



Servicio de “Tutorías Virtuales” en la Universidad de Valencia

Congreso Internacional de Informática Educativa
1999

-UNED-

Pedro Morillo Tena (UV-I.Robótica)

Marcos Fernández (UV-I.Robótica)

Vicente Cerverón (UV)

Ricardo Ferris (UV)

Índice Presentación :

- Motivación y objetivos
- Consideraciones generales
- Requisitos Usuario. Especificaciones
- Análisis Sistema
- Diseño Sistema. Cliente/Servidor
- Implementaciones. Cgi, Applet, RMI, ...
- Resultados
- Conclusiones y Trabajo Futuro

Motivaciones



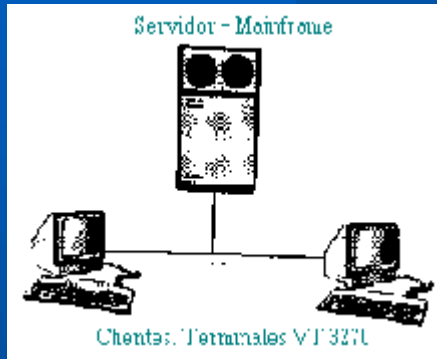
- **Universidad Virtual**
 - **Imposibilidad de accesos a campus**
 - **Geográficos**
 - **Laborales**
 - **Ambientales**
 - **Tutorías Virtuales + Gestor Videoconferencia**
- **Problemática de la U.V.:**
 - **Campus Distribuidos**
 - **N.P.E.**
- **Java como lenguaje en Internet**

Objetivos principales

- **Puesta en marcha de un SGTV en UV**
- Ilustración, V/I, de los 3 mecanismos principales de implementación en Java.
- Preparación y configuración servidor web con especificaciones mínimas del protocolo “http”.
- Estudio de la división de los E.P. de aplicaciones Java en subsistemas distribuidos en red.
- Verificar el “performance” de JDBC.

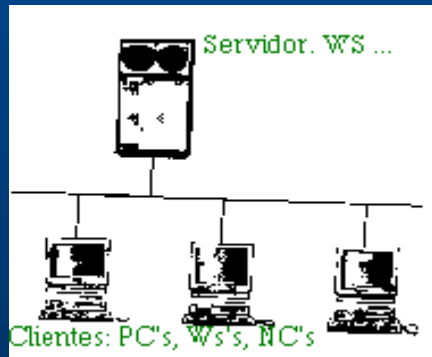
Conceptos y Consideraciones

Cliente-Servidor, Arquitectura de sistema



1-tier

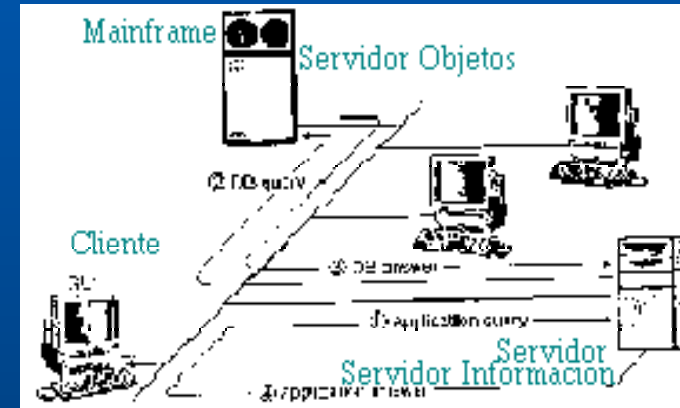
- Más antiguo ('80)
- Toda carga en servidor
- Terminales "tontos"
- Mala escalabilidad



2-tier

- Nacimiento real C/S
- >Repartición de carga
- Bloqueos.
- No SSDD

Arquitectura a Banda (tiers): Clientes+Servidores(Obj. +Inf.)



3-tier

- Paradigma real de SSDD
- Aparece Servidor Objetos
- Descarga a Cliente/Servidor y Red
- Liberación tipo almacén físico
- Mejor logística, seguridad (por bajos bloqueos) y eficiencia

Requisitos Usuario. Especificación Sistema

Requisitos de usuario: Procedimiento encuestas

.Aplicación funcionando sobre Internet

- **Visualización con navegadores típicos actuales**
- **Sin necesidad de lanzamientos de plug-ins adicionales**
- **No lanzamiento de cookies**
- **Al menos en computadores con conexión RTB**

Funcionalidad ejemplo:

Asignatura: Gestión de Bases de Datos

Fecha: 23/4/98

Usuario: Silvia Quesada Sesmero

Identificador pregunta: 21

Pregunta: ¿Es Postgress una base de datos orientada a objeto?

Respuesta:

Realmente no, pese a que incorpore ciertos mecanismos que le ofrezcan una apariencia de orientación a objetos hemos de catalogarla como Base de Datos Relacional Extendida.

Comentario 1 - Jose Serrano -

Pero, nosotros en prácticas hemos trabajado heredando tablas y hemos hechos referencias a OID's. ¿Cómo es eso?

Comentario 2 - J.J Relañó -

Es por eso que sigue siendo relacional, aun continua con la idea de tabla y no admite notación "punto" como Poet o Versant.

Especificaciones del sistema

- Aplicación C/S sobre TCP/IP siguiendo protocolo http
- Cualquier tipo de cliente (PC laboratorio, casa, Ws,..)
- Sin restricciones a priori para el servidor
- 2 tipos de usuarios
- *Interfaz de entrada común para todo tipo de usuario*



- Consulta al SGTV
- Búsqueda temas publicados
 - Búsqueda exhaustiva
 - Búsqueda por ítems
- Realización de comentarios
- Visualización profesores guardia

- Consulta al SGTV
- Respuesta a preguntas por e-mail
- Borrado respuestas descatalogada
- Búsqueda temas publicados
 - Búsqueda exhaustiva
 - Búsqueda por ítems
- Realización de comentarios

Materiales y Métodos

Sistema y servidores

- *Dependencia de la disponibilidad*
 - *P200 y 32 MB RAM con Linux Red Hat 5.0 y servidor Apache*
 - *PMMX 250 y 128 MB RAM con WNT 4.0 Server y servidor IIS*
 - *Silicon Graphics O2 con 64 Mb y servidor Netscape Fastrack 2.13*

• Elección motivos:

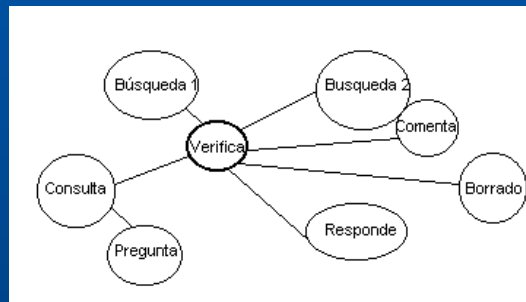
- *Novedad académica*
- *Coste*
- *Licencias de uso*
- *Portabilidad hacia el proyecto piloto*

Gestor Base de Datos

- Elección MiniSQL 2.0 de Hughes Technologies
 - **Coste 0 en desarrollo**
 - **Empleo de ANSI-SQL**
 - **Seguridad Look-up host**
 - Soporte de hasta 214 usuarios
 - Alta portabilidad por fuentes
 - Drivers ODBC y JDBC conexión

Análisis OMT. Rumbaugh:

- Modelo de objetos
- Modelo dinámico
- Modelo funcional



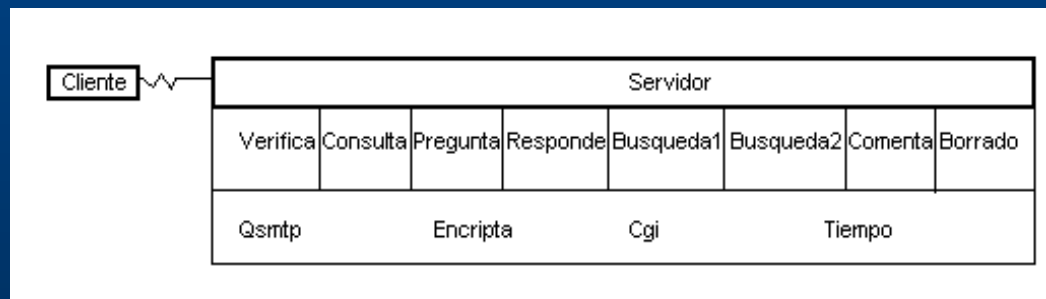
• *Distribución en estrella*

- Presencia de core
- Existen subsistema inherentemente concurrentes
- **Pese a ello, no consideración de hilos**

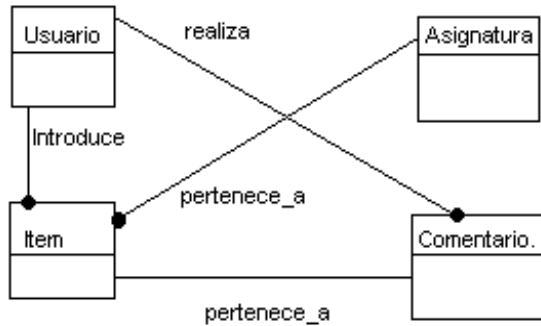
• *Organización de subsistemas en secuencias*



Disposición híbrida
partición vertical y
horizontal



Modelo entidad-relación:



Base de datos relacional

Tabla ASIGNATURA

Nombre del Campo	Tipo	Tamaño	Descripción
Asig_id	Entero	0..999	Identificador de la asignatura
Padre_id	Entero	0..99	ID del área de conocimiento
Nombre	Cadena caracteres	40	Nombre de la asignatura
Experto	Cadena caracteres	40	Nombre del experto en la materia
Correo	Cadena caracteres	30	Dirección de correo para atender cuestion

Tabla ITEM

Nombre del Campo	Tipo	Tamaño	Descripción
Item_id	Entero	-	Identificador de la pregunta/respuesta
Asig_id	Entero	0.999	ID de la asignatura a que pertenecen
Estudiante	Cadena caracteres	20	Nombre del user que hizo pregunta
Pregunta	Cadena caracteres	300	Texto del enunciado
Respuesta	Cadena caracteres	600	Texto de la respuesta (si la hubiera)
Fecha	Date	Fecha	Fecha de realización de consulta

Tabla COMENTARIO

Nombre del Campo	Tipo	Tamaño	Descripción
Coment_id	Entero	-	Identificador del comentario
Estudiante	Cadena caracteres	20	Nombre del estudiante que hizo comenta
Texto	Cadena caracteres	400	Texto del comentario
Referencia	Entero	-	P/R a la que hacen referencia

Tabla USUARIO

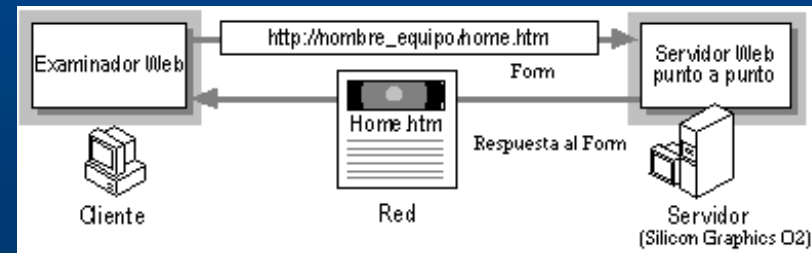
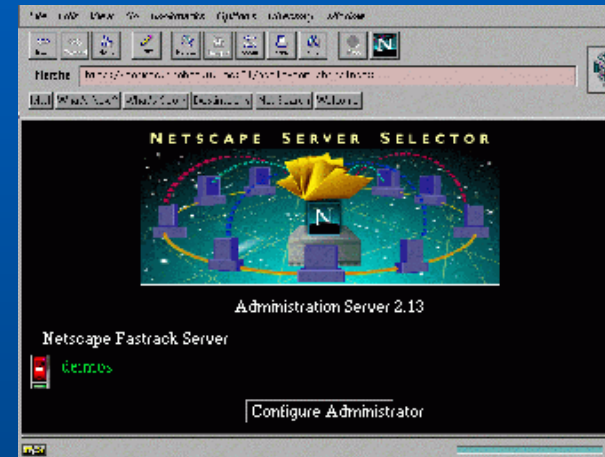
Nombre del Campo	Tipo	Tamaño	Descripción
Nombre	Cadena caracteres	20	Nombre en el sistema del usuario
Completo	Cadena caracteres	20	Nombre+Apellidos
Correo	Cadena caracteres	40	Dirección personal de correo
Cifrado	Cadena caracteres	20	Password de acceso
Tipo	Cadena caracteres	15	Tipo de usuario (experto - estudiante)

Implementación 1. Cgi's

1/2

Requisitos desarrollo:

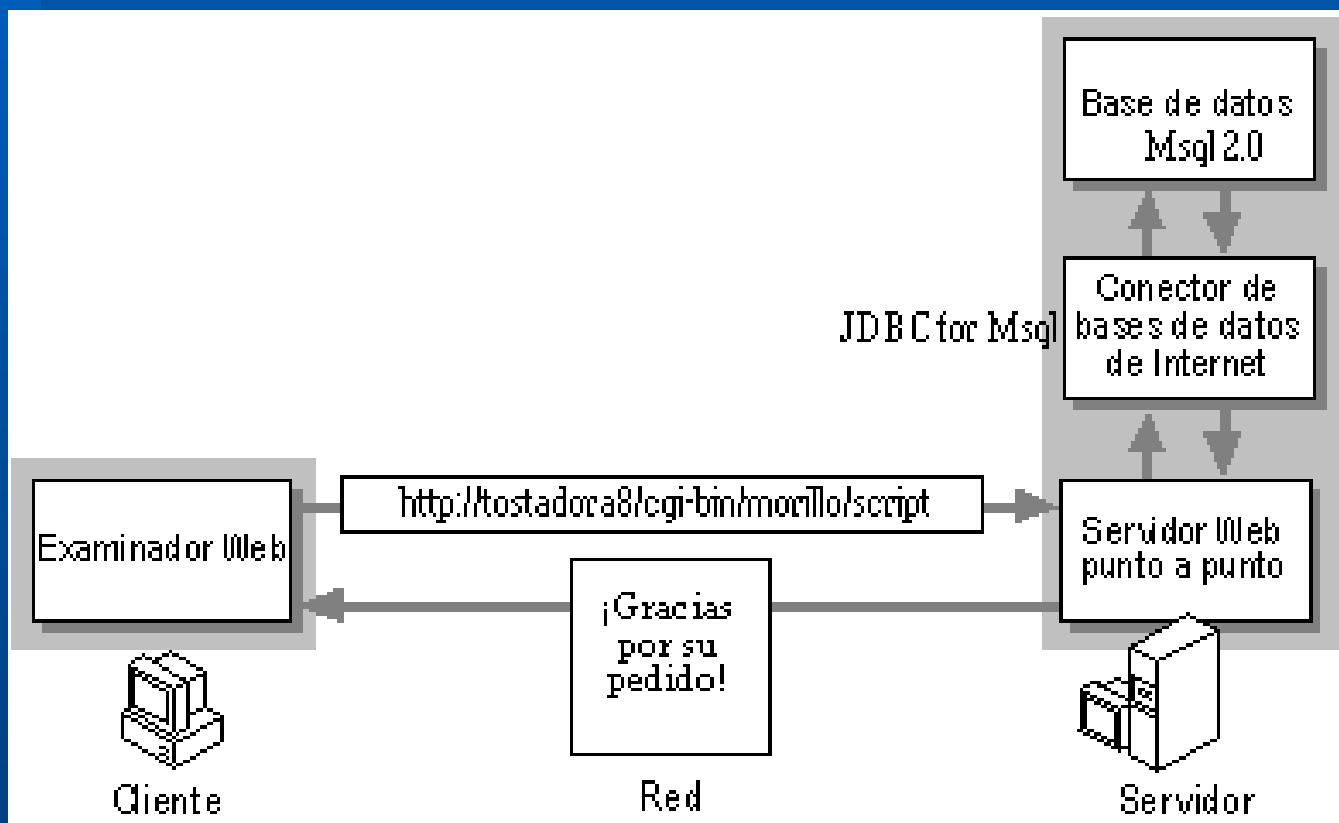
- Configuración servidor web
 - Tipología y scripts
 - Directorio de ejecución
 - Directorio de acceso restringido
 - REMOTE_USER
- Envío de mail siguiendo protocolo SMTP
- Forms de html. POST
 - Variables
 - Codificación URL
 - Procesamiento
- Modo trabajo base de datos



Implementación 1. Cgi's

2/2

Esquema de funcionamiento:



Implementación 2. Applet

1/2

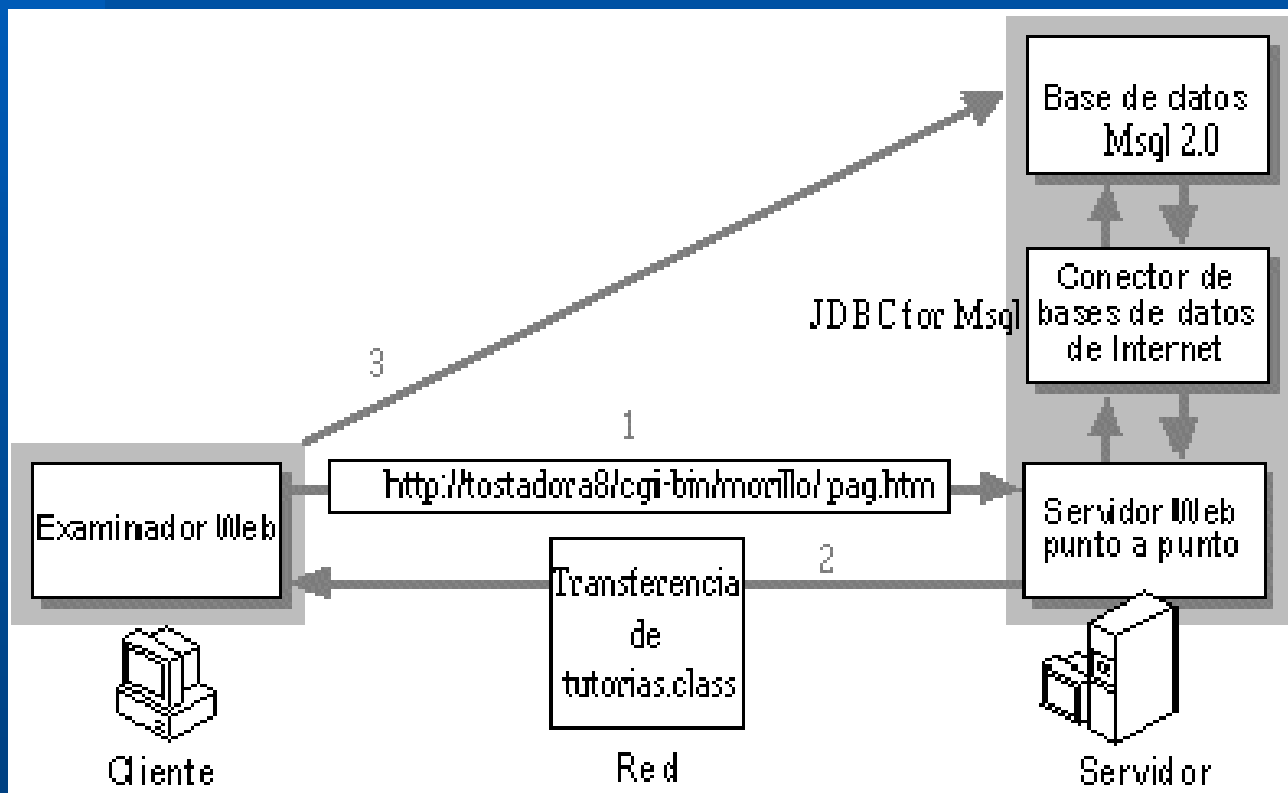
Requisitos desarrollo:

- Balance de cargas
 - Navegador cliente.
<APPLET CODE= nombre_de_clases.class></APPLET>
 - Funcionalidad del servidor web minimalista
 - Servidor en escucha por puerto 1114
- Metodología de aplicación basada en paneles y cards
- Empleo de AWT 1.1
- Modo de trabajo de Base de Datos

Implementación 2. Applet

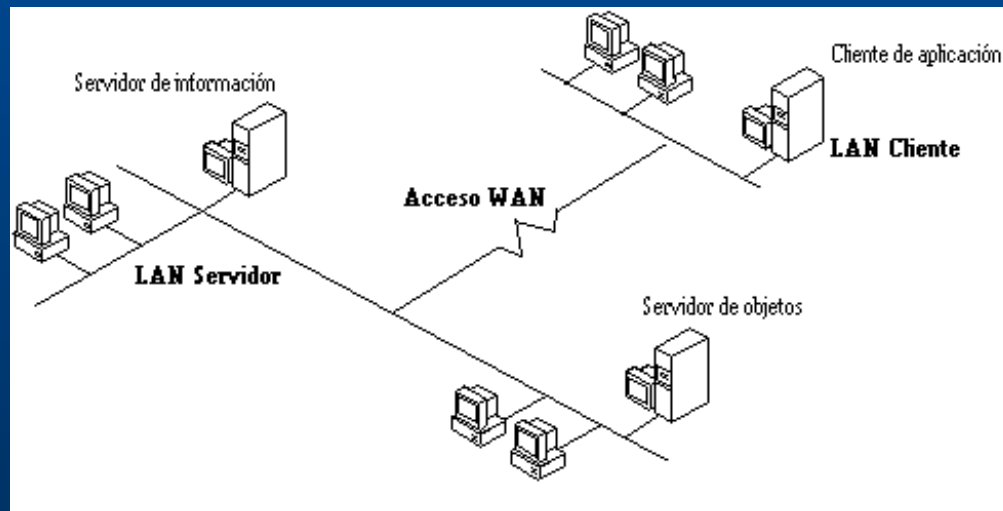
2/2

Esquema de funcionamiento



Implementación 3 - 4. RMI

- Elección del subsistema de seguridad. Jcrypt()
- Sistemas distribuidos
 - Componentes
 - RMI y CORBA
- Proceso y metodología



Resultados

Resultados temporales:

1/3

Operación en Tutorial Virtual	Implementación 1: "Http/cgi"	Implementación 2: "Applet"
<i>Acceso Tutorial</i>	693 ± 207.79	565.78 ± 119.8
<i>Mostrar Asignaturas</i>	907.17 ± 138.85	233 ± 66.7
<i>Inserción de Preguntas</i>	2107.23 ± 149.37	777.45 ± 68.34
<i>Respuesta a Preguntas</i>	1047 ± 75.38	552 ± 67.41
<i>Búsqueda Exhaustiva (75 ítems)</i>	4927.78 ± 564.19	3001.32 ± 221
<i>Búsqueda por Ítems</i>	1255.88 ± 356.1	799 ± 91.2
<i>Profesores de Guardia</i>	1141.84 ± 420	170.30 ± 83.62
<i>Borrado de Preguntas</i>	929.6 ± 125.79	139.2 ± 60.47
<i>Comentario a Preguntas</i>	1201.97 ± 357	522.58 ± 55.12

Tiempos de operaciones (ms) "cgi" vs "applet"

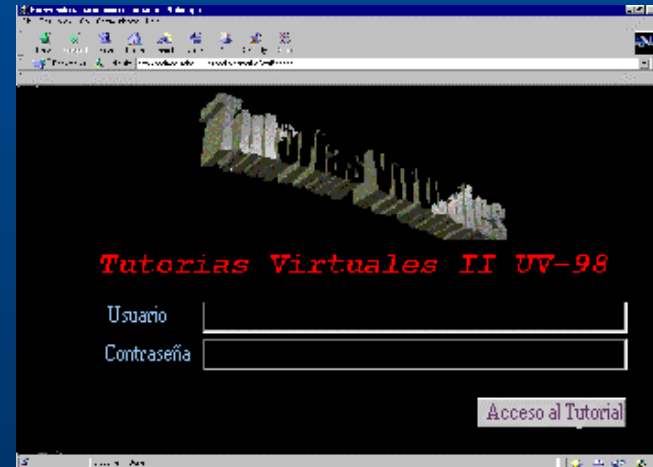
Implementación 3: "Centralizada"	Implementación 3-bis: "Distribuida"
Cliente:tostadora8.uv.es Servidor:tostadora8.uv.es Tiempo: 630.857 ± 115.9 ms Cliente:patan.uv.es Servidor:tostadora8.uv.es Tiempo:1701 ± 121.2 ms Cliente:fluid.etsiig.uniovi.es Servidor:tostadora8.uv.es Tiempo:3291.1 ± 288.22 ms Cliente: cvu.strath.ac.uk Servidor:tostadora8.uv.es Tiempo:5912.51 ± 33 ms	Cliente: tostadora8.uv.es Servidor Información: tostadora8.uv.es Servidor Objetos: tostadora8.uv.es Tiempo: 1026.75 ± 211.9 ms Cliente:patan.uv.es Servidor Información:tostadora8.uv.es Servidor Objetos:tostadora7.uv.es Tiempo:2191.88 ± 281.1 ms Cliente:fluid.etsiig.uniovi.es Servidor Información:tostadora8.uv.es Servidor Objetos:tostadora7.uv.es Tiempo:3698.32 ± 312 ms Cliente: cvu.strath.ac.uk Servidor Información: tostadora8.uv.es Servidor Objetos: tostadora7.uv.es Tiempo:6413.8 ± 452.32

Tiempos (ms) de accesos remotos

- Servicio de SGTV en la Universidad de Valencia.
- Dirigido a : Conjunto restringido de asignaturas de titulaciones:
Ingeniería Informática Ingeniería Técnica en Telecomunicaciones.
- Servidor: Silicon Graphics Origin 200.
- Oferta de implementaciones.



Vista implementacion 1("cgi")



Vista implementacion 2("applet")

Resultados

Consultas al servidor

3/3

Número de cuentas abiertas: 96

- Alumnos: 89
- Profesores: 7

Asignaturas: 9

<i>Asignatura</i>	<i>Accesos</i>	<i>P</i>	<i>R</i>	<i>C</i>
<i>Introducción a la programación</i>	356	25	24	51
<i>Lab. Introducción programación</i>	102	15	15	8
<i>Sistemas Distribuidos</i>	44	6	6	1
<i>Planificación y Gestión de Sistemas</i>	329	45	40	16
<i>Informática Gráfica</i>	78	11	10	6
<i>Lab. Informática Gráfica</i>	429	51	12	9
<i>Sistemas Operativos</i>	12	4	2	0
<i>Redes de Ordenadores</i>	135	10	10	34
<i>Gestión de Bases de Datos</i>	384	31	26	12

Tiempos (ms) de accesos remotos

P: nº preguntas

R: nº respuestas

C: nº comentarios

	<i>Implementación 1: "Http/cgi"</i>	<i>Implementación 2: "Applet"</i>
<i>Numero de accesos totales</i>	1321	646
<i>Porcentaje</i>	67.1%	32.9%

Tabla de acceso a implementaciones

Conclusiones finales

Puesta en marcha de un SGTV en la UV para 98-99

Implementación sistema atendiendo a 3 esquemas C/S

Configuración exitosa de un servidor web para ejecución cgi con java

Desarrollo aplicación usuario multiventana con AWT 1.1

Extrapolación resultados para V/I

Demostrar parcialización en subsistemas distribuidos

Dotar de seguridad UNIX a un applet

Mostrar idoneidad BdD relacional e interfaz JDBC a bajo coste (POSIX)

Trabajo futuro

- Ampliación del SGTV:
 - Fase 2: Plan de estudios Campus Ciencias.
 - Fase 3: Resto plan de estudios de la UV.
- Modificar el esquema de entorno distribuido para tareas críticas. C++ - Corba.
- Migración del esquema físico hacia BdD orientadas a objetos.
- Búsqueda, del posible punto inflexión en la ganancia del entorno distribuido.