

HTTP・Sinatra

SCCP2015・S4教室

d8161105 渡部未来



◇ 今回の目的

- ▶ 前回のおさらい
- ▶ HTTPの基礎を理解する
- ▶ WebフレームワークSinatraの使い方を学ぶ

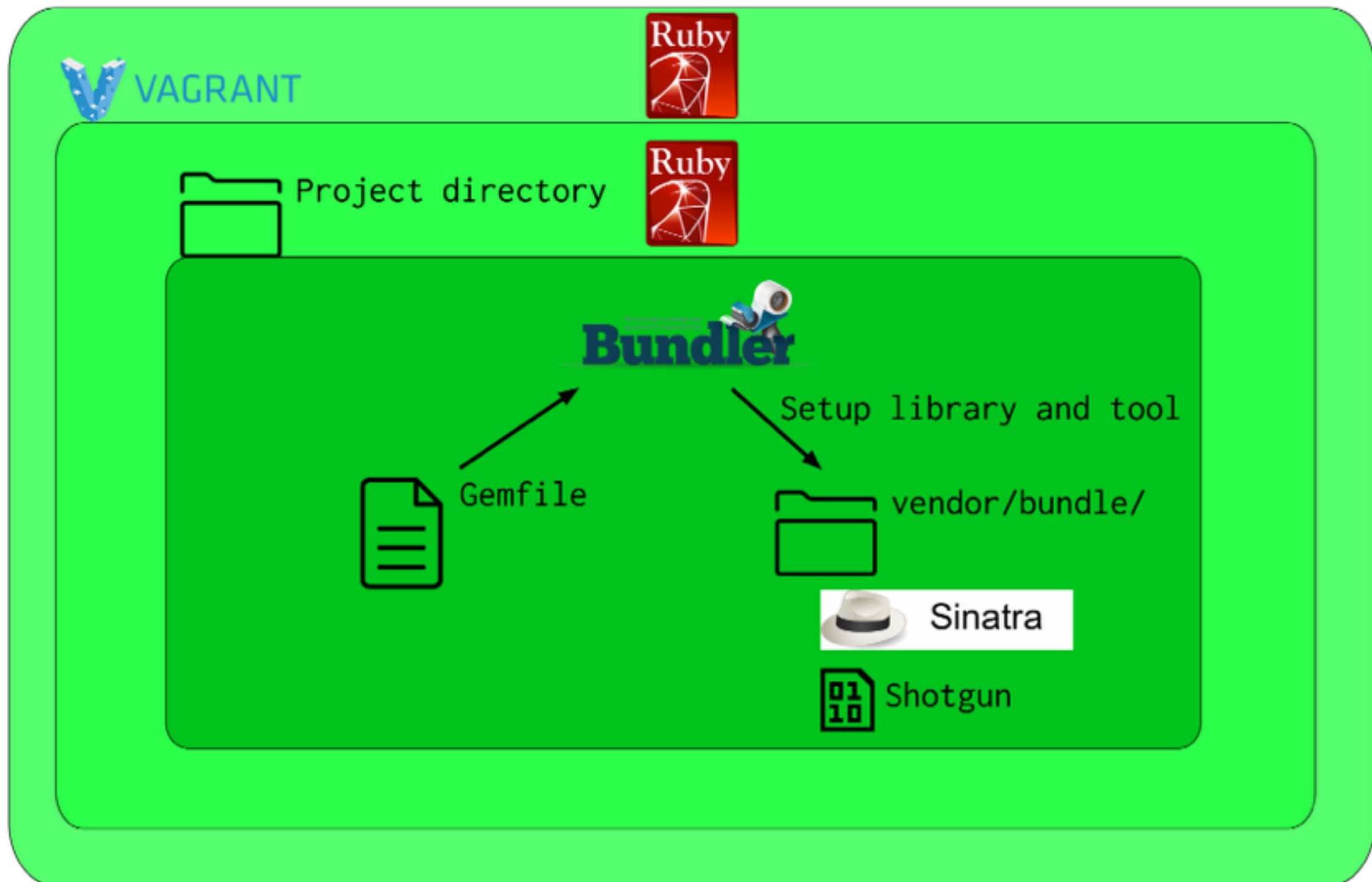
◇ 環境依存と問題点

- ▶ 環境(マシン・バーチャルマシン等)に対して、言語・ツール・ライブラリなどが入っている場合、**環境への依存性が高くなる。**
- ▶ 環境依存が高い状態で、プロジェクトが複数存在すると**ツール・ライブラリの競合状態が発生する可能性が高くなる。**

◇ 環境依存を無くすために

- ▶ 言語・ツール・ライブラリの依存性は、
ホストPC->VM(コンテナ)->プロジェクトディレク
トリの順に強くしていく。(完全にホストPCの依存
が無くなるのが理想)
- ▶ パッケージの管理や実行管理は、**ビルドツール**
(Makefile, Bundler, Gradle, Sbt等)にやらせる

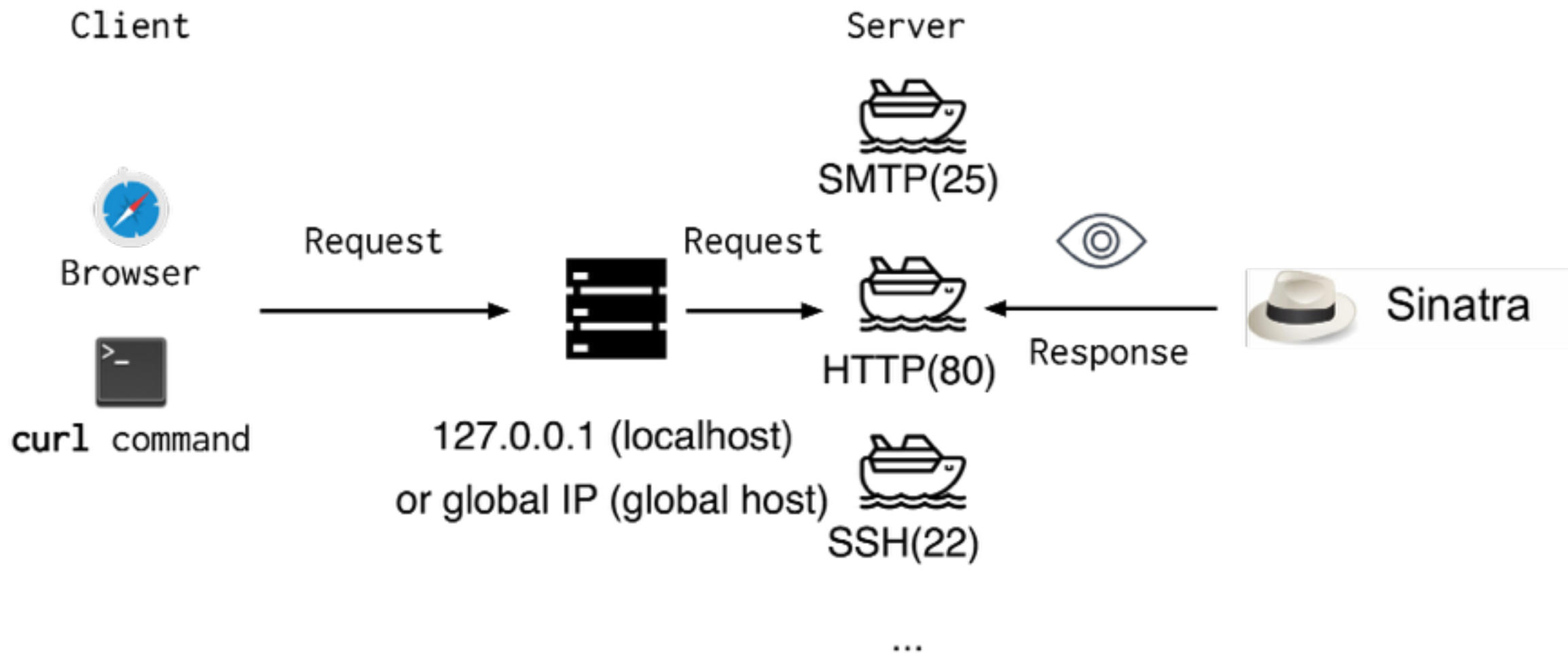
◇ 環境依存を無くすために



◇ HTTP

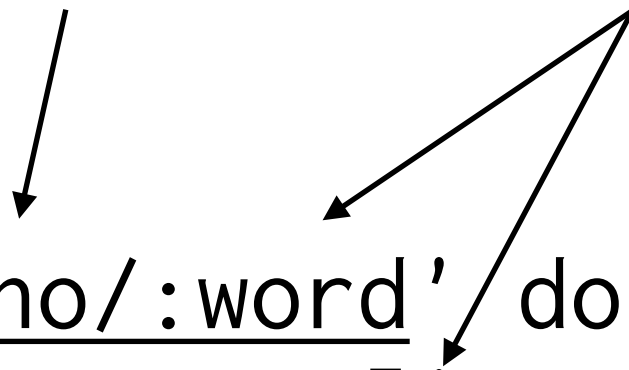
- ▶ HTTPは、通信プロトコルのひとつ
- ▶ クライアントからのリクエスト(要求)に対して、レスポンス(反応)を返す。
- ▶ HTTPポート(80)を監視し続け、リクエストに応じたレスポンスを返す仕組みを持ったWebフレームワークの一つがSinatra

◇ HTTP



◇ echoプログラム

- ▶ レスポンスを返すのは、結局はプログラム。自由に組むことが出来る。
- ▶ 重要なのはルーティングとパラメータ。



```
get '/echo/:word' do  
  "#{params['word']}"  
end
```

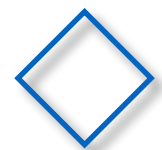
渡した文字をやまびこ(echo)するプログラム。

◇ いろんなレスポンス | 演習(step2)

- ▶ HelloWorld, echoプログラムを参考に3つ異なるレスポンスを返すプログラムを書け。

◇ メソッド

- ▶ HTTPは、**リソース**(URL・URI)に対して何をするか命令(メソッド)を選ぶことができる。
- ▶ **GET**(取得)・**POST**(送信)・**PUT**(更新)・**DELETE**(削除)
- ▶ POST・PUTのメソッドは、データをHTTPリクエストの**ボディ**に用意する必要がある。(GET・DELETEでも、データの送信自体は可能)



Sinatraにおけるメソッド分岐(step3)

```
get '/' do  
  'GET'  
end  
post '/' do  
  request.body.gets "\n"  
end  
put '/' do  
  request.body.gets + "\n"  
end  
delete '/' do  
  'DELETE'  
end
```

メソッド名

bodyの取得

◇ HTTPクライアント `curl`

- ▶ ブラウザのアクセスでは、基本GETリクエストしかだせないなので、いろいろなメソッドが試せない。
- ▶ 詳細な情報を確認しにくい。

メソッド名 ボディ

↓ ↓

```
$ curl -X POST -d hoge hoge http://localhost:9393/
```

レスポンスの詳細情報を取得

←

```
$ curl -X -i DELETE http://localhost:9393/
```

◇ データベース風Webサービスの実装| 演習(step4)

- ▶ Rubyのファイル操作を使って以下の仕様通りに実装せよ。データベース(テキストファイル)名はdataとする。
- ▶ GET → ファイル'data'の中身を返信。dataが存在しない場合は、空文字列を返信。
- ▶ POST → ファイル'data'にbodyで渡された文字列(改行を最後に含む)を追記。
- ▶ PUT → ファイル'data'の内容をbodyで渡された文字列で、書き換え。
- ▶ DELETE → ファイル'data'の中身を空に。

◇ データベース風Webサービスの実装| 宿題(step5)

- ▶ step4の実装を以下のようにリファクタリング(動作を変えずにコードを整理すること)すること。
- ▶ データベース操作(ファイル操作)を新しいクラスDatabaseクラスに分離せよ。
- ▶ Databaseクラスは、ファイル'data'のみを操作するクラスで構わない。

ヒント: シングルトンパターン or クラスメソッド

◇ まとめ

- ▶ VMや言語ごとのビルドツールを利用することで、環境依存を減らすか、無くすることができる。
- ▶ Sinatraは、HTTPプロトコルの機能を再現するためのWebフレームワークである。
- ▶ Sinatraでメソッド・ルーティングの振り分けや、Bodyの取得ができる。
- ▶ HTTPクライアントの一つcurlがある。