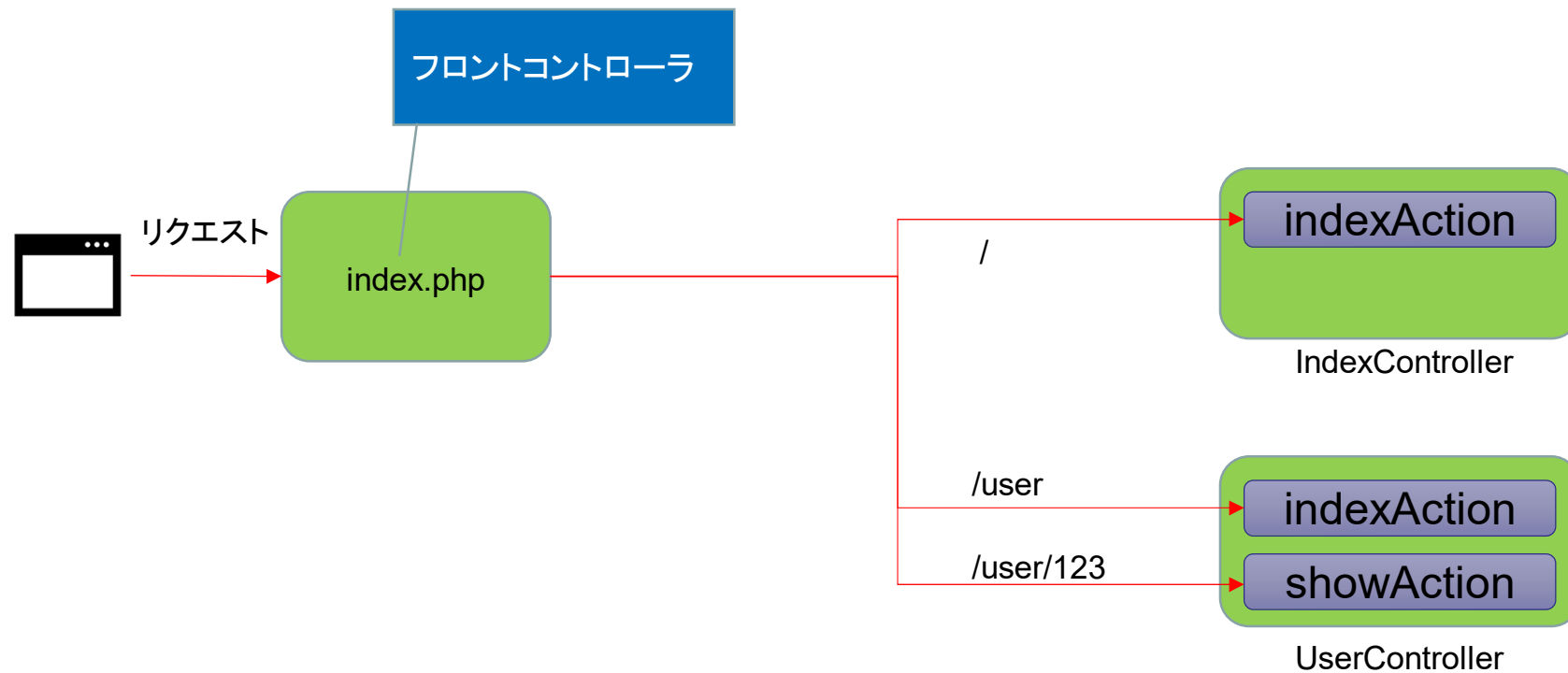


本ステップでおこなうこと

フロントコントローラを作ります。
index.phpですべてのHTTPリクエストを受け取り、
URL中のパス名に応じたコントローラー・メソッドを呼び出せるようにします。

- `http://enjoy-eats-step1:8888`
→ IndexControllerクラスのindexActionメソッドを呼び出す
- `http://enjoy-eats-step1:8888/user`
→ UserControllerクラスのindexActionメソッドを呼び出す
- `http://enjoy-eats-step1:8888/user/123`
→ UserControllerクラスのshowActionメソッドを呼び出し、
引数として「123」を渡す

本ステップでおこなうこと



ルーティング情報とディスパッチ(1)

パス名と、そのパス名に対応する処理をマッピングしたルール群のことをルーティング情報といいます。

/userにアクセス
されたら...

ルーティング情報

indexActionを呼び出す

/user/123にアクセス
されたら...

ルーティング情報

「123」の部分を引数
\$userIdにセットして、
showActionを呼び出す

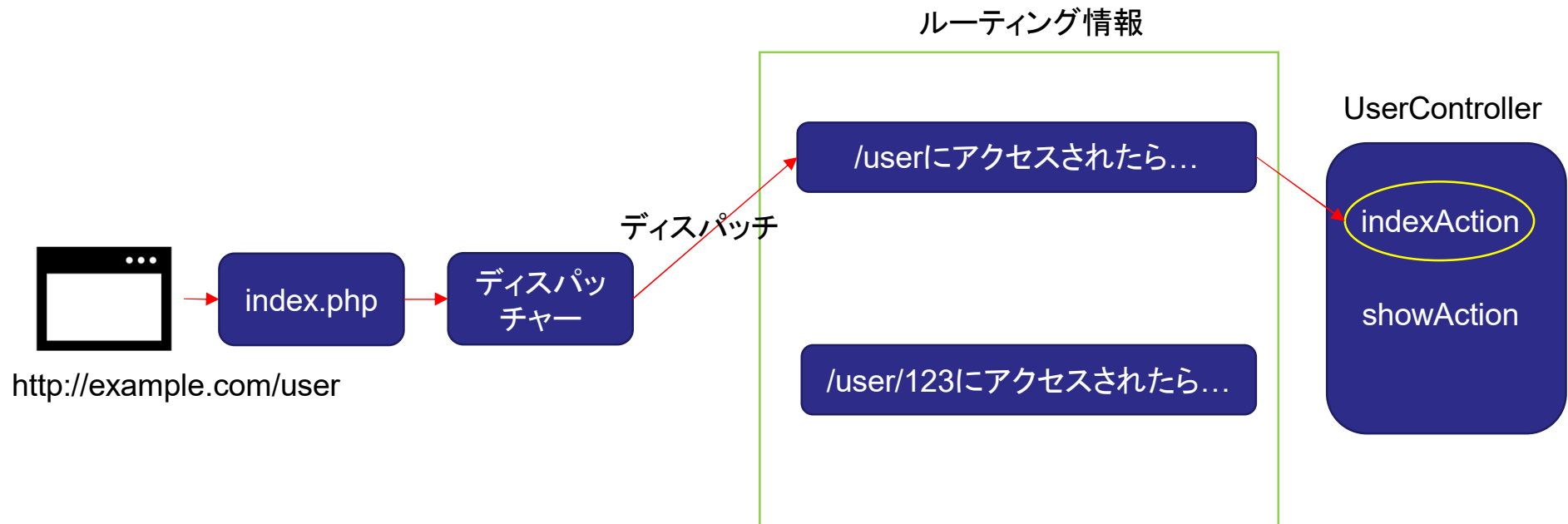
UserController.php

```
class UserController
{
    public function indexAction() {}

    public function showAction($userId) {}
}
```

ルーティング情報とディスパッチ(2)

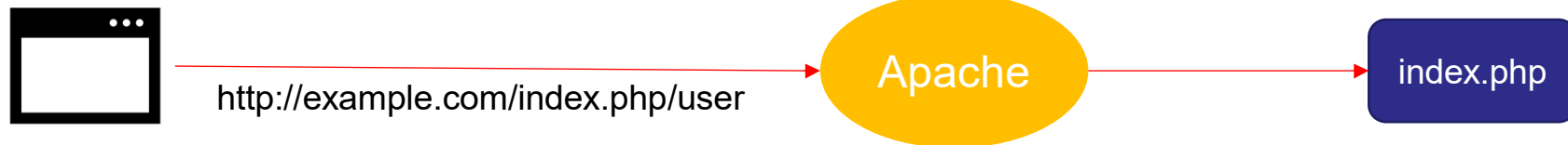
- アクセスされたパス名にマッチするルーティング情報を探しあて、そのルーティングルールに従った処理を行うことをディスパッチといいます。
- フロントコントローラが、アクセスされたパス名をディスパッチャーに渡します。



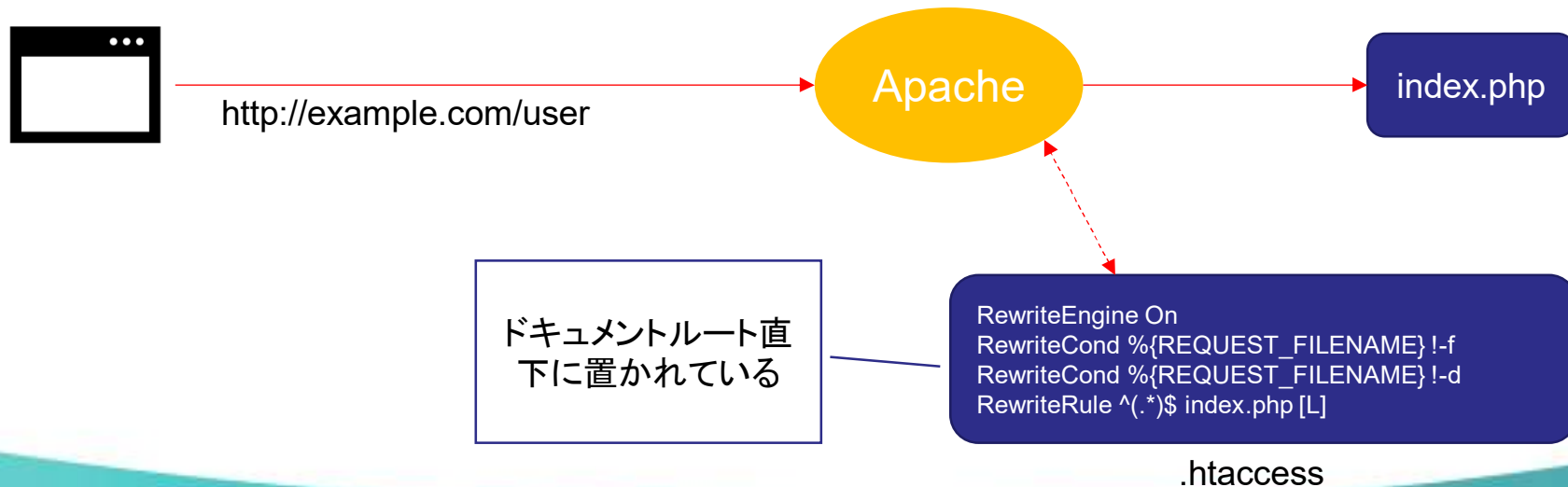
ルーティング情報とディスパッチ(3)

一般的に、アクセスするURLからindex.phpを省略するために.htaccessを使います。

●.htaccessがないときのアクセスURL



●.htaccessがあるときのアクセスURL

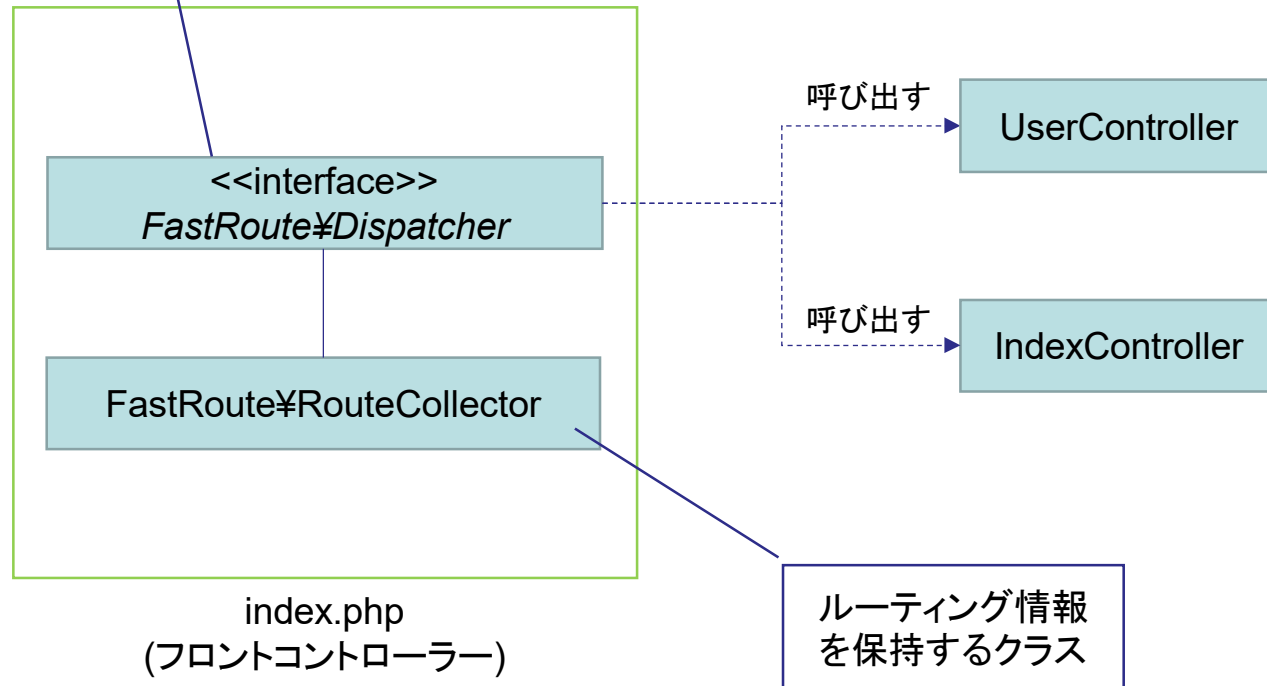


フロントコントローラの処理イメージ

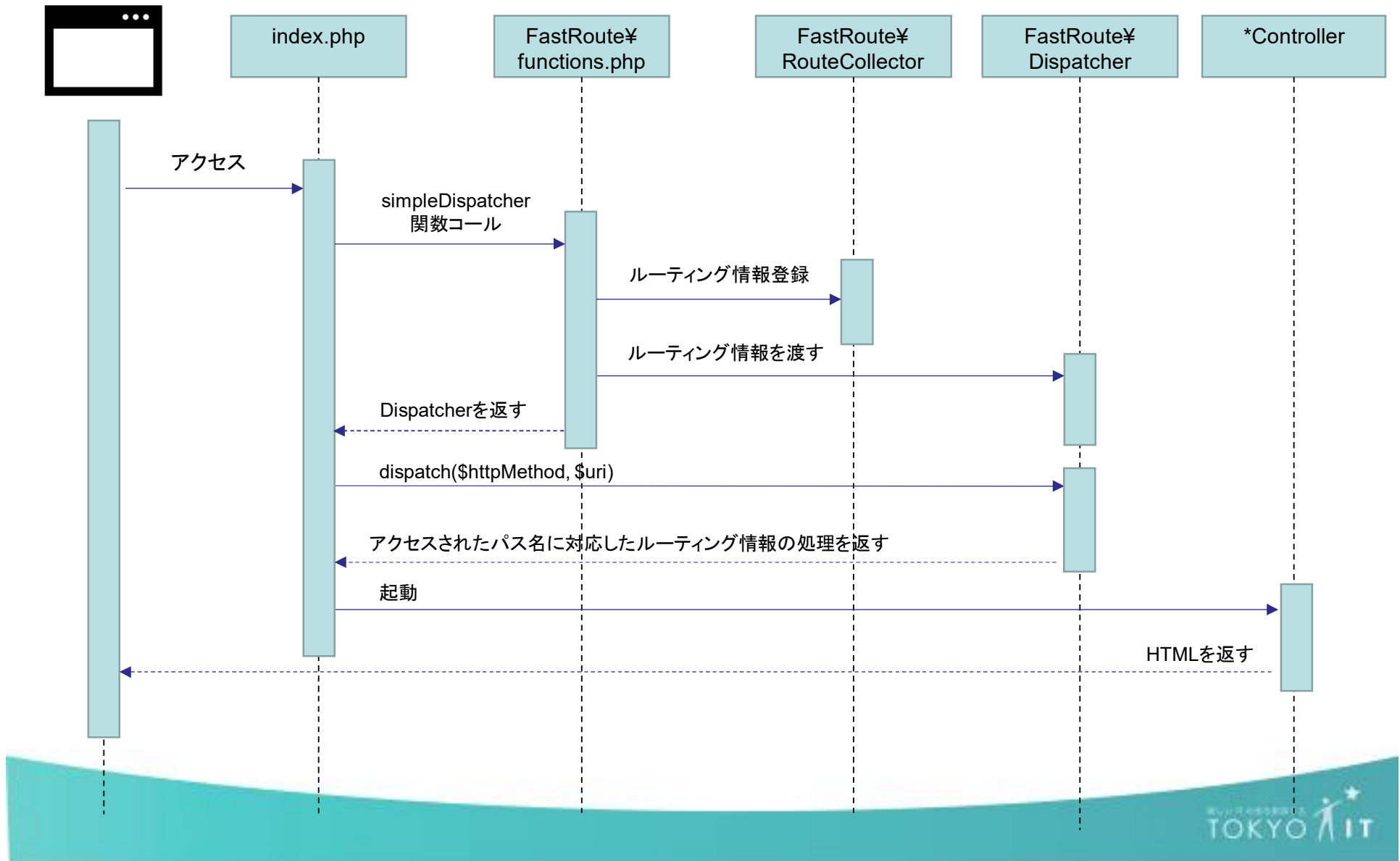
1. Webクライアントが、以下のURLにアクセスします。結果として、index.phpのプログラム処理が実行されます。
<http://example.com/user>
2. このときの`$_SERVER['REQUEST_URI']`の値は `/user` となります。
3. index.phpが、この値をディスパッチャーに渡すことで、コントローラクラスが起動します。

本ステップのクラス構成

ディスパッチャーのインターフェース。
このインターフェースを実装するクラスはいくつかありますが、ここではGroupCountBasedという実装クラスを使います



本ステップの処理の流れ



本ステップの変更ファイル一覧

●追加したファイル

- public/index.php
- public/.htaccess
- public/composer.json
- app/Modules/User/Controllers/UserController.php
- app/Modules/User/Controllers/IndexController.php

参考情報

- PHP本格入門(上)
「5-2-5 サーバー情報を取得する - \$_SERVER変数」