



# *Diseño y desarrollo de aplicaciones web multidispositivo*

*César Colado*  
*Ingeniero de Telecomunicación*

© Germinus, 2003

Índice

## Índice

- ➔ Introducción
- Independencia de dispositivo
- Arquitectura de un portal
- Decisiones de diseño
- Conclusiones



© Germinus, 2003

### Diseño y desarrollo de aplicaciones web multidispositivo

*"El universo de información accesible a través de la red"*

W3C. World Wide Web Consortium

- Telaraña mundial
- Información en cualquier formato: texto, gráfico, audio, ...
- Accesible desde navegadores
- Sistema basado en hipertexto

### Paradigma cliente/servidor



## ¿Qué es un portal?

*Conjunto de aplicaciones web que proporcionan un único punto de entrada a servicios e información dentro de una empresa.*

*Puerta de enlace para las necesidades informativas de la empresa. A menudo se define como el panel de control.*



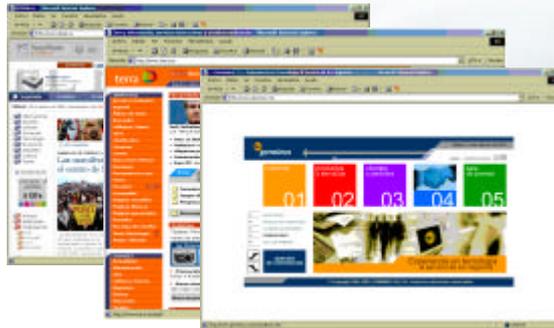
## Tipos de portales

- Portales de información de empresa (EIP). B2E
  - Gestión del conocimiento
  - Gestión de contenidos
  - Gestión de documentos
- Portales para e-Business
  - B2C. Conocer la oferta de servicios y productos, realizar pedidos, seguimiento de los pedidos: [elcorteingles.com](http://elcorteingles.com)
  - B2B. Relaciones entre empresas: [www.galiciab2b.com](http://www.galiciab2b.com)
- Portales públicos
  - Mega portales: Yahoo, AOL, Terra
  - Verticales. Centrados en nichos: [supermotor.com](http://supermotor.com), [barrapunto.com](http://barrapunto.com), etc.

## Introducción. ¿Qué es un portal?

### Características comunes

- Agrupación de aplicaciones web
- Medio para acceder a la información
- Medio para acceder a servicios



© Germinus, 2003

## Introducción. Expectativas

### Expectativas

- Aparecen multitud de nuevos dispositivos capaces de acceder a la web
- Los usuarios esperan poder acceder a la misma información y a los mismos servicios

*Compra de entradas para el cine*

*Conocer el pronóstico del tiempo*

*Averiguar qué farmacia está más cerca de donde me encuentro*



© Germinus, 2003

### Problemas

- El portal debe facilitar el acceso con independencia del dispositivo utilizado
- Los dispositivos presentan muy diferentes características
- Hacer portales diferentes para cada dispositivo se convierte en algo no abordable
- La experiencia en la navegación del usuario ha de ser no frustrante
- La identidad del portal debe mantenerse en los distintos dispositivos

### Índice

- Introducción
- ➔ Independencia de dispositivo
- Arquitectura de un portal
- Decisiones de diseño
- Conclusiones

## Las 4 Aes

ANYONE. Cualquiera  
ANYWHERE. Desde cualquier lugar  
ANYTIME. En cualquier momento  
ANYHOW. De cualquier forma



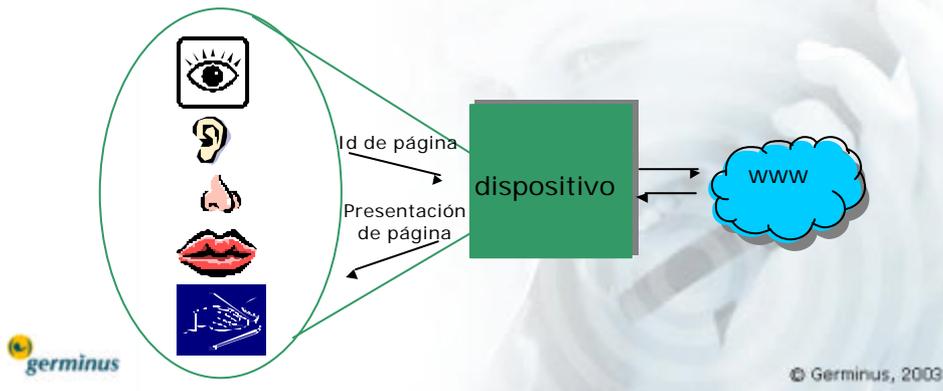
## Device Independence Working Group. W3C

- Creado en febrero de 2001
- Sus objetivos son:
  - Posibilitar el acceso a la web con independencia del dispositivo utilizado
  - Evitar repetir la autoría por dispositivo
- Definición de soluciones. Tres perspectivas:
  - El usuario
  - Las técnicas que permiten la creación y el acceso a la aplicación web
  - El mecanismo de entrega

## Independencia de dispositivo

### Perspectiva del usuario

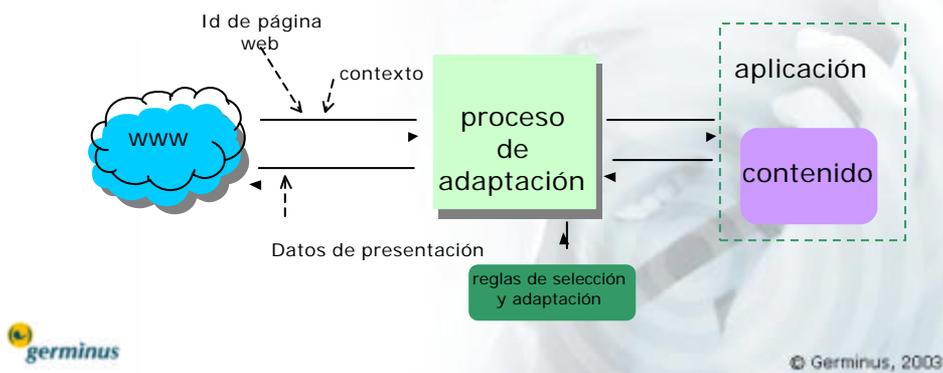
Debe ser posible obtener *una presentación funcional* en respuesta a la solicitud de una página con independencia del *mecanismo de acceso* utilizado



## Independencia de dispositivo

### Perspectiva de los autores

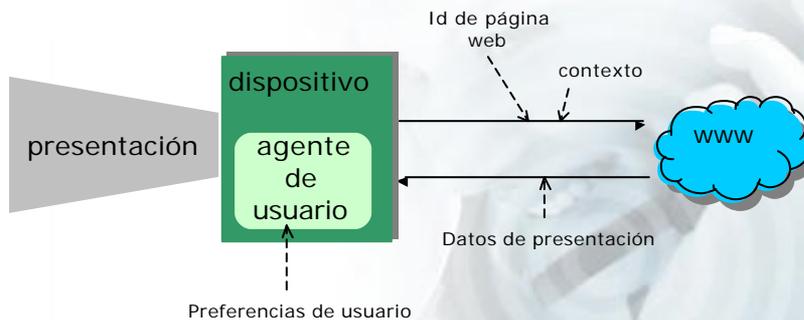
Debe ser posible obtener *una presentación funcional* mediante adaptaciones en función *del contexto*, siempre que el mecanismo de acceso sea adecuado, así como personalizar la respuesta.



## Independencia de dispositivo

### Perspectiva del mecanismo de entrega

El *agente de usuario* debe asociar información sobre el contexto a la solicitud de una página web

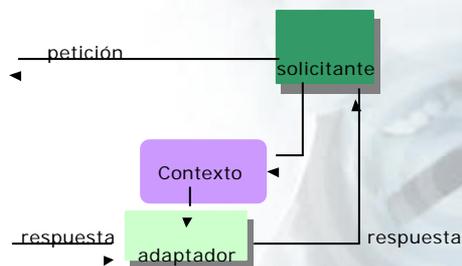


## Independencia de dispositivo

### Contexto

Características que definen las posibilidades del mecanismo de entrega así como las preferencias del usuario

- HTTP Headers
- Composite Capability/Preference Profiles (CC/PP). W3C
  - Basado en RDF
  - Extensible mediante la definición de vocabularios
  - UAPProf (WAP Forum)



## Independencia de dispositivo

Contexto. Ejemplo cabeceras HTTP.  
Móvil

```
GET /elpais.es HTTP/1.1
Host: 192.168.1.47
user-agent: Nokia Mobile Browser 3.01
Accept: application/vnd.wap.wmlscriptc,
        application/vnd.wap.wml+xml, application/vnd.wap.wbxml,
        application/vnd.wap.wmlc, application/xhtml+xml,
        image/vnd.wap.wbmp, image/gif, text/html, text/css, application/*,
        text/vnd.wap.wmlscript, text/vnd.wap.wml, text/plain, multipart/*
accept-charset: ISO-8859-1, US-ASCII, UTF-8; Q=0.8, ISO-10646-
        UCS-2; Q=0.6
accept-language: en, fi
Via: Nokia Activ Server 2.0 Professional (build 2451A)
X-Network-Info: UDP,127.0.0.1,security=0
Accept-Encoding:
```



© Germinus, 2003

## Independencia de dispositivo

Contexto. Ejemplo cabeceras HTTP.  
Explorer

```
GET /elpais.es HTTP/1.1
Accept: */*
Accept-Language: es
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 5.01; Windows NT 5.0)
Host: 192.168.1.47
Connection: Keep-Alive
```



© Germinus, 2003

### Contexto. Ejemplo de CC/PP

```
<RDF xmlns="http://..." xmlns:rdf="http://..." xmlns:ccpp="http://..."
  xmlns:uaprof="http://...">
  <Description about="http://www.example.com/MyProfile">
    <ccpp:component>
      <Description about="http://www.example.com/TerminalHardware">
        <type resource="http://...Schema#HardwarePlatform" />
        <uaprof:CPU>PPC</uaprof:CPU>
        <uaprof:ScreenSize>320x200</uaprof:ScreenSize>
      </Description>
    </ccpp:component>
    <ccpp:component>
      <rdf:Description rdf:about="http://...le#TerminalSoftware">
        <rdf:type rdf:resource="http://...#SoftwarePlatform" />
        <ex:name>EPOC</ex:name>
        <ex:version>2.0</ex:version>
      </rdf:Description>
    </ccpp:component>
  </Description>
</RDF>
```

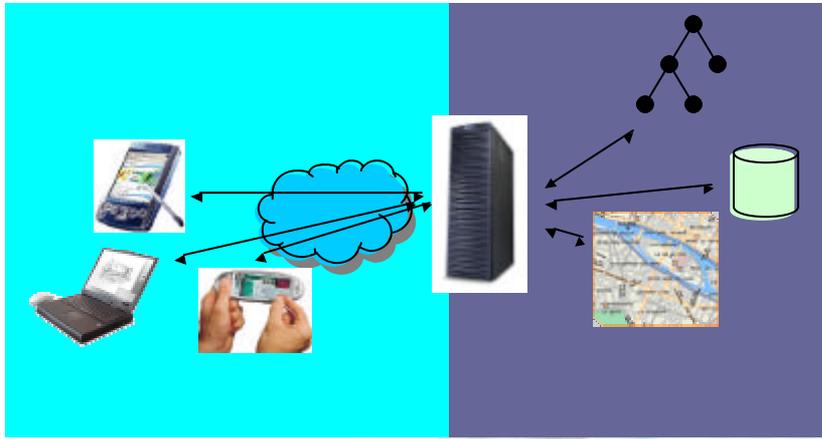


### Índice

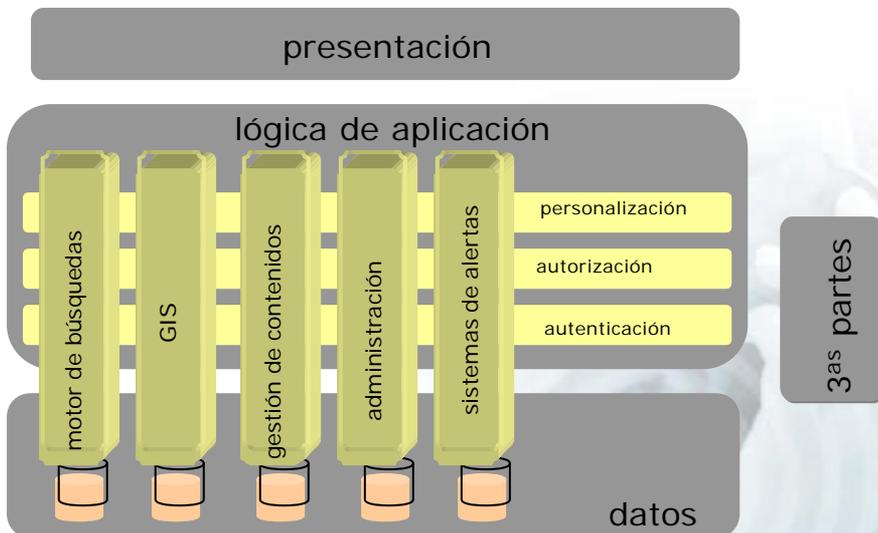
- Introducción
- Independencia de dispositivo
- ➔ Arquitectura de un portal
- Decisiones de diseño
- Conclusiones



# Arquitectura de un portal



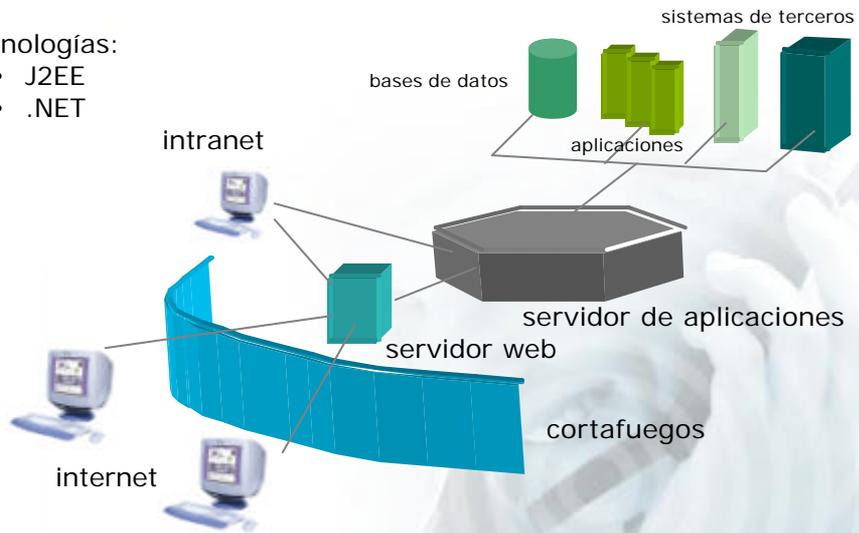
# Arquitectura de un portal



## Arquitectura de un portal

### Tecnologías:

- J2EE
- .NET



## Arquitectura de un portal

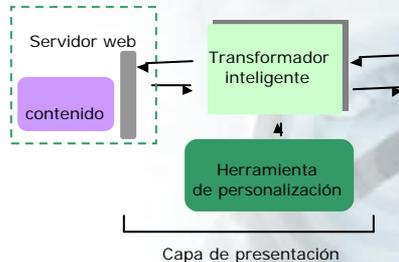
### Capa de presentación

- Composición de la vista
  - La lógica tiene como resultado la información a presentar, pero no la forma en que se presenta: una o dos tablas, con el logo de la empresa o sin él, etc.
- En esta capa es relevante el **contexto de entrega**
  - Características del dispositivo
  - Características de la red de acceso
  - Preferencias del usuario
- Herramientas como:
  - Everix
  - Volantis
  - WebCruiser

### WebCruiser

Reglas de adaptación separadas del resto de información de presentación

- La lógica tiene como resultado la información a presentar.
- Presentación de dos etapas:
  - Presentación general. Más burda, para cualquier dispositivo. No existe dependencia con el mecanismo de acceso.
  - Adaptación de la presentación. Se adapta el resultado de la etapa anterior, para obtener una presentación funcional.



### WebCruiser. Características

- Funciones de caché
- Paso transparente (pasarela)
- Transformación automática, en tiempo de petición.
- Transformación de imágenes automática: JPEG, GIF, TIFF, BMP, etc.
- Formularios. Interacción con el usuario
- Transformación personalizada



### WebCruiser. Personalización

- Referencia de elementos mediante expresiones XPath

```
/descendant::HTML[position()=1]/child::BODY[position()=2]/child::TABLE[position()=1]/child::TR[position()=1]/child::TD[position()=1]/child::TABLE[position()=1]/child::TR[position()=1]/child::TD[position()=1]/child::IMG[position()=1]
```

- Reglas para:
  - Inclusión (substitución)
  - Exclusión
  - Orden
- No es necesario conocer Xpath
- Transformaciones basadas en CSS



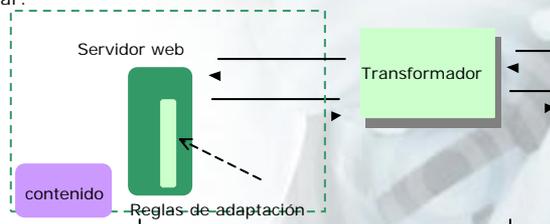
Elemento referenciado

### Everix



Reglas de adaptación almacenadas junto con el resto de información de presentación

- La lógica tiene como resultado la información a presentar.
- Presentación de dos etapas:
  - Presentación completa. Presentación para todos los mecanismos de acceso.
  - Filtrado y adaptación de la presentación generada, al mecanismo particular.

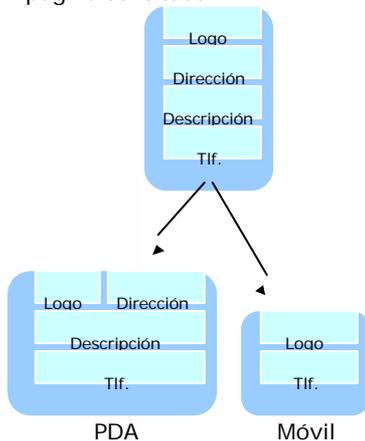


### Everix. Características

- Funciones de caché
- Información para la adaptación en etiquetas que se incluyen en el contenido
  - Inclusión
  - Exclusión
  - Orden
- Filtrado y adaptación en tiempo de petición.
- No transforma imágenes automáticamente, pero permite su selección en función del dispositivo

### Everix

página solicitada



```
<html>
<head><title>Prueba de concepto</title></head>
<body>
  <mm-group id="direccion" grouptitle="direccion" >
    <mm-head id="hd_direccion">Germinus Logo</mm-head>
    <mm-body id="bd_direccion1" idref="hd_direccion">
      <p>CManoterías 44, Madrid. España</p>
    </mm-body>
    <p>Segundo párrafo</p>
    <mm-body id="bd_direccion2" idref="bd_direccion1">
      <p>Párrafo 3</p>
    </mm-body>
  </mm-group>
  <table id="tabla_de_prueba">...</table>
  <mm-exclude useragents="xxx">
    <p>Esta empresa ..., bla, bla..... </p>
  </mm-exclude >
</body>
</html>
```

### Capa de lógica

- Capa integradora
  - Núcleo del portal. En esta capa se integran los distintos servicios y contenidos.
  - Funcionalidad, inteligencia.
  - Reusabilidad de componentes
- Es la capa que determina:
  - La respuesta a la solicitud de un usuario
  - Gestiona la interacción con los usuarios
- Servicios

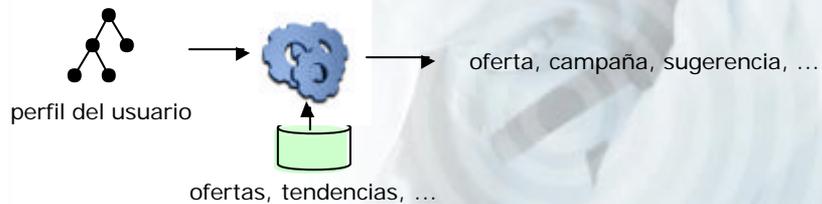
### Servicios de un portal 1/2

Servicios y funcionalidad requerida por el resto de servicios integrados en la capa de lógica

- Autenticación
  - Identificar al usuario
  - Autenticación basada en nombre de usuario y clave
  - Autenticación basada en firma digital
  - La autenticación puede ser dependiente del mecanismo de acceso
- Autorización.
  - Servicios o información a los que puede acceder el usuario

### Servicios de un portal 2/2

- Personalización
  - Perfil del usuario
    - Preferencias
    - Intereses
  - Comportamiento
  - Intereses y preferencias aprendidas de comportamientos pasados
  - El mecanismo de acceso es relevante para ser proactivo
  - Campañas de marketing y fidelización
  - Experiencia gratificante



### Otros servicios de un portal

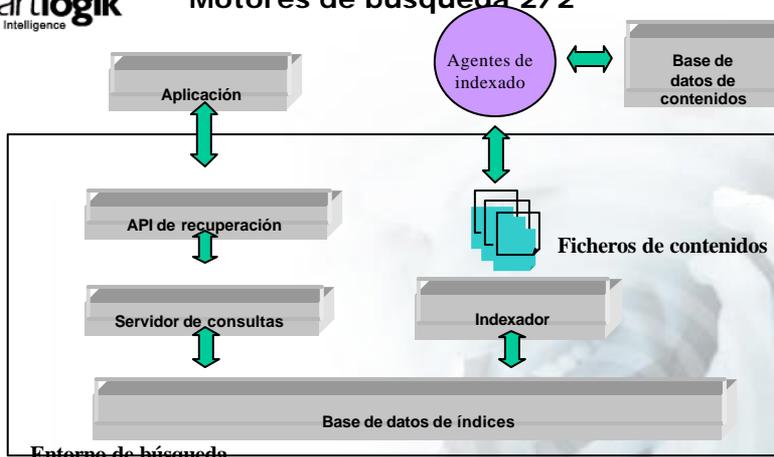
- Motor de búsquedas
- Sistema de información geográfica
- Gestión de contenidos
- Administración
- Sistema de alertas

## Motores de búsqueda 1/2

- Sistema capaz de buscar información entre los contenidos del portal
  - Motores de indexación
  - Distintas fuentes de información
- Problemática del multidispositivo
  - Recursos que indexa
  - Presentación del resultado de búsqueda
- Ejemplos:
  - Smartlogik, Verity, Autonomy, Lucene, ...



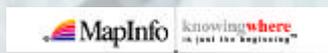
## Motores de búsqueda 2/2



## Arquitectura de un portal

### Sistemas de información geográfica 1/2

- Favorecido por el acceso multi-dispositivo. (Movilidad)
- Nuevos servicios en un portal:
  - Servicio de geolocalización. País, ciudad, calle y número
  - Puntos de interés
  - Rutas
- Basado en parámetros generalmente proporcionados en el *contexto de entrega* por la red de acceso:
  - Latitud y longitud
- La presentación es dependiente del mecanismo de acceso:
  - Formato de imágenes
  - Descripciones,
  - Etc.
- Ejemplos: Webraska , MapInfo

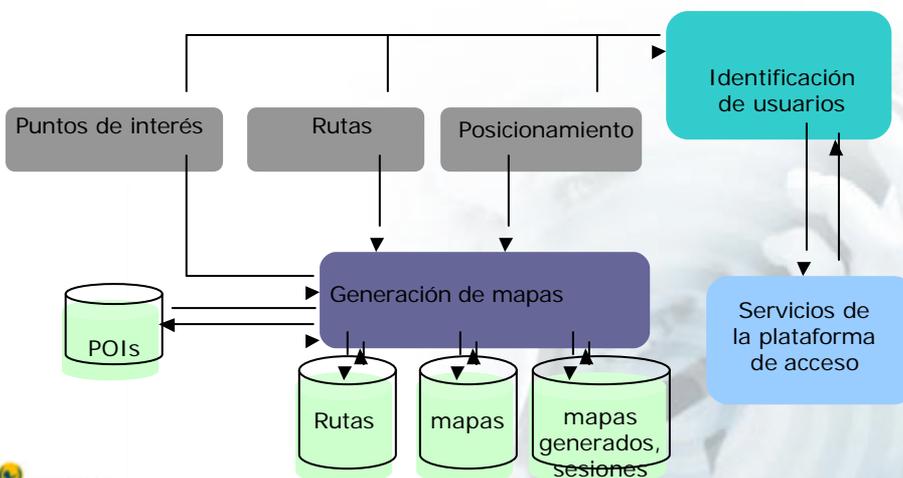


uebraska

© Germinus, 2003

## Arquitectura de un portal

### Sistemas de información geográfica 2/2



© Germinus, 2003

### Administración

- Acceso a la administración a través del propio portal
- Administración de los procesos de back-office
- Gestión de los distintos tipos de contenidos y sus conversiones
- Gestión de proveedores y servicios externos
- Gestión de usuarios
- Gestión de CHAT
- Gestión del servicio de búsquedas
- Gestión de mapas
- Definición de flujos de trabajo
- Adición y mantenimiento de funcionalidades horizontales y verticales
- ...

### Gestión de contenidos 1/2

- Distintos proveedores
- Proceso de normalización
- Ciclo de vida
- Definición de flujos
- Ejemplos: vignette, documentum, interwoven, slash, ...
- Problemática de ser multi-dispositivo:
  - Conversiones de contenido multimedia
    - Conversiones automáticas bajo petición
    - Conversión como parte del aprovisionamiento



### Gestión de contenidos 2/2



### Sistemas de alertas

- Sistema de información programado por el usuario o de forma automática según intereses o preferencias
  - Por aplicación
  - Alertas de gestión
- Portal multi-dispositivo
  - Posibilidad de utilización de distintos medios:
    - E-mail
    - Multimedia Messaging Services (MMS)
    - WAP Push
    - Short Message Service (SMS)
- Servicios asíncronos generalmente basados en sistemas de colas como Java Message Service (JMS)



### Capa de datos

- Almacen persistente de toda la información necesaria en el portal
  - Proveedores
  - Usuarios
  - Noticias, ...
  - Índices de los motores de búsqueda
- Distintos almacenes para distintas necesidades:
  - Bases de datos relacionales
    - Información sensible
    - Información no sensible
  - Sistema LDAP para almacenar los perfiles de usuarios y proveedores
  - Sistema de ficheros para algunos recursos del portal



open Idap



webraska

### Terceras partes

- Proveedores de contenidos externos
- Proveedores de servicios externos
- Es necesario gestionar la utilización de estos recursos

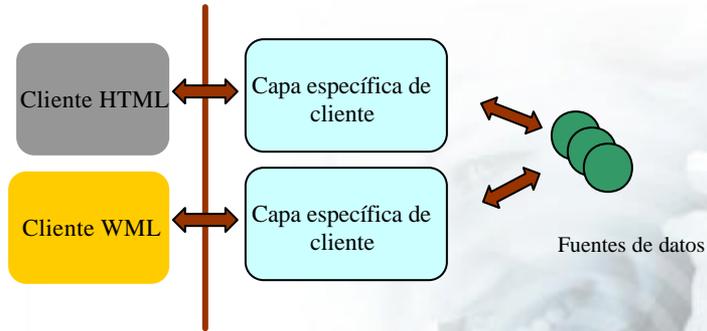
## Índice

- Introducción
- Independencia de dispositivo
- Arquitectura de un portal
- ➔ Decisiones de diseño
- Conclusiones

## Decisiones de diseño

- Primera aproximación al problema
  - Una web por dispositivo
  - Una web y un proceso de adaptación
- Diferentes tendencias en los diseños
  - Basado en los datos
  - Centrado en el gestor de contenidos
  - Arquitectura Modelo-Vista-Controlador (MVC)
  - Diseño basado en portlets

### Una web por dispositivo



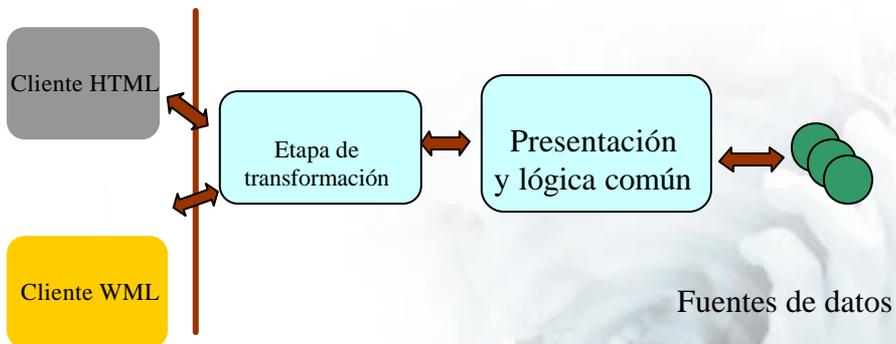
**Ventajas:**

- Eficiente
- Personalizable

**Problemas**

- Escalabilidad
- Mantenibilidad

### Una web multidispositivo



**Ventajas:**

- Fácilmente mantenible
- Escalable

**Problemas**

- Eficiencia
- Menor granularidad en el proceso de adaptación

### Tendencias en los diseños 1/2



### Tendencias en los diseños 2/2

- No son necesariamente excluyentes
- Es posible aplicar un tipo de diseño a un servicio y otro diferente a otro de los servicios
- Generalmente los diseños más complejos incluyen a los más simples

### Diseño basado en productos personalizados

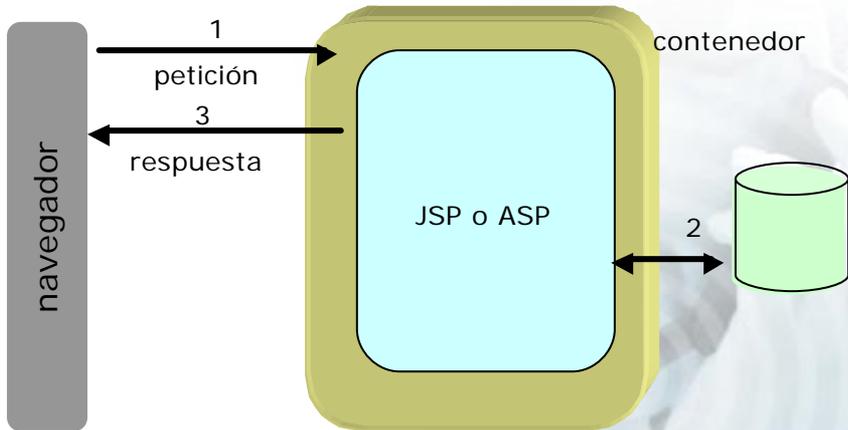
- Portales adaptados a la funcionalidad proporcionada por un paquete o producto concreto. Generalmente un gestor de contenidos.
- Existen soluciones para B2E, B2B, etc.
- Ventajas
  - Reducción en el desarrollo
  - Prácticamente es una solución cerrada
- Problemas
  - Difícil crecimiento a pesar de ser soluciones modulares
  - Coste de adaptación
- Ejemplos:
  - Vignette
  - Oracle Portal Server
  - BEA WebLogic Portal
  - PostNuke
  - PHPNuke
  - OpenCMS



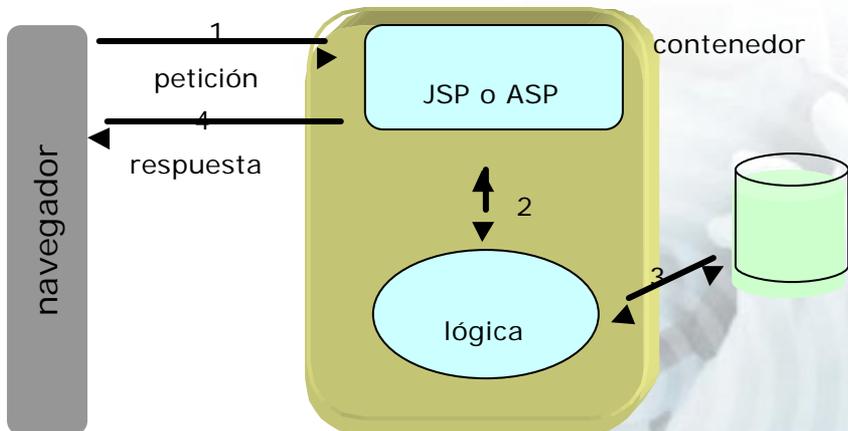
### Diseño basado en los datos 1/4

- Basados en páginas dinámicas
  - Java Server Pages (JSP)
  - Active Server Pages (ASP)
- Permiten el acceso a contenidos dinámicos
- Acceso a bases de datos
- Se unen la lógica de presentación y lógica de negocio

Diseño basado en los datos 2/4



Diseño basado en los datos 3/4



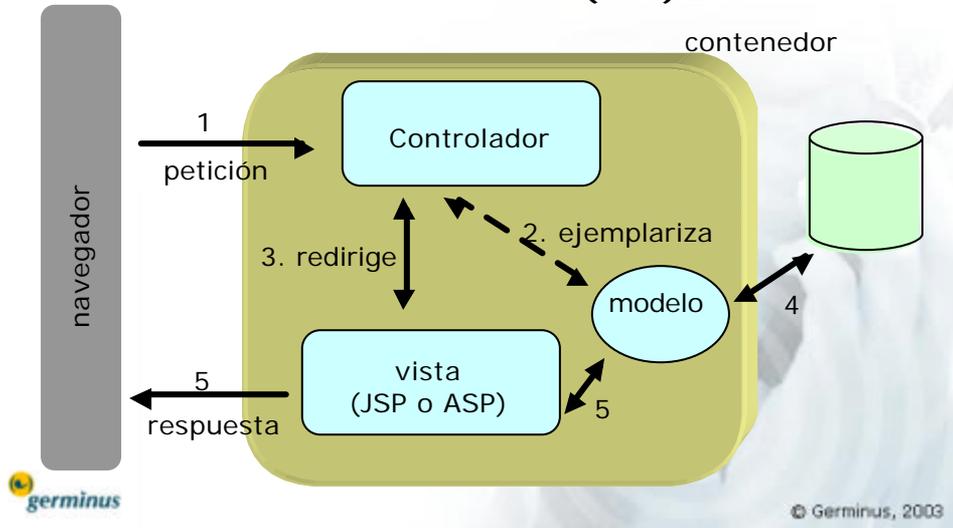
## Diseño basado en los datos 4/4

- Ventajas
  - Simplicidad
- Inconvenientes
  - Difícil escalabilidad
  - Difícil reutilización de componentes
  - Propenso al "Espaghetti code"

## Modelo Vista Controlador (MVC) 1/3

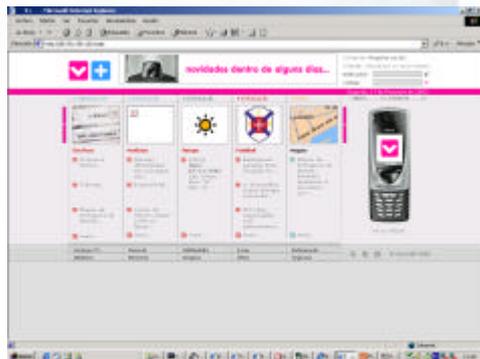
- Separación entre la lógica de negocio y la vista
- Servicios horizontales en el controlador
- Clara separación de responsabilidades

### Modelo Vista Controlador (MVC) 2/3



### Modelo Vista Controlador (MVC) 3/3

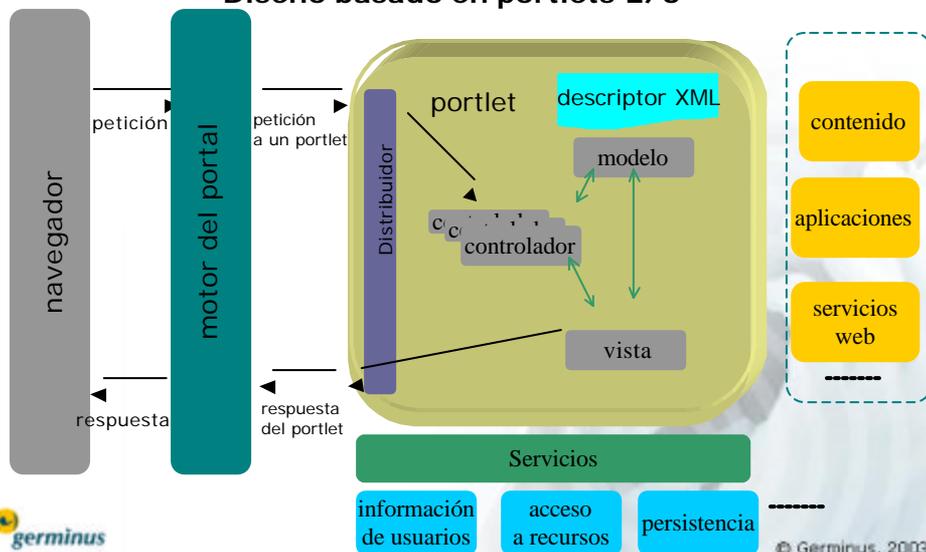
- Ventajas:
  - Es posible adaptar la vista fácilmente en función del dispositivo de acceso
  - Reusabilidad
  - Crecen ordenadamente
- Problemas
  - Complejidad



### Diseño basado en portlets 1/6

- Componente web controlado por un contenedor
- Fácilmente desarrollados y desplegados en el portal
- Arquitectura muy modular y extensible
- Integración absoluta en el portal:
  - El portal controla su situación y apariencia
  - El portlet tiene acceso a los servicios horizontales

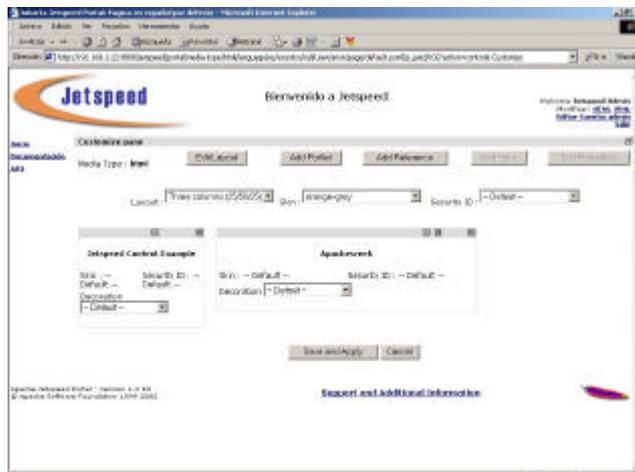
### Diseño basado en portlets 2/6



### Diseño basado en portlets 3/6



### Diseño basado en portlets 4/6





## Índice

- Introducción
- Independencia de dispositivo
- Arquitectura de un portal
- Decisiones de diseño
- ➔ Conclusiones

## Conclusiones

- Un portal es un sistema vivo que crece y debe ser capaz de adaptarse
  - Arquitectura modular y flexible
  - Escalabilidad horizontal y vertical
  - Alta disponibilidad y alto rendimiento
  - Seguridad
- Un portal multidispositivo es complejo de abordar
  - Arquitectura
  - Gestión de recursos
- Peor rendimiento del conjunto
- No existe la solución perfecta
- Nuevos servicios, nuevos usuarios, más negocio



**Si desea más información**

**[ccolado@germinus.com](mailto:ccolado@germinus.com)**  
**[germinus@germinus.com](mailto:germinus@germinus.com)**

**Madrid**

Avda. Manoteras, 44. 2ª Planta. 28050 Madrid  
Tel: 91 296 12 34

**Málaga**

Avda. Juan López Peñalver, 21  
Edificio Bic Euronova  
Parque Tecnológico de Andalucía  
29590 Campanilla – Málaga  
Tel: 95 101 05 83

[www.germinus.com](http://www.germinus.com)



© Germinus, 2003