

Sinatra::RocketIO



@shokai

私

- @shokai (しょうかい)

- 橋本 翔

- 学生

- 趣味：料理

- 好きなSNS：GitHub



Sinatra::RocketIO

Node.jsのSocket.IO

みたいなものを

Sinatraでも使いたい

Sinatra プラグイン

作るう

要求

- WebSocket & Comet
- C10K
- カッコイイコード

結果

- WebSocket & Comet

- 自動選択、切れても再接続

- C10K

- 1プロセスで15000クライアント接続

- カッコいいコード

- マジかっこよくなった (主観)

- Ruby 1.8.7 ~ 2.0.0
- Chrome, Safari, IE10, Android, iOS
- EventMachine & jQueryに依存
 - em-websocket + sinatra/streaming
 - Thin

Sinatra

```
% gem install sinatra
```

```
require "sinatra"
```

```
get "/" do  
  "Hello world."
```

```
end
```

Sinatra::RocketIO

```
% gem install sinatra-rocketio
```

```
require "sinatra"  
require "sinatra-rocketio"  
io = Sinatra::RocketIO  
  
io.on :connect do |client|  
  io.push :message, "Hello world."  
end
```

もう少し詳しく
説明します

setup

Server Side

```
require 'sinatra'  
require 'sinatra/rocketio'  
run Sinatra::Application
```

```
io = Sinatra::RocketIO
```

Client Side

```
<script src="/jquery.js"></script>  
<script src="<%= rocketio_js %>"></script>
```

```
var io = new RocketIO().connect();
```

接続できたので

データを

やりとりしたい

on & push

受信登録

`io.on("event_name", callback)`

送信

`io.push("event_name", data)`

Server --(WebSocket/Comet)--> Client

サーバープッシュ

Server Side

```
io.on :connect do |client|  
  puts "new client - <#{client.session}>"  
  io.push :chat, { :user => "shokai", :msg => "hello" }  
end
```

Client Side

```
io.on("chat", function(data){  
  alert(data.user + " : " + data.msg);  
});
```

push (送信)

on (イベント登録)

Client --(WebSocket/Ajax)--> Server

クライアントからプッシュ

Client Side

```
io.on("connect", function(){  
  io.push("chat", {msg:"Hello!!!", user:"shokai"} );  
});
```

push (送信)

on (イベント登録)

Server Side

```
io.on :chat do |data, client|  
  puts client.session  
  puts "#{data['user']} - #{data['msg']}"  
end
```

on(イベント登録)

push(送信)

サーバー/クライアントで

同じように使える

入門

- README嫁

- <https://github.com/shokai/sinatra-rocketio>

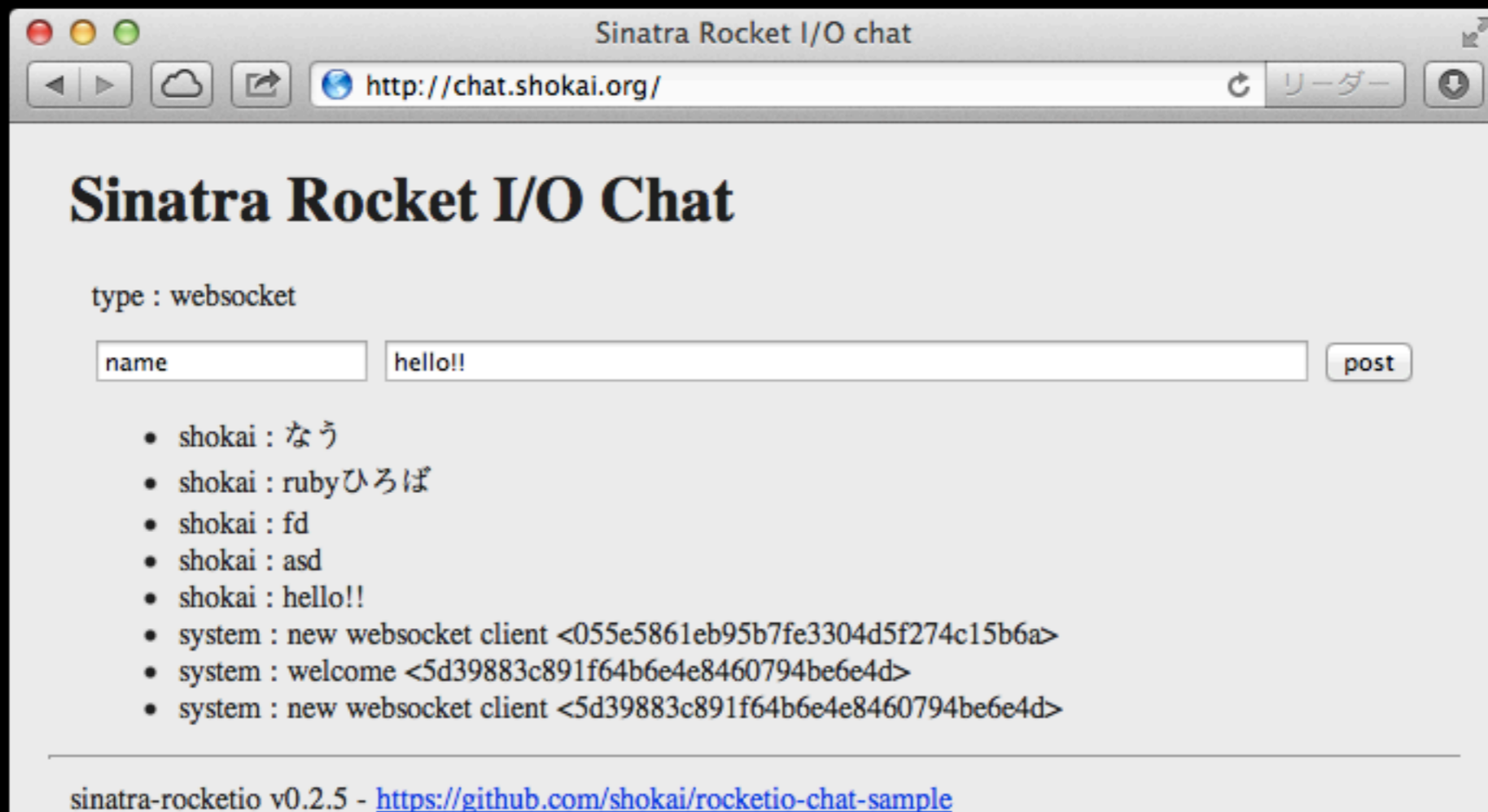
- Hello world

- <https://github.com/shokai/rocketio-hello-world>

アプリ例

ソースはGitHubに
あります

Chat



The screenshot shows a web browser window titled "Sinatra Rocket I/O chat". The address bar contains "http://chat.shokai.org/". The page content includes a title "Sinatra Rocket I/O Chat", a form with a "name" field and a "post" button, and a list of chat messages. The messages include user greetings and system messages about websocket clients.

Sinatra Rocket I/O chat

http://chat.shokai.org/ リーダー

Sinatra Rocket I/O Chat

type : websocket

name post

- shokai : なう
- shokai : rubyひろば
- shokai : fd
- shokai : asd
- shokai : hello!!
- system : new websocket client <055e5861eb95b7fe3304d5f274c15b6a>
- system : welcome <5d39883c891f64b6e4e8460794be6e4d>
- system : new websocket client <5d39883c891f64b6e4e8460794be6e4d>

sinatra-rocketio v0.2.5 - <https://github.com/shokai/rocketio-chat-sample>

Ruby+JS 合計40行で チャット作れる

```
1. sho@sho-mba:/Users/sho/projects/rocketio/rocketio-chat
File Edit Options Buffers Tools Javascript YASnippet Help
io = Sinatra::RocketIO

io.on :chat do |data, client|
  puts "#{data['name']} : #{data['message']} (from:#{client.session}, type:#{client.type})"
  io.push :chat, data
end

io.on :connect do |client|
  puts "new client <#{client.session}> (type:#{client.type})"
  io.push :chat, {:name => "system", :message => "new #{client.type} client <#{client.session}>"}
  io.push :chat, {:name => "system", :message => "welcome <#{client.session}>"}, {:to => client.session}
end

io.on :disconnect do |client|
  puts "disconnect client <#{client.session}> (type:#{client.type})"
  io.push :chat, {:name => "system", :message => "bye <#{client.session}>"}
end

get '/' do
  haml :index
end

- UU-: **--F1 main.rb All L1 Git-master (Ruby yas AC) - UU-: **--F1 index.js All L1 Git-master (Javascript yas
Beginning of buffer

$(function(){
  $("#chat #btn_send").click(post);
  $("#chat #message").keydown(function(e){
    if(e.keyCode == 13) post();
  });
});

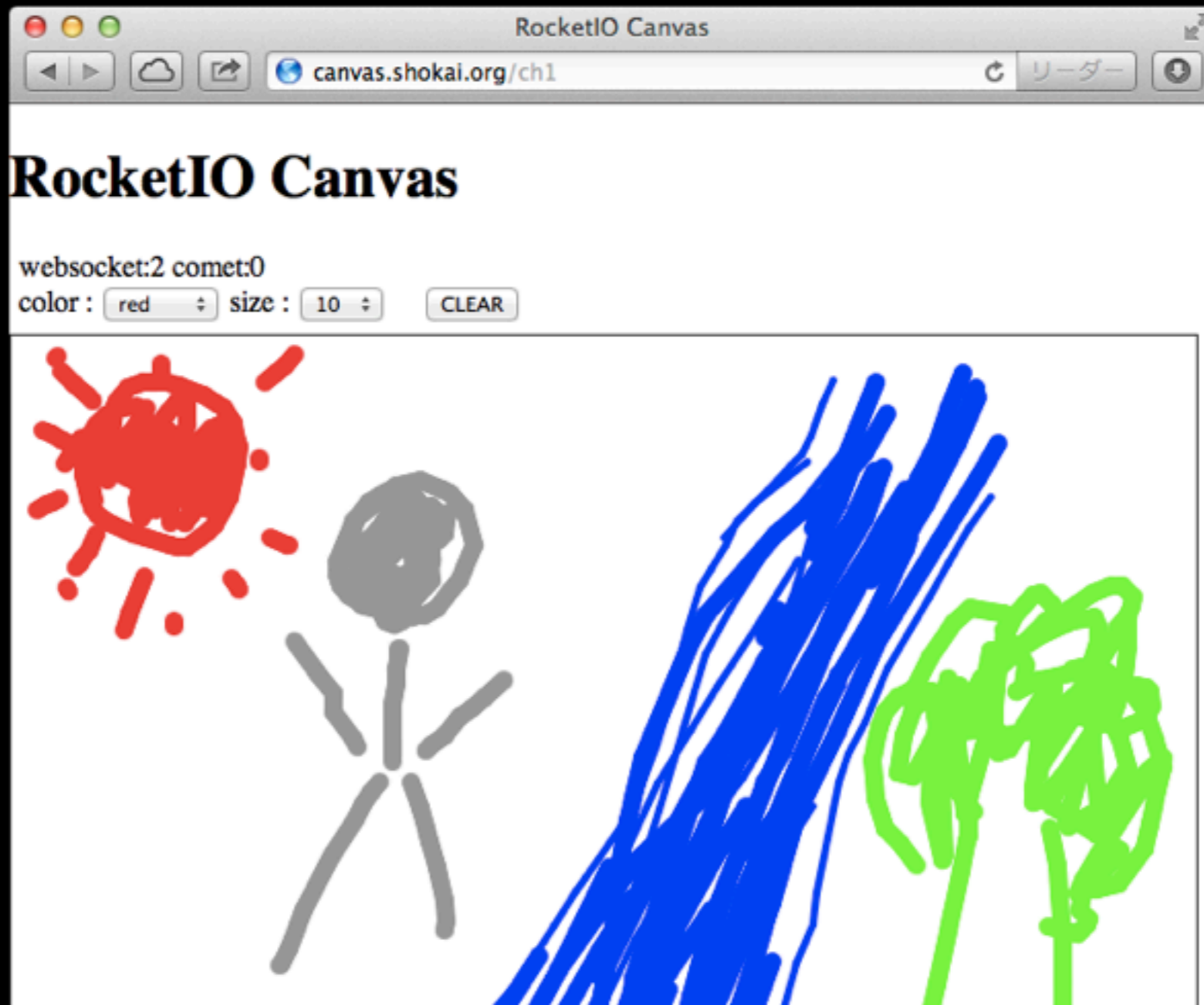
var io = new RocketIO().connect();

io.on("chat", function(data){
  var m = $("#<li>").text(data.name + " : " + data.message);
  $("#chat #timeline").prepend(m);
});

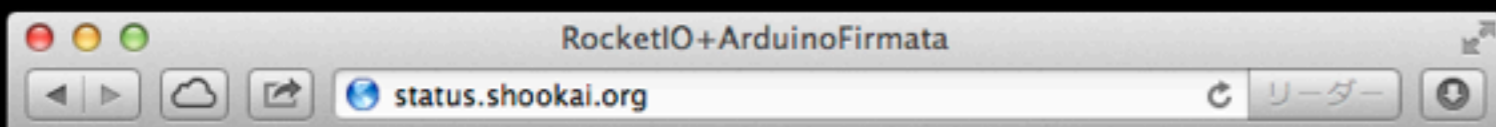
io.on("connect", function(){
  console.log("connect!! "+io.session);
  $("#type").text("type : "+io.type);
});

var post = function(){
  var name = $("#chat #name").val();
  var message = $("#chat #message").val();
  if(message.length < 1) return;
  io.push("chat", {name: name, message: message});
  $("#chat #message").val("");
};
```

Canvas



家のArduinoを見る



接続:websocket ..

気温:21.97265625度

明るさ:84

CPU:28.2%

メモリ:1.4%

websocket:1

comet:0



クワイアアント

ライブラリ

- `sinatra/rocketio/client`
 - `sinatra-rocketio` gemに付属
- `em-rocketio-client`
 - pure EventMachine実装
- `rocketio.js`
 - `sinatra-rocketio` gemに付属
 - ブラウザ拡張などにどうぞ

C10K

クライアントたくさん

接続する設定

/etc/security/limits.conf

```
shokai    soft    nofile    65536
shokai    hard    nofile    65536
```

config.ru

```
case RUBY_PLATFORM
when /linux/i then EM.epoll    ## for Linux
when /bsd/i   then EM.kqueue   ## for BSD
end
EM.set_descriptor_table_size 20000 ## 20K
```

<https://github.com/shokai/sinatra-websocketio/wiki/CIOK>

今後

- **Linda**という並列言語拡張を作ってる

- <http://shokai.org/blog/archives/7515>

- **Rack::RocketIO**にしたい

- Rack Hijacking API使いたい
- Railsでも使えるようになる

- **IE10**が**comet**で死ぬ

- 助けて

質問などあれば

- @shokai
- お気軽に
- お声かけください

