにあった感じですね。今は言語の設計も様変わりしてるのに、当時の悪いイメージがまだ残っている。今も教科書には使っちゃいけないって書いてあるんですかね？

基盤：なんだかカッコ悪いイメージはありますよね。

開発：あのころは玉ねぎ状の言語がカッコいいと思われてましたからね。一体何層になってるんだみ  
たいな。そういう意味ではCはカッコ悪い言語だった。

開発：構文的にひっかかったと言えば、例の switch がデフォで break だというやつです。なんか  
動きが変だなと思ったら。Cでは複数のcase が一つの処理になる場合を良く書くので、あれに相当  
する振る舞いをするキーワードとかあると良いのですが。

社長：goto でやるとか。

開発：それで、最初はそのままでやってたんですが、やはりGoの可変長文字列を使わない手は無い  
ということで、当初は固定長の []byte でやってたのを string に置き換えました。なんと言っても  
書きやすい。ここはCが一番苦手な部分ですし。ということで、もう元のCには戻れないプログラム  
になりました。

社長：まあどのくらいのオーバーヘッドがあるのかわからないですが、何しろ人間の手入力を相手にし  
ている部分ですしね。一桁くらい遅くなっただって構わないでしょう。ミリ秒で処理できればOK。

開発：ということでGoにしてgsh.goに組み込んでしまったので、逆にコマンド行エディタから自由  
にgshの内部状態にアクセスできるようになりました。履歴を戻るとか超簡単になりました。

社長：それで、コマンドラインIME的にはどのような？

開発：移植作業でちょっと疲れてしまったので、簡単なのを一つだけ作りました。いつも不便に思っ  
ていた 10進 / 16進変換支援です。%x/数値/ とすると数値が16進に、%d/数値/ とすると10進に  
変換されます。こんな感じ。まず入力をして。

```
iMac% vi gsh.go
iMac% gsh
!!! echo %x/1024/ %d/0x10000/
```

開発：¥j と押すと変換されます。

```
iMac% vi gsh.go
iMac% gsh
[あr]!!! echo 400 65536
```

基盤：うーん、これは普通に便利。ですが、¥j って、これも日本語変換っていう意味ですかw

開発：とりあえず流用してるだけです。¥ の後に意味を文字はかなり空いているので、色々できると  
思います。で、¥j は表示を変換するだけなので、ここで改行でコマンドを確定すると、こうなっ  
ちやいます。

```
[あr]!1! echo 400 65536
%x/1024/ %d/0x10000/
[あr]!2! █
```

開発：なので、現在表示されている文字列で元の文字列を置き換える ¥i というのを入れました。¥i  
してから改行するとこうなります。

```
iMac% gsh
[あr]!1! echo 400 65536
%x/1024/ %d/0x10000/
!1! echo 400 65536
400 65536
!3! █
```

社長：コントロールキーもファンクションキーも使わないんですね。

開発：その辺はユーザが使いたいならそうバインディングすれば良いと思います。今はハードコーデ  
ィングしていますが、これもコマンドキー用の辞書を作ってそれをインタープリットするようにすれ  
ば良いと思います。

基盤：VNCではコントロールキーがうまく通らなくてひどい目に合いましたか。これで助かりま  
す。

開発：そういうことで、昨日の作業のまとめを終わります。最後に恒例のこれを。

```
!7! echo Hello, sekai!
Hello, sekai!
[あr]!8! echo Hello, 世界!
Hello, sekai!
[あr]!9! echo Hello, 世界!
Hello, 世界!
```

社長：Hello -> こんにちはを辞書に入れとくとよいかもですね 😊

開発：私はここにカンマを入れる作法の意味がよくわからないんです (^-^;

— 2020-0823 SatoxITX

[GShell-0.1.8-by-SatoxITS](#)

ダウンロード