

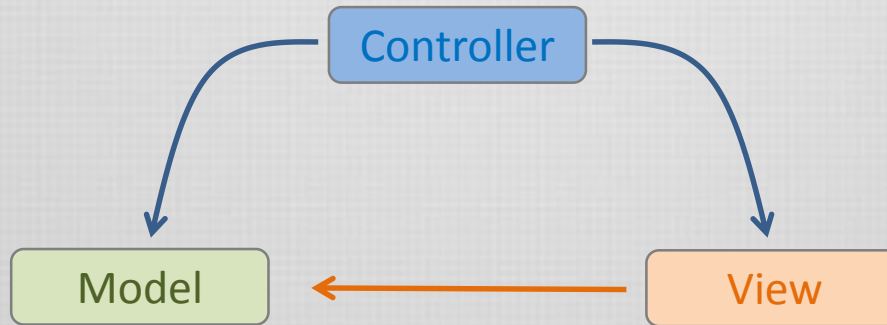
Ingeniería de Aplicaciones Web

Diego C. Martínez

Departamento de Ciencias e Ingeniería de la Computación
Universidad Nacional del Sur

Patrón general de diseño: MVC

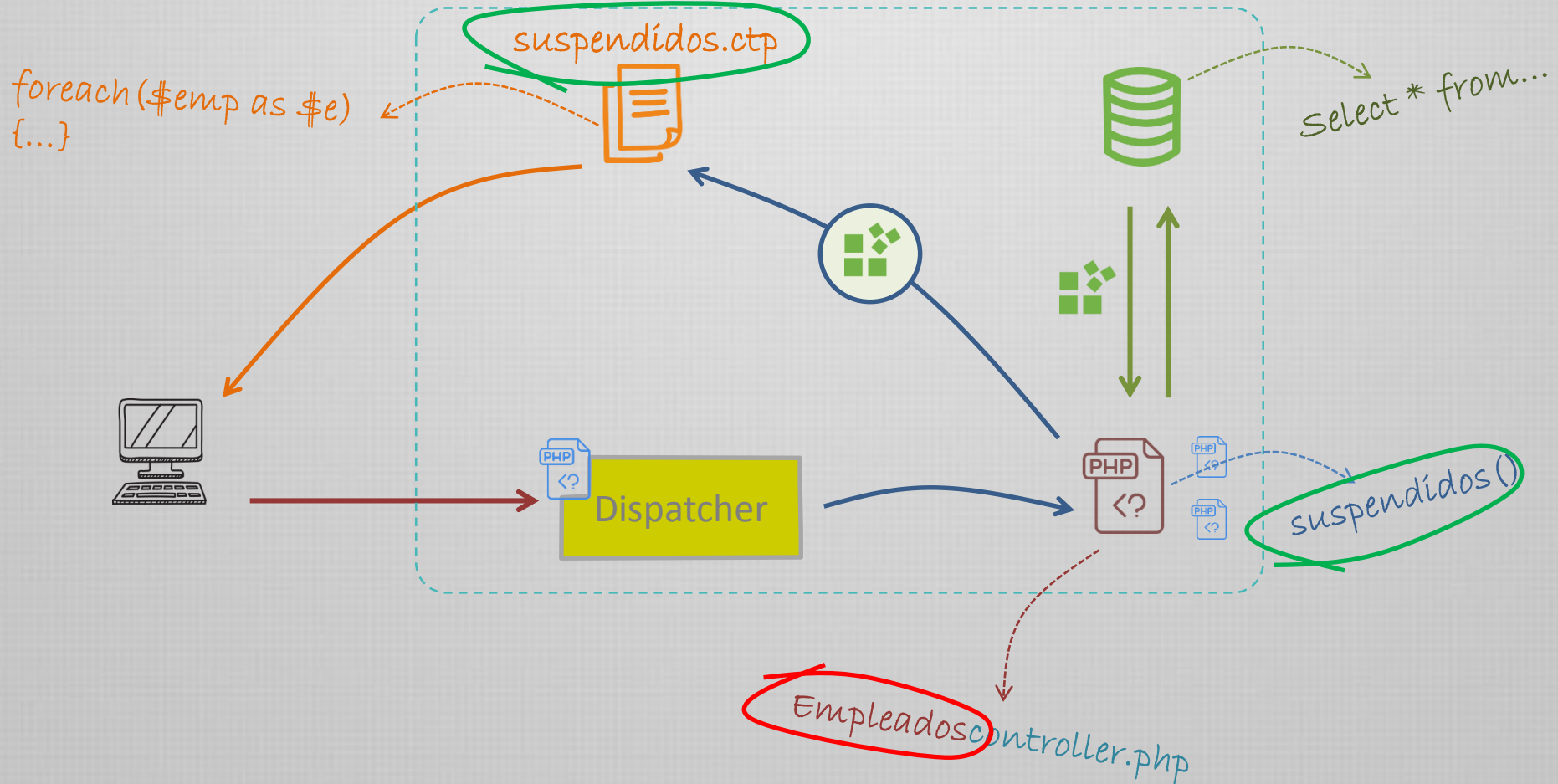
MVC es el patrón arquitectónico predominante en las aplicaciones web



*El **Controlador** administra el **Modelo** y la **Vista**.
La **Vista** es responsable de observar el **Modelo** para exteriorizar los datos.*

Web Frameworks: CakePHP

`http://nuestro.host.com/empleados/suspendidos`



Web Frameworks: CodeIgniter



Framework PHP, rápido y liviano
Más liberal que CakePHP en la aplicación del patrón MVC

Modelo

*representa las estructuras de datos y es usualmente
una interfaz de la base de datos*

No mandatorio!

Vista

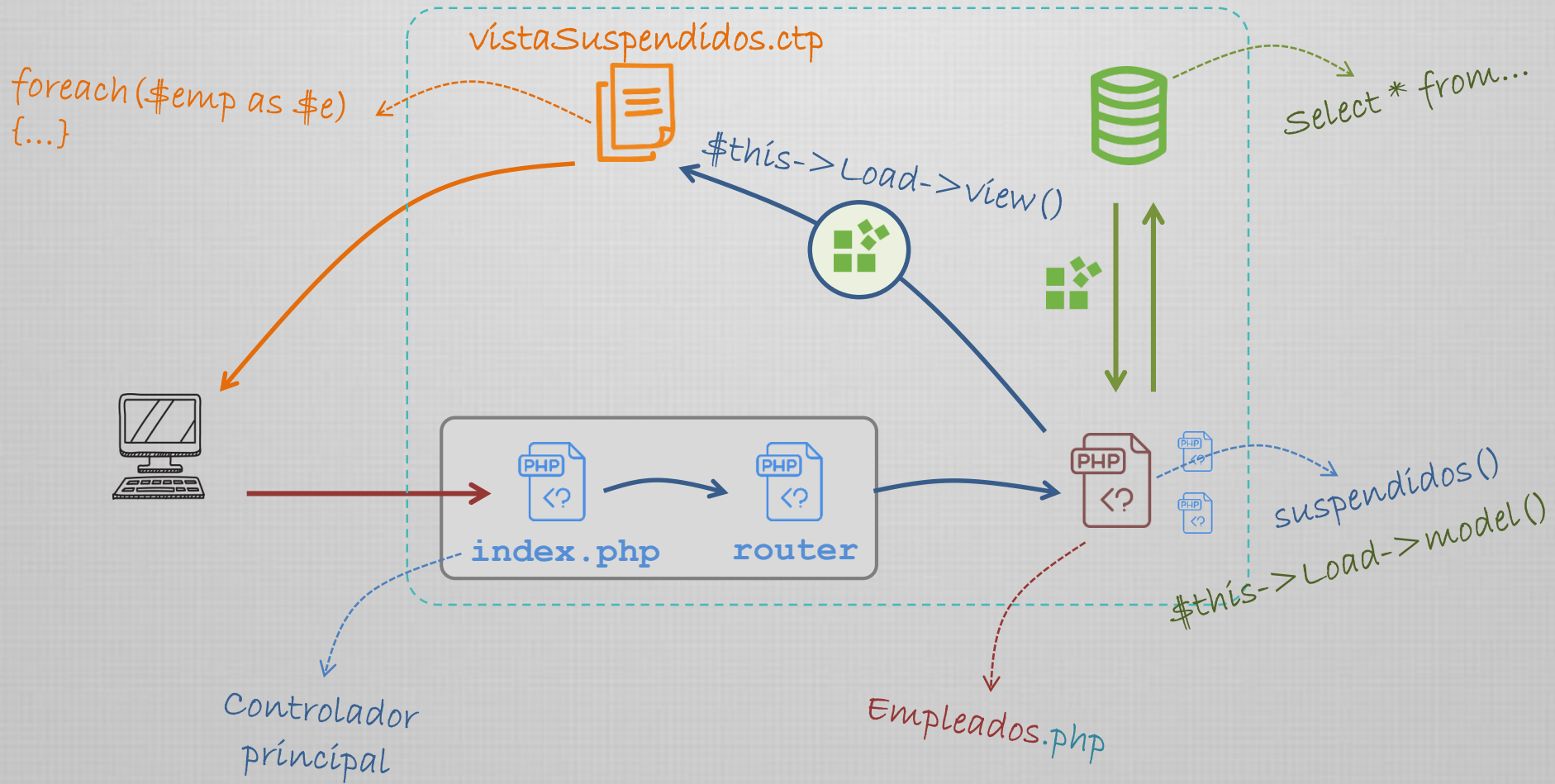
*representa la información presentable al usuario
(visual como en HTML o procesable como en RSS)*

Controlador

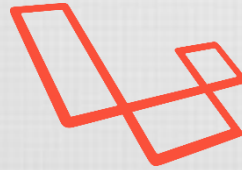
es el intermediario entre el modelo y la vista

Web Frameworks: Codeigniter

<http://nuestro.host.com/empleados/suspendidos>



Web Frameworks: [Laravel](#)



Framework PHP, basado en MVC
Actualmente uno de los frameworks mas populares



Creado por Taylor Otwell en 2011, como alternativa a CodeIgniter

Arquitectura bien definida

Rico ecosistema: Homestead, Valet, Lumen, etc

Comandos CLI

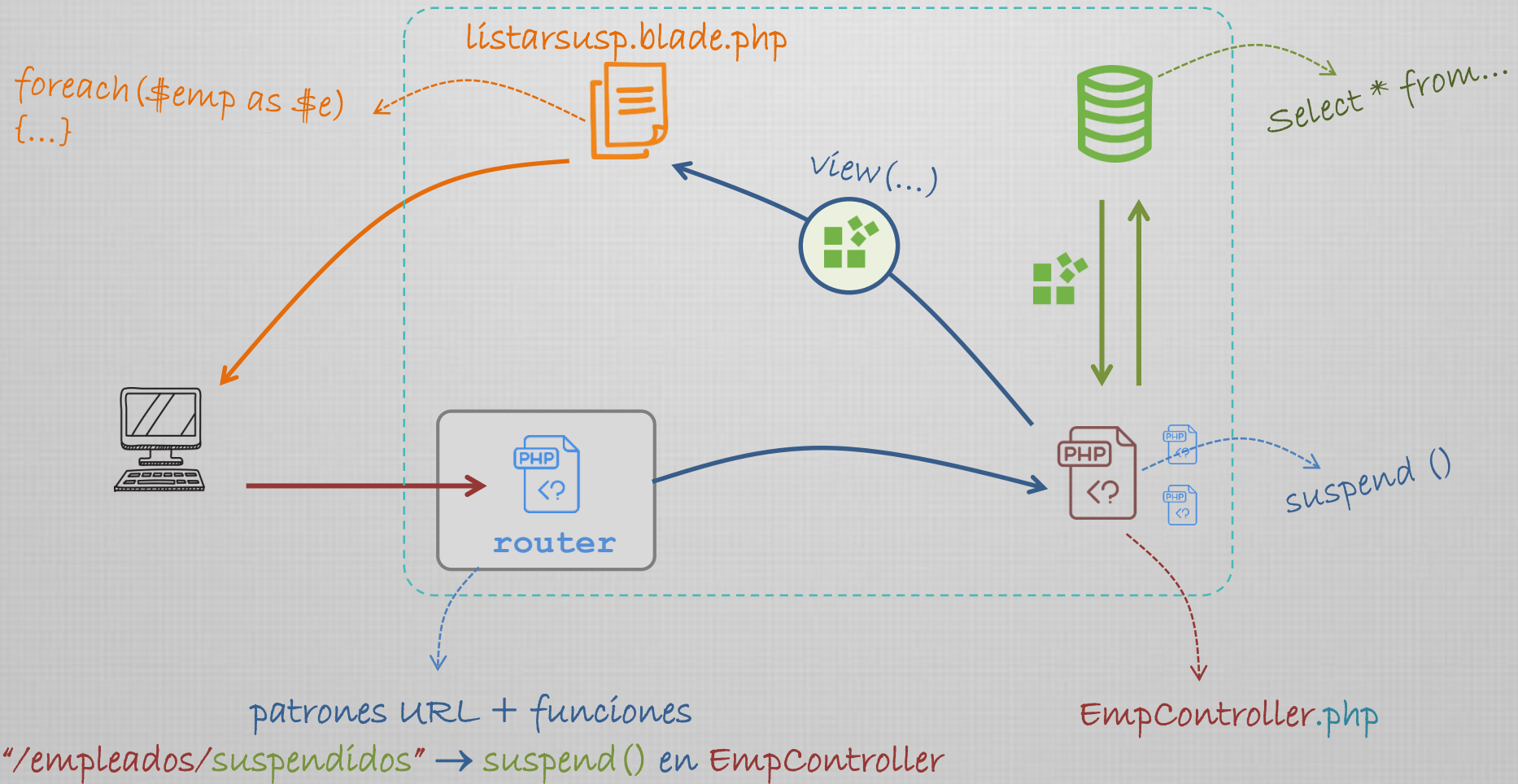
Mayor liberalismo en la implementación de modelos

Clara distribución del flujo de ejecución: route, middleware, controller, view

Templates propios: Blade

Web Frameworks: Laravel

`http://nuestro.host.com/empleados/suspendidos`



Web Frameworks: Spring MVC



Framework Java, que incluye una arquitectura MVC
Es un framework de aplicaciones amplio, con alcance web

Programación orientada a aspectos
Framework de acceso a datos (JDBC, Hibernate, etc)
Remote Access Framework
Aplicaciones web MVC

Web Frameworks: SpringMVC

src/main/java/hello/GreetingController.java

```
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestParam;

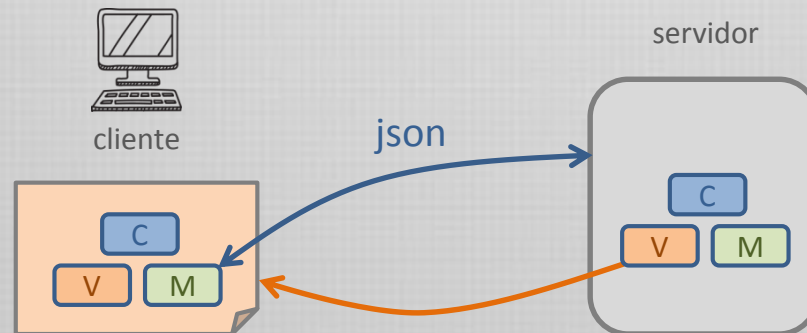
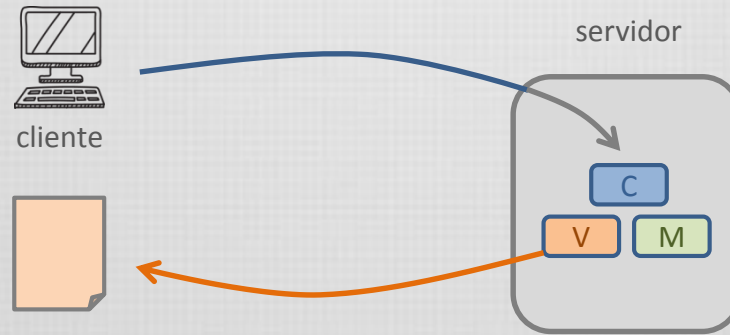
@Controller
public class GreetingController {

    @RequestMapping("/greeting")
    public String greeting(@RequestParam(value="name",required=false, defaultValue="Mundo")
                           String name, Model model) {
        model.addAttribute("name", name);
        return "greeting";
    }
}
```

src/main/resources/templates/greeting.html

```
<!DOCTYPE HTML>
<html xmlns:th="http://www.thymeleaf.org">
<head>
    <title>Saludo</title>
</head>
<body>
    <p th:text="'Hola, ' + ${name} + '!' " />
</body>
</html>
```

MVC – Clientes y Servidores



MVC del lado cliente

