

株式会社 ITS MORE

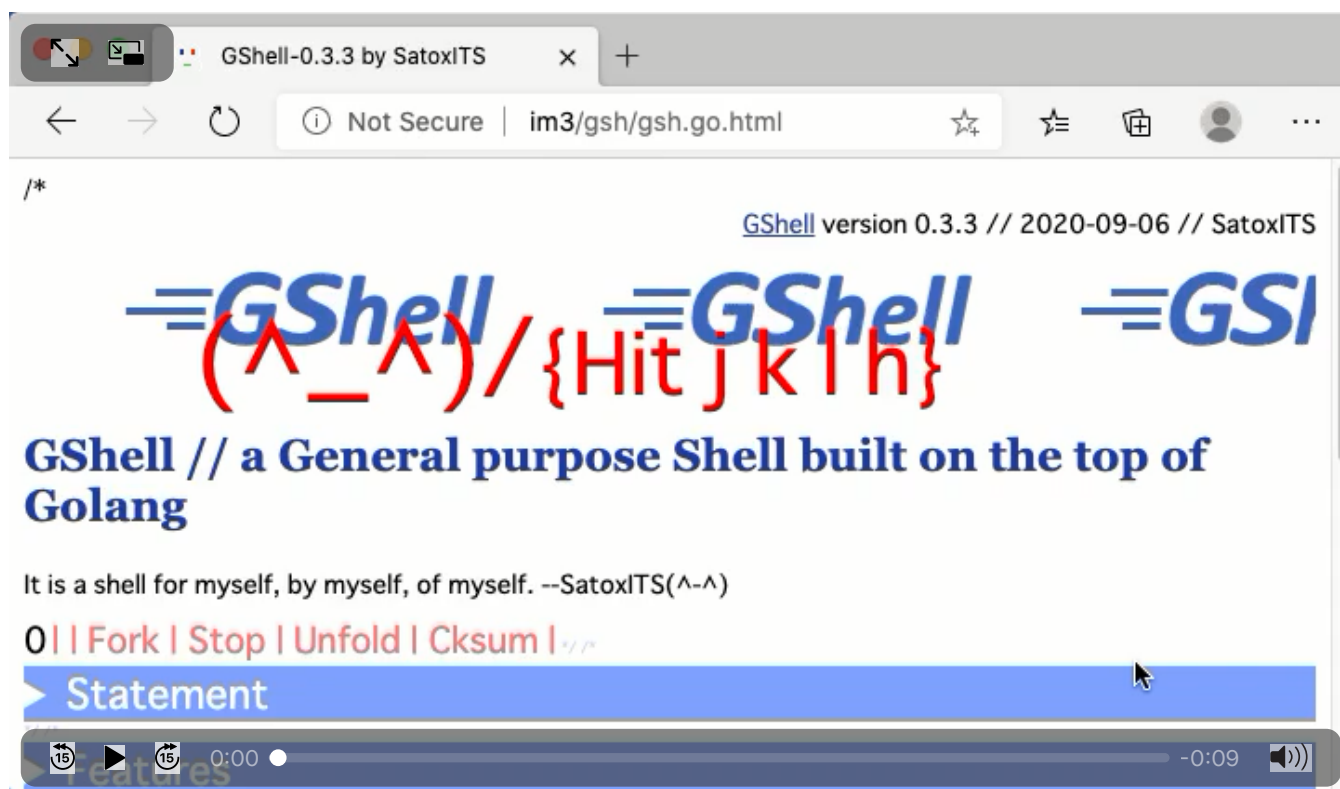
2020年4月設立

ITS more

2020年9月6日 投稿者: SATOXITS

GShell 0.3.3 - HTMLインスペクタ

(^_^)/{Hit j k l h}



社長：「そうだ、Shellを作ろう」の8月7日から今日で満一ヶ月となりました。

開発：この件はかなりみっちりやりました。

基盤：てか、他の案件はほぼ何もやってないですね。

開発：4万カウント祝でWordPressのphpを少々改訂しました。あと、UDPを使った遠隔データ転送高速化をやってますね。まあこれもGShellの一環でした。

開発環境と電力消費

基盤：8/11にはまだエアコン無しで頑張っていましたね。室温34.9度とブログに記録されていますw

開発：熱中症で逝ってたらこのブログ、沈没した潜水艦員のメモになるところでした。

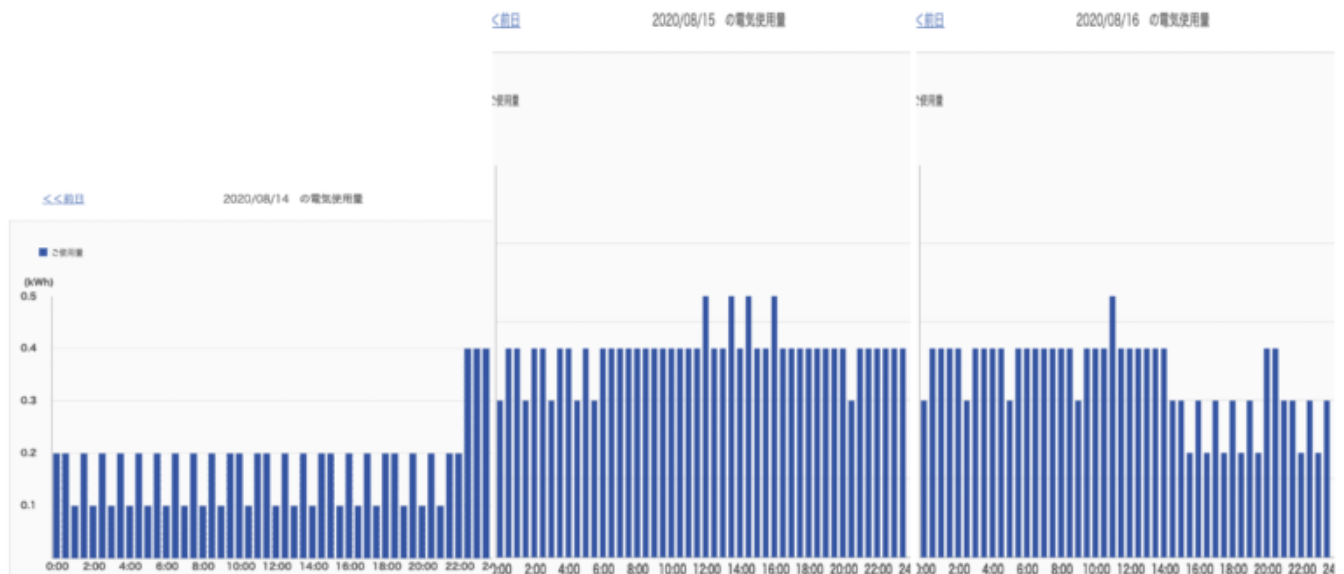
経理：でんこグラフによりますと、冷房を開始したのは8/14夜のようにです。

基盤：この時の決断がなかったら、我々の今はないかもしれませんね。

開発：ITS more、あくなき節電の闘いに倒れる、みたいな。

基盤：ほぼ犬死ですね。

社長：何事も囚われすぎは良くないです。時々強制的に吹っ切らないと。



基盤：これって、ふつうに1時間刻みで、期間設定できるグラフ表示とかないんですか？あるいはCSVで落とせるとか。

経理：さあ。

開発：でんこちゃんも、パートナーに恵まれてないのかも知れませんか。

社長：でも、棒の色は私の好みです。

開発：機微なデータを保護したいという名目なのかも知れませんが、だったらちゃんと電子証明書で顧客を認証してよねって思います。

基盤：アカウントの登録とかパスワード管理とか、余計な責任と運用コストを背負い込んでますよね。無駄な仕事で支配されている。そのせいで遅い、高い、まずい。

開発：まあ、これも雇用対策の一環なのかも知れませんが。

社長：ともかく、似たようなサービスを各社のサイトが独創的に作るのは全くナンセンスだと思います。単に、機微なデータを電子証明書・暗号で守る技術と、データをブラウザ上でグラフ化するアプリが標準パッケージになってれば、同じ使い勝手に、どこへ行って何を見るにも、早い、速い、易い、うまいとなると思うのですが。

開発：ローカルなツールはもっと多様ですよ。結局、コマンドを解析する、データを収集する、ビジュアルに表示する、変換する、リソースを操作する。共通の枠組みでできるはずですよ。

社長：そういうインフラになるのが目標だとすると、GShell という名前はイマイチなんですよ。

開発：とりあえず名前には囚われないということで。

開発のはじまり

開発：そういえばシステムのパフォーマンスのグラフをブラウザで描く話は、当社の出発点での計画の一つでしたが、未だに未着手です。

社長：macOSのパフォーマンスモニタはふざけてますからね。

基盤：それも今なら Golang GShell HTTPサーバ と、HTML GShell JavaScriptで実現できますよね。

社長：何を技術開発のハブにするべきかが見えるまでに4ヶ月かかりました。

開発：でも、これを始めた時には、1994年3月に DeleGate を始めた時みたいな、確信はなかったですね。これは使えるっ、ていう。

社長：まあ、DeleGateの場合には、本当にすぐに使いたい機能でしたからね。喉から手が出るというか、Gopher のファイアウォールを超えという。作った翌日には所内公開して、使われ始めた。

基盤：どっちにも出発点にモグラが出てきますねw

開発：DeleGateでHTTPを始めたのは開発開始から3週間後なんですよね。当時は、HTTPにはほとんどコンテンツがなくて。NCSA Mozaicのホームページを見て、おーすげーって思いましたっけ。

社長：はじめにHTTPから入ったら、DeleGateはああいう形にはならなかったでしょうね。Gopherのリクエストと応答を書き換えて無理やりプロキシするという。まあプロキシという一般名称もなかったから delegate なわけですが。

開発：DeleGateの場合、作り始めてからずっとコードの変遷が追えるので、それ自体が結構エンタメなんです。GShellについてもそれは念頭にあって、それなりに管理はしてきました。

基盤：一挙手一投足というか、タイムスタンプ付きで1キー入力ごとの記録が残っていると面白いでしょうねw

社長：で、全部のキー入力をタイムスタンプで署名すると。

基盤：ふつうのタイムスタンプにはミリ秒の精度はないんじゃないですかね。キー入力間隔100ms前後ですから、10ms精度では証明してくれないと。

社長：なければ作るだけです。自分で灯りを点けましょう。

開発：なんにしる、DeleGateを始めた時には世の中にほとんど何もなかったから面白苦しかった。コンセプトも部材も。今GShellをやってると、必要な部材は全部そろってるので楽ちんで、面白いだけです。まあでも、調べるのが苦手なので、そこは若干苦しい。

社長：MDNにしてもGolantサイトにしても、必要な情報は全てそこにあるんですが、検索がちょっとだと思います。

開発：それで、昔作った全文検索プログラムを掘り出して使おうと思うのですが。

社長：クローラにしてもエンジンにしても検索表示インターフェイスにしても、まっさらから今の技術で作り直したほうがいいんじゃないでしょうかね。あれは良い思い出として凍結保存。

マキシーのために

基盤：今日も「永遠の嘘をついてくれ」がリピートしてますが。

社長：やっぱ吉田拓郎いいですね。この朴訥としたおっさん臭い歌い方。聴き疲れが来ない。

開発：「君よ永遠の嘘をついてくれ いつまでもたねあかしをしないでくれ」この普遍的なテーマだ

け抽象的に歌ってればいいと思うんですけどね。具体事例の描写が中島みゆきの説明文臭いというか。

基盤：「永遠に嘘をついてくれ」が内容的には適切だと思うんですけど。

開発：詩的じゃないんでしょうね。当たり前すぎる。「永遠の嘘」とか「しじみ72個分のちから」だと、え？ってなってキャッチーですよ。

社長：かぐや姫のマキシーのためにを思い出すようなモチーフですね。はじめに具体例があったんですかね。

基盤：マキシーのために、編曲・吉田拓郎ってありますね。

社長：！！！！

開発：社長が道で倒れて出血しながら帰った日は、この曲を思い出しましたよ。

社長：つくばには坂道が少なく、詞になるような倒れ場所が無いですね。

基盤：マキシーには実在のモデルがいたそうです。

社長：喜多条さんとはこういう関係だったのか…

開発：かぐや姫はデビューからしばらくあのカラーでしたよね。

社長：最初買ったレコードが神田川と心の旅でしたw

基盤：バカなやつだったよ お前は本当に～♪

開発：おかしな連中集めて 自由な 自由なお城を作ろうと～♪

社長：共通性を感じますね。ペンネーム Satoxy にしようかな。

開発：結末が不吉過ぎるでしょうw

経理：マキシーが飲んだ100錠って、ジェネリックだったんでしょうか。

社長：100日ぶんもまとめて処方してくれますかね。

基盤：市販薬じゃないすか。

開発：智恵子が飲んだのは処方されたやつだったと思います。

社長：芸術と違って実用のソフトでは、機能には自分という手強い批判者がいるし、性能という客観尺度があるから外に救いを求める必要無くて楽です。

基盤：おっと、Parallelsアップデート来てるのでやっときましょう。

カーソル移動コマンド検討

開発：最近作戦会議が長くて実作業の時間がほとんど取れないんですが。

社長：でも、やりたいことがさくっとできちゃいますからねえ。

開発：で、基本はやはりエレメント間の移動コマンドかなと思います。

社長：できる限り vi 互換が良いです。

開発：j k は行。h l w b は、ユーザの見た目に見えるテキストレベル、物理的なビューと一致させるのが良いと思います。一方「隣のエレメント」は n:next、p:previous が自然と思います。vi では n は / か ? の繰り返し、p はペーストということになりますが、これは互換にするのは難しいですね。

社長：いつも何かを検索していることにして、n で移動するとか。で、何を検索するかは履歴から選べるとか。同じレベルのノードを検索する、みたいな。

基盤：カギカッコタグ言語は対応する開閉タグの間を跳ぶのが面倒ですね。% で出来ると良いと思います。

開発：オブジェクトレベルの内部表現ではこれはただのノードのツリーで、ソースレベルで見えるタグというものは無いんですよ。

基盤：なら、タグをそのまま表示するモードってというのがないと面白くないと思います。

開発：！

基盤：ていうか、HTMLとして編集するにはそういう表示になりますよね。

社長：まあ、構造エディタで、内部の木構造からテキスト表示を生成するというのは、そういうことです。

開発：具体的にはどうやって。

社長：私の場合には、抽象構文木とソースレベルのトークンの列の間を双方向リンクしちゃったような記憶があります。で、物理的表示レベルの移動は、トークンの列のほうでやっちゃうという単純化。使ってたコンピュータも遅かったし。たぶん今使ってるの、1/1000以下の性能で。

開発：ぱっと見、JavaScript ではそういう機能は提供してないですね。DOMは自在に見えるけど、表示結果から逆にたどれない。今マウスカーソルはどの要素の内部HTMLの何文字目を挿しているかとか。Stackoverflowなんかでも真面目に議論されている。

基盤：まあ一般にうちみたいにふざけて議論している人は少ないでしょう。

開発：そういう橋渡しの構造情報が、表示を生成するブラウザの実装レベルでは存在している可能性があります。JavaScriptに対してそういう情報を提供してくれているとは思えない。

社長：ブラウザのソースはあるので、あれを読めばわかるでしょうね。

基盤：とっかかり的には、hlコマンドが良いように思います。

開発：そうですね。

社長：あとせっかくなのでキーの「長押し」を活用したいですね。長めに単独でShiftを押すとシフトロックとか。

開発：検討しましょう。

Linux版復活

基盤：あと、キー入力プレイのために入れた `syscall.Select` の関係で、Linux では整数型不整合が起きて動かなくなってます。

開発：あれはなんか、そもそもSelectのビットマップの仕様が不明確なんです。調査しましょう。そもそもPollが使えればよいのですが… あれ？Cgoで、ナマポールすればいいってことですかね？

社長：ナマポール… そういえば、あそこの店は今ハッピーアワーかも知れない…

基盤：何かひとつ実作業はしましょう。

開発：uint32 か uint64 だけの違いなんで。そもそも、これを 0.2.5で作った時にドキュメント

と仕様が違ってるじゃんて事でした。Linuxは Golang 上のドキュメントの通りです。

社長：いっそのこと、`#ifdef linux` とかしたいところですね。

開発：やはり Cgo で直接 `poll` を使うのが早道に思えてきました。せっかくの機会だからやってみます。・・・要はデータの型をごまかすだけというか・・・出来ました。

```
// // 2020-0906 added,
// // <a href="https://golang.org/cmd/cgo/">CGO</a>
// #include <poll.h>
// typedef struct { struct pollfd fdv[8]; } pollFdv;
// int pollx(pollFdv *fdv, int nfds, int timeout){
//     return poll(fdv->fdv,nfds,timeout);
// }
import "C"

// // 2020-0906 added,
func CFpollIn1(fp*os.File, timeoutUs int)(ready uintptr){
    var fdv = C.pollFdv{}
    var nfds = 1
    var timeout = timeoutUs/1000

    fdv.fdv[0].fd = C.int(fp.Fd())
    fdv.fdv[0].events = C.POLLIN
    if( 0 < EventRecvFd ){
        fdv.fdv[1].fd = C.int(EventRecvFd)
        fdv.fdv[1].events = C.POLLIN
        nfds += 1
    }
    r := C.pollx(&fdv,C.int(nfds),C.int(timeout))
    if( r <= 0 ){
        return 0
    }
    if (int(fdv.fdv[1].revents) & int(C.POLLIN)) != 0 {
        //fprintf(stderr,"--De-- got Event\n");
        return uintptr(EventFdOffset + fdv.fdv[1].fd)
    }
    if (int(fdv.fdv[0].revents) & int(C.POLLIN)) != 0 {
        return uintptr(NormalFdOffset + fdv.fdv[0].fd)
    }
    return 0
}
```

社長：刹那的プログラミング炸裂ですね。

基盤：もはや何語で書いてあるのかよくわからないというか。

開発：一番手早く出来て見通しが良くてしかも動く。何か問題でも。とりあえず汎用でもきれいである必要ありません。で、まずMacで実行。


```
iMac% gsh
gsh/0.3.3 (2020-09-06) SatoxITS(^-^)/
!1! go version
go version go1.15 darwin/amd64
!2! date
Sun Sep 6 17:31:58 JST 2020
!3! replay
--Ir-- Replay(!2 x1 r1)
!4! date
Sun Sep 6 17:32:02 JST 2020
!5! █
```

開発：次にLinuxで実行。

```
jpl$ gsh
gsh/0.3.3 (2020-09-06) SatoxITS(^-^)/
!1! go version
go version go1.14.4 linux/amd64
!2! date
Sun Sep 6 17:25:22 JST 2020
!3! replay
--Ir-- Replay(!2 x1 r1)
!4! date
Sun Sep 6 17:25:25 JST 2020
!5! mit
```

基盤：確認。

社長：承認。

開発：CGo の練習になりました。

社長：これ、端末入力のリプレイのために作りましたけど、いずれ教習所の教習車方式でも使いたいですね。画面共有ならめ端末共有。

開発：キー入力の共有は問題ないかと思いますが、画面出力を共有するところが問題でしょうね。クライアントによって画面サイズが違う。端末も違うかも知れない。まあ強制すれば良いでしょうけど。

* * *

社長：刹那的とも思いましたが、標準イベント入力という標準的ファイルディスクリプタがあっても良いかもですね。

* * *

開発：何をやろうとしてたのか忘れまして。

基盤：h1 コマンドですね。

社長：見通しを付けてから飲みに行けると酒ウマと思います。

経理：そういえばカスミが300円で宅配サービスするってチラシが入っていました。

基盤：8/21 ~ 9/22の試行ですね。「電話でかんたん注文サービス」。

開発：電話のどこがかんたんなんだ。

基盤：岡田とかじゃなくアマゾンの傘下に入って普通にアマゾン宅配すればいいのに。

開発：これもコロナが背景にあるみたいですね。

社長：このコロナは色んな意味で社会構造を強制的に変えるきっかけになった、と歴史に記録されるかも知れないですね。

カーソル移動コマンド・プロトタイプ1

社長：気分を盛りあげるために H1 を打ちました (^-^)/ Go !

基盤：プリキュアもかけます。

開発：そもそもDOMにおいて、「現在の位置」という概念があるか、というところからですが、多分無いんだと思います。ユーザが定義するしか無い。定義して表示するには、そもそもの表示スタイルを色カタチ的にスタンドアウトさせるか、カーソルのように表示される何かを挿入するか、上空にマウスカーソルを誘致するか。くらいでしょうか。

基盤：マウスカーソルを引きずらないモードがあって欲しいですね。

開発：となると、文字単位のカーソル的なもの…素朴に、一文字移動するたびに、文字列の中に挿入する|みたいなの挿入位置をずらして行くみたいなの。

基盤：過激。

開発：処理の負荷的には大したことないと思います。

社長：その方式のカーソルだと、textarea の中では動かさなそうですね。

開発：たぶん。

社長：CSSでカーソルマークのついた透明シートを被せれば良いようにも思いますが。表示開始位置をずらしていく。z-indexとかで上にかぶせる。

開発：・・・ それはありですね。ですが、指している先との対応付けをどうやったものか…

社長：そもそも編集モードでは、固定ピッチのフォントにするとか言うのもありかと思います。

基盤：とりあえずグリッドみたいなのをかぶせて j k h l で動かせると面白いですね。

開発：・・・では今日の作業の計画変更。グリッドみたいな半透明プレーンを重ねて表示する。これをjklhで移動させられることとする。メッセージ的なものをそのプレーンに書けるようにする。これで行きたいと思います。

開発：えーと、position で位置を決めて、z-index で重ねると… テストなのでカーソルはでっかく (>_<) とかで… とりあえずモックアップ出来ました。

```
<!--
// 2020-0906 added,
https://developer.mozilla.org/en-US/docs/Web/CSS/z-index
https://developer.mozilla.org/en-US/docs/Web/CSS/position
-->
<span id="GshGrid">(&gt;_&lt;)</span>
<style id="gsh-style-def">
  #GshGrid {
    z-index:10; //
    opacity:0.0;
    position:absolute; top:50px; left:200px;
    width:10px; height:10px;
    color:#f00; font-size:50pt;
  }
</style>
```



基盤：イイ感じですね。

社長：その z-indexの後の //は何ですか？

開発：構文エラーだと思いますが、たまたまここにコメントを書こうとしたら動いたんです。とりあえず意味不明です。

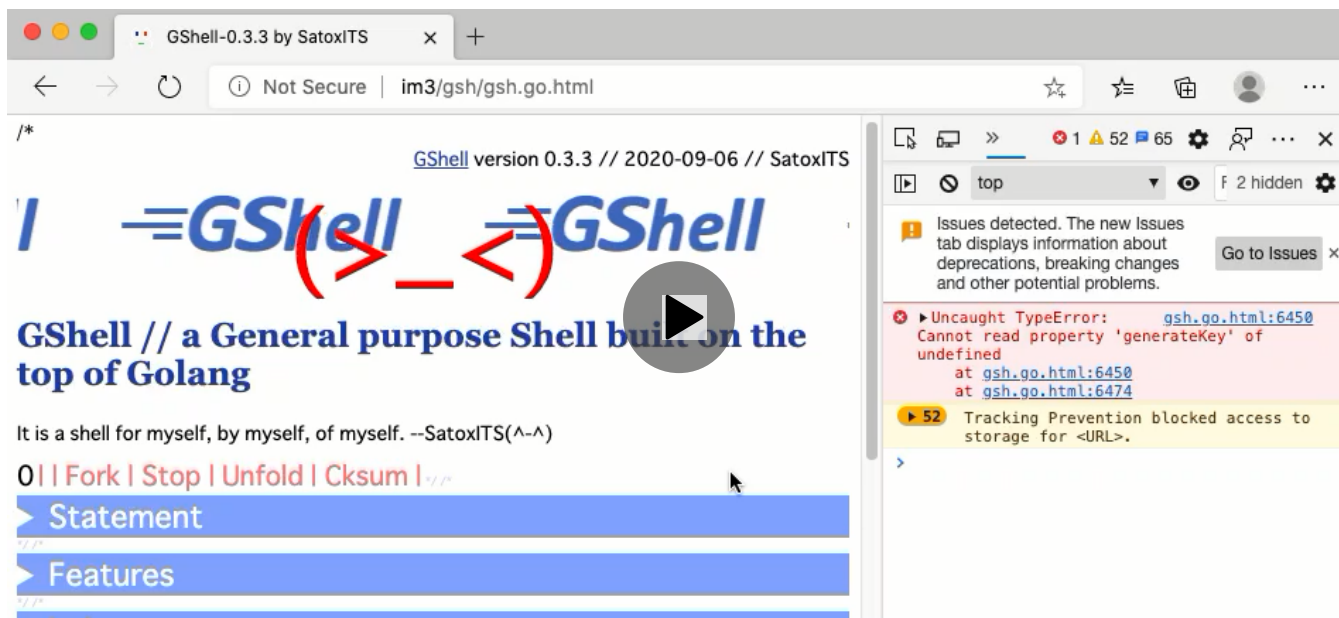
開発：それで、この top と left を JavaScript で書き換えれば良いはずですが。これはバナーをスライドしているのと同じ式でイけるでしょう… … いけないですね。なんでだろう。
console.log… おっと、この属性は文字列なのか。“px” とか付けないといけない？

開発：では刹那的に固定4地点移動にて。

```

if( e.code == "KeyJ" ){
    GshGrid.style.top = '200px';
    GshGrid.innerHTML = '( > < )Down';
}else
if( e.code == "KeyK" ){
    GshGrid.style.top = '20px';
    GshGrid.innerHTML = '( > < )Up';
}else
if( e.code == "KeyH" ){
    GshGrid.style.left = '20px';
    GshGrid.innerHTML = '( > < )Left';
}else
if( e.code == "KeyL" ){
    GshGrid.style.left = '200px';
    GshGrid.innerHTML = '( > < )Right';
}

```



社長：なるほど…

基盤：これは、ステータスとかメッセージとか表示するのに良さそうですね。

開発：エレメントをインスペクトした情報はここに表示すれば良いと思います。

社長：というか、スマホ版ではこの上の層にタッチしたイベントを拾えると思いますから、そこにいくつかボタンを配置すれば、JavaScriptで複数指同時タッチのIMEが出来そうに思います。

開発：表示しているHTMLと内部のノードの紐付け仕方がわかりませんが、その他は淡々と作るだけな感じですね。

社長：めでたいです。おなかすいたので飲みに行きましょう。

開発：昔 X Window でこのくらい簡単にGUIがいじれたら、ハマったでしょうね。

— 2020-0906 SatoxITS

http-im3-gsh-gsh-0.3.3.go_

ダウンロード



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 | Cksum | */ /*

▶ Statement

/ /

▶ Features

/ /

▶ Index

*/ //

▶ Go Source

//

▶ Considerations

// /*

▶ References

/ /

▶ Raw Source

/ /



▶ WebCrypto

*/ //