

# 株式会社 ITS MORE

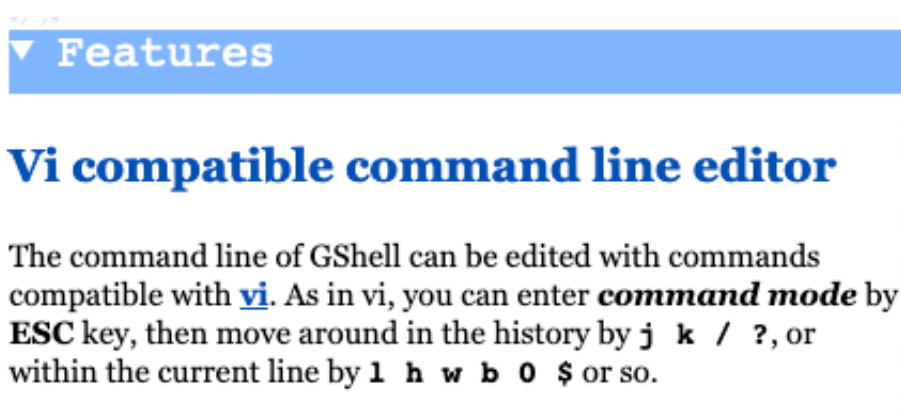
2020年4月設立

ITS more

2020年9月1日 投稿者: SATOXITS

## GShell 0.2.8 - vi互換コマンドIME

社長：vi互換のコマンド行編集機能をGShellの一丁目一番地に掲げたいと思います。  
Featuresの章を作って書いてみました。



基盤：あっさりしてますね。

開発：まあ「ESCでコマンドモードに入る」とだけ伝われば、あとは言わずもがなですからね。

社長：GShell 改め「ViShell」にしようかな…

開発：で、これはそのコマンド入力用IMEという位置づけなんですか？

社長：Comprehensive IMEで CMIE とかどうですかね…

# モードのありがたみ

基盤：なぜそんなにviが好きなんですか。

社長：まあ生まれた時からずっとviで育ったというのはあるんですけどね。なにしろ「モード」があるという事に尽きると思います。

開発：文章でもプログラムでも何度も何度も、長期に何年にも渡って読み返しては時々加筆修正するみたいな作業をしていると、ほとんどの時間帯は閲覧モードというか、readonly なんですよ。他人の書いたものとかの場合も多い。

社長：まあ何年か前の自分は他人に近いですからね。

開発：ずっと検索したりブラウズしたりして、よし、ここを変更しようと思った時に、入力モードに入るわけです。入力状態に居るということも明示的に意識される。

開発：これがモードレスだと、間違っってキーを触って文章とかプログラムを壊してしまうという恐ろしさとかストレスを感じるのです。特に文章の場合、コンパイラがチェックしてくれないですからね。

社長：なにしろ生まれつき粗忽ですからね。

社長：だから「繰り返し眺めて直すテキスト編集の作業モデル」と、「入力/コマンドモードというツールのモデル」は、とてもマッチしていると思うのです。人間に課す負担が少ない。

開発：まあGUIのIDEだと、入力する時以外はテキスト画面内でキーを打たないことが多いですけどね。これは vi の時代にはなかったモードです。でも、やっぱり間違っって何かキーを触って壊すことはあるわけです。

社長：Wordとかの文章入力ツールの「閲覧モード」とはか全然ハズレてるんですよ。単に改行マークとか外して readonly にしてくれるモードがあればよいのですが。

開発：ブラウザの画面内で / とか n とか g:/…/p なんて出来ると良いんですけどね。

社長：それはぜひ、gsh.go.html でやりましょう。

## キーの押し易さ

社長：それから、コマンドモードが入力モードと別れていると、普通は入力にされてしまうキー、打ち易い普通のアルファベットキーたちですね、をコマンドに使えるというメリットがあります。

社長：とにかく私はもう、2つのキーを同時に押すということ自体をストレスに感じているのです。Shiftキーだけは仕方が無いかなと思いますが。ほかのキーは、理想的には、全部やめたい。

開発：Alt だ Command だともう、キートップを引っこ抜きたいですね。昔はよく、CapsLock を引っこ抜いてましたけどw

基盤：入力モードとコマンドモードの他にはモードはあるんですか？

開発：・・・「:コマンド」が相当しますかね。行コマンドモードというか。

社長：そのへんは、シェルの入力部分としては、別のモードがあって良いと思いますね。

## 非英数字の入力

開発：本気でIMEをやるなら、IMEとしてのモードは導入せざるを得ないでしょうね。

基盤：モードと言えば、ローマ字入力じゃなくてひらがなの直接入力はやらないんですか？

開発：・・・それはありかも知れませんが。ローマ字辞書で統一計画が・・・w

基盤：かなで入力してローマ字にマッピングしてから辞書で変換するとか。

社長：日本語入力ではほとんどの人が、直接その語の文字（ひらがな）を入力するよりも、ローマ字かなでの変換入力を好んで選択してるんですよね。少なくともキーボード入力では。だから、他の言語の人たちもそうすればよいのではないかと思うことが多いです。

開発：そうすればハードウェア的にやたら沢山種類のある各国語キーボードもいらなくなりますね。

基盤：英字数字以外のキー、「:」と「;」の位置すら一定でないのにはめまいがします。

開発：案外「記号の入力」というのが、このIMEの生きる道かも知れませんね。

## 第3のモード

社長：「: コマンド」は設定画面への遷移にも使われるわけで、これは GShell でも設定モードということになるでしょうかね。

開発：GShell としてそういうコマンドがあれば良いのかなと思います。このへんが、テキストエディタが本業である vi とは違うのではないかと。それに、設定はやはり HTTP でブラウザからやりたいですね。GShell 自体が HTTP サーバになって。

社長：でも、キーボードだけでやりたいことも多いですよね。特に簡単な設定なら。だから、設定用の仮想的なファイルとか、コマンドヒストリとか、:g/…/pの結果とか、コマンドの実行結果を表す仮想的なファイル名を :e できると良いのかなと思います。設定ファイルでは、フォームみたいのが開く。

開発：松の第0ページでしたっけ、あれは原体験です。モードじゃなくて、あくまでページという。

基盤：モードは入力モードとコマンドモードだけ、あとは操作対象で操作の意味が変わ

るというのはスッキリしていると思います。

開発：プラス、非ASCII文字列用のIMEのモードというのは、避けられないと思います。onewはどうなっていましたっけ？

社長：あれは、Wnn / Canna の提供する入力バッファの編集状態的なモードだったと思いますね。自前のモードというより。

社長：ともかく、簡単な編集機能だけで良いので、フルスクリーンのエディタができると良いですね。フォーム入力対応の。

開発：さすがにそれは、curses 的なライブラリがないとキツイですねえ。

\* \* \*

基盤：夢は終わり無く始まる～♪

開発：この作詞者天才ですかね。

社長：でもこの人的な表現というか文体を作る翻訳機はできそうな気がします。

開発：アニメソングの面白いのはこういう直球メッセージを臆面も無く書けるんですよね。

社長：でもプリキュアを見るような世代にこれが通じてるんですかね？

\* \* \*

基盤：あれ？これ昨日買ってきた蒸しダコ、食べ忘れて室温で一日放置されてましたね。

社長：しまった！

開発：ちょっと食べてみる。うーん、匂いも味もヤバい感じはしませんが…

基盤：レンジでチンしては？

開発：ナイスアイデア。では70度で1分加熱殺菌・・・

レンジ：ゴー… パピペ、パピペ、…

開発：どれどれ。もぐもぐ。うん、うまい。あーでもちょっと不安だからもう1分。

レンジ：ゴー… パピペ、パピペ、…

開発：あーあ、タコから旨い汁が出ちゃってもったいない… もぐもぐ。味が薄くなっちゃった。

基盤：もぐもぐ。悲しい… もぐもぐ。

社長：もぐもぐ。これ、バター炒めにしたら良かったですね。出た汁が粘って絡みつくまで。

開発：お好み焼き屋でやるあれは最高ですよ。

基盤：今回はそれで対処しましょう。

## 単語単位の移動と編集

開発：vi モードでの単語単位のカーソル移動、超刹那的プログラミングにて完成しました。

社長：分かち書きの言語は単純ですが、日本語文字列の中での単語単位の移動っていうのは、深淵そうな課題ですね。

基盤：近似的には、次の非かな文字へのジャンプってことになりますかね。ああでも、かなだけの単語もありますしね。

社長：本気IMEでは形態素解析とかやるんでしょうね。でも、とりあえず漢字・ひらかな・カタカナの区切りを目印で良いと思います。

開発：といたしますか、そもそも内部データがローマ字状態の文字列で、変換表示しているモードではどうするか、なんですが。まあ単語検索は変換された結果のほうでやって、ローマ字表記での位置合わせはそこから逆算する式ですかね。

社長：cw とか dw とか yw がハイライトになりそうな気がします。

## 複数行対応

社長：vi でめっちゃ使うコマンドと言えば % ですよ。

開発：複数行でやりたいところですが…

社長：本式のスクリーンエディタじゃなくて良いので、数行のエディタをやってみてはどうですかね。上の行に移動するエスケープシーケンスだけでできそうですが。

開発：複数行のコマンドラインというのは面白いですね。継続行を理解する。

基盤：普通のshellでも括弧とかクオーテーションでの複数行入力はサポートされてますね。編集もできる。

社長：ほんとだ。編集できますね。知りませんでした…

開発：複数行に対応するとなると、j と k で履歴を前後するというキー割当てとぶつかっちゃいますね。

社長：というか、/ とか n とかもですね。これは複数行かどうかに関わらず。

開発：vi コマンドモードでの検索編集対象を現在の行の中に限るというモードが必要ということですかね。

社長：そうするしかないように思います。でも、そうめったにやることではないですから、この複雑化は受容できるんじゃないでしょうか。

基盤：b とか w でも履歴間を移動するというのアリでは無いでしょうか？

開発： . . .

社長：ヒストリは全部行単位で保管して、この行は何番目のコマンドの何番目の行、という形で扱うするのが良いかもしれないですね。

開発： . . .

開発：要するに、ずっとベタな一枚のベタテキストを入力している。ある行からある行までを折りたたんで表示したり扱うモードと、展開して編集するモード。という感じですかね。ヒストリはそのべたテキストへのインデックスであると。

社長：なんか `vin` のリスト表示の編集機能のようです。たしか、`c` で `catenate`、`i` で `unfold` だったと思います。

開発：で、複数のコマンド行をどのようにグループ化するかは後付けで追加できる。

社長：一行のコマンドの行内を編集するときと、複数行コマンドの中の一行を編集する時は、同じ見え方で同じ操作で。これがいいなと思います。

開発：ちょっと以前作ったヒストリ管理の部分をいじる必要はありますが、まあできると思います。たぶん、より簡単になる。

## 連続サーチモード

開発：一般的にテキストに向かっている時に、検索モードという作業状態というか心理状態があるように思います。基本的には自分では入力しない。検索する。検索した結果をコピペしたり再実行する。シェルのユーザにおいても、これはあると思います。

基盤：矢印キーで選んでは実行、という繰り返し状態はよくありますね。

開発：矢印キーは、入力モードにおいてもコマンドとして解釈されるので良いのですが、`jk` はデータキーになってしまうという問題があります。要は、一旦 `vi` コマンドモードに入った後の状態ですね。`jk` で選んで `RETURN` で実行する。その後に、`vi` コマンドが継続しているべきかいなか。



開発：何が言いたいかというと、vi コマンドモードに入って、? や / でコマンドを検索して、実行する。その後に、再び? や / で検索したいときに、vi コマンドが継続していてくれたほうが便利なのだと思います。

社長：実際、ある程度作業してヒストリが溜まったら、過去の入力の中から検索しては、部分的に編集して、再実行することは多いですね。それがやりやすいならなおさらそうなると思います。アドホックなようなコマンドラインですが、文章書きと違って、それほどクリエイティブに入力をしているわけではない。

開発：そういうケースを考えると、デフォが vi コマンドモードであるとか、サーチモードであるっていうのはアリかなと思います。コマンドを実行し終わったら、次のコマンド入力を求めるのではなく、検索文字列を求めるという。

基盤：メニュー形式のようなものですね。

社長：検索サイトが常に、検索モードであるみたいのと似てるかもですね。

開発：・・・

開発：表示形式そのものをメニューにする、というのがあっても良いかも知れませんね。過去に実行したコマンドからメニューを作る。その中から選ぶだけのモード。

## 包括的検索コマンド

社長：検索文字列そのものも、過去の検索文字列から検索できるのが良いですね。

開発：それはやりたいと思っていました。たぶん、検索モードというかサブコマンド状態になるんだと思いますが、もしフラットなモードでできると、何か面白い事ができるかも知れません。

社長：検索した結果をコマンドラインに呼び出すというコマンドがあったとすれば、それぞれにヒストリに入れておいて、ヒストリ検索で検索コマンドだけ抜き出して、そこから選択するというのが良いように思います。

開発：・・・

開発：つまり、履歴から条件検索した結果としてサブセットを作る。さらにその結果に対して検索で絞る。絞った結果からメニューで選ぶこともできる、みたいな感じですかね。

基盤：ファイル検索の結果が一時的な仮想的ファイルになってたり、履歴検索の結果が仮想的な履歴のビューになっていると良いと思いますね。AND OR 検索なんかもできて。

開発：つまり、履歴検索を包括できるような組み込みの汎用の検索機能があると良いということですね。その包括的な検索機能の適用例の一つとして履歴がある。別の情報の検索結果ともイケイケである。

基盤：IMEでの単語の候補も同じ枠組みでできると良いと思います。

開発：同じ読みの単語を検索すると・・・ まあこれは、辞書の検索コマンドは作ったので、あれと統合すれば良いのかな。ユニバーサル検索というか、コンプリヘンシブ検索というか・・・

基盤：whichでの検索、findでの検索、履歴検索、IME辞書検索、などなどですね。

社長：で、ぜんぶIMEの辞書につっこんじゃうとか。でもって、変換というユーザインタフェースで検索する。あるいは、日付とかで検索する。リソース使用量でソートするとか。

開発：・・・

社長：あと、Q&A形式のインタラクションも、shellでサポートすると良いように思います。

開発：・・・

開発：UIの深みですね。ともかく、/では まず現在の行内を探す、無かったら履歴

の中を探す、ということにしました。仮置きですけど。

<http-im3-gsh-gsh-0.2.8.go>

ダウンロード

```
/*
                                GShell version 0.2.8 // 2020-09-01 // SatoxITS
  ≡GShell   ≡GShell   ≡GS
```

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

It is a shell for myself, by myself, of myself. -SatoxITS(^-^)

0 | | Fork | Stop | Unfold | \*/ /\*

▶ Statement

\*/ /\*

▼ Features

### Vi compatible command line editor

The command line of GShell can be edited with commands compatible with **vi**. As in vi, you can enter **command mode** by **ESC** key, then move around in the history by **j k / ? n N**, or within the current line by **l h f w b 0 \$ %** or so.

\*/ /\*

▶ Index

\*/ //

▶ Go Source

//

▶ Considerations

// /\*

## ▶ References

*\*/ /\**

## ▶ Raw Source

*\*/ /\**



*-> \*//\**