

Tratamiento de información contextual en entornos inteligentes

Pablo A. Haya

Departamento de Ingeniería Informática
Av. Fco. Tomás y Valiente, 11, 28049 Madrid. Spain
Universidad Autónoma de Madrid
Pablo.Haya@uam.es

23 de Marzo de 2006, EPS-UAM



Índice

- Motivación
- Modelo de información contextual
- Arquitectura pizarra
- Demostradores
- Conclusiones y trabajos futuros

<http://odisea.ii.uam.es>

¿Qué es la Computación Ubicua?

Manifiesto

“La tecnologías más profundas son aquellas que desaparecen. Se imbrican en el tejido de la vida diaria hasta que son indistinguibles de ella.” [WEIS91]



Características

- Dispositivos de la vida cotidiana con capacidades de procesamiento.
- Procesadores baratos y de bajo costo.
- La interacción se expande a todo el espacio.
- Aplicaciones ubicuas.



<http://odisea.ii.uam.es>

Objetivos

Ubicuidad

Transparencia

<http://odisea.ii.uam.es>

UA Información contextual

IDENTIDAD:
- ¿Quién soy?
- ¿Con quién estoy?

LOCALIZACIÓN:
- ¿Dónde estoy?

ACTIVIDAD:
- ¿Qué estoy haciendo?

<http://odisea.ii.uam.es>

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

UA Problemas

Problemas

- Integración de tecnologías diferentes y muy heterogéneas.
- Componentes se encuentran distribuidos.
- Configuración del entorno es dinámica.
- Fuentes contextuales de distinta naturaleza.
- Información contextual de diferente nivel de abstracción y resolución.
- Interfaces generadas dinámicamente.

<http://odisea.ii.uam.es>

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

UA Niveles de abstracción

Información contextual

Cada variable contextual provee diferentes niveles de abstracción.

Directas	Puerta Abierta / Cerrada (Sensores)	<div style="text-align: right;">-</div> <div style="border-left: 1px solid black; border-bottom: 1px solid black; height: 100px; margin: 0 auto;"></div> <div style="text-align: left;">+</div>
	Detección de movimiento (VideoCamara)	
Deducidas	Localización del usuario	
	Actividad del usuario	

<http://odisea.ii.uam.es>

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

UA Propuesta

Nuestra propuesta

Una representación estándar del contexto que combine fuentes de diferente naturaleza y nivel de abstracción

+

Un repositorio de información centralizado dónde se almacena el modelo.

+

Un mecanismo flexible para poder almacenar y distribuir esa información

<http://odisea.ii.uam.es>

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

Modelo de información contextual

23 de Marzo de 2006, EPS-UAM

UAM Representación

Representación del contexto

- Basado en redes semánticas
- **Cada componente del modelo se representa por una entidad**
- Una entidad se describe por:
 - Nombre: **identificador único**
 - Tipo: **clase a la que pertenece**
- Cada entidad puede tener una, cero o más:
 - **Propiedades: pares nombre-valor. Describen propiedades intrínsecas y universales de las entidades.**
 - **Relaciones con otra entidad. Se definen mediante un el nombre de la relación y la entidad con la que se relacionan**

Esquema: en UML Instanciación del modelo:

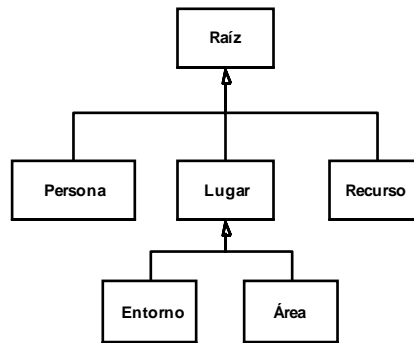
```

    graph TD
      EntidadA((EntidadA  
(Tipo))) -- Relación --- EntidadB((EntidadB  
(Tipo)))
      EntidadA -- Propiedad --> Valor[Valor de la  
propiedad]
    
```

<http://odisea.ii.uam.es>

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

Contexto primario



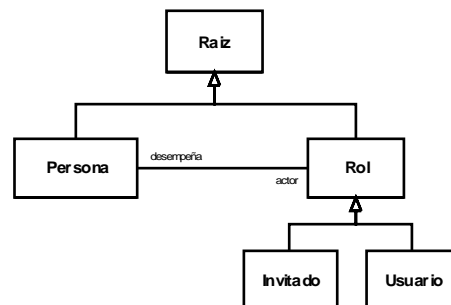
<http://odisea.ii.uam.es>

23 Mar, 2006

Tratamiento de información contextual en entornos inteligentes

Identidad

- Personas: Nombre completo, apellidos, correo-e
- Lugares: Nombre asociado al entorno
- Recursos: Tipo, Propiedades.



<http://odisea.ii.uam.es>

23 Mar, 2006

Tratamiento de información contextual en entornos inteligentes

UA Localización

Localización

- Un lugar nunca se puede solapar con otro lugar.
- Un entorno puede incluir a otros entornos y a otras áreas
- Un área puede incluir a otras áreas pero no a otros entornos.
- Todas las áreas que están incluidas en un mismo lugar son adyacentes.
- La adyacencia entre dos entornos siempre se realiza a través de una unión.

<http://odisea.ii.uam.es>

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

UA Localización

Presencia

- Propiedad espacial asociada a un lugar que aporta información sobre la existencia de cuerpos en su interior.
- Dos propiedades asociadas a *Lugar*.
 - *Vacío*
 - *Número de ocupantes*

Lugar

+vacío : boolean
+num_ocupantes : integer

<http://odisea.ii.uam.es>

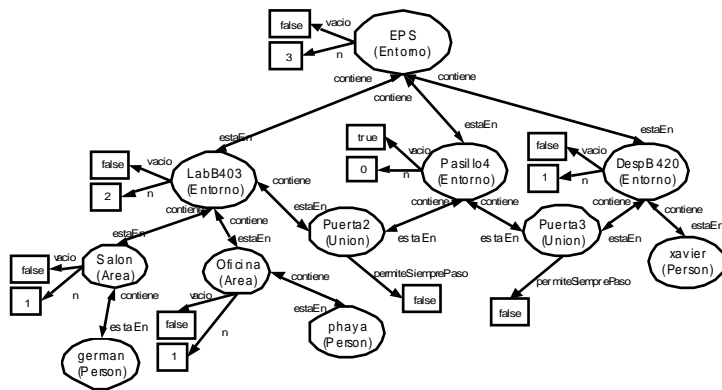
23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

Inclusión/Localización/Adyacencia

- **Inclusión:**
 - Relación *contiene*.
- **Localización:**
 - Relación *estaEn*.
- **Adyacencia:**
 - *Unión*
 - Relación *contiene/estaEn*.
 - Prop. *permiteSiemprePaso*.

<http://odisea.ii.uam.es>

Ejemplo localización



<http://odisea.ii.uam.es>

UAM Actividad

Actividad

- Ocupado
- No molestar

```

classDiagram
    class Persona
    class Actividad {
        +social: boolean
        +paraQue: ActividadFinalidad
        +conQue: ActividadFinalidad
    }
    class Recurso
    Persona "0..1" -- "0..n" Actividad : involucradoEn
    Persona "0..n" -- "0..1" Recurso : transporta
    Persona "0..n" -- "0..n" Recurso : usa
    Recurso "0..n" -- "0..n" Persona : usadoPor
    Recurso "0..n" -- "0..n" Actividad : usadoPor
  
```

<http://odisea.ii.uam.es>

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

UAM Actividad

Actividad

ActividadFinalidad

```

classDiagram
    class ActividadFinalidad
    class Obligacion
    class TiempoLibre
    class CuidadoPersona
    class Trabajo
    class TareaDomestica
    class Ocio
    class Descansar
    class Aseo
    class Medicarse
    class Alimentarse
    ActividadFinalidad <|-- Obligacion
    ActividadFinalidad <|-- TiempoLibre
    ActividadFinalidad <|-- CuidadoPersona
    Obligacion <|-- Trabajo
    Obligacion <|-- TareaDomestica
    TiempoLibre <|-- Ocio
    TiempoLibre <|-- Descansar
    CuidadoPersona <|-- Aseo
    CuidadoPersona <|-- Medicarse
    CuidadoPersona <|-- Alimentarse
  
```

<http://odisea.ii.uam.es>

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

UAM Actividad

Actividad

ActividadMedio

- Escribiendo
- Escuchando
- Leyendo
- Visionando
- Conversando
 - Conversando por teléfono
- Usando el ordenador

<http://odisea.ii.uam.es>

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

UAM Ejemplo

Ejemplo de instanciación del modelo

```

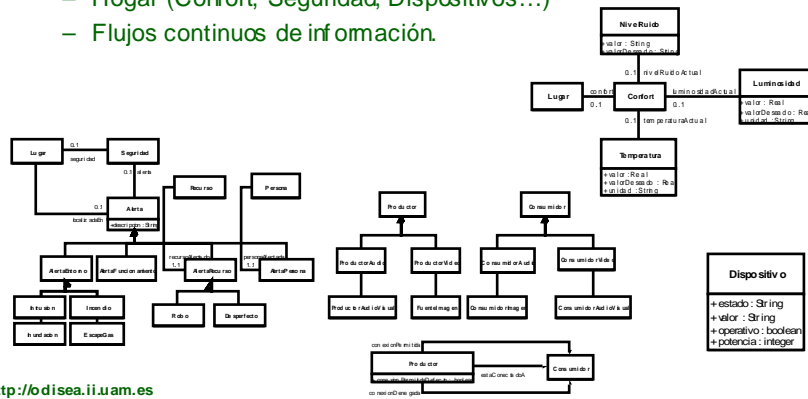
    graph TD
      Dave((Dave  
Persona)) -- involucradoEn --> A1((A1  
Actividad))
      Dave -- ocupa --> T1[true]
      Dave -- ocupa --> T2[true]
      Dave -- correo --> E[dave@uam.es]
      Dave -- noMolestar --> T3[true]
      Dave -- usa --> Altavoz1((Altavoz1  
Recurso))
      Dave -- estaEn --> LAB403((LAB403  
Entorno))
      Dave -- estaEn --> LAMP_1((LAMP_1  
Recurso))
      A1 -- conque --> Social[social]
      A1 -- conque --> False[false]
      A1 -- paraque --> TiempoLibre[TiempoLibre]
      Altavoz1 -- usadoPor --> A1
      LAB403 -- contiene --> Altavoz1
      LAB403 -- contiene --> LAMP_1
      LAMP_1 -- estado --> On[on]
  
```

<http://odisea.ii.uam.es>

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

Contexto secundario

- Extensión del modelo del contexto primario dependiente del dominio de aplicación.
 - Hogar (Confort, Seguridad, Dispositivos...)
 - Flujos continuos de información.



<http://odisea.ii.uam.es>

Espacio de nombres

- **Nombre** → `/name/lamp_1/props/status`
- **A partir de una relación** → `/name/lab403/contiene/*`
 - relación
- **A partir del tipo** → `/person/*/props/correoe`
 - tipo
- **Jerarquía predefinida** → `/roomresource/lab403/*`
 - Room type
 - Resource type

<http://odisea.ii.uam.es>


Arquitectura pizarra

23 de Marzo de 2006, EPS-UAM

UAM

Características

Capa contexto

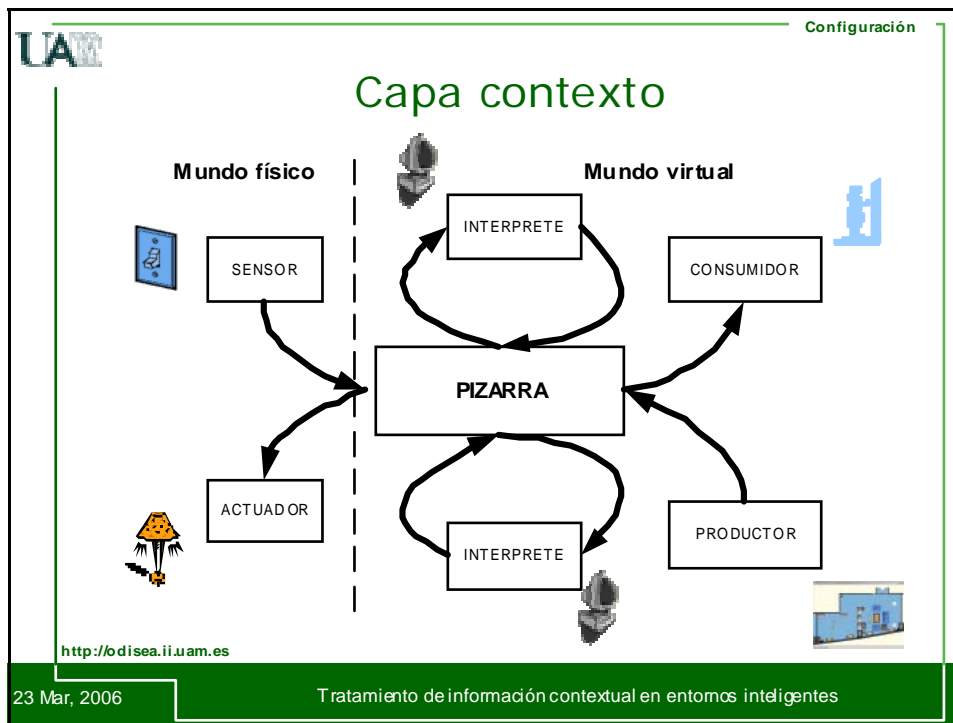


- Almacena una representación de las instancias del modelo de información contextual.
- Consiste en un repositorio común de información de cual se puede recuperar y almacenar la información contextual.
- Se basa en la metáfora de pizarra.
- Mecanismos: Sondeo y Publicación-Subscribirse
- Se pueden añadir nuevos componentes dinámicamente.

<http://odisea.ii.uam.es>

23 Mar, 2006

Tratamiento de información contextual en entornos inteligentes



- UA Desacoplo
- ## Desacoplo
- **Temporal:** Los dos participantes no tiene por qué estar activos simultáneamente.
 - **Espacial:** Los productores no conoce quiénes son los consumidores.
 - **Funcional:** Los productores no precisan conocer cuál es el uso que se le va dar a la información
- <http://odisea.ii.uam.es>
- 23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

UA Comunicación

Comunicación con la pizarra

- GET (Obtener)
- SET (Cambiar)
- ADD (Añadir)
- REMOVE (Borrar)
- ADD REL (Añadir Rel)
- REMOVE REL (Borrar Rel)
- SUBSCRIBE (Subscribirse)
- UNSUBSCRIBE (Describirse)

http://odisea.ii.uam.es

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

UA Operación GetContext

GetContext

Input: /get/roomperson/labB403*/props/fname

BLACKBOARD

```

graph TD
    LabB403((LabB403)) -- contains --> phaya((phaya))
    LabB403 -- contains --> xalaman((xalaman))
    phaya -- fname --> Pablo[Pablo]
    xalaman -- fname --> Xavier[Xavier]
        
```

Output:

```

<GetResponse>
  <entity name="labB403">
    <entity name="pha ya">
      <property name="fname"> Pablo</property>
    </entity>
    <entity name="xalaman">
      <property name="fname"> Xavier</property>
    </entity>
  </entity>
</GetResponse>
        
```

http://odisea.ii.uam.es

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

UAM Módulos

Módulos extras

- Controladores de dispositivo
- Histórico
- Control de acceso
- Arbitraje

<http://odisea.ii.uam.es>

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

UAM Control de acceso

Control de acceso

```

classDiagram
    class Agente
    class Grupo
    class Recurso

    Agente "0..n" -- "0..n" Recurso : esPropietario
    Recurso "0..n" -- "1..1" Agente : tienePropietario
    Agente "1..1" -- "0..n" Recurso : esAgenteRestringido
    Recurso "0..n" -- "1..1" Agente : esAgentePermitido
    Grupo "0..n" -- "0..n" Recurso : esPropietario
    Recurso "0..n" -- "1..1" Grupo : tienePropietario
    Grupo "1..1" -- "0..n" Recurso : esGrupoRestringido
    Recurso "0..n" -- "1..1" Grupo : esGrupoPermitido
    Agente "0..n" -- "0..n" Grupo : esAgentePermitido
    Grupo "0..n" -- "0..n" Agente : esGrupoPermitido
    Agente "0..n" -- "0..n" Grupo : esAgenteRestringido
    Grupo "0..n" -- "0..n" Agente : esGrupoRestringido
  
```

permitidoPorDefecto = true | false

<http://odisea.ii.uam.es>

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

Arbitraje

- Centralizado
- Mecanismo basado en colas de prioridades preferentes.
 - Diferentes roles (padre/hijo, propietario/invitado, profesor/alumno, administrador/usuario...).
 - Situaciones de emergencia.
- Por cada propiedad/relación existe un cola.
- Cada agente como máximo un comando en cada cola.
- Las colas se crean dinámicamente.
- Un agente puede eliminar los comandos de la cola.
- Límite de caducidad a las ordenes.
- La prioridad la calcula la pizarra en función del: rol agente (usuario/invitado), tipo de orden (preferencia/orden directa), estado del entorno.

<http://odisea.ii.uam.es>

Políticas específicas

- **Volumen altavoz.** La prioridad es inversamente proporcional al volumen que se quiera establecer en altavoz.
- **Primar apagar/cerrar.** Si la orden es de apagar (o cerrar) el recurso, está tiene la máxima prioridad. Sería también el caso del altavoz. Si un emisor decide apagarlo completamente tiene más prioridad que los que deciden mantenerlo encendido.
- **Primar encender/abrir.** Este sería el caso contrario al anterior. Se prima a aquellos comandos que enciendan (o abran) el recurso. Por ejemplo, sería la política a aplicar para el control de una puerta.

<http://odisea.ii.uam.es>

BBXML

23 de Marzo de 2006, EPS-UAM



classes

Ejemplo de definición de la clase de entidad *person*

```
<classes>  
  <class name="person" extends="root">  
    <property name="Nombre"/>  
    <property name="Apellidos"/>  
    <property name="NumTarjeta"/>  
    <property name="Correo"/>  
  </class>  
</classes>
```

<http://odisea.ii.uam.es>

Representación de una entidad

- Cada entidad se representa mediante un lenguaje-XML
- Cada elemento del entorno o del contexto del entorno se representa por una entidad
- Cada entida tiene un nombre, un tipo, unas propiedades, y unas relaciones.

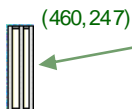
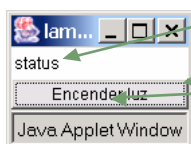
```

<entity name="speaker_1" type="Speaker">
  <property name="enable"/>
  <property name="ready"/>
  <property name="status"/>
  <property name="port"/>
  <property name="IP"/>
  <property name="bandwidth"/>
  <property name="left_Volume"/>
  <property name="right_Volume"/>
  <property name="master_Volume"/>
</entity>

<entity name="lab_B403" type="Room">
  <relation name="contains" destination="lamp_1" type="Light"/>
  <relation name="contains" destination="lamp_2" type="Light"/>
  <relation name="contains" destination="door_1" type="Lock"/>
  <relation name="contains" destination="speaker_1"
type="Speaker"/>
  <relation name="contains" destination="abraxas"/> type="PC"/>
</entity>
    
```

<http://odisea.ii.uam.es>

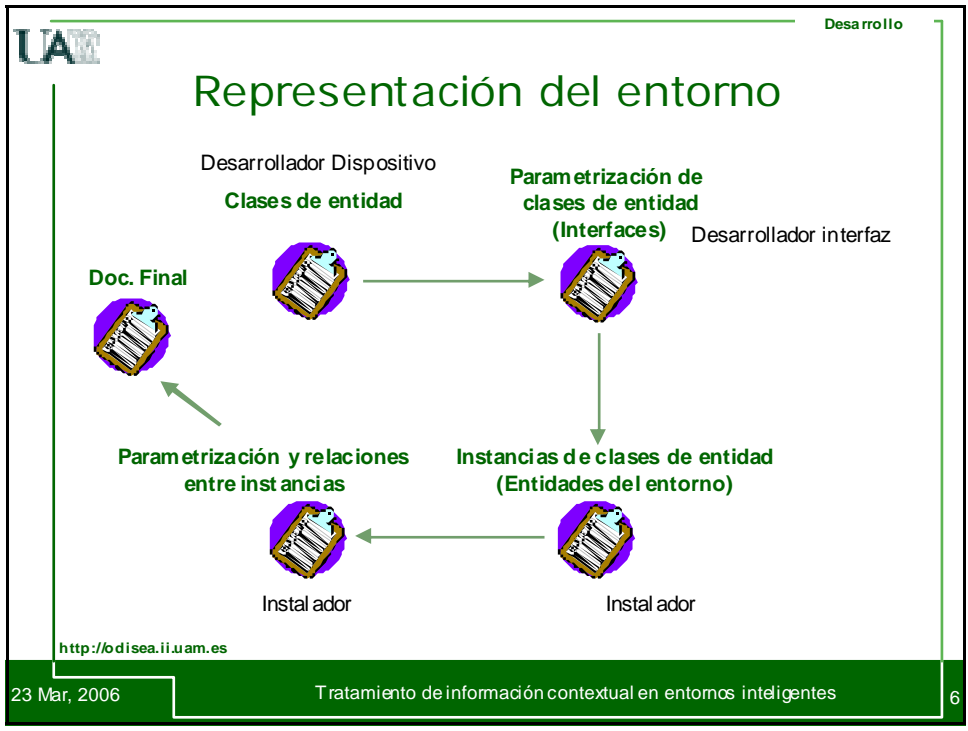
Jeffrey



```

<entity name="Lamp_1" id="1" type="0">
  <property name="Status">
    <paramSet name="jeffrey" id="2">
      <param name="type">switch</param>
      <param name="text_off">Turn on</param>
      <param name="cmd_off">1</param>
      <param name="text_on">Turn off</param>
      <param name="cmd_on">0</param>
      <param name="color_on">0x00FF00</param>
    </paramSet>
  </property>
  <paramSet name="jeffrey" id="2">
    <param name="image">lamp.gif</param>
    <param name="x">460</param>
    <param name="y">247</param>
  </paramSet>
</entity>
    
```

<http://odisea.ii.uam.es>



Universidad Autónoma de Madrid

Demostradores

23 de Marzo de 2006, EPS-UAM

Prototipo

Prototipo

Información multimedia

- TV
- Dos altavoces hifi
- Varios altavoces
- Radio
- DVD
- Dos Webcam IP
- Pantallas planas
- PC



- Cerradura electrónica
- RFID
- Luces
- Interruptores
- Teclado alfanumérico
- Cafetera

Domótica

Interacción vocal

- Dos micrófonos wireless
- Un micrófono de cabeza
- Sintetizadores

<http://odisea.ii.uam.es>

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

Capas

Arquitectura de Interact

Application Layer	Odisea Jeoffrey ContextAudio ContextImages Surveillance Meeting
Context Layer	Blackboard
Devices Management	SNMP
Communication Network	
Home Networking EIB X10 LonWork CeBus Zigbee	MM & Data Network Ethernet FireWire USB 2.0 IEEE 802.15 Bluetooth
Devices	Laptops TabletPCs Speakers VideoCameras Appliances PDAs PCs Sensors Microphones DVDs TVs


<http://odisea.ii.uam.es>

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

UA Proramación

Interacción con la capa de contexto

A) Navegador



The screenshot shows a Mozilla Firefox browser window with the address bar containing the URL: `http://odisea.ii.uam.es:8080/interact/bb/set/device/lamp_1/props/status?value=1`. Below the browser window, the XML response is displayed: `<SetResponse><property name="status" id="1">1</property></SetResponse>`. To the right of the XML is a control interface titled "Lamp_1 control" with two radio buttons: "Turn Off" (selected) and "Turn On". Below the radio buttons is a "Send command" button.

B) Formulario

C) Script

D) Librería Java

<http://odisea.ii.uam.es>

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

UA Agentes contextuales

Agentes contextuales

Sensores	Actuadores	Productores	Consumidores	Intérpretes
<ul style="list-style-type: none"> ▪ Sensor Puerta ▪ Lector RFID ▪ Sensor Teclado/Ratón ▪ Sensor On/Off 	<ul style="list-style-type: none"> ▪ Cerradura ▪ Altavoz ▪ Pantalla 	<ul style="list-style-type: none"> ▪ Productor Imagen ▪ Productor Audio 	<ul style="list-style-type: none"> ▪ Ag. AperturaPuerta ▪ Ag. Seguridad ▪ Consumidor Imágenes ▪ Identificador Reuniones 	<ul style="list-style-type: none"> ▪ Detector AlertaPuerta ▪ Detector Intrusos ▪ Detector Ocupantes ▪ Detector Localización ▪ Ag. Gestor Imágenes ▪ Ag. Gestor Audio

<http://odisea.ii.uam.es>

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

Aplicaciones sensibles al contexto

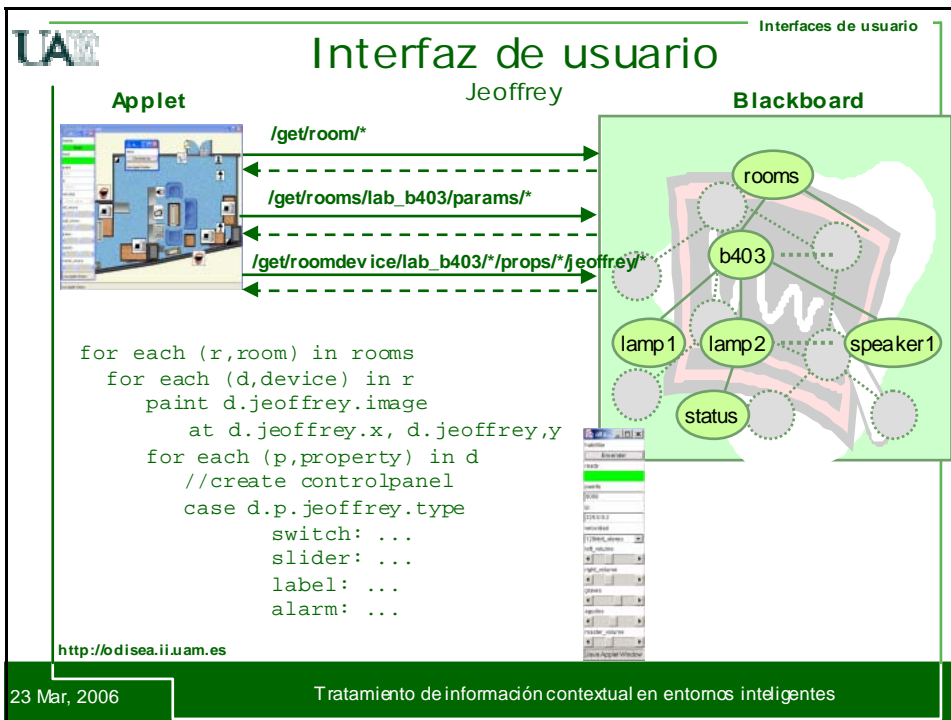
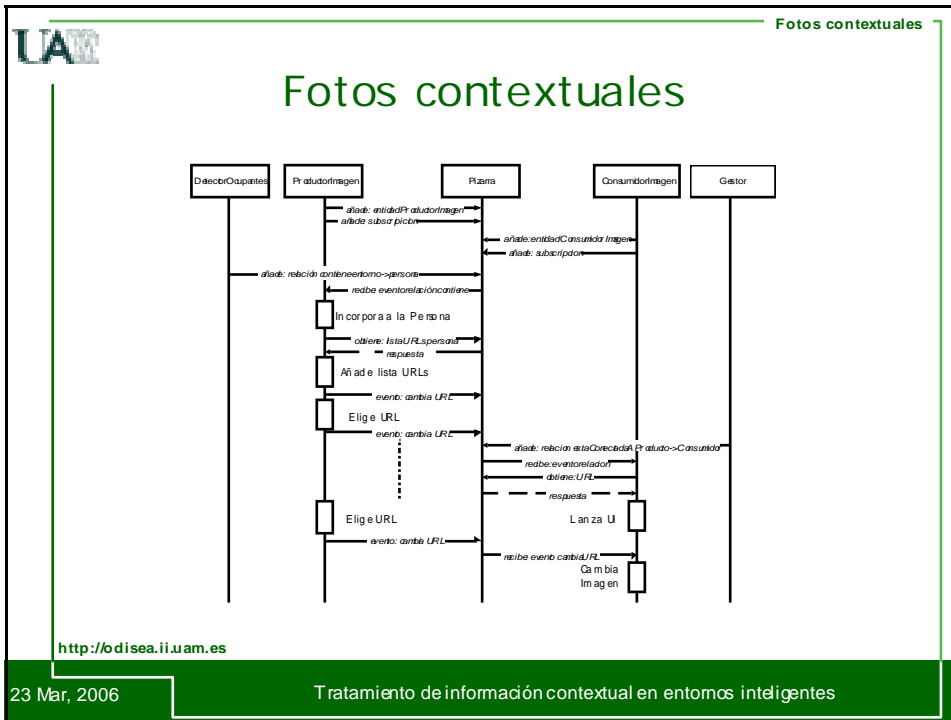
- Aplicaciones de control de acceso.
 - Vigilante puerta bien cerrada
 - Alarma contra intrusos
 - Control acceso al entorno
- Aplicaciones sensibles a la identidad del usuario.
 - Álbum de fotos contextuales
 - Aviso de reuniones
- Aplicaciones sensibles a la localización del usuario.
 - Gestor de flujos de audio
- Aplicaciones sensibles a la actividad.
 - Mensajes contextuales
- Interfaces de usuario
 - *Jeffrey*
 - *Odisea*

<http://odisea.ii.uam.es>

Agentes vs Aplicaciones

Aplicación	Sensor	Actuador	Productor	Consumidor	Intérprete
Vigilante puerta	Puerta			Ag. Seguridad	Detector AlertaPuerta
Alarma contra intrusos	Puerta			Ag. Seguridad	Detector Intrusos
Control acceso	LectorRFID	Cerradura		Ag. Apertura Puerta	
Album fotos		Pantallas	Productor Imagen	Consumidor Imagen	Gestor Imágenes Detector Localización
Aviso reunión				Identificador Reunión	
Audio Ctx		Altavoz	Productor Audio		Gestor audio, Detector Ocupantes, Detector Loc

<http://odisea.ii.uam.es>



Conclusiones y Trabajos futuros

23 de Marzo de 2006, EPS-UAM



Conclusiones

- El desarrollo de aplicaciones sensibles al contexto en Entornos Inteligentes está sujeto a diversos problemas.
 - Heterogeneidad: Tecnologías e Información Contextual.
 - Configuración dinámica.
- Se propone una capa de contexto para manejar esta complejidad:
 - Un modelo unificado del mundo compartido por todas los agentes.
 - Un repositorio de información que almacena el modelo.
 - Un mecanismo asíncrono de acceso al repositorio.

<http://odisea.ii.uam.es>

UAM

Conclusiones

Conclusiones

- Se añaden diversos módulos:
 - Controladores.
 - Histórico
 - Gestión de permisos.
 - Resolución de conflictos.

<http://odisea.ii.uam.es>

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

UAM

Conclusiones

Conclusiones

- Modelo de programación orientado a datos.
- Desacoplo temporal, espacial y funcional.
- La capa de contexto ha sido probada en un entorno real formado por:
 - Entorno.
 - Dispositivos.
 - Agentes contextuales.
 - Aplicaciones sensibles al contexto.

<http://odisea.ii.uam.es>

23 Mar, 2006 Tratamiento de información contextual en entornos inteligentes

Trabajo Futuro

- Tratamiento de la incertidumbre.
- Distribución de la implementación.
- RDF/OWL.
- Actividad.
- Privacidad.
- Aprendizaje.
- Preferencias de usuarios.

<http://odisea.ii.uam.es>

23 Mar, 2006

Tratamiento de información contextual en entornos inteligentes

FIN

<http://odisea.ii.uam.es>

23 Mar, 2006

Tratamiento de información contextual en entornos inteligentes