



Terminal With Your Life

ITOYANAGI Sakura
THE CONF 2018

Greeting



Hello, everyone!

Let me introduce myself



name

ITOYANAGI Sakura

GitHub

aycabta

maintainer

RDoc

Community: Asakusa.rb



Asakusa.rb every Ruby Tuesday

Company: Space Pirates, LLC.



Space Pirates, LLC.

I'm



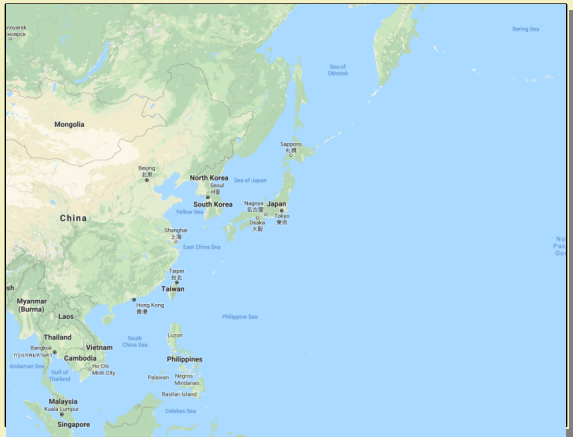
- from Japan
- a member of Ruby core team.

I'm



- **from Japan**
- a member of Ruby core team.

Japan



Japan



Brazil

Japan



Brazil is the backside of Japan

Japan



12742km(Through the earth)

Japan



Too far and too hard.

Japan



So many routes from Japan to Brazil.

Japan



So I decided to go to Brazil via Europe.

Japan



Europe

Hobby: Climbing



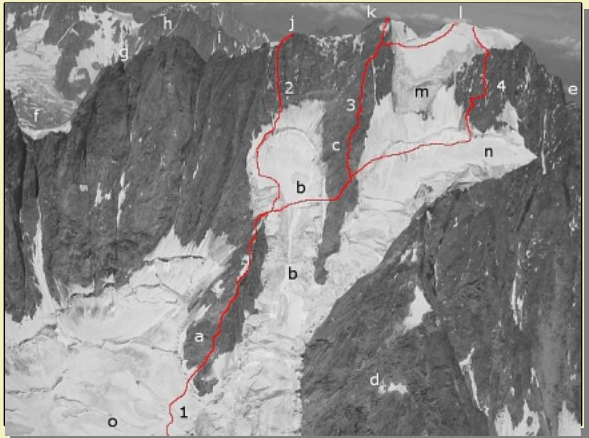
Mont Blanc mountain range

Hobby: Climbing



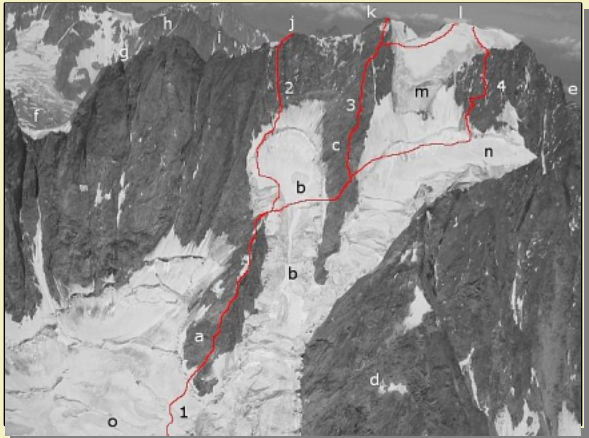
Mont Blanc, Grandes Jorasses

Hobby: Climbing



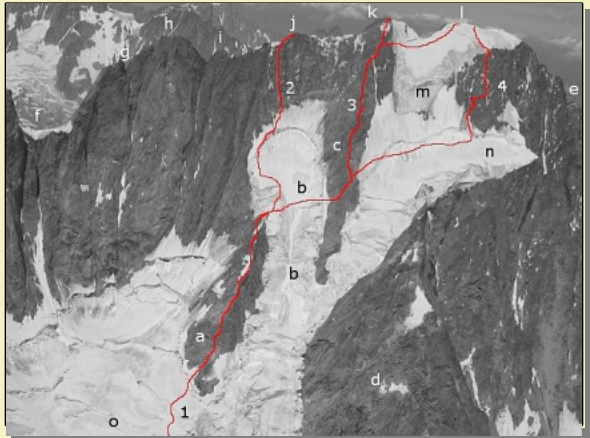
Grandes Jorasses via normale

Hobby: Climbing



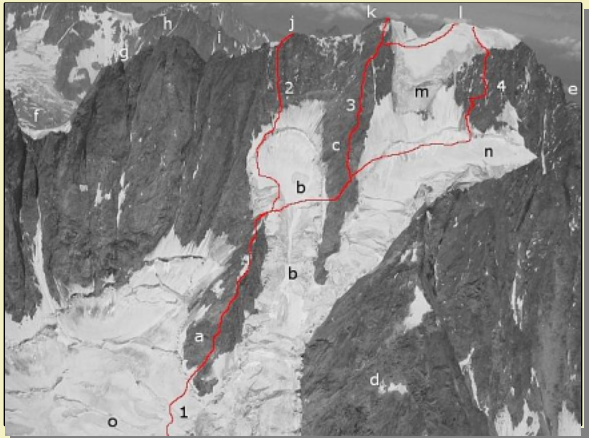
I climbed it, but...

Hobby: Climbing



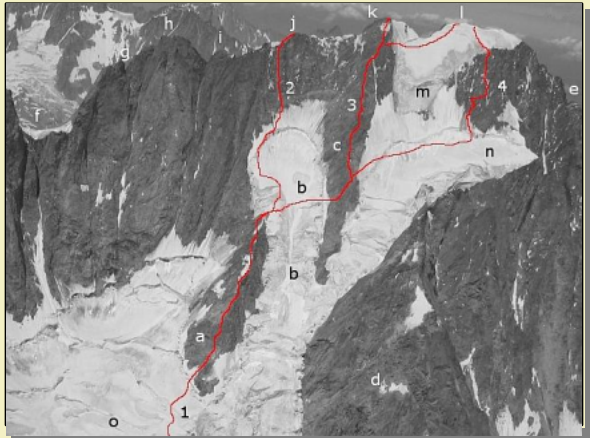
The glacier's condition was bad

Hobby: Climbing



I escaped from 3430m point

Hobby: Climbing



So I want to forget this climbing route as soon as possible

**Terminal
With
Your Life**



Terminal With Your Life



I'll talk about "confused terminal processing routes".

Development environment

What way do you use?

Development environment


- Console applications
 - shell, text editor, debugger, ...
- Browser(frontend)
 - with text editor
- IDE

Development environment

Console applications

 **shell, text editor,
debugger, ...**

Browser(frontend)

 with text editor

IDE

Console



On Unix like OSs:

- Users press key
- Keyboard driver
- Terminal emulator
- Line discipline
- TTY driver
- You see character!

Console



On Unix like OSs:

- **Users press key**
- Keyboard driver
- Terminal emulator
- Line discipline
- TTY driver
- You see character!

Console



On Unix like OSs:

- Users press key
- **Keyboard driver**
- Terminal emulator
- Line discipline
- TTY driver
- You see character!

Console



On Unix like OSs:

- Users press key
- Keyboard driver
- **Terminal emulator**
- Line discipline
- TTY driver
- You see character!

Terminal emulator



The rudimentary "terminal" is
a good mix of

- typewriter
- cable
- printer

Terminal emulator



Old teletype

Terminal emulator



Back then, all teletypes used different "characters and control codes set" on terminal.

Terminal emulator



So Unix started to provide compatible "terminal emulator" layer that is based on VT100.

Terminal emulator



VT100, famous real terminal

Terminal emulator



VT100 is introduced in August 1978 by DEC(Digital Equipment Corporation).

Terminal emulator



Most Unix like operating systems still has VT100 compatible terminal emulator.

Console



On Unix like OSs:

- Users press key
- Keyboard driver
- Terminal emulator
- **Line discipline**
- TTY driver
- You see character!

Line discipline



This is a simple "line editor" in kernel.

Line discipline



Two mode in line discipline:

- cooked(or canonical) mode
- raw mode

Line discipline: cooked (or canonical) mode



- Backspace
- Clear line
- Redraw
- ...

Line discipline: cooked (or canonical) mode



But this is too simple in the present day.

Line discipline: raw mode



So, this "raw mode" is important. I'll talk about this later.

Console



On Unix like OSs:

- Users press key
- Keyboard driver
- Terminal emulator
- Line discipline
- **TTY driver**
- You see character!

TTY driver



This is for multi "terminal emulator" with "session". This is for multi users at first, but this is for multi terminals in a day and age.

TTY driver



Do you use terminal
multiplexer?

- GNU Screen
- tmux
- ...

TTY driver



The terminal multiplexers handle the TTY drivers.

TTY driver



And these applications
connect TTY driver:

- Terminal.app(macOS)
- PuTTY(Windows)
- xterm(X Window System)
- ...

Console



On Unix like OSs:

- Users press key
- Keyboard driver
- Terminal emulator
- Line discipline
- TTY driver
- **You see character!**

You see character!



Congratulation! You are the
computer master!

Line discipline: raw mode



Tha get back to what I was saying...

Line discipline: raw mode

Some terminal applications
has so powerful user
interfaces:

- Emacs like key bindings
- History
- Complement

Line discipline: raw mode

In many cases, the "rich" user interface is provided by "GNU Readline" with "raw mode" of line discipline.

GNU Readline



I think that it's "brand-new
cooked mode line discipline
on application layer".

GNU Readline



Do you use GNU Readline?

GNU Readline



This is powerful line editor for terminal emulator. It's used by:

- shell(tcsh, Bash, Zsh, ...)
- REPL(IRB, Python Interpreter, ...)
- MySQL command-line tool
- The GNU Project

GNU Readline



Do you use GNU Readline?

GNU Readline



And now, I'm a member of
Ruby core team.

GNU Readline



Do you use Ruby?

GNU Readline on Ruby



Ruby has "readline" standard library, but it needs GNU Readline library **when you build Ruby.**

GNU Readline on Ruby



You should install GNU
Readline before installing
Ruby.

GNU Readline on Ruby



The "you build Ruby" means:



By hand



```
cd ruby-2.5.1 && ./configure  
&& make
```



By rbenv



```
rbenv install 2.5.1
```



...

GNU Readline on Ruby



How to install GNU Readline:



macOS



`brew install readline`



Ubuntu GNU/Linux



`sudo apt-get install
libreadline-dev`



FreeBSD



`sudo pkg install editline`

GNU Readline on Ruby



If you build Ruby without GNU Readline, Ruby **doesn't provide readline library.**

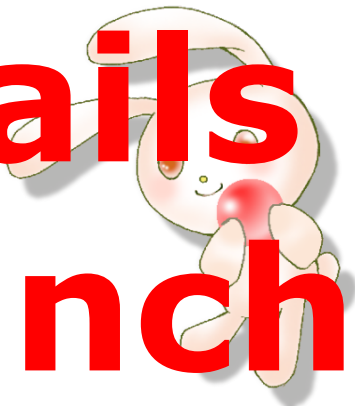
GNU Readline on Ruby



If Ruby doesn't provide
readline library:

- IRB can not completes
classes, modules, methods
and variables
- Pry **fails to launch**

Pry fails
to launch



Pry fails to launch



One day, a child receive a new computer as a birthday present from tender parents.

Pry fails to launch



The child says, "Thank you my tender parents, can I install Ruby to my great machine?"

Pry fails to launch



Tender parents nod in satisfaction.

Pry fails to launch



The child presses commands below:

```
$ wget http://.../ruby-3.0.0-preview100.tar.gz
$ tar xvzf ruby-3.0.0-preview100.tar.gz
$ cd ruby-3.0.0-preview100
$ ./configure
$ make
```

Pry fails to launch



```
$ irb
```

The child says, "can't
complete class names..."

Pry fails to launch



Tender parents say, "Ruby core team is developing many standard tools, you should use Pry".

Pry fails to launch



```
$ pry
```

```
Sorry, you can't use Pry without Readline or a compatible library.
```

```
Possible solutions:
```

- * Rebuild Ruby with Readline support using `--with-readline`
- * Use the rb-readline gem, which is a pure-Ruby port of Readline
- * Use the pry-coolline gem, a pure-ruby alternative to Readline

Pry fails to launch



```
$ pry
Sorry, you can't use Pry without Readline or a compatible library.
Possible solutions:
* Rebuild Ruby with Readline support using `--with-readline`
* Use the rb-readline gem, which is a pure-Ruby port of Readline
* Use the pry-coolline gem, a pure-ruby alternative to Readline
```

The child says, "so sad"

Pry fails to launch



LAPTOP

Pry fails to launch



TERMINATOR

Pry fails to launch



TERMINATOR: "Please wait,
I'm developing pure Ruby
Readline compatible library for
Ruby core"

Pry fails to launch



TERMINATOR: "It's Reline"

Pry fails to launch



The child: "Really?"

Pry fails to launch



TERMINATOR: "Yes"

Reline



Reline is the Readline compatible pure Ruby library for Ruby core.

Reline



If you build Ruby without GNU Readline, "readline" standard library uses Reline inside.

...maybe next Ruby 2.6...

Reline



If you build Ruby without GNU Readline, "readline" standard library uses Reline inside.

...maybe next Ruby 2.6...or...
2.7...

Reline



If you build Ruby without GNU Readline, "readline" standard library uses Reline inside.

...maybe next Ruby 2.6...or...
2.7...I can not promise...

Reline



I'll explain Reline
implementation.

Reline



Reline changes line discipline to "raw mode".

Reline



Reline handles all console screen:

- line editing
- completion
- Emacs binding
- settings
by `.inputrc`(including changes to vi binding)

Reline



And, Ruby core team values a Windows support, so Reline uses Windows API on Windows.

Reline



This is exactly what Ruby core team's works because this contains many "Unicode" handling for String class.

Reline



The child: "I'm glad to hear that"

Reline



But, original IRB author, keiju-san, is developing "Reidline". It's **multiline** editor like browser's JavaScript console for new IRB.

Reline



I'll join the development of
Reidline after Reline.

Ruby's line editors that I talked today



- GNU Readline
- readline(Ruby's standard library)
- Reline(pure Ruby re-implementation)
- Reidline(multiline editor library)

Ruby's line editors that I talked today



Maybe, Ruby 3,

Ruby's line editors that I talked today



Maybe, Ruby thre...future
Ruby must be greater more.

Ruby's line editors that I talked today



So you should NOT remember the complex names.

- GNU Readline
- readline(Ruby's standard library)
- Reline(pure Ruby re-implementation)

Thank you



- Terminal is complex
- TERMINATOR solves it
- Bright future of children with Ruby