

Propuesta de
actualización, integración y refactorización
de servicios web 2.0 para compartir medios
usando **Cake PHP**



Ing. Natalí Flores Lafosse



Introducción

Necesidad:

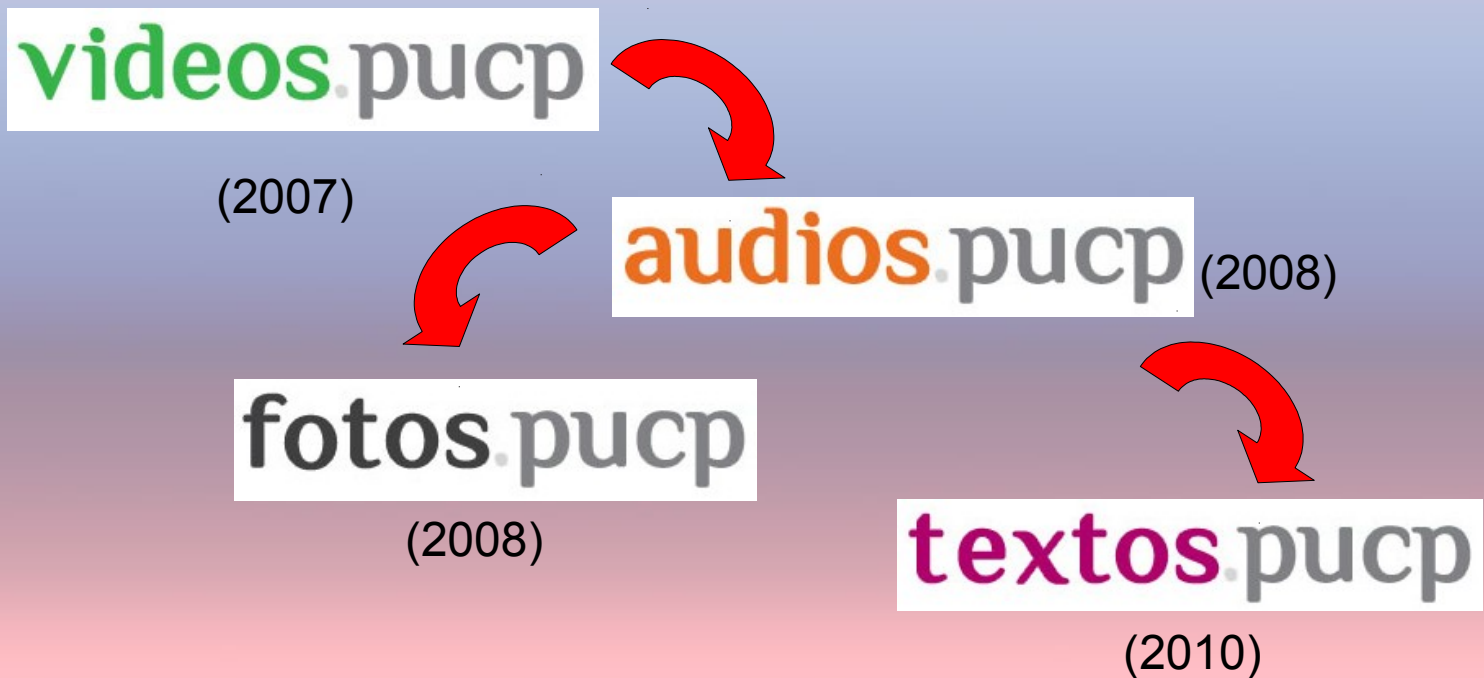
- Compartir medios
- Desarrollo rápido
- Direcciones amigables





Introducción

Cronología de Desarrollo





Introducción

Cake PHP

- Framework para PHP 5
- Utilización del M-V-C
- Plantillas
- Validación
- Asistentes (helpers) para HTML, Ajax y otros

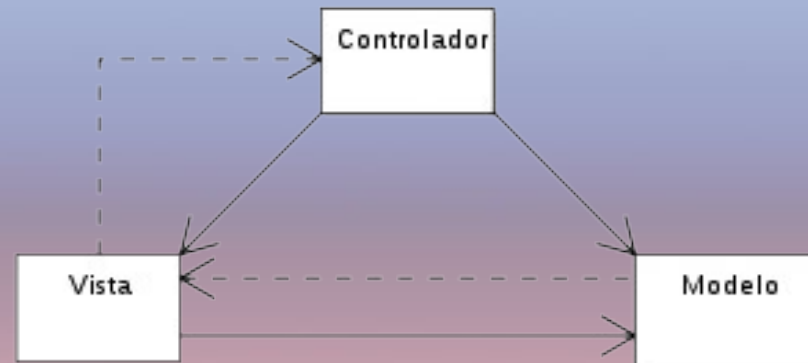




Introducción

Nociones sobre M-V-C

- Modelo: Datos
- Controlador: Lógica
- Vista: Interfaz

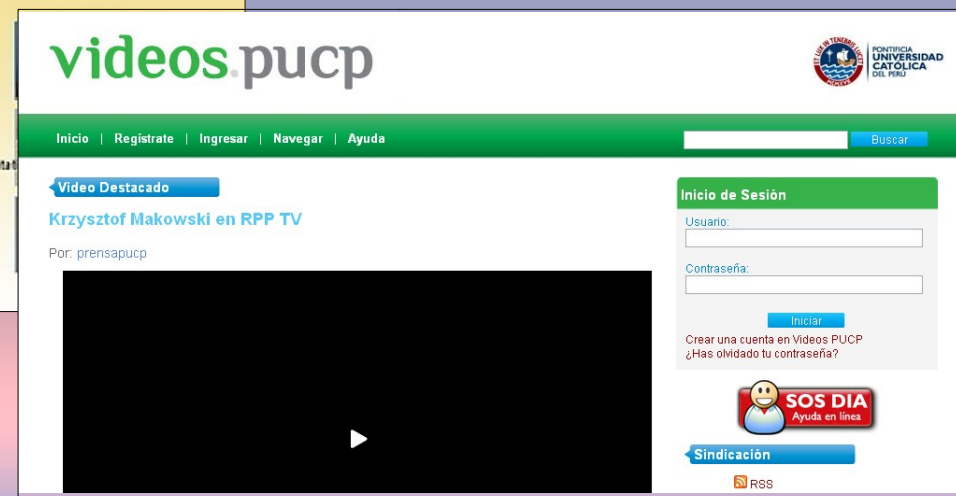
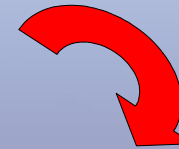




Identificación de Mejoras

Reutilización de Código (1)

- Cambios de diseños

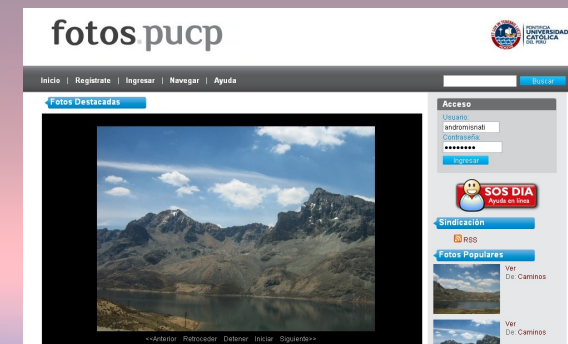
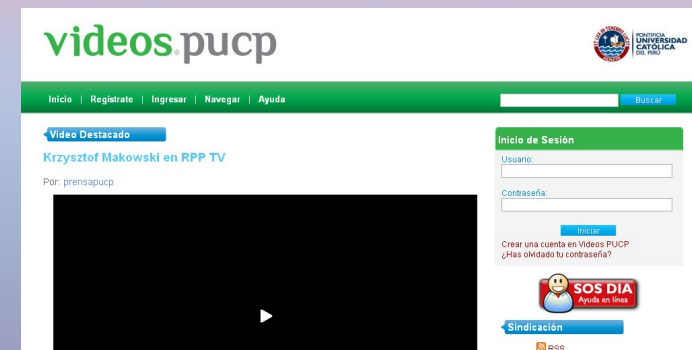




Identificación de Mejoras

Reutilización de Código (2)

- Cambios de diseños (2)

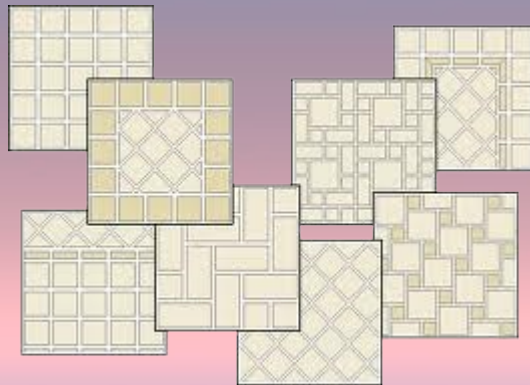




Identificación de Mejoras

Reutilización de Código (3)

- Características visuales similares
(paginación – perfiles – listas de elementos)

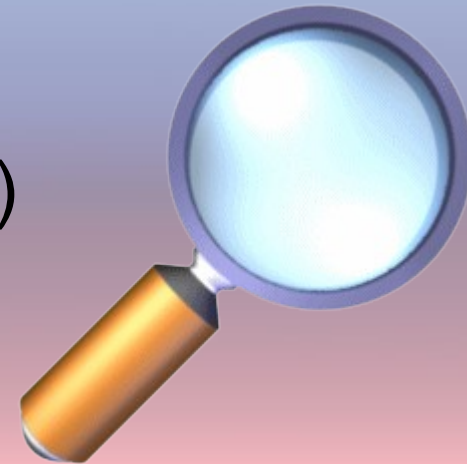




Identificación de Mejoras

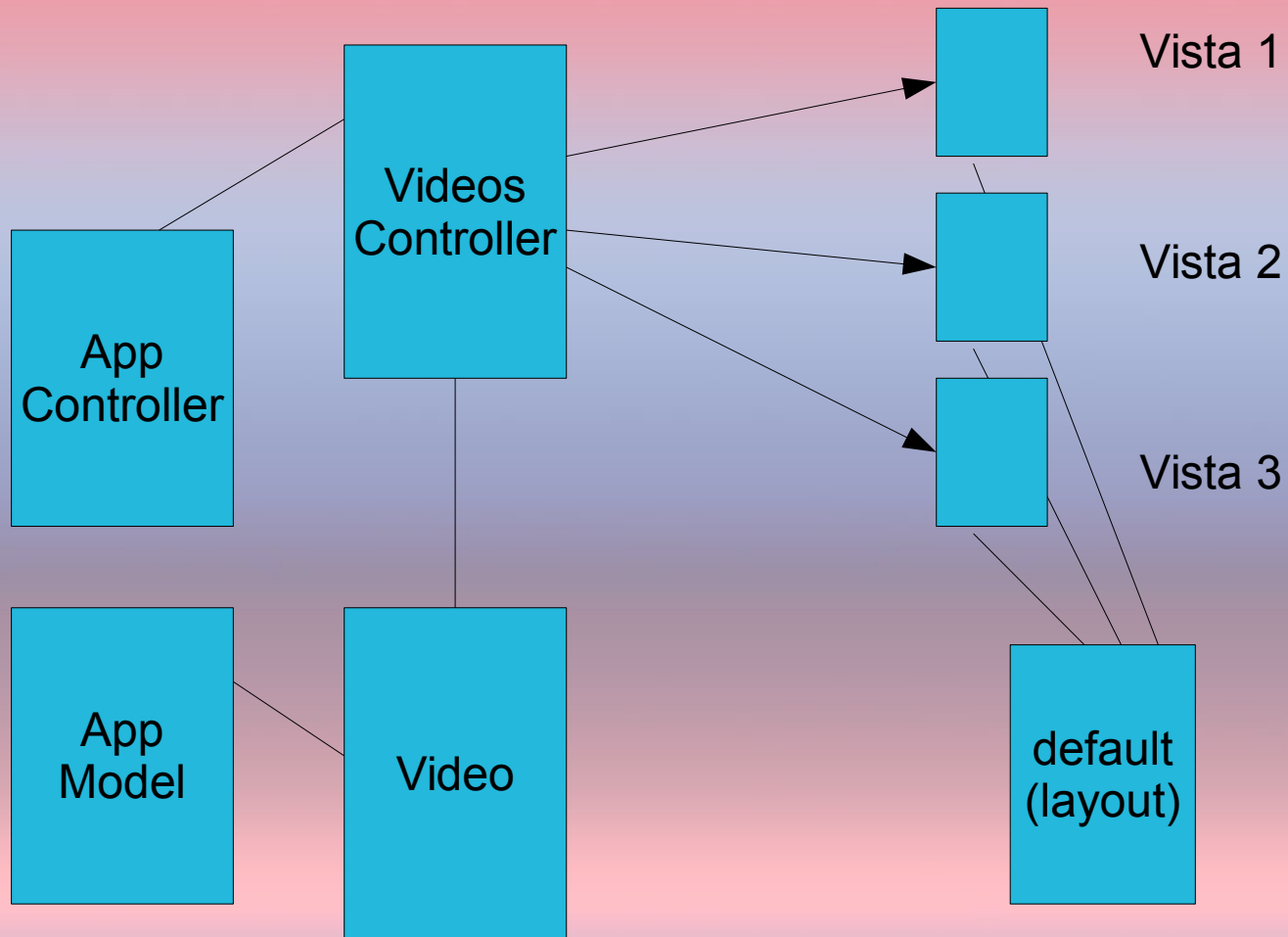
Reutilización de Código (3)

- Nuevas funcionalidades
- Optimización de código
(búsquedas – listas de medios)



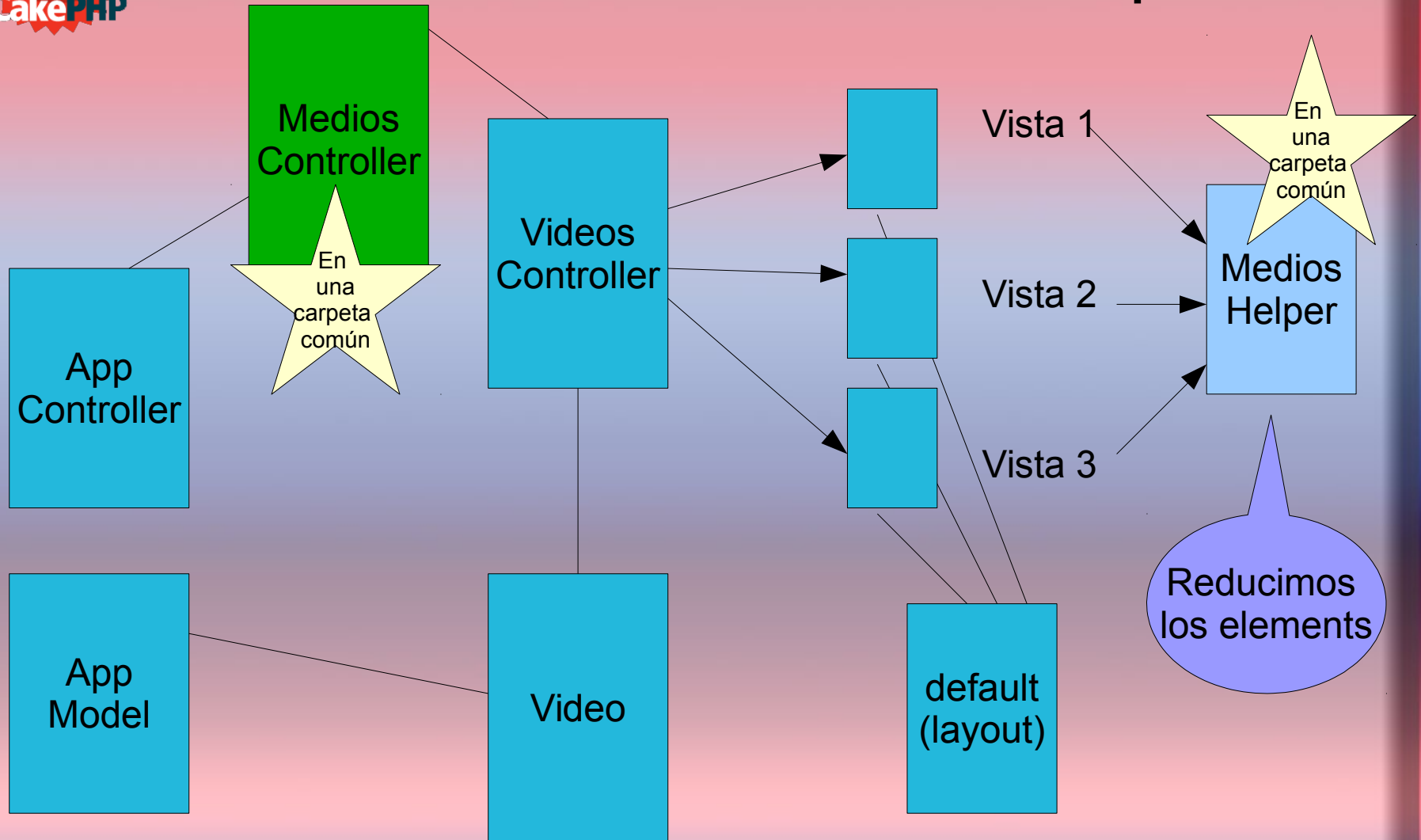


Modelo Actual





Modelo Propuesto





Avances

Paso 1: Actualización y Estandarización

- Aprovechar las mejoras
- Utilizar estándares de programación
- Documentar





Avances

Paso 2: Utilización de librería genérica

- Definir las partes comunes
- Pruebas de cada servicio
- Evaluación de performance





Ventajas

- Actualización de Cores “en lote”
- Implementación directa de nuevas funcionalidades
- Limpieza de Código





Próximos Pasos a Analizar

- Integración de cuentas de los servicios
- Crear una versión comunitaria





Referencias

- Sitio oficial CakePHP <http://cakephp.org>
- Frameworks en PHP
<http://oar.pucp.edu.pe/objetos/ver/1>
- Maintaining an Application-independant Code Library
<http://bakery.cakephp.org/articles/Sake/2008/05/03/maintaining-an-application-independant-code-library>



Preguntas

