

株式会社 ITS MORE

2020年4月設立

ITS more

2020年7月30日 投稿者: SATOXITS

ブラウザユニオン

経理：今日は .jp が500円だとxsoからSPAMが来てました。

基盤：.jp は通常2800で表示してますから、これはマジでお得ですよ。

社長：うーん、何か思いついたら買しましょう。でも、.jp で取りたいという動機はあまりないですよ。うちの社名関係はもう長期に押さえてあるし。

経理：一度 .inc の500円セールしたら xso の株も上がるのにね。

開発：一日で .inc 登録ドメイン数を1万件突破！とか。これは全世界に誇って良いでしょう。

基盤：xso株が暴落して経営破綻じゃね？

* * *

開発：さて、昨日は Chromium がデバッガにうんとも反応しなくて困ったわけですが、結局こういうことでした。

```
(lldb) bt
* thread #1, name = 'CrBrowserMain', queue = 'com.apple.main-thread', stop reason = signal SIGSTOP
  * frame #0: 0x00007fff71709cfa libsystem_kernel.dylib`mach_msg_trap + 10
    frame #1: 0x00007fff7170a170 libsystem_kernel.dylib`mach_msg + 60
    frame #2: 0x00007fff7376f6f5 CoreFoundation`__CFRunLoopServiceMachPort + 247
    frame #3: 0x00007fff7376f9c2 CoreFoundation`__CFRunLoopRun + 1319
    frame #4: 0x00007fff7376f430 CoreFoundation`CFRunLoopRunSpecific + 462
    frame #5: 0x00007fff36321abd HIToolbox`RunCurrentEventLoopInMode + 292
    frame #6: 0x00007fff36321d75 HIToolbox`ReceiveNextEventCommon + 584
    frame #7: 0x00007fff36321579 HIToolbox`BlockUntilNextEventMatchingListInModeWithFilter + 64
    frame #8: 0x00007fff34967039 AppKit`DPSNextEvent + 883
    frame #9: 0x00007fff34965000 AppKit`-[NSApplication(NSEvent)_nextEventMatchingEventMask:untilDate:inMode:dequeue:] + 1352
    frame #10: 0x000000010544c229 libchrome.dll.dylib`__71-[BrowserCrApplication nextEventMatchingMask:untilDate:inMode:dequeue:]_block_invoke(.block_descriptor=0x00007ffefbfe978) at chrome_
browser_application_mac.mm:227:15
    frame #11: 0x0000000131090422 libbase.dylib`base::CallWithEHFrame(void (*) block_pointer) at call_with_eh_frame_asm.S:39
    frame #12: 0x000000010544c8c4 libchrome.dll.dylib`-[BrowserCrApplication nextEventMatchingMask:untilDate:inMode:dequeue:](self=0x0000000135081210, _cmd="nextEventMatchingMask:untilDate:in
Mode:dequeue:", mask=18446744073709551615, expiration=0x00007ffefbfe4618, mode="kCFRunLoopDefaultMode", dequeue=YES) at chrome_browser_application_mac.mm:226:3
    frame #13: 0x00007fff3495750e AppKit`-[NSApplication run] + 658
    frame #14: 0x00000001310a2a25 libbase.dylib`base::MessagePumpNSApplication::DoRun(this=0x000000010072f8b0, delegate=0x000000010072b9b0) at message_pump_mac.mm:691:5
    frame #15: 0x00000001310a9200 libbase.dylib`base::MessagePumpCFRunLoopBase::Run(this=0x000000010072f8b0, delegate=0x000000010072b9b0) at message_pump_mac.mm:149:3
    frame #16: 0x000000013105cd00 libbase.dylib`base::sequence_manager::internal::ThreadControllerWithMessagePumpImpl::Run(this=0x000000010072b9b0, application_tasks_allowed=true, timeout=(de
lta = 9223372036854775807)) at thread_controller_with_message_pump_impl.cc:46:13
    frame #17: 0x00000001310b0e70 libbase.dylib`base::RunLoop::Run(this=0x000000010042a230) at run_loop.cc:124:14
    frame #18: 0x000000010042a230 libchrome.dll.dylib`ChromeBrowserMainParts::MainMessageLoopRun(this=0x000000013508a8f0, result_code=0x000000013508a7d8) at chrome_browser_main.cc:1705:15
    frame #19: 0x000000014002804 libcontent.dylib`content::BrowserMainLoop::RunMainMessageLoopParts(this=0x000000013508a7c0) at browser_main_loop.cc:1002:29
    frame #20: 0x0000000140e15c4 libcontent.dylib`content::BrowserMainRunnerImpl::Run(this=0x000000013505fc20) at browser_main_runner_impl.cc:150:15
    frame #21: 0x0000000140cc443 libcontent.dylib`content::BrowserMain(parameters=0x00007ffefbffa4f8) at browser_main.cc:47:28
    frame #22: 0x000000014395c12 libcontent.dylib`content::RunBrowserProcessMain(main_function_params=0x00007ffefbffa4f8, delegate=0x00007ffefbffa4f8) at content_main_runner_impl.cc:825:10
    frame #23: 0x0000000143957502 libcontent.dylib`content::ContentMainRunnerImpl::RunServiceManager(this=0x0000000100714b00, main_params=0x00007ffefbffa4f8, start_service_manager_only=false)
at content_main_runner_impl.cc:97:10
    frame #24: 0x0000000143954d05 libcontent.dylib`content::ContentMainRunnerImpl::Run(this=0x0000000100714b00, start_service_manager_only=false) at content_main_runner_impl.cc:881:12
    frame #25: 0x0000000143951d46 libcontent.dylib`content::ContentServiceManagerMainDelegate::RunEmbedderProcess(this=0x00007ffefbffa4f8) at content_service_manager_main_delegate.cc:60:32
    frame #26: 0x0000000130ba9c9 libembedder.dylib`service_manager::Main(params=0x00007ffefbffa4f8) at main.cc:403:29
    frame #27: 0x0000000143950447 libcontent.dylib`content::ContentMain(params=0x00007ffefbffa4f8) at content_main.cc:19:10
    frame #28: 0x0000000102004c10 libchrome.dll.dylib`::ChromeMain(argc=1, argv=0x00007ffefbffa4f8) at chrome_main.cc:117:12
    frame #29: 0x0000000100001305 Chromium`main(argc=1, argv=0x00007ffefbffa4f8) at chrome_exe_main_mac.cc:117:0
    frame #30: 0x00007fff715c8cc9 libdyld.dylib`start + 1
    frame #31: 0x00007fff715c8cc9 libdyld.dylib`start + 1
(lldb) cont
```

開発：それで、バックトレースに出てくる経路の各所にprintfをはさんで動作を観察したわけです。

開発：バックトレースを見ると、根っこからしばらくは環境設定の層で、途中でブラウザの層、その先にAppKitとかが挟まった先にさきっぽのカーネルに達しています。根っこのほうの層では設定をしてはその先のRunメソッドを呼び出さないので、そこに介入しやすい中心ループ的なものはありません。

開発：Firefox ではイベントをpollしては処理するし、実装上pollがラッパーになっているので簡単にフックできました。ですがMac版Chromiumでは mach_msgというシステムコールで待っている模様。これはタイムアウト付きのrecvみたいなものようで、poll 的な処理と分離ができない。なので、一番さきっぽの所で捕まえるのはちょっと難儀です。まあ、自分でmach_msg にラップしちゃうという手もあるかも知れませんが。

開発：ですがまじめに取り組むとすると、一番下の層に手を突っ込むのは暴力的過ぎると思います。Firefox でもですが、普通に外部からのコマンドを受け付けるスレッドを作ってそれをプロセスに入れ込むのが、まっとうな解法なんだろうと思いますね。そうなると、宿主のプログラムの構造を理解する必要がありますが、特に Chromium の場合、コードが意味不明な C++ 的呪文で刻まれていて、ちょっとツール無しで追う気になれないという状況です。

社長：まあ、設計のドキュメントを読むという路線もあるかも知れませんがね。

開発：ただ、こういうエンドユーザに対峙する最前線に居るプログラムが、今日もC++で書かれていて、Cのプログラムを普通に突っ込めるといふ点には、ちょっと感動的なものがありますね。

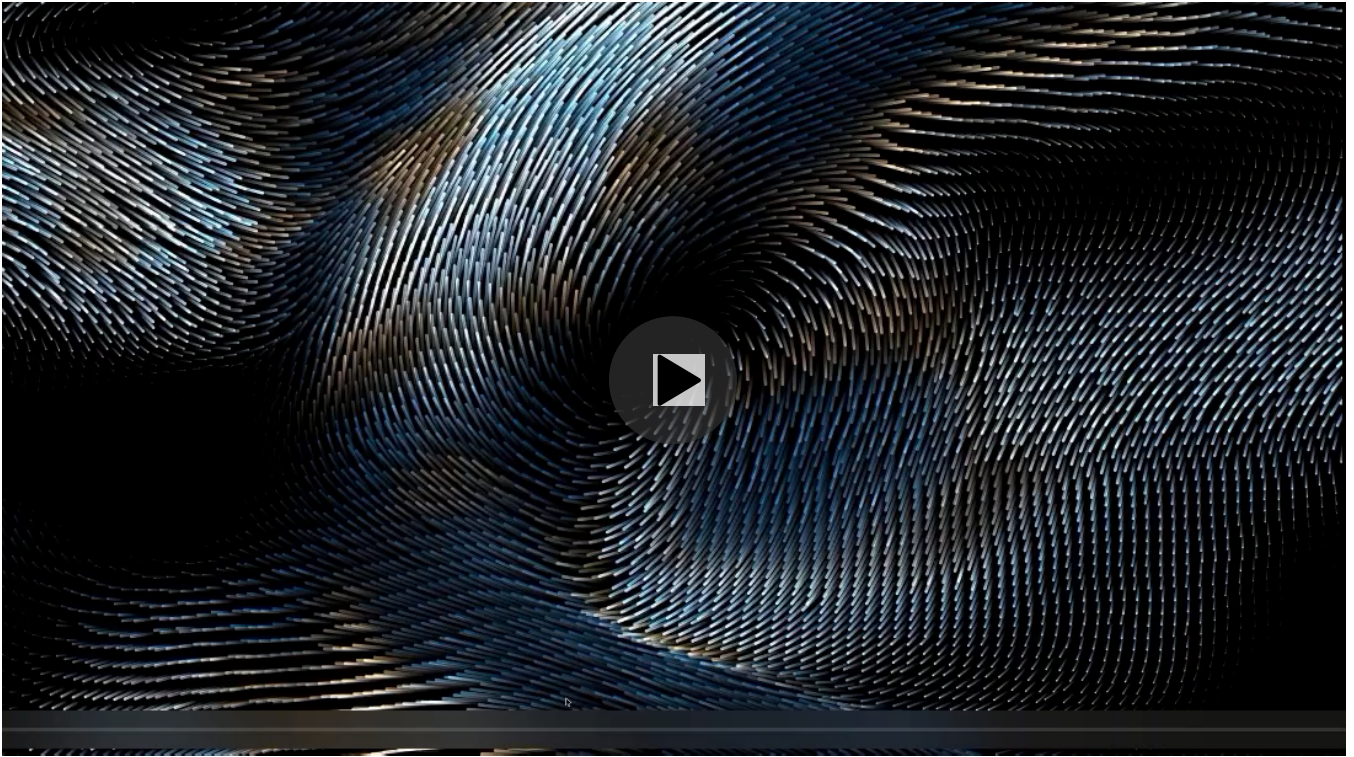
開発：なんにしても、Chromiumの作りはとっつきにくいし、MacOSXのガードも固いので手をツッコみにくい感じはします。なので、Mac版のFirefoxか、Linux版のChromiumでやれたらなと思います。

基盤：Chromiumのビルドにはストレージが120GB必要ですから、現行の仮想マシンのように仮想ディスクに置くのは大変ですね。外部にディスクを置かないと。いずれやる必要があるとは思ってましたので、この際にやってみましょうか。

社長：それはそうと、でんこちゃんの家計簿が早く開通しないかとわくわくしています。

事業所コード	お客さま番号	ご利用状況
		申請中

社長：それとこのスクリーンセーバー。初代液晶iMacのも印象的でしたが、時代はここまで来たか感があります。電気の無駄とは思いつつ。

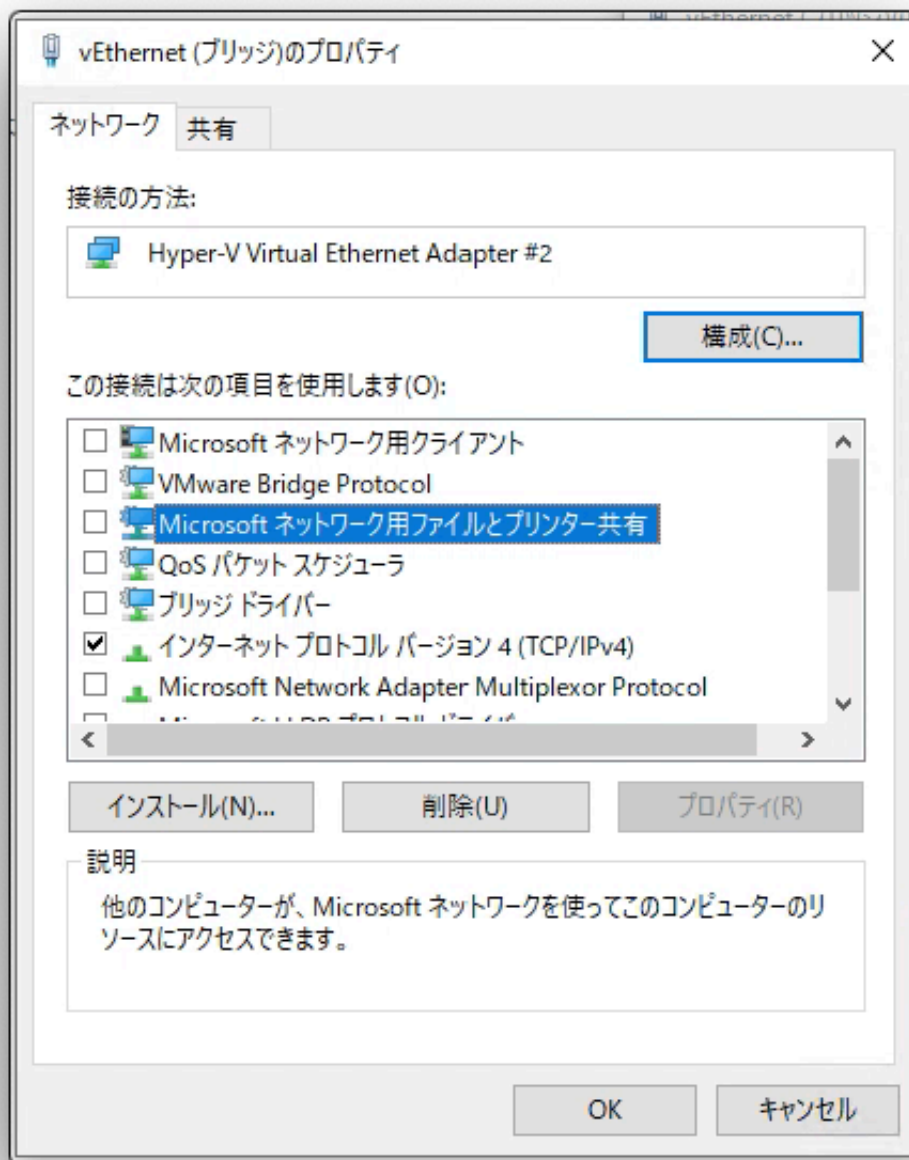


開発：iTunesの、音楽を画像化するやつも面白かったですね。

基盤：iTunesって死んじゃったんですかね？

* * *

基盤：ひーは一。いつの頃からか、どうもレノボ君がサンバでマウントさせてくれないと思ったら、こういうことになってました。



開発：ちょっとリストラし過ぎたようですねw

社長：太古の脳の中のブレーカーが落ちていたみたな。

開発：Windowsデビューの時代から一度も全体的な整理統合がされていないんでしょうね。

基盤：色々ひっかかりましたが、Linux用の、というか何からでもNFSでアクセスできるストレージ1TBを用意しました。

開発：それではさっそく。

基盤：手引は[GitHubにあるこれ](#)ですかね？

[How-to-build-chromium-on-Linux---intel_webml-polyfill-Wiki---GitHub](#)

ダウンロード

基盤：では言われるがままに… おっと、そもそも git がまだ無かった。sudo apt install git。なにやら準備した後に、chromium のソースを git clone…

開発：これって昨日Macで10分くらいで終わったやつですかね。

基盤：13GBのかたまりを13MiB/sくらいで落としてますから、20分くらいかかるかもですね。

社長：結局ストレージはQちゃんにしたんですね。

基盤：最初Hyper/Vのホスト直結のHDDをSMBでやろうとしたのですが、しょうもないところで引っかかりました。で、1時間かけてようやくつながったと思ったらやたらファイルの作成が遅い。ならQNAPのNFSにしようとしたら、今度はボリュームの作り方とかNFSの設定の仕方をすっかり忘れてしまっていて。思い出すと言うより探検し直すのに小一時間。

開発：Qちゃんのドライブがコロコロ言うのが面白いですね。懐かしい響き。

社長：HDDのコロコロ音は悪くないですよ。

システムの健康状態: 良好 (QN1 アップタイム 0日 01:11:18)

ハードウェア情報: システム 36°C/96°F, システム... 900 RPM

リソースモニター: CPU 30%, RAM 36%, Adapter 1 (77 MB/s ↓, 408 KB/s ↑)

ディスク健康状態: 良好 (NAS 1/2 ディスク)

ストレージ: work0 (準備完了)

- 使用済み: 46.94 GB
- 使用可能: 51.26 GB

 Public: 29.88 GB, work0: 953.69 MB, homes: 28.00 KB, Web: 12.00 KB, その他: - Byte(s)

```
Terminal — ssh -i ~/.ssh/u18.pem u18 — 101x5
ssh -i ~/.ssh/u18.pem u18
u18$ time git clone https://g Terminal — ssh -i ~/.ssh/u18.pem u18 n-src.git
Cloning into 'chromium-src'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
Receiving objects: 97% (12736557/12999042), 14.08 GiB | 12.72 MiB/s
```

基盤：もう少しで終了… 終了。？と思ったらすごいゴロゴロし始めましたね。

システムの健康状態: 良好 (QN1 アップタイム 0日 01:14:35)

ハードウェア情報: システム 35°C/95°F, システム... 930 RPM

リソースモニター: CPU 8%, RAM 36%, Adapter 1 (89 KB/s ↓, 1 MB/s ↑)

ディスク健康状態: 良好 (NAS 1/2 ディスク)

ストレージ: work0 (準備完了)

- 使用済み: 46.94 GB
- 使用可能: 51.26 GB

 Public: 29.88 GB, work0: 953.69 MB, homes: 28.00 KB, Web: 12.00 KB, その他: - Byte(s)

```
Terminal — ssh -i ~/.ssh/u18.pem u18 — 101x5
ssh -i ~/.ssh/u18.pem u18
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 12999042 (delta 2), reused 2 (delta 2), pack-reused 12999039
Receiving objects: 100% (12999042/12999042), 14.30 GiB | 12.51 MiB/s, done.
Resolving deltas: 1% (179367/10242873)
```

開発：Resolving deltas っていうから、ファイルをなめて差分チェックとかするんでしょうか。これは苦手分野。

基盤：うーん、全体で1000万あるのに、秒1000程度な感じなので、3時間くらいかかりそうです。これは… サムスン512GB SSDが半分空いているので、こっちでやり直しますかね？今ようやく10%。

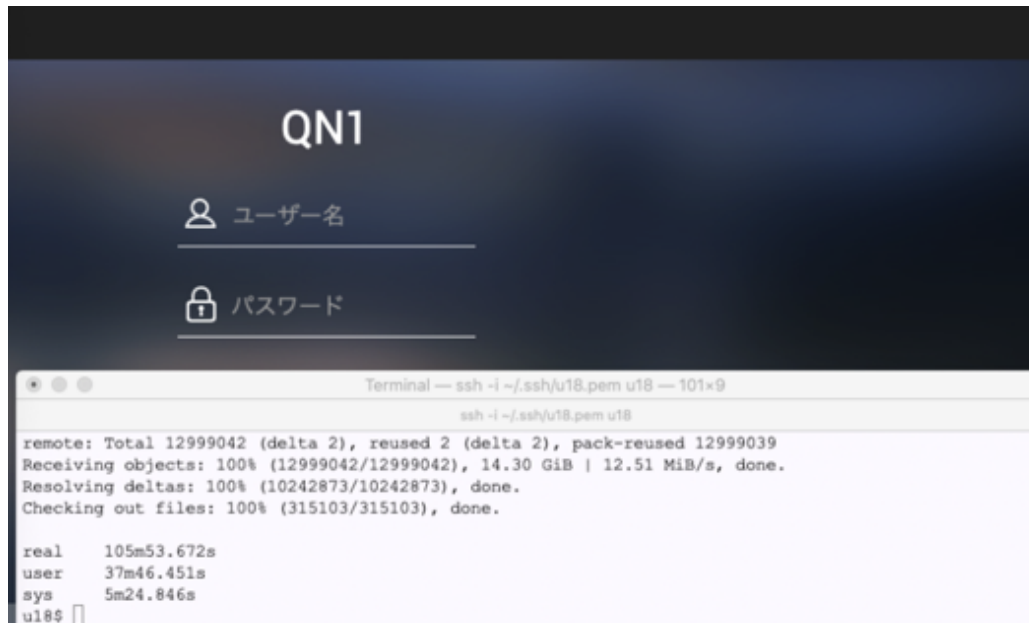
社長：これはこれで面白そうなので様子を見ましょう。食事に行ってきます。

* * *

社長：帰りましたはー

経理：酒臭いですね。

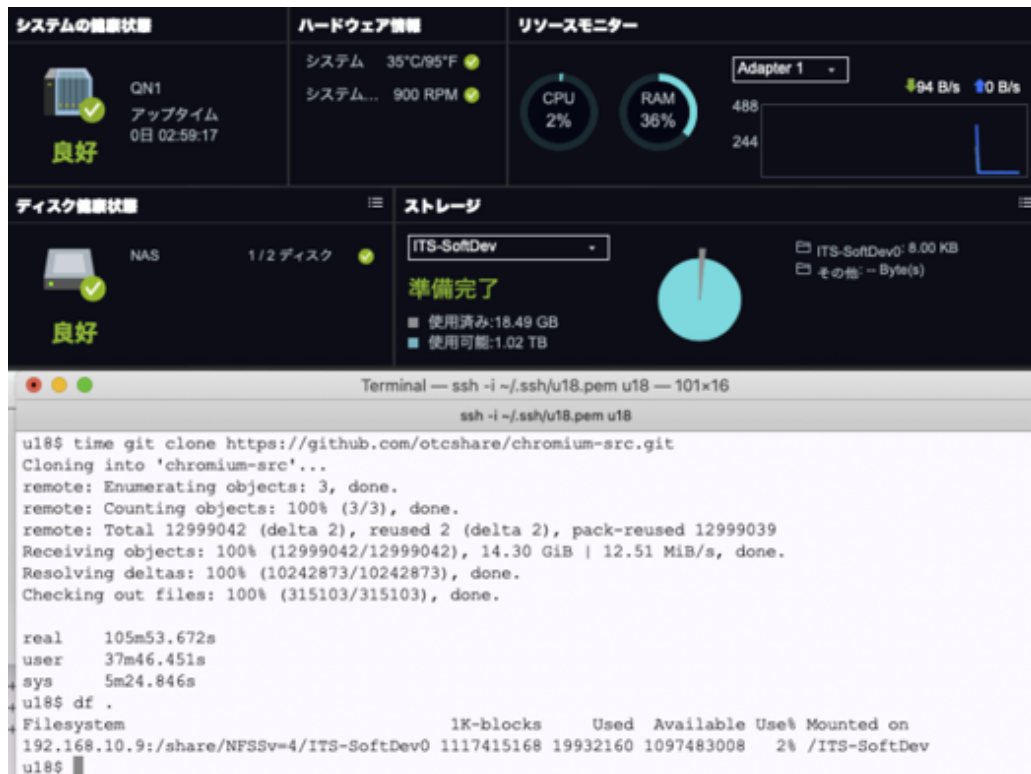
社長：うーん、中瓶の店だといつい2本飲んじゃうんですよね。よし悪し。ていうか、Qちゃんがおとなしくなってますね。



基盤：結局2時間弱で終わりました。

社長：飲みながら思ったんですが、結局仮想ディスクにするべきじゃないかなと。

基盤：そうですね。今の所20GBしかないの、オンメモリで数分で済んだ話かなと。



社長：それはともかく、Qちゃんの自動ログアウトの時間を長くしましょう。デフォルトでは1時間？。6時間にしましょう。そういう刻みは無い？では一日で。



社長：このダッシュボードせつかくよく出来てるのに、なんでディスクアクセス状況のメータとかグラフが無いんですかね？単独での表示にできないのも不思議です。

基盤：それはともかく、ビルドの手引に従って先に進めます。特に必要もないのにエディタにvimを使っていると書いてるのが面白いですね。依存性のチェック… aptでかなりいろんなパッケージをインストールしております。

開発：いまだに Chromiumの世界では Ubuntu 14.04 がふつーってすごすぎないですか？

How to build chromium on Linux

Bin Miao edited this page on 22 Jun · 7 revisions

System requirements

- A 64-bit Intel machine with at least 8GB of RAM. More than 16GB is highly recommended.
- At least 100GB of free disk space.
- You must have Git and Python v2 installed already.

Most development is done on Ubuntu (currently 14.04, Trusty Tahr). There are some instructions for other distros below, but they are mostly unsupported.

社長

社長：まあ、うまく行ってたものをいじって壊したくないでしょうからね。これだけ複雑なソフトだと。

基盤：それより、RAMは16GB以上を強く推奨、ディスクは100GB食うからねってちゃんと書いてあるのが良いですね。

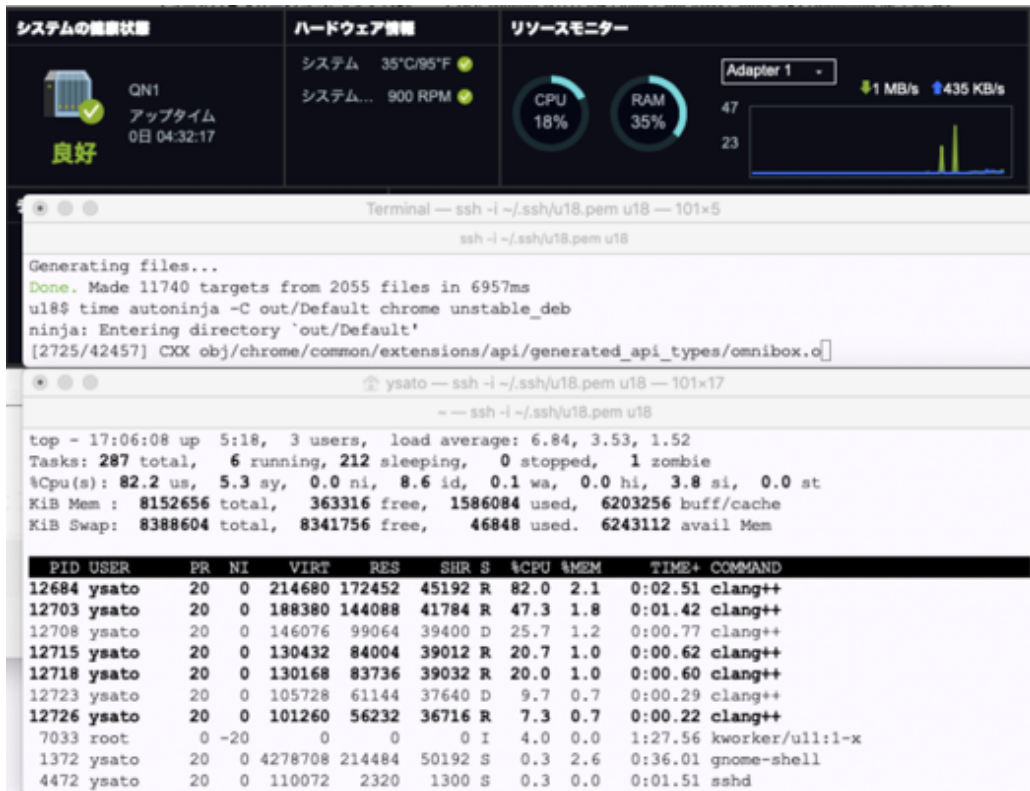
基盤：ツール類のアップデートは2分47秒で終わりました。では、ビルド… じゃなくてまだまだ準備があるんですね。5. Update your checkout and run the hooks… これってまさか、さっきのノリで2時間かかるとか…

社長：時間がかかるようならその間に他の方法も試しましょう。

基盤：では、`gclient sync` … ああ、これは数分で終わりそうな気配。



基盤：14分7秒でした。gclient runhooks。26秒。ビルドの設定。さて、ようやく autoninja の出番です。autoninja -C out/Default chrome unstable_deb...



開発：ディスクはちよっところころ言う程度ですね。

システムの健康状態

ハードウェア情報

リソースモニター

システム 35°C/95°F

システム... 900 RPM

CPU 6%

RAM 35%

Adapter 1

478 KB/s

331 KB/s

47

23

Terminal — ssh -i ~/ssh/u18.pem u18 — 101x5

```
Generating files...
Done. Made 11740 targets from 2055 files in 6957ms
u18$ time autoninja -C out/Default chrome unstable_deb
ninja: Entering directory `out/Default'
[4805/42457] CXX obj/third_party/angle/libANGLE_base/DisplayImpl.o
```

ysato — ssh -i ~/ssh/u18.pem u18 — 101x17

```
top - 17:11:07 up 5:23, 3 users, load average: 7.18, 5.90, 3.12
Tasks: 285 total, 6 running, 211 sleeping, 0 stopped, 1 zombie
%Cpu(s): 76.6 us, 8.2 sy, 0.0 ni, 10.1 id, 1.4 wa, 0.0 hi, 3.8 si, 0.0 st
KiB Mem : 8152656 total, 1619848 free, 1379832 used, 5152976 buff/cache
KiB Swap: 8388604 total, 8341756 free, 46848 used, 6449292 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
18166	ysato	20	0	65472	33120	7144	R	63.1	0.4	0:01.92	python
18299	ysato	20	0	162248	115220	41780	R	29.9	1.4	0:00.90	clang++
18304	ysato	20	0	126252	82332	40268	D	19.3	1.0	0:00.58	clang++
18307	ysato	20	0	112200	67668	38360	D	10.3	0.8	0:00.31	clang++
18310	ysato	20	0	100968	56608	37144	R	8.0	0.7	0:00.24	clang++
18313	ysato	20	0	99748	55120	36660	R	7.0	0.7	0:00.21	clang++
7033	root	0	-20	0	0	0	I	5.3	0.0	1:37.21	kworker/u11:1-x
18316	ysato	20	0	86916	40136	33320	R	1.0	0.5	0:00.03	clang++
6064	ysato	20	0	169400	157396	3144	S	0.7	1.9	0:08.36	ninja-linux64
10	root	20	0	0	0	0	S	0.3	0.0	0:00.76	ksoftirqd/0

基盤：順調に進みます。

開発：今気づいたんですが、右肩に現在時刻を表示してくれるのは macOS の top のサービスだったんですね。一方この、ターミナルはやる事が無いもんだから妙な情報をウィンドウのバーに表示している…ここに時刻でも出せばよいのに。

社長：これがまた、その場で右クリックで設定もできないという悲しい macOS の佐賀。あー、コンテキストメニューには一応あるんですね。なんでインスペクトという名前なのかわかりませんが。

開発：全てのアプリのウィンドウが「これと同じサイズの同じコンテンツのクローンを作る」というコンテキストメニューを持つべきですよ。Catalina にはぜひ、そういうサポートをして頂きたい。

基盤：バーへの表示文字列はエスケープシーケンスで指定できたりするかもですね。あるいは、出向元の zsh と連絡を取って現状を報告してはバーに反映させるとか。

開発：要するに現在実行しているプロセスの引数を表示しているだけみたいですから、shell で自分の引数を変えるようにすればそうなりますね。


```

src — 日本語でも行けるんだな — 001[RPM=0.81(0.0 0.4 0.1),IDLE=60s,ACT=1]-P9999 -- -----...
** NG, this executable is not signed
** checking DONE (took 0.282 seconds)
(UNIX) 17:53:06.579 [7095] Path-Normalized<<< /var/folders/bw/8dlt3xks44727rb0y662k
w_m0000gn/T/
(UNIX) 17:53:06.579 [7095] Path-Normalized>>> /var/folders/bw/8dlt3xks44727rb0y662k
w_m0000gn/T
<DeleGate/9.9.13+its01> [7095] -P9999 READY
[]

ysato — -zsh — 83x5
MacMini% ps x|grep delegate
 7144 s003 S+   0:00.00 grep delegate
 7095 s028 S+   0:00.31 delegated= -{002:001/001}[RPM=0.90(2.0 0.4 0.1),IDLE=45s
,ACT=1]-P9999 -- -----
MacMini%

```

基盤：つまり shell が pstitle というコマンドを持ってれば良いと…

開発：そういえば zsh はソースからビルドしましたから… find zsh -type f -exec grep -i pstitle '{}' ';' … ないですね。bash にはあるのかな。

基盤：あ、ただいま中間折り返し地点を通過。

システムの健康状態: QN1 アップタイム 0日 05:23:01 良好

ハードウェア情報: システム 35°C/95°F, システム... 930 RPM

リソースモニター: CPU 6%, RAM 35%, Adapter 1: 529 KB/s ↓, 311 KB/s ↑

```

Terminal — ssh -i ~/.ssh/u18.pem u18 — 101x5
Generating files...
Done. Made 11740 targets from 2055 files in 6957ms
ul18$ time autoninja -C out/Default chrome unstable_deb
ninja: Entering directory `out/Default'
[21807/42457] CXX obj/content/common/mojo_binding...refetched_signed_exchange_info.mojom-test-utils.

ysato — Building MyChromium — ssh -i ~/.ssh/u18.pem u18 — 101x17
top - 17:56:53 up 6:09, 3 users, load average: 7.08, 7.19, 7.04
Tasks: 285 total, 8 running, 209 sleeping, 0 stopped, 1 zombie
%Cpu(s): 78.0 us, 6.0 sy, 0.0 ni, 11.7 id, 0.1 wa, 0.0 hi, 4.3 si, 0.0 st
KiB Mem : 8152656 total, 1260396 free, 1700748 used, 5191512 buff/cache
KiB Swap: 8388604 total, 8341756 free, 46848 used, 6128452 avail Mem

  PID USER      PR  NI   VIRT   RES    SHR  S  %CPU  %MEM    TIME+  COMMAND
 2651 ysato    20   0 196452 152528 43956 R  48.3  1.9   0:01.45 clang++
 2654 ysato    20   0 181720 139672 44184 R  42.0  1.7   0:01.26 clang++
 2660 ysato    20   0 167496 120956 39692 R  33.3  1.5   0:01.00 clang++
 2657 ysato    20   0 161272 114408 39848 R  32.3  1.4   0:00.97 clang++
 2663 ysato    20   0 139572  93352 39332 R  25.0  1.1   0:00.75 clang++
 2666 ysato    20   0 144828  98688 39316 R  25.0  1.2   0:00.75 clang++
 2669 ysato    20   0 130228  83864 38908 R  19.7  1.0   0:00.59 clang++
30635 root      0  -20     0     0     0  I   5.3  0.0   0:17.05 kworker/u11:1-x
  11 root     20   0     0     0     0  I   0.3  0.0   0:13.95 rcu_sched
  30 root     20   0     0     0     0  S   0.3  0.0   0:01.26 ksofirqd/3

```

基盤：時間的な中間点じゃないですけど。

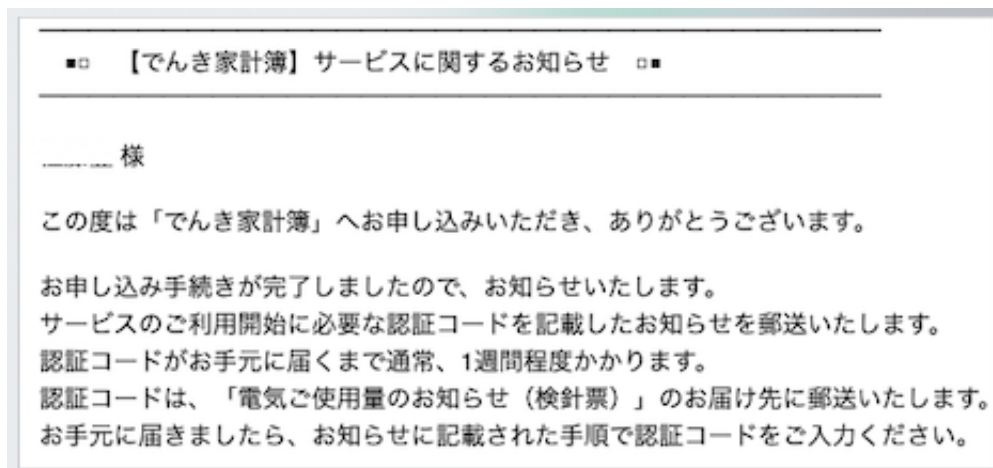
開発：Mac版のビルドと同じようなペースですね。

社長：shell の pstitle を使うとすると、時刻を表示するには、shell 自体に cron的な機能が必要ですね。

開発：shell の外付けにしてシグナルで起動するとかもありですかね。というか、タイムスリッパ式に動的ライブラリを差し替えれば行けそうな。しかもまさに、gettimeofday で。

社長：となると、shell に限らないですね。なんでもOK。ちょっと休憩しましょう。

経理：アース渦巻第2巻に着火しました。ところでお待ちかねのでんこちゃんはお1週間かかるそうです。



社長：マチカネタンホイザー…/(‘^)

* * *

基盤：それはそうと、MacMini からの引っ越しが済まない間に、Vivaldiが大変なことになっちゃってて、時々応答がなくなります。

社長：いやわたしは、ブックマークとかせこいこと言ってないで、一度開いたら一生そのまま開いてれば良いと思うんですよ。ああ、これは3ヶ月前に開いたページだなとか。こういう順序で情報を探索して理解したなとか。克明に思い出せるのです。

社長：なんなら、過去に開いた順序どおりに再生して開き直してくれるとありがたいで

AcrobatなんてMove to Trashですよ。

経理：月額1,700円相当で年間契約済みなんですけど。

* * *

基盤：3万ファイル通過しました。

The screenshot displays a system monitoring interface with three main sections: System Status, Hardware Information, and Resource Monitor. Below these are two terminal windows showing command execution and system performance metrics.

システムの健康状態
 QN1
 アップタイム
 0日 06:41:41
 良好

ハードウェア情報
 システム 34°C/93°F
 システム... 930 RPM

リソースモニター
 Adapter 1
 420 KB/s 233 KB/s
 CPU 7%
 RAM 35%
 5.7
 2.8

ディスク健康状態 | **ストレージ**

Terminal 1: ssh -i ~/.ssh/u18.pem u18 -- 101x5
 Generating files...
 Done. Made 11740 targets from 2055 files in 6957ms
 u18\$ time autoninja -C out/Default chrome unstable_deb
 ninja: Entering directory 'out/Default'
 [30816/42457] CXX obj/third_party/blink/renderer/core/dom/dom/window_event_context.o

Terminal 2: ysato -- Building MyChromium -- ssh -i ~/.ssh/u18.pem u18 -- 101x17
 top - 19:15:34 up 7:27, 3 users, load average: 7.17, 7.14, 7.10
 Tasks: 283 total, 6 running, 211 sleeping, 0 stopped, 1 zombie
 %Cpu(s): 85.7 us, 5.4 sy, 0.0 ni, 5.7 id, 0.2 wa, 0.0 hi, 3.1 si, 0.0 st
 KiB Mem : 10014288 total, 4551368 free, 2591212 used, 2871708 buff/cache
 KiB Swap: 8388604 total, 8343292 free, 45312 used, 7284600 avail Mem

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
29481	ysato	20	0	331784	281716	44388	R	61.8	2.8	0:03.86	clang++
29502	ysato	20	0	150112	103152	40268	R	28.6	1.0	0:00.86	clang++
29508	ysato	20	0	142348	96516	39600	D	25.9	1.0	0:00.78	clang++
29510	ysato	20	0	135364	88920	39632	D	23.3	0.9	0:00.70	clang++
29513	ysato	20	0	107856	63428	37556	R	11.0	0.6	0:00.33	clang++
29516	ysato	20	0	92916	48112	36232	R	4.3	0.5	0:00.13	clang++
26170	root	0	-20	0	0	0	I	3.3	0.0	0:06.47	kworker/u11:1-x
29519	ysato	20	0	84328	39056	34524	D	1.0	0.4	0:00.03	clang++
11	root	20	0	0	0	0	I	0.3	0.0	0:20.08	rcu_sched
6064	ysato	20	0	229516	217056	3144	S	0.3	2.2	0:23.19	ninja-linux64

基盤：通信がぴよこんと立っているのはさっきsshでログインしたからです。

社長：それにしてもこのQNAP、2万円でこれってすごすぎますね。

開発：ふつうにLinuxですしね。

```
ssh admin@qn1 -- 99x33
[~] # hostname
QN1
[~] # uname -a
Linux QN1 4.2.0 #1 SMP Thu Jul 2 09:32:05 CST 2020 aarch64 GNU/Linux
[~] # ps aux|wc
      314      1774      16876
[~] # df
Filesystem                Size      Used Available Use% Mounted on
none                      63.0M    41.4M    21.6M   66% /
devtmpfs                  744.9M     8.0K    744.9M    0% /dev
tmpfs                     64.0M    404.0K    63.6M    1% /tmp
tmpfs                     770.5M   120.0K    770.4M    0% /dev/shm
tmpfs                     16.0M     0        16.0M    0% /share
tmpfs                     16.0M     0        16.0M    0% /mnt/snapshot/export
/dev/md9                  493.5M   323.3M    170.1M   66% /mnt/HDA_ROOT
cgrouop_root              770.5M     0        770.5M    0% /sys/fs/cgroup
/dev/mapper/cachedev2    98.7G    46.9G    51.3G   48% /share/CACHEDEV2_DATA
/dev/md13                 417.0M   334.7M    82.3M   80% /mnt/ext
/dev/mapper/cachedev1    96.4G   289.1M    95.6G    0% /share/CACHEDEV1_DATA
tmpfs                     48.0M    76.0K    47.9M    0% /share/CACHEDEV2_DATA/.samba/lock/msg.lock
tmpfs                     16.0M     0        16.0M    0% /mnt/ext/opt/samba/private/msg.sock
tmpfs                     16.0M     0        16.0M    0% /share/NFSv=4
/dev/mapper/cachedev2    98.7G    46.9G    51.3G   48% /share/NFSv=4/Public
/dev/mapper/cachedev3    1.0T    33.2G    1.0T    3% /share/CACHEDEV3_DATA
/dev/mapper/cachedev3    1.0T    33.2G    1.0T    3% /share/NFSv=4/ITS-SoftDev0
[~] # uptime
19:21:55 up 6:48, load average: 0.15, 0.28, 0.37
[~] # █
```

```
ssh admin@qn1 -- 82x20
Mem: 1417340K used, 160716K free, 37584K shrd, 199716K buff, 520716K cached
CPU: 3.6% usr 2.7% sys 0.2% nic 92.1% idle 0.3% io 0.0% irq 0.9% sirq
Load average: 0.49 0.34 0.36 3/946 28747
  PID  PPID  USER  STAT  VSZ  %VSZ  CPU  %CPU  COMMAND
12820   2  admin  SW     0  0.0   3  1.2  [nfsd]
 6343   1  admin  S      42192 2.6   3  0.4  /mnt/ext/opt/Python/bin/python /mnt/
2839   1  admin  S     1521m 98.6   0  0.4  /sbin/hal_daemon -f
12819   2  admin  SW     0  0.0   0  0.4  [nfsd]
16619 16617  admin  S N    394m 25.5   0  0.1  {mytranscodesvr} /usr/local/medialib
11916   1  admin  S N    161m 10.4   1  0.1  /mnt/ext/opt/Python/bin/python2 /sbi
12818   2  admin  SW     0  0.0   2  0.1  [nfsd]
17090 14362  admin  S <   3304m214.2 0 0.0 /usr/local/apache/bin/apache_proxy -
24801   1  admin  S N    835m 54.1   0  0.0  /usr/local/mariadb/bin/mysqld --defa
25175   1  admin  S N    242m 15.7   0  0.0  /usr/local/bin/qsynsrv_monitor -pid
16230   1  admin  S     76440 4.8   1  0.0  /sbin/rfsd -i -f /etc/rfsd.conf
 6332   1  admin  S     29304 1.8   2  0.0  {redis-server} /mnt/ext/opt/netmgr/u
28196 17959  admin  R      3428 0.2   2  0.0  top
   7     2  admin  SW     0  0.0   2  0.0  [rcu_sched]
  20    2  admin  SW     0  0.0   3  0.0  [ksoftirqd/3]
  12    2  admin  SW     0  0.0   1  0.0  [ksoftirqd/1]
```



```

ssh admin@qn1 — 82x55
[~] # ls /bin
[[@      chgrp@      dmesg@      less@      naslogin*  sh*        unlzma@
[[@      chmod@      e2fsck@     ln@        netstat@  shalsum@  unxz@
addgroup@ chown@     e2fsck_64* login@     nslookup* sha256sum* usleep@
adduser@  cmp@       echo@       ls*        pidof@    sleep@    utelnetd*
ash@      cp*        egrep@     lsattr@   ping@     sync@     vi@
awk@      cpio@     false@    lspci@    ping6@    tar*      vim@
base64*   cut@      fgrep@     lzma@     ps@       tc*       watch@
bash@     date*     grep@      md5sum@   pwd@      touch@    wget-s*
blockdev* dd@       gunzip@    mkdir@    renice@   tr@       xz*
busybox*  delgroup@ gzip@      mknod*    rm@       true@     zcat@
bzipcat@  deluser@  host*     mktemp@   rmdir@   tty@
bzzip2*   df@       hostname@ more@     sed@      umount*
cat*      diff@     ip*       mount*    seq@     uname@
chattr@   dig*     kill@     mv@       setsid*  uniq@

[~] # ls /usr/sbin
agent@      firewall.sh@      rpc.mountd@
aplay@      fsck.fat@         rpc.nfsd@
avahi-autoipd@ getbindaddr@     rpc.rquotad@
avahi-browse@ get_geoinfo.sh@  rpc.statd@
avahi-browse-domains@ get_qcloud_device_name@ rpc.svcgssd@
avahi-daemon@ get_qcloud_host@  rsyslogd@
avahi-dnsmasq@ hciconfig@       rtk_transcoding_daemon@
avahi-publish@ hcitool@         scp@
avahi-publish-address@ inotifywait@     screen@
avahi-publish-service@ iostat@          send_http_mail@
avahi-resolve@ killall5@         sendmail@
avahi-resolve-address@ logrotate@       slapacl@
avahi-resolve-host-name@ lsof@            slapadd@
avahi-set-host-name@  lsusb@           slapauth@
bluetoothd@ mcu_tool@        slapcat@
check_dbus@ mDNSResponderPosix@ slapd@
chroot@      mkdosfs@         slapdn@
create_ddns_conf@ mkntfs@          slapindex@
create_dhcpd_conf@ mount.davfs@     slappasswd@
create_tftpd_conf@ nmcli@           slapschema@
crond@       nmd.sh@          slapttest@
cupsd@       nmlog@           sm-notify@
curlftps@   noip2c@          sms_sender@
davmigrate@ ntfsresize@      ssdutil@
dbus-daemon@ ntpd@            sshd@
dbus-daemon-launch-helper@ ntpdate@         ssmtp@
dbus-monitor@ ntpdated@        stunnel@
dbus-send@   openftpd@        udhcpd@
dbus-uuidgen@ parted@           ufsd@
dhclient@   partprobe@       umount.davfs@
dhcp6c@     pic_tool@        upsc*
dhcp6s@     qcloud_device_model_tool@ upsd@
dhcp6_source_route@ qcloud_tool@     upsmon@
dhcp_source_route@ qcloud_uninit_device_tool@ ups_stop_service@
ethtool@    radvd@           upsutil@
exportfs@   rdate@           vhost_cmd.php@
ez-ipupdate@ rdnsd@           zic@
fatlabel@   rpc.idmapd@

[~] # █

```

開発：Cortex-A72積んで Linux機として3万円位で売り出したら衝撃じゃないですか

ね？

社長：これってWordPress動くんですかね？

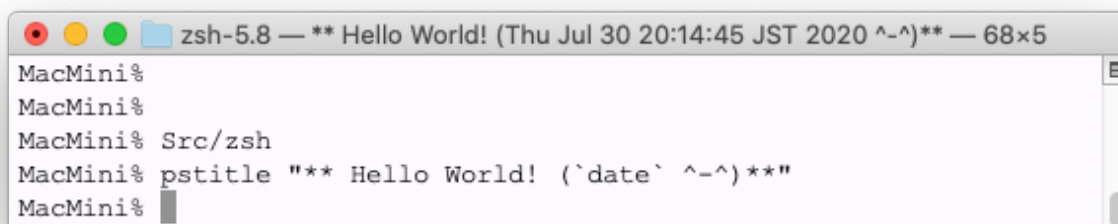
基盤：できるようです。というか正式サポートしているようです。

社長：うーん… 内部・開発系と、外部公開系は分けたいから、もう一台買いたいかな…

開発：あと1時間くらいかかりますかね。ちょっくら zsh に pstyle コマンドを入れてみますか。内蔵コマンドはどこかな？

```
MacMini% grep '"alias"' */*.c
Src/builtin.c:    BUILTIN("alias", BINF_MAGICEQUALS | BINF_PLUSOPTS,
bin_alias, 0, -1, 0, "Lgmrs", NULL),
```

開発：vi Src/builtin.c … へー、printf コマンド内蔵なんですね。じゃお隣におじゃましますよっと。引数をargv[0]にstrcpy()。これだけで良いかな？makeして。えいっ。



```
zsh-5.8 — ** Hello World! (Thu Jul 30 20:14:45 JST 2020 ^-^)** — 68x5
MacMini%
MacMini%
MacMini% Src/zsh
MacMini% pstyle "*** Hello World! (`date` ^-^)**"
MacMini%
```

基盤：おお！

社長：なんというか、古き良き時代が戻ってきた気分ですねえ…

開発：ちょっとした改変をするのにシェルなら5分。ブラウザは3日掛かり…

基盤：しかし、ターミナルがこれを見に来てくれるのに、1秒はかからないですが、0.5秒くらいかかるような感じですね。

社長：シェルとブラウザをどう組み合わせるかですね。実装上。それとスクリプト言語として。

開発：shellから見ると、ブラウザの内部状態がファイルシステムとして見えるというのが自然かなとは思いますが、で、特殊ファイルに書き込むとdocumentとして表示されたり、ウィンドウを開いたりする感じ。

基盤：JavaScriptのインタプリタとしても使えますね。電卓とかw

社長：こっち側はシンプルに実装できるのにねえ…

開発：あと、やはり bash も検討はしておいたほうが良いかなとは思いますが。

基盤：Go言語はどうするんでしょう？

開発：shellに動的リンクしちゃうと良いかも。

社長：すごいリッチなshellになりそうですね (^-^)。名前は ezsh なんてのがいいな。extended zsh、イージーシェル。

* * *

基盤：おっと、40000を超えました。もうそろそろ着陸態勢に。

システムの健康状態
QN1
アップタイム
0日 08:08:32
良好

ハードウェア情報
システム 35°C/95°F
システム... 930 RPM

リソースモニター
Adapter 1
7 MB/s 1 MB/s
CPU 10%
RAM 36%

ディスク健康状態 | **ストレージ**

```
Terminal -- ssh -i ~/.ssh/u18.pem u18 -- 101x5
Generating files...
Done. Made 11740 targets from 2055 files in 6957ms
u18$ time autoninja -C out/Default chrome unstable_deb
ninja: Entering directory `out/Default'
[40841/42457] CXX newlib_pnacl_nonsfi/obj/mojo/core/impl_for_embedder/entrypoints.o
```

```
ysato -- Building MyChromium -- ssh -i ~/.ssh/u18.pem u18 -- 101x17
top - 20:42:24 up 8:54, 3 users, load average: 7.27, 7.22, 7.15
Tasks: 301 total, 7 running, 220 sleeping, 0 stopped, 1 zombie
%Cpu(s): 70.7 us, 6.8 sy, 0.0 ni, 14.9 id, 2.3 wa, 0.0 hi, 5.3 si, 0.0 st
KiB Mem : 10014288 total, 2973656 free, 3407776 used, 3632856 buff/cache
KiB Swap: 8388604 total, 8343292 free, 45312 used, 6468036 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
29816	ysato	20	0	0	0	0	R	34.9	0.0	0:01.05	clang
29820	ysato	20	0	138180	100412	41788	R	29.2	1.0	0:00.88	clang
29851	ysato	20	0	119572	83216	41132	R	22.3	0.8	0:00.67	clang
29860	ysato	20	0	112704	75876	40624	R	17.9	0.8	0:00.54	clang
13954	root	0	-20	0	0	0	I	7.3	0.0	0:33.80	kworker/u11:2-x
29700	ysato	20	0	80548	22576	21032	D	3.0	0.2	0:00.09	clang++
29870	ysato	20	0	81328	43184	36828	R	1.7	0.4	0:00.05	clang
29871	ysato	20	0	52856	29432	25836	R	1.3	0.3	0:00.04	pnacl-llc
29798	ysato	20	0	41256	12104	6608	S	1.0	0.1	0:00.03	python
29805	ysato	20	0	37972	11540	6336	S	1.0	0.1	0:00.03	python

基盤：すごい勢いで進行しております… 大詰めのリンカーのフェイズにきました。

開発：Qちゃんがゴロゴロ言ってますね。

システムの健康状態
QN1
アップタイム
0日 08:12:33
良好

ハードウェア情報
システム 35°C/95°F
システム... 900 RPM

リソースモニター
Adapter 1
1 MB/s 7 MB/s
CPU 10%
RAM 36%

ディスク健康状態 | **ストレージ**

```
Terminal -- ssh -i ~/.ssh/u18.pem u18 -- 101x5
Generating files...
Done. Made 11740 targets from 2055 files in 6957ms
u18$ time autoninja -C out/Default chrome unstable_deb
ninja: Entering directory `out/Default'
[42441/42457] LINK ./chrome
```

```
ysato -- Building MyChromium -- ssh -i ~/.ssh/u18.pem u18 -- 101x17
top - 20:46:25 up 8:58, 3 users, load average: 4.88, 6.70, 7.00
Tasks: 267 total, 1 running, 197 sleeping, 0 stopped, 1 zombie
%Cpu(s): 0.1 us, 0.9 sy, 0.0 ni, 98.7 id, 0.0 wa, 0.0 hi, 0.3 si, 0.0 st
KiB Mem : 10014288 total, 3126572 free, 3161560 used, 3726156 buff/cache
KiB Swap: 8388604 total, 8343292 free, 45312 used, 6714252 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
11897	ysato	20	0	82084	23780	21076	D	4.7	0.2	0:00.32	clang++
16718	root	0	-20	0	0	0	I	4.3	0.0	0:18.60	kworker/u11:1-x
6069	ysato	20	0	47464	4564	3656	R	0.3	0.0	0:31.50	top
1	root	20	0	225900	7520	5056	S	0.0	0.1	0:03.50	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.01	kthreadd
3	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_gp
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_par_gp
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:0H-kb
9	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_percpu_wq
10	root	20	0	0	0	0	S	0.0	0.0	0:04.25	ksoftirqd/0

基盤：そして終了。

The screenshot displays a system monitoring interface with three main sections: System Status, Hardware Information, and Resource Monitor. Below these are two terminal windows. The first terminal window shows the execution of a 'time' command for 'autoninja' and the output of 'date'. The second terminal window shows the output of the 'top' command, including system load averages, task counts, CPU usage, and a list of running processes.

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
6069	ysato	20	0	47464	4564	3656	R	0.7	0.0	0:31.74	top
1	root	20	0	225900	7520	5056	S	0.0	0.1	0:03.50	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.01	kthreadd
3	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_gp
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_par_gp

基盤：3時間40分の長旅でした。

開発：てかもう9時じゃないですか。

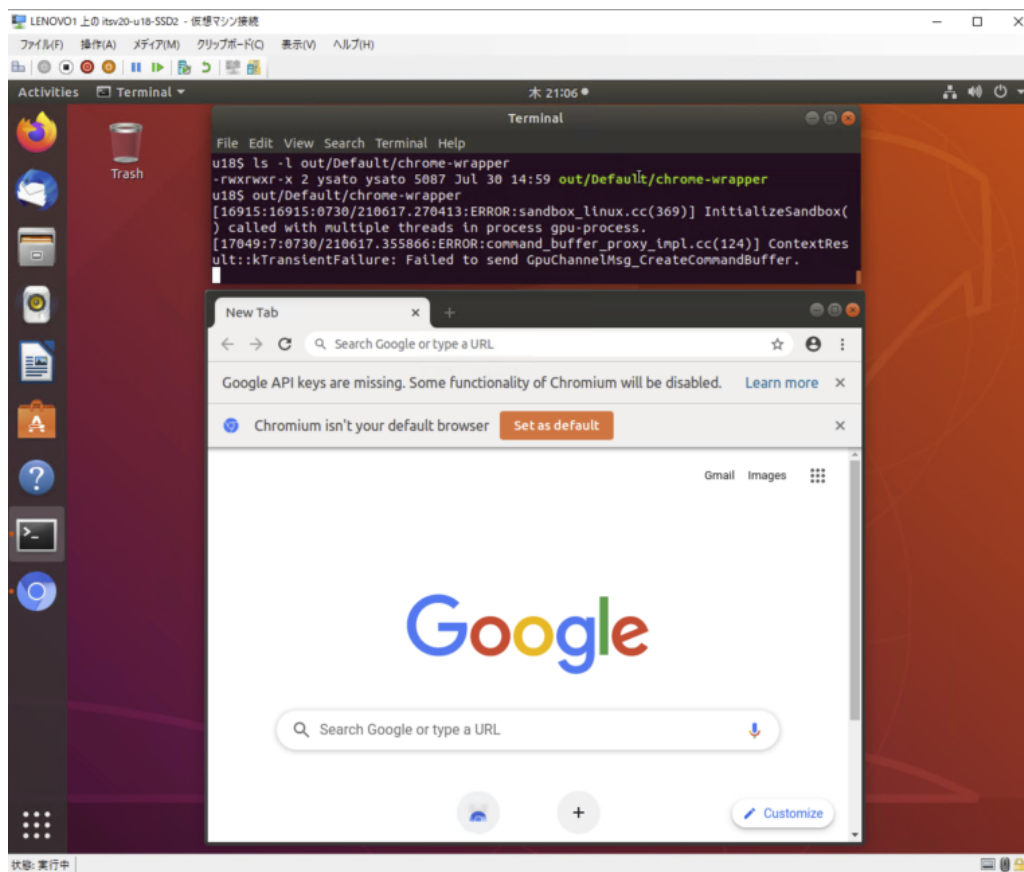
社長：いっぷくしましょう。

* * *

社長：いっぷくしてて思いついたんですが、mozilla と zsh をくっつけて mozsh とかどうですかね。xso で mozsh.jp を検索… ほら、無いから今日なら500円ですよ。

経理：まあ xso はしょっちゅう500円セールしてますけどね。

基盤：まあ、せっかく出来たので起動してみましようよ…



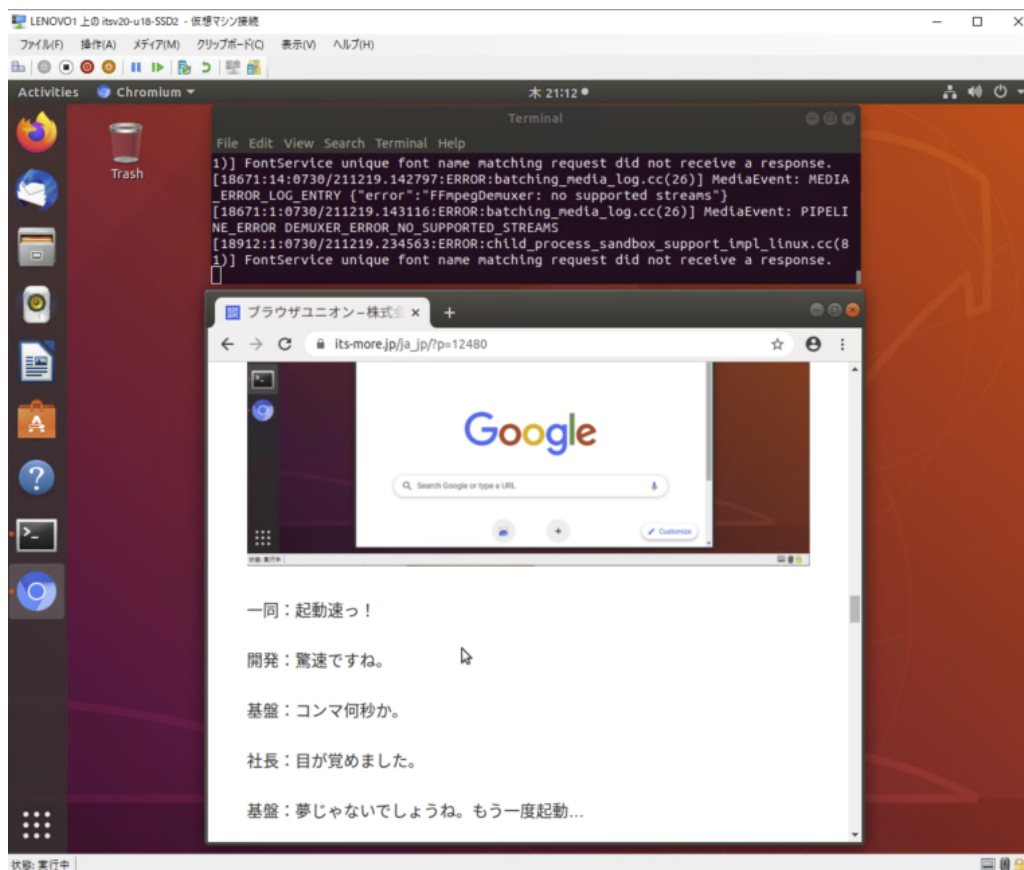
一同：起動速っ！

開発：驚速ですね。

基盤：コンマ何秒か。

社長：目が覚めました。

基盤：夢じゃないでしょうね。もう一度起動…



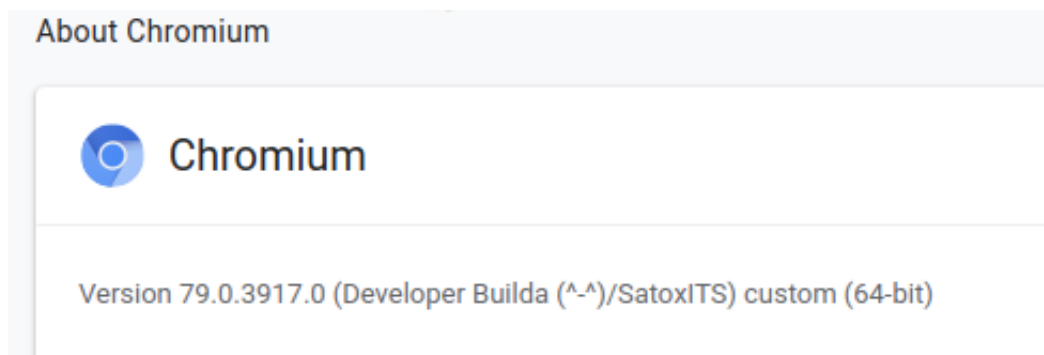
開発：これは使える。まったく使える。ふつうに使える。

社長：というか、macOS版がなんであんなに遅いのかよ、って事ですね。

開発：それではお約束のいたずらを。vi comp*/vers*p ... で autoninja -C ... うーん、やっぱりここをいじると再ビルドに10分くらいはかかりますね。

社長：うーん、我社はこれからは自家製 Chrome と自家製 zsh で生活ですかね。

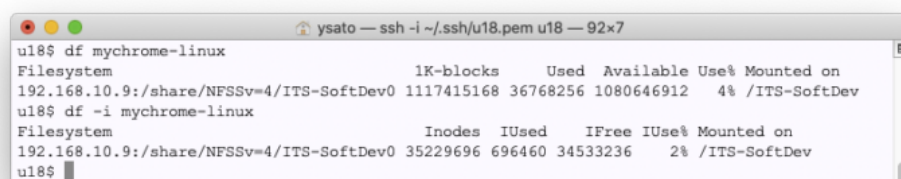
開発：できました。では起動。About Chromium ...



基盤：ぱちぱちぱち。

開発：「そのしゅん間、私は変わった」「私の心を大きく変化させた」「明るい未来が見えた気がした」 — Rika Kihira (2014)

基盤：しかし、ビルドの手引にはストレージ100GB用意しろとあるんですが、最終的にも34GBで終わってるんですよね。全体通してRAMだけでできちゃうサイズです。



```
ysato — ssh -i ~/.ssh/u18.pem u18 — 92x7
u18$ df mychrome-linux
Filesystem            1K-blocks    Used Available Use% Mounted on
192.168.10.9:/share/NFSSv=4/ITS-SoftDev0 1117415168 36768256 1080646912  4% /ITS-SoftDev
u18$ df -i mychrome-linux
Filesystem            Inodes    IUsed   IFree IUse% Mounted on
192.168.10.9:/share/NFSSv=4/ITS-SoftDev0 35229696 696460 34533236   2% /ITS-SoftDev
u18$
```

社長：なんにしても、Firefox と Chromium の開発環境が揃ったし、気分はバラ色の濡れ落ち葉ですね。飲みに行きますか。

— 2020-0730 SatoxITS